

Google Cloud Certified

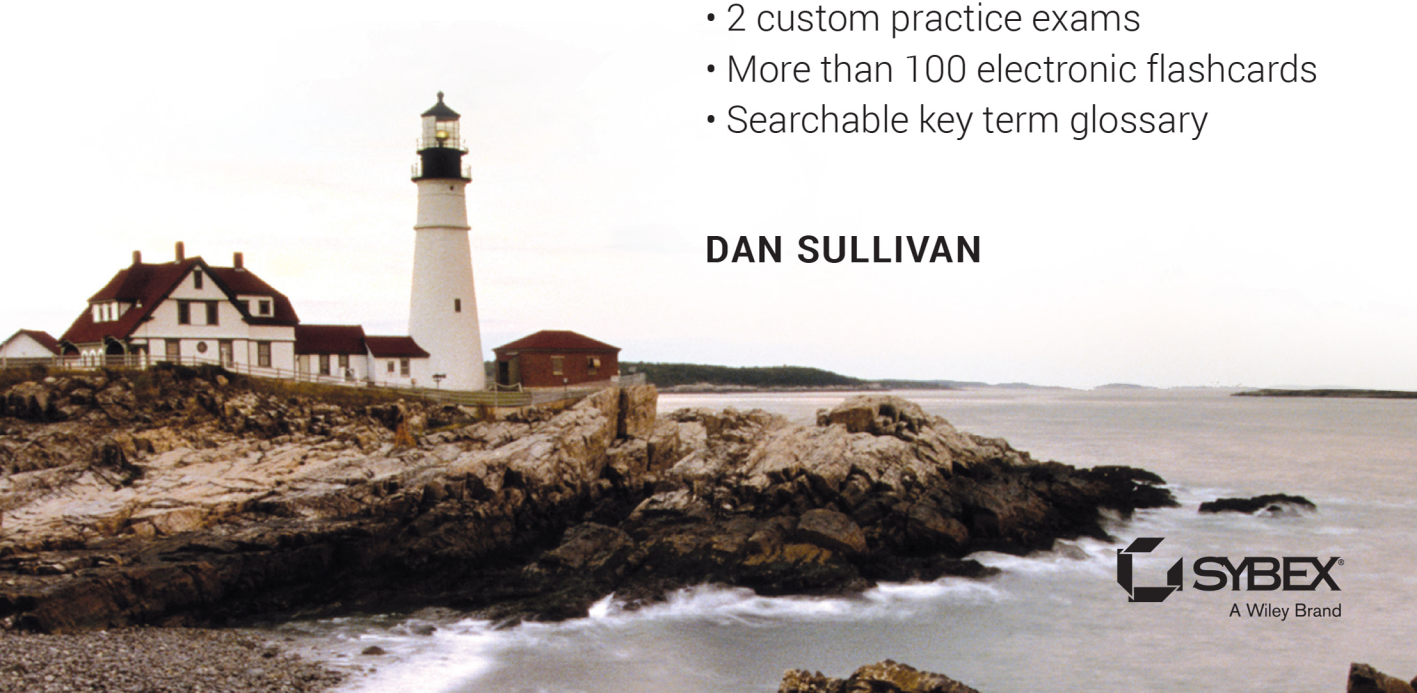
Associate Cloud Engineer Study Guide

Second Edition

Includes interactive online learning environment and study tools:

- 2 custom practice exams
- More than 100 electronic flashcards
- Searchable key term glossary

DAN SULLIVAN



Google Cloud

Certified Associate Cloud Engineer

Study Guide

Second Edition



Google Cloud Certified Associate Cloud Engineer

Study Guide
Second Edition



Dan Sullivan

 **SYBEX**
A Wiley Brand

Copyright © 2023 by Dan Sullivan. All rights reserved.

Published by John Wiley & Sons, Inc., Hoboken, New Jersey.

Published simultaneously in Canada and the United Kingdom.

ISBN: 978-1-119-87144-6

ISBN: 978-1-119-87145-3 (ebk.)

ISBN: 978-1-119-87146-0 (ebk.)

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning, or otherwise, except as permitted under Section 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 750-4470, or on the web at www.copyright.com. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at www.wiley.com/go/permission.

Trademarks: Wiley, the Wiley logo, and the Sybex logo are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates in the United States and other countries and may not be used without written permission. Google Cloud is a trademark of Google, LLC. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc. is not associated with any product or vendor mentioned in this book.

Limit of Liability/Disclaimer of Warranty: While the publisher and author have used their best efforts in preparing this book, they make no representations or warranties with respect to the accuracy or completeness of the contents of this book and specifically disclaim any implied warranties of merchantability or fitness for a particular purpose. No warranty may be created or extended by sales representatives or written sales materials. The advice and strategies contained herein may not be suitable for your situation. You should consult with a professional where appropriate. Further, readers should be aware that websites listed in this work may have changed or disappeared between when this work was written and when it is read. Neither the publisher nor authors shall be liable for any loss of profit or any other commercial damages, including but not limited to special, incidental, consequential, or other damages.

For general information on our other products and services or for technical support, please contact our Customer Care Department within the United States at (800) 762-2974, outside the United States at (317) 572-3993 or fax (317) 572-4002.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic formats. For more information about Wiley products, visit our web site at www.wiley.com.

Library of Congress Control Number: 2022945006

Cover image: © Jeremy Woodhouse/Getty Images

Cover design: Wiley

to Katherine

Acknowledgments

I am fortunate to have had the opportunity to work with the Wiley team once again. Jim Minatel, associate publisher at John Wiley & Sons; Pete Gaughan, managing editor; and Lily Miller, project manager, are a pleasure to work with and were essential to getting this second edition completed. I'd also like to thank Carole Jelen, VP of Waterside Productions, for all her help with yet another successful writing project.

Thank you to Kelly Kermode, Ammett Williams, and Robert Hales for their technical editing work. Google Cloud is complex and there are many opportunities for me to make mistakes or not explain something very well. Ammett's, Kelly's, and Robert's insight and knowledge have made this a better book.

About the Author

Dan Sullivan is a cloud and data architect specializing in Google Cloud, data architecture, data modeling, and machine learning. Dan is the author of *Google Cloud Certified Professional Architect Study Guide*, 2nd edition (Sybex, 2022); *Official Google Cloud Certified Professional Data Engineer Study Guide* (Sybex, 2020); *NoSQL for Mere Mortals* (Addison-Wesley Professional, 2015); and several LinkedIn Learning and Udemy courses on Google Cloud, databases, data science, and machine learning.

About the Technical Editors

Ammett Williams is a very simple and avid daydreamer who has more than 16 years of experience in the IT industry. Ammett has created the platform called Start Cloud Now with the aim to inspire others along their IT career path.

Ammett holds several IT certifications including CCIE #43569, CISSP, AWS, and a few Google Cloud professional level certs. In the daytime he is disguised as a Developer Relations Engineer @ Google. He can also be found online on LinkedIn www.linkedin.com/in/ammew and twitter @ammewtw.

Kelly Kermode is a self-professed renaissance gal with an insatiable penchant for learning. Kelly works as a cloud architect and engineer while residing in Grand Rapids, Michigan. With over 30 years of training and coaching experience, Kelly loves to think big and explore ways to think outside the box. From Bolivia to California to Michigan to South Africa, Kelly loves to share her love for collaborative problem-solving, architected solutions, data visualization, and geo-literacy. She leads consulting workshops, provides coaching, conducts technical bootcamps, and facilitates custom professional team development. Kelly is a Google Professional Cloud Architect, Google Cloud Certified Associate Cloud Engineer, member of the Google Earth Education Advisory Board, Google Innovator and Certified Trainer. In her free time, Kelly enjoys spending time with her two kids, reading, cooking, pottery, and gardening. Site: kellykermode.com and twitter: @coachk.

Robert Hales is an instructor from Western New York with over four years of training experience. Robert holds several Google, AWS, and Microsoft certifications and is a certified instructor in each domain. Robert is a US army veteran and loves to find ways to help other veterans get into the IT field. You can follow Robert on LinkedIn at www.linkedin.com/in/eventhorizonlearning.

Contents at a Glance

<i>Introduction</i>	<i>xxi</i>
<i>Assessment Test</i>	<i>xxxiii</i>
Chapter 1	Overview of Google Cloud 1
Chapter 2	Google Cloud Computing Services 17
Chapter 3	Projects, Service Accounts, and Billing 41
Chapter 4	Introduction to Computing in Google Cloud 67
Chapter 5	Computing with Compute Engine Virtual Machines 101
Chapter 6	Managing Virtual Machines 131
Chapter 7	Computing with Kubernetes 157
Chapter 8	Managing Standard Mode Kubernetes Clusters 179
Chapter 9	Computing with Cloud Run and App Engine 215
Chapter 10	Computing with Cloud Functions 237
Chapter 11	Planning Storage in the Cloud 253
Chapter 12	Deploying Storage in Google Cloud 285
Chapter 13	Loading Data into Storage 321
Chapter 14	Networking in the Cloud: Virtual Private Clouds and Virtual Private Networks 349
Chapter 15	Networking in the Cloud: DNS, Load Balancing, Google Private Access, and IP Addressing 375
Chapter 16	Deploying Applications with Cloud Marketplace and Cloud Foundation Toolkit 399
Chapter 17	Configuring Access and Security 425
Chapter 18	Monitoring, Logging, and Cost Estimating 447
Appendix	Answers to Review Questions 473
<i>Index</i>	<i>515</i>

Contents

Introduction

xxi

Assessment Test

xxxiii

Chapter 1	Overview of Google Cloud	1
	Types of Cloud Services	2
	Compute Resources	3
	Storage	4
	Networking	7
	Specialized Services	8
	Cloud Computing vs. Data Center Computing	8
	Rent Instead of Own Resources	8
	Pay-as-You-Go-for-What-You-Use Model	9
	Elastic Resource Allocation	9
	Specialized Services	10
	Summary	10
	Exam Essentials	10
	Review Questions	12
Chapter 2	Google Cloud Computing Services	17
	Computing Components of Google Cloud	18
	Computing Resources	19
	Storage Components of Google Cloud	23
	Storage Resources	23
	Databases	26
	Networking Components of Google Cloud	28
	Networking Services	28
	Identity Management and Security	30
	Development Tools	30
	Additional Components of Google Cloud	31
	Management and Observability Tools	31
	Specialized Services	32
	Summary	33
	Exam Essentials	33
	Review Questions	36
Chapter 3	Projects, Service Accounts, and Billing	41
	How Google Cloud Organizes Projects and Accounts	42
	Google Cloud Resource Hierarchy	42
	Organization Policies	45

	Managing Projects	46
	Roles and Identities	49
	Roles in Google Cloud	50
	Granting Roles to Identities	50
	Service Accounts	52
	Billing	53
	Billing Accounts	53
	Billing Budgets and Alerts	56
	Exporting Billing Data	57
	Enabling APIs	59
	Summary	60
	Exam Essentials	61
	Review Questions	62
Chapter 4	Introduction to Computing in Google Cloud	67
	Compute Engine	68
	Virtual Machine Images	68
	Virtual Machines Are Contained in Projects	77
	Virtual Machines Run in a Zone and Region	78
	Users Need Privileges to Create Virtual Machines	79
	Preemptible Virtual Machines	80
	Custom Machine Types	81
	Use Cases for Compute Engine Virtual Machines	82
	App Engine	83
	Structure of an App Engine Application	84
	App Engine Standard and Flexible Environments	85
	Use Cases for App Engine	86
	Kubernetes Engine	87
	Kubernetes Functionality	88
	Kubernetes Cluster Architecture	88
	Kubernetes Engine Use Cases	89
	Anthos	90
	Cloud Run	90
	Cloud Run Use Cases	91
	Cloud Functions	91
	Cloud Functions Execution Environment	91
	Cloud Functions Use Cases	93
	Summary	93
	Exam Essentials	95
	Review Questions	96
Chapter 5	Computing with Compute Engine Virtual Machines	101
	Creating and Configuring Virtual Machines with the Console	102
	Main Virtual Machine Configuration Details	104
	Advanced Configuration Details	109

	Creating and Configuring Virtual Machines with Cloud SDK	117
	Installing Cloud SDK	117
	Example Installation on Ubuntu Linux	118
	Creating a Virtual Machine with Cloud SDK	119
	Creating a Virtual Machine with Cloud Shell	120
	Basic Virtual Machine Management	121
	Starting and Stopping Instances	121
	Network Access to Virtual Machines	121
	Monitoring a Virtual Machine	123
	Cost of Virtual Machines	123
	Guidelines for Planning, Deploying, and Managing Virtual Machines	125
	Summary	125
	Exam Essentials	126
	Review Questions	127
Chapter 6	Managing Virtual Machines	131
	Managing Single Virtual Machine Instances	132
	Managing Single Virtual Machine Instances in the Console	132
	Managing a Single Virtual Machine Instance with Cloud Shell and the Command Line	141
	Introduction to Instance Groups	147
	Creating and Removing Instance Groups and Templates	147
	Instance Groups Load Balancing and Autoscaling	149
	Guidelines for Managing Virtual Machines	150
	Summary	150
	Exam Essentials	151
	Review Questions	152
Chapter 7	Computing with Kubernetes	157
	Introduction to Kubernetes Engine	158
	Kubernetes Cluster Architecture	159
	Kubernetes Objects	159
	Deploying Kubernetes Clusters	162
	Deploying Kubernetes Clusters Using Cloud Console	162
	Deploying Kubernetes Clusters Using Cloud Shell and Cloud SDK	167
	Deploying Application Pods	168
	Monitoring Kubernetes	172
	Summary	172
	Exam Essentials	173
	Review Questions	174

Chapter 8	Managing Standard Mode Kubernetes Clusters	179
	Viewing the Status of a Kubernetes Cluster	180
	Viewing the Status of Kubernetes Clusters Using Cloud Console	180
	Pinning Services to the Top of the Navigation Menu	182
	Viewing the Status of Kubernetes Clusters Using Cloud SDK and Cloud Shell	188
	Adding, Modifying, and Removing Nodes	193
	Adding, Modifying, and Removing Nodes with Cloud Console	193
	Adding, Modifying, and Removing Nodes with Cloud SDK and Cloud Shell	195
	Adding, Modifying, and Removing Pods	196
	Adding, Modifying, and Removing Pods with Cloud Console	196
	Adding, Modifying, and Removing Pods with Cloud SDK and Cloud Shell	200
	Adding, Modifying, and Removing Services	203
	Adding, Modifying, and Removing Services with Cloud Console	203
	Adding, Modifying, and Removing Services with Cloud SDK and Cloud Shell	205
	Creating Repositories in the Artifact Registry	207
	Viewing the Image Repository and Image Details with Cloud Console	207
	Summary	209
	Exam Essentials	209
	Review Questions	210
Chapter 9	Computing with Cloud Run and App Engine	215
	Overview of Cloud Run	216
	Cloud Run Services	216
	Cloud Run Jobs	217
	Creating a Cloud Run Service	218
	Creating a Cloud Run Job	222
	App Engine Components	223
	Deploying an App Engine Application	226
	Deploying an App Using Cloud Shell and SDK	226
	Scaling App Engine Applications	228
	Splitting Traffic Between App Engine Versions	229
	Summary	230
	Exam Essentials	231
	Review Questions	232

Chapter 10	Computing with Cloud Functions	237
	Introduction to Cloud Functions	238
	Events, Triggers, and Functions	238
	Runtime Environments	239
	Cloud Functions Receiving Events from Cloud Storage	241
	Deploying a Cloud Function for Cloud Storage	
	Events Using Cloud Console	241
	Deploying a Cloud Function for Cloud Storage	
	Events Using <i>gcloud</i> Commands	244
	Cloud Functions Receiving Events from Pub/Sub	245
	Deploying a Cloud Function for Cloud Pub/Sub	
	Events Using Cloud Console	245
	Deploying a Cloud Function for Cloud Pub/Sub	
	Events Using <i>gcloud</i> Commands	246
	Summary	247
	Exam Essentials	247
	Review Questions	249
Chapter 11	Planning Storage in the Cloud	253
	Types of Storage Systems	254
	Cache	255
	Persistent Storage	257
	Object Storage	258
	Storage Types When Planning a Storage Solution	264
	Storage Data Models	265
	Object: Cloud Storage	266
	Relational: Cloud SQL and Cloud Spanner	266
	Analytical: BigQuery	268
	NoSQL: Cloud Firestore and Bigtable	270
	Choosing a Storage Solution: Guidelines to Consider	277
	Summary	278
	Exam Essentials	278
	Review Questions	280
Chapter 12	Deploying Storage in Google Cloud	285
	Deploying and Managing Cloud SQL	286
	Creating and Connecting to a MySQL Instance	286
	Creating a Database, Loading Data, and Querying Data	288
	Backing Up MySQL in Cloud SQL	289
	Deploying and Managing Firestore	292
	Adding Data to a Firestore Database	292
	Backing Up Firestore	294

	Deploying and Managing BigQuery	294
	Estimating the Cost of Queries in BigQuery	294
	Viewing Jobs in BigQuery	296
	Deploying and Managing Cloud Spanner	297
	Deploying and Managing Cloud Pub/Sub	302
	Deploying and Managing Cloud Bigtable	306
	Deploying and Managing Cloud Dataproc	308
	Managing Cloud Storage	314
	Summary	316
	Exam Essentials	316
	Review Questions	317
Chapter 13	Loading Data into Storage	321
	Loading and Moving Data to Cloud Storage	322
	Loading and Moving Data to Cloud Storage Using the Console	322
	Loading and Moving Data to Cloud Storage Using the Command Line	327
	Importing and Exporting Data	328
	Importing and Exporting Data: Cloud SQL	328
	Importing and Exporting Data: Cloud Firestore	332
	Importing and Exporting Data: BigQuery	332
	Importing and Exporting Data: Cloud Spanner	337
	Exporting Data from Cloud Bigtable	339
	Importing and Exporting Data: Cloud Dataproc	340
	Streaming Data to Cloud Pub/Sub	341
	Summary	342
	Exam Essentials	342
	Review Questions	344
Chapter 14	Networking in the Cloud: Virtual Private Clouds and Virtual Private Networks	349
	Creating a Virtual Private Cloud with Subnets	350
	Creating a Virtual Private Cloud with Cloud Console	350
	Creating a Virtual Private Cloud with <i>gcloud</i>	354
	Creating a Shared Virtual Private Cloud Using <i>gcloud</i>	355
	Deploying Compute Engine with a Custom Network	357
	Creating Firewall Rules for a Virtual Private Cloud	359
	Structure of Firewall Rules	360
	Creating Firewall Rules Using Cloud Console	361
	Creating Firewall Rules Using <i>gcloud</i>	364
	Creating a Virtual Private Network	364
	Creating a Virtual Private Network Using Cloud Console	364
	Creating a Virtual Private Network Using <i>gcloud</i>	368

	Summary	368
	Exam Essentials	369
	Review Questions	370
Chapter 15	Networking in the Cloud: DNS, Load Balancing, Google Private Access, and IP Addressing	375
	Configuring Cloud DNS	376
	Creating DNS Managed Zones Using Cloud Console	376
	Creating DNS Managed Zones Using <i>gcloud</i>	381
	Configuring Load Balancers	382
	Types of Load Balancers	382
	Configuring Load Balancers Using Cloud Console	383
	Configuring Load Balancers Using <i>gcloud</i>	386
	Google Private Access	389
	Managing IP Addresses	389
	Expanding CIDR Blocks	390
	Reserving IP Addresses	390
	Summary	391
	Exam Essentials	392
	Review Questions	394
Chapter 16	Deploying Applications with Cloud Marketplace and Cloud Foundation Toolkit	399
	Deploying a Solution Using Cloud Marketplace	400
	Browsing Cloud Marketplace and Viewing Solutions	400
	Deploying Cloud Marketplace Solutions	403
	Building Infrastructure Using the Cloud Foundation Toolkit	411
	Deployment Manager Configuration Files	411
	Deployment Manager Template Files	414
	Launching a Deployment Manager Template	414
	Cloud Foundation Toolkit	415
	Config Connector	418
	Summary	418
	Exam Essentials	418
	Review Questions	420
Chapter 17	Configuring Access and Security	425
	Managing Identity and Access Management	426
	Viewing Account IAM Assignments	426
	Assigning IAM Roles to Accounts and Groups	428
	Defining Custom IAM Roles	432
	Managing Service Accounts	436
	Managing Service Accounts with Scopes	436
	Assigning a Service Account to a VM Instance	438
	Viewing Audit Logs	440

	Summary	441
	Exam Essentials	441
	Review Questions	443
Chapter 18	Monitoring, Logging, and Cost Estimating	447
	Cloud Monitoring	448
	Creating Dashboards	449
	Using Metric Explorer	450
	Creating Alerts	454
	Cloud Logging	458
	Log Routers and Log Sinks	458
	Configuring Log Sinks	459
	Viewing and Filtering Logs	459
	Viewing Message Details	462
	Using Cloud Diagnostics	463
	Overview of Cloud Trace	463
	Viewing Google Cloud Status	464
	Using the Pricing Calculator	464
	Summary	467
	Exam Essentials	468
	Review Questions	469
Appendix	Answers to Review Questions	473
	Chapter 1: Overview of Google Cloud	474
	Chapter 2: Google Cloud Computing Services	476
	Chapter 3: Projects, Service Accounts, and Billing	478
	Chapter 4: Introduction to Computing in Google Cloud	480
	Chapter 5: Computing with Compute Engine Virtual Machines	482
	Chapter 6: Managing Virtual Machines	485
	Chapter 7: Computing with Kubernetes	487
	Chapter 8: Managing Standard Mode Kubernetes Clusters	489
	Chapter 9: Computing with Cloud Run and App Engine	491
	Chapter 10: Computing with Cloud Functions	494
	Chapter 11: Planning Storage in the Cloud	496
	Chapter 12: Deploying Storage in Google Cloud	498
	Chapter 13: Loading Data into Storage	500
	Chapter 14: Networking in the Cloud: Virtual Private Clouds and Virtual Private Networks	502
	Chapter 15: Networking in the Cloud: DNS, Load Balancing, Google Private Access, and IP Addressing	504
	Chapter 16: Deploying Applications with Cloud Marketplace and Cloud Foundation Toolkit	507
	Chapter 17: Configuring Access and Security	509
	Chapter 18: Monitoring, Logging, and Cost Estimating	511
	<i>Index</i>	515

Introduction

Google Cloud is a leading public cloud that provides its users with some of the same software, hardware, and networking infrastructure used to power Google services. Businesses, organizations, and individuals can launch servers in minutes, store petabytes of data, and implement global virtual clouds with Google Cloud. It includes an easy-to-use console interface, command-line tools, and application programming interfaces (APIs) for managing resources in the cloud. Users can work with general resources, such as virtual machines (VMs) and persistent disks, or opt for highly focused services for Internet of Things (IoT), machine learning, media, and other specialized domains.

Deploying and managing applications and services in Google Cloud requires a clear understanding of the way Google structures user accounts and manages identities and access controls; you also need to understand the advantages and disadvantages of using various services. Certified Associate Cloud Engineers have demonstrated the knowledge and skills needed to deploy and operate infrastructure, services, and networks in Google Cloud.

This study guide is designed to help you understand Google Cloud in depth so that you can meet the needs of those operating resources in Google Cloud. Yes, this book will, of course, help you pass the Associate Cloud Engineer certification exam, but this is not an exam cram guide. You will learn more than is required to pass the exam; you will understand how to meet the day-to-day challenges faced by cloud engineers, including choosing services, managing users, deploying and monitoring infrastructure, and helping map business requirements into cloud-based solutions.

Each chapter in this book covers a single topic and includes an “Exam Essentials” section that outlines key information you should know to pass the certification exam. There are also exercises to help you review and reinforce your understanding of the chapter’s topic. Sample questions are included at the end of each chapter so that you can get a sense of the types of questions you will see on the exam. The book also includes flashcards and practice exams that cover all topics you’ll learn about with this guide.

What Does This Book Cover?

This book describes products and services in Google Cloud. It does not include G Suite administration topics.

Chapter 1: Overview of Google Cloud Platform In the opening chapter, we look into the types of services provided by Google Cloud, which include compute, storage, and networking services as well as specialized services, such as machine learning products. This chapter also describes some of the key differences between cloud computing and data center or on-premises computing.

Chapter 2: Google Cloud Computing Services This chapter provides an overview of infrastructure services such as computing, storage, and networking. It introduces the

concept of identity management and related services. It also introduces DevOps topics and tools for deploying and monitoring applications and resources. Google Cloud includes a growing list of specialized services, such as machine learning and natural language processing services. Those are briefly discussed in this chapter. The chapter introduces Google Cloud's organizational structure, with a look at regions and zones.

Chapter 3: Projects, Service Accounts, and Billing One of the first things you will do when starting to work with Google Cloud is to set up your accounts. In this chapter, you will learn how resources in accounts are organized into organizations, folders, and projects. You will learn how to create and edit these structures. You will also see how to enable APIs for particular projects as well as manage user identities and their access controls. This chapter describes how to create billing accounts and link them to projects. You will also learn how to create budgets and define billing alerts to help you manage costs.

Chapter 4: Introduction to Computing in Google Cloud In this chapter, you will see the variety of options available for running applications and services in Google Cloud. Options include Compute Engine, which provides VMs running Linux or Windows operating systems. Cloud Run and App Engine are platform as a service (PaaS) options that allows developers to run their applications without having to concern themselves with managing VMs. If you will be running multiple applications and services, you may want to take advantage of containers, which are a lightweight alternative to VMs. You will learn about containers and how to manage them with Kubernetes Engine. This chapter also introduces Cloud Functions, which is for event-driven, short-running tasks such as triggering the processing of an image loaded into Cloud Storage.

Chapter 5: Computing with Compute Engine Virtual Machines In this chapter, you will learn how to configure VMs, including selecting CPU, memory, storage options, and operating system images. You will learn how to use Google Cloud Console and Cloud Shell to work with VMs. In addition, you will see how to install the command-line interface and SDK, which you will use to start and stop VMs. The chapter also describes how to enable network access to VMs.

Chapter 6: Managing Virtual Machines In the previous chapter, you learned how to create VMs, and in this chapter you will learn how to manage individual and groups of VMs. You will start by managing a single instance of a VM using the Google Cloud console and then perform the same operations using Cloud Shell and the command line. You will also learn how to view currently running VMs. Next, you'll learn about instance groups, which allow you to create sets of VMs that you can manage as a single unit. In the section on instance groups, you will learn the difference between managed and unmanaged instance groups. You will also learn about preemptible instances, which are low-cost VMs that may be shut down by Google. You will learn about the

cost–benefit trade-offs of preemptible instances. Finally, the chapter closes with guidelines for managing VMs.

Chapter 7: Computing with Kubernetes This chapter introduces Kubernetes Engine, Google’s managed Kubernetes service. Kubernetes is a container orchestration platform created and released as open source by Google. In this chapter, you will learn the basics of containers, container orchestration, and the Kubernetes architecture. The discussion will include an overview of Kubernetes objects such as pods, services, volumes, and namespaces, as well as Kubernetes controllers such as ReplicaSets, Deployments, and Jobs.

Next, the chapter turns to deploying a Kubernetes cluster using Google Cloud console, Cloud Shell, and SDK. You will also see how to deploy pods, which includes downloading an existing Docker image, building a Docker image, creating a pod, and then deploying an application to the Kubernetes cluster. Of course, you will need to know how to monitor a cluster of servers. This chapter provides a description of how to set up monitoring and logging with Cloud Operations, which is Google’s application, service, container, and infrastructure monitoring service.

Chapter 8: Managing Standard Mode Kubernetes Clusters In this chapter you will learn the basics of managing a Kubernetes cluster, including viewing the status of the cluster, viewing the contents of the image repository, viewing details about images in the repository, and adding, modifying, and removing nodes, pods, and services. As in the chapter on managing VMs, in this chapter you will learn how to perform management operations with the three management tools: Google Cloud console, Cloud Shell, and SDK. The chapter concludes with a discussion of guidelines and good practices for managing a Kubernetes cluster.

Chapter 9: Computing with Cloud Run and App Engine Cloud Run and App Engine are part of Google Cloud’s serverless offerings. This chapter introduces Cloud Run, a service for running containers in the cloud. You will learn about the difference between Cloud Run Services and Cloud Run Jobs. Cloud Run will likely replace App Engine as the preferred choice for running containers in a serverless service, but App Engine is still in use and will be covered in this book. You will learn about App Engine components such as applications, services, versions, and instances. The chapter also covers how to define configuration files and specify dependencies of an application. In this chapter, you will learn how to view App Engine resources using Google Cloud console, Cloud Shell, and SDK. The chapter also describes how to distribute workload by adjusting traffic with splitting parameters. You will also learn about autoscaling in App Engine.

Chapter 10: Computing with Cloud Functions Cloud Functions is for event-driven, serverless computations. This chapter introduces Cloud Functions and shows you how to use it to receive events, evoke services, and return results. Next, you’ll see use cases for Cloud Functions, such as integrating with third-party APIs and event-driven processing. You will learn about Google’s Pub/Sub service for publication- and

subscription-based processing and how to use Cloud Functions with Pub/Sub. Cloud Functions are well suited to respond to events in Cloud Storage. The chapter describes Cloud Storage events and how to use Cloud Functions to receive and respond to those events. You will learn how to use Cloud Operations to monitor and log details of Cloud Function executions. Finally, the chapter concludes with a discussion of guidelines for using and managing Cloud Functions.

Chapter 11: Planning Storage in the Cloud Having described various compute options in Google Cloud, it is time to turn your attention to storage. This chapter describes characteristics of storage systems, such as their time to access, persistence, and data model. In this chapter, you will learn about differences between caches, persistent storage, and archival storage. You will learn about the cost–benefit trade-offs of using regional and multiregional persistent storage and using nearline versus Coldline and archival storage. The chapter includes details on the various Google Cloud storage options, including Cloud Storage for blob storage; Cloud SQL and Spanner for relational data; Firestore and Bigtable, for NoSQL storage; BigQuery for analytic data; and Cloud Firebase for mobile application data. The chapter includes detailed guidance on choosing a data store based on requirements for consistency, availability, transaction support, cost, latency, and support for various read/write patterns.

Chapter 12: Deploying Storage in Google Cloud Platform In this chapter, you will learn how to create databases, add data, list records, and delete data from each of Google Cloud’s storage systems. The chapter starts by introducing Cloud SQL, a managed database service that offers SQL Server, MySQL, and PostgreSQL managed instances. You will also learn how to create databases in Cloud Firestore, BigQuery, Bigtable, and Spanner. Next, you will turn your attention to Cloud Pub/Sub for storing data in message queues, followed by a discussion of Cloud Dataproc, a managed Hadoop and Spark cluster service, for processing big data sets. In the next section, you will learn about Cloud Storage for objects. The chapter concludes with guidance on how to choose a data store for a particular set of requirements.

Chapter 13: Loading Data into Storage There are a variety of ways of getting data into Google Cloud. This chapter describes how to use the command-line SDK to load data into Cloud SQL, Cloud Storage, Firestore, BigQuery, Bigtable, and Dataproc. It also describes bulk importing and exporting from those same services. Next, you will learn about two common data loading patterns: moving data from Cloud Storage and streaming data to Cloud Pub/Sub.

Chapter 14: Networking in the Cloud: Virtual Private Clouds and Virtual Private Networks In this chapter, you’ll turn your attention to networking with an introduction to basic networking concepts, including the following:

- IP addresses
- CIDR blocks

- Networks and subnetworks
- Virtual private clouds (VPCs)
- Routing and rules
- Virtual private networks (VPNs)
- Cloud DNS
- Cloud Routers
- Cloud Interconnect
- External peering

After being introduced to key networking concepts, you will learn how to create a VPC. Specifically, this includes defining a VPC, specifying firewall rules, creating a VPN, and working with load balancers. You will learn about different types of load balancers and when to use them.

Chapter 15: Networking in the Cloud: DNS, Load Balancing, Google Private Access, and IP Addressing In this chapter, you will learn about common network management tasks such as defining subnetworks, adding subnets to a VPC, managing CIDR blocks, and reserving IP addresses. You will learn how to perform each of these tasks using Cloud Console, Cloud Shell, and Cloud SDK.

Chapter 16: Deploying Applications with Cloud Marketplace and Cloud Foundation Toolkit Google Cloud Marketplace is Google Cloud's marketplace of preconfigured stacks and services. This chapter introduces Cloud Marketplace and describes some applications and services currently available. You will learn how to browse Cloud Marketplace, deploy applications from Cloud Marketplace, and shut down Cloud Marketplace applications. The chapter also discusses Deployment Manager templates that automate the deployment of an application and launch a Deployment Manager template to provision Google Cloud resources and configure an application automatically.

Chapter 17: Configuring Access and Security This chapter introduces identity management. In particular, you will learn about identities, roles, and assigning and removing identity roles. This chapter also introduces service accounts and how to create them, assign them to VMs, and work with them across projects. You will also learn how to view audit logs for projects and services. The chapter concludes with guidelines for configuring access control security.

Chapter 18: Monitoring, Logging, and Cost Estimating In the final chapter, we will discuss Cloud Operations alerts, logging, distributed tracing, and application debugging. Each of the corresponding Google Cloud services is designed to enable more efficient, functional, and reliable services. The chapter concludes with a review of the Pricing Calculator, which is helpful for estimating the cost of resources in Google Cloud.

Interactive Online Learning Environment and Test Bank



Like all exams, the Associate Cloud Engineer certification from Google Cloud is updated periodically and may eventually be retired or replaced. At some point after Google Cloud is no longer offering this exam, the old editions of our books and online tools will be retired. If you have purchased this book after the exam was retired, or are attempting to register in the Sybex online learning environment after the exam was retired, please know that we make no guarantees that this exam's online Sybex tools will be available once the exam is no longer available.

Studying the material in the *Google Cloud Certified Associate Cloud Engineer Study Guide, Second Edition* is an important part of preparing for the Associate Cloud Engineer certification exam, but we provide additional tools to help you prepare. The online Test Bank will help you understand the types of questions that will appear on the certification exam.

The sample tests in the Test Bank include all the questions in each chapter as well as the questions from the assessment test. In addition, there are two practice exams with 50 questions each. You can use these tests to evaluate your understanding and to identify areas where you may require additional study.

The flashcards in the Test Bank will push the limits of what you should know for the certification exam. There are 100 questions provided in digital format. Each flashcard has one question and one correct answer.

The online glossary is a searchable list of key terms introduced in this exam guide that you should know for the Associate Cloud Engineer certification exam.

To start using these to study for the Google Certified Associate Cloud Engineer exam, go to www.wiley.com/go/sybextestprep and register your book to receive your unique PIN. Once you have the PIN, return to www.wiley.com/go/sybextestprep, find your book and click Register or Login, and follow the link to register a new account or add this book to an existing account.



Exam policies can change from time to time. We highly recommend that you check <https://cloud.google.com/certification> for the most up-to-date information when you begin your preparation, when you register, and again a few days before your scheduled exam date.

Exam Objectives

The Associate Cloud Engineer certification is designed for people who create, deploy, and manage enterprise applications and infrastructure in Google Cloud. An Associate Cloud Engineer is comfortable working with Cloud Console, Cloud Shell, and Cloud SDK. Such individuals also understand products offered as part of Google Cloud and their appropriate use cases.

The exam will test your knowledge of the following:

- Planning a cloud solution using one or more Google Cloud services
- Creating a cloud environment for an organization
- Deploying applications and infrastructure
- Using monitoring and logging to ensure availability of cloud solutions
- Setting up identity management, access controls, and other security measures

Objective Map

The following are specific objectives defined by Google at <https://cloud.google.com/certification/guides/cloud-engineer>.

Section 1: Setting up a cloud solution environment

1.1 Setting up cloud projects and accounts. Activities include:

- Creating a resource hierarchy
- Applying organizational policies to the resource hierarchy
- Granting members IAM roles within a project
- Managing users and groups in Cloud Identity (manually and automated)
- Enabling APIs within projects
- Provisioning and setting up products in Google Cloud's operations suite

1.2 Managing billing configuration. Activities include:

- Creating one or more billing accounts
- Linking projects to a billing account
- Establishing billing budgets and alerts
- Setting up billing exports

1.3 Installing and configuring the command-line interface (CLI), specifically Cloud SDK (e.g., setting the default project)

Section 2: Planning and configuring a cloud solution

2.1 Planning and estimating Google Cloud product use using the Pricing Calculator

2.2 Planning and configuring compute resources. Considerations include:

- Selecting appropriate compute choices for a given workload (e.g., Compute Engine, Google Kubernetes Engine, Cloud Run, Cloud Functions)
- Using preemptible VMs and custom machine types as appropriate

2.3 Planning and configuring data storage options. Considerations include:

- Product choice (e.g., Cloud SQL, BigQuery, Firestore, Cloud Spanner, Cloud Bigtable)
- Choosing storage options (e.g., Zonal persistent disk, Regional balanced persistent disk, Standard, Nearline, Coldline, Archive)

2.4 Planning and configuring network resources. Tasks include:

- Differentiating load balancing options
- Identifying resource locations in a network for availability
- Configuring Cloud DNS

Section 3: Deploying and implementing a cloud solution

3.1 Deploying and implementing Compute Engine resources. Tasks include:

- Launching a compute instance using Cloud Console and Cloud SDK (`gcloud`) (e.g., assign disks, availability policy, SSH keys)
- Creating an autoscaled managed instance group using an instance template
- Generating/uploading a custom SSH key for instances
- Installing and configuring the Cloud Monitoring and Logging Agent
- Assessing compute quotas and requesting increases

3.2 Deploying and implementing Kubernetes Engine resources. Tasks include:

- Installing and configuring the command line interface (CLI) for Kubernetes (`kubectl`)
- Deploying a Google Kubernetes Engine cluster with different configurations including AutoPilot, regional clusters, private clusters, etc.
- Deploying a containerized application to Google Kubernetes Engine
- Configuring Kubernetes Engine monitoring and logging

3.3 Deploying and implementing Cloud Run and Cloud Functions resources.**Tasks include, where applicable:**

- Deploying an application and updating scaling configuration, versions, and traffic splitting
- Deploying an application that receives Google Cloud events (e.g., Pub/Sub events, Cloud Storage object change notification events)

3.4 Deploying and implementing data solutions. Tasks include:

- Initializing data systems with products (e.g., Cloud SQL, Firestore, BigQuery, Cloud Spanner, Cloud Pub/Sub, Cloud Bigtable, Dataproc, Dataflow, Cloud Storage)
- Loading data (e.g., command line upload, API transfer, import/export, load data from Cloud Storage, streaming data to Pub/Sub)

3.5 Deploying and implementing networking resources. Tasks include:

- Creating a VPC with subnets (e.g., custom-mode VPC, shared VPC)
- Launching a Compute Engine instance with custom network configuration (e.g., internal-only IP address, Google private access, static external and private IP address, network tags)
- Creating ingress and egress firewall rules for a VPC (e.g., IP subnets, network tags, service accounts)
- Creating a VPN between a Google VPC and an external network using Cloud VPN
- Creating a load balancer to distribute application network traffic to an application (e.g., global HTTP(S) load balancer, Global SSL Proxy load balancer, Global TCP Proxy load balancer, regional network load balancer, regional internal load balancer)

3.6 Deploying a solution using Cloud Marketplace. Tasks include:

- Browsing the Cloud Marketplace catalog and viewing solution details
- Deploying a Cloud Marketplace solution

3.7 Implementing resources via infrastructure as code. Tasks include:

- Building infrastructure via Cloud Foundation Toolkit templates and implementing best practices
- Installing and configuring Config Connector in Google Kubernetes Engine to create, update, delete, and secure resources

Section 4: Ensuring successful operation of a cloud solution**4.1 Managing Compute Engine resources. Tasks include:**

- Managing a single VM instance (e.g., start, stop, edit configuration, or delete an instance)
- Remotely connecting to the instance

- Attaching a GPU to a new instance and installing necessary dependencies
- Viewing current running VM inventory (instance IDs, details)
- Working with snapshots (e.g., create a snapshot from a VM, view snapshots, delete a snapshot)
- Working with images (e.g., create an image from a VM or a snapshot, view images, delete an image)
- Working with instance groups (e.g., set autoscaling parameters, assign instance template, create an instance template, remove an instance group)
- Working with management interfaces (e.g., Google Cloud console, Cloud Shell, Cloud SDK)

4.2 Managing Kubernetes Engine resources. Tasks include:

- Viewing current running cluster inventory (nodes, pods, services)
- Browsing Docker images and viewing their details in Artifact Registry
- Working with nodes pools (e.g., add, edit, or remove a node pool)
- Working with pods (e.g., add, edit, or remove pods)
- Working with services (e.g., add, edit, or remove a service)
- Working with stateful applications (e.g., persistent volumes, stateful sets)
- Managing Horizontal and Vertical autoscaling configurations
- Working with management interfaces (e.g., Google Cloud console, Cloud Shell, Cloud SDK, kubectl)

4.3 Managing Cloud Run resources. Tasks include:

- Adjusting application traffic-splitting parameters
- Setting scaling parameters for autoscaling instances
- Determining whether to run Cloud Run (fully managed) or Cloud Run for Anthos

4.4 Managing storage and database solutions. Tasks include:

- Managing and securing objects in and between Cloud Storage buckets
- Setting object life cycle management policies for Cloud Storage buckets
- Executing queries to retrieve data from data instances (e.g., Cloud SQL, BigQuery, Cloud Spanner, Datastore, Cloud Bigtable)
- Estimating costs of data storage resources
- Backing up and restoring database instances (e.g., Cloud SQL, Datastore)
- Reviewing job status in Dataproc, Dataflow, or BigQuery

4.5 Managing networking resources. Tasks include:

- Adding a subnet to an existing VPC
- Expanding a subnet to have more IP addresses
- Reserving static external or internal IP addresses
- Working with CloudDNS, CloudNAT, Load Balancers and firewall rules

4.6 Monitoring and logging. Tasks include:

- Creating Cloud Monitoring alerts based on resource metrics
- Creating and ingesting Cloud Monitoring custom metrics (e.g., from applications or logs)
- Configuring log sinks to export logs to external systems (e.g., on-premises or BigQuery)
- Configuring log routers
- Viewing and filtering logs in Cloud Logging
- Viewing specific log message details in Cloud Logging
- Using cloud diagnostics to research an application issue (e.g., viewing Cloud Trace data, using Cloud Debug to view an application point-in-time)
- Viewing Google Cloud status

Section 5: Configuring access and security**5.1 Managing Identity and Access Management (IAM). Tasks include:**

- Viewing IAM policies
- Creating IAM policies
- Managing the various role types and defining custom IAM roles (e.g., primitive, pre-defined and custom)

5.2 Managing service accounts. Tasks include:

- Creating service accounts
- Using service accounts in IAM policies with minimum permissions
- Assigning service accounts to resources
- Managing IAM of a service account
- Managing service account impersonation
- Creating and managing short-lived service account credentials

5.3 Viewing audit logs

How to Contact the Publisher

If you believe you've found a mistake in this book, please bring it to our attention. At John Wiley & Sons, we understand how important it is to provide our customers with accurate content, but even with our best efforts an error may occur.

In order to submit your possible errata, please email it to our Customer Service Team at wileysupport@wiley.com with the subject line "Possible Book Errata Submission."

Assessment Test

1. Instance templates are used to create a group of identical VMs. The instance templates include:
 - A. Machine type, boot disk image or container image, zone, and labels
 - B. Cloud Storage bucket definitions
 - C. A load balancer description
 - D. App Engine configuration file
2. The command-line command to create a Cloud Storage bucket is:
 - A. `gcloud mb`
 - B. `gsutil mb`
 - C. `gcloud mkbucket`
 - D. `gsutil mkbucket`
3. Your company has an object management policy that requires that objects stored in Cloud Storage be migrated from standard storage to nearline storage 90 days after the object is created. The most efficient way to do this is to:
 - A. Create a Cloud Function to copy objects from regional storage to nearline storage.
 - B. Set the `MigrateObjectAfter` property on the stored object to 90 days.
 - C. Copy the object to persistent storage attached to a VM and then copy the object to a bucket created on nearline storage.
 - D. Create a life cycle management configuration policy specifying an age of 90 days and `SetStorageClass` as nearline.
4. An education client maintains a site where users can upload videos, and your client needs to assure redundancy for the files; therefore, you have created two buckets for Cloud Storage. Which command do you use to synchronize the contents of the two buckets?
 - A. `gsutil rsync`
 - B. `gcloud cp sync`
 - C. `gcloud rsync`
 - D. `gsutil cp sync`
5. VPC resources are which of the following?
 - A. Regional
 - B. Zonal
 - C. Global
 - D. Subnet

6. A remote component in your network has failed, which results in a transient network error. When you submit a `gsutil` command, it fails because of a transient error. By default, the command will:
 - A. Terminate and log a message to Cloud Monitoring
 - B. Retry using a truncated binary exponential backoff strategy
 - C. Prompt the user to decide to retry or quit
 - D. Terminate and log a message to Cloud Shell
7. All of the following are components of firewall rules except which one?
 - A. Direction of traffic
 - B. Action on match
 - C. Time to live (TTL)
 - D. Protocol
8. Adding virtual machines to an instance group can be triggered in an autoscaling policy by all of the following, except which one?
 - A. CPU utilization
 - B. Cloud Monitoring metrics
 - C. IAM policy violation
 - D. Load balancing serving capacity
9. Your company's finance department is developing a new account management application that requires transactions and the ability to perform relational database operations using fully compliant SQL. Data store options in Google Cloud include:
 - A. Spanner and Cloud SQL
 - B. Firestore and Bigtable
 - C. Spanner and Cloud Storage
 - D. Firestore and Cloud SQL
10. The marketing department in your company wants to deploy a web application but does not want to have to manage servers or clusters. A good option for them is:
 - A. Compute Engine
 - B. Kubernetes Engine
 - C. Cloud Run
 - D. Cloud Functions
11. Your company is building an enterprise data warehouse and wants SQL query capabilities over petabytes of data, but does not want to manage servers or clusters. A good option for them is:
 - A. Cloud Storage
 - B. BigQuery
 - C. Bigtable
 - D. Firestore

12. You have been hired as a consultant to a startup in the Internet of Things (IoT) space. The startup will stream large volumes of data into Google Cloud. The data needs to be filtered, transformed, and analyzed before being stored in Google Cloud Firestore. A good option for the stream processing component is:
- A. Dataproc
 - B. Cloud Dataflow
 - C. Cloud Endpoints
 - D. Cloud Interconnect
13. Preemptible virtual machines may be shut down at any time but will always be shut down after running:
- A. 6 hours
 - B. 12 hours
 - C. 24 hours
 - D. 48 hours
14. You have been tasked with designing an organizational hierarchy for managing departments and their cloud resources. What organizing components are available in Google Cloud?
- A. Organization, folders, projects
 - B. Buckets, directories, subdirectories
 - C. Organizations, buckets, projects
 - D. Folders, buckets, projects
15. During an incident that has caused an application to fail, you suspect some resource may not have appropriate roles granted. The command to list roles granted to a resource is:
- A. `gutil iam list-grantable-roles`
 - B. `gcloud iam list-grantable-roles`
 - C. `gcloud list-grantable-roles`
 - D. `gcloud resources grantable-roles`
16. The availability of CPU platforms can vary between zones. To get a list of all CPU types available in a particular zone, you should use:
- A. `gcloud compute zones describe`
 - B. `gcloud iam zones describe`
 - C. `gutil zones describe`
 - D. `gcloud compute regions list`
17. To create a custom role, a user must possess which role?
- A. `iam.create`
 - B. `compute.roles.create`
 - C. `iam.roles.create`
 - D. `Compute.roles.add`

- 18.** You have been asked to create a network with 1,000 IP addresses. In the interest of minimizing unused IP addresses, which CIDR suffix would you use to create a network with at least 1,000 addresses but no more than necessary?
- A.** /20
 - B.** /22
 - C.** /28
 - D.** /32
- 19.** A team of data scientists have asked for your help setting up an Apache Spark cluster. You suggest they use a managed Google Cloud service instead of managing a cluster themselves on Compute Engine. The service they would use is:
- A.** Dataproc
 - B.** Cloud Dataflow
 - C.** Cloud Hadoop
 - D.** BigQuery
- 20.** You have created a web application that allows users to upload files to Cloud Storage. When files are uploaded, you want to check the file size and update the user's total storage used in their account. A serverless option for performing this action on load is:
- A.** Cloud Dataflow
 - B.** Dataproc
 - C.** Cloud Storage
 - D.** Cloud Functions
- 21.** Your company has just started using Google Cloud, and executives want to have a dedicated connection from your data center to the Google Cloud to allow for large data transfers. Which networking service would you recommend?
- A.** Google Cloud Carrier Internet Peering
 - B.** Google Cloud Dedicated Interconnect
 - C.** Google Cloud Internet Peering
 - D.** Google Cloud DNS
- 22.** You want to have Google Cloud manage cryptographic keys, so you've decided to use Cloud Key Management Services. Before you can start creating cryptographic keys, you must:
- A.** Enable Google Cloud Key Management Service (KMS) API and set up billing.
 - B.** Enable Google Cloud KMS API and create folders.
 - C.** Create folders and set up billing.
 - D.** Give all users grantable roles to create keys.

- 23.** In Kubernetes Engine, a node pool is:
- A.** A subset of nodes across clusters
 - B.** A set of VMs managed outside of Kubernetes Engine
 - C.** A set of preemptible VMs
 - D.** A subset of node instances within a cluster that all have the same configuration
- 24.** The Google Cloud service for storing and managing Docker containers is:
- A.** Cloud DevOps Repository
 - B.** Cloud Build
 - C.** Container Registry
 - D.** Docker Repository
- 25.** Code for Cloud Functions can be written in several languages, including:
- A.** Node.js and Python only
 - B.** Node.js, Python, and Go
 - C.** Python and Go
 - D.** Python and C

Answers to Assessment Test

1. A. Machine type, boot disk image or container image, zone, and labels are all configuration parameters or attributes of a VM and therefore would be included in an instance group configuration that creates those VMs.
2. B. `gsutil` is the command line for accessing and manipulating Cloud Storage from the command line. `mb` is the specific command for creating, or making, a bucket.
3. D. The life cycle configuration policy allows you to specify criteria for migrating data to other storage systems without having to concern yourself with running jobs to actually execute the necessary steps. The other options are inefficient or do not exist.
4. A. `gsutil` is the command-line tool for working with Cloud Storage. `rsync` is the specific command in `gsutil` for synchronizing buckets.
5. C. Google operates a global network, and VPCs are resources that can span that global network.
6. B. `gcloud` by default will retry a failed network operation and will wait a long time before each retry. The time to wait is calculated using a truncated binary exponential backoff strategy.
7. C. Firewall rules do not have TTL parameters. Direction of traffic, action on match, and protocol are all components of firewall rules.
8. C. IAM policy violations do not trigger changes in the size of clusters. All other options can be used to trigger a change in cluster size.
9. A. Only Spanner and Cloud SQL databases support transactions and have a SQL interface. Firestore has transactions but does not support fully compliant SQL; it has a SQL-like query language. Cloud Storage does not support transactions or SQL.
10. C. Cloud Run is a serverless service for running containers and allows developers to deploy full applications without having to manage servers or clusters. Compute Engine and Kubernetes Engine require management of servers. Cloud Functions is suitable for short-running Node.js or Python functions but not full applications.
11. B. BigQuery is designed for petabyte-scale analytics and provides a SQL interface.
12. B. Cloud Dataflow allows for stream and batch processing of data and is well suited for this kind of ETL work. Dataproc is a managed Hadoop and Spark service that is used for big data analytics. Cloud Endpoints is an API service, and Cloud Interconnect is a network service.
13. C. If a preemptible machine has not been shut down within 24 hours, Google will stop the instance.
14. A. Organizations, folders, and projects are the components used to manage an organizational hierarchy. Buckets, directories, and subdirectories are used to organize storage.

- 15. B. `gcloud` is the command-line tool for working with IAM, and `list-grantable-roles` is the correct command.
- 16. A. `gcloud` is the command-line tool for manipulating compute resources, and `zones describe` is the correct command.
- 17. C. `iam.roles.create` is correct; the other roles do not exist.
- 18. B. The `/22` suffix produces 1,022 usable IP addresses.
- 19. A. Dataproc is the managed Spark service. Cloud Dataflow is for stream and batch processing of data, BigQuery is for analytics, and Cloud Hadoop is not a Google Cloud service.
- 20. D. Cloud Functions respond to events in Cloud Storage, making them a good choice for taking an action after a file is loaded.
- 21. B. Google Cloud Dedicated Interconnect is the only option for a dedicated connection between a customer's data center and a Google data center.
- 22. A. Enabling the Google Cloud KMS API and setting up billing are steps common to using Google Cloud services.
- 23. D. A node pool is a subset of node instances within a cluster that all have the same configuration.
- 24. C. The Google Cloud service for storing and managing Docker containers is Artifact Registry. Cloud Build is for creating images. Cloud Source Repositories are private Git repositories hosted on Google Cloud. Docker Repository is not a Google Cloud service.
- 25. B. Node.js, Python, and Go are three of the languages supported by Cloud Functions.

Chapter 1

Overview of Google Cloud

**THIS CHAPTER COVERS THE FOLLOWING
OBJECTIVE OF THE GOOGLE ASSOCIATE
CLOUD ENGINEER CERTIFICATION EXAM:**

- ✓ 1.0 Setting up cloud projects and accounts





Google Cloud is a public cloud service that offers some of the same technologies used by Google to deliver its own products. This chapter describes the most important components of Google Cloud and discusses how it differs from on-premises data center-based computing.

Types of Cloud Services

Public cloud providers such as Google, Amazon, and Microsoft offer a range of services for deploying computing, storage, networking, and other infrastructures to run a wide array of business services and applications. Some cloud users are new companies that start in the cloud. They have never owned their own hardware infrastructure. Other cloud customers are enterprises with multiple data centers that use public clouds to supplement their data centers. These different kinds of users have different requirements.

A company that starts on the cloud can choose services that best fit its application and architectural needs without having to consider existing infrastructure. For example, a startup could use Google Cloud's Cloud Identity, including Identity Access Management services, for all authentication and authorization needs. A company that has already invested in a Microsoft Active Directory solution for identity management may want to leverage that system instead of working solely with the cloud's identity management system. This can lead to additional work to integrate the two systems and keep them synchronized.

Another area of concern for enterprises with their own infrastructure is establishing and maintaining a secure network between their on-premises resources and their public cloud resources. If there will be high-volume network traffic between the on-premises systems and the public cloud, the enterprise may need to invest in dedicated networking between its data center and a facility of the public cloud provider. If the volume of traffic does not justify the cost of a dedicated connection between facilities, then the company may use a virtual private network that runs over the public Internet. This requires additional network design and management that a company that is solely in the cloud would not have to address.

Public cloud providers offer services that fall into four broad categories:

- Compute resources
- Storage
- Networking
- Specialized services such as machine learning services

Cloud customers typically make use of services in more than one of these categories.

Compute Resources

Computing resources come in a variety of forms in public clouds.

Virtual Machines

Virtual machines (VMs) are a basic unit of computing resources and a good starting point for experimenting with the cloud. After you create an account with a cloud provider and provide billing information, you can create a project, which is a logical grouping of Google Cloud resources. After you have a project, you can use a portal or command-line tools to create VMs in a project. Google Cloud offers a variety of preconfigured VMs with varying numbers of vCPUs and amounts of memory. You can also create a custom configuration if the preconfigured offerings do not meet your needs.

Once you create a VM, you can log into it and administer it as you like. You have full access to the VM, so you can configure filesystems, add persistent storage, patch the operating system, or install additional packages. You decide what to run on the VM, who else will have access to it, and when to shut down the VM. A VM that you manage is like having a server in your office that you have full administrator rights to.

You can, of course, create multiple VMs running different operating systems and applications. Google Cloud also provides services, such as load balancers, that provide a single access point to a distributed back end. This is especially useful when you need to have high availability for your application. If one of the VMs in a cluster fails, the workload can be directed to the other VMs in the cluster. Autoscalers can add or remove VMs from the cluster based on the workload. This is called *autoscaling*. This helps both control cost by not running more VMs than needed and ensure that sufficient computing capacity is available when workloads increase.

Managed Kubernetes Clusters

Google Cloud gives you all the tools you need to create and manage clusters of servers. Many cloud users would rather focus on their applications and not the tasks needed to keep a cluster of servers up and running. For those users, managed clusters are a good option.

Managed clusters make use of containers. A *container* is like a lightweight VM that isolates processes running in one container from processes running in another container on the same server. In a managed cluster, you can specify the number of servers you would like to run and the containers that should run on them. You can also specify autoscaling parameters to optimize the number of containers running.

In a managed cluster, the health of containers is monitored for you. If a container fails, the cluster management software will detect it and start another container.

Containers are good options when you need to run applications that depend on multiple microservices running in your environment. The services are deployed through containers, and the cluster management service takes care of monitoring, networking, and some security management tasks.

Serverless Computing

Both VMs and managed Kubernetes clusters require some level of effort to configure and administer computing resources. Serverless computing is an approach that allows developers and application administrators to run their code in a computing environment that does not require setting up VMs or Kubernetes clusters.

Google Cloud has three serverless computing options: App Engine, Cloud Run, and Cloud Functions. App Engine is used for applications and containers that run for extended periods of time, such as a website back end, point-of-sale system, or custom business application. Cloud Run is also used to run containers when the full features of Kubernetes Engine are not needed. Cloud Run is used for containers when you want a fully managed service and rapid autoscaling for your stateless applications. Cloud Functions is a platform for running code in response to an event, such as uploading a file or adding a message to a message queue. This serverless option works well when you need to respond to an event by running a short process coded in a function or by calling a longer-running application that might be running on a VM, managed cluster, or App Engine.

Storage

Public clouds offer a few types of storage services that are useful for a wide range of application requirements. These types include the following:

- Object storage
- File storage
- Block storage
- Caches

Enterprise users of cloud services will often use a combination of these services.

Object Storage

Object storage is a system that manages the use of storage in terms of objects or blobs. Usually these objects are files, but it is important to note that the files are not stored in a conventional filesystem. Objects are grouped into *buckets*. Each object is individually addressable, usually by a URL.

Object storage is not limited by the size of disks or solid-state drives (SSDs) attached to a server. Objects can be uploaded without concern for the amount of space available on a disk. Multiple copies of objects are stored to improve availability and durability. In some cases, copies of objects may be stored in different regions to ensure availability even if a region becomes inaccessible.

Another advantage of object storage is that it is serverless. There is no need to create VMs and attach storage to them. Google Cloud's object storage, called Cloud Storage, is accessible from servers running in Google Cloud as well as from other devices with Internet access.

File Storage

File storage services provide a hierarchical storage system for files. Filesystem storage provides network-shared filesystems. Google Cloud has a file storage service called Cloud Filestore, which is based on the Network File System (NFS) storage system.

File storage is suitable for applications that require operating system–like file access to files. The file storage system decouples the filesystem from specific VMs. The filesystem, its directories, and its files exist independent of VMs or applications that may access those files.

Block Storage

Block storage uses a fixed-size data structure called a *block* to organize data. Block storage is commonly used in ephemeral and persistent disks attached to VMs. With a block storage system, you can install filesystems on top of the block storage, or you can run applications that access blocks directly. Some relational databases can be designed to access blocks directly rather than working through filesystems.

In Linux filesystems, 4 KB is a common block size. Relational databases often write directly to blocks, but they often use larger sizes, such as 8 KB or more.

Block storage is available on disks that are attached to VMs in Google Cloud. Block storage can be either persistent or ephemeral. A persistent disk continues to exist and store data, even if it is detached from a virtual server or the virtual server to which it is attached shuts down. Ephemeral disks exist and store data only as long as a VM is running. Ephemeral disks are deleted when the VM is shut down. Persistent disks are used when you want data to exist on a block storage device independent of a VM. These disks are good options when you have data that you want available independent of the life cycle of a VM, and support fast operating system– and filesystem-level access.

Object storage also keeps data independent of the life cycle of a VM, but it does not support operating system– or filesystem-level access; you have to use higher-level protocols like HTTP to access objects. It takes longer to retrieve data from object storage than to retrieve it from block storage. You may need a combination of object storage and block storage to meet your application needs. Object storage can store large volumes of data that are copied to persistent disk when needed. This combination gives the advantage of large volumes of storage along with operating system– and filesystem-based access when needed.

Caches

Caches are in-memory data stores that maintain fast access to data. The time it takes to retrieve data is called *latency*. The latency of in-memory stores is designed to be submillisecond. To give you a comparison, here are some other latencies:

- Making a main memory reference takes 100 nanoseconds, or 0.1 microsecond.
- Reading 4 KB randomly from an SSD takes 150 microseconds.
- Reading 1 MB sequentially from memory takes 250 microseconds.

- Reading 1 MB sequentially from an SSD takes 1,000 microseconds, or 1 millisecond.
- Reading 1 MB sequentially from disk takes 20,000 microseconds, or 20 milliseconds.

Here are some conversions for reference:

- 1,000 nanoseconds equal 1 microsecond.
- 1,000 microseconds equal 1 millisecond.
- 1,000 milliseconds equal 1 second.

These and other useful timing data are available at Jonas Bonér’s “Latency Numbers Every Programmer Should Know” at <https://gist.github.com/jboner/2841832>.

Let’s work through an example of reading 1 MB of data. If you have the data stored in an in-memory cache, you can retrieve the data in 250 microseconds, or 0.25 millisecond. If that same data is stored on an SSD, it will take four times as long to retrieve at 1 millisecond. If you retrieve the same data from a hard disk drive (HDD), you can expect to wait 20 milliseconds, or 80 times as long as reading from an in-memory cache.

Caches are quite helpful when you need to keep read latency to a minimum in your application. Of course, who doesn’t love fast retrieval times? Why don’t we always store our data in caches? There are three reasons:

- Memory is more expensive than SSD or HDD storage. It’s not practical in many cases to have as much in-memory storage as persistent block storage on SSDs or HDDs.
- Caches are volatile; you lose the data stored in the cache when power is lost or the operating system is rebooted. You can store data in a cache for fast access, but it should never be used as the only data store keeping the data. Some form of persistent storage should be used to maintain a “system of truth,” or a data store that always has the latest and most accurate version of the data.
- Caches can get out of synchronization with the system of truth. This can happen if the system of truth is updated but the new data is not written to the cache. When this happens, it can be difficult for an application that depends on the cache to detect the fact that data in the cache is invalid. If you decide to use a cache, be sure to design a cache update strategy that meets your requirements for consistency between the cache and the system of truth. This is such a challenging design problem that it has become memorialized in Phil Karlton’s well-known quip, “There are only two hard things in computer science: cache invalidation and naming things.” (See <https://martinfowler.com/bliki/TwoHardThings.html> for riffs on this rare example of computer science humor.)



Real World Scenario

Improving Database Query Response Time

Users expect web applications to be highly responsive. If a page takes more than 2 to 3 seconds to load, the user experience can suffer. It is common to generate the content of a page using the results of a database query, such as looking up account information by

customer ID. When a query is made to the database, the database engine will look up the data, which is usually on disk. The more users query the database, the more queries it has to serve. Databases keep a queue for queries that need to be answered but can't be processed yet because the database is busy with other queries. This can cause longer latency response time, since the web application will have to wait for the database to return the query results.

One way to reduce latency is to reduce the time needed to read the data. In some cases, it helps to replace hard disk drives with faster SSD drives. However, if the volume of queries is high enough that the queue of queries is long even with SSDs, another option is to use a cache.

When query results are fetched, they are stored in the cache. The next time that information is needed, it is fetched from the cache instead of the database. This can reduce latency because data is fetched from memory, which is faster than disk. It also reduces the number of queries to the database, so queries that can't be answered by looking up data in the cache won't have to wait as long in the query queue before being processed.

This approach would require changes to the application code to store query results in the cache and to check the cache for data before querying the database.

Networking

When working in the cloud, you'll need to work with networking between your cloud resources and possibly with your on-premises systems.

When you have multiple VMs running in your cloud environment, you will likely need to manage IP addresses at some point. Each network-accessible device or service in your environment will need an IP address. In fact, devices within Google Cloud can have both internal and external addresses. Internal addresses are accessible only to services in your internal network. Your internal Google Cloud network is defined as a virtual private cloud (VPC). External addresses are accessible from the Internet.

External IP addresses can be either static or ephemeral. Static addresses are assigned to a device for extended periods of time. Ephemeral external IP addresses are attached to VMs and released when the VM is stopped.

In addition to specifying IP addresses, you will often need to define firewall rules to control access to subnetworks and VMs in your VPC. For example, you may have a database server that you want to restrict access to so that only an application server can query the database. A firewall rule can be configured to limit inbound and outbound traffic to the IP address of the application server or load balancer in front of the application cluster.

You may need to share data and network access between an on-premises data center and your VPC. You can do this using one of several types of *peering*, which is the general term for linking distinct networks. Google Cloud offers several types of peering, including VPNs, Interconnects, Shared VPC, VPC networking peering, and Direct or Carrier peering.

Specialized Services

Most public cloud providers offer specialized services that can be used as building blocks of applications or as part of a workflow for processing data. Common characteristics of specialized services are as follows:

- They are serverless; you do not need to configure servers or clusters.
- They provide a specific function, such as translating text or analyzing images.
- They provide an application programming interface (API) to access the functionality of the service.
- As with other cloud services, you are charged based on your use of the service.

These are some of the specialized services in Google Cloud:

- AutoML, a machine learning service
- Cloud Natural Language, a service for analyzing text
- Speech-to-Text for converting spoken language to text
- Recommendations AI for personalized product recommendations

Specialized services encapsulate advanced computing capabilities and make them accessible to developers who are not experts in domains, such as natural language processing and machine learning. Expect to see more specialized services added to Google Cloud.

Cloud Computing vs. Data Center Computing

Although it may seem that running VMs in the cloud is not much different from running them in your data center, there are significant differences between operating IT environments in the cloud and an on-premises or colocated data center.

Rent Instead of Own Resources

Corporate data centers are filled with servers, disk arrays, and networking equipment. This equipment is often owned or leased for extended periods by the company, a model that requires companies to either spend a significant amount of money up front to purchase equipment or commit to a long-term lease for the equipment. This approach works well when an organization can accurately predict the number of servers and other equipment it will need for an extended period and it can utilize that equipment consistently.

The model does not work as well when companies have to plan for peak capacity that is significantly higher than the average workload. For example, a retailer may have an

average load that requires a cluster of 20 servers but during the holiday season the workload increases to the point where 80 servers are needed. The company could purchase 80 servers and let 60 idle for most of the year to have resources to accommodate peak capacity. Alternatively, it could purchase or lease fewer servers and tolerate the loss in business that would occur when its compute resources can't keep up with demand. Neither is an appealing option.

Public clouds offer an alternative of short-term rental of compute capacity. The retailer, for example, could run VMs in the cloud during peak periods in addition to its on-premises servers. This gives the retailer access to the servers it needs when it needs them without having to pay for them when they are not needed.

The unit cost of running servers in the cloud may be higher than that of running the equivalent server in the data center, but the total cost of on-premises and short-term in the cloud mix of servers may still be significantly less than the cost of purchasing or leasing for peak capacity and leaving resources idle.

Pay-as-You-Go-for-What-You-Use Model

Related to the short-term rental model of cloud computing is the pay-as-you-go model. When you run a virtual server in the cloud, you will typically pay for a minimum period, such as 1 minute, and pay per second thereafter. The unit cost per second will vary depending on the characteristics of the server. Servers with more CPUs and memory will cost more than servers with fewer CPUs and less memory.

It is important for cloud engineers to understand the pricing model of their cloud provider. It is easy to run up a large bill for servers and storage if you are not monitoring your usage. In fact, some cloud customers find that running applications in the cloud can be more expensive than running them on-premises.

Elastic Resource Allocation

Another key differentiator between on-premises and public cloud computing is the ability to add and remove compute and storage resources on short notice. In the cloud, you could start 20 servers in a matter of minutes. In an on-premises data center, it could take days or weeks to do the same thing if additional hardware must be provisioned.

Cloud providers design their data centers with extensive compute, storage, and network resources. They optimize their investment by efficiently renting these resources to customers. With sufficient data about customer use patterns, they can predict the capacity they need to meet customer demand. Since they have many customers, the variation in demand of any one customer has little effect on the overall use of their resources.

Extensive resources and the ability to quickly shift resources between customers enables public cloud providers to offer elastic resource allocation more efficiently than can be done in smaller data centers.

Specialized Services

Specialized services are, by their nature, not widely understood. Many developers understand how to develop user interfaces or query a database, but fewer have been exposed to the details of natural language processing or machine learning. Large enterprises may have the financial resources to develop in-house expertise in areas such as data science and machine vision, but many others don't.

By offering specialized services, cloud providers are bringing advanced capabilities to a wider audience of developers. Like investing in large amounts of hardware, public cloud vendors can invest in specialized services and recover their costs and make a profit because the specialized services are used by a large number of customers.

Summary

Google Cloud offers a variety of services for compute, storage, networking, and specialized services. Compute services include virtual machines and Kubernetes clusters, while storage services support object and file storage along with caching. Networking services provide virtual private clouds and other services including VPNs, Interconnects, Shared VPC, VPC networking peering, and Direct or Carrier peering. Specialized services include machine learning, Speech-to-Text, and recommendation services.

Cloud computing has several advantages over on-premises computing including: renting rather than owning infrastructure, a pay-as-you-go model, elastic resource allocation, and specialized services.

Exam Essentials

Understand different ways of delivering cloud computing resources. Computing resources can be allocated as individual VMs or clusters of VMs that you manage. You can also use managed Kubernetes clusters that relieve you of some of the operational overhead of managing a Kubernetes cluster. Serverless computing options relieve users of any server management. Instead, developers run their code in a containerized environment managed by the cloud provider or in a compute platform designed for short-running code. Developers and DevOps professionals have the most control over resources when they manage their own servers and clusters. Managed services and serverless options are good choices when you do not need control over the computing environment and will get more value from not having to manage compute resources.

Understand the different forms of cloud storage and when to use them. There are four main categories of storage: object, file, block, and in-memory caches. Object storage is designed for highly reliable and durable storage of objects, such as images or data sets.

Object storage has more limited functionality than filesystem-based storage systems. Filesystem-based storage provides hierarchical directory storage for files and supports common operating system and filesystem functions. Filesystem services provide network-accessible filesystems that can be accessed by multiple servers. Block storage is used for storing data on disks. Filesystems and databases make use of block storage systems. Block storage is used with persistent storage devices, such as SSDs and HDDs. Caches are in-memory data stores used to minimize the latency of retrieving data. They do not provide persistent storage and should never be considered a “system of truth.”

Understand the differences between running an IT environment on-premises or in the cloud. Running an IT environment in the cloud has several advantages, including short-term rental of resources, a pay-as-you-go model, elastic resource allocation, and the ability to use specialized services. The unit cost of cloud resources, such as the cost per minute of a mid-tier server, may be higher in the cloud than on-premises. It is important to understand the cost model of your cloud provider so that you can make decisions about the most efficient distribution of workload between cloud and on-premises resources.

Review Questions

You can find the answers in the Appendix.

1. Which of the following is an option for choosing a computing resource in Google Cloud?
 - A. Cache
 - B. Virtual machine (VM)
 - C. Block
 - D. Subnet
2. If you use a cluster that is managed by a cloud provider, which of these will be managed for you by the cloud provider?
 - A. Monitoring
 - B. Networking
 - C. Some security management tasks
 - D. All of the above
3. You need serverless computing for file processing and running the back end of a website; which two products can you choose from Google Cloud?
 - A. Kubernetes Engine and Compute Engine
 - B. Cloud Run and Cloud Functions
 - C. Cloud Functions and Compute Engine
 - D. Cloud Functions and Kubernetes Engine
4. You have been asked to design a storage system for a web application that allows users to upload large data files to be analyzed by a data analytics workflow. The files should be stored in a high-availability storage system. Filesystem functionality is not required. Which storage system in Google Cloud should be used?
 - A. Block storage
 - B. Object storage
 - C. Cache
 - D. Network File System
5. All block storage systems use what block size?
 - A. 4 KB.
 - B. 8 KB.
 - C. 16 KB.
 - D. Block size can vary.

6. You have been asked to set up network security in a virtual private cloud. Your company wants to have multiple subnetworks and limit traffic between the subnetworks. Which network security control would you use to control the flow of traffic between subnets?
 - A. Identity access management
 - B. Router
 - C. Firewall
 - D. IP address table
7. When you create a machine learning service to learn how to classify objects using tabular data, what type of servers should you choose to manage compute resources?
 - A. Virtual machines (VMs).
 - B. Clusters of VMs.
 - C. No servers; you should use specialized services, which are serverless.
 - D. VMs running Linux only.
8. When does investing in servers for extended periods of time, such as committing to use servers for three to five years, work well?
 - A. When a company is just starting up
 - B. When a company can accurately predict server need for an extended period of time
 - C. When a company has a fixed IT budget
 - D. When a company has a variable IT budget
9. Your company is based in North America and will be running a virtual server for batch processing invoices. What factor determines the unit per minute cost?
 - A. The time of day the virtual machine (VM) is run
 - B. The characteristics of the server
 - C. The application you run
 - D. None of the above
10. You plan to use AutoML to analyze sales data and predict product demand in the near future. You plan to analyze between 1,000 and 2,500 products per hour. How many VMs should you allocate to meet peak demand?
 - A. 1.
 - B. 10.
 - C. 25.
 - D. None; AutoML is a serverless service.

11. You have to run a number of services to support an application. Which of the following is a good deployment model?
 - A. Run on a large, single VM.
 - B. Use containers in a managed cluster.
 - C. Use two large VMs, making one of them read only.
 - D. Use a small VM for all services and increase the size of the VM when CPU utilization exceeds 90 percent.
12. You have created a VM. Which of the following system administration operations are you allowed to perform on it?
 - A. Configure the filesystem.
 - B. Patch operating system software.
 - C. Change file and directory permissions.
 - D. All of the above.
13. Cloud Filestore is based on what filesystem technology?
 - A. Network File System (NFS)
 - B. XFS
 - C. EXT4
 - D. ReiserFS
14. When creating resources in Google Cloud, those resources are always part of what?
 - A. Virtual private cloud
 - B. Subdomain
 - C. Cluster
 - D. None of the above
15. You need to store data for an application and are using a cache. How will the cache affect data retrieval?
 - A. A cache improves the execution of client-side JavaScript.
 - B. A cache will continue to store data even if power is lost, improving availability.
 - C. Caches can get out of sync with the system of truth.
 - D. Using a cache will reduce latency, since retrieving from a cache is faster than retrieving from SSDs or HDDs.
16. Why can cloud providers offer elastic resource allocation?
 - A. Cloud providers can take resources from lower-priority customers and give them to higher-priority customers.
 - B. Extensive resources and the ability to quickly shift resources between customers enables public cloud providers to offer elastic resource allocation more efficiently than can be done in smaller data centers.
 - C. They charge more the more resources you use.
 - D. They don't.

- 17.** What is not a characteristic of specialized services in Google Cloud?
- A.** They are serverless; you do not need to configure servers or clusters.
 - B.** They provide a specific function, such as translating text or analyzing images.
 - C.** They require monitoring by the user.
 - D.** They provide an API to access the functionality of the service.
- 18.** Your client's transactions must access a drive attached to a VM that allows for random access to parts of files. What kind of storage does the attached drive provide?
- A.** Object storage
 - B.** Block storage
 - C.** NoSQL storage
 - D.** SQL storage
- 19.** You are deploying a new relational database to support a web application. Which type of storage system would you use to store data files of the database?
- A.** Object storage
 - B.** Data storage
 - C.** Block storage
 - D.** Cache
- 20.** A user prefers services that require minimal setup; why would you recommend Cloud Storage, Cloud Run, and Cloud Functions?
- A.** They are charged only by time.
 - B.** They are serverless.
 - C.** They require a user to configure VMs.
 - D.** They can only run applications written in Go.

Chapter 2

Google Cloud Computing Services

THIS CHAPTER COVERS THE FOLLOWING OBJECTIVES OF THE GOOGLE ASSOCIATE CLOUD ENGINEER CERTIFICATION EXAM:

- ✓ 2.2 Planning and configuring compute resources
- ✓ 3.4 Deploying and implementing data solutions





Google Cloud is made up of a wide array of services that meet a variety of computing, storage, and networking needs. This chapter provides an overview of the most important Google Cloud computing services and describes some important use cases for these services.

Computing Components of Google Cloud

Google Cloud is a suite of cloud computing services that includes compute, storage, and networking services designed to meet the needs of a wide range of cloud computing customers. Small businesses may be attracted to virtual machines (VMs) and storage services. Large businesses and other sizable organizations may be more interested in access to highly scalable clusters of VMs, a variety of relational and NoSQL databases, specialized networking services, and advanced artificial intelligence and machine learning capabilities.

This chapter provides an overview of many of Google Cloud's services. The breadth of services available in the Google Cloud continues to grow. By the time you read this, Google may be offering additional services. Most of the services can be grouped into several core categories.

- Computing resources
- Storage resources
- Databases
- Networking services
- Identity management and security
- Development tools
- Management tools
- Specialized services

A Google-certified Associate Cloud Engineer should be familiar with the services in each category, how they are used, and the advantages and disadvantages of the various services in each category.

Computing Resources

Public cloud services provide a range of computing service options. At one end of the spectrum, customers can create and manage VMs themselves. This model gives the cloud user the greatest control of all the computing services. Users can choose the operating system to run, which packages to install, and when to back up and perform other maintenance operations. This type of computing service is typically referred to as infrastructure as a service (IaaS).

An alternative model is called platform as a service (PaaS), which provides a runtime environment to execute applications without the need to manage underlying servers, networks, and storage systems.

One of IaaS computing products is called Compute Engine, and the PaaS offerings are App Engine and Cloud Functions. In addition, Google offers Kubernetes Engine, which is a service for managing containers in a cluster; this type of service is an increasingly popular alternative to managing individual sets of VMs.

Compute Engine

Compute Engine is a service that allows users to create VMs, attach persistent storage to those VMs, and make use of other Google Cloud services, such as Cloud Storage.

VMs are abstractions of physical servers. They are essentially programs that emulate physical servers and provide CPU, memory, storage, and other services that you would find if you ran your favorite operating system on a server under your desk or in a data center. VMs run within a low-level service called a *hypervisor*. Google Cloud uses a security-hardened version of the KVM hypervisor. KVM stands for Kernel Virtual Machine and provides virtualization on Linux systems running on x86 hardware.

Hypervisors run operating systems like Linux or Windows Server. Hypervisors can run multiple operating systems, referred to as *guest operating systems*, while keeping the activities of each isolated from other guest operating systems. Each instance of an executing guest operating system is a VM instance. Figure 2.1 shows the logical organization of VM instances running on a physical server.

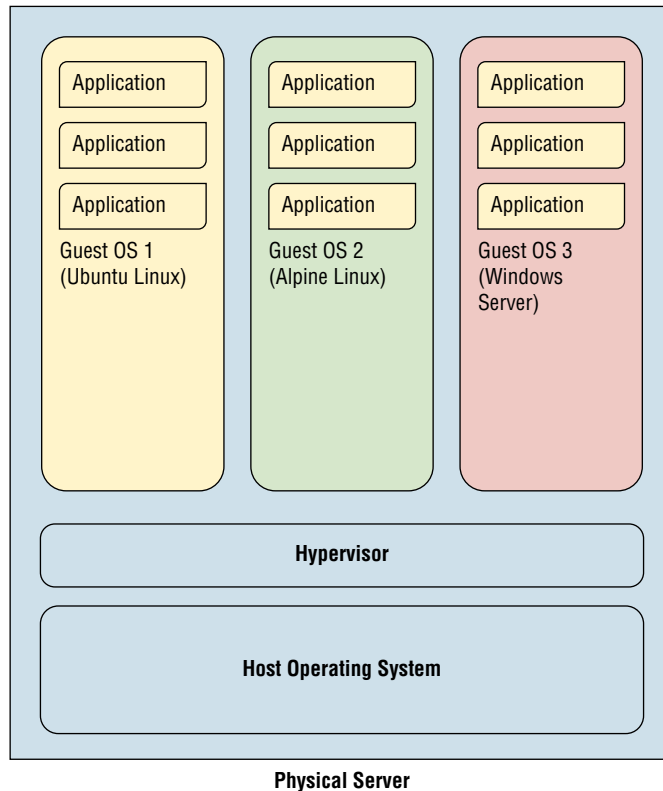
VMs come in a range of predefined sizes, but you can also create a customized configuration. When you create an instance, you can specify several parameters, including the following:

- The operating system
- The size of persistent storage
- Whether you'll add graphical processing units (GPUs) for compute-intensive operations like machine learning
- Whether you'll make the VM preemptible

The last option, making a VM preemptible, means you may be charged significantly less for the VM than normal (around 80 percent less), but your VM could be shut down at any time by Google. It will be shut down after the preemptible VM has run for at least 24 hours.

The latest version of preemptible VMs are known as *spot instances* and use the same pricing model as preemptible VM; however, spot instances do not have a maximum runtime and will not be shut down after 24 hours.

FIGURE 2.1 VM instances running within a hypervisor



Chapter 4, “Introduction to Computing in Google Cloud,” will introduce the details of managing Compute Engine VMs. To explore Compute Engine, log into the Google Cloud Console, navigate to the main menu on the left, and select Compute Engine.

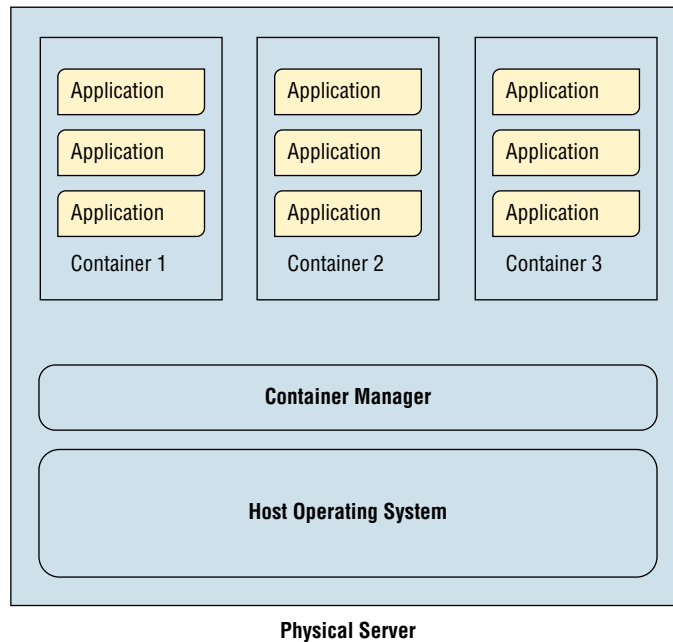
Kubernetes Engine

Kubernetes Engine is designed to allow users to easily run containerized applications on a cluster of servers. Containers are often compared to VMs because they are each used for isolating computing processing and resources. Containers take a different approach than VMs for isolating computing processes.

As mentioned, a VM runs a guest operating system on a physical server. The physical server runs an operating system as well, along with a hypervisor. Another approach to

isolating computing resources is to use features of the host operating system to isolate processes and resources. With this approach, there is no need for a hypervisor; the host operating system maintains isolation. Instead, a container manager is used. That is, a single container manager coordinates containers running on the server. No additional, or guest, operating systems run on top of the container manager. Instead, containers make use of host operating system functionality, while the operating system and container manager ensure isolation between the running containers. Figure 2.2 shows the logical structure of containers.

FIGURE 2.2 Containers running on a physical server



Kubernetes Engine is a Google Cloud product that allows users to describe the compute, storage, and memory resources they'd like to run their services. Kubernetes Engine then provisions the underlying resources. It's easy to add and remove resources from a Kubernetes cluster using a command-line interface or a graphical user interface.

In addition, Kubernetes monitors the health of servers in the cluster and automatically repairs problems, such as failed servers. Kubernetes Engine also supports autoscaling, so if the load on your applications increases, Kubernetes Engine will allocate additional resources.

Anthos clusters extend GKE for hybrid and multicloud environments by providing services to create, scale, and upgrade conformant Kubernetes clusters along with a common orchestration layer. Multiple clusters can be managed as a group known as a *fleet*. Anthos clusters can be connected using standard networking options, including VPNs, Dedicated Interconnect, and Partner Interconnects.

There are several key benefits to using Anthos to manage multiple Kubernetes clusters. These include:

- Centralized management of configuration as code
- Ability to roll back deployments with Git
- A single view of cluster infrastructure and applications
- Centralized and auditable workflows
- Instrumentation of code using Anthos Service Mesh
- Anthos Service Mesh authorization and routing

In addition, Anthos includes Migrate for Anthos for GKE, which is a service that allows you to orchestrate migrations using Kubernetes and Anthos.

The term “Anthos clusters” refers to Google Kubernetes Engine clusters that have been extended to function on-premises or in multicloud environments.

Chapter 7, “Computing with Kubernetes,” will describe the details of planning and managing Kubernetes Engine. To explore Kubernetes Engine, log into the Google Cloud Console, navigate to the main menu on the left, and select Kubernetes Engine.

App Engine

App Engine is a Google Cloud compute PaaS offering. With App Engine, developers and application administrators don’t need to concern themselves with configuring VMs or specifying Kubernetes clusters. Instead, developers create applications in a popular programming language such as Java, Go, Python, or Node.js and deploy that code to a serverless application environment.

App Engine manages the underlying computing and network infrastructure. There is no need to configure VMs or harden networks to protect your application. App Engine is well suited for web and mobile back-end applications.

App Engine is available in two types:

- In the **standard** environment, you run applications in a language-specific sandbox, so your application is isolated from the underlying server’s operating system as well as from other applications running on that server. The standard environment is well suited to applications that are written in one of a supported languages and do not need operating system packages or other compiled software that would have to be installed along with the application code.
- In the **flexible** environment, you run containerized applications in the App Engine environment. The flexible environment works well in cases where you have application code but also need libraries or other third-party software installed. As the name implies, the flexible environment gives you more options, including the ability to work with background processes and write to local disk.

Chapter 9, “Computing with Cloud Run and App Engine,” will introduce details for using and managing App Engine. To explore App Engine, log into the Google Cloud Console, navigate to the main menu on the left, and select App Engine.

Cloud Run

Cloud Run is a Google Cloud service for running stateless containers. When using the managed service, you pay per use and can have up to 1,000 container instances by default.

Unlike App Engine Standard, Cloud Run does not restrict you to using a fixed set of programming languages. Cloud Run services have regional availability.

A service is the main abstraction of computing in Cloud Run. A service is in a region and replicated across multiple zones. A service may have multiple revisions. Cloud Run will auto-scale the number of instances based on load.

Cloud Functions

Google Cloud Functions is a lightweight computing option that is well suited to event-driven processing. Cloud Functions runs code in response to an event, like a file being uploaded to Cloud Storage or a message being written to a message queue. The code that executes in the Cloud Functions environment must be short-running—this computing service is not designed to execute long-running code. If you need to support long-running applications or jobs, consider Compute Engine, Kubernetes Engine, or App Engine.

Cloud Functions is often used to call other services, such as third-party application programming interfaces (APIs) or other Google Cloud services, like a natural language translation service.

Like App Engine and Cloud Run, Cloud Functions is a serverless product. Users only need to supply code; they do not need to configure VMs or create containers. Cloud Functions will automatically scale as load increases.

In addition to these computing products, Google Cloud offers a number of storage resources.

Chapter 10, “Computing with Cloud Functions,” will describe the details of using and managing Cloud Functions. To explore Cloud Functions, log into the Google Cloud Console, navigate to the main menu on the left, and select Cloud Functions.

Storage Components of Google Cloud

Applications and services that run in the cloud must meet a wide range of requirements when it comes to storage.

Storage Resources

Sometimes an application needs fast read and write times for moderate amounts of data. Other times, a business application may need access to petabytes of archival storage but can tolerate minutes and even hours to retrieve a document. Google Cloud has several storage resources for storing objects and files.

Cloud Storage

Cloud Storage is Google Cloud's object storage system. Objects can be any type of file or binary large objects, or blob. Objects are organized into buckets, which are analogous to directories in a filesystem. It is important to remember that Cloud Storage is not a filesystem; it is a service that receives, stores, and retrieves files or objects from a distributed storage system. Cloud Storage is not part of a VM in the way an attached persistent disk is. Cloud Storage is accessible from VMs, containers, or any other network device with appropriate privileges and so complements filesystems on persistent disks.

Each stored object is uniquely addressable by a URL. For example, a PDF version of this chapter, called `chapter1.pdf`, that if stored in a bucket named *ace-certification-exam-prep* would be addressable as follows:

```
https://storage.cloud.google.com/ace-certification-exam-prep/  
chapter1.pdf
```

Google Cloud users and others can be granted permission to read and write objects to a bucket. Often, an application will be granted privileges through a service account with Identity and Access Management (IAM) roles to enable the application to read and write to buckets.

Cloud Storage is useful for storing objects that are treated as single units of data. For example, an image file is a good candidate for object storage. Images are generally read and written all at once. There is rarely a need to retrieve only a portion of the image. In general, if you write or retrieve an object all at once and you need to store it independently of servers that may or may not be running at any time, then Cloud Storage is a good option.

There are different location types of cloud storage. Regional storage keeps copies of objects in a single Google Cloud *region*. Regions are distinct geographic areas that can have multiple *zones*, or deployment areas. A zone is considered a single failure domain, which means that if all instances of your application are running in a zone and there is a failure, then all instances of your application will be inaccessible. Regional storage is well suited for applications that run in the same region and need low-latency access to objects in Cloud Storage.

Cloud Storage has some useful advanced features, such as support for multiple regions. This feature provides for storing replicas of objects in multiple Google Cloud regions, which is important for high availability, durability, and low latency.



Real World Scenario

Multi-Region Storage

If there was an outage in region `us-east1` and your objects were stored only in that region, then you would not be able to access those objects during the outage. However, if you enabled multiregion storage, then your objects stored in `us-east1` would be stored in another region, such as `us-west1`, as well.

In addition to high availability and durability, multiregion storage allows for faster access to data when users or applications are distributed across regions.

Sometimes data needs to be kept for extended periods of time but is rarely accessed. In those cases, nearline and coldline storage classes are good options. Use nearline when you will access objects less than once per month, and use coldline when you will access objects less than once every 90 days.

The archive storage class is low-cost archival storage designed for high durability and infrequent access. This class of storage is suitable for data that is accessed less than once per year.

A useful feature of Cloud Storage is the set of life cycle management policies that can automatically manage objects based on policies you define. For example, you could define a policy that moves all objects more than 60 days old in a bucket from standard storage class to nearline storage class, or deletes any object in an archive storage bucket that is older than five years.

Persistent Disk

Persistent disks are storage services that are attached to VMs in Compute Engine or Kubernetes Engine. Persistent disks provide block storage on solid-state drives (SSDs) and hard disk drives (HDDs). SSDs are often used for low-latency applications where persistent disk performance is important. SSDs cost more than HDDs, so applications that require large amounts of persistent disk storage but can tolerate longer read and write times can use HDDs to meet their storage requirements.

An advantage of persistent disks on the Google Cloud is that these disks support multiple readers without a degradation in performance. This allows for multiple instances to read a single copy of data. Disks can also be resized as needed while in use without the need to restart your VMs.

Persistent disks can be up to 64 TB in size using either SSDs or HDDs. Multiple persistent disks can be attached to a single VM.

Cloud Storage for Firebase

Mobile app developers may find Cloud Storage for Firebase to be the best combination of cloud object storage and the ability to support uploads and downloads from mobile devices with sometimes unreliable network connections.

The Cloud Storage for Firebase API is designed to provide secure transmission as well as robust recovery mechanisms to handle potentially problematic network quality. Once files, like photos or music recordings, are uploaded into Cloud Storage, you can access those files through the Cloud Storage command-line interface and software development kits (SDKs).

Cloud Filestore

Sometimes, developers need to have access to a filesystem housed on network-attached storage. For these use cases, the Cloud Filestore service provides a shared filesystem for use with Compute Engine and Kubernetes Engine.

Filestore can provide high numbers of input-output operations per second (IOPS) as well as variable storage capacity. Filesystem administrators can configure Cloud Filestore to meet their specific IOPS and capacity requirements.

Filestore implements the Network File System (NFS) protocol so that system administrators can easily mount shared filesystems on virtual servers.

Storage systems like the ones just described are used to store coarse-grained objects such as files. When data is more finely structured and has to be retrieved using query languages that describe the subset of data to return, then it is best to use a database management system.

Chapter 11, “Planning Storage in the Cloud,” describes details and guidance for planning storage services. To explore storage options, log into the Google Cloud Console, navigate to the main menu on the left, and select Storage or Filestore.

Databases

Google Cloud provides several database options. Some are relational databases, and some are NoSQL databases. Some are serverless and others require users to manage clusters of servers. Some provide support for atomic transactions, and others are better suited for applications with less stringent consistency and transaction requirements. Google Cloud users must understand their application requirements before choosing a service, and doing so is especially important when choosing a database, which often provides core storage services in the application stack.

Cloud SQL

Cloud SQL is Google Cloud managed relational database service that allows users to set up MySQL, PostgreSQL, and SQL Server databases on VMs without having to attend to database administration tasks, such as backing up databases or patching database software.

This database service includes management of replication and allows for automatic failover, providing for highly available databases.

Relational databases are well suited to applications with relatively consistent data structure requirements. For example, a banking database may track account numbers, customer names, addresses, and so on. Since virtually all records in the database will need the same information, this application is a good fit for a relational database.

Cloud Bigtable

Cloud Bigtable is designed for petabyte-scale applications that can manage up to billions of rows and thousands of columns. It is based on a NoSQL model known as a *wide-column data model*, which is different from relational databases such as Cloud SQL. Bigtable is suited for applications that require low-latency write and read operations. It is designed to support millions of operations per second.

Bigtable integrates with other Google Cloud services, such as Cloud Storage, Cloud Pub/Sub, Cloud Dataflow, and Cloud Dataproc. It also supports the HBase API, which is an API

for data access in the Hadoop big data ecosystem. Bigtable also integrates with open source tools for data processing, graph analysis, and time-series analysis.

Cloud Spanner

Cloud Spanner is Google's globally distributed relational database that combines the key benefits of relational databases, such as strong consistency and transactions, with the ability to scale horizontally like a NoSQL database. Spanner is a high-availability database with a 99.999 percent availability service level agreement (SLA), making it a good option for enterprise applications that demand scalable, highly available relational database services.

Cloud Spanner also has enterprise-grade security with encryption at rest and encryption in transit, along with identity-based access controls.

Cloud Spanner supports ANSI 2011 standard SQL.

Cloud Firestore

Cloud Firestore, formerly known as Cloud Datastore, is a NoSQL document database. This kind of database uses the concept of a document, or collection of key-value pairs, as the basic building block. Documents allow for flexible schemas. For example, a document about a book may have key-value pairs listing author, title, and date of publication. Some books may also have information about companion websites and translations into other languages. The set of keys that may be included does not have to be defined prior to use in document databases. This is especially helpful when applications must accommodate a range of attributes, some of which may not be known at design time.

Cloud Firestore is accessed via a REST API that can be used from applications running in Compute Engine, Kubernetes Engine, or App Engine. This database will scale automatically based on load. It will also *shard*, or partition, data as needed to maintain performance. Since Cloud Firestore is a managed service, it takes care of replication, backups, and other database administration tasks.

Although it is a NoSQL database, Cloud Firestore supports transactions, indexes, and SQL-like queries.

Cloud Firestore is well suited to applications that demand high scalability and structured data and do not always need strong consistency when reading data. Product catalogs, user profiles, and user navigation history are examples of the kinds of applications that use Cloud Datastore.

Cloud Memorystore

Cloud Memorystore is an in-memory cache service. Other databases offered in Google Cloud are designed to store large volumes of data and support complex queries, but Cloud Memorystore is a managed service for caching frequently used data in memory. Caches like this are used to reduce the time needed to read data into an application. Cloud Memorystore is designed to provide submillisecond access to data. Cloud Memorystore supports both Redis and memcached, two popular open source caching systems.

As a managed service, Cloud Memorystore allows users to specify the size of a cache while leaving administration tasks to Google. Google Cloud ensures high availability, patching, and automatic failover so users don't have to.

Chapter 12, “Deploying Storage in Google Cloud,” delves into details of how to create various types of databases, as well as how to load, delete, and query data. Each of the databases can be accessed from the main menu of the Google Cloud Console. From there you can begin to explore how each works and begin to see the differences.

Networking Components of Google Cloud

In this section, we will review the major networking components. Details on setting up networks and managing them are described in Chapter 14, “Networking in the Cloud: Virtual Private Clouds and Virtual Private Networks,” and Chapter 15, “Networking in the Cloud: DNS, Load Balancing, Google Private Access, and IP Addressing.”

Networking Services

Google Cloud provides a number of networking services designed to allow users to configure virtual networks within Google's global network infrastructure, link on-premises data centers to Google's network, optimize content delivery, and protect your cloud resources using network security services.

Virtual Private Cloud

When an enterprise operates its own data center, it controls what is physically located in that data center and connected to its network. Its infrastructure is physically isolated from those of other organizations running in other data centers. When an organization moves to a public cloud, it shares infrastructure with other customers of that public cloud. Although multiple enterprises will use the same cloud infrastructure, each enterprise can logically isolate its cloud resources by creating a virtual private cloud (VPC).

A distinguishing feature of Google Cloud is that a VPC can span the globe without relying on the public Internet. Traffic from any server on a VPC can be securely routed through the Google global network to any other point on that network. Another advantage of the Google network structure is that your back-end servers can access Google services, such as machine learning or Internet of Things (IoT) services, without creating a public IP address for back-end servers.

VPCs in Google Cloud can be linked to on-premises virtual private networks using Internet Protocol Security (IPSec).

Although a VPC is global, enterprises can use separate projects and billing accounts to manage different departments or groups within the organization. Firewalls can be used to restrict access to resources on a VPC as well.

Cloud Load Balancing

Google provides global load balancing to distribute workloads across your cloud infrastructure. Using a single cast IP address, Cloud Load Balancing can distribute the workload within and across regions, adapt to failed or degraded servers, and autoscale your compute resources to accommodate changes in workload. Cloud Load Balancing also supports internal load balancing, so no IP addresses need to be exposed to the Internet to get the advantages of load balancing.

Cloud Load Balancing is a software service that can load-balance HTTP, HTTPS, TCP/SSL, and UDP traffic.

Cloud Armor

Services exposed to the Internet can become targets of distributed denial-of-service (DDoS) attacks. Cloud Armor is a Google network security service that builds on the Global HTTP(s) Load Balancing service. Cloud Armor features include the following:

- Ability to allow or restrict access based on IP address
- Predefined rules to counter cross-site scripting attacks
- Ability to counter SQL injection attacks
- Ability to define rules from level 3 (network) to level 7 (application)
- Allows and restricts access based on the geolocation of incoming traffic

Cloud CDN

With content delivery networks (CDNs), users anywhere can request content from systems distributed in various regions. CDNs enable low-latency response to these requests by caching content on a set of endpoints across the globe. Google currently has more than 100 CDN endpoints that are managed as a global resource, so there is no need to maintain region-specific configurations.

CDNs are especially important for sites with large amounts of static content and a global audience. News sites, for example, could use the Cloud CDN service to ensure fast response to requests from any point in the world.

Cloud Interconnect

Cloud Interconnect is a set of Google Cloud services for connecting your existing networks to the Google network. Cloud Interconnect offers two types of connections: interconnects and peering.

Interconnect with direct access to networks uses the Address Allocation for Private Internets standard (RFC 1918) to connect to devices in your VPC. A direct network connection is maintained between an on-premises or hosted data center and one of Google's colocation facilities, which are in North America, South America, Europe, Asia, and Australia. Alternatively, if an organization cannot achieve a direct interconnect with a Google facility, it could use Partner Interconnect. This service depends on a third-party network provider to provide connectivity between the company's data center and a Google facility.

Partner Interconnect is the recommended way to connect to Google Cloud through providers, but if you need to access Google Workspace applications, then you can use carrier peering. Peering does not use Google Cloud resources such as interconnect connections or Cloud Routers.

For organizations that do not require the bandwidth of a direct or peered interconnect, Google offers VPN services that enable traffic to transmit between data centers, other vendor clouds, and Google Cloud using the public Internet.

Cloud DNS

Cloud DNS is a domain name service provided in Google Cloud. Cloud DNS is a high availability, low-latency service for mapping from domain names, such as `example.com`, to IP addresses, such as `74.120.28.18`.

Cloud DNS is designed to automatically scale so customers can have thousands or even millions of addresses without concern for scaling the underlying infrastructure. Cloud DNS also provides for private zones that allow you to create custom names for your VMs if you need those.

Identity Management and Security

Google Cloud Identity and Access Management (IAM) service enables customers to define fine-grained access controls on resources in the cloud. IAM uses the concepts of users, roles, and permissions.

Identities are abstractions about users of services, such as a human user. After an identity is authenticated by logging in or some other mechanism, the authenticated user can access resources and perform operations based on the permissions granted to that identity. For example, a user may have permissions to create a bucket in Cloud Storage or delete a VM running in Compute Engine.

Users often need similar sets of permissions. Someone who can create a VM will likely want to be able to modify or delete those VMs. Groups of related permissions can be bundled into roles. Roles are sets of permissions that can be assigned to an identity.

As a Google Certified Associate Cloud Engineer, you will become familiar with identities, roles, and permissions and how to administer them across organizations and projects.

You can find identity management tools under the IAM and Admin menu in the Google Cloud Console. Chapter 17, “Configuring Access and Security,” provides details on identity, roles, and best practices for their management.

Development Tools

Google Cloud is an excellent choice for developers and software engineers because of the easy access to infrastructure and data management services, but also for the tools it supports.

Cloud SDK is a command-line interface for managing Google Cloud resources, including VMs, disk storage, network firewalls, and virtually any other resource you might deploy

in Google Cloud. In addition to a command-line interface, Cloud SDK client libraries include libraries for Java, Python, Node.js, Ruby, Go, .NET, and PHP.

Google Cloud also supports deploying applications to containers with Container Registry, Cloud Build, and Cloud Source Repositories.

Google has also developed plug-ins to make it easy to work with popular development tools. These include the following:

- Cloud Tools for IntelliJ
- Cloud Tools for PowerShell
- Cloud Tools for Visual Studio
- Cloud Tools for Eclipse
- App Engine Gradle Plugin
- App Engine Maven Plugin

Of course, applications move from development to production deployment, and Google Cloud follows that flow with additional management tools to help monitor and maintain applications after they are deployed.

Additional Components of Google Cloud

Management tools are designed for those who are responsible for ensuring the reliability, availability, and scalability of applications.

Management and Observability Tools

The following are some of the most important tools in the management and observability tools category:

Cloud Monitoring This service collects performance data from Google Cloud, AWS resources, and application instrumentation, including popular open source systems like NGINX, Cassandra, and Elasticsearch.

Cloud Logging This service enables users to store and analyze and alert on log data from both Google Cloud and Amazon Web Services (AWS) logs.

Error Reporting This aggregates application crash information for display in a centralized interface.

Cloud Trace This is a distributed tracing service that captures latency data about an application to help identify performance problem areas.

Cloud Debugger This enables developers to inspect the state of executing code, inject commands, and view call stack variables.

Cloud Profiler This is used to collect CPU and memory utilization information across the call hierarchy of an application. Profiler uses statistical sampling to minimize the impact of profiling on application performance.

The combination of management and observability tools provides insights into applications as they run in production, enabling more effective monitoring and analysis of operational systems.

Specialized Services

In addition to IaaS and PaaS offerings, Google Cloud has specialized services for APIs, data analytics, and machine learning.

Apigee API Platform

The Apigee API platform is a management service for Google Cloud customers providing API access to their applications. The Apigee platform allows developers to deploy, monitor, and secure their APIs. It also generates API proxies based on the Open API Specification.

It is difficult to predict load on an API, and sometimes spikes in use can occur. For those times, the Apigee API platform provides routing and rate-limiting based on policies customers can define.

APIs can be authenticated using either OAuth 2.0 or SAML. Data is encrypted both in transit and at rest in the Apigee API platform.

Data Analytics and Data Pipelines

Google Cloud has a number of services designed for analyzing big data in batch and streaming modes. Some of the most important tools in this set of services are:

- BigQuery, a petabyte-scale analytics database service for data warehousing
- Cloud Dataflow, a framework for defining batch and stream processing pipelines
- Cloud Dataproc, a managed Hadoop and Spark service
- Cloud Dataprep, a service that allows analysts to explore and prepare data for analysis

Often, data analytics and data warehousing projects use several of these services together.

AI and Machine Learning

Google is a leader in AI and machine learning, so it is no surprise that Google Cloud includes several AI services. Vertex AI is a unified AI platform for building machine learning models. Specialized services in this area include the following:

AutoML This is a tool that allows developers without machine learning experience to develop machine learning models.

Translation AI This tool is for translating human language and includes AutoML Translation and Translation API for text translations and Media Translation API for audio translations.

Natural Language Analyze and extract features and concepts from text using machine learning methods.

Vision AI This is an image analysis platform for annotating images with metadata, extracting text, or filtering content.

Recommendations AI This is a service to provide personalized recommendations to customers at scale.

Summary

Google Cloud provides a full range of services to support information processing including computing resources, storage resources, databases, networking services, identity management and security services, development tools, management and operations services, as well as specialized services to support AI.

Exam Essentials

Understand the differences between Compute Engine, Kubernetes Engine, App Engine, Cloud Run, and Cloud Functions. Compute Engine is Google's VM service. Users can choose CPUs, memory, persistent disks, and operating systems. They can further customize a VM by adding graphics processing units for compute-intensive operations. VMs are managed individually or in groups of similar servers.

Kubernetes Engine manages groups of virtual servers and applications that run in containers. Containers are lighter weight than VMs. Kubernetes is called an *orchestration service* because it distributes containers across clusters, monitors cluster health, and scales as prescribed by configurations.

App Engine is Google's PaaS. Developers can run their code in a language-specific sandbox when using the standard environment or in a container when using the flexible environment. App Engine is a serverless service, so customers do not need to specify VM configurations or manage servers.

Cloud Run is a service for running stateless containers. This is a serverless option that provides some of the advantages of Kubernetes without requiring you to deploy your own clusters. Note that Cloud Run does not currently support applications that maintain state in the container.

Cloud Functions is a serverless service that is designed to execute short-running code that responds to events, such as file uploads or messages being published to a message queue. Functions may be written in Node.js or Python.

Understand what is meant by serverless. Serverless means customers using a service do not need to configure, monitor, or maintain the computing resources underlying the service. It does not mean there are no servers involved—there are always physical servers that run applications, functions, and other software. Serverless only refers to not needing to manage those underlying resources.

Understand the difference between object and file storage. Object stores are used to store and access file-based resources. These objects are referenced by a unique identifier, such as a URL. Object stores do not provide block or filesystem services, so they are not suitable for database storage. Cloud Storage is Google Cloud object storage service.

File storage supports block-based access to files. Files are organized into directories and sub-directories. Google's Filestore is based on NFS.

Know the different kinds of databases. Databases are broadly divided into relational and NoSQL databases.

Relational databases support transactions, strong consistency, and the SQL query languages. Relational databases have been traditionally difficult to horizontally scale. Cloud Spanner is a global relational database that provides the advantages of relational databases with the scalability previously found only in NoSQL databases.

NoSQL databases are designed to be horizontally scalable. Other features, such as strong consistency and support for standard SQL, are often sacrificed to achieve scalability and low-latency query responses. NoSQL databases may be key-value stores like Cloud Memorystore, document databases like Cloud Firestore, or wide-column databases such as Cloud Bigtable.

Understand virtual private clouds. A VPC is a logical isolation of an organization's cloud resources within a public cloud. In Google Cloud, VPCs are global; they are not restricted to a single zone or region. All traffic between Google Cloud services can be transmitted over the Google network without the need to send traffic over the public Internet.

Understand load balancing. Load balancing is the process of distributing a workload across a group of servers. Load balancers can route workload based on network-level or application-level rules. Google Cloud load balancers can distribute workloads globally.

Understand developer and management tools. Developer tools support common workflows in software engineering, including using version control for software, building containers to run applications and services, and making containers available to other developers and orchestration systems, such as Kubernetes Engine.

Management tools, such as Cloud Monitoring and Cloud Logging, are designed to provide systems administration information to developers and operators who are responsible for ensuring applications are available and operating as expected.

Know the types of specialized services offered by Google Cloud. Google Cloud includes a growing list of specialized services for data analytics as well as AI and machine learning.

Know the main differences between on-premises and public cloud computing. On-premises computing is computing, storage, networking, and related services that occur on infrastructure managed by a company or organization for its own use. Hardware may be located literally on the premises in a company building or in a third-party colocation facility. Colocation facilities provide power, cooling, and physical security, but the customers of the colocation facility are responsible for all the setup and management of the infrastructure.

Public cloud computing uses infrastructure and services provided by a cloud provider such as Google, AWS, or Microsoft. The cloud provider maintains all physical hardware and facilities. It provides a mix of services, such as VMs that are configured and maintained by customers and serverless offerings that enable customers to focus on application development, while the cloud provider takes on more responsibility for maintaining the underlying compute infrastructure.

Review Questions

You can find the answers in the Appendix.

1. You are planning to deploy an SaaS application for customers in North America, Europe, and Asia. To maintain scalability, you will need to distribute workload across servers in multiple regions. Which Google Cloud service would you use to implement the workload distribution?
 - A. Cloud DNS
 - B. Cloud Spanner
 - C. Cloud Load Balancing
 - D. Cloud CDN
2. You have decided to deploy a set of microservices using containers. The microservices will maintain state in the container. You could install and manage Docker on Compute Engine instances, but you'd rather have Google Cloud provide some container management services. Which are two Google Cloud services that allow you to run containers in a managed service?
 - A. App Engine standard environment and App Engine flexible environment
 - B. Kubernetes Engine and App Engine standard environment
 - C. Kubernetes Engine and Cloud Run environment
 - D. App Engine standard environment and Cloud Functions
3. Why would an API developer want to use the Apigee API platform?
 - A. To get the benefits of routing and rate-limiting
 - B. Authentication services
 - C. Version control of code
 - D. A and B
 - E. All of the above
4. You are deploying an API to the public Internet and are concerned that your service will be subject to DDoS attacks. Which Google Cloud service should you consider to protect your API?
 - A. Cloud Armor
 - B. Cloud CDN
 - C. Cloud IAM
 - D. VPCs
5. You have an application that uses a Pub/Sub message queue to maintain a list of tasks that are to be processed by another application. The application that consumes messages from the Pub/Sub queue removes the message only after completing the task. It takes approximately 10 seconds to complete a task. It is not a problem if two or more VMs perform the same task. What is a cost-effective configuration for processing this workload?
 - A. Use preemptible VMs
 - B. Use standard VMs
 - C. Use DataProc
 - D. Use Spanner

6. Your department is deploying an application that has a database back end. You are concerned about the read load on the database server and want to have data available in memory to reduce the time to respond to queries and to reduce the load on the database server. Which Google Cloud service would you use to keep data in memory?
 - A. Cloud SQL
 - B. Cloud Memorystore
 - C. Cloud Spanner
 - D. Cloud Firestore
7. The Cloud SDK can be used to configure and manage resources in which of the following services?
 - A. Compute Engine
 - B. Cloud Storage
 - C. Network firewalls
 - D. All of the above
8. What server configuration is required to use Cloud Functions?
 - A. VM configuration
 - B. Cluster configuration
 - C. Pub/Sub configuration
 - D. None
9. You have been assigned the task of consolidating log data generated by each instance of an application. Which management and observability tools would you use?
 - A. Cloud Monitoring
 - B. Cloud Trace
 - C. Cloud Debugger
 - D. Cloud Logging
10. Which specialized services are most likely to be used to build a data warehousing platform that requires complex extraction, transformation, and loading operations on batch data as well as processing streaming data?
 - A. Apigee API platform
 - B. Data analytics
 - C. AI and machine learning
 - D. Cloud SDK

11. Your company has deployed 100,000 Internet of Things (IoT) sensors to collect data on the state of equipment in several factories. Each sensor will collect and send data to a data store every 5 seconds. Sensors will run continuously. Daily reports will produce data on the maximum, minimum, and average values for each metric collected on each sensor. There is no need to support transactions in this application. Which database product would you recommend?
 - A. Cloud Spanner
 - B. Cloud Bigtable
 - C. Cloud SQL MySQL
 - D. Cloud SQL PostgreSQL
12. You are the lead developer on a medical application that uses patients' smartphones to capture biometric data. The app is required to collect data and store it on the smartphone when data cannot be reliably transmitted to the back-end application. You want to minimize the amount of development you have to do to keep data synchronized between smartphones and back-end data stores. Which data store option should you recommend?
 - A. Cloud Firestore
 - B. Cloud Spanner
 - C. Cloud CDN
 - D. Cloud SQL
13. A software engineer comes to you for a recommendation. They have implemented a machine learning algorithm to identify cancerous cells in medical images. The algorithm is computationally intensive, makes many floating-point calculations, requires immediate access to large amounts of data, and cannot be easily distributed over multiple servers. What kind of Compute Engine configuration would you recommend?
 - A. High memory, high CPU
 - B. High memory, high CPU, GPU
 - C. Mid-level memory, high CPU
 - D. High CPU, GPU
14. You are tasked with mapping the authentication and authorization policies of your on-premises applications to Google Cloud's authentication and authorization mechanisms. The Google Cloud documentation states that an identity must be authenticated in order to grant permissions to that identity. What does the term *identity* refer to?
 - A. VM ID
 - B. User
 - C. Role
 - D. Set of privileges

15. A client is developing an application that will need to analyze large volumes of text information. The client is not expert in text mining or working with language. What Google Cloud service would you recommend they use?
- A. Vertex AI
 - B. Recommendation AI
 - C. Natural Language
 - D. Text-to-Speech
16. Data scientists in your company want to use a machine learning library available only in Apache Spark. They want to minimize the amount of administration and DevOps work. How would you recommend they proceed?
- A. Use Cloud Spark.
 - B. Use Cloud Dataproc.
 - C. Use BigQuery.
 - D. Install Apache Spark on a cluster of VMs.
17. Database designers at your company are debating the best way to move a database to Google Cloud. The database supports an application with a global user base. Users expect support for transactions and the ability to query data using commonly used query tools. The database designers decide that any database service they choose will need to support ANSI SQL 2011 and global transactions. Which database service would you recommend?
- A. Cloud SQL
 - B. Cloud Spanner
 - C. Cloud Firestore
 - D. Cloud Bigtable
18. Which specialized service supports both batch and stream processing workflows?
- A. Cloud Dataflow
 - B. BigQuery
 - C. Cloud Firestore
 - D. AutoML
19. You have a Python application you'd like to run in a scalable environment with the least amount of management overhead. Which Google Cloud product would you select?
- A. App Engine flexible environment
 - B. Cloud Engine
 - C. App Engine standard environment
 - D. Kubernetes Engine

- 20.** A product manager at your company reports that customers are complaining about the reliability of one of your applications. The application is crashing periodically, but developers have not found a common pattern that triggers the crashes. They are concerned that they do not have good insight into the behavior of the application and want to perform a detailed review of all crash data. Which observability tool would you use to view consolidated crash information?
- A.** Cloud DataProc
 - B.** Cloud Monitoring
 - C.** Cloud Logging
 - D.** Error Reporting

Chapter 3

Projects, Service Accounts, and Billing

**THIS CHAPTER COVERS THE FOLLOWING
OBJECTIVES OF THE GOOGLE ASSOCIATE
CLOUD ENGINEER CERTIFICATION EXAM:**

- ✓ 1.1 Setting up cloud projects and accounts
- ✓ 1.2 Managing billing configuration





Before delving into computing, storage, and networking services, we need to discuss how Google Cloud organizes resources and links the use of those resources to a billing system. This chapter introduces the Google Cloud organizational hierarchy, which consists of organizations, folders, and projects. It also discusses service accounts, which are ways of assigning roles to compute resources so they can carry out functions on your behalf. Finally, the chapter briefly discusses billing.

How Google Cloud Organizes Projects and Accounts

When you use Google Cloud, you probably launch virtual machines or clusters, maybe create buckets to store objects, and make use of serverless computing services such as Cloud Run and Cloud Functions. The list of resources you use can grow quickly and can also change in dynamic, unpredictable ways as autoscaling services respond to workload.

If you run a single application or a few services for your department, you might be able to track all resources by viewing lists of resources in use. As the scope of your Google Cloud use grows, you will probably have multiple departments, each with its own administrators who need different privileges. Google Cloud provides a way to group resources and manage them as a single unit. This is called the *resource hierarchy*. The access to resources in the resource hierarchy is controlled by a set of policies that you can define.

Google Cloud Resource Hierarchy

The central abstraction for managing Google Cloud resources is the resource hierarchy. It consists of three levels:

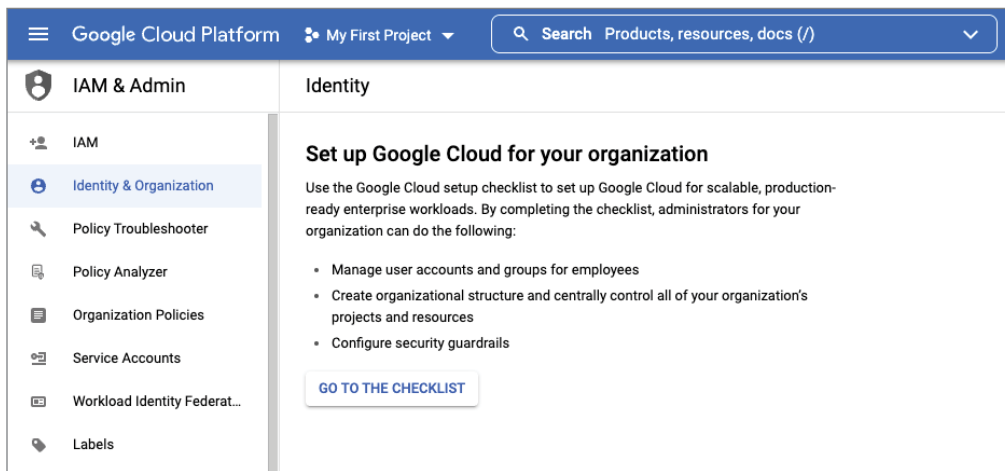
- Organization
- Folder
- Project

Let's describe how these three components relate to each other.

Organization

An organization is the root of the resource hierarchy and typically corresponds to a company or organization. Google Workspace domains and Cloud Identity accounts map to Google Cloud organizations. Google Workspace is Google's office productivity suite, which includes Gmail, Docs, Drive, Calendar, and other services. If your company uses Google Workspace, you can create an organization in your Google Cloud hierarchy. If your company does not use Google Workspace, you can use Cloud Identity, Google's identity as a service (IDaaS) offering (Figure 3.1).

FIGURE 3.1 You can create Cloud Identity accounts and manage Google Workspace users from the Identity & Organization console.



A single cloud identity is associated with at most one organization. Cloud identities have super admins, and those super admins assign the role of Organization Administrator Identity and Access Management (IAM) to users who manage the organization. In addition, Google Cloud will automatically grant Project Creator and Billing Account Creator IAM roles to all users in the domain. This allows any user to create projects and enable billing for the cost of resources used.

The users with the Organization Administrator IAM role are responsible for the following:

- Defining the structure of the resource hierarchy
- Defining identity and access management policies over the resource hierarchy
- Delegating other management roles to other users

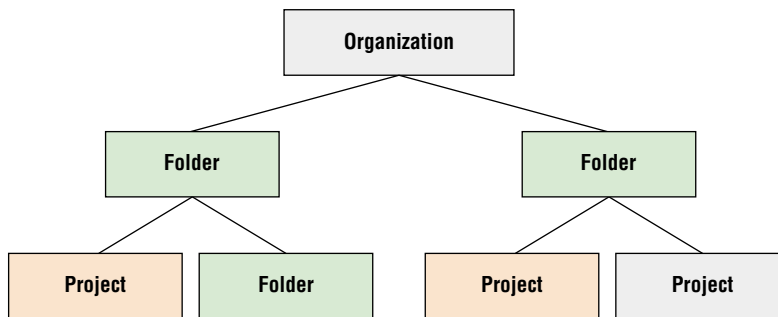
When a member of a Google Workspace organization/Cloud Identity account creates a billing account or project, Google Cloud will automatically create an organization resource.

All projects and billing accounts will be children of the organization resource. In addition, when the organization is created, all users in that organization are granted Project Creator and Billing Account Creator roles. From that point on, Google Workspace users will have access to Google Cloud resources.

Folder

Folders are the building blocks of multilayer organizational hierarchies. Organizations contain folders. Folders can contain other folders or projects. Folders, however, are optional and do not have to be used. A single folder may contain both folders and projects (see Figure 3.2). Folder organization is usually built around the kinds of services provided by resources in the contained projects and the policies governing folders and projects.

FIGURE 3.2 Generic organization folder project

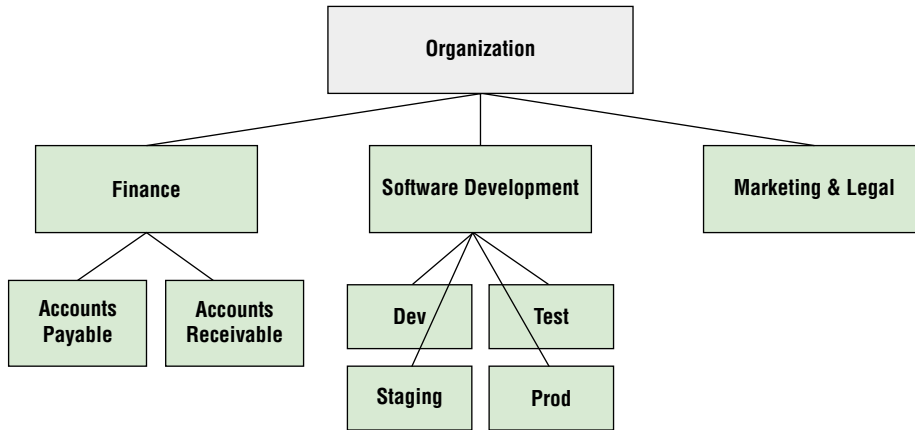


Consider an example resource hierarchy. An organization has four departments: finance, marketing, software development, and legal. The finance department has to keep its accounts receivable and accounts payable resources separate, so the administrator creates two folders within the Finance folder: Accounts Receivable and Accounts Payable. Software development uses multiple environments, including Dev, Test, Staging, and Production. Access to each of the environments is controlled by policies specific to that environment, so it makes sense to organize each environment into its own folder. Marketing and legal can have all their resources shared across members of the department, so a single folder is sufficient for both of those departments. Figure 3.3 shows the organization hierarchy for this organization.

Now that we have an organization defined and have set up folders that correspond to our departments and how different groups of resources will be accessed, we can create projects.

Project

Projects are in some ways the most important part of the hierarchy. It is in projects that we create resources, use Google Cloud services, manage permissions, and manage billing options.

FIGURE 3.3 Example organization folder project

The first step in working with a project is to create one. Anyone with the `resourcemanager.projects.create` IAM permission can create a project. By default, when an organization is created, every user in the domain is granted that permission.

Your organization will have a quota of projects it can create. The quota can vary between organizations. Google makes decisions about project quotas based on typical use, the customer's usage history, and other factors. If you reach your limit of projects and try to create another, you will be prompted to request an increase in the quota. You'll have to provide information such as the number of additional projects you need and what they will be used for.

After you have created your resource hierarchy, you can define policies that govern it.

Organization Policies

Google Cloud provides an Organization Policy Service. This service controls access to an organization's resources. The Organization Policy Service complements the IAM service.

IAM lets you assign permissions so that users or roles can perform specific operations in the cloud. The Organization Policy Service lets you specify limits on the ways resources can be used. One way to think of the difference is that IAM specifies who can do things, and the Organization Policy Service specifies what can be done with resources.

The organization policies are defined in terms of constraints on a resource.

Constraints on Resources

Constraints are restrictions on services. Google Cloud has list constraints and Boolean constraints.

List constraints are lists of values that are allowed or denied for a resource. The following are some types of list constraints:

- Allow a specific set of values.
- Deny a specific set of values.
- Deny a value and all its child values.
- Allow all allowed values.
- Deny all values.

Boolean constraints evaluate to true or false and determine whether or not the constraint is applied. For example, if you want to deny access to serial ports on VMs, you can set `constraints/compute.disableSerialPortAccess` to `TRUE`.

See organization policy constraints documentation at <https://cloud.google.com/resource-manager/docs/organization-policy/org-policy-constraints> for more details.

Policy Evaluation

Organizations may have standing policies to protect data and resources in the cloud. For example, there may be rules dictating who in the organization can enable a service API or create a service account. Your InfoSec department may require that all VMs disable serial port access. You could implement controls on each individual VM, but that is inefficient and prone to error. A better approach is to define a policy that constrains what can be done and attach that policy to an object in the resource hierarchy.

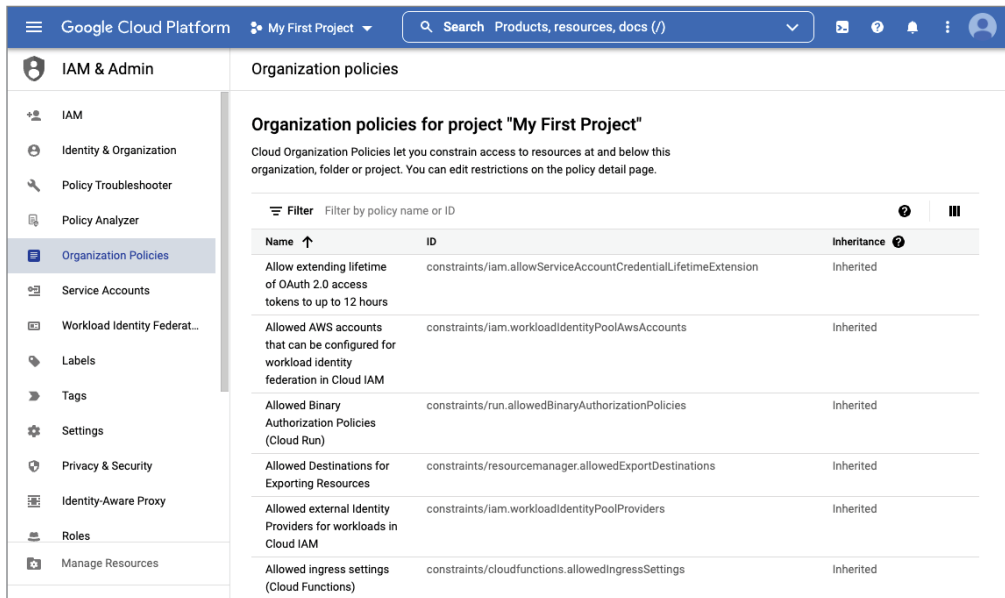
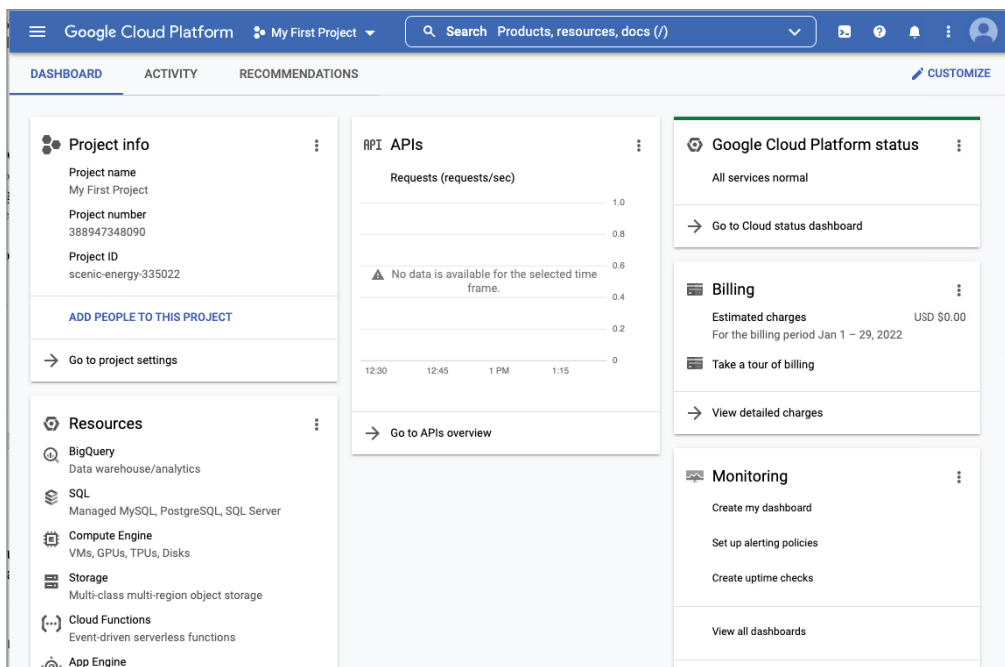
For example, since InfoSec wants all VMs to disable serial port access, you could specify a policy that constrains serial port access and then attach it to the organization. All folders and projects below the organization will inherit that policy. Since policies are inherited and cannot be disabled or overridden by objects lower in the hierarchy, this is an effective way to apply a policy across all organizational resources. There is, however, a way to disable inheriting from parents by setting the `inheritFromParent` parameter to `false`.

Policies are managed through the Organization Policies form in the IAM & Admin console. Figure 3.4 shows an example set of policies.

Multiple policies can be in effect for a folder or project. For example, if the organization had a policy on serial port access and a folder containing a project had a policy limiting who can create service accounts, then the project would inherit both policies and both would constrain what could be done with resources in that project.

Managing Projects

One of the first tasks you will perform when starting a new cloud initiative is to set up a project. This can be done with the Google Cloud Console. Assuming you have created an account with Google Cloud, navigate to the Google Cloud Console at <https://console.cloud.google.com> and log in. You will see the home page, which looks something like Figure 3.5.

FIGURE 3.4 Organizational policies are managed in the IAM & Admin console.**FIGURE 3.5** Home page console

From the Navigation menu in the upper-left corner, select IAM & Admin and then select Manage Resources (see Figure 3.6 and Figure 3.7).

FIGURE 3.6 Navigation menu

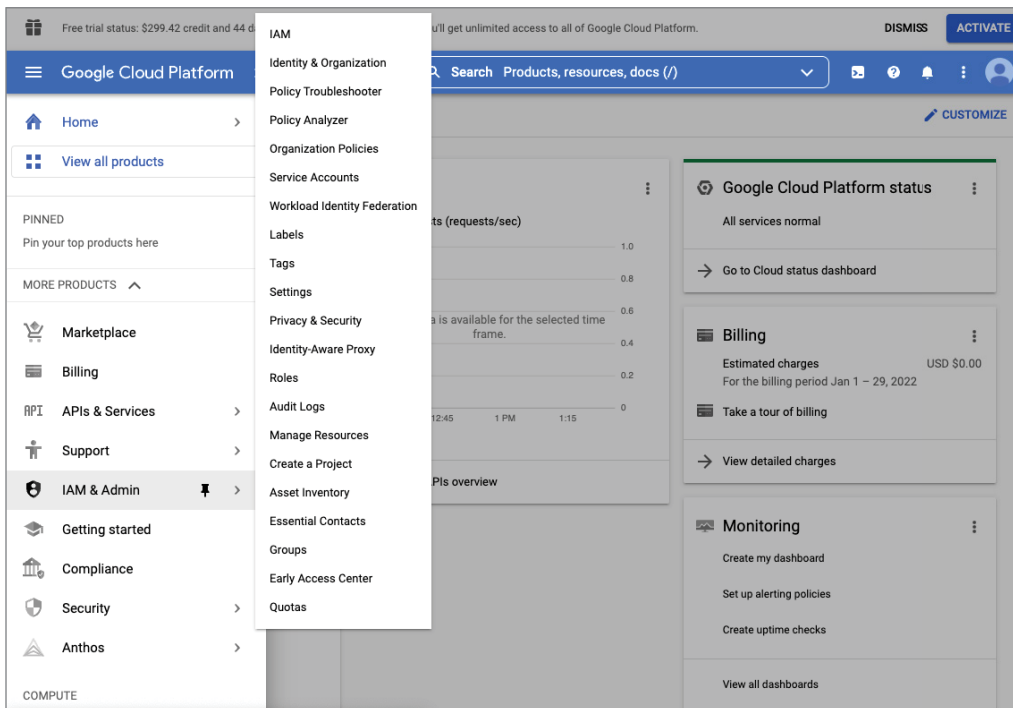
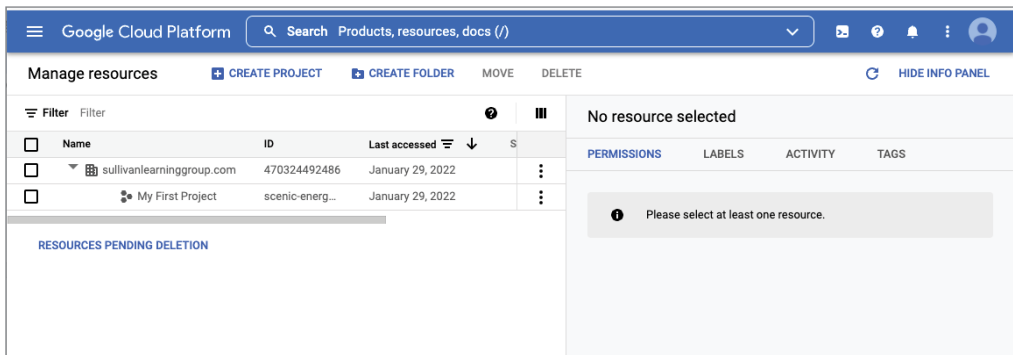


FIGURE 3.7 Managing resources



From there, you can click Create Project, which displays the New Project dialog box. Here, you can enter the name of a project and select an organization (Figure 3.8 and Figure 3.9).

FIGURE 3.8 Click Create Project.

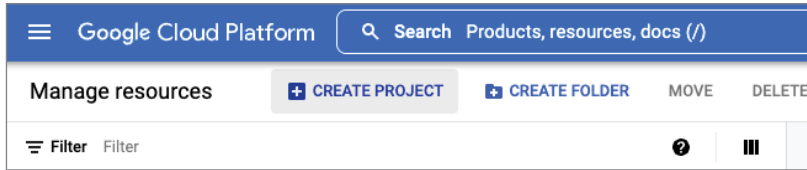


FIGURE 3.9 New Project dialog box

 A screenshot of the 'New Project' dialog box in Google Cloud Platform. The dialog has a title bar 'New Project'. It contains three main input fields: 'Project name *' with the text 'My Project 3502', 'Organization *' with the dropdown value 'sullivanlearninggroup.com', and 'Location *' with the dropdown value 'sullivanlearninggroup.com'. Below the 'Project name' field, it shows 'Project ID: silken-buttress-339721. It cannot be changed later. EDIT'. Below the 'Location' field, it says 'Parent organization or folder'. At the bottom are two buttons: 'CREATE' and 'CANCEL'.

Note that when you create a project, your remaining quota of projects is displayed. If you need additional projects, click the Manage Quotas link to request an increase in your quota.

Roles and Identities

In addition to managing resources, as a cloud engineer you will have to manage access to those resources. This is done with the use of roles and identities.

Roles in Google Cloud

A *role* is a collection of permissions. Roles are granted to users by binding a user to a role. When we talk of identities, we mean the object we use to represent a human user or service account in Google Cloud. For example, Alice is a software engineer developing applications in the cloud (the human user), and she has an identity with the name `alice@example.com`. Roles are assigned to `alice@example.com` within Google Cloud so that Alice can create, modify, delete, and use resources in Google Cloud.

There are three types of roles in Google Cloud:

- Basic roles
- Predefined roles
- Custom roles

Basic roles, formerly known as primitive roles, include Owner, Editor, and Viewer. These provide broad privileges that can be applied to most resources. It is a best practice to use predefined roles instead of basic roles when possible. Basic roles grant wide ranges of permissions that may not always be needed by a user. By using predefined roles, you can grant only the permissions a user needs to perform their function. This practice of only assigning permissions that are needed and no more is known as the *principle of least privilege*. It is one of the fundamental best practices in information security.

Predefined roles provide granular access to resources in Google Cloud, and they are specific to Google Cloud products and managed and updated by Google. (See Figure 3.10.) For example, App Engine roles include the following:

- `appengine.appAdmin`, which grants identities the ability to read, write, and modify all application settings in App Engine
- `appengine.ServiceAdmin`, which grants read-only access to application settings and write-level access to module-level and version-level settings in App Engine
- `appengine.appViewer`, which grants read-only access to applications in App Engine

Custom roles allow cloud administrators to create and administer their own roles. Custom roles are assembled using permissions defined in IAM. While you can use most permissions in a custom role, some, such as `iam.ServiceAccounts.getAccessToken`, are not available in custom roles.

Granting Roles to Identities

Once you have determined which roles you want to provide to users, you can assign roles to users through the IAM console. It is important to know that permissions cannot be assigned to users—they can be assigned only to roles. Roles are then assigned to users.

From the IAM console, you can select a project that will display a permission interface, such as in Figure 3.11.

From there, select the Add option to display another dialog box that prompts for usernames and roles (see Figure 3.12).

FIGURE 3.10 A sample list of roles in Google Cloud

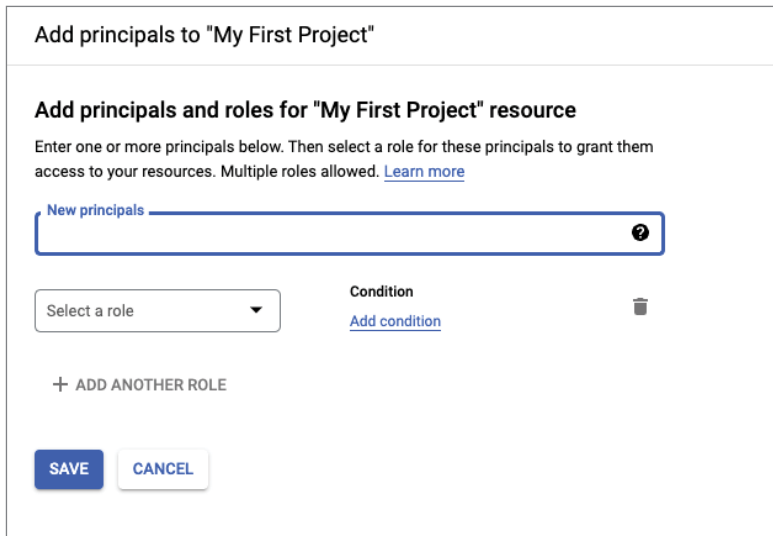
The screenshot shows the Google Cloud Platform interface for the 'My First Project'. The left sidebar is under the 'IAM & Admin' section, with 'Roles' selected. The main content area is titled 'Roles for "My First Project" project'. It includes a description: 'A role is a group of permissions that you can assign to principals. You can create a role and add permissions to it, or copy an existing role and adjust its permissions. [Learn more](#)'. Below this is a filter bar and a table of roles.

Type	Title	Used in	Status	
<input type="checkbox"/>	AAM Admin	Dialogflow	Enabled	⋮
<input type="checkbox"/>	AAM Conversational Architect	Dialogflow	Enabled	⋮
<input type="checkbox"/>	AAM Dialog Designer	Dialogflow	Enabled	⋮
<input type="checkbox"/>	AAM Lead Dialog Designer	Dialogflow	Enabled	⋮
<input type="checkbox"/>	AAM Viewer	Dialogflow	Enabled	⋮
<input type="checkbox"/>	Access Approval Approver	Access Approval	Enabled	⋮
<input type="checkbox"/>	Access Approval Config Editor	Access Approval	Enabled	⋮
<input type="checkbox"/>	Access Approval Viewer	Access Approval	Enabled	⋮
<input type="checkbox"/>	Access Context Manager Admin	Access Context Manager	Enabled	⋮
<input type="checkbox"/>	Access Context Manager Editor	Access Context Manager	Enabled	⋮
<input type="checkbox"/>	Access Context Manager Reader	Access Context Manager	Enabled	⋮
<input type="checkbox"/>	Access Transparency Admin	Organization Policy	Enabled	⋮
<input type="checkbox"/>	Actions Admin	Actions	Enabled	⋮
<input type="checkbox"/>	Actions Viewer	Actions	Enabled	⋮
<input type="checkbox"/>	Activity Analysis Viewer	Other	Enabled	⋮

FIGURE 3.11 IAM permissions

The screenshot shows the Google Cloud Platform interface for the 'My First Project'. The left sidebar is under the 'IAM & Admin' section, with 'IAM' selected. The main content area is titled 'Permissions for project "My First Project"'. It includes a description: 'These permissions affect this project and all of its resources. [Learn more](#)'. Below this is a 'View By' section with 'PRINCIPALS' selected and 'Include Google-provided role grants' checked. There is a filter bar and a table of principals.

Type	Principal	Name	Role	
<input type="checkbox"/>	388947348090-compute@developer.gserviceaccount.com	Compute Engine default service account	Cloud Data Fusion Runner Editor	1/1 x
				4671/4671 x

FIGURE 3.12 Adding a user

The screenshot shows a dialog box titled "Add principals to 'My First Project'". Inside, there's a section "Add principals and roles for 'My First Project' resource" with instructions: "Enter one or more principals below. Then select a role for these principals to grant them access to your resources. Multiple roles allowed. [Learn more](#)". Below this is a text input field labeled "New principals" with a question mark icon. Underneath the input field is a "Select a role" dropdown menu. To the right of the dropdown is a "Condition" section with an "Add condition" link and a trash icon. At the bottom left is a "+ ADD ANOTHER ROLE" link. At the bottom are "SAVE" and "CANCEL" buttons.

Service Accounts

Identities are usually associated with individual users. Sometimes it is helpful to have applications or VMs act on behalf of a user or perform operations that the user does not have permission to perform.

For example, you may have an application that needs to access a database but you do not want to allow users of the application to access the database directly. Instead, all user requests to the database should go through the application. You can create a service account that has access to the database. You can then assign that service account to the application so that the application can execute queries on behalf of users without having to grant database access to those users.

Service accounts are somewhat unusual in that we sometimes treat them as resources and sometime as identities. When we assign a role to a service account, we are treating it as an identity. When we give users permission to access a service account, we are treating it as a resource.

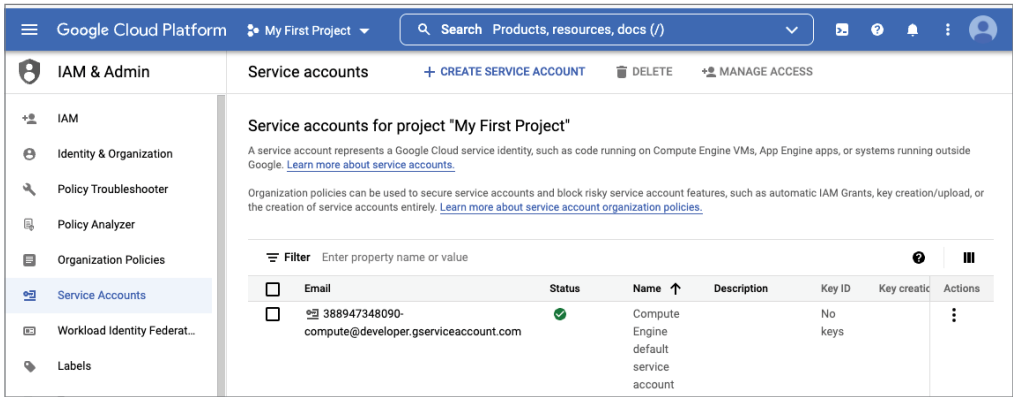
There are two types of service accounts: user-managed service accounts and Google-managed service accounts. Users can create up to 100 service accounts per project. When you create a project that has the Compute Engine API enabled, a Compute Engine service account is created automatically. Similarly, if you have an App Engine application in your project, Google Cloud will automatically create an App Engine service account. Both the Compute Engine and App Engine service accounts are granted editor roles on the projects in which they are created. You can also create custom service accounts in your projects.

Google may also create service accounts that it manages. These accounts are used with various Google Cloud services.

Service accounts can be managed as a group of accounts at the project level or at the individual service account level. For example, if you grant `iam.serviceAccountUser` to a user for a specific project, then that user can manage all service accounts in the project. If you prefer to limit users to manage only specific service accounts, you could grant `iam.serviceAccountUser` for a specific service account.

Service accounts are created automatically when resources are created. For example, a service account will be created for a VM when the VM is created. There may be situations in which you would like to create a service account for one of your applications. In that case, you can navigate to the IAM & Admin console and select Service Accounts. From there you can click Create Service Account at the top, as shown in Figure 3.13.

FIGURE 3.13 Service accounts’ listing in the IAM & Admin console



This brings up a form that prompts for the information needed to create a service account.

Billing

Using resources such as VMs, object storage, and specialized services usually incurs charges. The Google Cloud Billing API provides a way for you to manage how you pay for resources used.

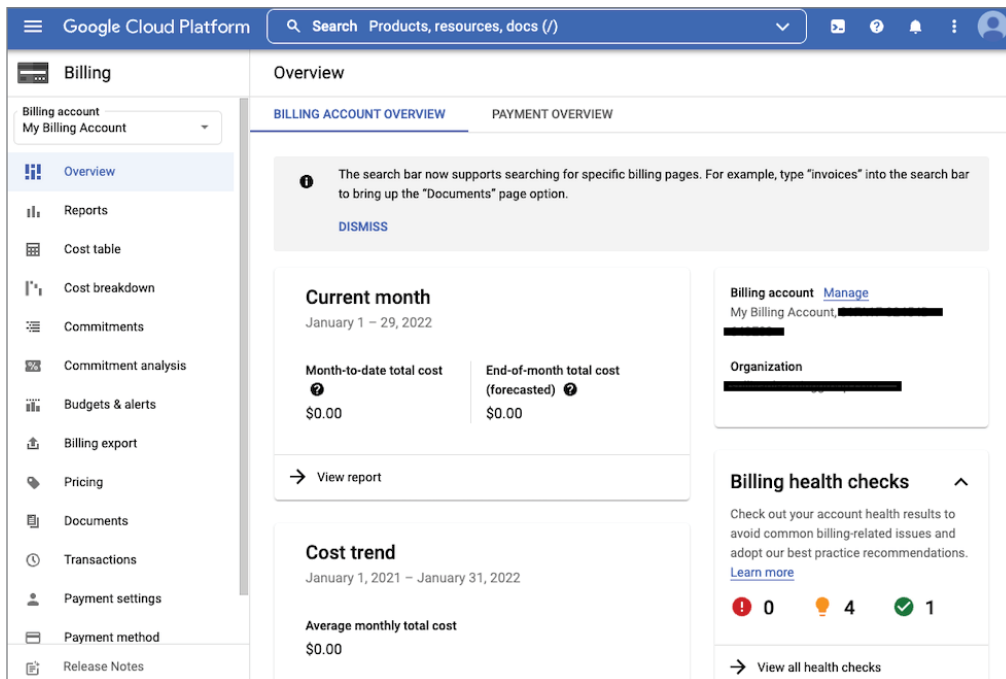
Billing Accounts

Billing accounts store information about how to pay charges for resources used. A billing account is associated with one or more projects. All projects must have a billing account unless they use only free services.

Billing accounts can follow a similar structure to the resource hierarchy. If you are working with a small company, you may have only a single billing account. In that case, all resource costs are charged to that one account. If your company is similar to the example from earlier in the chapter, with finance, marketing, legal, and software development departments, then you may want to have multiple billing accounts. You could have one billing account for each department, but that may not be necessary. If finance, marketing, and legal all pay for their cloud services from the same part of your company's budget, then they could use a single billing account. If software development services are paid from a different part of your company's budget, then it could use a different billing account.

From the main Google Cloud Console, you can navigate to the Billing console (see Figure 3.14), which lists existing billing accounts.

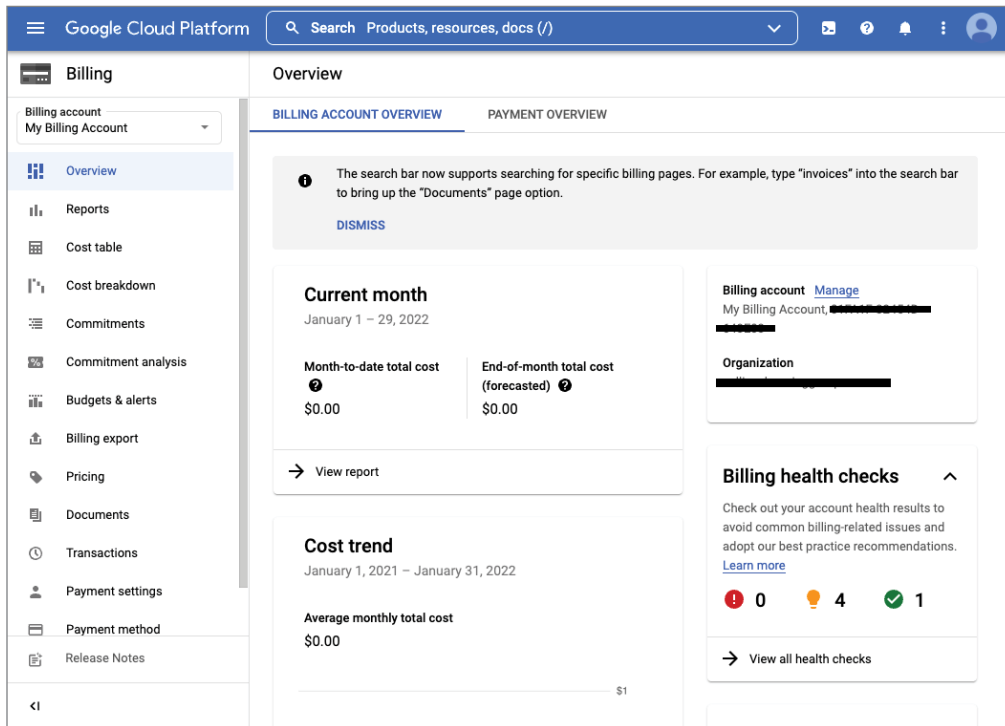
FIGURE 3.14 The main Billing form listing existing billing accounts



From here, you can create a new billing account, as shown in Figure 3.15.

From the Billing overview page, you can view and modify projects linked to billing accounts.

There are two types of billing accounts: self-serve and invoiced. Self-serve accounts are paid by credit card or direct debit from a bank account. The costs are charged automatically. The other type is an invoiced billing account, in which bills or invoices are sent to customers. This type of account is commonly used by enterprises and other large customers.

FIGURE 3.15 The form to create a new billing account

Several roles are associated with billing. It is important to know them for the exam. The billing roles are as follows:

- Billing Account Creator, which can create new self-service billing accounts
- Billing Account Administrator, which manages billing accounts but cannot create them
- Billing Account User, which enables a user to link projects to billing accounts
- Billing Account Viewer, which enables a user to view billing account cost and transactions

Few users will likely have Billing Account Creator, and those who do will likely have a financial role in the organization. Cloud admins may have Billing Account Administrator to manage the accounts. Any user who can create a project should have Billing Account User so that new projects can be linked to the appropriate billing account. Billing Account Viewer is useful for some, like an auditor who needs to be able to read billing account information but not change it.

Billing Budgets and Alerts

The Google Cloud Billing service includes an option for defining a budget and setting billing alerts. You can navigate to the budget form from the main console menu, select Billing, and then select Budgets & Alerts (see Figure 3.16).

FIGURE 3.16 The budget form enables you to have notices sent to you when certain percentages of your budget have been spent in a particular month.

Google Cloud Platform

Search Products, resources, docs (/)

Billing

Billing account
My Billing Account

- Overview
- Reports
- Cost table
- Cost breakdown
- Commitments
- Commitment analysis
- Budgets & alerts**
- Billing export
- Pricing
- Documents
- Transactions
- Payment settings
- Payment method
- Release Notes

Create Budget

1 Scope

Name *

A budget enables you to track your actual spend against your planned spend.

Time range
Monthly

The month starts on the first of the month and reset at the beginning of each month.

A budget can be scoped to focus on a specific set of resources.

Projects
All projects (1)

Services
All services (1938)

Credits
Selected credits are applied to the total cost. Budget tracks the total cost minus any applicable selected credits

☒ Discounts ?

☒ Promotions and others ?

NEXT

2 Amount

In the budget form, you can name your budget and specify a billing account to monitor. Note that a budget is associated with a billing account, not a project. One or more projects can be linked to a billing account, so the budget and alerts you specify should be based on what you expect to spend for all projects linked to the billing account.

You can specify a particular amount or specify that your budget is the amount spent in the previous month.

With a budget, you can set multiple alert percentages. By default, three percentages are set: 50 percent, 90 percent, and 100 percent. You can change those to percentages that work best for you. If you'd like more than three alerts, you can click Add Item in the Set Budget Alerts section to add additional alert thresholds.

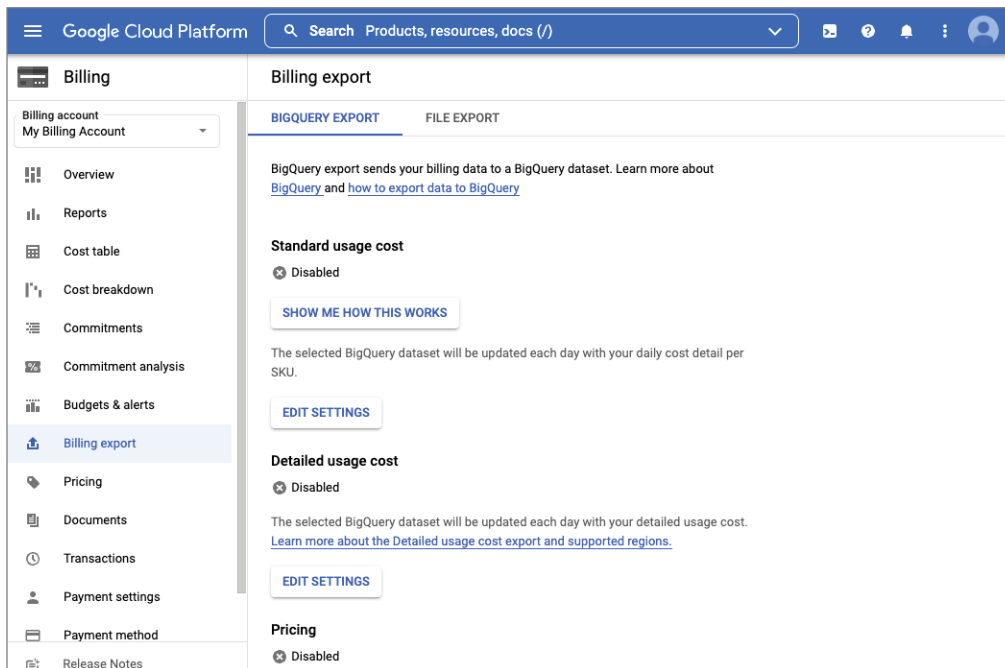
When that percentage of a budget has been spent, it will notify billing administrators and billing account users by email. If you would like to respond to alerts programmatically, you can have notifications sent to a Pub/Sub topic by checking the appropriate box in the Manage Notification sections.

Exporting Billing Data

You can export billing data for later analysis or for compliance reasons. Billing data can be exported to BigQuery.

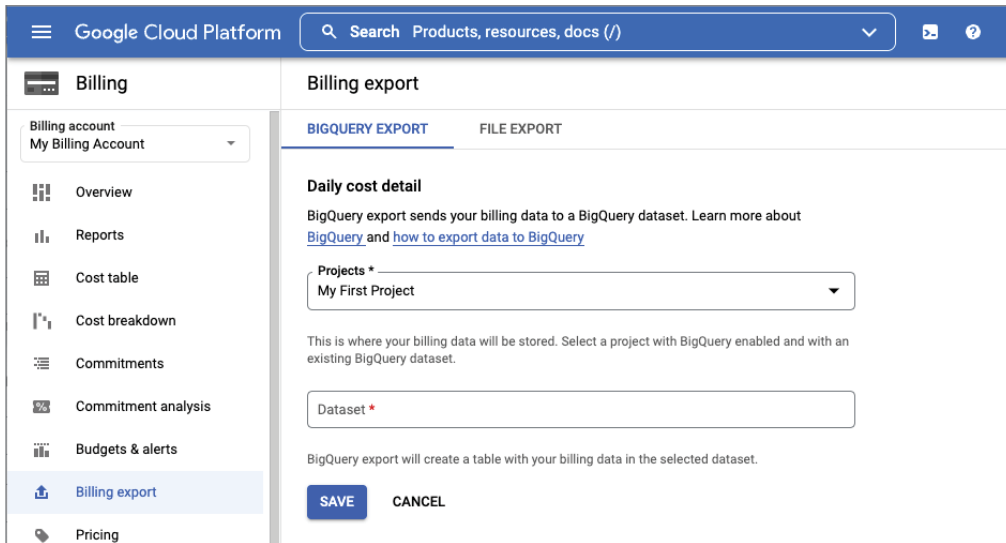
To export billing data to BigQuery, navigate to the Billing section of the console and select Billing Export from the menu. In the form that appears, select the billing account you would like to export (see Figure 3.17).

FIGURE 3.17 Billing export form



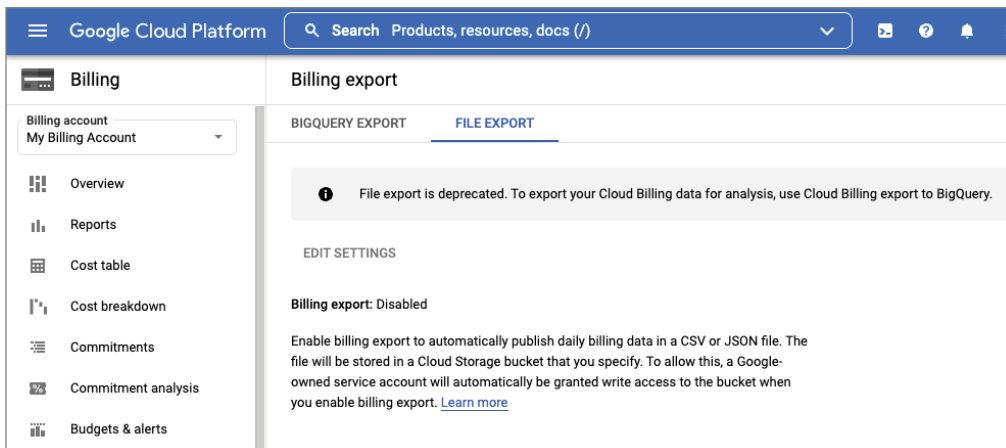
For BigQuery, click Edit Setting. Select the projects you want to include. You will need to create a BigQuery data set to hold the data. Click Go To BigQuery to open a BigQuery form. This will create a Billing export data set, which will be used to hold exported data. (See Figure 3.18.) For additional information on using BigQuery, see Chapter 12, “Deploying Storage in Google Cloud.”

FIGURE 3.18 Exporting to BigQuery



Alternatively, in the past you could export billing data to a file stored in Cloud Storage but that is no longer supported. A File Export option is available, but it no longer functions, as shown in Figure 3.19. By the time you read this, the File Export option may have been removed.

FIGURE 3.19 Exporting billing data to a file is now deprecated.



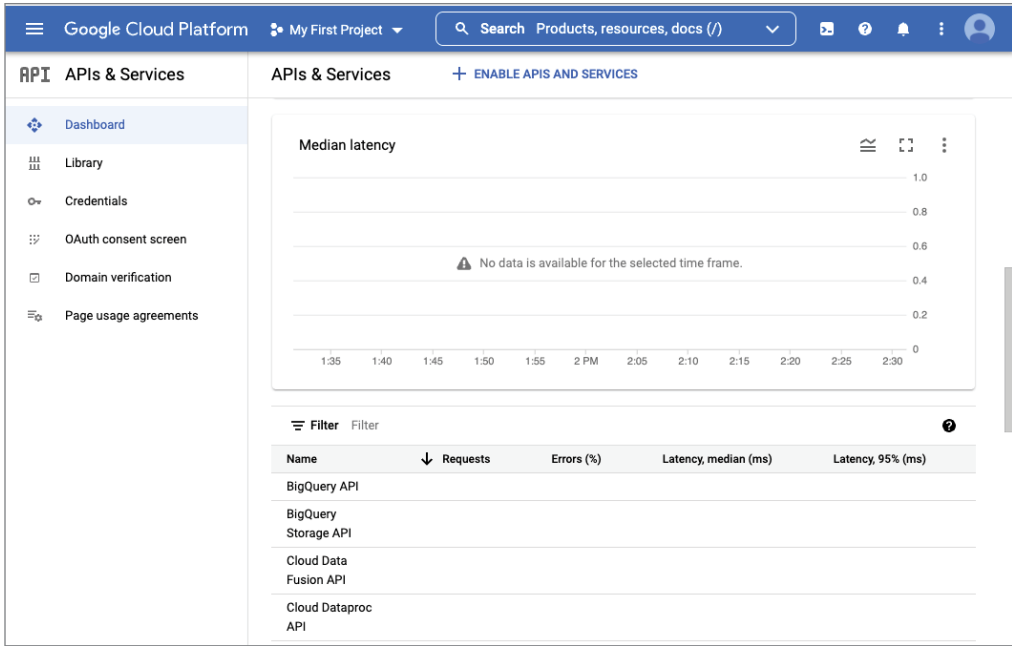
When exporting to a file, you will need to specify a bucket name and a report prefix. You have the option of choosing either the CSV or JSON file format. There may be questions on the exam about available file format options, so remember these two options.

Enabling APIs

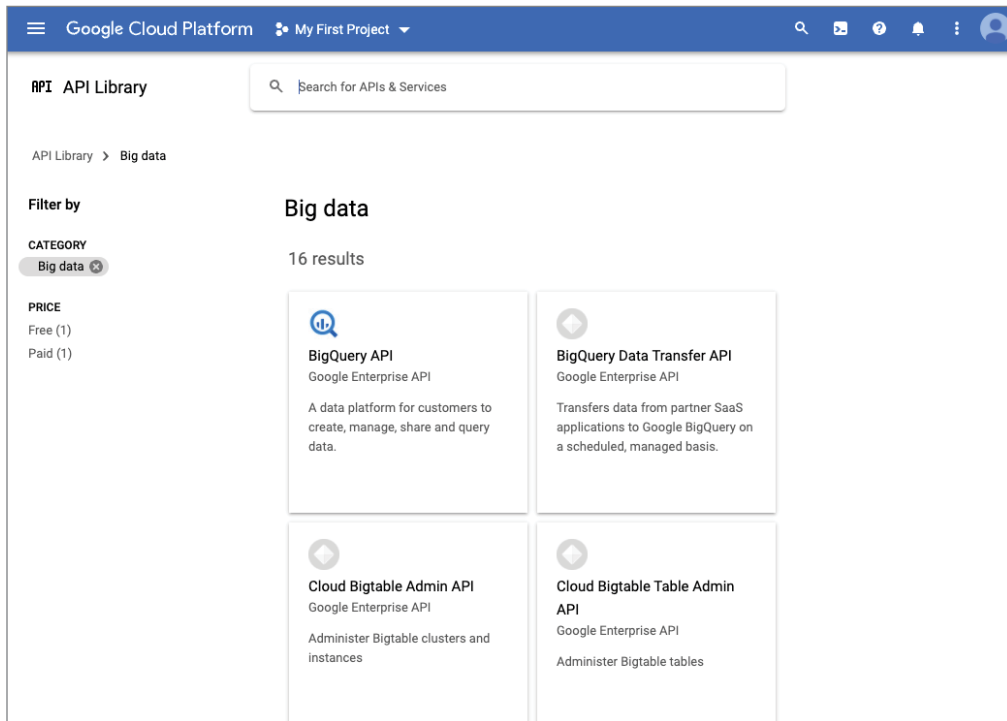
Google Cloud uses APIs to make services programmatically accessible. For example, when you use a form to create a VM or a Cloud Storage bucket, behind the scenes, API functions are executed to create the VM or bucket. All Google Cloud services have APIs associated with them. Most, however, are not enabled by default in a project.

To enable service APIs, you can select APIs & Services from the main console menu. This will display a dashboard, as shown in Figure 3.20.

FIGURE 3.20 An example API services dashboard



If you click the Enable APIs And Services link, you will see a list of services that you can enable, as shown in Figure 3.21.

FIGURE 3.21 Example services for Big Data operations

This form is a convenient way to enable APIs you know you will need. If you attempt an operation that requires an API that is not enabled, you may be prompted to decide if you want to enable the API.

Enabled APIs will have a Disable option. You can click that to disable the API. You can also click the name of an API in the list to drill down into details about API usage.

Summary

The central abstraction for managing Google Cloud resources is the resource hierarchy. It consists of three levels: organization, folder, and project. The Organization Policy Service and IAM together control access to an organization's resources. Billing accounts store information about how to pay charges for resources used. A billing account is associated with one or more projects. Google Cloud uses APIs to make services programmatically accessible and most are not enabled by default.

Exam Essentials

Understand the Google Cloud resource hierarchy. All resources are organized within your resource hierarchy. You can define the resource hierarchy using one organization and multiple folders and projects. Folders are useful for grouping departments, and other groups manage their projects separately. Projects contain resources such as VMs and cloud storage buckets. Projects must have billing accounts associated with them to use services that aren't free.

Understand organization policies. Organization policies restrict resources in the resource hierarchy. Policies include constraints, which are rules that define what can or cannot be done with a resource. For example, a constraint can be set to block access to the serial port on all VMs in a project. Also, understand the policy evaluation process and how to override inherited policies.

Understand service accounts and how they are used. Service accounts are identities that are not associated with a specific user but can be assigned to a resource, like a VM. Resources that are assigned a service account can perform operations that the service account has permission to perform. Understand service accounts and how to create them.

Understand Google Cloud Billing. Billing must be enabled to use services and resources beyond free services. Billing associates a billing method, such as a credit card or invoicing information, with a project. All costs associated with resources in a project are billed to the project's billing account. A billing account can be associated with more than one project. You manage your billing through the Billing API.

Review Questions

You can find the answers in the Appendix.

1. You are designing cloud applications for a healthcare provider. The records management application will manage medical information for patients. Access to this data is limited to a small number of employees. The billing department application will have insurance and payment information. Another group of employees will have access to billing information. In addition, the billing system will have two components: a private insurance billing system and a government payer billing system. Government regulations require that software used to bill the government must be isolated from other software systems. Which of the following resource hierarchies would meet these requirements and provide the most flexibility to adapt to changing requirements?
 - A. One organization, with folders for records management and billing. The billing folder would have private insurer and government payer folders within it. Common constraints would be specified in organization-level policies. Other policies would be defined at the appropriate folder.
 - B. One folder for records management, one for billing, and no organization. Policies defined at the folder level.
 - C. One organization, with folders for records management, private insurer, and government payer below the organization. All constraints would be specified in organization-level policies. All folders would have the same policy constraints.
 - D. None of the above.
2. When you create a hierarchy, you can have more than one of which structure?
 - A. Organization only
 - B. Folder only
 - C. Folder and project
 - D. Project only
3. You are designing an application that uses a series of services to transform data from its original form into a format suitable for use in a data warehouse. Your transformation application will write to the message queue as it processes each input file. You don't want to give users permission to write to the message queue. You could allow the application to write to the message queue by using which of the following?
 - A. Billing account
 - B. Service account
 - C. Messaging account
 - D. Folder

4. Your company has several policies that need to be enforced for all projects. You decide to apply policies to the resource hierarchy. Not long after you apply the policies, an engineer finds that an application that had worked prior to implementing policies is no longer working. The engineer would like you to create an exception for the application. How can you override a policy inherited from another entity in the resource hierarchy?
 - A. Inherited policies can be overridden by defining a policy at a folder or project level.
 - B. Inherited policies cannot be overridden.
 - C. Policies can be overridden by linking them to service accounts.
 - D. Policies can be overridden by linking them to billing accounts.
5. Constraints are used in resource hierarchy policies. Which of the following are types of constraints allowed?
 - A. Allow a specific set of values.
 - B. Deny a specific set of values.
 - C. Deny a value and all its child values.
 - D. Allow all allowed values.
 - E. All of the above.
6. A team with four members wants you to set up a project that needs only general permissions for all resources. You are granting each person a basic role for different levels of access, depending on their responsibilities in the project. Which of the following are not included as basic roles in Google Cloud?
 - A. Owner
 - B. Publisher
 - C. Editor
 - D. Viewer
7. You are deploying a new custom application and want to delegate some administration tasks to DevOps engineers. They do not need all the privileges of a full application administrator, but they do need a subset of those privileges. What kind of role should you use to grant those privileges?
 - A. Basic
 - B. Predefined
 - C. Advanced
 - D. Custom
8. An app for a finance company needs access to a database and a Cloud Storage bucket. There is no predefined role that grants all the needed permissions without granting some permissions that are not needed. You decide to create a custom role. When defining custom roles, you should follow which of the following principles?
 - A. Rotation of duties
 - B. Least principle
 - C. Defense in depth
 - D. Least privilege

9. How many organizations can you create in a resource hierarchy?
- A. 1
 - B. 2
 - C. 3
 - D. Unlimited
10. You are contacted by the finance department of your company for advice on how to automate payments for Google Cloud services. What kind of account would you recommend setting up?
- A. Service account
 - B. Billing account
 - C. Resource account
 - D. Credit account
11. You are experimenting with Google Cloud for your company. You do not have permission to incur costs. How can you experiment with Google Cloud without incurring charges?
- A. You can't; all services incur charges.
 - B. You can use a personal credit card to pay for charges.
 - C. You can use only free services in Google Cloud.
 - D. You can use only serverless products, which are free to use.
12. The CFO of your company is concerned that they will learn of unusually high cloud computing bills only after charges have been incurred. What mechanism in Google Cloud could be used to address the CFO's concern?
- A. Cloud Monitoring
 - B. Cloud Logging
 - C. Budgeting and Alerting
 - D. Policy Constraints
13. A large enterprise is planning to use Google Cloud across several subdivisions. Each subdivision is managed independently and has its own budget. Most subdivisions plan to spend tens of thousands of dollars per month. How would you recommend they set up their billing account(s)?
- A. Use a single self-service billing account.
 - B. Use multiple self-service billing accounts.
 - C. Use a single invoiced billing account.
 - D. Use multiple invoiced billing accounts.

14. An application administrator is responsible for managing all resources in a project. They want to delegate responsibility for several service accounts to another administrator. If additional service accounts are created, the other administrator should manage those as well. What is the best way to delegate privileges needed to manage the service accounts?
- A. Grant `iam.serviceAccountUser` to the administrator at the project level.
 - B. Grant `iam.serviceAccountUser` to the administrator at the service account level.
 - C. Grant `iam.serviceProjectAccountUser` to the administrator at the project level.
 - D. Grant `iam.serviceProjectAccountUser` to the administrator at the service account level.
15. You work for a retailer with a large number of stores. Every night the stores upload daily sales data. You have been tasked with creating a service that verifies the uploads every night. You decide to use a service account. Your manager questions the security of your proposed solution, particularly about authenticating the service account. You explain the authentication mechanism used by service accounts. What authentication mechanism is used?
- A. Username and password
 - B. Two-factor authentication
 - C. Encryption keys
 - D. Biometrics
16. What objects in Google Cloud are sometimes treated as resources and sometimes as identities?
- A. Billing accounts
 - B. Service accounts
 - C. Projects
 - D. Roles
17. You plan to develop a web application using products from the Google Cloud that already include established roles for managing permissions such as read-only access or the ability to delete old versions. Which of the following roles offers these capabilities?
- A. Basic roles
 - B. Predefined roles
 - C. Custom roles
 - D. Application roles
18. You are reviewing a new Google Cloud account created for use by the finance department. An auditor has questions about who can create projects by default. You explain who has privileges to create projects by default. Who is included?
- A. Only project administrators
 - B. All users
 - C. Only users without the role `resourcemanager.projects.create`
 - D. Only billing account users

- 19.** How many projects can be created in an account?
- A.** 10.
 - B.** 25.
 - C.** There is no limit.
 - D.** Each account has a limit determined by Google.
- 20.** You are planning how to grant privileges to users of your company's Google Cloud account. You need to document what each user will be able to do. Auditors are most concerned about a role called Organization Administrator. You explain that users with that role can perform a number of tasks, which include all of the following except which one?
- A.** Defining the structure of the resource hierarchy
 - B.** Determining what permissions a user should be assigned
 - C.** Defining IAM policies over the resource hierarchy
 - D.** Delegating other management roles to other users

Chapter 4

Introduction to Computing in Google Cloud

**THIS CHAPTER COVERS THE FOLLOWING
OBJECTIVE OF THE GOOGLE ASSOCIATE
CLOUD ENGINEER CERTIFICATION EXAM:**

✓ 2.2 Planning and configuring compute resources





In this chapter, you will learn about each of the compute options available in Google Cloud and when to use them. We will also discuss preemptible virtual machines and when they can help reduce your overall computing costs.

Compute Engine

Compute Engine is a service that provides virtual machines (VMs) that run on Google Cloud. We usually refer to a running VM as an *instance*. When you use Compute Engine, you create and manage one or more instances.

Virtual Machine Images

Instances run images, which contain operating systems, libraries, and other code. You may choose to run a public image provided by Google (Figure 4.1). Both Linux and Windows images are available. In addition to the images maintained by Google, there are other public images provided by open source projects or third-party vendors.

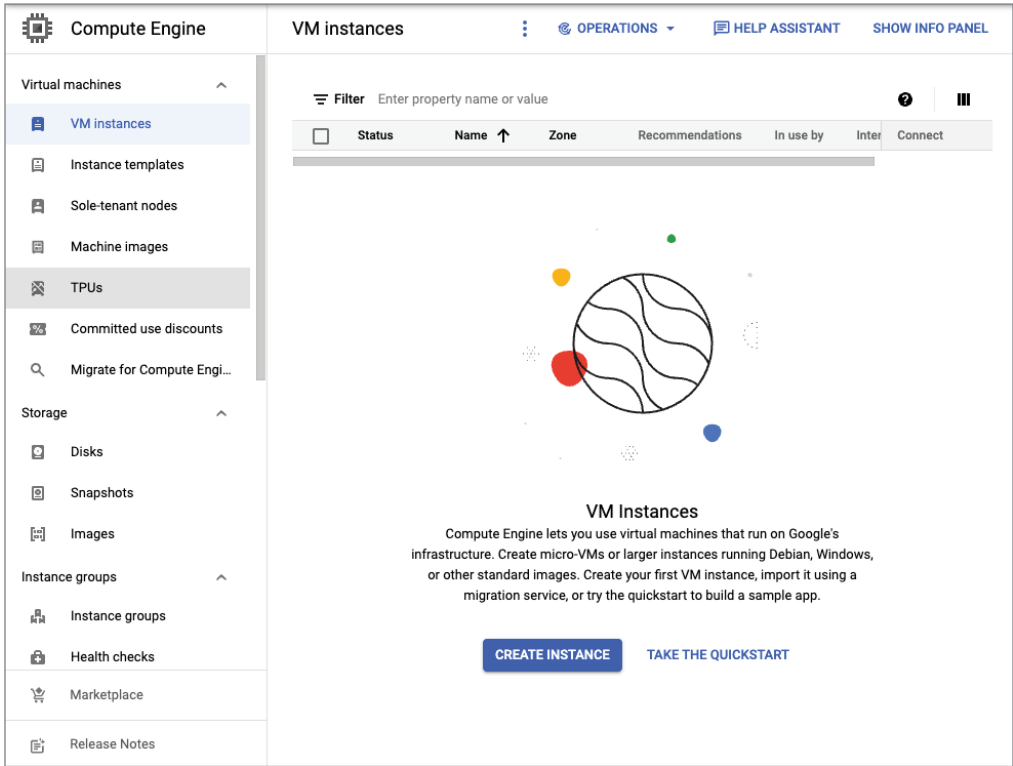
FIGURE 4.1 A subset of operating system images available in Compute Engine

Compute Engine									
Images									
CREATE IMAGE REFRESH DELETE HELP ASSISTANT SHOW INFO PANEL									
<div>Virtual machines</div> <div>VM instances</div> <div>Instance templates</div> <div>Sole-tenant nodes</div> <div>Machine images</div> <div>TPUs</div> <div>Committed use discounts</div> <div>Migrate for Compute Eng...</div> <div>Storage</div> <div>Disks</div> <div>Snapshots</div> <div>Images</div> <div>Instance groups</div> <div>Instance groups</div> <div>Health checks</div> <div>Marketplace</div> <div>Release Notes</div>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	vzuzz2uz2z-debian-10						
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	c2-deeplearning-pytorch-1-10-cu110-v20220202-debian-10	asia, eu, us	—	50 GB	Debian	pytorch-1-10-gpu-debian-10	Feb 2, 2022, 1:34:16 PM UTC-08:00
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	c2-deeplearning-pytorch-1-10-xla-v20220202-debian-10	asia, eu, us	—	50 GB	Debian	pytorch-1-10-xla-debian-10	Feb 2, 2022, 12:27:53 PM UTC-08:00
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	centos-7-v20200403	asia, eu, us	—	20 GB	CentOS	centos-7	Apr 6, 2020, 2:51:35 PM UTC-07:00
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	centos-7-v20220126	asia, eu, us	—	20 GB	CentOS	centos-7	Jan 26, 2022, 2:27:27 PM UTC-08:00
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	centos-stream-8-v20220128	asia, eu, us	—	20 GB	CentOS	centos-stream-8	Jan 28, 2022, 11:16:25 AM UTC-08:00
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	cos-69-10895-385-0	asia, eu, us	—	10 GB	Google	cos-69-its	Oct 8, 2019, 11:25:22 PM UTC-07:00
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	cos-73-11647-656-0	asia, eu, us	—	10 GB	Google	cos-73-its	Sep 5, 2020, 2:25:50 PM UTC-07:00
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	cos-77-12371-1109-0	asia, eu, us	—	10 GB	Google	cos-77-its	Jan 11, 2021, 11:36:57 AM UTC-08:00
	<input type="checkbox"/>	<input checked="" type="checkbox"/>							

The public images include a range of operating systems, such as CentOS, Container-Optimized OS from Google, Debian, Red Hat Enterprise Linux, SUSE Enterprise Linux Server, Ubuntu, and Windows Server.

If there is no public image that meets your needs, you can create a custom image from a boot disk or by starting with another image. To create a VM from the console, navigate to Compute Engine and then to VM Instances. You will see a screen similar to Figure 4.2.

FIGURE 4.2 Creating a VM in Compute Engine



Click Create Instance to open the page for creating an instance. Here, as shown in Figure 4.3, you can set the name of the instance, choose the machine configuration, add graphics processing units (GPUs), and set other features of the instance.

Other configurable features of an instance are shown in Figure 4.4. For example, for high-security applications, you can use the Confidential VM service to encrypt data in memory. You can also specify a name, size, image, and type of the boot disk. VMs have an associated identity called a service account associated with them. Service accounts are identities, like users and groups, but are not associated with human users. Service accounts can be assigned roles so that they can have permissions to perform actions in Google Cloud. (For more on service accounts, see Chapter 3, “Projects, Service Accounts, and Billing.”)

FIGURE 4.3 Part 1 of creating an instance in Compute Engine

← Create an instance HELP ASSISTANT

To create a VM instance, select one of the options:

- New VM instance**
Create a single VM instance from scratch
- New VM instance from template
Create a single VM instance from an existing template
- New VM instance from machine image
Create a single VM instance from an existing machine image
- Marketplace
Deploy a ready-to-go solution onto a VM instance

Name *
instance-1

Labels
[+ ADD LABELS](#)

Region *
us-west4 (Las Vegas) ?
Region is permanent

Zone *
us-west4-b ?
Zone is permanent

Monthly estimate
\$28.65
That's about \$0.04 hourly
You have \$299.42 free trial credits remaining
Pay for what you use: No upfront costs and per second billing
[DETAILS](#)

Machine configuration

Machine family
[GENERAL-PURPOSE](#) [COMPUTE-OPTIMIZED](#) [MEMORY-OPTIMIZED](#)

Machine types for common workloads, optimized for cost and flexibility

Series
E2

CPU platform selection based on availability

Machine type
e2-medium (2 vCPU, 4 GB memory)

	vCPU	Memory
	1 shared core	4 GB

[CPU PLATFORM AND GPU](#)

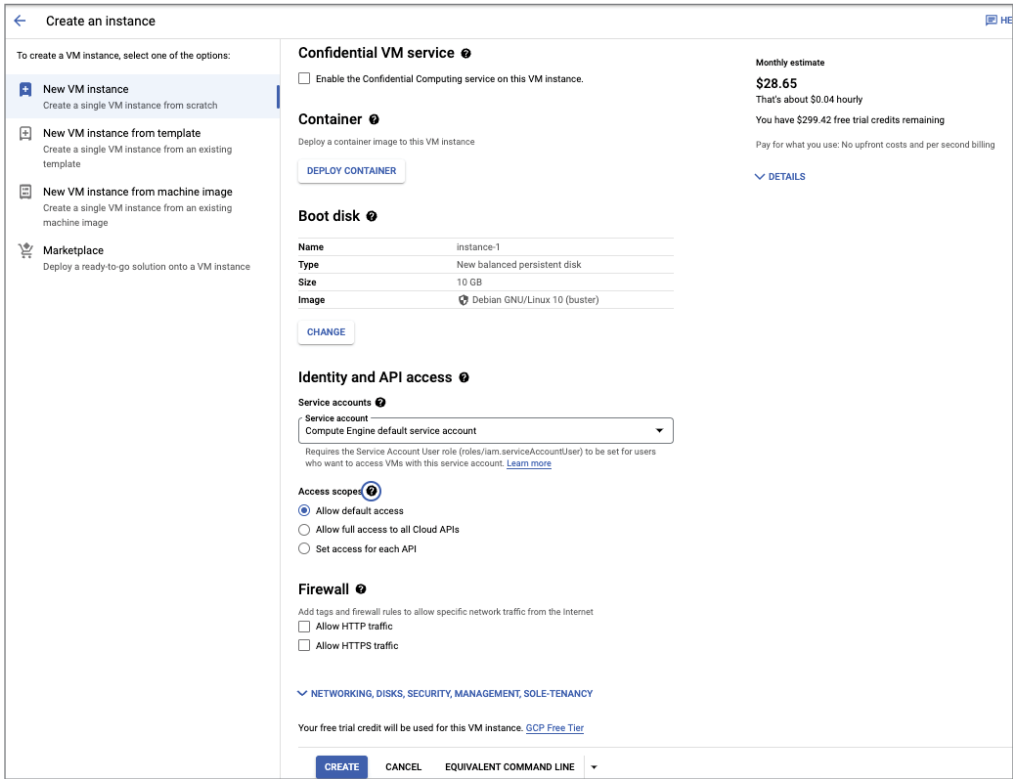
Display device
Enable to use screen capturing and recording tools.
☐ Enable display device

You can also control what actions an instance can perform by setting access scopes. Access scopes are a legacy access control mechanism that existed before the Identity and Access Management (IAM) service. By default, access scopes allow minimal access, including the ability to read from storage and to write to monitoring and logging services. IAM is the preferred method for controlling access granted to a Compute Engine instance.

You can also specify whether HTTP or HTTPS traffic is allowed to the instance.

Figure 4.5 shows networking configurations for an instance. In the Networking section of the Create Instance page, you can specify network tags, a hostname, and network performance configurations, as well as add additional network interfaces. One network interface is created by default.

FIGURE 4.4 Part 2 of creating an instance in Compute Engine



If you would like additional disks, along with the boot disk, you can add and configure disks on this page as well. Figure 4.6 shows the options for configuring disks. You can provide a name, description, disk type, size, a backup schedule for the disk, and encryption settings. Note that all data in Google Cloud is encrypted when stored (known as *encryption at rest*) in persistent storage. We do not have the option to persistently store data without encryption, but we can choose how encryption keys are managed. Currently, the choices are Google-managed encryption keys, customer-managed encryption keys, and customer-supplied encryption keys. With Google-managed encryption keys, Google creates and manages keys. With customer-supplied encryption keys, customers create their own keys but Google manages them. When we use customer-supplied encryption keys, the customers create and manage keys outside of Google Cloud.

FIGURE 4.5 Configuring network properties in a Compute Engine instance

Networking
Hostname and network interfaces

Network tags

Hostname
Set a custom hostname for this instance or leave it default. Choice is permanent

IP forwarding
☐ Enable

Network performance configuration
Network interface card

Network bandwidth
☐ Increase total egress bandwidth
Maximum outbound network bandwidth: 2Gbps

Network interfaces
Network interface is permanent

default default (10.182.0.0/20)

ADD NETWORK INTERFACE

To create another network interface you need to have a new network first.

Disks can be attached as read/write disks or as read-only disk. A disk by default is kept when an instance is deleted, but you can choose to have the disk deleted when the instance is deleted.

In the Security section of the Create Instance page, you can specify some advanced security features. (See Figure 4.7.) Secure Boot protects against boot-level and kernel-level malicious code, such as rootkits. The Virtual Trusted Platform Module (vTPM) validates boot integrity and provides additional protections for key generation and protection. When vTPM is enabled, you have the option of enabling Integrity Monitoring, which verifies the runtime integrity of the virtual machine.

FIGURE 4.6 Configuring disks in a Compute Engine instance

Add new disk

Name *

disk-1

?

Name is permanent

Description

Source

Create a blank disk, apply a bootable disk image, or restore a snapshot of another disk in this project.

Disk source type *

Blank disk

Disk settings

Disk type *

Balanced persistent disk

?

COMPARE DISK TYPES

Size *

100

GB

?

Provision between 10 and 65,536 GB

Snapshot schedule (Recommended)

Use snapshot schedules to automate disk backups. [Learn more](#)

Select a snapshot schedule

Encryption

Data is encrypted automatically. Select an encryption key management solution.

☒ Google-managed encryption key

No configuration required

☐ Customer-managed encryption key (CMEK)

Manage via Google Cloud Key Management Service

☐ Customer-supplied encryption key (CSEK)

Manage outside of Google Cloud

Labels ?

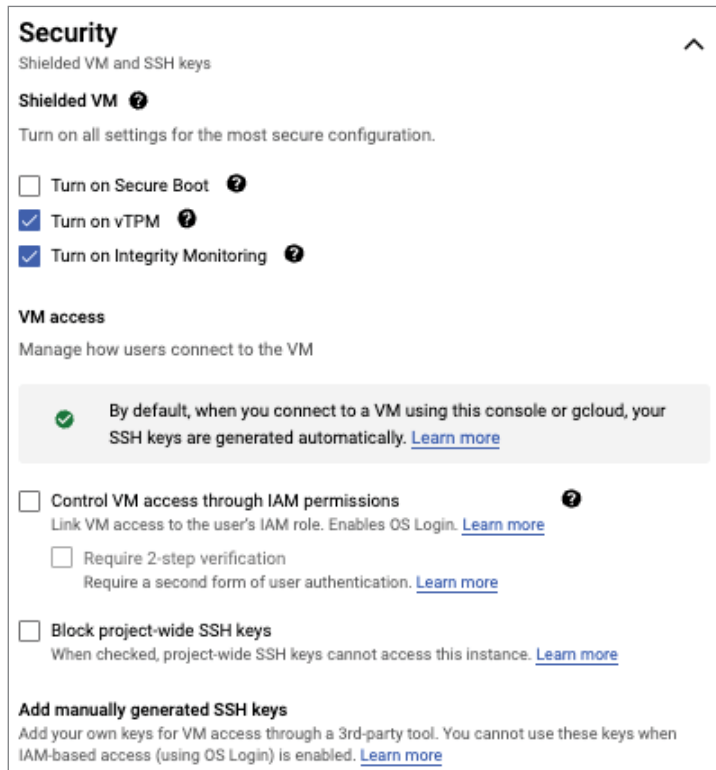
+ ADD LABEL

Attachment settings

Mode

SAVE

CANCEL

FIGURE 4.7 Configuring security in a Compute Engine instance

The screenshot shows the 'Security' settings page for a Compute Engine instance. The page is titled 'Security' with a subtitle 'Shielded VM and SSH keys'. Under the 'Shielded VM' section, there is a note 'Turn on all settings for the most secure configuration.' and three checkboxes: 'Turn on Secure Boot' (unchecked), 'Turn on vTPM' (checked), and 'Turn on Integrity Monitoring' (checked). The 'VM access' section has a subtitle 'Manage how users connect to the VM' and a green checkmark icon followed by the text 'By default, when you connect to a VM using this console or gcloud, your SSH keys are generated automatically. [Learn more](#)'. Below this are three checkboxes: 'Control VM access through IAM permissions' (unchecked), 'Require 2-step verification' (unchecked), and 'Block project-wide SSH keys' (unchecked). Each checkbox has a brief description and a 'Learn more' link. At the bottom, there is a section titled 'Add manually generated SSH keys' with a subtitle 'Add your own keys for VM access through a 3rd-party tool. You cannot use these keys when IAM-based access (using OS Login) is enabled. [Learn more](#)'.

Security
Shielded VM and SSH keys

Shielded VM ⓘ
Turn on all settings for the most secure configuration.

☐ Turn on Secure Boot ⓘ
☒ Turn on vTPM ⓘ
☒ Turn on Integrity Monitoring ⓘ

VM access
Manage how users connect to the VM

✓ By default, when you connect to a VM using this console or gcloud, your SSH keys are generated automatically. [Learn more](#)

☐ Control VM access through IAM permissions ⓘ
Link VM access to the user's IAM role. Enables OS Login. [Learn more](#)

☐ Require 2-step verification
Require a second form of user authentication. [Learn more](#)

☐ Block project-wide SSH keys
When checked, project-wide SSH keys cannot access this instance. [Learn more](#)

Add manually generated SSH keys
Add your own keys for VM access through a 3rd-party tool. You cannot use these keys when IAM-based access (using OS Login) is enabled. [Learn more](#)

You can further restrict access to an instance through IAM roles. When this feature is enabled, only users with the Compute OS Login role, Compute OS Admin Login role, or other roles that have permissions to enable IAM-based access can login. Another way to block access is to disallow the use of project-based SSH keys, which by default would allow access to any VM instance in a project.

Figure 4.8 shows the options for specifying management features. These include a description, the ability to block deletion of the instance, instance reservations (a way of purchasing blocks of instance time at a discount), and whether you want an automation script to run on startup. You can also configure availability parameters, including choosing to make this a preemptible VM. Preemptible VMs cost less but can be shut down at any time by Google Cloud. Originally, preemptible VMs would run for a maximum of 24 hours before being shut down. Google Cloud now offers spot VMs, which are billed like preemptible VMs but are not necessarily shut down after 24 hours. You can also specify if the instance should be migrated to another server during server maintenance and started automatically if there is a hardware failure or other non-user-initiated shutdown.

FIGURE 4.8 Configuring management features in a Compute Engine instance

Management

Description, deletion protection, reservations, automation, and availability policies

Description

Deletion protection

☐ Enable deletion protection

Reservations

Automatically use created reservation

Use an existing reservation when creating this VM instance

Automation

Startup script

You can choose to specify a startup script that will run when your instance boots up or restarts. Startup scripts can be used to install software and updates, and to ensure that services are running within the virtual machine. [Learn more](#)

Metadata

You can set custom metadata for an instance or project outside of the server-defined metadata. This is useful for passing in arbitrary values to your project or instance that can be queried by your code on the instance. [Learn more](#)

+ ADD ITEM

Availability policy

Preemptibility

Off (Recommended)

A preemptible VM costs much less, but lasts only 24 hours. It can be terminated sooner due to system demands. [Learn more](#)

On host maintenance

Migrate VM instance (Recommended)

When Compute Engine performs periodic infrastructure maintenance it can migrate your VM instances to other hardware without downtime

Automatic restart

On (recommended)

Compute Engine can automatically restart VM instances if they are terminated for non-user-initiated reasons (maintenance event, hardware failure, software failure and so on)

Month

\$28.

That's

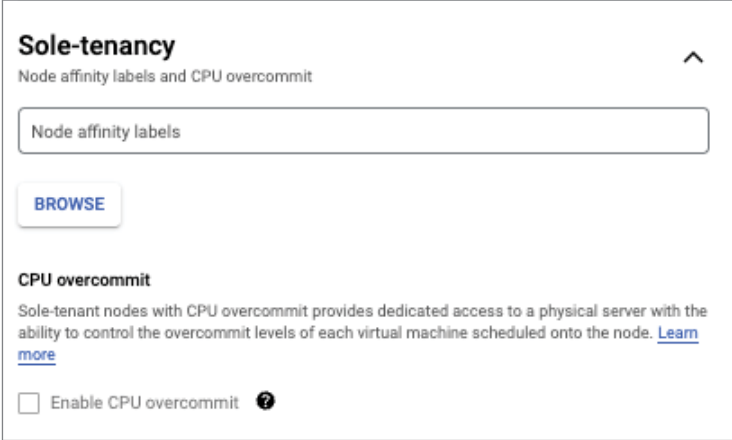
You h

Pay fo

▼ DE

There may be times when you do not want virtual machines from other projects running on the same server as your project's virtual machines. In such cases, you can choose Sole Tenancy for your instance (see Figure 4.9). Only VMs from your project with node affinity labels matching the labels you specify here will run on the same server together. You also have the option of overcommitting the CPUs on a server that is configured as sole tenant. This can increase performance by scheduling VMs with more CPU requirements than actually available if the VMs do not need all the committed resources at the same time. For example, if two instances are running on a server and one instance has a peak load in the morning and the other has a peak load in the evening, you may be able to overcommit without adversely impacting performance of either instance.

FIGURE 4.9 Configuring Sole Tenancy features in a Compute Engine instance



Sole-tenancy
Node affinity labels and CPU overcommit

Node affinity labels

BROWSE

CPU overcommit
Sole-tenant nodes with CPU overcommit provides dedicated access to a physical server with the ability to control the overcommit levels of each virtual machine scheduled onto the node. [Learn more](#)

☐ Enable CPU overcommit ?

If you are going to create additional instances with the same configuration, you can create an *instance template*. A template is a description of a VM configuration. The process of creating an instance template is similar to creating a VM as just described but instead of creating a VM when complete, you will have created a template. You can then use that template to create a new instance without having to specify all the configuration parameters manually.

Another way to create an instance is from a machine image that you create. Figure 4.10 shows the dialog box for creating a machine image from an existing VM. You specify a name, description, source VM, and location to store the image. You can also specify how encryption keys are managed.

FIGURE 4.10 Creating a machine image

Create a machine image

A machine image contains a VM's properties, metadata, permissions, and data from all its attached disks. You can use a machine image to create, backup, or restore a VM. [Learn more](#)

Name *

Name is permanent

Description

Source VM instance *

Location

☐ Multi-regional
 ☐ Regional

Select location

Encryption

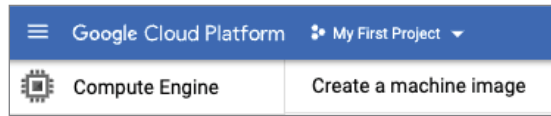
☒ Google-managed encryption key
No configuration required
 ☐ Customer-managed encryption key (CMEK)
Manage via Google Cloud Key Management Service
 ☐ Customer-supplied encryption key (CSEK)
Manage outside of Google Cloud

Virtual Machines Are Contained in Projects

When you create an instance, you specify a project to contain the instance. As you may recall, projects are part of the Google Cloud resource hierarchy. Projects are the lowest-level structure in the hierarchy. Projects allow you to manage related resources with common policies.

When you open Google Cloud Console, you will notice at the top of the form either the name of a project or the phrase *Select A Project*, as shown in Figure 4.11.

FIGURE 4.11 The current project name or the option to select one is displayed in Google Cloud Console.



When you choose Select A Project, a form like the one in Figure 4.12 appears. From there, you can select the project you want to store your resources, including VMs.

FIGURE 4.12 Choosing a project from existing projects in an account



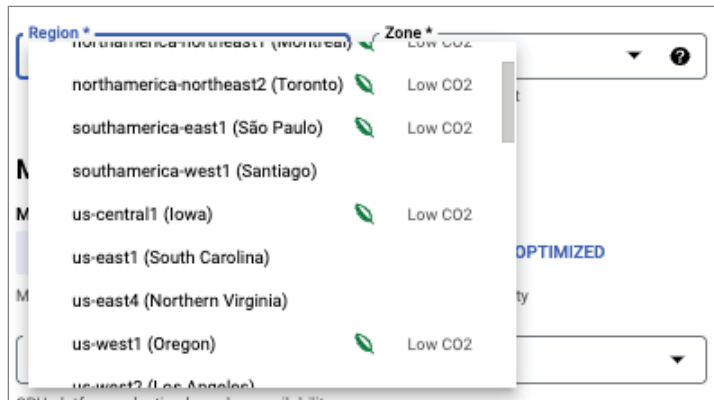
Virtual Machines Run in a Zone and Region

In addition to having a project, VM instances have a zone assigned. Zones are data center–like resources, but they may consist of one or more closely coupled data centers. They are located within regions. A *region* is a geographical location, such as `asia-east1`, `eu-west2`, and `us-east4`. The zones within a region are linked by low-latency, high-bandwidth network connections.

You specify a region and a zone when you create a VM. As you can see in Figure 4.13 and Figure 4.14, the Create VM form includes drop-down lists from which you can select the region and zone.

You may want to consider several factors when choosing where to run your VM:

- Cost, which can vary between regions.
- Data locality regulations, such as keeping data about European Union citizens in the European Union.
- High availability. If you are running multiple instances, you may want them in different zones and possibly different regions. If one of the zones or regions becomes inaccessible, the instances in other zones and regions can still provide services.

FIGURE 4.13 Selecting a region in the Create VM form**FIGURE 4.14** Once a region is selected, you can choose a zone within that region.

- Latency, which is important if you have users in different parts of the world. Keeping instances and data geographically close to application users can help reduce latency.
- Need for specific hardware platforms, which can vary by region. For example, europe-west1 may have a processor available that is not available in europe-west2.
- The carbon intensity of the power generation in the region.

Users Need Privileges to Create Virtual Machines

To create Compute Engine resources in a project, users must be members of the project or a specific resource and have appropriate permissions to perform specific tasks. Users can be associated with projects as follows:

- Individual users
- A Google group
- A Google Workspace domain
- A service account

Once a user, or a set of users, is added to a project, you can assign permissions by granting roles to the user or set of users. This process is explained in detail in Chapter 17, “Configuring Access and Security.” Predefined roles are especially useful because they group together permissions that are often needed for a user to carry out a set of tasks. Here are some examples of predefined roles:

Compute Admin Users with this role have full control over Compute Engine instances.

Compute Network Admin Users with this role can create, modify, and delete most networking resources, and it provides read-only access to firewall rules and SSL certificates. This role does not give the user permission to create or alter instances.

Compute Security Admin Users with this role can create, modify, and delete SSL certificates and firewall rules.

Compute Viewer Users with this role can get and list Compute Engine resources but cannot read data from those resources.

When privileges are granted to users at the project level, then those permissions apply to all resources within a project. For example, if a user is granted the Compute Admin role at the project level, then that person can administer all Compute Engine instances in the project.

An alternative way to control access to resources is to attach IAM policies directly to resources. In this way, privileges can be tailored to specific resources instead of for all resources in a project. For example, you could specify that user Alice has the Compute Engine Admin role on one instance and Bob has the same role on another instance. Alice and Bob would be able to administer their own VM instances, but they could not administer other instances.

Preemptible Virtual Machines

Consider if you have a workload that is the opposite of needing high availability. Preemptible VMs are short-lived compute instances suitable for running certain types of workloads—particularly for applications that perform financial modeling, rendering, big data, continuous integration, and web crawling operations. These VMs offer the same configuration options as regular compute instances and persist for up to 24 hours; spot VMs do not have this time limitation. If an application is fault-tolerant and can withstand possible instance interruptions (with a 30-second warning), then using preemptible VM instances and spot VMs can reduce Google Compute Engine costs significantly.

Some big data analysis jobs run on clusters of servers running software like Hadoop and Spark. The platforms are designed to be resilient to failure. If a node goes down in the middle of a job, the platform detects the failure and moves the workload to other nodes in the server. You may have analytic jobs that are well served by a combination of reliable VMs and preemptible VMs. With some percentage of reliable VMs, you know you can get your

jobs processed within your time constraints, but if you add low-cost, preemptible VMs, you can often finish your jobs faster and at lower overall cost.

Limitations of Preemptible Virtual Machines

As you decide where to use preemptible VMs, keep in mind their limitations and differences compared to conventional VM instances in Google Cloud. Preemptible VMs have the following characteristics:

- May terminate at any time. If they terminate within 1 minute of starting, you will not be charged for that time.
- Will be terminated within 24 hours except for spot VMs.
- May not always be available. Availability may vary across zones and regions.
- Cannot migrate to a regular VM.
- Cannot be set to automatically restart.
- Are not covered by any service level agreement (SLA).

Custom Machine Types

Compute Engine has dozens of predefined machine types grouped into standard types, high-memory machines, high-CPU machines, shared core type, and memory-optimized machines. These predefined machine types vary in the number of virtual CPUs (vCPUs) and amount of memory. Here are some examples:

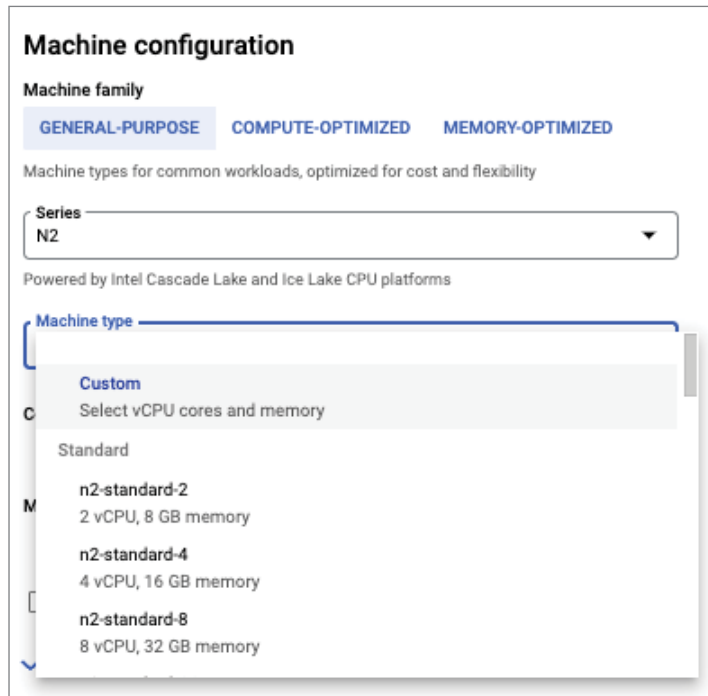
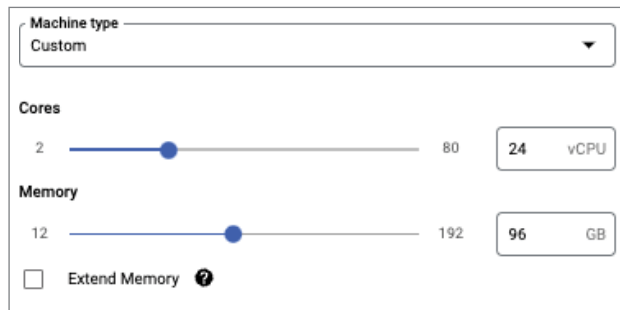
- n2-standard-2 has 2 vCPU and 8 GB of memory.
- n2-standard-32 has 32 vCPUs and 128 GB of memory.
- m2-megamem-416 has 416 vCPUs and 5.75 TB of memory.
- m2-ultramem-208 has 208 vCPUs and 5.75 TB of memory.

The predefined options for VMs will meet the needs of many use cases, but there may be times where your workload could run more cost effectively and faster on a configuration that is not already defined. In that case, you may want to use a custom machine type.

To create a custom image, select the Create VM option in the console. Click the Customize link in the Machine Type section (see Figure 4.15).

This expands the Machine Type section, as shown in Figure 4.16. From there you can adjust the sliders to increase or decrease the number of CPUs and the amount of memory you require.

The options available to create a custom machine configuration will vary by series. For example, custom machine types based on the N2 series can have between 2 and 80 vCPUs and up to 640 GB of memory. The price of a custom configuration is based on the number of vCPUs and the memory allocated. Custom machine types based on N2D series can have up to 96 cores and up to 768 GB of memory. You can select Extend Memory to increase the amount of memory relative to CPUs.

FIGURE 4.15 Choosing a custom machine type from the MachineType drop-down menu**FIGURE 4.16** Customizing a VM by adjusting the number of CPUs and the amount of memory

Use Cases for Compute Engine Virtual Machines

Compute Engine is a good option when you need maximum control over VM instances. With Compute Engine, you can do the following:

- Choose the specific image to run on the instance.
- Install software packages or custom libraries.

- Have fine-grained control over which users have permissions on the instance.
- Have control over SSL certificates and firewall rules for the instance.

Relative to other computing services in Google Cloud, Google Compute Engine provides the least amount of management. Google does provide public images and a set of VM configurations, but you as an administrator must make choices about which image to use, the number of CPUs, the amount of memory to allocate, how to configure persistent storage, and how to configure network configurations.

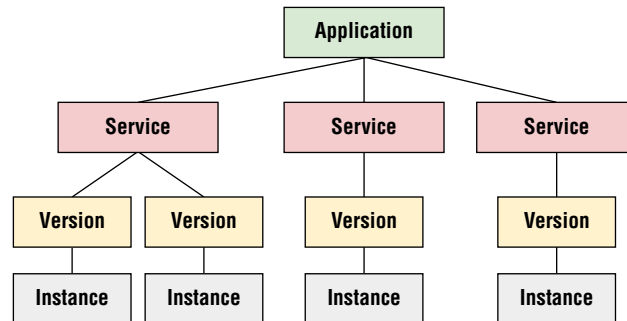
In general, the more control over a resource you have in Google Cloud, the more responsibility you have for configuring and managing the resource.

App Engine

App Engine is a PaaS compute service that provides a managed platform for running applications. When you use App Engine, your focus is on your application and not on the VMs that run the application. Instead of configuring VMs, you specify some basic resource requirements along with your application code, and Google will manage the resources needed to run the code. This means that App Engine users have less to manage, but they also have less control over the compute resources that are used to execute the application.

Like VM instances, applications in App Engine are created within a project. Unlike Compute Engine, when creating an App Engine service, you are not providing a lot of detail for configuring virtual machines. Instead, you are configuring your application to run as a service in App Engine (see Figure 4.17).

FIGURE 4.17 When using App Engine, the focus is on applications, not infrastructure.



App Engine is not included as a topic in the Google Cloud Associate Cloud Engineer exam guide, but it is included here because the service is still available and continues to be used.

Structure of an App Engine Application

App Engine applications have a common structure, and they consist of services. Services provide a specific function, like computing sales tax in a retail web application or updating inventory as products are sold on a site. Services have versions, and this allows multiple versions to run at one time. Each version of a service runs on an instance that is managed by App Engine (see Figure 4.18).

FIGURE 4.18 The structure of an App Engine application

App Engine Create app [LEARN](#)

1 **Configure application** — 2 Get started

Region

Select a region for your App Engine application. Please remember, once selected the region is permanently tied to the project.

Select a region *
us-central

Identity and API access

Select a service account

If no service account is selected the default App Engine service account will be used.

NEXT

The number of instances used to provide an application depends on your configuration for the application and the current load on the application. As the load increases, Google can add more instances to meet the need. Similarly, if the load lessens, instances can be shut down to save on the cost of unutilized instances. This kind of autoscaling is available with dynamic instances.

In addition to dynamic instances, App Engine also provides resident instances. You can add or remove resident instances manually.

When the number of deployed instances changes frequently, it can be difficult to estimate the costs of running instances. Fortunately, Google Cloud allows users to set up daily spending limits as well as create budgets and set alarms.

App Engine Standard and Flexible Environments

App Engine provides two types of runtime environments: standard and flexible. The standard environment provides language runtimes, whereas the flexible environment is a more generalized container execution platform. In both environments, your code runs in container instances running on Google Cloud managed infrastructure.

App Engine Standard Environment

The standard environment is the original App Engine environment. It consists of a preconfigured, language-specific runtime. There are currently two generations of the standard environment. The second generation improves on the performance of the first generation and has fewer limitations.

Currently, App Engine standard environment users can choose from the following supported languages:

First Generation

- Python 2.7
- Java 8
- PHP 5.5
- Go 1.11

Second Generation

- Python 3
- Java 11, 17
- Node.js
- PHP 7/8
- Ruby
- Go 1.12+

With the second-generation standard environment, developers can use any language extension, but in the first generation only a select set of extensions and libraries are allowed. Network access is restricted in the first generation, but users have full network access in the second generation.

App Engine services are scaled using automatic, manual, or basic scaling. With basic scaling, App Engine tries to keep costs low so it does not start another instance until there is a request that cannot be serviced by an existing instance. This can cause a delay in the time

to process the request because the instance has to start. With automatic scaling, App Engine automatically creates new instances as load increases. With manual scaling, you specify the number of instances for each version of a service.

App Engine standard environment is especially appealing from a cost perspective because you only pay for what you need and applications can scale to zero instances when there is no traffic to the application.

An App Engine service gets compute and memory resources based on the instance class configured for the service.

For first-generation runtimes, the default instance class for front end services, called F1, has up to 128 MB of memory and a CPU limit of 600 MHz. The default instance class for back-end services, called B2, has 256 MB of memory and a 1.2 GHz CPU limit. There are several other classes for both front-end and back-end instance classes.

For second-generation environments, F1 has 256 MB of memory and a 600 MHz CPU limit whereas the B2 instance has 512 MB of memory and a 1.2 GHz CPU limit.

Front-end instance classes scale automatically, and back-end instance classes support manual and basic scaling.

App Engine Flexible Environment

The App Engine flexible environment provides more options and control to developers who would like the benefits of a platform as a service (PaaS) like App Engine but without the language and customization constraints of the App Engine standard environment.

Like App Engine Standard, the App Engine flexible environment uses containers as the basic building block abstraction; however, in App Engine Flexible users can customize their runtime environments by configuring a container. The flexible environment uses Docker containers, so developers familiar with Dockerfiles can specify base operating system images, additional libraries and tools, and custom tools. It also has native support for Java, Python, Node.js, Ruby, PHP, .NET core, and Go. See App Engine documentation for specific versions supported.

In some ways, the App Engine flexible environment is similar to the Kubernetes Engine, which will be discussed in the next section. Both Google products can run customized Docker containers. The App Engine flexible environment provides a fully managed PaaS and is a good option when you can package your application and services into a small set of containers. These containers can then be autoscaled according to load. Kubernetes Engine, as you will see shortly, is designed to manage containers executing in a cluster that you control. With Kubernetes Engine you have control over your cluster but must monitor and manage that cluster using tools such as Cloud Monitoring and autoscaling. With the App Engine flexible environment, the health of App Engine servers is monitored by Google and corrected as needed without any intervention on your part.

Use Cases for App Engine

The App Engine product is a good choice for a computing platform when you have little need to configure and control the underlying operating system or storage system. App Engine

manages underlying VMs and containers and relieves developers and DevOps professionals of some common system administration tasks, like patching and monitoring servers.

When to Use App Engine Standard Environment

The App Engine standard environment is designed for applications written in one of the supported languages. The standard environment provides a language-specific runtime that comes with its own constraints. The constraints are fewer in the second-generation App Engine standard environment.



If you are starting a new development effort and plan to use the App Engine standard environment, then it is best to choose second-generation instances. First-generation instances will continue to be supported, but that kind of instance should be used only for applications that already exist and were designed for that platform.

When to Use App Engine Flexible Environment

The App Engine flexible environment is well suited for applications that can be decomposed into services and where each service can be containerized. For example, one service could use a Django application to provide an application user interface, another could embed business logic for data storage, and another service could schedule batch processing of data uploaded through the application. If you need to install additional software or run commands during startup, you can specify those in the Dockerfile. For example, you could add a `run` command to a Dockerfile to run `apt-get update` to get the latest version of installed packages. Dockerfiles are text files with commands for configuring a container, such as specifying a base image to start with and specifying package manager commands, like `apt-get` and `yum`, for installing packages.

The App Engine standard environment scales down to no running instances if there is no load, but this is not the case with the flexible environment. There will always be at least one container running with your service, and you will be charged for that time even if there is no load on the system.

Kubernetes Engine

Compute Engine allows you to create and manage VMs either individually or in groups called *instances groups*. Instance groups let you manage similar VMs as a single unit. This is helpful if you have a fleet of servers that all run the same software and have the same operational life cycle. Modern software, however, is often built as a collection of services, sometimes referred to as *microservices*. Different services may require different configurations of VMs, but you still may want to manage the various instances as a single resource, or cluster. You can use Kubernetes Engine for that.

Kubernetes is an open source tool created by Google for administering clusters of virtual and bare-metal machines. (Kubernetes is sometimes abbreviated K8s.) Kubernetes is a container orchestration service that helps you. It allows you to do the following:

- Create clusters of VMs or bare metal machines that run the Kubernetes orchestration software for containers.
- Deploy containerized applications to the cluster.
- Administer the cluster.
- Specify policies, such as autoscaling.
- Monitor cluster health.

Kubernetes Engine is Google Cloud's managed Kubernetes service. If you wanted, you could deploy a set of VMs, install Kubernetes on your VMs, and manage the Kubernetes platform yourself. With Kubernetes Engine you get the benefits of Kubernetes without the administrative overhead.

Kubernetes Engine supports two modes: GKE Standard and GKE Autopilot. With GKE Standard, you pay-per-node for resources in your GKE cluster, and you are responsible for configuring and managing nodes. With GKE Autopilot you pay per pod, which is a single unit of resources for providing a service, and GKE manages configuration and infrastructure.

Kubernetes Functionality

Kubernetes is designed to support clusters that run a variety of applications. This is different from other cluster management platforms that provide a way to run one application over multiple servers. Spark, for example, is a big data analytics platform that runs Spark services on a cluster of servers. Spark is not a general-purpose cluster management platform like Kubernetes.

Kubernetes Engine provides the following functions:

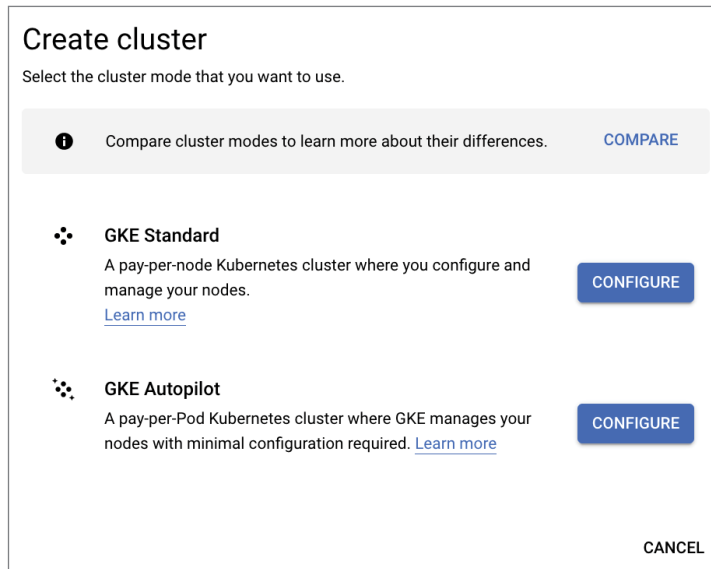
- Load balancing across Compute Engine VMs that are deployed in a Kubernetes cluster
- Automatic scaling of nodes (VMs) in the cluster
- Automatic upgrading of cluster software as needed
- Node monitoring and health repair
- Logging
- Support for node pools, which are collections of nodes all with the same configuration

Kubernetes Cluster Architecture

A Kubernetes cluster includes a cluster control plane and one or more worker nodes.

The control plane manages the cluster. Cluster services, such as the Kubernetes API server, resource controllers, and schedulers, run on the control plane. The Kubernetes API Server is the coordinator for all communications to the cluster. The control plane determines what containers and workloads are run on each node.

FIGURE 4.19 Kubernetes Engine supports clusters that you can manage using Standard mode, or you can have Kubernetes Engine manage many of your cluster operations using Autopilot mode.



When a Kubernetes cluster is created from either Google Cloud Console or a command line, a number of nodes are created as well. These are Compute Engine VMs, and you can specify the machine type when creating the cluster.

Kubernetes deploys containers in abstract compute units known as *pods*. They often have a single container but may have more than one. Containers within a single pod share storage and network resources. Containers within a pod share an IP address and port space. Containers are deployed and scaled as a unit.

Kubernetes Engine Use Cases

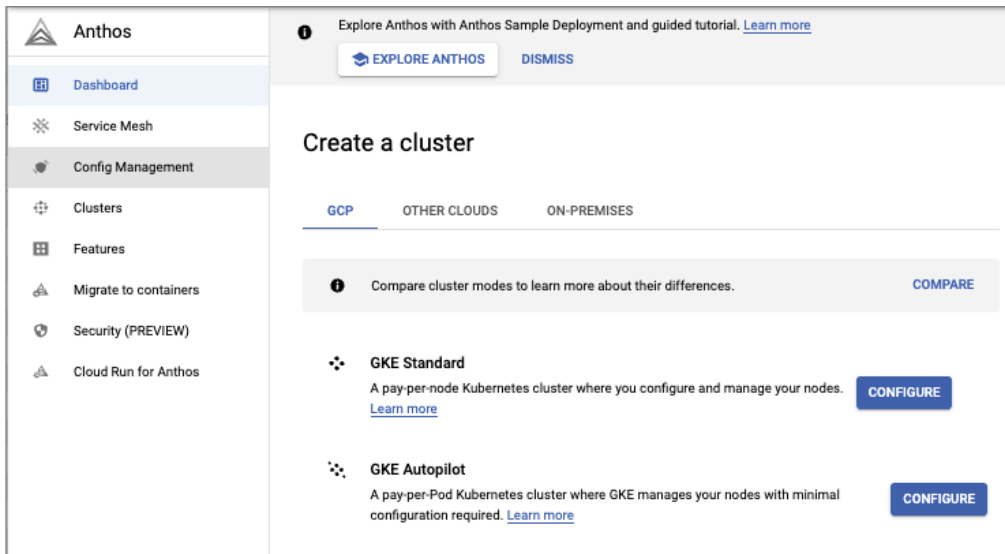
Kubernetes Engine is a good choice for large-scale applications that require high availability and high reliability. Kubernetes Engine supports the concept of pods and deployment sets, which allow application developers and administrators to manage services as a logical unit. This can help if you have a set of services that support a user interface, another set that implements business logic, and a third set that provides back-end services. Each of these different groups of services can have different life cycles and scalability requirements. Kubernetes helps to manage these at levels of abstraction that make sense for users, developers, and DevOps professionals.

Anthos

Anthos is not a compute service, like Compute Engine or Kubernetes Engine, but it is an increasingly important service that is used to managed services and resources across clouds and on-premises environments.

Anthos is a managed service for centrally configuring and managing the way you deploy services. (See Figure 4.20.) With Anthos, you can manage multiple GKE clusters running on virtual machines as well as bare-metal servers. Anthos can manage clusters running in other clouds and on-premises as well.

FIGURE 4.20 Anthos supports the management of Kubernetes clusters in Google Cloud, other clouds, and on-premises.



One of the advantages of using Anthos is that you can define and enforce policies across environments. Anthos Service Mesh is a service for managing complex microservices architectures and consistently securing and monitoring services running in Kubernetes.

Cloud Run

Cloud Run is a managed service for running containers. Specifically, Cloud Run is used to deploy stateless containers. By *stateless*, we mean that any instance of a container running a service can respond to requests from that service. No data is maintained in a service about a particular connection or user of the service.

Cloud Run, like App Engine, is a managed service for running containers. (See Figure 4.21) When you deploy a service to Cloud Run, you specify a container image, a service name, a

region, CPU allocation configuration, autoscaling parameters, as well as traffic configuration and authentication information.

Cloud Run Use Cases

The key thing to keep in mind when using Cloud Run is that the service runs containers. This puts it in group with Kubernetes Engine and App Engine, which also run containers. Cloud Run does not provide virtual machines; those are provided by Compute Engine.

If you are primarily interested in running your code in containers and do not want to manage infrastructure, then Cloud Run is the recommended option if your application is stateless.

Cloud Functions

Cloud Functions is a serverless computing platform designed to run single-purpose pieces of code in response to events in the Google Cloud environment. There is no need to provision or manage VMs, containers, or clusters when using Cloud Functions. Code that is written in Node.js, Python, Go, Java, .NET, Ruby, and PHP can be run on Cloud Functions. See the Cloud Functions documentation for information on supported versions of these languages.

Cloud Functions is not a general-purpose computing platform like Compute Engine or App Engine. Cloud Functions provides the “glue” between services that are otherwise independent.

For example, one service may create a file and upload it to Cloud Storage, and another service has to pick up those files and perform some processing on the file. Both services can be developed independently. There is no need for either to know about the other. However, you will need some way to detect that a new file has been written to Cloud Storage, and then the other application can begin processing it.

We don’t want to write applications in ways that make assumptions about other processes that may provide input or consume output. Services can change independently of each other. We should not have to keep track of dependencies between services if we can avoid it. Cloud Functions helps us avoid that situation.

Cloud Functions Execution Environment

Google Cloud manages everything that is needed to execute your code in a secure, isolated environment. Of course, below the serverless abstraction, there are virtual and physical servers running your code, but you as a cloud engineer do not have to administer any of that infrastructure. Three key things to remember about Cloud Functions are the following:

- The functions execute in a secure, isolated execution environment.
- Compute resources scale as needed to run as many instances of Cloud Functions as needed without you having to do anything to control scaling.
- The execution of one function is independent of all others. The life cycles of Cloud Functions are not dependent on each other.

FIGURE 4.21 When deploying an application to Cloud Run, you will specify a container, a location to run the container, and a minimal set of configuration parameters.

Cloud Run

Create service

A service exposes a unique endpoint and automatically scales the underlying infrastructure to handle incoming requests. Service name and region cannot be changed later.

☒ Deploy one revision from an existing container image

Container image URL *

SELECT

[TEST WITH A SAMPLE CONTAINER](#)

Should listen for HTTP requests on \$PORT and not rely on local state. [How to build a container?](#)

☐ Continuously deploy new revisions from a source repository

Service name *

Region *

us-central1 (Iowa)

[How to pick a region?](#)

CPU allocation and pricing

☒ CPU is only allocated during request processing
You are charged per request and only when the container instance processes a request.

☐ CPU is always allocated
You are charged for the entire lifecycle of the container instance.

Autoscaling

Minimum number of instances *

0

Maximum number of instances

100

Set to 1 to reduce cold starts. [Learn more](#)

Ingress

☒ Allow all traffic

☐ Allow internal traffic and traffic from Cloud Load Balancing

☐ Allow internal traffic only

Authentication *

☐ Allow unauthenticated invocations
Check this if you are creating a public API or website.

☐ Require authentication
Manage authorized users with Cloud IAM.

Container, Variables & Secrets, Connections, Security

CREATE

CANCEL

There is an important corollary to these key points: Cloud Functions may be running in multiple instances at one time. If two mobile app users uploaded an image file for processing at the same time, two different instances of Cloud Functions would execute at roughly the same time. You do not have to do anything to prevent conflicts between the two instances; they are independent.

Since each invocation of a Cloud Function runs in a separate instance, functions do not share memory or variables. In general, this means that Cloud Functions should be stateless. That means the function does not depend on the state of memory to compute its output. This is a reasonable constraint in many cases, but sometimes you can optimize processing if you can save state between invocations. Cloud Functions does offer some ways of doing this, which will be described in Chapter 11, “Planning Storage in the Cloud.”

Cloud Functions Use Cases

Cloud Functions is well suited to short-running, event-based processing. If your workflows upload, modify, or otherwise alter files in Cloud Storage or use message queues to send work between services, then the Cloud Functions service is a good option for running code that starts the next step in processing. Some application areas that fit this pattern include the following:

- Internet of Things (IoT), in which a sensor or other device can send information about the state of a sensor. Depending on the values sent, Cloud Functions could trigger an alert or start processing data that was uploaded from the sensor.
- Mobile applications that, like IoT apps, send data to the cloud for processing.
- Asynchronous workflows in which each step starts at some time after the previous steps completes, but there are no assumptions about when the processing steps will complete.

As with other serverless compute options, when using Cloud Functions, you specify parameters about your service, in this case a function, and do not need to concern yourself with the underlying infrastructure (see Figure 4.22).

Summary

Google Cloud offers several computing options. The options vary in the level of control that you, as a user of Google Cloud, have over the computing platform. Generally, with more control comes more responsibility and management overhead. Your objective when choosing a computing platform is to choose one that meets your requirements while minimizing DevOps overhead and cost.

Compute Engine is the Google Cloud service that lets you provision VMs. You can choose from predefined configurations, or you can create a custom configuration with the best combination of virtual CPUs and memory for your needs. If you can tolerate some disruption in VM functioning, you can save a significant amount of money by using pre-emptible VMs.

FIGURE 4.22 Configuring a Cloud Function

Cloud Functions | Create function

1 Configuration — 2 Code

Basics

Environment
1st gen

Function name *
function-1

Region
us-central1

Trigger

HTTP

Trigger type
HTTP

URL
https://us-central1-scenic-energy-335022.cloudfunctions.net/function-1

Authentication

☐ Allow unauthenticated invocations
Check this if you are creating a public API or website.

☒ Require authentication
Manage authorized users with Cloud IAM.

☒ Require HTTPS

SAVE CANCEL

Modern software applications are built on multiple services that may have different computing requirements and change on different life cycles. Kubernetes Engine runs clusters of servers that can be used to run a variety of services while efficiently allocating work to servers as needed. Kubernetes Engine also provides monitoring, scaling, and remediation when something goes wrong with a VM in the cluster.

As enterprises adopt Kubernetes and run multiple clusters, they can turn to Anthos for managing Kubernetes clusters in Google Cloud, other clouds, and on-premises.

Cloud Run is a managed service for running stateless containers. If you do not need the full functionality and feature-richness of Kubernetes Engine, Cloud Run is a good option for deploying stateless containers.

Loosely coupled applications may be strung together to implement complex workflows. Often, we want each component to be independent of others. In such cases, we often need to execute “glue” code that moves workload from one stage to another. Cloud Functions is the serverless computing option designed to meet this need.

Exam Essentials

Understand how images are used to create instances of VMs and how VMs are organized in projects. Instances run images, which contain operating systems, libraries, and other code. When you create an instance, you specify a project to contain the instance.

Know that Google Cloud has multiple geographic regions and regions have one or more zones. VMs run in zones. A region is a geographical location, such as asia-east1, europe-west2, and us-east4. The zones within a region are linked by low-latency, high-bandwidth network connections.

Understand what preemptible VMs are and when they are appropriate to use. Also understand when *not* to use them. Google Cloud offers an option called a preemptible VM for workloads that can be disrupted without creating problems.

Understand the difference between the App Engine standard and flexible environments. The standard environment runs a language-specific platform, and the App Engine flexible environment allows you to run custom containers. App Engine is well suited for HTTP(S)-based applications.

Know that Kubernetes is a container orchestration platform. It also runs containers in a cluster.

Understand Kubernetes. It provides load balancing, automatic scaling, logging, and node health checks and repair. Also know that Anthos is used to manage multiple Kubernetes clusters across Google Cloud, other clouds, and on-premises.

Understand Cloud Run. Cloud Run is a managed service for running stateless containers and is a good option when you do not need the full functionality of Kubernetes Engine.

Understand Cloud Functions. This service is used to run programs in response to events, such as file upload or a message being added to a queue.

Review Questions

You can find the answers in the Appendix.

1. You are deploying a Python web application to Google Cloud. The application uses only custom code and basic Python libraries. You expect to have sporadic use of the application for the foreseeable future and want to minimize both the cost of running the application and the DevOps overhead of managing the application. Which computing service is the best option for running the application?
 - A. Compute Engine
 - B. App Engine standard environment
 - C. App Engine flexible environment
 - D. Kubernetes Engine
2. Your manager is concerned about the rate at which the department is spending on cloud services. You suggest that your team use preemptible VMs for all of the following except which one?
 - A. Database server
 - B. Batch processing with no fixed time requirement to complete
 - C. High-performance computing cluster
 - D. None of the above
3. What parameters need to be specified when creating a VM in Compute Engine?
 - A. Project and zone
 - B. Username and admin role
 - C. Billing account
 - D. Cloud Storage bucket
4. Your company has licensed a third-party software package that runs on Linux. You will run multiple instances of the software in Docker containers. Which of the following Google Cloud services could you use to deploy this software package?
 - A. Compute Engine only
 - B. Kubernetes Engine only
 - C. Compute Engine, Kubernetes Engine, and the App Engine flexible environment only
 - D. Compute Engine, Kubernetes Engine, the App Engine flexible environment, or the App Engine standard environment
5. You can specify packages to install into a Docker container by including commands in which file?
 - A. `Docker.cfg`
 - B. `Dockerfile`
 - C. `Config.dck`
 - D. `install.cfg`

6. Which of the following could be managed using Anthos?
 - A. Kubernetes clusters in Google Cloud only
 - B. App Engine Flexible containers and Kubernetes clusters in Google Cloud
 - C. App Engine Flexible containers, Cloud Functions, and Kubernetes clusters in Google Cloud
 - D. Kubernetes clusters in Google Cloud, AWS, and on-premises
7. Your manager is making a presentation to executives in your company advocating that you start using Kubernetes Engine. You suggest that the manager highlight all the features Kubernetes provides to reduce the workload on DevOps engineers. You describe several features, including all of the following except which one?
 - A. Load balancing across Compute Engine VMs that are deployed in a Kubernetes cluster
 - B. Security scanning for vulnerabilities
 - C. Automatic scaling of nodes in the cluster
 - D. Automatic upgrading of cluster software as needed
8. Your company is about to release an online service that builds on a new user interface experience driven by a set of services that will run on your servers. A separate set of services manage authentication and authorization. A third set of services keeps track of account information. All three sets of services must be highly reliable and scale to meet demand. Which of the Google Cloud services is the best option for deploying this?
 - A. App Engine standard environment
 - B. Compute Engine
 - C. Cloud Functions
 - D. Kubernetes Engine
9. A mobile application uploads images for analysis, including identifying objects in the image and extracting text that may be embedded in the image. A third party has created the mobile application, and you have developed the image analysis service. You both agree to use Cloud Storage to store images. You want to keep the two services completely decoupled, but you need a way to invoke the image analysis as soon as an image is uploaded. How should this be done?
 - A. Change the mobile app to start a VM running the image analysis service and have that VM copy the file from storage into local storage on the VM. Have the image service run on the VM.
 - B. Write a function in Python that is invoked by Cloud Functions when a new image file is written to the Cloud Storage bucket that receives new images. The function should submit the URL of the uploaded file to the image analysis service. The image analysis service will then load the image from Cloud Storage, perform analysis, and generate results, which can be saved to Cloud Storage.
 - C. Have a Kubernetes cluster running continuously, with one pod dedicated to listing the contents of the upload bucket and detecting new files in Cloud Storage and another pod dedicated to running the image analysis software.
 - D. Have a Compute Engine VM running and continuously listing the contents of the upload bucket in Cloud Storage to detect new files. Another VM should be continually running the image analysis software.

10. Your team is developing a new pipeline to analyze a stream of data from sensors on manufacturing devices. The old pipeline occasionally corrupted data because parallel threads overwrote data written by other threads. You decide to use Cloud Functions as part of the pipeline. As a developer of a Cloud Function, what do you have to do to prevent multiple invocations of the function from interfering with each other?
- A. Include a check in the code to ensure another invocation is not running at the same time.
 - B. Schedule each invocation to run in a separate process.
 - C. Schedule each invocation to run in a separate thread.
 - D. Nothing. Google Cloud ensures that function invocations do not interfere with each other.
11. A client of yours processes personal and health information for hospitals. All health information needs to be protected according to government regulations. Your client wants to move their application to Google Cloud but wants to use the encryption library that they have used in the past. You suggest that all VMs running the application have the encryption library installed. Which kind of image would you use for that?
- A. Custom image
 - B. Public image
 - C. CentOS 6 or 7
 - D. Ubuntu 18 or later
12. What is the lowest level of the resource hierarchy?
- A. Folder
 - B. Project
 - C. File
 - D. VM instance
13. Your company is seeing a marked increase in the rate of customer growth in Europe. Latency is becoming an issue because your application is running in us-central1. You suggest deploying your services to a region in Europe. You have several choices. You should consider all the following factors except which one?
- A. Cost
 - B. Latency
 - C. Regulations
 - D. Reliability
14. What role gives users full control over Compute Engine instances?
- A. Compute Manager role
 - B. Compute Admin role
 - C. Compute Regional Manager role
 - D. Compute Security Admin

15. Which of the following are limitations of a preemptible VM?
- A. Will be terminated within 24 hours.
 - B. May not always be available. Availability may vary across zones and regions.
 - C. Cannot migrate to a regular VM.
 - D. All of the above.
16. Which of the following would eliminate Cloud Run as an option for deploying an application in Google Cloud?
- A. The application uses a mix of Java and Python application code.
 - B. The application stores data about a session in memory for use across multiple requests during a session.
 - C. The application runs in a container.
 - D. The container configuration is specified in a Dockerfile.
17. When using the App Engine standard environment, which of the following languages' runtime is not supported?
- A. Java
 - B. Python
 - C. C
 - D. Go
18. You want to be sure all services running in a Kubernetes Engine cluster use the same authentication and monitoring services. What service would you use?
- A. Cloud Functions
 - B. Anthos Service Mesh
 - C. App Engine Flexible
 - D. App Engine Standard
19. You are deploying a set of virtual machines in Compute Engine. You want to ensure that malware does not compromise the operating system, so you want to validate boot integrity. What feature of Compute Engine would you enable?
- A. Customer-supplied encryption keys
 - B. vTPM
 - C. Sole tenancy
 - D. Identity and access management roles
20. A client has brought you in to help reduce their DevOps overhead. Engineers are spending too much time patching servers and optimizing server utilization. They want to move to serverless platforms as much as possible. Your client has heard of Cloud Functions and wants to use them. You recommend all the following types of applications except which one?
- A. Long-running data warehouse data load procedures
 - B. IoT back-end processing
 - C. Mobile application event processing
 - D. Asynchronous workflows

Chapter 5

Computing with Compute Engine Virtual Machines

THIS CHAPTER COVERS THE FOLLOWING OBJECTIVES OF THE GOOGLE ASSOCIATE CLOUD ENGINEER CERTIFICATION EXAM:

- ✓ 1.3 Installing and configuring the command line interface (CLI), specifically Cloud SDK (e.g., setting the default project)
- ✓ 2.2 Planning and configuring compute resources.
Considerations include:
 - Selecting appropriate compute choices for a given workload (e.g., Compute Engine, Google Kubernetes Engine, Cloud Run, Cloud Functions)
 - Using preemptible VMs and custom machine types as appropriate





In this chapter, you will learn about Google Cloud Console, a graphical user interface for working with Google Cloud. You will learn how to install Google Cloud SDK and use it to create virtual machine instances and how to use Cloud Shell as an alternative to installing Google Cloud SDK locally.

Creating and Configuring Virtual Machines with the Console

Let's create a VM in Compute Engine. We have three options for doing this: we can use Google Cloud Console, the Google Cloud Software Development Kit (SDK), or Google Cloud Shell. Let's start with the console.

Google Cloud Console is a web-based graphical user interface (GUI) for creating, configuring, and managing resources in Google Cloud. In this chapter, we will use it to create a VM.

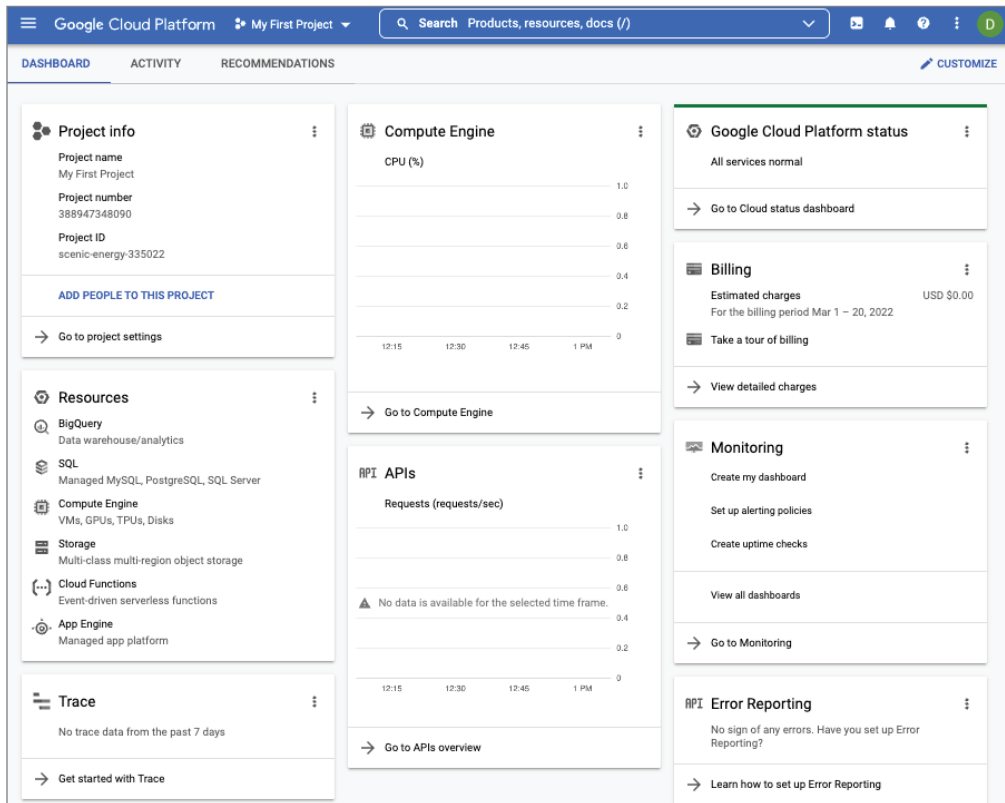
To open the console, navigate in your browser to <https://console.cloud.google.com> and log in. Figure 5.1 shows an example of the main form in the console.

In the console, Select A Project option to display the existing projects. You can also create a new project from this form, as shown in Figure 5.2.

After you select an existing project or create a new project, you can return to the main console panel. The first time you try to work with a VM you will have to create a billing account if one has not already been created.

Click Enable Billing if prompted and fill in the billing information, such as name, address, and credit card. Once billing is enabled, you can navigate to the Compute Engine console page (see Figure 5.3).

Click the Create Instance button at the top of the panel to bring up a VM configuration, as shown in Figure 5.4.

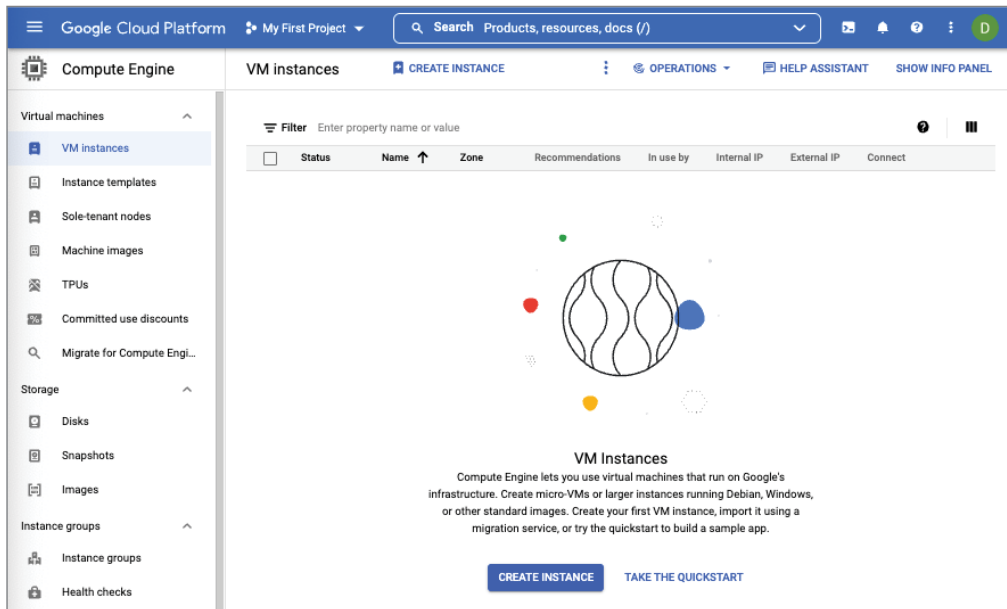
FIGURE 5.1 The main starting form of Google Cloud Console**FIGURE 5.2** The Project form lets you choose the project you want to work with when creating VMs. You can also create a new project here.

Select from **SULLIVANLEARNINGGROUP.COM** NEW PROJECT

Search projects and folders

RECENT STARRED ALL

Name	ID
✓ ☆ My First Project ?	scenic-energy-335022
🏠 sullivanlearninggroup.com ?	470324492486

FIGURE 5.3 The starting panel for creating a VM

Main Virtual Machine Configuration Details

Within the console, you can specify all the needed details about the configuration of the VM that you are creating, including the following:

- Name of the VM
- Region and zone where the VM will run
- Machine type, which determines the number of CPUs and the amount of memory in the VM
- Boot disk, which includes the operating system the VM will run

You can choose the name of your VM. This is primarily for your use. Google Cloud uses other identifiers internally to manage VMs.

You will need to specify a region. Regions are major geographical areas. A partial list of regions is shown in Figure 5.5.

After you select a region, you can select a zone. Remember, a zone is a data center–like facility within a region. Figure 5.6 shows an example list of zones available in the us-east-1 region.

FIGURE 5.4 Part of the main configuration form for creating VMs in Compute Engine

Name *
instance-1

?

Labels ?
[+ ADD LABELS](#)

Region *
us-central1 (Iowa) ?
Region is permanent

Zone *
us-central1-a ?
Zone is permanent

Monthly estimate
\$25.46
That's about \$0.03 hourly
Pay for what you use: No upfront costs
and per second billing

[▼ DETAILS](#)

Machine configuration

Machine family

[GENERAL-PURPOSE](#) [COMPUTE-OPTIMIZED](#) [MEMORY-OPTIMIZED](#) [GPU](#)

Machine types for common workloads, optimized for cost and flexibility

Series
E2

▼

CPU platform selection based on availability

Machine type
e2-medium (2 vCPU, 4 GB memory)

▼



vCPU
1 shared core

Memory
4 GB

[▼ CPU PLATFORM AND GPU](#)

Display device

Enable to use screen capturing and recording tools.

☐ Enable display device

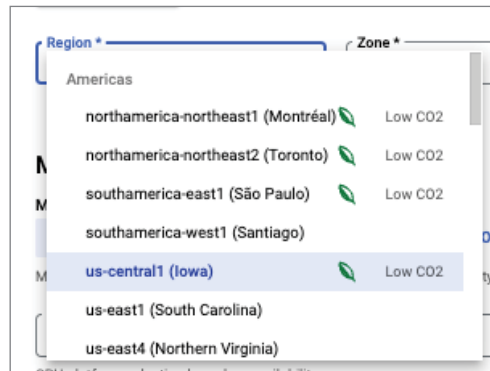
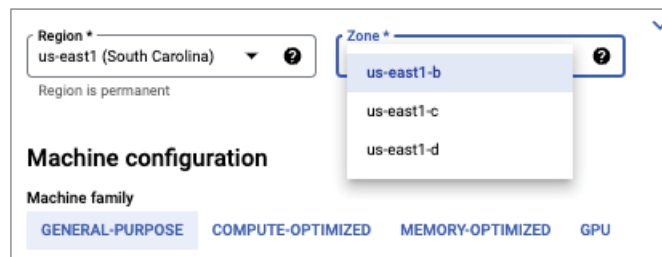
Confidential VM service ?

☐ Enable the Confidential Computing service on this VM instance.

Container ?

Deploy a container image to this VM instance

[DEPLOY CONTAINER](#)

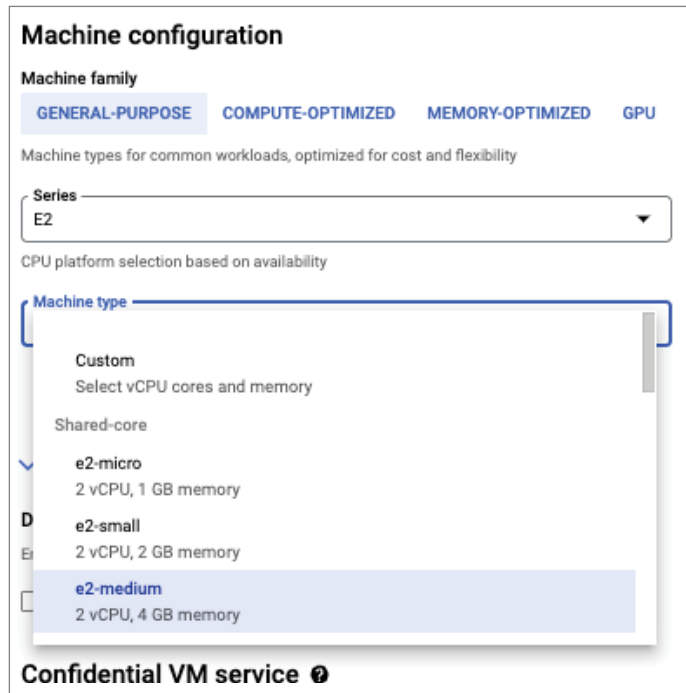
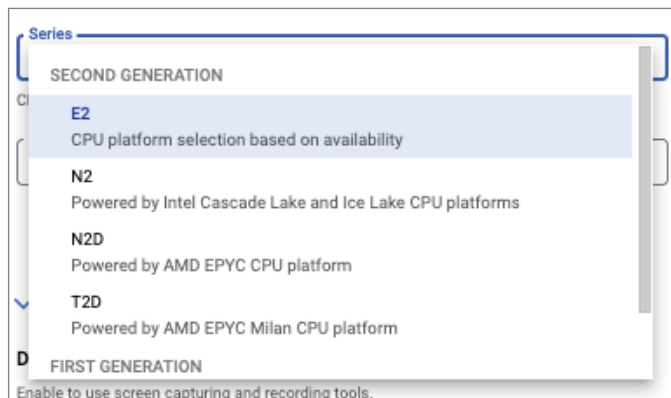
FIGURE 5.5 A partial list of regions providing Compute Engine services**FIGURE 5.6** A list of zones within the us-east1 region

After you specify a region and a zone, Google Cloud can determine the VMs available in that zone. Not all zones have the same availability. Figure 5.7 shows an example listing of machine types available for the E2 series in the us-east1-b zone.

Google Cloud organizes virtual machines into machine families, series, and machine type. A machine family is a set of processor and hardware configurations designed for particular workloads, such as general purpose, compute optimized, and memory optimized. Within a family, machines are organized into series and generation, as shown in Figure 5.8.

Within a series, you will have the option of one or more machine types, which vary based on the number of virtual CPUs and the amount of memory.

For applications and services requiring high security, you can enable Confidential VM Service, which keeps data in memory encrypted using encryption keys that Google does not have access to.

FIGURE 5.7 A partial list of machine types available in the us-east1-b zone**FIGURE 5.8** Virtual machines within a machine family are further organized into series and generations based on the type of processor.

You have the option of choosing to run a container in your virtual machine. If you decide to do that, you must specify a container that is in a public repository or Google Container Registry. This can be useful if you want to run a container with specialized software or some custom configuration.

The Boot Disk section lists a default configuration. Clicking the Change button brings up the Boot Disk form, as shown in Figure 5.9.

FIGURE 5.9 Form for configuring the boot disk of the VM

Boot disk

Select an image or snapshot to create a boot disk; or attach an existing disk. Can't find what you're looking for? Explore hundreds of VM solutions in [Marketplace](#)

PUBLIC IMAGES CUSTOM IMAGES SNAPSHOTS EXISTING DISKS

Operating system
Debian

Version *
Debian GNU/Linux 10 (buster)

amd64 built on 20220317, supports Shielded VM features

Boot disk type *
Balanced persistent disk

Size (GB) *
10

[SHOW ADVANCED CONFIGURATION](#)

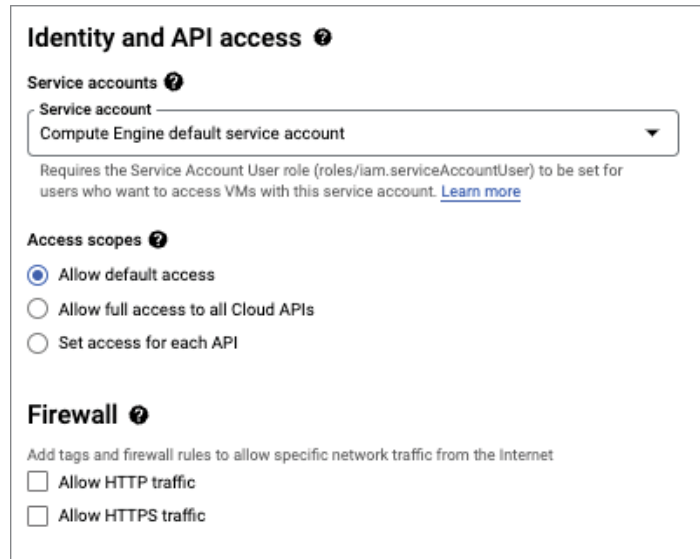
SELECT **CANCEL**

Here you can choose the operating system you want to use. You can also choose the boot disk type: Balanced Persistent Disk, Extreme Persistent Disk, SSD Persistent Disk, or Standard Persistent Disk. You can also specify the size of the disk.

- Balanced persistent disks use solid-state drives (SSDs) and balance cost and performance.
- Extreme persistent disks use SSDs but provide high performance and allow you to provision your desired level of input-output operations per second (IOPS).
- SSD persistent disks use solid-state drives.
- Standard persistent disks use standard hard disk drives (HDDs).

Following the Boot Disk section is the Identity And API Access section (see Figure 5.10). Here you can specify a service account for the VM and set the scope of API access. If you want the processes running on this VM to use only some APIs, you can use these options to limit the VM's access to specific APIs.

FIGURE 5.10 Identity And API Access and Firewall configurations



The screenshot shows the 'Identity and API access' configuration panel. It has a title 'Identity and API access' with a help icon. Below the title is a section 'Service accounts' with a help icon. Inside this section is a dropdown menu labeled 'Service account' with the selected option 'Compute Engine default service account'. Below the dropdown is a note: 'Requires the Service Account User role (roles/iam.serviceAccountUser) to be set for users who want to access VMs with this service account. [Learn more](#)'. Below the 'Service accounts' section is a section 'Access scopes' with a help icon. It contains three radio button options: 'Allow default access' (which is selected), 'Allow full access to all Cloud APIs', and 'Set access for each API'. Below the 'Access scopes' section is a section 'Firewall' with a help icon. It contains a note: 'Add tags and firewall rules to allow specific network traffic from the Internet'. Below the note are two checkbox options: 'Allow HTTP traffic' and 'Allow HTTPS traffic', both of which are currently unchecked.

In the next section, you can select whether you want the VM to accept HTTP or HTTPS traffic.

Advanced Configuration Details

Click Management, Security, Disks, Networking, and Sole Tenancy to display advanced configuration options. This will expand a list of advanced configuration options.

Management Tab

The Management tab of the form (Figure 5.11) provides a space where you can describe the VM and its use. You can also create labels, which are key-value pairs. You can assign any label you like. Labels and a general description are often used to help manage your VMs and illustrate how they are being used. Labels are particularly important when your number of servers grows. It is a best practice to include a description and labels for all VMs.

FIGURE 5.11 The first part of the Management tab of the VM creation form

Management

Description, deletion protection, reservations, automation, and availability policies

Description

Deletion protection ⓘ
☐ Enable deletion protection

Reservation name
Automatically use created reservation
Use an existing reservation when creating this VM instance

Automation

Startup script

You can choose to specify a startup script that will run when your instance boots up or restarts. Startup scripts can be used to install software and updates, and to ensure that services are running within the virtual machine. [Learn more](#)

Metadata
You can set custom metadata for an instance or project outside of the server-defined metadata. This is useful for passing in arbitrary values to your project or instance that can be queried by your code on the instance. [Learn more](#)

+ ADD ITEM

Availability policy

Preemptibility
Off (Recommended)

A preemptible VM costs much less, but lasts only 24 hours. It can be terminated sooner due to system demands. [Learn more](#)

On host maintenance
Migrate VM instance (Recommended)

When Compute Engine performs periodic infrastructure maintenance it can migrate your VM instances to other hardware without downtime

Automatic restart
On (recommended)

Compute Engine can automatically restart VM instances if they are terminated for non-user-initiated reasons (maintenance event, hardware failure, software failure and so on)

If you want to force an extra confirmation before deleting an instance, you can select the Deletion Protection option. If someone tries to delete the instance, the operation will fail.

If you have reserved Compute Engine instance resources, they will be automatically used, but you can indicate that reservations should not be used for a particular instance.

You can specify a startup script to run when the instance starts. Copy the contents of the startup script to the Automation text box. For example, you could paste a Bash or Python script directly into the text box.

The Metadata section allows you to specify key-value pairs associated with the instance. These values are stored in a metadata server, which is available for querying using the Compute Engine API. Metadata tags are especially useful if you have a common script you want to run on startup or shutdown but want the behavior of the script to vary according to some metadata values.

Under Availability Policy are three drop-down menus:

- VM Provisioning Model, which is standard or spot. Spot allows Google to shut down the server with a 30-second notice. In return, the cost of a preemptible server is much lower than that of a nonpreemptible server.
- On Host Maintenance, which indicates whether the virtual server should be migrated to another physical server when a maintenance event occurs.
- Automatic Restart, which indicates if the server stops because of a hardware failure, maintenance event, or some other non-user-controlled event.

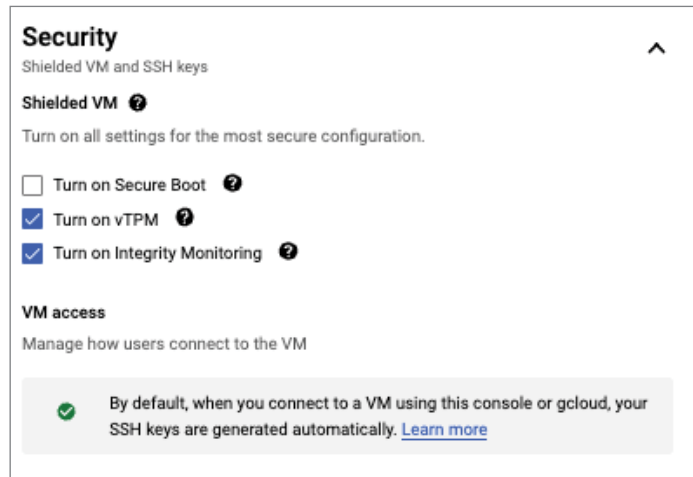
Security Tab

On the Security tab, you can specify whether you want to use Shielded VMs and Secure Shell (SSH) keys.

Shielded VMs are configured to have additional security mechanisms that you can choose to run (see Figure 5.12). These include the following:

- Secure Boot, which ensures that only authenticated operating system software runs on the VM. It does this by checking the digital signatures of the software. If a signature check fails, the boot process will halt.
- Virtual Trusted Platform Module (vTPM), which is a virtualized version of a Trusted Platform Module (TPM). A TPM is a specialized computer chip designed to protect security resources, like keys and certificates.
- Integrity Monitoring, which uses a known good baseline of boot measurements to compare to recent boot measurements. If the check fails, that means some difference exists between the baseline measurement and the current measurements.

Google Cloud supports the concept of project-wide SSH keys, which are used to give users project-wide access to VMs. You can block that behavior at the VM if you use project-wide SSH keys and do not want all project users to have access to this machine.

FIGURE 5.12 You can place additional security controls on VMs.

Boot Disks and Additional Disks

In the Boot Disk tab of the Create Instance page, you can specify advanced configuration options, as shown in Figure 5.13. Under Deletion Rule, you can specify whether the boot disk should be deleted when the instance is deleted. You can also select how you would like to manage encryption keys for the boot disk. By default, Google manages those keys.

On the Disk configuration tab, you also have the option of adding a new disk or attaching an existing disk. Figure 5.14 shows the tab for adding a new disk.

When adding a new disk, the form in Figure 5.14 appears. Here, you specify a name and description and source information. The source specifies if you want to use a blank disk or create one using a snapshot or image. You also specify the disk size and type. If you want to automatically back up your disk, you can specify a snapshot schedule. By default, Google will manage encryption keys for the disk, but you can also specify customer-managed encryption keys (CMEKs) or customer-supplied encryption keys (CSEKs).

Adding an existing disk displays the form shown in Figure 5.15. Here you choose a disk from a list of existing disks and specify whether the disk should be attached as Read/Write or Read-Only. You can also specify whether the disk should be deleted when the VM is deleted. The default is to keep the disk. Finally, you can provide a custom disk name.

FIGURE 5.13 Boot disk advanced configuration

Boot disk

Select an image or snapshot to create a boot disk; or attach an existing disk. Can't find what you're looking for? Explore hundreds of VM solutions in [Marketplace](#)

PUBLIC IMAGES CUSTOM IMAGES SNAPSHOTS EXISTING DISKS

Operating system
Debian

Version *
Debian GNU/Linux 10 (buster)
amd64 built on 20220317, supports Shielded VM features

Boot disk type *
Balanced persistent disk

Size (GB) *
10

Deletion rule
When deleting instance
☐ Keep boot disk
☒ Delete boot disk

Encryption
Data is encrypted automatically. Select an encryption key management solution.
☒ Google-managed encryption key
No configuration required
☐ Customer-managed encryption key (CMEK)
Manage via Google Cloud Key Management Service
☐ Customer-supplied encryption key (CSEK)
Manage outside of Google Cloud

Snapshot schedule
Use snapshot schedules to automate disk backups. [Learn more](#)

Select a snapshot schedule

Device name ⓘ
Used to reference the device for mounting or resizing.
☐ Use a custom device name

Device name
instance-1
Based on instance name (default)

[^ HIDE ADVANCED CONFIGURATION](#)

FIGURE 5.14 Adding a new disk to a Compute Engine instance

Add new disk

Name *

disk-1

?

Name is permanent

Description

Source

Create a blank disk, apply a bootable disk image, or restore a snapshot of another disk in this project.

Disk source type *

Blank disk

Disk settings

Disk type *

Balanced persistent disk

?

COMPARE DISK TYPES

Size *

100

GB

?

Provision between 10 and 65,536 GB

Snapshot schedule (Recommended)

Use snapshot schedules to automate disk backups. [Learn more](#)

Select a snapshot schedule

Encryption

Data is encrypted automatically. Select an encryption key management solution.

☒ Google-managed encryption key

No configuration required

☐ Customer-managed encryption key (CMEK)

Manage via Google Cloud Key Management Service

☐ Customer-supplied encryption key (CSEK)

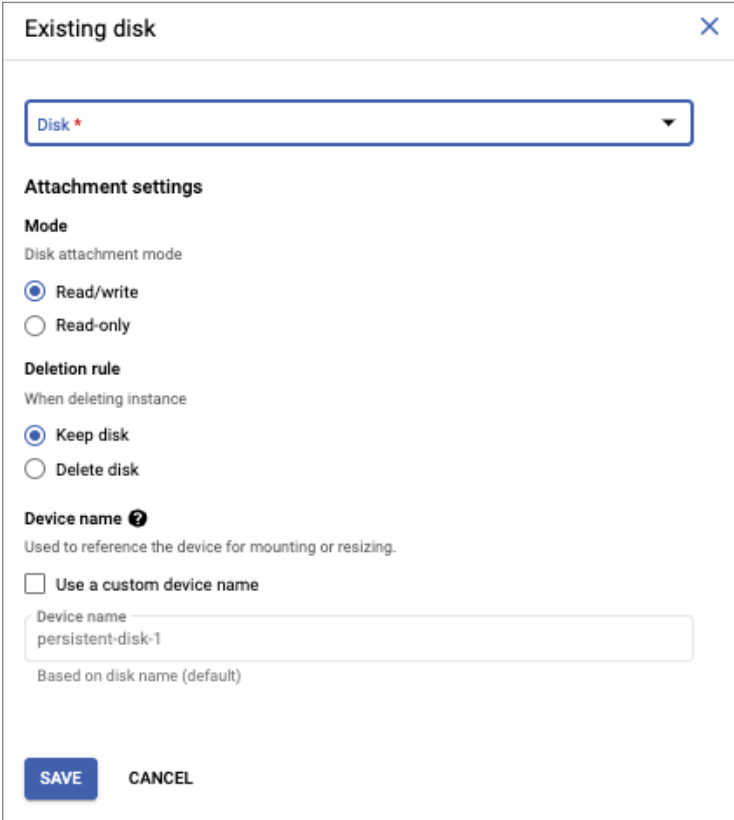
Manage outside of Google Cloud

Labels ?

+ ADD LABEL

SAVE

CANCEL

FIGURE 5.15 Form for adding an existing disk to a VM

The screenshot shows a modal window titled "Existing disk" with a close button (X) in the top right corner. Inside the modal, there is a dropdown menu labeled "Disk" with a red asterisk and a downward arrow. Below this, the "Attachment settings" section includes a "Mode" subsection with "Disk attachment mode" and two radio buttons: "Read/write" (selected) and "Read-only". The "Deletion rule" subsection, labeled "When deleting instance", has two radio buttons: "Keep disk" (selected) and "Delete disk". The "Device name" section, marked with a question mark icon, includes the text "Used to reference the device for mounting or resizing." and a checkbox labeled "Use a custom device name". Below the checkbox is a text input field containing "persistent-disk-1" and a note "Based on disk name (default)". At the bottom of the modal are two buttons: "SAVE" and "CANCEL".

Networking Tab

On the Networking tab, you can see the network interface information, including the IP address of the VM. If you have two networks, you have the option of adding another network interface to that other network. This use of dual network interfaces can be useful if you are running some type of proxy or server that acts as a control for the flow of some traffic between the networks. In addition, you can add network tags on this tab (see Figure 5.16).

Sole-Tenancy Tab

If you need to ensure that your VMs run on a server only with your other VMs, then you can specify *sole tenancy*. The Sole-Tenancy tab allows you to specify labels regarding sole tenancy for the server (see Figure 5.17).

FIGURE 5.16 Options for network configuration of a VM

Networking ^
Hostname and network interfaces

Network tags ?

Hostname ?
Set a custom hostname for this instance or leave it default. Choice is permanent

IP forwarding ?
☐ Enable

Network performance configuration
Network interface card ▾
—

Network bandwidth
☐ Increase total egress bandwidth
Maximum outbound network bandwidth: 1Gbps

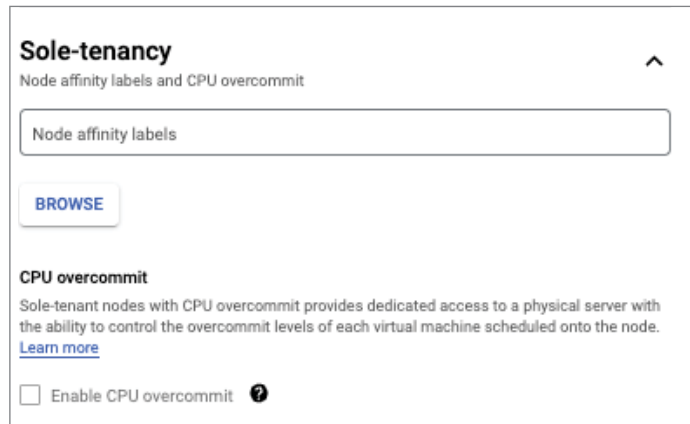
Network interfaces ?
Network interface is permanent

default default (10.142.0.0/20) ▾

ADD NETWORK INTERFACE

i To create another network interface you need to have a new network first.

Sole-tenant nodes can be configured to allow overcommitting CPU resources, but this is permitted only on machines with four or more CPUs. Node affinity labels are used to determine where a VM can run.

FIGURE 5.17 Sole tenancy configuration options

The screenshot shows a configuration panel titled "Sole-tenancy" with a subtitle "Node affinity labels and CPU overcommit". It features a text input field for "Node affinity labels" with a "BROWSE" button below it. The "CPU overcommit" section includes a description of sole-tenant nodes and a checkbox to "Enable CPU overcommit" with a help icon.

Sole-tenancy
Node affinity labels and CPU overcommit

Node affinity labels

BROWSE

CPU overcommit
Sole-tenant nodes with CPU overcommit provides dedicated access to a physical server with the ability to control the overcommit levels of each virtual machine scheduled onto the node.
[Learn more](#)

☐ Enable CPU overcommit ?

Creating and Configuring Virtual Machines with Cloud SDK

A second way to create and configure VMs is with Google Cloud SDK, which provides a command-line interface. (CLI) To use the Cloud SDK, you will first need to install it on your local device.

Installing Cloud SDK

You have three options for interacting with Google Cloud resources:

- Using a command-line interface
- Using a RESTful interface
- Using the Cloud Shell

Before using either of the first two options from your local system, you will need to install Cloud SDK on your machine. Cloud Console is a GUI you can access through a browser at <https://console.cloud.google.com>.

Cloud SDK can be installed on Linux, Windows, or Mac computers.

Installing Cloud SDK on Linux

If you are using Linux, you can install Cloud SDK using your operating system's package manager. Ubuntu and other Debian distributions use `apt-get` to install packages. Red Hat Enterprise, CentOS, and other Linux distributions use `yum`. For instructions on using `apt-get`, see <https://cloud.google.com/sdk/docs/install-sdk#deb>. For instructions on installing on Red Hat Enterprise or CentOS, see <https://cloud.google.com/sdk/docs/install-sdk#rpm>.

Cloud SDK on macOS

Instructions for installing on a Mac and the installation file for Cloud SDK are available at <https://cloud.google.com/sdk/docs/install-sdk#mac>. The first step is to verify that you have Python 3 installed. There are three versions of Cloud SDK, one for 32-bit macOS; one for 64-bit macOS running on x86 processors; and one for 64-bit macOS running on arm64, the Apple M1 processor.

Installing Cloud SDK on Windows

To install Cloud SDK on a Windows platform, you will need to download the appropriate installer. You can find instructions at <https://cloud.google.com/sdk/docs/install-sdk#windows>.

Example Installation on Ubuntu Linux

The first step in installing Cloud SDK is to get the appropriate version of the package for your operating system. The following commands are for installing Cloud SDK on Ubuntu. See <https://cloud.google.com/sdk/docs/install-sdk#deb> for any updates to this procedure.

The first step is adding the `gcloud` CLI URI as a source for packages:

```
echo "deb [signed-by=/usr/share/keyrings/cloud.google.gpg]
https://packages.cloud.google.com/apt cloud-sdk main" | sudo tee -a
/etc/apt/sources.list.d/google-cloud-sdk.list
```

You also need to import the Google Cloud public key, which you do with this command:

```
curl https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-
key --keyring /usr/share/keyrings/cloud.google.gpg add -
```

Finally, you need to update the `apt-get` package list and then use `apt-get` to install Cloud SDK:

```
sudo apt-get update && sudo apt-get install google-cloud-cli
```

Now that Cloud SDK is installed, you can execute commands using it. The first step is to initialize Cloud SDK using the `gcloud init` command, as shown here:

```
gcloud init
```

When you receive an authentication link, copy it into your browser. You are prompted to authenticate with Google when you go to that URL. Next, a response code appears in your browser. Copy that to your terminal window and paste it in response to the prompt that should appear.

Next, you are prompted to enter a project. If projects already exist in your account, they will be listed. You also have the option of creating a new project at this point. The project you select or create will be the default project used when issuing commands through Cloud SDK.

Creating a Virtual Machine with Cloud SDK

To create a VM from the command line, you will use the `gcloud` command. You use this command for many cloud management tasks, including working with the following services:

- Compute Engine
- Cloud SQL instances
- Kubernetes Engine
- Cloud Dataproc
- Cloud DNS
- Cloud Deployment Manager

The `gcloud` command is organized into a hierarchy of groups, such as the `compute` group for Compute Engine commands. We'll discuss other groups in later chapters; the focus here is on Compute Engine.

A typical `gcloud` command starts with the group, as shown here:

```
gcloud compute
```

A subgroup is used in Compute Engine commands to indicate what type of compute resource you are working with. To create an instance, you use this command:

```
gcloud compute instances
```

And the action you want to take is to create an instance, so you use this:

```
gcloud compute instances create ace-instance-1 ace-instance-2
```

If you do not specify additional parameters, such as a zone, Google Cloud will use your information from your default project. You can view your project information using the following `gcloud` command:

```
gcloud compute project-info describe
```

To create a VM in the us-central1-a zone, add the zone parameter like this:

```
gcloud compute instances create ace-instance-1 ace-instance-2 --  
zone=us-central1-a
```

You can list the VMs you've created using this:

```
gcloud compute instances list
```

The following are parameters commonly used with the `create instance` command:

- `--boot-disk-size` is the size of the boot disk for a new disk. Disk size may be between 10 GB and 2 TB.
- `--boot-disk-type` is the type of disk. Use `gcloud compute disk-types list` for a list of disk types available in the zone the VM is created in.
- `--labels` is the list of key-value pairs in the format of `KEY=VALUE`.
- `--machine-type` is the type of machine to use. If not specified, it uses `n1-standard-1`. Use `gcloud compute machine-types list` to view a list of machine types available in the zone you are using.
- `--preemptible`, if included, specifies that the VM will be preemptible.

For additional parameters, see the `gcloud compute instance create` documentation at <https://cloud.google.com/sdk/gcloud/reference/compute/instances/create>.

To create a standard VM with 8 CPUs and 30 GB of memory, you can specify `n1s8-standard-2` as the machine type:

```
gcloud compute instances create ace-instance-n1s8 --machine-type=e2-standard-2
```

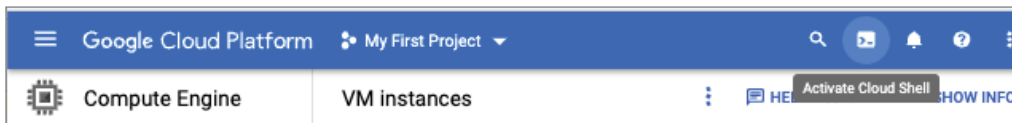
If you want to make this instance preemptible, you add the `preemptible` parameter:

```
gcloud compute instances create --machine-type=n1-  
standard-8 --preemptible ace-instance-1
```

Creating a Virtual Machine with Cloud Shell

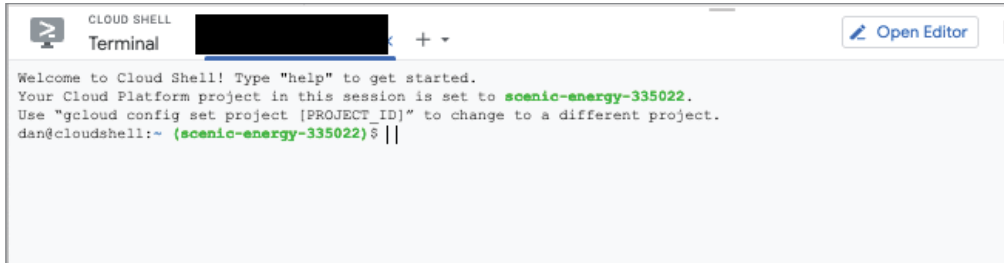
An alternative to running `gcloud` commands locally is to run them in a cloud instance. Cloud Shell provides this capability. To use Cloud Shell, start it from Cloud Console by clicking the shell icon in the upper-right corner of the browser, as shown in Figure 5.18.

FIGURE 5.18 Cloud Shell is activated through Cloud Console.



Cloud SDK is installed and Cloud Shell provides a Linux command line, as shown in Figure 5.19. All `gcloud` commands that you can enter on your local device with Cloud SDK installed can be used in Cloud Shell.

FIGURE 5.19 Cloud Shell opens a command-line window in the browser.



Basic Virtual Machine Management

When VMs are running, you can perform basic management tasks by using the console or by using `gcloud` commands.

Starting and Stopping Instances

In the console you view a list of instances by selecting Compute Engine and then VM Instances from the left-side panel of the console. You can then select a VM to operate on and list command options by clicking the ellipsis icons on the right. Figure 5.20 shows an example.

Note that you can start a stopped instance using the `start` command that is enabled in the pop-up for stopped instances.

You can also use `gcloud` to stop an instance with the following command, where *INSTANCE-NAME* is the name of the instance:

```
gcloud compute instances stop INSTANCE-NAME
```

Network Access to Virtual Machines

As a cloud engineer, you will sometimes need to log into a VM to perform some administration tasks. The most common way is to use SSH when logging into a Linux server or use Remote Desktop Protocol (RDP) when logging into a Windows server.

FIGURE 5.20 Basic operations on VMs can be performed using a pop-up menu in the console.

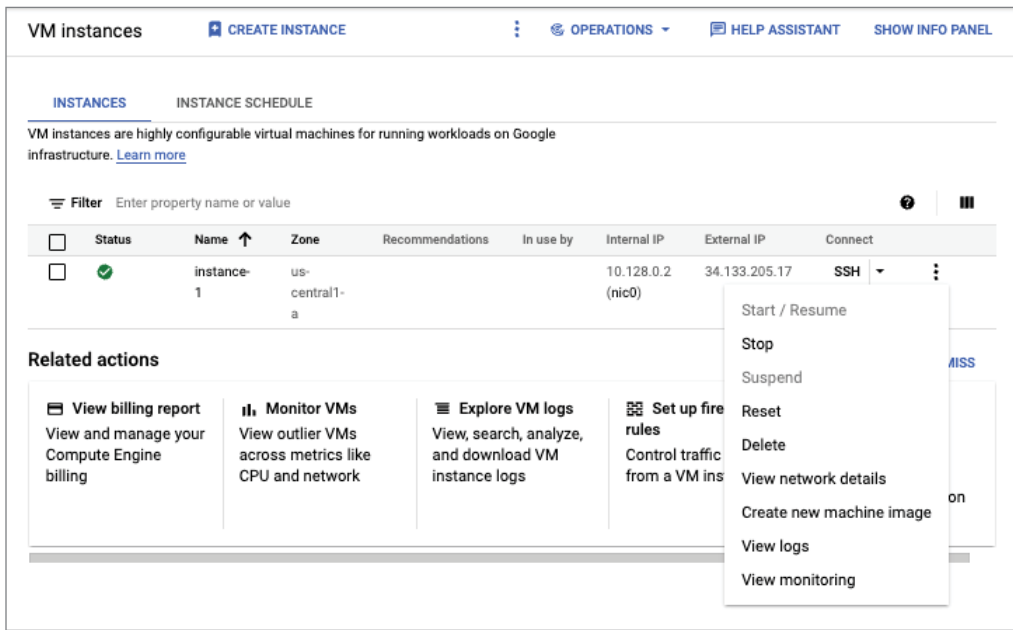
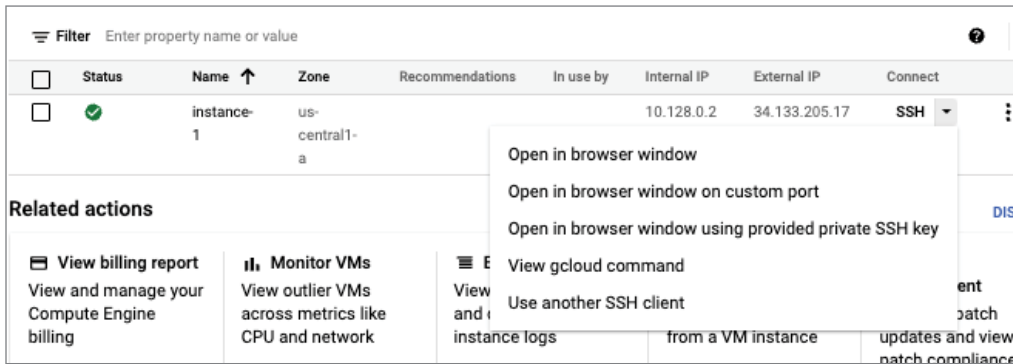


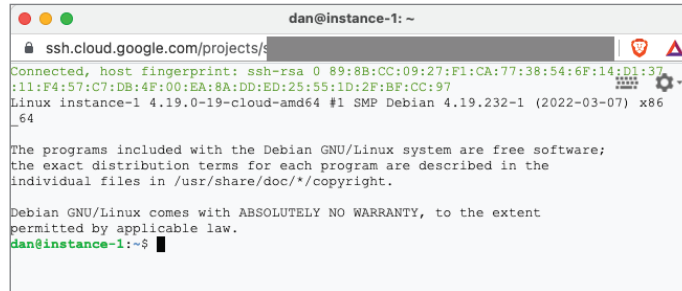
Figure 5.21 shows the set of options for using SSH from the console. This list of options appears when you click the SSH button associated with a VM.

FIGURE 5.21 From the console, you can start an SSH session to log into a Linux server.



Choosing the Open In Browser Window option will open a new browser window and display a terminal window for accessing the command line on the server, as shown in Figure 5.22.

FIGURE 5.22 A terminal window opens in a new browser window when using SSH-in-browser.



Monitoring a Virtual Machine

While your VM is running, you can monitor CPU, disk, and network load by viewing the Monitoring tab on the VM Instance Details page.

To access monitoring information in the console, select a VM instance from the VM Instance page by clicking the name of the VM you want to monitor. This will display the Details page of the VM. Select the Monitoring option near the top of the page to view monitoring details.

Figure 5.23 shows the information displayed about CPU, network utilization, and disk operations.

Cost of Virtual Machines

Part of the basic management of a VM is tracking the costs of the instances you are running. If you want to track costs automatically, you can enable Cloud billing and set up Billing Export. This will produce daily reports on the usage and cost of VMs.

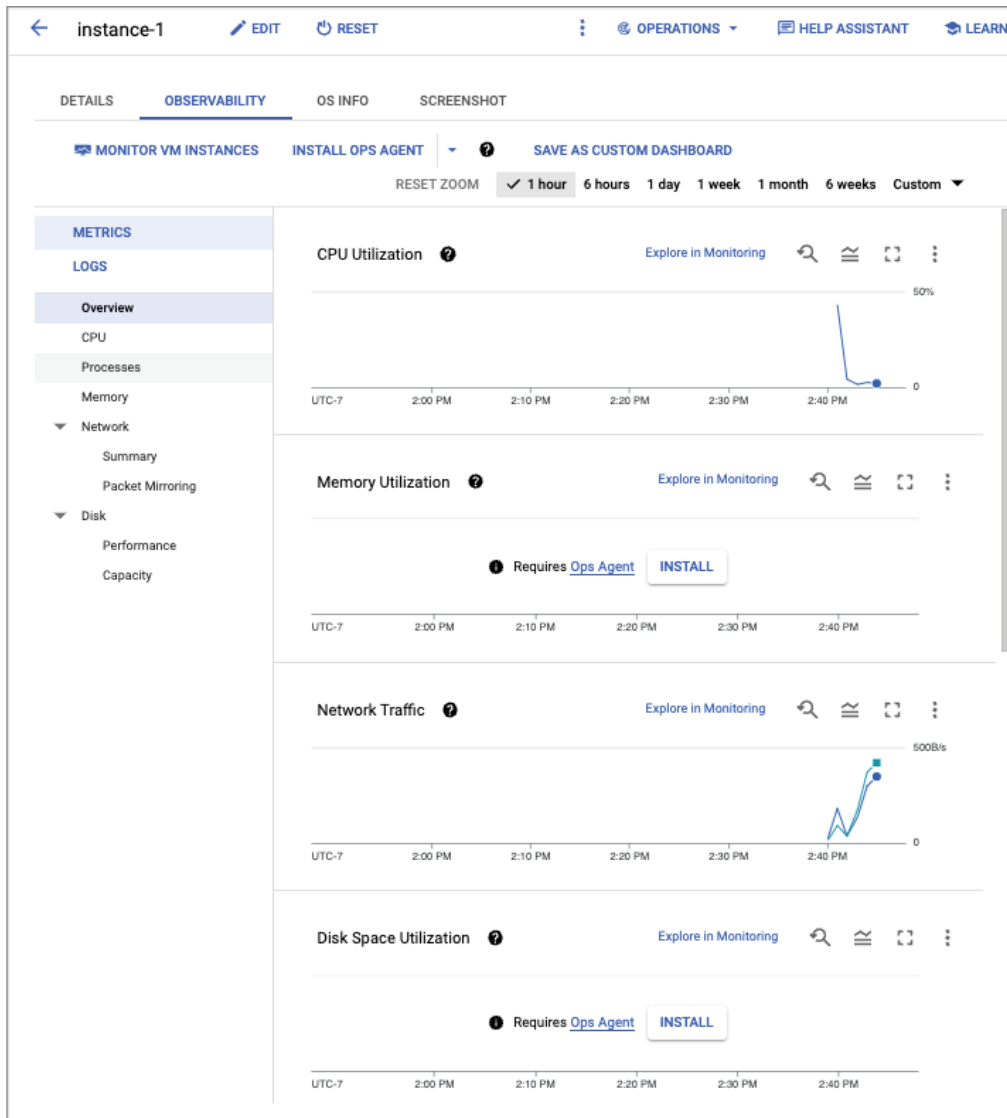
The following are the most important things to remember about VM costs:

- VMs are billed in 1-second increments.
- The cost is based on machine type. The more CPUs and memory used, the higher the cost.
- Google offers discounts for sustained usage, but discounts are not offered on all instance types.

- VMs are charged for a minimum of 1 minute of use.
- Spot VMs can save you up to 80 percent of the cost of a VM.

For additional information on pricing, see <https://cloud.google.com/compute/vm-instance-pricing>.

FIGURE 5.23 The Observability tab of the VM Instance Details page



Guidelines for Planning, Deploying, and Managing Virtual Machines

Consider the following guidelines to help with streamlining your work with VMs. These guidelines apply to working with a small number of VMs. Later chapters will provide additional guidelines for working with clusters and instance groups, which are sets of similarly configured VMs.

- Choose a machine type with the fewest CPUs and the smallest amount of memory that still meets your requirements, including peak capacity. This will minimize the cost of the VM.
- Use the console for ad hoc administration of VMs. Use scripts with `gcloud` commands for tasks that will be repeated.
- Use startup scripts to perform software updates and other tasks that should be performed on startup.
- If you make several modifications to a machine image, consider saving it and using it with new instances rather than running the same set of modifications on every new instance.
- If you can tolerate unplanned disruptions, use spot VMs to reduce cost.
- Use SSH or RDP to access a VM to perform operating system-level tasks.
- Use Cloud Console, Cloud Shell, or Cloud SDK to perform VM-level tasks.

Summary

Google Cloud Console is a web-based graphical user interface for managing Google Cloud resources. Cloud SDK is a command-line package that allows engineers to manage cloud resources from the command line of their local device. Cloud Shell is a web-based terminal interface to VMs. Cloud SDK is installed in Cloud Shell.

When creating a VM, you must specify a number of parameters, including a name for the VM, a region and zone where the VM will run, a machine type that specifies the number of vCPUs and the amount of memory, and a boot disk that includes an operating system.

`gcloud` is the top-level command of the hierarchical command structure in Cloud SDK.

Common tasks when managing VMs are starting and stopping instances, using SSH to access a terminal on the VM, monitoring, and tracking the cost of the VM.

Exam Essentials

Understand how to use Cloud Console and Cloud SDK to create, start, and stop VMs. Parameters that you will need to provide when creating a VM include name, machine type, region, zone, and boot disk. Understand the need to create a VM in a project.

Know how to configure a spot VM using Cloud Console and the `gcloud` commands. Know when to use a spot VM and when not to. Know that spot VMs cost up to 80 percent less than standard VMs.

Know the purpose of advanced options, including Shielded VMs and advanced boot disk configurations. Know that advanced options provide additional security. Understand the kinds of protections provided.

Know how to use `gcloud` compute instance commands to list, start, and stop VMs. Know the structure of `gcloud` commands. `gcloud` commands start with `gcloud` followed by a service, such as `compute`, followed by a resource type, such as `instances`, followed by a command or verb, like `create`, `list`, or `describe`.

Understand how to monitor a VM. Know where to find CPU utilization, network monitoring, and disk monitoring in the VM Instances pages of the console. Know the difference between listing and describing instances with a `gcloud` command.

Know the factors that determine the cost of a VM. Know that Google charges by the second with a 1-minute minimum. Understand that the costs of a machine type may be different in different locations. Know that cost is based on the number of vCPUs and memory.

Review Questions

You can find the answers in the Appendix.

1. You have just opened the Google Cloud console at <http://console.google.com>. You have authenticated with the user you want to use. What is one of the first things you should do before performing tasks on VMs?
 - A. Open Cloud Shell.
 - B. Verify you can log into a VM using SSH.
 - C. Verify that the selected project is the one you want to work with.
 - D. Review the list of running VMs.
2. What is a one-time task you will need to complete before using the console?
 - A. Set up billing.
 - B. Create a project.
 - C. Create a storage bucket.
 - D. Specify a default zone.
3. A colleague has asked for your assistance setting up a test environment in Google Cloud. They have never worked in Google Cloud. You suggest starting with a single VM. Which of the following is the minimal set of information you will need?
 - A. A name for the VM and a machine type
 - B. A name for the VM, a machine type, a region, and a zone
 - C. A name for the VM, a machine type, a region, a zone, and a CIDR block
 - D. A name for the VM, a machine type, a region, a zone, and an IP address
4. An architect has suggested a particular machine type for your workload. You are in the console creating a VM and you don't see the machine type in the list of available machine types. What could be the reason for this?
 - A. You have selected the incorrect subnet.
 - B. That machine type is not available in the zone you specified.
 - C. You have chosen an incompatible operating system.
 - D. You have not specified a correct memory configuration.
5. Your manager asks for your help with understanding cloud computing costs. Your team runs dozens of VMs for three different applications. Two of the applications are for use by the marketing department and one is used by the finance department. Your manager wants a way to bill each department for the cost of the VMs used for their applications. What would you suggest to help solve this problem?
 - A. Access controls
 - B. Persistent disks
 - C. Labels and descriptions
 - D. Descriptions only

6. If you wanted to set the preemptible property using Cloud Console, in which section of the Create An Instance page would you find the option?
 - A. Availability Policy
 - B. Identity And API Access
 - C. Sole Tenancy
 - D. Networking
7. You need to set up a server with a high level of security. You want to be prepared in case of attacks on your server by someone trying to inject a rootkit (a kind of malware that can alter the operating system). Which option should you select when creating a VM?
 - A. Firewall
 - B. Shielded VM
 - C. Project-wide SSH keys
 - D. Boot disk integrity control service
8. All of the following parameters can be set when adding an additional disk through Google Cloud Console, except:
 - A. Disk type
 - B. Encryption key management
 - C. Block size
 - D. Source image for the disk
9. You lead a team of cloud engineers who maintain cloud resources for several departments in your company. You've noticed a problem with configuration drift. Some machine configurations are no longer in the same state as they were when created. You can't find notes or documentation on how the changes were made or why. What practice would you implement to solve this problem?
 - A. Have all cloud engineers use only command-line interface in Cloud Shell.
 - B. Write scripts using `gcloud` commands to change configuration and store those scripts in a version control system.
 - C. Take notes when making changes to configuration and store them in Google Drive.
 - D. Limit privileges so only you can make changes, and you will always know when and why configurations were changed.
10. When you're using the Cloud SDK command-line interface, which of the following is part of commands for administering resources in Compute Engine?
 - A. `gcloud compute instances`
 - B. `gcloud instances`
 - C. `gcloud instances compute`
 - D. None of the above

11. A newly hired cloud engineer is trying to understand what VMs are running in a particular project. How could the engineer get summary information on each VM running in a project?
 - A. Execute the command `gcloud compute list`.
 - B. Execute the command `gcloud compute instances list`.
 - C. Execute the command `gcloud instances list`.
 - D. Execute the command `gcloud list instances`.
12. When creating a VM using the command line, how should you specify labels for the VM?
 - A. Use the `--labels` option with labels in the format of `KEYS:VALUES`.
 - B. Use the `--labels` option with labels in the format of `KEYS=VALUE`.
 - C. Use the `--labels` option with labels in the format of `KEYS,VALUES`.
 - D. This is not possible in the command line.
13. In the boot disk advanced configuration, which operations can you specify when creating a new VM?
 - A. Add a new disk, reformat an existing disk, attach an existing disk.
 - B. Add a new disk and reformat an existing disk.
 - C. Add a new disk and attach an existing disk.
 - D. Reformat an existing disk and attach an existing disk.
14. You have acquired a 10 GB data set from a third-party research firm. A group of data scientists would like to access this data from their statistics programs written in R. R works well with Linux and Windows filesystems, and the data scientists are familiar with file operations in R. The data scientists would each like to have their own dedicated VM with the data available in the VM's filesystem. What is a way to make this data readily available on a VM and minimize the steps the data scientists will have to take?
 - A. Store the data in Cloud Storage.
 - B. Create VMs using a source image created from a disk with the data on it.
 - C. Store the data in Google Drive.
 - D. Load the data into BigQuery.
15. The Networking tab of the Create VM form is where you would perform which of the following operations?
 - A. Set the IP address of the VM.
 - B. Add a network interface to the VM.
 - C. Specify a default router.
 - D. Change firewall configuration rules.

16. You want to create a VM using the `gcloud` command. What parameter would you include to specify the type of boot disk?
- A. `boot-disk-type`
 - B. `boot-disk`
 - C. `disk-type`
 - D. `type-boot-disk`
17. Which of the following commands will create a VM with four CPUs that is named `web-server-1`?
- A. `gcloud compute instances create --machine-type=n1-standard-4\`
`web-server-1`
 - B. `gcloud compute instances create --cpus=4 web-server-1`
 - C. `gcloud compute instances create --machine-type=n1-standard-4\`
`-instance-name=web-server-1`
 - D. `gcloud compute instances create --machine-type=n1-4-cpu\`
`web-server-1`
18. Which of the following commands will stop a VM named `web-server-1`?
- A. `gcloud compute instances halt web-server-1`
 - B. `gcloud compute instances --terminate web-server1`
 - C. `gcloud compute instances stop web-server-1`
 - D. `gcloud compute stop web-server-1`
19. You have just created an Ubuntu VM and want to log into the VM to install some software packages. Which network service would you use to access the VM?
- A. FTP
 - B. SSH
 - C. RDP
 - D. `ipconfig`
20. Your management team is considering three different cloud providers. You have been asked to summarize billing and cost information to help the management team compare cost structures between clouds. Which of the following would you mention about the cost of VMs in Google Cloud?
- A. VMs are billed in 1-second increments, cost varies with the number of CPUs and amount of memory in a machine type, you can create custom machine types, preemptible VMs cost up to 80 percent less than standard VMs, and Google offers discounts for sustained usage.
 - B. VMs are billed in 1-second increments and VMs can run up to 24 hours before they will be shut down.
 - C. Google offers discounts for sustained usage in only some regions, cost varies with the number of CPUs and amount of memory in a machine type, you can create custom machine types, preemptible VMs cost up to 80 percent less than standard VMs.
 - D. VMs are charged for a minimum of 1 hour of use, and cost varies with the number of CPUs and amount of memory in a machine type.

Chapter 6

Managing Virtual Machines

**THIS CHAPTER COVERS THE FOLLOWING
OBJECTIVES OF THE GOOGLE ASSOCIATE
CLOUD ENGINEER CERTIFICATION EXAM:**

- ✓ 4.1 Managing Compute Engine resources





After creating virtual machines, you will need to work with both single instances of virtual machines (VMs) and groups of VMs that run the same configuration. The latter are called *instance groups* and are introduced in this chapter.

This chapter begins with a description of common management tasks and how to complete them in the console, followed by a description of how to complete them in Cloud Shell or with the Cloud SDK command line. Next, you will learn how to configure and manage instance groups. The chapter concludes with a discussion of guidelines for managing VMs.

Managing Single Virtual Machine Instances

We begin by discussing how to manage a single instance of a VM. By single instance, we mean one created by itself and not in an instance group or other type of cluster. Recall from previous chapters that there are three ways to work with instances: in Cloud Console, in Cloud Shell, and with the Cloud SDK command line. Both Cloud Shell and the Cloud SDK command line make use of `gcloud` commands, so we will describe Cloud Shell and Cloud SDK together in this section.

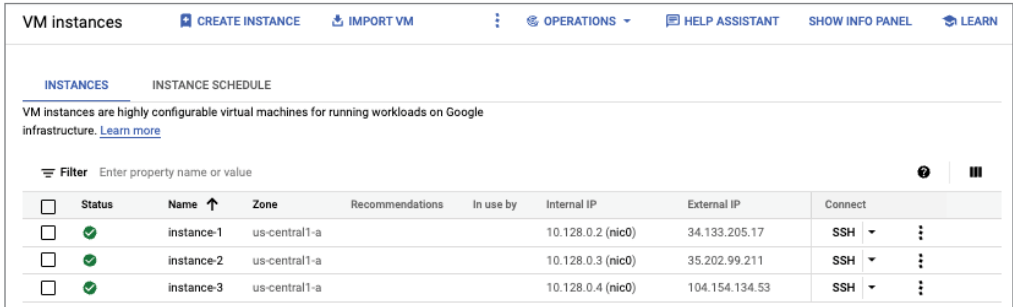
Managing Single Virtual Machine Instances in the Console

The basic VM management tasks that you should be familiar with are creating, stopping, and deleting instances. We covered creating instances in the previous chapter, so we'll focus on the other tasks here. You should also be familiar with listing VMs, attaching graphics processing units (GPUs) to VMs, and working with snapshots and images.

Starting, Stopping, and Deleting Instances

To start working, open the console and select Compute Engine. Then select VM instances. This will display a window like the one in Figure 6.1, but with different VMs listed. In this example, there are three VMs.

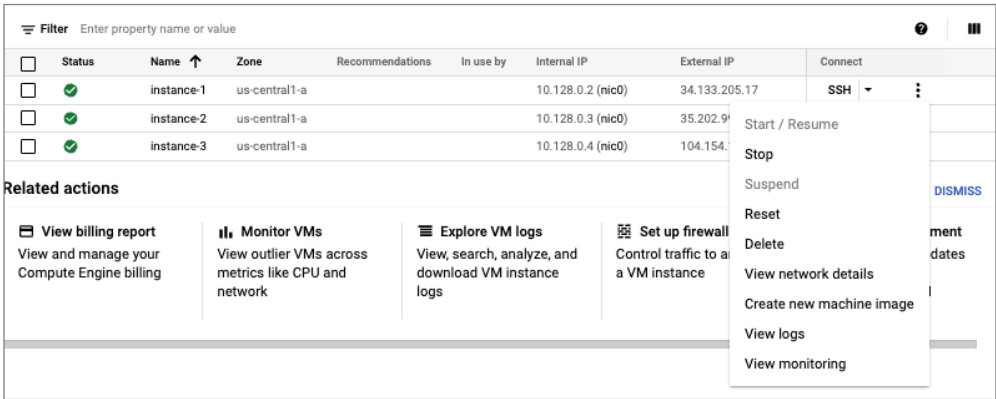
FIGURE 6.1 The VM Instance panel in the Compute Engine section of Cloud Console



Source: Google LLC

The three instances in Figure 6.1 are all running. You can stop the instances by clicking the three-dot icon on the right side of the line listing the VM attributes. This action displays a list of commands. Figure 6.2 shows the list of commands available for `instance-1`.

FIGURE 6.2 The list of commands available from the console for changing the state of a VM

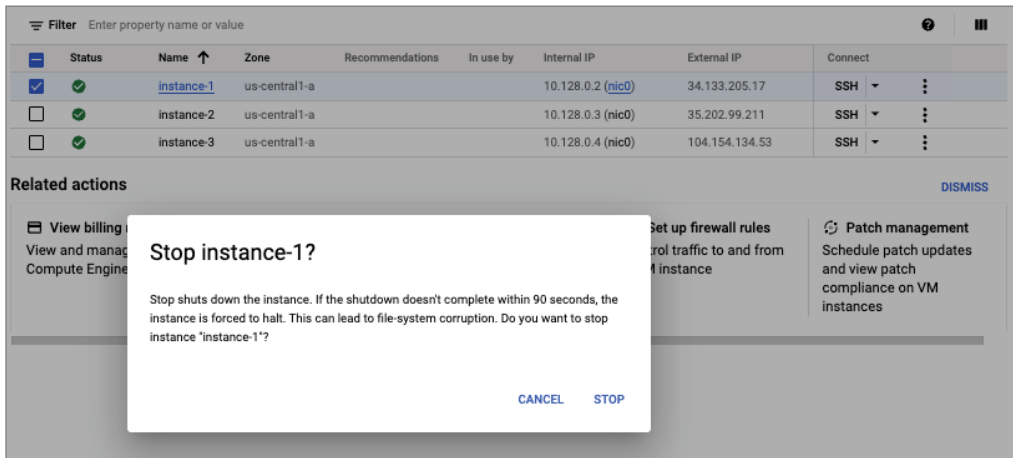


Source: Google LLC

If you select Stop from the command menu, the instance will be stopped. When an instance is stopped, it is not consuming compute resources, so you will not be charged. The instance still exists and can be started again when you need it. Figure 6.3 shows a warning form that indicates you are about to stop a VM. You can click the dialog box in the lower left to suppress this message.

When you stop a VM, the green check mark on the left changes to a gray circle with a white square, and the SSH option is disabled, as shown in Figure 6.4.

FIGURE 6.3 A warning message that may appear about stopping a VM



Source: Google LLC

FIGURE 6.4 When VMs are stopped, the icon on the left changes and SSH is no longer available.

Filter	Enter property name or value							
<input checked="" type="checkbox"/>	Status	Name	Zone	Recommendations	In use by	Internal IP	External IP	Connect
<input type="checkbox"/>	⚙	instance-1	us-central1-a			10.128.0.2 (nic0)	None	SSH ▾ ⋮
<input type="checkbox"/>	✔	instance-2	us-central1-a			10.128.0.3 (nic0)	35.202.99.211	SSH ▾ ⋮
<input type="checkbox"/>	✔	instance-3	us-central1-a			10.128.0.4 (nic0)	104.154.134.53	SSH ▾ ⋮

Source: Google LLC

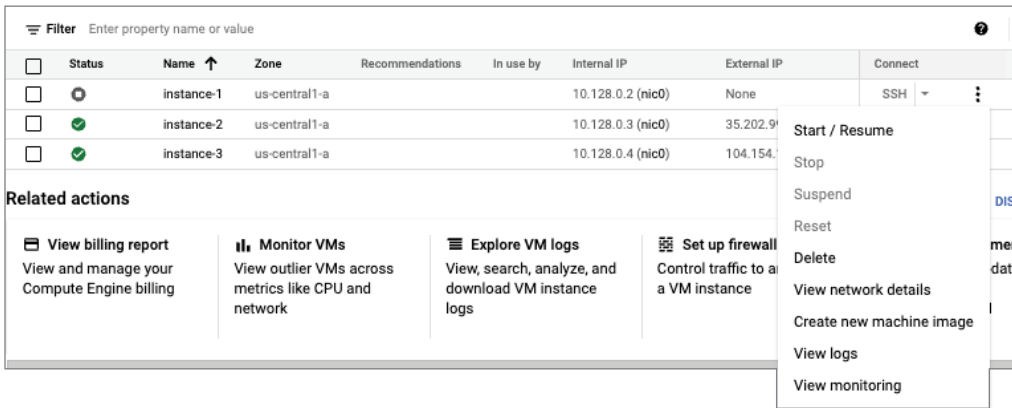
To start a stopped VM, click the three-dot icon on the right to display the menu of available commands. Notice in Figure 6.5 that Start is now available but Stop and Reset are not. The Reset command restarts a VM. The properties of the VM will not change, but data in memory will be lost.



When a VM is restarted, the contents of memory are lost. If you need to preserve data between reboots or for use on other VMs, save the data to a persistent disk or Cloud Storage.

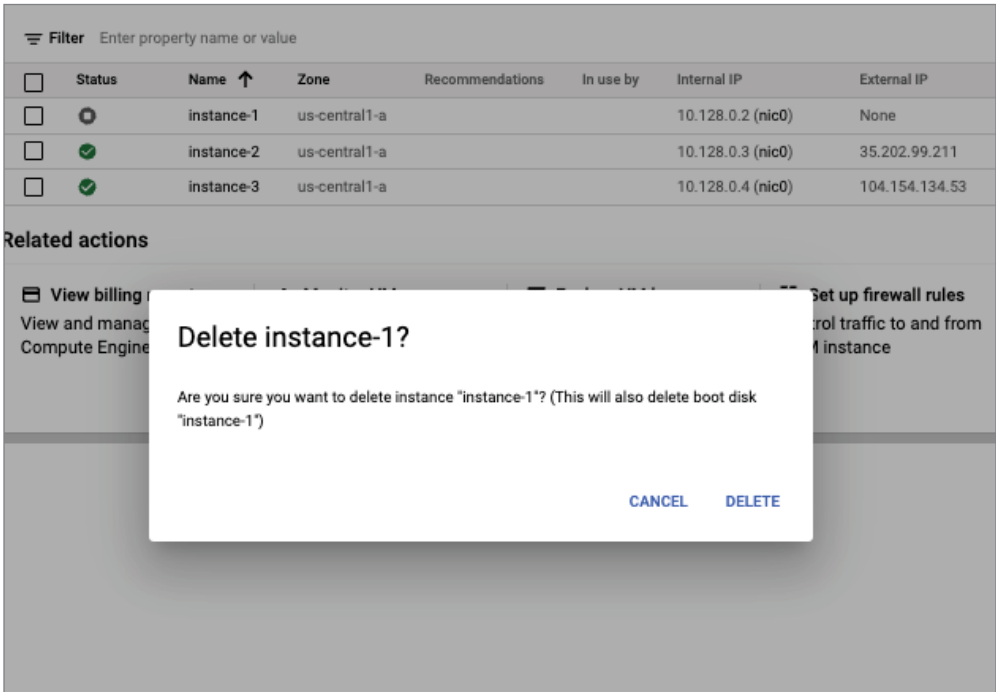
When you are done with an instance and no longer need it, you can delete it. Deleting a VM removes it from Cloud Console and releases resources, like the storage used to keep the VM image when stopped. Deleting an instance from Cloud Console will display a warning message, shown in Figure 6.6.

FIGURE 6.5 When VMs are stopped, Stop and Reset are no longer available, but Start / Resume is available as a command.



Source: Google LLC

FIGURE 6.6 Deleting an instance from the console will display a warning message such as this.

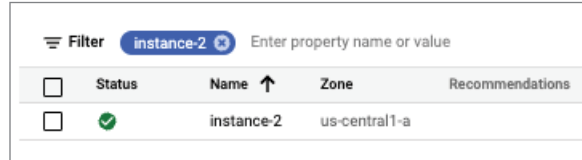


Source: Google LLC

Viewing Virtual Machine Inventory

The VM Instances page of Cloud Console will show a list of VMs, if any exist in the current project. If you have a large number of instances, it can help to filter the list to see only instances of interest. Do this by using the Filter VM Instances box above the list of VMs, as shown in Figure 6.7.

FIGURE 6.7 List of instances filtered by search criteria



The screenshot shows the Google Cloud VM Instances page. At the top, there is a 'Filter' button with a dropdown menu showing 'instance-2'. To the right of the filter button is a text input field with the placeholder 'Enter property name or value'. Below the filter bar is a table with the following columns: 'Status', 'Name', 'Zone', and 'Recommendations'. The table contains one row with the following data: 'Status' is 'Running' (indicated by a green checkmark), 'Name' is 'instance-2', 'Zone' is 'us-central1-a', and 'Recommendations' is empty.

	Status	Name	Zone	Recommendations
<input type="checkbox"/>	Running	instance-2	us-central1-a	

Source: Google LLC

In this example, we have specified that we want to see only the instance named `instance-2`. In addition to specifying instance names, you can also filter by the following:

- Labels
- Internal IP
- External IP
- Status
- Zone
- Network
- Deletion protection

If you set multiple filter conditions, then all must be true for a VM to be listed unless you explicitly state the OR operator.

Attaching GPUs to an Instance

GPUs are used for math-intensive applications such as visualizations and machine learning. GPUs perform math calculations and allow some work to be offloaded from the CPU to the GPU. Compute Engine has a machine family specifically designed for VMs with GPUs. To use GPUs, you will also need to install GPU drivers or use an image that has GPU drivers already installed.

When creating an instance in the console, you can choose the GPU machine family to see the options for working with GPUs. (See Figure 6.8.)

FIGURE 6.8 GPU machine family supports a variety of GPU types, and a number of GPUs and CPU platforms.

Machine configuration

Machine family

GENERAL-PURPOSE

COMPUTE-OPTIMIZED

MEMORY-OPTIMIZED

GPU

Optimized for machine learning, high performance computing, and visualization workloads

GPU type

NVIDIA Tesla A100

Number of GPUs

1

☐ Enable Virtual Workstation (NVIDIA GRID)

1

To enable Virtual Workstation (NVIDIA GRID), choose a different GPU such as NVIDIA Tesla T4, P4 or P100. [Learn more.](#)

Series

A2


Powered by NVIDIA A100 graphics cards.

Machine type

a2-highgpu-1g (12 vCPU, 85 GB memory)

vCPU

12

Memory

85 GB

CPU platform

Automatic

Source: Google LLC

To add a GPU to an instance, you must start an instance in which GPU libraries have been installed or will be installed. For example, you can use one of the Google Cloud images that has GPU libraries installed, including the Deep Learning images, as shown in Figure 6.8. You must also verify that the instance will run in a zone that has GPUs available.

The parameters you can configure include GPU Type and Number Of GPUs. Figure 6.9 shows some GPU options. The type of GPU will determine the number of GPUs available. For example, currently the NVIDIA Tesla A100 can be used in 1, 2, 4, 8, or 16 GPU configurations whereas the NVIDIA Tesla T4 can be used in 1, 2, or 4 GPU configurations.

FIGURE 6.9 Some GPU options available in Compute Engine

GPU type

NVIDIA Tesla A100

NVIDIA Tesla K80

NVIDIA Tesla P4

NVIDIA Tesla T4

NVIDIA Tesla V100

Number of GPUs

1

☐ Enable Virtual Workstation (NVIDIA GRID)

1

To enable Virtual Workstation (NVIDIA GRID), choose a different GPU such as NVIDIA Tesla T4, P4 or P100. [Learn more.](#)

Source: Google LLC

As with other machine families, you can specify a machine type. You can also specify a CPU platform, such as Intel Skylake or later or Intel Ivy Bridge or later. These are microarchitecture options. Compute Engine will automatically choose a CPU platform by default.

There are some restrictions on the use of GPUs; for example, GPUs cannot be attached to shared memory machines. For the latest documentation on GPU restrictions and a list of zones with GPUs, see <https://cloud.google.com/compute/docs/gpus>.

Working with Snapshots

Snapshots are copies of data on a persistent disk. You use snapshots to save data on a disk so that you can restore it. This is a convenient way to make multiple persistent disks with the same data or to back up a disk so that you can recover the state of the disk at a particular point in time.

When you first create a snapshot, Google Cloud will make a full copy of the data on the persistent disk. The next time you create a snapshot from that disk, Google Cloud will copy only the data that has changed since the last snapshot. This optimizes storage while keeping the snapshot up-to-date with the data that was on the disk the last time a snapshot operation occurred.

If you are running a database or other application that may buffer data in memory before writing to disk, be sure to flush disk buffers before you create the snapshot; otherwise, data in memory that should be written to disk may be lost. The way to flush the disk buffers will vary by application. For example, MySQL has a FLUSH statement.

To work with snapshots, a user must be assigned the Compute Storage Admin role. Go to the Identity Access Management (IAM) page, select Roles, and then specify the email address of a user to be assigned the role.

To create a snapshot from Cloud Console, display the Compute Engine options and select Snapshots from the left panel, as shown in Figure 6.10.

Then, click Create Snapshot to display the form in Figure 6.11. Specify a name and, optionally, a description. You can add labels to the snapshot as well. It is a good practice to label all resources with a consistent labeling convention. In the case of snapshots, the labels may indicate the type of data on the disk and the application that uses the data.

You have the option of storing the snapshot regionally or multiregionally.

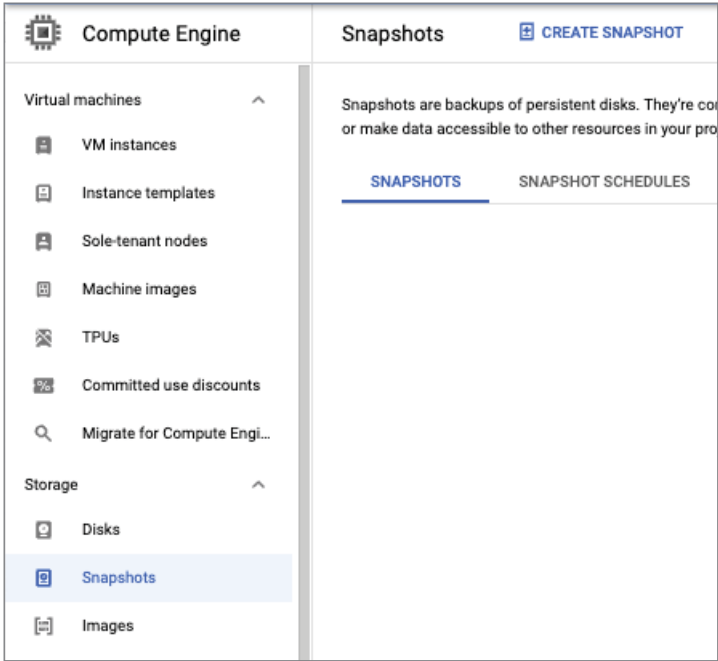
Working with Images

Images are similar to snapshots in that they are copies of disk contents. The difference is that snapshots are used to make data available on a disk, whereas images are used to create VMs. VMs can also be created from snapshots, as long as that snapshot is made from a boot disk. The main storage difference between images and snapshots is that snapshots offer incremental backups, while images are a single complete backup. Images can be created from the following:

- Disk
- Snapshot
- Image

- Cloud storage file
- Virtual disk

FIGURE 6.10 Creating a snapshot using Cloud Console



Source: Google LLC

To create an image, choose the Image option from the Compute Engine page in Cloud Console, as shown in Figure 6.12. This lists available images.

Select Create Image to show the form in Figure 6.13. This form allows you to create a new image by specifying a name, description, and labels. Images have an optional attribute called Family, which allows you to group images. When a family is specified, the latest, nondeprecated image in the family is used.

The form provides a drop-down list of options for the source of the image, as shown in Figure 6.14.

FIGURE 6.11 Form for creating a snapshot

← Create a snapshot

Snapshots are backups of persistent disks. They're commonly used to recover, transfer, or make data accessible to other resources in your project. [Learn more](#)

Name *
snapshot-1
Name is permanent

Description

Source disk *

Location ?
There may be a network transfer fee if you choose to store this snapshot in a location different than the source disk. [Learn more](#)
☐ Multi-regional
☐ Regional

Select location

Labels ?
[+ ADD LABEL](#)

Your free trial credit will be used for this snapshot. [GCP Free Tier](#)

Source: Google LLC

When you choose Image as the source type, you can choose an image from the current project or other projects (see Figure 6.15).

If you choose a Cloud Storage file as a source, you can browse your Cloud Storage bucket to find a file to use as the source (see Figure 6.16).

After you have created an image, you can delete it or deprecate it by checking the box next to the image name and selecting Delete or Deprecate from the line of commands above the list. You can delete and deprecate only custom images, not Google Cloud-supplied images.

FIGURE 6.12 Images available. From here, you can create additional images.

IMAGES										
IMAGE IMPORT HISTORY										
IMAGE EXPORT HISTORY										
<div>Filter Enter property name or value</div>										
<div>Show deprecated images</div>										
<input type="checkbox"/>	Status	Name	Location	Archive size	Disk size	Created by	Family	Architecture	Actions	
<input type="checkbox"/>	✓	c0-deeplearning-common-cpu-v20221026-debian-10	asia, eu, us	—	50 GB	Debian	common-cpu-debian-10	—	⋮	
<input type="checkbox"/>	✓	c0-deeplearning-common-cu113-v20221026-debian-10	asia, eu, us	—	50 GB	Debian	common-dl-gpu-debian-10	—	⋮	
<input type="checkbox"/>	✓	c1-deeplearning-tf-1-15-gpu-cu110-v20221026-debian-10	asia, eu, us	—	50 GB	Debian	tf-1-15-gpu-debian-10	—	⋮	

Source: Google LLC

Delete removes the image, and Deprecated marks the image as no longer supported and allows you to specify a replacement image to use going forward. Google’s deprecated images are available for use but may not be patched for security flaws or other updates. Deprecation is a useful way of informing users of the image that it is no longer supported and that they should plan to test their applications with the newer, supported versions of the image. Eventually, deprecated images will no longer be available, and users of the deprecated images will need to use different versions.

After you have created an image, you can create an instance using that image by selecting the Create Instance option in the line of commands above the image listing.

Managing a Single Virtual Machine Instance with Cloud Shell and the Command Line

In addition to managing VMs through the console, you can manage compute resources using the command line. The same commands can be used in Cloud Shell or in your local environment after you have installed Google Cloud SDK, which was covered in Chapter 5, “Computing with Compute Engine Virtual Machines.”

FIGURE 6.13 Cloud Console form for creating an image

The screenshot shows the 'Create an image' form in the Google Cloud Console. The form is titled 'Create an image' with a back arrow. It contains several sections: 'Name' with a text input 'image-1' and a help icon; 'Source' with a dropdown menu set to 'Disk' and a help icon; 'Source disk' with a dropdown menu and a help icon; 'Location' with radio buttons for 'Multi-regional' and 'Regional', a 'Select location' dropdown, and a 'Family' dropdown; a 'Description' text area; 'Labels' with an '+ ADD LABEL' button; and 'Encryption' with three radio button options: 'Google-managed encryption key' (selected), 'Customer-managed encryption key (CMEK)', and 'Customer-supplied encryption key (CSEK)'. Each option has a brief description below it.

Source: Google LLC

FIGURE 6.14 Options for the source of an image

This screenshot shows the 'Source' dropdown menu from the previous form. The menu is open, displaying five options: 'Disk' (highlighted in blue), 'Snapshot', 'Image', 'Cloud Storage file', and 'Virtual disk (VMDK, VHD)'. Each option has a help icon to its right.

Source: Google LLC

FIGURE 6.15 When using an image as a source, you can choose a source image from another project.

Source *

Image

Source project for images *

sc [redacted] ? CHANGE

Source image *

Source: Google LLC

FIGURE 6.16 When using a Cloud Storage file as a source, you browse your storage buckets for a file.

name is permanent

Source *

Cloud Storage file

Cloud Storage file * ? BROWSE

Your image source must use the .tar.gz extension and the file inside the archive must be named disk.raw. [Learn more](#)

Source: Google LLC

This section describes the most important commands for working with instances. Commands have their own specific sets of parameters; however, all `gcloud` commands support sets of flags. These are referred to as `gcloud`-wide flags, also known as `gcloud` global flags, and include the following:

- `--account` specifies a Google Cloud account to use overriding the default account.
- `--configuration` uses a named configuration file that contains key-value pairs.
- `--flatten` generates separate key-value records when a key has multiple values.
- `--format` specifies an output format, such as default (human-readable) CSV, JSON, YAML, text, or other possible options.
- `--help` displays a detailed help message.
- `--project` specifies a Google Cloud project to use, overriding the default project.
- `--quiet` disables interactive prompts and uses defaults.
- `--verbosity` specifies the level of detailed output messages. Options are `debug`, `info`, `warning`, and `error`.

Throughout this section, commands can take an optional `--zone` parameter. We assume a default zone was set when you ran `gcloud init`.

Starting Instances

To start an instance, use the `gcloud` command, specifying that you are working with a compute service and instances specifically. You also need to indicate that you will be starting an instance by specifying `start`, followed by the name of one or more instances.

The command syntax is as follows:

```
gcloud compute instances start INSTANCE_NAMES
```

An example is as follows:

```
gcloud compute instances start instance-1 instance-2
```

The `instance start` command also takes optional parameters. The `--async` parameter returns immediately without waiting for the operations to complete. The `--verbose` option in many Linux commands provides similar functionality. Here's an example:

```
gcloud compute instances start instance-1 instance-2 --async
```

Google Cloud needs to know in which zone to create an instance. This can be specified with the `--zone` parameter as follows:

```
gcloud compute instances start ch06-instance-1 ch06-instance-2 \
--zone=us-central1-c
```

You can get a list of zones with the following command:

```
gcloud compute zones list
```

If no zone is specified, the command will prompt for one.

Stopping Instances

To stop an instance, use `gcloud compute instances` and specify `stop` followed by the name of one or more instances.

The command syntax is as follows:

```
gcloud compute instances stop INSTANCE_NAMES
```

Here's an example:

```
gcloud compute instances stop instance-3 instance-4
```

Like the `instance start` command, the `stop` command takes optional parameters:

```
gcloud compute instances stop ch06-instance-1 ch06-instance-2 --async
```

Google Cloud needs to know which zone contains the instance to stop. This can be specified with the `--zone` parameter as follows:

```
gcloud compute instances stop ch06-instance-1 ch06-instance-2 \
--zone=us-central1-c
```

You can get a list of zones with the following command:

```
gcloud compute zones list
```

Deleting Instances

When you are finished working with a VM, you can delete it with the `delete` command. Here's an example:

```
gcloud compute instances delete instance-1
```

The `delete` command takes the `--zone` parameter to specify where the VM to delete is located. Here's an example:

```
gcloud compute instances delete ch06-instance-1 --zone=us-central1-b
```

When an instance is deleted, the disks on the VM may be deleted or saved by using the `--delete-disks` and `--keep-disks` parameters, respectively. You can specify `all` to keep all disks, `boot` to specify the partition of the root filesystem, and `data` to specify nonboot disks.

For example, the following command keeps all disks:

```
gcloud compute instances delete ch06-instance-1 --zone=us-central2-b\
--keep-disks=all
```

while the following deletes all nonboot disks:

```
gcloud compute instances delete ch06-instance-1 --zone=us-central2-b\
--delete-disks=data
```

Viewing VM Inventory

The command to view the set of VMs in your inventory is as follows:

```
gcloud compute instances list
```

This command takes an optional name of an instance. To list VMs in a particular zone, you can use the following:

```
gcloud compute instances list --filter="zone:ZONE"
```

where *ZONE* is the name of a zone. You can list multiple zones using a comma-separated list.

The `--limit` parameter is used to limit the number of VMs listed, and the `--sort-by` parameter is used to reorder the list of VMs by specifying a resource field. You can see the resource fields for a VM by running the following:

```
gcloud compute instances describe
```

Working with Snapshots

You can create a snapshot of a disk using the following command:

```
gcloud compute disks snapshot DISK_NAME --snapshot-names=NAME
```

where *DISK_NAME* is the name of a disk and *NAME* is the name of the snapshot. To view a list of snapshots, use the following:

```
gcloud compute snapshots list
```

For detailed information about a snapshot, use the following:

```
gcloud compute snapshots describe SNAPSHOT_NAME
```

where *SNAPSHOT_NAME* is the name of the snapshot to describe. To create a disk, use this:

```
gcloud compute disks create DISK_NAME --source-snapshot=SNAPSHOT_NAME
```

You can also specify the size of the disk and disk type using the `--size` and `--parameters`. Here's an example:

```
gcloud compute disks create disk-1 --source-snapshot=ch06-snapshot --size=100\
--type=pd-standard
```

This will create a 100 GB disk using the `ch06-snapshot` using a standard persistent disk.

Working with Images

Google Cloud provides a wide range of images to use when creating a VM; however, you may need to create a specialized image of your own. This can be done with the following command:

```
gcloud compute images create IMAGE_NAME
```

where *IMAGE_NAME* is the name given to the images. The source for the images is specified using one of the source parameters, which are as follows:

- `--source-disk`
- `--source-image`
- `--source-image-family`
- `--source-snapshot`
- `--source-uri`

The `source-disk`, `source-image`, and `source-snapshot` parameters are used to create an image using a disk, image, and snapshot, respectively. The `source-image-family` parameter uses the latest version of an image in the family. Families are groups of related images, which are usually different versions of the same underlying image. The `source-uri` parameter allows you to specify an image using a web address.

An image can have a description and a set of labels. These are assigned using the `--description` and `--labels` parameters.

Here is an example of creating a new image from a disk:

```
gcloud compute images create image-1 --source-disk=disk-1
```

You can also delete images when they are no longer needed using this:

```
gcloud compute images delete IMAGE_NAME
```

It is often helpful to store images on Cloud Storage. You can export an image to Cloud Storage with the following command:

```
gcloud compute images export --destination-uri=DESTINATION_URI\  
--image=IMAGE_NAME
```

where *DESTINATION_URI* is the address of a Cloud Storage bucket where you want to store the image.

Introduction to Instance Groups

Instance groups are sets of VMs that are managed as a single entity. Any `gcloud` or console command applied to an instance group is applied to all members of the instance group. Google provides two types of instance groups: managed and unmanaged.

Managed groups consist of groups of identical VMs. They are created using an instance template, which is a specification of a VM configuration, including machine type, boot disk image, zone, labels, and other properties of an instance. Managed instance groups can automatically scale the number of instances in a group and be used with load balancing to distribute workloads across the instance group. If an instance in a group crashes, it will be re-created automatically. Managed groups are the preferred type of instance group.

Unmanaged groups should be used only when you need to work with different configurations within different VMs in the group.

Creating and Removing Instance Groups and Templates

To create an instance group, you must first create an instance group template. To create an instance template, use the following command:

```
gcloud compute instance-templates create INSTANCE
```

You can specify an existing VM as the source of the instance template by using the `--source-instance` parameter. Here's an example:


```
gcloud compute instance-templates create instance-template-1\  
--source-instance=instance-1
```

Instance group templates can also be created in the console using the Instance Groups Template page, as shown in Figure 6.17.


Instance groups can contain instances in a single zone or across a region. The first is called a *zonal* managed instance group, and the second is called a *regional* managed instance group. Regional managed instance groups are recommended because that configuration spreads the workload across zones, increasing resiliency.

FIGURE 6.17 Instance group templates can be created in the console using a form similar to the create instance form.

Set up automatic management for a group of stateless VMs, including updates, regional deployments, load balancing, autoscaling, and autohealing. [Learn more](#)

Name *
instance-group-1 
Name is permanent



Description

Instance template * 

Number of instances
Based on autoscaling configuration



Location
For higher availability, select multiple zones in a region instead of a single zone. [Learn more](#)


☒ Single zone
☐ Multiple zones


Region * us-central1 (Iowa)  **Zone *** us-central1-a 

Autoscaling
Use autoscaling to automatically add and remove instances to the group for periods of high and low load. [Learn more](#)


Autoscaling mode
On: add and remove instances to the group

Minimum number of instances * 1  **Maximum number of instances *** 10 

Autoscaling metrics 
Use metrics to help determine when to scale the group. [Learn more](#)

CPU utilization: 60% (default)
Predictive autoscaling is off 

[ADD METRIC](#)

Autoscaling schedules 

Source: Google LLC

You can specify a distribution policy. Even distribution will evenly distribute across zones. Balanced distribution will distribute as evenly as possible across zones based on available resources. The Any distribution will deploy managed instances to zones based on availability and reservations.

You can remove instance templates by deleting them from the Instance Group Template page in the console. Select the instance group template by selecting the check box in the list of templates and then delete it by clicking the delete icon.

You can also delete an instance group template using the following command:

```
gcloud compute instance-templates delete INSTANCE-TEMPLATE-NAME
```

where *INSTANCE-TEMPLATE-NAME* is the name of the template you want to delete.

To list templates and instance groups, use the following:

```
gcloud compute instance-templates list
gcloud compute instance-groups managed list-instances
```

To list the instances in an instance group, use the following:

```
gcloud compute instance-groups managed list-instances INSTANCE-GROUP-NAME
```

Instance Groups Load Balancing and Autoscaling

To deploy a scalable, highly available application, you can run that application on a load-balanced set of instances. Google Cloud offers several types of load balancing, and they all require use of an instance group.

In addition to load balancing, managed instance groups can be configured to autoscale. You can configure an autoscaling policy to trigger adding or removing instances based on CPU utilization, monitoring metric, load-balancing capacity, or queue-based workloads.

No More Peak Capacity Planning

Prior to the advent of the cloud, IT organizations often had to plan their hardware purchases around the maximum expected load. This is called *peak capacity planning*. If there is little variation in load, peak capacity planning is a sound approach. Businesses with highly variable workloads, such as retailers in the United States that have high demand during the last two months of the year, would have to support idle capacity for months out of the year. Cloud computing and autoscaling have eliminated the need for peak capacity planning. Additional servers are acquired in minutes, not weeks or months. When capacity is not needed, it is dropped. Instance groups automate the process of adding and removing VMs, allowing cloud engineers to fine-tune when to add and when to remove VMs.



When autoscaling, ensure you leave enough time for VMs to boot up or shut down before triggering another change in the cluster configuration. If the time between checks is too small, you may find that a recently added VM is not fully started before another is added. This can lead to more VMs being added than are actually needed.

Guidelines for Managing Virtual Machines

Here are some guidelines for managing VMs:

- Use labels and descriptions. This will help you identify the purpose of an instance and help when filtering lists of instances.
- Use managed instance groups to enable autoscaling and load balancing. These are key to deploying scalable and highly available services.
- Use GPUs for numeric-intensive processing, such as machine learning and high-performance computing. For some applications, GPUs can give a greater performance benefit than adding another CPU.
- Use snapshots to save the state of a disk or to make copies. These can be saved in Cloud Storage and act as backups.
- Use preemptible instances for workloads that can tolerate disruption. Spot VMs are lower-cost VMs that are suitable for 60%–91% less than standard priced VMs. They are preemptible instances but are not limited to maximum runtimes of 24 hours like the original preemptible VMs are.

Summary

In this chapter, you learned how to manage a single VM instances and instance groups. Single VM instances can be created, configured, stopped, started, and deleted using Cloud Console or `gcloud` commands from Cloud Shell or your local machine if you have SDK installed.

Snapshots are copies of disks and are useful as backups and for copying data to other instances. Images are complete backups of a boot disk, so are used to create VMs. Snapshots made from a boot disk can also be used to create a VM.

The main command used to manage VMs is the `gcloud compute instances` command. `gcloud` uses a hierarchical structure to order the command elements. The command begins with `gcloud`, followed by a Google Cloud component, such as `compute` for Compute Engine, followed by an entity type such as `instances` or `snapshots`. An operation is then specified, such as `create`, `delete`, `list`, or `describe`.

GPUs can be attached to instances that have GPU libraries installed in the operating system. GPUs are used for compute-intensive tasks, such as building machine learning models.

Instance groups are groups of instances that are managed together. Managed instance groups have instances that are the same. These groups support load balancing and autoscaling.

Exam Essentials

Understand how to navigate Cloud Console. Cloud Console is the graphical interface for working with Google Cloud. You can create, configure, delete, and list VM instances from the Compute Engine area of the console.

Understand how to install Cloud SDK. Cloud SDK allows you to configure default environment variables, such as a preferred zone, and issue commands from the command line. If you use Cloud Shell, Cloud SDK is already installed.

Know how to create a VM in the console and at the command line. You can specify machine type, choose an image, and configure disks with the console. You can use commands at the command line to list and describe, and you can find the same information in the console. Understand when to use customized images and how to deprecate them. Images are copies of contents of a disk, and they are used to create VMs. Deprecated marks an image as no longer supported.

Understand why GPUs are used and how to attach them to a VM. GPUs are used for compute-intensive operations; a common use case for using GPUs is machine learning. It is best to use an image that has GPU libraries installed. Understand how to determine which locations have GPUs available, because there are some restrictions. The CPU must be compatible with the GPU selected, and GPUs cannot be attached to shared memory machines. Know how GPU costs are charged.

Understand images and snapshots. Snapshots save the contents of disks for backup and data-sharing purposes. Images save the operating system and related configurations so that you can create identical copies of the instance.

Understand instance groups and instance group templates. Instance groups are sets of instances managed as a single entity. Instance group templates specify the configuration of an instance group and the instances in it. Managed instance groups support autoscaling and load balancing.

Review Questions

You can find the answers in the Appendix.

1. Which page in Google Cloud Console would you use to create a single instance of a VM?
 - A. Compute Engine
 - B. App Engine
 - C. Kubernetes Engine
 - D. Cloud Functions
2. You view a list of Linux VM instances in the console. All have public IP addresses assigned. You notice that the SSH option is disabled for one of the instances. Why might that be the case?
 - A. The instance is preemptible and therefore does not support SSH.
 - B. The instance is stopped.
 - C. The instance was configured with the No SSH option.
 - D. The SSH option is never disabled.
3. You have noticed unusually slow response time when issuing commands to a Linux server, and you decide to reboot the machine. Which command would you use in the console to reboot?
 - A. Reboot
 - B. Reset
 - C. Restart
 - D. Shutdown followed by Startup
4. In the console, you can filter the list of VM instances by which of the following?
 - A. Labels only
 - B. Member of managed instance group only
 - C. Labels, status, or deletion prevention
 - D. Labels and status only
5. You will be building several machine learning models on an instance and attaching GPU to the instance. When you run your machine learning models they take an unusually long time to run. It appears that GPU is not being used. What could be the cause of this?
 - A. GPU libraries are not installed.
 - B. The operating system is based on Ubuntu.
 - C. You do not have at least eight CPUs in the instance.
 - D. There isn't enough persistent disk space available.

6. When you add a GPU to an instance, you must ensure that:
 - A. The GPU and CPU choices are compatible.
 - B. The instance is preemptible.
 - C. The instance does not have nonboot disks attached.
 - D. The instance is running Ubuntu 18.02 or later.
7. You are using snapshots to save copies of a 100 GB disk. You make a snapshot and then add 10 GB of data. You create a second snapshot. How much storage is used in total for the two snapshots (assume no compression)?
 - A. 210 GB, with 100 GB for the first and 110 GB for the second
 - B. 110 GB, with 100 GB for the first and 10 GB for the second
 - C. 110 GB, with 110 GB for the second (the first snapshot is deleted automatically)
 - D. 221 GB, with 100 GB for the first, 110 GB for the second, plus 10 percent of the second snapshot (11 GB) for metadata overhead
8. You have decided to delegate the task of making backup snapshots to a member of your team. What role would you need to grant to your team member to create snapshots?
 - A. Compute Image Admin
 - B. Storage Admin
 - C. Compute Snapshot Admin
 - D. Compute Storage Admin
9. The source of an image may be:
 - A. Only disks
 - B. Snapshots or disks only
 - C. Disks, snapshots, or another image
 - D. Disks, snapshots, or any database export file
10. You have built images using Ubuntu 18.04 and now want users to start using Ubuntu 20.04. You don't want to just delete images based on Ubuntu 18.04, but you want users to know they should start using Ubuntu 20.04. What feature of images would you use to accomplish this?
 - A. Redirection
 - B. Deprecated
 - C. Unsupported
 - D. Migration
11. You want to generate a list of VMs in your inventory and have the results in JSON format. What command would you use?
 - A. `gcloud compute instances list`
 - B. `gcloud compute instances describe`
 - C. `gcloud compute instances list --format=json`
 - D. `gcloud compute instances list --output=json`

12. You would like to understand details of how Google Cloud starts a virtual instance. Which optional parameter would you use when starting an instance to display those details?
- A. `--verbose`
 - B. `--async`
 - C. `--describe`
 - D. `--details`
13. Which command will delete an instance named `ch06-instance-3`?
- A. `gcloud compute instances delete instance=ch06-instance-3`
 - B. `gcloud compute instance stop ch06-instance-3`
 - C. `gcloud compute instances delete ch06-instance-3`
 - D. `gcloud compute delete ch06-instance-3`
14. You are about to delete an instance named `ch06-instance-1` but want to keep its boot disk. You do not want to keep other attached disks. What `gcloud` command would you use?
- A. `gcloud compute instances delete ch06-instance-1\ --keep-disks=boot`
 - B. `gcloud compute instances delete ch06-instance-1\ --save-disks=boot`
 - C. `gcloud compute instances delete ch06-instance-1\ --keep-disks=filesystem`
 - D. `gcloud compute delete ch06-instance-1 --keep-disks=filesystem`
15. You want to view a list of fields you can use to sort a list of instances. What command would you use to see the field names?
- A. `gcloud compute instances list`
 - B. `gcloud compute instances describe`
 - C. `gcloud compute instances list --detailed`
 - D. `gcloud compute instances describe -detailed`
16. You are deploying an application that will need to scale and be highly available. Which of these Compute Engine components will help achieve scalability and high availability?
- A. Preemptible instances
 - B. Instance groups
 - C. Cloud Storage
 - D. GPUs
17. Before creating an instance group, what do you need to create?
- A. Instances in the instance group
 - B. Instance template
 - C. Boot disk image
 - D. Source snapshot

18. How would you delete an instance group using the command line?
- A. `gcloud compute instances instance-template delete`
 - B. `gcloud compute instance-templates delete`
 - C. `gcloud compute delete instance-template`
 - D. `gcloud compute delete instance-templates`
19. What can be the basis for scaling up an instance group?
- A. CPU utilization and operating system updates
 - B. Disk usage and CPU utilization only
 - C. Network latency, load balancing capacity, and CPU utilization
 - D. Disk usage and operating system updates only
20. An architect is moving a legacy application to Google Cloud and wants to minimize the changes to the existing architecture while administering the cluster as a single entity. The legacy application runs on a load-balanced cluster that runs nodes with two different configurations. The two configurations are required because of design decisions made several years ago. The load on the application is consistent, so there is rarely a need to scale up or down. What Google Cloud Compute Engine resource would you recommended using?
- A. Preemptible instances
 - B. Unmanaged instance groups
 - C. Managed instance groups
 - D. GPUs

Chapter 7

Computing with Kubernetes

THIS CHAPTER COVERS THE FOLLOWING OBJECTIVES OF THE GOOGLE ASSOCIATE CLOUD ENGINEER CERTIFICATION EXAM:

- ✓ 3.2 Deploying and implementing Google Kubernetes Engine resources





This chapter introduces Kubernetes, a container orchestration system created and open sourced by Google. You will learn about the architecture of Kubernetes and the ways it manages workloads across nodes in a cluster. You will also learn how to manage Kubernetes resources with Cloud Console, Cloud Shell, and Cloud SDK. The chapter also covers how to deploy application pods (a Kubernetes structure) and monitor and log Kubernetes resources.

Introduction to Kubernetes Engine

Kubernetes Engine is a Google Cloud–managed Kubernetes service. With this service, Google Cloud customers can create and maintain their own Kubernetes clusters without having to manage the Kubernetes platform. Google Kubernetes Engine is sometimes abbreviated as GKE.

Kubernetes runs containers on a cluster of virtual machines (VMs). It determines where to run containers, monitors the health of containers, and manages the full life cycle of VM instances. This collection of tasks is known as *container orchestration*.

It may sound as if a Kubernetes cluster is like an instance group, which was discussed in Chapter 6, “Managing Virtual Machines.” There are some similarities, and in fact, GKE uses instance groups to manage the underlying VMs in a GKE cluster.

Containers offer a highly portable, lightweight means of distributing and scaling your applications or workloads, like VMs, without replicating the guest OS. They can start and stop much faster (usually in seconds) and use fewer resources. You can think of a container as similar to shipping containers for applications and workloads. Like shipping containers that can ride on ships, trains, and trucks without reconfiguration, application containers can be moved from development laptops to testing and production servers without reconfiguration. Instance groups have configurable monitoring and can restart instances that fail, but Kubernetes has much more flexibility with regard to maintaining a cluster of servers.

Let’s look at Kubernetes architecture, which consists of several objects and a set of controllers.

Keep in mind that when you use Kubernetes Engine, you will manage Kubernetes and your applications and workloads running in containers on the Kubernetes platform.

Kubernetes Cluster Architecture

A Kubernetes cluster consists of a control plane and one or more worker machines called nodes. The control plane manages the cluster and can be replicated and distributed for high availability and fault tolerance.

The control plane manages services provided by Kubernetes, such as the Kubernetes API, controllers, and schedulers. All interactions with the cluster are done through the control plane using the Kubernetes API. The control plane issues the command that performs an action on a node. Users can also interact with a cluster using the `kubectl` command.

The basic components of Kubernetes are:

- API Server, which is a component of the control plane that exposes the Kubernetes API
- Scheduler, a control plane component that assigns pods to nodes
- Controller Manager, a control plane component that manages resource controllers, such as node controller, job controller, and service account controller
- etcd, a highly available key-value store
- Kubelet, an agent that runs on each node in a cluster
- Container Runtime, the software responsible for running containers
- Kube-proxy, a network proxy that runs on each node in the cluster

Nodes execute the workloads run on the cluster. Nodes are VMs that run containers configured to run an application. Nodes are primarily controlled by the control plane, but some commands can be run manually. The nodes run an agent called *kubelet*, which is the service that communicates with the control plane.

When you create a GKE cluster, you can specify a machine type. These VMs run specialized operating systems optimized to run containers. Some of the memory and CPU is reserved for Kubernetes and so is not available to applications running on the node.

Kubernetes organizes processing into workloads. There are several organizing objects that make up the core functionality of how Kubernetes processes workloads.

Kubernetes Objects

Workloads are distributed across nodes in a Kubernetes cluster. To understand how work is distributed, it is important to understand some basic concepts, in particular the following:

- Pods
- Services
- Deployments
- ReplicaSets
- StatefulSets

- Job
- Volumes
- Namespaces
- Node pools

Each of these objects contributes to the logical organization of workloads.

Pods

Pods are single instances of a running process in a cluster. Pods contain at least one container. They often run a single container but can run multiple containers. Multiple containers are used when two or more containers must share resources or are tightly coupled. Pods also use shared networking and storage across containers. Each pod gets a unique IP address and a set of ports. Containers connect to a port. Multiple containers in a pod connect to different ports and can talk to each other on localhost. This structure is designed to support running one instance of an application within the cluster as a pod. A pod allows its containers to behave as if they are running on an isolated VM, sharing common storage, one IP address, and a set of ports. By doing this, you can deploy multiple instances of the same application, or different instances of different applications on the same node or different nodes, without having to change their configuration.

Pods treat the multiple containers as a single entity for management purposes.

Pods are generally created in groups. Replicas are copies of pods and constitute a group of pods that are managed as a unit. Pods support autoscaling as well. Pods are considered ephemeral; that is, they are expected to terminate. If a pod is unhealthy—for example, if it is stuck in a waiting mode or crashing repeatedly—it is terminated. The mechanism that manages scaling and health monitoring is known as a *controller*.

Services

Since pods are ephemeral and can be terminated by a controller, other services that depend on pods should not be tightly coupled to particular pods. For example, even though pods have unique IP addresses, applications should not depend on that IP address to reach an application. If the pod with that address is terminated and another is created, it may have another IP address. The IP address may be reassigned to another pod running a different container.

Kubernetes provides a level of indirection between applications running in pods and other applications that call them: it is called a *service*. A service, in Kubernetes terminology, is an object that provides API endpoints with a stable IP address that allow applications to discover pods running a particular application. Services update when changes are made to pods, so they maintain an up-to-date list of pods running an application.

Deployments

Another important concept in Kubernetes is the deployment. Deployments are sets of identical pods. The members of the set may change as some pods are terminated and others are started, but they are all running the same application. The pods all run the same application because they are created using the same pod template.

A *pod template* is a definition of how to run a pod. The description of how to define the pod is a *pod specification*. Kubernetes uses this definition to keep a pod in the state specified in the template. That is, if the specification has a minimum number of pods that should be in the deployment and the number falls below that, then additional pods will be added to the deployment by calling on a ReplicaSet.

ReplicaSets

A ReplicaSet is a controller used by a deployment that ensures the correct number of identical pods are running. For example, if a pod is determined to be unhealthy, a controller will terminate that pod. The ReplicaSet will detect that not enough pods for that application or workload are running and will create another. ReplicaSets are also used to update and delete pods. In general, it is a good practice to use deployment rather than ReplicaSets unless you require custom update orchestration or do not require any updates at all.

StatefulSets

Deployments are well suited to stateless applications. Those are applications that do not need to keep track of their state. For example, an application that calls an API to perform a calculation on the input values does not need to keep track of previous calls or calculations. An application that calls that API may reach a different pod each time it makes a call. There are times, however, when it is advantageous to have a single pod respond to all calls for a client during a single session.

StatefulSets are like deployments, but they assign unique identifiers to pods. This enables Kubernetes to track which pod is used by which client and keep them together. StatefulSets are used when an application needs a unique network identifier or stable persistent storage.

Jobs

A job is an abstraction about a workload. Jobs create pods and run them until the application completes a workload. Job specifications are specified in a configuration file and include specifications about the container to use and what command to run.

Volumes

Volumes are a storage mechanism provided by Kubernetes. Volumes store data independently of the life of a pod. If a pod fails and is restarted, the contents of a volume attached to the failed pod will continue to exist after the pod is restarted, and that volume will be

attached to the new instance of the pod. This ensures that if a pod crashes or restarts, data saved to a volume will be available for the replacement pod. Volumes are also used to share files across containers running in a pod.

Namespaces

A namespace is a logical abstraction for separating groups of resources in a cluster. Namespaces are used when clusters host a variety of projects, teams, or other groups that may have different policies or requirements for using cluster resources. Kubernetes creates a default namespace, which is used for objects with no other namespace defined. Kubernetes also creates namespaces for administering the cluster.

Node Pools

A node pool is a set of nodes in a cluster that have the same configuration. When the cluster is first created, all nodes are in the same node pool. You can add other nodes and node pools after the cluster is created. Node pools are useful if you want to group nodes with similar features, such as nodes that run on preemptible virtual machines. A node pool of preemptible VMs would allow you to assign some workloads to nodes on those preemptible while preventing other workloads from running on them.

Now that you're familiar with how Kubernetes is organized and how workloads are run, we'll cover how to deploy a Kubernetes cluster using Kubernetes Engine.

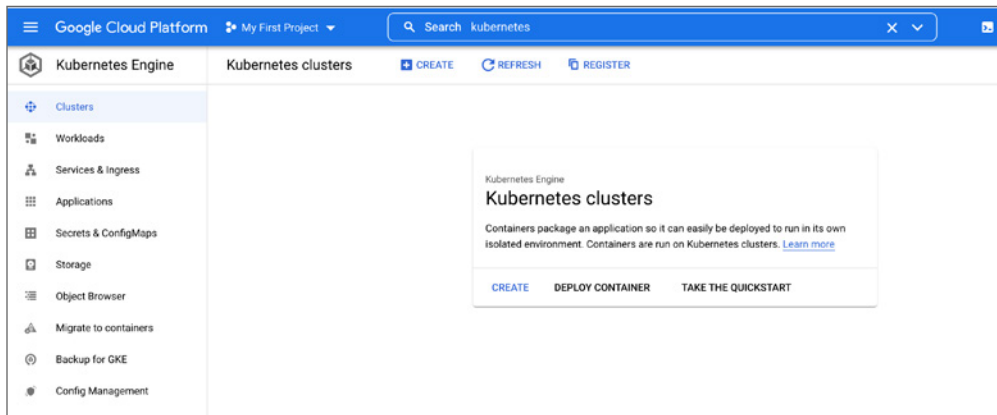
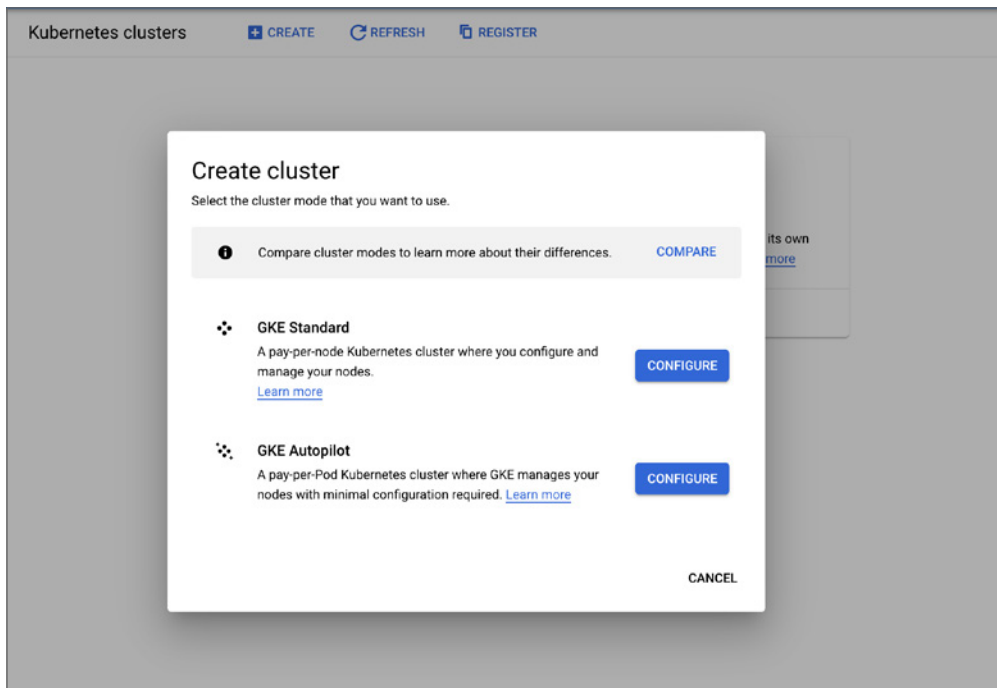
Deploying Kubernetes Clusters

Kubernetes clusters can be deployed using either Cloud Console or the command line in Cloud Shell, or your local environment if Cloud SDK is installed.

Deploying Kubernetes Clusters Using Cloud Console

To use Kubernetes Engine, you will need to enable the Kubernetes Engine API. Once you have enabled the API, you can navigate to the Kubernetes Engine page in Cloud Console. Figure 7.1 shows the Overview page.

When you create a cluster, you will have the option to create the cluster in standard mode or autopilot mode. In standard mode you pay for the cluster resources you provision, manage the node infrastructure, and determine the configuration of the nodes. In autopilot mode, GKE manages the cluster and node infrastructure and you pay only for the resources used when your applications are running. Autopilot mode clusters use preconfigured and optimized cluster configurations (see Figure 7.2). Autopilot is the recommended mode for using GKE.

FIGURE 7.1 The Overview page of the Kubernetes Engine section of Cloud Console**FIGURE 7.2** When creating a GKE, you specify standard mode or autopilot mode.

When you click an autopilot cluster, GKE will automatically manage and configure node infrastructure, VPC-native traffic routing for public and private clusters, use Shielded GKE nodes, as well as logging and monitoring. You will specify a cluster name, cluster description,

and a region. You also specify if the cluster is private or public. In private clusters, nodes only have private IP addresses and all communication between the control plane and node are via private addresses only. See Figure 7.3.

FIGURE 7.3 Creating an autopilot GKE cluster

← Create an Autopilot cluster

Create an Autopilot cluster by specifying a name and region. After the cluster is created, you can deploy your workload through Kubernetes and we'll take care of the rest, including:

- ✓ **Nodes:** Automated node provisioning, scaling, and maintenance
- ✓ **Networking:** VPC-native traffic routing for public or private clusters
- ✓ **Security:** Shielded GKE Nodes and Workload Identity
- ✓ **Telemetry:** Cloud Operations logging and monitoring

Name

autopilot-cluster-1

?

Region

us-central1

▼

?

Networking

Define how applications in this cluster communicate with each other and how clients can reach them.

☒ Public cluster

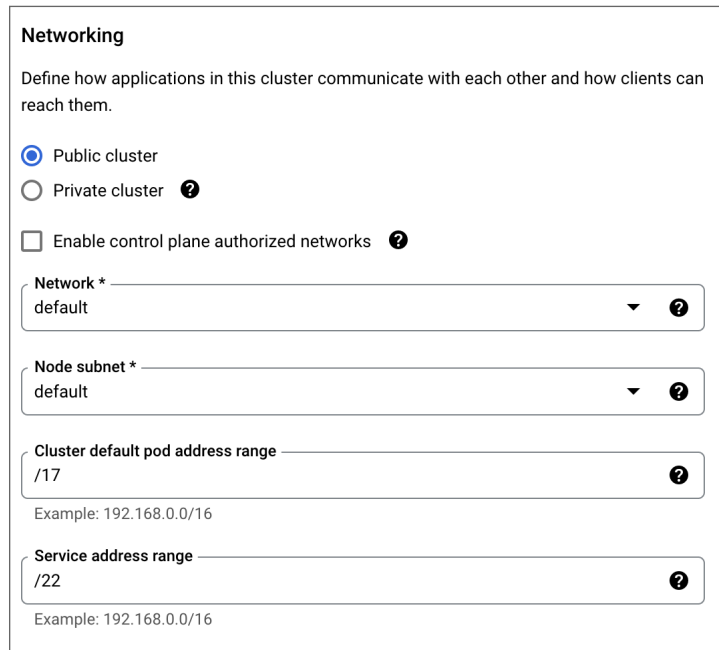
☐ Private cluster ?

▼ NETWORKING OPTIONS

▼ ADVANCED OPTIONS

Click **Create** to create the cluster with these settings turned on.

Expanding the Networking Options area in the Create An Autopilot Cluster page shows additional network configurations, as you can see in Figure 7.4. You can enable control plane-authorized networks to block nontrusted non-Google Cloud source IP addresses from accessing the control plane using HTTPS. You can also specify a network, node subnet, and address ranges for pods and services. When specifying address ranges, you use CIDR notation; for example, 192.168.0.0/16.

FIGURE 7.4 Networking options in autopilot mode

Networking

Define how applications in this cluster communicate with each other and how clients can reach them.

☒ Public cluster

☐ Private cluster ?

☐ Enable control plane authorized networks ?

Network *
default ?

Node subnet *
default ?

Cluster default pod address range
/17 ?
Example: 192.168.0.0/16

Service address range
/22 ?
Example: 192.168.0.0/16

In the Advanced Options area in the Create An Autopilot Cluster page, you can specify a maintenance window to specify a time for running routine Kubernetes maintenance operations. By default, these operations can run at any time. You can also enable security features, including Google Groups for RBAC, to grant roles to members of a Google Workspace Group, application-layer secrets encryption to encrypt secrets stored in etcd (part of the control plane), and enable the use of a customer-managed key to encrypt the boot disk of nodes. You can add labels and a description to the cluster. See Figure 7.5.

From the listing of clusters, you can edit, delete, and connect to a cluster. When you click Connect, you receive a `gcloud` command to connect to the cluster from the command line. You also have the option of viewing the Workloads page, as shown in Figure 7.6.

When you choose to configure a standard mode cluster using the cloud console, you will see a form like that shown in Figure 7.7. You will specify a name and location of the cluster. If you choose to create a zonal cluster, the location will be a zone. If you choose to create a regional cluster, the location will be a region. Regional clusters by default have nodes in three zones, but you can specify default node locations if you want to specify specific zones to run nodes.

By default, clusters are created with a release channel configuration, which enables automatic upgrading of the cluster software. If you want more control over the upgrade process, you can choose to configure a static channel. See Figure 7.7.

FIGURE 7.5 Advanced options in autopilot mode

Automation

☐ Enable Maintenance Window ?

* Indicates required field

[+ ADD MAINTENANCE EXCLUSION](#)

Security

☐ Enable Google Groups for RBAC ?

☐ Enable Application-layer Secrets Encryption ?

☐ Enable customer-managed encryption for boot disk ?

Metadata

Add a description and labels to organize your cluster.

Description

?

Labels

To organize your project, add arbitrary labels as key/value pairs to your resources. Use labels to indicate different environments, services, teams, and so on. [Learn more](#)

[+ ADD LABEL](#)

[^ HIDE ADVANCED OPTIONS](#)

FIGURE 7.6 Once the autopilot clusters are deployed, it will be listed on the GKE page of the console.

<div>Kubernetes Engine</div> <div>Clusters</div> <div>Workloads</div> <div>Services & Ingress</div> <div>Applications</div> <div>Secrets & ConfigMaps</div>	Kubernetes clusters CREATE DEPLOY REFRESH REGISTER OPERATIONS HELP ASSISTANT									
	<div>OVERVIEW</div> <div>COST OPTIMIZATION</div>									
	<div>Filter</div> Enter property name or value									
	<input type="checkbox"/>	Status	<input type="checkbox"/>	Name	Location	Mode	Number of nodes	Total vCPUs	Total memory	Notifications
	<input checked="" type="checkbox"/>			autopilot-cluster-1	us-central1	Autopilot		0	0 GB	—

FIGURE 7.7 Initial steps to configure a standard cluster

Cluster basics

The new cluster will be created with the name, version, and in the location you specify here. After the cluster is created, name and location can't be changed.

❗

To experiment with an affordable cluster, try **My first cluster** in the **Cluster set-up guides**

Name

cluster-1

?

Location type

☒ Zonal

☐ Regional

Zone

us-central1-c

▼

?

☐ Specify default node locations

?

Current default: us-central1-c

Control plane version

Choose a release channel for automatic management of your cluster's version and upgrade cadence. Choose a static version for more direct management of your cluster's version. [Learn more.](#)

☐ Static version

☒ Release channel

Release channel

Regular channel (default)

▼

Version

1.21.9-gke.1002 (default)

▼

Deploying Kubernetes Clusters Using Cloud Shell and Cloud SDK

Like other Google Cloud services, Kubernetes Engine can be managed using the command line. The basic command for working with Kubernetes Engine is the following `gcloud` command:

```
gcloud container
```

This `gcloud` command has many parameters, including the following:

- Project
- Zone
- Machine type
- Image type
- Disk type
- Disk size
- Number of nodes

A basic command for creating a standard mode cluster looks like this:

```
gcloud container clusters create cluster1 --num-nodes=3 --region=us-central1
```

There are a large number of parameters for the `gcloud container clusters create` command that allow you to specify many different configurations for a cluster. For details on the parameters, visit <https://cloud.google.com/sdk/gcloud/reference/container/clusters/create>.

The command `gcloud container clusters create-auto` is used to create auto-pilot mode GKE clusters.

Deploying Application Pods

Now that you have created a cluster, let's deploy an application.

From the Clusters page of Kubernetes Engine on Cloud Console, select Create Deployment. A form such as the one in Figure 7.8 appears. Use this form to specify the following:

- Container image
- Environment variables
- Initial command

After specifying the initial parameters, you can continue to add configuration parameters (see Figure 7.9):

- Application name
- Namespace
- Labels
- Cluster

FIGURE 7.8 The Create Deployment option provides a form to specify a container to run and an initial command to start the application running.

The screenshot shows a web interface for creating a deployment. At the top, there's a back arrow and the title 'Create a deployment'. Below this, a progress indicator shows '1 Container' and '2 Configuration'. The 'Container' section is active and contains a modal titled 'Edit container'. Inside this modal, there are two radio buttons: 'Existing container image' (selected) and 'New container image'. Below the radio buttons is a text input field for 'Image path *' with the value 'nginx:latest' and a 'SELECT' button. A note below the input field says: 'Enter your image path, or choose from Google Container Registry. You can also try to deploy with official nginx image nginx:latest.' Below the modal, there's a section for 'Environment variables' with a '+ ADD ENVIRONMENT VARIABLE' button. Underneath that is an 'Initial command' text input field with a note: 'Overrides the default entrypoint of the container image.' At the bottom right of the modal are 'CANCEL' and 'DONE' buttons. Below the modal, there's an 'ADD CONTAINER' button. At the bottom of the 'Container' section is a 'CONTINUE' button.

Once you have specified a deployment, you can display the corresponding YAML specification, which can be saved and used to create deployments from the command line. The core elements of the Kubernetes template include `apiVersion`, `kind`, `metadata`, and `spec`. Listing 7.1 shows an example deployment YAML file. The output is always displayed in YAML format.

FIGURE 7.9 Configuring a deployment

[←](#) Create a deployment

2 Configuration

A deployment is a configuration which defines how Kubernetes deploys, manages, and scales your container image. Kubernetes will ensure your system matches this configuration.

Application name *
nginx-1

Namespace *
default

Labels

Key 1 *
app

Value 1
nginx-1

[+ ADD KUBERNETES LABEL](#)

Configuration YAML

Kubernetes deployments are defined declaratively using YAML files. The best practice is to store these files in version control, so you can track changes to your deployment configuration over time.

[VIEW YAML](#)

Cluster

Kubernetes Cluster
autopilot-cluster-1 (us-central1) ▼

Cluster in which the deployment will be created.

[CREATE NEW CLUSTER](#)

[DEPLOY](#)

Listing 7.1: Sample YAML configuration specification for a deployment

```
apiVersion: "apps/v1"
kind: "Deployment"
metadata:
  name: "nginx-1"
  namespace: "default"
  labels:
    app: "nginx-1"
spec:
  replicas: 3
  selector:
    matchLabels:
      app: "nginx-1"
  template:
    metadata:
      labels:
        app: "nginx-1"
    spec:
      containers:
        - name: "nginx-1"
          image: "nginx:latest"
---
apiVersion: "autoscaling/v2beta1"
kind: "HorizontalPodAutoscaler"
metadata:
  name: "nginx-1-hpa-5fkn"
  namespace: "default"
  labels:
    app: "nginx-1"
spec:
  scaleTargetRef:
    kind: "Deployment"
    name: "nginx-1"
    apiVersion: "apps/v1"
  minReplicas: 1
  maxReplicas: 5
  metrics:
```

```
- type: "Resource"
  resource:
    name: "cpu"
    targetAverageUtilization: 80
```

In addition to installing Cloud SDK, you will need to install the Kubernetes command-line tool `kubectl` to work with clusters from the command line. You can do this with the following command:

```
gcloud components install kubectl
```

You can then use `kubectl` to run a Docker image on a cluster by using the `kubectl run` command. To run a container within a deployment, use the `create deployment` command. Here's an example:

```
kubectl create deployment app-deploy1 --image=app1 --port=8080
```

This will run a Docker image called `app1` and make its network accessible on port 8080. If after some time you'd like to scale up the number of replicas in the deployment, you can use the `kubectl scale` command:

```
kubectl scale deployment app-deploy1 --replicas=5
```

This example would create five replicas.

Monitoring Kubernetes

Cloud Operations Suite is Google Cloud's comprehensive monitoring, logging, and alerting product and includes Cloud Monitoring and Cloud Logging services, which can be used to monitor Kubernetes clusters.

GKE provides for multiple sources of application and system performance metrics, including System metrics, Managed Service for Prometheus, and Workload metrics. System metrics describe low-level cluster resources such as CPUs, memory, and storage. Prometheus is a widely used open source system for collecting performance metrics. The Managed Service for Prometheus is a service provided by Google Cloud for customers who want to use Prometheus but who do not want to manage the infrastructure and applications that make up Prometheus. Workload metrics are a set of deprecated metrics exposed by GKE workloads.

When you create a cluster, you can indicate that metrics be sent to Cloud Monitoring and logs be sent to Cloud Logging. Both are enabled by default.

Summary

Kubernetes Engine is a container orchestration system for deploying applications to run in clusters. Kubernetes is architected with a single cluster manager and worker nodes.

Kubernetes uses the concept of pods, or instances running a container. It is possible to run multiple containers in a pod, but this is usually only done when the two containers are

tightly coupled. ReplicaSets are controllers for ensuring that the correct number of pods are running. Deployments are sets of identical pods. StatefulSets are a type of deployment used for stateful applications.

Kubernetes clusters can be deployed through Cloud Console or by using `gcloud` commands. You deploy applications by bundling the application in a container and using the console or the `kubectl` command to create a deployment that runs the application on the cluster.

Cloud Operations Suite includes Cloud Monitoring and Cloud Logging, which is used to monitor instances in clusters.

Exam Essentials

Understand that Kubernetes is a container orchestration system. Kubernetes Engine is a Google Cloud product that provides Kubernetes to Google Cloud customers. Kubernetes manages containers that run in a set of VM instances.

Understand that Kubernetes uses a control plane to manage nodes and workloads. Kubernetes uses the control plane to coordinate execution and monitor the health of pods. If there is a problem with a pod, the control plane can correct the problem and reschedule the disrupted job.

Be able to describe pods. Pods are single instances of a running process, services provide a level of indirection between pods and clients calling services in the pods, a ReplicaSet is a kind of controller that ensures that the right number of pods are running, and a deployment is a set of identical pods.

Kubernetes can be deployed using Cloud Console or using `gcloud` commands. `gcloud` commands manipulate the Kubernetes Engine service, whereas `kubectl` commands are used to manage the internal state of clusters from the command line. The base command for working with Kubernetes Engine is `gcloud container`. Note that `gcloud` and `kubectl` have different command syntaxes. `kubectl` commands specify a verb and then a resource, as in `kubectl scale deployment ...`, whereas `gcloud` specifies a resource before the verb, as in `gcloud container clusters create`. Deployments are created using Cloud Console or at the command line using a YAML specification.

Be able to define Kubernetes objects. Deployments are sets of identical pods. StatefulSets are a type of deployment used for stateful applications. Kubernetes is monitored using Cloud Operations. Cloud Operations can be configured to generate alerts and notify you on a variety of channels. To monitor the state of a cluster, you can create a policy that monitors a metric, like CPU utilization, and have notifications sent to email or other channels.

Review Questions

You can find the answers in the Appendix.

1. A new engineer is asking for clarification about when it is best to use Kubernetes and when to use instance groups. You point out that Kubernetes uses instance groups. What purpose do instance groups play in a Kubernetes cluster?
 - A. They monitor the health of instances.
 - B. They create pods and deployments.
 - C. They create sets of VMs that can be managed as a unit.
 - D. They create alerts and notification channels.
2. What components are required in a Kubernetes cluster?
 - A. A control plane and nodes to execute workloads.
 - B. A control plane, nodes to execute workloads, and monitoring nodes to monitor node health.
 - C. Kubernetes nodes; all instances are the same.
 - D. Instances with at least six vCPUs.
3. What is a pod in Kubernetes?
 - A. A set of containers
 - B. Application code deployed in a Kubernetes cluster
 - C. A single instance of a running application in a cluster
 - D. A controller that manages communication between clients and Kubernetes services
4. You have developed an application that calls a service running in a Kubernetes cluster. The service runs in pods that can be terminated if they are unhealthy and replaced with other pods that might have a different IP address. How should you code your application to ensure it functions properly in this situation?
 - A. Query Kubernetes for a list of IP addresses of pods running the service you use.
 - B. Communicate with Kubernetes Services so that applications do not have to be coupled to specific pods.
 - C. Query Kubernetes for a list of pods running the service you use.
 - D. Use a `gcloud` command to get the IP addresses needed.
5. You have noticed that an application's performance has degraded significantly. You have recently made some configuration changes to resources in your Kubernetes cluster and suspect that those changes have altered the number of pods running in the cluster. Where would you look for details on the number of pods that should be running?
 - A. Deployment config
 - B. Cloud Operations Suite
 - C. Container Runtime
 - D. Jobs

6. You are deploying a high-availability application in Kubernetes Engine. You want to maintain availability even if there is a major network outage in a data center. What feature of Kubernetes Engine would you employ?
 - A. Multiple instance groups
 - B. Regional cluster
 - C. Regional deployments
 - D. Load balancing
7. You want to write a script to deploy a Kubernetes cluster with GPUs. You have deployed clusters before, but you are not sure about all the required parameters. You need to deploy this script as quickly as possible. What is one way to develop this script quickly?
 - A. Use the GPU template in the Kubernetes Engine cloud console to generate the `gcloud` command to create the cluster.
 - B. Search the web for a script.
 - C. Review the documentation on `gcloud` parameters for adding GPUs.
 - D. Use an existing script and add parameters for attaching GPUs.
8. What `gcloud` command will create a cluster named `ch07-cluster-1` with four nodes?
 - A. `gcloud container clusters create ch07-cluster-1 --num-nodes=4`
 - B. `gcloud container clusters create ch07-cluster-1 --size=4`
 - C. `gcloud container clusters create ch07-cluster-1 --region-nodes=4`
 - D. `gcloud beta container clusters create ch07-cluster-1 4`
9. When using Create Deployment from Cloud Console, which of the following cannot be specified for a deployment?
 - A. Container image
 - B. Application name
 - C. Time to Live (TTL)
 - D. Initial command
10. Deployment configuration files created in Cloud Console use what type of file format?
 - A. CSV
 - B. YAML
 - C. TSV
 - D. JSON
11. What command is used to run a Docker image on a cluster?
 - A. `gcloud container run`
 - B. `gcloud container clusters run`
 - C. `kubectl run`
 - D. `kubectl container run`

12. What command would you use to have 10 replicas of a deployment named `ch07-app-deploy`?
 - A. `kubectl upgrade deployment ch07-app-deploy --replicas=5`
 - B. `gcloud containers deployment ch07-app-deploy --replicas=5`
 - C. `kubectl scale deployment ch07-app-deploy --replicas=10`
 - D. `kubectl scale deployment ch07-app-deploy --pods=5`
13. Cloud Operations Suite is used for what operations on Kubernetes clusters?
 - A. Notifications only
 - B. Monitoring and notifications only
 - C. Logging only
 - D. Notifications, monitoring, and logging
14. You want to use Cloud Logging and Cloud Monitoring with your GKE clusters. What must you do to enable this when creating a cluster?
 - A. Specify the `--monitoring=True` and `--logging=True` parameters in the `gcloud container create cluster` command.
 - B. Create a node pool and configure it for monitoring and logging.
 - C. Create a namespace and configure it for monitoring and logging.
 - D. Nothing; metrics and logs are sent to Cloud Logging and Cloud Monitoring by default.
15. What popular open source monitoring tool is available in Google Cloud as a managed service?
 - A. Prometheus
 - B. Apache Flink
 - C. MongoDB
 - D. Spark
16. You want to create a Kubernetes Engine cluster and want to minimize the amount of configuring and infrastructure management. What kind of cluster would you create?
 - A. Standard mode cluster
 - B. Autopilot mode cluster
 - C. Minimal mode cluster
 - D. Template mode cluster
17. You want the greatest degree of control over your Kubernetes cluster. What kind of cluster would you create?
 - A. Standard mode cluster
 - B. Autopilot mode cluster
 - C. Minimal mode cluster
 - D. Template mode cluster

- 18.** You want to create a Kubernetes cluster, but you do not want GKE to automatically upgrade the cluster. How would you configure the cluster?
- A.** With a release channel
 - B.** With a static channel
 - C.** With multiple node pools
 - D.** With a ReplicaSet
- 19.** You are attempting to execute commands to initiate a deployment on a Kubernetes cluster. The commands are not having any effect. You suspect that a Kubernetes component is not functioning correctly. What component could be the problem?
- A.** The Kubernetes API
 - B.** A StatefulSet
 - C.** Cloud SDK `gcloud` commands
 - D.** ReplicaSet
- 20.** You have deployed an application to a Kubernetes cluster. You have noticed that several pods are starved for resources for a period of time and the pods are shut down. When resources are available, new instantiations of those pods are created. Clients are still able to connect to pods even though the new pods have different IP addresses from the pods that were terminated. What Kubernetes component makes this possible?
- A.** Services
 - B.** ReplicaSet
 - C.** Alerts
 - D.** StatefulSet

Chapter 8

Managing Standard Mode Kubernetes Clusters

**THIS CHAPTER COVERS THE FOLLOWING
OBJECTIVE OF THE GOOGLE ASSOCIATE
CLOUD ENGINEER CERTIFICATION EXAM:**

- ✓ 4.2 Managing Google Kubernetes Engine resources





This chapter describes how to perform basic Kubernetes management tasks, including the following:

- Viewing the status of Kubernetes clusters
- Viewing image repositories and image details
- Adding, modifying, and removing nodes
- Adding, modifying, and removing pods
- Adding, modifying, and removing services

You'll see how to perform each of these tasks using Google Cloud Console and Cloud SDK, which you can use locally on your development machines, on Google Cloud virtual machines, and by using Cloud Shell.

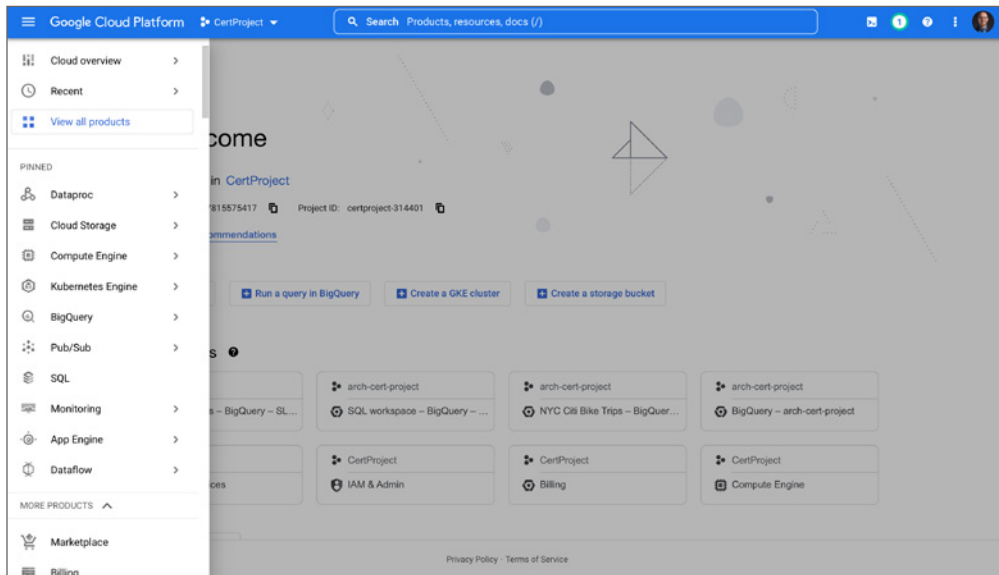
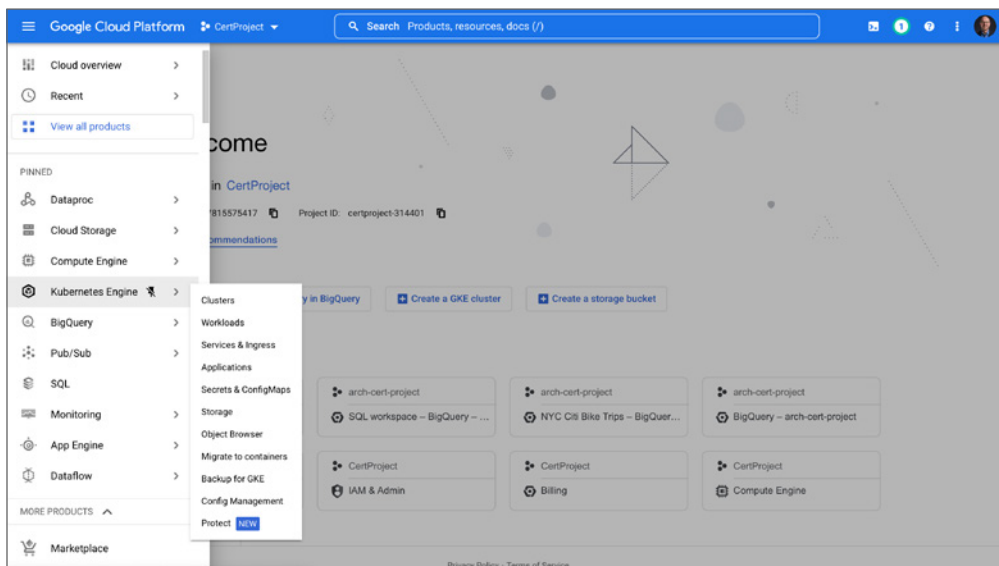
Viewing the Status of a Kubernetes Cluster

Assuming you have created a cluster using the steps outlined in Chapter 7, “Computing with Kubernetes,” you can view the status of a Kubernetes cluster using either Google Cloud Console or the `gcloud` commands.

Viewing the Status of Kubernetes Clusters Using Cloud Console

Starting from the Cloud Console home page, open the navigation menu by clicking the three stacked lines icon in the upper-left corner. This displays the list of Google Cloud services, as shown in Figure 8.1.

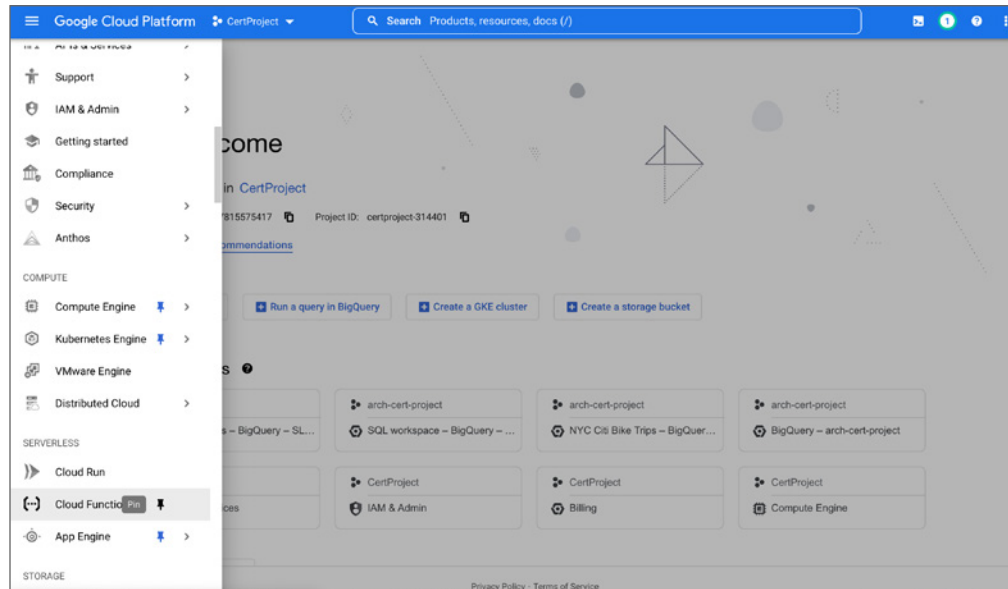
Select Kubernetes Engine from the lists of services to open the submenu shown in Figure 8.2.

FIGURE 8.1 Navigation menu in Google Cloud Console**FIGURE 8.2** Selecting Kubernetes Engine from the navigation menu

Pinning Services to the Top of the Navigation Menu

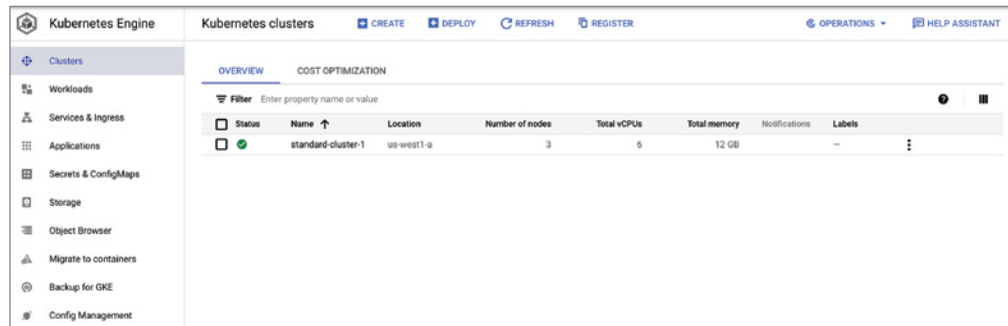
In Figure 8.2, Kubernetes Engine has been *pinned* so it is displayed at the top. You can pin any service in the navigation menu by mousing over the product and clicking the pin icon that appears, as shown in Figure 8.3. In that figure, Compute Engine and Kubernetes Engine are already pinned, and Cloud Functions can be pinned by clicking the gray pin icon.

FIGURE 8.3 Pinning a service to the top of the navigation menu



After clicking Kubernetes Engine in the navigation menu, you will see a list of running clusters, as shown in Figure 8.4, which shows a single cluster called `standard-cluster-1`.

FIGURE 8.4 Example list of clusters in Kubernetes Engine



Hover over the name of the cluster to highlight it, as in Figure 8.5, and click the name to display details of the cluster, as shown in Figure 8.6.

FIGURE 8.5 Click the name of a cluster to display its details.

OVERVIEW			
OBSERVABILITY			
COST OPTIMIZATION			
Filter Enter property name or value			
Status	Name ↑	Location	Number of nodes
<input checked="" type="checkbox"/>	standard-cluster-1	us-west1-a	3

FIGURE 8.6 The first part of the cluster Details page describes the configuration of the cluster.

Kubernetes Engine

Clusters

Workloads

Services & Ingress

Applications

Secrets & ConfigMaps

Storage

Object Browser

Migrate to Containers

Backup for GKE NEW

Security Posture

Config & Policy

Config

Marketplace

Release Notes

←

Clusters

EDIT

DELETE

ADD NODE POOL

DEPLOY

⋮

OPERATIONS

✓

standard-cluster-1

DETAILS

NODES

STORAGE

OBSERVABILITY

LOGS

Cluster basics

Name	standard-cluster-1	
Location type	Zonal	
Control plane zone	us-west1-a	
Default node zones	us-west1-a	
Release channel	Regular channel	UPGRADE AVAILABLE
Version	1.24.7-gke.900	
Total size	3	
External endpoint	34.168.24.163 Show cluster certificate	
Internal endpoint	10.138.0.10 Show cluster certificate	

Automation

Maintenance window	Any time	
Maintenance exclusions	None	
Notifications	Disabled	
Vertical Pod Autoscaling	Disabled	

Clicking the Nodes option shows details of node pools and replicas (see Figure 8.7). In the Node Pools section, you will see the number of nodes, machine type, image type and other attributes. In the Nodes section, you will see nodes and their status, requested and allocatable CPUs, memory, and storage.

FIGURE 8.7 Add-on and permission details for a cluster

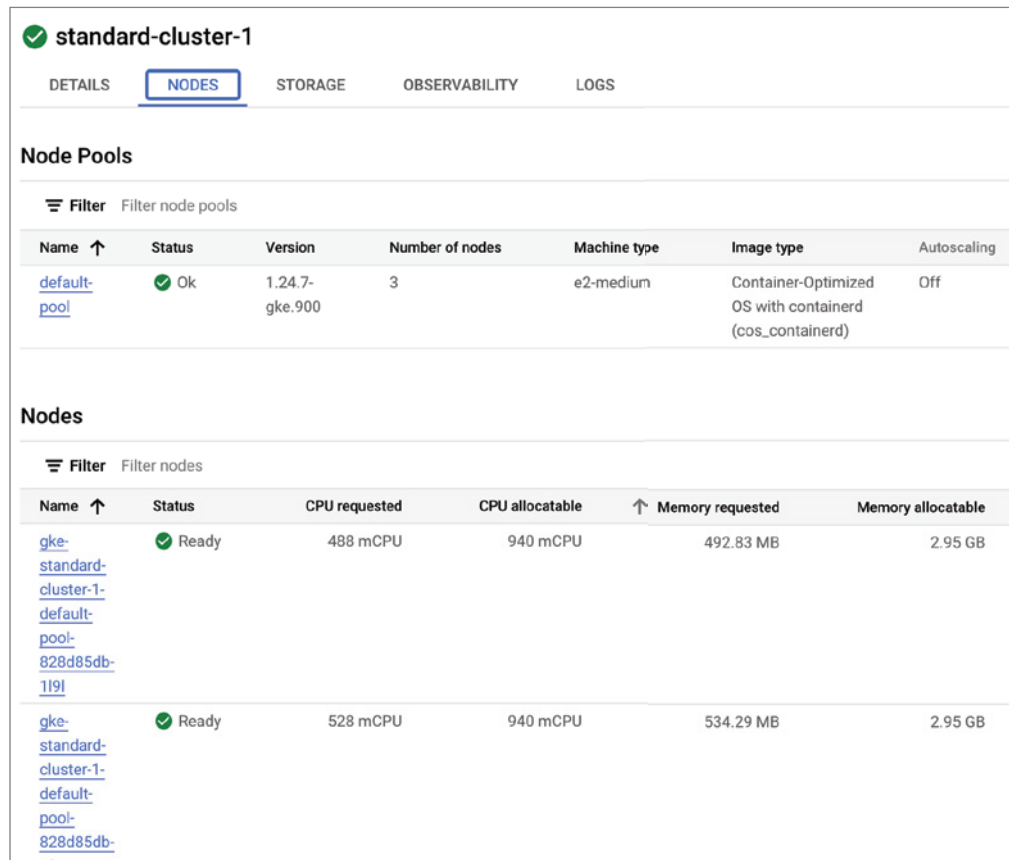


Figure 8.8 shows example details of node pools, which are separate instance groups running in a Kubernetes cluster. The details in this section include the node image running on the nodes, the machine type, the total number of vCPUs (listed as Total Cores), the disk type, and whether the nodes are preemptible.

Below the name of the cluster is a horizontal list of several options: Details, Nodes, Storage, Observability and Logs. So far, we have described the contents of the Details page. Click Storage to display information like that shown in Figure 8.9, which displays persistent volumes and the storage classes used by the cluster.

This cluster does not have persistent volumes but uses standard storage. Persistent volumes are durable disks that are managed by Kubernetes and implemented using Compute Engine persistent disks. A storage class is a type of storage with a set of policies specifying quality of service, backup policy, and a provisioner (which is a service that implements the storage).

FIGURE 8.8 Details about node pools in the cluster

Node pool details

REFRESH

EDIT

DELETE

RESIZE

default-pool

Cluster

standard-cluster-1

Node version

1.21.11-gke.1100

Size

Number of nodes

3

Autoscaling

Off

Node zones

us-west1-a

Nodes

Image type

Container-Optimized OS with containerd (cos_containerd)

Machine type

e2-medium

Boot disk type

Standard persistent disk

Boot disk size (per node)

100 GB

Boot disk encryption

Google-managed

Provisioning Model

Standard

Networking

Pod IP Address Range

10.36.0.0/14 (gke-standard-cluster-1-pods-d3f4a502)
(inherited from standard-cluster-1)

Maximum Pods per Node

110 (inherited from standard-cluster-1)

Management

Auto-upgrade ?

Enabled

Auto-repair ?

Enabled

Surge upgrade ?

Enabled

Max surge

1

Max unavailable

0

The Observability section shows metrics about cluster performance. Under the Logs option of the cluster status menu, you can see a log of messages, as shown in Figure 8.10.

Click the name of one of the nodes to see detailed status information, as shown in Figure 8.11. The node details include CPU utilization, memory consumption, and disk I/O. There is also a list of pods running on the node.

Click the name of a pod to see its details. The pod display is similar to the node display, with CPU, memory, and disk statistics. Configuration details include when the pod was created, the labels assigned, links to logs, and the status (which is shown as Running in Figure 8.12).

Other possible statuses are Pending, which indicates the pod is downloading images; Succeeded, which indicates the pod terminated successfully; Failed, which indicates at least one container failed; and Unknown, which means the control plane cannot reach the node and status cannot be determined.

At the bottom of the pod display is a list of containers running. Click the name of a container to see its details. Figure 8.13 shows the details of a Pod. Information includes the status, the start time, the command that is running, and the volumes mounted.

FIGURE 8.9 Storage information about a cluster

standard-cluster-1

DETAILS

NODES

STORAGE

OBSERVABILITY

LOGS

Storage classes

GKE automatically deploys and manages the Kubernetes Filestore Container Storage Interface (CSI) driver. Enable the CSI driver to add Filestore (NFS) storage. If enabled, Filestore storage classes will appear in the table below. [Learn more](#)

Filter

Filter storage classes

Name ↑	Provisioner	Type	Zone
premium-rwo	pd.csi.storage.gke.io	pd-ssd	
standard	kubernetes.io/gce-pd	pd-standard	
standard-rwo	pd.csi.storage.gke.io	pd-balanced	

Persistent volumes

Filter

Filter persistent volumes

Name ↑	Status	Type	Source	Read only	Storage Class	Claim
No persistent volume to display. Use Cloud Shell for YAML file creation and kubectl operations.						

FIGURE 8.10 Log of nodes in the cluster

← Clusters

EDIT

DELETE

ADD NODE POOL

DEPLOY

CONNECT

DUPLICATE

OPERATIONS

HELP ASSISTANT

standard-cluster-1

DETAILS

NODES

STORAGE

LOGS

You can find your general cluster logs and your cluster's Autoscaler logs below.

CLUSTER LOGS

AUTOSCALER LOGS

Showing 30 log entries

Severity

Default

Filter logs

2022-05-27T15:04:11.884153Z

Kubernetes Apiservice Requests

update

kube-node-lease:gke-standard-cluster-

system:node:gke-standard-cluster-

(@type: type.googleapis.com/go-

2022-05-27T15:04:12.202448Z

Kubernetes Apiservice Requests

update

kube-system:vpa-recommender

system:vpa-recommender

(@type: type.googleapis.com/google.cloud.audit.AuditLog, au-

2022-05-27T15:04:12.566576Z

Kubernetes Apiservice Requests

update

kube-system:clustermetrics

system:clustermetrics

(@type: type.googleapis.com/google.cloud.audit.AuditLog, auth-

2022-05-27T15:04:12.562854Z

Kubernetes Apiservice Requests

update

kube-system:ingress-gce-lock

system:l7-lb-controller

(@type: type.googleapis.com/google.cloud.audit.AuditLog, ..

2022-05-27T15:04:12.634946Z

Kubernetes Apiservice Requests

update

kube-system:managed-certificate-contr_

system:managed-certificate-controller

(@type: type.googleapis.com/goog-

2022-05-27T15:04:12.642112Z

Kubernetes Apiservice Requests

update

kube-system:managed-certificate-contr_

system:managed-certificate-controller

(@type: type.googleapis.com/goog-

2022-05-27T15:04:12.717126Z

Kubernetes Apiservice Requests

update

kube-system:cluster-kubestore

system:kubestore-collector

(@type: type.googleapis.com/google.cloud.audit.AuditL...

2022-05-27T15:04:12.666317Z

Kubernetes Apiservice Requests

update

kube-system:cluster-autoscaler

system:cluster-autoscaler

(@type: type.googleapis.com/google.cloud.audit.AuditL...

2022-05-27T15:04:13.081985Z

Kubernetes Apiservice Requests

update

kube-system:snapshot-controller-leader

system:snapshot-controller

(@type: type.googleapis.com/google.cloud.aud...

2022-05-27T15:04:13.253134Z

Kubernetes Apiservice Requests

update

kube-system:kube-controller-manager

system:kube-controller-manager

(@type: type.googleapis.com/google.cloud.au...

2022-05-27T15:04:13.258708Z

Kubernetes Apiservice Requests

update

kube-system:kube-scheduler

system:kube-scheduler

(@type: type.googleapis.com/google.cloud.audit.Auditlog, auth-

2022-05-27T15:04:14.528855Z

Kubernetes Apiservice Requests

update

kube-system:clustermetrics

system:clustermetrics

(@type: type.googleapis.com/google.cloud.audit.Auditlog, auth-

2022-05-27T15:04:14.577963Z

Kubernetes Apiservice Requests

update

kube-system:ingress-gce-lock

system:l7-lb-controller

(@type: type.googleapis.com/google.cloud.audit.Auditlog, ..

2022-05-27T15:04:14.656655Z

Kubernetes Apiservice Requests

update

kube-system:managed-certificate-contr_

system:managed-certificate-controller

(@type: type.googleapis.com/goog-

2022-05-27T15:04:14.736022Z

Kubernetes Apiservice Requests

update

kube-system:cluster-kubestore

system:kubestore-collector

(@type: type.googleapis.com/google.cloud.audit.AuditL...

2022-05-27T15:04:15.269895Z

Kubernetes Apiservice Requests

update

kube-system:kube-controller-manager

system:kube-controller-manager

(@type: type.googleapis.com/google.cloud.au...

2022-05-27T15:04:16.215854Z

Kubernetes Apiservice Requests

update

kube-system:vpa-recommender

system:vpa-recommender

(@type: type.googleapis.com/google.cloud.audit.Auditlog, au...

2022-05-27T15:04:16.591108Z

Kubernetes Apiservice Requests

update

kube-system:ingress-gce-lock

system:l7-lb-controller

(@type: type.googleapis.com/google.cloud.audit.Auditlog, ..

2022-05-27T15:04:16.675467Z

Kubernetes Apiservice Requests

update

kube-system:managed-certificate-contr_

system:managed-certificate-controller

(@type: type.googleapis.com/goog-

metes/clusters/details/us-west1-a/standard-cluster-1/logs?authuser=1&project=scenic-energy-335022

kubestore

system:kubestore-collector

(@type: type.googleapis.com/google.cloud.audit.Auditl...

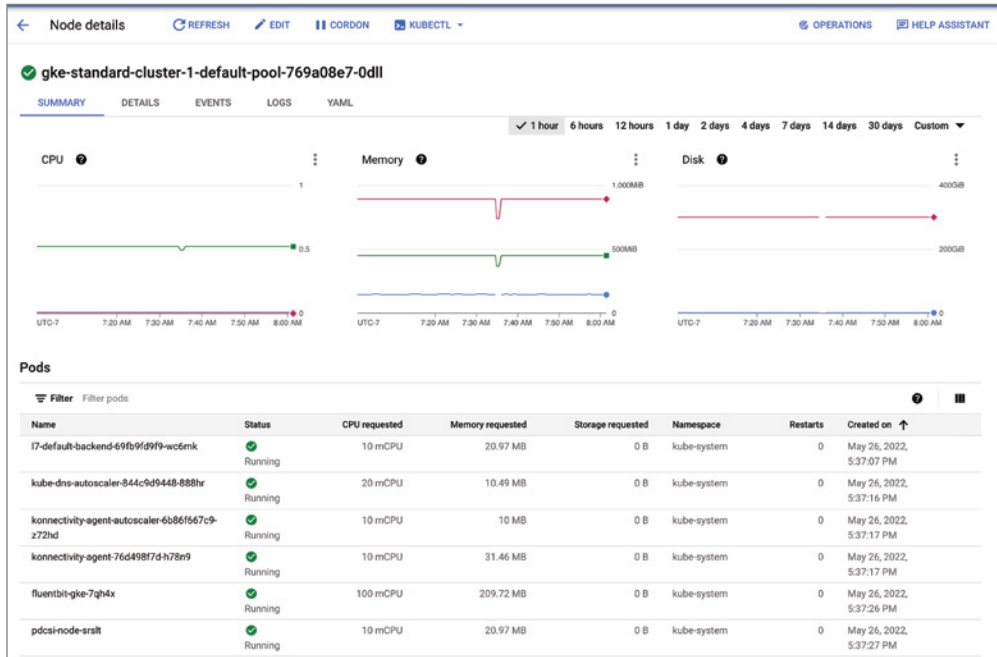
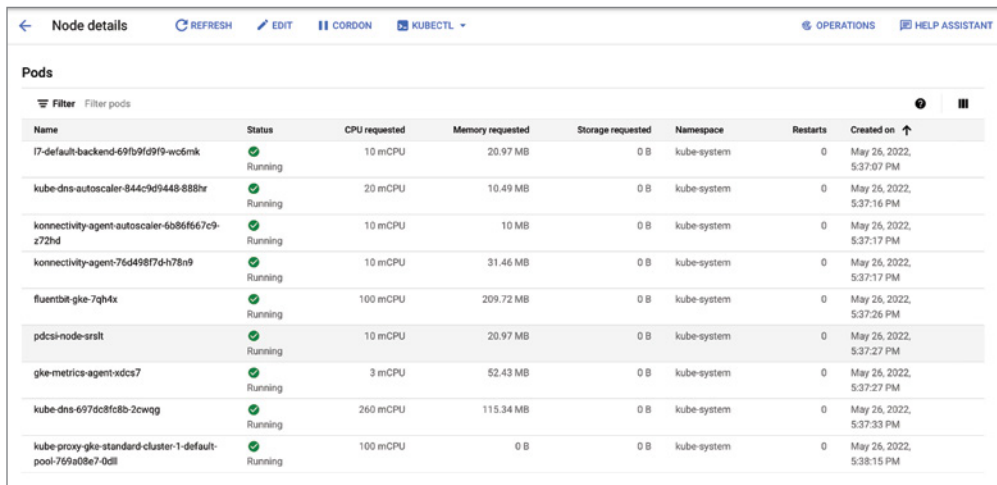
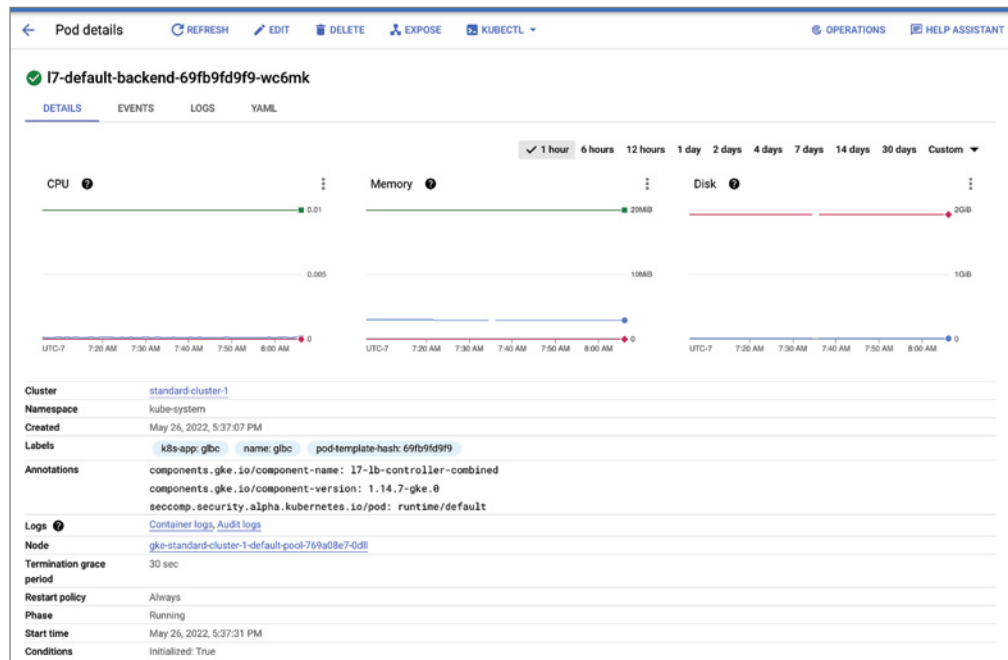
FIGURE 8.11 Example details of a node running in a Kubernetes cluster**FIGURE 8.12** Pod status display, with the status Running

FIGURE 8.13 Details of a pod running on a node



Using Cloud Console, you can list all clusters and view details of their configuration and status. You can then drill down into each node, pod, and container to view their details.

Viewing the Status of Kubernetes Clusters Using Cloud SDK and Cloud Shell

You can also use the command line to view the status of a cluster. The `gcloud container clusters list` command is used to show those details.

To list the names and basic information of all clusters, use this command:

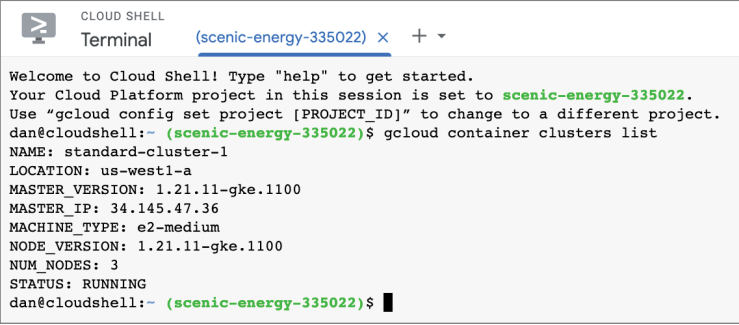
```
gcloud container clusters list
```

This produces the output shown in Figure 8.14.

Why Don't Commands Start with *gcloud kubernetes*?

`gcloud` commands start with the word `gcloud` followed by the name of the service, for example, `gcloud compute` for Compute Engine commands and `gcloud sql` for Cloud SQL commands. You might expect the Kubernetes Engine commands to start with `gcloud kubernetes`, but the service was originally called Google Container Engine. In November 2017, Google renamed the service Kubernetes Engine, but the `gcloud` commands remained the same.

FIGURE 8.14 Example output from the `gcloud container clusters list` command



```

Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to scenic-energy-335022.
Use "gcloud config set project [PROJECT_ID]" to change to a different project.
dan@cloudshell:~ (scenic-energy-335022)$ gcloud container clusters list
NAME: standard-cluster-1
LOCATION: us-west1-a
MASTER_VERSION: 1.21.11-gke.1100
MASTER_IP: 34.145.47.36
MACHINE_TYPE: e2-medium
NODE_VERSION: 1.21.11-gke.1100
NUM_NODES: 3
STATUS: RUNNING
dan@cloudshell:~ (scenic-energy-335022)$

```

To view the details of a cluster, use the `gcloud container clusters describe` command. You will need to pass in the name of a zone or region using the `--zone` or `--region` parameter. For example, to describe a cluster named `standard-cluster-1` located in the `us-central1-a` zone, you would use this command:

```
gcloud container clusters describe --zone us-central1-a standard-cluster-1
```

This command will display details like those shown in Figure 8.15 and Figure 8.16. Note that the `describe` command also displays authentication information such as client certificate, username, and password. That information is not shown in the figures.

FIGURE 8.15 Part 1 of the information displayed by the `gcloud container clusters describe` command

```

dan@cloudshell:~ (scenic-energy-335022)$ gcloud container clusters describe --zone us-west1-a standard-cluster-1
addonsConfig:
  dnsCacheConfig: {}
  gcePersistentDiskCsiDriverConfig:
    enabled: true
  horizontalPodAutoscaling: {}
  httpLoadBalancing: {}
  kubernetesDashboard:
    disabled: true
  networkPolicyConfig:
    disabled: true
  authenticatorGroupsConfig: {}
  autoscaling:
    autoscalingProfile: BALANCED
  binaryAuthorization: {}
  clusterIps: {}
  clusterIpv4Cidr: 10.36.0.0/14
  createTime: '2022-05-27T00:34:03+00:00'
  currentMasterVersion: 1.21.11-gke.1100
  currentNodeCount: 3
  currentNodeVersion: 1.21.11-gke.1100
  databaseEncryption:
    state: DECRYPTED
  defaultMaxPodsConstraint:
    maxPodsPerNode: '110'
  endpoint: 34.145.47.36
  id: d3f4a50281b04f22baf0a23b876d2809d7a1729135db43afbc13275b8a5d9ba9
  initialClusterVersion: 1.21.11-gke.1100
  instanceGroupUrls:
  - https://www.googleapis.com/compute/v1/projects/scenic-energy-335022/zones/us-west1-a/instanceGroupManagers/gke-standard-cluster-1-default-pool-769a08e7-grp
  ipAllocationPolicy:
    clusterIpv4Cidr: 10.36.0.0/14
    clusterIpv4CidrBlock: 10.36.0.0/14
    clusterSecondaryRangeName: gke-standard-cluster-1-pods-d3f4a502
    servicesIpv4Cidr: 10.40.0.0/20
    servicesIpv4CidrBlock: 10.40.0.0/20
    servicesSecondaryRangeName: gke-standard-cluster-1-services-d3f4a502
    useIpAliases: true
  labelFingerprint: a9dcl6a7
  legacyAbac: {}
  location: us-west1-a
  locations:
  - us-west1-a
  loggingConfig:
    componentConfig:
      enableComponents:
      - SYSTEM_COMPONENTS
      - WORKLOADS
    loggingService: logging.googleapis.com/kubernetes

```

FIGURE 8.16 Part 2 of the information displayed by the `gcloud container clusters describe` command

```

masterAuthorizedNetworksConfig: {}
monitoringConfig:
  componentConfig:
    enableComponents:
      - SYSTEM_COMPONENTS
monitoringService: monitoring.googleapis.com/kubernetes
name: standard-cluster-1
network: default
networkConfig:
  datapathProvider: LEGACY_DATAPATH
  defaultSnatStatus: {}
  network: projects/scenic-energy-335022/global/networks/default
  serviceExternalIpsConfig: {}
  subnetwork: projects/scenic-energy-335022/regions/us-west1/subnetworks/default
nodeConfig:
  diskSizeGb: 100
  diskType: pd-standard
  imageType: COS_CONTAINERD
  machineType: e2-medium
  metadata:
    disable-legacy-endpoints: 'true'
  oauthScopes:
    - https://www.googleapis.com/auth/devstorage.read_only
    - https://www.googleapis.com/auth/logging.write
    - https://www.googleapis.com/auth/monitoring
    - https://www.googleapis.com/auth/servicecontrol
    - https://www.googleapis.com/auth/service.management.readonly
    - https://www.googleapis.com/auth/trace.append
  serviceAccount: default
  shieldedInstanceConfig:
    enableIntegrityMonitoring: true
nodePoolAutoConfig: {}
nodePoolDefaults:
  nodeConfigDefaults: {}
nodePools:
- autoscaling: {}
  config:
    diskSizeGb: 100
    diskType: pd-standard
    imageType: COS_CONTAINERD
    machineType: e2-medium
    metadata:
      disable-legacy-endpoints: 'true'
    oauthScopes:
      - https://www.googleapis.com/auth/devstorage.read_only
      - https://www.googleapis.com/auth/logging.write
      - https://www.googleapis.com/auth/monitoring
      - https://www.googleapis.com/auth/servicecontrol
      - https://www.googleapis.com/auth/service.management.readonly

```

To list information about nodes and pods, use the `kubectl` command.

First, you need to ensure you have a properly configured `kubeconfig` file, which contains information on how to communicate with the cluster API. Run the command `gcloud container clusters get-credentials` with the name of a zone or region and the name of a cluster. Here's an example:

```
gcloud container clusters get-credentials \
--zone us-central1-a standard-cluster-1
```

This command will configure the `kubeconfig` file on a cluster named `standard-cluster-1` in the `us-central1-a` zone. Figure 8.17 shows an example output of that command, which includes the status of fetching and setting authentication data.

FIGURE 8.17 Example output of the `get-credentials` command

```
dan@cloudshell:~ (scenic-energy-335022)$ gcloud container clusters get-credentials --zone us-west1-a standard-cluster-1
Fetching cluster endpoint and auth data.
kubeconfig entry generated for standard-cluster-1.
dan@cloudshell:~ (scenic-energy-335022)$
```

You can list the nodes in a cluster using the following:

```
kubectl get nodes
```

This command produces output like that shown in Figure 8.18, which shows the status of three nodes.

FIGURE 8.18 Example output of the `kubectl get nodes` command

```
dan@cloudshell:~ (scenic-energy-335022)$ kubectl get nodes
W0527 15:18:44.254844 1253 gcp.go:120] WARNING: the gcp auth plugin is deprecated in v1.22+, unavailable in v1.25+; use gcloud instead.
To learn more, consult https://cloud.google.com/blog/products/containers-kubernetes/kubectl-auth-changes-in-gke
NAME                                STATUS    ROLES    AGE   VERSION
gke-standard-cluster-1-default-pool-769a08e7-0d11  Ready    <none>   14h   v1.21.11-gke.1100
gke-standard-cluster-1-default-pool-769a08e7-bz3h  Ready    <none>   14h   v1.21.11-gke.1100
gke-standard-cluster-1-default-pool-769a08e7-v63z  Ready    <none>   14h   v1.21.11-gke.1100
dan@cloudshell:~ (scenic-energy-335022)$
```

Similarly, to list pods, use the following command:

```
kubectl get pods
```

This command produces output like that shown in Figure 8.19, which lists pods and their status.

For more details about nodes and pods, use these commands:

```
kubectl describe nodes
kubectl describe pods
```

FIGURE 8.19 Example output of the `kubectl get pods` command

```

dan@cloudshell:~ (scenic-energy-335022) $ kubectl get pods -n kube-system
NAME                                READY    STATUS    RESTARTS   AGE
event-exporter-gke-7549f5d5c8-ng2cd 2/2      Running   0           15h
fluentbit-gke-7hd6g                 2/2      Running   0           15h
fluentbit-gke-7qh4x                 2/2      Running   0           15h
fluentbit-gke-hlgnp                 2/2      Running   0           15h
gke-metrics-agent-8g5zv             1/1      Running   0           15h
gke-metrics-agent-bwsj2             1/1      Running   0           15h
gke-metrics-agent-xdcs7             1/1      Running   0           15h
konnectivity-agent-76d498f7d-dblgc 1/1      Running   0           15h
konnectivity-agent-76d498f7d-h78n9 1/1      Running   0           15h
konnectivity-agent-76d498f7d-tq9w5 1/1      Running   0           15h
konnectivity-agent-autoscaler-6b86f667c9-z72hd 1/1      Running   0           15h
kube-dns-697dc8fc8b-2cwgq           4/4      Running   0           15h
kube-dns-697dc8fc8b-46vcc           4/4      Running   0           15h
kube-dns-autoscaler-844c9d9448-888hr 1/1      Running   0           15h
kube-proxy-gke-standard-cluster-1-default-pool-769a08e7-0dl1 1/1      Running   0           15h
kube-proxy-gke-standard-cluster-1-default-pool-769a08e7-bz3h 1/1      Running   0           15h
kube-proxy-gke-standard-cluster-1-default-pool-769a08e7-v63z 1/1      Running   0           15h
l7-default-backend-69f9fd9f9-8f6mk 1/1      Running   0           15h
metrics-server-v0.4.5-bbb794dccc-wc7k6g 2/2      Running   0           15h
pdcsl-node-p2sg8                    2/2      Running   0           15h
pdcsl-node-r4u5s                    2/2      Running   0           15h
pdcsl-node-srslt                    2/2      Running   0           15h
dan@cloudshell:~ (scenic-energy-335022) $

```

Figure 8.20 and Figure 8.21 show partial listings of the results. Note that the `kubectl describe pods` command also includes information about containers, names, labels, conditions, network addresses, and system information.

FIGURE 8.20 Partial listing of the details shown by the `kubectl describe nodes` command

```

Name: gke-standard-cluster-1-default-pool-769a08e7-0dl1
Roles: <none>
Labels: beta.kubernetes.io/arch=amd64
        beta.kubernetes.io/instance-type=e2-medium
        beta.kubernetes.io/os=linux
        cloud.google.com/gke-boot-disk=pd-standard
        cloud.google.com/gke-container-runtime=containerd
        cloud.google.com/gke-nodepool=default-pool
        cloud.google.com/gke-os-distribution=cos
        cloud.google.com/machine-family=e2
        failure-domain.beta.kubernetes.io/region=us-west1
        failure-domain.beta.kubernetes.io/zone=us-west1-a
        kubernetes.io/arch=amd64
        kubernetes.io/hostname=gke-standard-cluster-1-default-pool-769a08e7-0dl1
        kubernetes.io/os=linux
        node.kubernetes.io/instance-type=e2-medium
        topology.gke.io/zone=us-west1-a
        topology.kubernetes.io/region=us-west1
        topology.kubernetes.io/zone=us-west1-a
Annotations: container.googleapis.com/instance_id: 350793183704489540
        csi.volume.kubernetes.io/nodeid:
        {
          "pd.csi.storage.gke.io": "projects/scenic-energy-335022/zones/us-west1-a/instances/gke-standard-cluster-1-default-pool-769a08e7-0dl1"
        }
        node.alpha.kubernetes.io/ttl: 0
        node.gke.io/last-applied-node-labels:
        cloud.google.com/gke-boot-disk=pd-standard,cloud.google.com/gke-container-runtime=containerd,cloud.google.com/gke-nodepool=default-pool,c...
        node.gke.io/last-applied-node-taints:
        volumes.kubernetes.io/controller-managed-attach-detach: true
CreationTimestamp: Fri, 27 May 2022 00:37:25 +0000
Taints: <none>
Unschedulable: false
Lease:
HolderIdentity: gke-standard-cluster-1-default-pool-769a08e7-0dl1
AcquireTime: <unset>
RenewTime: Fri, 27 May 2022 15:41:29 +0000
Conditions:
  Type            Status    LastHeartbeatTime    LastTransitionTime    Reason    Message
  ----            -
  CorruptDockerOverlay2  False    Fri, 27 May 2022 15:39:04 +0000    Fri, 27 May 2022 00:37:27 +0000    NoCorruptDockerOverlay2    docker overlay2 is functioning properly
  FrequentUnregisterNetDevice  False    Fri, 27 May 2022 15:39:04 +0000    Fri, 27 May 2022 00:37:27 +0000    NoFrequentUnregisterNetDevice    node is functioning properly
  FrequentKubeletRestart  False    Fri, 27 May 2022 15:39:04 +0000    Fri, 27 May 2022 00:37:27 +0000    NoFrequentKubeletRestart    kubelet is functioning properly
  FrequentDockerRestart  False    Fri, 27 May 2022 15:39:04 +0000    Fri, 27 May 2022 00:37:27 +0000    NoFrequentDockerRestart    docker is functioning properly
  FrequentContainerdRestart  False    Fri, 27 May 2022 15:39:04 +0000    Fri, 27 May 2022 00:37:27 +0000    NoFrequentContainerdRestart    containerd is functioning properly
  KernelDeadlock  False    Fri, 27 May 2022 15:39:04 +0000    Fri, 27 May 2022 00:37:27 +0000    KernelHasNoDeadlock    kernel has no deadlock
  ReadOnlyFilesystem  False    Fri, 27 May 2022 15:39:04 +0000    Fri, 27 May 2022 00:37:27 +0000    FilesystemIsNotReadOnly    Filesystem is not read-only
  NetworkUnavailable  False    Fri, 27 May 2022 00:37:26 +0000    Fri, 27 May 2022 00:37:26 +0000    RouteCreated    kubelet has sufficient memory available
  MemoryPressure  False    Fri, 27 May 2022 15:37:25 +0000    Fri, 27 May 2022 00:37:10 +0000    KubeletHasSufficientMemory    kubelet has sufficient memory available
  DiskPressure  False    Fri, 27 May 2022 15:37:25 +0000    Fri, 27 May 2022 00:37:10 +0000    KubeletHasNoDiskPressure    kubelet has no disk pressure

```

FIGURE 8.21 Partial listing of the details shown by the `kubectl describe pods` command

```

Name:          event-exporter-gke-5479fd58c-ng2cd
Namespace:     kube-system
Priority:       0
Node:          gke-standard-cluster-1-default-pool-769a08e7-bz3h/10.138.0.2
Start Time:    Fri, 27 May 2022 00:37:35 +0000
Labels:        k8s-app=event-exporter
               pod-template-hash=5479fd58c8
               version=v0.3.5
Annotations:   components.gke.io/component-name: event-exporter
               components.gke.io/component-version: 1.0.10
Status:        Running
IP:            10.36.0.3
IPs:           IP: 10.36.0.3
Controlled By: ReplicaSet/event-exporter-gke-5479fd58c8
Containers:
  event-exporter:
    Container ID:  containerd://19fb483de3f0ecdb8dbda3b89434599940c8f0dca35224e42f92da2539b85fe8
    Image:         gke.gcr.io/event-exporter:v0.3.5-gke.0
    Image ID:      gke.gcr.io/event-exporter@sha256:c9e908d7ea0020f47cc1279eaf6ce1b4dc0debb9c9493b8550aaecd3c82c7e
    Port:         <none>
    Host Port:     <none>
    Command:
      /event-exporter
      --sink-opts=--stackdriver-resource-model=new --endpoint=https://logging.googleapis.com
      --prometheus-endpoint=:8080
    State:         Running
      Started:     Fri, 27 May 2022 00:37:49 +0000
      Ready:       True
      Restart Count: 0
    Environment:  <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-lfv1t (ro)
  prometheus-to-sd-exporter:
    Container ID:  containerd://e6a0d73252606f2f0c9e0edc5369418d61f7c969a0c9cb3ede709913b736ffcf1
    Image:         gke.gcr.io/prometheus-to-sd:v0.10.0-gke.0
    Image ID:      gke.gcr.io/prometheus-to-sd@sha256:c5e12480a431990d5e39ed249dc43a7672e99f7ef94a9928be40c12f418f62f
    Port:         <none>
    Host Port:     <none>
    Command:
      /monitor
      --stackdriver-prefix=container.googleapis.com/internal/addons
      --api-override=https://monitoring.googleapis.com/
      --source=event_exporter:http://localhost:8080?whitelisted=stackdriver_sink_received_entry_count,stackdriver_sink_request_count,stackdriver_sink_successfully_sent_entry_count
      --pod-id=$(POD_NAME)
      --namespace-id=$(POD_NAMESPACE)
      --node-name=$(NODE_NAME)
    State:         Running

```

To view the status of clusters from the command line, use the `gcloud container` commands, but to get information about Kubernetes managed objects, like nodes, pods, and containers, use the `kubectl` command.

Adding, Modifying, and Removing Nodes

You can add, modify, and remove nodes from a cluster using either Cloud Console or Cloud SDK in your local environment, on a Google Cloud virtual machine, or in Cloud Shell.

Adding, Modifying, and Removing Nodes with Cloud Console

From Cloud Console, navigate to the Kubernetes Engine page and display a list of clusters. Click the name of a cluster to display its details, as shown in Figure 8.22.

FIGURE 8.22 Details of a cluster in Cloud Console

← Clusters

EDITDELETEADD NODE POOL

OPERATIONSHELP ASSISTANT

✓ standard-cluster-1

DETAILSNODESSTORAGELOGS

Cluster basics

Name	standard-cluster-1	🔒
Location type	Zonal	🔒
Control plane zone	us-central1-c	🔒
Default node zones ?	us-central1-c	✎
Release channel	Regular channel	✎ UPGRADE AVAILABLE
Version	1.22.8-gke.201	
Total size	3	①
Endpoint	34.133.226.46 Show cluster certificate	🔒

Automation

Maintenance window	Any time	✎
Maintenance exclusions	None	✎
Notifications	Disabled	✎
Vertical Pod Autoscaling	Disabled	✎
Node auto-provisioning	Disabled	✎
Autoscaling profile	Balanced	✎

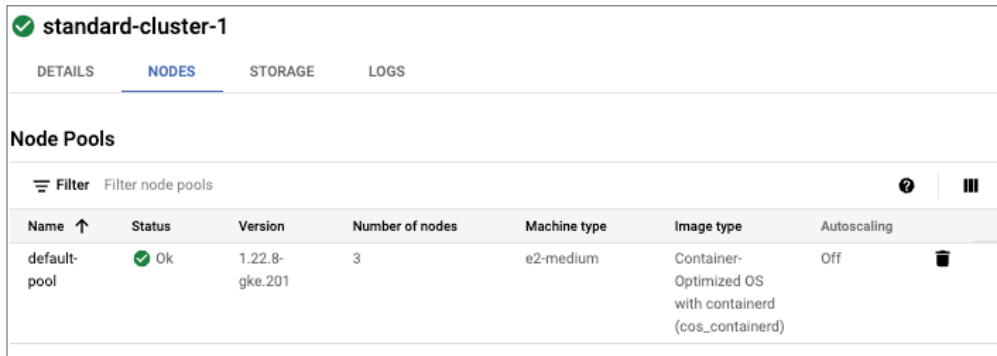
Networking

Private cluster	Disabled	🔒
Network	default	🔒
Subnet	default	🔒
VPC-native traffic routing	Enabled	🔒
Cluster pod address range (default)	10.124.0.0/14	🔒 ⌵
Maximum pods per node	110	🔒
Service address range	10.0.0.0/20	🔒
Intranode visibility	Disabled	✎
NodeLocal DNSCache	Disabled	✎
HTTP Load Balancing	Enabled	✎

Select the Nodes tab to display the Node Pools and Nodes sections. The Node Pools section lists the name, status, GKE version, number of nodes, machine type, and image type.

It also indicates whether Autoscaling is enabled on the node pool. Select the Edit option to change the number of nodes in the node pool. Figure 8.23 shows details of a node pool.

FIGURE 8.23 Details of a node pool in Cloud Console



standard-cluster-1						
DETAILS NODES STORAGE LOGS						
Node Pools						
Filter Filter node pools						
Name ↑	Status	Version	Number of nodes	Machine type	Image type	Autoscaling
default-pool	Ok	1.22.8-gke.201	3	e2-medium	Container-Optimized OS with containerd (cos_containerd)	Off

To add nodes, increase the size to the number of nodes you would like. To remove nodes, decrease the size to the number of nodes you'd like to have.

Adding, Modifying, and Removing Nodes with Cloud SDK and Cloud Shell

The command to add or modify nodes is `gcloud container clusters resize`. The command takes three parameters:

- cluster name
- node pool name
- cluster size

For example, assume you have a cluster named `standard-cluster-1` running a node pool called `default-pool`. To increase the size of the node pool from 3 to 5, use this command:

```
gcloud container clusters resize standard-cluster-1 \
--node-pool default-pool --num-nodes 5 --region=us-central1
```

The number of nodes you specify in the command will be the number of nodes in the pool if you are using a zonal cluster. If you are using a regional cluster, the number of nodes will be the number of nodes for each zone the node pool is in.

Once a cluster has been created, you can modify it using the `gcloud container clusters update` command. For example, to enable Autoscaling, use the update

command to specify the maximum and minimum number of nodes. The command to update a cluster named `standard-cluster-1` running in a node pool called `default-pool` is as follows:

```
gcloud container clusters update standard-cluster-1 \
--enable-autoscaling --min-nodes 1 \
--max-nodes 5 --zone us-central1-a \
--node-pool default-pool
```



Real World Scenario

Keeping Up with Demand with Autoscaling

Often it is difficult to predict demand on a service. Even if there are regular patterns, such as large batch jobs run during nonbusiness hours, there can be variation in when those peak loads run. Rather than keep manually changing the number of vCPUs in a cluster, enable Autoscaling to automatically add or remove nodes as needed based on demand. Autoscaling can be enabled when creating clusters with either Cloud Console or `gcloud`. This approach is more resilient to unexpected spikes and shifts in long-term patterns of peak use. It will also help optimize the cost of your cluster by not running too many servers when not needed. It will also help maintain performance by having sufficient nodes to meet demand.

Adding, Modifying, and Removing Pods

You can add, modify, and remove pods from a cluster using either Cloud Console or Cloud SDK in your local environment, on a Google Cloud VM, or in Cloud Shell.

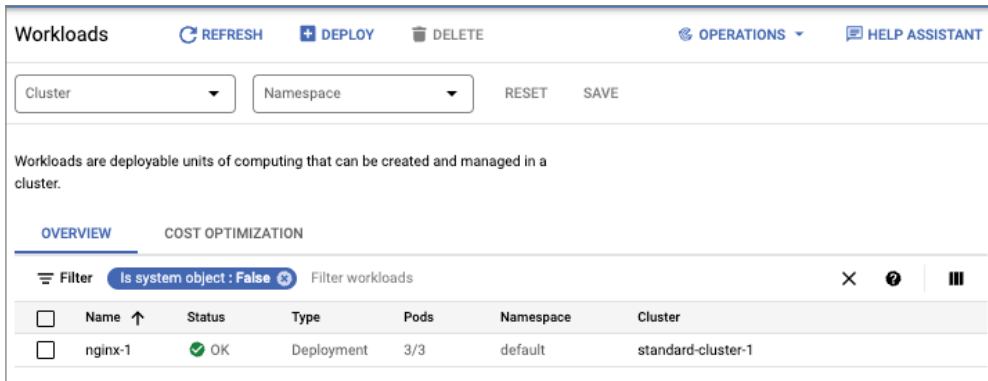
It is considered a best practice to not manipulate pods directly. Kubernetes will maintain the number of pods specified for a deployment. If you would like to change the number of pods, you should change the deployment configuration.

Adding, Modifying, and Removing Pods with Cloud Console

Pods are managed through deployments. A deployment includes a configuration parameter called *replicas*, which are the number of pods running the application specified in the deployment. This section describes how to use Cloud Console to change the number of replicas, which will in turn change the number of pods.

From Cloud Console, select the Workloads options from the navigation menu on the left. This displays a list of deployments, as shown in Figure 8.24.

FIGURE 8.24 Deployment list of a cluster



Click the name of the deployment you want to modify; a form is displayed with details (see Figure 8.25).

Click the name of a pod in the Managed Pods section (see Figure 8.26) to display details of the pod. Note there is a button that allows you to delete the pod in the horizontal menu bar at the top of the page. Again, this is not a best practice in general and pods should be managed by Kubernetes.

Select the Actions option from the three vertical dot icon to display Actions, then select Scale to display a dialog box that allows you to set a new size for the workload, as shown in Figure 8.27. In this example, the number of replicas has been changed to 2.

You can also have Kubernetes automatically add and remove replicas (and pods) depending on need by specifying Autoscaling. You can choose Autoscale from the Actions menu, which is shown in Figure 8.28. In the resulting form, you can specify a minimum and maximum number of replicas to run.

The Actions menu also provides options for exposing a service on a port, as shown in Figure 8.29, and to specify parameters to control rolling updates to deployed code, as shown in Figure 8.30. The parameters include the minimum seconds to wait before considering the pod updated, the maximum number of pods above target size allowed, and the maximum number of unavailable pods.

FIGURE 8.25 Multiple forms contain details of a deployment and include a menu of actions you can perform on the deployment.

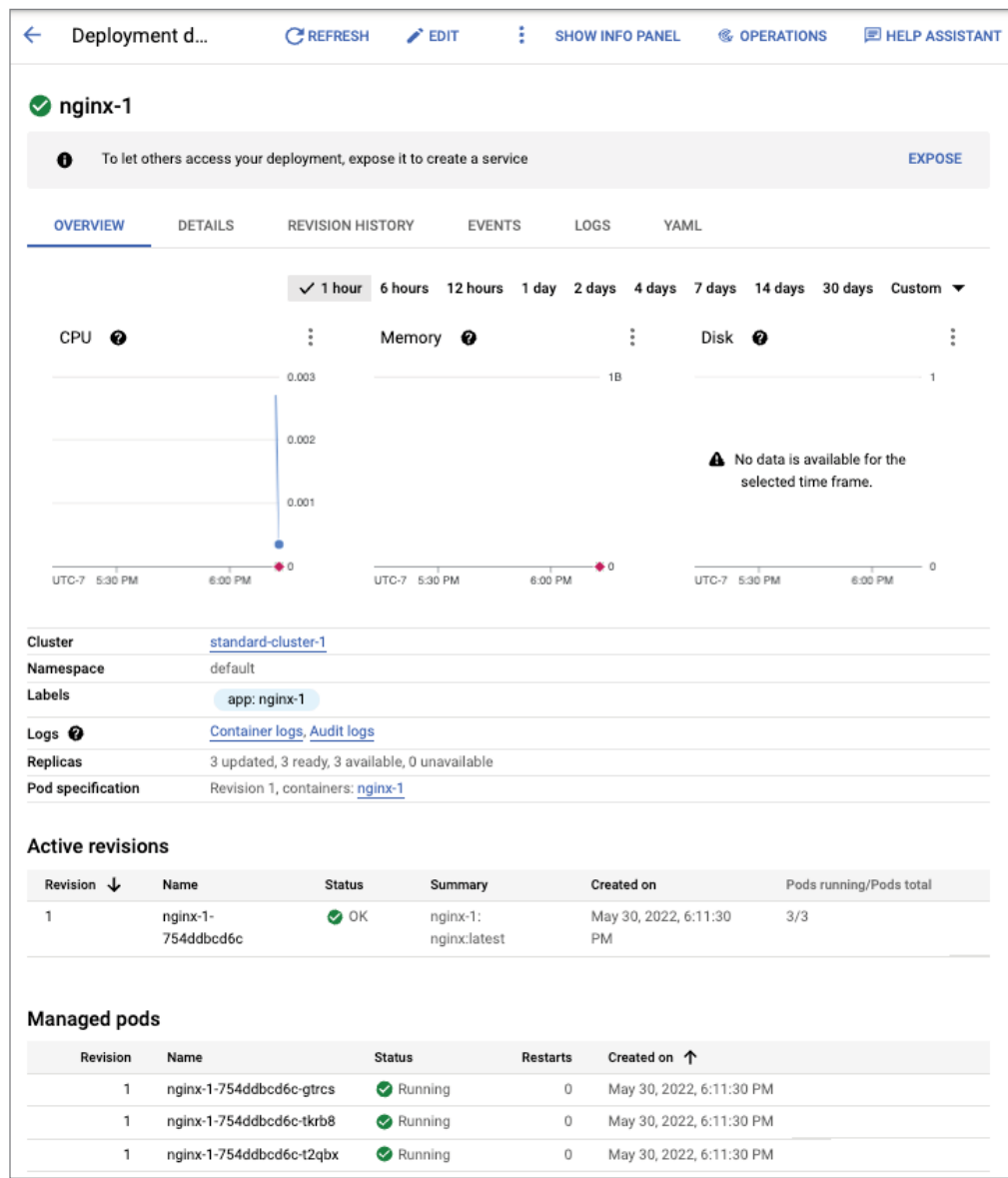


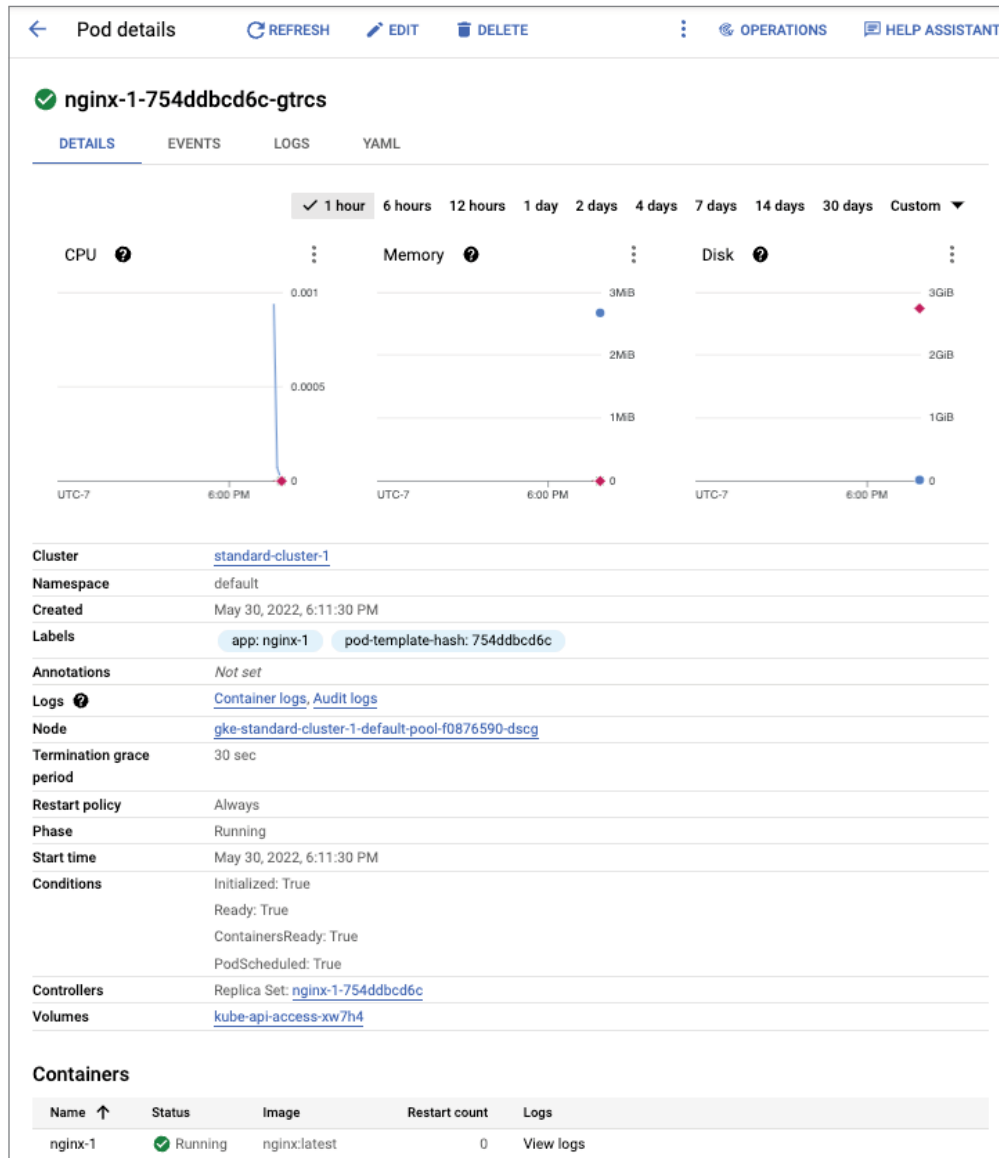
FIGURE 8.26 Details of a pod running in GKE

FIGURE 8.27 Set the number of replicas for a deployment.

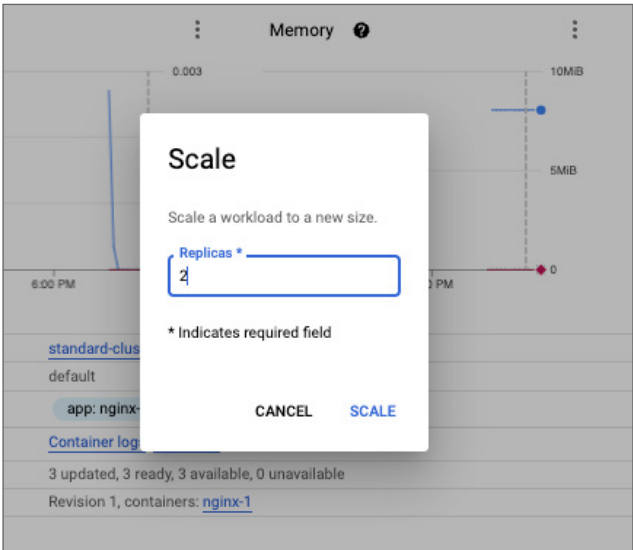
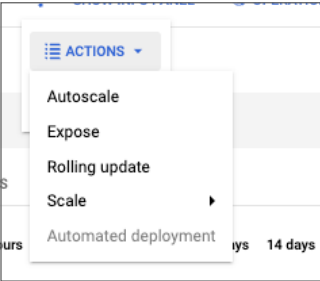


FIGURE 8.28 Enable Autoscaling to automatically add and remove replicas as needed depending on load.



Adding, Modifying, and Removing Pods with Cloud SDK and Cloud Shell

Working with pods in Cloud SDK and Cloud Shell is done by working with deployments; deployments were explained earlier in the section “Adding, Modifying, and Removing Pods with Cloud Console.” You can use the `kubectl` command to work with deployments.

FIGURE 8.29 Form to expose services running on pods

The screenshot shows the Kubernetes dashboard interface for a deployment named 'nginx-1'. A modal window titled 'Expose' is open, allowing the user to expose the deployment's pods using a Kubernetes Service. The modal includes a 'Port mapping' section with fields for 'Port 1' (set to 80), 'Target port 1', and 'Protocol 1' (set to TCP). There is an '+ ADD PORT MAPPING' button below these fields. The 'Service type' is set to 'Cluster IP'. A note indicates that an asterisk (*) denotes a required field. At the bottom of the modal are 'CANCEL' and 'EXPOSE' buttons. The background shows the 'Overview' tab of the deployment, with a table of active revisions at the bottom.

Revision	Name	Status	Summary	Created on	Pods running/Pods total
1	nginx-1-754ddbcd6c	OK	nginx-1: nginx:latest	May 30, 2022, 6:11:30 PM	3/3

To list deployments, use the following command:

```
kubectl get deployments
```

To add and remove pods, change the configuration of deployments using the `kubectl scale deployment` command. For this command, you have to specify the deployment name and number of replicas. For example, to set the number of replicas to 5 for a deployment named `nginx-1`, use this:

```
kubectl scale deployment nginx-1 --replicas 5
```

FIGURE 8.30 Form to specify parameters for rolling updates of code running in pods

The screenshot shows the Kubernetes dashboard for a deployment named 'nginx-1'. A modal window titled 'Rolling update' is open, allowing configuration of update parameters. The background shows the 'OVERVIEW' tab with various metrics like CPU, Disk, and a table of revisions.

Rolling update

Update workload Pods to a new application version.

Minimum seconds ready: 0

Maximum surge: 25%

Maximum unavailable: 25%

Container images

Image of nginx-1 *: nginx:latest

* Indicates required field

CANCEL UPDATE

Revision ↓	Name	Status	Summary	Created on	Pods running/Pods tot
1	nginx-1-754ddbcd6c	OK	nginx-1: nginx:latest	May 30, 2022, 6:11:30 PM	3/3

Managed pods

To have Kubernetes manage the number of pods based on load, use the `autoscale` command. The following command will add or remove pods as needed to meet demand based on CPU utilization. If CPU usage exceeds 80 percent, up to 10 additional pods or replicas will be added. The deployment will always have at least one pod or replica.

```
kubectl autoscale deployment nginx-1 --max 10 --min 1 --cpu-percent 80
```

To remove a deployment, use the `delete deployment` command like so:

```
kubectl delete deployment nginx-1
```



Services vs services

In the next section we discuss a Kubernetes abstraction known as Services. A Kubernetes Service is a way to expose an application on a set of pods to other applications and users on a network. The term services is also used in a more generic sense as a synonym for an application. For example, an application that provides an API that returns information about weather might be called a “weather service” and an application that computes tax on a sale might be called a “tax service.” To minimize potential confusion, in the following section we use the term “Service” with a capital S to refer to the Kubernetes abstraction and “service” with a lower-case s to refer to the synonym for applications.

Adding, Modifying, and Removing Services

You can add, modify, and remove Services from a cluster using either Cloud Console or Cloud SDK in your local environment, on a Google Cloud VM, or in Cloud Shell.

A service is an abstraction that groups a set of pods as a single resource.

Adding, Modifying, and Removing Services with Cloud Console

Kubernetes Services are added through deployments. In Cloud Console, select the Workloads option from the navigation menu to display a deployment list, as shown in Figure 8.31. Note the Deploy option in the horizontal menu at the top of the page.

FIGURE 8.31 Deployment list along with a Deploy command to create new services

Workloads

REFRESH

DEPLOY

DELETE

OPERATIONS

HELP ASSISTANT

Cluster

Namespace

RESET

SAVE

Workloads are deployable units of computing that can be created and managed in a cluster.

OVERVIEW

COST OPTIMIZATION

Filter

Is system object : False

Filter workloads

<input type="checkbox"/>	Name ↑	Status	Type	Pods	Namespace	Cluster
<input type="checkbox"/>	nginx-1	<div>OK</div>	Deployment	3/3	default	standard-cluster-1

Click Deploy to display the deployment form, shown in Figure 8.32.

FIGURE 8.32 Form that lets you specify a new deployment for a service

← Create a deployment

1 Container

Edit container

☒ Existing container image
☐ New container image

Image path *
nginx:latest [SELECT](#)

Enter your image path, or choose from Google Container Registry. You can also try to deploy with official nginx image nginx:latest.

Environment variables

[+ ADD ENVIRONMENT VARIABLE](#)

Initial command

Overrides the default endpoint of the container image.

[CANCEL](#) [DONE](#)

[ADD CONTAINER](#)

[CONTINUE](#)

2 Configuration

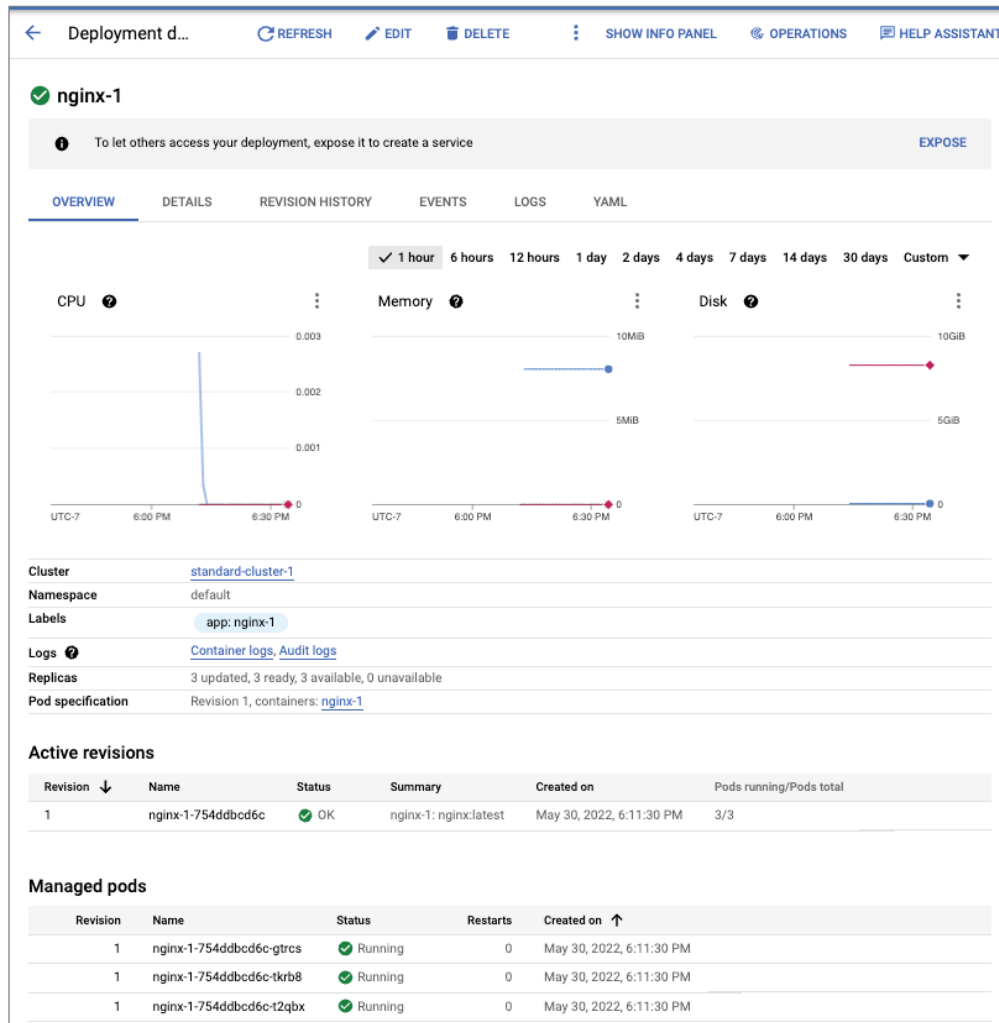
In the Container Image parameter, you can specify the name of an image or select one from the Google Container Repository. To specify a name directly, specify a path to the image using a URL such as this:

```
gcr.io/google-samples/hello-app:2.0
```

You can specify labels, the initial command to run, and a name for your application.

When you click the name of a deployment you will see details of that deployment, including a list of Services, like that shown in Figure 8.33.

Clicking the name of a Service opens the Detail form of the Service, which includes a Delete option in the horizontal menu. Figure 8.34 shows the delete dialog.

FIGURE 8.33 Details of Services exposing a deployment

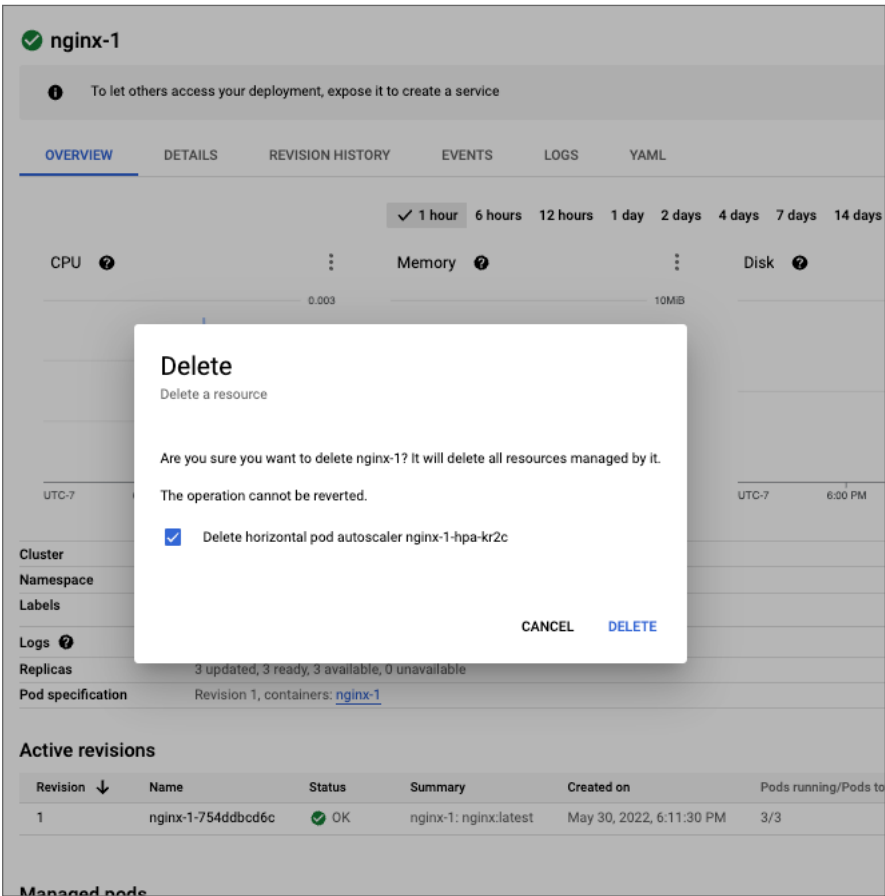
Adding, Modifying, and Removing Services with Cloud SDK and Cloud Shell

Use the `kubectl get services` command to list Services.

To add a Service, use the `kubectl create deployment` command to start a Service. For example, to add a Service called `hello-server` using the sample application by the same name provided by Google, use the following command:

```
kubectl create deployment hello-server --image=gcr.io/google/samples/
hello-app:1.0 \
--port 8080
```

FIGURE 8.34 Navigate to the Service Details page to delete a service using the Delete option in the horizontal menu.



This command will download and start running the image found at the path `gcr.io/google-samples/` called `hello-app`, version 1. It will be accessible on port 8080. Deployments need to be exposed to be accessible to resources outside the cluster. This can be set using the `expose` command, as shown here:

```
kubectl expose deployment hello-server --type="LoadBalancer"
```

This command exposes the Service by having a load balancer act as the endpoint for outside resources to contact the service.

To remove a Service, use the `delete service` command, as shown here:

```
kubectl delete service hello-server
```

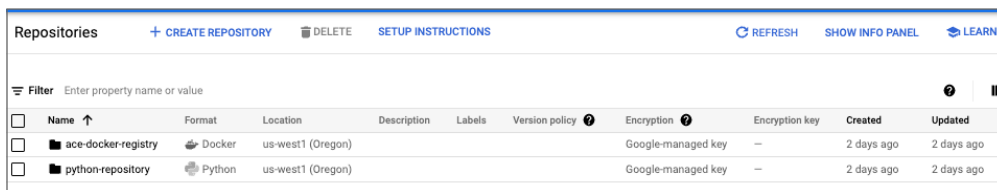
Creating Repositories in the Artifact Registry

Artifact Registry is a Google Cloud service for storing container images. Container Registry is a service used in the past to manage images, but Artifact Registry is now the recommended service for managing images. Once you have created a registry and pushed images to it, you can view the contents of the registry and image details using Cloud Console and Cloud SDK and Cloud Shell.

Viewing the Image Repository and Image Details with Cloud Console

In Cloud Console, select Artifact Registry from the navigation menu to display example registries (see Figure 8.35).

FIGURE 8.35 A listing of repositories in Artifact Registry



	Name	Format	Location	Description	Labels	Version policy	Encryption	Encryption key	Created	Updated
<input type="checkbox"/>	ace-docker-registry	Docker	us-west1 (Oregon)				Google-managed key	—	2 days ago	2 days ago
<input type="checkbox"/>	python-repository	Python	us-west1 (Oregon)				Google-managed key	—	2 days ago	2 days ago

To create a registry, click the + icon to display a dialog box like the one shown in Figure 8.36. You can see, Artifact Registry can have multiple types of registries, including one for Docker, Maven (a Java framework), and Python, among others.

Depending on the type of repository you create, you must take additional steps to set up the repository. Artifact Registry provides detailed commands for configuring the repository. Figure 8.37, for example, shows a command for configuring a Docker repository.

FIGURE 8.36 Creating a repository in Artifact Registry

The screenshot shows the 'Create repository' form in the Google Cloud Artifact Registry console. The form includes the following sections:

- Name:** A text input field with a red asterisk indicating it is required.
- Format:** A group of radio buttons for selecting the package format: Docker (selected), Maven, npm, Python, Apt, Yum, and Kubeflow Pipelines (with a 'PREVIEW' label).
- Location type:** A group of radio buttons for selecting the location type: Region (selected) and Multi-region.
- Region:** A dropdown menu with a red asterisk indicating it is required.
- Description:** A text area for providing a description of the repository.
- Labels:** A section with a '+ ADD LABEL' button.
- Encryption:** A group of radio buttons for selecting the encryption method: Google-managed encryption key (selected, with the note 'No configuration required') and Customer-managed encryption key (CMEK) (with the note 'Manage via Google Cloud Key Management Service').
- Buttons:** 'CREATE' and 'CANCEL' buttons at the bottom.

FIGURE 8.37 Example instructions for configuring a Docker repository

The screenshot shows the Google Cloud Artifact Registry console. The left pane displays a breadcrumb trail: 'My First Project' > 'Images for ace-docker-registry' > 'us-west1-docker.pkg.dev' > 'scenic-energy-335022' > 'ace-docker-registry'. Below the breadcrumb trail is a table with columns 'Name', 'Created', and 'Updated', but it shows 'No rows to display'. The right pane, titled 'Setup instructions', contains the following content:

Follow the steps below to configure your client to push and pull packages using this repository. You can also view more detailed instructions [here](#). For more information about working with artifacts in this repository, see the [documentation](#).

Initialize gcloud

The [Google Cloud SDK](#) is used to generate an access token when authenticating with Artifact Registry. Make sure that it is installed and initialized with [Application Default Credentials](#) before proceeding.

Configure Docker

Run the following command to configure `gcloud` as the credential helper for the Artifact Registry domain associated with this repository's location:

```
$ gcloud auth configure-docker \
  us-west1-docker.pkg.dev
```

Kubernetes Engine makes use of container images stored in a Docker Repository. The contents of the Docker Repository can be viewed in summary and in detail using both Cloud Console and the command-line Cloud SDK, including in Cloud Shell.

Summary

In this chapter, you learned how to perform basic management tasks for working with Kubernetes clusters, nodes, pods, and services. The chapter also described how to list the contents of container image repositories. You learned how to pin services in the Cloud Console menu, view the status of Kubernetes clusters, and view image repository and image details using `gcloud` commands. This chapter also described how to modify and remove nodes and pods. You also saw the benefits of autoscaling in a real-world scenario.

Both Cloud Console and Cloud SDK, including Cloud Shell, can be used to add, remove, and modify nodes, pods, and services. They both can be used to review the contents of an image repository. Some of the most useful commands include `gcloud container clusters create` and `gcloud container clusters resize`. The `kubectl` command is used to modify Kubernetes resources such as deployments and pods.

Exam Essentials

Know how to view the status of a Kubernetes cluster. Use Cloud Console to list clusters and drill down into clusters to see details of the cluster, including node, pod, and container details. Know the `gcloud container clusters` command and its options.

Understand how to add, modify, and remove nodes. Use Cloud Console to modify nodes and know how to add and remove nodes by changing deployments. Use the `gcloud container clusters resize` command to add and remove nodes.

Understand how to add, modify, and remove pods. Use Cloud Console to modify pods and to add and remove pods by changing deployments. Use `kubectl get deployments` to list deployments, `kubectl scale deployment` to modify the number of deployments, and `kubectl autoscale deployment` to enable Autoscaling.

Understand how to add, modify, and remove Services. Use Cloud Console to modify Services and add and remove Services by changing deployments. Use `kubectl create deployment` to start Services and `kubectl expose deployment` to make a Service accessible outside the cluster. Delete a service using the `kubectl delete service` command.

Review Questions

You can find the answers in the Appendix.

1. You are running several microservices in a Kubernetes cluster. You've noticed some performance degradation. After reviewing some logs, you begin to think the cluster may be improperly configured, and you open Cloud Console to investigate. How do you see the details of a specific cluster?
 - A. Type the cluster name into the search bar.
 - B. Click the cluster name.
 - C. Use the `gcloud cluster details` command.
 - D. None of the above.
2. You are viewing the details of a cluster in Cloud Console and want to see how many vCPUs are available in the cluster. Where would you look for that information?
 - A. Node Pools section of the Nodes Details page
 - B. Labels section of the Cluster Details page
 - C. Summary line of the Cluster Listing page
 - D. None of the above
3. You have been assigned to help diagnose performance problems with applications running on several Kubernetes clusters. The first thing you want to do is understand, at a high level, the characteristics of the clusters. Which command should you use?
 - A. `gcloud container list`
 - B. `gcloud container clusters list`
 - C. `gcloud clusters list`
 - D. None of the above
4. When you first try to use the `kubectl` command, you get an error message indicating that the resource cannot be found or you cannot connect to the cluster. What command would you use to try to eliminate the error?
 - A. `gcloud container clusters access`
 - B. `gcloud container clusters get-credentials`
 - C. `gcloud auth container`
 - D. `gcloud auth container clusters`
5. An engineer recently joined your team and is not aware of your team's standards for creating clusters and other Kubernetes objects. In particular, the engineer has not properly labeled several clusters. You want to modify the labels on the cluster from Cloud Console. How would you do it?
 - A. Click the Connect button.
 - B. Click the Deploy menu option.
 - C. Click the Edit menu option.
 - D. Type the new labels in the Labels section.

6. You receive a page in the middle of the night informing you that several services running on a Kubernetes cluster have high latency when responding to API requests. You review monitoring data and determine that there are not enough resources in the cluster to keep up with the load. You decide to add six more VMs to the cluster. What parameters will you need to specify when you issue the `cluster resize` command?
 - A. Cluster size
 - B. Cluster name
 - C. Node pool name
 - D. All of the above
7. You want to modify the number of pods in a cluster. What is the best way to do that?
 - A. Modify pods directly
 - B. Modify deployments
 - C. Modify node pools directly
 - D. Modify nodes
8. You want to see a list of deployments. Which option from the Kubernetes Engine navigation menu would you select?
 - A. Clusters
 - B. Storage
 - C. Workloads
 - D. Deployments
9. What actions are available from the Actions menu when viewing deployment details?
 - A. Scale and Autoscale only
 - B. Autoscale, Expose, and Rolling Update
 - C. Add, Modify, and Delete
 - D. None of the above
10. What is the command to list deployments from the command line?
 - A. `gcloud container clusters list-deployments`
 - B. `gcloud container clusters list`
 - C. `kubectl get deployments`
 - D. `kubectl deployments list`
11. What parameters of a deployment can be set in the Create Deployment page in Cloud Console?
 - A. Container image
 - B. Cluster name
 - C. Application name
 - D. All of the above

12. Where can you view a list of applications when using Cloud Console?
 - A. In the Deployment Details page
 - B. In the Container Details page
 - C. In the Cluster Details page
 - D. None of the above
13. What `kubectl` command is used to create a deployment?
 - A. `run`
 - B. `start`
 - C. `initiate`
 - D. `deploy`
14. You are supporting machine learning engineers who are testing a series of classifiers. They have five classifiers, called `ml-classifier-1`, `ml-classifier-2`, etc. They have found that `ml-classifier-3` is not functioning as expected, and they would like it removed from the cluster. What would you do to delete a service called `ml-classifier-3`?
 - A. Run the command `kubectl delete service ml-classifier-3`.
 - B. Run the command `kubectl delete ml-classifier-3`.
 - C. Run the command `gcloud service delete ml-classifier-3`.
 - D. Run the command `gcloud container service delete ml-classifier-3`.
15. What service is responsible for managing container images?
 - A. Kubernetes Engine
 - B. Compute Engine
 - C. Artifact Registry
 - D. Container Engine
16. What command is used to list container images in the command line?
 - A. `gcloud container images list`
 - B. `gcloud container list images`
 - C. `kubectl list container images`
 - D. `kubectl container list images`
17. A data warehouse designer wants to deploy an extract, transform, and load process to Kubernetes. The designer provided you with a list of libraries that should be installed, including drivers for GPUs. You have a number of container images that you think may meet the requirements. How could you get a detailed description of each of those containers?
 - A. Run the command `gcloud container images list details`.
 - B. Run the command `gcloud container images describe`.
 - C. Run the command `gcloud image describe`.
 - D. Run the command `gcloud container describe`.

18. You have just created a deployment and want applications outside the cluster to have access to the pods provided by the deployment. What do you need to do to the deployment?
- A. Give it a public IP address.
 - B. Issue a `kubectl expose deployment` command.
 - C. Issue a `gcloud expose deployment` command.
 - D. Nothing; making it accessible must be done at the cluster level.
19. You have deployed an application to a Kubernetes cluster that processes sensor data from a fleet of delivery vehicles. The volume of incoming data depends on the number of vehicles making deliveries. The number of vehicles making deliveries is dependent on the number of customer orders. Customer orders are high during daytime hours, holiday seasons, and when major advertising campaigns are run. You want to make sure you have enough nodes running to handle the load, but you want to keep your costs down. How should you configure your Kubernetes cluster?
- A. Deploy as many nodes as your budget allows.
 - B. Enable Autoscaling.
 - C. Monitor CPU, disk, and network utilization and add nodes as necessary.
 - D. Write a script to run `gcloud` commands to add and remove nodes when peaks usually start and end, respectively.
20. When using Kubernetes Engine, which of the following might a cloud engineer need to configure?
- A. Nodes, pods, services, and clusters only
 - B. Nodes, pods, services, clusters, and container images
 - C. Nodes, pods, clusters, and container images only
 - D. Pods, services, clusters, and container images only

Chapter 9

Computing with Cloud Run and App Engine

**THIS CHAPTER COVERS THE FOLLOWING
OBJECTIVE OF THE GOOGLE ASSOCIATE
CLOUD ENGINEER CERTIFICATION EXAM:**

- ✓ 3.3 Deploying and implementing Cloud Run and Cloud Functions resources





This chapter describes how to deploy containerized services using Cloud Run and App Engine. Cloud Run, an alternative to App Engine for running containers, is a managed, serverless service. App Engine is no longer included in the Associate

Cloud Engineer Certification Exam Guide; however, it is still included in this chapter because it is still an option in Google Cloud and cloud engineers should be able to support App Engine services even if they are not asked questions about App Engine on an exam.

Cloud Run is designed to support highly scalable, containerized applications written in any language. Cloud Run integrates with developer tools such as Cloud Build, Artifact Registry, and Docker.

App Engine is a platform-as-a-service (PaaS) offering from Google Cloud that allows developers to work within a set of language specific frameworks and deploy scalable applications while requiring only minimal attention to scaling concerns.

Overview of Cloud Run

Cloud Run is a serverless, managed service for running containerized applications. Unlike with App Engine Standard and Cloud Function, you are not restricted to using a limited set of programming languages. Cloud Run supports any application that can be run in a container. An advantage of using Cloud Run is that you do not have to manage infrastructure, such as virtual machines. Cloud Run supports two ways to run code: as a service and as a job.

Cloud Run services are used when your code is used to respond to web requests or events. For example, an API that returns data from a Cloud SQL database could be implemented using containers and run as a service in Cloud Run.

Cloud Run jobs is used when the code executes until a workload is complete. For example, if you needed to transform a set of files stored in Cloud Storage and then load it into Cloud SQL you could run the application in a container using Cloud Run jobs.

Cloud Run Services

Cloud Run services are well suited for web applications, microservices, APIs, and stream data processing.

Cloud Run services are designed to listen to an HTTPS endpoint and respond to requests made to that endpoint. Each Cloud Run service has an endpoint on a unique subdomain of the `run.app` domain and custom domains can be used as well. Endpoints can scale to up to 1,000 container instances with default quotas; you can request a higher quota if needed.

You can also specify a maximum number of container instances to run if you want to limit the number of containers and therefore the cost of running those containers. In addition to providing scalable resources to support traffic to the endpoint, Cloud Run manages TLS. You can use WebSockets, HTTP/2 (end-to-end), and gRPC with these endpoints.

With Cloud Run, you deploy immutable versions of a service. To make an update to a service, you would create a new container image and deploy that as a new version. You can run multiple revisions of the same service in Cloud Run. Also, you can route traffic between different revisions. This is useful when you release a new version and you want to send a small amount of traffic to the latest version so that you can monitor for any problems before rolling out the latest version to all users. If you do discover problems with a revision, you can roll it back and have traffic routed to an earlier, more stable revision.

Cloud Run services are deployed privately by default and require authentication to access. You can control access to services in the following ways:

- With a Cloud IAM policy
- Using ingress settings
- Allowing only authenticated users with Cloud Identity Aware Proxy (IAP)

With Cloud IAM policies, you can assign a role to a group of users so that the group has the permissions specified in the group. For example, to make a service publicly accessible, you can allow access to unauthenticated users. You may want to grant a group of developers permissions to create new versions of a service, and you can do that by assigning the `run.developer` role.

You can also control access at the network level. By default, a Cloud Run endpoint is accessible from anywhere on the Internet using the `run.app` subdomain or a custom domain you define. In addition to using IAM roles to control access to a service, you can control network traffic to the endpoint by specifying an ingress setting. Ingress setting options include:

- Internal, which is the most restrictive and allows only traffic from internal HTTP(S) load balancers, resources within the VPC Service Controls perimeter, VPC networks in the same project or VPC Service Controls perimeter, as well as Eventarc, Cloud Pub/Sub, and Cloud Workflow services in the same project or service control perimeter
- Internal and Cloud Load Balancing, which includes traffic allowed by the Internal setting along with External HTTP(S) load balancers
- All, which is the least restrictive and allows all requests to run that are sent to the service endpoint

Cloud IAP is a security service that protects services by only allowing traffic to the services to come from proxies. When a user tries to access a Cloud Run service protected by Cloud IAM, they are first subject to authentication and authorization by IAP.

Cloud Run Jobs

Cloud Run jobs are programs or scripts that run for a period of time while completing a task and then stop. For example, you could use Cloud Run jobs to run a script to validate files uploaded to a Cloud Storage bucket and then import data in those files to a Cloud SQL

database. Unlike a service, which continues to accept requests to perform tasks, jobs perform a task and terminate. Cloud Run jobs can be scheduled to run on regular schedules. They also array jobs, which can be parallelized. The file processing example just mentioned is a good example of a job that can be parallelized. Instead of processing each file one at a time, multiple containers can be started so that multiple files can be processed simultaneously.

Creating a Cloud Run Service

You can create a Cloud Run service using the console, the Cloud SDK, or programmatically using the API. In this section, we'll review how to create a Cloud Run service using the console.

In the cloud console, navigate to the Cloud Run page and select the option for creating a service. The Create Service page shown in Figure 9.1 opens.

FIGURE 9.1 The form for creating a Cloud Run service

Cloud Run **Create service**

A service exposes a unique endpoint and automatically scales the underlying infrastructure to handle incoming requests. Service name and region cannot be changed later.

☒ Deploy one revision from an existing container image

Container image URL
us-docker.pkg.dev/cloudrun/container/hello [SELECT](#)

[TEST WITH A SAMPLE CONTAINER](#)

Should listen for HTTP requests on \$PORT and not rely on local state. [How to build a container?](#)

☐ Continuously deploy new revisions from a source repository

Service name *
hello

Region *
us-west1 (Oregon) [How to pick a region?](#)

CPU allocation and pricing ?

☒ CPU is only allocated during request processing
You are charged per request and only when the container instance processes a request.

☐ CPU is always allocated
You are charged for the entire lifecycle of the container instance.

Autoscaling ?

Minimum number of instances *
0

Maximum number of instances
100

Set to 1 to reduce cold starts. [Learn more](#)

Cloud Run pricing

Free tier

First 180,000 vCPU-seconds/month
First 360,000 GiB-seconds/month
2 million requests/month

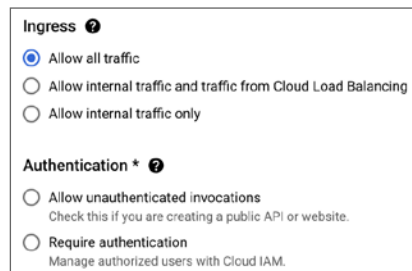
[Check paid tiers details](#)

On this page you will specify a container image URL. You type in a URL or select an image from the Container Registry or the Artifact Registry. By default, you will deploy one revision, but you can select the option to continuously deploy new versions as the source repository is updated.

You will also specify a service name and choose a region to run your service. You have the option of paying only for the time CPU resources are allocated to processing a request or for paying for CPU resources that are always allocated. You can also specify a minimum and maximum number of instances.

Figure 9.2 shows how we can specify an ingress rule. The ingress configuration options were described above. You can also change the default requirement for authentication to allow for unauthenticated access. Unauthenticated access is typically used for websites or public APIs.

FIGURE 9.2 When creating a Cloud Run service, we can choose one of three ingress options.



The image shows a configuration panel for Cloud Run Ingress. It has two sections: 'Ingress' and 'Authentication'. The 'Ingress' section has three radio button options: 'Allow all traffic' (selected), 'Allow internal traffic and traffic from Cloud Load Balancing', and 'Allow internal traffic only'. The 'Authentication' section has two radio button options: 'Allow unauthenticated invocations' (with a subtext 'Check this if you are creating a public API or website.') and 'Require authentication' (with a subtext 'Manage authorized users with Cloud IAM.').

Next, you can specify additional configuration options for containers, connections and security.

For containers (Figure 9.3), you can specify a port, a container command, and container arguments. You can also configure the amount of memory and number of CPUs. Currently, the max memory is 32 GB in preview and 16 GB in general release. Up to 8 CPUs are currently supported in preview and up to 4 in general release. (Services in preview are not covered by the Google Cloud SLA, but services in general release are covered by SLAs.)

By default, requests will time out after 5 minutes, but you can specify a shorter or longer period ranging from 1 to 60 minutes.

Cloud Run has two execution environments: first generation and second generation. The second generation supports features such as filesystem access and faster performance. By default, Cloud Run will choose an environment for you. You can also specify environment variables for the container and reference secrets in the container.

On the Connections tab (Figure 9.4), you can indicate if you want to use HTTP/2 end-to-end, which supports gRPC streaming services, and if you want to support session affinity. Session affinity will route requests from a client to the same container, if possible. You can also specify a Cloud SQL connection for services that use a Cloud SQL database. You can also use create a VPC Connector to use Serverless VPC Access to connect your Cloud Run service to other resources in your VPC, such as Compute Engine instances or a Cloud Memorystore cache.

FIGURE 9.3 Configuring container parameters in a Cloud Run service

Container, Connections, Security

CONTAINER

CONNECTIONS

SECURITY

General

Container port

8080

Requests will be sent to the container on this port. We recommend listening on \$PORT instead of this specific number.

Container command

Leave blank to use the entry point command defined in the container image.

Container arguments

Arguments passed to the entry point command.

Capacity

Memory

512 MiB

▼

Memory to allocate to each container instance.

CPU

1

▼

Number of vCPUs allocated to each container instance.

Request timeout

300

seconds

Time within which a response must be returned (maximum 3600 seconds).

Maximum requests per container

80

The maximum number of concurrent requests that can reach each container instance.
[What is concurrency?](#)

Execution environment

The execution environment your container runs in. [Learn More](#)

☒ Default

Cloud Run will select a suitable execution environment for you.

☐ First generation

☐ Second generation

PREVIEW

File system access, full Linux compatibility, faster performance.

Environment variables

+ ADD VARIABLE

Secrets ?

REFERENCE A SECRET

CREATE

CANCEL

FIGURE 9.4 Configuring connection parameters in a Cloud Run service

The screenshot shows the 'Container, Connections, Security' configuration panel for a Cloud Run service. The 'CONNECTIONS' tab is selected, indicated by a blue underline. The panel has three tabs: 'CONTAINER', 'CONNECTIONS', and 'SECURITY'. Below the tabs, there is a text block: 'Connect to other Google Cloud services like Google Cloud Storage or Google Cloud Firestore directly from your code. [Learn more](#)'. Below this, there are two checkboxes: 'Use HTTP/2 end-to-end' and 'Session affinity'. The 'Session affinity' checkbox is followed by a 'PREVIEW' badge and a description: 'Best effort to route requests from the same client to the same container instance.' Below these, there is a section for 'Cloud SQL connections' with a help icon and a '+ ADD CONNECTION' button. The 'VPC Connector' section has a 'Network' dropdown menu set to 'None' and a refresh icon. Below the dropdown, there is a text block: 'Access resources on a VPC. [Learn more](#) or [create a Serverless VPC Connector](#)'. Below this, there are two radio buttons: 'Route only requests to private IPs through the VPC connector' (selected) and 'Route all traffic through the VPC connector'. At the bottom, there are two buttons: 'CREATE' and 'CANCEL'.

Container, Connections, Security

CONTAINER **CONNECTIONS** SECURITY

Connect to other Google Cloud services like Google Cloud Storage or Google Cloud Firestore directly from your code. [Learn more](#)

☐ Use HTTP/2 end-to-end
Use if your container is a gRPC streaming server or is able to directly handle requests in HTTP/2 cleartext. [Learn more](#)

☐ Session affinity **PREVIEW**
Best effort to route requests from the same client to the same container instance.

Cloud SQL connections ?

+ ADD CONNECTION

VPC Connector

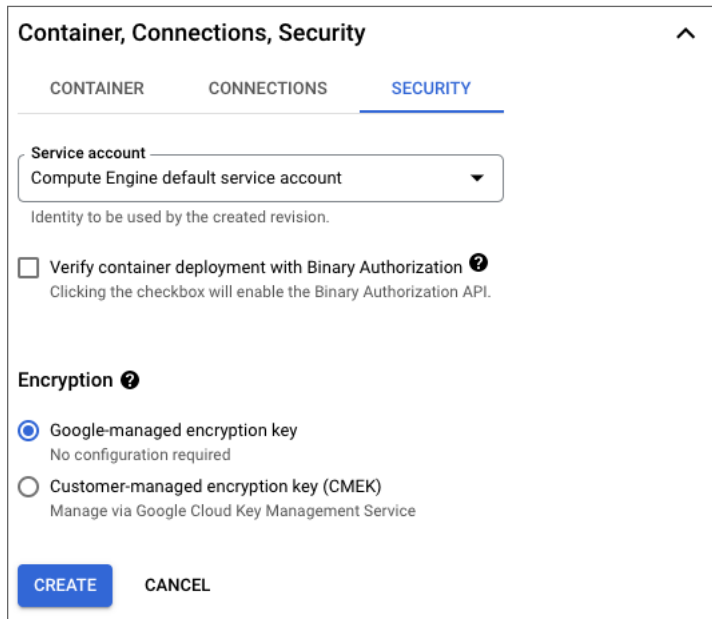
Network
None

Access resources on a VPC. [Learn more](#) or [create a Serverless VPC Connector](#)

☒ Route only requests to private IPs through the VPC connector
☐ Route all traffic through the VPC connector

CREATE **CANCEL**

On the Security tab (Figure 9.5), you can specify a service account to use with this service. You can also require Binary Authorization before deploying a container. Binary Authorization is a service that verifies that containers meet requirements specified in a policy governing the deployment of containers in Cloud Run and Kubernetes Engine, among other services. You can also specify if you want to use Google-managed or customer-managed encryption keys.

FIGURE 9.5 Configuring security parameters in a Cloud Run service

The screenshot shows a configuration window titled "Container, Connections, Security" with a close button (upward arrow) in the top right corner. Below the title are three tabs: "CONTAINER", "CONNECTIONS", and "SECURITY", with "SECURITY" being the active tab. The "Service account" section features a dropdown menu currently set to "Compute Engine default service account" and a note stating "Identity to be used by the created revision." Below this is an unchecked checkbox labeled "Verify container deployment with Binary Authorization" with a help icon and a note: "Clicking the checkbox will enable the Binary Authorization API." The "Encryption" section, also with a help icon, contains two radio button options: "Google-managed encryption key" (selected) with the subtext "No configuration required", and "Customer-managed encryption key (CMEK)" with the subtext "Manage via Google Cloud Key Management Service". At the bottom are two buttons: a blue "CREATE" button and a "CANCEL" button.

Creating a Cloud Run Job

Creating a job in Cloud Run is similar to creating a service. From the Cloud Run page on the cloud console, select the Jobs tab and Create A Job to open a form similar to Figure 9.6.

As when creating a service, you will specify a container image URL and region. You will also provide a job name and the number of times you want to run the container; the default is one time.

On the General tab, you can specify container configuration parameters (Figure 9.7). Some parameters, such as Container Command, Container Arguments, Memory, and CPU are similar to what you saw when configuring a Cloud Run service. In addition, you can specify the number of retries of failed tasks and a parallelism parameter to control the number of concurrent tasks. There is also an option to execute a job immediately.

On the Variables & Secrets tab (Figure 9.8), you can specify environment variables and references to stored secrets.

As with Cloud Run services, you can specify Cloud SQL connections and a VPC connector on the Connections tab (Figure 9.9). On the Security tab (Figure 9.10), you can specify a service account for the service and encryption key management.

Before the release of Cloud Run, developers often chose to run their services on App Engine.

FIGURE 9.6 Creating a Cloud Run job

Cloud Run

← Create job **PREVIEW**

A Cloud Run job executes containers to completion. Job name and region cannot be changed later.

Container image URL [SELECT](#)

[TEST WITH A SAMPLE CONTAINER](#)

[How to build a container?](#)

Job name *

i Cloud Run Admin API needs to be enabled to use this option. [ENABLE](#)

Number of tasks *
1

The number of times to run the container. All tasks must succeed in order for a job to succeed.

Container, Variables & Secrets, Connections, Security ▼

☐ Execute job immediately

CREATE **CANCEL**

App Engine Components

App Engine is available in a Standard and a Flexible version. App Engine Standard applications consist of four components:

- Application
- Service
- Version
- Instance

An App Engine application is a high-level resource created in a project; that is, each project can have one App Engine application. All resources associated with an App Engine app are created in the region specified when the app is created.

FIGURE 9.7 Configuring container parameters for a Cloud Run job

Container, Variables & Secrets, Connections, Security

< GENERAL

VARIABLES & SECRETS

CONNECTIONS >

Container command

Leave blank to use the entry point command defined in the container image.

Container arguments

Arguments passed to the entry point command.

Task capacity

Memory

512 MiB

Memory to allocate to each container instance.

CPU

1

Number of vCPUs allocated to each container instance.

Task timeout *

600

seconds

The maximum amount of time an instance can run for (maximum 3600 seconds).

Number of retries per failed task *

0

Parallelism

The maximum number of tasks running at the same time.

☒ Run as many tasks concurrently as possible

For 1 CPU Cloud Run can run up to 100 tasks at a time.

☐ Limit the number of concurrent tasks

Use this option to limit the number of concurrent requests to backing resources such as databases or file systems.

☐ Execute job immediately

FIGURE 9.8 Configuring variables and secrets for a Cloud Run job

Container, Variables & Secrets, Connections, Security

< GENERAL

VARIABLES & SECRETS

CONNECTIONS >

Environment variables

+ ADD VARIABLE

Secrets ?

REFERENCE A SECRET

FIGURE 9.9 Configuring connection parameters for a Cloud Run job

The screenshot shows the 'Connections' tab in the App Engine console. The tab is titled 'Container, Variables & Secrets, Connections, Security' with a dropdown arrow. Below the title bar are three tabs: 'GENERAL', 'VARIABLES & SECRETS', and 'CONNECTIONS' (which is selected and highlighted in blue). The main content area is titled 'Cloud SQL connections' with a help icon. Below this is a button labeled '+ ADD CONNECTION'. Further down is the 'VPC Connector' section. It features a 'Network' dropdown menu currently set to 'None' with a refresh icon to its right. Below the dropdown is a link: 'Access resources on a VPC. [Learn more](#) or [create a Serverless VPC Connector](#)'. At the bottom are two radio button options: 'Route only requests to private IPs through the VPC connector' (which is selected) and 'Route all traffic through the VPC connector'.

FIGURE 9.10 Configuring security parameters for a Cloud Run job

The screenshot shows the 'Security' tab in the App Engine console. The tab is titled 'Container, Variables & Secrets, Connections, Security' with a dropdown arrow. Below the title bar are three tabs: 'VARIABLES & SECRETS', 'CONNECTIONS', and 'SECURITY' (which is selected and highlighted in blue). The main content area is titled 'Service account' with a dropdown menu showing 'Compute Engine default service account'. Below the dropdown is the text 'Identity to be used by the job.' Further down is the 'Encryption' section with a help icon. It features two radio button options: 'Google-managed encryption key' (which is selected) with the subtext 'No configuration required', and 'Customer-managed encryption key (CMEK)' with the subtext 'Manage via Google Cloud Key Management Service'.

Apps have at least one service, which is the code executed in the App Engine environment. Because multiple versions of an application's code base can exist, App Engine supports versioning of apps. A service can have multiple versions, and these are usually slightly different, with newer versions incorporating new features, bug fixes, and other changes relative to earlier versions. When a version executes, it creates an instance of the app.

Services are typically structured to perform a single function with complex applications made up of multiple services, known as *microservices*. One microservice may handle API requests for data access, while another microservice performs authentication and a third records data for billing purposes.

Services are defined by their source code and their configuration file. The combination of those files constitutes a version of the app. If you slightly change the source code or configuration file, it creates another version. In this way, you can maintain multiple versions of your application at one time, which is especially helpful for testing new features on a small number of users before rolling the change out to all users. If bugs or other problems occur with a version, you can easily roll back to an early version. Another advantage of keeping multiple versions is that they allow you to migrate and split traffic, which we'll describe in more detail later in the chapter.

Deploying an App Engine Application

The Google Associate Cloud Engineer certification exam does not require engineers to write an application, but we are expected to know how to deploy one. In this section, you will download a Hello World example from Google and use it as a sample application that you will deploy. The app is written in Python, so you'll use the Python runtime in App Engine.

Deploying an App Using Cloud Shell and SDK

First, you will work in a terminal window using Cloud Shell, which you can start from the console by clicking the Cloud Shell icon. Make sure `gcloud` is configured to work with App Engine by using the following command:

```
gcloud components install app-engine-python
```

This command will install or update the App Engine Python library as needed. If the library is up-to-date, you will receive a message saying that.

When you open Cloud Shell, you may have a directory named `python-docs-samples`. This contains a number of example applications, including the Hello World app we'll use. If you do not see this directory, you can download the Hello World app from Google using this:

```
git clone https://github.com/GoogleCloudPlatform/python-docs-samples
```

Next, change your working directory to the directory with the Hello World app, using the following:

```
cd python-docs-samples/appengine/standard_python3/hello_world
```

If you list the files in the directory, you will see five files:

- `app.yaml`
- `main.py`
- `main_test.py`
- `requirements.txt`
- `requirements-test.txt`

Here you are primarily concerned with the `app.yaml` file. (Figure 9.11).

FIGURE 9.11 The contents of an `app.yaml` file for a Python 3 application

```
1 # Copyright 2021 Google LLC
2 #
3 # Licensed under the Apache License, Version 2.0 (the "License");
4 # you may not use this file except in compliance with the License.
5 # You may obtain a copy of the License at
6 #
7 #     http://www.apache.org/licenses/LICENSE-2.0
8 #
9 # Unless required by applicable law or agreed to in writing, software
10 # distributed under the License is distributed on an "AS IS" BASIS,
11 # WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
12 # See the License for the specific language governing permissions and
13 # limitations under the License.
14
15 runtime: python27
16 api_version: 1
17 threadsafe: true
18
19 handlers:
20 - url: /*
21   script: main.app
```

In this example, the app configuration file specifies the version of Python to use. Depending on the version of Python you are using, the `app.yaml` file may also contain the API version you are deploying, a Python parameter called `threadsafe`, and environment variables.

To deploy your app, you can use the following command:

```
gcloud app deploy app.yaml
```

However, `app.yaml` is the default, so if you are using that for the filename, you do not have to specify `app.yaml` in the deploy command.

This command must be executed from the directory with the `app.yaml` file. The `gcloud app deploy` command has some optional parameters:

- `--version`, to specify a custom version ID
- `--project`, to specify the project ID to use for this app
- `--no-promote`, to deploy the app without routing traffic to it

You can see the output of the Hello World program by navigating in a browser to your project URL, such as `https://gcp-pace-project.appspot.com`. The project URL is the project name followed by `.appspot.com`.



You can also assign a custom domain if you would rather not use an `appspot.com` URL. You can do this from the Add New Custom Domain function on the App Engine Settings page.

You can stop serving versions using the `gcloud app versions stop` command and passing a list of versions to stop. For example, to stop serving versions named `v1` and `v2`, use the following:

```
gcloud app versions stop v1 v2
```

Scaling App Engine Applications

Instances are created to execute an application on an App Engine–managed server. App Engine can automatically add or remove instances as needed based on load. When instances are scaled based on load, they are called *dynamic* instances. These dynamic instances help optimize your costs by shutting down when demand is low.

Alternatively, you can configure your instances to be resident or running all the time. These are optimized for performance so that users will wait less time while an instance is started.

Your configuration determines whether an instance is resident or dynamic. If you configure autoscaling or basic scaling, then instances will be dynamic. If you configure manual scaling, then your instances will be resident.

To specify automatic scaling, add a section to `app.yaml` that includes the term `automatic_scaling` followed by key-value pairs of configuration options. These include the following:

- `target_cpu_utilization`
- `target_throughput_utilization`
- `max_concurrent_requests`
- `max_instances`
- `min_instances`
- `max_pending_latency`
- `min_pending_latency`

target_cpu_utilization Specifies the maximum CPU utilization that occurs before additional instances are started.

target_throughput_utilization Specifies the maximum number of concurrent requests before additional instances are started. This is specified as a number between 0.5 and 0.95.

max_concurrent_requests Specifies the maximum concurrent requests an instance can accept before starting a new instance. The default is 10; the max is 80.

max_instances and min_instances Indicates the range of number of instances that can run for this application.

max_pending_latency and min_pending_latency Indicates the maximum and minimum time a request will wait in the queue to be processed.

You can also use basic scaling to enable automatic scaling. The only parameters for basic scaling are `idle_timeout` and `max_instances`.



Real World Scenario

Microservices vs. Monolithic Applications

Scalable applications are often written as collections of microservices. This has not always been the case. In the past, many applications were monolithic, or designed to include all functionality in a single compiled program or script. This may sound like a simpler, easy way to manage applications, but in practice it creates more problems than it solves:

- Any changes to the application require redeploying the entire application, which can take longer than deploying microservices. Developers tended to bundle changes before releasing them.
- If a bundled release had a bug in a feature change, then all feature changes would be rolled back when the monolithic application was rolled back.
- It was difficult to coordinate changes when teams of developers had to work with a single file or a small number of files of source code.

Microservices divide application code into single-function applications, allowing developers to change one service and roll it out without impacting other services. Source code management tools, like Git, make it easy for multiple developers to contribute components of a larger system by coordinating changes to source code files. This single-function code and the easy integration with other code promote more frequent updates and the ability to test new versions before rolling them out to all users at once.

Splitting Traffic Between App Engine Versions

If you have more than one version of an application running, you can split traffic between the versions. App Engine provides three ways to split traffic: by IP address, by HTTP cookie, and by random selection. IP address splitting provides some stickiness, so a client is always routed to the same split, at least as long as the IP address does not change. HTTP cookies are useful when you want to assign users to versions. Random selection is useful when you want to evenly distribute workload.

When using IP address splitting, App Engine creates a hash—that is, a number generated based on an input string between 0 and 999, using the IP address of each version. This can create problems if users change IP address, such as if they start working with the app in the office and then switch to a network in a coffee shop. If state information is maintained in a version, it may not be available after an IP address change.

The preferred way to split traffic is with a cookie. When you use a cookie, the HTTP request header for a cookie named GOOGAPPUID contains a hash value between 0 and 999. With cookie splitting, a user will access the same version of the app even if the user's IP address changes. If there is no GOOGAPPUID cookie, then the traffic is routed randomly.

The command to split traffic is `gcloud app services set-traffic`. Here's an example:

```
gcloud app services set-traffic serv1 --splits v1=.4,v2=.6
```

This command will split traffic, with 40 percent going to version 1 of the service named `serv1` and 60 percent going to version 2. If no service name is specified, then all services are split.

The `gcloud app services set-traffic` command takes the following parameters:

- `--migrate` indicates that App Engine should migrate traffic from the previous version to the new version.
- `--split-by` specifies how to split traffic using either IP or cookies. Possible values are `ip`, `cookie`, and `random`.

You can also migrate traffic from the console. Navigate to the Versions page and select the Migrate command.

Summary

Cloud Run is a serverless, managed service for running containerized applications. Cloud Run supports any application that can be run in a container. Cloud Run services are used when your code is used to respond to web requests or events. Cloud Run jobs are used when the code executes until a workload is complete. When working with services or jobs, you can configure several categories of parameters, including container, connection, and security settings.

App Engine Standard is a serverless platform for running applications in language-specific environments. As a cloud engineer, you are expected to know how to deploy and scale App Engine applications. App Engine applications consist of services, versions, and instances. You can have multiple versions running at one time. You can split traffic between versions and have all traffic automatically migrate to a new version. App Engine applications are configured through `app.yaml` configuration files. You can specify the language environment, scaling parameters, and other parameters to customize your deployment. App Engine is no longer listed in the Google Cloud Associate Cloud Engineer Exam Guide, but it is included here because cloud engineers should be familiar with this popular Google Cloud service.

Exam Essentials

Be able to describe Cloud Run as a serverless service for running containers. Cloud Run is a serverless, managed service for deploying, scaling, and managing services. Although there are no servers to configure, you can specify parameters to control the number of instances running at any time, the security used to protect the service, as well as connection configuration details.

Know how Cloud Run services are used to run long-lived services like websites and API servers. Cloud Run services run containers continuously. You have the option to pay only for CPU resources used when responding to requests, or you can choose to have a container always available and pay for the time the CPU resources are allocated.

Know how Cloud Run jobs are used to run tasks, such as loading data into a database. Cloud Run jobs are configured similarly to Cloud Run services. You can specify that jobs use multiple containers running simultaneously. This is useful when running parallelizable workloads.

Be able to describe the structure of App Engine Standard applications. These consist of services, versions, and instances. Services usually provide a single function. Versions are different versions of code running in the App Engine environment. Instances are managed instances running the service.

Know how to deploy an App Engine app. This includes configuring the App Engine environment using the `app.yaml` file. Know that a project can have only one App Engine app at a time. Know how to use the `gcloud app deploy` command.

Understand the various scaling options. Three scaling options are autoscaling, basic scaling, and manual scaling. Only autoscaling and basic scaling are dynamic. Manual scaling creates resident instances. Autoscaling allows for more configuration options than basic scaling.

Review Questions

You can find the answers in the Appendix.

1. You want to provide your customers with an API to allow them to query a database with proprietary industry data. You want your developers to focus on adding new features and not on administering servers. Which of the following Google Cloud services would you choose?
 - A. Compute Engine managed instance groups
 - B. Computer Engine unmanaged instance groups
 - C. Cloud Run services
 - D. Cloud Run jobs
2. You are working for a biomedical research group that has several hundred data files stored in Cloud Storage. They have a statistical analysis program that analyzes a data file and writes the output to another Cloud Storage bucket. They have agreed with you that deploying the program in a container is the best option, but they are unsure which Google Cloud service to use to run the container. What would you recommend?
 - A. Kubernetes Engine
 - B. Compute Engine
 - C. App Engine Flexible
 - D. Cloud Run jobs
3. You are working for a climate change research group that has tens of thousands of public weather data files stored in Cloud Storage. They are building a model to predict sea levels in the near future. The data in each file can be analyzed independently of other files. They plan to use Cloud Run jobs for this task. What feature of Cloud Run jobs would you recommend they use?
 - A. Customer-managed encryption keys
 - B. Array jobs
 - C. Cloud SQL Connection
 - D. A private IP address
4. An application administrator has asked for your help with configuring a Cloud Run service. The application administrator would like to have all client requests routed to the same container if possible. How would you suggest the administrator accomplish this?
 - A. Use Cloud SQL Connection.
 - B. Use array jobs.
 - C. Configure the connection in the Cloud Run Service to support session affinity.
 - D. Use a private IP address.

5. You are deploying a service on Cloud Run. The service has access to personal identifiable information (PII) and for compliance reasons, you do not want to expose the service to any traffic outside of internal traffic in your Google Cloud environment. What ingress configuration would you use?
 - A. Internal
 - B. Internal and Cloud Load Balancing
 - C. All
 - D. PII proxy traffic
6. You want to use a service account specifically created for a Cloud Run service. Where would you specify that in the cloud console?
 - A. On the Connections tab
 - B. On the Security tab
 - C. On the Container tab
 - D. On the Variables & Secrets tab
7. A group of developers need the ability to deploy new versions of a service running in Cloud Run. How would you configure that access?
 - A. Using IAM
 - B. Using Cloud Identity Aware Proxy (IAP)
 - C. Using an ingress policy
 - D. Using the Security tab in the Cloud Run console
8. Your team deployed a Cloud Run service last month that accesses a Cloud SQL database. The database team has changed their system and now use a Memcached cache running in Cloud Memorystore. You have to change your Cloud Run service to access the Cloud Memorystore cache. What would you use to do that?
 - A. Cloud SQL Connection
 - B. Cloud IAP Proxy
 - C. VPC Connection
 - D. Session affinity
9. A service is deployed to Cloud Run services and will communicate with clients using gRPC. What should you configure to enable this protocol to work with the service?
 - A. External Load Balancing
 - B. Cloud Identity Aware Proxy (IAP)
 - C. Session affinity
 - D. HTTP/2 end-to-end

10. What Google Cloud services can be used to store and access container images accessible from Cloud Run?
 - A. Container Registry only
 - B. Container Registry and Artifact Registry
 - C. Artifact Registry only
 - D. Container Registry, Artifact Registry, and Kubernetes Engine
11. You have designed a microservice that you want to deploy to production. Before it can be deployed, you have to review how you will manage the service life cycle. The architect is particularly concerned about how you will deploy updates to the service with minimal disruption. What aspect of App Engine components would you use to minimize disruptions during updates to the service?
 - A. Services
 - B. Versions
 - C. Instance groups
 - D. Instances
12. You've just released an application running in App Engine Standard. You notice that there are peak demand periods in which you need up to 12 instances, but most of the time 5 instances are sufficient. What is the best way to ensure that you have enough instances to meet demand without spending more than you have to?
 - A. Configure your app for autoscaling and specify max instances of 12 and min instances of 5.
 - B. Configure your app for basic scaling and specify max instances of 12 and min instances of 5.
 - C. Create a cron job to add instances just prior to peak periods and remove instances after the peak period is over.
 - D. Configure your app for instance detection and do not specify a max or minimum number of instances.
13. What command should you use to deploy an App Engine app from the command line?
 - A. `gcloud components app deploy`
 - B. `gcloud app deploy`
 - C. `gcloud components instances deploy`
 - D. `gcloud app instance deploy`
14. You have deployed a Django 1.5 Python application to App Engine. This version of Django requires Python 3. For some reason, App Engine is trying to run the application using Python 2. What file would you check and possibly modify to ensure that Python 3 is used with this application?
 - A. `app.config`
 - B. `app.yaml`
 - C. `services.yaml`
 - D. `deploy.yaml`

15. You are concerned that as users make connections to your application, the performance will degrade. You want to make sure that more instances are added to your App Engine application when there are more than 20 concurrent requests. What parameter would you specify in `app.yaml`?
- A. `max_concurrent_requests`
 - B. `target_throughput_utilization`
 - C. `max_instances`
 - D. `max_pending_latency`
16. What parameters can be configured with basic scaling?
- A. `max_instances` and `min_instances`
 - B. `idle_timeout` and `min_instances`
 - C. `idle_timeout` and `max_instances`
 - D. `idle_timeout` and `target_throughput_utilization`
17. The runtime parameter in `app.yaml` is used to specify what?
- A. The script to execute
 - B. The URL to access the application
 - C. The language runtime environment
 - D. The maximum time an application can run
18. You work for a startup, and costs are a major concern. You are willing to take a slight performance hit if it will save you money. How should you configure the scaling for your apps running in App Engine?
- A. Use dynamic instances by specifying autoscaling or basic scaling.
 - B. Use resident instances by specifying autoscaling or basic scaling.
 - C. Use dynamic instances by specifying manual scaling.
 - D. Use resident instances by specifying manual scaling.
19. What parameter to `gcloud app services set-traffic` is used to specify the method to use when splitting traffic?
- A. `--split-traffic`
 - B. `--split-by`
 - C. `--traffic-split`
 - D. `--split-method`
20. What are valid methods for splitting traffic in App Engine?
- A. By IP address only
 - B. By HTTP cookie only
 - C. Randomly and by IP address only
 - D. By IP address, HTTP cookies, and randomly

Chapter 10

Computing with Cloud Functions

**THIS CHAPTER COVERS THE FOLLOWING
OBJECTIVES OF THE GOOGLE ASSOCIATE
CLOUD ENGINEER CERTIFICATION EXAM:**

- ✓ 3.3 Deploying and implementing Cloud Run and Cloud Functions resources





In this chapter, we describe the purpose of Cloud Functions as well as how to implement and deploy the functions. We will use examples of the functions written in Python. If you are unfamiliar with Python, that should not dissuade you from following along, as we will explain the important details of Python functions. You will learn how to use the Cloud Console and `gcloud` commands to create and manage Cloud Functions.

Introduction to Cloud Functions

Cloud Functions is a serverless compute service provided by Google Cloud. Cloud Functions is similar to Cloud Run in that they are both serverless compute options. A primary difference is that Cloud Run supports both services with HTTP endpoints that can run continuously and batch jobs that run to completion and terminate, whereas Cloud Functions are relatively short-running functions (up to 60 minutes for HTTP functions and 10 minutes for event-driven functions).

Cloud Functions are well suited for event-driven processing. For example, your customers may upload files to Cloud Storage, which are analyzed for quality control checks, and if the checks are passed, a message is written to a Pub/Sub topic, a messaging service in GCP, which is read by another service that continues the processing.

At the time of writing, there are two supported versions of Cloud Functions: the original Cloud Functions and Second-Generation Cloud Functions. The Second-Generation Cloud Functions offer larger instances, improved concurrency, pre-warmed instances, and traffic management. Second-Generation Functions also support Eventarc, a GCP service that supports managing the flow of events in microservices architectures. Eventarc greatly expands the range of event sources supported by Cloud Functions. The Second-Generation functions also support CloudEvents, an open specification for describing cloud events.

Events, Triggers, and Functions

Here are some terms you need to know before going any further into Cloud Functions:

- Events
- Triggers
- Functions

Events are a particular action that happens in the cloud, such as a file is uploaded to Cloud Storage, or a message that is written to a Pub/Sub message (called a *topic*) queue. There are different kinds of actions associated with each of the events. The first generation of Cloud Functions, GCP supports events in several categories:

- HTTP
- Cloud Storage
- Cloud Pub/Sub
- Cloud Firestore
- Cloud Firebase

The HTTP type of event allows developers to invoke a function by making an HTTP request using POST, GET, PUT, DELETE, and OPTIONS calls. Events in Cloud Storage include uploading, deleting, and archiving a file. Cloud Pub/Sub has an event for publishing a message. Cloud Firestore is a NoSQL document database, and Cloud Functions supports create, update, delete, and write events. Firebase is a database service used for mobile application development and supports database triggers, remote configuration triggers, and authentication triggers.

Second-Generation Cloud Functions use Eventarc triggers, which are configured based on a provider, such as services supported in First-Generation Cloud Functions like Cloud Pub/Sub; additional GCP services, such as Cloud Task, Cloud Dataproc, Cloud DNS, and Network Management; as well as non-GCP specific services such as OAuth 2.0. The specific event types will vary by provider. For example, OAuth 2.0 providers support GetToken, GetTokenInfo, and RevokeToken events. Network Management events include CreateConnectivityTest, GetConnectivityTest, ListConnectivityTest.

For each of the Cloud Functions-enabled events that can occur, you can define a trigger. A *trigger* is a way of responding to an event.

Triggers have an associated *function*. The function, passed arguments with data about the event, executes in response to the event.

Runtime Environments

Functions run in their own environment. Each time a function is invoked, it is run in a separate instance from all other invocations. There is no way to share information between invocations of functions using only Cloud Functions. If you need to coordinate the updating of data, such as keeping a global count, or need to keep information about the state of functions, such as the name of the last event processed, then you should use a database like Cloud Firestore or a file in Cloud Storage.

Google currently supports several runtime environments:

- Node.js
- Python
- Go

- Java
- .NET
- Ruby
- PHP

For each of these runtimes, particular versions may be recommended over others. For example, at the time of writing, the recommended Node.js is Node.js 16 and the recommended version of Python is 3.9. See Cloud Functions documentation (<https://cloud.google.com/functions>) for the latest supported, recommended, and deprecated versions of the runtimes.

Let's walk through an example function. Say you want to record information about file uploads to a particular bucket in Cloud Storage. You can do this by writing a Python function that receives information about an event and then issues print commands to send a description of that data to a log file. Here is the Python code:

```
def cloud_storage_function_test(event_data, event_context):  
    print('Event ID: {}'.format(event_context.event_id))  
    print('Event type: {}'.format(event_context.event_type))  
    print('File: {}'.format(event_data['name']))
```

The first line begins the creation of a function called `cloud_storage_function_test`. It takes two arguments, `event_data` and `event_context`. These are Python data structures with information about the object of the event and about the event itself. The next three lines print the values of the `event_id`, `event_type`, and name of the file. Since this code will be run as a function, and not interactively, the output of a print statement will go to the function's log file.

Python functions should be saved in a file called `main.py`.



Real World Scenario

Making Documents Searchable

Litigation, or lawsuits, between businesses often involve reviewing a large volume of documents. Electronic documents may be in readily searchable formats, such as Microsoft Word documents or PDF files. Others may be scanned images of paper documents. In that case, the file needs to be preprocessed using an optical character recognition (OCR) program.

Functions can be used to automate the OCR process. When a file is uploaded, a Cloud Storage trigger fires and invokes a function. The function determines whether the file is in a searchable format or needs to be preprocessed by the OCR program. If the file does require OCR processing, the function writes the location of the file into a Pub/Sub topic.

A second function is bound to a new message event. When a file location is written in a message, the function calls the OCR program to scan the document and produce a searchable version of the file. That searchable version is written to a Cloud Storage bucket, where it can be indexed by the search tool along with other searchable files.

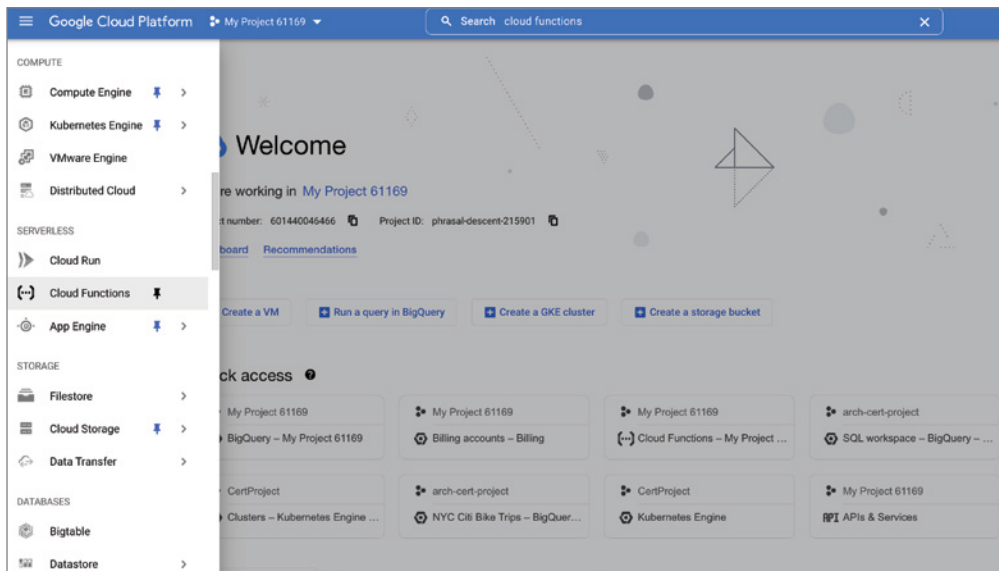
Cloud Functions Receiving Events from Cloud Storage

Cloud Storage is GCP's object storage. This service allows you to store files in containers known as *buckets*. We will go into more detail about Cloud Storage in Chapter 11, “Planning Storage in the Cloud,” but for this chapter you just need to understand that Cloud Storage uses buckets to store files. When files are created, deleted, or archived, or their meta-data changes, an event can invoke a function. Let's go through an example of deploying a function for Cloud Storage Events using Cloud Console and `gcloud` commands in Cloud SDK and Cloud Shell.

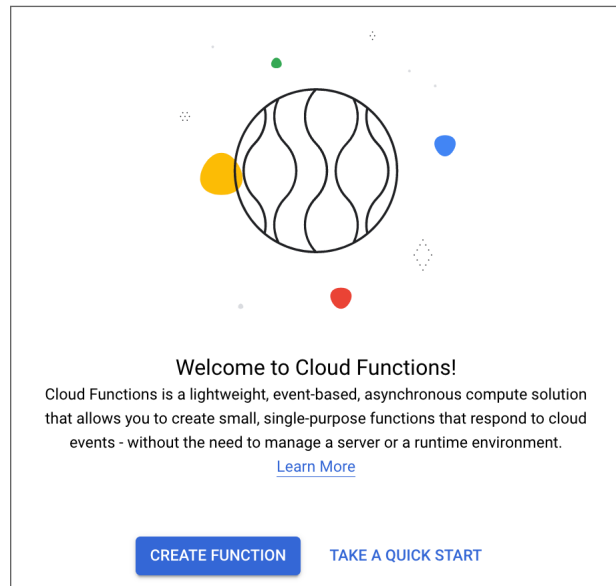
Deploying a Cloud Function for Cloud Storage Events Using Cloud Console

To create a function using Cloud Console, select the Cloud Function options from the vertical menu in the console, as shown in Figure 10.1.

FIGURE 10.1 Opening the Cloud Functions console



In the Cloud Functions console, you may be prompted to enable the Cloud Functions API if it is not already enabled. After the API is enabled, you will have the option to create a new function, as shown in Figure 10.2.

FIGURE 10.2 The Create Function button in Cloud Console

When you create a new function in the console, a form such as the one in Figure 10.3 appears. In Figure 10.3, the options, which have been filled in, include:

- Function name
- Region
- Trigger type
- Event type
- Bucket

In the following example, we are uploading a file containing the function code. The contents of that file are as follows:

```
def cloud_storage_function_test(event_data, event_context):  
    print('Event ID: {}'.format(event_context.event_id))  
    print('Event type: {}'.format(event_context.event_type))  
    print('File: {}'.format(event_data['name']))
```

The function name is what Google Cloud will use to refer to this function. Memory Allocated is the amount of memory that will be available to the function. Memory options range from 128 MB to 8 GB for original Cloud Functions and 16 GB with Second-Generation Cloud Functions. Trigger is one of the defined triggers, such as HTTP, Cloud Pub/Sub, and Cloud Storage. There are several options for specifying where to find the source code, including uploading it, getting it from Cloud Storage or a Cloud Source repository, or

entering the code in an editor. Runtime indicates which runtime to use to execute the code. The editor is where you can enter function code. Finally, the function to execute is the name of the function in the code that should run when the event occurs.

FIGURE 10.3 Creating a function in the console

Cloud Functions | Create function

1 Configuration — 2 Code

Basics

Environment
1st gen

Function name *
cloud_storage_function_test

Region
us-central1

Trigger

Cloud Storage

Trigger type
Cloud Storage

Event type *
Finalize/Create

Bucket *
slg-cf-1 BROWSE

☐ Retry on failure

SAVE CANCEL

Runtime, build, connections and security settings

After a function is created, you will see a list of functions in the Cloud Functions console, as shown in Figure 10.4.

FIGURE 10.4 List of functions in the console

Cloud Functions

Functions

RELEASE NOTES

LEARN

Filter

Filter functions

<input type="checkbox"/>	<div></div>	Environment	Name <div></div>	Last deployed	Region	Trigger	Runtime	Memory allocated
<input type="checkbox"/>	<div></div>	1st gen	function-1	Nov 19, 2022, 9:54:25 AM	us-central1	HTTP	Node.js 16	256 MB

Deploying a Cloud Function for Cloud Storage Events Using *gcloud* Commands

The first step to using `gcloud` commands for Cloud Functions is to make sure you have the latest version of the commands installed. You can update standard `gcloud` commands using this:

```
gcloud components update
```

If any of the commands for the environment you choose are in beta, you can ensure that they are installed with the following command:

```
gcloud components install beta
```

Let's assume you have created a Cloud Storage bucket called `gcp-ace-exam-test-bucket`. You can deploy a function using the `gcloud functions deploy` command. This command takes the name of a function as its argument. There are also three parameters you will need to pass in:

- `runtime`
- `trigger-resource`
- `trigger-event`

`runtime` indicates whether you are using Python 3.7, Node.js 6, or Node.js 8. `trigger-resource` indicates the bucket name associated with the trigger. `trigger-event` is the kind of event that will trigger the execution of the function. The possible options are as follows:

- `google.storage.object.finalize`
- `google.storage.object.delete`
- `google.storage.object.archive`
- `google.storage.object.metadataUpdate`

`finalize` is the term used to describe when a file is fully uploaded.

Whenever a new file is uploaded to the bucket called `gcp-ace-exam-test-bucket`, we want to execute the `cloud_storage_function_test`. We accomplish this by issuing the following command:

```
gcloud functions deploy cloud_storage_function_test \
    --runtime python39 \
    --trigger-resource gcp-ace-exam-test-bucket \
    --trigger-event google.storage.object.finalize
```

When you upload a file to the bucket, the function will execute and create a log message. When you are done with the function and want to delete it, you can use the `gcloud` function's `delete` command, like so:

```
gcloud functions delete cloud_storage_function_test
```

Cloud Functions Receiving Events from Pub/Sub

A function can be executed each time a message is written to a Pub/Sub topic. You can use Cloud Console or `gcloud` commands to deploy functions triggered by a Cloud Pub/Sub event.

Deploying a Cloud Function for Cloud Pub/Sub Events Using Cloud Console

Assume you are using a function similar to one used in the previous Cloud Storage example. This time we'll call the function `pub_sub_function_test`.

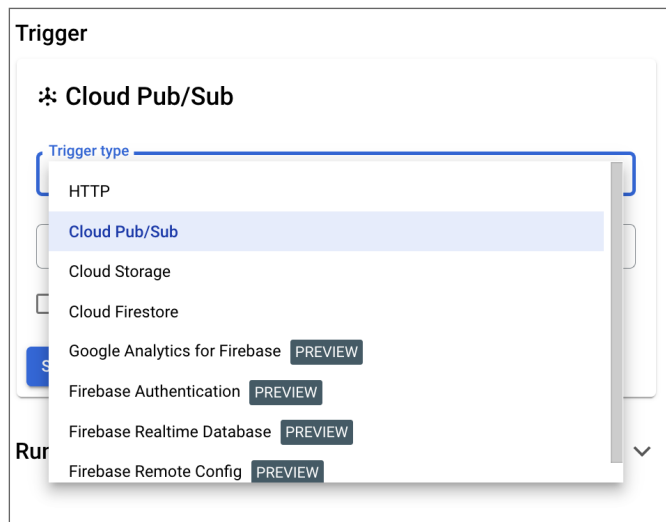
To create a function using Cloud Console, select the Cloud Function options from the vertical menu in the console. In the Cloud Functions console, you may be prompted to enable the Cloud Functions API if it is not already enabled. After the API is enabled, you will have the option to create a new function. When creating a function, you will need to specify several parameters, including the cloud function name, memory allocated, event type, and source code. Here is the source code for `pub_sub_function_test`:

```
def pub_sub_function_test(event_data, event_context):
    import base64
    print('Event ID: {}'.format(event_context.event_id))
    print('Event type: {}'.format(event_context.event_type))
    if 'name' in event_data:
        name = base64.b64decode(event_data['name']).decode('utf-8')
        print('Message name: {}'.format(event_data['name']))
```

This function prints the event ID and event type associated with the message. If the event data has a key-value pair with the key of name, then the function will also print the name in the message. Note that this function has an import statement and uses a function called `base64.b64decode`. This is because messages in Pub/Sub are encoded to allow for binary data in a place where text data is expected, and the `base64.b64decode` function is used to convert it to a more common text encoding called UTF-8.

The code is deployed in the same way as the previous Cloud Storage example with two exceptions. Instead of selecting a Cloud Storage trigger, choose Cloud Pub/Sub from the list of triggers, as shown in Figure 10.5.

FIGURE 10.5 Selecting a trigger from options in Cloud Console



You can also specify the name of the Cloud Pub/Sub topic after specifying this is a Cloud Pub/Sub trigger. If the topic does not exist, it can be created, as shown in Figure 10.6.

Deploying a Cloud Function for Cloud Pub/Sub Events Using *gcloud* Commands

To deploy this function, you use the `gcloud functions deploy` command. When deploying a Cloud Pub/Sub function, you specify the name of the topic that will contain messages that will trigger the function. Like deploying for Cloud Storage, you have to specify the runtime environment you want to use. Here's an example:

```
gcloud functions deploy pub_sub_function_test --runtime python39 --trigger-topic gcp-ace-exam-test-topic
```

FIGURE 10.6 Creating a Pub/Sub topic while creating a Cloud Function

The screenshot shows a 'Create a topic' dialog box. At the top, it says 'Create a topic' and 'A topic forwards messages from publishers to subscribers.' Below this is a text input field for 'Topic ID *' with a question mark icon. Underneath the input field, it says 'Topic name: projects/scenic-energy-335022/topics/'. There are three checkboxes: 'Add a default subscription' (checked), 'Use a schema' (unchecked), and 'Set message retention duration (not free)' (unchecked). Below these is an 'Encryption' section with two radio buttons: 'Google-managed encryption key' (selected) and 'Customer-managed encryption key (CMEK)'. At the bottom right are 'CANCEL' and 'CREATE TOPIC' buttons.

You can delete this function using the `gcloud functions delete` command. Here's an example:

```
gcloud functions delete pub_sub_function_test
```

Summary

In this chapter, we worked with Cloud Functions and saw how to implement and deploy functions. We used examples of functions written in Python, but they could have been written in Node.js or one of several other supported languages as well. Functions can be created using either the Google Cloud Console or the command line. To use Cloud Functions, it is important to understand the relationship between events, triggers, and functions. Events are actions that happen in the cloud. Different services have different types of events. Triggers are how you indicate you want to execute a function when an event occurs. Functions refer to the code that is executed when an event occurs that has a trigger defined for it.

Exam Essentials

Know the relationship between events, triggers, and functions. Events are actions that happen, such as when a file is uploaded to Cloud Storage or a message is written to a

Cloud Pub/Sub topic. Triggers are declarations that an action should be taken when an event occurs. Functions associated with triggers define what actions are taken when an event occurs.

Know when to use Cloud Functions. Cloud Functions is a service that supports single-purpose functions that respond to events in the cloud. Cloud Run is also a serverless computing option, but it is used to deploy multifunction applications, including those that users interact with directly.

Know the runtimes and generations supported in Cloud Functions. Cloud Functions support the following runtimes: Node.js, Python, Go, Java, .NET, Ruby, and PHP. There are two generations of Cloud Functions, the original is known as First Generation and the other Second Generation. Second-Generation Cloud Functions have fewer constraints and more functionality.

Know the parameters for defining a cloud function on a Cloud Storage event. Parameters for Cloud Storage include the following:

- Cloud function name
- Memory allocated for the function
- Trigger
- Event type
- Source of the function code
- Runtime
- Source code
- Name of the function to execute

Know the parameters for defining a Cloud Function on a Cloud Pub/Sub event. Parameters for Pub/Sub include the following:

- Cloud function name
- Memory allocated for the function
- Trigger
- Topic
- Source of the function code
- Runtime
- Source code
- Name of the function to execute

Know the `gcloud` commands for working with Cloud Functions. These include the following:

- `gcloud functions deploy`
- `gcloud functions delete`

Review Questions

You can find the answers in the Appendix.

1. A product manager is proposing a new application that will require several back-end services and three business logic services. Each service will provide a single function, and it will require several of these services to complete a business task. Service execution time is dependent on the size of input and is expected to take up to 90 minutes in some cases. Which GCP product is a good serverless option for running this related service?
 - A. Cloud Functions
 - B. Compute Engine
 - C. Cloud Run
 - D. Cloud Storage
2. You have been asked to deploy a cloud function to reformat image files as soon as they are uploaded to Cloud Storage. You notice after a few hours that about 10 percent of the files are not processed correctly. After reviewing the files that failed, you realize they are all substantially larger than average. What could be the cause of the failures?
 - A. There is a syntax error in the function code.
 - B. The wrong runtime was selected.
 - C. The timeout is too low to allow enough time to process large files.
 - D. There is a permissions error on the Cloud Storage bucket containing the files.
3. When an action occurs in GCP, such as a file being written to Cloud Storage or a message being added to a Cloud Pub/Sub topic, what is that action called?
 - A. An incident
 - B. An event
 - C. A trigger
 - D. A log entry
4. All of the following generate events that can be triggered using Cloud Functions, except which one?
 - A. Cloud Storage
 - B. Cloud Pub/Sub
 - C. SSL
 - D. Firebase
5. Which runtimes are supported in Cloud Functions?
 - A. Node.js and Python only
 - B. Node.js, Python, and Ruby only
 - C. Node.js, Python, .NET, and Go only
 - D. Node.js, Python, Go, Java, .NET, Ruby, and PHP only

6. An HTTP trigger can be invoked by making a request using which of the following?
 - A. GET only
 - B. POST and GET only
 - C. DELETE, POST, and GET
 - D. DELETE, POST, REVERSE, and GET
7. What types of events are available to Cloud Functions working with Cloud Storage?
 - A. Upload or finalize and delete only
 - B. Upload or finalize, delete, and list only
 - C. Upload or finalize, delete, and metadata update only
 - D. Upload or finalize, delete, archive, and metadata update
8. You are tasked with designing a function to execute in Cloud Functions. The function will need more than the default amount of memory and should be applied only when a finalize event occurs after a file is uploaded to Cloud Storage. The function should only apply its logic to files with a standard image file type. Which of the following required features cannot be specified in a parameter and must be implemented in the function code?
 - A. Cloud function name
 - B. Memory allocated for the function
 - C. File type to apply the function to
 - D. Event type
9. How much memory can be allocated to a Cloud Function when using Second-Generation functions?
 - A. 128 MB to 256 MB
 - B. 128 MB to 512 MB
 - C. 128 MB to 1 GB
 - D. 128 MB to 16 GB
10. How long can a Second-Generation event type Cloud Function run by default before timing out?
 - A. 30 seconds
 - B. 1 minute
 - C. 10 minutes
 - D. 20 minutes
11. You want to use the command line to manage Cloud Functions that will be written in Python. What command should you run to ensure your command-line SDK is up to date?
 - A. `gcloud components install`
 - B. `gcloud install components functions`
 - C. `gcloud functions install components`
 - D. `gcloud functions install`

12. You want to create a cloud function to transform audio files into different formats. The audio files will be uploaded into Cloud Storage. You want to start transformations as soon as the files finish uploading. Which trigger would you specify in the Cloud Function to cause it to execute after the file is uploaded?
- A. `google.storage.object.finalize`
 - B. `google.storage.object.upload`
 - C. `google.storage.object.archive`
 - D. `google.storage.object.metadataUpdate`
13. You are defining a Cloud Function to write a record to a database when a file in Cloud Storage is archived. What parameters will you have to set when creating that function?
- A. `runtime` only
 - B. `trigger-resource` only
 - C. `runtime`, `trigger-resource`, `trigger-event` only
 - D. `runtime`, `trigger-resource`, `trigger-event`, `file-type`
14. You'd like to stop using a Cloud Function and delete it from your project. Which command would you use from the command line to delete a Cloud Function?
- A. `gcloud functions delete`
 - B. `gcloud components function delete`
 - C. `gcloud components delete`
 - D. `gcloud delete functions`
15. You have been asked to deploy a Cloud Function to work with Cloud Pub/Sub. As you review the Python code, you notice a reference to a Python function called `base64.b64decode`. Why would a decode function be required in a Pub/Sub cloud function?
- A. It's not required and should not be there.
 - B. Messages in Pub/Sub topics are encoded to allow binary data to be used in places where text data is expected. Messages need to be decoded to access the data in the message.
 - C. It is required to add padding characters to the end of the message to make all messages the same length.
 - D. The decode function maps data from a dictionary data structure to a list data structure.
16. Which of these commands will deploy a Python Cloud Function called `pub_sub_function_test`?
- A. `gcloud functions deploy pub_sub_function_test`
 - B. `gcloud functions deploy pub_sub_function_test --runtime python37`
 - C. `gcloud functions deploy pub_sub_function_test --runtime python37 --trigger-topic gcp-ace-exam-test-topic`
 - D. `gcloud functions deploy pub_sub_function_test --runtime python --trigger-topic gcp-ace-exam-test-topic`

- 17.** When specifying a Cloud Storage Cloud Function, you have to specify an event type, such as `finalize`, `delete`, or `archive`. When specifying a Cloud Pub/Sub Cloud Function, you do not have to specify an event type. Why is this the case?
- A.** Cloud Pub/Sub does not have triggers for event types.
 - B.** Cloud Pub/Sub has triggers on only one event type, when a message is published.
 - C.** Cloud Pub/Sub determines the correct event type by analyzing the function code.
 - D.** The statement in the question is incorrect; you do have to specify an event type with Cloud Pub/Sub functions.
- 18.** Your company has a web application that allows job seekers to upload résumé files. Some files are in Microsoft Word, some are PDFs, and others are text files. You would like to store all résumés as PDFs. How could you do this in a way that minimizes the time between upload and conversion and with minimal amounts of coding?
- A.** Write a Cloud Run application with multiple services to convert all documents to PDF.
 - B.** Implement a Cloud Function on Cloud Storage to execute on a `finalize` event. The function checks the file type, and if it is not PDF, the function calls a PDF converter function and writes the PDF version to the bucket that has the original.
 - C.** Add the names of all files to a Cloud Pub/Sub topic and have a batch job run at regular intervals to convert the original files to PDF.
 - D.** Implement a Cloud Function on Cloud Pub/Sub to execute on a `finalize` event. The function checks the file type, and if it is not PDF, the function calls a PDF converter function and writes the PDF version to the bucket that has the original.
- 19.** What are options for uploading code to a cloud function?
- A.** Inline editor
 - B.** Zip upload
 - C.** Cloud source repository
 - D.** All of the above
- 20.** What type of trigger allows developers to use HTTP POST, GET, and PUT calls to invoke a Cloud Function?
- A.** HTTP
 - B.** Webhook
 - C.** Cloud HTTP
 - D.** None of the above

Chapter 11

Planning Storage in the Cloud

**THIS CHAPTER COVERS THE FOLLOWING
OBJECTIVE OF THE GOOGLE ASSOCIATE
CLOUD ENGINEER CERTIFICATION EXAM:**

- ✓ 2.3 Planning and configuring data storage options





As a cloud engineer, you will have to understand the various storage options provided in Google Cloud Platform (Google Cloud). You will be expected to choose the appropriate option for a given use case while knowing the relative trade-offs, such as having access to SQL for a query language versus the ability to store and query petabytes of data streaming into your database.

Unlike most other chapters in the book, this chapter focuses more on storage concepts than on performing specific tasks in Google Cloud. The material here will help you answer questions about choosing the best storage solution. Chapter 12, “Deploying Storage in Google Cloud,” will provide details on deploying and implementing data solutions.

To choose between storage options, it helps to understand how storage solutions vary by:

- Time to access data
- Data model
- Other features, such as consistency, availability, and support for transactions

This chapter includes guidelines for choosing storage solutions for different kinds of requirements.

Types of Storage Systems

A main consideration when you choose a storage solution is the time in which the data must be accessed. At one extreme, data in an L1 cache on a CPU chip can be accessed in 0.5 nanoseconds (ns). At the other end of the spectrum some services can require hours to return data files. Most storage requirements fall between these extremes.

Nanoseconds, Milliseconds, and Microseconds

Some storage systems operate at speeds as unfamiliar to us as what happens under an electron microscope. One second is an extremely long time when talking about the time it takes to access data in-memory or on disk. We measure time to access, or “latency,” with three units of measure:

- Nanosecond (ns), which is 10^{-9} second
- Microsecond (μ s), which is 10^{-6} second
- Millisecond (ms), which is 10^{-3} second

Note that the number 10^{-3} is in scientific notation and means 0.001 second. Similarly, 10^{-6} is the same as 0.000001, and 10^{-9} is the same as 0.000000001 second.

Another consideration is persistence. How durable is the data stored in a particular system? Caches offer the lowest latency for accessing data, but this type of volatile data exists only as long as power is supplied to memory. Shut down the server and away goes your data. Disk drives have higher durability rates, but they can fail. Redundancy helps here. By making copies of data and storing them on different servers, in different racks, in different zones, and in different regions, you reduce the risk of losing data due to hardware failures.

Google Cloud has several storage services, including the following:

- A managed service for caching based on Redis and Memcached
- Persistent disk storage for use with VMs
- Object storage for shared access to files across resources
- Archival storage for long-term, infrequent access requirements

Cache

A *cache* is an in-memory data store designed to provide applications with submillisecond access to data. Its primary advantage over other storage systems is its low latency. Caches are limited in size by the amount of memory available, and if the machine hosting the cache shuts down, then the contents of the cache are lost. These are significant limitations, but in some use cases, the benefits of fast access to data outweigh the disadvantages.

Memorystore

Google Cloud offers Memorystore, a managed service that provides Redis or Memcached compatible caching. Both Redis and Memcached are widely used open source cache systems. Since Memorystore is protocol-compatible with Redis and Memcached, tools and applications written to work with either should work with Memorystore.

Caches are usually used with an application that cannot tolerate long latencies when retrieving data. For example, an application that reads from a hard disk drive might have to wait 80 times longer than if the data were read from an in-memory cache. Application developers can use caches to store data that is retrieved from a database and then retrieved from the cache instead of the disk the next time that data is needed.

When you use Memorystore, you create instances that run either Redis or Memcached. A Redis instance is configured with up to 300 GB of memory. It can also be configured for high availability, in which case Memorystore creates failover replicas. Memcached instances are configured as a set of up to 20 nodes, and each node can have a maximum of 256 GB. An instance can support up to 5 TB of memory.

Configuring Memorystore

Memorystore caches can be used with applications running in Compute Engine, App Engine, and Kubernetes Engine. Figure 11.1 shows the parameters used to configure Memorystore. You can navigate to this form by choosing Memorystore from the main console menu and then selecting the option to create a Redis instance.

FIGURE 11.1 Configuration parameters for a Memorystore Redis cache

← Create a Redis instance

Cloud Memorystore for Redis is a fully managed Redis service for the Google Cloud Platform. [Learn more](#)

Name your instance

The instance ID is a permanent and unique identifier. The display name is optional and for display purposes only.

Instance ID *

Use lowercase letters, numbers, and hyphens. Start with a letter.

Display name

Tier Selection

Determines availability, cost, and performance.

☐ Basic

Lower cost. Does not provide high availability.

☒ Standard

Supports automatic failover for high availability and up to 5 read replicas for scaling reads. [Learn more.](#)

Capacity

16 GB

Provision upto 300 GB of memory.

Summary

Tier	Standard
Location	us-central1
Estimated maximum throughput (MB/s) ?	1250 / 2000

Cost estimate

Based on instance tier, region, and capacity. [Pricing details](#)

16 GB with 2 read replicas\$805.92/month

To configure a Redis cache in Memorystore, you will need to specify an instance ID, a display name, and a Redis version. You can choose to have a replica in a different zone for high availability by selecting the Standard instance tier. The Basic instance tier does not include a replica but costs less. The configuration of a Memcached is similar but also has parameters for configuring a cluster of nodes.

You will need to specify a region and zone along with the amount of memory you want to dedicate to your cache. The cache can be 1 GB to 300 GB in size. The Redis instance will be accessible from the default network unless you specify a different network. (See Chapter 14, “Networking in the Cloud: Virtual Private Clouds and Virtual Private Networks,” and Chapter 15, “Networking in the Cloud: DNS, Load Balancing, Google Private Access, and IP Addressing,” for more on networks in Google Cloud.) The advanced options for Memorystore allow you to assign labels and define an IP range from which the IP address will be assigned.

The configuration of a Memcached is similar.

Persistent Storage

In Google Cloud, persistent disks provide durable block storage. Persistent disks can be attached to VMs in Google Compute Engine (GCE) and Google Kubernetes Engine (GKE). Since persistent disks are block storage devices, you can create filesystems on these devices. Persistent disks are not directly attached to physical servers hosting your VMs but are network accessible. VMs can have locally attached solid-state drives (SSDs), but the data on those drives is lost when the VM is terminated. The data on persistent disks continues to exist after VMs are shut down and terminated. Persistent disks exist independently of virtual machines; local attached SSDs do not.

Features of Persistent Disks

Persistent disks are available in SSD and hard disk drive (HDD) configurations. SSDs are used when high throughput is important. SSDs provide consistent performance for both random access and sequential access patterns. HDDs have longer latencies but cost less, so HDDs are a good option when storing large amounts of data and performing batch operations that are less sensitive to disk latency than interactive applications. Persistent disks are available in the following types:

- Zonal standard persistent disks, which provide efficient and reliable block storage within a zone using standard hard disk drives
- Regional standard persistent disks, which are like zonal standard persistent disks in performance but also provide for synchronous replication across two zones within a region
- Zonal balanced persistent disks, which are cost effective and reliable storage using SSDs
- Regional balanced persistent disks, which are like zonal balanced persistent disks in performance but also provide for synchronous replication across two zones within a region
- Zonal SSD persistent disks, which provide fast and reliable block storage within a zone
- Regional SSD persistent disks, which are like zonal SSD persistent disks in performance but also provide for synchronous replication across two zones within a region
- Zonal extreme persistent disks, which offer the highest performance block storage of persistent disks and use SSDs

In addition to persistent disks, Google Cloud offers Local SSDs, which are high-performance local block storage but have no redundancy. Persistent disks have a maximum capacity of 64 TB whereas Local SSDs have a fixed capacity of 375 GB.

Persistent disks can be mounted on multiple VMs to provide multireader storage. Snapshots of disks can be created in minutes, so additional copies of data on a disk can be distributed for use by other VMs. If a disk created from a snapshot is mounted to a single VM, it can support both read and write operations.

The size of persistent disks can be increased while mounted to a VM. If you do resize a disk, you may need to perform operating system commands to make that additional space accessible to the filesystem. Both SSD and HDD disks can be up to 64 TB.

Persistent disks automatically encrypt data on the disk.

When planning your storage options, you should also consider whether you want your disks to be zonal or regional. Zonal disks store data across multiple physical drives in a single zone. If the zone becomes inaccessible, you will lose access to your disks. Alternatively, you could use regional persistent disks, which replicate data blocks across two zones within a region but are more expensive than zonal storage.

Configuring Persistent Disks

You can create and configure persistent disks from the console by navigating to Compute Engine and selecting Disks. From the Disk page, click Create A Disk to display a form like that in Figure 11.2.

You will need to provide a name for the disk, but the description is optional. There are two types of disk: standard and SSD persistent disk. For higher availability, you can have a replica created within the region. You will need to specify a region and zone. Labels are optional but recommended to help keep track of each disk's purpose.

Persistent disks can be created blank or from an image or snapshot. Use the image option if you want to create a persistent boot disk. Use a snapshot if you want to create a replica of another disk.

When you store data at rest in Google Cloud, it is encrypted by default. When creating a disk, you can choose to have Google manage encryption keys, in which case no additional configuration is required. You could use Google Cloud's Cloud Key Management Service to manage keys yourself and store them in Google Cloud's key repository. Choose the customer-managed encryption key (CMEK) option for this. You will need to specify the name of a key you have created in Cloud Key Management Service. If you create and manage keys using another key management system, then select customer-supplied encryption key (CSEK). You will have to enter the key into the form if you choose the customer-supplied key option.

Object Storage

Caches are used for storing relatively small amounts of data that must be accessible with submillisecond latency. Persistent storage devices can store up to 64 TB on a single disk and provide up to hundreds of IOPS for read and write operations. When you need to store large volumes of data—that is, up to exabytes—and share it widely, object storage is a good option. Google Cloud's object storage is Cloud Storage.

FIGURE 11.2 Form to create a persistent disk

←

Create a disk

Name *

disk-1

?

Name is permanent

Description

Location

☒ Single zone

☐ Regional

Create a failover replica in the same region for high availability. [Learn more](#)

Region *

us-central1 (Iowa)

▼

?

Zone *

us-central1-a

▼

?

Source

Create a blank disk, apply a bootable disk image, or restore a snapshot of another disk in this project.

Disk source type *

Blank disk

▼

Disk settings

Disk type *

Balanced persistent disk

▼

?

COMPARE DISK TYPES

Size *

100

GB

?

Provision between 10 and 65,536 GB

Snapshot schedule (Recommended)

Use snapshot schedules to automate disk backups. [Learn more](#)

☒ Enable snapshot schedule

Select or create a snapshot schedule *

default-schedule-1

▼

Every day, starts between 8:00 AM and 9:00 AM

Encryption

Data is encrypted automatically. Select an encryption key management solution.

☒ Google-managed encryption key

No configuration required

☐ Customer-managed encryption key (CMEK)

Manage via Google Cloud Key Management Service

☐ Customer-supplied encryption key (CSEK)

Manage outside of Google Cloud

CREATE

CANCEL

EQUIVALENT COMMAND LINE

▼

Features of Cloud Storage

Cloud Storage is an object storage system, which means files that are stored in the system are treated as atomic units—that is, you cannot operate on part of the file, such as reading only a section of the file. You can perform operations on an object, like creating or deleting it, but Cloud Storage does not provide functionality to manipulate subcomponents of a file. For example, there is no Cloud Storage command for overwriting a section of the file. Also, Cloud Storage does not support concurrency and locking. If multiple clients are writing to a file, then the last data written to the file is stored and persisted.

Cloud Storage is well suited for storing large volumes of data without requiring any consistent data structure. You can store different types of data in a bucket, which is the logical unit of organization in Cloud Storage. Buckets are resources within a project. It is important to remember that buckets share a global namespace, so each bucket name must be globally unique. We shouldn't be surprised if we can't name a bucket "mytestbucket," but it's not too difficult to find a unique filename.

It is important to remember that object storage does not provide a filesystem. Buckets are analogous to directories in that they help organize objects into groups, but buckets are not true directories that support features such as subdirectories. Google does support an open source project called Cloud Storage Fuse, which provides a way to mount a bucket as a filesystem on Linux and Mac operating systems. Using Cloud Storage Fuse, you can download and upload files to buckets using filesystem commands, but it does not provide full filesystem functionality. Cloud Storage Fuse has the same limitations as Cloud Storage. Its purpose is to make it more convenient to move data in and out of buckets when working in a Linux or Mac filesystem.

Cloud Storage provides four different classes of object storage: standard, nearline, coldline, and archive. For each class of storage, we can choose to store the data in a single region, dual regions, or multi-regions.

Storage Classes

Standard storage is the best option for frequently used data, which is sometimes referred to as "hot data" or data that is being stored for short periods of time. Dual region replication can increase availability over single region storage. Multi-region storage is a good option when the data will be read from multiple regions and you want to reduce latency to accessing data from multiple regions. Dual and multi-region Standard storage have 99.95 percent availability, while single region has 99.9 percent availability.

For infrequently accessed data, the nearline and coldline storage classes are good options. Nearline storage is designed for use cases in which you expect to access files less than once per month. Coldline storage is designed, and priced, for files expected to be accessed once per 90 days or less.

Nearline storage has a 99.95 percent typical monthly availability in multiregional locations and a 99.9 percent typical availability in regional locations. The SLAs for nearline are 99.9 percent in multiregional locations and 99.0 percent in regional locations. These lower SLAs come with a significantly lower cost per gigabyte stored, but before you start moving

all your regional and multiregional data to nearline to save on costs, you should know that Google adds a data retrieval charge to nearline and coldline storage. There is also a minimum 30-day storage duration for nearline storage.

Coldline storage has a 99.95 percent typical monthly availability in multiregional locations and a 99.9 percent typical availability in regional locations. The SLAs are 99.9 percent for multiregional locations and 99.0 percent for regional locations. Coldline also has a lower cost per gigabyte than nearline storage. Remember, that is only the storage charge. Like nearline storage, coldline storage has access charges. Google expects data in coldline storage to be accessed once per 90 days or less and have at least a 90-day minimum storage.

Archive storage is designed for long-term storage for archiving, disaster recovery, and other use cases where the data will be accessed less than once per year and will be stored for at least 365 days. The SLA for Archive storage is 99.9 percent for multi-region and dual region and 99.0 percent for region location types.

It is more important to understand the relative cost relationships than the current prices. Prices can change, but the costs of each class relative to other classes of storage are more likely to stay the same.

Regional, Dual Regional, and Multi-Regional Storage

When you create a bucket, you specify a location to create the bucket. The bucket and its contents are stored in this location. You can store your data in a single region, dual regions, or multiple regions. A region is a specific geographic location, such as Northern Virginia, Paris, and Mumbai. A dual-region is a pair of regions. A multi-region is a large geographic area, such as the United States, European Union, and Asia. The availability SLA for regional storage is 99.9 percent while dual-region and multi-region have a 99.95 percent availability SLQ. Regional buckets are redundant across zones.

Multiregional buckets are used when content needs to be stored in multiple regions to ensure acceptable times to access content. It also provides redundancy in case of zone-level failures. These benefits come with a higher cost, however. (You are not likely to be asked about specific prices on the Associate Cloud Engineer exam, but you should know the relative costs so that you can identify the lowest-cost solution that meets a set of requirements.)

Both regional and multiregional storage are used for generally used data. If you have an application where users download and access files often, such as more than once per month, then it is most cost-effective to choose regional or multiregional. You choose between regional and multiregional based on the location of your users. If users are globally dispersed and require access to synchronized data, then multiregional may provide better performance and availability.



A note on terminology: Google sometimes uses the term *georedundant*. Georedundant data is stored in at least two locations that are at least 100 miles apart. If your data is in multiregional locations, then it is georedundant.

Versioning and Object Life Cycle Management

Buckets in Cloud Storage can be configured to retain versions of objects when they are changed. When versioning is enabled on a bucket, a copy of an object is archived each time the object is overwritten or when it is deleted. The latest version of the object is known as the live version. Versioning is useful when you need to keep a history of changes to an object or want to mitigate the risk of accidentally deleting an object.

Cloud Storage also provides life cycle management policies to automatically change an object's storage class or delete the object after a specified period. A life cycle policy, sometimes called a configuration, is a set of rules. The rules include a condition and an action. If the condition is true, then the action is executed. Life cycle management policies are applied to buckets and affect all objects in the bucket.

Conditions are often based on age. Once an object reaches a certain age, it can be deleted or moved to a lower-cost storage class. In addition to age, conditions can check the number of versions, whether the version is live, whether the object was created before a specific date, and whether the object is in a particular storage class.

You can delete an object or change its storage class. Both unversioned and versioned objects can be deleted. If the live version of a file is deleted, then instead of actually deleting it, the object is archived. If an archived version of an object is deleted, the object is permanently deleted.

You can also change the storage class of an object using life cycle management. There are restrictions on which classes can be assigned. Standard storage objects can be changed to nearline, coldline, or archive. Nearline can be changed only to coldline or archive, whereas coldline can be changed to archive.

Configuring Cloud Storage

You can create buckets in Cloud Storage using the console. From the main menu, navigate to Storage and select Create Bucket. This will display a form similar to Figure 11.3.

When creating a bucket, you need to supply some basic information, including a bucket name and storage class. You can optionally add labels and choose either Google-managed keys or customer-managed keys for encryption. You can also set a retention policy to prevent changes to files or deletion of files before the time you specify.

Once you have created a bucket, you define a life cycle policy. Choose Lifecycle from the horizontal menu to display the form shown in Figure 11.4.

Notice that the Lifecycle column indicates whether a life cycle configuration is enabled. Choose a bucket to create or modify a life cycle and click None or Enabled in the Lifecycle column. This will display the form shown in Figure 11.5.

FIGURE 11.3 Form to create a storage bucket from the console. Advanced options are displayed.

Create a bucket

- Name your bucket**
 Pick a globally unique, permanent name. [Naming guidelines](#)

 Tip: Don't include any sensitive information
 ✓ LABELS (OPTIONAL)
- Choose where to store your data**
 Location: us (multiple regions in United States)
 Location type: Multi-region
- Choose a default storage class for your data**
 Default storage class: Standard
- Choose how to control access to objects**
 Public access prevention: Off
 Access control: Uniform
- Choose how to protect object data**
 Protection tools: None
 Data encryption: Google-managed key

Good to know

Location pricing
 Storage rates vary depending on the storage class of your data and location of your bucket. [Pricing details](#)

Current configuration: Multi-region / Standard

Item	Cost
us (multiple regions in United States)	\$0.026 per GB-month

ESTIMATE YOUR MONTHLY COST

When you add a rule, you need to specify the object condition and the action. Condition options are Age, Creation Data, Storage Class, Newer Versions, and Live State. Live State applies to version objects, and you can set your condition to apply to either live or archived versions of an object. The action can be to set the storage class to either nearline, coldline, or archive.

Let's look at an example policy. From the Browser section of Cloud Storage in the console, you can see a list of buckets, as shown in Figure 11.6.

FIGURE 11.4 The list of buckets includes a link to define or modify life cycle policies.

Location

us-west1 (Oregon)

Storage class

Standard

Public access

Not public

Protection

None

OBJECTS

CONFIGURATION

PERMISSIONS

PROTECTION

LIFECYCLE

OBSERVABILITY

NEW

Lifecycle rules let you apply actions to a bucket's objects when certain conditions are met — for example, switching objects to colder storage classes when they reach or pass a certain age. [Learn more](#)

If an object meets the conditions for multiple rules:

- Deletion takes precedence over a change in storage class.
- Changing objects to colder storage classes takes precedence over changing to warmer ones (ex. objects will switch to the Archive storage class instead of Coldline if there are rules for both).

Rules

ADD A RULE

DELETE ALL

Action	Object condition	Works with
You haven't added any lifecycle rules to this bucket.		

Storage Types When Planning a Storage Solution

When planning a storage solution, a factor you should consider is the time required to access data. Caches, like Memorystore, offer the fastest access time but are limited to the amount of memory available. Caches are volatile; when the server shuts down, the contents of the cache are lost. You should save the contents of the cache to persistent storage at regular intervals to enable recovery to the point in time when the contents of the cache were last saved.

Persistent storage is used for block storage devices, such as disks attached to VMs. Google Cloud offers SSD and HDD drives. SSDs provide faster performance but cost more. HDDs are used when large volumes of data need to be stored in a filesystem but users of the data do not need the fastest access possible.

Object storage is used for storing large volumes of data for extended periods of time. Cloud Storage has both regional and multiregional storage classes and supports life cycle management and versioning.

In addition to choosing an underlying storage system, you will have to consider how data is stored and accessed. For this, it is important to understand the data models available and when to use them.

FIGURE 11.5 When creating a life cycle policy, click the Add Rule option, which is in the lower horizontal menu. to define a rule.

← Add object lifecycle rule

After you add or edit a rule, it may take up to 24 hours to take effect.

- **Select an action**
 - ☒ Set storage class to Nearline
Best for backups and data accessed less than once a month
 - i** Coldline and Archive objects will not be changed to Nearline.

Early deletion fees could apply if the object being modified hasn't met the minimum duration requirements for its current class. [Learn more](#)
 - ☐ Set storage class to Coldline
Best for disaster recovery and data accessed less than once a quarter
 - ☐ Set storage class to Archive
Best for long-term digital preservation of data accessed less than once a year
 - ☐ Delete object

CONTINUE

- **Select object conditions**

CREATE CANCEL

FIGURE 11.6 Listing of buckets in Cloud Storage Browser

Filter Filter buckets						?	
<input type="checkbox"/> Name ↑	Created	Location type	Location	Default storage class	?		
<input type="checkbox"/> dataflow-staging-us-west1-38894734...	Oct 23, 2022, 11:06:45 AM	Region	us-west1	Standard			⋮
<input type="checkbox"/> gcf-sources-388947348090-us-central1	Nov 19, 2022, 9:52:45 AM	Region	us-central1	Standard			⋮
<input type="checkbox"/> slg-cloud-storage-2	Oct 26, 2022, 6:55:12 AM	Region	us-west1	Standard			⋮

Storage Data Models

There are four broad categories of data models available in Google Cloud: object, relational, analytical, and NoSQL.

Object: Cloud Storage

The object storage data model treats files as atomic objects. You cannot use object storage commands to read blocks of data or overwrite parts of the object. If you need to update an object, you must copy it to a server, make the change, and then copy the updated version back to the object storage system.

Object storage is used when you need to store large volumes of data and do not need fine-grained access to data within an object while it is in the object store. This data model is well suited for archived data, machine learning training data, and old Internet of Things (IoT) data that needs to be saved but is no longer actively analyzed.

Relational: Cloud SQL and Cloud Spanner

Relational databases have been the primary data store for enterprises for decades. Relational databases support frequent queries and updates to data. They are used when it is important for users to have a consistent view of data. For example, if two users are reading data from a relational table at the same time, they will see the same data. This is not always the case with databases that may have inconsistencies between replicas of data, such as some NoSQL databases.

Relational databases, like Cloud SQL and Cloud Spanner, support database transactions. A transaction is a set of operations that is guaranteed to succeed or fail in its entirety—there is no chance that some operations are executed and others are not. For example, when a customer purchases a product, the count of the number of products available is decremented in the inventory table, and a record is added to a customer-purchased products table. With transactions, if the database fails after updating inventory but before updating the customer-purchased products table, the database will roll back the partially executed transaction when the database restarts.

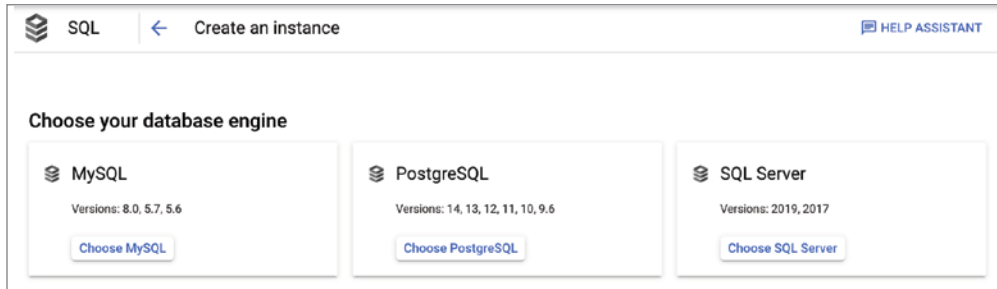
Cloud SQL and Cloud Spanner are used when data is structured and modeled for relational databases. Cloud SQL is a managed database service that provides MySQL, SQL Server, and PostgreSQL databases. Cloud SQL is used for databases that do not need to scale horizontally—that is, by adding additional servers to a cluster. Cloud SQL databases scale vertically—that is, by running on servers with more memory and more CPU. Cloud Spanner is used when you have extremely large volumes of relational data or data that needs to be globally distributed while ensuring consistency and transaction integrity across all servers.

Large enterprises often use Cloud Spanner for applications like global supply chains and financial services applications, whereas Cloud SQL is often used for web applications, and e-commerce applications.

Configuring Cloud SQL

You can create a Cloud SQL instance by navigating to Cloud SQL in the main menu of the console and selecting Create Instance. You will be prompted to choose a MySQL, PostgreSQL, or SQL Server instance, as shown in Figure 11.7.

FIGURE 11.7 Cloud SQL provides MySQL, PostgreSQL, and SQL Server instances.



To configure a MySQL instance, you will need to specify a name, root password, region, and zone. The configuration options include the following:

- MySQL version.
- Connectivity, where you can specify whether to use a public or a private IP address.
- Machine type. The default is a db-n1-standard-1 with 1 vCPU and 3.75 GB of memory.
- Automatic backups.
- Failover replicas.
- Database flags. These are specific to MySQL and include the ability to set a database read-only flag and set the query cache size.
- A maintenance time window.
- Labels.

Figure 11.8 shows the configuration form for MySQL instances, Figure 11.9 shows the configuration for SQL Server instances, and Figure 11.10 shows the configuration for PostgreSQL instances.

Configuring Cloud Spanner

If you need to create a global, consistent database with support for transactions, you should consider Cloud Spanner. Given the advanced nature of Spanner, its configuration is surprisingly simple. In the console, navigate to Cloud Spanner and select Create Instance to display the form in Figure 11.11.

FIGURE 11.8 Configuration form for a MySQL instance

← Create a MySQL instance

Instance info

Instance ID *

Use lowercase letters, numbers, and hyphens. Start with a letter.

Password *

Set a password for the root user. [Learn more](#)

☐ No password

Database version *

MySQL 8.0

Choose region and zonal availability

For better performance, keep your data close to the services that need it. Region is permanent, while zone can be changed any time.

Region

us-central1 (Iowa)

Zonal availability

☐ Single zone

In case of outage, no failover. Not recommended for production.

☒ Multiple zones (Highly available)

Automatic failover to another zone within your selected region. Recommended for production instances. Increases cost.

▼ SPECIFY ZONES

Customize your instance

You can also customize instance configurations later

▼ SHOW CONFIGURATION OPTIONS

CREATE INSTANCE

CANCEL

Summary

Region	us-central1 (Iowa)
DB Version	MySQL 8.0
vCPUs	4 vCPU
Memory	26 GB
Storage	100 GB
Network throughput (MB/s)	1,000 of 2,000
Disk throughput (MB/s)	Read: 48.0 of 240.0 Write: 48.0 of 240.0
IOPS	Read: 3,000 of 15,000 Write: 3,000 of 15,000
Connections	Public IP
Backup	Automated
Availability	Multiple zones (Highly available)
Point-in-time recovery	Enabled

You need to provide an instance name, instance ID, and number of nodes. You will also have to choose either a regional or multiregional configuration to determine where nodes and data are located. This will determine cost and replication storage location. If you select Regional, you will choose from the list of available regions, such as us-west1, asia-east1, and europe-north1.

Analytical: BigQuery

BigQuery is a service designed for a data warehouse and analytic applications. BigQuery is designed to store petabytes of data. BigQuery works with large numbers of rows and columns of data and is not suitable for transaction-oriented applications, such as e-commerce or support for interactive web applications.

FIGURE 11.9 Configuration form for a SQL Server instance

← Create a SQL Server instance

Instance info

Instance ID *

Use lowercase letters, numbers, and hyphens. Start with a letter.

Password *

🔍

GENERATE

Your default service admin username is "sqlserver" [Learn more](#)

Database version *

SQL Server 2019 Standard

Choose region and zonal availability

For better performance, keep your data close to the services that need it. Region is permanent, while zone can be changed any time.

Region

us-central1 (Iowa)

Zonal availability

☐ Single zone

In case of outage, no failover. Not recommended for production.

☒ Multiple zones (Highly available)

Automatic failover to another zone within your selected region. Recommended for production instances. Increases cost.

[SPECIFY ZONES](#)

Customize your instance

You can also customize instance configurations later

[SHOW CONFIGURATION OPTIONS](#)

CREATE INSTANCE

CANCEL

Summary

Region	us-central1 (Iowa)
DB Version	SQL Server 2019 Standard
vCPUs	4 vCPU
Memory	26 GB
Storage	100 GB
Network throughput (MB/s)	1,000 of 2,000
Disk throughput (MB/s)	Read: 48.0 of 240.0 Write: 48.0 of 240.0
IOPS	Read: 3,000 of 15,000 Write: 3,000 of 15,000
Connections	Public IP
Backup	Automated
Availability	Multiple zones (Highly available)

Configuring BigQuery

BigQuery is a serverless analytics service, which provides storage plus query, statistical, and machine learning analysis tools. BigQuery does not require you to configure instances. Instead, when you first navigate to BigQuery from the console menu, you will see the form shown in Figure 11.12.

The first task for using BigQuery is to create a dataset to hold data. You do this by clicking Create Dataset to display the form shown in Figure 11.13.

When creating a dataset, you will have to specify a name and select a region in which to store it. Not all regions support BigQuery. Currently you have a choice of most locations across the United States, Europe, and Asia.

FIGURE 11.10 Configuration form for a PostgreSQL instance

Create a PostgreSQL instance

Instance info

Instance ID *

Use lowercase letters, numbers, and hyphens. Start with a letter.

Password *

Set a password for the default admin user "postgres". [Learn more](#)

✓

PASSWORD POLICY

Database version *

PostgreSQL 14

Choose region and zonal availability

For better performance, keep your data close to the services that need it. Region is permanent, while zone can be changed any time.

Region

us-central1 (Iowa)

Zonal availability

☐ Single zone

In case of outage, no failover. Not recommended for production.

☒ Multiple zones (Highly available)

Automatic failover to another zone within your selected region. Recommended for production instances. Increases cost.

✓

SPECIFY ZONES

Customize your instance

You can also customize instance configurations later

✓

SHOW CONFIGURATION OPTIONS

CREATE INSTANCE

CANCEL

Summary

Region	us-central1 (Iowa)
DB Version	PostgreSQL 14
vCPUs	4 vCPU
Memory	26 GB
Storage	100 GB
Network throughput (MB/s)	1,000 of 2,000
Disk throughput (MB/s)	Read: 48.0 of 240.0 Write: 48.0 of 240.0
IOPS	Read: 3,000 of 15,000 Write: 3,000 of 15,000
Connections	Public IP
Backup	Automated
Availability	Multiple zones (Highly available)
Point-in-time recovery	Enabled

In Chapter 12, we will discuss how to load and query data in BigQuery and other Google Cloud databases.

NoSQL: Cloud Firestore and Bigtable

NoSQL databases do not use the relational model and do not require a fixed structure or schema. Database schemas define what kinds of attributes can be stored. When no fixed schema is required, developers have the option to store different attributes in different records. Google Cloud has a document database called Cloud Firestore and a wide column database called Bigtable.

FIGURE 11.11 The Cloud Spanner configuration form in Cloud Console

Create an instance

Name your instance

An instance has both a name and an ID. The name is for display purposes only. The ID is a permanent and unique identifier.

Instance name *

Name must be 4-30 characters long

Instance ID *

Lowercase letters, numbers, hyphens allowed

Choose a configuration

Determines where your nodes and data are located. Affects cost, performance, and replication. A multi-region configuration will select the default leader region for your leader replicas. You can change your leader region at any time with a DDL statement. [Learn more](#)

COMPARE REGION CONFIGURATIONS

☒ Regional

☐ Multi-region

Select a configuration

Allocate compute capacity

Your compute capacity determines the amount of data throughput, queries per second (QPS), and storage limits in your instance. One node equals 1,000 processing units. Affects billing.

Unit *

Processing units

Quantity *

1000

Integers only. Enter in increments of 100 up to 1,000, followed by increments of 1,000.

COMPUTE CAPACITY GUIDANCE

CREATE

CANCEL

Summary

Storage cost depends on GB stored per month. Compute cost refers to the hourly charge of nodes or processing units in your instance.

Configuration	---
Replicas	---
Availability	---
Compute cost	---
Storage cost	---
Maximum storage capacity	---

FIGURE 11.12 BigQuery user interface for creating and querying data

Explorer

+ ADD DATA

Q Type to search

Viewing all resources. [Show starred resources only.](#)

scenic-energy-335022

☆

Editor

RUN

SAVE

SHARE

SCHEDULE

Type a query to get started

1

FIGURE 11.13 Form to create a data set in BigQuery

Create dataset

Project ID
scenic-energy-335022 [CHANGE](#)

Dataset ID *
Letters, numbers, and underscores allowed

Data location ▼ ⓘ

Default table expiration

☐ Enable table expiration ⓘ

Default maximum table age Days

Advanced options ▼

[CREATE DATASET](#) [CANCEL](#)

Firestore Features

Firestore is a document database. That does not mean it is used to store documents like spreadsheets or text files but that the data in the database is organized into a structure called a document. Documents are made up of sets of key-value pairs. A simple example is as follows:

```
{
  book : "ACE Study Guide",
    chapter: 11,
  length: 20,
  topic: "storage"
}
```

This example describes the characteristics of a chapter in a book. There are four keys or properties in this example: `book`, `chapter`, `length`, and `topic`. This set of key-value pairs is called an *entity* in Firestore terminology. Entities often have properties in common, but since Firestore is a schema-less database, there is no requirement that all entities have the same set of properties. Here's an example:

```
{
  book : "ACE Study Guide",
    Chapter: 11,
```

```
topic: "computing",
number_of_figures: 8
}
```

Firestore is a managed database, so users of the service do not need to manage servers or install database software. Firestore automatically partitions data and scales up or down as demand warrants.

Firestore is used for nonanalytic, nonrelational storage needs. It is a good choice for product catalogs, which have many types of products with varying characteristics or properties. It is also a good choice for storing user profiles associated with an application.

Firestore has some features in common with relational databases, such as support for transactions and indexes to improve query performance. The main difference is that Firestore does not require a fixed schema or structure and does not support relational operations, such as joining tables, or computing aggregates, such as sums and counts.

Cloud Firestore is the latest generation of document databases in Google Cloud. Cloud Datastore preceded Cloud Firestore as a document database.

Configuring Firestore

Firestore, like BigQuery, is a serverless database service that does not require you to specify node configurations. Instead, you can work from the console to add entities to the database. Figure 11.14 shows the initial form that appears when you first navigate to Firestore in Cloud Console. The first thing you must do when using Firestore is choose between Native mode, which automatically scales to millions of clients, or Datastore mode, which automatically scales to millions of writes per second.

Once you have chosen a mode, you choose where to store your data (see Figure 11.15). You have the option of using multiregional storage or regional storage.

Once you have configured Cloud Firestore for your project in Datastore mode, you can create entities. When creating an entity, you specify a namespace, which is a way to group entities much like schemas group tables in a relational database. You will need to specify a kind, which is analogous to a table in a relational database. Each entity requires a key, which can be an autogenerated numeric key or a custom-defined key.

Next, you will add one or more properties that have names, types, and values. Types include string, date and time, Boolean, and other structured types like arrays.

Firestore Native Mode, provides a different data model based on documents and collections. Documents are collections of key-value pairs and collections are sets of documents.

Additional details on loading and querying data in Firestore are in Chapter 12.


Bigtable Features

Bigtable is another NoSQL database, but unlike Firestore, it is a wide-column database, not a document database. Wide-column databases, as the name implies, store tables that can have a large number of columns. Not all rows need to use all columns, so in that way it is like Firestore—neither requires a fixed schema to structure the data.

FIGURE 11.14 The Firestore user interface allows you to choose between Native and Datastore modes.

1 Select a Cloud Firestore mode — 2 Choose where to store your data

Cloud Firestore is the next generation of Cloud Datastore. You can use Cloud Firestore in either Native mode or Datastore mode, each with distinct system behavior optimized for different types of projects. [Pricing](#) for both modes is based on location, stored data, operations, and network egress with a daily free quota for each. [Learn more about choosing a mode](#)

 The mode you select here will be permanent for this project

	Native mode	Datastore mode
	Enable all of Cloud Firestore's features, with offline support and real-time synchronization. SELECT NATIVE MODE	Leverage Cloud Datastore's system behavior on top of Cloud Firestore's powerful storage layer. SELECT DATASTORE MODE
API	Firestore	Datastore
Scalability	Automatically scales to millions of concurrent clients	Automatically scales to millions of writes per second
App engine support	Not supported in the App Engine standard Python 2.7 and PHP 5.5 runtimes	All runtimes
Max writes per second	10,000	No limit
Real-time updates	✓	✗
Mobile/web client libraries with offline data persistence	✓	✗

Bigtable is designed for petabyte-scale databases. Both operational databases, like storing IoT data, and analytic processing, like data science applications, can effectively use Bigtable. This database is designed to provide consistent, low-millisecond latency. Bigtable runs in clusters and scales horizontally.

Bigtable is designed for applications with high data volumes and a high-velocity ingest of data. Time series, IoT, and financial applications all fall into this category.

FIGURE 11.15 Choosing a storage location

Get started

✓ Select a Cloud Firestore mode — 2 **Choose where to store your data**

You selected Cloud Firestore in Native mode. Now choose a database location.

The location of your database affects its cost, availability, and durability. Choose a regional location (lower write latency, lower cost) or a multi-region location (higher availability, higher cost). [Learn more](#)

⚠ Choose carefully: your location selection is permanent and will also apply to this project's App Engine app

Select a location

Multi-region (99.999% SLA)

eur3 (Europe)

nam5 (United States)

Regional (99.99% SLA)

asia-east1 (Taiwan)

asia-east2 (Hong Kong)

asia-northeast1 (Tokyo)

Configuring Bigtable

From Cloud Console, navigate to Bigtable and click Create Instance to open the form shown in Figure 11.16.

In this form, you will need to provide an instance name and an instance ID. Next, choose either Production or Development mode. Production clusters have a minimum of three nodes and provide for high availability. Development mode uses low-cost instances without replication or high availability. You will also need to choose either SSD or HDD for persistent disks used by the database.

Bigtable can support multiple clusters. For each cluster you will need to specify a cluster ID, a region and zone location, and the number of nodes in the cluster. The cluster can be replicated to improve availability.

In Chapter 12, we will describe how to load and query data in Bigtable.

FIGURE 11.16 Configuration form for Bigtable

← Create an instance

A Bigtable instance is a container for your clusters. [Learn more](#)

\$468 per month (estimated)

That's about \$0.65 an hour with 0 GB stored.

✓ SHOW DETAILS

1 Name your instance

Instance name *

For display purposes only

Instance ID *

ID is permanent

CONTINUE

2 Select your storage type

3 Configure your first cluster

✓ SHOW ADVANCED OPTIONS

CREATE CANCEL



Real World Scenario

The Need for Multiple Databases

Healthcare organizations and medical facilities store and manage a wide range of data about patients, their treatments, and the outcomes. A patient's medical records include demographic information, such as name, address, age, and so on. Medical records also store detailed information about medical conditions and diagnoses as well as treatment, such as drugs prescribed and procedures performed. This kind of data is highly structured. Transaction support and strong consistency are required. Relational databases, like Cloud SQL, are a good solution for this kind of application.

The medical data stored in transactional, relational databases is valuable for analyzing patterns in treatments and recovery. For example, data scientists could use medical records to identify patterns associated with readmission to the hospital. However, transactional relational databases are not suited for analytics. A better option is to use BigQuery and build a data warehouse with data structured in ways that make it easier to analyze data. Data from the transactional system is extracted, transformed, and loaded into a BigQuery dataset.

Choosing a Storage Solution: Guidelines to Consider

Google Cloud offers multiple storage solutions. As a cloud engineer, you may have to help plan and implement storage solutions for a wide range of applications. The different storage solutions lend themselves to different use cases, and in many enterprise applications, you will find that you need two or more storage products to support the full range of application requirements. Here are several factors to keep in mind when choosing storage solutions:

Read and Write Patterns Some applications, such as accounting and retail sales applications, read and write data frequently. There are also frequent updates in these applications. They are best served by a storage solution such as Cloud SQL if the data is structured; however, if you need a global database that supports relational read/write operations, then Cloud Spanner is a better choice. If you are writing data at consistently high rates and in large volumes, consider Bigtable. If you are writing files and then downloading them in their entirety, Cloud Storage is a good option.

Consistency Consistency ensures that a user reading data from the database will get the same data no matter which server in a cluster responds to the request. If you need strong consistency, which is always reading the latest data, then Cloud SQL and Cloud Spanner are good options. Firestore can be configured for strong consistency, but I/O operations will take longer than if a less strict consistency configuration is used. Firestore is a good option if your data is unstructured; otherwise, consider one of the relational databases. NoSQL databases offer at least eventual consistency, which means some replicas may not be in sync for a short period of time. During those periods it is possible to read stale data. If your application can tolerate that, then you may find that less strict consistency requirements can lead to faster read and write operations.

Transaction Support If you need to perform atomic transactions in your application, use a database that supports them. You may be able to implement transaction support in your application, but that code can be difficult to develop and maintain. The relational databases, Cloud SQL and Spanner, and Firestore provide transaction support.

Cost The cost of using a particular storage system will depend on the amount of data stored, the amount of data retrieved or scanned, and per-unit charges of the storage system. If you are using a storage service in which you provision VMs, you will have to account for that cost as well.

Latency Latency is the time between the start of an operation, like a request to read a row of data from a database, to the time it completes. Bigtable provides consistently low-millisecond operations. Spanner can have longer latencies, but with those longer latencies you get a globally consistent, scalable database.

In general, choosing a data store is about making trade-offs. In an ideal world, we could have a low-cost, globally scalable, low-latency, strongly consistent database. We don't live in an ideal world. We have to give up one or more of those characteristics.

In the next chapter, you will learn how to use each of the storage solutions described here, with an emphasis on loading and querying data.

Summary

When planning cloud storage, consider the types of storage systems and types of data models. The storage systems provide the hardware and basic organizational structure used for storing data. The data models organize data into logical structures that determine how data is stored and queried within a database.

The main storage systems available in Google Cloud are Memorystore, a managed cache service, and persistent disks, which are network-accessible disks for VMs in Compute Engine and Kubernetes Engine. Cloud Storage is Google Cloud's object storage system.

The primary data models are object, relational, and NoSQL. NoSQL databases in Google Cloud are further subdivided into document and wide-column databases. Cloud Storage uses an object data model. Cloud SQL and Cloud Spanner use relational databases for transaction processing applications. BigQuery uses a relational model for data warehouse and analytic applications. Firestore is a document database. Bigtable is a wide-column table.

When choosing data storage systems, consider read and write patterns, consistency requirements, transaction support, cost, and latency.

Exam Essentials

Know the major storage system types, including caches, persistent disks, and object storage. Caches are used to improve application performance by reducing the need to read from databases on disk. Caches are limited by the amount of available memory. Persistent disks are network devices that are attached to VMs. Persistent disks may be attached to multiple VMs in read-only mode. Object storage is used for storing files for shared access and long-term storage.

Know the major kinds of data models. Relational databases are used for transaction processing systems that require transaction support and strong consistency. Cloud SQL and Cloud Spanner are relational databases used for transaction processing applications. BigQuery uses an analytical model but is designed for data warehouses and analytics. The object model is an alternative to a filesystem model. Objects, stored as files, are treated as atomic units. NoSQL data models include document data models and wide-column models. Firestore is a document database. Bigtable is a wide-column database.

Know the various classes in Cloud Storage. Standard, nearline, coldline, and archive are the four storage classes. Standard is designed for data that is accessed frequently (more than once per month) or only stored in Cloud Storage for a short time. Nearline is designed for infrequent access, less than once per month. Coldline storage is designed for long-term storage, with files being accessed less than once per 90 days. Archive storage is designed for data that is not accessed more frequently than once per year. Nearline, Coldline, and Archive storage incur retrieval charges in addition to charges based on the size of the data.

Know that cloud applications may require more than one kind of data store. For example, an application may need a cache to reduce latency when querying data in Cloud SQL, object storage for the long-term storage of data files, and BigQuery for data warehousing reporting and analysis.

Know that you can apply lifecycle configurations on Cloud Storage buckets. Lifecycles are used to delete files and change storage class. Standard class objects can be changed to Nearline, Coldline, or Archive. Nearline storage can change to Coldline and Archive. Coldline can be changed to Archive.

Know the characteristics of different data stores that help you determine which is the best option for your requirements. Read and write patterns, consistency requirements, transaction support, cost, and latency are often factors.

Review Questions

You can find the answers in the Appendix.

1. You are tasked with defining life cycle configurations on buckets in Cloud Storage. You need to consider all possible options for transitioning from one storage class to another. All of the following transitions are allowed except for which one?
 - A. Nearline to Coldline
 - B. Coldline to Archive
 - C. Standard to Nearline
 - D. Archive to Standard
2. Your manager has asked for your help in reducing Cloud Storage charges. You know that some of the files stored in Cloud Storage are rarely accessed more than once every 90 days. What kind of storage would you recommend for those files?
 - A. Nearline
 - B. Standard
 - C. Coldline
 - D. Archive
3. You are working with a startup developing analytics software for IoT data. You need to ingest large volumes of data consistently and store it for several months. The startup has several applications that will need to query this data. Volumes are expected to grow to petabyte volumes. Which database should you use?
 - A. Cloud Spanner
 - B. Bigtable
 - C. BigQuery
 - D. Firestore
4. A software developer on your team is asking for your help improving the query performance of a database application. The developer is using a Cloud SQL MySQL database and is willing to modify some parts of the application but wants to continue to use a relational database. Which options would you recommend?
 - A. Memorystore and SSD persistent disks
 - B. Memorystore and HDD persistent disks
 - C. Firestore and SSD persistent disks
 - D. Firestore and HDD persistent disks

5. You are creating a set of persistent disks to store data for exploratory data analysis. The disks will be mounted on a virtual machine in the us-west2-a zone. The data is historical data retrieved from Cloud Storage. The data analysts do not need peak performance and are more concerned about cost than performance. The data will be stored in a local relational database. Which type of storage would you recommend?
 - A. SSDs
 - B. HDDs
 - C. Firestore
 - D. Bigtable
6. Which of the following statements about Cloud Storage is not true?
 - A. Cloud Storage buckets can have retention periods.
 - B. Lifecycle configurations can be used to change storage class from Archive to Standard.
 - C. Cloud Storage does not provide block-level access to data within files stored in buckets.
 - D. Cloud Storage is designed for high durability.
7. When using versioning on a bucket, what is the latest version of the object called?
 - A. Live version
 - B. Top version
 - C. Active version
 - D. Safe version
8. A product manager has asked for your advice on which database services might be options for a new application. Transactions and support for tabular data are important. Ideally, the database would support common query tools. What databases would you recommend the product manager consider?
 - A. BigQuery and Spanner
 - B. Cloud SQL and Spanner
 - C. Cloud SQL and Bigtable
 - D. Bigtable and Spanner
9. The Cloud SQL service provides fully managed relational databases. What two types of databases are available in Cloud SQL?
 - A. Oracle and MySQL
 - B. Oracle and PostgreSQL
 - C. PostgreSQL and MySQL
 - D. MySQL and DB2
10. Which of the following Cloud Spanner configurations would have the highest hourly cost?
 - A. Located in us-central1
 - B. Located in nam3
 - C. Located in us-west1-a
 - D. Located in nam-eur-asia1

11. Which of the following are database services that do not require you to specify configuration information for VMs?
 - A. BigQuery only
 - B. Firestore only
 - C. Bigtable only
 - D. BigQuery and Firestore
12. What kind of data model is used by Firestore?
 - A. Relational
 - B. Document
 - C. Wide-column
 - D. Graph
13. You have been tasked with creating a data warehouse for your company. It must support tens of petabytes of data and use SQL for a query language. Which managed database service would you choose?
 - A. BigQuery
 - B. Bigtable
 - C. Cloud SQL
 - D. IBM DB2
14. A team of mobile developers is developing a new application. It will require synchronizing data between mobile devices and a back-end database. Which database service would you recommend?
 - A. BigQuery
 - B. Firestore
 - C. Spanner
 - D. Bigtable
15. A product manager is considering a new set of features for an application that will require additional storage. What features of storage would you suggest the product manager consider?
 - A. Read and write patterns only.
 - B. Cost only.
 - C. Consistency and cost only.
 - D. They are all relevant considerations.
16. What is the maximum size of a Memorystore cache when using Redis?
 - A. 100 GB
 - B. 300 GB
 - C. 400 GB
 - D. 50 GB

17. Once a bucket has its storage class set to Archive, what are other storage classes it can transition to?
- A. Standard
 - B. Nearline
 - C. Coldline
 - D. None of the above
18. Before you can start storing data in BigQuery, what must you create?
- A. A dataset
 - B. A bucket
 - C. A persistent disk
 - D. An entity
19. What features can you configure when running a MySQL database in Cloud SQL?
- A. Machine type
 - B. Maintenance windows
 - C. Failover replicas
 - D. All of the above
20. A colleague is wondering why some storage charges are so high. They explain that they have moved all their storage to Nearline and Coldline storage and then costs increased. They routinely access most of the objects on any given day. What is one possible reason the storage costs are higher than expected?
- A. Nearline and Coldline incur access charges.
 - B. Transfer charges are involved.
 - C. Egress charges are involved.
 - D. None of the above.

Chapter 12

Deploying Storage in Google Cloud

**THIS CHAPTER COVERS THE FOLLOWING
OBJECTIVES OF THE GOOGLE ASSOCIATE
CLOUD ENGINEER CERTIFICATION EXAM:**

- ✓ 3.4 Deploying and implementing data solutions
- ✓ 4.4 Managing storage and database solutions





In this chapter, we will discuss how to create data storage systems in several Google Cloud products, including Cloud SQL, Cloud Datastore, BigQuery, Bigtable, Cloud Spanner, Cloud Pub/Sub, Cloud Dataproc, and Cloud Storage. You will learn how to create databases, buckets, and other basic data structures as well as how to perform key management tasks, such as backing up data and checking the status of jobs.

Deploying and Managing Cloud SQL

Cloud SQL is a managed relational database service. In this section, you will learn how to do the following:

- Create a database instance.
- Connect to the instance.
- Create a database.
- Load data into the database.
- Query the database.
- Back up the database.

We will use a MySQL instance in this section, but the following procedures are similar for PostgreSQL and SQL Server.

Creating and Connecting to a MySQL Instance

We described how to create and configure a MySQL instance in Chapter 11, “Planning Storage in the Cloud,” but will review the steps here.

From the console, navigate to Cloud SQL and click Create Instance. Choose MySQL to open the page shown in Figure 12.1.

After a few minutes, the instance is created; the MySQL list will look similar to Figure 12.2.

FIGURE 12.1 Creating a MySQL instance

←

Create a MySQL instance

Instance info

Instance ID *

Use lowercase letters, numbers, and hyphens. Start with a letter.

Password *

GENERATE

Set a password for the root user. [Learn more](#)

☐ No password

Database version *

MySQL 8.0

Choose region and zonal availability

For better performance, keep your data close to the services that need it. Region is permanent, while zone can be changed any time.

Region

us-central1 (Iowa)

Zonal availability

☐ Single zone
In case of outage, no failover. Not recommended for production.

☒ Multiple zones (Highly available)
Automatic failover to another zone within your selected region. Recommended for production instances. Increases cost.

SPECIFY ZONES

Customize your instance

You can also customize instance configurations later

SHOW CONFIGURATION OPTIONS

CREATE INSTANCE

CANCEL

Summary

Region	us-central1 (Iowa)
DB Version	MySQL 8.0
vCPUs	4 vCPU
Memory	26 GB
Storage	100 GB
Network throughput (MB/s)	1,000 of 2,000
Disk throughput (MB/s)	Read: 48.0 of 240.0 Write: 48.0 of 240.0
IOPS	Read: 3,000 of 15,000 Write: 3,000 of 15,000
Connections	Public IP
Backup	Automated
Availability	Multiple zones (Highly available)
Point-in-time recovery	Enabled

FIGURE 12.2 A listing of MySQL instances

SQL

Instances

+ CREATE INSTANCE

HELP ASSISTANT

SHOW INFO PANEL

Filter

Enter property name or value

?

<input type="checkbox"/>	Instance ID <div><div>?</div><div>↑</div></div>	Type	Public IP address	Private IP address	Instance connection name	Actions
<input type="checkbox"/>	<div><div>✓</div><div>ace-exam-mysql</div></div>	MySQL 8.0	35.238.89.86		scenic-energy-33502... <div><div>▼</div></div>	<div><div></div><div></div><div></div></div>

After the database is created, you can connect by starting Cloud Shell and using the `gcloud sql connect` command. This command takes the name of the instance to connect to and optionally a username and password. It is a good practice to not specify a password in the command line. Instead, you will be prompted for it, and it will not be displayed as

you type. You may see a message about allowing the listing of your IP address; this is a security measure and will allow you to connect to the instance from Cloud Shell.

To connect to the instance called `ace-exam-mysql`, use the following command:

```
gcloud sql connect ace-exam-mysql --user=root
```

This opens a command-line prompt to the MySQL instance, as shown in Figure 12.3.

FIGURE 12.3 Command-line prompt to work with MySQL after connecting using `gcloud sql connect`

```
dan@cloudshell:~ (scenic-energy-335022)$ gcloud sql connect ace-exam-mysql --user=root
Allowlisting your IP for incoming connection for 5 minutes...done.
Connecting to database with SQL user [root].Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 257320
Server version: 8.0.26-google (Google)

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> █
```

Creating a Database, Loading Data, and Querying Data

In the MySQL command-line environment, you use MySQL commands, not `gcloud` commands. MySQL uses standard SQL, so the command to create a database is `CREATE DATABASE`. You indicate the database to work with (there may be many in a single instance) by using the `USE` command. For example, to create a database and set it as the default database to work with, use this:

```
CREATE DATABASE ace_exam_book;
USE ace_exam_book
```

You can then create a table using `CREATE TABLE`. Data is inserted using the `INSERT` command. For example, the following commands create a table called `books` and inserts two rows:

```
CREATE TABLE books (title VARCHAR(255), num_chapters INT,
entity_id INT NOT NULL AUTO_INCREMENT, PRIMARY KEY (entity_id));
INSERT INTO books (title,num_chapters)
VALUES ('ACE Exam Study Guide', 18);
INSERT INTO books (title,num_chapters)
VALUES ('Architecture Exam Study Guide', 18);
```

To query the table, you use the `SELECT` command. Here's an example:

```
SELECT * FROM books;
```

This command will list all the rows in the table, as shown in Figure 12.4.

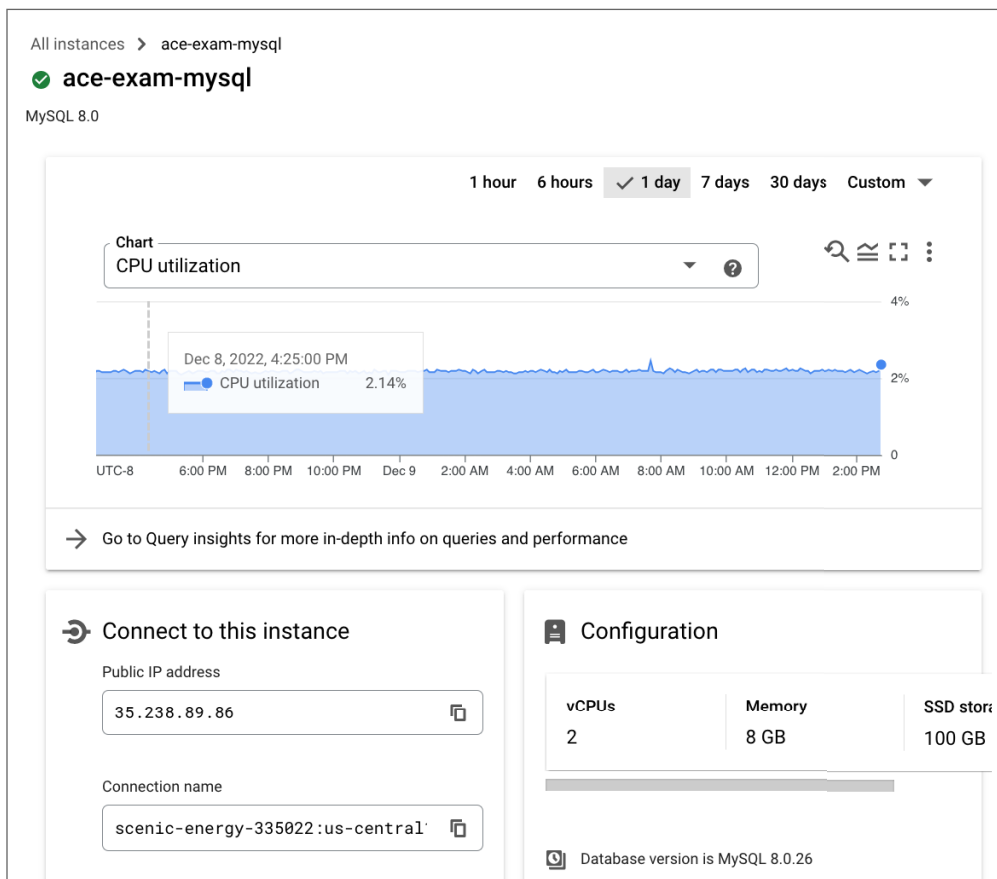
FIGURE 12.4 Listing the contents of a table in MySQL

```
mysql> SELECT * FROM books;
+-----+-----+-----+
| title                | num_chapters | entity_id |
+-----+-----+-----+
| ACE Exam Study Guide |          18  |         1 |
| Architecture Exam Study Guide |          18  |         2 |
+-----+-----+-----+
2 rows in set (0.00 sec)
```

Backing Up MySQL in Cloud SQL

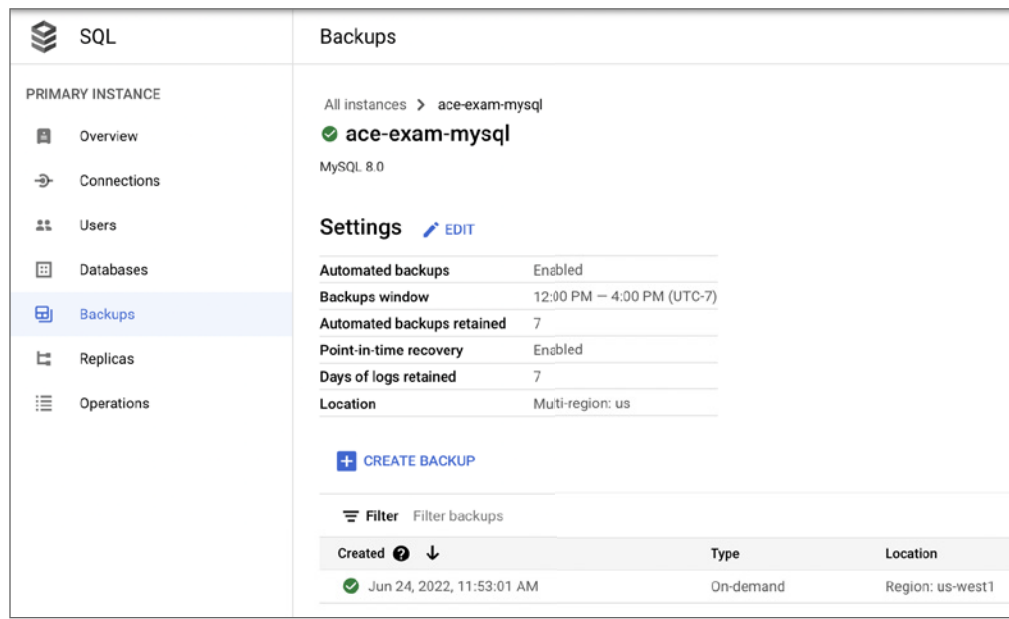
Cloud SQL enables both on-demand and automatic backups.

To create an on-demand backup, click the name of the instance on the Instances page on the console. This will display the Instance Details page (see Figure 12.5).

FIGURE 12.5 Partial listing of MySQL Instance Details page with vertical menu displayed on the left

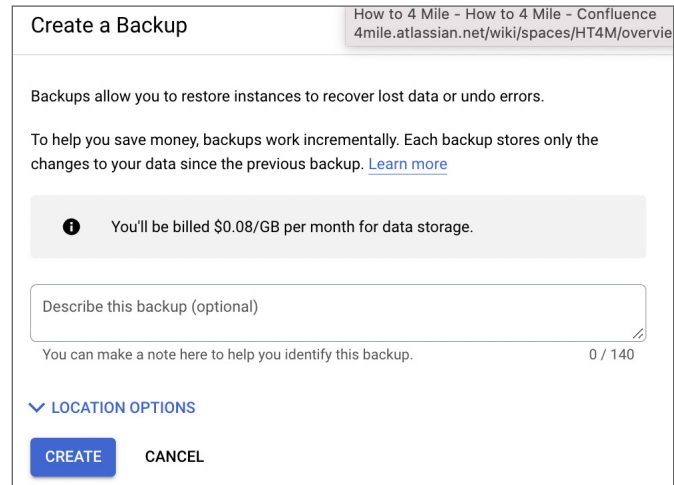
Click the Backups menu option to display the Backups page (see Figure 12.6).

FIGURE 12.6 Create Backup button



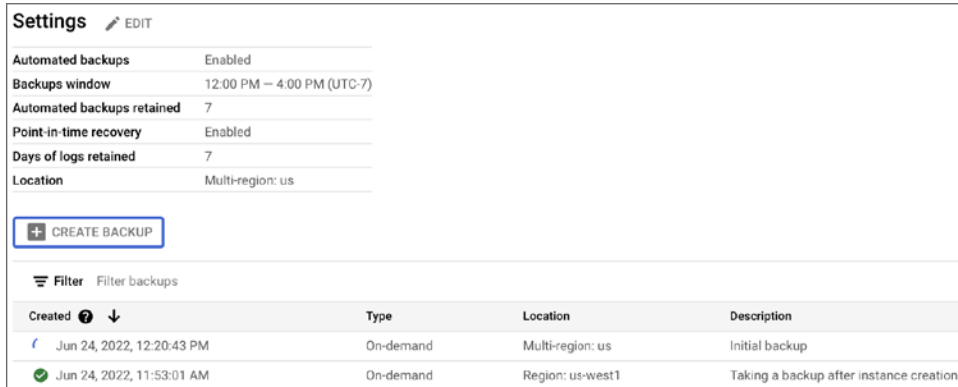
Clicking Create Backup opens the window shown in Figure 12.7.

FIGURE 12.7 Assign a description to a backup and create it.



Fill in the optional description and click Create. When the backup is complete, it will appear in the list of backups, as shown in Figure 12.8.

FIGURE 12.8 Listing of backups available for this instance



Settings <small>EDIT</small>	
Automated backups	Enabled
Backups window	12:00 PM – 4:00 PM (UTC-7)
Automated backups retained	7
Point-in-time recovery	Enabled
Days of logs retained	7
Location	Multi-region: us

CREATE BACKUP

Created	Type	Location	Description
Jun 24, 2022, 12:20:43 PM	On-demand	Multi-region: us	Initial backup
Jun 24, 2022, 11:53:01 AM	On-demand	Region: us-west1	Taking a backup after instance creation

You can also create a backup using the `gcloud sql backups` command, which has this form:

```
gcloud sql backups create --async --instance [INSTANCE_NAME]
```

Here, `[INSTANCE_NAME]` is the name, such as `ace-exam-mysql`, and the `--async` parameter is optional.

To create an on-demand backup for the `ace-exam-mysql` instance, use the following command:

```
gcloud sql backups create --async --instance ace-exam-mysql
```

You can also have Cloud SQL automatically create backups.

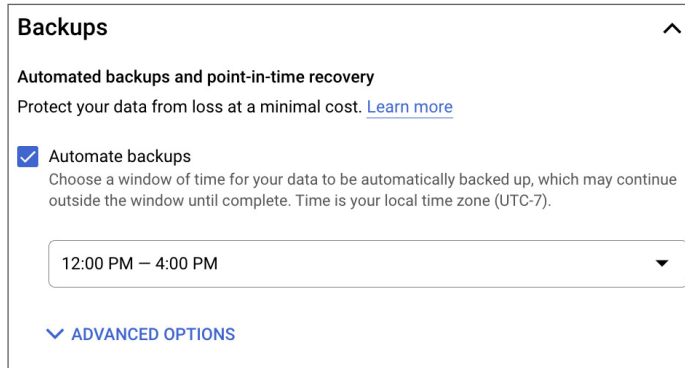
From the console, navigate to the Cloud SQL Instance page, click the name of the instance, and then click Edit Instance. Open the Enabled Auto Backups section and fill in the details of when to create the backups (see Figure 12.9). You must specify a time range for when automatic backups should occur. You can also enable binary logging, which is needed for more advanced features, such as point-in-time recovery.

To enable automatic backups from the command line, use the `gcloud` command:

```
gcloud sql instances patch [INSTANCE_NAME] --backup-start-time [HH:MM]
```

For this example instance, you could run automatic backups at 1:00 a.m. with the following command:

```
gcloud sql instances patch ace-exam-mysql --backup-start-time 01:00
```

FIGURE 12.9 Enabling automatic backups in Cloud Console

Deploying and Managing Firestore

Chapter 11 described how to initialize a Firestore document database. Now, you will see how to create entities and add properties to a document database. You'll also review backup and restore operations. Cloud Firestore is the latest generation of Cloud Datastore. Cloud Firestore has two modes: Native and Datastore mode.

Cloud Firestore's features include strong consistency, document data model, real-time updates, and mobile and web client libraries. Real-time updates and mobile and web client libraries are only available in native mode. Datastore mode can scale to millions of writes per second and is a good option for a document data store when you do not need the real-time or mobile features of Native mode. Datastore mode also supports the GQL query language, which is similar to SQL.

Adding Data to a Firestore Database

You add data to a Firestore database in Native Mode using the Start Collection option in the Firestore section of the console. The Collections data structure is analogous to a schema in relational databases.

You create an entity by clicking Start Collection and filling in the form that appears. Here you will provide a collection ID and then add documents, which are key-value pairs with a data type on the value. (See Figure 12.10.)

After creating entities, you can view data in the console, as shown in Figure 12.11.

FIGURE 12.10 Adding data to a Firestore collection

Start a collection

A collection is a set of one or more documents that contain data. Start a collection at this path by adding its first document. [Learn more](#)

Give the collection an ID

Parent path
/

Collection ID *
ace_exam_questions

Choose an ID that describes the documents you'll add to this collection.

Add its first document ?

Document ID
YncGFFXRCua4TU0brrHB

Assigned automatically. Customize if needed.

Field name	Field type	Field value
exam_chapter	string	12
title	string	"Managing Data"

+ ADD FIELD

SAVE SAVE & ADD ANOTHER CANCEL

FIGURE 12.11 Viewing data in Firestore, Native mode

Data		
Cloud Firestore in Native mode Database location: us-west1		
/ > ace_exam_questions > YncGFFXRCua4TU0brrHB		
Root	ace_exam_questions	YncGFFXRCua4TU0brrHB
+ START COLLECTION	+ ADD DOCUMENT	+ START COLLECTION
ace_exam_questions	30mJlwGNtzYuMwFAnAIQ	+ ADD FIELD
	YncGFFXRCua4TU0brrHB	exam_chapter: "12"
		title: "Managing Data"

Backing Up Firestore

To back up a Firestore database, you need to create a Cloud Storage bucket to hold a backup file and grant appropriate permissions to users performing backup.

You can create a bucket for backups using the `gsutil` command:

```
gsutil mb gs://[BUCKET_NAME]/
```

Here, `[BUCKET_NAME]` is the name, such as `ace_exam_backups`. In our example, we use `ace_exam_backups` and create that bucket using the following:

```
gsutil mb gs://ace_exam_backups/
```

Users creating backups need the `datastore.databases.export` permission. (Cloud Datastore was renamed to Cloud Firestore but at the time of writing, the IAM roles still refer to Datastore.) If you are importing data, you will need `datastore.databases.import`. The Cloud Datastore Import Export Admin role has both permissions; see Chapter 17, “Configuring Access and Security,” for details on assigning roles to users.

To create a backup by exporting from Firestore, you can use a command like this one:

```
gcloud firestore export gs://ace_exam_backups
```

To import a backup file, use the `gcloud firestore import` command:

```
gcloud firestore import gs://ace_exam_backups
```

Deploying and Managing BigQuery

BigQuery is a fully managed database service, so Google takes care of backups and other basic administrative tasks. As a Cloud Engineer, you still have some administrative tasks when working with BigQuery. Two of those tasks are estimating the cost of a query and checking on the status of a job.

Estimating the Cost of Queries in BigQuery

In the console, choose BigQuery from the main navigation menu to display the BigQuery query interface, as partially shown in Figure 12.12.

Here you can enter a query in the Query Editor, such as a query about names and genders in the `usa_1910_2013` table, as shown in Figure 12.13.

Notice in the upper-right corner that BigQuery provides an estimate of how much data will be scanned. You can also use the command line to get this estimate by using the `bq` command with the `--dry-run` option:

```
bq --location=[LOCATION] query --use_legacy_sql=false --dry_run [SQL_QUERY]
```

FIGURE 12.12 The BigQuery console

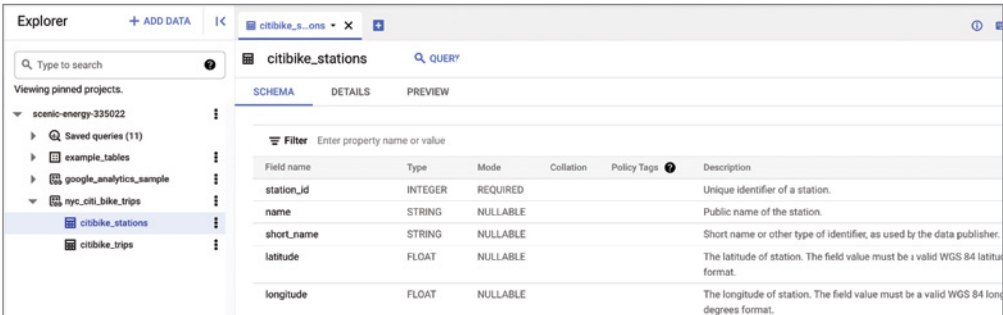
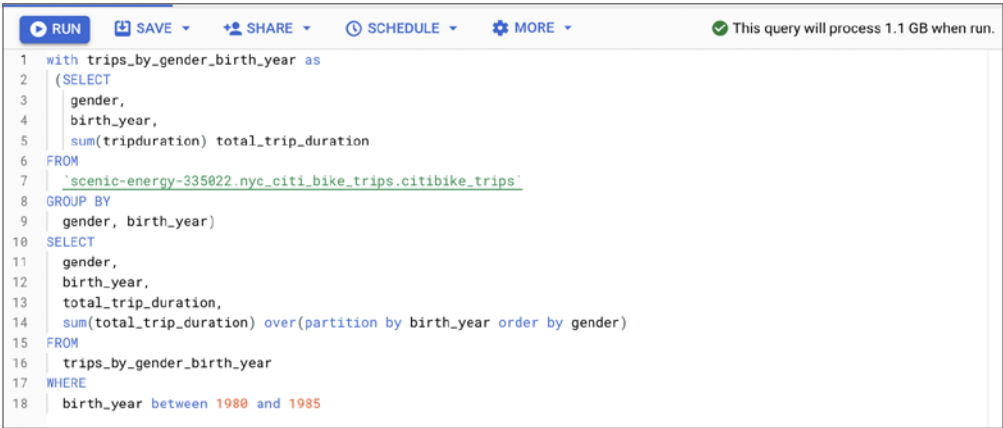


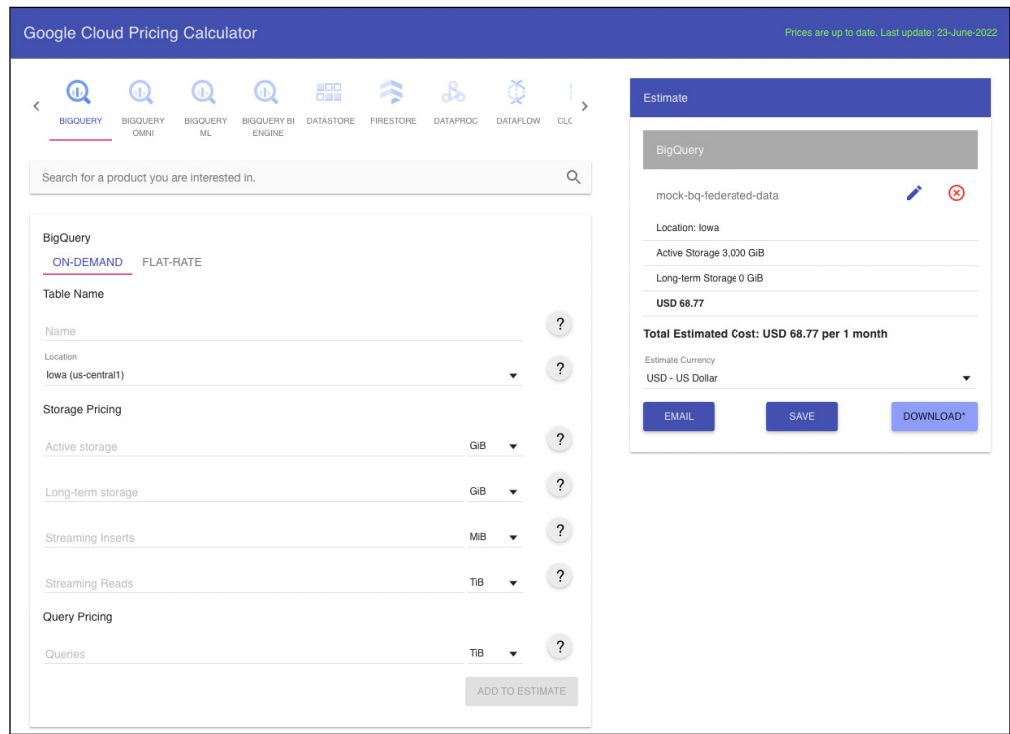
FIGURE 12.13 Example query with estimated amount of data scanned



Here, *[Location]* is the location in which you created the data set you are querying, and *[SQL_QUERY]* is the SQL query you are estimating.

You can use this number with the Pricing Calculator to estimate the cost. The Pricing Calculator is available at <https://cloud.google.com/products/calculator>. After selecting BigQuery, navigate to the On-Demand tab, enter the name of the table you are querying, set the amount of storage to 0, and then enter the size of the query in the Queries line of the Queries Pricing section. Be sure to use the same size unit as displayed in the BigQuery console. When you click Add To Estimate, the Pricing Calculator will display the cost (see Figure 12.14).

FIGURE 12.14 Using the Pricing Calculator to estimate the cost of a query



Viewing Jobs in BigQuery

Jobs in BigQuery are processes used to load, export, copy, and query data. Jobs are automatically created when you start any of these operations.

To view the status of jobs, navigate to the BigQuery console and click Personal History or Project History in the lower section of the edit window. Notice in Figure 12.15 that the top job in the list has a check mark, indicating that the job completed successfully. This is an example of an expanded view of a job entry. Below that is a single-line summary of a job that failed. The failure is indicated by the exclamation point icon next to the job ID.

FIGURE 12.15 A listing of job statuses in BigQuery

PERSONAL HISTORY							REFRESH	
Filter Enter property name or value								
Job ID	Creation time	Owner	Type	Summary	Sessi	Actions		
✓ bqjob_4428709e_1818e183371	Jun 22, 2022, 6:06:00 PM	dan@sullivanlearninggroup..	QUERY	SELECT s.station_id, count(*) FROM `scenic-energy-3350..		⋮		
✓ bqjob_3f9e061_1818e126794	Jun 22, 2022, 6:00:00 PM	dan@sullivanlearninggroup..	QUERY	SELECT * FROM `scenic-energy-335022.nyc_citi_bike_tri..		⋮		
✓ bqjob_485f7cf_18182033afb	Jun 20, 2022, 1:34:50 PM	dan@sullivanlearninggroup..	QUERY	SELECT * FROM `scenic-energy-335022.google_analytic..		⋮		
! bqjob_6929b4d_1817eb55ba0	Jun 19, 2022, 6:23:44 PM	dan@sullivanlearninggroup..	QUERY	SELECT * FROM Unnest((SELECT [(1,foo), (2, bar), (3, b...		⋮		

You could also view the status of a BigQuery job by using the `bq show` command. For example, the following command shows the status of the specified job:

```
bq --location=US show -j gcp-space-project:US.bq.job_119adae7_167c373d5c3
```

Deploying and Managing Cloud Spanner

Now, let's turn our attention to Cloud Spanner, the global relational database. In this section, you will create a database, define a schema, insert some data, and then query it.

First, you will create a Cloud Spanner instance. Navigate to the Cloud Spanner page in the console and click Create Instance. This will display the page shown in Figure 12.16.

FIGURE 12.16 Creating a Cloud Spanner instance

Create an instance

Name your instance
An instance has both a name and an ID. The name is for display purposes only. The ID is a permanent and unique identifier.

Instance name *
ace-exam-spanner
Name must be 4-30 characters long

Instance ID *
ace-exam-spanner
Lowercase letters, numbers, hyphens allowed

Choose a configuration
Determines where your nodes and data are located. Affects cost, performance, and replication. A multi-region configuration will select the default leader region for your leader replicas. You can change your leader region at any time with a DDL statement. [Learn more](#)

[COMPARE REGION CONFIGURATIONS](#)

☒ Regional
☐ Multi-region

us-west1 (Oregon)

Allocate compute capacity
Your compute capacity determines the amount of data throughput, queries per second (QPS), and storage limits in your instance. One node equals 1,000 processing units. Affects billing.

Unit *
Processing units

Quantity *
100
Integers only. Enter in increments of 100 up to 1,000, followed by increments of 1,000.

[COMPUTE CAPACITY GUIDANCE](#)


[CREATE](#) [CANCEL](#)

Summary
Storage cost depends on GB stored per month. Compute cost refers to the hourly charge of nodes or processing units in your instance.

Configuration	us-west1 (Oregon)
Replicas	3 read-write replicas in 3 separate zones within the region us-west1
Availability	99.99% availability SLA
Compute cost	\$0.09 per hour Save up to 20% by committing to 1 year and 40% by committing to 3 years using Committed Use Discounts. Learn more
Storage cost	\$0.30 per GB/month
Maximum storage capacity	410 GB

Next, you need to create a database in the instance. Click **Create Database** at the top of the Instance Details page to show a page similar to Figure 12.17.

FIGURE 12.17 Create a database within a Cloud Spanner instance.

 Create a database in ace-exam-spanner

Name your database

Enter a permanent name for your database of at least two characters, starting with a letter.

Database name *
ace-spanner-db1

Lowercase letters, numbers, hyphens, underscores allowed

Select database dialect

Choose between Google Standard SQL and PostgreSQL dialects for your Spanner database.

☒ Google Standard SQL

☐ PostgreSQL

Define your schema (optional)

Add Spanner Data Definition Language SQL statements below. Separate statements with a semicolon. [Learn more](#)

DDL TEMPLATES ▼ **SHORTCUTS**

Press Alt+F1 for Accessibility Options.

```
1 CREATE TABLE <table_name> (  
2   <col_name> <col_type>,  
3 ) PRIMARY KEY (<col_name>;
```

✓ SHOW ENCRYPTION OPTIONS

CREATE

CANCEL

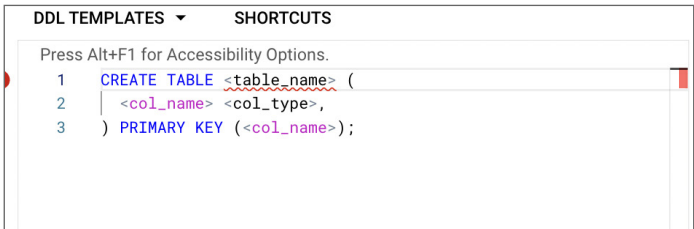
When creating a database, you will need to use the SQL data definition language (DDL) to define the structure of tables. SQL DDL is the set of SQL commands for creating tables, indexes, and other data structures (see Table 12.1). Figure 12.18 shows an example of using DDL templates provided by Google Cloud. In this case, the template for creating a table is displayed.

TABLE 12.1 SQL data definition commands

Command	Description
CREATE TABLE	Creates a table with columns and data types specified
CREATE INDEX	Creates an index on the specified column(s)
ALTER TABLE	Changes table structure
DROP TABLE	Removes the table from the database schema
DROP INDEX	Removes the index from the database schema

In addition to creating a table, other templates are shown in Figure 12.19.

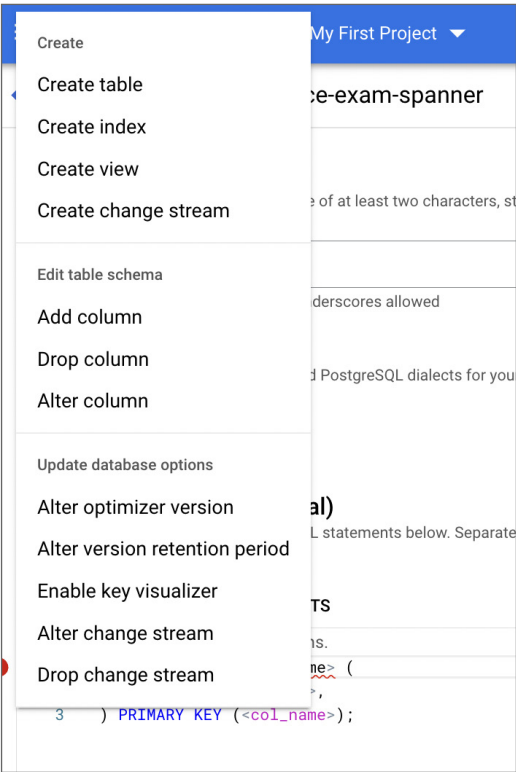
FIGURE 12.18 Creating a table using a DDL template



Once a table is created, you can view the structure and properties of the table, as shown in Figure 12.20.

From the schema description of the table, you can navigate to Cloud Logging to see a history of changes to the table, as shown in Figure 12.21.

FIGURE 12.19 DDL templates available to help you create database objects in Spanner



Finally, you can review and add Spanner-related roles to principals from the Spanner console. From the Spanner instance list, select the check box for the instance. A panel will appear on the right similar to Figure 12.22.

Cloud Spanner is a managed database service, so you will not have to patch, back up, or perform other basic data administration tasks. Your tasks, and those of data modelers and software engineers, will focus on design tables and queries.

FIGURE 12.20 Details of the table created in Spanner

All instances

>

INSTANCE

ace-exam-spanner: Overview

>

GOOGLE STANDARD SQL DATABASE

ace-spanner-db1: Overview

>

TABLE

my_table: Schema

Schema

Name

my_table

Schema updates

Update completed

To view all updates go to [Cloud Logging](#).

Primary Key(s): id_column (asc)

	Column	Type	Nullable	Order	Watched by
	id_column	INT64	No	asc	
	name_column	STRING(MAX)	Yes	—	

[SHOW EQUIVALENT DDL](#)

Interleaved tables

There are no tables in my table. Add a table to get started.

ADD TABLE

FIGURE 12.21 Log of changes to Spanner table

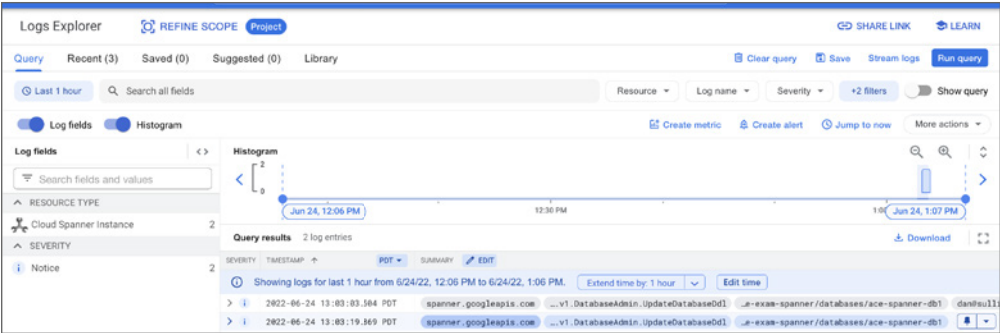
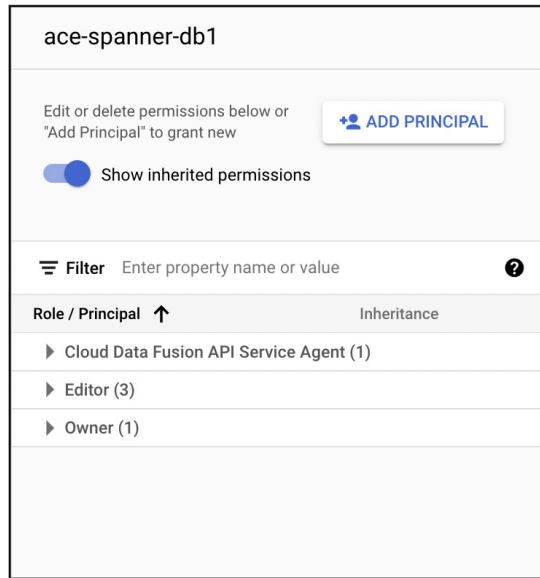


FIGURE 12.22 From the Show Info panel, you can view and manage Spanner-related roles.



Deploying and Managing Cloud Pub/Sub

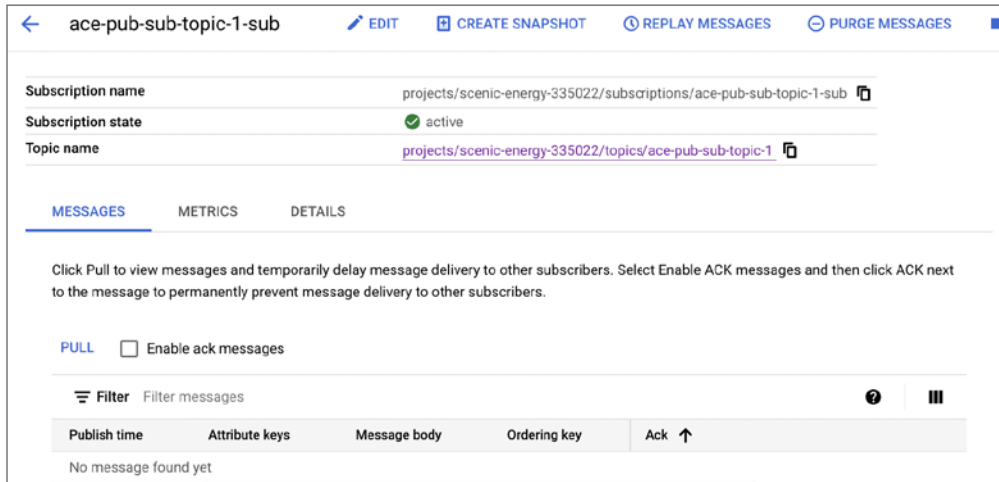
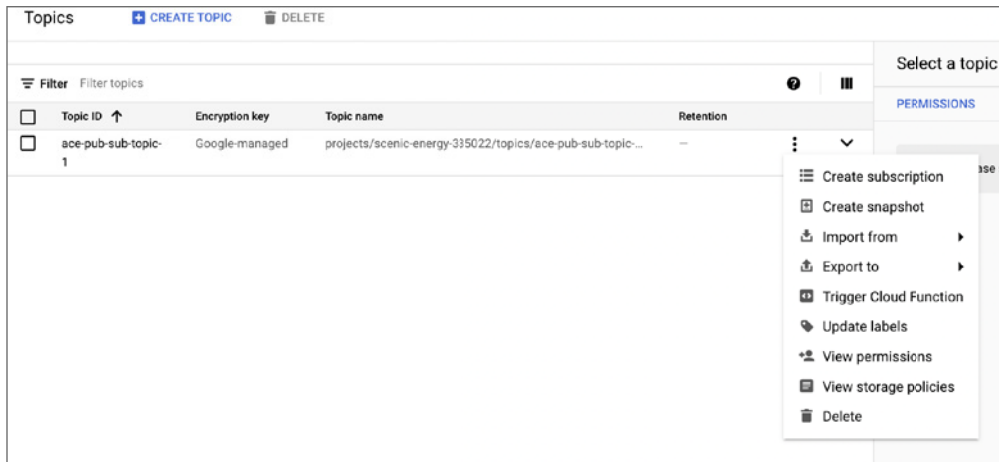
Two tasks are required to deploy a Pub/Sub message queue: creating a topic and creating a subscription. A topic is a structure where applications can send messages. Pub/Sub receives the messages and keeps them until they are read by an application. Applications read messages by using a subscription.

The first step for working with Pub/Sub is to navigate to the Pub/Sub page in Cloud Console. The first time you use Pub/Sub, the form will be similar to Figure 12.23.

After you click Create A Topic, you will see a list of subscriptions, as shown in Figure 12.24.

You will see a list of topics displayed in the Topics page after creating the first topic, as shown in Figure 12.25.

To create a subscription to a topic, click the ellipsis icon at the end of the topic summary line in the listing. The menu that appears includes the Create Subscription option (see Figure 12.26). Click Create Subscription to create a subscription to that topic. This will display a page like that shown in Figure 12.27.

FIGURE 12.25 Subscription details**FIGURE 12.26** Creating a subscription to a topic

Once a message is read, the application reading the message acknowledges receiving the message. Pub/Sub will wait the period of time specified in the Acknowledgment Deadline parameter. The time to wait can range from 10 to 600 seconds.

You can also specify a retention period, which is the length of time to keep a message that cannot be delivered. After the retention period passes, messages are deleted from the topic.

When you complete creating a subscription, you will see a list of subscriptions like that shown in Figure 12.28.

FIGURE 12.27 The options for creating a subscription

← Create subscription

A subscription directs messages on a topic to subscribers. Messages can be pushed to subscribers immediately, or subscribers can pull messages as needed.

Subscription ID *

ace-exam-subscription-1

?

Subscription name:

projects/scenic-energy-335022/subscriptions/ace-exam-subscription-1

Select a Cloud Pub/Sub topic *

projects/scenic-energy-335022/topics/ace-pub-sub-topic-1

▼

Delivery type ?

☒ Pull

☐ Push

Message retention duration ?

Duration is from 10 minutes to 7 days

Days

7

▼

Hours

0

▼

Minutes

0

▼

☐ Retain acknowledged messages ?

When enabled, acknowledged messages are retained for the message retention duration specified above. This increases message storage fees. [Learn more](#)

Expiration period ?

☒ Expire after this many days of inactivity (up to 365)

31

Days

A subscription is inactive if there is no subscriber activity such as open connections, active pulls, or successful pushes.

☐ Never expire

The subscription will never expire no matter the activity.

Acknowledgement deadline ?

FIGURE 12.28 A list of subscriptions

Subscriptions									
		+ CREATE SUBSCRIPTION		🗑 DELETE					
Filter Filter subscriptions									
<input type="checkbox"/>	State	Subscription ID ↑	Delivery type	Topic name	Ack deadline	Retention			
<input type="checkbox"/>	✓	ace-exam-subscription-2	Pull	projects/scenic-e...	10 seconds	7 days	C	⋮	▼
<input type="checkbox"/>	✓	ace-pub-sub-topic-1-sub	Pull	projects/scenic-e...	10 seconds	7 days	C	⋮	▼

In addition to using the console, you can use `gcloud` commands to create topics and subscriptions. The commands to create topics and subscriptions are as follows:

```
gcloud pubsub topics create [TOPIC-NAME]
gcloud pubsub subscriptions create [SUBSCRIPTION-NAME] --topic [TOPIC-NAME]
```

Deploying and Managing Cloud Bigtable

As a Cloud Engineer, you may need to create a Bigtable cluster, or set of servers running Bigtable services, as well as create tables, add data, and query that data.

To create a Bigtable instance, navigate to the Bigtable console and click Create Instance. This will display the page shown in Figure 12.29. (See Chapter 11 for additional details on creating a Bigtable instance.)

Once an instance is created, you can view a summary of performance data in the Instance Details page, such as shown in Figure 12.30.

Much of the work you will do with Bigtable is done at the command line.

To create a table, open a Cloud Shell browser and install the `cbt` command. Unlike relational databases, Bigtable is a NoSQL database and does not use the SQL command. Instead, the `cbt` command has subcommands to create tables, insert data, and query tables (see Table 12.2).

TABLE 12.2 `cbt` commands

Command	Description
<code>createtable</code>	Creates a table
<code>createfamily</code>	Creates a column family
<code>read</code>	Reads and displays rows
<code>ls</code>	Lists tables and columns

To configure `cbt` in Cloud Shell, enter these commands:

```
gcloud components update
gcloud components install cbt
```

FIGURE 12.29 Creating a Bigtable instance

← Create an instance

A Bigtable instance is a container for your clusters. [Learn more](#)

✓ Name your instance

✓ Select your storage type

3 Configure your first cluster

\$468 per month (estimated)
That's about \$0.65 an hour with 0 GB stored.
[SHOW DETAILS](#)

Select a cluster ID

ID is permanent

Cluster ID *
ace-bigtable-1-c1

Select a location

Choice is permanent. Determines where cluster data is stored. To reduce latency and increase throughput, store your data near the services that need it. [Learn more](#)

Region *
Zone

Choose node scaling mode

Nodes are compute resources that Bigtable uses to manage your data and perform maintenance tasks. Adding nodes helps a cluster handle larger workloads.

Scaling mode and configurations can be changed at any time.

☒ Manual allocation
Set your node count for fixed costs and compute resources.

For better instance performance, keep your CPU utilization under the recommended threshold for your [app profile routing policy](#). [Contact us](#) if you need to increase your quota.

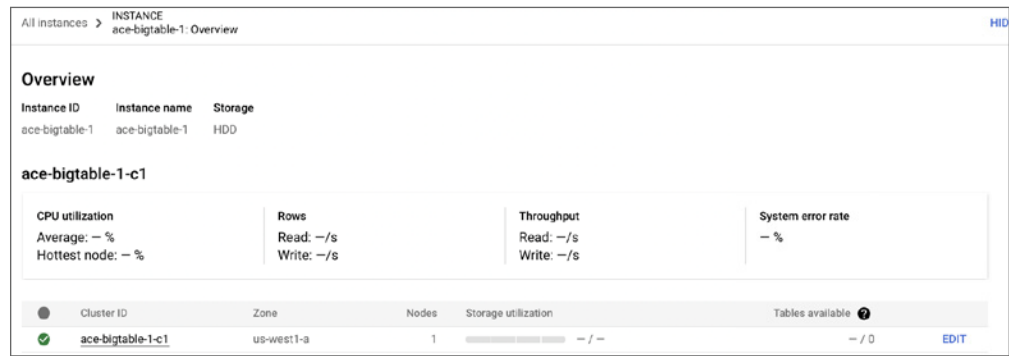
Quantity *
1
Nodes

☐ Autoscaling
Let Bigtable automatically add and remove nodes.

[SHOW ENCRYPTION OPTIONS](#)

Bigtable requires an environment variable called `instance` to be set by including it in a CBT configuration file called `.cbtrc`, which is kept in the home directory.

FIGURE 12.30 Instance details, including performance data



For example, to set the instance to `ace-exam-bigtable`, enter this command at the command-line prompt:

```
echo instance = ace-exam-bigtable >> ~/.cbtrc
```

Now `cbt` commands will operate on that instance. To create a table, issue a command such as this:

```
cbt createtable ace-exam-bt-table
```

The `ls` command lists tables. Here's an example:

```
cbt ls
```

This will display a list of all tables. Tables contain columns, but Bigtable also has a concept of column families. To create a column family called `colfam1`, use the following command:

```
cbt createfamily ace-exam-bt-table colfam1
```

To set a value of the cell with the column `colfam1` in a row called `row1`, use the following command:

```
cbt set ace-exam-bt-table row1 colfam1:col1=ace-exam-value
```

To display the contents of a table, use a `read` command like this:

```
cbt read ace-exam-bt-table
```

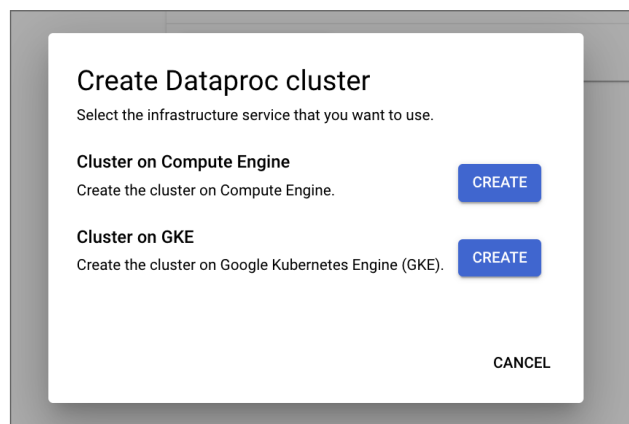
Deploying and Managing Cloud Dataproc

Cloud Dataproc is Google's managed Apache Spark and Apache Hadoop service. Both Spark and Hadoop are designed for "big data" applications. Spark supports analysis and machine learning, whereas Hadoop is well suited to batch, big data applications. As a Cloud Engineer,

you should be familiar with creating a Dataproc cluster and submitting jobs to run in the cluster.

To create a cluster, navigate to the Dataproc part of Cloud Console and select Create Cluster; then choose the underlying infrastructure, which can be Compute Engine or Google Kubernetes Engine (see Figure 12.31). Google Kubernetes Engine is a good option if you have an existing GKE cluster and want to use it for a Cloud Dataproc managed Spark/Hadoop cluster. If you do not have a GKE cluster or do not want to run Cloud Dataproc clusters on your GKE clusters, then using Compute Engine is the better option.

FIGURE 12.31 Choose an infrastructure for your cluster, either Compute Engine or Google Kubernetes Engine.



Create a Dataproc cluster by completing the Create Cluster options. You will need to specify the name of the cluster and a region and zone. You'll also need to specify the cluster mode, which can be Standard, Single Node, or High Availability. Single Node is useful for development. Standard has only one master node, so if it fails, the cluster becomes inaccessible. The High Availability mode uses three masters.

You will also need to specify machine configuration information for the master nodes and the worker nodes. You'll specify CPUs, memory, and disk information. The cluster mode determines the number of master nodes, but you can choose the number of worker nodes. If you choose to expand the list of advanced options, you can indicate you'd like to use preemptible VMs and specify the number of preemptible VMs you want to run (not shown in figures). Figure 12.32 shows the options for creating a cluster on Compute Engine, and Figure 12.33 shows the options for creating a cluster on Google Kubernetes Engine.

When the cluster is running, you can submit jobs using the Submit A Job page shown in Figure 12.34.

FIGURE 12.32 Creating a Dataproc cluster on Compute Engine

←

Create a Dataproc cluster on Compute Engine

- Set up cluster**
 Begin by providing basic information.
- Configure nodes (optional)**
 Change node compute and storage capabilities.
- Customize cluster (optional)**
 Add cluster properties, features, and actions.
- Manage security (optional)**
 Change access, encryption, and security settings.

CREATE

CANCEL

EQUIVALENT COMMAND LINE ▼

Name

Cluster Name *
 cluster-5691

Location

Region *
 us-central1

Zone *
 us-central1-c

Cluster type

☒ Standard (1 master, N workers)

☐ Single Node (1 master, 0 workers)
Provides one node that acts as both master and worker. Good for proof-of-concept or small-scale processing

☐ High Availability (3 masters, N workers)
Hadoop High Availability mode provides uninterrupted YARN and HDFS operations despite single-node failures or reboots

Autoscaling
Automates cluster resource management based on an autoscaling policy.

Policy
 None

Enhanced Flexibility Mode
Dataproc Enhanced Flexibility Mode (EFM) manages shuffle data to minimize job progress delays caused by the removal of nodes from a running cluster. EFM offloads shuffle data in one of two user-selectable modes, primary worker shuffle and Hadoop Compatible File System (HCFS) shuffle. [Learn more](#)

ⓘ

 An autoscaling policy must be selected to configure EFM.

Versioning
Use a custom image to load pre-installed packages. [Learn more](#)

You will need to specify the cluster on which to run the job and the type of job, which can be Spark, PySpark, SparkR, Hive, Spark SQL, Pig, or Hadoop. The JAR files are the Java programs that will be executed, and the Main Class or JAR is the name of the function or method that should be invoked to start the job. If you choose PySpark, you will submit a Python program; if you submit SparkR, you will submit an R program. When running Hive or SparkSQL, you will submit query files. You can also pass in optional arguments.

FIGURE 12.33 Creating a Dataproc cluster on Google Kubernetes Engine

← Create a Dataproc cluster on GKE

- **Set up cluster**
Begin by providing basic information.
- **Configure Node pools**
Change the shape and size of your Kubernetes node pools.
- **Customize cluster (optional)**
Add cluster properties, features, and actions.

CREATE

CANCEL

Name

Cluster Name *
gke-cluster-f923

Location

Region *
us-central1

Versioning

Image Type and Version
dataproc-2.0

Release Date
First released on 1/22/2021.

CHANGE

Kubernetes Cluster

Enter an underlying Kubernetes cluster.

Kubernetes Cluster *

BROWSE ?

Cloud Storage staging bucket

Cloud Storage staging bucket to be used for storing cluster job dependencies, job driver output, and cluster config files.

Storage staging bucket *

BROWSE

Workload identity

Dataproc on GKE requires the use of [GKE Workload Identity](#). If you have the necessary permissions, the 'Setup workload identity' item is enabled. Make this selection to have Google Cloud Console set up the Workload Identity bindings for you. If this selection is disabled or you decide to set up your own bindings, see [Dataproc on GKE IAM Roles and Identity](#) for more information.

☒ Setup workload identity

You can also create workflow templates in Cloud Dataproc (see Figure 12.35). Workflow templates allow you to define and execute workflows specified as a directed graph of jobs. With workflow templates, you can specify if you want to use a managed cluster, which would enable the workflow to create a cluster, run the jobs, and then shut down the cluster automatically. Alternatively, you can specify a cluster on which to run the jobs. Workflow templates are useful when you have to run complex jobs on Cloud Dataproc.

FIGURE 12.34 Submitting a job and choosing a job type

The screenshot shows the Google Cloud Dataproc 'Submit a job' interface. On the left is a navigation sidebar with the Dataproc logo and the following sections: 'Jobs on Clusters' (containing Clusters, Jobs, Workflows, and Autoscaling policies), 'Serverless' (containing Batches), and 'Utilities' (containing Component exchange, Metastore, and Workbench). The 'Jobs' option is selected. The main panel is titled 'Submit a job' and contains the following fields:

- Job ID ***: A text input field containing 'job-325f9d63'.
- Region ***: A dropdown menu set to 'us-west1'. Below it is a note: 'Specifies the Cloud Dataproc regional service, which determines what clusters are available.'
- Cluster ***: A dropdown menu set to 'ace-dataproc-cluster-1'.
- Job type ***: A dropdown menu with 'Hadoop' selected. The menu is open, showing the following options: Hadoop, Spark, SparkR, PySpark, Hive, SparkSql, and Pig.
- Archive files**: A text input field. Below it is a note: 'Archive files are extracted in the Spark working directory. Can be a GCS file with the gs:// prefix, an HDFS file on the cluster with the hdfs:// prefix, or a local file on the cluster with the file:// prefix. Supported file types: .jar, .tar, .tar.gz, .tgz, .zip.'
- Arguments**: A text input field. Below it is a note: 'Additional arguments to pass to the main class. Press Return after each argument.'
- Max restarts per hour**: A text input field. Below it is a note: 'Leave blank if you don't want to allow automatic restarts on job failure. [Learn more](#)'.

In addition to allowing to create clusters and workflows, Cloud Dataproc supports a Serverless Spark option. You can run batch jobs by choosing the Batches option under the Serverless section of the Dataproc navigation pane, as shown in Figure 12.36. With the Serverless option, you do not have to configure cluster resources or manage clusters.

In addition to using the console, you can create a cluster using the `gcloud dataproc clusters` command. Here's an example:

```
gcloud dataproc clusters create cluster-bc3d --zone us-west2-a
```

FIGURE 12.35 Creating a workflow template

This command will create a default cluster in the `us-west2-a` zone. You can also specify additional parameters for machine types, disk configurations, and other cluster characteristics.

You use the `gcloud dataproc jobs` command to submit jobs from the command line. Here's an example:

```
gcloud dataproc jobs submit spark --cluster cluster-bc3d \
--jar ace_exam_jar.jar
```

This will submit a job running the `ace_exam_jar.jar` program on the `cluster-bc3d` cluster.



Real World Scenario

Spark for Machine Learning

Retailers collect large volumes of data about shoppers' purchases, and this is especially helpful for understanding customers' preferences and interests. The transaction processing systems that collect much of this data are not designed to analyze large volumes of data. For example, if retailers wanted to recommend products to customers based on their interests, they could build machine learning models trained on their sales data. Spark has a machine learning library, called MLlib, that is designed for just this kind of problem. Engineers can export data from transaction processing systems, load it into Spark, and then apply a variety of machine learning algorithms, such as clustering and collaborative filtering, for recommendations. The output of these models includes products that are likely to be of interest to particular customers. It's applications like these that drive the adoption of Spark and other analytics platforms.

FIGURE 12.36 Serverless options allow you to run jobs without configuring clusters.

Dataproc

Jobs on Clusters

Clusters
Jobs
Workflows
Autoscaling policies

Serverless

Batches

Utilities

Component exchange
Metastore
Workbench

Release Notes

←
Create batch

Batch info

Batch ID *
batch-9aaf

Region *
us-central1

Container

Batch type *
Spark

☒ Main class
The fully qualified name of a class in a provided or standard jar file, for example, com.example.wordcount.

Main class *

☐ Main jar URI
A provided jar file to use the main class of that jar file.

Custom container image
Specify a custom container image to add Java or Python dependencies not provided by the default container image. You must host your custom container on [Container Registry](#).

Jar files
Jar files are included in the CLASSPATH. Can be a GCS file with the gs:// prefix, an HDFS file on the cluster with the hdfs:// prefix, or a local file on the cluster with the file:// prefix.

Files
Files are included in the working directory of each executor. Can be a GCS file with the gs:// prefix, an HDFS file on the cluster with the hdfs:// prefix, or a local file on the cluster with the file:// prefix.

Archive files
Archive files are extracted in the Spark working directory. Can be a GCS file with the gs:// prefix, an HDFS file on the cluster with the hdfs:// prefix, or a local file on the cluster with the file:// prefix. Supported file types: .jar, .tar, .tar.gz, .tgz, .zip.

Managing Cloud Storage

In Chapter 11, you saw how to use life cycle management policies to automatically change a bucket's storage class. For example, you could create a policy to change a regional storage class bucket to a nearline bucket after 90 days. There may be times, however, when you would like to manually change a bucket's storage class. In those cases, you can use the `gsutil rewrite` command and specify the `-s` flag. Here's an example:

```
gsutil rewrite -s [STORAGE_CLASS] gs://[PATH_TO_OBJECT]
```

Here, `[STORAGE_CLASS]` is the new storage class.

Another common task with Cloud Storage is moving objects between buckets. You can do this using the `gsutil mv` command. The form of the command is as follows:

```
gsutil mv gs://[SOURCE_BUCKET_NAME]/[SOURCE_OBJECT_NAME] \
gs://[DESTINATION_BUCKET_NAME]/[DESTINATION_OBJECT_NAME]
```

Here, `[SOURCE_BUCKET_NAME]` and `[SOURCE_OBJECT_NAME]` are the original bucket name and filename, and `[DESTINATION_BUCKET_NAME]` and `[DESTINATION_OBJECT_NAME]` are the target bucket and filename, respectively.

The move command can also be used to rename an object, similar to the `mv` command in Linux. For an object in Cloud Storage, you can use this command:

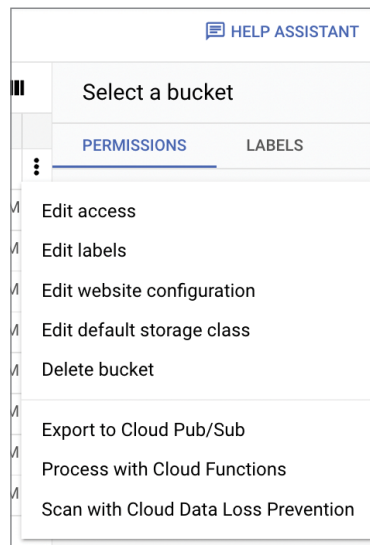
```
gsutil mv gs://[BUCKET_NAME]/[OLD_OBJECT_NAME] \
gs://[BUCKET_NAME]/[NEW_OBJECT_NAME]
```

You can also use the console to perform an array of operations (see Figure 12.37), including:

- Editing access
- Editing labels
- Deleting a bucket
- Exporting to Cloud Pub/Sub
- Processing with Cloud Functions
- Scanning with Cloud Data Loss Protection service

Google Cloud has added a `gcloud storage` command to the `gcloud` utility. It has similar functionality to `gsutil` and is generally faster for both uploads and downloads.

FIGURE 12.37 Operations you can perform on buckets in Cloud Storage



Summary

In this chapter, you learned how to perform basic deployment and management tasks for a number of Google Cloud services, including Cloud SQL, Cloud Datastore, BigQuery, Bigtable, Cloud Spanner, Cloud Pub/Sub, Cloud Dataproc, and Cloud Storage. You have seen how to use the console and command-line tools. While `gcloud` is often used, several of the services have their own command-line tools. There was some discussion of how to create database structures, insert data, and query that data in the various database services. We also discussed basic Cloud Storage management operations, such as moving and renaming objects.

Exam Essentials

Understand how to initialize Cloud SQL and Cloud Spanner. Cloud SQL and Cloud Spanner are the two managed relational databases for transaction processing systems. BigQuery is an analytical database designed for data warehouse and analytics. Understand the need to create databases and tables. Know that SQL is used to query these databases.

Understand how to initialize Cloud Firestore and Cloud Bigtable. These are two NoSQL offerings. You can add small amounts of data to Cloud Firestore through the console and query it with a SQL-like language called GQL. Cloud Bigtable is a wide-column database that does not support SQL. Bigtable is managed with the `cbt` command-line tool.

Know how to export data from BigQuery, estimate the cost of a query, and monitor jobs in BigQuery. BigQuery is designed to work with petabyte-scale data warehouses. SQL is used to query data. Know how to export data using the console. Understand that the `bq` command line, not `gcloud`, is the tool for working with BigQuery from the command line.

Know how to convert Cloud Storage bucket storage classes. Life cycle policies can change storage classes of buckets when events occur, such as a period of time passes. Know that `gsutil rewrite` is used to change the storage class of a bucket interactively. Know how to use the console and the command line to move and rename objects.

Understand that Pub/Sub is a message queue. Applications write data to topics, and applications receive messages through subscriptions to topics. Subscriptions can be push or pull. Unread messages have a retention period after which they are deleted.

Understand that Cloud Dataproc is a managed Spark and Hadoop service. These platforms are used for big data analytics, machine learning, and large-scale batch jobs, such as large volume extraction, transformation, and load operations. Spark is a good option for analyzing transaction data, but data must be loaded into Spark from its source system.

Know the four command-line tools: `gcloud`, `gsutil`, `bq`, and `cbt`. `gcloud` is used for most products but not all. `gsutil` and the newer `gcloud` storage commands are used to work with Cloud Storage from the command line. If you want to work with BigQuery from the command line, you need to use `bq`. To work with Bigtable, you use the `cbt` command.

Review Questions

You can find the answers in the Appendix.

1. Cloud SQL is a fully managed relational database service, but database administrators still have to perform some tasks. Which of the following tasks do Cloud SQL users need to perform?
 - A. Applying security patches
 - B. Performing regularly scheduled backups
 - C. Creating databases
 - D. Tuning the operating system to optimize Cloud SQL performance
2. Which of the following commands is used to create a backup of a Cloud SQL database?
 - A. `gcloud sql backups create`
 - B. `gsutil sql backups create`
 - C. `gcloud sql create backups`
 - D. `gcloud sql backups export`
3. Which of the following commands will run an automatic backup at 3:00 a.m. on an instance called `ace-exam-mysql`?
 - A. `gcloud sql instances patch ace-exam-mysql \`
`--backup-start-time 03:00`
 - B. `gcloud sql databases patch ace-exam-mysql \`
`--backup-start-time 03:00`
 - C. `cbt sql instances patch ace-exam-mysql \`
`--backup-start-time 03:00`
 - D. `bq gcloud sql instances patch ace-exam-mysql \`
`--backup-start-time 03:00`
4. What is the query language used by Firestore in Datastore mode?
 - A. SQL
 - B. MDX
 - C. GQL
 - D. DataFrames
5. What is the correct command-line structure to export data from Firestore?
 - A. `gcloud firestore export collection gs://[BUCKET_NAME]`
 - B. `gcloud firestore dump collection gs://[BUCKET_NAME]`
 - C. `gcloud firestore export gs://[BUCKET_NAME]`
 - D. `gcloud firestore dump gs://[BUCKET_NAME]`

6. When you enter a query into the BigQuery query form, BigQuery analyzes the query and displays an estimate of what metric?
 - A. Time required to enter the query
 - B. Cost of the query
 - C. Amount of data scanned
 - D. Number of bytes passed between servers in the BigQuery cluster
7. You want to get an estimate of the volume of data scanned by BigQuery from the command line. Which option shows the command structure you should use?
 - A. `gcloud BigQuery query estimate [SQL_QUERY]`
 - B. `bq --location=[LOCATION] query --use_legacy_sql=false \ --dry_run [SQL_QUERY]`
 - C. `gsutil --location=[LOCATION] query --use_legacy_sql=false \ --dry_run [SQL_QUERY]`
 - D. `cbt BigQuery query estimate [SQL_QUERY]`
8. You are using Cloud Console and want to check on some jobs running in BigQuery. You navigate to the BigQuery part of the console. Which menu item would you click to view jobs?
 - A. Personal History or Project History.
 - B. Active Jobs.
 - C. My Jobs.
 - D. You can't view job status in the console; you have to use `bq` on the command line.
9. You want to estimate the cost of running a BigQuery query. What two services within Google Cloud will you need to use?
 - A. BigQuery and Billing
 - B. Billing and Pricing Calculator
 - C. BigQuery and Pricing Calculator
 - D. Billing and `bq` command
10. You have just created a Cloud Spanner instance. You have been tasked with creating a way to store data about a product catalog. What is the next step after creating a Cloud Spanner instance that you would perform to enable you to load data?
 - A. Run `gcloud spanner update-security-patches`.
 - B. Create a database within the instance.
 - C. Create tables to hold the data.
 - D. Use the Cloud Spanner console to import data into tables created with the instance.
11. You have created a Cloud Spanner instance and database. According to Google best practices, how often should you update VM packages using `apt-get`?
 - A. Every 24 hours.
 - B. Every 7 days.
 - C. Every 30 days.
 - D. Never; Cloud Spanner is a managed service.

12. Your software team is developing a distributed application and wants to send messages from one application to another. Once the consuming application reads a message, it should be deleted. You want your system to be robust to failure, so messages should be available for at least three days before they are discarded. Which Google Cloud service is best designed to support this use case?
- A. Bigtable
 - B. Dataproc
 - C. Cloud Pub/Sub
 - D. Cloud Spanner
13. Your manager asks you to set up a bare-bones Pub/Sub system as a sandbox for new developers to learn about messaging systems. What are the two resources within Pub/Sub you will need to create?
- A. Topics and tables
 - B. Topics and databases
 - C. Topics and subscriptions
 - D. Tables and subscriptions
14. Your company is launching an IoT service and will receive large volumes of streaming data. You have to store this data in Bigtable. You want to explore the Bigtable environment from the command line. What command would you run to ensure you have command-line tools installed?
- A. `apt-get install bigtable-tools`
 - B. `apt-get install cbt`
 - C. `gcloud components install cbt`
 - D. `gcloud components install bigtable-tools`
15. You need to create a table called `iot-ingest-data` in Bigtable. What command would you use?
- A. `cbt createtable iot-ingest-data`
 - B. `gcloud bigtable tables create ace-exam-bt-table`
 - C. `gcloud bigtable create tables ace-exam-bt-table`
 - D. `gcloud create ace-exam-bt-table`
16. Cloud Dataproc is a managed service for which two big data platforms?
- A. Spark and Cassandra
 - B. Spark and Hadoop
 - C. Hadoop and Cassandra
 - D. Spark and TensorFlow

17. Your department has been asked to analyze large batches of data every night. The jobs will run for about three to four hours. You want to shut down resources as soon as the analysis is done, so you decide to write a script to create a Dataproc cluster every night at midnight. What command would you use to create a cluster called `spark-nightly-analysis` in the `us-west2-a` zone?
- A. `bq dataproc clusters create spark-nightly-analysis \`
`--zone us-west2-a`
 - B. `gcloud dataproc clusters create spark-nightly-analysis \`
`--zone us-west2-a`
 - C. `gcloud dataproc clusters spark-nightly-analysis \`
`--zone us-west2-a`
 - D. None of the above
18. You have a number of buckets containing old data that is hardly ever used. You don't want to delete it, but you want to minimize the cost of storing it. You decide to change the storage class to Coldline for each of those buckets. What is the command structure that you would use?
- A. `gcloud rewrite -s [STORAGE_CLASS] gs://[PATH_TO_OBJECT]`
 - B. `gsutil rewrite -s [STORAGE_CLASS] gs://[PATH_TO_OBJECT]`
 - C. `cbt rewrite -s [STORAGE_CLASS] gs://[PATH_TO_OBJECT]`
 - D. `bq rewrite -s [STORAGE_CLASS] gs://[PATH_TO_OBJECT]`
19. You want to rename an object stored in a bucket. What command structure would you use?
- A. `gsutil cp gs://[BUCKET_NAME]/[OLD_OBJECT_NAME] \`
`gs://[BUCKET_NAME]/[NEW_OBJECT_NAME]`
 - B. `gsutil mv gs://[BUCKET_NAME]/[OLD_OBJECT_NAME] \`
`gs://[BUCKET_NAME]/[NEW_OBJECT_NAME]`
 - C. `gsutil mv gs://[OLD_OBJECT_NAME] gs://[NEW_OBJECT_NAME]`
 - D. `gcloud mv gs://[OLD_OBJECT_NAME] gs://[NEW_OBJECT_NAME]`
20. An executive in your company emails you asking about creating a recommendation system that will help sell more products. The executive has heard there are some Google Cloud solutions that may be good fits for this problem. What Google Cloud service would you recommend the executive look into?
- A. Cloud Dataproc, especially Spark and its machine learning library
 - B. Cloud Dataproc, especially Hadoop
 - C. Cloud Spanner, which is a global relational database that can hold a lot of data
 - D. Cloud SQL, because SQL is a powerful query language

Chapter 13

Loading Data into Storage

**THIS CHAPTER COVERS THE FOLLOWING
OBJECTIVES OF THE GOOGLE ASSOCIATE
CLOUD ENGINEER CERTIFICATION EXAM:**

- ✓ 3.4 Deploying and implementing data solutions





In this chapter, we will delve into the details of loading and moving data into various storage and processing systems in Google Cloud. We'll start by explaining how to load and move data in Cloud Storage using the console and the command line.

The bulk of the chapter will describe how to import and export data into data storage and analysis services, including Cloud SQL, Cloud Firestore, BigQuery, Cloud Spanner, Cloud Bigtable, and Cloud Dataproc. The chapter wraps up with a look into streaming data into Cloud Pub/Sub.

Loading and Moving Data to Cloud Storage

Cloud Storage is used for a variety of storage use cases, including long-term storage and archiving, file transfers, and data sharing. This section describes how to create storage buckets, load data into storage buckets, and move objects between storage buckets.



Google Cloud recently introduced the `gcloud storage` command which has similar functionality as `gsutil`. `gcloud storage` is generally more performant than `gsutil`.

Loading and Moving Data to Cloud Storage Using the Console

Loading data into Cloud Storage is a common task that's easily done using Cloud Console.

Navigate to the Cloud Storage page of Cloud Console. You will see a list of existing buckets and an option to create a new bucket. Figure 13.1 shows a listing of buckets and the Create Bucket button above the list.

When you create a bucket, you are prompted to specify a name and a location where you want to store your data, as shown in Figure 13.2. The bucket name must be globally unique. The location can be Multi-Region for highest availability and highest cost, Dual-Region for high availability and low latency across two regions, or Region, which has the lowest latency within a single region. If you choose Multi-Region, your options include United States, Europe, and Asia Pacific (see the console for the latest list of multi-regions). If you choose Dual-Region, you can specify two regions within a continent, with the current options being United States, Europe, and Asia Pacific. If you choose Region, then you can choose any one of the regions available.

FIGURE 13.1 The first step in loading data into Cloud Storage is to create a bucket.

Buckets						
		+ CREATE	↻ REFRESH	HELP ASSISTANT	LEARN	
<div><div>Filter</div><div>Filter buckets</div></div>						
<input type="checkbox"/>	Name ↑	Created	Location type	Location	Default storage class	
<input type="checkbox"/>	dataflow-staging-us-west1-38894734...	Oct 23, 2022, 11:06:45 AM	Region	us-west1	Standard	⋮
<input type="checkbox"/>	gcf-sources-388947348090-us-central1	Nov 19, 2022, 9:52:45 AM	Region	us-central1	Standard	⋮
<input type="checkbox"/>	slg-cloud-storage-2	Oct 26, 2022, 6:55:12 AM	Region	us-west1	Standard	⋮
<input type="checkbox"/>	slg-cloud-storage-data-transfer-1	Oct 23, 2022, 8:19:09 AM	Region	us-east1	Standard	⋮
<input type="checkbox"/>	slg-ml-training-data	Aug 14, 2022, 6:24:01 PM	Region	us-west1	Standard	⋮
<input type="checkbox"/>	us.artifacts.scenic-energy-335022.ap...	Nov 19, 2022, 9:53:20 AM	Multi-region	us	Standard	⋮

FIGURE 13.2 Defining a regional bucket in us-west1

☒

Name your bucket

Pick a globally unique, permanent name. [Naming guidelines](#)

ace-exam-test-bucket

Tip: Don't include any sensitive information

✓ LABELS (OPTIONAL)

CONTINUE

•

Choose where to store your data

This permanent choice defines the geographic placement of your data and affects cost, performance, and availability. [Learn more](#)

Location type

☐ Multi-region

Highest availability across largest area

☐ Dual-region

High availability and low latency across 2 regions

☒ Region

Lowest latency within a single region

us-west1 (Oregon)


CONTINUE




Remember that buckets are regional resources, and buckets are replicated across zones in the region.

Next, you will need to choose your storage class (see Figure 13.3). The options are Standard, which is best for short-term storage and frequently accessed objects; Nearline for objects accessed less than once every 30 days; Coldline storage for objects accessed less than once every 90 days; and Archive, which is used for objects accessed less than once per year.

FIGURE 13.3 Choosing a storage class and access control method

 **Choose a default storage class for your data**

A storage class sets costs for storage, retrieval, and operations. Pick a default storage class based on how long you plan to store your data and how often it will be accessed. [Learn more](#)


☒ **Standard** 
Best for short-term storage and frequently accessed data

☐ **Nearline**
Best for backups and data accessed less than once a month

☐ **Coldline**
Best for disaster recovery and data accessed less than once a quarter

☐ **Archive**
Best for long-term digital preservation of data accessed less than once a year

[CONTINUE](#)

 **Choose how to control access to objects**

Prevent public access

Restrict data from being publicly accessible via the internet. Will prevent this bucket from being used for web hosting. [Learn more](#)

☐ Enforce public access prevention on this bucket

Access control

☒ **Uniform**
Ensure uniform access to all objects in the bucket by using only bucket-level permissions (IAM). This option becomes permanent after 90 days. [Learn more](#)

☐ **Fine-grained**
Specify access to individual objects by using object-level permissions (ACLs) in addition to your bucket-level permissions (IAM). [Learn more](#)

[CONTINUE](#)

Here you will also need to decide how you want to control access to the bucket. Since buckets are web addressable, you can allow anyone with the URL to your bucket to access the contents of that bucket. Google Cloud gives you the option of explicitly preventing this kind of public access by providing the Enforce Public Access Prevention On This Bucket option.

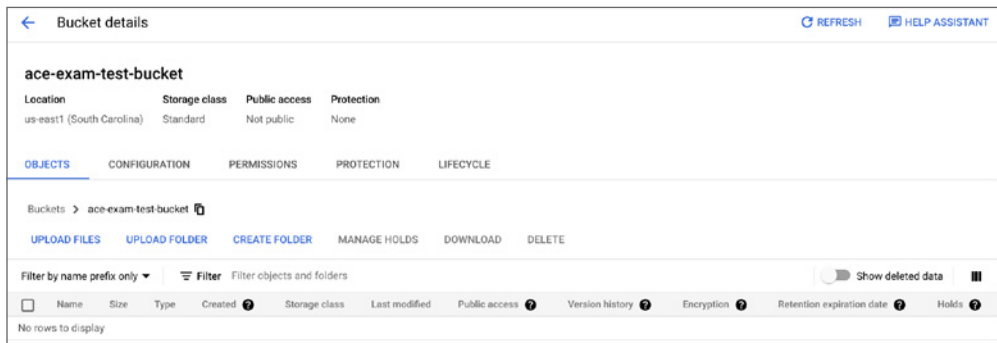
Google Cloud originally used access control lists on buckets to manage access to buckets. This is now called the Fine-grained access option, and it allows you to specify access controls on individual objects as well as on buckets. Although fine-grained access is still available, the preferred option is to use uniform access, in which access to objects in the bucket are controlled by bucket-level permissions managed by the IAM service. Uniform access control is the default and using it is considered a best practice.



Google Cloud certification exams may test you on your knowledge of Google recommended practices. Using uniform access control instead of fine-grained access control is one of those recommended practices.

After you create a bucket, you can view the bucket's details, as shown in Figure 13.4.

FIGURE 13.4 The Bucket Details page shows information on Objects, Configuration, Permissions, Protection, and Lifecycle.



When you click Upload Files, you are prompted to do so using your client device's filesystem. When you upload a folder, you are also prompted by your local operating system tools (see Figure 13.5).

It's easy to move objects between buckets. Just click the ellipsis at the end of a line to display a list of operations, which includes Move. Clicking Move will open the page shown in Figure 13.6.

When moving an object, you are prompted for a destination bucket and folder, as shown in Figure 13.7.

FIGURE 13.5 Upload Files prompts you for a folder using the client device's filesystem tools.

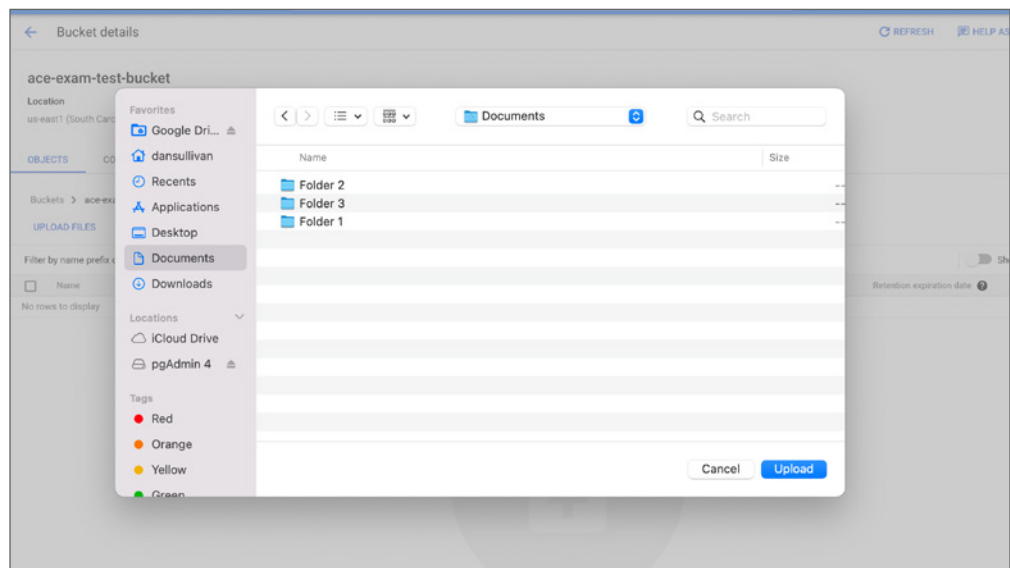


FIGURE 13.6 Objects can be moved by using the move command in the Operations menu.

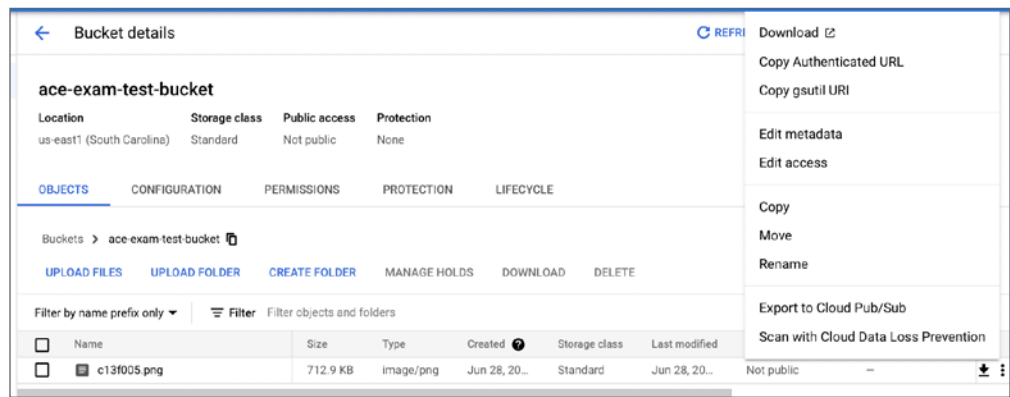


FIGURE 13.7 When moving an object in the console, you will be prompted for a destination bucket and folder.

Move object

Source
ace-exam-test-bucket/c13f005.png

Destination * BROWSE

☐ Keep source permissions ?

☒ Use default permissions at destination ?

[GSUTIL EQUIVALENT](#)

Loading and Moving Data to Cloud Storage Using the Command Line

Loading and moving data can be done in the command line using the `gsutil` command.

To create a bucket, use the `gsutil mb` command. `mb` is short for “make bucket.”

```
gsutil mb gs://[BUCKET_NAME]/
```

Keep in mind that bucket names must be globally unique. To create a bucket named `ace-exam-bucket1`, use the following command:

```
gsutil mb gs://ace-exam-bucket1/
```

To upload a file from your local device or a Google Cloud virtual machine (VM), you can use the `gsutil cp` command to copy files. The command is as follows:

```
gsutil cp [LOCAL_OBJECT_LOCATION] gs://[DESTINATION_BUCKET_NAME]/
```

For example, to copy a file called `README.txt` from `/home/mydir` to the bucket `ace-exam-bucket1`, you’d execute the following command from your client device command line:

```
gsutil cp /home/mydir/README.txt gs://ace-exam-bucket1/
```

Similarly, if you’d like to download a copy of your data from a Cloud Storage bucket to a directory on a VM, you could log into the VM using SSH and issue a command such as this:

```
gsutil cp gs://ace-exam-bucket1/README.txt /home/mydir/
```

In this example, the source object is on Cloud Storage, and the target file is on the VM from which you are running the command.

The `gsutil` tool has a `move` command; its structure is as follows:

```
gsutil mv gs://[SOURCE_BUCKET_NAME]/[SOURCE_OBJECT_NAME] \
          gs://[DESTINATION_BUCKET_NAME]/[DESTINATION_OBJECT_NAME]
```

To move the `README.txt` file from `ace-exam-bucket1` to `ace-exam-bucket2` and keep the same filename, you'd use this command:

```
gsutil mv gs://ace-exam-bucket1/README.txt gs://ace-exam-bucket2/
```

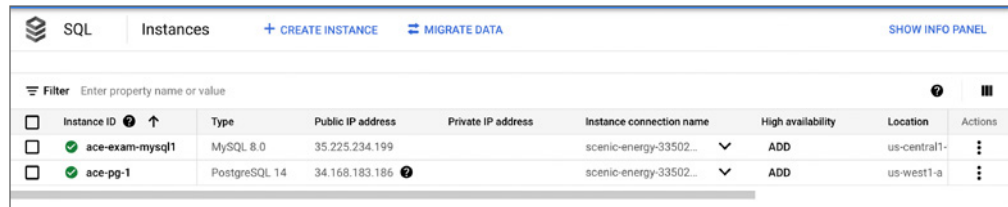
Importing and Exporting Data

As a Cloud Engineer, you may need to perform bulk data operations, such as importing and exporting data from databases. These operations are done with command-line tools and sometimes the console. We will not look into how to programmatically insert data into databases; that is more of an application developer and database administrator task.

Importing and Exporting Data: Cloud SQL

To export a Cloud SQL database using the console, navigate to the Cloud SQL page of the console to list database instances, as shown in Figure 13.8.

FIGURE 13.8 Listing of database instances on the Cloud SQL page of the console



The screenshot shows the Google Cloud SQL console interface. At the top, there's a navigation bar with the 'SQL' icon, the 'Instances' tab selected, and buttons for '+ CREATE INSTANCE' and 'MIGRATE DATA'. A 'SHOW INFO PANEL' link is on the right. Below the navigation bar is a filter input field labeled 'Filter' with the placeholder text 'Enter property name or value'. The main content area is a table listing database instances. The table has columns for Instance ID, Type, Public IP address, Private IP address, Instance connection name, High availability, Location, and Actions. Two instances are listed: 'ace-exam-mysql1' (MySQL 8.0) and 'ace-pg-1' (PostgreSQL 14). Both instances have a public IP address and are located in 'us-central1'.

Instance ID	Type	Public IP address	Private IP address	Instance connection name	High availability	Location	Actions
ace-exam-mysql1	MySQL 8.0	35.225.234.199		scenic-energy-33502...	ADD	us-central1-	
ace-pg-1	PostgreSQL 14	34.168.183.186		scenic-energy-33502...	ADD	us-west1-a	

Open the Instance Details page by double-clicking the name of the instance (see Figure 13.9).

Select the Export tab to open the Export Data page. You will need to specify a bucket in which to store the backup file (see Figure 13.10).

You will also need to choose SQL or CSV output. The SQL output is useful if you plan to import the data to another relational database. CSV is a good choice if you need to move this data into a nonrelational database or other tool that is not a relational database.

After you create an export file, you can import it. Follow the same instructions as for exporting, but choose the Import option instead of the Export option. This will display the options shown in Figure 13.11. Specify the source file, the file format, and the database to import the data to.

FIGURE 13.9 The Instance Details page has Import and Export tabs.

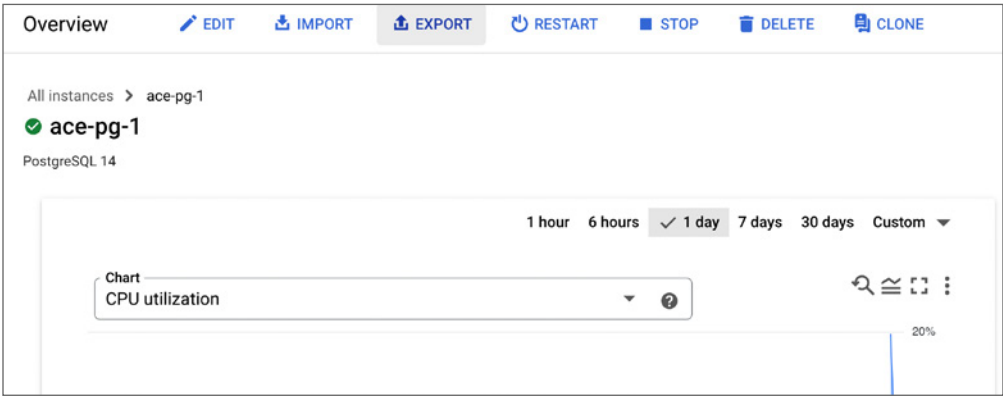


FIGURE 13.10 Exporting a database requires you to specify a bucket for storing the export file and a file format.

←

Export data to Cloud Storage

Source

Choose the format for your export, and the data you'd like to export from this instance.
[Learn more](#)

File format

☒ SQL

A plain text file with a sequence of SQL commands, like the output of `pg_dump`.

☐ CSV

Exports a plain text file with one line per row and comma-separated fields. Requires SQL `SELECT` query.

Data to export

Choose a database from this instance to export.

Database *

▼

☐ Offload export to a temporary instance

Makes your export serverless to reduce strain on the source instance, allowing you to perform other operations while the export is in progress. Affects cost. [Learn more](#)

Destination

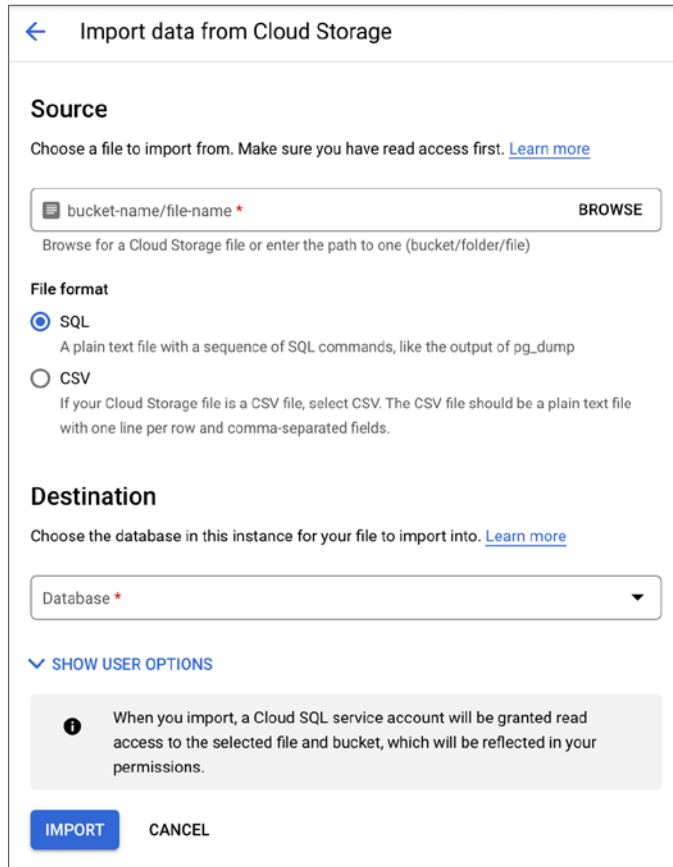
Choose a Cloud Storage location to export into. Make sure you have the required permissions. [Learn more](#)

bucket-name/file-name *

BROWSE

Browse for a Cloud Storage location or enter the path to one

FIGURE 13.11 Importing a database requires you to specify a path to the bucket and object storing the export file, a file format, and a target database within the instance.



← Import data from Cloud Storage

Source

Choose a file to import from. Make sure you have read access first. [Learn more](#)

bucket-name/file-name * BROWSE

Browse for a Cloud Storage file or enter the path to one (bucket/folder/file)

File format

☒ SQL
A plain text file with a sequence of SQL commands, like the output of pg_dump

☐ CSV
If your Cloud Storage file is a CSV file, select CSV. The CSV file should be a plain text file with one line per row and comma-separated fields.

Destination

Choose the database in this instance for your file to import into. [Learn more](#)

Database *

▼ SHOW USER OPTIONS

i When you import, a Cloud SQL service account will be granted read access to the selected file and bucket, which will be reflected in your permissions.

IMPORT CANCEL

You can also create, import, and export a database using the command line. Use the `gsutil` command to create a bucket:

```
gsutil mb gs://ace-exam-bucket1/
```

You need to ensure that the service account can write to the bucket, so get the name of the service account by describing the instance with the following command:

```
gcloud sql instances describe [INSTANCE_NAME]
```

In this example, this command would be as follows:

```
gcloud sql instances describe ace-exam-mysql1
```

This command will produce a detailed listing about the instance. See Figure 13.12 for an example of the output.

FIGURE 13.12 Details about a database instance generated by the `gcloud sql instances describe` command

```
dan@cloudshell:~ (scenic-energy-335022)$ gcloud sql instances describe ace-exam-mysql1
backendType: SECOND_GEN
connectionName: scenic-energy-335022:us-central1:ace-exam-mysql1
createTime: '2022-06-30T00:32:53.562Z'
databaseInstalledVersion: MYSQL_8_0_26
databaseVersion: MYSQL_8_0
etag: 152a1dc634a450414a1e93798b9d00074b6634d07ac4f4aa789d8b45e39ba251
gceZone: us-central1-f
instanceType: CLOUD_SQL_INSTANCE
ipAddresses:
- ipAddress: 35.225.234.199
  type: PRIMARY
kind: sql#instance
maintenanceVersion: MYSQL_8_0_26.R20220508.01_03
name: ace-exam-mysql1
project: scenic-energy-335022
region: us-central1
selfLink: https://sqladmin.googleapis.com/sql/v1beta4/projects/scenic-energy-335022/instances/ace-exam-mysql1
serverCaCert:
  cert: |-
    -----BEGIN CERTIFICATE-----
    MIIDfzCCAmegAwIBAgIBADANBgkqhkiG9w0BAQsFADB3MS0wKwYDVQQeYyRhZWNm
    ZTVhNC0zYzowLT04NjAtYTMwOC1hYzNiYzJiMzoxODYxTzAhBgNVBAMTGkdub2ds
```

You can create an export of a database using this command:

```
gcloud sql export sql [INSTANCE_NAME]
                        gs://[BUCKET_NAME]/[FILE_NAME] \
                        --database=[DATABASE_NAME]
```

For example, the following command will export the MySQL database to a SQL dump file written to the `ace-exam-bucket1` bucket:

```
gcloud sql export sql ace-exam-mysql1 \
                        gs://ace-exam-bucket1/ace-exam-mysqlexport.sql \
                        --database=mysql
```

If you prefer to export to a CSV file, you will change `sql` to `csv` in the previous command. Here's an example:

```
gcloud sql export csv ace-exam-mysql1 \
                        gs://ace-exam-bucket1/ace-exam-mysql-export.csv \
                        --database=mysql
```

Importing to a database uses a similarly structured command:

```
gcloud sql import sql [INSTANCE_NAME] \
                        gs://[BUCKET_NAME]/[IMPORT_FILE_NAME] \
                        --database=[DATABASE_NAME]
```

Using the example database, bucket, and export file, you can import the file using this command:

```
gcloud sql import sql ace-exam-mysql1 \
                        gs://ace-exam-bucket1/ace-exam-mysql-export.sql \
                        --database=mysql
```

Importing and Exporting Data: Cloud Firestore

To export data from Cloud Firestore in Native mode, you can use this command:

```
gcloud firestore export gs://{BUCKET}
```

Importing and exporting data from Firestore in Datastore mode is done through the command line. Datastore mode uses a namespace data structure to group entities that are exported. You will need to specify the name of the namespace used by the entities you are exporting. The default namespace is simply `(default)`.

The Cloud Datastore export command is as follows:

```
gcloud datastore export --namespaces="(default)" gs://{BUCKET}
```

You can export to a bucket called `ace-exam-datastore1` using this command:

```
gcloud datastore export --namespaces="(default)" gs://ace-exam-datastore1
```

The Cloud Datastore import command is as follows:

```
gcloud datastore import gs://{BUCKET}/[PATH]/[FILE].overall_export_metadata
```

The export process will create a folder named `ace-exam-datastore1` using the data and time of the export. The folder will contain a metadata file and a folder containing the exported data. The metadata filename will use the same date and time used for the containing folder. The data folder will be named after the namespace of the exported Datastore database. An example import command is as follows:

```
gcloud datastore import gs://ace-exam-datastore1/2018-12-20T19:13:55_64324/2018-12-20T19:13:55_64324.overall_export_metadata
```

Importing and Exporting Data: BigQuery

BigQuery users can export and import tables using Cloud Console and the command line.

To export a table using the console, navigate to the BigQuery console interface. Under Resources, open the data set containing the table you want to export. Click the table name to list the table description, as shown in Figure 13.13. Notice the Export option in the upper right.

At the far right, click Export to display a list of four export locations: Google Sheets, Google Cloud Storage, Looker Studio (formerly Data Studio), which is an analysis tool in Google Cloud, or Scanning with Data Loss Prevention service (see Figure 13.14).

Selecting Cloud Storage displays the options shown in Figure 13.15. Enter the bucket name in which you want to store the export file. Choose a file format. The options are CSV, Avro, and JSON. Choose a compression type. The options are None or Gzip for CSV and Deflate and Snappy for Avro.

FIGURE 13.13 Detailed list of a BigQuery table

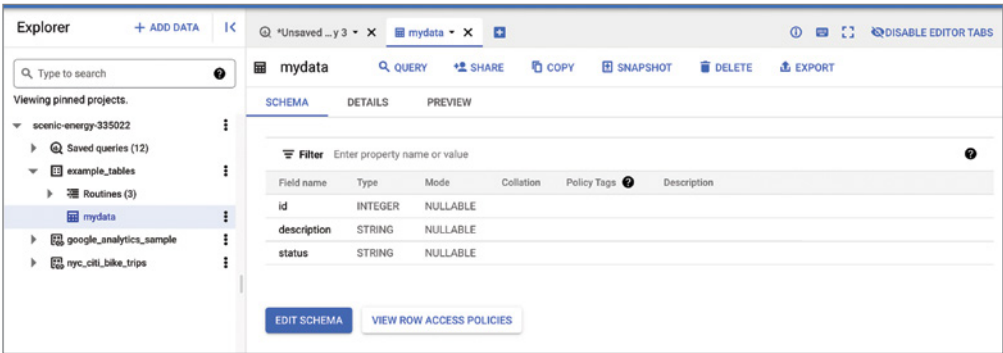
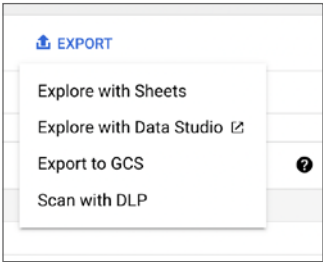


FIGURE 13.14 Choosing a target location for a BigQuery export



File Formats

BigQuery offers several export file options. CSV, short for comma-separated values, is a human-readable format suitable for small data sets that will be imported into tools that only support the CSV format. CSV is not optimized for storage, so it does not compress or use a more efficient encoding than text. It's not the best option when exporting large data sets.

JSON is also a human-readable format that has advantages and disadvantages similar to CSV. One difference is that JSON includes schema information with each record, whereas CSV uses an optional header row with column names at the beginning of the file to describe the schema.

Gzip is a widely used lossless compression utility.

Avro is a compact binary format that supports complex data structures. When data is saved in the Avro format, a schema is written to the file along with data. Schemas are defined in JSON. Avro is a good option for large data sets, especially when importing data into other applications that read the Avro format, including Apache Spark, which is available as a managed service in Cloud Dataproc. Avro files can be compressed using either the deflate or the Snappy utility. Deflate produces smaller compressed files, but Snappy is faster.

FIGURE 13.15 Specifying the output parameters for a BigQuery export operation

Export table to Google Cloud Storage

GCS Location * ☒ ace-exam-test-bucket/ace-test-export1 BROWSE ?

Export format * Avro

Compression * None

SAVE CANCEL

To export data from the command line, use the `bq extract` command. The structure is as follows:

```
bq extract --destination_format [FORMAT] --compression
[COMPRESSION_TYPE] --field_delimiter [DELIMITER] --print_header
[BOOLEAN] [PROJECT_ID]:[DATASET].[TABLE] gs://[BUCKET]/[FILENAME]
```

Here's an example:

```
bq extract --destination_format CSV --compression GZIP 'mydataset.mytable' \
gs://example-bucket/myfile.zip
```



Remember, the command-line tool for working with BigQuery is `bq`, not `gcloud`.

To import data into BigQuery, navigate to the BigQuery console page and select a dataset you'd like to import data into. Click a dataset and then select **Create Table**, as shown in Figure 13.16.

The **Create Table** page has several parameters, including an optional source table, a destination project, the dataset name, the table type, and the table name (see Figure 13.17).

The **Create Table From** field indicates where to find the source data, if any. This field provides a way to create a table based on data in an existing table, but defaults to an empty table (see Figure 13.18).

You will also need to specify the file format of the file that will be imported. The options include CSV, JSONL (Newline Delimited JSON), Avro, Parquet, ORC, and Cloud Datastore Backup (see Figure 13.19).

FIGURE 13.16 When viewing a data set, you have the option to create a table.

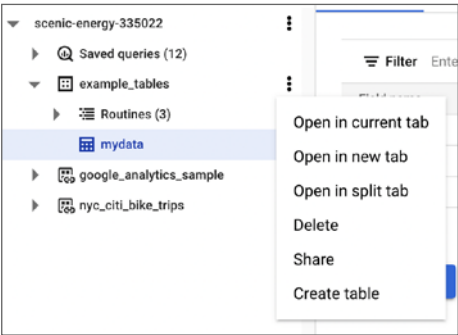


FIGURE 13.17 Creating a table in BigQuery

Create table

Source

Create table from
Empty table

Destination

Project *
scenic-energy-335022

Dataset *
example_tables

Table *
mydata2
Unicode letters, marks, numbers, connectors, dashes or spaces allowed.

Table type
Native table

Schema

☐ Edit as text

+

CREATE TABLE

CANCEL

FIGURE 13.18 Data can be imported from multiple kinds of locations.

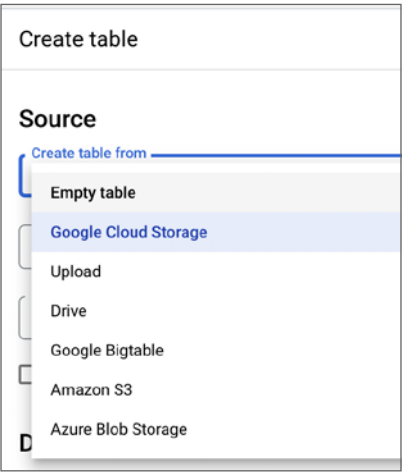
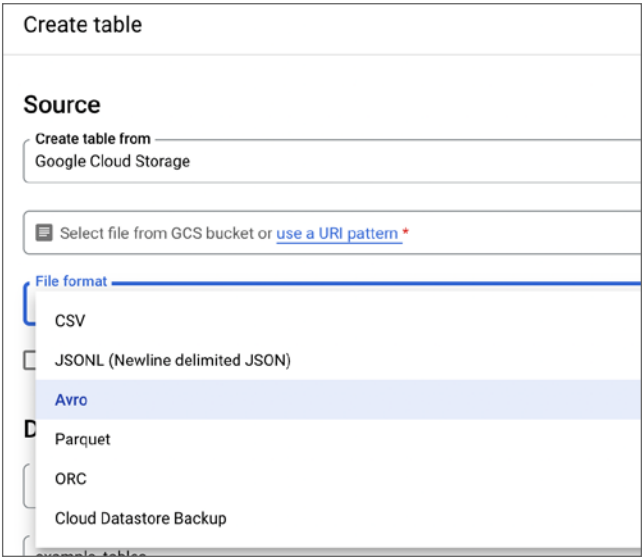


FIGURE 13.19 File format options for importing



Provide destination information, including project, data set name, table type, and table name. Table type may be Native type or an external table. If the table is external, the data is kept in the source location, and only metadata about the table is stored in BigQuery. This type is used when you have large data sets and do not want to load them all into BigQuery.

After specifying all parameters, click Create Table to create the table and load the data. To load data from the command line, use the `bq load` command. Its structure is as follows:

```
bq load --autodetect --source_format=[FORMAT] [DATASET].[TABLE] \
[PATH_TO_SOURCE]
```

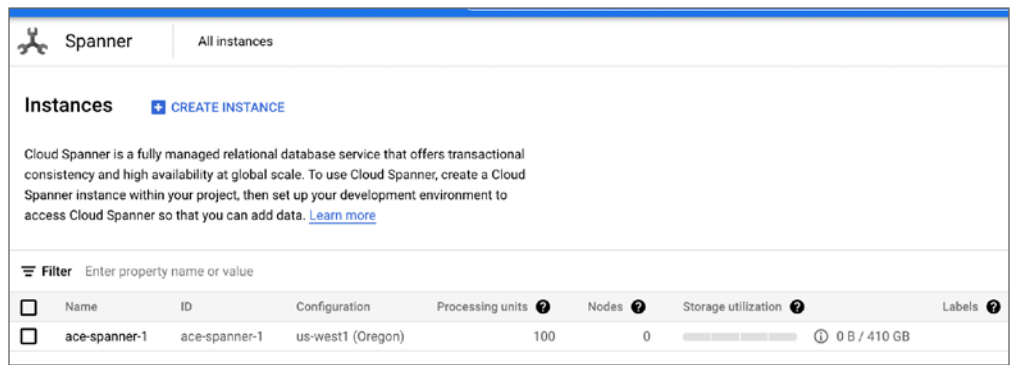
The `--autodetect` parameter has `bq load` automatically detect the table schema from the source file. An example command is as follows:

```
bq load --autodetect --source_format=CSV mydataset.mytable \
gs://ace-exam-biquery/mydata.csv
```

Importing and Exporting Data: Cloud Spanner

Cloud Spanner users can import and export data using Cloud Console. To export data from Cloud Spanner, navigate to the Cloud Spanner section of the console. You will see a list of Spanner instances, as shown in Figure 13.20.

FIGURE 13.20 Listing of Spanner instances



Click the name of the instance that is the source of data to export. This will show the Instance Details page (see Figure 13.21). Click Export to display the Export options, as shown in Figure 13.22. You will need to enter a destination bucket, the database to export, and a region to run the job. Notice that you must confirm that you understand there will be charges for running Cloud Dataflow and that there may be data egress charges for data sent between regions. To import data, click Import to display the Import page (see Figure 13.23). You will need to specify a source bucket, a destination database, and a region to run a job.

FIGURE 13.21 Import/Export page

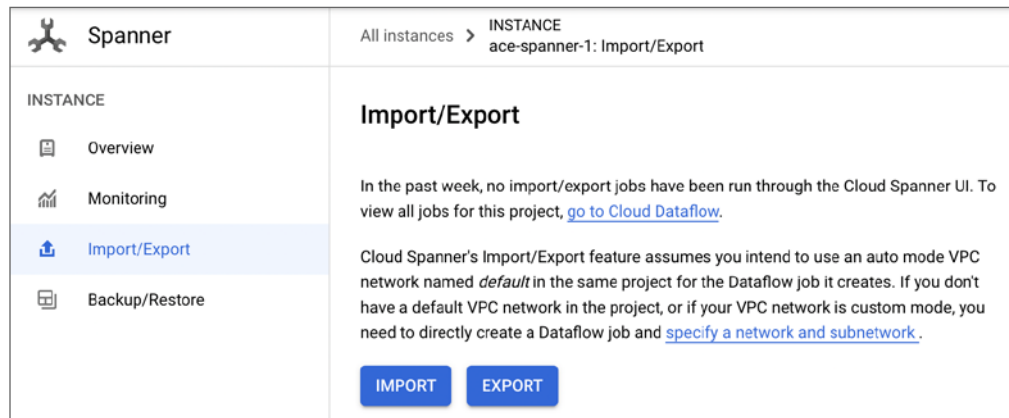


FIGURE 13.22 Export options for Cloud Spanner

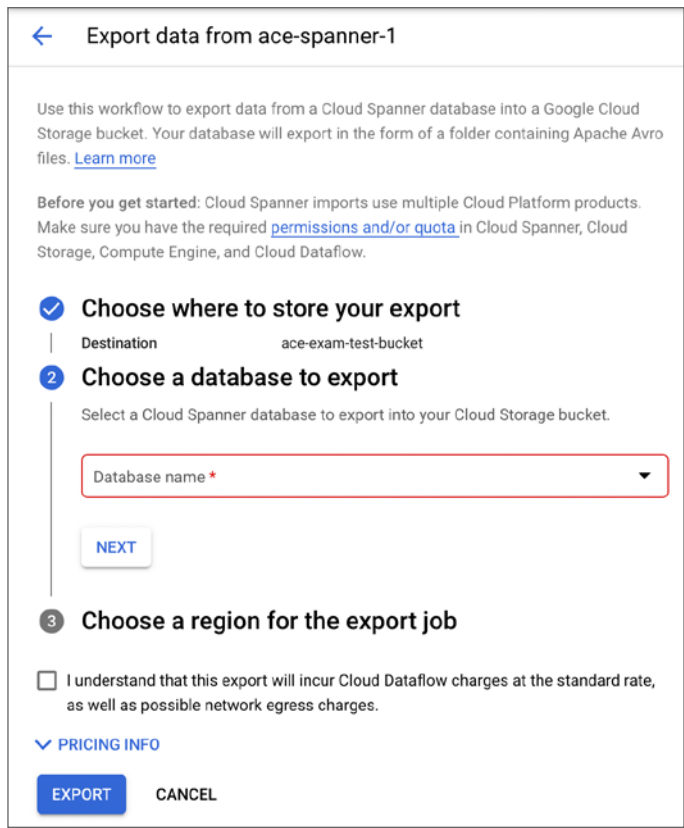


FIGURE 13.23 Import options for Cloud Spanner

← Import data into ace-spanner-1

Use this workflow to import data from a previous Cloud Spanner export. To import other data, use the Cloud Dataflow API. [Learn more](#)

Before you get started: Cloud Spanner imports use multiple Cloud Platform products. Make sure you have the required [permissions and/or quota](#) in Cloud Spanner, Cloud Storage, Compute Engine, and Cloud Dataflow.

✓ **Choose a source**
Source ace-exam-test-bucket

2 **Select database dialect**
Choose the dialect of your original exported database.

☒ Google Standard SQL
☐ PostgreSQL

NEXT

3 **Name your database**

4 **Choose a region for the import job**

☐ I understand that this import will incur Cloud Dataflow charges at the standard rate, as well as possible network egress charges.

✓ PRICING INFO

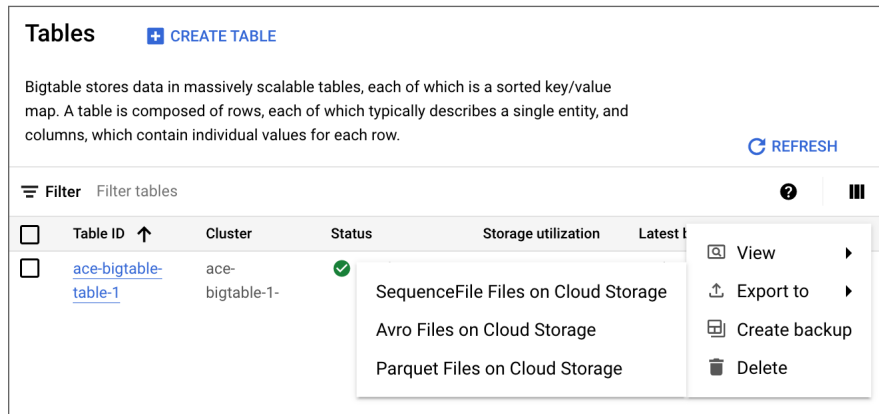
IMPORT CANCEL

Cloud Spanner does not have a `gcloud` command to export data, but you can use Dataflow to export data. The details of constructing Dataflow jobs is outside the scope of this section. For more details, see the Cloud Dataflow documentation at <https://cloud.google.com/dataflow/docs>.

Exporting Data from Cloud Bigtable

Cloud Bigtable supports exporting data through the console. Navigate to the Bigtable console and select Tables from the menu bar on the left. This will display an export dialog as shown in Figure 13.24.

Bigtable exports are stored in Cloud Storage and can use one of three formats: SequenceFile, Avro, or Parquet.

FIGURE 13.24 Export page for Cloud Bigtable

Importing and Exporting Data: Cloud Dataproc

Cloud Dataproc is not a database like Cloud SQL or Bigtable; rather, it is a data analysis platform. These platforms are designed more for data manipulation, statistical analysis, machine learning, and other complex operations than for data storage and retrieval. Cloud Dataproc is not designed to be a persistent store of data. For that you should use Cloud Storage or persistent disks to store the data files you want to analyze.

Cloud Dataproc does have Import and Export commands to save and restore cluster configuration data. These commands are available using `gcloud`.

The command to export a Dataproc cluster configuration is as follows:

```
gcloud dataproc clusters export [CLUSTER_NAME] \
--destination=[PATH_TO_EXPORT_FILE]
```

Here's an example:

```
gcloud dataproc clusters export ace-exam-dataproc-cluster \
--destination=gs://ace-exam-bucket1/mydataproc.yaml
```

To import a configuration file, use the `import` command:

```
gcloud dataproc clusters import [SOURCE_FILE]
```

For example, to import the file created in the previous export example, use the following:

```
gcloud dataproc clusters import gs://ace-exam-bucket1/mydataproc.yaml
```

Importing and exporting data are common operations. Google Cloud provides console and command-line tools for most database services. There are also beta commands for exporting and importing cluster configuration data for Dataproc.

Streaming Data to Cloud Pub/Sub

So far in this chapter you have spent most of your time on moving data into and around Cloud Storage, along with importing and exporting data to databases. Let's now turn our attention to working with Cloud Pub/Sub, the messaging queue.

As a Cloud Engineer, you may need to create message queues for application developers. Although developers will most likely write services that use Pub/Sub, Cloud Engineers should be able to test Pub/Sub topics and subscriptions. We discussed how to create message queues in Chapter 12, "Deploying Storage in Google Cloud." Here our focus will be on creating messages on topics and receiving those messages through subscriptions.

The `gcloud pubsub` commands you will use are `create`, `publish`, and `pull`. To create a topic, you use the following command:

```
gcloud pubsub topics create [TOPIC_NAME]
```

The command to create a subscription is as follows:

```
gcloud pubsub subscriptions create --topic [TOPIC_NAME] [SUBSCRIPTION_NAME]
```

For example, to create a topic called `ace-exam-topic1` and a subscription to that topic called `ace-exam-sub1`, you can use these commands:

```
gcloud pubsub topics create ace-exam-topic1
gcloud pubsub subscriptions create --topic=ace-exam-topic1 ace-exam-sub1
```

Now, to test whether the message queue is working correctly, you can send data to the topic using the following command:

```
cloud pubsub topics publish [TOPIC_NAME] --message [MESSAGE]
```

and then read that message from the subscription using the following:

```
gcloud pubsub subscriptions pull --auto-ack [SUBSCRIPTION_NAME]
```

To write a message to the topic and read it from the subscription you just created, you can use the following:

```
gcloud pubsub topics publish ace-exam-topic1 \
--message "first ace exam message"
gcloud pubsub subscriptions pull --auto-ack ace-exam-sub1
```



Real World Scenario

Decoupling Services Using Message Queues

One of the challenges with distributed systems is that sometimes one service cannot keep up with the inflow of data. This can create a backlog in services that depend on the lagging service.

For example, a sudden spike in traffic on a retail site may put a high load on an inventory tracking service, which updates inventory as customers add or remove items from their baskets. The inventory program may be slow to respond to a service that added an item to the cart. If that service is waiting for a response from the inventory service, it too will be delayed. This kind of synchronous communication is problematic when distributed systems are under load.

A better option is to decouple the direct connection between services. For example, the user interface could write a message to a Pub/Sub topic each time an item is added or removed from a customer's basket. The inventory management service can subscribe to this topic and update the inventory system as new messages come in. If the inventory system slows down, it will not affect the user interface because it is writing to a Pub/Sub topic, which can scale along with the load generated by the user interface.

Summary

In this chapter, we looked at the different ways you can load data into storage, database, and message queue systems. Cloud Storage is organized around objects in buckets. The `gsutil` command and Cloud Console can be used to upload data as well as move it between buckets. You saw that the `gsutil cp` command can be used to copy files between Cloud Storage and VMs.

The database services provide import and export utilities. Each supports a variety of file formats.

Cloud Pub/Sub can be used to decouple applications and improve resiliency to spikes in load. You saw how to create a topic and subscriptions and how to push data to the message queue, where it can be read by subscribers.

Know that Cloud Spanner uses the Dataflow service for importing and exporting. There can be additional charges when using Dataflow and moving data between regions.

Exam Essentials

Know how to load data into and move data around Cloud Storage. Cloud Storage is widely used for a variety of use cases, including long-term storage and archiving, file transfers, and data sharing. Understand the structure of `gsutil` commands, which is different from `gcloud`. `gsutil` commands start with `gsutil` followed by an operation, such as `copy` or `make bucket`. Be sure to know the syntax of the `copy (cp)`, `move (mv)`, and `make bucket (mb)` commands. You can copy files from Cloud Storage to VMs, and vice versa. Also, know

that the `gsutil acl ch -u` command is used to change permissions on objects. You can use the `gsutil acl ch` command to change permissions on a Cloud Storage bucket.

Understand how import and export work with Cloud SQL. Importing and exporting data from databases are common operations. You can perform imports and exports from the console and from the command line.

Know that you can export entities from a Cloud Firestore. Exports and imports are done at the database level when in Native mode and at the level of namespaces when the database is in Datastore mode.

Understand how to export and import data from BigQuery. BigQuery has a range of options for the source of data to import. Data can be compressed when exported to save on space. BigQuery can export data in multiple formats, including CSV, JSON, and Avro. Know that the `bq` command is used for importing and exporting from the command line.

Know that Pub/Sub is used to send messages between services. Pub/Sub allows for greater resiliency to fluctuations in load. If one service lags, its work can accumulate in a Pub/Sub queue without forcing the service that generates that data to wait.

Review Questions

You can find the answers in the Appendix.

1. Which of the following commands is used to create buckets in Cloud Storage?
 - A. `gcloud storage create buckets`
 - B. `gsutil storage buckets create`
 - C. `gsutil mb`
 - D. `gcloud mb`
2. You need to copy files from your local device to a bucket in Cloud Storage. What command would you use? Assume you have Cloud SDK installed on your local computer.
 - A. `gsutil copy`
 - B. `gsutil cp`
 - C. `gcloud cp`
 - D. `gcloud storage objects copy`
3. You are migrating a large number of files from a local storage system to Cloud Storage. You want to use the Cloud Console instead of writing a script. Which of the following Cloud Storage operations can you perform in the console?
 - A. Upload files only
 - B. Upload folders only
 - C. Upload files and folders
 - D. Compare local files with files in the bucket using the `diff` command
4. A software developer asks for your help exporting data from a Cloud SQL database. The developer tells you which database to export and which bucket to store the export file in, but hasn't mentioned which file format should be used for the export file. What are the options for the export file format?
 - A. CSV and XML
 - B. CSV and JSON
 - C. JSON and SQL
 - D. CSV and SQL
5. A database administrator has asked for an export of a MySQL database in Cloud SQL. The database administrator will load the data into another relational database and would like to do it with the least amount of work. Specifically, the loading method should not require the database administrator to define a schema. What file format would you recommend for this task?
 - A. SQL
 - B. CSV
 - C. XML
 - D. JSON

6. Which command will export a MySQL database called `ace-exam-mysql1` to a file called `ace-exam-mysql-export.sql` in a bucket named `ace-exam-bucket1`?
- A. `gcloud storage export sql ace-exam-mysql1 \`
`gs://ace-exam-bucket1/ace-exam-mysql-export.sql \`
`--database=mysql`
 - B. `gcloud sql export ace-exam-mysql1 \`
`gs://ace-exam-bucket1/ace-exam-mysql-export.sql \`
`--database=mysql`
 - C. `gcloud sql export sql ace-exam-mysql1 \`
`gs://ace-exam-bucket1/ace-exam-mysql-export.sql \`
`--database=mysql`
 - D. `gcloud sql export sql ace-exam-mysql1 \`
`gs://ace-exam-mysql-export.sql/ace-exam-bucket1/ \`
`--database=mysql`
7. Which of the following file formats is not an option for an export file when exporting from BigQuery?
- A. CSV
 - B. XML
 - C. Avro
 - D. JSON
8. Which of the following file formats is not supported when importing data into BigQuery?
- A. CSV
 - B. Parquet
 - C. Avro
 - D. YAML
9. You have received a large data set from an Internet of Things (IoT) system. You want to use BigQuery to analyze the data. What command-line command would you use to make data available for analysis in BigQuery?
- A. `bq load --autodetect --source_format=[FORMAT] \`
`[DATASET].[TABLE] [PATH_TO_SOURCE]`
 - B. `bq import --autodetect --source_format=[FORMAT] \`
`[DATASET].[TABLE] [PATH_TO_SOURCE]`
 - C. `gcloud BigQuery load --autodetect --source_format=[FORMAT] \`
`[DATASET].[TABLE] [PATH_TO_SOURCE]`
 - D. `gcloud BigQuery load --autodetect --source_format=[FORMAT] \`
`[DATASET].[TABLE] [PATH_TO_SOURCE]`

10. You have set up a Cloud Spanner process to export data to Cloud Storage. You notice that each time the process runs you incur charges for another Google Cloud service, which you think is related to the export process. What other Google Cloud service might be incurring charges during the Cloud Spanner export?
- A. Dataproc
 - B. Dataflow
 - C. Firestore
 - D. bq
11. As a developer on a project using Bigtable for an IoT application, you will need to export data from Bigtable to make some data available for analysis with another tool. What would you use to export the data, assuming you want to minimize the amount of effort required on your part?
- A. A Java program designed for importing and exporting data from Bigtable
 - B. `gcloud bigtable table export`
 - C. `bq bigtable table export`
 - D. An import tool provided by the analysis tool
12. You have just exported from a Dataproc cluster. What have you exported?
- A. Data in Spark DataFrames
 - B. All tables in the Spark database
 - C. Configuration data about the cluster
 - D. All tables in the Hadoop database
13. A team of data scientists has requested access to data stored in Bigtable so that they can train machine learning models. They explain that Bigtable does not have the features required to build machine learning models. Which of the following Google Cloud services are they most likely to use to build machine learning models?
- A. Firestore
 - B. Dataflow
 - C. Dataproc
 - D. DataAnalyze
14. Which of the following is the correct command to create a Pub/Sub topic?
- A. `gcloud pubsub topics create`
 - B. `gcloud pubsub create topics`
 - C. `bq pubsub create topics`
 - D. `cbt pubsub topics create`

15. Which of the following commands will create a subscription on the topic `ace-exam-topic1`?
- A. `gcloud pubsub create --topic=ace-exam-topic1 ace-exam-sub1`
 - B. `gcloud pubsub subscriptions create --topic=ace-exam-topic1`
 - C. `gcloud pubsub subscriptions create --topic=ace-exam-topic1 ace-exam-sub1`
 - D. `gsutil pubsub subscriptions create --topic=ace-exam-topic1 ace-exam-sub1`
16. What is one of the direct advantages of using a message queue in distributed systems?
- A. It increases security.
 - B. It decouples services, so if one lags, it does not cause other services to lag.
 - C. It supports more programming languages.
 - D. It stores messages until they are read by default.
17. To ensure you have installed beta `gcloud` commands, which command should you run?
- A. `gcloud components beta install`
 - B. `gcloud components install beta`
 - C. `gcloud commands install beta`
 - D. `gcloud commands beta install`
18. What parameter is used to tell BigQuery to automatically detect the schema of a file on import?
- A. `autodetect`
 - B. `autoschema`
 - C. `detectschema`
 - D. `dry_run`
19. The compression options Deflate and Snappy are available for what file types when exporting from BigQuery?
- A. Avro
 - B. CSV
 - C. XML
 - D. Thrift
20. You want to read a message from a Pub/Sub topic and acknowledge reading that message in the same command. Which of the following would you use?
- A. `gcloud pubsub subscriptions pull --auto-ack`
 - B. `gcloud pubsub topic pull --auto-ack`
 - C. `gsutil pubsub topic pull --with-acknowledgement`
 - D. `gcloud pubsub subscription pull --with-acknowledgement`

Chapter 14

Networking in the Cloud: Virtual Private Clouds and Virtual Private Networks

**THIS CHAPTER COVERS THE FOLLOWING
OBJECTIVES OF THE GOOGLE ASSOCIATE
CLOUD ENGINEER CERTIFICATION EXAM:**

- ✓ 2.4 Planning and configuring network resources
- ✓ 4.5 Managing networking resources





In this chapter we turn our attention to networking, starting with virtual private clouds (VPCs). You will learn how to create VPCs with default and custom subnets. You'll learn about creating custom network configurations in Compute Engine for cases when default network configurations do not meet your needs. Finally, we will show you how to configure firewall rules and create virtual private networks (VPNs).

Creating a Virtual Private Cloud with Subnets

VPCs are software versions of physical networks that link resources in a project. Google Cloud automatically creates a VPC when you create a project. You can create additional VPCs and modify the VPCs created by Google Cloud.

VPCs are global resources, so they are not tied to a specific region or zone. Resources, such as Compute Engine virtual machines (VMs) and Kubernetes Engine clusters, can communicate with each other, assuming traffic is not blocked by a firewall rule.

VPCs contain subnetworks, called *subnets*, which are regional resources. Subnets have a range of IP addresses associated with them. Subnets provide private internal addresses. Resources use these addresses to communicate with each other and with Google APIs and services.

In addition to VPCs associated with projects, you can create a shared VPC within an organization. The shared VPC is hosted in a common project. Users in other projects who have sufficient permissions can create and use resources in the shared VPC. You can also use VPC network peering for interproject connectivity, even if an organization is not defined.

In this section, you will create a VPC with subnets using Cloud Console and `gcloud`, and then turn your attention to creating a shared VPC.

Creating a Virtual Private Cloud with Cloud Console

To create a VPC in Cloud Console, navigate to the VPC Networks page, as shown in Figure 14.1.

Clicking Create VPC Network opens the page shown in Figure 14.2. Figure 14.2 shows that you can assign a name and description to a new VPC. It also shows a list of subnets that will be created in the VPC. When an automatic mode VPC is created, subnets are created in each region. Google Cloud chooses a range of IP addresses for each subnet when creating an auto mode network.

FIGURE 14.1 The VPC Network page of Cloud Console

VPC networks

+

CREATE VPC NETWORK

↻

REFRESH

🗨️

HELP ASSISTANT

NETWORKS IN CURRENT PROJECT

SUBNETS IN CURRENT PROJECT

💡

SMTP port 25 disallowed in this project

?

VPC networks

≡

Filter

Enter property name or value

?

⌵

Name	Subnets	MTU	Mode	Internal IP ranges	Gateways	Firewall rules	Global dynamic routing
default	36	1460	Auto			4	Off

FIGURE 14.2 Creating a VPC in Cloud Console, part 1

← Create a VPC network

Name *

Lowercase letters, numbers, hyphens allowed

Description

Subnets

Subnets let you create your own private cloud topology within Google Cloud. Click Automatic to create a subnet in each region, or click Custom to manually define the subnets. [Learn more](#)

Subnet creation mode ?

☐ Custom

☒ Automatic

IP stack type

IPv4 (single-stack)

⚠️

These IP address ranges will be assigned to each region in your VPC network. When an instance is created for your VPC network, it will be assigned an IP from the appropriate region's address range.

Region ↑	IP address range
asia-east1	10.140.0.0/20
asia-northeast1	10.146.0.0/20
asia-south1	10.160.0.0/20
asia-southeast1	10.148.0.0/20
australia-southeast1	10.152.0.0/20

Alternatively, you can create one or more custom subnets by selecting the Custom tab in the Subnet section (Figure 14.3). This displays another page that allows you to specify a region and an IP address range. The IP range is specified in classless interdomain routing (CIDR) notation. (See the upcoming sidebar “Understanding CIDR Notation” for details on how to specify IP addresses using that notation.) You can turn on Private Google Access. That allows VMs on the subnet to access Google services without assigning an external IP address to the VM. You can also turn on logging of network traffic by setting the Flow Logs option to On.

FIGURE 14.3 Creating a custom subnet

The screenshot shows the 'Create a VPC network' form. At the top, there is a back arrow and the title 'Create a VPC network'. Below this, there is a 'Name' field with a red asterisk and a help icon. A note below the field states 'Lowercase letters, numbers, hyphens allowed'. Below the name field is a 'Description' text area. Further down is the 'VPC network ULA internal IPv6 range' section with a help icon. A note states 'Enabling this feature will assign a /48 from Google defined ULA prefix fd20::/20.' There are two radio buttons: 'Enabled' and 'Disabled', with 'Disabled' being selected. Below this is the 'Subnets' section. It contains a paragraph explaining subnets and a link to 'Learn more'. Underneath is the 'Subnet creation mode' section with two radio buttons: 'Custom' and 'Automatic', with 'Custom' being selected. At the bottom, there is a 'New subnet' dropdown menu and a blue 'ADD SUBNET' button.

Figure 14.4 shows the second part of the VPC page, which includes firewall rules, dynamic routing setting, and a DNS server policy. The Firewall Rules section lists rules that can be applied to the VPC. In the example in Figure 14.4, one of the rules allows ingress, which is incoming TCP traffic on port 22, to allow for SSH access. The IP range of 0.0.0.0/0 allows traffic from all source IP addresses.

FIGURE 14.4 Creating a VPC in Cloud Console, part 2

←

Create a VPC network

Firewall rules ?

Select any of the firewall rules below that you would like to apply to this VPC network. Once the VPC network is created, you can manage all firewall rules on the Firewall rules page.

IPv4 FIREWALL RULES

IPv6 FIREWALL RULES

<input type="checkbox"/>	Name	Type	Targets	Filters	Protocols / ports	Action	Priority ↑	
<input type="checkbox"/>	allow-custom ?	Ingress	Apply to all	IP ranges:	all	Allow	65,534	EDIT
<input type="checkbox"/>	allow-icmp ?	Ingress	Apply to all	IP ranges:	icmp	Allow	65,534	
<input type="checkbox"/>	allow-rdp ?	Ingress	Apply to all	IP ranges:	tcp:3389	Allow	65,534	
<input type="checkbox"/>	allow-ssh ?	Ingress	Apply to all	IP ranges:	tcp:22	Allow	65,534	
	deny-all-ingress ?	Ingress	Apply to all	IP ranges:	all	Deny	65,535	
	allow-all-egress ?	Egress	Apply to all	IP ranges:	all	Allow	65,535	

Dynamic routing mode ?

☒ Regional

Cloud Routers will learn routes only in the region in which they were created

☐ Global

Global routing lets you dynamically learn routes to and from all regions with a single VPN or interconnect and Cloud Router

?

Enable DNS API to pick a DNS policy

ENABLE

Maximum transmission unit (MTU)

1460

CREATE

CANCEL

The dynamic routing option determines what routes are learned. Regional routing will have Google Cloud Routers learn routes within the region. Global routing will enable Google Cloud Routers to learn routes on all subnetworks in the VPC.

The optional DNS server policy lets you choose a DNS policy that enables DNS name resolution provided by Google Cloud or makes changes to name resolution order. (See Chapter 15, “Networking in the Cloud: DNS, Load Balancing, Google Private Access, and IP Addressing,” for more details.)

Once you have specified the parameters and created a VPC, it will appear in the VPC listing and show information about the VPC and its subnets, as shown in Figure 14.5.

FIGURE 14.5 Listing of VPCs and subnets

<input type="checkbox"/>	Name ↑	Region	Stack Type	Internal IP ranges	External IP ranges	Secondary IPv4 ranges
<input type="checkbox"/>	ace-vpc-1	us-west2	IPv4	10.10.0.0/24	None	None
<input type="checkbox"/>	default	us-central1	IPv4	10.128.0.0/20	None	None
<input type="checkbox"/>	default	europe-west1	IPv4	10.132.0.0/20	None	None
<input type="checkbox"/>	default	us-west1	IPv4	10.138.0.0/20	None	None
<input type="checkbox"/>	default	asia-east1	IPv4	10.140.0.0/20	None	None
<input type="checkbox"/>	default	us-east1	IPv4	10.142.0.0/20	None	None

Creating a Virtual Private Cloud with *gcloud*

The `gcloud` command to create a VPC is `gcloud compute networks create`. For example, to create a VPC in the default project with automatically generated subnets, you would use the following command:

```
gcloud compute networks create ace-exam-vpc1 --subnet-mode=auto
```

You can also configure custom subnets by creating a VPC network specifying the custom option and then creating subnets in that VPC. The first command to create a custom VPC called `ace-exam-vpc1` is as follows:

```
gcloud compute networks create ace-exam-vpc1 --subnet-mode=custom
```

Next, you can create a subnet using the `gcloud compute networks subnet create` command. This command requires that you specify a VPC, the region, and the IP range. You can optionally turn on the Private Google Access and Flow Logs settings by adding the appropriate flags.

Here is an example command to create a subnet called `ace-exam-vpc-subnet1` in the `ace-exam-vpc1` VPC. This subnet is created in the `us-west2` region with an IP range of `10.10.0.0/16`. The Private IP Access and Flow Logs settings are turned on.

```
gcloud compute networks subnets create ace-exam-vpc-subnet1 --
network=ace-exam-vpc1 --region=us-west2 --range=10.10.0.0/16 --
enable-private-ip-google-access --enable-flow-logs
```

Understanding CIDR Notation

When you specify ranges of IP addresses, you use something called *classless interdomain routing* (CIDR). The name stems from early IP networks that were defined into three primary fixed classes: A, B, and C. A classless network address structure was created to overcome the limitations of a class-based routing structure, particularly the lack of flexibility in creating different-sized subnets.

CIDR uses variable-length subnet masking (VLSM) to allow network administrators to define networks with the number of addresses that they need, not the fixed numbers that were allocated to the older class model interdomain routine.

CIDR addresses consist of two sets of numbers: a network address for identifying a subnet and a host identifier. These numbers are written out using CIDR notation, which consists of a network address and a network mask. Example network addresses, according to the RFC1918 specification are:

```
10.0.0.0 - 10.255.255.255 (/8)
172.16.0.0 - 172.31.255.355 (/12)
192.168.0.0 - 192.168.255.255 (/16)
```

CIDR notation adds a slash (/) and a number indicating how many bits of an IP address to allocate to the network mask, which determines which addresses are within the block of the address and which are not.

For example, 192.168.0.0/16 means that 16 bits of the 32 bits of an IP address are used to specify the network, and 16 bits are used to specify the host address. With 16 bits, you can create $2^{16}-2$, or 65,534 host addresses.

The CIDR block 172.16.0.0/12 indicates that 12 bits are used for specifying the network, and 20 bits are used to specify host addresses. With 20 bits, you can create up to 1,048,574 host addresses. In general, the smaller the number after the slash, the more host addresses are available. You can experiment with CIDR block options using a CIDR calculator such as the one at www.subnet-calculator.com/cidr.php.

Creating a Shared Virtual Private Cloud Using *gcloud*

If you want to create a shared VPC, you can use the `gcloud` command `gcloud compute shared-vpc`.

Before executing commands to create a shared VPC, you will need to assign an org member the Shared VPC Admin role at the organization level or the folder level. To assign the Shared VPC Admin role, which uses the descriptor `roles/compute.xpnAdmin`, issue this command:

```
gcloud organizations add-iam-policy-binding [ORG_ID]
```

```
--member='user:[EMAIL_ADDRESS] '
--role="roles/compute.xpnAdmin"
```

[*ORG_ID*] is the organization identifier of the organization using the policy. You can find an organization ID with the command `gcloud organizations list`. If you prefer to assign the Shared VPC Admin role to a folder, you can use this command:

```
gcloud resource-manager folders add-iam-policy-binding [FOLDER_ID]
--member='user:[EMAIL_ADDRESS] '
--role="roles/compute.xpnAdmin"
```

[*FOLDER_ID*] is the identifier of the folder of the policy. You can get folder IDs by using this command:

```
gcloud resource-manager folders list --organization=[ORG_ID]
```

For more on roles and privileges, see Chapter 17, “Configuring Access and Security.”

Once you have set the Shared VPC Admin role at the organization level, you can issue the `shared-vpc` command:

```
gcloud compute shared-vpc enable [HOST_PROJECT_ID]
```

If you are sharing the VPC at the folder level, use this command:

```
gcloud compute shared-vpc enable [HOST_PROJECT_ID]
```

Now that the shared VPC is created, you can associate projects using the `gcloud compute shared-vpc associate-projects` command. At the organization level, you can use this command:

```
gcloud compute shared-vpc associated-projects add [SERVICE_PROJECT_ID] \
--host-project [HOST_PROJECT_ID]
```

At the folder level, the command to associate folders is as follows:

```
gcloud compute shared-vpc associated-projects add [SERVICE_PROJECT_ID] \
--host-project [HOST_PROJECT_ID]
```

Alternatively, VPC network peering can be used for interproject traffic when an organization does not exist. VPC network peering is implemented using the `gcloud compute networks peerings create` command. For example, you peer two VPCs by specifying peerings on each network. Here’s an example:

```
gcloud compute networks peerings create peer-ace-exam-1 \
--network ace-exam-network-A \
--peer-project ace-exam-project-B \
--peer-network ace-exam-network-B \
--auto-create-routes
```

And then create a peering on the other network using:

```
gcloud compute networks peerings create peer-ace-exam-1 \
```

```
--network ace-exam-network-B \
--peer-project ace-exam-project-A \
--peer-network ace-exam-network-A \
--auto-create-routes
```

This peering will allow private traffic to flow between the two VPCs.

Deploying Compute Engine with a Custom Network

You can deploy a VM with custom network configurations using the console and the command line.

Navigate to the Compute Engine section of the console and open the Create Instance page, shown in Figure 14.6.

FIGURE 14.6 Preliminary options to create an instance in Cloud Console

The screenshot shows the 'Create Instance' page in the Google Cloud Console. The form is divided into several sections:

- Name:** A text input field containing 'instance-1' with a help icon.
- Labels:** A section with a '+ ADD LABELS' button and a help icon.
- Region:** A dropdown menu set to 'us-central1 (Iowa)' with a help icon. Below it, it says 'Region is permanent'.
- Zone:** A dropdown menu set to 'us-central1-a' with a help icon. Below it, it says 'Zone is permanent'.
- Machine configuration:**
 - Machine family:** A section with four tabs: 'GENERAL-PURPOSE' (selected), 'COMPUTE-OPTIMIZED', 'MEMORY-OPTIMIZED', and 'GPU'. Below the tabs, it says 'Machine types for common workloads, optimized for cost and flexibility'.
 - Series:** A dropdown menu set to 'E2' with a help icon. Below it, it says 'CPU platform selection based on availability'.
 - Machine type:** A dropdown menu set to 'e2-medium (2 vCPU, 4 GB memory)' with a help icon.
- Hardware specifications:**
 - vCPU:** 1-2 vCPU (1 shared core)
 - Memory:** 4 GB
- Expandable section:** A blue arrow icon and the text 'CPU PLATFORM AND GPU' at the bottom.

In the horizontal menu toward the bottom of the page, click Management > Security > Disks > Networking > Sole Tenancy to expand the optional forms and then click the Networking tab to display a page similar to Figure 14.7.

Note that on this page, you can set network tags, which are used for defining firewall rules and routes. Click Add Network Interface to display a page like that shown in Figure 14.8. Here you can choose a custom network. In this example, we are choosing `ace-exam-vpc1`, which we created earlier in the chapter. We also selected a subnet.

FIGURE 14.7 Networking configuration options

Networking
Hostname and network interfaces

Network tags

Hostname

Set a custom hostname for this instance or leave it default. Choice is permanent

IP forwarding

☐ Enable

Network performance configuration

Network interface card

—

Network bandwidth

☐ Increase total egress bandwidth

Maximum outbound network bandwidth: 2Gbps

Network interfaces

Network interface is permanent

default default (10.128.0.0/20)

[ADD NETWORK INTERFACE](#)

Here, you can also specify a static IP address or choose a custom ephemeral address using the Primary Internal IP setting. The External IP drop-down menu allows you to have an ephemeral external IP or use a static external IP.

You can also create an instance to run in a particular subnet using the `gcloud compute instances create` command with subnet and zone parameters.

```
gcloud compute instances create [INSTANCE_NAME] --subnet [SUBNET_NAME] --zone [ZONE_NAME]
```

FIGURE 14.8 Options to add a custom network interface

Network interfaces ?

Network interface is permanent

Interface	Subnet	IP Address	Actions
default default	(10.168.0.0/20)		

New network interface ^

Network *
ace-exam-vpc1

Subnetwork *
ace-exam-vpc-subnet1 IPv4 (10.10.0.0/16)

IP stack type
☒ IPv4 (single-stack)
☐ IPv4 and IPv6 (dual-stack)

Primary internal IP
Ephemeral (Automatic)

Alias IP ranges
[+ ADD IP RANGE](#)

External IPv4 address
Ephemeral

Network Service Tier

Creating Firewall Rules for a Virtual Private Cloud

Firewall rules are defined at the network level and used to control the flow of network traffic to VMs.

Firewall rules allow or deny a specified type of traffic on a port; for example, a rule may allow TCP traffic to port 22. They also are applied to traffic in one direction, either incoming (ingress) or outgoing (egress) traffic. It is important to note that the firewall is *stateful*, which means if traffic is allowed in one direction and a connection established, it is allowed in the

other direction. Firewalls rulesets are stateful, so if a connection is allowed, like establishing an SSH connection on port 22, then all later traffic matching this rule is permitted as long as the connection is active. An active connection is one with at least one packet exchanged every 10 minutes.

Structure of Firewall Rules

Firewall rules consist of several components:

Direction Either ingress or egress.

Priority Highest-priority rules are applied; any rule with a lower priority that matches are not applied. Priority is specified by an integer from 0 to 65535. 0 is the highest priority, and 65535 is the lowest.

Action Either allow or deny. Only one can be chosen.

Target An instance to which the rule applies. Targets can be all instances in a network, instances with particular network tags, or instances using a specific service account.

Source/Destination Source applies to ingress rules and specifies source IP ranges, instances with particular network tags, or instances using a particular service account. You can also use combinations of source IP ranges and network tags and combinations of source IP ranges and service accounts used by instances. The IP address 0.0.0.0/0 indicates any IP address. The Destination parameter uses only IP ranges.

Protocol and Port A network protocol such as TCP, UDP, or ICMP and a port number. If no protocol is specified, then the rule applies to all protocols.

Enforcement Status Firewall rules are either enabled or disabled. Disabled rules are not applied even if they match. Disabling is sometimes used to troubleshoot problems with traffic getting through when it should not or not getting through when it should.

All VPCs start with two implied rules: one allows egress traffic to all destinations (IP address 0.0.0.0/0), and one denies all incoming traffic from any source (IP address 0.0.0.0/0). Both implied rules have priority 65535, so you can create other rules with higher deny or allow traffic as you need. You cannot delete an implied rule.

When a VPC is automatically created, the default network is created with four network rules. These rules allow the following:

- Incoming traffic from any VM instance on the same network
- Incoming TCP traffic on port 22, allowing SSH

- Incoming TCP traffic on port 3389, allowing Microsoft Remote Desktop Protocol (RDP)
 - Incoming Internet Control Message Protocol (ICMP) from any source on the network
- The default rules all have priority 65534.

Creating Firewall Rules Using Cloud Console

To create or edit firewall rules, navigate to the VPC section of the console and select the Firewall option from the VPC menu. Figure 14.9 shows a list of firewall rules.

FIGURE 14.9 List of firewall rules in the VPC section of Cloud Console

Filter Enter property name or value												
<input type="checkbox"/>	Name	Type	Targets	Filters	Protocols / ports	Action	Priority	Network ↑	Logs	Hit count ⓘ	Last hit ⓘ	Insights
<input type="checkbox"/>	default-allow-icmp	Ingress	Apply to all	IP ranges: 0.0.0.0/0	icmp	Allow	65534	default	Off	—	—	▼
<input type="checkbox"/>	default-allow-internal	Ingress	Apply to all	IP ranges: 10.0.0.0/8	tcp:0-65535 udp:0-65535 icmp	Allow	65534	default	Off	—	—	▼
<input type="checkbox"/>	default-allow-rdp	Ingress	Apply to all	IP ranges: 0.0.0.0/0	tcp:3389	Allow	65534	default	Off	—	—	▼
<input type="checkbox"/>	default-allow-ssh	Ingress	Apply to all	IP ranges: 0.0.0.0/0	tcp:22	Allow	65534	default	Off	—	—	▼

Click Create Firewall Rule at the top of the page to create a new firewall rule. This opens the page shown in Figure 14.10.

Here, you specify a name and description of the firewall rule. You can choose to turn logging on or off. If it is on, logging information will be captured in Cloud Logging. (See Chapter 18, “Monitoring, Logging, and Cost Estimating,” for more on Cloud Logging.) You also need to specify the network in the VPC to apply the rule to.

Next, you will need to specify a priority, direction, action, targets, and sources. Priority can be integers in the range from 0 to 65535. Direction can be Ingress or Egress. Action can be Allow or Deny. Choose targets are the drop-down list; the options are shown in Figure 14.11.

If you choose tags or service accounts, you will be able to specify the tags or the name of the service account. You can also specify source filters as either IP ranges, subnets, source tags, or service accounts. Google Cloud allows a second source filter if you’d like to use a combination of conditions. A list of source filters is shown in Figure 14.12.

Finally, you specify protocol and ports by choosing between the Allow All and Specified Protocols and Ports options. If you choose the latter, you can specify protocols and ports.

Figure 14.13 shows the listing of the firewall rule created using the parameters specified in Figure 14.10.

FIGURE 14.10 Creating a firewall rule

← Create a firewall rule

Description

Example firewall rule

Logs

Turning on firewall logs can generate a large number of logs which can increase costs in Cloud Logging. [Learn more](#)

☐ On

☒ Off

Network *

default

Priority *

1000

[CHECK PRIORITY OF OTHER FIREWALL RULES](#)

Priority can be 0 - 65535

Direction of traffic ?

☒ Ingress

☐ Egress

Action on match ?

☒ Allow

☐ Deny

Targets

All instances in the network

Source filter

IPv4 ranges

Source IPv4 ranges *

0.0.0.0/0 for example, 0.0.0.0/0, 192.168.2.0/24

Second source filter

None

Protocols and ports ?

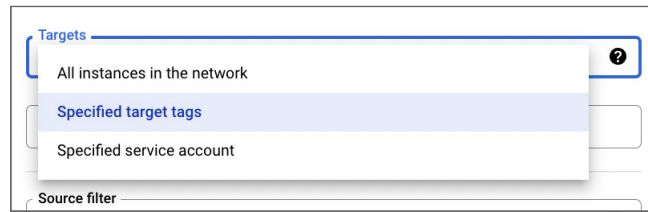
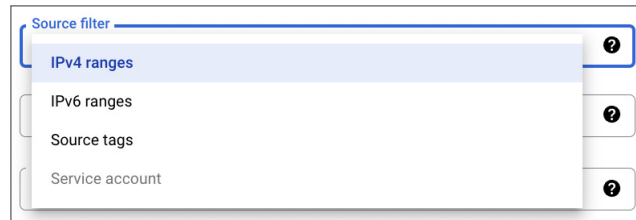
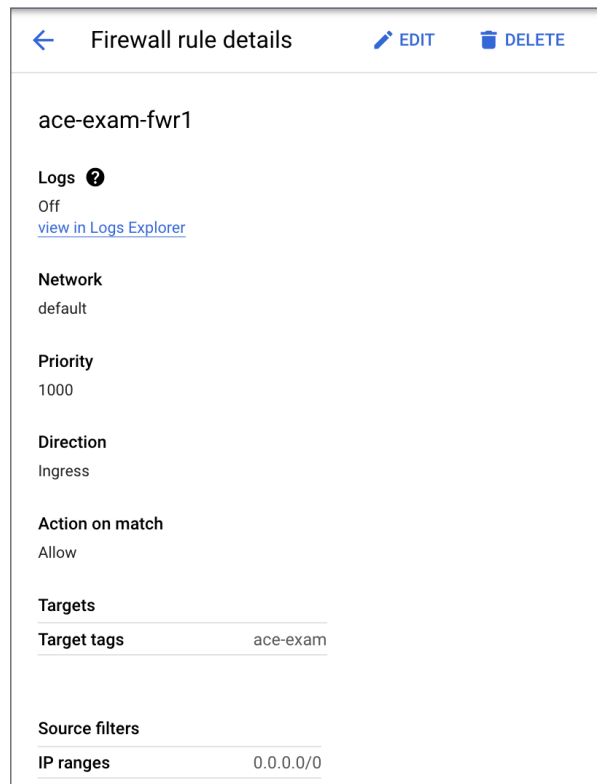
☒ Allow all

☐ Specified protocols and ports

▼ DISABLE RULE

CREATE

CANCEL

FIGURE 14.11 List of target types**FIGURE 14.12** List of source filter types**FIGURE 14.13** Listing of the firewall rule created using the earlier configuration

Creating Firewall Rules Using *gcloud*

The command for working with firewall rules from the command line is `gcloud compute firewall-rules`. With this command, you can create, delete, describe, update, and list firewall rules.

A number of parameters are used with `gcloud compute firewall-rules create`:

- `--action`
- `--allow`
- `--description`
- `--destination-ranges`
- `--direction`
- `--network`
- `--priority`
- `--source-ranges`
- `--source-service-accounts`
- `--source-tags`
- `--target-service-accounts`
- `--target-tags`

For example, to allow all TCP traffic on ports 20000 to 25000, use this:

```
gcloud compute firewall-rules create ace-exam-fwr2 \  
--network ace-exam-vpc1 --allow tcp:20000-25000
```

Creating a Virtual Private Network

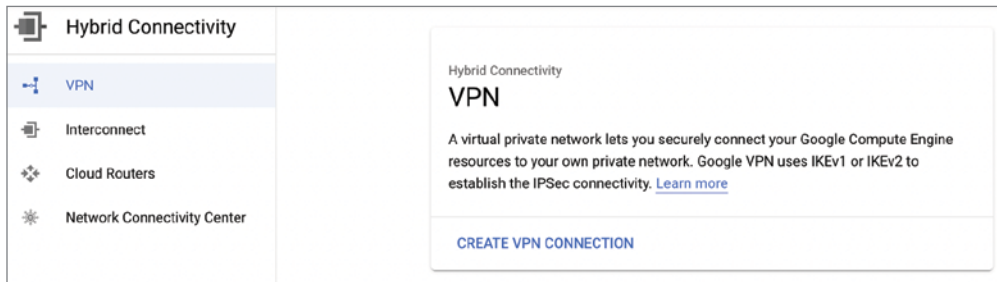
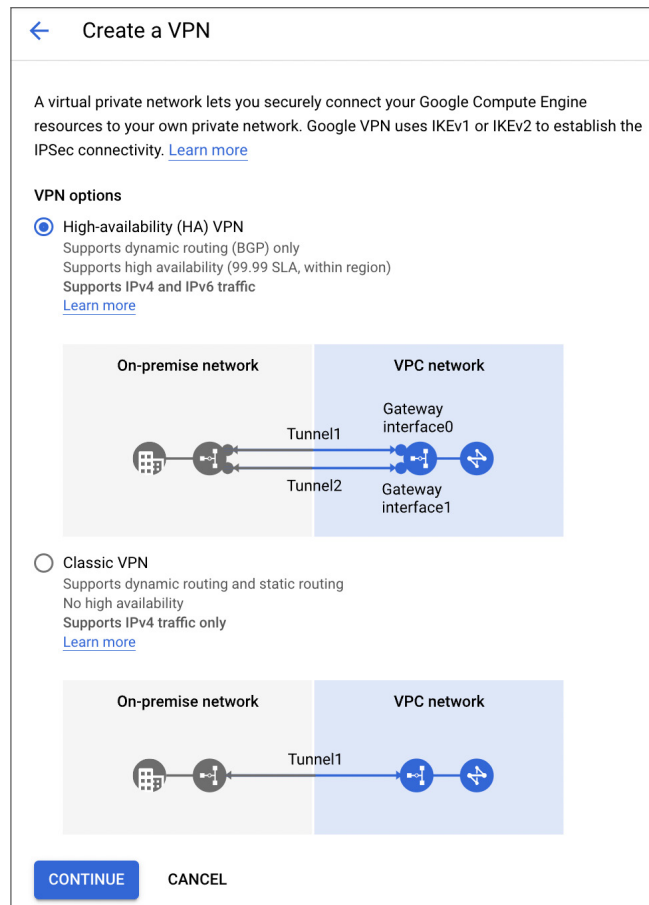
VPNs allow you to securely send network traffic from the Google network to your own network. You can create a VPN using Cloud Console or the command line.

Creating a Virtual Private Network Using Cloud Console

To create a VPN using Cloud Console, navigate to the Hybrid Connectivity section of the console, as shown in Figure 14.14.

Click **Create VPN Connection** to display the page shown in Figure 14.15.

You have the option of creating either a High-availability (HA) VPN or a Classic VPN. HA VPNs support dynamic routing using the Border Gateway Protocol (BGP) as well as a high availability 99.99 SLA within a region. High availability is provided by using two tunnels instead of just one. You can use either IPv4 or IPv6 addresses in an HA VPN.

FIGURE 14.14 Hybrid Connectivity section of Cloud Console**FIGURE 14.15** Creating a VPN connection, part 1

In the past, Google Cloud has offered a Classic VPN that supported both dynamic and static routing but only IPv4 addresses, and it did not provide for high availability. Classic VPN has been partially deprecated. You can continue to use Classic VPN tunnels that use dynamic routing when connecting to a Compute Engine VM running a VPN gateway. You cannot use Classic VPN tunnels for connections outside of Google Cloud. (See <https://cloud.google.com/network-connectivity/docs/vpn/deprecations/classic-vpn-deprecation> for additional details.)

Figure 14.16 shows the first part of the page for creating an HA VPN. You specify a VPN gateway name, a network, and a region.

FIGURE 14.16 Creating a high availability VPN

The screenshot shows the 'Create a VPN' wizard in the Google Cloud console. The first step is 'Create Cloud HA VPN gateway'. The page includes a back arrow and the title 'Create a VPN'. Below the step indicator, there is a description of HA VPN gateways and a 'Learn more' link. The form contains three required fields: 'VPN gateway name' (with the value 'ace-exam-ha-vpn-1' and a note that lowercase letters, numbers, and hyphens are allowed), 'Network' (with the value 'ace-exam-vpc1'), and 'Region' (with the value 'us-west2 (Los Angeles)' and a note that the region is permanent). Below these fields, there are sections for 'VPN gateway public IP address' (noting that two IP addresses will be automatically allocated) and 'VPN tunnel inner IP stack type' (with radio buttons for 'IPv4 (single-stack)' and 'IPv4 and IPv6 (dual-stack)'). At the bottom of the first step are 'CREATE & CONTINUE' and 'CANCEL' buttons. A progress indicator on the left shows four steps: '1 Create Cloud HA VPN gateway', '2 Add VPN tunnels', '3 Configure BGP sessions', and '4 Summary and reminder'.

← Create a VPN

1 Create Cloud HA VPN gateway

High Availability (HA) capable Cloud VPN gateways are regional resources with two interfaces, each interface with its own external IP address. HA VPN connects to an on-premises VPN gateway or another Cloud VPN gateway. [Learn more](#)

VPN gateway name *
ace-exam-ha-vpn-1 ?
Lowercase letters, numbers, hyphens allowed

Network *
ace-exam-vpc1 ?

Region *
us-west2 (Los Angeles) ?
Region is permanent

VPN gateway public IP address ?
Two IP addresses will be automatically allocated for each of your gateway interfaces

VPN tunnel inner IP stack type
The IP stack type will apply to all the tunnels associated with this VPN gateway.

☒ IPv4 (single-stack)
☐ IPv4 and IPv6 (dual-stack) ?

CREATE & CONTINUE CANCEL

2 Add VPN tunnels

3 Configure BGP sessions

4 Summary and reminder

Next, you will add VPN tunnels (see Figure 14.17). Tunnels connect the VPN gateway to a peer gateway that exists on premises, in another cloud, or in Google Cloud.

FIGURE 14.17 Configuring tunnels in an HA VPN

2

Add VPN tunnels

A VPN tunnel connects the Cloud VPN gateway to a peer gateway. Traffic sent through the tunnel is encrypted using the IPSec protocol operating in tunnel mode. [Learn more](#)

VPC network	ace-exam-vpc1
Region	us-west2
VPN gateway name	ace-exam-ha-vpn-1
Interfaces	0 : 34.124.1.127 1 : 34.104.67.235

Peer VPN gateway

☒ On-prem or Non Google Cloud
☐ Google Cloud

Peer VPN gateway name *

You can add more VPN tunnels to the same VPN gateway afterwards

CREATE & CONTINUE

CANCEL

In the Tunnels section, you configure the other network endpoint in the VPN. You specify a name, description, and IP address of the VPN gateway on your network. You will have the option to choose an existing peer VPN gateway or to create one. If you choose to create a peer VPN gateway, you will need to provide a name for it; specify 1, 2 or 4 interfaces; and provide the external IP address.



Real World Scenario

Analytics in the Cloud

Data science and data analysis are increasingly important to businesses. To derive insights from these practices, you need both the data and the tools. Data about customers, sales, and other kinds of transactions are often stored in a database in a company's data center. The tools analysts want to use, such as Spark and machine learning services, are readily available in the cloud. Many organizations have security practices to protect data and would not allow an analyst, for example, to download some data and then copy it over an unsecure Internet connection to the cloud. Instead, network and cloud engineers would create a

VPN between the company's data center and Google Cloud. This would ensure that network traffic between the data center and the cloud is encrypted. Analysts get access to the data and tools they need, and the information security professionals in the organization are able to protect the confidentiality and integrity of the data.

Creating a Virtual Private Network Using *gcloud*

To create a VPN at the command line, you can use these three commands:

- `gcloud compute target-vpn-gateways`
- `gcloud compute forwarding-rule`
- `gcloud compute vpn-tunnels`

The format of the `gcloud compute target-vpn-gateways` command for creating a Classic VPN is as follows:

```
gcloud compute vpn-tunnels create NAME --peer-address=PEER_ADDRESS \
--shared-secret=SHARED_SECRET --target-vpn-gateway=TARGET_VPN_GATEWAY
```

NAME is the name of the tunnel. *PEER_ADDRESS* is the IPv4 address of the remote tunnel endpoint. *SHARED_SECRET* is a secret string. *TARGET_VPN_GATEWAY* is a reference to the target VPN gateway IP.

When creating an HA VPN, you will need to specify either the `--peer-gcp-gateway` or the `--peer-external-gateway` parameter as well.

The format of `gcloud compute forwarding-rule` is as follows:

```
gcloud compute forwarding-rules create NAME --TARGET_SPECIFICATION=VPN_GATEWAY
```

NAME is the name of the forwarding rule. *TARGET_SPECIFICATION* is one of several target types, including `target-instance`, `target-http-proxy`, and `--target-vpn-gateway`. For additional details, see the documentation at <https://cloud.google.com/sdk/gcloud/reference/compute/forwarding-rules/create>.

The format of the `gcloud compute vpn-tunnels` command is as follows:

```
gcloud compute vpn-tunnels create NAME --peer-address=PEER_ADDRESS \
--shared-secret=SHARED_SECRET --target-vpn-gateway=TARGET_VPN_GATEWAY
```

NAME is the name of the VPN tunnel, *PEER_ADDRESS* is the IPv4 address of the remote tunnel, *SHARED_SECRET* is a secret string, and *TARGET_VPN_GATEWAY* is a reference to a VPN gateway.

Summary

This chapter reviewed how to create VPCs and VPNs. VPCs define networks in the Google Cloud to link your Google Cloud resources. VPNs in Google Cloud are used to link your Google Cloud networks to your internal networks. We discussed how to create VPCs, shared

VPCs, and subnets, and we described CIDR notation. You also learned how to configure VMs with custom network connections. Next, we reviewed firewall rules and how to create them. The chapter concluded with discussing the steps required to create a VPN.

Exam Essentials

Know that VPCs are logical data centers in the cloud and that VPNs are secure connections between your VPC subnets and your internal network. Your cloud resources are in a VPC. VPCs have subnets and routing rules for routing traffic between subnets. You control the flow of traffic using firewall rules.

Know that VPCs create subnets in each region when in auto mode. You can create additional subnets. Each subnet has a range of IP addresses. Firewall rules are applied to subnets, also called networks. Routers can be configured to learn just regional routes or global routes.

Understand how to read and calculate CIDR notation. CIDR notation represents a subnet mask and the size of available IP address in the IP range. The smaller the subnet mask size, which is the number after the slash in a CIDR block, the more IP addresses are available. The format of the CIDR address is an IP address followed by a slash, followed by the size of the subnet mask, such as 10.0.0.0/8.

Know that VPCs can be created using `gcloud` commands. A VPC can be created with `gcloud compute networks create`. A shared VPC can be created using `gcloud beta compute shared-vpc`. Shared VPCs can be shared at the network or the folder level. You will need to bind identity and access management (IAM) policies at the organizational or folder level to enable Shared VPC Admin roles. VPC peering can be used for interproject connectivity.

Understand that you can add network interfaces to a VM. You can configure these interfaces to use a particular subnet. You can assign ephemeral or static IP addresses.

Know that firewall rules control the flow of network traffic. Firewall rules consist of direction, priority, action, target, source/destination, protocols and port, and enforcement status. Firewall rules are applied to a subnet.

Know how to create a VPN with Cloud Console. VPNs route traffic between your cloud resources and your internal network. VPNs include gateways, forwarding rules, and tunnels. Both Classic and High Availability (HA) VPNs are available.

Review Questions


You can find the answers in the Appendix.

1. What kinds of a resource are virtual private clouds in Google Cloud?
 - A. Zonal
 - B. Regional
 - C. Super-regional
 - D. Global
2. You have been tasked with defining CIDR ranges to use with a project. The project includes two VPCs with several subnets in each VPC. How many CIDR ranges will you need to define?
 - A. One for each VPC
 - B. One for each subnet
 - C. One for each region
 - D. One for each zone
3. The legal department needs to isolate its resources on its own VPC. You want to have the network provide routing to any other service available on the global network. The VPC network has not learned global routes. What parameter may have been missed when creating the VPC subnets?
 - A. DNS server policy
 - B. Dynamic routing
 - C. Static routing policy
 - D. Systemic routing policy
4. The command used to create a VPC from the command line is:
 - A. `gcloud compute networks create`
 - B. `gcloud networks vpc create`
 - C. `gsutil networks vpc create`
 - D. `gcloud compute create networks`
5. You have created several subnets. Most of them are sending logs to Cloud Logging. One subnet is not sending logs. What option may have been misconfigured when creating the subnet that is not forwarding logs?
 - A. Flow Logs
 - B. Private IP Access
 - C. Cloud Logging
 - D. Variable-length subnet masking

6. At what levels of the resource hierarchy can a shared VPC be created?
 - A. Folders and resources
 - B. Organizations and project
 - C. Organizations and folders
 - D. Folders and subnets
7. You are using Cloud Console to create a VM that you want to exist in a custom subnet you just created. What section of the Create Instance page would you use to specify the custom subnet?
 - A. Networking tab of the Management, Security, Disks, Networking, Sole Tenancy section
 - B. Management tab of the Management, Security, Disks, Networking, Sole Tenancy section
 - C. Sole Tenancy tab of Management, Security, Disks, Networking, Sole Tenancy
 - D. Sole Tenancy tab of Management, Security, Disks, Networking
8. You want to implement interproject communication between VPCs. Which feature of VPCs would you use to implement this?
 - A. VPC network peering
 - B. Interproject peering
 - C. VPN
 - D. Interconnect
9. You want to limit traffic to a set of instances. You decide to set a specific network tag on each instance. What part of a firewall rule can reference the network tag to determine the set of instances affected by the rule?
 - A. Action
 - B. Target
 - C. Priority
 - D. Direction
10. What part of a firewall rule determines whether a rule applies to incoming or outgoing traffic?
 - A. Action
 - B. Target
 - C. Priority
 - D. Direction
11. You want to define a CIDR range that applies to all destination addresses. What IP address would you specify?
 - A. 0.0.0.0/0
 - B. 10.0.0.0/8
 - C. 172.16.0.0/12
 - D. 192.168.0.0/16

12. You are using `gcloud` to create a firewall rule. Which command would you use?
- A. `gcloud network firewall-rules create`
 - B. `gcloud compute firewall-rules create`
 - C. `gcloud network rules create`
 - D. `gcloud compute rules create`
13. You are using `gcloud` to create a firewall rule. Which parameter would you use to specify the subnet it should apply to?
- A. `--subnet`
 - B. `--network`
 - C. `--destination`
 - D. `--source-ranges`
14. An application development team is deploying a set of specialized service endpoints and wants to limit traffic so that only traffic going to one of the endpoints is allowed through by firewall rules. The service endpoints will accept any UDP traffic, and each endpoint will use a port in the range of 20000–30000. Which of the following commands would you use?
- A. `gcloud compute firewall-rules create fwr1 \`
`--allow=udp:20000-30000 --direction=ingress`
 - B. `gcloud network firewall-rules create fwr1 \`
`--allow=udp:20000-30000 --direction=ingress`
 - C. `gcloud compute firewall-rules create fwr1 --allow=udp`
 - D. `gcloud compute firewall-rules create fwr1 --direction=ingress`
15. You have a rule to allow inbound traffic to a VM. You want it to apply only if there is not another rule that would deny that traffic. What priority should you give this rule?
- A. 0
 - B. 1
 - C. 1000
 - D. 65535
16. You want to create a VPN using Cloud Console. What section of Cloud Console should you use?
- A. Compute Engine
 - B. App Engine
 - C. Hybrid Connectivity
 - D. IAM & Admin

17. Your company needs to ensure they have at least a 99.99 percent availability SLA for networking between on-premises networks and a VPC in Google Cloud. What should you use to ensure you have this level of availability?
- A. Classic VPN
 - B. HA VPN
 - C. Shared VPC
 - D. VPC network peering
18. You want the router on a tunnel you are creating to learn routes from all Google Cloud regions on the network. What feature of Google Cloud routing would you enable?
- A. Global dynamic routing
 - B. Regional routing
 - C. VPC
 - D. Firewall rules
19. What `gcloud` command would you use to create tunnels for a VPN?
- A. `gcloud network vpn-tunnels create`
 - B. `gcloud compute vpn-tunnels create`
 - C. `gcloud newtwork create vpn-tunnels`
 - D. `gcloud compute create vpn-tunnels`
20. You are using `gcloud` to create a VPN. Which command(s) would you use?
- A. `gcloud compute target-vpn-gateways only`
 - B. `gcloud compute forwarding-rule` and `gcloud compute target-vpn-gateways only`
 - C. `gcloud compute vpn-tunnels only`
 - D. `gcloud compute forwarding-rule`, `gcloud compute target-vpn-gateways`, and `gcloud compute vpn-tunnels`



Chapter 15

Networking in the Cloud: DNS, Load Balancing, Google Private Access, and IP Addressing

THIS CHAPTER COVERS THE FOLLOWING OBJECTIVES OF THE GOOGLE ASSOCIATE CLOUD ENGINEER CERTIFICATION EXAM:

- ✓ 2.4 Planning and configuring network resources
- ✓ 3.5 Deploying and implementing networking resources
- ✓ 4.5 Managing networking resources



This chapter continues the focus on networking, specifically configuring the Domain Name System (DNS), load balancing, Google Private Access, and managing IP addresses. Cloud DNS is a managed service providing authoritative domain naming services. It is designed for high availability, low latency, and scalability. Load balancing services in Google Cloud offer several types of load balancers to address a range of needs. In this chapter, you will see how HTTP(S), SSL Proxy, TCP Proxy, Network TCP/UDP, and Internal TCP/UDP Network differ and when to use each. Cloud engineers should also be familiar with managing IP addresses, in particular managing classless interdomain routing (CIDR) blocks and understanding how to reserve IP addresses. This chapter, in combination with Chapter 14, “Networking in the Cloud: Virtual Private Clouds and Virtual Private Networks,” provides an overview of the networking topics covered on the Associate Cloud Engineer exam.

Configuring Cloud DNS

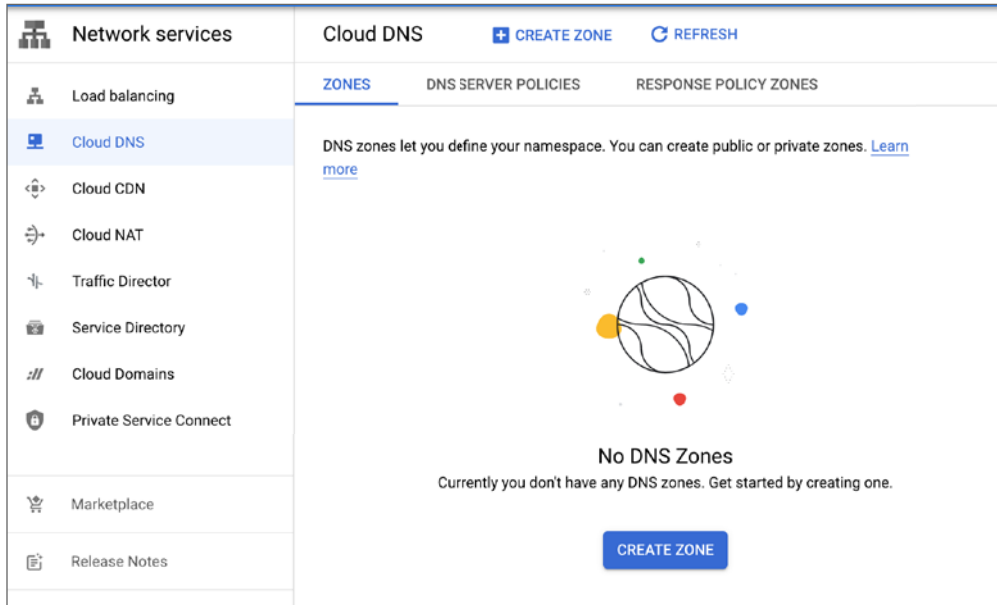
Cloud DNS is a Google service that provides domain name resolution. At the most basic level, DNS services map domain names, such as `example.com`, to IP addresses, such as `35.20.24.107`. A managed zone contains DNS records associated with a DNS name suffix, such as `aceexamdns1.com`. DNS records contain specific details about a zone. For example, an A record maps a hostname to IP addresses in IPv4. AAAA records are used in IPv6 to map names to IPv6 addresses. CNAME records map an alias to the canonical name of the domain. In this section, you will learn how to configure DNS services in Google Cloud, which consists of creating zones and adding records.

Creating DNS Managed Zones Using Cloud Console

To create a managed zone using Cloud Console, navigate to the Network Services section of the console. Click Cloud DNS to access the page shown in Figure 15.1.

Click Create Zone to the page shown in Figure 15.2.

First, select a zone type, which can be Public or Private. Then specify a zone name, which must be unique within the project.

FIGURE 15.1 Network Services Cloud DNS page

Public zones are accessible from the Internet. These zones provide name servers that respond to queries from any source. Private zones provide name services to your Google Cloud resources, such as virtual machines (VMs) and load balancers. Private zones respond only to queries that originate from resources in the same project as the zone.

In the form, provide a zone name and description. Specify the DNS name, which should be the suffix of a DNS name, such as `aceexamdns1.com`.

You can enable DNSSEC, which is DNS security. It provides strong authentication of clients communicating with DNS services. DNSSEC is designed to prevent spoofing (a client appearing to be some other client) and cache poisoning (a client sending incorrect information to update the DNS server).

If you choose to create a private zone, you will have the option of choosing settings that provide additional configurations for a private zone, as shown in Figure 15.3.

In addition to the parameters set for a public zone, you will need to specify the networks that will have access to the private zone.

After you've created some zones, the Cloud DNS page will list the zones, as shown in Figure 15.4.

FIGURE 15.2 Creating a public DNS zone

← Create a DNS zone

A DNS zone is a container of DNS records for the same DNS name suffix. In Cloud DNS, all records in a managed zone are hosted on the same set of Google-operated authoritative name servers. [Learn more](#)

If you don't have a domain yet, purchase one through [Cloud Domains](#).

Zone type ?

☐ Private

☒ Public

Zone name * ?

Example: example-zone-name

DNS name * ?

Example: myzone.example.com

DNSSEC *

Off ▼ ?

Description

Cloud Logging ?

☐ On

☒ Off

After creating your zone, you can add resource record sets and modify the networks your zone is visible on.

CREATE CANCEL

EQUIVALENT COMMAND LINE ▼

FIGURE 15.3 Additional configuration options for private DNS zones

Options * ?

Default (private)

Forward queries to another server ?

DNS Peering

Managed reverse lookup zone

Use a service directory namespace

ks your

CREATE CANCEL

FIGURE 15.4 List of DNS zones

The screenshot shows the Google Cloud Platform interface for Network services. The 'Cloud DNS' section is selected in the left sidebar. The 'ZONES' tab is active, showing a list of DNS zones. The table below represents the data shown in the screenshot:

Zone name	DNS name	DNSSEC	Description	Zone type
ace-exam-dns-zone	aceexamdns1.com.			Private

Click the name of a zone to see its details. As shown in Figure 15.5, the zone details include a list of records associated with the zone. When a zone is created, NS and SOA records are added. NS is a *name server* record that has the address of an authoritative server that manages the zone information. SOA is a *start of authority* record, which has authoritative information about the zone. You can add other records, such as A and CNAME records.

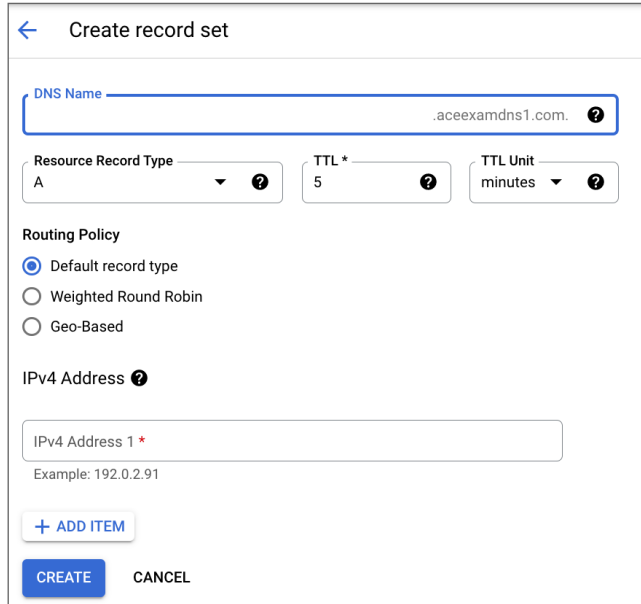
FIGURE 15.5 List of records in a DNS zone

The screenshot shows the 'Zone details' page for the 'ace-exam-dns-zone'. The page displays the zone's DNS name and type, and a list of record sets. The table below represents the data shown in the screenshot:

DNS name	Type	TTL (seconds)	Routing policy
aceexamdns1.com.	SOA	21600	Default
aceexamdns1.com.	NS	21600	Default

To add an A record, click Add Record Set to display the page shown in Figure 15.6.

FIGURE 15.6 Creating an A record set



The screenshot shows the 'Create record set' form in Google Cloud DNS. At the top, there is a back arrow and the title 'Create record set'. Below this is a 'DNS Name' field with a blue border, containing '.aceexamdns1.com.' and a help icon. Underneath are three fields: 'Resource Record Type' with a dropdown menu showing 'A', 'TTL *' with a value of '5', and 'TTL Unit' with a dropdown menu showing 'minutes'. Below these is the 'Routing Policy' section with three radio buttons: 'Default record type' (selected), 'Weighted Round Robin', and 'Geo-Based'. The 'IPv4 Address' section has a header with a help icon and a text input field labeled 'IPv4 Address 1 *'. Below the input field is an example: 'Example: 192.0.2.91'. At the bottom left is a '+ ADD ITEM' button, and at the bottom right are 'CREATE' and 'CANCEL' buttons.

Select A as a resource record type and specify an IPv4 address of the server that maps domain names to IP addresses for this zone.

The TTL (time to live) and TTL Unit parameters specify how long the record can live in a cache—in other words, the period of time DNS resolvers should cache the data before querying for the value again. DNS resolvers perform lookup operations mapping domain names to IP addresses. If you want to specify multiple IP addresses in the record, click Add Item to add other IP addresses.

You can also add canonical name records using the Create Record Set page. In this case, select CNAME as the Resource Record Type, as shown in Figure 15.7.

The CNAME record takes a name, or alias, of a server. The DNS name and TTL parameters are the same as in the A record example.

Also, DNS Forwarding is now available, which allows your DNS queries to be passed to an on-premises DNS server if you are using Cloud VPN or Interconnect.

FIGURE 15.7 Creating a CNAME record

← Create record set

DNS Name .aceexamdns1.com. ⓘ

Resource Record Type CNAME ⓘ

TTL * 5 ⓘ

TTL Unit minutes ⓘ

Routing Policy

☒ Default record type

☐ Weighted Round Robin

☐ Geo-Based

Canonical name ⓘ

Canonical name 1 *

Example: server-1.example.com.

+ ADD ITEM

CREATE CANCEL

Creating DNS Managed Zones Using *gcloud*

To create DNS zones and add records, you will use `gcloud dns managed-zones` and `gcloud dns record-sets transaction`.

To create a managed public zone called `ace-exam-zone1` with the DNS suffix `aceexamzone.com`, use this:

```
gcloud dns managed-zones create ace-exam-zone1 --description= "A
sample zone" --dns-name=aceexamzone.com.
```

To make this a private zone, you add the `--visibility` parameter set to `private`:

```
gcloud dns managed-zones create ace-exam-zone1 --description= "A
sample zone" --dns-name=aceexamzone.com. --visibility=private --
networks=default
```

To add an A record, you start a transaction, add the A record information, and then execute the transaction.

Transactions are started with `gcloud dns record-sets transaction start`. Record sets are added using `gcloud dns record-sets transaction add`, and

transactions are completed using `gcloud dns record-sets transaction execute`. Together, the steps are as follows:

```
gcloud dns record-sets transaction start --zone=ace-exam-zone1
gcloud dns record-sets transaction add 192.0.2.91 --name=aceexamzone.com.
--ttl=300 --type=A --zone=ace-exam-zone1
gcloud dns record-sets transaction execute --zone=ace-exam-zone1.
```

To create a CNAME record, you would use similar commands:

```
gcloud dns record-sets transaction start --zone=ace-exam-zone1
gcloud dns record-sets transaction add server1.aceexamezone.com. \
--name=www2.aceexamzone.com. --ttl=300 --type=CNAME --zone=ace-exam-zone1
gcloud dns record-sets transaction execute --zone=ace-exam-zone1
```

Configuring Load Balancers

Load balancers distribute workload to servers running an application. In this section, we will discuss the different types of load balancers and how to configure them.

Types of Load Balancers

Load balancers can distribute load within a single region or across multiple regions. The several load balancers offered by Google Cloud are characterized by three features:

- Global versus regional load balancing
- External versus internal load balancing
- Traffic type, such as HTTP and TCP

Global load balancers are used when an application is globally distributed. There are four global load balancers:

- Global External HTTP(S) Load Balancing, which balances HTTP and HTTPS loads across a set of back-end instances globally on a Premium network service tier.
- Global External HTTP(S) Load Balancing (classic), which balances HTTP and HTTPS loads across a set of back-end instances globally on Premium tier networking and regionally on Standard tier networking.
- SSL Proxy, which terminates SSL/TLS connections, which are Secure Socket Layer connections. This type is used for non-HTTPS traffic.
- TCP Proxy, which terminates TCP sessions at the load balancer and then forwards traffic to back-end servers.

Regional load balancers are used when resources providing an application are in a single region. The regional load balancers are as follows:

- Regional External HTTP(S) Load Balancing, which balances HTTP(S) regionally on Standard tier networking
- Internal HTTP(S) Load Balancing, which balances HTTP(S) regionally on Premium tier networking only
- Internal TCP/UDP Load Balancing, which balances TCP/UDP regionally on Premium tier networking only
- External TCP/UDP Network Load Balancing, which enables balancing of TCP, UDP, and other protocols regionally on Standard or Premium tier networking



Real World Scenario

Load Balancing and High Availability

Applications that need to be highly available should use load balancers to distribute traffic and to monitor the health of VMs in the back end. A company offering API access to customer data will need to consider how to scale up and down in response to changes in load and how to ensure high availability.

The combination of instance groups (Chapter 6, “Managing Virtual Machines”) and load balancers solves both problems. Instance groups can manage autoscaling, and load balancers can perform health checks. If a VM is not functioning, the health checks will fail and take the failed VM out of rotation for traffic. Users of the API are less likely to get failed response codes when instance groups keep an appropriate number of VMs active and load balancers prevent any traffic from being routed to failed servers.

Configuring Load Balancers Using Cloud Console

To create a load balancer in Cloud Console, navigate to the Network Services section and select Load Balancing, as shown in Figure 15.8.

The first step to creating a load balancer is deciding on the type. In this example, you will create a TCP load balancer (see Figure 15.9).

After you select the TCP Load Balancing option, the page shown in Figure 15.10 appears. Select Only Between My VMs for private load balancing. This load balancer will be used in a single region, and you will not offload TCP or SSL processing.

FIGURE 15.8 Network Services, Load Balancing section

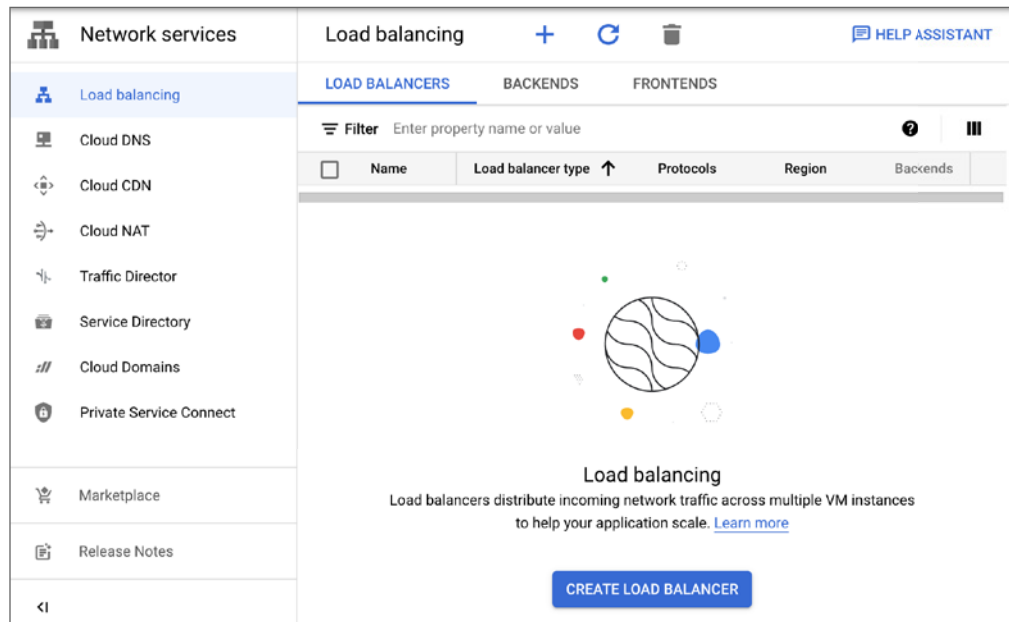
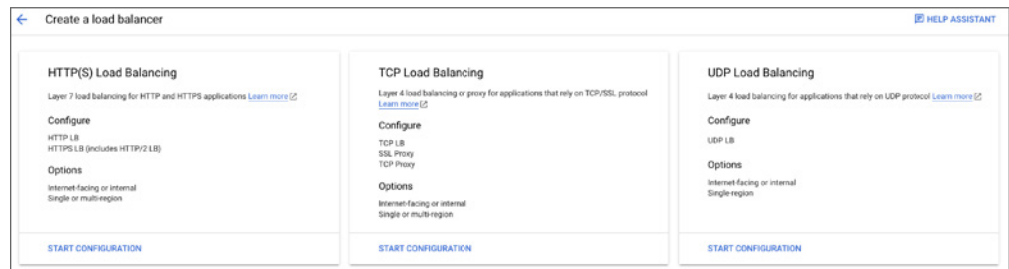


FIGURE 15.9 Create A Load Balancer options



You will need to specify if you want the load balancer to handle traffic from the Internet to your VMs or only between VMs on your network. Next, specify if you want to support a single region or multiple regions. You will also specify a back-end type, which can be Backend Service, Target Pool, or Target Instance. Backend Service allows you to specify how to distribute traffic as well as support for connection draining, TCP health checks, managed instance groups, and failover groups. Target Pools are instances within a region that are identified by a list or URLs that specify what VMs can receive traffic.

FIGURE 15.10 Creating a TCP balancer

← Create a load balancer

Please answer a few questions to help us select the right load balancing type for your application

Internet facing or internal only

Do you want to load balance traffic from the Internet to your VMs or only between VMs in your network?

☐ From Internet to my VMs

☒ Only between my VMs

Multiple regions or single region

Do you want to place the backends for your load balancer in a single region or across multiple regions?

☐ Multiple regions (or not sure yet)

☒ Single region only

Load Balancer type

Do you want a pass-through load balancer or a proxy load balancer? ?

☒ Pass-through ?

☐ Proxy ?

CONTINUE

Figure 15.11 shows the parameters for configuring a back end, including IP Stack Type, Health Check, and Session Affinity.

You can configure a health check for the back end. This will bring up a separate page, as shown in Figure 15.12.

In the health check, you specify a name, a protocol and a port, and a set of health criteria. In this case, you check back ends every 5 seconds and will wait for a response for up to 5 seconds. If you have two consecutive periods where the health check fails, then the server will be considered unhealthy and taken out of the load balancing rotation.

Next, you configure the front end using the page in Figure 15.13. You specify a name, subnetwork, and an internal IP configuration, which in this case is ephemeral (see “Managing IP Addresses” later in this chapter for more on types of IP addresses). You also specify the port that will have its traffic forwarded to the back end. In this example, you are forwarding traffic on port 80.

The last step prior to creating the front end is to review the configuration and then create the load balancer.

FIGURE 15.11 Configuring the back end

← New TCP load balancer

Name *

Lowercase, no spaces.
Name is permanent

Region *

Backend configuration

Frontend configuration

Review and finalize (optional)

Backend configuration

Backend service

Backends

New backend

IP stack type

IPv4 (single-stack)

IPv4 and IPv6 (dual-stack)

Instance group *

☐ Use this instance group as a failover group for backup

CANCEL

DONE

ADD BACKEND

Health check *

Session affinity

None

ADVANCED CONFIGURATIONS

CREATE

CANCEL

Configuring Load Balancers Using *gcloud*

In this section, we will review the steps needed to create a network load balancer. These are good options when you need to load balance protocols in addition to HTTP(S).

FIGURE 15.12 Creating a health check

Health Check

Name *

?

Lowercase, no spaces.

Description

Region

us-west1 (Oregon)

?

Protocol

TCP

Port *

80

?

Proxy protocol

NONE

Request

?

Response

?

Logs

☐ On

Turning on Health check logs can increase costs in Cloud Logging.

☒ Off

Health criteria

Define how health is determined: how often to check, how long to wait for a response, and how many successful or failed attempts are decisive

Check interval *

5

seconds

?

Timeout *

5

seconds

?

Healthy threshold *

2

consecutive successes

?

Unhealthy threshold *

2

consecutive failures

?

SAVE

CANCEL

FIGURE 15.13 Configuring the front end

← New TCP load balancer

Name * ?
Lowercase, no spaces.
Name is permanent

Region *
us-west1 (Oregon) ?

- Backend configuration
- Frontend configuration**
- Review and finalize (optional)

Frontend configuration

Specify an IP address, port and protocol. This IP address is the frontend IP for your clients requests.

New Frontend IP and port

Name (Optional) ?
Lowercase, no spaces.
Name is permanent

Description

Protocol
TCP

IP version
IPv4

Network Service Tier ?
☒ Premium (Current project-level tier, [change](#)) ?
☐ Standard ?

IP address
Ephemeral

Ports ?
☒ Single
☐ Multiple
☐ All

Port number *

CREATE CANCEL

The `gcloud compute forward-rules` command is used to forward traffic that matches an IP address to the load balancer:

```
gcloud compute forwarding-rules create ace-exam-lb --port=80 \
--target-pool ace-exam-pool
```

This command routes traffic to any VM in the `ace-exam-pool` to the load balancer called `ace-exam-lb`.

Target pools are created using the `gcloud compute target-pools create` command. Instances are added to the target pool using the `gcloud compute target-pools add-instances` command. For example, to add VMs `ig1` and `ig2` to the target pool called `ace-exam-pool`, use the following command:

```
gcloud compute target-pools add-instances ace-exam-pool --instances ig1,ig2
```

Google Private Access

VMs running in a VPC can use external IP addresses to connect to Google APIs and other services. However, if you do not want to assign external IP addresses to VMs, you can use one of the Private Google Access options.

Private Google Access is used to reach Google Cloud resources without using an external IP address. This allows you to connect to Google APIs through the VPC's default network gateway. This option supports most Google Cloud services and APIs. Traffic to Google APIs from on-premises systems needs to be sent to either `private.googleapis.com` or `restricted.googleapis.com`. Also, routes must be in place for traffic to flow from on-premises systems to `private.googleapis.com` or `restricted.googleapis.com`.

Private Service Access is used to access a Google or third-party managed VPC network through a VPC Peering connection. This supports some Google Cloud services and third-party services.

Private Service Connect is used with Google Cloud resources that may or may not have external IP addresses as well as on-premises systems. With this option, you connect to a Private Service Connect endpoint in your VPC network and that endpoint will forward requests to Google APIs and services.

If you are using Cloud Run, App Engine Standard, and Cloud Functions, then you can use Serverless VPC Access to reach private IP addresses from those services.

Managing IP Addresses

The exam topics for the Associate Cloud Engineer certification specifically identifies two IP address-related topics: expanding CIDR blocks and reserving IP addresses.



It is also important to understand the difference between ephemeral and static IP addresses. Static IP addresses are assigned to a project until they are released. They are used if you need a fixed IP address for a service, such as a website. Ephemeral IP addresses exist only as long as the resource is using the IP address, such as on a VM running an application only accessed by other VMs in the same project. If you delete or stop a VM, ephemeral addresses are released.

Expanding CIDR Blocks

CIDR blocks define a range of IP addresses that are available for use in a subnet. If you need to increase the number of addresses available—for example, if you need to expand the size of clusters running in a subnet—you can use the `gcloud compute networks subnets expand-ip-range` command. It takes the name of the subnet and a new prefix length. The prefix length determines the size of the network mask.

For example, to increase the number of addresses in `ace-exam-subnet1` to 65,536, you set the prefix length to 16:


```
gcloud compute networks subnets expand-ip-range ace-exam-subnet1 \
--prefix-length 16
```


This assumes the prefix length was larger than 16 prior to issuing this command. The `expand-ip-range` command is used only to increase the number of addresses. You cannot decrease them, though. You would have to re-create the subnet with a smaller number of addresses.


Reserving IP Addresses


Static external IP addresses can be reserved using Cloud Console or the command line. To reserve a static IP address using Cloud Console, navigate to the Virtual Private Cloud (VPC) section of the console and select IP Addresses. This will display a page like the one shown in Figure 15.14.


FIGURE 15.14 VPC Network IP Address page


 VPC network

 VPC networks

 IP addresses

 Bring your own IP

 Firewall

 Routes

IP addresses

<

ALL

INTERNAL IP ADDRESSES

EXTERNAL IP ADDRESSES

IPv4 ADDRESSES

>

Filter

Enter property name or value

?

⋮

<input type="checkbox"/>	Name	IP address	Access type	Region	Type ↓	Version
<input type="checkbox"/>	default-ip-range	10.87.224.0	Internal		Static	IPv4

Click Reserve External Static Address to display the page shown in Figure 15.15, where you can reserve an IP address.

When reserving an IP address, you will need to specify a name and optional description. You may have the option of using the lower-cost Standard service tier for networking, which uses the Internet for some transfer of data. The Premium tier routes all traffic over Google's global network. You will also need to determine whether the address is in IPv4 or IPv6 and whether it's regional or global. You can attach the static IP address to a resource as part of the reservation process, or you can keep it unattached.

FIGURE 15.15 Reserving a static IP address

The screenshot shows the 'Reserve a static address' form in the Google Cloud console. The form includes the following fields and options:

- Name ***: A text input field with a help icon. Below it, a note states: 'Lowercase letters, numbers, hyphens allowed'.
- Description**: A text input field with a help icon.
- Network Service Tier**: Two radio button options:
 - ☒ Premium (Current project-level tier, [change](#))
 - ☐ Standard
- IP version**: Two radio button options:
 - ☒ IPv4
 - ☐ IPv6
- Type**: Two radio button options:
 - ☒ Regional
 - ☐ Global (to be used with Global forwarding rules [Learn more](#))
- Region**: A dropdown menu showing 'us-central1 (Iowa)' with a help icon.
- Attached to**: A dropdown menu showing 'None' with a help icon. Below it, a note states: 'Some of the instances may be disabled due to the 'External IPs for VM instances' organization policy. [Learn more](#)'.

At the bottom, there is a warning box with a yellow triangle icon: 'Static IP addresses not attached to an instance or load balancer are billed at a higher hourly rate [Pricing details](#)'. Below the warning box are two buttons: 'RESERVE' (in blue) and 'CANCEL'.

Reserved addresses stay attached to a VM when it is not in use and stay attached until released. This is different from ephemeral addresses, which are released automatically when a VM shuts down.

To reserve an IP address using the command line, use the `gcloud` command `gcloud compute addresses create`. For example, to create a static IP address in the `us-west2` region, which uses the Premium tier, use this command:

```
gcloud compute addresses create ace-exam-reserved-static1 \
--region=us-west2 --network-tier=PREMIUM
```

Summary

The Associate Cloud Engineer exam may test your knowledge of Cloud DNS, load balancing, and managing IP addresses. Cloud DNS is an authoritative name service for mapping domain names to IP addresses. You can set up public or private DNS zones. You will

also need to be familiar with load balancing and the different types of load balancers. Some load balancers are regional, and some are global. Some are for internal use only, and others support external sources of traffic. The chapter also reviewed how to expand the number of addresses available in a subnet and discussed how to reserve IP addresses.

Exam Essentials

Understand that Cloud DNS is used to map domain names to IP addresses. If you want to support queries from the Internet, use a public DNS zone. Use a private DNS zone only if you want to accept queries from resources in your project.

Know that DNS entries, like `example.com`, can have multiple records associated with them. The A record specifies the address of a DNS resolver that maps domain names to IP addresses. CNAME records store the canonical name of the domain.

Know how load balancers are distinguished. Load balancers are distinguished based on global versus regional load balancing, external versus internal load balancing, and the protocols supported. Global balancers distribute load across regions, whereas regional load balancers work within a region. Internal load balancers balance traffic only from within Google Cloud, not external sources. Some load balancers are protocol-specific, such as HTTP and SSL load balancers.

Know the types of load balancers and when they should be used. HTTP(S), SSL Proxy, TCP Proxy, and TCP/UDP. Load balancers distribute load regionally or globally. Internal load balancers distribute load from internal traffic. External load balancers distribute load from external traffic.

HTTP(S) balances HTTP and HTTPS load.

SSL Proxy terminates SSL/TLS connections.

TCP Proxy terminates TCP sessions.

TCP/UDP balances TCP/UDP traffic on private networks hosting internal VMs.

Understand that configuring a load balancer can require configuring both the front end and back end. The network load balancer can be configured by specifying a forwarding rule that routes traffic to the load balancer to VMs in the target pool.

Know Google Private Access options. Private Google Access is used for private access to most Google Cloud services, while Private Service Access is used with third-party services and some Google Cloud service. Private Service Connect uses a VPC endpoint for forwarding traffic to Google Cloud services. Serverless VPC Access allows Cloud Run, Cloud Functions, and App Engine Standard to reach VMs with private addresses.

Know how to increase the number of IP addresses in a subnet. Use the `gcloud compute network subnets expand-ip-range` command to increase IP addresses in a subnet. The number of addresses can only increase. The `expand-ip-range` command cannot be used to decrease the number of addresses.

Know how to reserve an IP address using the console and the `gcloud compute address create` command. Reserved IP addresses continue to be available to your project even if they are not attached to a resource. Know the difference between Premium and Standard tier network services.

Review Questions


You can find the answers in the Appendix.

1. What record type is used to specify the IPv4 address of a domain?
 - A. AAAA
 - B. A
 - C. NS
 - D. SOA
2. The CEO of your startup just read a news report about a company that was attacked by something called cache poisoning. The CEO wants to implement additional security measures to reduce the risk of DNS spoofing and cache poisoning. What would you recommend?
 - A. Using DNSSEC
 - B. Adding SOA records
 - C. Adding CNAME records
 - D. Deleting CNAME records
3. What do the TTL parameters specify in a DNS record?
 - A. Time a record can exist in a cache before it should be queried again
 - B. Time a client has to respond to a request for DNS information
 - C. Time allowed to create a CNAME record
 - D. Time before a human has to manually verify the information in the DNS record
4. What command is used to create a DNS zone in the command line?
 - A. `gsutil dns managed-zones create`
 - B. `gcloud dns managed-zones create`
 - C. `gcloud managed-zones create`
 - D. `gcloud create dns managed zones`
5. What parameter is used to make a DNS zone private?
 - A. `--private`
 - B. `--visibility=private`
 - C. `--private=true`
 - D. `--status=private`
6. Which load balancers provide global load balancing?
 - A. Global External HTTP(S) Load Balancing and Global External HTTP(S) Load Balancing (classic) only
 - B. SSL Proxy and TCP Proxy only
 - C. Global External HTTP(S) Load Balancing, Global External HTTP(S) Load Balancing (classic), SSL Proxy, and TCP Proxy
 - D. Internal TCP/UDP, HTTP(S), SSL Proxy, and TCP Proxy

7. Which regional load balancer balances HTTP(S) regionally on Premium tier networking only?
 - A. Global External HTTP(S) Load Balancing
 - B. SSL Proxy
 - C. TCP Proxy
 - D. Internal HTTP(S) Load Balancing
8. You are configuring a load balancer and want to implement private load balancing. Which option would you select?
 - A. Only Between My VMs
 - B. Enable Private
 - C. Disable Public
 - D. Local Only
9. What two components need to be configured when creating a TCP Proxy load balancer?
 - A. Front end and forwarding rule
 - B. Front end and back end
 - C. Forwarding rule and back end only
 - D. Back end and forwarding rule only
10. A health check is used to check what resources?
 - A. Organization policies
 - B. VMs
 - C. Storage buckets
 - D. Persistent disks
11. Where do you specify the ports on a TCP Proxy load balancer that should have their traffic forwarded?
 - A. Back end
 - B. Front end
 - C. Network Services section
 - D. VPC
12. What command is used to create a network load balancer at the command line?
 - A. `gcloud compute forwarding-rules create`
 - B. `gcloud network forwarding-rules create`
 - C. `gcloud compute create forwarding-rules`
 - D. `gcloud network create forwarding-rules`

13. A team is setting up a web service for internal use. They want to use the same IP address for the foreseeable future. What type of IP address would you assign?
- A. Internal
 - B. External
 - C. Static
 - D. Ephemeral
14. You are starting up a VM to experiment with a new Python data science library. You'll use SSH to connect to the VM, use the Python interpreter interactively for a while, and then shut down the machine. What type of IP address would you assign to this VM?
- A. Ephemeral
 - B. Static
 - C. Permanent
 - D. IPv8
15. You have created a subnet called sn1 using 192.168.0.0 with 65,534 addresses. You realize that you will not need that many addresses, and you'd like to reduce that number to 254. Which of the following commands would you use?
- A. `gcloud compute networks subnets expand-ip-range sn1 \ --prefix-length=24`
 - B. `gcloud compute networks subnets expand-ip-range sn1 \ --prefix-length=-8`
 - C. `gcloud compute networks subnets expand-ip-range sn1 --size=256`
 - D. There is no command to reduce the number of IP addresses available.
16. You have created a subnet called sn1 using 192.168.0.0. You want it to have 14 addresses. What prefix length would you use?
- A. 32
 - B. 28
 - C. 20
 - D. 16
17. You want all your network traffic to route over the Google network and not traverse the public Internet. What level of network service should you choose?
- A. Standard
 - B. Google-only
 - C. Premium
 - D. Non-Internet

- 18.** You have a website hosted on a Compute Engine VM. Users can access the website using the domain name you provided. You do some maintenance work on the VM and stop the server and restart it. Now users cannot access the website. No other changes have occurred on the subnet. What might be the cause of the problem?
- A.** The restart caused a change in the DNS record.
 - B.** You used an ephemeral instead of a static IP address.
 - C.** You do not have enough addresses available on your subnet.
 - D.** Your subnet has changed.
- 19.** You are deploying a distributed system. Messages will be passed between Compute Engine VMs using a reliable UDP protocol. All VMs are in the same region. You want to use the load balancer that best fits these requirements. Which kind of load balancer would you use?
- A.** Internal TCP/UDP
 - B.** TCP Proxy
 - C.** SSL Proxy
 - D.** Global External HTTP(S) Load Balancing
- 20.** You want to use Cloud Console to review the records in a DNS entry. What section of Cloud Console would you navigate to?
- A.** Compute Engine
 - B.** Network Services
 - C.** Kubernetes Engine
 - D.** Hybrid Connectivity



Chapter 16

Deploying Applications with Cloud Marketplace and Cloud Foundation Toolkit

**THIS CHAPTER COVERS THE FOLLOWING
OBJECTIVES OF THE GOOGLE ASSOCIATE
CLOUD ENGINEER CERTIFICATION EXAM:**

- ✓ 3.6 Deploying a solution using Cloud Marketplace
- ✓ 3.7 Implementing resources via infrastructure as code



Throughout this study guide, you have learned how to deploy computing, storage, and networking resources, and now you will turn your attention to deploying applications. Cloud Marketplace is a Google Cloud service for finding and deploying preconfigured applications that are ready to run the Google Cloud. Cloud Marketplace lets users deploy applications and necessary compute, storage, and network resources without having to configure those resources themselves. Cloud Foundation Toolkit is a suite of tools used to streamline deploying infrastructure as code.

Deploying a Solution Using Cloud Marketplace

Cloud Marketplace is a central repository of applications and data sets that can be deployed to your Google Cloud environment. Working with the Cloud Marketplace is a two-step process: browsing for a solution that fits your needs and then deploying the solution.

Browsing Cloud Marketplace and Viewing Solutions

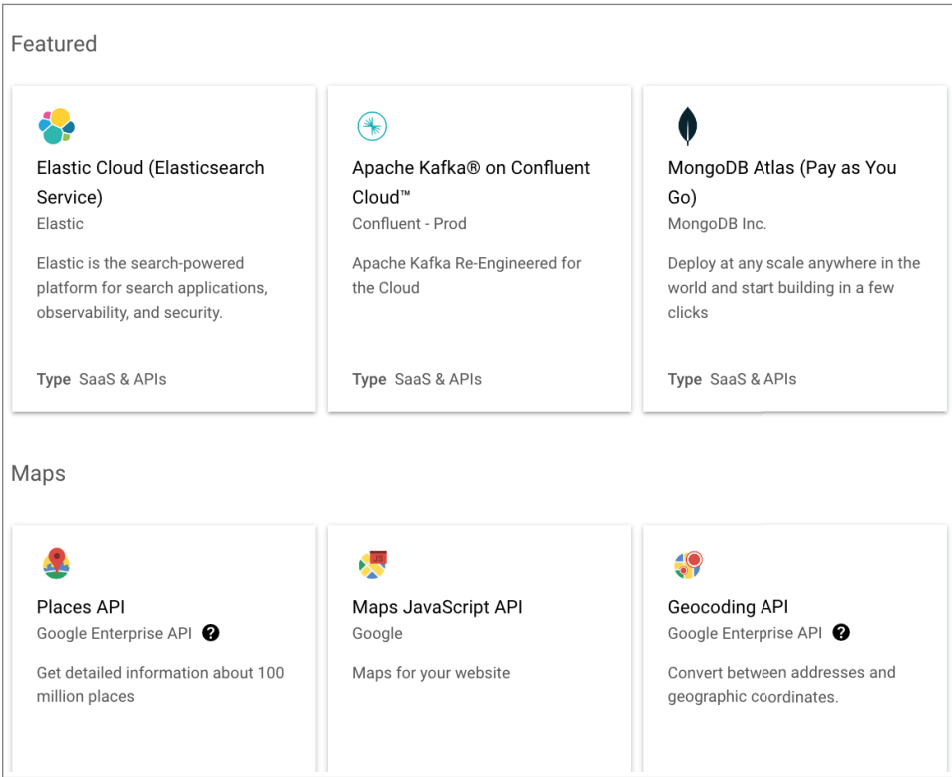
To view the solutions available in Cloud Marketplace, navigate to the Marketplace section. This will display a page like that shown in Figure 16.1.

The main page of Cloud Marketplace shows some featured solutions. The solutions shown in Figure 16.1 include Elastic Cloud (Elasticsearch Service), Apache Kafka on Confluent Cloud, MongoDB Atlas, as well as APIs for geocoding and directions.

You can either search or browse by filter to see the list of solutions. Figure 16.2 shows the list of categories of available solutions.

You can narrow the set of solutions displayed on the main page by choosing a particular category. For example, if you filter to see Big Data only, you will see a list of options, as shown in Figure 16.3. You can see a list of available operating systems in Figure 16.4.

FIGURE 16.1 Cloud Marketplace main page



Notice that you can further filter the list of operating systems by license type. The license types are free, flat hourly rate, usage fees, and bring your own license (BYOL). Free operating systems include Linux and FreeBSD options. The operating systems available for a fee include Windows and enterprise-supported Linux. You will be charged a fee based on your usage, and that charge will be included in your Google Cloud billing. The BYOL option includes two supported Linux operating systems that require you to have a valid license to run the software. You are responsible for acquiring the license before running the software.

Figure 16.5 shows a sample of developer tools available in Cloud Marketplace. These include WordPress, Joomla, and Alfresco.

Let’s take a look at the kind of information provided along with the solutions listed in Cloud Marketplace. Figure 16.6 shows the bulk of the information available. It includes an overview, pricing information, and details about the contents of the package. There is also information on where the solution will run within Google Cloud.

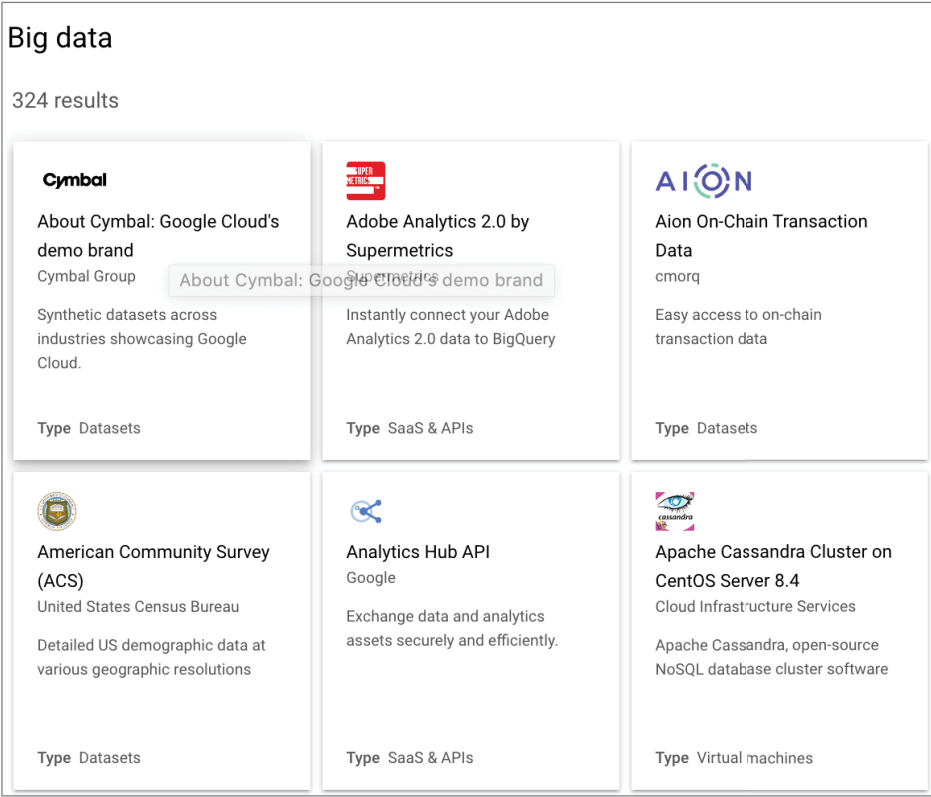
FIGURE 16.2 Filtering by category

Filter Type to filter	
Category ^	
Maps	(30)
Big data	(324)
Analytics	(420)
Databases	(265)
Machine learning	(163)
Type ^	
Virtual machines	(1,225)
SaaS & APIs	(891)
Google Cloud Platform	(51)
Kubernetes apps	(101)
Container images	(103)
Price v	
Partner v	

Pricing information (see Figure 16.7) is also shown on the overview page. These are estimated costs for running the solution, as configured, for one month, which includes the costs of VMs, persistent disks, and any other resources. The price estimate also includes discounts for sustained usage of Google Cloud resources, which are applied as you reach a threshold based on the amount of time a resource is used.

The last sections of the product overview page provide information and links to documentation and tutorials as well as support information.

FIGURE 16.3 Big Data options available in Cloud Marketplace

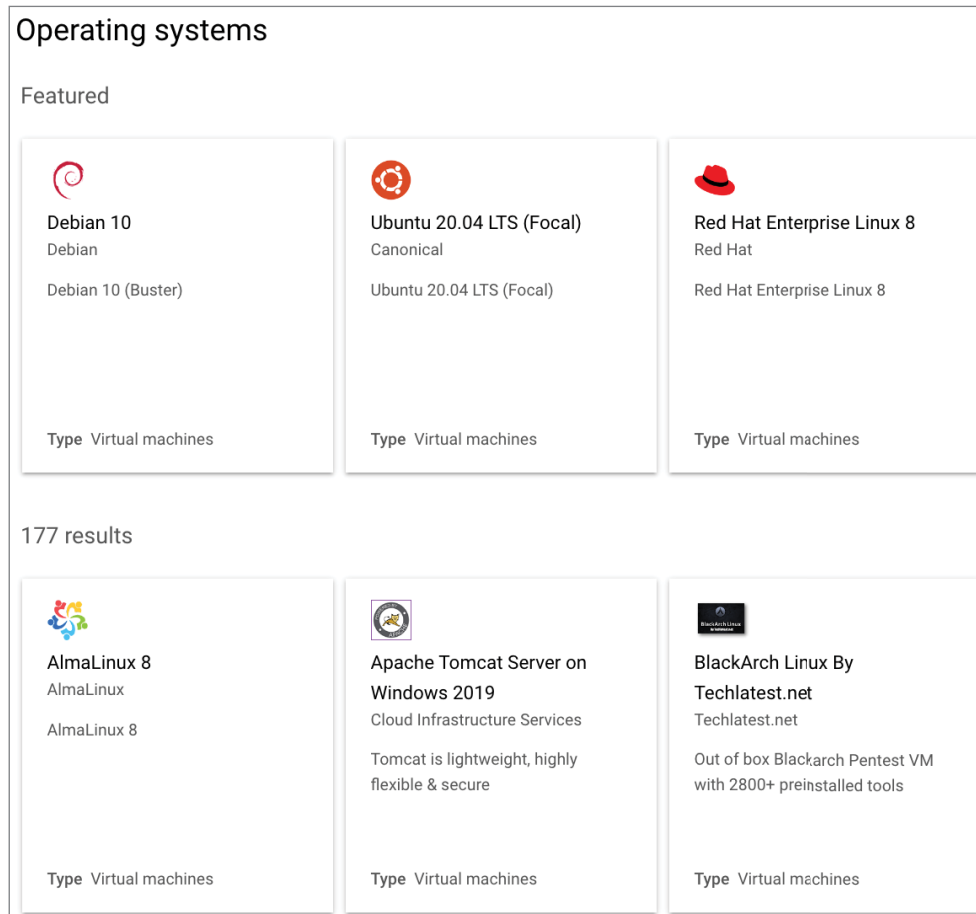


Deploying Cloud Marketplace Solutions

After you identify a solution that meets your needs, you can launch it from Cloud Marketplace.

Go to the overview page of the product you would like to launch, as shown in Figure 16.9, and select Launch.

This will open the page shown in Figure 16.10. You may see a message stating that additional APIs must be enabled to deploy a solution. In that case, enable the additional APIs, and once you do, the page in Figure 16.10 will appear.

FIGURE 16.4 Operating systems available in Cloud Marketplace


The contents of this page will vary by application, but many parameters are common across solutions. On this page, you specify a name for the deployment, a zone, and the machine type.

You can choose the type and size of the persistent disk. In this example, the solution will deploy to a 2 vCPU server with 8 GB of memory and a 10 GB boot disk using standard persistent disks. If you wanted, you could opt for an SSD disk for the boot disk. You can also change the size of the boot disk.

FIGURE 16.5 Developer tools available in Cloud Marketplace

Developer tools


Featured



WordPress Certified by Bitnami and Automattic
Bitnami

Up-to-date, secure, and ready to run.


Type Virtual machines



Joomla! packaged by Bitnami
Bitnami

Up-to-date, secure, and ready to run.

Type Virtual machines




Alfresco Community packaged by Bitnami
Bitnami

Up-to-date, secure, and ready to run.

Type Virtual machines

431 results




Cymbal

About Cymbal: Google Cloud's demo brand

Cymbal Group

Synthetic datasets across industries showcasing Google Cloud.


Type Datasets



Actifio Global Manager
Actifio

Web scale management for Actifio Sky appliances

Type Virtual machines




Actifio Sky
Actifio

Enterprise Class Backup and Recovery

Type Virtual machines

In the Networking section, you can specify the network and subnet to launch the VM. You can also configure firewall rules to allow HTTP and HTTPS traffic. In addition, you can specify source IP ranges for HTTP and HTTPS traffic. If you expand the Networking section, you will see additional parameters for specifying network, subnetwork, and external IP addresses. (See Figure 16.11.)

FIGURE 16.6 Overview page of a WordPress solution



WordPress Certified by Bitnami and Automattic

Bitnami

Up-to-date, secure, and ready to run.

[LAUNCH](#)[VIEW PAST DEPLOYMENTS](#)

[OVERVIEW](#)[PRICING](#)[DOCUMENTATION](#)[SUPPORT](#)

Overview

Bitnami, the leaders in application packaging, and Automattic, the experts behind WordPress, have teamed up to offer this official WordPress image on Google Cloud Marketplace.

WordPress is the world's most popular content management platform. Whether it's for an enterprise or small business website, or a personal or corporate blog, content authors can easily create content using its new Gutenberg editor, and developers can extend the base platform with additional features.

For content authors, the Jetpack plugin (enabled by default) offers access to additional professional themes, performance improvements, scanning, site activity, and marketing tools. Other popular plugins like Akismet, All in One SEO Pack, WP Mail and Google Analytics for WordPress also come pre-installed. Optional automatic backup and priority support are available from Automattic.

For developers, this image features the AMP for WordPress plugin. This plugin automatically adds Accelerated Mobile Pages (Google AMP Project) support to deliver a faster, higher-performance and more flexible web experience across distribution platforms. It helps you reduce the operating and development costs of your site by pairing your content to the format required by the destination platform and making the user experience consistent across devices.

This image includes the latest version of WordPress, PHP, Apache, and MySQL. It is secure by default, as all ports except HTTP and HTTPS ports are closed. HTTP/2 and Let's Encrypt auto-configuration are supported.

Additional details

Runs on: Google Compute Engine

Type: [Virtual machines](#), Single VM

Last updated: 7/4/22

Category: [Blog & CMS](#)

Version: 6.0.0-6-r03

Operating System: Debian 11

Package contents: Akismet 4.2.4, Simple Image 3.0.2, AMP 2.3.0, WordPress Mail SMTP 3.4.0, All-in-One WP Migration 7.61.0, All in One SEO Pack 4.2.2-0, W3 Total Cache 2.2.3-0, Google Analytics Dashboard 8.6.0, Jetpack 9.9.1-0, WordPress Amazon Polly Plugin 4.3.2, WordPress 6.0.0, mod_pagespeed library 1.13.35-2, mod_pagespeed_ap24 library 1.13.35-2, ModSecurity Apache Connector 0.20210819.0, Apache utilities (APR) 1.6.1, Apache Portable Runtime (APR) 1.7.0, ModSecurity 3.0.7, ModSecurity2 2.9.5, Apache 2.4.54, Apache PageSpeed Module 1.13.35-2, MariaDB 10.6.8, Composer 2.3.7, PECL APC User Cache 5.1.21, MaxMind DB Reader PHP API 1.11.0, libmemcached 3.2.0, PECL PHP driver for Xdebug 3.1.5, libmaxminddb 1.6.0, PECL PHP driver for imagick 3.7.0, PECL PHP driver for MongoDB 1.13.0, IMAP 2007.0.0, PHP 8.0.20, qpress 11.0.0-0, Percona XtraBackup 8.0.28-21, vmtoolsd-querystring 2.0.3, Varnish 6.6.2, phpMyAdmin 5.2.0, Bndiagnostic Tool 0.9.17, wait-for-port 1.0.3, Conit 0.2.6, MySQL 8.0.29, gosu 1.14.0, Brotli 1.0.9, WP-CLI 2.6.0, Bncert Tool 0.7.4, render-template 1.0.3, ini-file 1.4.3



Add to Service Catalog: [Deployment .zip file](#) 

FIGURE 16.7 Pricing estimates for the WordPress solution

Pricing

The table below shows the estimated costs using the default configuration. You can customize the configuration later when deploying this solution.

Bitnami WordPress Usage Fee Bitnami does not charge a usage fee.	USD 0.00/mo
Infrastructure fee	
VM instance: 1 shared vCPU + 1.7 GB memory (g1-small)	USD 18.76/mo
Standard Persistent Disk: 10GB	USD 0.47/mo
Sustained use discount 	- USD 5.63/mo
Estimated monthly total	USD 13.60/mo

We're currently using USD to calculate costs, which can be changed in the billing setup. Final prices in your bill will be set in accordance with your billing setup, and might be subject to exchange rates.

Price estimates based on 30-day, 24hrs per day usage of the listed resources in the Central US region. The Estimated Monthly Infrastructure Fee calculation may not reflect all Google Cloud Platform IaaS resources actually created or consumed by this product (or the fees charged for such consumption). Bitnami may be able to provide a more accurate estimate of monthly GCP IaaS consumption.

Google Cloud Platform Free Trial

New Google Cloud customers may be eligible for free trial.

[Learn more about Google Cloud pricing & free trial](#)

FIGURE 16.8 Tutorial and support information

Tutorials and documentation

[Access using SSH](#)
Configure SSH keys to access the application as the user "bitnami".

[Using SFTP](#)
Use this guide to upload files using SFTP.

[MariaDB access credentials](#)
Use username "root" and the temporary password to access MariaDB.

[Change your MariaDB root password](#)
Change your temporary mariadb root password by following these instructions

[Accessing phpMyAdmin](#)
Access phpMyAdmin via an SSH tunnel using this guide.

[Adding plugins with privileges](#)
Some plugins need privileged access to install. Edit privileges with this guide.

[Installation directory structure](#)
Learn how application files, libraries and configuration files are organized.

Support

Bitnami provides technical support for installation and setup issues through [our support center](#).

[Learn more](#)

Terms of Service


By using this product you agree to the [GCP Marketplace Terms of Service](#) and the terms and conditions of the following software license(s): [End User License Agreement](#).

In addition to the parameters described earlier, the launch page will also display links to related documentation, as shown in Figure 16.12.

Click the Deploy button to launch the deployment. That will open Deployment Manager and show the progress of the deployment (see Figure 16.13).

When the launch process completes, you will see a summary about the deployment and a button to launch the admin panel, as shown in Figure 16.14.

FIGURE 16.9 Launch a Cloud Marketplace solution from the overview page of the product.



WordPress Multisite Certified by Bitnami and Automattic

[Bitnami](#)

Up-to-date, secure, and ready to run.

LAUNCHVIEW PAST DEPLOYMENTS

OVERVIEWPRICINGSUPPORT

Overview

Bitnami, the leaders in application packaging, and Automattic, the experts behind WordPress, have teamed up to offer this official WordPress image on Google Cloud Marketplace.

WordPress Multisite enables you to make one WordPress deployment to manage multiple, independent websites. These websites can all have unique domain names and layouts while sharing assets like themes and plugins. It is ideal for universities, corporations, and agencies that need to enable many people to manage their own websites.

For content authors, the Jetpack plugin (enabled by default) offers access to additional professional themes, performance improvements, scanning, site activity, and marketing tools. Other popular plugins like Akismet, All in One SEO Pack, WP Mail and Google Analytics for WordPress also come pre-installed. Optional automatic backup and priority support are available from Automattic.

For developers, this image features the AMP for WordPress plugin. This plugin automatically adds Accelerated Mobile Pages (Google AMP Project) support to deliver a faster, higher-performance and more flexible web experience across distribution platforms. It helps you reduce the costs of your site by pairing your content to the format required by the destination platform and making the user experience consistent across devices.

This image includes the latest version of WordPress, PHP, Apache, and MySQL. It is secure by default, as all ports except HTTP and HTTPS ports are closed. HTTP/2 and Let's Encrypt auto-configuration are supported.

Why use Bitnami Certified Apps?

Additional details

Runs on: Google Compute Engine
Type: [Virtual machines](#), Single VM
Last updated: 7/5/22
Category: [Blog & CMS](#)
Version: 6.0.0-11-r65
Operating System: Debian 11

Package contents: Akismet 4.2.4 , All-in-One WP Migration 7.62.0 , Simple Tags 3.6.2 , AMP 2.3.0 , WordPress Mail SMTP 3.4.0 , All in One SEO Pack 4.2.2.0 , W3 Total Cache 2.2.3-0 , Google Analytics Dashboard 8.6.0 , Jetpack 9.9.1-0 , WordPress Amazon Polly Plugin 4.3.2 , WordPress 6.0.0 , mod_pagespeed library 1.13.35-2 , mod_pagespeed_ap24 library 1.13.35-2 , ModSecurity Apache Connector 0.20210819.0 , Apache utilities (APR) 1.6.1 , Apache Portable Runtime (APR) 1.7.0 , ModSecurity 3.0.7 , ModSecurity2 2.9.5 , Apache 2.4.54 , Apache PageSpeed Module 1.13.35-2 , MariaDB 10.6.8 , Composer 2.3.7 , PECL APC User Cache 5.1.21 , MaxMind DB Reader PHP API 1.11.0 , libmemcached 3.2.0 , PECL PHP driver for Xdebug 3.1.5 , libmaxminddb 1.6.0 , PECL PHP driver for Imagick 3.7.0 , PECL PHP driver for MongoDB 1.13.0 , IMAP 2007.0.0 , PHP 8.0.10 , qpress 11.0.0-0 , Percona XtraBackup 8.0.28-21 , vmtoolsd-querystring 2.0.3 , Varnish 6.6.2 , phpMyAdmin 5.2.0 , Bndiagnostic Tool 0.9.17 , wait-for-port 1.0.3 , Gonic 0.2.6 , MySQL 8.0.29 , gosu 1.14.0 , Brotli 1.0.9 , WP-CLI 2.6.0 , Bncert Tool 0.8.0 , render-template 1.0.3 , ini-file 1.4.3


Add to Service Catalog: [Deployment .zip file](#) 

FIGURE 16.10 The launch page for a WordPress solution in Cloud Marketplace

New WordPress Certified by Bitnami and Automattic deployment

Deployment name *
wordpress-1

Zone
us-south1-c


Machine type
Machine family
GENERAL-PURPOSE

Machine types for common workloads, optimized for cost and flexibility


Series
N2

Powered by Intel Cascade Lake and Ice Lake CPU platforms


Machine type
n2-standard-2 (2 vCPU, 8 GB memory)



vCPU
2




Memory
8 GB



WordPress Certified by Bitnami and Automattic overview

Product provided by Bitnami

Bitnami WordPress Usage Fee	USD 0.00/mo
Bitnami does not charge a usage fee.	
Infrastructure fee	
VM instance: 2 vCPUs + 8 GB memory (n2-standard-2)	USD 83.66/mo
Standard Persistent Disk: 10GB	USD 0.47/mo
Sustained use discount 	- USD 25.10/mo
Estimated monthly total	USD 59.03/mo

Price estimates based on 30-day, 24hrs per day usage of the listed resources in the selected region. The Estimated Monthly Infrastructure Fee calculation may not reflect all Google Cloud Platform IaaS resources actually created or consumed by this product (or the fees charged for such consumption). Bitnami may be able to provide a more accurate estimate of monthly GCP IaaS consumption.

FIGURE 16.11 Additional network parameters

Networking

Network interfaces

Network interface

Network

default

Subnetwork

default

External IP

Ephemeral

DONE

FIGURE 16.12 Links to related documentation are available on the deployment page.

Documentation

[Access using SSH](#)

Configure SSH keys to access the application as the user "bitnami".

[Using SFTP](#)

Use this guide to upload files using SFTP.

[MariaDB access credentials](#)

Use username "root" and the temporary password to access MariaDB.

[Change your MariaDB root password](#)

Change your temporary mariadb root password by following these instructions

[Accessing phpMyAdmin](#)

Access phpMyAdmin via an SSH tunnel using this guide.

[Adding plugins with privileges](#)

Some plugins need privileged access to install. Edit privileges with this guide.

[Installation directory structure](#)

Learn how application files, libraries and configuration files are organized.

Terms of Service

By deploying the software or accessing the service you are agreeing to comply with the [Bitnami terms of service](#), [GCP Marketplace terms of service](#) and the terms of applicable open source software licenses bundled with the software or service. Please review these terms and licenses carefully for details about any obligations you may have related to the software or service. To the limited extent an open source software license related to the software or service expressly supersedes the GCP Marketplace Terms of Service, that open source software license governs your use of that software or service.

By using this product, you understand that certain account and usage information may be shared with Bitnami for the purposes of financial accounting, sales attribution, performance analysis, and support.

Google is providing this software or service "as-is" and any support for this software or service will be provided by Bitnami under their terms of service.

FIGURE 16.13 Cloud Deployment Manager launching WordPress

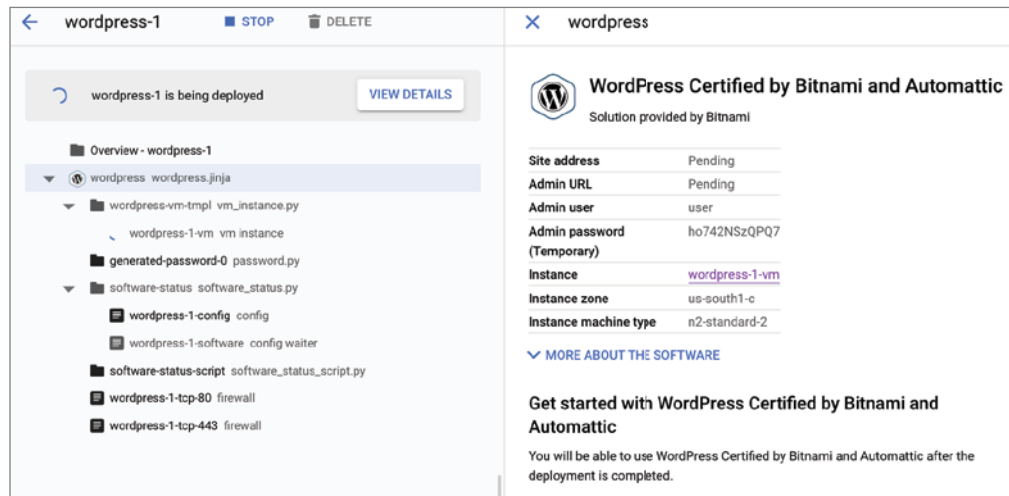
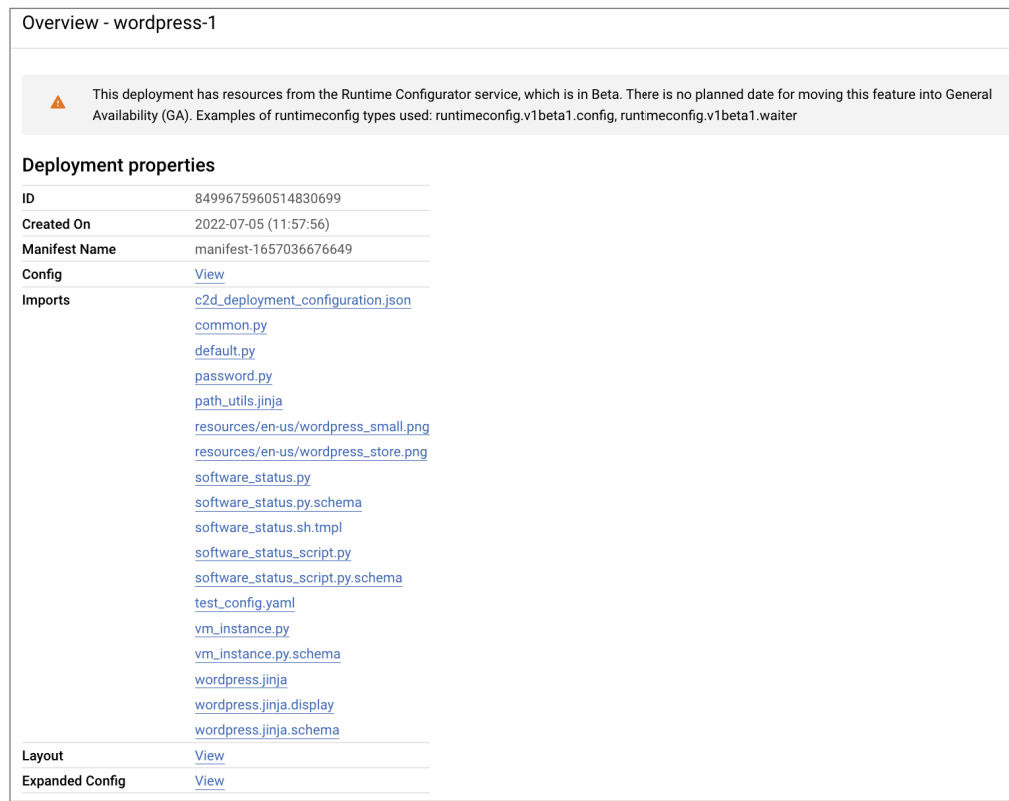


FIGURE 16.14 Information about the deployed WordPress instance



Building Infrastructure Using the Cloud Foundation Toolkit

In addition to launching the solutions listed in Cloud Marketplace, you can create your own solution configuration files so that users can launch preconfigured solutions using Deployment Manager configuration files as well as Terraform-based specifications using the Cloud Foundation Toolkit. Terraform is an open source tool for specifying infrastructure as code. A third option, Config Connector, is available to those who want to manage Google Cloud resources using Kubernetes.

Deployment Manager Configuration Files

Deployment Manager configuration files are written in YAML syntax. The configuration files start with the word `resources`, followed by resource entities, which are defined using three fields:

- `name`, which is the name of the resource.
- `type`, which is the type of the resource, such as `compute.v1.instance`.
- `properties`, which are key-value pairs that specify configuration parameters for the resource. For example, a VM has properties for specifying machine type, disks, and network interfaces.



For information on YAML syntax, see the official documentation at yaml.org.

A simple example defining a virtual machine called `ace-exam-deployment-vm` starts with the following:

```
resources:
```

```
- type: compute.v1.instance
  name: ace-exam-deployment-vm
```

Next, you can add properties, such as the machine type, disk configuration, and network interfaces.

The properties section of the configuration file starts with the word `properties`. For each property, there is a single key-value pair or a list of key-value pairs. The machine type property has a single key-value pair, with the key being `machineType`. Disks have multiple properties, so following the word `disks`, there is a list of key-value pairs. Continuing the example of `ace-exam-deployment-vm`, the structure is as follows:

```
resources:
```

```
- type: compute.v1.instance
  name: ace-exam-deployment-vm
```

properties:

```
machineType: [MACHINE_TYPE_URL]
```

In this example, machineType would be a URL to a Google API resource specification, such as the following:

```
www.googleapis.com/compute/v1/projects/[PROJECT_ID]/zones/us-
central1-f/machineTypes/f1-micro
```

Note that there is a reference to *[PROJECT_ID]*, which you'd replace with an actual project ID in a configuration file. Disks have properties such as a deviceName and type, and Booleans indicating whether the disk is a boot disk or should be autodeleted. Let's continue the previous example by adding the machine type specification and some disk properties:

resources:

```
- type: compute.v1.instance
```

```
  name: ace-exam-deployment-vm
```

properties:

```
  machineType: www.googleapis.com/compute/v1/projects/[PROJECT_ID]/
  zones/us-central1-f/machineTypes/f1-micro
```

disks:

```
- deviceName: boot
```

```
  type: PERSISTENT
```

```
  boot: true
```

```
  autoDelete: true
```

Listing 16.1 shows the full configuration file from the Google Deployment Manager documentation. The following code is available at <https://cloud.google.com/deployment-manager/docs/quickstart> (source: https://github.com/GoogleCloudPlatform/deploymentmanager-samples/blob/master/examples/v2/quick_start/vm.yaml).

Listing 16.1: examples/v2/quick_start/vm.yaml

```
# Copyright 2016 Google Inc. All rights reserved.#
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
#     www.apache.org/licenses/LICENSE-2.0
#
```

```
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.

# Put all your resources under 'resources:'. For each resource, you need:
# - The type of resource. In this example, the type is a Compute VM instance.
# - An internal name for the resource.
# - The properties for the resource. In this example, for VM instances, you add
#   the machine type, a boot disk, network information, and so on.
#
# For a list of supported resources,
# see https://cloud.google.com/deployment-manager/docs/configuration/supported-resource-types

resources:
- type: compute.v1.instance
  name: quickstart-deployment-vm
  properties:
    # The properties of the resource depend on the type of resource. For a
    # list of properties, see the API reference for the resource.
    zone: us-central1-f
    # Replace [MY_PROJECT] with your project ID
    machineType: www.googleapis.com/compute/v1/projects/[MY_PROJECT]/
zones/us-central1-f/machineTypes/f1-micro
    disks:
      - deviceName: boot
        type: PERSISTENT
        boot: true
        autoDelete: true
        initializeParams:
          # Replace [FAMILY_NAME] with the image family name.
          # See a full list of image families at
          # https://cloud.google.com/compute/docs/images#os-compute-support
          sourceImage: www.googleapis.com/compute/v1/projects/debian-cloud/
global/images/family/[FAMILY_NAME]
          # Replace [MY_PROJECT] with your project ID
```

```

networkInterfaces:
- network: www.googleapis.com/compute/v1/projects/[MY_PROJECT]/
  global/networks/default
  # Access Config required to give the instance a public IP address
accessConfigs:
- name: External NAT
  type: ONE_TO_ONE_NAT

```

This configuration specifies a deployment named `quickstart-deployment-vm`, which will run in the `us-central1-f` zone. The deployment will use a `f1-micro` virtual machine running a Debian distribution of Linux. An external IP address will be assigned.

Before executing this template, you would need to replace `[MY_PROJECT]` with your project ID and `[FAMILY_NAME]` with the name of a Debian image family, such as `debian-9`. You can find a list of images in the Compute Engine section of Cloud Console on the Images tab. You can also list images using the `gcloud compute images list` command.

Deployment Manager Template Files

If your deployment configurations are becoming complicated, you can use deployment templates. Templates are another text file you use to define resources and import those resources into configuration files. This allows you to reuse resource definitions in multiple places. Templates can be written in Python or Jinja2, a templating language.



For information on Jinja2 syntax, see the official documentation at <http://jinja.pocoo.org/docs/2.10>.

As an Associate Cloud Engineer, you should know that Google recommends using Python to create template files unless the templates are relatively simple, in which case it is appropriate to use Jinja2.

Launching a Deployment Manager Template

You can launch a deployment template using the `gcloud deployment-manager deployments create` command. For example, to deploy the template from the Google documentation, use the following:

```
gcloud deployment-manager deployments create quickstart-deployment
--config=vm.yaml
```

You can also describe the state of a deployment with the `describe` command, as follows:

```
gcloud deployment-manager deployments describe quickstart-deployment
```

Providing a Deployable Service

In large enterprises, different groups often want to use the same service, such as a data science application, to understand customer purchasing patterns. Product managers across the organization may want to use this. Software developers could create a single instance of the application's resources and have multiple users work with that one instance. This is a co-hosted structure, which has some advantages if you have a single DevOps team supporting all users.

Alternatively, you could allow each user or small group of users to have their own application instance. This approach has several advantages. Users could run the application in their own projects, simplifying allocating charges for resources, since the project would be linked to the users' billing accounts. Also, users could scale the resources up or down as needed for their use case.

A potential disadvantage is that users may not be comfortable configuring Google Cloud resources. Deployment Manager addresses that problem by making it relatively simple to deploy an application and resources in a repeatable process. Someone who can run a `gcloud deployment-manager` command could deploy application resources similar to the way users deploy applications from Cloud Marketplace.

Cloud Foundation Toolkit

The Cloud Foundation Toolkit is an open source project that provides infrastructure as code templates using Deployment Manager and Terraform templates.

The Cloud Foundation Toolkit includes blueprints, which are packages of deployable configuration specification as well as policies for implementing a solution to a particular class of problems. These blueprints encapsulate best practices for configuring infrastructure and granting access to resources. Blueprints are available for Terraform and Kubernetes. The Kubernetes blueprints are used with the Config Connector. For examples of blueprints, see <https://cloud.google.com/docs/terraform/blueprints/terraform-blueprints>.

In addition to blueprints that are designed for solving broad needs, such as deploying a data warehouse, templates are also available for configuring specific Google Cloud service resources. For example, Listing 16.2 shows a template for creating a virtual machine.

Listing 16.2: https://github.com/terraform-google-modules/terraform-google-vm/blob/master/modules/compute_instance/main.tf

```
/**
 * Copyright 2018 Google LLC
 *
```

```

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*     www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

locals {
  hostname      = var.hostname == "" ? "default" : var.hostname
  num_instances = length(var.static_ips) == 0 ? var.num_instances :
length(var.static_ips)

  # local.static_ips is the same as var.static_ips with a dummy element
  # appended
  # at the end of the list to work around "list does not have any elements
  # so cannot
  # determine type" error when var.static_ips is empty
  static_ips = concat(var.static_ips, ["NOT_AN_IP"])
  project_id = length(regexall("/projects/([^/]*)", var.instance_template)) >
0 ? flatten(regexall("/projects/([^/]*)", var.instance_template))[0] : null

  # When no network or subnetwork has been defined, we want to use the
  # settings from
  # the template instead.
  network_interface = length(format("%s%s", var.network, var.subnetwork)) == 0
? [] : [1]
}

#####
# Data Sources
#####
data "google_compute_zones" "available" {
  project = local.project_id
  region  = var.region
}

#####
# Instances

```

```
#####
```

```
resource "google_compute_instance_from_template" "compute_instance" {
  provider      = google
  count         = local.num_instances
  name          = var.add_hostname_suffix ? format("%s%s%s", local
.hostname, var.hostname_suffix_separator, format("%03d", count.index + 1)) :
local.hostname
  project       = local.project_id
  zone          = var.zone == null ? data.google_compute_zones
.available.names[count.index % length(data.google_compute_zones.available
.names)] : var.zone
  deletion_protection = var.deletion_protection

  dynamic "network_interface" {
    for_each = local.network_interface

    content {
      network          = var.network
      subnetwork       = var.subnetwork
      subnetwork_project = var.subnetwork_project
      network_ip       = length(var.static_ips) == 0 ? "" : element(local
.static_ips, count.index)
      dynamic "access_config" {
        for_each = var.access_config
        content {
          nat_ip          = access_config.value.nat_ip
          network_tier    = access_config.value.network_tier
        }
      }
    }

    dynamic "alias_ip_range" {
      for_each = var.alias_ip_ranges
      content {
        ip_cidr_range      = alias_ip_range.value.ip_cidr_range
        subnetwork_range_name = alias_ip_range.value.subnetwork_range_name
      }
    }
  }

  source_instance_template = var.instance_template
}
```

Config Connector

Config Connector is a Kubernetes add-on that allows you to manage Google Cloud resources through Kubernetes. This is useful for those who have already managed Kubernetes resources using Kubernetes configurations and want to extend the scope of those tools to include Google Cloud resources. Config Connector provides a collection of Kubernetes custom resource definitions (CRDs) and controllers for managing Google Cloud resources.

To install the Config Connector, you pass a parameter to the `gcloud container clusters create` command specifying the ConfigConnector add-on. For example:

```
gcloud container cluster create ace-gke-cluster1 \
  --addons ConfigConnector
  --workload-pool=ace-project-dw1
  --logging=SYSTEM
  --monitoring=SYSTEM
```

To use Config Connector, you will have to enable Workload Identity, a way to link IAM identities to Kubernetes identities. You will also need to enable Kubernetes Engine monitoring and use a supported version of Kubernetes. Configurations of Config Connector are applied using `kubectl`.

For more on Config Connector solutions, see <https://github.com/GoogleCloudPlatform/cloud-foundation-toolkit/tree/master/config-connector/solutions>.

Summary

Cloud Marketplace and Cloud Deployment Manager are designed to make it easy to deploy resources in Google Cloud. Cloud Marketplace is where third-party vendors can offer deployable applications based on proprietary or open source software. When an application is deployed from Cloud Marketplace, resources such as VMs, storage buckets, and persistent disks are created automatically without additional human intervention. Deployment Manager gives cloud engineers the ability to define configuration files that describe the resources they would like to deploy. Cloud engineers can then use `gcloud` commands to deploy the resources and list their status. Deployment Manager is especially useful in organizations where you want to easily deploy resources without requiring users of those resources to understand the details of how to configure Google Cloud resources.

The Cloud Foundation Toolkit provides templates and blueprints that encode best practices for deploying solutions and individual resources to Google Cloud. The Config Connector add-on to Kubernetes allows you to manage Google Cloud resources using Kubernetes.

Exam Essentials

Understand how to browse for solutions using the Cloud Marketplace section of Cloud Console. You can use filters to narrow your search to specific kinds of solutions, such as

operating systems and developer tools. There may be multiple options for a single application, such as WordPress. This is because multiple vendors provide configurations. Review the description of each to understand which best fits your needs.

Know how to deploy a solution in Cloud Marketplace. Understand how to configure a Cloud Marketplace deployment in Cloud Console. Understand that when you launch a solution, you may be prompted for application-specific configurations. For example, with WordPress you may be prompted to install phpMyAdmin. You may also have the opportunity to configure common configuration attributes, such as the machine type and boot disk type.

Understand how to use the Deployment Manager section of the console to monitor deployment. It may be a few minutes from the time you launch a configuration to the time it is ready to use. Note that once the application is ready, you may be prompted for additional information, such as a username and password to log in.

Know that Deployment Manager is a Google Cloud service for creating configuration files that define resources to use with an application. These configuration files use YAML syntax. They are made up of resource specifications that use key-value pairs to define properties of the resource.

Know that resources in a configuration file are defined using a name, type, and set of properties. The properties vary by type. The machine type can be defined using just a URL that points to a type of machine available in a region. Disks have multiple properties, including a device name, a type, and whether the disk is a boot disk.

Know that you can use templates with configuration files. If your configuration files are getting long or complicated, you can modularize them using templates. Templates define resources and can be imported into other templates. Templates are text files written in Jinja2 or Python.

Know how to launch a deployment configuration file using
`gcloud deployment-manager deployment create.`

You can review the status of a deployment using
`gcloud deployment-manager deployments-describe.`

Know the purpose of Cloud Foundation Toolkit and Config Connector. Cloud Foundation Toolkit is an open source project with blueprints and example configurations that capture Google Cloud-recommended best practices for deploying solutions. Config Connector is a Kubernetes add-on for managing Google Cloud resources from Kubernetes.

Review Questions

You can find the answers in the Appendix.

1. What are the categories of Cloud Marketplace solutions?
 - A. Data sets only
 - B. Operating systems only
 - C. Developer tools and operating systems only
 - D. Data sets, operating systems, and developer tools
2. You want to use Terraform for managing infrastructure as code and you would also like to follow Google Cloud–recommended best practices. What would you use to start implementing such a solution?
 - A. Cloud Deployment Manager
 - B. Cloud Foundation Toolkit
 - C. Config Connector
 - D. Cloud Build
3. Where do you navigate to launch a Cloud Marketplace solution?
 - A. Overview page of the solution
 - B. Main Cloud Marketplace page
 - C. Network Services
 - D. None of the above
4. You want to quickly identify the set of operating systems available in Cloud Marketplace. Which of these steps would help with that?
 - A. Use Google Search to search the web for a listing.
 - B. Use filters in Cloud Marketplace.
 - C. Scroll through the list of solutions displayed on the start page of Cloud Marketplace.
 - D. It is not possible to filter to operating systems.
5. You want to use Cloud Marketplace to deploy a WordPress site. You notice there is more than one WordPress option. Why is that?
 - A. It's a mistake. Submit a ticket to Google support.
 - B. Multiple vendors may offer the same application.
 - C. It's a mistake. Submit a ticket to the vendors.
 - D. You will never see such an option.

6. You have used Cloud Marketplace to deploy a WordPress site and would now like to deploy a database. You notice that the configuration page for the databases is different from the one used with WordPress. Why is that?
 - A. It's a mistake. Submit a ticket to Google support.
 - B. You've navigated to a different subform of Cloud Marketplace.
 - C. Configuration properties are based on the application you are deploying and will be different depending on what application you are deploying.
 - D. This cannot happen.
7. You have been asked by your manager to deploy a WordPress site. You expect heavy traffic, and your manager wants to make sure the VM hosting the WordPress site has enough resources. Which resources can you configure when launching a WordPress site using Cloud Marketplace?
 - A. Machine type
 - B. Disk type
 - C. Disk size
 - D. All of the above
8. You would like to define as code the configuration of a set of application resources. What is the Google Cloud service for creating resources using a configuration file made up of resource specifications defined in YAML syntax?
 - A. Compute Engine
 - B. Deployment Manager
 - C. Marketplace Manager
 - D. Marketplace Deployer
9. What file format is used to define Deployment Manager configuration files?
 - A. XML
 - B. JSON
 - C. CSV
 - D. YAML
10. A Deployment Manager configuration file starts with what word?
 - A. deploy
 - B. resources
 - C. properties
 - D. YAML
11. Which of the following are used to define a resource in a Cloud Deployment Manager configuration file?
 - A. Type only
 - B. Properties only
 - C. Name and type only
 - D. Type, properties, and name

12. What properties may be set when defining a disk on a VM?
 - A. A device name only
 - B. A Boolean indicating a boot disk and a Boolean indicating autodelete
 - C. A Boolean indicating autodelete only
 - D. A device name, a Boolean indicating a boot disk, and a Boolean indicating autodelete
13. You need to look up what images are available in the zone in which you want to deploy a VM. What command would you use?
 - A. `gcloud compute images list`
 - B. `gcloud images list`
 - C. `gsutil compute images list`
 - D. `gcloud compute list images`
14. You want to use a template file with Deployment Manager. You expect the file to be complicated. What language would you use?
 - A. Jinja2
 - B. Ruby
 - C. Go
 - D. Python
15. What command launches a deployment?
 - A. `gcloud deployment-manager deployments create`
 - B. `gcloud cloud-launcher deployments create`
 - C. `gcloud deployment-manager deployments launch`
 - D. `gcloud cloud-launcher deployments launch`
16. A DevOps engineer is noticing a spike in CPU utilization on your servers. You explain that you have just launched a deployment. You'd like to show the DevOps engineer the details of a deployment you just launched. What command would you use?
 - A. `gcloud cloud-launcher deployments describe`
 - B. `gcloud deployment-manage deployments list`
 - C. `gcloud deployment-manager deployments describe`
 - D. `gcloud cloud-launcher deployments list`
17. If you expand the More link in the Networking section when deploying a Cloud Marketplace solution, what will you be able to configure?
 - A. IP addresses
 - B. Billing
 - C. Access controls
 - D. Custom machine type

- 18.** What are the license types referenced in Cloud Marketplace?
- A.** Free only
 - B.** Free and flat hourly only
 - C.** Free and bring your own license (BYOL) only
 - D.** Free, flat hourly, usage fees, and bring your own license (BYOL)
- 19.** Which license type will add charges to your Google Cloud bill when using Cloud Marketplace with this type of license?
- A.** Free
 - B.** Flat hourly and usage fees
 - C.** BYOL
 - D.** Chargeback
- 20.** You are deploying a Cloud Marketplace application that includes an LAMP stack. What software will this deploy?
- A.** Apache server and Linux only
 - B.** Linux only
 - C.** MySQL and Apache only
 - D.** Apache, MySQL, Linux, and PHP

Chapter 17

Configuring Access and Security

**THIS CHAPTER COVERS THE FOLLOWING
OBJECTIVES OF THE GOOGLE ASSOCIATE
CLOUD ENGINEER CERTIFICATION EXAM:**

- ✓ 5.1 Managing Identity and Access Management (IAM)
- ✓ 5.2 Managing service accounts
- ✓ 5.3 Viewing audit logs





Google Cloud engineers can expect to spend a significant amount of time working with access controls. This chapter provides instruction on how to perform several common tasks, including managing identity and access management (IAM) assignments, creating custom roles, managing service accounts, and viewing audit logs.

It is important to know that the preferred way of assigning permissions to users, groups, and service accounts is through the IAM system. However, Google Cloud did not always have IAM. Before that, permissions were granted using what are now known as basic roles, which are fairly coarse-grained. Basic roles, may have more permissions than you want an identity to have. You can constrain permissions using scopes. In this chapter, we will describe how to use basic roles and scopes as well as IAM. Going forward, it is a best practice to use IAM for access control.

Managing Identity and Access Management

When you work with IAM, there are a few common tasks you need to perform:

- Viewing account IAM assignments
- Assigning IAM roles
- Defining custom roles

Let's look at how to perform each of these tasks.

Viewing Account IAM Assignments

You can view account IAM assignments in Cloud Console by navigating to the IAM & Admin section. In that section, select IAM from the navigation menu to display the page shown in Figure 17.1. The example in the figure shows a list of identities filtered by member name.

In this example, the user `dan@sullivanlearninggroup.com` has three roles: Compute Organization Resource Admin, Organization Administrator, and Owner. App Engine Admin and BigQuery Admin are predefined IAM roles. Owner is a basic role.

FIGURE 17.1 Permissions listing filtered by member

IAM + ADD - REMOVE			
PERMISSIONS RECOMMENDATIONS HISTORY			
Permissions for project "My First Project" These permissions affect this project and all of its resources. Learn more			
View By: PRINCIPALS ROLES			
Filter Enter property name or value			
<input type="checkbox"/> Type	Principal ↑	Name	Role
<input type="checkbox"/>	388947348090-compute@developer.gserviceaccount.com	Compute Engine default service account	Cloud Data Fusion Runner Editor
<input type="checkbox"/>	388947348090@cloudbuild.gserviceaccount.com		Cloud Build Service Account
<input type="checkbox"/>	388947348090@cloudservices.gserviceaccount.com	Google APIs Service Agent ?	Editor
<input type="checkbox"/>	dan@sullivanlearninggroup.com	Dan Sullivan	Compute Organization Resource Admin Organization Administrator Owner
<input type="checkbox"/>	scenic-energy-335022@appspot.gserviceaccount.com	App Engine default service account	Owner

Basic roles were used prior to IAM. There are three basic roles: Owner, Editor, and Viewer. Viewers have permission to perform read-only operations. Editors have viewer permissions and permission to modify an entity. Owners have editor permissions and can manage roles and permission on an entity. Owners can also set up billing for a project.

IAM roles are collections of permissions. They are tailored to provide identities with just the permissions they need to perform a task and no more. To see a list of users assigned a role (see Figure 17.2), click the Roles tab on the IAM page.

FIGURE 17.2 List of identities assigned to Cloud Build Service Account and Cloud Data Fusion Runner roles

IAM + ADD - REMOVE HELP ASSISTANT LEARN			
PERMISSIONS RECOMMENDATIONS HISTORY			
Permissions for project "My First Project" These permissions affect this project and all of its resources. Learn more			
View By: PRINCIPALS ROLES <input type="checkbox"/> Include Google-provided role grants ?			
Filter Enter property name or value ?			
<input type="checkbox"/> Role / Principal ↑	Name	Inheritance	
<input type="checkbox"/> ▼ Cloud Build Service Account (1)			
<input type="checkbox"/> 388947348090@cloudbuild.gserviceaccount.com			
<input type="checkbox"/> ▼ Cloud Data Fusion Runner (1)			
<input type="checkbox"/> 388947348090-compute@developer.gserviceaccount.com	Compute Engine default service account		

This page shows a list of roles with the number of identities assigned to that role in parentheses. Click the arrow next to the name of a role to display a list of identities with that role.

You can also see a list of users and roles assigned across a project by using the command `gcloud projects get-iam-policy`. For example, to list roles assigned to users in a project with the project ID `ace-exam-project`, use this:

```
gcloud projects get-iam-policy ace-exam-project
```

Predefined roles are grouped by service. For example, App Engine has five roles:

- App Engine Admin, which grants read, write, and modify permission to application and configuration settings. The role name used in `gcloud` commands is `roles/appengine.appAdmin`.
- App Engine Service Admin, which grants read-only access to configuration settings and write access to module-level and version-level settings. The role name used in `gcloud` commands is `roles/appengine.serviceAdmin`.
- App Engine Deployer, which grants read-only access to application configuration and settings and write access to create new versions. Users with only the App Engine Deployer role cannot modify or delete existing versions. The role name used in `gcloud` commands is `roles/appengine.deployer`.
- App Engine Viewer, which grants read-only access to application configuration and settings. The role name used in `gcloud` commands is `roles/appengine.appViewer`.
- App Engine Code Viewer, which grants read-only access to all application configurations, settings, and deployed source code. The role name used in `gcloud` commands is `roles/appengine.codeViewer`.



Although you do not have to know all of them, it helps to review predefined roles to understand patterns of how they are defined. For more details, see the Google Cloud documentation at <https://cloud.google.com/iam/docs/understanding-roles>.

Assigning IAM Roles to Accounts and Groups

To add IAM roles to accounts and groups, navigate to the IAM & Admin section of the console. Select IAM from the menu. Click the Add link at the top to display a page like that shown in Figure 17.3.

Specify the name of a user or group in the field labeled New Principals. Click Select A Role to add a role. You can add multiple roles. When you click the down arrow in the Select A Role field, you will see a list of services and their associated roles. You can choose the roles from that list. See Figure 17.4 for an example of a subset of the list, showing the roles for BigQuery.

FIGURE 17.3 The Add option in IAM opens this page, where you can assign one or more roles to users or groups.

The screenshot shows a web interface titled "Add principals to 'My First Project'". Below the title is a subtitle "Add principals, roles to 'My First Project' project" and a paragraph of instructions: "Enter one or more principals below. Then select a role for these principals to grant them access to your resources. Multiple roles allowed. [Learn more](#)". There are two input fields: "New principals *" and "Select a role *". To the right of the "Select a role *" field is a "Condition" section with a link "Add condition" and a trash icon. Below these fields is a button "+ ADD ANOTHER ROLE". At the bottom are "SAVE" and "CANCEL" buttons.

FIGURE 17.4 The drop-down list in the Select A Role field shows available roles grouped by service.

This screenshot shows the same interface as Figure 17.3, but with the "Select a role *" dropdown menu open. The menu is divided into two columns: "Quick access" and "Roles". The "Quick access" column lists "Currently used", "Basic", "By product or service", "Access Approval", "Access Context Manager", and "Actions". The "Roles" column lists "Vertex AI Service Agent", "Artifact Registry Service Agent", "BigQuery Data Transfer Service Agent", "Cloud Build Service Account", "Cloud Build Service Agent", and "Cloud Functions Service". At the bottom of the menu is a "MANAGE ROLES" link. The background shows the same form as in Figure 17.3.

If you want to know which of the fine-grained permissions are granted when you assign a role, you can list those permissions at the command line or in the console. You can also see what permissions are assigned to a role by using the command `gcloud iam roles describe`. For example, Figure 17.5 shows the list of permissions in the Spanner Database Admin role.

FIGURE 17.5 A partial listing of permissions using the `gcloud iam roles describe` command

```
iansullivanblk@cloudshell:~ (gdg-project-294122)$ gcloud iam roles describe roles/spanner.databaseAdmin
description: Full control of Cloud Spanner databases.
tag: AA==
includedPermissions:
+ monitoring.timeSeries.list
+ resourceManager.projects.get
+ resourceManager.projects.list
+ spanner.databaseOperations.cancel
+ spanner.databaseOperations.delete
+ spanner.databaseOperations.get
+ spanner.databaseOperations.list
+ spanner.databases.beginOrRollbackReadWriteTransaction
+ spanner.databases.beginPartitionedDmlTransaction
+ spanner.databases.beginReadOnlyTransaction
+ spanner.databases.create
+ spanner.databases.drop
+ spanner.databases.get
+ spanner.databases.getDdl
+ spanner.databases.getIamPolicy
+ spanner.databases.list
+ spanner.databases.partitionQuery
+ spanner.databases.partitionRead
+ spanner.databases.read
+ spanner.databases.select
```

You can also use Cloud Console to view permissions. Navigate to the IAM & Admin section and select Roles from the menu. This displays a list of roles. Click the check box next to a role name to display a list of permissions on the right, as shown in Figure 17.6 for Cloud SQL Admin.

You can assign roles to a member in a project using the following command:

```
gcloud projects add-iam-policy-binding [RESOURCE-NAME] \
--member= user:[USER-EMAIL] --role= [ROLE-ID]
```

For example, to grant the Editor basic role to a user identified by `jane@acexam.com`, you could use this:


```
gcloud projects add-iam-policy-binding ace-exam-project \
--member=user:jane@acexam.com --role='roles/editor'
```

FIGURE 17.6 Using Cloud Console to view a partial listing of permissions available for Cloud SQL Admin

←

Cloud SQL Admin

+ EDIT ROLE

 CREATE FROM ROLE

ID

roles/cloudsql.admin

Role launch stage

General Availability

Description

Full control of Cloud SQL resources.

71 assigned permissions

cloudsql.backupRuns.create

cloudsql.backupRuns.delete

cloudsql.backupRuns.get

cloudsql.backupRuns.list

cloudsql.databases.create

cloudsql.databases.delete

cloudsql.databases.get

cloudsql.databases.list

cloudsql.databases.update

cloudsql.instances.addServerCa

cloudsql.instances.clone

cloudsql.instances.connect

cloudsql.instances.create

cloudsql.instances.createTagBinding

cloudsql.instances.delete

cloudsql.instances.deleteTagBinding

cloudsql.instances.demoteMaster

cloudsql.instances.export

cloudsql.instances.failover

cloudsql.instances.get

cloudsql.instances.import

cloudsql.instances.list

cloudsql.instances.listEffectiveTags

cloudsql.instances.listServerCas

cloudsql.instances.listTagBindings

cloudsql.instances.login

cloudsql.instances.promoteReplica

cloudsql.instances.resetSslConfig

cloudsql.instances.restart

cloudsql.instances.restoreBackup

cloudsql.instances.rotateServerCa

cloudsql.instances.startReplica

cloudsql.instances.stopReplica

cloudsql.instances.truncateLog

cloudsol.instances.update



Real World Scenario

IAM Roles Support Least Privilege and Separation of Duties

Two security best practices are assigning least privileges and maintaining a separation of duties. The principle of least privileges says you grant only the smallest set of permissions that is required for a user or service account to perform their required tasks. For example, if users can do everything they need to do with only read permission to a database, then they should not have write permission.

In the case of separation of duties, the idea is that a single user should not be able to perform multiple sensitive operations that together could present a risk. In high-risk domains, such as finance or defense, you would not want a developer to be able to modify an application and deploy that change to production without review. A malicious engineer, for example, could modify code in a finance application to suppress application logging when funds are transferred to a bank account controlled by the malicious engineer. If that engineer were to put that code in production, it could be some time before auditors discover that logging has been suppressed and there may have been fraudulent transactions.

IAM roles support least privilege by assigning minimal permissions to predefined roles. It also supports separation of duties by allowing some users to have the ability to change code and others to deploy code.

Another common security practice is defense in depth, which applies multiple, overlapping security controls. That is also a practice that should be adopted. IAM can be applied as one of the layers of defense.

Defining Custom IAM Roles

If the set of predefined IAM roles does not meet your needs, you can define a custom role.

To define a custom role in Cloud Console, navigate to the Roles option in the IAM & Admin section of the console. Click the Create Role link at the top of the page. This will display a page like that shown in Figure 17.7.

On this page you can specify a name for the custom role, a description, an identifier, a launch stage, and a set of permissions. The launch stage options are as follows: Alpha, Beta, General Availability, and Disabled.

You can click Add Permissions to display a list of permissions. The list in Figure 17.8 is filtered to include only permissions in the Cloud SQL Admin role.

FIGURE 17.7 Creating a role in Cloud Console

←

Create Role

Custom roles let you group permissions and assign them to principals in your project or organization. You can manually select permissions or import permissions from another role. [Learn more](#)

Title *

Custom Role

11 / 100

Description

Created on: 2022-07-23

22 / 256

ID *

CustomRole

Role launch stage

Alpha

+ ADD PERMISSIONS

No assigned permissions

Filter

Enter property name or value

?

|||

<input type="checkbox"/>	Permission ↑	Status
No rows to display		

i

Some permissions might be associated with and checked by third parties. These permissions contain the third party's service and domain name in the permission prefix.

CREATE

CANCEL

FIGURE 17.8 List of available permissions filtered by role

Add permissions

Filter permissions by role Cloud SQL Admin

Filter Enter property name or value ? |||

<input type="checkbox"/>	Permission ↑	Status
<input type="checkbox"/>	cloudsql.backupRuns.create	Supported
<input type="checkbox"/>	cloudsql.backupRuns.delete	Supported
<input type="checkbox"/>	cloudsql.backupRuns.get	Supported
<input type="checkbox"/>	cloudsql.backupRuns.list	Supported
<input type="checkbox"/>	cloudsql.databases.create	Supported
<input type="checkbox"/>	cloudsql.databases.delete	Supported
<input type="checkbox"/>	cloudsql.databases.get	Supported
<input type="checkbox"/>	cloudsql.databases.list	Supported
<input type="checkbox"/>	cloudsql.databases.update	Supported
<input type="checkbox"/>	cloudsql.instances.addServerCa	Supported

1 – 10 of 71 < >

CANCEL ADD

You can also define a custom role using the `gcloud iam roles create` command. The structure of that command is as follows:

```
gcloud iam roles create [ROLE-ID] --project [PROJECT-ID] \
--title=[ROLE-TITLE] --description= [ROLE -DESCRIPTION] \
--permissions= [PERMISSIONS-LIST] --stage=[LAUNCH-STAGE]
```

For example, to create a role that has only App Engine application update permission, you could use the following command:

```
gcloud iam roles create customAppEngine1 --project ace-exam-project \
--title='Custom Update App Engine' \
--description='Custom update' --permissions=appengine.applications.update \
--stage=alpha
```

FIGURE 17.9 The permissions section of the Create Role page with permissions added

←

Create Role

Custom roles let you group permissions and assign them to principals in your project or organization. You can manually select permissions or import permissions from another role. [Learn more](#)

Title *

Custom Role

11 / 100

Description

Created on: 2022-07-23

22 / 256

ID *

CustomRole

Role launch stage

Alpha

+ ADD PERMISSIONS

6 assigned permissions

Filter

Enter property name or value

?

⋮

<input checked="" type="checkbox"/>	Permission ↑	Status
<input checked="" type="checkbox"/>	cloudsql.backupRuns.create	Supported
<input checked="" type="checkbox"/>	cloudsql.backupRuns.delete	Supported
<input checked="" type="checkbox"/>	cloudsql.backupRuns.get	Supported
<input checked="" type="checkbox"/>	cloudsql.backupRuns.list	Supported
<input checked="" type="checkbox"/>	cloudsql.databases.get	Supported
<input checked="" type="checkbox"/>	cloudsql.databases.list	Supported

ⓘ

Some permissions might be associated with and checked by third parties. These permissions contain the third party's service and domain name in the permission prefix.

✓ SHOW ADDED AND REMOVED PERMISSIONS

CREATE

CANCEL

Managing Service Accounts

Service accounts are used to provide identities independent of human users. Service accounts are identities that can be granted roles. Service accounts are assigned to VMs, which then use the permissions available to the service accounts to carry out tasks.

Three things cloud engineers are expected to know how to do are working with scopes, assigning service accounts to VMs, and granting access to a service account to another project.

Managing Service Accounts with Scopes

Scopes are permissions granted to a VM to perform some operation. Scopes authorize the access to API methods. The service account assigned to a VM has roles associated with it. To configure access controls for a VM, you will need to configure both IAM roles and scopes. We have discussed how to manage IAM roles, so now we will turn our attention to scopes.

A scope is specified using a URL that starts with `www.googleapis.com/auth` and is then followed by permission on a resource. For example, the scope allowing a VM to insert data into BigQuery is as follows:

```
www.googleapis.com/auth/bigquery.insertdata
```

The scope that allows viewing data in Cloud Storage is as follows:

```
www.googleapis.com/auth/devstorage.read_only
```

And to write to Compute Engine logs, use this:

```
www.googleapis.com/auth/logging.write
```

An instance can only perform operations allowed by both IAM roles assigned to the service account and scopes defined on the instance. For example, if a role grants only read-only access to Cloud Storage but a scope allows write access, then the instance will not be able to write to Cloud Storage.

To set scopes in an instance, navigate to the VM instance page in Cloud Console. Stop the instance if it is running. On the Instance Detail page, click the Edit link. In the middle of the Edit page, you will see the Access Scopes section, as shown in Figure 17.10.

The options are Allow Default Access, Allow Full Access To All Cloud APIs, and Set Access For Each API. Default access is usually sufficient. If you are not sure what to set, you can choose Allow Full Access, but be sure to assign IAM roles to limit what the instance can do. If you want to choose scopes individually, choose Set Access For Each API. This will display a list of services and scopes like that shown in Figure 17.11.

FIGURE 17.10 Access Scopes section in VM instance detail edit page

Identity and API access ?

Service accounts ?

Service account

Compute Engine default service account

Requires the Service Account User role (roles/iam.serviceAccountUser) to be set for users who want to access VMs with this service account. [Learn more](#)

Access scopes ?

☒ Allow default access
☐ Allow full access to all Cloud APIs
☐ Set access for each API

Firewall ?

Add tags and firewall rules to allow specific network traffic from the Internet

☐ Allow HTTP traffic
☐ Allow HTTPS traffic

FIGURE 17.11 A partial list of services and scopes that can be individually configured

Access scopes ?

☐ Allow default access
☐ Allow full access to all Cloud APIs
☒ Set access for each API

BigQuery

None

Bigtable Admin

None

Bigtable Data

None

Cloud Datastore

None

Cloud Debugger

None

Cloud Platform

None

Cloud Pub/Sub

None

Cloud Source Repositories

None

Cloud SQL

None

Compute Engine

None

You can also set scopes using the `gcloud compute instances set-service-account` command. The structure of the command is as follows:

```
gcloud compute instances set-service-account [INSTANCE_NAME] \
  [--service-account [SERVICE_ACCOUNT_EMAIL] | [--no-service-account] \
  [--no-scopes | --scopes [SCOPES,...]]
```

An example scope assignment using `gcloud` is as follows:

```
gcloud compute instances set-service-account ace-instance \
  --service-account examadmin@ace-exam-project.iam.gserviceaccount.com \
  --scopes compute-rw,storage-ro
```

Assigning a Service Account to a VM Instance

You can assign a service account to a VM instance. First, create a service account by navigating to the Service Accounts section of the IAM & Admin section of the console. Click Create Service Account to display a page like that shown in Figure 17.12.

FIGURE 17.12 Creating a service account in the console

Create service account

- Service account details**

Display name for this service account

*
✕ ↺

Email address:

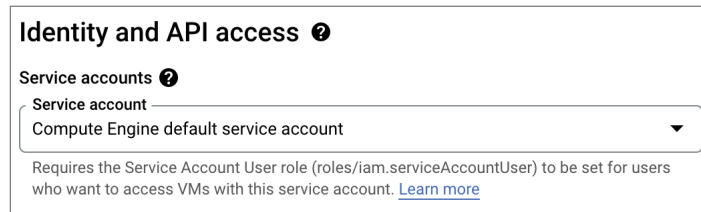
📋

Describe what this service account will do
- Grant this service account access to project (optional)**
- Grant users access to this service account (optional)**

After specifying a name, identifier, and description, click Create and continue. Next, you can assign roles as described earlier, using the console or `gcloud` commands. Once you have assigned the roles you want the service account to have, you can assign it to a VM instance.

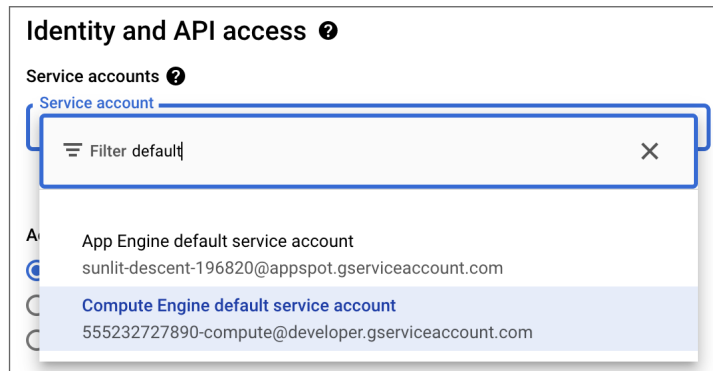
Navigate to the VM Instances page in the Compute Engine section of the console. Select a VM instance and click Edit. This will display a page with a parameter for the instance. Scroll down to see the parameter labeled Service Account (see Figure 17.13).

FIGURE 17.13 Section of Edit Instance page showing the Service Account parameter



From the drop-down list, select the service account you want assigned to that instance, as shown in Figure 17.14.

FIGURE 17.14 List of service accounts that can be assigned to the instance



You can also specify a service instance at the command line when you create an instance by using the `gcloud compute instances create` command. It has the following structure:

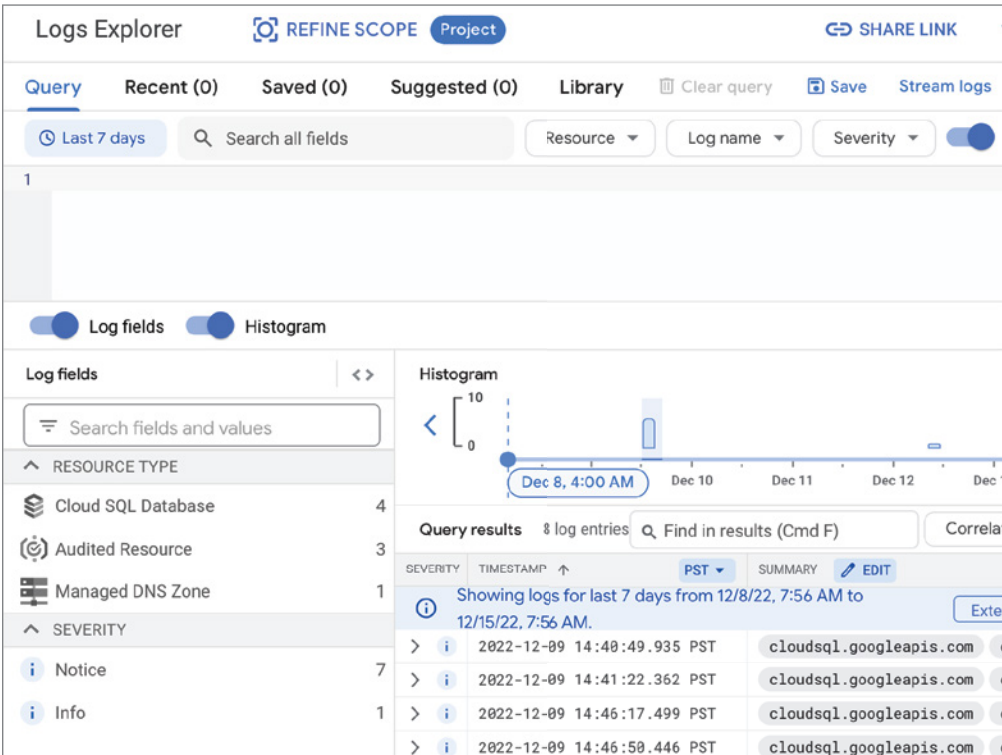
```
gcloud compute instances create [INSTANCE_NAME] \
--service-account [SERVICE_ACCOUNT_EMAIL]
```

To grant access to a project, navigate to the IAM page of the console and add a member. Use the service accounts email as the entity to add.

Viewing Audit Logs

To view audit logs, navigate to the Cloud Logging page in Cloud Console. This will show a listing like that in Figure 17.15.

FIGURE 17.15 Default listing of the Cloud Logging page



Notice you can select the resource, types of logs to display, the log level, and the period of time from which to display entities. Using the Log Name, search for **activity** to see Activity audit logs, **data_access** to see Data Access audit logs, **system_event** to see System Event audit logs, and **policy** to see Policy Denied audit logs.

For additional information on logging, see Chapter 18, “Monitoring, Logging, and Cost Estimating.”

Summary

Access controls in Google Cloud are managed using IAM, basic roles, and scopes. The three basic roles are Owner, Editor, and Viewer. They provide coarse-grained access controls to resources. Scopes are access controls that apply to instances of VMs. They are used to limit operations that can be performed by an instance. The set of operations that an instance can perform is determined by the scopes assigned and the roles assigned to a service account used by the instance. IAM provides predefined roles. These roles are grouped by service. The roles are designed to provide the minimal set of permissions needed to carry out a logical task, such as writing to a bucket or deploying an App Engine application. When predefined roles do not meet your needs, you can define custom roles.

Service accounts are used to enable VMs to perform operations with a set of permissions. The permissions are granted to service accounts through the roles assigned to the service account. You can use the default service account provided by Google Cloud for an instance or you can assign your own.

Exam Essentials

Know the three types of roles: basic, predefined, and custom. Basic roles include Owner, Editor, and Viewer. These were developed prior to the release of IAM. Predefined roles are IAM roles. Permissions are assigned to these roles, and then the roles are assigned to users, groups, and service accounts. Custom roles include permissions selected by the user creating the custom role.

Understand that scopes are a type of access control applied to VM instances. The VM can only perform operations allowed by scopes and IAM roles assigned to the service account of the instance. You can use IAM roles to constrain scopes and use scopes to constrain IAM roles.

Know how to view roles assigned to identities. You can use the Roles tab in the IAM & Admin section of the console to list the identities assigned particular roles. You can also use the `gcloud projects get-iam-policy` command to list roles assigned to users in a project.

Understand that IAM roles support separation of duties and the principle of least privilege. Basic roles did not support least privilege and separation of duties because they are too coarse-grained. Separation of duties ensures that two or more people are required to complete a sensitive task.

Know how to use `gcloud iam roles describe` to view details of a role, including permissions assigned to a role. You can also view roles users have been granted by drilling down into a role in the Roles page of the IAM & Admin section of the console. When working with IAM, you will be using the `gcloud` command when working from the command line.

Understand the different options for accessing scopes when creating an instance. The options are Default Access, Full Access, and Set Access For Each API. If you aren't sure which to use, you can grant full access, but be sure to limit what the instance can do by assigning roles that constrain allowed operations.

Know that Cloud Logging collects logging events. They can be filtered and displayed in the Logging section of Cloud Console. You can filter by resource, type of log, log level, and period of time to display.

Review Questions

You can find the answers in the Appendix.

1. What does IAM stand for?
 - A. Identity and authorization management
 - B. Identity and access management
 - C. Identity and auditing management
 - D. Individual access management
2. When you navigate to IAM & Admin in Cloud Console, what appears in the main body of the page?
 - A. Members and roles assigned
 - B. Roles only
 - C. Members only
 - D. Roles and permissions assigned
3. Why are basic roles classified in a category in addition to IAM?
 - A. They are part of IAM.
 - B. They were created before IAM.
 - C. They were created after IAM.
 - D. They are not related to access control.
4. A developer intern is confused about what roles are used for. You describe IAM roles as a collection of what?
 - A. Identities
 - B. Permissions
 - C. Access control lists
 - D. Audit logs
5. You want to list roles assigned to users in a project called ace-exam-project. What `gcloud` command would you use?
 - A. `gcloud iam get-iam-policy ace-exam-project`
 - B. `gcloud projects list ace-exam-project`
 - C. `gcloud projects get-iam-policy ace-exam-project`
 - D. `gcloud iam list ace-exam-project`

6. You are working in the form displayed after clicking the Add link on the IAM page of IAM & Admin in Cloud Console. There is a field called New Members. What items would you enter in that parameter?
 - A. Individual users only
 - B. Individual users or groups
 - C. Roles or individual users
 - D. Roles or groups
7. You have been assigned the App Engine Deployer role. What operations can you perform?
 - A. Write new versions of an application only.
 - B. Read application configuration and settings only.
 - C. Read application configuration and settings and write new configurations.
 - D. Read application configuration and settings and write new versions.
8. You want to list permissions in a role using Cloud Console. Where would you go to see that?
 - A. IAM & Admin; select Roles. All permissions will be displayed.
 - B. IAM & Admin; select Roles. Check the box next to a role to display the permissions in that role.
 - C. IAM & Admin; select Audit Logs.
 - D. IAM & Admin; select Service Accounts and then Roles.
9. You are meeting with an auditor to discuss security practices in the cloud. The auditor asks how you implement several best practices. You describe how IAM predefined roles help to implement which security best practice(s)?
 - A. Least privilege
 - B. Separation of duties
 - C. Defense in depth
 - D. Options A and B
10. What launch stages are available when creating custom roles?
 - A. Alpha and beta only
 - B. General availability only
 - C. Disabled only
 - D. Alpha, beta, general availability, and disabled
11. What is the `gcloud` command used to create a custom role?
 - A. `gcloud project roles create`
 - B. `gcloud iam roles create`
 - C. `gcloud project create roles`
 - D. `gcloud iam create roles`

12. A DevOps engineer is confused about the purpose of scopes. Scopes are access controls that are applied to what kind of resources?
- A. Storage buckets
 - B. VM instances
 - C. Persistent disks
 - D. Subnets
13. A scope is identified using what kind of identifier?
- A. A randomly generated ID
 - B. A URL beginning with `www.googleusercontent.com`
 - C. A URL beginning with `www.googleapis.com/auth`
 - D. A URL beginning with `www.googleapis.com/auth/PROJECT_ID`
14. A VM instance is trying to read from a Cloud Storage bucket. Reading the bucket is allowed by IAM roles granted to the service account of the VM. Reading buckets is denied by the scopes assigned to the VM. What will happen if the VM tries to read from the bucket?
- A. The application performing the read will skip over the read operation.
 - B. The read will execute because the most permissive permission is allowed.
 - C. The read will not execute because both scopes and IAM roles are applied to determine what operations can be performed.
 - D. The read operation will succeed, but a message will be logged to Cloud Logging.
15. What are the options for setting scopes in a VM?
- A. Allow Default Access and Allow Full Access only.
 - B. Allow Default Access, Allow Full Access, and Set Access For Each API.
 - C. Allow Full Access or Set Access For Each API Only.
 - D. Allow Default Access and Set Access For Each API Only.
16. What `gcloud` command would you use to set scopes?
- A. `gcloud compute instances set-scopes`
 - B. `gcloud compute instances set-service-account`
 - C. `gcloud compute service-accounts set-scopes`
 - D. `gcloud compute service-accounts define-scopes`
17. What `gcloud` command would you use to assign a service account when creating a VM?
- A. `gcloud compute instances create [INSTANCE_NAME] \`
`--service-account [SERVICE_ACCOUNT_EMAIL]`
 - B. `gcloud compute instances create-service-account [INSTANCE_NAME] \`
`[SERVICE_ACCOUNT_EMAIL]`
 - C. `gcloud compute instances define-service-account [INSTANCE_NAME] \`
`[SERVICE_ACCOUNT_EMAIL]`
 - D. `gcloud compute create instances-service-account [INSTANCE_NAME] \`
`[SERVICE_ACCOUNT_EMAIL]`

- 18.** What Google Cloud service is used to view audit logs?
- A.** Compute Engine
 - B.** Cloud Storage
 - C.** Cloud Logging
 - D.** Custom logging
- 19.** What options are available for filtering log messages when viewing audit logs?
- A.** Period time and log level only
 - B.** Resource, type of log, log level, and period of time only
 - C.** Resource and period of time only
 - D.** Type of log only
- 20.** An auditor needs to review audit logs. You assign read-only permission to a custom role you create for auditors. What security best practice are you following?
- A.** Defense in depth
 - B.** Least privilege
 - C.** Separation of duties
 - D.** Vulnerability scanning

Chapter 18

Monitoring, Logging, and Cost Estimating

**THIS CHAPTER COVERS THE FOLLOWING
OBJECTIVES OF THE GOOGLE ASSOCIATE
CLOUD ENGINEER CERTIFICATION EXAM:**

- ✓ 2.1 Planning and estimating Google Cloud product use using the Pricing Calculator
- ✓ 4.6 Monitoring and logging





Monitoring system performance is an essential part of cloud engineering. In this chapter, you will learn about Cloud Operations suite, a Google Cloud service for resource monitoring, logging, and tracing. You will start by creating alerts based on resource metrics and custom metrics. Next, you will turn your attention to logging, with a discussion of how to create log sinks to store logging data outside of Cloud Operations. You'll also see how to view and filter log data. Cloud Operations includes diagnostic tools such as Cloud Trace, which you'll learn about as well. We'll close out the chapter with a review of the Pricing Calculator for estimating the cost of Google Cloud resources and services.

Cloud Monitoring

Cloud Monitoring is a service for collecting performance metrics, logs, and event data from our resources. Metrics include measurements such as the average percentage of CPU utilization over the past minute and the number of bytes written to a storage device in the last minute. Cloud Monitoring includes many predefined metrics. Some examples are shown in Table 18.1 that you can use to assess the health of your resources and, if needed, trigger alerts to bring your attention to resources or services that are not meeting service-level objectives.

TABLE 18.1 Example Cloud Monitoring metrics

Google Cloud Product	Metric
Compute Engine	CPU utilization
Compute Engine	Disk bytes read
BigQuery	Execution times
Bigtable	CPU load
Cloud Functions	Execution count

Cloud Monitoring works in hybrid environments, with support for Google Cloud, Amazon Web Services, and on-premises resources.

Creating Dashboards

Metrics are defined measurements on a resource collected at regular intervals. Metrics return aggregate values, such as the maximum, minimum, or average value of the item measured, which could be CPU utilization, amount of memory used, or number of bytes written to a network interface.

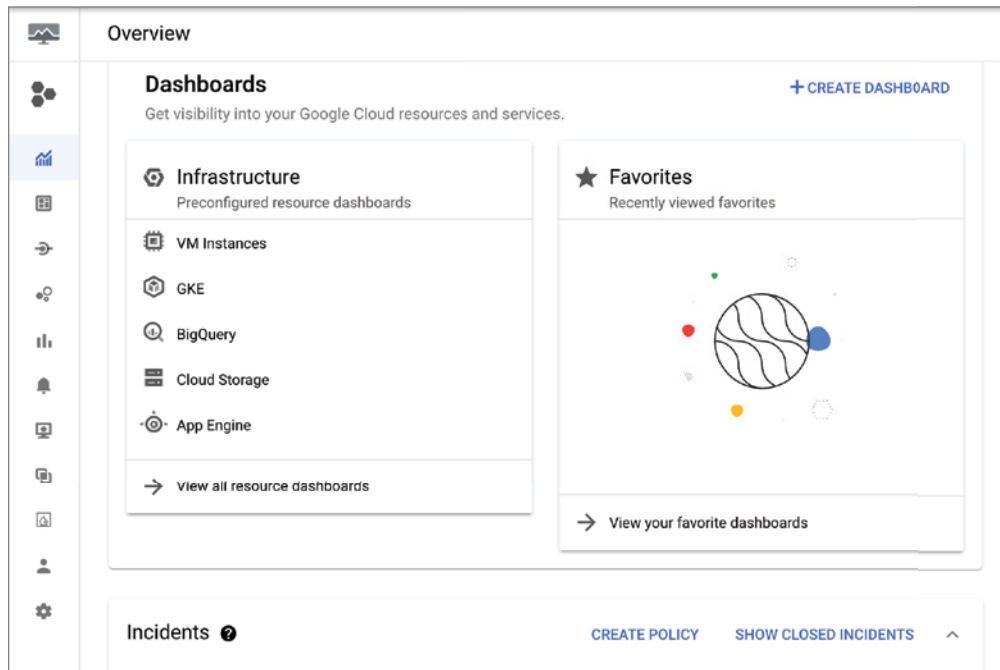
For this example, assume you are working with a VM that has Apache Server and PHP installed. VMs will collect basic metrics and logs, but for more detailed metrics, you can install the Ops Agent, which includes both monitoring and logging support. To install the Ops Agent on a Linux VM, execute the following command at the shell prompt (note that these are not `gcloud` commands):

```
curl -sSO https://dl.google.com/cloudagents/add-google-cloud-ops-agent-repo.sh
sudo bash add-google-cloud-ops-agent-repo.sh --also-install
```

VMs with agents installed collect monitoring and logging data and send it to Cloud Monitoring and Cloud Logging.

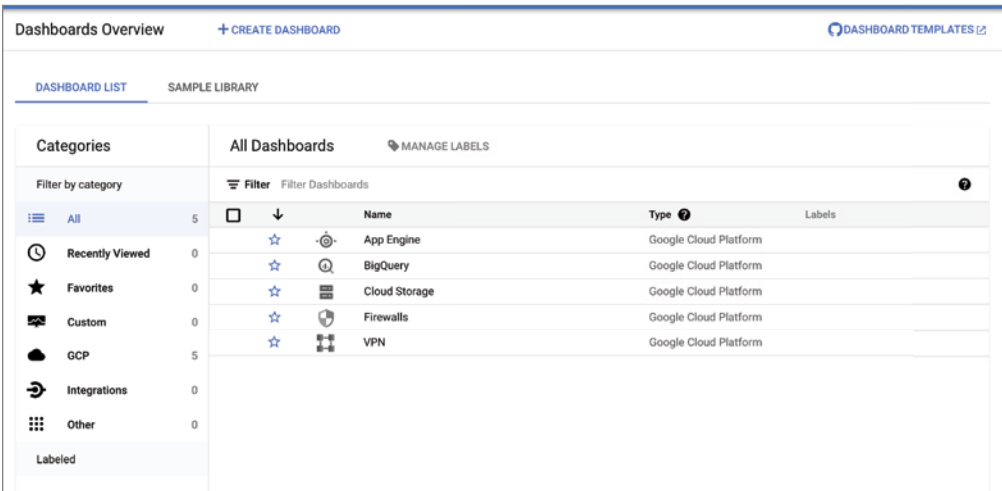
Figure 18.1 shows an example of the Cloud Monitoring Overview page. It includes information on the status of monitoring setup, available dashboards, as well as links to related articles and blog posts. Details on incidents and health checks are available in the overview as well.

FIGURE 18.1 Partial view of Cloud Monitoring Overview page



In the left panel of the Cloud Monitoring page, you can select other monitoring views, including dashboards. An example list of default dashboards is shown in Figure 18.2. The list includes dashboards for App Engine, BigQuery, Cloud Storage, Firewalls, and VPN.

FIGURE 18.2 Available dashboards in Cloud Monitoring



Categories		All Dashboards MANAGE LABELS		
Filter by category		Filter Filter Dashboards ?		
All	5	<input type="checkbox"/>	Name	Type ?
Recently Viewed	0	<input type="checkbox"/>	Labels	
Favorites	0	<input type="checkbox"/>		
Custom	0	<input type="checkbox"/>		
GCP	5	<input type="checkbox"/>		
Integrations	0	<input type="checkbox"/>		
Other	0	<input type="checkbox"/>		
Labeled		<input type="checkbox"/>		

	App Engine	Google Cloud Platform
	BigQuery	Google Cloud Platform
	Cloud Storage	Google Cloud Platform
	Firewalls	Google Cloud Platform
	VPN	Google Cloud Platform

Each of the dashboards shows information relevant to the service. For example, the Cloud Storage dashboard shows data on incidents, buckets, requests and network traffic sent, as shown in Figure 18.3.

In addition to the predefined dashboards available in Cloud Monitoring, you can create your own by clicking Create Dashboard to display the window shown in Figure 18.4.

If you choose to create a line chart, a chart component is added to the dashboard, as shown in Figure 18.5. In this example, the mean CPU Utilization metric will be plotted.

Using Metric Explorer

Metric Explorer is another feature of Cloud Monitoring. It allows you to view a wide variety of metrics by choosing from a list of metrics. Figure 18.6 shows the main page of Metric Explorer, and Figure 18.7 shows details of metrics you can view related to Cloud Storage buckets.

After you select the object count metric for Cloud Storage buckets, Metric Explorer displays a line graph, as shown in Figure 18.8.

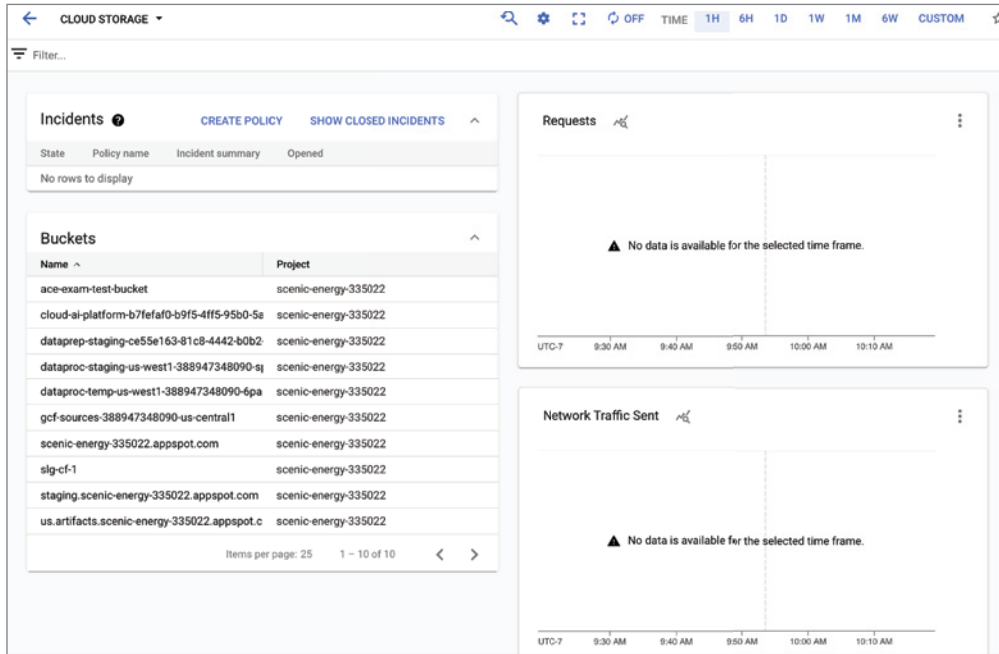
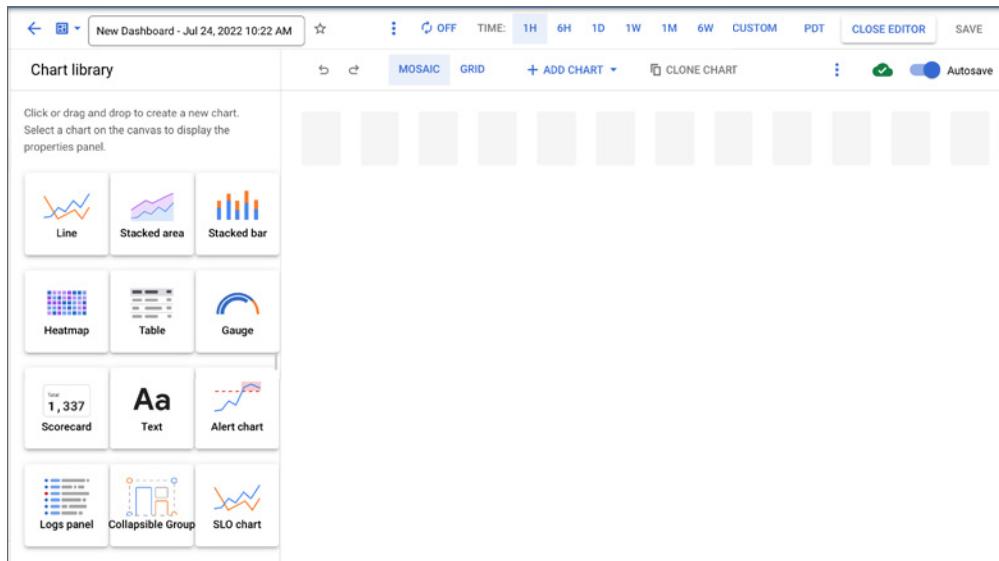
FIGURE 18.3 Cloud Storage monitoring dashboard**FIGURE 18.4** Creating your own dashboard begins with choosing a chart.

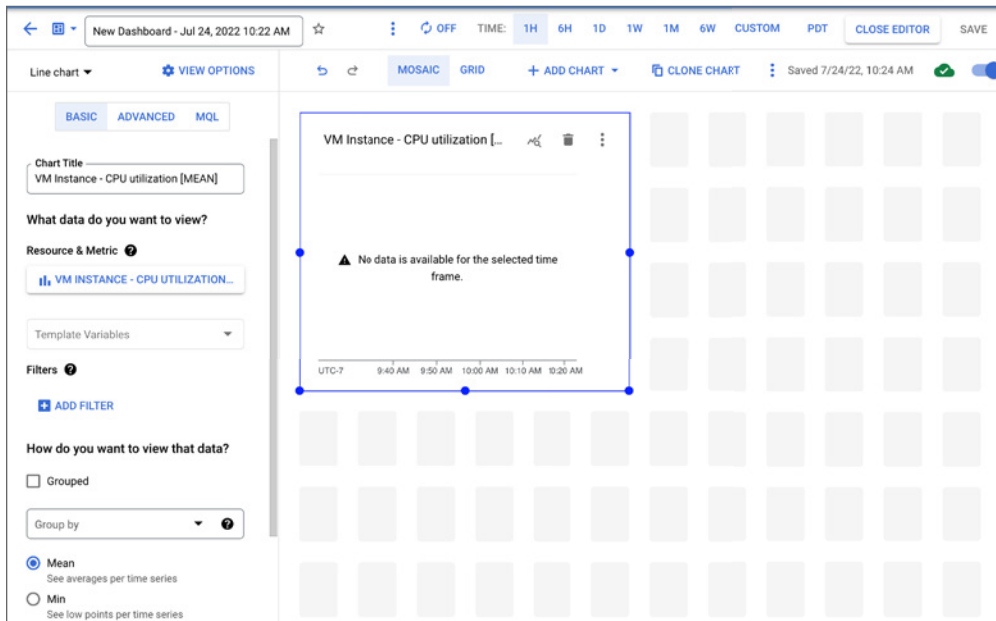
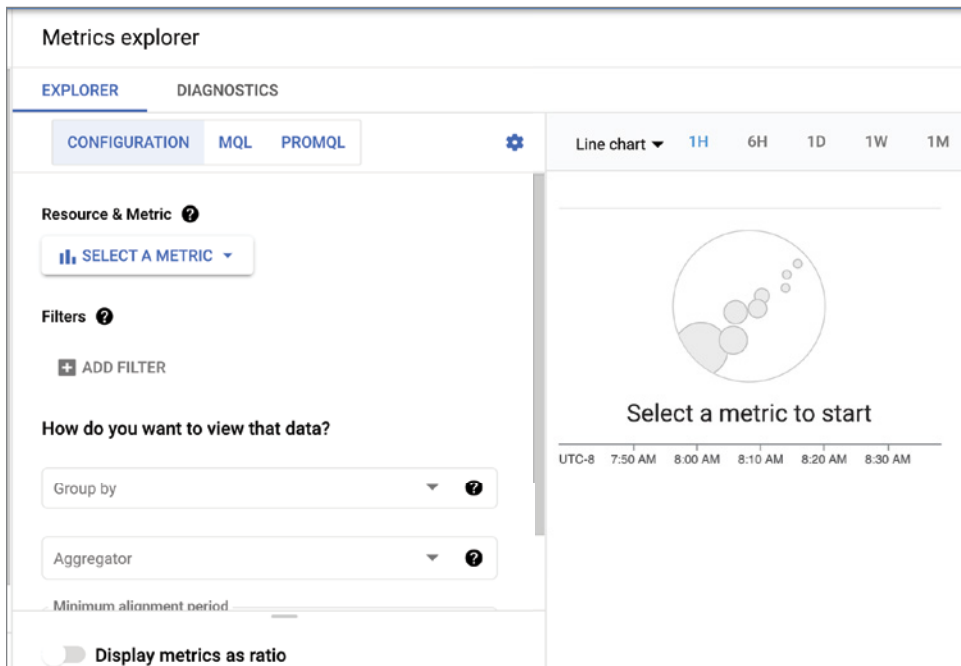
FIGURE 18.5 Adding a line chart to display mean CPU utilization**FIGURE 18.6** Main page of Metric Explorer

FIGURE 18.7 Metrics available for Cloud Storage Buckets

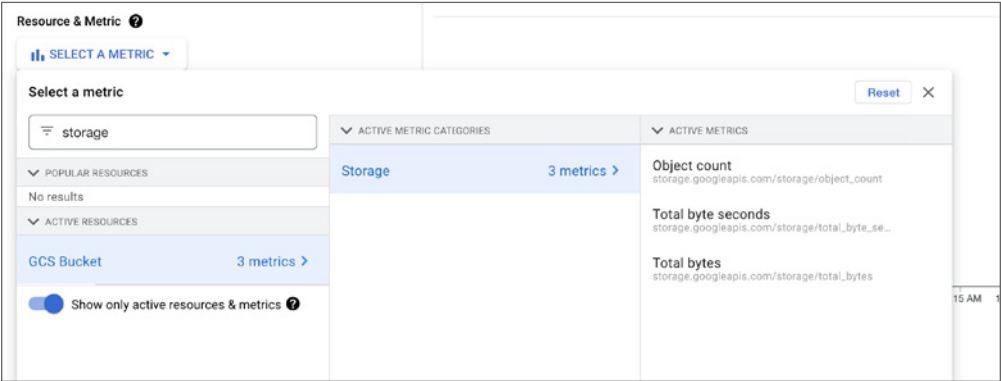
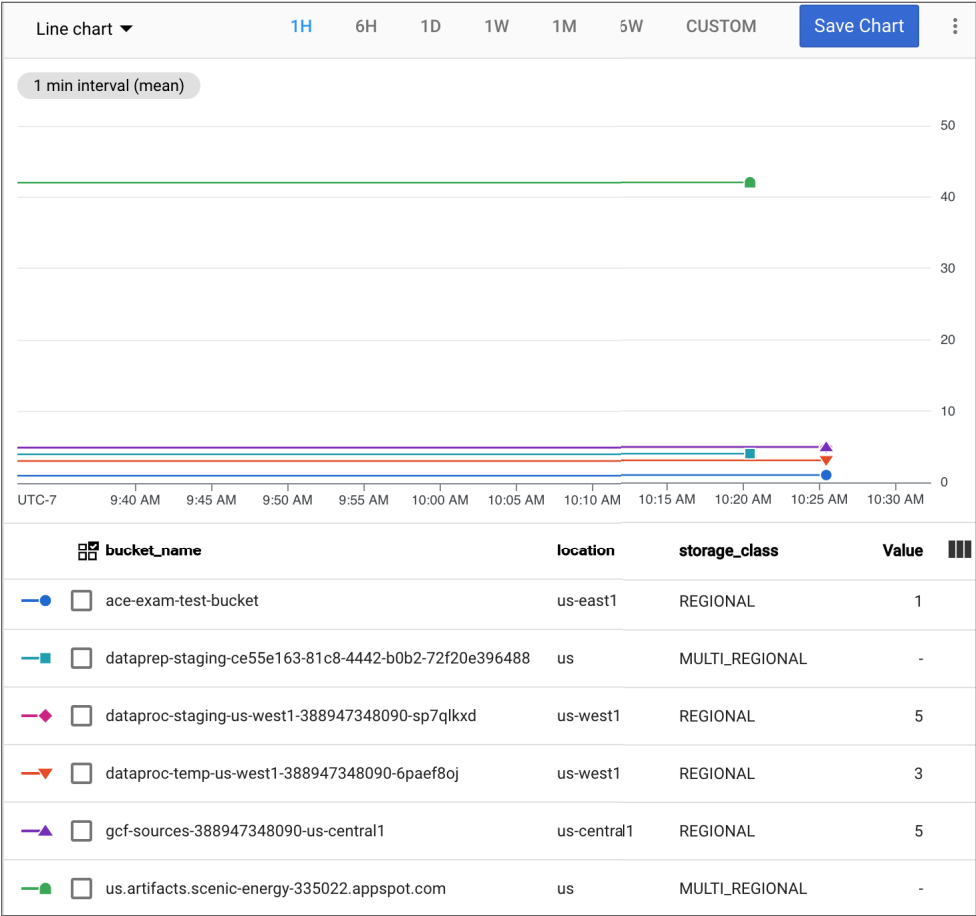


FIGURE 18.8 Line chart of object count metric for Cloud Storage buckets

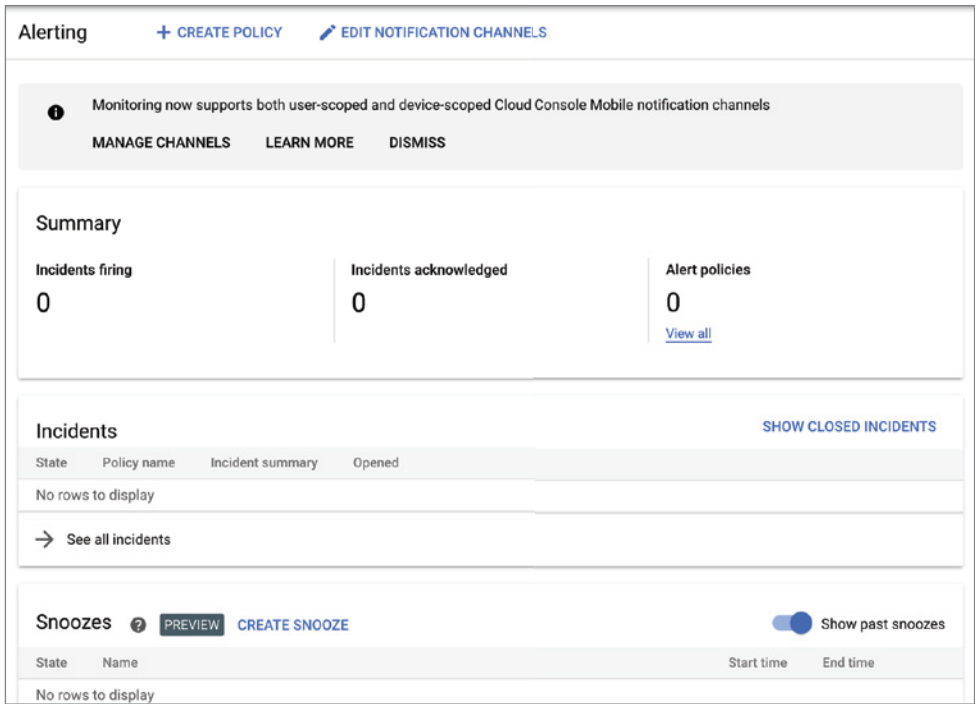


Creating Alerts

Dashboards are useful for getting a quick overview of a set of key metrics, and Metric Explorer is useful when you are investigating an issue and need to view a variety of metrics. If you want to be automatically notified when a metric exceeds some threshold, you can create alerts.

The main Alerting page in Cloud Monitoring is shown in Figure 18.9. It includes a summary count of incidents firing, incidents responded to, and alert policies. There are also detailed listings of incidents and policies.

FIGURE 18.9 Alerting main page of Cloud Logging



To create an alert, you create a policy. A policy is defined for a metric. For example, Figure 18.10 shows the start of defining a policy to alert you when there is a backlog of unacknowledged messages in a Cloud Pub/Sub topic.

FIGURE 18.10 Creating a policy for a Pub/Sub backlog

The screenshot shows a configuration interface for creating an alert policy. It is divided into several sections:

- Select a metric**: A dropdown menu is set to "CLOUD PUB/SUB SUBSCRIPTION - BACKLOG SIZE".
- Add filters**: Labeled as "Optional". A note states: "Selections made on the chart do not affect the alert policy [Learn more](#)". Below this is an "ADD FILTER" button.
- Transform data**:
 - Within each time series**:
 - Rolling window ***: A dropdown menu set to "5 min". A note below says: "Adjust the length of time a signal is calculated for. Example: Mean of CPU utilization for 5 minutes is above 80%".
 - Rolling window function ***: A dropdown menu set to "mean". A note below says: "Function applied to the rolling window".
 - Across time series**: A dropdown menu with a downward arrow.
 - Add secondary data transformation**: A toggle switch that is currently turned off.
- NEXT**: A button at the bottom right of the configuration area.

For a policy, you also specify the condition type, which can be a threshold or a metric absence. A threshold condition triggers when the metric value is above or falls below the specified value for the specified period of time. A metric absence condition is based on the absence of data for a specified period of time (see Figure 18.11).

You also specify an alert trigger, which specifies the scope of the data you consider when checking the alert condition. Figure 18.12 shows the possible options, which include Any Time Series Violates, Percent Of Time Series Violates, Number Of Time Series Violates, and All Time Series Violates.

The last step of creating a policy is to specify notification channels, as shown in Figure 18.13.

Options for notification channels include:

- Email, which sends messages to an email address
- Slack, which sends messages to Slack channels
- SMS, which sends text messages
- Cloud Pub/Sub, which post messages to a Cloud Pub/Sub topic
- PagerDuty, which sends messages to a popular SaaS platform for DevOps
- Webhooks, which invokes an HTTP-based callback function to send messages to an app

FIGURE 18.11 Configuring an alert

The screenshot shows the 'Configure alert trigger' form. At the top, there are links for '+ ADD ALERT CONDITION' and 'DELETE ALERT CONDITION'. The form is titled 'Configure alert trigger'. Under 'Condition type', there are two radio buttons: 'Threshold' (selected) and 'Metric absence'. Below these, there are two dropdown menus: 'Alert trigger' (set to 'Any time series violates') and 'Threshold position' (set to 'Above threshold'). There is a text input field for 'Threshold value' with a unit selector set to 'B'. Below this is a section for 'Advanced Options' which is currently collapsed. It contains a text input field for 'Condition name *' with the value 'Cloud Pub/Sub Subscription - Backlog size'. At the bottom of the form is a 'NEXT' button. At the very bottom of the page are two buttons: 'CREATE POLICY' and 'PROVIDE FEEDBACK'.

FIGURE 18.12 Alert trigger options


The screenshot shows the 'Alert trigger' dropdown menu. The options are: 'Any time series violates' (highlighted), 'Percent of time series violates', 'Number of time series violates', and 'All time series violate'.


You can also specify a period for automatically closing the alert, labels, and documentation to be included with the alert.


FIGURE 18.13 Creating notification channels for an alert

Configure notifications and finalize alert

Configure notifications Recommended


 Use notification channel

Notification Channels 

 We recommend that you create multiple notification channels for redundancy purposes. Google has no control of many of the delivery systems after we have passed the notification to that system. Additionally, a single Google service supports Cloud Console Mobile App, PagerDuty, Webhooks, and Slack. If you use one of these notification channels, then use email, SMS, or Pub/Sub as the redundant channel.
[LEARN MORE](#)

☐ Notify on incident closure


Incident autoclose duration *

7 days 

If data is absent, select a duration after which Incident will automatically close.


Policy user labels


Policy user labels allow you to add your own labels to alert policies for organization. The labels are included in the notification and incident details.(Optional)



Documentation Optional

Enter any documentation you would like included with the notification. You can use markdown, variables, and channel-specific controls. Markdown formatting may not apply to all notification channels.

Text Field 



Too Many Alerts Are as Bad as Too Few

Be careful when crafting monitoring policies. You do not want to subject engineers to so many alerts that they begin to ignore them. This is sometimes called *alert fatigue*. Policies that are too sensitive will generate alerts when no human intervention is required. For example, CPU utilization may regularly spike for brief periods of time. If this is a normal pattern for your environment, and it is not adversely impacting your ability to meet service level agreements, then there is little reason to alert on them. Design policies to identify conditions that actually require the attention of an engineer and are not likely to resolve on their own. Use thresholds that are long enough so that conditions are not triggered on transient states that will not last long. Often by the time an engineer resolves it, the condition is no longer triggering. Designing policies for monitoring is something of an art. You should assume you will need multiple iterations to tune your policies to find the right balance of generating just the right kinds of useful alerts without also generating alerts that are not helpful.

Cloud Logging

Cloud Logging is a service for collecting, storing, filtering, and viewing log and event data. Logging is a managed service, so you do not need to configure or deploy servers to use the service.

The Associate Cloud Engineering Exam guidelines note several logging tasks a cloud engineer should be familiar with:

- Configuring log routers
- Configuring log sinks
- Viewing and filtering logs
- Viewing message details

We'll review each of these in this section.

Log Routers and Log Sinks

Log data is ingested by the Cloud Logging API. From there, log messages are routed to one of three types of sinks: the Required log sink, the Default log sink, or a user-defined log sink.

Sinks are associated with a Google Cloud resource, such as a billing account, project, folder, or organization. Google creates a Required and a Default sink for each billing account, project, folder, or organization.

The Log Router is a service that receives log messages and applies inclusion and exclusion filters to determine which log sinks should receive the message. Log Router supports using combinations of sinks to route logs to multiple storage locations.

Configuring Log Sinks

The Required log sink is used to store admin activity, system events, and access transparency logs. These logs are stored for 400 days, and that duration cannot be changed.

The Default log sink receives log messages that are not sent to the Required log sink. These logs are stored for 30 days by default, but you can change that by configuring a custom retention policy. A 30-day retention is sufficient if you use logs to diagnose operational issues but rarely view the logs after a few days. Your organization may need to keep logs longer to comply with government or industry regulations. You may also want to analyze logs to gain insight into application performance. For these use cases, it is best to export logging data to a long-term storage system like Cloud Storage or BigQuery.

You can create user-defined log buckets in a project. This allows you to route a subset of log messages to a specific Cloud Storage bucket. You can configure a custom retention on a user-defined log bucket.

In addition to storing log messages, Cloud Logging supports log metrics. These are metrics that are based on the content of log messages. If a log message meets a log metric pattern, that message is reflected in the Cloud Monitoring metric associated with the pattern.

Cloud Logging supports several destinations where messages can be routed:

- Cloud Storage, for long-term storage of logs in JSON format
- BigQuery, for logs that will be analyzed
- Cloud Pub/Sub, for JSON messages that are consumed by third-party integrations
- Cloud Logging, for viewing and storing for user-configurable time periods

Viewing and Filtering Logs

To view the contents of logs, navigate to the Cloud Logging section of the console to view the Log Explorer page, shown in Figure 18.14.

Log Explorer allows you to view log messages. Since logs are often quite large, it is important to be able to quickly filter messages to only those you are interested in. Log Explorer allows you to filter messages based on:

- Time (see Figure 18.15)
- Resource type (see Figure 18.16)
- Severity (see Figure 18.17)
- Log query (see Figure 18.18)

FIGURE 18.14 Log Explorer page of the Cloud Logging console

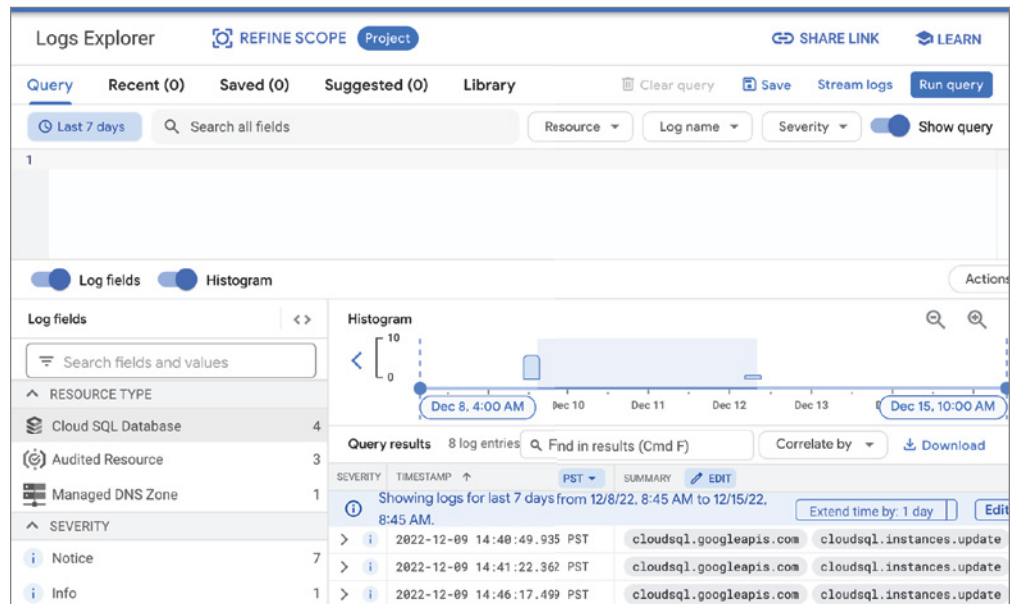


FIGURE 18.15 Time restriction options in Log Explorer

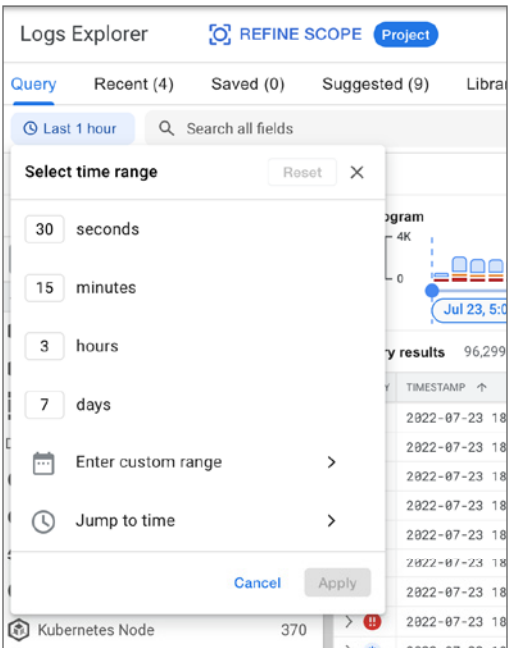


FIGURE 18.16 Resource filtering options in Log Explorer

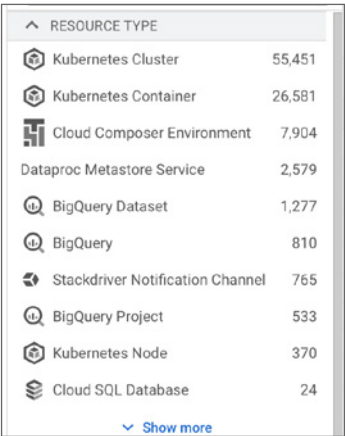


FIGURE 18.17 Severity filtering options in Log Explorer

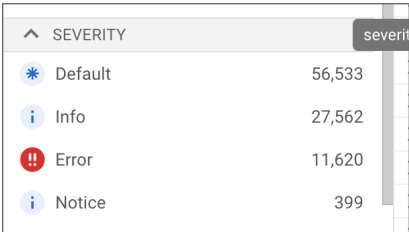
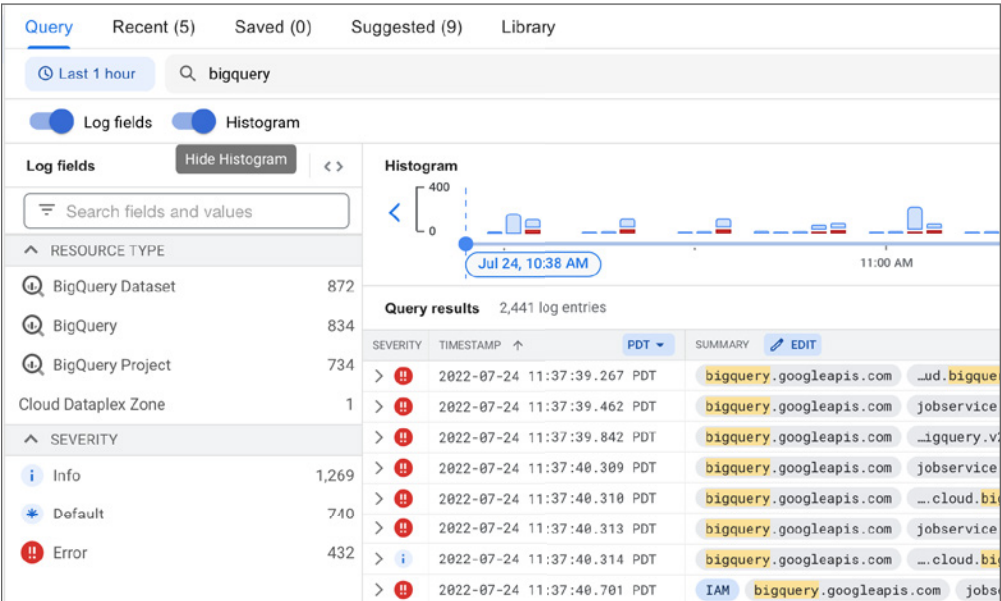


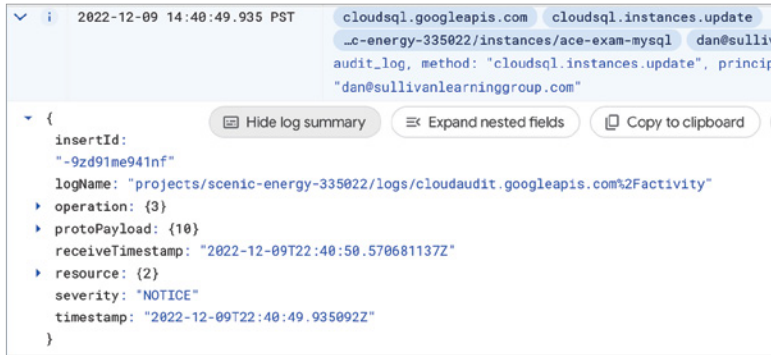
FIGURE 18.18 Queries in Log Explorer can be as simple as keyword searches.



Viewing Message Details

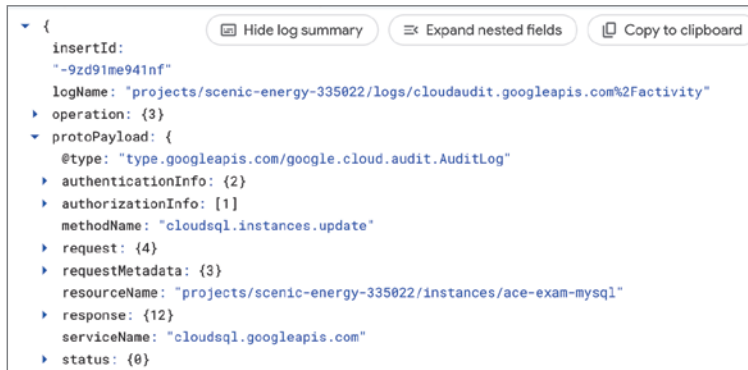
Each log entry is displayed as a single line when you view the contents of logs. Notice the triangle icon at the left of the line. If you click that icon, the line will expand to show additional details. For example, Figure 18.19 shows a log entry expanded by one level.

FIGURE 18.19 A log entry expanded by one level



In the case of the first-level expansion, you see high-level information such as `insertId`, `logName`, and `receiveTimestamp`. You also see other structured data elements, such as `protoPayload` and `resource`. Figure 18.20 shows the `protoPayload` structure expanded.

FIGURE 18.20 A log entry with the `protoPayload` structure expanded



You can continue to drill down individually into each structure if there is a triangle at the left. For example, in the `protoPayload` structure, you could drill down into `authenticationInfo`, `authorizationInfo`, and `requestMetadata`, among others. Figure 18.21 shows the `requestMetadata` section expanded.

FIGURE 18.21 Details of the requestMetadata section of a log message

Query results 2,441 log entries

SEVERITY	TIMESTAMP ↑	PDT ▼	SUMMARY	EDIT
			<pre> protoPayload: { @type: "type.googleapis.com/google.cloud.audit.AuditLog" authenticationInfo: {1} authorizationInfo: [1] metadata: {2} methodName: "google.cloud.bigquery.v2.JobService.InsertJob" requestMetadata: { callerIp: "23.22.133.206" callerSuppliedUserAgent: "Looker/22.12.21 (GPN:Looker;) Google-HTTP-Java-Client/1.39.2 (gzip),gzip(gfe)" } resourceName: "projects/sunlit-descent-196820/jobs/68116593-53e7-4d77-b9b8-fa59fc96cf1a" serviceName: "bigquery.googleapis.com" status: {2} } receiveTimestamp: "2022-07-24T18:37:39.483974140Z" resource: {2} severity: "ERROR" timestamp: "2022-07-24T18:37:39.267066Z" } </pre>	

Using Cloud Trace and Google Cloud Status

Google Cloud provides diagnostic tools that software developers can use to collect information about the performance and functioning of their applications. Specifically, developers can use Cloud Trace to collect data as their applications execute.

Overview of Cloud Trace

Cloud Trace is a distributed tracing system for collecting latency data from an application. This helps developers understand where applications are spending their time and to identify cases where performance is degrading.

From the Cloud Trace console, you can list traces generated by applications running in a project. Traces are generated when developers specifically call Cloud Trace from their applications. In addition to seeing lists of traces, you can create reports.

For the purpose of the Associate Cloud Engineering Exam, remember that Cloud Trace is a distributed tracing application that helps developers and DevOps engineers identify sections of code that are performance bottlenecks.

Viewing Google Cloud Status

In addition to understanding the state of your applications and services, you need to be aware of the status of Google Cloud services. You can find this status in the Google Cloud Status Dashboard, which displays information on service status: Available, Service Disruption, or Service Outage.

To view the status of Google Cloud services, navigate to <https://status.cloud.google.com>. Figure 18.22 shows the overview status of major geographical areas.

FIGURE 18.22 Overview status of Google Cloud services

Service Health

This page provides status information on the services that are part of Google Cloud. Check back here to view the current status of the services listed below. If you are experiencing an issue not listed here, please [contact Support](#). Learn more about what's posted on the dashboard in [this FAQ](#). For additional information on these services please visit <https://cloud.google.com/>.

[Overview](#) [Americas \(regions\)](#) [Europe \(regions\)](#) [Asia Pacific \(regions\)](#) [Multi-regions](#)

Check status by product and location. Click the other tabs to check the status for specific regions and multi-regions.

Multi-regions: Services in a multi-region location are managed by Google to be redundant and distributed across multiple regions in a large geographic area. [Learn more](#)

Global: status for a specific globally distributed service offered to the product. This status does not refer to all product service around the world, just the specific global service.

✓ Available

ⓘ Service information

⚠ One or more regions affected

Products	Americas (regions)	Europe (regions)	Asia Pacific (regions)	Multi-regions
Access Approval				
Access Context Manager	✓	✓	✓	
Access Transparency				
AI Platform Prediction	✓	✓	✓	

There are also tabs for seeing more detail within major geographic regions. For example, Figure 18.23 shows more details about the status of American regions.

Using the Pricing Calculator

Google provides a Pricing Calculator to help Google Cloud users understand the costs associated with the services and configuration of resources they choose to use. You will find the Pricing Calculator at <https://cloud.google.com/products/calculator> (see Figure 18.24).

FIGURE 18.23 More detailed view of American service status

Overview	<u>Americas (regions)</u>	Europe (regions)	Asia Pacific (regions)	Multi-regions
----------	---------------------------	------------------	------------------------	---------------

Check status by region and product in the Americas.

✔ Available
ℹ Service information
⚠ Service disruption
✖ Service outage

Products	us-central1 Iowa	us-east1 South Carolina	us-east4 Northern Virginia	us-east5 Columbus	us-south1 Dallas	us-west1 Oregon
Access Context Manager	✔	✔	✔	✔		✔
AI Platform Prediction	✔	✔	✔			✔
AI Platform Training	✔	✔	✔			✔
Anthos Service Mesh	✔	✔	✔	✔	✔	✔

FIGURE 18.24 Google Cloud Pricing Calculator

Google Cloud Pricing Calculator

Prices are up to date. Last update: 20-July-2022

COMPUTE ENGINE

GKE STANDARD

GKE AUTOPILOT

BACKUP FOR GKE

CLOUD TPU

ALLOYDB

VERTEX AI TRAINING

VERTEX AI PREDICTION

VE SE

Estimate

Search for a product you are interested in.

?

Instances

Number of instances

?

What are these instances for?

?

Operating System / Software

Free: Debian, CentOS, CoreOS, Ubuntu or BYOL (Bring Your Own License)

?

Provisioning model

Regular

?

Machine Family

General purpose

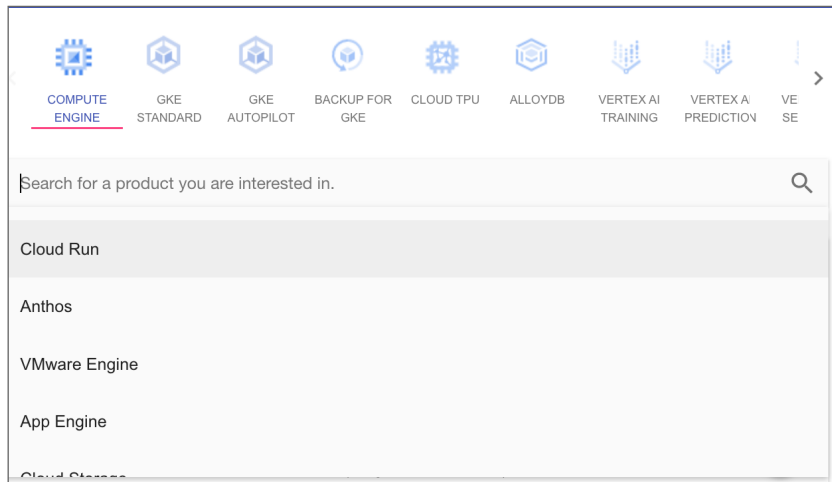
?

Series

E2

?

With the Pricing Calculator, you can specify the configuration of resources, the time they will be used, and in the case of storage, the amount of data that will be stored. Other parameters can be specified too. Those will vary according to the service you are calculating charges for. Figure 18.25 shows some of the services available to use with the Pricing Calculator.

FIGURE 18.25 Partial list of services available in the Pricing Calculator

After selecting a service, you can specify a configuration specific to that service. For example, when estimating the price of a Compute Engine virtual machine, you will provide:

- Number of instances
- Machine types
- Operating system
- Average usage per day and week
- Persistent disks
- Load balancing
- Cloud tensor processing units (TPUs) (for machine learning applications)




After you enter data in the fields, the Pricing Calculator will generate an estimate, such as that shown in Figure 18.26.

Different resources will require different parameters for an estimate. For example, when estimating the price of using BigQuery you will need to specify both storage and query parameters. The storage parameters are for active and long-term storage as well as the volume of data in streaming inserts and streaming reads. For queries, you will need to specify the volume of data queried because BigQuery charges based on the amount of data processed or scanned to get query results. Currently, the first 1 TB of data processed during querying per month is free.

FIGURE 18.26 Example price estimate for five e2-standard-2 VMs

Estimate

Compute Engine

5 x Workloads   

Region: Iowa

3,650 total hours per month

Provisioning model: Regular

Instance type: e2-standard-2 USD 244.59

Operating System / Software: Free

Estimated Component Cost: USD 244.59 per 1 month

Total Estimated Cost: USD 244.59 per 1 month

Estimate Currency
USD - US Dollar ▼

EMAIL

SAVE

DOWNLOAD*

Summary

As a cloud engineer, you are responsible for monitoring the health and performance of applications and cloud services. Google Cloud provides multiple tools, including monitoring, logging, and tracing services.

Cloud Monitoring allows you to define alerts on metrics, such as CPU utilization, so that you can be notified if part of your infrastructure is not performing as expected. Cloud Logging collects, stores, and manages log entries. Logs can be stored in Cloud Logging–provided buckets or user-defined buckets. Log messages can be routed to Cloud Storage, BigQuery, or Cloud Pub/Sub. Cloud Trace provides distributed tracing services to identify slow-running parts of code.

You can always get the status of Google Cloud services at the Google Cloud Status Dashboard at <https://status.cloud.google.com>.

The Pricing Calculator is designed to help you estimate the cost of services in the Google Cloud. It is available at <https://cloud.google.com/products/calculator>.

Exam Essentials

Understand the need for monitoring and the role of metrics. Metrics provide data on the state of applications and infrastructure. You create conditions, like CPU exceeding 80 percent for 5 minutes, to trigger alerts. Alerts are delivered by notification channels. Google Cloud has a substantial number of predefined metrics, but you can create custom metrics as well.

Know how to collect, store, filter, and display log data using Cloud Logging. Logs can come from virtually any source. Logging keeps log data in the Default bucket for 30 days unless a custom retention policy is specified. If you need to keep log data longer than that, you need to export the data to a log sink. Log sinks may be a Cloud Storage bucket, a Big-Query data set, or a Cloud Pub/Sub topic.

Know how to filter logs. Logs can contain a large amount of data. Use filters to search for text or labels, limit log entries by log type and severity, and restrict the time range to a period of interest.

Log entries are hierarchical. Cloud Logging shows a single-line summary for a log entry by default, but you can drill down into the details of a log entry. Use the Expand All and Collapse All options to quickly view or hide the full details of a log entry.

Know how to use the Cloud Trace distributed tracing service. Software developers include Cloud Trace code in their applications to record trace data. Trace data can be viewed as individual traces, or you can create reports that include parameters specifying a subset of traces you want to include.

Know where Google Cloud publishes the status of services. The Google Cloud Status page includes a list of all services, their current status, and the status over the near past. If there is an incident in a service, you will find additional details on the impact and root cause of the problem.

Know how to use the Pricing Calculator to estimate the cost of resources and services in the Google Cloud. The calculator is available at <https://cloud.google.com/products/calculator>. There is a separate calculator for each service. Each service has its own set of parameters for estimating costs. The Pricing Calculator allows you to estimate the cost of multiple services and generate a total estimate for all those services.

Review Questions

1. What Cloud Operations service is used to generate alerts when the CPU utilization of a VM exceeds 80 percent?
 - A. Cloud Logging
 - B. Cloud Monitoring
 - C. Cloud Trace
 - D. Cloud Debugger
2. You have just created a virtual machine, and you'd like to collect detailed metrics about the VM. What do you need to do to the VM to have this happen?
 - A. Install a Cloud Operations image.
 - B. Install the Ops Agent on the VM.
 - C. Edit the VM configuration in Cloud Console and select the Monitor With Cloud Monitoring option.
 - D. Set a notification channel.
3. Where can Cloud Monitoring be used to monitor resources?
 - A. In Google Cloud only
 - B. In Google Cloud and Amazon Web Services only
 - C. In Google Cloud and on-premises data centers
 - D. In Google Cloud, Amazon Web Services, and on-premises data centers
4. You are responsible for the reliability and availability of several services running in Kubernetes Engine. You have determined that you need to monitor several metrics to get information on the state of the services. You'd like to see all of these metrics displayed as line charts, one for each metric. All of the line charts should be available on a single-page view. What would you use to create such a page view?
 - A. Cloud Monitoring Dashboard
 - B. Cloud Logging sink
 - C. Cloud Monitoring Alert
 - D. BigQuery data set
5. You have created a condition of CPU utilization, and you want to receive notifications. Which of the following are options?
 - A. Email only
 - B. PagerDuty only
 - C. Webhooks and PagerDuty
 - D. Email, PagerDuty, and Webhooks

6. When you create a policy to notify you of a potential problem with your infrastructure, you can specify optional documentation. Why would you bother putting documentation in that form?
 - A. It is saved to Cloud Storage for future use.
 - B. It can help you or a colleague understand the purpose of the policy.
 - C. It can contain information that would help someone diagnose and correct the problem.
 - D. Options B and C.
7. What is alert fatigue, and why is it a problem?
 - A. Too many alert notifications are sent for events that do not require human intervention, and eventually DevOps engineers begin to pay less attention to notifications.
 - B. Too many alerts put unnecessary load on your systems.
 - C. Too few alerts leave DevOps engineers uncertain of the state of your applications and infrastructure.
 - D. Too many log messages make it hard to find important messages.
8. How long is log data stored in the Default bucket of Cloud Logging?
 - A. 7 days
 - B. 15 days
 - C. 30 days
 - D. 60 days
9. You need to store log entries for a longer period of time than Cloud Logging retains them in the Default bucket. What is the best option for preserving log data?
 - A. There is no option; once the data retention period passes, Cloud Logging deletes the data.
 - B. Create a user-defined bucket and configure a retention policy.
 - C. Write a Python script to use the Cloud Logging API to write the data to Cloud Storage.
 - D. Write a Python script to use the Cloud Logging API to write the data to BigQuery.
10. Which of the following are options for logging sinks?
 - A. Cloud Storage bucket only
 - B. BigQuery dataset and Cloud Storage bucket only
 - C. Cloud Pub/Sub topic only
 - D. Cloud Storage bucket, BigQuery dataset, and Cloud Pub/Sub topic
11. Which of the following can be used to filter log entries when viewing logs in Cloud Logging?
 - A. Log query only
 - B. Resource type and severity only
 - C. Time and severity only
 - D. Log query, resource type, severity, and time

12. Which of the following is not a standard log level that can be used to filter log viewings?
- A. Critical
 - B. Halted
 - C. Warning
 - D. Info
13. You are viewing log entries and spot one that looks suspicious. You are not familiar with that kind of log entry, and want to find out what, specifically, is in a field called `metadataRequest`. What would you do?
- A. Expand the `metadataRequest` field in the JSON structure of the message.
 - B. View the message in Metric Explorer.
 - C. Write a Python script to reformat the log entry.
 - D. Click the Show Detail link next to the log entry.
14. What Cloud Operations service is best for identifying where bottlenecks exist in your application?
- A. Monitoring
 - B. Logging
 - C. Trace
 - D. Debugger
15. There is a performance problem in a microservice. You have reviewed application outputs but cannot identify the problem. What Cloud Operations service would you use to gain insight into the performance of the services throughout execution?
- A. Monitoring
 - B. Logging
 - C. Trace
 - D. Debugger
16. You believe there may be a problem with BigQuery in the us-central zone. Where would you go to check the status of the BigQuery service for the quickest access to details?
- A. Email Google Cloud Support.
 - B. Check <https://status.cloud.google.com>.
 - C. Check <https://bigquery.status.cloud.google.com>.
 - D. Call Google tech support.
17. You would like to estimate the cost of Google Cloud resources you will be using. Which services would require you to have information on the virtual machines you will be using?
- A. Compute Engine and BigQuery
 - B. Compute Engine and Kubernetes Engine
 - C. BigQuery and Kubernetes Engine
 - D. BigQuery and Cloud Pub/Sub

- 18.** You are generating an estimate of the cost of using BigQuery. One of the parameters is Query Pricing. You have to specify a value in TB units. What is the value you are specifying?
- A.** The amount of data stored in BigQuery
 - B.** The amount of data returned by the query
 - C.** The amount of data scanned by the query
 - D.** The number of partitions used
- 19.** Why do you need to specify the operating system to be used when estimating the cost of a VM?
- A.** All operating systems are charged a fixed rate.
 - B.** Some operating systems incur a cost.
 - C.** It's not necessary; it is only included for documentation.
 - D.** To estimate the cost of Bring Your Own License configurations.
- 20.** Which types of log messages are sent to the Required log sink?
- A.** Operating system messages only
 - B.** Admin activity messages only
 - C.** Admin activity and system events only
 - D.** Admin activity, system events, and access transparency

Appendix

Answers to Review Questions



Chapter 1: Overview of Google Cloud

1. B. The correct answer is B. A basic unit for purchasing computing resources in Google Cloud is the virtual machine (VM). Option A is incorrect; a cache is a low-latency storage system. Option C is incorrect; a block is a unit of storage on persistent disks. Option D is incorrect; a subnet is a networking abstraction.
2. D. The correct answer is D. When using managed clusters, the cloud provider will monitor the health of servers, also known as nodes, in the cluster; set up networking between nodes in the cluster; and configure firewall and other security controls.
3. B. The correct answer is B. Cloud Run is a serverless platform for running containers, and Cloud Functions is a service for executing short-running functions in response to events. Kubernetes Engine is a managed cluster service, and both Kubernetes Engine and Compute Engine require you to configure servers. Neither Compute Engine nor Kubernetes is a serverless option.
4. B. The correct answer is B. Object storage, like Cloud Storage, provides redundantly stored objects without limits on the amount of data you can store, which makes option B correct. Since filesystem functionality is not required, option D is not a good option. Block storage could be used, but you would have to manage your own replication to ensure high availability and it would cost more than object storage. Caches are transient, in-memory storage and are not high-availability, persistent storage systems.
5. D. The correct answer is D. Block sizes in a block storage system can vary. Block size is established when a filesystem is created. In Linux, 4 KB block sizes are commonly used.
6. C. The correct answer is C. Firewalls in Google Cloud are software-defined network controls that limit the flow of traffic into and out of a network or subnetwork. Routers are used to move traffic to appropriate destinations on the network. Identity access management is used for authenticating and authorizing users; it is not relevant to network controls between subnetworks. IP address tables are not a security control.
7. C. Option C is correct because specialized services in Google Cloud, like AutoML are serverless. Google manages the compute resources used by the services. There is no need for a user to allocate or manage servers.
8. B. Option B is correct; investing in servers works well when an organization can accurately predict the number of servers and other equipment it will need for an extended period and can utilize that equipment consistently. Startups are not established businesses with histories that can guide expected needs in three to five years. It does not matter if a budget is fixed or variable; investing in servers should be based on demand for server capacity.
9. B. The characteristics of the server, such as the number of virtual servers, the amount of memory, and the region where you run the VM, influence the cost, so option B is correct. Time of day is not a factor, nor is the type of application you run on the VM.
10. D. AutoML is one of Google Cloud's specialized services. Users of the service do not need to configure any VMs to use the service.

11. B. Containers give the most flexibility for using the resources of a cluster efficiently, and orchestration platforms reduce the operations overhead, which makes option B correct. Running in a single VM is not recommended because if the server fails, all services will be down. Using two VMs with one read-only is not useful. Read-only servers are sometimes used with databases, but there was no mention of databases in the question. Using a small VM and upgrading when it is no longer able to keep up with the workload delivers poor-quality service to users and should be avoided.
12. D. The correct answer is D. All of the operations are available to a system administrator after creating a VM.
13. A. Option A is correct; Cloud Filestore is based on the Network File System (NFS), which is a distributed file management system. The other options are filesystems supported by Linux.
14. A. When you create resources, they are created within a VPC. Resources are added to the VPC and are not accessible outside the VPC unless you explicitly configure them to be. A subdomain is related to web domains and not related to the organization of Google Cloud resources. Clusters, such as Kubernetes clusters, may be in your network, but not all resources are necessarily in a cluster.
15. D. The correct answer is D. Caches use memory, and that makes them the fastest storage type for reading data. Caches are data stores on the back end of distributed systems, not the clients. A cache would have no effect on client-side JavaScript execution. Caches can lose data in the cache if power is lost and the data would have to be reloaded. Caches can get out of sync with the system of truth because the system of truth could be updated, but the cache may not be updated. Caches have faster read times than SSDs and HDDs.
16. B. Option B is correct; cloud providers have large capacity and can quickly allocate those resources to different customers. With a mix of customers and workloads, they can optimize the allocation of resources. Option A is incorrect; cloud providers do not take resources from one customer to give them to another, with the exception of preemptible instances. Option C is incorrect; cloud providers usually offer discounts for increased use.
17. C. Option C is correct. Specialized services are monitored by Google so that users do not have to monitor them. Specialized services provide a specific compute functionality but do not require the user to configure any resources. They also provide APIs.
18. B. The correct answer is B. Attached drives are block storage devices. Cloud Storage is an object storage service and does not attach directly to a VM. NoSQL is a type of database, not a storage system. There is no such thing as SQL storage; SQL is a query language used in relational databases.
19. C. The correct answer is C. Databases require persistent storage on block devices. Object storage does not provide data block or filesystem storage. Data storage is not a type of storage system. Caches are often used with databases to improve read performance, but they are volatile and are not suitable for persistently storing data files.
20. B. The correct answer is B. All three services are serverless, so the user does not need to configure VMs. Cloud Storage is charged based on time and size of data stored. Cloud Run and Cloud Functions are not restricted to just the Go language.

Chapter 2: Google Cloud Computing Services

1. C. The correct answer is C. Cloud Load Balancing distributes workloads within and across regions, provides health checks, and implements autoscaling. Cloud DNS provides domain name services, such as translating a URL like `www.example.com` to an IP address. Cloud Spanner is a distributed relational database but does not implement workload distribution. Cloud CDN distributes content across regions to reduce latency when delivering content to users across the globe.
2. C. The correct answer is C. Cloud Run allows you to run containers in a serverless service. Kubernetes Engine is an orchestration platform for running containers. Both provide container management services and support stateful applications. Cloud Run allows for running containers in a managed service but does not currently support managing state within the container. The App Engine standard environment runs applications in language-specific sandboxes and is not a general container management system. Cloud Functions is a serverless service for running code in response to events.
3. D. The correct answer is D. Options A and B are both correct answers. The Apigee API platform provides policy-based rate-limiting and routing services to help accommodate spikes in traffic. It also provides OAuth 2.0 and SAML authentication. It does not provide version control; Cloud Source Repositories is the service used for version control.
4. A. The correct answer is A. Cloud Armor builds on Google Cloud's load balancing services to provide the ability to allow or restrict access based on IP address, deploy rules to counter cross-site scripting attacks, and provide countermeasures to SQL injection attacks. Cloud CDN is a content distribution service, not a security service. Identity and access management is a security service, but it is for authorization, not denial-of-service mitigation. Virtual private clouds are used to restrict network access to an organization's resources, but it does not have features to mitigate denial-of-service attacks. Also, Cloud CDN acts as a first line of defense in the case of DDoS attacks.
5. A. The correct answer is A. This is a good use case for preemptible VMs because they could reduce the cost of running the second application without the risk of losing work. Since tasks are deleted from the queue only after they are completed, if a preemptible VM is shut down before completing the task another VM can perform the task. Also, there is no harm in running a task more than once, so if two VMs do the same task, it will not adversely affect the output of the application. DataProc is a managed Hadoop and Spark cluster and Spanner is a globally scalable relational database; neither are appropriate products for this task.
6. B. The correct answer is B. Cloud Memorystore is the Google Cloud managed service for caching data in memory using either Redis or memcached. Cloud SQL is a relational database service and might be a good option for the back-end database. Cloud Spanner is a global relational database and is a good option when you need a globally scalable, relational database. Cloud Firestore is a document database suitable for product catalogs, user profiles, and other semi-structured data.

7. D. The correct answer is D. All three of the services listed, Compute Engine, Cloud Storage, and network firewalls, can be managed and configured using Cloud SDK.
8. D. The correct answer is D. Cloud Functions is a serverless product, so no configuration is required.
9. D. The correct answer is D. The Cloud Logging service is used to consolidate and manage logs generated by applications and servers.
10. B. The correct answer is B. The data analytics set of specialized services includes products that help with extraction, transformation, and loading (ETL) and work with both batch and streaming data. The Apigee API platform is used for managing APIs and does not meet the needs described. AI and machine learning might be useful for analyzing data in the data warehouse, but the services in that set are not always helpful for ETL operations. Cloud SDK is used to control services but by itself is not directly able to perform the operations needed.
11. B. The correct answer is B. Bigtable is designed to accept billions of rows of data. Spanner is a relational database and supports transactions, but they are not needed. Cloud SQL MySQL and Cloud SQL PostgreSQL would be difficult to scale to this level of read and write performance.
12. A. The correct answer is A. Cloud Firestore is a database service that can synchronize data between mobile devices and centralized storage. Spanner is a global relational database for large-scale applications that require transaction support in highly scaled databases. Cloud CDN is a distributed storage system for reducing latency when delivering static content to web application users. Cloud SQL could be used but would require more custom development to synchronize data between mobile devices and the centralized data store.
13. B. The correct answer is B. A computationally intensive application obviously requires high CPUs, but the fact that there are many floating-point calculations indicates that a GPU should be used. You might consider running this in a cluster, but the work is not easily distributed over multiple servers, so you will need to have a single server capable of handling the load. Immediate access to large amounts of data indicates that a high-memory machine should be recommended.
14. B. The correct answer is B. Identities are abstractions of users. They can also represent characteristics of processes that run on behalf of a human user or a VM in the Google Cloud; these are known as service accounts. Identities are not related to VM IDs. Roles are collections of privileges that can be granted to identities. Option D is synonymous with option C.
15. C. The correct answer is C. Natural Language services provides functionality for analyzing text. Vertex AI is a unified platform for building machine learning models, but since the client is not an expert in machine learning, a specialized service such as Natural Language is a better option. Recommendation AI is used to make product recommendations to customers. Text-to-Speech is a service for converting natural language text to human-sounding speech.

16. B. The correct answer is B. Both options B and D would meet the need of running Spark, which would give the data scientists access to the machine library they need. However, option D requires that they manage and monitor the cluster of servers, which would require more DevOps and administration work than if they used the Dataproc service. Option C, BigQuery, is a scalable database, not a platform for running Spark. Cloud Spark is a fictitious product and does not exist in Google Cloud.
17. B. The correct answer is B. Spanner supports ANSI SQL 2011 and global transactions. Cloud SQL supports standard SQL but does not have global transaction. Firestore and Bigtable are NoSQL databases.
18. A. Dataproc is designed to execute workflows in both batch and streaming modes, which makes option A correct. BigQuery is a data warehouse service. Firestore is a document database. AutoML is a machine learning service.
19. C. The correct answer is C. App Engine standard environment provides a serverless Python sandbox that scales automatically. App Engine flexible environment runs containers and requires more configuration. Cloud Engine and Kubernetes Engine both require significant management and monitoring.
20. D. The correct answer is D. Error reporting consolidates crash information. Cloud Monitoring collects metrics on application and server performance. Logging is a log management service. Cloud Dataproc is not an observability tool but is a managed Hadoop and Spark service.

Chapter 3: Projects, Service Accounts, and Billing

1. A. Option A, the correct answer, separates the two main applications into their own folders and further allows separating private insurance from government payer by using folders for each. This satisfies the regulatory need to keep the government payer software isolated from other software. Option B does not include an organization, which is the root of the resource hierarchy. Option C is not flexible regarding differences in constraints on different applications. Option D is false because option A does meet the requirements.
2. C. Resource hierarchies have a single organization at the root, which makes option C correct. Below that, there are folders that can contain other folders or projects. Folders can contain multiple folders and multiple projects.
3. B. Service accounts are designed to give applications or VMs permission to perform tasks. Billing accounts are for associating charges with a payment method. Folders are part of resource hierarchies and have nothing to do with enabling an application to perform a task. Messaging accounts are a fictitious option.
4. A. Inherited policies can be overridden by defining a policy at a folder or project level. Service accounts and billing accounts are not part of the resource hierarchy and are not involved in overriding policies.

5. E. All of the listed types of constraints are supported in policies.
6. B. Option B is the correct answer because Publisher is not a primitive role. Owner, Editor, and Viewer are the three basic roles in Google Cloud.
7. D. Basic roles only include the Owner, Editor, and View permissions. Predefined roles are designed for Google Cloud products and services, like App Engine and BigQuery. For a custom application, you can create sets of privileges that give the user with that role as much permission as needed but not more.
8. D. Users should have only the privileges that are needed to carry out their duties. This is the principle of least privilege. Rotation of duties is another security principle related to having different people perform a task at different times. Defense in depth is the practice of using multiple security controls to protect the same asset. Option B is not a real security principle.
9. A. A resource hierarchy has only one organization, which makes option A correct. You can, however, create multiple folders and projects within a resource hierarchy.
10. B. In option B, the correct answer, the billing account is used to specify payment information and should be used to set up automatic payments. Service accounts are used to grant privileges to a VM and are not related to billing and payments. Resource accounts and credit accounts do not exist.
11. C. Google Cloud offers a free service level for many products, which makes option C the correct answer. You can use these services without having to set up a billing account. Google charges for serverless products, such as Cloud Functions and App Engine, when customers exceed the amount allowed under the free tier.
12. C. The correct answer is C. Budgeting and Alerting allows you to specify a budget. When specified percentages of that budget is spent, alerts can be generated. Cloud Monitoring is an observability service for application and infrastructure performance, not billing. Cloud Logging is an observability service for collecting information about events in services and infrastructure. Policy Constraints are a mechanism for restricting how resources can be used.
13. D. Large enterprises should use invoicing when incurring large charges, which makes option D the right answer. A self-service account is appropriate only for amounts that are within the credit limits of credit cards. Since the subdivisions are independently managed and have their own budgets, each should have its own billing accounts.
14. A. When a user is granted `iam.serviceAccountUser` at the project level, that user can manage all service accounts in the project, so option A is correct. If a new service account is created, they will automatically have privilege to manage that service account. You could grant `iam.serviceAccountUser` to the administrator at the service account level, but that would require setting the role for all service accounts. If a new service account is created, the application administrator would have to grant `iam.serviceAccountUser` to the other administrator on the new service account. `iam.serviceProjectAccountUser` is a fictional role.

15. C. When a service account is created, Google generates encryption keys for authentication, making option C correct. Usernames and passwords are not an option for service accounts. Two-factor authentication is an authentication practice that requires two forms of authentication, such as a username password pair and a code from an authentication device. Biometrics cannot be used by services and is not an option.
16. B. Service accounts are resources that are managed by administrators, but they also function as identities that can be assigned roles, which makes option B the correct answer. Billing accounts are not related to identities. Projects are not identities; they cannot take on roles. Roles are resources but not identities. They can take on privileges, but those privileges are used only when they are attached to an identity.
17. B. Predefined roles are defined for a particular product, such as Cloud Run or Compute Engine, so option B is the right answer. They bundle privileges often needed together when managing or using a service. Basic roles are building blocks for other roles. Custom roles are created by users to meet their particular needs; the Application role is a fictitious role.
18. B. By default all users in an organization can create projects, which makes option B correct. The role `resourcemanager.projects.create` allows users to create projects. The billing account is not associated with creating projects.
19. D. The maximum number of organizations is determined on a per-account basis by Google, so option D is the correct answer. If you need additional organizations, you can contact Google and ask for an increase in your limit.
20. B. Users with the Organization Administrator role are not necessarily responsible for determining what permissions should be assigned to users. That is determined based on the person's role in the organization and the security policies established within the organization, which makes option B correct.

Chapter 4: Introduction to Computing in Google Cloud

1. B. The App Engine standard environment can run Python applications, which can autoscale down to no instances when there is no load and thereby minimize costs. Compute Engine and the App Engine flexible environment both require more configuration management than the App Engine standard environment. Kubernetes Engine is used when a cluster of servers is needed to support large or multiple applications using the same computing resources.
2. A. Database servers require high availability to respond to queries from users or applications. Preemptible machines are certain to shut down in at most 24 hours unless they are spot VMs. A batch processing job with no fixed time requirements could use preemptible machines as long as the VM is restarted. High-performance computing clusters can use preemptible machines because work on a preemptible machine can be automatically rescheduled for another node on the cluster when a server is preempted. Option D is incorrect because there is a correct answer in the set of options.

3. A. VMs are created in projects, which are part of the resource hierarchy. They are also located in geographic regions and data centers, so a zone is specified as well. Usernames and admin roles are not specified during creation. The billing account is tied to a project and so does not have to be specified when the VM is created. Cloud storage buckets are created independently of VMs. Not all VMs will make use of storage buckets.
4. C. Compute Engine can run Docker containers if you install Docker on the VM. Kubernetes and the App Engine flexible environment support Docker containers. The App Engine standard environment provides language-specific runtime environments and does not allow customers to specify custom Docker images for use.
5. B. The name of the file that is used to build and configure a Docker container is `Dockerfile`.
6. D. Anthos is a managed service for administering Kubernetes clusters in Google Cloud, other clouds, and on-premises. App Engine Flexible and Cloud Functions are not managed by Anthos.
7. B. Kubernetes provides load balancing, scaling, and automatic upgrading of software. It does not provide vulnerability scanning. Google Cloud's Web Security Scanner service and the Container Analysis service can detect vulnerabilities, but they are separate from Kubernetes Engine.
8. D. The scenario described is a good fit for Kubernetes. Each of the groups of services can be structured in pods and deployed using Kubernetes deployment. Kubernetes Engine manages node health, load balancing, and scaling. App Engine Standard Edition has language-specific sandboxes and is not a good fit for this use case. Cloud Functions is designed for short-running event processing and is not the kind of continuous processing needed in this scenario. Compute Engine could meet the requirements of this use case, but it would require more effort on the part of application administrators and DevOps professionals to configure load balancers, monitor health, and manage software deployments.
9. B. This is an ideal use case for Cloud Functions. The cloud function is triggered by a file upload event. The cloud function calls the image processing service. With this setup, the two services are independent. No additional servers are required. Option A violates the requirement to keep the services independent. Options C and D incur more management overhead and will probably cost more to operate than option B.
10. D. Each invocation of a cloud function runs in a secure, isolated runtime environment. There is no need to check whether other invocations are running. With the Cloud Functions service, there is no way for a developer to control code execution at the process or thread level.
11. A. You would create a custom image after you installed the custom code, in this case the encryption library. A public image does not contain custom code, but it could be used as the base that you add custom code to. Both CentOS and Ubuntu are Linux distributions. You could use either as the base image that you add custom code to, but on their own, they do not have custom code.

- 12. B. Projects are the lowest level of the resource hierarchy. The organization is at the top of the hierarchy, and folders are between the organization and projects. VM instances are not part of the resource hierarchy.
- 13. D. All Google regions have the same level of service level agreement, so reliability is the same. Costs may differ between regions. Regulations may require that data stay within a geographic area, such as the European Union. Latency is a consideration when you want a region that is close to end users or data you will need is already stored in a particular region.
- 14. B. The Compute Engine Admin role gives users complete control over instances. Options A and C are fictitious roles. Compute Engine Security Admin gives users the privileges to create, modify, and delete SSL certificates and firewall rules.
- 15. D. Preemptible VMs will be terminated after 24 hours with the exception of spot VMs. Google does not guarantee that preemptible VMs will be available. Once an instance is started as a preemptible machine, it cannot migrate to a regular VM. You could, however, save a snapshot and use that to create a new regular instance.
- 16. B. The application maintains state and therefore cannot run in Cloud Run. Cloud Run is a managed service for running applications in containers, including Docker-based containers. Containers can run applications written in a variety of languages.
- 17. C. The C programming language is not supported in the App Engine standard environment. If you need to run a C application, it can be compiled and run in a container running in the App Engine flexible environment.
- 18. B. Anthos Service Mesh is a managed service that enables consistent security and monitoring services in Kubernetes clusters. Cloud Functions is used for event processing. App Engine Standard and App Engine Flexible are services for running containerized applications.
- 19. B. vTPM verifies the boot integrity of Compute Engine instances and is used to prevent rootkits and other malicious software from compromising the operating system. Customer-supplied encryption keys are used to encrypt data at rest. Sole tenancy limits which instances can run on a server but does not validate boot integrity. Identity and access management is used to assign roles and permissions to control access to resources in Google Cloud.
- 20. A. Cloud Functions is best suited for event-driven processing, such as a file being uploaded to Cloud Storage or an event being written to a Pub/Sub queue. Long-running jobs, such as loading data into a data warehouse, are better suited to Compute Engine or App Engine.

Chapter 5: Computing with Compute Engine Virtual Machines

- 1. C. You should verify the project selected because all operations you perform will apply to resources in the selected project, making option C the correct answer. You do not need to open Cloud Shell unless you want to work with the command line, and if you did, you should verify that the project is correctly selected first. Logging into a VM using SSH is one of the tasks that requires you to be working with the correct project to see the VMs

associated with that project, so logging in via SSH should not happen before verifying the project. The list of VMs in the VM Instance window is a list of VMs in the current project. You should verify which project you are using to ensure you are viewing the set of VMs you think you are using.

2. A. You will need to set up billing if it is not already enabled when you start using the console, so option A is the right answer. You may create a project, but you will be able to do this only if billing is enabled. You do not need to create a storage bucket to work with the console. Specifying a default zone is not a one-time task; you may change zones throughout the life of your project.
3. B. The name of the VM, the region and zone, and the machine type can all be specified in the console along with other parameters, so option B is correct. Option A is missing required parameters. A CIDR block is a range of IP addresses that is associated with a subnet and is not needed to create a VM. An IP address is assigned automatically so it is not required.
4. B. Different zones may have different machine types available, so you will need to specify a region first and then a zone to determine the set of machine types available. If the machine type does not appear in the list, it is not available in that zone. This makes option B the correct answer. Options A and C are incorrect. Subnets and IP addresses are not related to the machine types available. Unless you are specifying a custom machine type, you do not specify the amount of memory; that is defined by the machine type, so option D is incorrect.
5. C. Labels and descriptions help you track your own attributes of resources. As the number of servers grows, it can become difficult to track which VMs are used for which applications and services, so option C is the correct answer. Labels and a general description will help administrators track numbers of VMs and their related costs. Options A and B are used for security and storage but do not help with managing multiple VMs. Option D is only partially correct. Descriptions are helpful but so are labels.
6. A. The Availability Policy section within the Management tab is where you set the pre-emptible option, so option A is correct. Identity And API Access is used to control the VM's access to Google Cloud APIs and which service account is used with the VM. Sole Tenancy is used if you need to run your VMs on physical servers that only run your VMs. Networking is used to set network tags and change the network interface.
7. B. Shielded VM is an advanced set of security controls that includes Integrity Monitoring, a check to ensure boot images have not been tampered with, which makes option B the right answer. Firewalls are used to control ingress and egress of network traffic to a server or subnet. Project-wide SSH keys are used for authenticating users across servers within a project. Boot disk integrity control service is a fictional feature.
8. C. Block size is not an option in under Additional Disks, so option C is correct. Encryption key management, disk type, and the option of specifying a source image are all available options.
9. B. Using version-controlled scripts is the best approach of the four options. Scripts can be documented with reasons for the changes, and they can be run repeatedly on different machines to implement the same change. This reduces the chance of error when manually entering a command. Option A does not help to improve documenting why changes

were made. Option C could help improve documentation, but executable scripts are precise and accurate reflections of what was executed. Notes may miss details. Option D is not advisable. You could become a bottleneck to making changes, changes cannot be made when you are unavailable, and your memory may not be a reliable way to track all configuration changes.

10. A. `gcloud compute instances` is the start of commands for administering Compute Engine resources, making option A the right answer. Option B, `gcloud instances`, is missing the `compute` keyword that indicates we are working with Compute Engine. Option C has switched the order of `compute` and `instances`. Option D is false because option A is the correct answer.
11. B. Option B follows the pattern of the `gcloud` command, which is hierarchical and starts with the `gcloud` name of the service, in this case `compute` for Compute Engine, followed by the next level down, which in this case is `instances`. Finally, there is the action or verb, in this case `list`. Option A is missing the term `instances` to indicate you are working with VM instances. Option C is missing the `compute` keyword to indicate you are working with Compute Engine. Option D is missing the `compute instance` keyword and has switched the order of `instances` and `list`.
12. B. The correct format is to use the `--labels` parameter and specify the key followed by an equal sign followed by the value in option B. Options A and C have the wrong character separating the key and value. Option D is incorrect because it is possible to specify labels in the command line.
13. C. The two operations you can specify when using the boot disk configuration are adding a new disk and attaching an existing disk, so option C is correct. Reformatting an existing disk is not an option, so options A, B, and D cannot be the correct answer.
14. B. 10 GB of data is small enough to store on a single disk. By creating an image of a disk with the data stored on it, you can specify that source image when creating a VM. Option A would require the data scientist to copy the data from Cloud Storage to a disk on the VM. Option C would similarly require copying the data. Option D would load data into a database, not a filesystem as specified in the requirements.
15. B. On the Networking tab of the VM form, you can add another network interface, so option B is correct. GCP sets the IP address, so option A is incorrect. There is no option to specify a router or change firewall rules on the Networking tab, so options C and D are incorrect.
16. A. The correct option is `boot-disk-type`, which is option A. The other three options are not parameters to the `gcloud compute instances` command.
17. A. Option A is the correct command. It is the only option that includes a correct machine type and properly specifies the name of the instance. Option B uses the `--cpus` parameter, which does not exist. Option C uses the parameter `instance-name`, which does not exist. The instance name is passed as an argument and does not need a parameter name. Option D is incorrect because `machine type n1-4-cpu` is not a valid machine type.

18. C. Option C is the correct command, which is `gcloud compute instances`, to indicate you are working with VMs, followed by the `stop` command and the name of the VM. Option A is incorrect because `halt` is not an option. Option B is incorrect because `terminate` is not a parameter. Option D is missing the word `instances`, which indicates you are working with VMs.
19. B. SSH is a service for connecting to a remote server and logging into a terminal window. Once logged in, you would have access to a command line, so option B is the right answer. FTP is a file transfer protocol and does not allow you to log in and perform system administration tasks. RDP is a protocol used to remotely access Windows servers, not Ubuntu, which is a Linux distribution. `ipconfig` is a command-line utility for configuring IP stacks on a device and does not allow you to log into a remote server.
20. A. All of the statements in option A are true and relevant to billing and costs. Option B is correct that VMs are billed in 1-second increments, but the only preemptible VMs are shut down within 24 hours of starting. Option C is incorrect because discounts are not limited to some regions. Option D is incorrect because VMs are not charged for a minimum of 1 hour.

Chapter 6: Managing Virtual Machines

1. A. The Compute Engine page is where you have the option of creating a single VM instance, so option A is the correct answer. App Engine is used for containers and running applications in language-specific runtime environments. Kubernetes Engine is used to create and manage Kubernetes clusters. Cloud Functions is where you would create a function to run in Google's serverless cloud function environment.
2. B. Instances can be stopped, and when they are, then you cannot connect to them via SSH, which makes option B the correct answer. Starting the instance will enable SSH access. Option A is not correct because you can log into preemptible machines. Option C is incorrect because there is no No SSH option. Option D is incorrect because the SSH option can be disabled.
3. B. The Reset command can be used to restart a VM; thus, option B is correct. The properties of the VM will not change, but data in memory will be lost. There is no Reboot, Restart, Shutdown, or Startup option in the console.
4. C. Labels, deletion protection, and status are all available for filtering, so option C is the correct answer. You can also filter by internal IP, external IP, zone, network, deletion protection, and member of a managed or unmanaged instance group.
5. A. To function properly, the operating system must have GPU libraries installed, so option A is correct. The operating system does not have to be Ubuntu-based, and there is no need to have at least eight CPUs in an instance before you can attach and use a GPU. Available disk space does not determine whether or not a GPU is used.

6. A. If you add a GPU to a VM, you must have compatible CPUs and GPUs. The instance does not need to be preemptible and it can have nonboot disks attached. The instance is not required to run Ubuntu 18.02 or later.
7. B. When you first create a snapshot, Google Cloud will make a full copy of the data on the persistent disk. The next time you create a snapshot from that disk, Google Cloud will only copy the data that has changed since the last snapshot. Option A is incorrect; Google Cloud does not store a full copy for the second snapshot. Option C is incorrect; the first snapshot is not deleted automatically. Option D is incorrect; subsequent snapshots do not incur 10 percent overhead.
8. D. To work with snapshots, a user must be assigned the Compute Storage Admin role, which makes option D the correct answer. The other options are fictitious roles.
9. C. Images can be created from disks, snapshots, cloud storage files, a virtual disk, or another image, so option C is the right answer. Database export files are not sources for images.
10. B. Deprecated marks the image as no longer supported and allows you to specify a replacement image to use going forward, making option B the correct answer. Deprecated images are available for use but may not be patched for security flaws or have other updates. The other options are fictitious features of images.
11. C. The base command for working with instances is `gcloud compute instances`, which makes option C the correct answer. The `list` command is used to show details of all instances. By default, output is in human-readable form, not `json`. Using the `--format json` option forces the output to be in JSON format. `--output` is not a valid option.
12. B. `--async` causes information about the start process to be displayed; therefore, option B is correct. `--verbose` is an analogous parameter in many Linux commands. `--describe` provides details about an instance but not necessarily the startup process. `--details` is not a valid parameter.
13. C. The command to delete an instance is `gcloud compute instances delete` followed by the name of the instance, so option C is correct. Option A is incorrect because there is no `instance` parameter. Option B is incorrect because that command stops but does not delete the instance. Option D is missing `instances` in the command, which is required to indicate what type of entity is being deleted.
14. A. `gcloud compute instances` is the base command followed by `delete`, the name of the instance, and `--keep-disks=boot`, so option A is correct. There is no `--save-disk` parameter. Option C is wrong because `filesystem` is not a valid value for the `keep-disk` parameter. Option D is missing the `instances` option, which is required in the command.
15. B. The correct answer is option B, which is to use the `describe` command. Option A will show some fields but not all. Options C and D are incorrect because there is no `detailed` parameter.
16. B. Instance groups are sets of VMs that can be configured to scale and are used with load balancers, which contribute to improving availability, so option B is correct. Preemptible instances are not highly available because they can be shut down at any time by Google Cloud. Cloud Storage is not a Compute Engine component. GPUs can help improve throughput for math-intensive operations but do not contribute to high availability.

17. B. An instance template is used to specify how the instance group should be created, which makes option B the correct answer. Option A is incorrect because instances are created automatically when an instance group is created. Boot disk images and snapshots do not have to be created before creating an instance group.
18. B. The command to delete an instance group is `gcloud compute instance-template delete`, so option B is correct. Option A incorrectly includes the term `instances`. Option C is in incorrect order. Option D is wrong because `instance-template` is in the wrong position and is plural in the option.
19. C. You can configure an autoscaling policy to trigger adding or removing instances based on CPU utilization, monitoring metric, load balancing capacity, or queue-based workloads. Disk, network latency, and memory can trigger scaling if monitoring metrics on those resources are configured. So, option C is correct.
20. B. Unmanaged instance groups are available for limited use cases such as this. Unmanaged instance groups are not recommended in general. Managed instance groups are the recommended way to use instance groups, but the two different configurations prevent their use. Preemptible instances and GPUs are not relevant to this scenario.

Chapter 7: Computing with Kubernetes

1. C. Kubernetes creates instance groups as part of the process of creating a cluster, which makes option C the correct answer. Cloud Monitoring and Cloud Logging, not instance groups, is used to monitor the health of nodes and to create alerts and notifications. Kubernetes creates pods and deployments; they are not provided by instance groups.
2. A. A Kubernetes cluster has a single control plane and one or more nodes to execute workloads, so option A is the correct answer. There is no monitoring node in Kubernetes, but it does generate metrics that can be sent to Cloud Monitoring. Kubernetes does not require instances with at least six vCPUs.
3. C. Pods are single instances of a running application in a cluster, so option C is correct. Pods run containers but are not simply sets of containers. Application code runs in containers that are deployed in pods. Pods are not controllers, so they cannot manage communication with clients and Kubernetes services.
4. B. Services are Kubernetes components providing API endpoints that allow applications to discover pods running a particular application, making option B correct. Options A and C, if they could be coded using the API designed for managing clusters, would require more code than working with services and are subject to changes in a larger set of API functions. Option D is not an actual option.
5. A. A deployment config specifies how many nodes to create in a ReplicaSet. Cloud Operations Suite is a monitoring and logging service that monitors but does not control Kubernetes clusters. Container Runtime is a component of Kubernetes that is responsible for running containers. Jobs is an abstraction of workloads and is not tied to the number of pods running in a cluster.

6. B. Regional clusters are available in Kubernetes Engine and are used to provide resiliency to an application, so option B is correct. Option A refers to instance groups that are a feature of Compute Engine, not directly of Kubernetes Engine. Option C is incorrect; regional deployments is a fictitious term. Load balancing distributes load and is part of Kubernetes by default. If load is not distributed across zones or regions, it does not help to add resiliency across data centers.
7. A. Option A is the best answer. Starting with an existing template, filling in parameters, and generating the `gcloud` command is the most reliable way. Option D may work, but multiple parameters that are needed for your configuration may not be in the script you start with. There may be some trial and error with this option. Options B and C may lead to a solution but could take some time to complete.
8. A. The correct command is option A. Option B has `size` instead of `num-nodes`. Option C has `region-nodes` instead of `num-nodes`. Option D is missing the `--num-nodes` parameter name.
9. C. Time to Live is not an attribute of deployments, so option C is the correct answer. Application name, container image, and initial command can all be specified.
10. B. Deployment configuration files created in Cloud Console are saved in YAML format. CSV, TSV, and JSON are not used.
11. C. The `kubectl` command is used to control workloads on a Kubernetes cluster once it is created, so option C is correct. Options A and B are incorrect because `gcloud` is not used to manipulate Kubernetes processes. Option D is wrong because `container` is not required in `kubectl` commands.
12. C. Option C is the correct command. Option A uses the term `upgrade` instead of `scale`. Option B incorrectly uses `gcloud`. Option D uses the incorrect parameter pods.
13. D. Cloud Operations Suite is a comprehensive monitoring, logging, alerting, and notification service that can be used to monitor Kubernetes clusters.
14. D. GKE sends metrics and logs to Cloud Monitoring and Cloud Logging by default, so you do not need to do anything other than accept the default configuration for monitoring and logging. There are no `--monitoring=True` and `--logging=True` parameters. Node pools are used to group nodes with similar configurations and not required for monitoring and logging. Namespaces are used to logically separate workloads on clusters and do not need to be individually configured to enable monitoring and logging.
15. A. Prometheus is a popular open source monitoring tool available as a managed service in Google Cloud. Apache Flink is a stream and batch processing platform similar to Cloud Dataflow. MongoDB is a NoSQL database that uses a document storage model. Spark is a data analysis tool that is available as a managed service in Google Cloud, but it is not a monitoring tool.
16. B. Autopilot mode clusters require the least configuration and infrastructure management, so B is the correct answer. Standard mode clusters require you to specify infrastructure and configuration options. Options C and D are fictitious cluster modes.

17. A. Standard mode clusters require you to make configuration and infrastructure choices, so A is the correct answer. Autopilot mode clusters use preconfigured optimized infrastructure and do not give you as much control over configuration and infrastructure as standard mode does. Options C and D are fictitious cluster modes.
18. B. B is the correct answer because with a static channel configuration, GKE will not automatically upgrade the cluster. Option A is the correct choice if you want automatic upgrades. Node pools and ReplicaSets are not related to upgrading configurations.
19. A. All interactions with the cluster are done through the master using the Kubernetes API. If an action is to be taken on a node, the command is issued by the control plane, so option A is the correct answer. Options B and D are incorrect because they are controllers within the cluster and do not impact how commands are received from client devices. Option C is incorrect because `kubectl`, not `gcloud`, is used to initiate deployments.
20. A. Services provide a level of indirection to accessing pods. Pods are ephemeral. Clients connect to services, which can discover pods. ReplicaSets and StatefulSets provide managed pods. Alerts are for reporting on the state of resources.

Chapter 8: Managing Standard Mode Kubernetes Clusters

1. B. When on the Cloud Console pages, you can click the cluster name to see a Details page, so option B is the correct answer. Typing the name of a cluster in the search bar does not always return cluster details; it can return instance group details. There is no such command as `gcloud cluster details`.
2. A. You can find the number of vCPUs on the cluster listing in the Node Pools section of the Nodes Details page. The other sections do not have vCPU details.
3. B. The correct command includes `gcloud container` to describe the service, `clusters` to indicate the resource you are referring to, and `list` to indicate the command, which makes option B the correct answer. Options A and C are not valid commands.
4. B. It is likely you do not have access privileges to the cluster. The `gcloud container clusters get-credentials` command is the correct command to configure `kubectl` to use Google Cloud credentials for the cluster, so option B is the right option. Options A, C, and D are invalid commands.
5. C. Clicking the Edit button allows you to change, add, or remove labels, so option C is the correct answer. The Connect button is on the cluster listing page, and the Deploy button is for creating new deployments. There is no way to enter labels under the Labels section when displaying details.

6. D. When resizing, the `gcloud container clusters resize` command requires the name of the cluster and the node pool to modify. The size is required to specify how many nodes should be running. Therefore, option D is correct.
7. B. Pods are used to implement replicas of a deployment. It is a best practice to modify the deployments, which are configured with a specification of the number of replicas that should always run, so option B is the correct answer. Option A is incorrect; you should not modify pods directly. Options C and D are incorrect because they do not change the number of pods running an application.
8. C. Deployments are listed under Workloads, making option C the correct answer. The Cluster option shows details about clusters but does not have details on deployments. Storage shows information about persistent volumes and storage classes. Deployments is not an option.
9. B. There are four actions available for deployments (Autoscale, Expose, Rolling Update, and Scale), so option B is correct. Add, Modify, and Delete are not options.
10. C. Since deployments are managed by Kubernetes and not Google Cloud, we need to use a `kubectl` command and not a `gcloud` command, which makes option C correct. Option D is incorrect because it follows the `gcloud` command structure, not the `kubectl` command structure. The `kubectl` command has the verb, like `get`, before the resource type, like `deployments`, for example.
11. D. You can specify container image, cluster name, and application name along with the labels, initial command, and namespace; therefore, option D is the correct answer.
12. A. The Deployment Details page includes applications, so option A is the correct answer. Containers are used to implement services; service details are not available there. The Clusters Detail page does not contain information on services running in the cluster.
13. A. `kubectl run` is the command used to start a deployment. It takes a name for the deployment, an image, and a port specification. The other options are not valid `kubectl` commands.
14. A. Option A shows the correct command, which is `kubectl delete service ml-classifier-3`. Option B is missing the service term. Options C and D cannot be correct because services are managed by Kubernetes, not Google Cloud.
15. C. The Container Registry is the service for managing images that can be used in other services, including Kubernetes Engine and Compute Engine, making option C correct. Both Compute Engine and Kubernetes Engine use images but do not manage them. There is no service called Container Engine.
16. A. Images are managed by Google Cloud, so the correct command will be a `gcloud` command, making option A the correct answer. Option B is incorrect because the verb is placed before the resource. Options C and D are incorrect because `kubectl` is for managing Kubernetes resources, not Google Cloud resources like container images.

17. B. The correct command is `gcloud container images describe`, which makes option B the right answer. `describe` is the `gcloud` verb or operation for showing the details of an object. All other options are invalid commands.
18. B. The `kubectl expose deployment` command makes a service accessible, so option B is the correct answer. IP addresses are assigned to VMs, not services. The command `gcloud` does not manage Kubernetes services, so option C is incorrect. Option D is incorrect because making a service accessible is not a cluster-level task.
19. B. Autoscaling is the most cost-effective and least burdensome way to respond to changes in demand for a service, so option B is the correct answer. Option A may run nodes even when they are not needed. Option C is manually intensive and requires human intervention. Option D reduces human intervention but does not account for unexpected spikes or lulls in demand.
20. B. Cloud engineers working with Kubernetes will need to be familiar with working with clusters, nodes, pods, and container images. They will also need to be familiar with deployment. Option B is the correct answer because the other options are all missing an important component of Kubernetes that cloud engineers will have to manage.

Chapter 9: Computing with Cloud Run and App Engine

1. C. The correct answer is C. Cloud Run services are managed, serverless services for running containers and are designed to support containers that run continuously, which is what is needed for an API service. Compute Engine requires that you manage servers, so options A and B are incorrect. Option D is incorrect because Cloud Run jobs are for containers that perform a task and then terminate.
2. D. The correct answer is D. Cloud Run jobs are managed, serverless services for running containers and are designed to support executables that run until a task is completed. Kubernetes Engine is used to run containers but is best suited to running large numbers of containers in complicated environments, such as environments that need to support multiple namespaces. Compute Engine could be used but requires more administration than using Cloud Run. App Engine Flexible could be used to run a container, but Cloud Run is preferred over App Engine.
3. B. The correct answer is B; array jobs in Cloud Run jobs allow multiple containers to run and process the workload in parallel. Since the data in each file is independent of the data in other files, they can be processed in any order and in parallel. Option A is incorrect because the data is publicly available and there is no need for customer-managed encryption keys. Option C is incorrect because there is no mention of a need to write data to a Cloud SQL database. Option D is incorrect; a private vs. public IP address is not relevant to the question.

4. C. The correct answer is C; by supporting session affinity in the Connection configuration, Cloud Run will route all requests from a client to the same container if possible. Option A is incorrect; Cloud SQL Connection is used to connect a Cloud Run service to Cloud SQL. Option B is incorrect; array jobs in Cloud Run jobs allow multiple containers to run and process the workload in parallel. Option D is incorrect; a private vs. public IP address is not relevant to the question.
5. A. The correct answer is A; an internal ingress setting will restrict network traffic to internal Google Cloud traffic. Option B is incorrect; that would allow traffic entering through external load balancing. Option C is incorrect; that would allow all traffic. Option D is incorrect; there is no such setting for PII Proxy Traffic in Cloud Run.
6. B. The correct answer is B; you specify a service account for a Cloud Run service on the Security tab of the Create Service page in the Cloud Run console. Option A is incorrect; that is used for configuring network connection. Option C is incorrect; that is used for configuring the running container. Option D is incorrect; that is for setting environment variables and referencing secrets.
7. A. The correct answer is A; you would grant a role with appropriate permissions to a group that includes the developers who need access. Option B is incorrect; Cloud Identity Aware Proxy ensures users are authenticated and authorized but does not grant permissions. Option C is incorrect; the ingress policy controls network traffic, not users. Option D is incorrect; the Security tab does not grant access to users.
8. C. The correct answer is C; a VPC Connection enables the use of Serverless VPC Access to connect your Cloud Run service to other resources in your VPC. Option A is incorrect; that is used for connecting to a Cloud SQL database. Option B is incorrect; IAP Proxy is used to authenticate and authorize using fine-grained access controls. Option D is incorrect; session affinity is used to send all requests from a client to the same container.
9. D. The correct answer is D; configuring the service to use HTTP/2 end-to-end will allow for the use of gRPC protocol. Option A is incorrect; support for external load balancing traffic is configured using the ingress setting and not necessarily required for using gRPC. Option B is incorrect; IAP Proxy is used to authenticate and authorize using fine-grained access controls. Option C is incorrect; session affinity is used to send all requests from a client to the same container.
10. B. The correct answer is B; both Container Registry and Artifact Registry can be used to store and make container images accessible to Cloud Run. Options A and C are both missing a valid option for serving container images. Option D is incorrect; Kubernetes is used for container orchestration but does not provide container image registry services for Cloud Run.
11. B. Versions support migration. An app can have multiple versions, and by deploying with the `--migrate` parameter, you can migrate traffic to the new version, so option B is the correct answer. Services are a higher-level abstraction and represent the functionality of a microservice. An app may have multiple services, but they serve different purposes. Instances execute

code in a version. Instances may be added and removed as needed, but they will run only one version of a service. Instance groups are part of Compute Engine and are not an App Engine component.

12. A. Autoscaling enables setting a maximum and minimum number of instances, which makes option A correct. Basic scaling does not support maximum and minimum instances. Option C is not recommended because it is difficult to predict when load will peak, and even if the schedule is predictable today, it may change over time. Option D is wrong; there is no instance detection option.
13. B. The correct command is `gcloud app deploy`, which is option B. Options A and C are incorrect because `gcloud components` commands are used to install `gcloud` commands for working with parts of App Engine, such as the Python runtime environment. Option D is incorrect; you do not need to specify an instance in the command.
14. B. The `app.yaml` file is used to configure an App Engine application, which makes option B correct. The other options are not files used to configure App Engine.
15. A. `max_concurrent_requests` lets you specify the maximum number of concurrent requests before another instance is started, which makes option A correct. `target_throughput_utilization` functions similarly but uses a 0.05 to 0.95 scale to specify maximum throughput utilization. `max_instances` specifies the maximum number of instances but not the criteria for adding instances. `max_pending_latency` is based on the time a request waits, not the number of requests.
16. C. Basic scaling only allows for idle time and maximum instances, so option C is the right answer. `min_instances` is not supported. `target_throughput_utilization` is an autoscaling parameter, not a basic scaling parameter.
17. C. The `runtime` parameter specifies the language environment to execute in, which makes option C correct. The script to execute is specified by the `script` parameter. The URL to access the application is based on the project name and the domain `appspot.com`. There is no parameter for specifying the maximum time an application can run.
18. A. Using dynamic instances by specifying autoscaling or basic scaling will automatically adjust the number of instances in use based on load, so option A is correct. Option B is incorrect because autoscaling and basic scaling only create dynamic instances. Options C and D are incorrect because manual scaling will not adjust instances automatically, so you may continue to run more instances than needed at some points.
19. B. `--split-by` is the parameter used to specify the method for splitting traffic. Valid options are `cookie`, `ip`, and `random`. All other options are not valid parameters to the `gcloud app services set-traffic` command.
20. D. All three methods listed, IP address, HTTP cookie, and random splitting, are allowed methods for splitting traffic.

Chapter 10: Computing with Cloud Functions

1. C. Cloud Run is a serverless service for running containerized applications that run continuously and provide an endpoint, making option C the correct answer. This is unlike Cloud Functions, which is designed to support single-purpose functions that operate independently and in response to isolated events in the Google Cloud and complete within a specified period of time. Compute Engine is not a serverless option. Cloud Storage is not a computing product.
2. C. A timeout period that is too low would explain why the smaller files are processed in time but the largest are not, which makes option C the right answer. If only 10 percent of the files are failing, then it is not a syntax error or the wrong runtime selected, as in options A and B. Those errors would affect all files, not just the largest ones. Similarly, if there was a permission problem with the Cloud Storage bucket, it would affect all files.
3. B. Those actions are known as events in Google Cloud terminology; thus, option B is the correct answer. An incident may be a security or performance-related occurrence, but those are unrelated to the expected and standardized actions that constitute events. A trigger is a declaration that a certain function should execute when an event occurs. A log entry is related to applications recording data about significant events. Log entries are helpful for monitoring and compliance, but in themselves are not event-related actions.
4. C. The correct answer is option C because SSL is a secure protocol for remotely accessing servers. It is used, for example, to access instances in Compute Engine. It does not have events that can be triggered using Cloud Functions. The three GCP products listed do generate events that can have triggers associated with them.
5. D. The correct answer is D; all other options are missing two or more supported environments. The supported runtime environments include Node.js, Python, Go, Java, .NET, Ruby, and PHP.
6. C. HTTP requests using GET, POST, DELETE, PUT, and OPTIONS can invoke an HTTP trigger in Cloud Functions, so option C is the right answer.
7. D. The correct answer, option D, shows the four events supported in Cloud Storage:

```
google.storage.object.finalize  
google.storage.object.delete  
google.storage.object.archive  
google.storage.object.metadataUpdate
```
8. C. There is no option to specify the file type to apply the function to, so option C is correct. You can, however, specify the bucket to which the function is applied. You could only save files or the types you want processed in that bucket, or you could have your function check file type and then execute the rest of the function or not, based on type. All the other options listed are parameters to a Cloud Storage function.

9. D. Second-Generation Cloud Functions can have between 128 MB and 16 GB of memory allocated, which makes option D the correct answer.
10. B. By default, Cloud Functions can run for up to 1 minute before timing out, so option B is correct. You can, however, set the `timeout` parameter for a Cloud Function for periods of up to 9 minutes before timing out.
11. A. Option A will install standard `gcloud` commands. Options B, C, and D are not valid `gcloud` commands.
12. A. The correct trigger in option A is `google.storage.object.finalize`, which occurs after a file is uploaded. Option B is not a valid trigger name. Option C triggers when a file is archived, not uploaded. Option D is triggered when some metadata attribute changes, but not necessarily only after a file uploads.
13. C. The three parameters are `runtime`, `trigger-resource`, and `trigger-event`, as listed in option C. All must be set, so options A and B are incorrect. `file-type` is not a parameter to creating a Cloud Function on Cloud Storage, so option D is incorrect.
14. A. The correct answer is option A, `gcloud functions delete`. Option B references components, which is incorrect. You do need to reference components when installing or updating `gcloud` commands but not when deleting a cloud function, so options B and C are incorrect. Option D is incorrect because the Google Cloud entity type, in this case `functions`, comes before the name of the operation, in this case `delete`, in a `gcloud` command.
15. B. Messages are stored in a text format, base64, so that binary data can be stored in the message in a text format, so option B is correct. Option A is incorrect; it is needed to map from a binary encoding to a standard text encoding. Option C is incorrect because the function does not pad with extra characters to make them the same length. Option D is incorrect; it does not change dictionary data types into list data types.
16. C. Option C is correct because it includes the name of the function, the runtime environment, and the name of the Pub/Sub topic. Option A is incorrect because it's missing both the runtime and the topic. Option B is incorrect because it is missing the topic. Option D is incorrect because the runtime specification is incorrect; you have to specify `python37` and not `python` as the runtime.
17. B. There is only one type of event that is triggered in Cloud Pub/Sub, and that is when a message is published, which is option B. Option A is incorrect; Cloud Pub/Sub has one event type that can have a trigger. Option C is incorrect; Cloud Pub/Sub does not analyze the code to determine when it should be run. Option D is incorrect; you do not have to specify an event type with Cloud Pub/Sub functions.
18. B. The correct answer is option B because it uses a Cloud Storage `finalize` event to trigger conversion if needed. There is minimal delay between the time the file is uploaded and when it is converted. Option A is a possibility but would require more coding than option B. Option C is not a good option because files are not converted until the batch job runs. Option D is incorrect because you cannot create a Cloud Function for Cloud Pub/Sub using a `finalize` event. That event is for Cloud Storage, not Cloud Pub/Sub.

- 19. D. All of the options are available along with zip from Cloud Storage.
- 20. A. The HTTP trigger allows for the use of POST, GET, and PUT calls, so option A is the correct answer. Webhook and Cloud HTTP are not valid trigger types.

Chapter 11: Planning Storage in the Cloud

- 1. D. An Archive storage class cannot be changed to standard or to any of the other storage classes. All other options are allowed.
- 2. C. The goal is to reduce cost, so you would want to use the least costly storage option. Coldline is designed for objects that are accessed no more than once every 90 days, so option C is correct. Nearline and standard costs more than coldline, so those are not good options. Archive should only be used for objects accessed no more than once per year.
- 3. B. Bigtable is a wide-column database that can ingest large volumes of data consistently, so option B is correct. It also supports low-millisecond latency, making it a good choice for supporting querying. Cloud Spanner is a global relational database that is not suitable for high-speed ingestion of large volumes of data. Firestore is an object data model and not a good fit for IoT or other time series data. BigQuery is an analytics database and not designed for ingestion of large volumes of data with short write latencies.
- 4. A. Option A is correct because Memorystore is a managed cache. The cache can be used to store the results of queries. Follow-on queries that reference the data stored in the cache can read it from the cache, which is much faster than reading from persistent disks. SSDs have significantly lower latency than hard disk drives and should be used for performance-sensitive applications like databases. Options B and D are incorrect because HDD persistent disks do give the best performance with respect to IOPS. Options C and D are incorrect because Firestore is a managed NoSQL database and would not meet the requirement of continuing to use a relational database.
- 5. B. HDDs are the better choice for persistent disks for a local database when performance is not the primary concern and you are trying to keep costs down, so option B is correct. Option A is wrong because SSDs are more expensive and the users do not need the lowest latency available. Options C and D are wrong; both of those are other databases that would not be used to store data in a local relational database.
- 6. B. Lifecycle configurations cannot change the storage class from archive to standard, so option B is the right answer. Option A is true; you can set retention periods when creating a bucket. Option C is true; Cloud Storage does not provide filesystem-like access to internal data blocks. Option D is true because Cloud Storage is highly durable.

7. A. The most recent version of an object is called the live version, so option A is correct. Options B, C, and D are incorrect; top and active are not terms used to refer to versions.
8. B. Both Cloud SQL and Spanner are relational databases and are well suited for transaction-processing applications, so option B is correct. Option A is incorrect because BigQuery is an analytic database designed for data warehousing and analytics, not transaction processing. Options C and D are incorrect because Bigtable is a wide-column NoSQL database, not a relational database.
9. C. Both MySQL and PostgreSQL are Cloud SQL options, so option C is correct. Options A and B are incorrect; Oracle is not a Cloud SQL option. Option D is incorrect because DB2 is not a Cloud SQL option. You could choose to run DB2 or Oracle on your instances but you would have to manage them, unlike Cloud SQL managed databases.
10. D. The multiregional and multi-super-regional location of nam-eur-aisa1 is the most expensive, which makes option D the right answer. Option A is a region that costs less than the multi-super-regional nam-eur-asia1. Option C is incorrect; that is a zone, and Spanner is configured to regions or super regions. Option B is incorrect; it is only a single super region, which costs less than deploying to multiple super regions.
11. D. BigQuery and Firestore are all fully managed services that do not require you to specify configuration information for VMs, which makes option D correct. Cloud SQL and Bigtable require you to specify some configuration information for VMs.
12. B. Firestore is a document database, which makes option B correct. Cloud SQL and Spanner are relational databases. Bigtable is a wide-column database. Google does not offer a managed graph database.
13. A. BigQuery is a managed service designed for data warehouses and analytics. It uses standard SQL for querying, which makes option A the right answer. Bigtable can support the volume of data described, but it does not use SQL as a query language. Cloud SQL is not the best option to scale to tens of petabytes. IBM DB2 is a relational database but it is not a Google Cloud-managed database service.
14. B. Firestore is a document database that has mobile supporting features, like data synchronization, so option B is the right answer. BigQuery is for analytics, not mobile or transactional applications. Spanner is a global relational database but does not have mobile-specific features. Bigtable could be used with mobile devices, but it does not have mobile-specific features like synchronization.
15. D. In addition to read and write patterns, cost, and consistency, you should consider transaction support and latency, which makes option D correct.
16. B. Option B is correct because Memorystore can be configured to use between 1 GB and 300 GB of memory.
17. D. Once a bucket is set to archive, it cannot be changed to another storage class; thus, option D is correct. Standard can change to Nearline, Coldline, or Archive. Nearline can change to Coldline and Archive. Coldline can change to Archive.

18. A. To use BigQuery to store data, you must have a dataset to store it, which makes option A the right answer. Buckets are used by Cloud Storage, not BigQuery. You do not manage persistent disks when using BigQuery. An entity is a data structure in Firestore, not BigQuery.
19. D. With a MySQL database, you can configure the MySQL version, connectivity, machine type, automatic backups, failover replicas, database flags, maintenance windows, and labels, so option D is correct.
20. A. Access charges are used with Nearline and Coldline storage, which makes option A correct. There is no transfer charge involved. Option C is incorrect because egress charges would have applied before the change to Nearline and Coldline. Option D is incorrect because nearline and coldline incur access charges.

Chapter 12: Deploying Storage in Google Cloud

1. C. Creating databases is the responsibility of database administrators or other users of Cloud SQL, so option C is correct. Google applies security patches and performs other maintenance, so option A is incorrect. Google Cloud performs regularly scheduled backups, so option B is incorrect. Database administrators need to schedule backups, but Google Cloud makes sure they are performed on schedule. Cloud SQL users can't use SSH to connect to a Cloud SQL server, so they can't tune the operating system. That's not a problem; Google takes care of that.
2. A. Cloud SQL is controlled using the `gcloud` command; the sequence of terms in `gcloud` commands is `gcloud` followed by the service, in this case `SQL`; followed by a resource, in this case `backups`; and a command or verb, in this case `create`. Option A is the correct answer. Option B is incorrect because `gsutil` is used to work with Cloud Storage, not Cloud SQL. Option C is wrong because the order of terms is incorrect; `backups` comes before `create`. Option D is incorrect because the command or verb should be `create`.
3. A. Option A is the correct answer. The base command is `gcloud sql instances patch`, which is followed by the instance name and a start time passed to the `--backup-start-time` parameter. Option B is incorrect because `databases` is not the correct resource to reference; `instances` is. Option C uses the `cbt` command, which is for use with Bigtable, so it is incorrect. Similarly, Option D is incorrect because it uses the `bq` command, which is used to manage BigQuery resources.
4. C. Datastore mode uses a SQL-like query language called GQL, so option C is correct. Option A is incorrect; SQL is not used with this database. Option B is incorrect; MDX is a query language for online analytic processing (OLAP) systems. Option D is incorrect because DataFrames is a data structure used in Spark.
5. C. Option C is the correct command. It has the correct base command, `gcloud firestore export`, followed by the name of a Cloud Storage bucket to hold the export file. Option A is incorrect because the `collection` parameter name is not required. Option B is incorrect because `dump` is not a valid operation and the term `collection` is not required. Option D is incorrect because it uses the command or verb `dump` instead of `export`.

6. C. Option C is correct; BigQuery displays an estimate of the amount of data scanned. This is important because BigQuery charges for data scanned in queries. Option A is incorrect; knowing how long it took you to enter a query is not helpful. Option B is incorrect; you need to use the scanned data estimate with the Pricing Calculator to get an estimate cost. Option D is incorrect; you do not create clusters in BigQuery as you do with Bigtable and Dataproc. Network I/O data is not displayed.
7. B. Option B shows the correct bq command structure, which includes `location` and the `--dry_run` option. This option calculates an estimate without actually running the query. Options A and C are incorrect because they use the wrong command; `gcloud` and `gsutil` are not used with BigQuery. Option D is also wrong. `cbt` is a tool for working with Bigtable, not BigQuery. Be careful not to confuse the two because their names are similar.
8. A. Option A is correct; the menu option is Personal History or Project History. Options B and C are incorrect; there is no Active Jobs or My Jobs option. Job History shows active jobs, completed jobs, and jobs that generated errors. Option D is incorrect; you can get job status in the console.
9. C. BigQuery provides an estimate of the amount of data scanned, and the Pricing Calculator gives a cost estimate for scanning that volume of data. Options A, B, and D are incorrect; the Billing service tracks charges incurred—it is not used to estimate future or potential charges.
10. B. Option B is correct; the next step is to create a database within the instance. Once a database is created, tables can be created, and data can be loaded into tables. Option A is incorrect; Cloud Spanner is a managed database, so you do not need to apply security patches. Option C is incorrect because you can't create tables without first having created a database. Option D is incorrect; no tables are created that you could import data into when an instance is created.
11. D. Option D is correct because there is no need to apply patches to the underlying compute resources when using Cloud Spanner because Google manages resources used by Cloud Spanner. Updating packages is a good practice when using VMs, for example, with Compute Engine, but it is not necessary with a managed service.
12. C. This use case is well suited to Pub/Sub, so option C is correct. It involves sending messages to the topic, and the subscription model is a good fit. Pub/Sub has a retention period to support the three-day retention period. Option A is incorrect; Bigtable is designed for storing large volumes of data. Dataproc is for processing and analyzing data, not passing it between systems. Cloud Spanner is a global relational database. You could design an application to meet this use case, but it would require substantial development and be costly to run.
13. C. Pub/Sub works with topics, which receive and hold messages, and subscriptions, which make messages available to consuming applications; therefore, option C is correct. Option A is incorrect; tables are data structures in relational databases, not message queues. Similarly, option B is wrong because databases exist in instances of database management systems, not messaging systems. Option D is wrong because tables are not a resource in messaging systems.
14. C. The correct command is `gcloud components install cbt` to install the Bigtable command-line tool, so option C is correct. Options A and B are incorrect; `apt-get` is used to install packages on some Linux systems but is not specific to Google Cloud. Option D is incorrect; there is no such command as `bigtable-tools`.

15. A. You would need to use a `cbt` command, which is the command-line tool for working with Bigtable, so option A is correct. All other options reference `gcloud` and are therefore incorrect.
16. B. Cloud Dataproc is a managed service for Spark and Hadoop, so option B is correct. Cassandra is a big data distributed database but is not offered as a managed service by Google, so options A and C are incorrect. Option D is incorrect because TensorFlow is a deep learning platform not included in Dataproc.
17. B. The correct command is `gcloud dataproc clusters create` followed by the name of the cluster and the `--zone` parameter, so option B is correct. Option A is incorrect because `bq` is the command-line tool for BigQuery, not Dataproc. Option C is a `gcloud` command missing a verb or command, so it does not create a cluster.
18. B. `gsutil` is the correct command, so option B is correct. Option A is incorrect because `gcloud` commands are not used to manage Cloud Storage. Similarly, options C and D are incorrect because `cbt` is used for working with Bigtable and `bq` is used for working with BigQuery.
19. B. The command in option B correctly renames an object from an old name to a new name. Option A is incorrect because it uses a `cp` command instead of `mv`. Option C does not include bucket names, so it is incorrect. Option D uses `gcloud`, but `gsutil` is the command-line tool for working with Cloud Storage.
20. A. Dataproc with Spark and its machine learning library are ideal for this use case, so option A is correct. Option B suggests Hadoop, but it is not a good choice for machine learning applications. Option C is incorrect because Spanner is designed as a global relational database with support for transaction processing systems, not analytic and machine learning systems. Option D is incorrect; SQL is a powerful query language, but it does not support the kinds of machine learning algorithms needed to solve the proposed problem.

Chapter 13: Loading Data into Storage

1. C. `gsutil` is the command-line utility for working with Cloud Storage. Option C is the correct answer because `mb`, short for “make bucket,” is the verb that follows `gsutil` to create a bucket. Option D is wrong because it is not a complete command. Option A is incorrect because `create` and `buckets` are in the wrong order. The command `gcloud storage buckets create` could also be used to create a bucket. Option B is wrong because it uses `gsutil` with a command syntax used by `gcloud`.
2. B. The correct answer is option B; `gsutil` is the command to copy files to Cloud Storage. Option A is incorrect; the verb is `cp`, not `copy`. Option C is incorrect because `gcloud cp` is not a complete command. Option D is not a valid `gcloud storage` command.
3. C. From the console, you can upload both files and folders. Options A and B are incorrect because they are missing an operation that can be performed in the console. Option D is incorrect because there is no `diff` operation in Cloud Console.

4. D. When exporting a database from Cloud SQL, the export file format options are CSV and SQL, which makes option D correct. Option A is incorrect because XML is not an option. Options B and C are incorrect because JSON is not an option.
5. A. Option A, SQL format, exports a database as a series of SQL data definition commands. These commands can be executed in another relational database without having to first create a schema. Option B could be used, but that would require mapping columns to columns in a schema that was created before loading the CSV, and the database administrator would like to avoid that. Options C and D are incorrect because they are not export file format options.
6. C. Option C is the correct command, `gcloud sql export sql`, indicating that the service is Cloud SQL, the operation is export, and the export file format is SQL. The filename and target bucket are correctly formed. Option A is incorrect because it references `gcloud storage`, not `gcloud sql`. Option B is incorrect because it is missing an export file format parameter. Option D is incorrect because the bucket name and filename are in the wrong order.
7. B. Option B is correct because XML is not an option in BigQuery's export process. All other options are available.
8. D. Option D is correct because YAML is not a file storage format; it is used for specifying configuration data. Options A, B, and C are all supported import file types.
9. A. The correct command is `bq load` in option A. The `autodetect` and `source_format` parameters and path to source are correctly specified in all options. Option B is incorrect because it uses the term `import` instead of `load`. Options C and D are incorrect because they use `gcloud` instead of `bq`.
10. B. The correct answer is B because Dataflow is a pipeline service for processing streaming and batch data that implements workflows used by Cloud Spanner. Option A is incorrect; Dataproc is a managed Hadoop and Spark service, which is used for data analysis. Option C is incorrect; Firestore is a NoSQL database. Option D is incorrect because `bq` is used with BigQuery only.
11. A. Bigtable data is exported using a compiled Java program, so option A is correct. Option B is incorrect; there is no `gcloud bigtable` command. Option C is incorrect; `bq` is not used with Bigtable. Option D is incorrect because it does not export data from Bigtable.
12. C. Exporting from Dataproc exports data about the cluster configuration, which makes option C correct. Option A is incorrect; data in DataFrames is not exported. Option B is incorrect; Spark does not have tables for persistently storing data like relational databases. Option D is incorrect; no data from Hadoop is exported.
13. C. The correct answer is option C; the service Dataproc supports Apache Spark, which has libraries for machine learning. Options A and B are incorrect; neither is an analysis or machine learning service. Option D, DataAnalyze, is not an actual service.

14. A. The correct command in option A uses `gcloud` followed by the service, in this case `pubsub`, followed by the resource, in this case `topics`; and finally the verb, in this case `create`. Option B is incorrect because the last two terms are out of order. Options C and D are incorrect because they do not use `gcloud`. `bq` is the command-line tool for BigQuery. `cbt` is the command-line tool for Bigtable.
15. C. The correct answer, option C, uses `gcloud pubsub subscriptions create` followed by the topic and the name of the subscription. Option A is incorrect because it is missing the term `subscriptions`. Option B is incorrect because it is missing the name of the subscription. Option D is incorrect because it uses `gsutil` instead of `gcloud`.
16. B. Using a message queue between services decouples the services, so if one lags it does not cause other services to lag, which makes option B correct. Option A is incorrect because adding a message queue does not directly mitigate any security risks that might exist in the distributed system, such as overly permissive permissions. Option C is incorrect; adding a queue is not directly related to programming languages. Option D is incorrect; by default, message queues have a retention period.
17. B. The correct answer is B; `gcloud components` followed by `install` and then `beta`. Option A is incorrect because `beta` and `install` are in the wrong order. Options C and D are wrong because `commands` is used instead of `components`.
18. A. The correct parameter name is `autodetect`, which is option A. Options B and C are not actually valid `bq` parameters. Option D is a valid parameter, but it returns the estimated size of data scanned to when executing a query.
19. A. Avro supports Deflate and Snappy compression. CSV supports Gzip and no compression. XML and Thrift are not export file type options.
20. A. The correct answer is A. You would include the `auto-ack` parameter in the `gcloud pubsub subscriptions pull` command. Option B is incorrect, you pull from a subscription, not a topic. Option C is incorrect, you do not use `gsutil` to work with Pub/Sub and `with-acknowledgement` is not a valid parameter. Option D is incorrect because `with-acknowledgement` is not a valid parameter.

Chapter 14: Networking in the Cloud: Virtual Private Clouds and Virtual Private Networks

1. D. Virtual private clouds are global, so option D is correct. By default, they have subnets in all regions. Resources in any region can be accessed through the VPC. Options A, B, and C are all incorrect.
2. B. IP ranges are assigned to subnets, so option B is correct. Each subnet is assigned an IP range for its exclusive use. IP ranges are assigned network structures, not zones and regions. VPCs can have multiple subnets but each subnet has its own address range.

3. B. Option B is correct; dynamic routing is the parameter that specifies whether routes are learned regionally or globally. Option A is incorrect; DNS is a name resolution service and is not involved with routing. Option C is incorrect; there is no static routing policy parameter. Option D is incorrect because systematic routing is not an actual option.
4. A. The correct answer is `gcloud compute networks create`, which is option A. Option B is incorrect; `networks vpc` is not a correct part of the command. Option C is incorrect because `gsutil` is the command used to work with Cloud Storage. Option D is incorrect because there is no such thing.
5. A. The Flow Log option of the `create vpc` command determines whether logs are sent to Cloud Logging, so option A is correct. Option B, Private IP Access, determines whether an external IP address is needed by a VM to use Google services. Option C is incorrect because Cloud Logging is the service, not a parameter used when creating a subnet. Option D is incorrect because variable-length subnet masking has to do with CIDR addresses, not logging.
6. C. Shared VPCs can be created at the organization or folder level of the resource hierarchy, so option C is correct. Options A and B are incorrect; shared VPCs are not created at the resource or project levels. Option D is incorrect; shared VPCs are not applied at subnets, which are resources in the resource hierarchy.
7. A. The correct answer is the Networking tab of the Management, Security, Disks, Networking, Sole Tenancy section of the page, which makes option A correct. The Management tab is not about subnet configurations. Option D is incorrect because it does not lead to Sole Tenancy options.
8. A. VPC network peering is used for interproject communications. Option B is incorrect; there is no interproject peering. Options C and D are incorrect; they have to do with linking on-premises networks with networks in Google Cloud.
9. B. The target can be all instances in a network, instances with network tags, or instances using a specific service account, so option B is correct. Option A is incorrect; Action is either Allow or Deny. Option C is incorrect; Priority determines which of all the matching rules is applied. Option D is incorrect; it specifies whether the rule is applied to incoming or outgoing traffic.
10. D. Direction specifies whether the rule is applied to incoming or outgoing traffic, which makes option D the right answer. Option A is incorrect; Action is either Allow or Deny. Option B is incorrect; Target specifies the set of instances that the rule applies to. Option C is incorrect; Priority determines which of all matching rules is applied.
11. A. The `0.0.0.0/0` matches all IP addresses, so option A is correct. Option B represents a block of 16,777,214 addresses. Option C represents a block of 1,048,574 addresses. Option D represents a block of 65,534. You can experiment with CIDR block options using a CIDR calculator such as the one at www.subnet-calculator.com/cidr.php.
12. B. The product you are working with is compute and the resource you are creating is a firewall rule, so option B is correct. Options A and C references `network` instead of `compute`. Option D references `rules` instead of `firewall-rules`.

13. B. The correct parameter is `--network`, which makes option B correct. Option A is incorrect; `--subnet` is not a parameter to `gcloud` to create a firewall. Option C is incorrect; `--destination` is not a valid parameter. Option D is incorrect; `--source-ranges` is for specifying sources of network traffic the rule applies to.
14. A. The rule in option A uses the correct `gcloud` command and specifies the `allow` and `direction` parameters. Option B is incorrect because it references `gcloud network` instead of `gcloud compute`. Option C is incorrect because it does not specify the port range. Option D is incorrect because it does not specify the protocol or port range.
15. D. Option D is correct because it is the largest number allowed in the range of values for priorities. The larger the number, the lower the priority. Having the lowest priority will ensure that other rules that match will apply.
16. C. The VPC create option is available in the Hybrid Connectivity section, so option C is correct. Compute Engine, App Engine, and IAM & Admin do not have features related to VPNs.
17. B. The correct answer is B; HA VPN is a virtual private network that can provide a 99.99 percent availability SLA and connect on-premises networks to Google Cloud. Classic VPNs do not provide high availability, so option A is incorrect. Option C, Shared VPC, is used for making resources in a host project available to other projects. Option D, VPC network peering, is used to enable a flow of traffic between VPCs, including VPCs in different organizations.
18. A. Option A is correct because global dynamic routing is used to learn all routes on a network. Option B is incorrect; regional routing would learn only routes in a region. Options C and D are incorrect because they are not used to configure routing options.
19. B. The correct command is `B, gcloud compute vpn-tunnels create`. Options A and C are incorrect; `gcloud network` is not the start of a valid command for creating VPN tunnels. Option D is incorrect; the `create` term is in the wrong position.
20. D. When using `gcloud` to create a VPN, you need to create forwarding rules, tunnels, and gateways, so all the `gcloud` commands listed would be used.

Chapter 15: Networking in the Cloud: DNS, Load Balancing, Google Private Access, and IP Addressing

1. B. The A record is used to map a domain name to an IPv4 address, so option B is correct. Option A is incorrect because the AAAA record is used for IPv6 addresses. Option C is incorrect; NS is a name server record. Option D is incorrect; SOA is a start of authority record.
2. A. DNSSEC is a secure protocol designed to prevent spoofing and cache poisoning, so option A is correct. Options B and C are incorrect because SOA and CNAME records contain data about the DNS record; they are not an additional security measure. Option D is incorrect because deleting a CNAME record does not improve security.

3. A. The TTL parameters specify the time a record can be in a cache before the data should be queried again, so option A is correct. Option B is incorrect; this time period is not related to timeouts. Option C is incorrect; the TTLs are not related to time restriction on data change operations. Option D is not correct; there is no manual review required.
4. B. The correct answer, option B, is `gcloud dns managed-zones create`. Option A is incorrect; it uses the `gsutil` command, which is used to work with Cloud Storage. Option C is incorrect; it is missing `dns`. Option D is incorrect; `create` is in the wrong position.
5. B. The `visibility` parameter is the parameter that can be set to `private`, so option B is correct. Option A is not a valid parameter. Option C is incorrect; `private` is not a parameter. Similarly, option D is incorrect; `status` is not a valid parameter for making a DNS zone private.
6. C. The global load balancers are Global External HTTP(S) Load Balancing, Global External HTTP(S) Load Balancing (classic), SSL Proxy, and TCP Proxy, so option C is correct. Options A and B are missing at least one global load balancer. Option D is incorrect because Internal TCP/UDP is a regional load balancer.
7. D. Internal HTTP(S) Load Balancing distributes traffic regionally on Premium tier networking. The others are global load balancers.
8. A. In the console there is an option to select between From Internet To My VMs and Only Between My VMs. This is the option to indicate private or public, so option A is correct. Options B, C, and D are all fictitious parameters.
9. B. TCP Proxy load balancers require you to configure both the front end and the back end, so option B is correct. Options A and D are incorrect because they are missing one component. Option C is incorrect; forwarding rules are the one component specified with network load balancing.
10. B. Health checks monitor the health of VMs used with load balancers, so option B is correct. Option A is incorrect; there is no need for health checks on policies. Option C and D are incorrect; storage devices or buckets are not health checked.
11. B. You specify ports to forward when configuring the front end, so option B is correct. The back end is where you configure how traffic is routed to VMs. Option C is incorrect; Network Services is a high-level area of the console. Option D is incorrect; VPCs are not where you specify load balancer configurations.
12. A. The correct answer, option A, is `gcloud compute forwarding-rules create`. Option B is incorrect; the service should be `compute`, not `network`. Option C is incorrect; `create` comes after `forwarding-rules`. Option D is incorrect because it has the wrong service and the verb is in the wrong position.
13. C. Static addresses are assigned until they are released, so option C is correct. Options A and B are incorrect because internal and external addresses determine whether traffic is routed into and out of the subnet. External addresses can have traffic reach them from the Internet; internal addresses cannot. Option D is incorrect; ephemeral addresses are released when a VM shuts down or is deleted.

14. A. An ephemeral address is sufficient, since resources outside the subnet will not need to reach the VM and you can SSH into the VM from the console, so option A is correct. Option B is incorrect because there is no need to assign a static address, which would then have to be released. Option C is incorrect; there is no Permanent type. Option D is incorrect; there is no IPv8 address.
15. D. You cannot reduce the number of addresses using any of the commands, so option D is correct. Option A is incorrect because the prefix length specified in the `expand-ip-range` command must be a number less than the current length. If there are 65,534 addresses, then the prefix length is 16. Option B is incorrect for the same reason, and the prefix length cannot be a negative number. Option C is incorrect; there is no `--size` parameter.
16. B. The prefix length specifies the length in bits of the subnet mask. The remaining bits of the IP address are used for device addresses. Since there are 32 bits in an IP address, you subtract the length of the mask to get the number of bits used to represent the address. 16 is equal to 2^4 , so you need 4 bits to represent 14 addresses. $32 - 4$ is 28, so option B is the correct answer. Option A would leave 1 address, option C would provide 4,094 addresses, and option D would provide 65,534.
17. C. Premium is the network service level that routes all traffic over the Google network, so option C is correct. Option A is incorrect; the Standard tier may use the public Internet when routing traffic. Options B and D are incorrect; there are no service tiers called Google-only or non-Internet.
18. B. Stopping and starting a VM will release ephemeral IP addresses, so option B is correct. Use a static IP address to have the same IP address across reboots. Option A is incorrect; rebooting a VM does not change a DNS record. Option C is incorrect because if you had enough addresses to get an address when you first started the VM and you then released that IP address, there should be at least one IP address assuming no other devices are added to the subnet. Option D is incorrect because no other changes, including changes to the subnet, were made.
19. A. Internal TCP/UDP is a good option. It is a regional load balancer that supports UDP, so option A is correct. Options B, C, and D are all global load balancers. Option B supports TCP, not UDP. Option D supports HTTP and HTTPS, not UDP.
20. B. Network Services is the section of Cloud Console that has the Cloud DNS console, so option B is correct. Option A is incorrect; Compute Engine does not have DNS management forms. Neither does option C, Kubernetes Engine. Option D is related to networking, but the services in Hybrid Connectivity are for services such as VPNs.

Chapter 16: Deploying Applications with Cloud Marketplace and Cloud Foundation Toolkit

1. D. Categories of solutions include all of the categories mentioned, so option D is correct. Others include Kubernetes Apps, API & Services, and Databases.
2. B. The correct answer is B; the Cloud Foundation Toolkit provides blueprints and other configurations for common solutions, such as data warehouses, as well as templates for specific resources, such as virtual machines. Option A, Cloud Deployment Manager, is a service for deploying solutions but is not a set of example solutions. Option C, Config Connector, is a Kubernetes add-on for managing Google Cloud resources from Kubernetes. Option D, Cloud Build, is a Google Cloud service for building containers.
3. A. You launch a solution by clicking the Launch On Compute Engine link on the overview page, so option A is correct. Option B is incorrect; the main page has summary information about the products. Option C is incorrect; Network Services is unrelated to this topic. Option D is incorrect because option A is the correct answer.
4. B. Cloud Marketplace has a set of predefined filters, including filtering by operating system, so option B is correct. Option A may eventually lead to the correct information, but it is not efficient. Option D is incorrect because it is impractical for such a simple task.
5. B. Multiple vendors may offer configurations for the same applications, so option B is correct. This gives users the opportunity to choose the one best suited to their requirements. Options A and C are incorrect; this is a feature of Cloud Launcher. Option D is incorrect because option B is the correct answer.
6. C. Cloud Launcher will display configuration options appropriate for the application you are deploying, so option C is correct. For example, when deploying WordPress, you will have the option of deploying an administration tool for PHP. Option A is incorrect; this is a feature of Cloud Launcher. Option B is incorrect; you are not necessarily on the wrong page. Option D is incorrect; this is a feature of Cloud Launcher.
7. D. You can change the configuration of any of the items listed, so option D is correct. You can also specify firewall rules to allow both HTTP and HTTPS traffic or change the zone in which the VM runs.
8. B. Deployment Manager is the name of the service for creating application resources using a YAML configuration file, so option B is correct. Option A is incorrect, although you could use scripts with `gcloud` commands to deploy resources in Compute Engine. Options C and D are incorrect because those are fictitious names of products.

9. D. Configuration files are defined in YAML syntax, so option D is correct.
10. B. Configuration files define resources and start with the word `resources`, so option B is correct.
11. D. All three—type, properties, and name—are used when defining resources in a Cloud Deployment Manager configuration file, so option D is correct.
12. D. All three can be set; specifically, the keys are `deviceName`, `boot`, and `autodelete`. Option D is correct.
13. A. Option A is the correct command. Option B is incorrect; it is missing the term `compute`. Option C is incorrect; `gsutil` is the command for working with Cloud Storage. Option D is incorrect because the terms `list` and `images` are in the wrong order.
14. D. Google recommends using Python for complicated templates, so option D is correct. Option A is incorrect because Jinja2 is recommended only for simple templates. Options B and C are incorrect; neither language is supported for templates.
15. A. The correct answer is `gcloud deployment-manager deployments create`, so option A is correct. Options B and D are incorrect; the service is not called `cloud-launcher` in the command. Option C is incorrect; `launch` is not a valid verb for this command.
16. C. The correct answer is `gcloud deployment-manager deployments describe`, so option C is correct. Options A and D are incorrect; `cloud-launcher` is not the name of the service. Option B is incorrect; `list` displays a brief summary of each deployment. `describe` displays a detailed description.
17. A. You will be able to configure IP addresses, so option A is correct. You cannot configure billing or access controls in Deployment Manager, so options B and C are incorrect. You can configure the machine type, but that is not in the More section of Networking.
18. D. The correct answer is option D because `free`, `flat hourly`, `usage fees`, and `BYOL` are all license options used in Cloud Marketplace.
19. B. The `flat hourly` and `usage fees` license types include payment for the license in your Google Cloud charges, so option B is correct. The `free` license type does not incur charges. The `BYOL` license type requires you to work with the software vendor to get and pay for a license. There is no such license type as `chargeback`, so option D is incorrect.
20. D. LAMP is short for Linux, Apache, MySQL, and PHP. All are included when installing LAMP solutions, so option D is correct.

Chapter 17: Configuring Access and Security

1. B. IAM stands for identity and access management, so option B is correct. Option A is incorrect; the A does not stand for authorization, although that is related. Option C is incorrect; the A does not stand for auditing, although that is related. Option D is incorrect. IAM also works with groups, not just individuals.
2. A. Members and their roles are listed, so option A is correct. Options B and C are incorrect because they are missing the other main piece of information provided in the listing. Option D is incorrect; permissions are not displayed on that page.
3. B. Basic roles were created before IAM and provided coarse-grained access controls, so option B is correct. Option A is incorrect; they are used for access control. Option C is incorrect; IAM is the newer form of access control. Option D is incorrect; they do provide access control functionality.
4. B. Roles are used to group permissions that can then be assigned to identities, so option B is correct. Option A is incorrect; roles do not have identities, but identities can be granted roles. Option C is incorrect; roles do not use access control lists. Option D is incorrect; roles do not include audit logs. Logs are collected and managed by Stackdriver Logging.
5. C. The correct answer is `gcloud projects get-iam-policy ace-exam-project`, so option C is correct. Option A is incorrect because the resource should be `projects` and not `iam`. Option B is incorrect; `list` does not provide detailed descriptions. Option D is incorrect because `iam` and `list` are incorrectly referenced.
6. B. New members can be users, indicated by their email addresses, or groups, so option B is correct. Option A is incorrect; it does not include groups. Options C and D are incorrect because roles are not added there.
7. D. Deployers can read application configurations and settings and write new application versions, so option D is correct. Option A is incorrect because it is missing the ability to read configurations and settings. Option B is incorrect because it is missing writing new versions. Option C is incorrect because it references writing new configurations.
8. B. The correct steps are navigating to IAM & Admin, selecting Roles, and then checking the box next to a role, so option B is correct. Option A is incorrect; all roles are not displayed automatically. Option C is incorrect; audit logs do not display permissions. Option D is incorrect; there is no Roles option in Service Accounts.
9. D. Predefined roles help implement both least privilege and separation of duties, so option D is correct. Predefined roles do not implement defense in depth by themselves but could be used with other security controls to implement defense in depth.

10. D. The four launch stages available are alpha, beta, general availability, and disabled, so option D is correct.
11. B. The correct answer, option B, is `gcloud iam roles create`. Option A is incorrect because it references `project` instead of `iam`. Option C is incorrect because it references `project` instead of `iam`, and the terms `create` and `roles` are out of order. Option D is also incorrect because the terms `create` and `roles` are out of order.
12. B. Scopes are permissions granted to VM instances, so option B is correct. Scopes in combination with IAM roles assigned to service accounts assigned to the VM instance determine what operations the VM instance can perform. Options A and C are incorrect; scopes do not apply to storage resources. Option D is incorrect; scopes do not apply to subnets.
13. C. Scope identifiers start with `www.googleapis.com/auth` and are followed by a scope-specific name, such as `devstorage.read_only` or `logging.write`, so option C is correct. Option A is incorrect; scope IDs are not randomly generated. Option B is incorrect; the domain name is not `googleserviceaccounts`. Option D is incorrect; scopes are not linked directly to projects.
14. C. Both scopes and IAM roles assigned to service accounts must allow an operation for it to succeed, so option C is correct. Option A is incorrect; access controls do not affect the flow of control in applications unless explicitly coded for that. Option B is incorrect; the most permissive permission is not used. Option D is incorrect; the operation will not succeed.
15. B. The options for setting scopes are Allow Default Access, Allow Full Access, and Set Access For Each API, so option B is correct. Option A is incorrect; it is missing Set Access For Each API. Option C is incorrect; it is missing Allow Default Access. Option D is incorrect; it is missing Allow Full Access.
16. B. The correct command is `gcloud compute instances set-service-account`, so option B is correct. Option A is incorrect; there is no `set-scopes` command verb. Option C is incorrect; the command verb is not `set-scopes`. Option D is incorrect; there is no command verb `define-scopes`.
17. A. You can assign a service account when creating a VM using the `create` command. Option B is incorrect; there is no `create-service-account` command verb. Option C is incorrect; there is no `define-service-account` command verb. Option D is incorrect; there is no `instances-service-account` command; also, `create` should come at the end of the command.
18. C. Cloud Logging collects, stores, and displays log messages, so option C is correct. Option A is incorrect; Compute Engine does not manage logs. Option B is incorrect; Cloud Storage is not used to view logs, although log files can be stored there. Option D is incorrect; custom logging solutions are not Google Cloud services.
19. B. Logs can be filtered by resource, type of log, log level, and period of time only, so option B is correct. Options A, C, and D are incorrect because they are each missing at least one option.

20. B. This is an example of assigning the least privilege required to perform a task, so option B is correct. Option A is incorrect; defense in depth combines multiple security controls. Option C is incorrect because it is having different people perform sensitive tasks. Option D is incorrect; vulnerability scanning is a security measure applied to applications that helps reveal potential vulnerabilities in an application that an attacker could exploit.

Chapter 18: Monitoring, Logging, and Cost Estimating

1. B. The Monitoring service is used to set a threshold on metrics and generate alerts when a metric exceeds the threshold for a specified period of time, so option B is correct. Option A is incorrect; Logging is for collecting log messages about events. Option C is incorrect; Cloud Trace is for application tracing. Option D is incorrect; Debugger is a deprecated service that was used to debug applications. It will no longer be available after May 2023.
2. B. Option B is correct. You would install the Ops Agent on the VM. The agent will collect data and send it to Cloud Monitoring and Cloud Logging. Option A is incorrect because there is no such thing as a Cloud Operations image. Option C is incorrect; there is no Monitor With Cloud Monitoring option on the VM configuration page. Option D is incorrect because you set notification channels in an alerting policy, not on a VM.
3. D. Cloud Monitoring can monitor resources in Google Cloud, AWS, and on-premises data centers, so option D is correct. Options A through C are incorrect because they do not include two other correct options.
4. A. The correct answer is A. A dashboard in Cloud Monitoring would allow you to view a set of charts in one place. Option B, a Cloud Logging sink, is used to route log messages to a storage location. Option C is incorrect; a Cloud Monitoring Alert is for sending notifications when a metric exceeds or falls below a threshold or monitoring data is missing for a period of time. Option D is incorrect; a BigQuery data set is used to store a set of related tables and views.
5. D. All three options are valid notification channels in Cloud Monitoring, so option D is correct. PagerDuty is a popular DevOps tool.
6. D. The documentation is useful for documenting the purpose of the policy and for providing guidance for solving the problem, so option D is correct. Option A is incorrect; where a policy is stored is irrelevant to its usefulness. Options B and C alone are partially correct, but option D is a better answer.
7. A. Alert fatigue is a state caused by too many alert notifications being sent for events that do not require human intervention, so option A is correct. This creates the risk that eventually DevOps engineers will begin to pay less attention to notifications. Option B is incorrect, although it is conceivable that too many alerts could adversely impact performance, but that is not likely. Option C is a potential problem, too, but that is not alert fatigue. Option D is incorrect because logging services have a variety of ways to filter log messages that allow users to find precisely the messages they are interested in.

8. C. Cloud Logging stores log entries in the Default bucket for 30 days, so option C is correct.
9. B. Option B is correct. The best option is to use create a user-defined bucket with a custom retention policy. Option A is incorrect; there is a way to export data. Options C and D are incorrect because writing a custom script would take more time to develop and maintain than using Logging's export functionality.
10. D. All three, Cloud Storage buckets, BigQuery data sets, and Cloud Pub/Sub topics, are available as sinks for logging exports, so option D is correct.
11. D. Option D is correct. All of the options listed can be used to filter log messages.
12. B. The correct answer, option B, is halted. There is no such standard log level status. Statuses include Critical, Error, Warning, Info, and Debug.
13. A. The correct answer is A. You can expand the `metadataRequest` field in the JSON structure of the message in Log Explorer. Option B is incorrect; Metric Explorer is used with Cloud Monitoring metrics, not log messages. Option C is more time-consuming than using the functionality built into Cloud Logging. Option D is incorrect; there is no such link.
14. C. Cloud Trace is a distributed tracing application that provides details on how long different parts of code run, so option C is correct. Option A is incorrect; Monitoring is used to notify DevOps engineers when resources are not functioning as expected. Option B is incorrect; Logging is for collecting, storing, and viewing log data, and although log entries might help diagnose bottlenecks, they are not specifically designed for that purpose. Option D is incorrect; Debugger is a deprecated service that was used to debug applications. It will no longer be available after May 2023.
15. C. Option C, Trace is correct. It is a distributed tracing application that provides details on how longer different parts of code run. Option A is incorrect, Monitoring is used to observe metrics about services and alert on unwanted conditions. Option B is incorrect, Logging is for collecting, storing, and viewing log data. Option D is incorrect, Debugger is a deprecated service.
16. B. The Google Cloud Status Dashboard at <https://status.cloud.google.com> has information on the status of Google Cloud services, so option B is correct. Options A and D might lead to information, but they would take longer. Option C is not a link to a source of information on BigQuery.
17. B. Both Compute Engine and Kubernetes Engine will require details about the VMs' configurations, so option B is correct. The other options are incorrect because BigQuery and Cloud Pub/Sub are serverless services.
18. C. Query pricing in BigQuery is based on the amount of data scanned, so option C is correct. Option A is incorrect; the amount of data storage is specified in the Storage Pricing section. Option B is incorrect; query pricing is not based on the volume of data returned. Option D is incorrect because the number of partitions is not a factor in BigQuery pricing.

- 19.** B. Some operating systems, like Microsoft Windows Server, require a license, so option B is correct. Google sometimes has arrangements with vendors to collect fees for using proprietary software. Option A is incorrect; there is no fixed rate charge for operating systems. Option C is incorrect; the information is sometimes needed to compute charges. Option D is incorrect because if you bring your own license, there is no additional license charge.
- 20.** D. Option D is correct. The Required bucket is used to store admin activity, system events, and access transparency. Option A is incorrect; operating system messages are not routed to the Required bucket. Options B and C are incorrect because they only include some of the types of logs routed to the Required bucket.

Index

A

- AAAA records, 376
- accounts
 - assigning IAM roles to, 428–432
 - billing, 53–55
 - service, 436–440
 - viewing IAM assignments, 426–428
- Address Allocation for Private Internets standard (RFC 1918), 29
- AI (artificial intelligence), 32–33
- alerts
 - billing, 56–57
 - creating, 454–458
- ALTER TABLE command, 299
- analytical storage data model, 258–270
- analytics, in the cloud, 367–368
- Anthos, 21–22, 90
- Apache Hadoop, 308
- Apache Spark, 308, 312–313, 333
- Apigee API platform, 32
- APIs, enabling, 59–60
- App Engine
 - about, 22, 83, 216, 230
 - components, 223–226
 - deploying applications, 226–228
 - exam essentials, 231
 - flexible environment, 86, 87
 - review questions, 232–235, 491–493
 - roles for, 428
 - scaling applications, 228–229
 - splitting traffic between versions, 229–230
 - standard environment, 85–86, 87
 - structure of applications, 84–85
 - use cases for, 86–87
- App Engine Admin role, 428

- App Engine Code Viewer role, 428
- App Engine Deployer role, 428
- App Engine Service Admin role, 428
- App Engine Viewer role, 428
- application pods, deploying, 168–172
- architecture, of Kubernetes clusters,
 - 88–89, 159
- Archive storage, 261
- Artifact Registry, creating repositories
 - in, 207–209
- artificial intelligence (AI), 32–33
- audit logs, viewing, 440
- Automatic Restart, 111
- AutoML, 32
- autoscaling
 - instance groups, 147
 - using, 196
- Avro files, 333

B

- backing up
 - Firestore, 294
 - MySQL in Cloud SQL, 289–292
- BGP (Border Gateway Protocol), 364
- BigQuery
 - about, 57–59, 258–270, 294
 - deploying, 294–297
 - estimating cost of queries in, 294–296
 - importing and exporting data, 332–337
 - metric for, 448
 - viewing jobs in, 296–297
- billing
 - about, 42, 53–59, 60
 - accounts for, 53–55

- budgets and alerts, 56–57
- enabling APIs, 59–60
- exam essentials, 61
- exporting data for, 57–59
- organizing projects and accounts, 42–49
- review questions, 62–66, 478–480
- roles and identities, 49–52
- service accounts, 52–53
- blocks
 - CIDR, 390
 - defined, 5
 - storage of, 5
- Bonér, Jonas
 - “Latency Numbers Every Programmer Should Know,” 6
- Boot Disks, 112–115
- Border Gateway Protocol (BGP), 364
- bq command, 294
- bq `extract` command, 334
- bq `load` command, 337
- bq `show` command, 297
- buckets, 241
- budgets, billing, 56–57

C

- caches, 5–6, 255–257
- `cbt` command, 306–307
- CDNs (content delivery networks), 29
- CIDR (classless interdomain routing)
 - notation
 - about, 352, 355
 - expanding blocks, 390
- classes, storage, 260–261
- classless interdomain routing (CIDR)
 - notation
 - about, 352, 355
 - expanding blocks, 390
- Cloud Armor, 29
- Cloud Bigtable
 - deploying, 304–308
 - exporting data from, 339–340
 - managing, 304–308
 - metric for, 448
 - using, 270–276
- Cloud CDN, 29
- cloud computing
 - categories of, 2–8
 - data center computing compared
 - with, 8–10
 - exam essentials, 10–11
 - review questions, 12–15, 474–475
 - types of, 2–8
- Cloud Dataproc
 - deploying, 308–313
 - exporting data, 340
 - importing data, 340
 - managing, 308–313
- Cloud Debugger, 32
- Cloud DNS
 - about, 30
 - configuring, 376–382
 - creating DNS managed zones
 - using Cloud Console, 376–381
 - using `gcloud` command, 381–382
- Cloud Firestore
 - about, 5, 292
 - adding data to databases, 292–293
 - backing up, 294
 - deploying, 292–294
 - exporting data, 332
 - importing data, 332
 - using, 270–276
- Cloud Foundation Toolkit, 411–418
- Cloud Functions
 - about, 23, 91, 238, 247
 - events, 238–239
 - exam essentials, 247–248
 - execution environment, 91–93
 - functions, 238–239
 - metric for, 448
 - receiving events
 - from Cloud Storage, 241–245
 - from Pub/Sub, 245–247

- review questions, 249–252, 494–496
 - runtime environments, 239–240
 - triggers, 238–239
 - use cases, 93
- Cloud Interconnect, 29–30
- Cloud Load Balancing, 29
- Cloud Logging
- about, 31, 448, 458, 467
 - configuring log sinks, 459
 - diagnostics, 469–472
 - exam essentials, 468
 - filtering logs, 459–461
 - log routers, 458–459
 - log sinks, 458–459
 - Pricing Calculator, 469–472
 - review questions, 469–472, 511–513
 - viewing
 - logs, 459–461
 - message details, 462–463
- Cloud Marketplace, deploying solutions using, 400–410
- Cloud Monitoring
- about, 31, 448, 467
 - creating
 - alerts, 454–458
 - dashboards, 449–450
 - exam essentials, 468
 - review questions, 469–472, 511–513
 - using Metric Explorer, 450–453
- cloud networking. *See* networking
- Cloud Profiler, 32
- Cloud Pub/Sub
- deploying, 300–304
 - managing, 300–304
 - receiving events from, 245–247
 - streaming data to, 341
- Cloud Run
- about, 23, 90–91, 216, 230
 - creating
 - jobs, 222–223
 - services, 218–222
 - exam essentials, 231
 - jobs, 217–218
 - review questions, 232–235, 491–493
 - services, 216–217
 - use cases, 91
- Cloud SDK
- about, 30–31
 - adding
 - nodes with, 195–196
 - Pods with, 200–202
 - services with, 205–207
 - creating virtual machines with, 117–121
 - deploying
 - applications using, 226–228
 - Kubernetes clusters using, 167–168
 - installing, 117–118
 - modifying
 - nodes with, 195–196
 - Pods with, 200–202
 - services with, 205–207
 - removing
 - nodes with, 195–196
 - Pods with, 200–202
 - services with, 205–207
 - viewing status of Kubernetes clusters using, 187–193
- Cloud Shell
- adding
 - nodes with, 195–196
 - Pods with, 200–202
 - services with, 205–207
 - creating virtual machines with, 120–121
 - deploying
 - applications using, 226–228
 - Kubernetes clusters using, 167–168
 - managing single virtual machine instances with, 141–147
 - modifying
 - nodes with, 195–196
 - Pods with, 200–202
 - services with, 205–207
 - removing
 - nodes with, 195–196
 - Pods with, 200–202
 - services with, 205–207
 - viewing status of Kubernetes clusters using, 187–193

Cloud Spanner

- about, 256–268
- deploying, 297–300
- exporting, 337–339
- importing, 337–339
- managing, 297–300

Cloud SQL

- about, 256–268, 286
- backing up MySQL in, 289–292
- creating
 - databases, 288–289
 - MySQL instances, 286–288
- deploying, 286–292
- exporting data, 328–331
- importing data, 328–331
- loading data, 288–289
- querying data, 288–289

Cloud Storage

- about, 146–147
- configuring, 262–264
- features of, 260
- loading and moving data to
 - using command line, 327–328
 - using Google Cloud Console, 322–327
- managing, 314–315
- receiving events from, 241–245

Cloud Storage Fuse, 260

Cloud Trace, 31, 463

clusters. *See* Kubernetes clusters

CNAM records, 376

Coldline storage, 261

command line

- loading and moving data to Cloud Storage
 - using, 327–328
- managing single virtual machine instances
 - with, 141–147

commands. *See also* gcloud command

- ALTER TABLE, 299
- bq, 294
- bq extract, 334
- bq load, 337
- bq show, 297
- cbt, 306–307

CREATE DATABASE, 288

CREATE INDEX, 299

create instance, 120

CREATE TABLE, 288, 299

delete, 145

DROP INDEX, 299

DROP TABLE, 299

export, 340

gcloud app deploy app.yml, 227

gcloud app services set-traffic, 230

gcloud app versions stop, 228

gcloud components install

app-engine-python, 226

gcloud components install

beta, 244

gcloud components update, 244

gcloud compute addresses create, 391

gcloud compute firewall-rules, 364

gcloud compute forwarding-rule, 368

gcloud compute forward-rules, 388

gcloud compute instances, 144

gcloud compute instances

create, 358, 439

gcloud compute networks

create, 354

gcloud compute networks subnet

create, 354

gcloud compute shared-vpc, 355

gcloud compute target-pools

create, 389

gcloud compute target-vpn-

gateways, 368

gcloud compute vpn-tunnels, 368

gcloud container cluster

list, 187–189

gcloud container clusters

describe, 189–191

gcloud container clusters

resize, 195–196

gcloud dataproc clusters, 312

gcloud dataproc jobs, 312

gcloud deployment-manager

deployments create, 414

- gcloud firestore export, 332
- gcloud firestore import, 294, 332
- gcloud functions delete, 247
- gcloud iam roles create, 434
- gcloud iam roles describe, 430
- gcloud pubsub topics, 341
- gcloud sql backups, 291
- gcloud sql export, 331
- gcloud sql import, 331
- gcloud sql instances describe, 331
- gsutil, 294, 330
- gsutil mb, 327
- gsutil mv, 314
- gsutil rewrite, 314
- import, 340
- INSERT, 288
- instance start, 144
- kubect, 200
- kubect delete deployment, 202
- kubect delete service, 207
- kubect describe nodes, 192–193
- kubect expose deployment, 206–207
- kubect get deployments, 201–202
- kubect get pods, 191–192
- kubect get services, 205–206
- kubect scale, 172
- ls, 307–308
- move, 328
- mv, 314
- read, 308
- SELECT, 288–289
- sql connect, 287–288
- start, 144
- stop, 144
- comma-separated values (CSV) files, 333
- Compute Admin role, 80
- Compute Engine
 - about, 19–20, 257
 - custom machine types, 81–82
 - deploying with custom networks, 357–359
 - metric for, 448
 - preemptible virtual machines, 80–81
 - use cases for virtual machines, 82–83
 - users needing privileges to create virtual machines, 79–80
 - virtual machine images, 68–77
 - virtual machines
 - contained in projects, 77–78
 - run in zones and regions, 78–79
- Compute Engine virtual machines
 - about, 125
 - basic management of, 121–124
 - Boot Disks, 112–115
 - configuration details, 104–117
 - cost of, 123–124
 - creating
 - with Cloud SDK, 117–121
 - with Google Cloud Console, 102–117
 - exam essentials, 126
 - example installation on Ubuntu
 - Linux, 118–119
 - guidelines for planning, deploying, and managing, 125
 - installing Cloud SDK, 117–118
 - Management tab, 109–111
 - monitoring, 123
 - network access to, 121–123
 - Networking tab, 115
 - review questions, 127–130, 482–485
 - Security tab, 111–112
 - Sole-Tenancy tab, 115–117
 - starting instances, 121
 - stopping instances, 121
- Compute Network Admin role, 80
- compute resources, forms of, 3–4
- Compute Security Admin role, 80
- Compute Viewer role, 80
- computing
 - about, 93–95
 - App Engine, 83–87
 - Cloud Functions, 91–93
 - Cloud Run, 90–91
 - components of
 - about, 18–19
 - App Engine, 22
 - Cloud Functions, 23

- Cloud Run, 23
- Compute Engine, 19–20
- computing resources, 19–23
- of Google Cloud, 11–23
- Kubernetes Engine, 20–22
- Compute Engine, 68–83
- custom machine types, 81–82
- exam essentials, 95
- Kubernetes Engine, 87–90
- preemptible virtual machines, 80–81
- review questions, 96–99, 480–482
- use cases for Compute Engine virtual machines, 82–83
- users needing privileges to create virtual machines, 79–80
- virtual machine images, 68–77
- virtual machines
 - contained in projects, 77–78
 - in zones and regions, 78–79
- Config Connector, 417–418
- configuration files, for Deployment Manager, 411–414
- consistency, storage and, 277
- constraints, on resources, 45–46
- container, 3
- content delivery networks (CDNs), 29
- control plane, 159
- controller, 160
- cost
 - estimating of queries in BigQuery, 294–296
 - storage and, 277
 - of virtual machines, 123–124
- CREATE DATABASE command, 288
- CREATE INDEX command, 299
- create instance command, 120
- CREATE TABLE command, 288, 299
- CSV (comma-separated values) files, 333
- custom machine types, 81–82

D

- dashboards, creating, 449–450
- data
 - adding to Firestore databases, 292–293

- loading, 288–289
- querying, 288–289
- streaming to Cloud Pub/Sub, 341
- data analytics, 32
- data center computing, cloud computing
 - compared with, 8–10
- data definition language (DDL), 299–300
- data models
 - analytical, 268–270
 - NoSQL, 270–276
 - object, 266
 - relational, 266–268
 - storage, 265–276
- data pipelines, 32
- database queries, improving response time for, 6–7
- databases
 - creating, 288–289
 - Firestore, 292–293
 - need for multiple, 276
 - relational, 256–268
- DDL (data definition language), 299–300
- decoupling services, using message queues, 341–342
- delete command, 145
- deployable services, 415
- deploying
 - about, 161, 400, 418
 - App Engine applications, 226–228
 - application pods, 168–172
 - BigQuery, 294–297
 - building infrastructure using Cloud Foundation Toolkit, 411–418
 - Cloud Bigtable, 304–308
 - Cloud Dataproc, 308–313
 - Cloud Function for Cloud Pub/Sub events
 - using Cloud Console, 245–246
 - using `gcloud` commands, 246–247
 - Cloud Function for Cloud Storage events
 - using Cloud Console, 241–244
 - using `gcloud` commands, 244–245
 - Cloud Pub/Sub, 300–304
 - Cloud Spanner, 297–300
 - Cloud SQL, 286–292
 - Compute Engine with custom networks, 357–359

- exam essentials, 418–419
- Firestore, 292–294
- Kubernetes clusters, 162–168
- review questions, 420–423, 507–508
- solutions using Cloud
 - Marketplace, 400–410
 - virtual machines, 125
- Deployment Manager
 - configuration files, 411–414
 - launching templates, 414
 - template files, 414
- development tools, 30–31
- diagnostics
 - about, 463, 467
 - Cloud Logging, 469–472
 - Cloud Trace, 463
 - exam essentials, 468
 - Google Cloud Status, 464
- documents, making searchable, 240
- DROP INDEX command, 299
- DROP TABLE command, 299
- dual regional storage, 261
- durability rates, 255
- dynamic routing, 353

E

- elastic resource allocation, 9
- enabling APIs, 59–60
- entity, 272
- ephemeral IP addresses, 389
- Error Reporting, 31
- events
 - Cloud Functions, 238–239
 - receiving
 - from Cloud Storage, 241–245
 - from Pub/Sub, 245–247
- exam essentials
 - App Engine, 231
 - cloud computing, 10–11
 - Cloud Functions, 247–248
 - Cloud Logging, 468
 - Cloud Monitoring, 468
 - Cloud Run, 231

- Compute Engine virtual machines, 126
- computing, 95
- deploying, 418–419
- diagnostics, 468
- Google Cloud, 33–35
- identity and access management
 - (IAM), 441–442
- Kubernetes clusters, 209
- Kubernetes Engine, 173
- loading data, 342–343
- networking, 369, 392–393
- Pricing Calculator, 468
- projects, service accounts, and billing, 61
- storage, 278–279, 315–316
- virtual machines (VMs), 151
- execution environment, in Cloud
 - Functions, 91–93
- export command, 340
- exporting billing data, 57–59

F

- file formats, 333
- file storage, 5
- firewall rules, creating for virtual private
 - clouds, 359–364
- fixed IP addresses, 389
- fleet, 21
- flexible environment, App Engine and,
 - 22, 86, 87
- folders, in resource hierarchy, 44
- functionality, of Kubernetes Engine, 88
- functions, 238–239

G

- GCE (Google Compute Engine). *See*
 - Compute Engine
- gcloud app deploy app.yaml command, 227
- gcloud app services set-traffic
 - command, 230
- gcloud app versions stop
 - command, 228

gcloud command

- about, 119–120, 143, 167–168, 291, 304
- configuring load balancers using, 386–389
- creating
 - DNS managed zones using, 381–382
 - firewall rules using, 364
 - shared virtual private clouds
 - using, 355–357
 - virtual private clouds with, 354–355
 - virtual private network (VPN)
 - using, 368
- deploying
 - Cloud Function for Cloud Pub/Sub events using, 246–247
 - Cloud Function for Cloud Storage events using, 244–245
- gcloud components install app-engine-python command, 226
- gcloud components install beta command, 244
- gcloud components update command, 244
- gcloud compute addresses create command, 391
- gcloud compute firewall-rules command, 364
- gcloud compute forwarding-rule command, 368
- gcloud compute forward-rules command, 388
- gcloud compute instances command, 144
- gcloud compute instances create command, 358, 439
- gcloud compute networks create command, 354
- gcloud compute networks subnet create command, 354
- gcloud compute shared-vpc command, 355
- gcloud compute target-pools create command, 389
- gcloud compute target-vpn-gateways command, 368
- gcloud compute vpn-tunnels command, 368
- gcloud container cluster list command, 187–189

gcloud container clusters

- describe command, 189–191
- gcloud container clusters resize command, 195–196
- gcloud dataproc clusters command, 312
- gcloud dataproc jobs command, 312
- gcloud deployment-manager deployments create command, 414
- gcloud firestore export command, 332
- gcloud firestore import command, 294, 332
- gcloud functions delete command, 247
- gcloud iam roles create command, 434
- gcloud iam roles describe command, 430
- gcloud pubsub topics commands, 341
- gcloud sql backups command, 291
- gcloud sql export command, 331
- gcloud sql import command, 331
- gcloud sql instances describe command, 331
- georedundant, 261
- GKE (Google Kubernetes Engine), 21–22, 257
- Google Cloud
 - about, 2, 10, 33
 - additional components of, 31–33
 - computing components of, 11–23
 - exam essentials, 33–35
 - networking components of, 28–31
 - resource hierarchy, 42–45
 - review questions, 36–40, 476–478
 - roles in, 50
 - storage components of, 23–28
- Google Cloud Console
 - about, 46–49, 77–78
 - adding
 - nodes with, 193–195
 - Pods with, 196–200
 - services with, 203–205
 - configuring load balancers using, 383–386
 - creating
 - DNS managed zones using, 376–381
 - firewall rules using, 361–363
 - virtual machines with, 102–117
 - virtual private clouds with, 350–354
 - virtual private network (VPN)
 - using, 364–366

- deploying
 - Cloud Function for Cloud Pub/Sub events using, 245–246
 - Cloud Function for Cloud Storage events using, 241–244
 - Kubernetes clusters using, 162–167
- loading and moving data to Cloud Storage using, 322–327
- managing single virtual machine instances in, 132–141
- modifying
 - nodes with, 193–195
 - pods with, 196–200
 - services with, 203–205
- removing
 - nodes with, 193–195
 - pods with, 196–200
 - services with, 203–205
- viewing
 - image repository/image details with, 207–209
 - status of Kubernetes clusters using, 180–181
- Google Cloud Status, 464
- Google Compute Engine (GCE). *See* Compute Engine
- Google Kubernetes Engine (GKE), 21–22, 257
- Google Workspace, 43–44
- GPUs, attaching to instances, 136–138
- groups, assigning IAM roles to, 428–432. *See also* instance groups
- gsutil command, 294, 330
- gsutil mb command, 327
- gsutil mv command, 314
- gsutil rewrite command, 314
- Gzip files, 333

H

- hard disk drive (HDD)
 - configurations, 257–258
- high availability, load balancing and, 383

- Host Maintenance, 111
- hot data, 260

I

- IAM. *See* Identity and Access Management (IAM)
- identities, granting roles to, 50–52
- Identity and Access Management (IAM)
 - about, 30, 426, 441
 - assigning roles to accounts and groups, 428–432
 - defining custom roles, 432–435
 - exam essentials, 441–442
 - managing
 - about, 426–435
 - service accounts, 436–440
 - review questions, 443–446, 509–511
 - viewing account assignments, 426–428
- images
 - virtual machines (VMs), 68–77
 - working with, 138–141, 146–147
- import command, 340
- infrastructure, building using Cloud Foundation Toolkit, 411–418
- INSERT command, 288
- instance groups
 - about, 87, 147
 - autoscaling, 147
 - creating, 147–149
 - defined, 132
 - load balancing, 147
 - removing, 147–149
- instance start command, 144
- instances
 - attaching GPUs to, 136–138
 - deleting, 132–135, 145
 - managing single virtual machine, 132–147
 - MySQL, 286–288
 - starting, 121, 132–135, 144
 - stopping, 121, 132–135, 144
 - VM, 438–439
- Integrity Monitoring, 111

IP addresses

- managing, 389–391
- reserving, 390–391

J

jobs

- about, 161
- Cloud Run, 217–218, 222–223
- viewing in BigQuery, 296–297

JSON files, 333

K

Karlton, Phil, 6

- kubectl command, 200
- kubectl delete deployment command, 202
- kubectl delete service command, 207
- kubectl describe nodes command, 192–193
- kubectl expose deployment
 - command, 206–207
- kubectl get deployments command, 201–202
- kubectl get pods command, 191–192
- kubectl get services command, 205–206
- kubectl scale command, 172
- kubelet, 159
- Kubernetes clusters
 - about, 209
 - adding
 - nodes, 193–196
 - Pods, 196–202
 - services, 203–207
 - architecture of, 159
 - creating repositories in Artifact Registry, 207–209
 - deploying, 162–168
 - exam essentials, 209
 - modifying
 - nodes, 193–196
 - Pods, 196–202
 - services, 203–207
 - removing

nodes, 193–196

Pods, 196–202

services, 203–207

review questions, 210–213, 489–491

viewing status of, 180–193

Kubernetes Engine

about, 20–22, 87–88, 158, 172–173

Anthos, 90

cluster architecture, 88–89, 159

deploying

application Pods, 168–172

Kubernetes clusters, 162–168

exam essentials, 173

functionality, 88

monitoring, 172

objects, 159–162

pinning services to top of Navigation

menu, 182–187

review questions, 174–177, 487–489

use cases, 89

L

latency

defined, 5

storage and, 277

“Latency Numbers Every Programmer Should Know” (Bonér), 6

Linux, installing Cloud SDK on, 118–119

load balancers

configuring

about, 382–383

using Cloud Console, 383–386

using gcloud command, 386–389

instance groups, 147

types of, 382–383

loading data, 288–289

loading data, into storage

about, 322, 342

to Cloud Storage

using command line, 327–328

using Google Cloud Console, 322–327

exam essentials, 342–343

- importing and exporting data
 - BigQuery, 332–337
 - from Cloud Bigtable, 339–340
 - Cloud Dataproc, 340
 - Cloud Firestore, 332
 - Cloud Spanner, 337–339
 - Cloud SQL, 328–331
- review questions, 344–347, 500–502
- streaming data to Cloud Pub/Sub, 341–342
- Log Explorer page, 459–461
- log routers, 458–459
- log sinks, 458–459
- logs, viewing and filtering, 459–461
- ls command, 307–308

M

- machine learning (ML)
 - about, 32–33
 - Apache Spark for, 312–313
- macOS, installing Cloud SDK on, 118
- managed Kubernetes clusters, 3
- management and observability tools, for
 - Google Cloud, 31–32
- Management tab (VM creation form), 109–111
- Memcached, 255–257
- Memorystore, 255–257
- message queues, decoupling services
 - using, 341–342
- Metric Explorer, 450–453
- microseconds, 254–255
- microservices, 87, 229
- milliseconds, 254–255
- MLib, 312–313
- monolithic applications, 229
- move command, 328
- multi-region/multiregional storage, 260, 261
- mv command, 314
- MySQL
 - backing up in Cloud SQL, 289–292
 - instances, 286–288

N

- namespaces, 162
- nanoseconds, 254–255
- Natural Language, 33
- Navigation menu, pinning services to top
 - of, 182–187
- Nearline storage, 260–261
- network access, to virtual machines, 121–123
- Network File System (NFS) storage system, 5
- networking
 - about, 7, 350, 368–369, 376, 391–392
 - configuring
 - Cloud DNS, 376–382
 - load balancers, 382–389
 - creating
 - firewall rules for virtual private clouds, 359–364
 - virtual private clouds with subnets, 350–357
 - virtual private networks, 364–368
 - deploying Compute Engine with custom networks, 357–359
 - exam essentials, 369, 392–393
 - Google Private Access, 389
 - managing IP addresses, 389–391
 - review questions, 370–373, 394–397, 502–506
- networking components
 - Cloud Armor, 29
 - Cloud CDN, 29
 - Cloud DNS, 30
 - Cloud Interconnect, 29–30
 - Cloud Load Balancing, 29
 - development tools, 30–31
 - of Google Cloud, 28–31
 - identity management and security, 30
 - networking services, 28–30
 - virtual private cloud (VPC), 28
- Networking tab (VM creation form), 115
- NFS (Network File System) storage system, 5
- node pools, 162
- nodes
 - about, 159

- adding, 193–196
- modifying, 193–196
- removing, 193–196

NoSQL storage data model, 270–276

O

- object storage, 4, 258–261
- object storage data model, 266
- objects (Kubernetes), 159–162
- optical character recognition (OCR), 240
- organizations
 - policies of, 45–46
 - in resource hierarchy, 43–44
- owning resources, 8–9

P

- pay-as-you-go-for-what-you-use model, 9
- peak capacity planning, 147
- peering, 7
- persistent disks, 257–258
- persistent storage, 257–258
- pod specification, 161
- pod template, 161
- pods
 - about, 160
 - adding, 196–202
 - modifying, 196–202
 - removing, 196–202
- policies, of organizations, 45–46
- preemptible virtual machines, 80–81
- Pricing Calculator
 - about, 295, 464–467
 - Cloud Logging, 469–472
 - exam essentials, 468
- principle of least privilege, 50, 432
- Private Google Access, 389
- Private Service Connect, 389
- projects
 - about, 42, 60
 - billing, 53–59

- enabling APIs, 59–60
- exam essentials, 61
- managing, 46–49
- organizing projects and accounts, 42–49
- in resource hierarchy, 44–45
- review questions, 62–66, 478–480
- roles and identities, 49–52
- service accounts, 52–53
- virtual machines (VMs) contained
 - in, 77–78

Q

- queries
 - data, 288–289
 - estimating cost of in BigQuery, 294–296

R

- RDP (Remote Desktop Protocol), 121–123
- Read and Write patterns, storage and, 277
- read command, 308
- Recommendations AI, 33
- Redis, 255–257
- regional managed instance group, 147
- regional storage, 261
- regions, running virtual machines in, 78–79
- relational databases, 256–268
- Remote Desktop Protocol (RDP), 121–123
- renting resources, 8–9
- replicas, 196
- ReplicaSets, 161
- repositories, creating in Artifact
 - Registry, 207–209
- reserving IP addresses, 390–391
- resources
 - computing, 19–23
 - constraints on, 45–46
 - hierarchy of, 42–45
 - renting compared with owning, 8–9
 - storage, 23–26
- review questions

- App Engine, 232–235, 491–493
 - cloud computing, 12–15, 474–475
 - Cloud Functions, 249–252, 494–496
 - Cloud Logging, 469–472, 511–513
 - Cloud Monitoring, 469–472, 511–513
 - Cloud Run, 232–235, 491–493
 - Compute Engine virtual machines, 127–130, 482–485
 - computing, 96–99, 480–482
 - deploying, 420–423, 507–508
 - Google Cloud, 36–40, 476–478
 - identity and access management (IAM), 443–446, 509–511
 - Kubernetes clusters, 210–213, 489–491
 - Kubernetes Engine, 174–177, 487–489
 - loading data, 344–347, 500–502
 - networking, 370–373, 394–397, 502–506
 - projects, service accounts, and billing, 62–66, 478–480
 - storage, 280–283, 317–320, 496–500
 - virtual machines (VMs), 152–155, 485–487
 - roles
 - for App Engine, 428
 - in Google Cloud, 50
 - granting to identities, 50–52
 - IAM, 428–435
 - runtime environments, for Cloud Functions, 239–240
 - billing, 53–59
 - enabling APIs, 59–60
 - exam essentials, 61
 - managing, 436–440
 - organizing projects and accounts, 42–49
 - review questions, 62–66, 478–480
 - roles and identities, 49–52
 - services
 - about, 160
 - adding, 203–207
 - Cloud Run, 216–217, 218–222
 - decoupling using message queues, 341–342
 - deployable, 415
 - modifying, 203–207
 - networking, 28–30
 - pinning to top of Navigation menu, 182–187
 - removing, 203–207
 - Snappy utility, 333
 - snapshots, working with, 138, 145–146
 - SOA (start of authority) record, 379
 - Sole-Tenancy tab (VM creation form), 115–117
 - specialized services
 - about, 8, 10
 - for Google Cloud, 32–33
 - sql connect command, 287–288
 - standard environment, App Engine and, 22, 85–86, 87
 - start command, 144
 - start of authority (SOA) record, 379
 - stateful firewalls, 359–360
 - StatefulSets, 161
 - static IP addresses, 389
 - stop command, 144
 - storage
 - about, 4, 254, 278, 315
 - block storage, 5
 - caches, 5–6, 255–257
 - classes, 260–261
 - data models for, 265–276
 - deploying and managing
-
- S**
- scaling App Engine applications, 228–229
 - scopes, managing service accounts with, 436–438
 - searchable documents, 240
 - Secure Boot, 111
 - Security tab (VM creation form), 111–112
 - SELECT command, 288–289
 - separation of duties, IAM roles and, 432
 - serverless computing, 4
 - service accounts
 - about, 42, 52–53, 60

- BigQuery, 294–297
- Cloud Bigtable, 304–308
- Cloud Dataproc, 308–313
- Cloud Firestore, 292–294
- Cloud Pub/Sub, 300–304
- Cloud Spanner, 297–300
- Cloud SQL, 286–292
 - in Google Cloud, 286–320
- dual regional, 261
- exam essentials, 278–279, 315–316
- file storage, 5
- guidelines for choosing, 277–278
- loading data into (*See* loading data, into storage)
- managing Cloud Storage, 314–315
- multiregional/multi-regional, 261
- object, 258–261
- object storage, 4
- persistent, 257–258
- regional, 261
- review questions, 280–283, 317–320, 496–500
- types of, 264–265
- types of systems for, 254–265
- versioning, 262
- storage components
 - about, 23
 - Cloud Bigtable, 26–27
 - Cloud Firestore, 25–26, 27
 - Cloud Memorystore, 27–28
 - Cloud Spanner, 27
 - Cloud SQL, 26
 - cloud storage, 24–25
 - cloud storage for Firebase, 25
 - databases, 26–28
 - of Google Cloud, 23–28
 - multi-region storage, 24
 - persistent disks, 25
 - storage resources, 23–26
- streaming data, to Cloud Pub/Sub, 341
- subnets, creating virtual private cloud (VPC)
 - with, 350–357

T

- target pools, 389
- template files, in Deployment Manager, 414
- templates, creating and removing, 147–149
- tools, development, 30–31
- topics, 239
- transaction support, storage and, 277
- Translation AI, 33
- triggers, Cloud Functions, 238–239

U

- Ubuntu Linux, installing Cloud SDK
 - on, 118–119
- use cases
 - for App Engine, 86–87
 - Cloud Functions, 93
 - Cloud Run, 91
 - for Compute Engine virtual machines, 82–83
 - Kubernetes Engine, 89
- users, needing privileges to create virtual machines, 79–80

V

- versioning, 262
- virtual machines (VMs). *See also* Compute Engine virtual machines
 - about, 3, 150
 - contained in projects, 77–78
 - custom types, 81–82
 - exam essentials, 151
 - guidelines for managing, 150
 - images, 68–77
 - instance groups, 147–149
 - managing, 132–155
 - managing single instances, 132–147
 - preemptible, 80–81

- review questions, 152–155, 485–487
- running in zones and regions, 78–79
- use cases for, 82–83
- users needing privileges to create, 79–80
- viewing inventory of, 136, 145

virtual private cloud (VPC)

- about, 28
- creating
 - firewall rules for, 359–364
 - with `gcloud` command, 354–355
 - with Google Cloud Console, 350–354
 - shared, with `gcloud`
 - command, 355–357
 - with subnets, 350–357

virtual private network (VPN),

- creating, 364–368

Virtual Trusted Platform (vTPM), 111

Vision AI, 33

VM instances, assigning service accounts to, 438–439

VM Provisioning Model, 111

VMs (virtual machines). *See also* Compute Engine virtual machines

- about, 3, 150
- contained in projects, 77–78
- custom types, 81–82
- exam essentials, 151
- guidelines for managing, 150
- images, 68–77
- instance groups, 147–149
- managing, 132–155
- managing single instances, 132–147
- preemptible, 80–81
- review questions, 152–155, 485–487

- running in zones and regions, 78–79
- use cases for, 82–83
- users needing privileges to create, 79–80
- viewing inventory of, 136, 145

volumes, 161–162

VPC (virtual private cloud)

- about, 28
- creating
 - firewall rules for, 359–364
 - with `gcloud` command, 354–355
 - with Google Cloud Console, 350–354
 - shared, with `gcloud`
 - command, 355–357
 - with subnets, 350–357

VPN (virtual private network),

- creating, 364–368

vTPM (Virtual Trusted Platform), 111

W

websites

- Cloud Dataflow documentation, 339
- Cloud Functions documentation, 240
- Google Cloud Console, 46
- Google Cloud documentation, 428
- Pricing Calculator, 295

Windows, installing Cloud SDK on, 118

Workload Identity, 418

Z

zonal managed instance group, 147

zones, running virtual machines in, 78–79

Comprehensive Online Learning Environment

Register to gain one year of FREE access after activation to the online interactive test bank to help you study for your Google Cloud Associate Cloud Engineer certification exam—including with your purchase of this book!

The online test bank includes the following:

- **Assessment Test** to help you focus your study on specific objectives
- **Chapter Tests** to reinforce what you've learned
- **Practice Exams** to test your knowledge of the material
- **Digital Flashcards** to reinforce your learning and provide last-minute test prep before the exam
- **Searchable Glossary** to define the key terms you'll need to know for the exam

Register and Access the Online Test Bank

To register your book and get access to the online test bank, follow these steps:

1. Go to www.wiley.com/go/sybextestprep (this address is case sensitive)! You'll see the "How to Register Your Book for Online Access" instructions.
2. Click "Click here to register" and then select your book from the list.
3. Complete the required registration information, including answering the security verification to prove book ownership. You will be emailed a pin code.
4. Follow the directions in the email or go to www.wiley.com/go/sybextestprep.

5. Find your book on that page and click the “Register or Login” link with it. Then enter the pin code you received and click the “Activate PIN” button.
6. On the Create an Account or Login page, enter your username and password, and click Login or, if you don’t have an account already, create a new account.
7. At this point, you should be in the test bank site with your new test bank listed at the top of the page. If you do not see it there, please refresh the page or log out and log back in.

WILEY END USER LICENSE AGREEMENT

Go to www.wiley.com/go/eula to access Wiley's ebook EULA.