CertyIQ


# CertyIQ

## Premium exam material

Get certification quickly with the CertyIQ Premium exam material.
Everything you need to prepare, learn & pass your certification exam easily. Lifetime free updates
First attempt guaranteed success.

https://www.CertyIQ.com

# About CertyIQ

We here at CertyIQ eventually got enough of the industry's greedy exam paid for. Our team of IT professionals comes with years of experience in the IT industry Prior to training CertIQ we worked in test areas where we observed the horrors of the paywall exam preparation system.

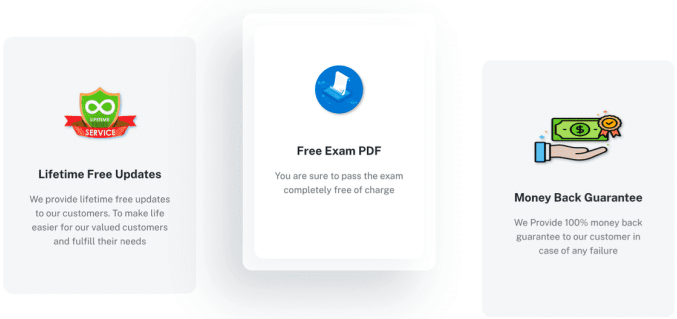
The misuse of the preparation system has left our team disillusioned. And for that reason, we decided it was time to make a difference. We had to make In this way, CertyIQ was created to provide quality materials without stealing from everyday people who are trying to make a living.

# Doubt Support

We have developed a very scalable solution using which we are able to solve 400+ doubts every single day with an average rating of 4.8 out of 5.

https://www.certyiq.com

Mail us on - certyiqofficial@gmail.com




**John**

October 19, 2022

★★★★★

Thanks you so much for your help. I scored 972 in my exam today. More than 90% were from your PDFs!

**Dana**

September 04, 2022

★★★★★

Thanks a lot for this updated AZ-900 Q&A. I just passed my exam and got 974, I followed both of your Az-900 videos and the 6 PDF, the PDFs are very much valid, all answers are correct. Could you please create a similar video/PDF for DP900, your content/PDF's is really awesome. The team did a really good job. Thank You 😊.

**Ahamed Shibly**

2 months ago

★★★★★

Customer support is realy fast and helpful, I just finished my exam and this video along with the 6 PDF helped me pass! Definitely recommend getting the PDFs. Thank you!

October 22, 2022

★★★★★

Passed my exam today with 891 marks. Out of 52 questions, 51 were from certyiq PDFs including Contoso case study.
Thank You certyiq team!

**Henry Rome**

2 months ago

★★★★★

These questions are real and 100 % valid. Thank you so much for your efforts, also your 4 PDFs are awesome, I passed the DP900 exam on 1 Sept. With 968 marks. Thanks a lot, buddy!

**Esmaria**

2 months ago

★★★★★

Simple easy to understand explanations. To anyone out there wanting to write AZ900, I highly recommend 6 PDF's.Thank you so much, appreciate all your hard work in having such great content. Passed my exam Today - 3 September with 942 score.

# Google

(Associate Cloud Engineer)

Associate Cloud Engineer

Total: **291 Questions**

Link: https://certyiq.com/papers/google/associate-cloud-engineer

**Question: 1** CertyIQ

Every employee of your company has a Google account. Your operational team needs to manage a large number of instances on Compute Engine. Each member of this team needs only administrative access to the servers. Your security team wants to ensure that the deployment of credentials is operationally efficient and must be able to determine who accessed a given instance. What should you do?

A. Generate a new SSH key pair. Give the private key to each member of your team. Configure the public key in the metadata of each instance.

B. Ask each member of the team to generate a new SSH key pair and to send you their public key. Use a configuration management tool to deploy those keys on each instance.

C. Ask each member of the team to generate a new SSH key pair and to add the public key to their Google account. Grant the compute.osAdminLogin role to the Google group corresponding to this team.

D. Generate a new SSH key pair. Give the private key to each member of your team. Configure the public key as a project-wide public SSH key in your Cloud Platform project and allow project-wide public SSH keys on each instance.

**Answer: C**

**Explanation:**

Option C is the most suitable solution because it leverages Google Accounts for identity management and the compute.osAdminLogin role for access control, fulfilling both operational efficiency and security requirements. By adding individual SSH public keys to each team member's Google account, Google automatically manages key distribution to instances upon their first SSH connection. Granting the compute.osAdminLogin role to the team's Google Group ensures that members have the necessary administrative privileges to the instances. This approach provides individual accountability through audit logs linked to Google accounts, allowing the security team to track instance access. Options A and D introduce security risks with shared private keys, while option B involves manual key management and deployment, making Option C the most streamlined and secure solution.

**Supporting Links:**

**Managing instance access using OS Login:** https://cloud.google.com/compute/docs/oslogin/setup/enable-os-login
**Granting the compute.osAdminLogin role:** https://cloud.google.com/compute/docs/oslogin/configure-os-login#grant_the_required_iam_roles
**Understanding OS Login and its benefits:** https://cloud.google.com/compute/docs/oslogin

**Question: 2** CertyIQ

You need to create a custom VPC with a single subnet. The subnet's range must be as large as possible. Which range should you use?

A. 0.0.0.0/0

B. 10.0.0.0/8

C. 172.16.0.0/12

D. 192.168.0.0/16

**Answer: B**

**Explanation:**

Option B, **10.0.0.0/8**, is the correct answer because it represents the largest possible subnet range among the options provided due to its CIDR notation. A /8 prefix allocates more IP addresses (approximately 16 million)

compared to /12 or /16 prefixes, which are smaller network ranges. The smaller the CIDR suffix number, the larger the network and the number of available IP addresses within that range. Option A, 0.0.0.0/0, is not a valid subnet range but rather represents the default route, encompassing all networks. Options C (172.16.0.0/12) and D (192.168.0.0/16) are valid private IP ranges, but they are smaller in size than 10.0.0.0/8. For further understanding of VPC subnet ranges and CIDR notation, consult the official Google Cloud documentation on VPC networks: https://cloud.google.com/vpc/docs/vpc and https://cloud.google.com/vpc/docs/subnet-networks.

**Question: 3** CertyIQ

You want to select and configure a cost-effective solution for relational data on Google Cloud Platform. You are working with a small set of operational data in one geographic location. You need to support point-in-time recovery. What should you do?

A. Select Cloud SQL (MySQL). Verify that the enable binary logging option is selected.

B. Select Cloud SQL (MySQL). Select the create failover replicas option.

C. Select Cloud Spanner. Set up your instance with 2 nodes.

D. Select Cloud Spanner. Set up your instance as multi-regional.

**Answer: A**

**Explanation:**

Rationale: Cloud SQL for MySQL is a cost-effective, managed relational database service suitable for small datasets and single geographic locations. Enabling binary logging in Cloud SQL (MySQL) is essential for point-in-time recovery, allowing you to restore your database to a specific point in time. Cloud Spanner, while offering point-in-time recovery, is designed for global scale and higher availability, making it more expensive and less cost-effective for a small, single-region dataset. Failover replicas in Cloud SQL enhance availability but are not the primary mechanism for point-in-time recovery, and choosing them without binary logging doesn't fulfill the requirement.

Supporting Links:

1. **Cloud SQL Point-in-Time Recovery:** https://cloud.google.com/sql/docs/mysql/backup-recovery/point-in-time-recovery
2. **Cloud SQL Pricing:** https://cloud.google.com/sql/docs/mysql/pricing
3. **Cloud Spanner Pricing:** https://cloud.google.com/spanner/docs/pricing

**Question: 4** CertyIQ

You want to configure autohealing for network load balancing for a group of Compute Engine instances that run in multiple zones, using the fewest possible steps.
You need to configure re-creation of VMs if they are unresponsive after 3 attempts of 10 seconds each. What should you do?

A. Create an HTTP load balancer with a backend configuration that references an existing instance group. Set the health check to healthy (HTTP)

B. Create an HTTP load balancer with a backend configuration that references an existing instance group. Define a balancing mode and set the maximum RPS to 10.

C. Create a managed instance group. Set the Autohealing health check to healthy (HTTP)

D. Create a managed instance group. Verify that the autoscaling setting is on.

**Answer: C**

**Explanation:**

Rationale: To configure autohealing with the fewest steps, creating a Managed Instance Group (MIG) and setting its autohealing policy is the most direct approach because MIGs are specifically designed to automatically replace unhealthy instances based on health checks, fulfilling the requirement of VM re-creation after unresponsiveness. This method directly addresses autohealing without requiring additional components like load balancers for this specific purpose. Options A and B involve load balancers, which are not the most direct way to achieve autohealing for instance groups, and option D focuses on autoscaling, a separate feature from autohealing.

Supporting Links:

Managed Instance Groups Documentation: Explains the purpose and features of Managed Instance Groups, including autohealing.
Autohealing Instances in a MIG: Details how to configure autohealing for Managed Instance Groups.

**Question: 5** CertyIQ

You are using multiple configurations for gcloud. You want to review the configured Kubernetes Engine cluster of an inactive configuration using the fewest possible steps. What should you do?

A. Use gcloud config configurations describe to review the output.
B. Use gcloud config configurations activate and gcloud config list to review the output.
C. Use kubectl config get-contexts to review the output.
D. Use kubectl config use-context and kubectl config view to review the output.

**Answer: D**

**Explanation:**

**Justification:** Option D is the correct answer because to review the Kubernetes Engine cluster configuration through kubectl, you must first select the appropriate Kubernetes context, which can be done using kubectl config use-context <context-name>. Subsequently, using kubectl config view displays the entire kubectl configuration file, including details about the currently used or specified context and its associated Kubernetes cluster, thus allowing for review of the configured cluster. This two-command approach is the most direct way to inspect Kubernetes cluster details using kubectl commands for a potentially inactive gcloud configuration's context.

**Supporting Links:**

**kubectl config use-context:** https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands/#config
**kubectl config view:** https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands/#config

**Question: 6** CertyIQ

Your company uses Cloud Storage to store application backup files for disaster recovery purposes. You want to follow Google's recommended practices. Which storage option should you use?

A. Multi-Regional Storage
B. Regional Storage
C. Nearline Storage

D. Coldline Storage

**Answer: D**

**Explanation:**

**Justification:** Coldline Storage is the recommended option for disaster recovery backups due to its lowest storage cost and suitability for infrequently accessed data, aligning with Google's best practices for cost optimization. Disaster recovery backups are typically accessed infrequently, making Coldline's higher latency and retrieval costs a reasonable trade-off for significant cost savings. Google recommends Coldline and Archive Storage for backup and archival scenarios where data access is infrequent and cost is a primary concern. For disaster recovery, data durability and cost-effectiveness are prioritized over frequent and fast access, making Coldline an ideal choice. Regional and Multi-Regional Storage are designed for frequently accessed data and are more expensive options not optimized for infrequent disaster recovery needs. Nearline Storage, while cheaper than Regional, is still more expensive than Coldline and intended for more frequent access than typical disaster recovery backups.

**Supporting Links:**

**Cloud Storage pricing:** https://cloud.google.com/storage/pricing (Illustrates the cost difference between storage classes, highlighting Coldline's low cost.)
**Storage classes:** https://cloud.google.com/storage/docs/storage-classes (Explains the use cases for each storage class, including Coldline for infrequently accessed data like backups.)
**Choosing a Cloud Storage option:** https://cloud.google.com/storage/docs/choose-storage-options (Provides guidance on selecting the appropriate storage class based on access frequency and cost considerations.)
**Disaster Recovery Planning Guide:** https://cloud.google.com/solutions/disaster-recovery/ (Google Cloud's documentation on disaster recovery, implicitly suggesting cost-effective storage for backups.)

**Question: 7** CertyIQ

Several employees at your company have been creating projects with Cloud Platform and paying for it with their personal credit cards, which the company reimburses. The company wants to centralize all these projects under a single, new billing account. What should you do?

A. Contact [email protected] with your bank account details and request a corporate billing account for your company.
B. Create a ticket with Google Support and wait for their call to share your credit card details over the phone.
C. In the Google Platform Console, go to the Resource Manage and move all projects to the root Organizarion.
D. In the Google Cloud Platform Console, create a new billing account and set up a payment method.

**Answer: D**

**Explanation:**

**Justification:** To centralize billing, the most direct approach is to create a new billing account within the Google Cloud Platform Console and configure a corporate payment method, as described in option D; this allows the company to consolidate all project costs under a single, centrally managed billing account, as further detailed in Google Cloud documentation on managing billing accounts.

**Supporting Links:**

**Create a Cloud Billing account:** https://cloud.google.com/billing/docs/how-to/manage-billing-account

**Question: 8** CertyIQ

You have an application that looks for its licensing server on the IP 10.0.3.21. You need to deploy the licensing server on Compute Engine. You do not want to change the configuration of the application and want the application to be able to reach the licensing server. What should you do?

A. Reserve the IP 10.0.3.21 as a static internal IP address using gcloud and assign it to the licensing server.

B. Reserve the IP 10.0.3.21 as a static public IP address using gcloud and assign it to the licensing server.

C. Use the IP 10.0.3.21 as a custom ephemeral IP address and assign it to the licensing server.

D. Start the licensing server with an automatic ephemeral IP address, and then promote it to a static internal IP address.

**Answer: A**

**Explanation:**

The correct answer is A because reserving the IP address 10.0.3.21 as a static internal IP and assigning it to the licensing server ensures that the server is reachable at the exact IP address expected by the application, without requiring any changes to the application's configuration. This method directly fulfills the requirement of maintaining the application's existing IP dependency. Options B, C, and D are incorrect because using a public IP (B) is unnecessary for internal communication, ephemeral IPs (C and D) are not static and can change, and promoting an ephemeral IP to static (D) doesn't guarantee the specific IP address 10.0.3.21 from the beginning.

**Supporting Links:**

**Static internal IP addresses:** https://cloud.google.com/compute/docs/ip-addresses/reserve-static-internal-ip-address
**Internal IP addresses in Compute Engine:** https://cloud.google.com/compute/docs/ip-addresses#internaladdresses

**Question: 9** CertyIQ

You are deploying an application to App Engine. You want the number of instances to scale based on request rate. You need at least 3 unoccupied instances at all times. Which scaling type should you use?

A. Manual Scaling with 3 instances.

B. Basic Scaling with min_instances set to 3.

C. Basic Scaling with max_instances set to 3.

D. Automatic Scaling with min_idle_instances set to 3.

**Answer: D**

**Explanation:**

**Justification:** Automatic Scaling with min_idle_instances set to 3 is the correct choice because it configures App Engine to dynamically adjust the number of instances based on request rate while ensuring that at least 3 instances are always kept idle and ready to serve incoming requests immediately, meeting both scaling and unoccupied instance requirements. Manual scaling does not scale automatically, Basic scaling's min_instances refers to the minimum instances started upon traffic, not idle ones, and Basic scaling's max_instances limits the total instances, not idle instances.

**Supporting Links:**

1. **App Engine scaling types:** https://cloud.google.com/appengine/docs/standard/python3/how-

instances-are-managed#scaling_types

2. **Automatic scaling settings:**
   https://cloud.google.com/appengine/docs/standard/python3/config/appref#automatic_scaling
   (Specifically look for min_idle_instances description)

**Question: 10** CertyIQ

You have a development project with appropriate IAM roles defined. You are creating a production project and want to have the same IAM roles on the new project, using the fewest possible steps. What should you do?

A. Use gcloud iam roles copy and specify the production project as the destination project.

B. Use gcloud iam roles copy and specify your organization as the destination organization.

C. In the Google Cloud Platform Console, use the 'create role from role' functionality.

D. In the Google Cloud Platform Console, use the 'create role' functionality and select all applicable permissions.

**Answer: A**

**Explanation:**

To replicate IAM roles in a new production project with the fewest steps, utilize the gcloud iam roles copy command, designating the production project as the destination. This command efficiently copies custom IAM roles directly from the development project to the production project. Using gcloud iam roles copy minimizes manual configuration, streamlining the process compared to console-based methods or broader organizational approaches. This command directly duplicates the role definitions and associated permissions, ensuring consistency between projects. For more details on gcloud iam roles copy, refer to the official documentation. https://cloud.google.com/sdk/gcloud/reference/iam/roles/copy

**Question: 11** CertyIQ

You need a dynamic way of provisioning VMs on Compute Engine. The exact specifications will be in a dedicated configuration file. You want to follow Google's recommended practices. Which method should you use?

A. Deployment Manager

B. Cloud Composer

C. Managed Instance Group

D. Unmanaged Instance Group

**Answer: A**

**Explanation:**

Deployment Manager is the recommended method because it is Google's Infrastructure-as-Code (IaC) service, allowing you to define VM specifications in declarative configuration files (YAML or Python). This approach enables dynamic provisioning by simply modifying the configuration files to adjust VM parameters. Deployment Manager facilitates repeatable and consistent infrastructure deployments, aligning with Google's best practices for managing cloud resources. It directly addresses the requirement of using configuration files for VM specifications and offers a dynamic and automated way to provision Compute Engine VMs. In contrast, Cloud Composer is for workflow orchestration, while Managed and Unmanaged Instance Groups are focused on managing groups of VMs rather than provisioning individual VMs based on varied configurations. For further research, refer to the Deployment Manager documentation

https://cloud.google.com/deployment-manager/docs/ and Google's best practices for Infrastructure as Code on Google Cloud https://cloud.google.com/solutions/infrastructure-as-code.

**Question: 12** CertyIQ

You have a Dockerfile that you need to deploy on Kubernetes Engine. What should you do?

A. Use kubectl app deploy <dockerfilename>.

B. Use gcloud app deploy <dockerfilename>.

C. Create a docker image from the Dockerfile and upload it to Container Registry. Create a Deployment YAML file to point to that image. Use kubectl to create the deployment with that file.

D. Create a docker image from the Dockerfile and upload it to Cloud Storage. Create a Deployment YAML file to point to that image. Use kubectl to create the deployment with that file.

**Answer: C**

**Explanation:**

The correct answer is **C**. To deploy a Dockerfile on Kubernetes Engine, you must first build a Docker image from it and push this image to a container registry like Google Container Registry. Kubernetes then uses a Deployment YAML file to define how to run this image, pulling it from the registry and deploying it onto the cluster using kubectl. Option A is incorrect because kubectl app deploy is not a valid kubectl command for deploying Dockerfiles directly. Option B is incorrect as gcloud app deploy is used for Google App Engine deployments, not Kubernetes Engine. Option D is less suitable because while Cloud Storage can store files, Container Registry is the recommended and integrated service for storing and managing container images for use with Kubernetes Engine.

**Supporting Links:**

1. **Building Docker Images:** https://docs.docker.com/build/
2. **Google Container Registry:** https://cloud.google.com/container-registry
3. **Kubernetes Deployments:** https://kubernetes.io/docs/concepts/workloads/controllers/deployment/
4. **kubectl apply command:** https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands/#apply

**Question: 13** CertyIQ

Your development team needs a new Jenkins server for their project. You need to deploy the server using the fewest steps possible. What should you do?

A. Download and deploy the Jenkins Java WAR to App Engine Standard.

B. Create a new Compute Engine instance and install Jenkins through the command line interface.

C. Create a Kubernetes cluster on Compute Engine and create a deployment with the Jenkins Docker image.

D. Use GCP Marketplace to launch the Jenkins solution.

**Answer: D**

**Explanation:**

GCP Marketplace offers pre-built solutions like Jenkins, simplifying deployment with minimal manual configuration, thus requiring the fewest steps compared to manually setting up Jenkins on Compute Engine, App Engine, or Kubernetes. Launching Jenkins from the GCP Marketplace pre-configures the necessary

infrastructure and software, significantly reducing the complexity and time involved in setting up a Jenkins server. This streamlined approach allows users to quickly deploy a functional Jenkins environment with just a few clicks, fulfilling the requirement of deploying with the fewest steps.

**Supporting Links:**

1. **GCP Marketplace Overview:** https://cloud.google.com/marketplace - Provides general information about GCP Marketplace and its benefits.
2. **Jenkins Solution in GCP Marketplace:** https://console.cloud.google.com/marketplace/details/jenkinsx/jenkins - Direct link to the Jenkins solution in GCP Marketplace, demonstrating its availability.
3. **Deploying from Marketplace:** https://cloud.google.com/marketplace/docs/deploying - Documentation on how to deploy solutions from GCP Marketplace, showcasing the simplified process.

**Question: 14** CertyIQ

You need to update a deployment in Deployment Manager without any resource downtime in the deployment. Which command should you use?

A. gcloud deployment-manager deployments create --config <deployment-config-path>

B. gcloud deployment-manager deployments update --config <deployment-config-path>

C. gcloud deployment-manager resources create --config <deployment-config-path>

D. gcloud deployment-manager resources update --config <deployment-config-path>

**Answer: B**

**Explanation:**

The command gcloud deployment-manager deployments update --config <deployment-config-path> is used to modify an existing Deployment Manager deployment configuration, facilitating updates without service interruption. Deployment Manager orchestrates updates in a controlled manner, aiming for minimal to zero downtime depending on the resources and configuration changes. In contrast, create is for initiating new deployments, and resources create or resources update are focused on individual resources rather than the entire deployment update. Therefore, update command is the correct choice for modifying a deployment while minimizing downtime.

**Supported Links:**

gcloud deployment-manager deployments update

**Question: 15** CertyIQ

You need to run an important query in BigQuery but expect it to return a lot of records. You want to find out how much it will cost to run the query. You are using on-demand pricing. What should you do?

A. Arrange to switch to Flat-Rate pricing for this query, then move back to on-demand.

B. Use the command line to run a dry run query to estimate the number of bytes read. Then convert that bytes estimate to dollars using the Pricing Calculator.

C. Use the command line to run a dry run query to estimate the number of bytes returned. Then convert that bytes estimate to dollars using the Pricing Calculator.

D. Run a select count (*) to get an idea of how many records your query will look through. Then convert that number of rows to dollars using the Pricing Calculator.

**Answer: B**

**Explanation:**

The correct answer is **B**. Here's why:

To estimate the cost of a BigQuery query using on-demand pricing, you should use a dry run to determine the estimated bytes read by the query, as on-demand pricing is based on the volume of data processed. A dry run allows you to get this estimate without actually running the query and incurring charges for the full data processing. You can then use the Google Cloud Pricing Calculator to convert the estimated bytes read into a dollar amount based on the on-demand pricing rates. Options A, C, and D are incorrect because switching to flat-rate pricing is irrelevant for on-demand cost estimation (A), bytes returned are not the basis for on-demand pricing (C), and SELECT COUNT(*) actually runs a query and uses row count instead of bytes processed for cost estimation (D).

**Supporting Links:**

**BigQuery pricing:** https://cloud.google.com/bigquery/pricing - This page explains the BigQuery pricing models, including on-demand pricing which is based on data processed.
**Run a dry run query:** https://cloud.google.com/bigquery/docs/dry-run-queries - This documentation details how to use dry run queries in BigQuery to estimate query cost before execution.
**Google Cloud Pricing Calculator:** https://cloud.google.com/products/calculator - This is the official tool to calculate the cost of Google Cloud services, including BigQuery, based on usage parameters like bytes processed.

**Question: 16** CertyIQ

You have a single binary application that you want to run on Google Cloud Platform. You decided to automatically scale the application based on underlying infrastructure CPU usage. Your organizational policies require you to use virtual machines directly. You need to ensure that the application scaling is operationally efficient and completed as quickly as possible. What should you do?

A. Create a Google Kubernetes Engine cluster, and use horizontal pod autoscaling to scale the application.

B. Create an instance template, and use the template in a managed instance group with autoscaling configured.

C. Create an instance template, and use the template in a managed instance group that scales up and down based on the time of day.

D. Use a set of third-party tools to build automation around scaling the application up and down, based on Stackdriver CPU usage monitoring.

**Answer: B**

**Explanation:**

The correct answer is **B**. Here's the justification: Managed Instance Groups (MIGs) in Google Cloud Platform are designed to automatically scale virtual machines based on metrics like CPU utilization. By creating an instance template and using it within a MIG configured with autoscaling, you can efficiently and quickly scale your application's VMs up or down in response to CPU load. This approach leverages native GCP services for autoscaling, making it operationally efficient and faster compared to manual or third-party solutions. Instance templates ensure consistent VM configurations for scaling, further streamlining the process. Options A, C, and D are less optimal because A introduces containerization which might be an overhead for a single binary application running directly on VMs as requested, C utilizes time-based scaling instead of CPU-based autoscaling, and D adds complexity by requiring third-party tools when GCP provides native autoscaling within MIGs.

**Supporting Links:**

**Managed Instance Groups (MIGs):** https://cloud.google.com/compute/docs/instance-groups/
**Autoscaling MIGs:** https://cloud.google.com/compute/docs/instance-groups/autoscaling/
**Instance Templates:** https://cloud.google.com/compute/docs/instance-templates/

**Question: 17** CertyIQ

You are analyzing Google Cloud Platform service costs from three separate projects. You want to use this information to create service cost estimates by service type, daily and monthly, for the next six months using standard query syntax. What should you do?

A. Export your bill to a Cloud Storage bucket, and then import into Cloud Bigtable for analysis.
B. Export your bill to a Cloud Storage bucket, and then import into Google Sheets for analysis.
C. Export your transactions to a local file, and perform analysis with a desktop tool.
D. Export your bill to a BigQuery dataset, and then write time window-based SQL queries for analysis.

**Answer: D**

**Explanation:**

The correct answer is **D**. Here's the justification: Exporting your Cloud Billing data to BigQuery is the recommended approach for analyzing GCP service costs because BigQuery is a fully-managed, serverless data warehouse designed for large-scale data analysis and supports standard SQL queries, allowing you to efficiently analyze cost data from multiple projects, aggregate costs by service type, and perform time-window based analysis to generate daily and monthly cost estimates as required. This method leverages BigQuery's analytical capabilities for complex queries and large datasets which are typical in cost analysis. In contrast, options A, B, and C are less efficient or unsuitable for handling large datasets and performing complex analytical queries with standard SQL syntax for cost estimation and forecasting.

**Supporting Links:**

**Export Cloud Billing data to BigQuery:** https://cloud.google.com/billing/docs/how-to/export-data-bigquery
**BigQuery Documentation:** https://cloud.google.com/bigquery/docs
**Analyzing your billing data with BigQuery:** https://cloud.google.com/billing/docs/how-to/analyze-billing-data-bigquery

**Question: 18** CertyIQ

You need to set up a policy so that videos stored in a specific Cloud Storage Regional bucket are moved to Coldline after 90 days, and then deleted after one year from their creation. How should you set up the policy?

A. Use Cloud Storage Object Lifecycle Management using Age conditions with SetStorageClass and Delete actions. Set the SetStorageClass action to 90 days and the Delete action to 275 days (365 " 90)
B. Use Cloud Storage Object Lifecycle Management using Age conditions with SetStorageClass and Delete actions. Set the SetStorageClass action to 90 days and the Delete action to 365 days.
C. Use gsutil rewrite and set the Delete action to 275 days (365-90).
D. Use gsutil rewrite and set the Delete action to 365 days.

**Answer: B**

**Explanation:**

Option B is the correct answer because Cloud Storage Object Lifecycle Management using Age conditions

with SetStorageClass and Delete actions is the appropriate method to manage object lifecycles based on age, and setting SetStorageClass to 90 days moves objects to Coldline after 90 days, while setting Delete to 365 days ensures deletion one year after creation as required. Cloud Storage Object Lifecycle Management Documentation details how to automate storage class transitions and deletions based on object age, fulfilling the question's requirements directly. Option A is incorrect because setting the Delete action to 275 days would result in deletion after 275 days from the Coldline transition, which is not one year from creation. Options C and D are incorrect as gsutil rewrite is not the intended tool for setting up automated lifecycle policies; Object Lifecycle Management is the dedicated feature for this purpose.

**Question: 19** CertyIQ

You have a Linux VM that must connect to Cloud SQL. You created a service account with the appropriate access rights. You want to make sure that the VM uses this service account instead of the default Compute Engine service account. What should you do?

A. When creating the VM via the web console, specify the service account under the 'Identity and API Access' section.

B. Download a JSON Private Key for the service account. On the Project Metadata, add that JSON as the value for the key compute-engine-service- account.

C. Download a JSON Private Key for the service account. On the Custom Metadata of the VM, add that JSON as the value for the key compute-engine- service-account.

D. Download a JSON Private Key for the service account. After creating the VM, ssh into the VM and save the JSON under ~/.gcloud/compute-engine-service- account.json.

**Answer: A**

**Explanation:**

Option A is the correct answer because specifying the service account under the 'Identity and API Access' section during VM creation in the web console is the most straightforward and recommended method to ensure the VM uses the designated service account instead of the default Compute Engine service account. This approach directly associates the service account with the VM instance at creation time, ensuring all API calls from the VM are authenticated using the specified service account's credentials.
https://cloud.google.com/compute/docs/access/create-enable-service-accounts-for-instances
https://cloud.google.com/iam/docs/service-accounts-compute

**Question: 20** CertyIQ

You created an instance of SQL Server 2017 on Compute Engine to test features in the new version. You want to connect to this instance using the fewest number of steps. What should you do?

A. Install a RDP client on your desktop. Verify that a firewall rule for port 3389 exists.

B. Install a RDP client in your desktop. Set a Windows username and password in the GCP Console. Use the credentials to log in to the instance.

C. Set a Windows password in the GCP Console. Verify that a firewall rule for port 22 exists. Click the RDP button in the GCP Console and supply the credentials to log in.

D. Set a Windows username and password in the GCP Console. Verify that a firewall rule for port 3389 exists. Click the RDP button in the GCP Console, and supply the credentials to log in.

**Answer: B**

**Explanation:**

The correct answer is **B**. Here's the justification:

Option B offers the fewest steps to connect to your SQL Server 2017 instance. You simply need to have an RDP client on your desktop, which is a standard tool for Windows environments. Setting a Windows username and password in the GCP Console is a quick and direct way to establish credentials for your instance. Once you have these credentials, you can use your RDP client to connect to the instance's external IP address and log in using the created username and password. This method bypasses extra steps like manually configuring firewall rules or relying solely on the GCP Console's RDP button, making it the most straightforward approach.

Google Compute Engine documentation on connecting to Windows instances generally outlines setting a Windows password in the GCP console and using an RDP client as a standard method to connect, supporting option B's approach.

**Question: 21** CertyIQ

You have one GCP account running in your default region and zone and another account running in a non-default region and zone. You want to start a new
Compute Engine instance in these two Google Cloud Platform accounts using the command line interface. What should you do?

A. Create two configurations using gcloud config configurations create [NAME]. Run gcloud config configurations activate [NAME] to switch between accounts when running the commands to start the Compute Engine instances.

B. Create two configurations using gcloud config configurations create [NAME]. Run gcloud configurations list to start the Compute Engine instances.

C. Activate two configurations using gcloud configurations activate [NAME]. Run gcloud config list to start the Compute Engine instances.

D. Activate two configurations using gcloud configurations activate [NAME]. Run gcloud configurations list to start the Compute Engine instances.

**Answer: A**

**Explanation:**

**Justification:** To manage multiple GCP accounts effectively using the gcloud CLI, create separate configurations with gcloud config configurations create [NAME] for each account, allowing you to store distinct project settings and credentials; switch between these account contexts by activating the corresponding configuration with gcloud config configurations activate [NAME] before executing gcloud compute instances create commands, ensuring instances are launched in the intended GCP account and specified region/zone.

**Supporting Links:**

**Creating configurations:** https://cloud.google.com/sdk/gcloud/reference/config/configurations/create
**Activating configurations:** https://cloud.google.com/sdk/gcloud/reference/config/configurations/activate
**Managing gcloud CLI configurations:** https://cloud.google.com/sdk/docs/configurations
**Creating Compute Engine instances:**
https://cloud.google.com/sdk/gcloud/reference/compute/instances/create

**Question: 22** CertyIQ

You significantly changed a complex Deployment Manager template and want to confirm that the dependencies of all defined resources are properly met before committing it to the project. You want the most rapid feedback on your changes. What should you do?

A. Use granular logging statements within a Deployment Manager template authored in Python.

B. Monitor activity of the Deployment Manager execution on the Stackdriver Logging page of the GCP Console.

C. Execute the Deployment Manager template against a separate project with the same configuration, and monitor for failures.

D. Execute the Deployment Manager template using the "-preview option in the same project, and observe the state of interdependent resources.

**Answer: D**

**Explanation:**

The correct answer is **D**.

**Justification:**Executing Deployment Manager with the -preview option provides the most rapid feedback to confirm dependencies are met in a changed template because it simulates the deployment process and displays potential issues without actual deployment, thus allowing quick validation of resource dependencies before committing the changes to the project. This preview mode allows you to observe the intended state of interdependent resources and identify potential dependency conflicts or errors before any resources are created or modified in your actual project.

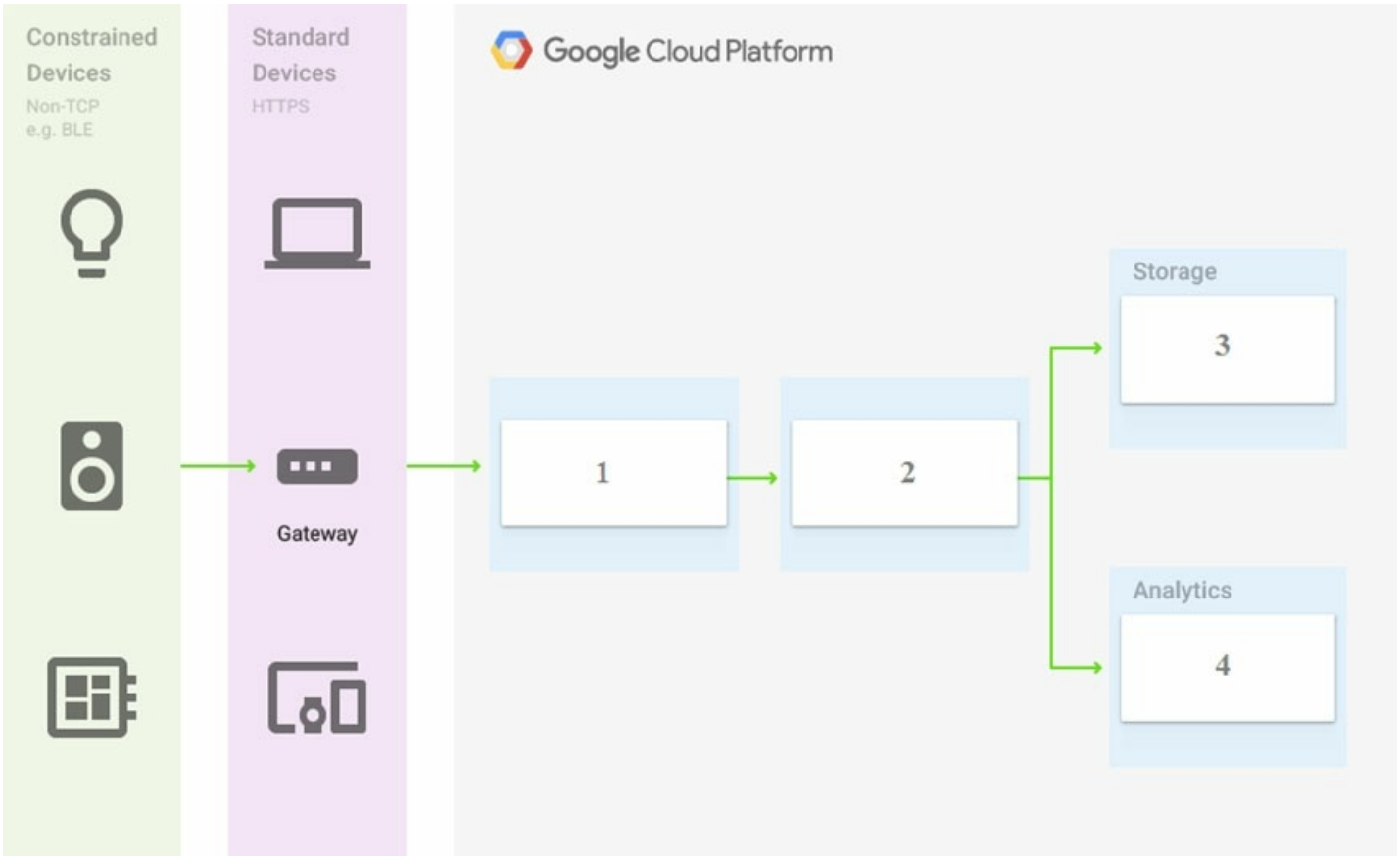
**Supporting Links:**

**gcloud deployment-manager deployments create | Cloud Deployment Manager Documentation:** https://cloud.google.com/deployment-manager/docs/reference/gcloud-dm/gcloud-dm-deployments/create - This documentation explains the gcloud deployment-manager deployments create command and highlights the --preview flag, detailing its functionality to preview deployments without applying changes.

**Previewing a Deployment | Cloud Deployment Manager Documentation:** https://cloud.google.com/deployment-manager/docs/deployments/previewing-a-deployment - This page specifically discusses how to use the preview feature in Deployment Manager for validating configurations before deployment, emphasizing its role in identifying potential issues and dependencies.

**Question: 23** CertyIQ

You are building a pipeline to process time-series data. Which Google Cloud Platform services should you put in boxes 1,2,3, and 4?




A. Cloud Pub/Sub, Cloud Dataflow, Cloud Datastore, BigQuery

B. Firebase Messages, Cloud Pub/Sub, Cloud Spanner, BigQuery

C. Cloud Pub/Sub, Cloud Storage, BigQuery, Cloud Bigtable

D. Cloud Pub/Sub, Cloud Dataflow, Cloud Bigtable, BigQuery

**Answer: D**

**Explanation:**

Cloud Pub/Sub, Cloud Dataflow, Cloud Bigtable, BigQuery

Reference:

https://cloud.google.com/solutions/correlating-time-series-dataflow

**Question: 24** CertyIQ

You have a project for your App Engine application that serves a development environment. The required testing has succeeded and you want to create a new project to serve as your production environment. What should you do?

A. Use gcloud to create the new project, and then deploy your application to the new project.

B. Use gcloud to create the new project and to copy the deployed application to the new project.

C. Create a Deployment Manager configuration file that copies the current App Engine deployment into a new project.

D. Deploy your application again using gcloud and specify the project parameter with the new project name to create the new project.

**Answer: A**

**Explanation:**

Answer: A. Use gcloud to create the new project, and then deploy your application to the new project.

Justification: To create a separate production environment for your App Engine application, you should first create a new Google Cloud project using gcloud projects create. Subsequently, you should deploy your application code to this newly created project using gcloud app deploy, ensuring that the production environment is isolated and independently managed from your development environment. This approach ensures a clean separation of environments and is the recommended practice for setting up production deployments in Google Cloud. https://cloud.google.com/appengine/docs/standard/python3/creating-project, https://cloud.google.com/appengine/docs/standard/python3/deploying-and-managing-services-and-versions

**Question: 25** CertyIQ

You need to configure IAM access audit logging in BigQuery for external auditors. You want to follow Google-recommended practices. What should you do?

A. Add the auditors group to the 'logging.viewer' and 'bigQuery.dataViewer' predefined IAM roles.
B. Add the auditors group to two new custom IAM roles.
C. Add the auditor user accounts to the 'logging.viewer' and 'bigQuery.dataViewer' predefined IAM roles.
D. Add the auditor user accounts to two new custom IAM roles.

**Answer: A**

**Explanation:**

Option A is the most appropriate solution as it leverages the predefined 'logging.viewer' role, which adheres to Google's best practice of using predefined roles to grant the least privilege necessary for accessing audit logs, and it also utilizes a group for managing auditor access, aligning with recommended practices for efficient IAM administration and scalability. To grant auditors access to IAM audit logs, the 'logging.viewer' role is essential as it provides read-only permissions to view logs, including audit logs, within the project. While 'bigquery.dataViewer' grants read-only access to BigQuery datasets, it may be included to allow auditors to view BigQuery audit logs if they are stored in BigQuery, or it could be an additional, though potentially less necessary, permission for a comprehensive audit view. Employing a group for auditors is a best practice for managing permissions efficiently, rather than assigning roles to individual user accounts, which is less scalable and harder to maintain.

**Supporting Links:**

**IAM Predefined Roles:** https://cloud.google.com/iam/docs/understanding-roles
**Logging Roles:** https://cloud.google.com/logging/docs/access-control
**Best practices for using groups for IAM:** https://cloud.google.com/iam/docs/best-practices-for-using-groups

**Question: 26** CertyIQ

You need to set up permissions for a set of Compute Engine instances to enable them to write data into a particular Cloud Storage bucket. You want to follow
Google-recommended practices. What should you do?

A. Create a service account with an access scope. Use the access scope 'https://www.googleapis.com/auth/devstorage.write_only'.
B. Create a service account with an access scope. Use the access scope 'https://www.googleapis.com/auth/cloud-platform'.
C. Create a service account and add it to the IAM role 'storage.objectCreator' for that bucket.

D. Create a service account and add it to the IAM role 'storage.objectAdmin' for that bucket.

**Answer: C**

**Explanation:**

Justification: To adhere to Google's best practices, you should create a service account and assign it the storage.objectCreator IAM role on the designated Cloud Storage bucket; this method leverages IAM for fine-grained permission control, aligning with the principle of least privilege and is the recommended approach for granting access to cloud resources, as opposed to using less secure and less granular access scopes. (https://cloud.google.com/compute/docs/access/service-accounts, https://cloud.google.com/storage/docs/access-control/iam-roles, https://cloud.google.com/iam/docs/service-accounts)

**Question: 27** CertyIQ

You have sensitive data stored in three Cloud Storage buckets and have enabled data access logging. You want to verify activities for a particular user for these buckets, using the fewest possible steps. You need to verify the addition of metadata labels and which files have been viewed from those buckets. What should you do?

A. Using the GCP Console, filter the Activity log to view the information.
B. Using the GCP Console, filter the Stackdriver log to view the information.
C. View the bucket in the Storage section of the GCP Console.
D. Create a trace in Stackdriver to view the information.

**Answer: B**

**Explanation:**

The correct answer is **B**. Here's why:

**Justification:**

To verify user activities such as adding metadata labels and viewing files in Cloud Storage buckets with data access logging enabled, you should utilize **Cloud Logging** (formerly Stackdriver Logging). Cloud Logging centrally collects and allows you to filter logs, including Data Access logs, to monitor actions within your GCP projects. By filtering logs within the Cloud Logging interface in the GCP Console, you can efficiently search for activities performed by a specific user across the three buckets, identifying metadata label additions and file view events with minimal steps. Options A, C, and D are less suitable because the term "Activity log" is not the precise GCP service name for comprehensive logging, viewing buckets directly does not provide audit trails, and Cloud Trace is designed for performance analysis, not audit logging.

**Supporting Links:**

**Cloud Logging Overview:** https://cloud.google.com/logging/docs/overview - This document provides a general overview of Cloud Logging and its capabilities.
**Data Access Logs:** https://cloud.google.com/logging/docs/audit/data-access - This document explains Data Access audit logs, which are essential for tracking who accessed user-provided data.
**Viewing Logs in the Logs Explorer:** https://cloud.google.com/logging/docs/view/query-logs - This document details how to use the Logs Explorer in the GCP Console to filter and analyze logs, including data access logs for Cloud Storage.

**Question: 28** CertyIQ

You are the project owner of a GCP project and want to delegate control to colleagues to manage buckets and files in Cloud Storage. You want to follow Google- recommended practices. Which IAM roles should you grant your colleagues?

A. Project Editor
B. Storage Admin
C. Storage Object Admin
D. Storage Object Creator

**Answer: B**

**Explanation:**

The Storage Admin role (roles/storage.admin) is the most suitable choice as it grants comprehensive control over Cloud Storage resources, enabling colleagues to manage both buckets and objects, aligning with Google's recommendation to use predefined roles for common administrative tasks. This role provides the necessary permissions to create, delete, and modify buckets and objects, as well as manage IAM policies within Cloud Storage, fulfilling the requirement to delegate bucket and file management. Granting Project Editor would be overly permissive, violating the principle of least privilege, as it provides broad access to all GCP services in the project, not just Cloud Storage. Storage Object Admin and Storage Object Creator are more restrictive roles focused on object-level operations and do not provide the full bucket management capabilities needed for this scenario. Therefore, Storage Admin strikes the right balance, offering sufficient control for managing Cloud Storage while adhering to Google-recommended IAM practices.

**Supporting Links:**

**IAM roles for Cloud Storage:** https://cloud.google.com/storage/docs/access-control/iam-roles
**IAM best practices:** https://cloud.google.com/iam/docs/best-practices

**Question: 29** CertyIQ

You have an object in a Cloud Storage bucket that you want to share with an external company. The object contains sensitive data. You want access to the content to be removed after four hours. The external company does not have a Google account to which you can grant specific user-based access privileges. You want to use the most secure method that requires the fewest steps. What should you do?

A. Create a signed URL with a four-hour expiration and share the URL with the company.
B. Set object access to 'public' and use object lifecycle management to remove the object after four hours.
C. Configure the storage bucket as a static website and furnish the object's URL to the company. Delete the object from the storage bucket after four hours.
D. Create a new Cloud Storage bucket specifically for the external company to access. Copy the object to that bucket. Delete the bucket after four hours have passed.

**Answer: A**

**Explanation:**

Option A is the most secure method with the fewest steps because signed URLs grant temporary, time-limited access to specific Cloud Storage objects without requiring Google accounts, perfectly fitting the scenario. Signed URLs allow controlled access for the external company to view the sensitive data for four hours, after which the link automatically expires and access is revoked. This approach avoids making the object publicly accessible, which would be less secure and against best practices for sensitive data. Option A directly addresses all requirements: security, time limit, external access without Google accounts, and minimal steps.

https://cloud.google.com/storage/docs/access-control/signed-urls

**Question: 30** CertyIQ

You are creating a Google Kubernetes Engine (GKE) cluster with a cluster autoscaler feature enabled. You need to make sure that each node of the cluster will run a monitoring pod that sends container metrics to a third-party monitoring solution. What should you do?

A. Deploy the monitoring pod in a StatefulSet object.

B. Deploy the monitoring pod in a DaemonSet object.

C. Reference the monitoring pod in a Deployment object.

D. Reference the monitoring pod in a cluster initializer at the GKE cluster creation time.

**Answer: B**

**Explanation:**

The correct answer is **B. Deploy the monitoring pod in a DaemonSet object.** A DaemonSet ensures that a copy of a pod runs on all (or a subset of) nodes in a Kubernetes cluster, which directly addresses the requirement of having a monitoring pod on each node. This is ideal for node-level agents like monitoring solutions that need to collect metrics from every node, including nodes added by the cluster autoscaler. Deployments and StatefulSets are not designed to guarantee one pod per node, and cluster initializers are for initial setup, not continuous pod management across all nodes. DaemonSets are specifically built for this kind of node-level operational task.

**Supporting Links:**

**Kubernetes DaemonSets:** https://kubernetes.io/docs/concepts/workloads/controllers/daemonset/
**Choosing the right controller:** https://kubernetes.io/docs/concepts/workloads/controllers/ (This page explains different workload controllers and their use cases, including DaemonSets).

**Question: 31** CertyIQ

You want to send and consume Cloud Pub/Sub messages from your App Engine application. The Cloud Pub/Sub API is currently disabled. You will use a service account to authenticate your application to the API. You want to make sure your application can use Cloud Pub/Sub. What should you do?

A. Enable the Cloud Pub/Sub API in the API Library on the GCP Console.

B. Rely on the automatic enablement of the Cloud Pub/Sub API when the Service Account accesses it.

C. Use Deployment Manager to deploy your application. Rely on the automatic enablement of all APIs used by the application being deployed.

D. Grant the App Engine Default service account the role of Cloud Pub/Sub Admin. Have your application enable the API on the first connection to Cloud Pub/ Sub.

**Answer: A**

**Explanation:**

The correct answer is **A**. To use Cloud Pub/Sub, the API must be explicitly enabled in your GCP project. Enabling the Cloud Pub/Sub API through the API Library in the GCP Console is the standard and direct method to activate the service for your project, making it accessible to your App Engine application. Relying on automatic API enablement (option B and C) is not a guaranteed or recommended practice, especially in production environments. Option D is incorrect because service account roles manage permissions after the

API is enabled, and applications cannot programmatically enable APIs themselves. Therefore, manually enabling the API via the GCP Console is the necessary first step to allow your App Engine application to interact with Cloud Pub/Sub.

**Supporting Links:**

**Enabling and disabling APIs:** https://cloud.google.com/apis/design/enablement
**Google Cloud APIs:** https://console.cloud.google.com/apis/library

**Question: 32** CertyIQ

You need to monitor resources that are distributed over different projects in Google Cloud Platform. You want to consolidate reporting under the same Stackdriver
Monitoring dashboard. What should you do?

A. Use Shared VPC to connect all projects, and link Stackdriver to one of the projects.

B. For each project, create a Stackdriver account. In each project, create a service account for that project and grant it the role of Stackdriver Account Editor in all other projects.

C. Configure a single Stackdriver account, and link all projects to the same account.

D. Configure a single Stackdriver account for one of the projects. In Stackdriver, create a Group and add the other project names as criteria for that Group.

**Answer: C**

**Explanation:**

Rationale: To consolidate monitoring from multiple GCP projects into a single Stackdriver Monitoring dashboard, you should configure a single Stackdriver account and link all the projects to this account. Cloud Monitoring is designed to handle multi-project monitoring, allowing a central project to act as a monitoring account and collect metrics from other projects. Linking projects to a single monitoring account provides a unified view of resources across all projects in one dashboard. This approach avoids the complexity and redundancy of managing multiple Stackdriver accounts or relying on networking solutions like Shared VPC for monitoring consolidation. Option C directly addresses the requirement for consolidated reporting within a single dashboard, making it the most efficient and recommended solution.

Supporting Links:

Cloud Monitoring documentation on multi-project monitoring: This document explains how to configure and use Cloud Monitoring for multiple Google Cloud projects.
Google Cloud documentation on Monitored projects: This page details the concept of monitored projects and how they relate to a monitoring scope in Cloud Monitoring.

**Question: 33** CertyIQ

You are deploying an application to a Compute Engine VM in a managed instance group. The application must be running at all times, but only a single instance of the VM should run per GCP project. How should you configure the instance group?

A. Set autoscaling to On, set the minimum number of instances to 1, and then set the maximum number of instances to 1.

B. Set autoscaling to Off, set the minimum number of instances to 1, and then set the maximum number of instances to 1.

C. Set autoscaling to On, set the minimum number of instances to 1, and then set the maximum number of instances to 2.

D. Set autoscaling to Off, set the minimum number of instances to 1, and then set the maximum number of instances to 2.

**Answer: A**

**Explanation:**

Option A is the correct answer because setting autoscaling to 'On' with a minimum and maximum instance count of 1 ensures that the managed instance group maintains exactly one VM instance at all times, automatically replacing it if it becomes unhealthy, thus guaranteeing the application is always running while adhering to the single instance limit per project. Managed Instance Groups with autoscaling enabled leverage health checks to ensure application availability and automatically replace failing instances. https://cloud.google.com/compute/docs/instance-groups/autoscaling, https://cloud.google.com/compute/docs/instance-groups/creating-groups-of-managed-instances

**Question: 34** CertyIQ

You want to verify the IAM users and roles assigned within a GCP project named my-project. What should you do?

A. Run gcloud iam roles list. Review the output section.

B. Run gcloud iam service-accounts list. Review the output section.

C. Navigate to the project and then to the IAM section in the GCP Console. Review the members and roles.

D. Navigate to the project and then to the Roles section in the GCP Console. Review the roles and status.

**Answer: C**

**Explanation:**

**Justification:** Navigating to the IAM section within the GCP Console for the specified project (my-project) directly displays a list of all members (users, service accounts, groups) and their corresponding roles assigned at the project level, effectively verifying the IAM configuration as requested. This graphical interface within the console offers a user-friendly and immediate way to review the project's IAM policy. https://cloud.google.com/iam/docs/using-console, https://cloud.google.com/iam/docs/overview

**Question: 35** CertyIQ

You need to create a new billing account and then link it with an existing Google Cloud Platform project. What should you do?

A. Verify that you are Project Billing Manager for the GCP project. Update the existing project to link it to the existing billing account.

B. Verify that you are Project Billing Manager for the GCP project. Create a new billing account and link the new billing account to the existing project.

C. Verify that you are Billing Administrator for the billing account. Create a new project and link the new project to the existing billing account.

D. Verify that you are Billing Administrator for the billing account. Update the existing project to link it to the existing billing account.

**Answer: B**

**Explanation:**

To create a new billing account and link it to an existing Google Cloud Platform project, you must first verify you have the Project Billing Manager role on the GCP project to manage billing settings for that project. Then, proceed to create a new billing account following Google Cloud documentation. Finally, link the newly created billing account to your existing GCP project within the Google Cloud Console or using gcloud commands. Option B accurately reflects these steps by advising verification of Project Billing Manager role and subsequent creation of a new billing account followed by linking it to the existing project.

Supporting Links:

1. **Create a new Cloud Billing account:** https://cloud.google.com/billing/docs/how-to/manage-billing-account#create_billing_account
2. **Link a project to a Cloud Billing account:** https://cloud.google.com/billing/docs/how-to/link-project-billing
3. **Cloud Billing Access Control:** https://cloud.google.com/billing/docs/concepts/access-control

**Question: 36** CertyIQ

You have one project called proj-sa where you manage all your service accounts. You want to be able to use a service account from this project to take snapshots of VMs running in another project called proj-vm. What should you do?

A. Download the private key from the service account, and add it to each VMs custom metadata.
B. Download the private key from the service account, and add the private key to each VM's SSH keys.
C. Grant the service account the IAM Role of Compute Storage Admin in the project called proj-vm.
D. When creating the VMs, set the service account's API scope for Compute Engine to read/write.

**Answer: C**

**Explanation:**

**Rationale:** Option C is correct because to allow a service account from proj-sa to manage resources in proj-vm, you must grant it the necessary IAM permissions within proj-vm. Granting the Compute Storage Admin role to the service account from proj-sa in proj-vm provides the required permissions to manage Compute Engine storage resources, including taking snapshots of VMs in proj-vm. This approach adheres to the principle of least privilege by granting specific permissions rather than sharing private keys or modifying VM configurations unnecessarily.

**Supporting Links:**

**Granting roles to service accounts:** https://cloud.google.com/iam/docs/granting-roles-to-service-accounts - This documentation explains how to grant IAM roles to service accounts, which is essential for cross-project access.
**Compute Storage Admin role:** https://cloud.google.com/iam/docs/understanding-roles#compute_storage_admin - This page details the permissions included in the Compute Storage Admin role, confirming it allows managing storage resources like snapshots.
**Service Accounts Overview:** https://cloud.google.com/iam/docs/service-accounts - This provides a general understanding of service accounts and their use in Google Cloud IAM.

**Question: 37** CertyIQ

You created a Google Cloud Platform project with an App Engine application inside the project. You initially configured the application to be served from the us- central region. Now you want the application to be served

from the asia-northeast1 region. What should you do?

A. Change the default region property setting in the existing GCP project to asia-northeast1.

B. Change the region property setting in the existing App Engine application from us-central to asia-northeast1.

C. Create a second App Engine application in the existing GCP project and specify asia-northeast1 as the region to serve your application.

D. Create a new GCP project and create an App Engine application inside this new project. Specify asia-northeast1 as the region to serve your application.

**Answer: D**

**Explanation:**

The correct answer is **D**. Here's the justification:

App Engine region is set during application creation and cannot be changed after the application is deployed. If you need to serve your application from a different region like asia-northeast1 instead of the initial us-central, you cannot modify the region of the existing App Engine application. Therefore, you must create a new App Engine application in a new GCP project, specifying asia-northeast1 as the desired region during the new application creation. This approach is necessary because App Engine region configuration is a project-level setting defined at the time of application creation and is immutable afterwards. Options A and B are incorrect as they suggest modifying the region setting, which is not possible. Option C, while technically feasible, would result in two separate applications in different regions within the same project, not a region change for the original application.

Supporting documentation:

**App Engine Locations:** https://cloud.google.com/appengine/docs/locations - This document clearly states "After you deploy your application, you cannot change its location."

**Choosing a region (App Engine Standard Environment):** https://cloud.google.com/appengine/docs/standard/python/configuration/app-settings#region - Although specific to Python, the principle holds true across all App Engine environments: "You select a region when you create your App Engine application. You cannot change the region later."

**Question: 38** CertyIQ

You need to grant access for three users so that they can view and edit table data on a Cloud Spanner instance. What should you do?

A. Run gcloud iam roles describe roles/spanner.databaseUser. Add the users to the role.

B. Run gcloud iam roles describe roles/spanner.databaseUser. Add the users to a new group. Add the group to the role.

C. Run gcloud iam roles describe roles/spanner.viewer - -project my-project. Add the users to the role.

D. Run gcloud iam roles describe roles/spanner.viewer - -project my-project. Add the users to a new group. Add the group to the role.

**Answer: B**

**Explanation:**

Option B is the correct answer because roles/spanner.databaseUser grants the necessary permissions to view and edit table data within a Cloud Spanner instance. Adding users to a group and then assigning this group the roles/spanner.databaseUser role is a best practice for efficient IAM management, promoting scalability and

simplified administration compared to assigning roles to individual users. This approach ensures organized and manageable access control for multiple users. Options C and D are incorrect as roles/spanner.viewer only allows viewing data, not editing, which fails to meet the requirement. Option A, while functional, is less manageable for multiple users compared to using groups.

**Supporting Links:**

**Cloud Spanner IAM Roles:** https://cloud.google.com/spanner/docs/iam#predefined
**IAM Best Practices - Use Groups:** https://cloud.google.com/iam/docs/best-practices-for-managing-iam#use-groups

**Question: 39** CertyIQ

You create a new Google Kubernetes Engine (GKE) cluster and want to make sure that it always runs a supported and stable version of Kubernetes. What should you do?

A. Enable the Node Auto-Repair feature for your GKE cluster.
B. Enable the Node Auto-Upgrades feature for your GKE cluster.
C. Select the latest available cluster version for your GKE cluster.
D. Select Container-Optimized OS (cos) as a node image for your GKE cluster.

**Answer: B**

**Explanation:**

**Option B is the correct answer.** GKE Node Auto-Upgrades automatically updates the Kubernetes version on your cluster's nodes to a stable and supported version, ensuring continuous security and feature updates. This feature is designed to keep your cluster running a supported Kubernetes version, which is crucial for stability and receiving security patches. Node Auto-Repair focuses on node health, not Kubernetes version updates, and selecting the latest version is a one-time action, not continuous maintenance. Container-Optimized OS is a node image and doesn't manage Kubernetes version upgrades. For more details, refer to the Google Cloud documentation on node auto-upgrades: https://cloud.google.com/kubernetes-engine/docs/how-to/node-auto-upgrades

**Question: 40** CertyIQ

You have an instance group that you want to load balance. You want the load balancer to terminate the client SSL session. The instance group is used to serve a public web application over HTTPS. You want to follow Google-recommended practices. What should you do?

A. Configure an HTTP(S) load balancer.
B. Configure an internal TCP load balancer.
C. Configure an external SSL proxy load balancer.
D. Configure an external TCP proxy load balancer.

**Answer: A**

**Explanation:**

The correct answer is **A. Configure an HTTP(S) load balancer.** Here's the justification: An HTTP(S) load balancer is specifically designed for load balancing HTTP and HTTPS traffic and is the Google-recommended solution for serving public web applications over HTTPS. It terminates SSL connections at the load balancer,

thus offloading SSL processing from the backend instances within the instance group, improving performance and security. HTTP(S) load balancers offer features tailored for web applications, such as content-based routing and integration with Google Cloud CDN. Internal TCP load balancers are for internal traffic, while external TCP and SSL proxy load balancers are not optimized for web traffic like HTTP(S) load balancers and are generally used for other TCP-based protocols or specific SSL proxying needs.

Supporting Links:

Choosing a load balancer
HTTP(S) Load Balancing Overview
SSL certificates overview

**Question: 41** CertyIQ

You have 32 GB of data in a single file that you need to upload to a Nearline Storage bucket. The WAN connection you are using is rated at 1 Gbps, and you are the only one on the connection. You want to use as much of the rated 1 Gbps as possible to transfer the file rapidly. How should you upload the file?

A. Use the GCP Console to transfer the file instead of gsutil.
B. Enable parallel composite uploads using gsutil on the file transfer.
C. Decrease the TCP window size on the machine initiating the transfer.
D. Change the storage class of the bucket from Nearline to Multi-Regional.

**Answer: B**

**Explanation:**

**Justification:** To maximize the utilization of a 1 Gbps WAN connection for uploading a 32 GB file to Nearline Storage, enabling parallel composite uploads using gsutil is the optimal approach as it breaks the large file into smaller pieces and uploads them concurrently, thereby leveraging the available bandwidth more efficiently and reducing the overall transfer time. This method significantly improves upload speed for large files compared to standard sequential uploads or using the GCP Console, and is specifically designed for optimizing large data transfers to Google Cloud Storage. Disabling parallel uploads or changing storage class would not address the goal of maximizing upload speed. Decreasing TCP window size would likely hinder, not help, the transfer speed.

**Supporting Links:**

**Parallel composite uploads:** https://cloud.google.com/storage/docs/parallel-composite-uploads
**gsutil cp command and parallel uploads:** https://cloud.google.com/storage/docs/gsutil/commands/cp (Search for "Parallel transfers" on the page)
**Optimizing gsutil Performance:** https://cloud.google.com/storage/docs/gsutil/performance

**Question: 42** CertyIQ

You've deployed a microservice called myapp1 to a Google Kubernetes Engine cluster using the YAML file specified below:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: myapp1-deployment
spec:
  selector:
    matchLabels:
      app: myapp1
  replicas: 2
  template:
    metadata:
      labels:
        app: myapp1
    spec:
      containers:
      - name: main-container
        image: gcr.io/my-company-repo/myapp1:1.4
        env:
        - name: DB_PASSWORD
          value: "t0ugh2guess!"
        ports:
        - containerPort: 8080
```

You need to refactor this configuration so that the database password is not stored in plain text. You want to follow Google-recommended practices. What should you do?

A. Store the database password inside the Docker image of the container, not in the YAML file.

B. Store the database password inside a Secret object. Modify the YAML file to populate the DB_PASSWORD environment variable from the Secret.

C. Store the database password inside a ConfigMap object. Modify the YAML file to populate the DB_PASSWORD environment variable from the ConfigMap.

D. Store the database password in a file inside a Kubernetes persistent volume, and use a persistent volume claim to mount the volume to the container.

**Answer: B**

**Explanation:**

it is good practice to use Secrets for confidential data (like API keys) and ConfigMaps for non-confidential data (like port numbers). B is correct

**Question: 43** CertyIQ

You are running an application on multiple virtual machines within a managed instance group and have autoscaling enabled. The autoscaling policy is configured so that additional instances are added to the group if the CPU utilization of instances goes above 80%. VMs are added until the instance group reaches its maximum limit of five VMs or until CPU utilization of instances lowers to 80%. The initial delay for HTTP health checks against the instances is set to 30 seconds.
The virtual machine instances take around three minutes to become available for users. You observe that when the instance group autoscales, it adds more instances then necessary to support the levels of end-user traffic. You

want to properly maintain instance group sizes when autoscaling. What should you do?

A. Set the maximum number of instances to 1.
B. Decrease the maximum number of instances to 3.
C. Use a TCP health check instead of an HTTP health check.
D. Increase the initial delay of the HTTP health check to 200 seconds.

**Answer: D**

**Explanation:**

The autoscaler might be adding more instances than necessary because the initial health check delay is shorter than the time it takes for the VMs to become fully operational, leading to premature scaling decisions based on incomplete metrics. Increasing the initial delay of the HTTP health check to 200 seconds, which is closer to the VM's 3-minute startup time, ensures that the autoscaler waits until new instances are fully ready before evaluating CPU utilization and making further scaling adjustments, thus preventing overscaling. This allows the autoscaler to accurately reflect the impact of newly added instances on CPU utilization before adding more. Options A and B restrict scaling capacity, while option C, using TCP health checks, doesn't address the timing issue related to VM startup and application readiness.

Supporting Links:

Autoscaling based on HTTP(S) health check
Health checks for autoscaling
Best practices for autoscaling

**Question: 44** CertyIQ

You need to select and configure compute resources for a set of batch processing jobs. These jobs take around 2 hours to complete and are run nightly. You want to minimize service costs. What should you do?

A. Select Google Kubernetes Engine. Use a single-node cluster with a small instance type.
B. Select Google Kubernetes Engine. Use a three-node cluster with micro instance types.
C. Select Compute Engine. Use preemptible VM instances of the appropriate standard machine type.
D. Select Compute Engine. Use VM instance types that support micro bursting.

**Answer: C**

**Explanation:**

**Justification:** Compute Engine preemptible VMs offer substantial cost savings suitable for fault-tolerant batch jobs, aligning with the need to minimize service costs for nightly processing. Preemptible VMs are significantly cheaper than regular VMs and are ideal for batch workloads that can tolerate interruptions, as the jobs are run nightly and can be restarted if preempted. Utilizing standard machine types within Compute Engine allows for selecting the appropriate resources for the 2-hour job duration, optimizing performance and cost. GKE introduces unnecessary complexity and overhead for simple batch processing compared to Compute Engine VMs. Micro bursting, while beneficial for network performance, is not the primary cost optimization strategy for batch processing like preemptible VMs.

**Supporting Links:**

1. **Preemptible VMs:** https://cloud.google.com/compute/docs/instances/preemptible - This page details preemptible VMs, their cost benefits, and suitability for batch processing and fault-tolerant workloads.

2. **Compute Engine Pricing:** https://cloud.google.com/compute/vm-instances/pricing - This page provides pricing information for Compute Engine VMs, highlighting the cost difference between regular and preemptible instances.
3. **Batch Processing on Google Cloud:** https://cloud.google.com/batch - While focusing on the Batch service, it reinforces the concept of using cost-optimized compute resources like VMs for batch workloads on Google Cloud.

**Question: 45** CertyIQ

You recently deployed a new version of an application to App Engine and then discovered a bug in the release. You need to immediately revert to the prior version of the application. What should you do?

A. Run gcloud app restore.

B. On the App Engine page of the GCP Console, select the application that needs to be reverted and click Revert.

C. On the App Engine Versions page of the GCP Console, route 100% of the traffic to the previous version.

D. Deploy the original version as a separate application. Then go to App Engine settings and split traffic between applications so that the original version serves 100% of the requests.

**Answer: C**

**Explanation:**

To immediately revert to the prior working version of an App Engine application after discovering a bug, the most efficient approach is to route 100% of traffic to the previously deployed version from the App Engine Versions page in the Google Cloud Console. App Engine automatically retains previous versions upon each deployment, allowing for quick rollbacks by simply adjusting traffic distribution. This method leverages App Engine's built-in versioning and traffic management capabilities for seamless reversion. Options like gcloud app restore are not valid commands, and deploying the original version as a separate application is an unnecessary and complex approach for a simple rollback. Utilizing the Versions page for traffic routing ensures a fast and direct reversion to the stable application state. For more information on managing versions and traffic splitting in App Engine, refer to the official Google Cloud documentation on App Engine Versions and Traffic Splitting.

**Question: 46** CertyIQ

You deployed an App Engine application using gcloud app deploy, but it did not deploy to the intended project. You want to find out why this happened and where the application deployed. What should you do?

A. Check the app.yaml file for your application and check project settings.

B. Check the web-application.xml file for your application and check project settings.

C. Go to Deployment Manager and review settings for deployment of applications.

D. Go to Cloud Shell and run gcloud config list to review the Google Cloud configuration used for deployment.

**Answer: D**

**Explanation:**

Rationale: To determine the project where the App Engine application was mistakenly deployed and understand the cause, the most direct approach is to inspect the Google Cloud SDK configuration used during deployment. Running gcloud config list in Cloud Shell reveals the active configuration at the time of

deployment, including the project ID, which directly influences where gcloud app deploy sends the application. This command helps identify if the wrong project was configured in the Cloud SDK during deployment, explaining why the application landed in an unintended project. Checking app.yaml (A) or web-application.xml (B) might show application settings but not the deployment project. Deployment Manager (C) is for broader infrastructure deployments, not typically used for standard App Engine application deployments via gcloud app deploy.

Supporting Links:

gcloud config list
gcloud app deploy

**Question: 47** CertyIQ

You want to configure 10 Compute Engine instances for availability when maintenance occurs. Your requirements state that these instances should attempt to automatically restart if they crash. Also, the instances should be highly available including during system maintenance. What should you do?

A. Create an instance template for the instances. Set the 'Automatic Restart' to on. Set the 'On-host maintenance' to Migrate VM instance. Add the instance template to an instance group.

B. Create an instance template for the instances. Set 'Automatic Restart' to off. Set 'On-host maintenance' to Terminate VM instances. Add the instance template to an instance group.

C. Create an instance group for the instances. Set the 'Autohealing' health check to healthy (HTTP).

D. Create an instance group for the instance. Verify that the 'Advanced creation options' setting for 'do not retry machine creation' is set to off.

**Answer: A**

**Explanation:**

Option A is the correct answer because it utilizes an instance template to ensure consistent configuration across instances and sets 'Automatic Restart' to 'on' to automatically recover from crashes. Setting 'On-host maintenance' to 'Migrate VM instance' ensures high availability during infrastructure maintenance by live migrating VMs to other hosts. Adding the instance template to an instance group allows for managing these instances as a scalable and highly available group. This combination addresses both automatic restart and high availability requirements, including during system maintenance events.

**Supporting Links:**

**Instance Templates:** https://cloud.google.com/compute/docs/instance-templates
**Automatic Restart:** https://cloud.google.com/compute/docs/instances/restarting-terminated-instances
**Live Migration (On-host maintenance):** https://cloud.google.com/compute/docs/instances/live-migration
**Instance Groups:** https://cloud.google.com/compute/docs/instance-groups

**Question: 48** CertyIQ

You host a static website on Cloud Storage. Recently, you began to include links to PDF files on this site. Currently, when users click on the links to these PDF files, their browsers prompt them to save the file onto their local system. Instead, you want the clicked PDF files to be displayed within the browser window directly, without prompting the user to save the file locally. What should you do?

A. Enable Cloud CDN on the website frontend.

B. Enable 'Share publicly' on the PDF file objects.

C. Set Content-Type metadata to application/pdf on the PDF file objects.

D. Add a label to the storage bucket with a key of Content-Type and value of application/pdf.

**Answer: C**

**Explanation:**

Setting the Content-Type metadata to application/pdf on the PDF file objects informs the browser that the file is a PDF document, which is the standard MIME type for PDF files. Browsers are typically configured to display PDF files inline when they receive this Content-Type header, rather than prompting users to download them. This action directly addresses the browser's handling of the PDF file by correctly identifying its content type. Options A, B, and D do not directly control the browser's behavior regarding file display based on content type.

**Supporting Links:**

**Google Cloud Storage Metadata:** https://cloud.google.com/storage/docs/metadata
**MDN Web Docs on Content-Type:** https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Content-Type

**Question: 49** CertyIQ

You have a virtual machine that is currently configured with 2 vCPUs and 4 GB of memory. It is running out of memory. You want to upgrade the virtual machine to have 8 GB of memory. What should you do?

A. Rely on live migration to move the workload to a machine with more memory.
B. Use gcloud to add metadata to the VM. Set the key to required-memory-size and the value to 8 GB.
C. Stop the VM, change the machine type to n1-standard-8, and start the VM.
D. Stop the VM, increase the memory to 8 GB, and start the VM.

**Answer: D**

**Explanation:**

The correct answer is **D**. To increase the memory of a Google Cloud Virtual Machine, you must first stop the VM instance. Once stopped, you can modify the instance configuration to increase the memory to the desired 8 GB. After adjusting the memory, you need to start the VM again for the changes to take effect. Options A and B are incorrect as live migration does not change VM resources, and metadata cannot be used to directly increase VM memory. Option C, while also involving stopping and starting the VM, suggests changing the machine type, which might over-provision resources and is not the most direct way to simply increase memory to 8 GB as requested.

**Supporting Links:**

**Modifying a VM instance:** https://cloud.google.com/compute/docs/instances/changing-machine-type-instance (This document explains how to change machine type and also implicitly covers customizing memory as part of instance modification which requires stopping the instance).
**Custom machine types:** https://cloud.google.com/compute/docs/instances/custom-machine-types (This document details how to customize CPU and memory for VMs, reinforcing the need to stop the VM for such modifications).

**Question: 50** CertyIQ

You have production and test workloads that you want to deploy on Compute Engine. Production VMs need to be in

a different subnet than the test VMs. All the
VMs must be able to reach each other over Internal IP without creating additional routes. You need to set up VPC and the 2 subnets. Which configuration meets these requirements?

A. Create a single custom VPC with 2 subnets. Create each subnet in a different region and with a different CIDR range.

B. Create a single custom VPC with 2 subnets. Create each subnet in the same region and with the same CIDR range.

C. Create 2 custom VPCs, each with a single subnet. Create each subnet in a different region and with a different CIDR range.

D. Create 2 custom VPCs, each with a single subnet. Create each subnet in the same region and with the same CIDR range.

**Answer: A**

**Explanation:**

Here is the justification for the answer A:

Creating a single custom VPC allows resources within the network to communicate using internal IP addresses by default, satisfying the requirement for VMs to reach each other without additional routes. Utilizing two subnets within this VPC ensures the separation of production and test VMs into distinct network segments. Subnets within the same VPC, even in different regions, can communicate privately. Assigning different CIDR ranges to these subnets is essential for proper network segmentation and to avoid IP address conflicts within the VPC. Option B is incorrect because subnets in the same VPC cannot have overlapping CIDR ranges. Options C and D, with separate VPCs, would require VPC peering or other network connectivity solutions, contradicting the requirement of no additional routes for internal communication.

**Supporting Links:**

**VPC overview:** https://cloud.google.com/vpc/docs/vpc
**Subnets:** https://cloud.google.com/vpc/docs/subnets
**Internal IP Addresses:** https://cloud.google.com/vpc/docs/ip-addresses#internal-ip
**VPC Peering:** https://cloud.google.com/vpc/docs/vpc-peering

**Question: 51** CertyIQ

You need to create an autoscaling managed instance group for an HTTPS web application. You want to make sure that unhealthy VMs are recreated. What should you do?

A. Create a health check on port 443 and use that when creating the Managed Instance Group.

B. Select Multi-Zone instead of Single-Zone when creating the Managed Instance Group.

C. In the Instance Template, add the label 'health-check'.

D. In the Instance Template, add a startup script that sends a heartbeat to the metadata server.

**Answer: A**

**Explanation:**

Rationale: To ensure unhealthy VMs in a Managed Instance Group (MIG) for an HTTPS web application are automatically recreated, you should configure a health check on port 443, the standard port for HTTPS, and associate it with the MIG. This configuration enables the MIG to periodically check the health of instances by sending requests to port 443; instances that fail the health check are considered unhealthy and will be automatically recreated by the MIG, thus ensuring the application's high availability and responsiveness. Options B, C, and D are incorrect because selecting Multi-Zone improves availability across zones but does

not ensure VM recreation based on health, labels are for metadata and do not enable health checks, and while a startup script can implement a custom health check, using GCP's built-in health check mechanism is the recommended and more integrated approach for MIGs.

Supported Links:

1. **Health checks for MIGs**: https://cloud.google.com/compute/docs/instance-groups/configuring-health-checks
2. **Creating health checks**: https://cloud.google.com/load-balancing/docs/health-checks

**Question: 52** CertyIQ

Your company has a Google Cloud Platform project that uses BigQuery for data warehousing. Your data science team changes frequently and has few members.
You need to allow members of this team to perform queries. You want to follow Google-recommended practices.
What should you do?

A. 1. Create an IAM entry for each data scientist's user account. 2. Assign the BigQuery jobUser role to the group.

B. 1. Create an IAM entry for each data scientist's user account. 2. Assign the BigQuery dataViewer user role to the group.

C. 1. Create a dedicated Google group in Cloud Identity. 2. Add each data scientist's user account to the group. 3. Assign the BigQuery jobUser role to the group.

D. 1. Create a dedicated Google group in Cloud Identity. 2. Add each data scientist's user account to the group. 3. Assign the BigQuery dataViewer user role to the group.

**Answer: C**

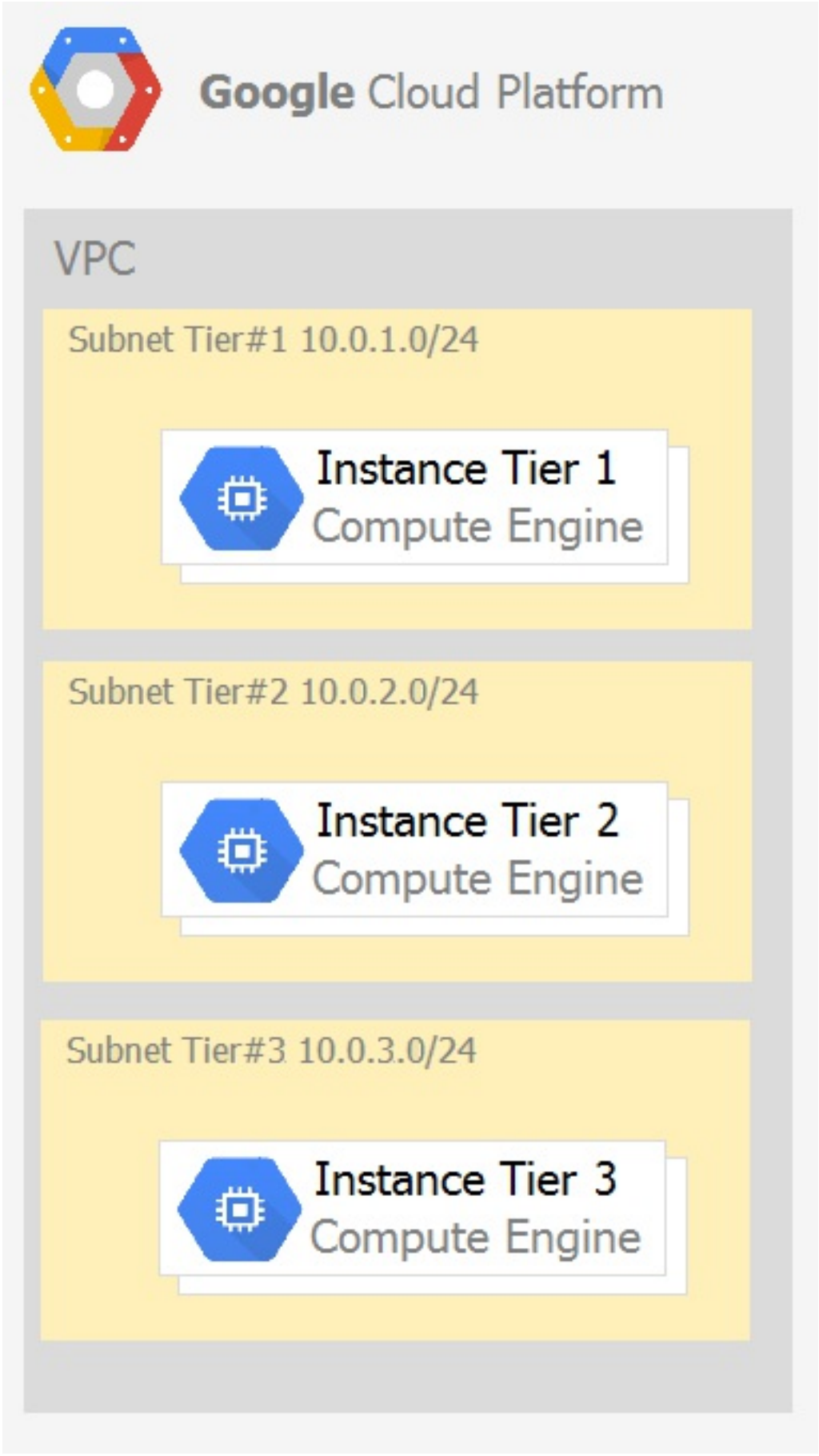
**Explanation:**

Rationale: Option C is the most suitable approach because it aligns with Google's best practices for IAM by utilizing Google Groups to manage team access, which simplifies user management for frequently changing teams and reduces administrative overhead compared to managing individual user accounts; assigning the BigQuery Job User role to the group grants the necessary permissions for data scientists to run queries and other jobs in BigQuery, as this role is specifically designed to allow users to run jobs including queries, without granting broader data access or administrative privileges, thus adhering to the principle of least privilege. Utilizing Google Groups for IAM management is a recommended practice for efficient and scalable permission control in Google Cloud. The BigQuery Job User role is appropriate for users who need to run queries and jobs in BigQuery. Managing permissions at the group level is more efficient than managing individual user permissions, especially for teams with frequent membership changes. This approach ensures that permissions are consistently applied to the team members and simplifies the process of adding or removing team members.

Link for Google Groups IAM Best Practices: https://cloud.google.com/iam/docs/best-practices#groupsLink for BigQuery Predefined Roles: https://cloud.google.com/bigquery/docs/access-control-iam-roles

**Question: 53** CertyIQ

Your company has a 3-tier solution running on Compute Engine. The configuration of the current infrastructure is shown below.




Each tier has a service account that is associated with all instances within it. You need to enable communication on TCP port 8080 between tiers as follows:
* Instances in tier #1 must communicate with tier #2.
* Instances in tier #2 must communicate with tier #3.
What should you do?

A. 1. Create an ingress firewall rule with the following settings: ¢ Targets: all instances ¢ Source filter: IP ranges (with the range set to 10.0.2.0/24) ¢ Protocols: allow all 2. Create an ingress firewall rule with the following settings: ¢ Targets: all instances ¢ Source filter: IP ranges (with the range set to 10.0.1.0/24) ¢ Protocols: allow all

B. 1. Create an ingress firewall rule with the following settings: ¢ Targets: all instances with tier #2 service account ¢ Source filter: all instances with tier #1 service account ¢ Protocols: allow TCP:8080 2. Create an ingress firewall rule with the following settings: ¢ Targets: all instances with tier #3 service account ¢ Source filter: all instances with tier #2 service account ¢ Protocols: allow TCP: 8080

C. 1. Create an ingress firewall rule with the following settings: ¢ Targets: all instances with tier #2 service account ¢ Source filter: all instances with tier #1 service account ¢ Protocols: allow all 2. Create an ingress firewall rule with the following settings: ¢ Targets: all instances with tier #3 service account ¢ Source filter: all instances with tier #2 service account ¢ Protocols: allow all

D. 1. Create an egress firewall rule with the following settings: ¢ Targets: all instances ¢ Source filter: IP ranges (with the range set to 10.0.2.0/24) ¢ Protocols: allow TCP: 8080 2. Create an egress firewall rule with the following settings: ¢ Targets: all instances ¢ Source filter: IP ranges (with the range set to 10.0.1.0/24) ¢ Protocols: allow TCP: 8080

**Answer: B**

**Explanation:**

B is correct.

1. Create an ingress firewall rule with the following settings: ¢ Targets: all instances with tier #2 service account ¢ Source filter: all instances with tier #1 service account ¢ Protocols: allow TCP:8080 2. Create an ingress firewall rule with the following settings: ¢ Targets: all instances with tier #3 service account ¢ Source filter: all instances with tier #2 service account ¢ Protocols: allow TCP: 8080

**Question: 54** CertyIQ

You are given a project with a single Virtual Private Cloud (VPC) and a single subnetwork in the us-central1 region. There is a Compute Engine instance hosting an application in this subnetwork. You need to deploy a new instance in the same project in the europe-west1 region. This new instance needs access to the application. You want to follow Google-recommended practices. What should you do?

A. 1. Create a subnetwork in the same VPC, in europe-west1. 2. Create the new instance in the new subnetwork and use the first instance's private address as the endpoint.

B. 1. Create a VPC and a subnetwork in europe-west1. 2. Expose the application with an internal load balancer. 3. Create the new instance in the new subnetwork and use the load balancer's address as the endpoint.

C. 1. Create a subnetwork in the same VPC, in europe-west1. 2. Use Cloud VPN to connect the two subnetworks. 3. Create the new instance in the new subnetwork and use the first instance's private address as the endpoint.

D. 1. Create a VPC and a subnetwork in europe-west1. 2. Peer the 2 VPCs. 3. Create the new instance in the new subnetwork and use the first instance's private address as the endpoint.

**Answer: A**

**Explanation:**

The correct answer is A because creating a subnetwork in the same VPC in the europe-west1 region allows the new instance to communicate with the existing instance using private IP addresses within the same Virtual Private Cloud (VPC) network, which is a Google-recommended practice for network connectivity within a project. This approach leverages the global nature of VPCs to span multiple regions and provides simple and efficient private communication between instances in different regions within the same network. Options B, C, and D introduce unnecessary complexity by suggesting the creation of new VPCs, internal load balancers, Cloud VPN, or VPC peering, which are not required for basic cross-region connectivity within a single VPC and project. Using private IP addresses within the same VPC is more secure and cost-effective compared to exposing applications through load balancers or setting up VPN/peering for internal communication. Therefore, Option A represents the most straightforward and Google-recommended solution for connecting instances in different regions within the same project and VPC.

**Question: 55** CertyIQ

Your projects incurred more costs than you expected last month. Your research reveals that a development GKE container emitted a huge number of logs, which resulted in higher costs. You want to disable the logs quickly using the minimum number of steps. What should you do?

A. 1. Go to the Logs ingestion window in Stackdriver Logging, and disable the log source for the GKE container resource.

B. 1. Go to the Logs ingestion window in Stackdriver Logging, and disable the log source for the GKE Cluster Operations resource.

C. 1. Go to the GKE console, and delete existing clusters. 2. Recreate a new cluster. 3. Clear the option to enable legacy Stackdriver Logging.

D. 1. Go to the GKE console, and delete existing clusters. 2. Recreate a new cluster. 3. Clear the option to enable legacy Stackdriver Monitoring.

**Answer: A**

**Explanation:**

Rationale: Option A is correct because Cloud Logging allows you to control log ingestion using exclusion filters, which can be configured through the Logs ingestion window (now referred to as Log Router in the Cloud Console, specifically under "Logs Exclusions"). This method directly targets the problematic GKE container logs by creating an exclusion filter for that specific resource, effectively stopping further ingestion of these logs and minimizing costs quickly with minimal steps. Option B is less precise as disabling GKE Cluster Operations logs might impact visibility into cluster management events, and might not directly address the container logs. Options C and D are overly complex and time-consuming, involving cluster deletion and recreation, which is not a quick solution and is far from the minimum number of steps required to disable logs. Disabling legacy Stackdriver Logging or Monitoring during cluster creation (options C and D) is not the most efficient way to stop excessive logs from a running container.

Supported Links:

1. **Cloud Logging Exclusions**: https://cloud.google.com/logging/docs/exclusions - This documentation explains how to use exclusion filters in Cloud Logging to prevent certain logs from being ingested, which directly aligns with the action described in option A.
2. **Route logs to destinations supported by Cloud Logging**: https://cloud.google.com/logging/docs/routing/overview - This document provides an overview of log routing, including how to exclude logs, confirming that Cloud Logging provides mechanisms to control log ingestion.
3. **Exclusion filters**: https://cloud.google.com/logging/docs/exclusions-filters - This page details how to create exclusion filters based on various criteria, including resource types (like GKE containers), further supporting the feasibility and effectiveness of option A.

**Question: 56** CertyIQ

You have a website hosted on App Engine standard environment. You want 1% of your users to see a new test version of the website. You want to minimize complexity. What should you do?

A. Deploy the new version in the same application and use the --migrate option.

B. Deploy the new version in the same application and use the --splits option to give a weight of 99 to the current version and a weight of 1 to the new version.

C. Create a new App Engine application in the same project. Deploy the new version in that application. Use the

App Engine library to proxy 1% of the requests to the new version.

D. Create a new App Engine application in the same project. Deploy the new version in that application. Configure your network load balancer to send 1% of the traffic to that new application.

**Answer: B**

**Explanation:**

Answer B is the most suitable choice because App Engine's --splits option during deployment directly supports traffic splitting between different versions within the same application, allowing you to easily route 1% of user traffic to the new version and 99% to the existing one by assigning corresponding weights. This method minimizes complexity by leveraging built-in App Engine features for version management and traffic control without requiring additional infrastructure or complex configurations. Option A's --migrate is for complete traffic migration, not splitting. Options C and D introduce unnecessary complexity by creating new applications and using proxying or load balancers, which are not needed for simple version-based traffic splitting within App Engine. Refer to the official Google Cloud documentation on App Engine traffic splitting for more details on using the --splits option: https://cloud.google.com/appengine/docs/standard/python/how-instances-are-routed#splitting_traffic.

**Question: 57** CertyIQ

You have a web application deployed as a managed instance group. You have a new version of the application to gradually deploy. Your web application is currently receiving live web traffic. You want to ensure that the available capacity does not decrease during the deployment. What should you do?

A. Perform a rolling-action start-update with maxSurge set to 0 and maxUnavailable set to 1.

B. Perform a rolling-action start-update with maxSurge set to 1 and maxUnavailable set to 0.

C. Create a new managed instance group with an updated instance template. Add the group to the backend service for the load balancer. When all instances in the new managed instance group are healthy, delete the old managed instance group.

D. Create a new instance template with the new application version. Update the existing managed instance group with the new instance template. Delete the instances in the managed instance group to allow the managed instance group to recreate the instance using the new instance template.

**Answer: B**

**Explanation:**

Option B is the correct answer because setting maxSurge to 1 allows the managed instance group to create a new instance before taking down an old one, ensuring that there is always at least the original capacity available during the rolling update. Setting maxUnavailable to 0 ensures that no instances are taken offline until a replacement is ready, further guaranteeing no reduction in serving capacity during the deployment. This configuration facilitates a safe and gradual rollout of the new application version without impacting the availability of the web application. Option A is incorrect because maxUnavailable=1 allows capacity to decrease. Option C is a valid deployment strategy (Blue/Green), but it is more complex than a rolling update for gradual deployment and might be an overkill. Option D will definitely cause a temporary capacity decrease as instances are deleted before being recreated.

**Justification Sentence:**

Option B is correct because a rolling update with maxSurge set to 1 and maxUnavailable set to 0 ensures that a new instance is created before an old one is removed, thus maintaining or slightly increasing capacity during the update and preventing any reduction in available serving capacity as per Google Cloud documentation on rolling updates and instance group update settings.

**Question: 58** CertyIQ

You are building an application that stores relational data from users. Users across the globe will use this application. Your CTO is concerned about the scaling requirements because the size of the user base is unknown. You need to implement a database solution that can scale with your user growth with minimum configuration changes. Which storage solution should you use?

A. Cloud SQL
B. Cloud Spanner
C. Cloud Firestore
D. Cloud Datastore

**Answer: B**

**Explanation:**

The correct answer is **B. Cloud Spanner**.

**Justification:** Cloud Spanner is a globally distributed, scalable, and strongly consistent relational database service specifically designed for applications requiring global scale and automatic scaling with minimal operational overhead. It automatically handles sharding, replication, and scaling across multiple regions, ensuring high availability and performance for a globally distributed user base with minimal configuration changes as user growth increases.

**Supporting Links:**

1. **Google Cloud Spanner Overview:** https://cloud.google.com/spanner/docs/overview - This documentation provides a general overview of Cloud Spanner, highlighting its key features such as global scalability and strong consistency for relational data.
2. **Cloud Spanner Global Scalability:** https://cloud.google.com/spanner/docs/instances#regional-vs-multiregional - This section explains how Cloud Spanner can be configured for regional or multi-regional deployments, enabling global scalability and high availability for applications with worldwide users.
3. **Cloud Spanner Autoscaling and Management:** https://cloud.google.com/spanner/docs/autoscaling/understanding-autoscaling - Although the term 'autoscaling' in Spanner refers to automated capacity management, the core principle is that Spanner is designed to handle scaling with minimal manual intervention, aligning with the requirement of minimal configuration changes for growth.

**Question: 59** CertyIQ

You are the organization and billing administrator for your company. The engineering team has the Project Creator role on the organization. You do not want the engineering team to be able to link projects to the billing account. Only the finance team should be able to link a project to a billing account, but they should not be able to make any other changes to projects. What should you do?

A. Assign the finance team only the Billing Account User role on the billing account.
B. Assign the engineering team only the Billing Account User role on the billing account.
C. Assign the finance team the Billing Account User role on the billing account and the Project Billing Manager role on the organization.

D. Assign the engineering team the Billing Account User role on the billing account and the Project Billing Manager role on the organization.

**Answer: A**

**Explanation:**

**Justification:** To allow the finance team to link projects to the billing account and prevent the engineering team from doing so, assign the finance team the Billing Account User role on the billing account. This role specifically grants the permission to link projects to the billing account without providing broader project management capabilities, thus fulfilling the requirement that only the finance team can link projects and not make other project changes. The engineering team, lacking this role, will be unable to link projects to the billing account, even with the Project Creator role.

**Supporting Links:**

**Billing roles:** https://cloud.google.com/billing/docs/how-to/billing-access (Specifically look for "Billing Account User")
**Predefined IAM roles:** https://cloud.google.com/iam/docs/understanding-roles (Search for "Billing Account User" and "Project Creator")

**Question: 60** CertyIQ

You have an application running in Google Kubernetes Engine (GKE) with cluster autoscaling enabled. The application exposes a TCP endpoint. There are several replicas of this application. You have a Compute Engine instance in the same region, but in another Virtual Private Cloud (VPC), called gce-network, that has no overlapping IP ranges with the first VPC. This instance needs to connect to the application on GKE. You want to minimize effort. What should you do?

A. 1. In GKE, create a Service of type LoadBalancer that uses the application's Pods as backend. 2. Set the service's externalTrafficPolicy to Cluster. 3. Configure the Compute Engine instance to use the address of the load balancer that has been created.

B. 1. In GKE, create a Service of type NodePort that uses the application's Pods as backend. 2. Create a Compute Engine instance called proxy with 2 network interfaces, one in each VPC. 3. Use iptables on this instance to forward traffic from gce-network to the GKE nodes. 4. Configure the Compute Engine instance to use the address of proxy in gce-network as endpoint.

C. 1. In GKE, create a Service of type LoadBalancer that uses the application's Pods as backend. 2. Add an annotation to this service: cloud.google.com/load-balancer-type: Internal 3. Peer the two VPCs together. 4. Configure the Compute Engine instance to use the address of the load balancer that has been created.

D. 1. In GKE, create a Service of type LoadBalancer that uses the application's Pods as backend. 2. Add a Cloud Armor Security Policy to the load balancer that whitelists the internal IPs of the MIG's instances. 3. Configure the Compute Engine instance to use the address of the load balancer that has been created.

**Answer: C**

**Explanation:**

Rationale: Option C is the most efficient solution because it leverages an Internal Load Balancer to expose the GKE application within its VPC, and utilizes VPC Peering to establish private network connectivity between the two VPCs, enabling the Compute Engine instance to access the application via the internal load balancer IP. This approach avoids public internet exposure, minimizes configuration effort by using managed services like Internal Load Balancer and VPC Peering, and ensures secure and private communication between the VPCs. Option A creates an external load balancer which is not ideal for internal communication and is more exposed. Option B is overly complex with manual proxy setup. Option D uses an external load balancer and adds unnecessary Cloud Armor complexity for internal connectivity.

Supported Links:

Internal Load Balancing: Explains how to set up internal load balancers in GKE.
VPC Peering: Describes VPC Peering and how it enables private communication between VPC networks.
Services in Kubernetes: General documentation on Kubernetes Services, including LoadBalancer types.

**Question: 61** CertyIQ

Your organization is a financial company that needs to store audit log files for 3 years. Your organization has hundreds of Google Cloud projects. You need to implement a cost-effective approach for log file retention. What should you do?

A. Create an export to the sink that saves logs from Cloud Audit to BigQuery.
B. Create an export to the sink that saves logs from Cloud Audit to a Coldline Storage bucket.
C. Write a custom script that uses logging API to copy the logs from Stackdriver logs to BigQuery.
D. Export these logs to Cloud Pub/Sub and write a Cloud Dataflow pipeline to store logs to Cloud SQL.

**Answer: B**

**Explanation:**

Option B is the most cost-effective solution because Cloud Storage Coldline is designed for infrequently accessed data with low storage costs, making it suitable for long-term archival of audit logs. Exporting logs to Coldline via a logging sink is a streamlined and managed service, avoiding the complexity and potential costs of custom scripts or data pipelines. BigQuery, while powerful for analysis, is more expensive for long-term storage than Coldline and less suitable for simple archival. Pub/Sub and Dataflow leading to Cloud SQL introduce unnecessary complexity and higher operational costs compared to direct export to Cloud Storage. Utilizing logging sinks for export provides a centralized and efficient mechanism to manage log retention across hundreds of projects. For detailed information on Cloud Storage pricing and storage classes, refer to the Cloud Storage Pricing documentation.

**Question: 62** CertyIQ

You want to run a single caching HTTP reverse proxy on GCP for a latency-sensitive website. This specific reverse proxy consumes almost no CPU. You want to have a 30-GB in-memory cache, and need an additional 2 GB of memory for the rest of the processes. You want to minimize cost. How should you run this reverse proxy?

A. Create a Cloud Memorystore for Redis instance with 32-GB capacity.
B. Run it on Compute Engine, and choose a custom instance type with 6 vCPUs and 32 GB of memory.
C. Package it in a container image, and run it on Kubernetes Engine, using n1-standard-32 instances as nodes.
D. Run it on Compute Engine, choose the instance type n1-standard-1, and add an SSD persistent disk of 32 GB.

**Answer: A**

**Explanation:**

Rationale: Cloud Memorystore for Redis is a fully managed, in-memory data store service, perfectly suited for caching and low-latency access, directly addressing the 30GB in-memory cache requirement. While Redis is not a reverse proxy itself, it can be effectively used as a cache layer for one. Compute Engine instances, particularly with custom configurations or larger instance types like n1-standard-32, can be more expensive than a managed Redis instance for this specific caching purpose, especially considering the low CPU utilization. Kubernetes Engine introduces unnecessary complexity and cost overhead for a single reverse proxy application. Using an SSD persistent disk (Option D) does not fulfill the in-memory cache requirement,

and would introduce higher latency compared to in-memory solutions like Redis. Hence, utilizing Cloud Memorystore for Redis is the most cost-effective and appropriate solution for a latency-sensitive website requiring a large in-memory cache for its reverse proxy.

Link for Redis as cache: https://redis.io/docs/use-cases/caching/Link for Cloud Memorystore: https://cloud.google.com/memorystore/docs/redis

**Question: 63** CertyIQ

You are hosting an application on bare-metal servers in your own data center. The application needs access to Cloud Storage. However, security policies prevent the servers hosting the application from having public IP addresses or access to the internet. You want to follow Google-recommended practices to provide the application with access to Cloud Storage. What should you do?

A. 1. Use nslookup to get the IP address for storage.googleapis.com. 2. Negotiate with the security team to be able to give a public IP address to the servers. 3. Only allow egress traffic from those servers to the IP addresses for storage.googleapis.com.

B. 1. Using Cloud VPN, create a VPN tunnel to a Virtual Private Cloud (VPC) in Google Cloud. 2. In this VPC, create a Compute Engine instance and install the Squid proxy server on this instance. 3. Configure your servers to use that instance as a proxy to access Cloud Storage.

C. 1. Use Migrate for Compute Engine (formerly known as Velostrata) to migrate those servers to Compute Engine. 2. Create an internal load balancer (ILB) that uses storage.googleapis.com as backend. 3. Configure your new instances to use this ILB as proxy.

D. 1. Using Cloud VPN or Interconnect, create a tunnel to a VPC in Google Cloud. 2. Use Cloud Router to create a custom route advertisement for 199.36.153.4/30. Announce that network to your on-premises network through the VPN tunnel. 3. In your on-premises network, configure your DNS server to resolve *.googleapis.com as a CNAME to restricted.googleapis.com.

**Answer: D**

**Explanation:**

Option D is the most suitable solution because it utilizes Private Google Access by establishing a secure connection via Cloud VPN or Interconnect and directs traffic to restricted.googleapis.com for Cloud Storage access, ensuring compliance with security policies by avoiding public internet exposure and aligning with Google's recommended practices for private network access to Google Cloud services.

**Supporting Links:**

**Private Google Access options:** https://cloud.google.com/vpc/docs/private-google-access
**Configuring Private Google Access for on-premises hosts:** https://cloud.google.com/vpc/docs/configure-private-google-access-on-premises
**Using Restricted Google Access:** https://cloud.google.com/vpc/docs/configure-private-google-access#restricted-google-access

**Question: 64** CertyIQ

You want to deploy an application on Cloud Run that processes messages from a Cloud Pub/Sub topic. You want to follow Google-recommended practices. What should you do?

A. 1. Create a Cloud Function that uses a Cloud Pub/Sub trigger on that topic. 2. Call your application on Cloud Run from the Cloud Function for every message.

B. 1. Grant the Pub/Sub Subscriber role to the service account used by Cloud Run. 2. Create a Cloud Pub/Sub subscription for that topic. 3. Make your application pull messages from that subscription.

C. 1. Create a service account. 2. Give the Cloud Run Invoker role to that service account for your Cloud Run

application. 3. Create a Cloud Pub/Sub subscription that uses that service account and uses your Cloud Run application as the push endpoint.

D. 1. Deploy your application on Cloud Run on GKE with the connectivity set to Internal. 2. Create a Cloud Pub/Sub subscription for that topic. 3. In the same Google Kubernetes Engine cluster as your application, deploy a container that takes the messages and sends them to your application.

**Answer: C**

**Explanation:**

Option C is the correct answer because it leverages Cloud Pub/Sub's push mechanism to directly send messages to your Cloud Run application as HTTP requests, which is a Google-recommended practice for integrating these services. Creating a dedicated service account and granting it the Cloud Run Invoker role ensures secure authorization for Pub/Sub to trigger your Cloud Run service. This approach avoids unnecessary intermediary services and aligns with serverless best practices by directly pushing messages to the application endpoint. Utilizing a push subscription with a service account for authentication provides a streamlined, secure, and scalable solution for processing Pub/Sub messages in Cloud Run.

**Supporting Links:**

**Cloud Run and Pub/Sub integration:** https://cloud.google.com/run/docs/integrate/pubsub
**Push subscriptions for Pub/Sub:** https://cloud.google.com/pubsub/docs/push
**Service accounts in Google Cloud:** https://cloud.google.com/iam/docs/service-accounts
**Cloud Run Invoker role:** https://cloud.google.com/run/docs/securing/managing-access

**Question: 65** CertyIQ

You need to deploy an application, which is packaged in a container image, in a new project. The application exposes an HTTP endpoint and receives very few requests per day. You want to minimize costs. What should you do?

A. Deploy the container on Cloud Run.

B. Deploy the container on Cloud Run on GKE.

C. Deploy the container on App Engine Flexible.

D. Deploy the container on GKE with cluster autoscaling and horizontal pod autoscaling enabled.

**Answer: A**

**Explanation:**

**Justification:** Cloud Run is the most cost-effective option because it is a serverless platform that automatically scales to zero when not in use, ensuring you only pay for request processing and not for idle resources, which is ideal for applications with very few requests per day. This eliminates the need to manage servers or pay for instance uptime when the application is not actively serving requests, unlike App Engine Flexible or GKE based solutions that incur costs even during periods of low traffic. Cloud Run's pay-per-use model directly aligns with the requirement of minimizing costs for applications with infrequent usage.

**Supporting Links:**

**Cloud Run Pricing:** https://cloud.google.com/run/pricing - This page details Cloud Run's pricing model, emphasizing the pay-per-use nature and scaling to zero.
**Serverless Computing on Google Cloud:** https://cloud.google.com/serverless - Provides an overview of serverless computing on Google Cloud and highlights the benefits of cost optimization and automatic scaling, which are key features of Cloud Run.
**Choosing a compute option:** https://cloud.google.com/compute/docs/choose-compute-options - This

documentation helps in selecting the appropriate compute option on Google Cloud based on various factors, including cost and application requirements, and often recommends Cloud Run for cost-optimized, containerized applications with variable traffic.

**Question: 66** CertyIQ

Your company has an existing GCP organization with hundreds of projects and a billing account. Your company recently acquired another company that also has hundreds of projects and its own billing account. You would like to consolidate all GCP costs of both GCP organizations onto a single invoice. You would like to consolidate all costs as of tomorrow. What should you do?

A. Link the acquired company's projects to your company's billing account.

B. Configure the acquired company's billing account and your company's billing account to export the billing data into the same BigQuery dataset.

C. Migrate the acquired company's projects into your company's GCP organization. Link the migrated projects to your company's billing account.

D. Create a new GCP organization and a new billing account. Migrate the acquired company's projects and your company's projects into the new GCP organization and link the projects to the new billing account.

**Answer: A**

**Explanation:**

To consolidate GCP costs onto a single invoice by tomorrow, link the acquired company's projects to your company's billing account, as this directly aggregates all project spending under one billing account for unified invoicing. This approach is the most immediate and efficient way to achieve the goal of consolidated billing for both organizations. Linking projects to a central billing account ensures all associated costs are billed together, fulfilling the requirement for a single invoice. This method avoids the complexities and time delays associated with organization or project migrations, allowing for near-immediate cost consolidation. Reviewing the "Billing overview" and "Manage billing for your Google Cloud organization" documentation confirms that linking projects to a billing account is the standard practice for consolidating billing.

**Supporting Links:**

1. **Billing overview**: https://cloud.google.com/billing/docs/concepts/billing-overview
2. **Manage billing for your Google Cloud organization**: https://cloud.google.com/billing/docs/how-to/manage-billing-account-org

**Question: 67** CertyIQ

You built an application on Google Cloud that uses Cloud Spanner. Your support team needs to monitor the environment but should not have access to table data.
You need a streamlined solution to grant the correct permissions to your support team, and you want to follow Google-recommended practices. What should you do?

A. Add the support team group to the roles/monitoring.viewer role

B. Add the support team group to the roles/spanner.databaseUser role.

C. Add the support team group to the roles/spanner.databaseReader role.

D. Add the support team group to the roles/stackdriver.accounts.viewer role.

**Answer: A**

**Explanation:**

Granting the roles/monitoring.viewer role allows the support team to view monitoring data, including Spanner metrics, without granting access to the sensitive table data itself, which aligns with the principle of least privilege. The roles/spanner.databaseUser role grants access to read and modify data, which is not desired. Similarly, roles/spanner.databaseReader also grants access to read data. While roles/stackdriver.accounts.viewer allows viewing Stackdriver accounts information, it does not provide necessary metrics for Spanner monitoring. Therefore, roles/monitoring.viewer is the most appropriate role for this scenario.

Supporting links:

Predefined IAM roles for Cloud Monitoring
Predefined IAM roles for Cloud Spanner

**Question: 68** CertyIQ

For analysis purposes, you need to send all the logs from all of your Compute Engine instances to a BigQuery dataset called platform-logs. You have already installed the Cloud Logging agent on all the instances. You want to minimize cost. What should you do?

A. 1. Give the BigQuery Data Editor role on the platform-logs dataset to the service accounts used by your instances. 2. Update your instances' metadata to add the following value: logs-destination: bq://platform-logs.

B. 1. In Cloud Logging, create a logs export with a Cloud Pub/Sub topic called logs as a sink. 2. Create a Cloud Function that is triggered by messages in the logs topic. 3. Configure that Cloud Function to drop logs that are not from Compute Engine and to insert Compute Engine logs in the platform-logs dataset.

C. 1. In Cloud Logging, create a filter to view only Compute Engine logs. 2. Click Create Export. 3. Choose BigQuery as Sink Service, and the platform-logs dataset as Sink Destination.

D. 1. Create a Cloud Function that has the BigQuery User role on the platform-logs dataset. 2. Configure this Cloud Function to create a BigQuery Job that executes this query: INSERT INTO dataset.platform-logs (timestamp, log) SELECT timestamp, log FROM compute.logs WHERE timestamp > DATE_SUB(CURRENT_DATE(), INTERVAL 1 DAY) 3. Use Cloud Scheduler to trigger this Cloud Function once a day.

**Answer: C**

**Explanation:**

Option C is the most cost-effective approach because Cloud Logging exports directly to BigQuery, avoiding the need for intermediate services like Cloud Pub/Sub and Cloud Functions, which would incur additional costs. Using a filter in Cloud Logging ensures only the desired Compute Engine logs are exported, minimizing data ingestion into BigQuery. This method leverages the built-in functionality of Cloud Logging for direct data transfer to BigQuery, optimizing both operational overhead and cost. Cloud Logging export documentation, BigQuery Sink documentation

**Question: 69** CertyIQ

You are using Deployment Manager to create a Google Kubernetes Engine cluster. Using the same Deployment Manager deployment, you also want to create a
DaemonSet in the kube-system namespace of the cluster. You want a solution that uses the fewest possible services. What should you do?

A. Add the cluster's API as a new Type Provider in Deployment Manager, and use the new type to create the DaemonSet.

B. Use the Deployment Manager Runtime Configurator to create a new Config resource that contains the DaemonSet definition.

C. With Deployment Manager, create a Compute Engine instance with a startup script that uses kubectl to create the DaemonSet.

D. In the cluster's definition in Deployment Manager, add a metadata that has kube-system as key and the DaemonSet manifest as value.

**Answer: A**

**Explanation:**

**Justification for Answer A:**

The most efficient and service-minimal approach for creating both a GKE cluster and a DaemonSet within the same Deployment Manager deployment is to utilize the cluster's API as a Type Provider. Option A directly leverages Deployment Manager's capacity to interact with external APIs, allowing it to manage Kubernetes resources as if they were native Deployment Manager resources. By adding the GKE API as a type provider, you gain the ability to define and manage Kubernetes objects, like DaemonSets, using YAML or Jinja2 templates directly within your Deployment Manager configuration. This avoids creating intermediary resources or adding unnecessary complexity.

Option B, using the Runtime Configurator, is less suitable for this purpose. The Configurator is primarily for managing configuration data across services, not for deploying Kubernetes resources. It adds an extra layer of abstraction and complexity. Option C, creating a Compute Engine instance with a startup script, introduces an unnecessary VM and requires configuring kubectl, which adds more moving parts and increases the chances of failure and management overhead. Option D, using metadata, is not a valid way to create Kubernetes resources and lacks the proper declarative approach of Deployment Manager.

Using the cluster's API as a Type Provider offers a streamlined solution that utilizes the strengths of Deployment Manager's resource management capabilities. It reduces the need for additional services and promotes a more declarative infrastructure-as-code approach, aligning well with cloud best practices. This method is more maintainable and less error-prone.

**Relevant Links for further research:**

**Deployment Manager Type Providers:** https://cloud.google.com/deployment-manager/docs/configuration/type-providers
**Google Cloud Platform Deployment Manager:** https://cloud.google.com/deployment-manager/docs/
**Kubernetes API:** https://kubernetes.io/docs/reference/kubernetes-api/
**GKE API (for Deployment Manager):** https://cloud.google.com/deployment-manager/docs/reference/type/gcp-kubernetes-engine-v1beta1-cluster

**Question: 70** CertyIQ

You are building an application that will run in your data center. The application will use Google Cloud Platform (GCP) services like AutoML. You created a service account that has appropriate access to AutoML. You need to enable authentication to the APIs from your on-premises environment. What should you do?

A. Use service account credentials in your on-premises application.

B. Use gcloud to create a key file for the service account that has appropriate permissions.

C. Set up direct interconnect between your data center and Google Cloud Platform to enable authentication for your on-premises applications.

D. Go to the IAM & admin console, grant a user account permissions similar to the service account permissions, and use this user account for authentication from your data center.

**Answer: B**

**Explanation:**

The correct answer is B, "Use gcloud to create a key file for the service account that has appropriate permissions." Here's why:

Service accounts are the recommended way for applications to authenticate with Google Cloud APIs. Unlike user accounts, they don't require interactive logins. To use a service account from outside GCP (like your on-premises data center), you need a private key associated with it. This private key allows your application to prove its identity.

Option B directly addresses this need. The gcloud command-line tool can generate a JSON key file containing the service account's private key. Your on-premises application can then use this key file to authenticate with GCP, gaining the permissions associated with the service account. This approach is secure and efficient.

Option A is incorrect because directly embedding service account credentials in your application's code is a significant security risk. It exposes the private key and should be avoided.

Option C, setting up direct interconnect, is relevant for high-bandwidth, low-latency connections but is not primarily for authentication. While it allows for more efficient data transfer, it doesn't directly address how your on-premises application authenticates with GCP APIs. It's overkill for the task.

Option D is also incorrect because using a user account for application authentication goes against best practices. User accounts are intended for human users, not automated processes. Furthermore, managing user accounts and distributing their credentials for application authentication is cumbersome and less secure compared to service accounts.

In summary, using a service account key file generated by gcloud provides a secure and practical way to authenticate your on-premises application with GCP APIs, in line with recommended security practices.

**Supporting Links:**

**Google Cloud Service Accounts:** https://cloud.google.com/iam/docs/service-accounts
**Creating Service Account Keys:** https://cloud.google.com/iam/docs/creating-managing-service-account-keys
**Authenticating as a Service Account:** https://cloud.google.com/docs/authentication/getting-started

**Question: 71** CertyIQ

You are using Container Registry to centrally store your company's container images in a separate project. In another project, you want to create a Google
Kubernetes Engine (GKE) cluster. You want to ensure that Kubernetes can download images from Container Registry. What should you do?

A. In the project where the images are stored, grant the Storage Object Viewer IAM role to the service account used by the Kubernetes nodes.

B. When you create the GKE cluster, choose the Allow full access to all Cloud APIs option under 'Access scopes'.

C. Create a service account, and give it access to Cloud Storage. Create a P12 key for this service account and use it as an imagePullSecrets in Kubernetes.

D. Configure the ACLs on each image in Cloud Storage to give read-only access to the default Compute Engine service account.

**Answer: A**

**Explanation:**

The correct answer is A. To enable Kubernetes nodes in one project to pull images from Container Registry in another project, you need to grant the appropriate permissions. Container Registry utilizes Cloud Storage for

image storage, and access to these storage objects is controlled by Identity and Access Management (IAM). Kubernetes nodes typically operate under a service account, often the Compute Engine default service account or a custom service account defined during node pool creation.

Option A directly addresses this need by granting the "Storage Object Viewer" IAM role to the service account used by the Kubernetes nodes in the GKE cluster's project. This role provides the necessary read-only permissions to access the container images stored in the Container Registry project's Cloud Storage bucket. Without this permission, Kubernetes will be unable to download the images, and deployments will fail.

Option B is incorrect because granting "Allow full access to all Cloud APIs" is overly permissive and not a best practice. It provides unnecessary access beyond what is needed for pulling container images, potentially increasing the attack surface.

Option C is also incorrect. Manually creating service accounts and managing P12 keys for imagePullSecrets adds unnecessary complexity and is not the recommended way to handle cross-project Container Registry access. IAM roles provide a more centralized and secure solution.

Option D is incorrect. While it aims to grant access, it attempts to do so on each image individually using ACLs, which is highly impractical for managing access to a large number of images. IAM is the preferred method for controlling access at the bucket or project level.

Therefore, granting the Storage Object Viewer role to the Kubernetes node service account provides the least privilege approach to enabling cross-project image pulling. This solution adheres to security best practices and is the most straightforward and manageable method.

**Relevant Links:**

**Granting access to Container Registry:** https://cloud.google.com/container-registry/docs/access-control
**IAM roles for Cloud Storage:** https://cloud.google.com/storage/docs/access-control/iam-roles
**GKE node identity:** https://cloud.google.com/kubernetes-engine/docs/how-to/workload-identity

**Question: 72** CertyIQ

You deployed a new application inside your Google Kubernetes Engine cluster using the YAML file specified below.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: myapp-deployment
spec:
  selector:
    matchLabels:
      app: myapp
  replicas: 2
  template:
    metadata:
      labels:
        app: myapp
    spec:
      containers:
      - name: myapp
        image: myapp:1.1
        ports:
        - containerPort: 80
```

```
apiVersion: v1
kind: Service
metadata:
  name: myapp-service
spec:
  ports:
  - port: 8000
    targetPort: 80
    protocol: TCP
  selector:
    app: myapp
```

You check the status of the deployed pods and notice that one of them is still in PENDING status:

```
kubectl get pods -l app=myapp
NAME                                READY     STATUS    RESTART   AGE
myapp-deployment-58ddbbb995-lp86m   0/1       Pending   0         9m
myapp-deployment-58ddbbb995-qjpkg   1/1       Running   0         9m
```

You want to find out why the pod is stuck in pending status. What should you do?

A. Review details of the myapp-service Service object and check for error messages.
B. Review details of the myapp-deployment Deployment object and check for error messages.
C. Review details of myapp-deployment-58ddbbb995-lp86m Pod and check for warning messages.
D. View logs of the container in myapp-deployment-58ddbbb995-lp86m pod and check for warning messages.

**Answer: C**

**Explanation:**

C is the correct answer.

If a Pod is stuck in Pending it means that it can not be scheduled onto a node. Generally this is because there are insufficient resources of one type or another that prevent scheduling. Look at the output of the kubectl describe ... command above. There should be messages from the scheduler about why it can not schedule your Pod

Reference:

https://cloud.google.com/run/docs/gke/troubleshooting

**Question: 73** CertyIQ

You are setting up a Windows VM on Compute Engine and want to make sure you can log in to the VM via RDP. What should you do?

A. After the VM has been created, use your Google Account credentials to log in into the VM.

B. After the VM has been created, use gcloud compute reset-windows-password to retrieve the login credentials for the VM.

C. When creating the VM, add metadata to the instance using 'windows-password' as the key and a password as the value.

D. After the VM has been created, download the JSON private key for the default Compute Engine service account. Use the credentials in the JSON file to log in to the VM.

**Answer: B**

**Explanation:**

The correct answer is B. Here's a detailed justification:

When setting up a Windows VM on Google Compute Engine, standard Google account credentials won't directly work for RDP login. Windows instances require a local user account and password. Option A is incorrect because it incorrectly suggests that your Google account credentials will work for the Windows VM's login. Option C is also incorrect as it is not the appropriate method for setting a Windows password using metadata. While metadata can be used to configure various aspects of an instance, handling sensitive data like passwords this way is generally discouraged for security reasons. Option D is incorrect because the service account's JSON key is used for programmatic access to Google Cloud services and not for direct RDP logins to the VM.

The gcloud compute reset-windows-password command is specifically designed to handle the initial setup of a Windows user account and password on Compute Engine. It generates a new password and associates it with a default user, typically "administrator". After executing this command, you can use the generated username and password to log into the Windows VM via RDP. This process provides a secure and reliable way to gain initial access to your Windows instance. This command doesn't modify the default service account configuration; rather, it creates a local user account specifically for the VM.

This approach is a standard and recommended practice when managing Windows VMs in Google Cloud. It ensures that user credentials are created securely and in a way that's compatible with Windows authentication mechanisms.

For more information on managing Windows passwords on Compute Engine, you can refer to the official Google Cloud documentation:

Setting up Windows passwords
Resetting Windows passwords

**Question: 74** CertyIQ

You want to configure an SSH connection to a single Compute Engine instance for users in the dev1 group. This instance is the only resource in this particular
Google Cloud Platform project that the dev1 users should be able to connect to. What should you do?

A. Set metadata to enable-oslogin=true for the instance. Grant the dev1 group the compute.osLogin role. Direct them to use the Cloud Shell to ssh to that instance.

B. Set metadata to enable-oslogin=true for the instance. Set the service account to no service account for that instance. Direct them to use the Cloud Shell to ssh to that instance.

C. Enable block project wide keys for the instance. Generate an SSH key for each user in the dev1 group. Distribute the keys to dev1 users and direct them to use their third-party tools to connect.

D. Enable block project wide keys for the instance. Generate an SSH key and associate the key with that instance. Distribute the key to dev1 users and direct them to use their third-party tools to connect.

**Answer: A**

**Explanation:**

The correct answer is **A**. Here's why:

Option A leverages OS Login, a Google Cloud service that manages SSH access using IAM roles. By setting enable-oslogin=true in the instance metadata, we enable OS Login for that specific instance. Then, granting the compute.osLogin role to the dev1 group allows members of that group to establish SSH connections to instances configured for OS Login. Importantly, OS Login uses Google identities, not instance-specific keys, and grants access based on IAM roles. Cloud Shell provides a convenient browser-based SSH client, making the connection process streamlined and managed. This adheres to the principle of least privilege as dev1 users are granted only the necessary access for SSH and the service is used for user access management rather than a key distribution scheme.

Options B, C, and D are incorrect: Option B, disabling the service account, does not relate to the ssh use case and might break instance functionality. Option C and D introduce individual key management overhead, are less scalable than OS login and can easily become a security risk. Relying on distributing SSH keys to multiple users, especially through third-party tools, creates difficulties in user access management and key rotation, potentially increasing security vulnerabilities. These manual key distribution methods also increase the complexity and administrative burden compared to the managed OS Login solution.

https://cloud.google.com/compute/docs/oslogin/setup/enable-osloginhttps://cloud.google.com/iam/docs/understanding-roles#compute.osloginhttps://cloud.google.com/shell/docs/using-cloud-shell

**Question: 75** CertyIQ

You need to produce a list of the enabled Google Cloud Platform APIs for a GCP project using the gcloud command line in the Cloud Shell. The project name is my-project. What should you do?

A. Run gcloud projects list to get the project ID, and then run gcloud services list --project <project ID>.

B. Run gcloud init to set the current project to my-project, and then run gcloud services list --available.

C. Run gcloud info to view the account value, and then run gcloud services list --account <Account>.

D. Run gcloud projects describe <project ID> to verify the project value, and then run gcloud services list --available.

**Answer: A**

**Explanation:**

Here's a detailed justification for why option A is the correct approach to list enabled APIs in a GCP project using gcloud:

Option A, gcloud projects list followed by gcloud services list --project <project ID>, is the correct method because it directly addresses the task. First, gcloud projects list is used to retrieve a list of all projects associated with the current account. This step is crucial because, although the question specifies the project

name (my-project), gcloud commands often require the project ID (a unique, alphanumeric identifier), which might be different from the name. The command provides output that allows users to determine the specific ID corresponding to a project name. The second part, gcloud services list --project <project ID>, then uses the extracted project ID to list all services (APIs) enabled within that particular project. The --project flag is essential as it targets the operation to the specified project.

Options B, C, and D are incorrect. Option B uses gcloud init, which is for configuring gcloud, not listing APIs. It also uses --available, which lists all available APIs, not just enabled ones for the project. Option C uses gcloud info to get account info, which is not relevant for listing project APIs, and it also uses --account which doesn't apply to services. Option D uses gcloud projects describe, which provides details of a project, not the enabled APIs. Like option B, option D utilizes --available, which, again, lists all available APIs, not just the ones enabled in the specified project.

Therefore, option A accurately outlines the necessary steps to retrieve the project ID and subsequently list the enabled APIs for a specific project.

Relevant documentation:

gcloud projects list
gcloud services list

**Question: 76** CertyIQ

You are building a new version of an application hosted in an App Engine environment. You want to test the new version with 1% of users before you completely switch your application over to the new version. What should you do?

A. Deploy a new version of your application in Google Kubernetes Engine instead of App Engine and then use GCP Console to split traffic.

B. Deploy a new version of your application in a Compute Engine instance instead of App Engine and then use GCP Console to split traffic.

C. Deploy a new version as a separate app in App Engine. Then configure App Engine using GCP Console to split traffic between the two apps.

D. Deploy a new version of your application in App Engine. Then go to App Engine settings in GCP Console and split traffic between the current version and newly deployed versions accordingly.

**Answer: D**

**Explanation:**

The correct answer is **D**. Here's why: App Engine is designed for seamless version management and traffic splitting, making it ideal for A/B testing or gradual rollouts. Option D leverages this built-in capability directly. You deploy the new version alongside the existing one within the same App Engine application. The GCP Console's App Engine settings then allow you to precisely control the percentage of traffic directed to each version. Specifically, you can set a 1% traffic split to the new version while 99% remains on the old version, achieving the desired testing scenario. This approach minimizes complexity and avoids unnecessary infrastructure changes.

Options A and B involve migrating away from App Engine to either Google Kubernetes Engine (GKE) or Compute Engine. These options are overkill for a simple traffic split and introduce unnecessary management overhead. While GKE and Compute Engine are powerful platforms, they require additional setup and configurations for traffic management that App Engine already handles. Option C suggests deploying the new version as a completely separate App Engine application. This approach while possible, complicates resource management and traffic routing significantly as you'd have two separate app deployments. The easiest most efficient solution is to utilize App Engines built in versioning and traffic splitting within the same app

deployment.

Key concepts supporting this answer:

**App Engine Versioning:** App Engine allows multiple versions of the same application to run concurrently.
**Traffic Splitting:** App Engine provides a mechanism to distribute incoming requests across different versions.
**Gradual Rollouts:** The traffic splitting feature is perfect for gradually introducing changes and monitoring their impact.
**Minimizing Complexity:** Choosing the built-in feature of App Engine aligns with best practices for simplicity and efficiency.

Authoritative links:

**App Engine documentation on Traffic Splitting:**
https://cloud.google.com/appengine/docs/standard/python/how-to/traffic-splitting
**App Engine documentation on Versioning:** https://cloud.google.com/appengine/docs/standard/python/how-to/managing-versions

**Question: 77** CertyIQ

You need to provide a cost estimate for a Kubernetes cluster using the GCP pricing calculator for Kubernetes. Your workload requires high IOPs, and you will also be using disk snapshots. You start by entering the number of nodes, average hours, and average days. What should you do next?

A. Fill in local SSD. Fill in persistent disk storage and snapshot storage.
B. Fill in local SSD. Add estimated cost for cluster management.
C. Select Add GPUs. Fill in persistent disk storage and snapshot storage.
D. Select Add GPUs. Add estimated cost for cluster management.

**Answer: A**

**Explanation:**

To accurately estimate costs for a Kubernetes cluster requiring high IOPs and disk snapshots in the GCP pricing calculator, after entering basic node details, the next step is to **fill in local SSD** to address the high IOPs requirement and **fill in persistent disk storage and snapshot storage** to account for storage costs and snapshot needs. Local SSDs are chosen for high IOPs, while persistent disks and snapshots cover storage and data protection aspects. Options B, C, and D are less relevant as cluster management is a general cost and GPUs are not directly related to high IOPs storage requirements. Option A directly addresses both the high IOPs and snapshot requirements mentioned in the question, making it the most appropriate next step for accurate cost estimation.

**Justification:** To accurately estimate the cost for a Kubernetes cluster requiring high IOPs and disk snapshots, after specifying the basic node details, you should next configure local SSD for high IOPs performance and include the costs for persistent disk storage and snapshot storage to reflect the storage requirements of the workload.

**Supporting Links:**

**GCP Pricing Calculator:** https://cloud.google.com/products/calculator
**Kubernetes Storage Options on GCP:** https://cloud.google.com/kubernetes-engine/docs/concepts/storage
**Persistent Disk Snapshots:** https://cloud.google.com/compute/docs/disks/snapshots
**Local SSDs:** https://cloud.google.com/compute/docs/disks/local-ssd

**Question: 78** CertyIQ

You are using Google Kubernetes Engine with autoscaling enabled to host a new application. You want to expose this new application to the public, using HTTPS on a public IP address. What should you do?

A. Create a Kubernetes Service of type NodePort for your application, and a Kubernetes Ingress to expose this Service via a Cloud Load Balancer.

B. Create a Kubernetes Service of type ClusterIP for your application. Configure the public DNS name of your application using the IP of this Service.

C. Create a Kubernetes Service of type NodePort to expose the application on port 443 of each node of the Kubernetes cluster. Configure the public DNS name of your application with the IP of every node of the cluster to achieve load-balancing.

D. Create a HAProxy pod in the cluster to load-balance the traffic to all the pods of the application. Forward the public traffic to HAProxy with an iptable rule. Configure the DNS name of your application using the public IP of the node HAProxy is running on.

**Answer: A**

**Explanation:**

Option A is the correct answer because it leverages Kubernetes Ingress, which when used in Google Kubernetes Engine, automatically provisions a Google Cloud Load Balancer to expose the application publicly. This setup allows for HTTPS termination at the load balancer and provides a stable public IP address, fulfilling the requirement of exposing the application via HTTPS on a public IP. Creating a NodePort Service makes the application accessible within the cluster and provides a backend for the Ingress to route traffic to the application pods. This approach is the most managed and scalable way to expose applications externally in GKE, ensuring high availability and automatic integration with Google Cloud's networking infrastructure.

**Supporting Links:**

**Kubernetes Ingress:** https://kubernetes.io/docs/concepts/services-networking/ingress/
**Google Cloud Load Balancer:** https://cloud.google.com/load-balancing
**Exposing Services - Kubernetes Documentation:** https://kubernetes.io/docs/tasks/access-application-cluster/service-access-application-cluster/

**Question: 79** CertyIQ

You need to enable traffic between multiple groups of Compute Engine instances that are currently running two different GCP projects. Each group of Compute
Engine instances is running in its own VPC. What should you do?

A. Verify that both projects are in a GCP Organization. Create a new VPC and add all instances.

B. Verify that both projects are in a GCP Organization. Share the VPC from one project and request that the Compute Engine instances in the other project use this shared VPC.

C. Verify that you are the Project Administrator of both projects. Create two new VPCs and add all instances.

D. Verify that you are the Project Administrator of both projects. Create a new VPC and add all instances.

**Answer: B**

**Explanation:**

The correct answer is **B**. Here's the justification:

To enable traffic between Compute Engine instances in different GCP projects and VPCs, Shared VPC is the recommended approach. Shared VPC allows you to designate one project as a host project and share its VPC network with other service projects, enabling instances in different projects to communicate securely and

privately over the shared VPC. Verifying that both projects are within the same GCP Organization is a prerequisite for Shared VPC as it simplifies administration and billing. By sharing a VPC, you avoid the complexity and potential security concerns of routing traffic over the public internet or through external IP addresses, and it's more efficient than migrating instances to a new VPC.

Supporting Links:

1. **Shared VPC Overview**: https://cloud.google.com/vpc/docs/shared-vpc
2. **Setting up Shared VPC**: https://cloud.google.com/vpc/docs/shared-vpc/set-up-shared-vpc

**Question: 80** CertyIQ

You want to add a new auditor to a Google Cloud Platform project. The auditor should be allowed to read, but not modify, all project items.
How should you configure the auditor's permissions?

A. Create a custom role with view-only project permissions. Add the user's account to the custom role.
B. Create a custom role with view-only service permissions. Add the user's account to the custom role.
C. Select the built-in IAM project Viewer role. Add the user's account to this role.
D. Select the built-in IAM service Viewer role. Add the user's account to this role.

**Answer: C**

**Explanation:**

The most efficient way to grant read-only access to all project items for an auditor is by using the built-in IAM Project Viewer role, as it is specifically designed to provide comprehensive read-only permissions at the project level. This predefined role includes permissions to view resources across all GCP services within the project without the ability to modify them. Creating a custom role with view-only permissions is also an option but adds unnecessary complexity when a suitable built-in role already exists. Service-level viewer roles would not grant access to all project items and would require more granular management, making the Project Viewer role the most straightforward and appropriate solution. Using built-in roles like Viewer aligns with Google Cloud's best practices for IAM.

**Supporting Links:**

1. **IAM Roles - Predefined roles:** https://cloud.google.com/iam/docs/understanding-roles#predefined_roles
2. **IAM Roles - Project roles:** https://cloud.google.com/iam/docs/understanding-roles#project-roles
3. **IAM Best Practices - Use predefined roles:** https://cloud.google.com/iam/docs/best-practices#grant-least-privilege

**Question: 81** CertyIQ

You are operating a Google Kubernetes Engine (GKE) cluster for your company where different teams can run non-production workloads. Your Machine Learning
(ML) team needs access to Nvidia Tesla P100 GPUs to train their models. You want to minimize effort and cost.
What should you do?

A. Ask your ML team to add the accelerator: gpu annotation to their pod specification.
B. Recreate all the nodes of the GKE cluster to enable GPUs on all of them.
C. Create your own Kubernetes cluster on top of Compute Engine with nodes that have GPUs. Dedicate this cluster to your ML team.

D. Add a new, GPU-enabled, node pool to the GKE cluster. Ask your ML team to add the cloud.google.com/gke-accelerator: nvidia-tesla-p100 nodeSelector to their pod specification.

**Answer: D**

**Explanation:**

The most efficient and cost-effective solution is to add a dedicated GPU-enabled node pool to your existing GKE cluster and use a nodeSelector in the ML team's pod specifications to ensure their workloads are scheduled on these GPU nodes, minimizing cost by only provisioning GPUs where needed and reducing management overhead by leveraging the existing cluster infrastructure.

Supporting Links:

1. **GKE Node Pools:** https://cloud.google.com/kubernetes-engine/docs/concepts/node-pools - This documentation explains the concept of node pools in GKE, allowing for heterogeneous node configurations within a cluster.
2. **Adding GPUs to Node Pools:** https://cloud.google.com/kubernetes-engine/docs/how-to/gpus#add-gpu-node-pool - This documentation specifically details how to add GPU-enabled node pools to a GKE cluster.
3. **Assigning Pods to Nodes:** https://kubernetes.io/docs/concepts/scheduling-eviction/assign-pod-node/ - This Kubernetes documentation explains node selectors and how they are used to constrain pods to run on specific nodes based on labels, which is essential for directing ML workloads to the GPU node pool.

**Question: 82** CertyIQ

Your VMs are running in a subnet that has a subnet mask of 255.255.255.240. The current subnet has no more free IP addresses and you require an additional
10 IP addresses for new VMs. The existing and new VMs should all be able to reach each other without additional routes. What should you do?

A. Use gcloud to expand the IP range of the current subnet.

B. Delete the subnet, and recreate it using a wider range of IP addresses.

C. Create a new project. Use Shared VPC to share the current network with the new project.

D. Create a new subnet with the same starting IP but a wider range to overwrite the current subnet.

**Answer: A**

**Explanation:**

Expanding the IP range of the existing subnet using gcloud is the correct solution because Google Cloud allows you to expand a subnet's IP range without needing to delete and recreate it, thus avoiding disruption to running VMs. This approach directly addresses the need for additional IP addresses within the current subnet, ensuring seamless connectivity for both existing and new VMs without requiring additional routing configurations. Option B is disruptive and unnecessary, Option C is an overly complex solution for a simple IP address shortage, and Option D is not a valid operation in Google Cloud Networking as subnets cannot be overwritten in that manner. For more details, refer to the Google Cloud documentation on expanding subnet IP ranges: https://cloud.google.com/vpc/docs/expand-subnet and the gcloud compute networks subnets expand-ip-range command documentation:
https://cloud.google.com/sdk/gcloud/reference/compute/networks/subnets/expand-ip-range.

**Question: 83** CertyIQ

Your organization uses G Suite for communication and collaboration. All users in your organization have a G Suite account. You want to grant some G Suite users access to your Cloud Platform project. What should you do?

A. Enable Cloud Identity in the GCP Console for your domain.

B. Grant them the required IAM roles using their G Suite email address.

C. Create a CSV sheet with all users' email addresses. Use the gcloud command line tool to convert them into Google Cloud Platform accounts.

D. In the G Suite console, add the users to a special group called [email protected] Rely on the default behavior of the Cloud Platform to grant users access if they are members of this group.

**Answer: B**

**Explanation:**

**Option B is correct because** Google Cloud Platform (GCP) Identity and Access Management (IAM) directly integrates with Google Accounts, which G Suite accounts are. You can grant IAM roles to users by using their G Suite email addresses, enabling them to access your Cloud Platform project resources according to the assigned roles. This method leverages the existing identity management provided by G Suite, making it the simplest and most direct approach for granting access in this scenario. Option A is not necessary as G Suite already provides identity. Option C is incorrect as G Suite accounts are already valid Google Cloud accounts and do not need conversion. Option D is misleading as there is no default special group like [email protected] that automatically grants GCP access.

**Supporting Links:**

**IAM Overview:** https://cloud.google.com/iam/docs/overview
**Granting, changing, and revoking access to resources:** https://cloud.google.com/iam/docs/granting-changing-revoking-access

**Question: 84** CertyIQ

You have a Google Cloud Platform account with access to both production and development projects. You need to create an automated process to list all compute instances in development and production projects on a daily basis. What should you do?

A. Create two configurations using gcloud config. Write a script that sets configurations as active, individually. For each configuration, use gcloud compute instances list to get a list of compute resources.

B. Create two configurations using gsutil config. Write a script that sets configurations as active, individually. For each configuration, use gsutil compute instances list to get a list of compute resources.

C. Go to Cloud Shell and export this information to Cloud Storage on a daily basis.

D. Go to GCP Console and export this information to Cloud SQL on a daily basis.

**Answer: A**

**Explanation:**

**Justification:** Option A is the most effective solution because it utilizes gcloud config to manage project-specific configurations, allowing for seamless switching between development and production environments within a script. By activating each configuration and using gcloud compute instances list, the script can systematically retrieve compute instance details from both projects. This approach ensures automation and leverages the appropriate gcloud commands for managing configurations and listing compute resources as required.

**Supporting Links:**

**gcloud config configurations create:**
https://cloud.google.com/sdk/gcloud/reference/config/configurations/create
**gcloud config configurations activate:**
https://cloud.google.com/sdk/gcloud/reference/config/configurations/activate
**gcloud compute instances list:** https://cloud.google.com/sdk/gcloud/reference/compute/instances/list

**Question: 85** CertyIQ

You have a large 5-TB AVRO file stored in a Cloud Storage bucket. Your analysts are proficient only in SQL and need access to the data stored in this file. You want to find a cost-effective way to complete their request as soon as possible. What should you do?

A. Load data in Cloud Datastore and run a SQL query against it.

B. Create a BigQuery table and load data in BigQuery. Run a SQL query on this table and drop this table after you complete your request.

C. Create external tables in BigQuery that point to Cloud Storage buckets and run a SQL query on these external tables to complete your request.

D. Create a Hadoop cluster and copy the AVRO file to NDFS by compressing it. Load the file in a hive table and provide access to your analysts so that they can run SQL queries.

**Answer: C**

**Explanation:**

Option C is the most cost-effective and quickest solution because BigQuery external tables allow querying AVRO files directly in Cloud Storage without loading data into BigQuery storage, thus saving on storage costs and data loading time. This approach enables analysts to use their SQL skills immediately on the data residing in Cloud Storage, fulfilling the requirements efficiently. BigQuery is designed for querying large datasets and supports AVRO format natively, making external tables an ideal solution for this scenario. Options A, B, and D are less efficient due to data loading overhead, incompatibility with SQL (Datastore), or higher operational complexity and cost (Hadoop cluster).

**Supporting Links:**

**BigQuery External Tables:** https://cloud.google.com/bigquery/docs/external-tables
**Querying AVRO files in BigQuery:** https://cloud.google.com/bigquery/docs/querying-avro

**Question: 86** CertyIQ

You need to verify that a Google Cloud Platform service account was created at a particular time. What should you do?

A. Filter the Activity log to view the Configuration category. Filter the Resource type to Service Account.

B. Filter the Activity log to view the Configuration category. Filter the Resource type to Google Project.

C. Filter the Activity log to view the Data Access category. Filter the Resource type to Service Account.

D. Filter the Activity log to view the Data Access category. Filter the Resource type to Google Project.

**Answer: A**

**Explanation:**

To verify the creation time of a service account, you should filter the Activity log (now known as Cloud

Logging Admin Activity logs) and select the **Configuration** category because service account creation is an administrative configuration change within your Google Cloud project. Filtering by **Resource type** as **Service Account** specifically targets audit logs related to service account operations, allowing you to pinpoint the creation event and its timestamp. Data Access logs (options C & D) are intended for operations that access user-provided data, not configuration changes like service account creation. Filtering by Google Project (option B & D) might show project-level changes but is less specific and efficient than filtering directly for Service Account resources when looking for service account creation.

**Supporting Links:**

**Cloud Logging Admin Activity logs:** https://cloud.google.com/logging/docs/audit#admin-activity-logs
**Cloud Logging Overview:** https://cloud.google.com/logging/docs/overview
**Filtering logs in Cloud Logging:** https://cloud.google.com/logging/docs/view/query-logs

**Question: 87** CertyIQ

You deployed an LDAP server on Compute Engine that is reachable via TLS through port 636 using UDP. You want to make sure it is reachable by clients over that port. What should you do?

A. Add the network tag allow-udp-636 to the VM instance running the LDAP server.

B. Create a route called allow-udp-636 and set the next hop to be the VM instance running the LDAP server.

C. Add a network tag of your choice to the instance. Create a firewall rule to allow ingress on UDP port 636 for that network tag.

D. Add a network tag of your choice to the instance running the LDAP server. Create a firewall rule to allow egress on UDP port 636 for that network tag.

**Answer: C**

**Explanation:**

The correct answer is C because to allow external clients to reach the LDAP server on UDP port 636, you need to configure Google Cloud Firewall to permit ingress traffic. Network tags are used to apply firewall rules to specific Compute Engine instances. By adding a network tag to the LDAP server instance and creating an ingress firewall rule that allows UDP traffic on port 636 for instances with that tag, you effectively open the necessary port for external access. Option A is incomplete because network tags alone do not open firewall ports; a firewall rule is required. Option B is incorrect as routes manage network traffic direction, not firewall rules for allowing ingress. Option D is wrong because egress firewall rules control outbound traffic, while making the LDAP server reachable requires allowing inbound (ingress) traffic.

**Supporting Links:**

**Firewall Rules Overview:** https://cloud.google.com/firewall/docs/firewall-rules
**Using Network Tags:** https://cloud.google.com/firewall/docs/firewalls-tags
**Ingress vs Egress Firewall Rules:** https://cloud.google.com/vpc/docs/firewalls#direction_of_connection

**Question: 88** CertyIQ

You need to set a budget alert for use of Compute Engineer services on one of the three Google Cloud Platform projects that you manage. All three projects are linked to a single billing account. What should you do?

A. Verify that you are the project billing administrator. Select the associated billing account and create a budget and alert for the appropriate project.

B. Verify that you are the project billing administrator. Select the associated billing account and create a budget and a custom alert.

C. Verify that you are the project administrator. Select the associated billing account and create a budget for the appropriate project.

D. Verify that you are project administrator. Select the associated billing account and create a budget and a custom alert.

**Answer: A**

**Explanation:**

To set a budget alert, you must first have the necessary billing administrator permissions for the billing account. After verifying permissions, navigate to the billing section in the Google Cloud Console and select the associated billing account. Within the billing account settings, create a new budget and configure it to apply to the specific project you are managing and for Compute Engine services. Ensure you configure budget alerts as part of the budget creation process to receive notifications when spending thresholds are reached. Option A accurately describes these steps by emphasizing the need for billing administrator permissions and creating a budget and alert for the appropriate project within the billing account.

**Supporting Links:**

**Creating Budgets and Setting Budget Alerts:** https://cloud.google.com/billing/docs/how-to/budgets
**Billing Roles and Permissions:** https://cloud.google.com/billing/docs/how-to/billing-access

**Question: 89** CertyIQ

You are migrating a production-critical on-premises application that requires 96 vCPUs to perform its task. You want to make sure the application runs in a similar environment on GCP. What should you do?

A. When creating the VM, use machine type n1-standard-96.

B. When creating the VM, use Intel Skylake as the CPU platform.

C. Create the VM using Compute Engine default settings. Use gcloud to modify the running instance to have 96 vCPUs.

D. Start the VM using Compute Engine default settings, and adjust as you go based on Rightsizing Recommendations.

**Answer: A**

**Explanation:**

The correct answer is A because the question explicitly states the application requires 96 vCPUs, and the n1-standard-96 machine type in Compute Engine provides exactly 96 vCPUs. Choosing a predefined machine type like n1-standard-96 ensures that the VM is provisioned with the necessary compute resources from the start, aligning with production-critical requirements. Option B is insufficient as specifying CPU platform alone doesn't guarantee 96 vCPUs. Options C and D are not suitable for production-critical migrations as they suggest starting with default settings which are unlikely to meet the 96 vCPU requirement and involve potential adjustments later, introducing unnecessary risks. For more information on Compute Engine machine types, refer to the official Google Cloud documentation: https://cloud.google.com/compute/docs/general-purpose-machines.

**Question: 90** CertyIQ

You want to configure a solution for archiving data in a Cloud Storage bucket. The solution must be cost-effective. Data with multiple versions should be archived after 30 days. Previous versions are accessed once a month for reporting. This archive data is also occasionally updated at month-end. What should you do?

A. Add a bucket lifecycle rule that archives data with newer versions after 30 days to Coldline Storage.

B. Add a bucket lifecycle rule that archives data with newer versions after 30 days to Nearline Storage.

C. Add a bucket lifecycle rule that archives data from regional storage after 30 days to Coldline Storage.

D. Add a bucket lifecycle rule that archives data from regional storage after 30 days to Nearline Storage.

**Answer: B**

**Explanation:**

The most cost-effective solution for archiving data accessed monthly and updated occasionally after 30 days is Nearline Storage. Nearline Storage is designed for data accessed less than once a month, balancing cost and availability, making it suitable for monthly reporting access. Implementing a lifecycle rule to transition data to Nearline after 30 days automates the archiving process and optimizes costs. Coldline Storage, while cheaper, is intended for even less frequent access and has higher retrieval costs, making Nearline a better fit for monthly access. Regional Storage is designed for frequent access and would be more expensive for archival purposes. You can learn more about Cloud Storage classes https://cloud.google.com/storage/docs/storage-classes and lifecycle management https://cloud.google.com/storage/docs/lifecycle.

**Question: 91** CertyIQ

Your company's infrastructure is on-premises, but all machines are running at maximum capacity. You want to burst to Google Cloud. The workloads on Google
Cloud must be able to directly communicate to the workloads on-premises using a private IP range. What should you do?

A. In Google Cloud, configure the VPC as a host for Shared VPC.

B. In Google Cloud, configure the VPC for VPC Network Peering.

C. Create bastion hosts both in your on-premises environment and on Google Cloud. Configure both as proxy servers using their public IP addresses.

D. Set up Cloud VPN between the infrastructure on-premises and Google Cloud.

**Answer: D**

**Explanation:**

Cloud VPN establishes a secure IPsec tunnel between your on-premises infrastructure and Google Cloud VPC, enabling private IP communication necessary for workloads to communicate directly across environments. This method fulfills the requirement for bursting workloads to Google Cloud while maintaining private connectivity with on-premises resources. Shared VPC and VPC Peering are relevant for network organization within Google Cloud, not for connecting to on-premises. Bastion hosts using public IPs do not provide private IP communication for workloads and are more suited for secure administrative access. Therefore, Cloud VPN is the most suitable solution for secure, private hybrid connectivity in this scenario. For further research, refer to the Cloud VPN Documentation.

**Question: 92** CertyIQ

You want to select and configure a solution for storing and archiving data on Google Cloud Platform. You need to support compliance objectives for data from one geographic location. This data is archived after 30 days and needs to be accessed annually. What should you do?

A. Select Multi-Regional Storage. Add a bucket lifecycle rule that archives data after 30 days to Coldline

Storage.

B. Select Multi-Regional Storage. Add a bucket lifecycle rule that archives data after 30 days to Nearline Storage.

C. Select Regional Storage. Add a bucket lifecycle rule that archives data after 30 days to Nearline Storage.

D. Select Regional Storage. Add a bucket lifecycle rule that archives data after 30 days to Coldline Storage.

**Answer: D**

**Explanation:**

The correct answer is **D**. Here's the justification:

**Option D is the most suitable choice because it leverages Regional Storage to meet geographic compliance requirements for data originating from one location, ensuring data locality.** Regional Storage offers cost-effectiveness compared to Multi-Regional when data locality within a single region is sufficient. **Coldline Storage is the ideal storage class for archiving data accessed infrequently, such as annually, due to its low storage cost.** Implementing a bucket lifecycle rule to transition data to Coldline after 30 days automates the archiving process according to the stated requirement. **This combination of Regional Storage and Coldline storage optimized for infrequent access and cost-effectiveness while adhering to compliance needs.** Nearline storage (options B and C) is designed for more frequent access than annually, making it less cost-optimal than Coldline for this scenario, and Multi-Regional storage (options A and B) is unnecessary for single geographic location compliance, adding unnecessary cost and complexity.

**Supporting Links:**

**Storage Classes:** https://cloud.google.com/storage/docs/storage-classes
**Lifecycle Management:** https://cloud.google.com/storage/docs/lifecycle
**Choosing a Storage Option:** https://cloud.google.com/storage/docs/choosing-storage

**Question: 93** CertyIQ

Your company uses BigQuery for data warehousing. Over time, many different business units in your company have created 1000+ datasets across hundreds of projects. Your CIO wants you to examine all datasets to find tables that contain an employee_ssn column. You want to minimize effort in performing this task.
What should you do?

A. Go to Data Catalog and search for employee_ssn in the search box.

B. Write a shell script that uses the bq command line tool to loop through all the projects in your organization.

C. Write a script that loops through all the projects in your organization and runs a query on INFORMATION_SCHEMA.COLUMNS view to find the employee_ssn column.

D. Write a Cloud Dataflow job that loops through all the projects in your organization and runs a query on INFORMATION_SCHEMA.COLUMNS view to find employee_ssn column.

**Answer: A**

**Explanation:**

Rationale: Data Catalog is a fully managed and scalable metadata management service that allows users to discover, understand, and manage their data assets. It automatically catalogs metadata from BigQuery, including table schemas and column names. Searching Data Catalog for "employee_ssn" would directly identify tables containing this column across all projects accessible to the user, requiring minimal effort compared to scripting and manual iteration through projects and datasets. Options B, C, and D involve scripting and querying INFORMATION_SCHEMA, which are more complex and time-consuming than using Data Catalog's built-in search functionality for metadata discovery.

Link for research:

Google Cloud Data Catalog Documentation - Searching Catalog
Google Cloud Data Catalog Documentation - BigQuery metadata

**Question: 94** CertyIQ

You create a Deployment with 2 replicas in a Google Kubernetes Engine cluster that has a single preemptible node pool. After a few minutes, you use kubectl to examine the status of your Pod and observe that one of them is still in Pending status:

```
$ kubectl get pods -l app=myapp
NAME                                READY     STATUS    RESTART   AGE
myapp-deployment-58ddbbb995-lp86m   0/1       Pending   0         9m
myapp-deployment-58ddbbb995-qjpkg   1/1       Running   0         9m
```

What is the most likely cause?

A. The pending Pod's resource requests are too large to fit on a single node of the cluster.

B. Too many Pods are already running in the cluster, and there are not enough resources left to schedule the pending Pod.

C. The node pool is configured with a service account that does not have permission to pull the container image used by the pending Pod.

D. The pending Pod was originally scheduled on a node that has been preempted between the creation of the Deployment and your verification of the Pods' status. It is currently being rescheduled on a new node.

**Answer: B**

**Explanation:**

Correct Answer is (B):

Reasons for a Pod Status Pending:

Troubleshooting Reason #1: Not enough CPU

Troubleshooting Reason #2: Not enough memory

Troubleshooting Reason #3: Not enough CPU and memory

https://managedkube.com/kubernetes/k8sbot/troubleshooting/pending/pod/2019/02/22/pending-pod.html

**Question: 95** CertyIQ

You want to find out when users were added to Cloud Spanner Identity Access Management (IAM) roles on your Google Cloud Platform (GCP) project. What should you do in the GCP Console?

A. Open the Cloud Spanner console to review configurations.

B. Open the IAM & admin console to review IAM policies for Cloud Spanner roles.

C. Go to the Stackdriver Monitoring console and review information for Cloud Spanner.

D. Go to the Stackdriver Logging console, review admin activity logs, and filter them for Cloud Spanner IAM roles.

**Answer: D**

**Explanation:**

To find out when users were added to Cloud Spanner IAM roles, you should navigate to the Stackdriver Logging (now Cloud Logging) console because it records admin activity logs, which include changes to IAM policies. Admin Activity logs capture administrative operations such as granting or revoking IAM roles. By filtering these logs for Cloud Spanner and IAM-related activities, you can pinpoint the specific events of users being added to Cloud Spanner IAM roles and their timestamps. The Cloud Spanner console, IAM & admin console, and Cloud Monitoring console do not provide historical audit logs of IAM role changes in the same way that Cloud Logging does.

**Supporting Links:**

**Cloud Logging Documentation:** https://cloud.google.com/logging/docs/audit - This documentation explains Google Cloud Audit Logs, including Admin Activity logs, which record administrative operations like IAM policy changes.
**Viewing Audit Logs:** https://cloud.google.com/logging/docs/view/query-logs - This document explains how to view and query logs in Cloud Logging, which is necessary to filter for specific events like IAM role changes.
**Cloud Spanner Audit Logging:** https://cloud.google.com/spanner/docs/audit-logging - While not explicitly about IAM role changes, this document confirms that Cloud Spanner operations are logged in Cloud Logging, reinforcing the idea that IAM changes related to Spanner would also be logged there.

**Question: 96** CertyIQ

Your company implemented BigQuery as an enterprise data warehouse. Users from multiple business units run queries on this data warehouse. However, you notice that query costs for BigQuery are very high, and you need to control costs. Which two methods should you use? (Choose two.)

A. Split the users from business units to multiple projects.

B. Apply a user- or project-level custom query quota for BigQuery data warehouse.

C. Create separate copies of your BigQuery data warehouse for each business unit.

D. Split your BigQuery data warehouse into multiple data warehouses for each business unit.

E. Change your BigQuery query model from on-demand to flat rate. Apply the appropriate number of slots to each Project.

**Answer: BE**

**Explanation:**

The chosen methods to control high BigQuery costs are B and E. Option B is valid because applying user or project-level custom query quotas directly restricts resource consumption and prevents excessive spending. Option E is also correct as transitioning to a flat-rate pricing model with allocated slots can be more cost-effective for consistent, high query volumes compared to on-demand pricing. Options A, C, and D are less effective or counterproductive for cost control; A only helps with cost segregation, while C and D increase costs through data duplication and infrastructure splitting.

**Supported Links for Research:**

**BigQuery Quotas and Limits:** https://cloud.google.com/bigquery/quotas - This documentation explains how to set and manage quotas in BigQuery.
**BigQuery Pricing:** https://cloud.google.com/bigquery/pricing - This page details the on-demand and flat-rate pricing models for BigQuery and helps understand when flat-rate is more cost-effective.
**Controlling costs in BigQuery:** https://cloud.google.com/bigquery/docs/best-practices-costs - This document provides comprehensive best practices for managing and controlling BigQuery costs.

**Question: 97** CertyIQ

You are building a product on top of Google Kubernetes Engine (GKE). You have a single GKE cluster. For each of your customers, a Pod is running in that cluster, and your customers can run arbitrary code inside their Pod. You want to maximize the isolation between your customers' Pods. What should you do?

A. Use Binary Authorization and whitelist only the container images used by your customers' Pods.

B. Use the Container Analysis API to detect vulnerabilities in the containers used by your customers' Pods.

C. Create a GKE node pool with a sandbox type configured to gvisor. Add the parameter runtimeClassName: gvisor to the specification of your customers' Pods.

D. Use the cos_containerd image for your GKE nodes. Add a nodeSelector with the value cloud.google.com/gke-os-distribution: cos_containerd to the specification of your customers' Pods.

**Answer: C**

**Explanation:**

Option C is the most suitable choice because using gvisor, a sandbox container runtime, significantly enhances isolation between customer Pods by providing a separate kernel for each Pod, minimizing the shared kernel attack surface. Configuring a GKE node pool with a sandbox type set to gvisor and specifying runtimeClassName: gvisor in the Pod specification ensures that these Pods run with gvisor, effectively isolating them from each other and the node's kernel. This approach directly addresses the requirement to maximize isolation for customer Pods running arbitrary code within a shared GKE cluster. Options A and B focus on image security and vulnerability scanning, not runtime isolation, while Option D, using cos_containerd, improves node security but doesn't provide the same level of pod isolation as gvisor. For further details on GKE sandbox node pools and gvisor, you can refer to the official Google Cloud documentation. 1, 2

**Question: 98** CertyIQ

Your customer has implemented a solution that uses Cloud Spanner and notices some read latency-related performance issues on one table. This table is accessed only by their users using a primary key. The table schema is shown below.

```
CREATE TABLE Persons (
     person_id INT64 NOT NULL,     // sequential number based on number of registration
     account_creation_date DATE,   // system date
     birthdate DATE,               // customer birthdate
     firstname STRING (255),           // first name
     lastname STRING (255),            // last name
     profile_picture BYTES (255)       // profile picture
) PRIMARY KEY (person_id)
```

You want to resolve the issue. What should you do?

A. Remove the profile_picture field from the table.

B. Add a secondary index on the person_id column.

C. Change the primary key to not have monotonically increasing values.

D. Create a secondary index using the following Data Definition Language (DDL):

```
CREATE INDEX person_id_ix
ON Persons (
     person_id,
     firstname,
     lastname
) STORING (
     profile_picture
)
```

**Answer: C**

**Explanation:**

C: this is to avoid having hotspots. If the PK is monotonic, then there is a higher chance of requests being routed to the same spanner server and thus overloading it.

D: Primary key is an index already. The question states that the users only accesses this table by PK.

**Question: 99** CertyIQ

Your finance team wants to view the billing report for your projects. You want to make sure that the finance team does not get additional permissions to the project. What should you do?

A. Add the group for the finance team to roles/billing user role.
B. Add the group for the finance team to roles/billing admin role.
C. Add the group for the finance team to roles/billing viewer role.
D. Add the group for the finance team to roles/billing project/Manager role.

**Answer: C**

**Explanation:**

**Justification:** To allow the finance team to view billing reports without granting extra project permissions, the roles/billing.viewer role is the most appropriate choice as it specifically provides read-only access to billing information, enabling them to view reports without the ability to modify billing settings or project resources. This aligns with the principle of least privilege, granting only the necessary permissions for the intended task.

**Supporting Links:**

1. **Google Cloud Documentation on Billing Access Control:** https://cloud.google.com/billing/docs/how-to/billing-access - This document explains how to control access to your Cloud Billing account and highlights different billing roles.
2. **Google Cloud IAM Roles Documentation (Billing Roles):** https://cloud.google.com/iam/docs/understanding-roles#billing-roles - This page provides a comprehensive list of predefined IAM roles for Cloud Billing, including the roles/billing.viewer role and its associated permissions.

**Question: 100** CertyIQ

Your organization has strict requirements to control access to Google Cloud projects. You need to enable your Site Reliability Engineers (SREs) to approve requests from the Google Cloud support team when an SRE opens a support case. You want to follow Google-recommended practices. What should you do?

A. Add your SREs to roles/iam.roleAdmin role.
B. Add your SREs to roles/accessapproval.approver role.
C. Add your SREs to a group and then add this group to roles/iam.roleAdmin.role.
D. Add your SREs to a group and then add this group to roles/accessapproval.approver role.

**Answer: D**

**Explanation:**

D is the correct answer. Granting the roles/accessapproval.approver role to a group containing SREs aligns with Google's recommended practice of using Access Approval to manage Google Support access requests, ensuring strict control as required. This role specifically allows SREs to approve access requests from Google Cloud support, meeting the question's requirement directly. Utilizing a group is a best practice for managing permissions for multiple users, simplifying administration and enhancing security. Options A and C are incorrect because roles/iam.roleAdmin provides overly broad permissions beyond just approving support access, violating the principle of least privilege. Option B is partially correct by using the right role, but option D is better by incorporating the best practice of group management.

**Supporting Links:**

**Access Approval Overview:** https://cloud.google.com/access-approval/docs/overview
**Access Approval Roles:** https://cloud.google.com/access-approval/docs/access-approval-iam#accessapproval.approver

**Question: 101** CertyIQ

You need to host an application on a Compute Engine instance in a project shared with other teams. You want to prevent the other teams from accidentally causing downtime on that application. Which feature should you use?

A. Use a Shielded VM.
B. Use a Preemptible VM.
C. Use a sole-tenant node.
D. Enable deletion protection on the instance.

**Answer: D**

**Explanation:**

Justification: Enabling deletion protection on the Compute Engine instance prevents accidental termination by other teams in the shared project, directly addressing the requirement to avoid unintended downtime, as documented in Google Cloud's feature for preventing accidental VM deletion [1].

[1] Preventing accidental VM deletion: https://cloud.google.com/compute/docs/instances/preventing-accidental-vm-deletion

**Question: 102** CertyIQ

Your organization needs to grant users access to query datasets in BigQuery but prevent them from accidentally deleting the datasets. You want a solution that follows Google-recommended practices. What should you do?

A. Add users to roles/bigquery user role only, instead of roles/bigquery dataOwner.

B. Add users to roles/bigquery dataEditor role only, instead of roles/bigquery dataOwner.

C. Create a custom role by removing delete permissions, and add users to that role only.

D. Create a custom role by removing delete permissions. Add users to the group, and then add the group to the custom role.

**Answer: D**

**Explanation:**

**Justification:**

Option D is the most suitable solution because it adheres to the principle of least privilege and Google-recommended practices for IAM. Creating a custom role allows for granular control by removing delete permissions, specifically preventing accidental dataset deletion while granting necessary query access. Assigning users to a group and then the group to the custom role is a best practice for scalable user management in Google Cloud IAM, simplifying permission administration and enhancing security. Predefined roles like roles/bigquery.user (Option A) or roles/bigquery.dataEditor (Option B) might not offer the precise control needed to prevent dataset deletion while ensuring query access, and directly assigning roles to individual users (Option C) is less manageable than using groups for larger organizations.

**Supporting Links:**

**Best practices for using IAM:** https://cloud.google.com/iam/docs/best-practices-for-using-iam (Specifically mentions using groups to manage users)
**Create and manage custom roles:** https://cloud.google.com/iam/docs/creating-custom-roles (Explains the benefits and process of custom roles for granular access)
**Predefined BigQuery roles:** https://cloud.google.com/bigquery/docs/access-control-iam (Details the permissions included in roles/bigquery.user and roles/bigquery.dataEditor, highlighting that neither is perfectly tailored to prevent dataset deletion while allowing queries as specifically required)

**Question: 103** CertyIQ

You have a developer laptop with the Cloud SDK installed on Ubuntu. The Cloud SDK was installed from the Google Cloud Ubuntu package repository. You want to test your application locally on your laptop with Cloud Datastore. What should you do?

A. Export Cloud Datastore data using gcloud datastore export.

B. Create a Cloud Datastore index using gcloud datastore indexes create.

C. Install the google-cloud-sdk-datastore-emulator component using the apt get install command.

D. Install the cloud-datastore-emulator component using the gcloud components install command.

**Answer: C**

**Explanation:**

To test your application locally with Cloud Datastore on Ubuntu where Cloud SDK was installed from the Google Cloud Ubuntu package repository, you should install the Cloud Datastore emulator using the apt-get install command, specifically for the google-cloud-sdk-datastore-emulator component. This approach is necessary because the Cloud SDK was initially installed via Ubuntu's package manager (apt-get), and components should be managed using the same package manager to ensure proper system integration and

dependency management. Using apt-get to install google-cloud-sdk-datastore-emulator ensures that the emulator is correctly installed and integrated with your existing Cloud SDK setup. This method is consistent with how packages are managed within the Ubuntu environment when using the Google Cloud package repository.

**Supporting Links:**

While a direct link explicitly stating to use apt-get install google-cloud-sdk-datastore-emulator might be slightly harder to find as documentation might generally favor gcloud components install, understanding the context of package manager installation is key.

1. **Installing Cloud SDK via apt on Debian/Ubuntu:** https://cloud.google.com/sdk/docs/install/linux-apt - This documentation confirms that apt-get is used to install the Cloud SDK on Debian/Ubuntu. While it doesn't explicitly list all component package names, it sets the context of using apt-get.
2. **Cloud SDK Components:** https://cloud.google.com/sdk/docs/components - This page lists components, including cloud-datastore-emulator. While it primarily talks about gcloud components install, the principle of components being installed separately still applies.
3. **General understanding of package management in Linux (Ubuntu/Debian):** Basic understanding of how apt-get manages packages in Ubuntu/Debian reinforces the idea that if the base SDK is installed via apt-get, components are also best managed via apt-get for consistency and dependency resolution.

By combining these pieces of information, we can deduce that option C is the correct approach for the given scenario.

**Question: 104** CertyIQ

Your company set up a complex organizational structure on Google Cloud. The structure includes hundreds of folders and projects. Only a few team members should be able to view the hierarchical structure. You need to assign minimum permissions to these team members, and you want to follow Google-recommended practices. What should you do?

A. Add the users to roles/browser role.
B. Add the users to roles/iam.roleViewer role.
C. Add the users to a group, and add this group to roles/browser.
D. Add the users to a group, and add this group to roles/iam.roleViewer role.

**Answer: C**

**Explanation:**

Option C is the correct answer because roles/browser grants the minimum necessary permissions to view the organizational hierarchy, as it includes permissions to read folders and projects [1], and using a group to assign roles is a Google-recommended IAM best practice for managing permissions efficiently and centrally [2].

[1] https://cloud.google.com/iam/docs/roles/browser[2] https://cloud.google.com/iam/docs/best-practices-for-using-iam#use-groups-for-managing-users

**Question: 105** CertyIQ

Your company has a single sign-on (SSO) identity provider that supports Security Assertion Markup Language (SAML) integration with service providers. Your company has users in Cloud Identity. You would like users to authenticate using your company's SSO provider. What should you do?

A. In Cloud Identity, set up SSO with Google as an identity provider to access custom SAML apps.

B. In Cloud Identity, set up SSO with a third-party identity provider with Google as a service provider.

C. Obtain OAuth 2.0 credentials, configure the user consent screen, and set up OAuth 2.0 for Mobile & Desktop Apps.

D. Obtain OAuth 2.0 credentials, configure the user consent screen, and set up OAuth 2.0 for Web Server Applications.

**Answer: B**

**Explanation:**

The correct answer is **B**. The scenario describes a situation where an organization wants to leverage its existing, third-party SAML-based SSO provider to authenticate users who also exist in Cloud Identity. Cloud Identity acts as a directory service in this context. For seamless integration, we need to configure Cloud Identity to trust the external SSO provider as the source of authentication. Option **B**, which states "In Cloud Identity, set up SSO with a third-party identity provider with Google as a service provider," accurately reflects this approach. Cloud Identity becomes the service provider (relying on the third-party SSO provider) in this configuration. The SAML assertions sent by the external identity provider are used by Cloud Identity to grant access. Option **A** is incorrect because it would make Google the identity provider, contradicting the prompt where the company has an existing third-party SSO provider. Options **C** and **D** are incorrect because they are related to OAuth 2.0, which is not used for authenticating users via a SAML-based SSO, but for granting delegated access to APIs and resources.

Relevant concepts:

**SAML (Security Assertion Markup Language):** An XML-based open standard for exchanging authentication and authorization data between an identity provider and a service provider.
**SSO (Single Sign-On):** Allows users to access multiple applications with one set of login credentials.
**Identity Provider (IdP):** A system that creates, maintains, and manages identity information for users while authenticating them.
**Service Provider (SP):** A service that relies on an identity provider to authenticate users.
**Cloud Identity:** Google's identity and access management service.

Authoritative Links:

Set up SSO with a third-party identity provider
SAML Overview

**Question: 106** CertyIQ

Your organization has a dedicated person who creates and manages all service accounts for Google Cloud projects. You need to assign this person the minimum role for projects. What should you do?

A. Add the user to roles/iam.roleAdmin role.

B. Add the user to roles/iam.securityAdmin role.

C. Add the user to roles/iam.serviceAccountUser role.

D. Add the user to roles/iam.serviceAccountAdmin role.

**Answer: D**

**Explanation:**

The correct answer is **D** because the roles/iam.serviceAccountAdmin role grants the necessary permissions to create, delete, and manage service accounts within a Google Cloud project, aligning with the requirement of managing service accounts. This role is specifically designed for service account management, unlike roles/iam.roleAdmin and roles/iam.securityAdmin which provide broader, less specific permissions not needed for this task. roles/iam.serviceAccountUser only allows users to use service accounts, not manage them. Therefore, roles/iam.serviceAccountAdmin is the minimum role that fulfills the requirement of creating and managing service accounts. For further research, you can refer to the Google Cloud documentation on IAM roles and specifically the Service Account Admin role.

**Question: 107** CertyIQ

You are building an archival solution for your data warehouse and have selected Cloud Storage to archive your data. Your users need to be able to access this archived data once a quarter for some regulatory requirements. You want to select a cost-efficient option. Which storage option should you use?

A. Cold Storage
B. Nearline Storage
C. Regional Storage
D. Multi-Regional Storage

**Answer: A**

**Explanation:**

Cold Storage is the most cost-effective option for archiving data accessed infrequently, such as quarterly, due to its lowest storage costs among Cloud Storage classes, despite higher retrieval costs and latency compared to Nearline, Regional, or Multi-Regional Storage, which are designed for more frequent access and higher performance needs. Cold Storage is specifically designed for archival and backup scenarios where data access is infrequent and cost optimization is a priority. Nearline is suitable for data accessed less than once a month, making Cold Storage a better fit for quarterly access. Regional and Multi-Regional Storage are designed for frequently accessed data with higher availability and performance requirements, making them significantly more expensive for archival purposes. Therefore, for quarterly access and cost-efficient archival, Cold Storage is the optimal choice.

https://cloud.google.com/storage/docs/storage-classeshttps://cloud.google.com/storage/pricing

**Question: 108** CertyIQ

A team of data scientists infrequently needs to use a Google Kubernetes Engine (GKE) cluster that you manage. They require GPUs for some long-running, non- restartable jobs. You want to minimize cost. What should you do?

A. Enable node auto-provisioning on the GKE cluster.
B. Create a VerticalPodAutscaler for those workloads.
C. Create a node pool with preemptible VMs and GPUs attached to those VMs.
D. Create a node pool of instances with GPUs, and enable autoscaling on this node pool with a minimum size of 1.

**Answer: D**

**Explanation:**

Option D is the most suitable choice because it balances cost-effectiveness with the infrequent and long-running nature of the data scientists' jobs requiring GPUs. Creating a dedicated node pool with GPUs ensures the necessary hardware is available when needed. Enabling autoscaling on this node pool, with a minimum size of 1, addresses the "infrequent" use case; the cluster will scale down to only one node (reducing cost) when not in use, and scale up when jobs are submitted. This setup avoids the constant cost associated with keeping a larger node pool continuously running. While preemptible VMs (Option C) can be cheaper, they are unsuitable for "long-running, non-restartable jobs" as they can be terminated with short notice, leading to job loss. Node auto-provisioning (Option A) introduces complexity and might provision more resources than needed, increasing costs. Vertical Pod Autoscaling (Option B) is not appropriate in this context, as it adjusts CPU/memory within a Pod, not the underlying GPU resources needed for a node.

**Key Concepts:**

**Node Pools:** Provide a way to group nodes with similar configurations within a GKE cluster.
**Autoscaling:** Automatically adjusts the number of nodes in a node pool based on resource utilization.
**Preemptible VMs:** Cost-effective but can be terminated with little warning.
**GPUs:** Specialized hardware accelerators for computationally intensive workloads.

**Why other options are not ideal:**

**A (Node Auto-provisioning):** Might over-provision resources and lead to unpredictable costs.
**B (VerticalPodAutoscaler):** Does not address the need for GPU resources at the node level.
**C (Preemptible VMs):** Unsuitable for non-restartable jobs due to potential interruptions.

**Authoritative Links:**

**GKE Node Pools:** https://cloud.google.com/kubernetes-engine/docs/concepts/node-pools
**GKE Autoscaling:** https://cloud.google.com/kubernetes-engine/docs/how-to/node-autoscaling
**Preemptible VMs:** https://cloud.google.com/compute/docs/instances/preemptible
**GPUs on GKE:** https://cloud.google.com/kubernetes-engine/docs/how-to/gpus
**Vertical Pod Autoscaling:** https://cloud.google.com/kubernetes-engine/docs/how-to/vertical-pod-autoscaling

**Question: 109** CertyIQ

Your organization has user identities in Active Directory. Your organization wants to use Active Directory as their source of truth for identities. Your organization wants to have full control over the Google accounts used by employees for all Google services, including your Google Cloud Platform (GCP) organization. What should you do?

A. Use Google Cloud Directory Sync (GCDS) to synchronize users into Cloud Identity.
B. Use the cloud Identity APIs and write a script to synchronize users to Cloud Identity.
C. Export users from Active Directory as a CSV and import them to Cloud Identity via the Admin Console.
D. Ask each employee to create a Google account using self signup. Require that each employee use their company email address and password.

**Answer: A**

**Explanation:**

The correct answer is **A**.

**Justification:** Google Cloud Directory Sync (GCDS) is the recommended tool for synchronizing user identities from on-premises Active Directory to Google Cloud Identity. GCDS automates the synchronization process, ensuring Active Directory remains the source of truth for user identities. By using GCDS, the organization gains full control over the Google accounts associated with their employees within their Google Cloud Platform organization. This approach fulfills the requirement of managing Google accounts centrally and

leveraging existing Active Directory infrastructure for identity management. GCDS handles ongoing synchronization, reflecting changes made in Active Directory in Cloud Identity, maintaining consistency and reducing administrative overhead. Options B, C, and D are less efficient, less secure, or do not provide the desired level of centralized control and automated synchronization compared to GCDS.

**Supporting Links:**

**Google Cloud Directory Sync (GCDS) Documentation:** https://support.google.com/a/answer/106368?hl=en - This link provides comprehensive information about GCDS, its features, and how to use it for synchronizing Active Directory with Google Workspace and Cloud Identity.

**Cloud Identity Documentation:** https://cloud.google.com/identity - This link leads to the general documentation for Google Cloud Identity, explaining its purpose and how it integrates with Google Cloud services and external identity providers like Active Directory.

**Best practices for integrating with Active Directory:** https://cloud.google.com/architecture/identity/federating-gcp-with-active-directory - This document discusses various ways to integrate GCP with Active Directory, including using Cloud Identity and synchronization tools like GCDS to centralize identity management and control.

**Question: 110** CertyIQ

You have successfully created a development environment in a project for an application. This application uses Compute Engine and Cloud SQL. Now you need to create a production environment for this application. The security team has forbidden the existence of network routes between these 2 environments and has asked you to follow Google-recommended practices. What should you do?

A. Create a new project, enable the Compute Engine and Cloud SQL APIs in that project, and replicate the setup you have created in the development environment.

B. Create a new production subnet in the existing VPC and a new production Cloud SQL instance in your existing project, and deploy your application using those resources.

C. Create a new project, modify your existing VPC to be a Shared VPC, share that VPC with your new project, and replicate the setup you have in the development environment in that new project in the Shared VPC.

D. Ask the security team to grant you the Project Editor role in an existing production project used by another division of your company. Once they grant you that role, replicate the setup you have in the development environment in that project.

**Answer: A**

**Explanation:**

Creating a new Google Cloud project for the production environment ensures complete isolation from the development environment at the project level, aligning with the security team's requirement of no network routes and adhering to Google's recommended best practices for environment separation as projects act as fundamental organizational units and provide strong isolation boundaries for resources. This approach enhances security, simplifies management, and prevents unintended interactions between development and production environments. https://cloud.google.com/resource-manager/docs/resource-hierarchy, https://cloud.google.com/architecture/framework/organizational-structure/best-practices, https://cloud.google.com/architecture/framework/security/design-principles

**Question: 111** CertyIQ

Your management has asked an external auditor to review all the resources in a specific project. The security team

has enabled the Organization Policy called
Domain Restricted Sharing on the organization node by specifying only your Cloud Identity domain. You want the auditor to only be able to view, but not modify, the resources in that project. What should you do?

A. Ask the auditor for their Google account, and give them the Viewer role on the project.

B. Ask the auditor for their Google account, and give them the Security Reviewer role on the project.

C. Create a temporary account for the auditor in Cloud Identity, and give that account the Viewer role on the project.

D. Create a temporary account for the auditor in Cloud Identity, and give that account the Security Reviewer role on the project.

**Answer: C**

**Explanation:**

Option C is the most suitable choice because creating a temporary account within your Cloud Identity domain ensures compliance with the Domain Restricted Sharing policy, which restricts resource access to users within your domain. Assigning the Viewer role to this temporary account grants the auditor the necessary read-only permissions to examine project resources without allowing any modifications, effectively fulfilling the requirement of view-only access for the audit. This approach also aligns with security best practices by providing controlled and revocable access for external auditors. Granting access with external Google accounts (options A and B) would likely be blocked by the Domain Restricted Sharing policy, and while the Security Reviewer role (options B and D) offers read-only access, the Viewer role (options A and C) is sufficient for the auditor's task of reviewing resources and is the least privileged role that meets the requirements.

**Supporting Links:**

**Organization policies - Domain restricted sharing:** https://cloud.google.com/resource-manager/docs/organization-policy/restricting-domains
**Predefined roles - Viewer:** https://cloud.google.com/iam/docs/understanding-roles#viewer
**Identity and Access Management (IAM) Overview:** https://cloud.google.com/iam/docs/overview

**Question: 112** CertyIQ

You have a workload running on Compute Engine that is critical to your business. You want to ensure that the data on the boot disk of this workload is backed up regularly. You need to be able to restore a backup as quickly as possible in case of disaster. You also want older backups to be cleaned automatically to save on cost. You want to follow Google-recommended practices. What should you do?

A. Create a Cloud Function to create an instance template.

B. Create a snapshot schedule for the disk using the desired interval.

C. Create a cron job to create a new disk from the disk using gcloud.

D. Create a Cloud Task to create an image and export it to Cloud Storage.

**Answer: B**

**Explanation:**

The correct answer is **B**. Creating a snapshot schedule is the Google-recommended practice for regularly backing up Compute Engine persistent disks, as it provides automated, incremental backups at desired intervals. Snapshot schedules allow for quick restoration by creating new disks from snapshots, and they support retention policies to automatically delete older snapshots, optimizing cost. This approach aligns with best practices for disaster recovery and data protection on Google Cloud.

Supporting Links:

1. **Google Cloud Documentation on Scheduled Snapshots:** https://cloud.google.com/compute/docs/disks/scheduled-snapshots
2. **Google Cloud Documentation on Disk Snapshots:** https://cloud.google.com/compute/docs/disks/snapshots

**Question: 113** CertyIQ

You need to assign a Cloud Identity and Access Management (Cloud IAM) role to an external auditor. The auditor needs to have permissions to review your
Google Cloud Platform (GCP) Audit Logs and also to review your Data Access logs. What should you do?

A. Assign the auditor the IAM role roles/logging.privateLogViewer. Perform the export of logs to Cloud Storage.

B. Assign the auditor the IAM role roles/logging.privateLogViewer. Direct the auditor to also review the logs for changes to Cloud IAM policy.

C. Assign the auditor's IAM user to a custom role that has logging.privateLogEntries.list permission. Perform the export of logs to Cloud Storage.

D. Assign the auditor's IAM user to a custom role that has logging.privateLogEntries.list permission. Direct the auditor to also review the logs for changes to Cloud IAM policy.

**Answer: B**

**Explanation:**

The correct answer is **B**. The roles/logging.privateLogViewer IAM role grants the auditor permission to view both Admin Activity logs (often referred to as GCP Audit Logs) and Data Access logs, fulfilling the requirement to review both log types. Directing the auditor to review logs for changes to Cloud IAM policy is crucial because Data Access logs often contain information about IAM-related actions, which are essential for a comprehensive security audit. Options A and C suggest exporting logs to Cloud Storage, which is not necessary for simply reviewing logs and adds complexity. Options C and D propose custom roles with logging.privateLogEntries.list, while roles/logging.privateLogViewer is a pre-defined role that directly addresses the need to view private logs, making it a simpler and more appropriate solution.

**Supporting Links:**

**Cloud Logging Access Control:** https://cloud.google.com/logging/docs/access-control - This documentation explains the different Cloud Logging roles and their associated permissions, including roles/logging.privateLogViewer and its ability to view private logs such as Data Access logs.
**Cloud Audit Logs Overview:** https://cloud.google.com/audit-logging - This page provides an overview of Cloud Audit Logs, including Admin Activity and Data Access logs, and their importance for security and compliance.

**Question: 114** CertyIQ

You are managing several Google Cloud Platform (GCP) projects and need access to all logs for the past 60 days. You want to be able to explore and quickly analyze the log contents. You want to follow Google-recommended practices to obtain the combined logs for all projects. What should you do?

A. Navigate to Stackdriver Logging and select resource.labels.project_id="*"

B. Create a Stackdriver Logging Export with a Sink destination to a BigQuery dataset. Configure the table expiration to 60 days.

C. Create a Stackdriver Logging Export with a Sink destination to Cloud Storage. Create a lifecycle rule to delete objects after 60 days.

D. Configure a Cloud Scheduler job to read from Stackdriver and store the logs in BigQuery. Configure the table expiration to 60 days.

**Answer: B**

**Explanation:**

The correct answer is **B**. Creating a Stackdriver Logging Export with a Sink destination to BigQuery is the recommended approach for centralizing logs from multiple GCP projects for analysis, as it allows you to combine and explore logs in BigQuery's powerful analytics environment. Exporting logs to BigQuery facilitates efficient querying and analysis of large log volumes, aligning with the requirement to quickly analyze log contents. BigQuery's table expiration feature directly addresses the 60-day retention requirement, automatically managing storage. Using Cloud Logging Exports is a best practice for routing logs to other Google Cloud services, and BigQuery is ideal for log analysis and exploration. This method allows for efficient and cost-effective log management and analysis compared to other options like manual querying or using Cloud Storage for immediate analysis.

**Supporting Links:**

**Cloud Logging Exports:** https://cloud.google.com/logging/docs/export
**BigQuery as a Logging Sink:** https://cloud.google.com/logging/docs/export/bigquery
**BigQuery Table Expiration:** https://cloud.google.com/bigquery/docs/table-expiration
**Centralized Logging:** https://cloud.google.com/logging/docs/central-logging

**Question: 115** CertyIQ

You need to reduce GCP service costs for a division of your company using the fewest possible steps. You need to turn off all configured services in an existing
GCP project. What should you do?

A. 1. Verify that you are assigned the Project Owners IAM role for this project. 2. Locate the project in the GCP console, click Shut down and then enter the project ID.

B. 1. Verify that you are assigned the Project Owners IAM role for this project. 2. Switch to the project in the GCP console, locate the resources and delete them.

C. 1. Verify that you are assigned the Organizational Administrator IAM role for this project. 2. Locate the project in the GCP console, enter the project ID and then click Shut down.

D. 1. Verify that you are assigned the Organizational Administrators IAM role for this project. 2. Switch to the project in the GCP console, locate the resources and delete them.

**Answer: A**

**Explanation:**

**Justification:** Option A is the correct answer because shutting down a GCP project is the quickest way to stop all running services and immediately reduce costs. Project Owners have sufficient permissions to shut down a project as documented in Understanding project roles. Deleting resources individually (Option B and D) is a more time-consuming and less comprehensive approach, failing to be the "fewest possible steps". Organizational Administrator role (Option C and D) is not necessary for shutting down a single project, as Project Owner is sufficient. The "Shut down" feature in the GCP console, as described in Option A, is the direct tool for this action, further detailed in Shutting down projects. Therefore, Option A accurately outlines the fewest steps to turn off all services by shutting down the project with the appropriate Project Owner role.

**Question: 116** CertyIQ

You are configuring service accounts for an application that spans multiple projects. Virtual machines (VMs) running in the web-applications project need access to BigQuery datasets in crm-databases-proj. You want to follow Google-recommended practices to give access to the service account in the web-applications project. What should you do?

A. Give project owner for web-applications appropriate roles to crm-databases-proj.
B. Give project owner role to crm-databases-proj and the web-applications project.
C. Give project owner role to crm-databases-proj and bigquery.dataViewer role to web-applications.
D. Give bigquery.dataViewer role to crm-databases-proj and appropriate roles to web-applications.

**Answer: D**

**Explanation:**

**Justification:** To adhere to Google-recommended practices and the principle of least privilege for cross-project service account access, you should grant the bigquery.dataViewer role on the crm-databases-proj project directly to the service account associated with the VMs in the web-applications project, allowing them to read BigQuery datasets without granting excessive permissions like Project Owner. This approach ensures that only the necessary BigQuery access is granted in the target project (crm-databases-proj), while "appropriate roles to web-applications" would refer to any roles needed within the web-applications project for the service account to function, which is less relevant to the core problem of cross-project BigQuery access. This method aligns with best practices for securing cloud resources by limiting permissions to only what is required. Granting access to project members and Service accounts documentation support this approach of granting specific roles to service accounts for cross-project access. BigQuery IAM roles details the bigquery.dataViewer role for read-only access to BigQuery data.

**Question: 117** CertyIQ

An employee was terminated, but their access to Google Cloud Platform (GCP) was not removed until 2 weeks later. You need to find out this employee accessed any sensitive customer information after their termination. What should you do?

A. View System Event Logs in Stackdriver. Search for the user's email as the principal.
B. View System Event Logs in Stackdriver. Search for the service account associated with the user.
C. View Data Access audit logs in Stackdriver. Search for the user's email as the principal.
D. View the Admin Activity log in Stackdriver. Search for the service account associated with the user.

**Answer: C**

**Explanation:**

The correct answer is **C**. Data Access audit logs in Google Cloud Platform (GCP) specifically track access to data resources and are the appropriate logs to investigate who accessed sensitive customer information. Searching these logs using the terminated employee's email address as the principal will reveal any actions performed by that user after their termination. System Event Logs (Option A and B) are more focused on system operations and infrastructure events, not detailed data access. Admin Activity logs (Option D) record administrative actions, not necessarily data access by regular users.

**Supporting Links:**

**Google Cloud Audit Logs Overview:** https://cloud.google.com/logging/docs/audit - This document provides a general overview of Cloud Audit Logs, explaining the different types of audit logs including Data Access audit

logs.
**Data Access Audit Logs:** https://cloud.google.com/logging/docs/audit/data-access - This page specifically describes Data Access audit logs and what type of operations they record, which is relevant to tracking access to sensitive data.
**Cloud Logging Query Language:** https://cloud.google.com/logging/docs/query/language - While not directly about audit logs, this resource is useful for understanding how to search and filter logs in Cloud Logging (formerly Stackdriver Logging), which is essential for finding the specific user's activity.

**Question: 118** CertyIQ

You need to create a custom IAM role for use with a GCP service. All permissions in the role must be suitable for production use. You also want to clearly share with your organization the status of the custom role. This will be the first version of the custom role. What should you do?

A. Use permissions in your role that use the 'supported' support level for role permissions. Set the role stage to ALPHA while testing the role permissions.

B. Use permissions in your role that use the 'supported' support level for role permissions. Set the role stage to BETA while testing the role permissions.

C. Use permissions in your role that use the 'testing' support level for role permissions. Set the role stage to ALPHA while testing the role permissions.

D. Use permissions in your role that use the 'testing' support level for role permissions. Set the role stage to BETA while testing the role permissions.

**Answer: A**

**Explanation:**

Answer A is the most suitable because using 'supported' permissions ensures the custom IAM role is production-ready, as these permissions are stable and recommended for production environments. Setting the role stage to 'ALPHA' is appropriate for the first version of a custom role as it clearly communicates to your organization that the role is still under development and testing, allowing for transparency about its status before wider production deployment. This approach balances the need for production-suitable permissions with the practical need for testing and communication during the initial phase of a custom role's lifecycle.

**Supporting Links:**

1. **Understanding custom roles:** https://cloud.google.com/iam/docs/understanding-custom-roles - This document explains the use of custom roles in IAM and best practices.
2. **Role stages:** https://cloud.google.com/iam/docs/understanding-roles#role-stages - This documentation details the different role stages (ALPHA, BETA, GA) and their implications, highlighting that ALPHA is for early development and testing.
3. **Permissions support levels:** https://cloud.google.com/iam/docs/permissions-reference - While not explicitly stating "supported" vs "testing" levels, this document implicitly highlights that permissions used in production roles should be stable and well-documented, which aligns with the concept of 'supported' permissions for production use. (Note: the term "testing" as a formal support level for permissions isn't directly documented in the same way as role stages or GA/Preview features but is implied by the existence of ALPHA/BETA stages for roles and the general concept of software development lifecycle).

**Question: 119** CertyIQ

Your company has a large quantity of unstructured data in different file formats. You want to perform ETL

transformations on the data. You need to make the data accessible on Google Cloud so it can be processed by a Dataflow job. What should you do?

A. Upload the data to BigQuery using the bq command line tool.

B. Upload the data to Cloud Storage using the gsutil command line tool.

C. Upload the data into Cloud SQL using the import function in the console.

D. Upload the data into Cloud Spanner using the import function in the console.

**Answer: B**

**Explanation:**

**Justification:** Cloud Storage is designed to store large amounts of unstructured data in various formats, offering scalability and cost-effectiveness, making it ideal for staging data before ETL processing with Dataflow, which can directly read and transform data from Cloud Storage.

**Supporting Links:**

**Google Cloud Storage:** https://cloud.google.com/storage
**Google Cloud Dataflow:** https://cloud.google.com/dataflow
**gsutil Tool:** https://cloud.google.com/storage/docs/gsutil

**Question: 120** CertyIQ

You need to manage multiple Google Cloud projects in the fewest steps possible. You want to configure the Google Cloud SDK command line interface (CLI) so that you can easily manage multiple projects. What should you do?

A. 1. Create a configuration for each project you need to manage. 2. Activate the appropriate configuration when you work with each of your assigned Google Cloud projects.

B. 1. Create a configuration for each project you need to manage. 2. Use gcloud init to update the configuration values when you need to work with a non-default project

C. 1. Use the default configuration for one project you need to manage. 2. Activate the appropriate configuration when you work with each of your assigned Google Cloud projects.

D. 1. Use the default configuration for one project you need to manage. 2. Use gcloud init to update the configuration values when you need to work with a non-default project.

**Answer: A**

**Explanation:**

The most efficient way to manage multiple Google Cloud projects with the gcloud CLI is to create a separate configuration for each project and activate the relevant configuration when working on a specific project, as this approach allows for quick switching between project environments without re-initialization. This method leverages the gcloud config configurations create command to set up configurations and the gcloud config configurations activate command for seamless switching, ensuring each project's settings are isolated and readily accessible. Using configurations eliminates the need to repeatedly update project settings using gcloud init, which is a more time-consuming process intended for initial setup or significant configuration changes rather than frequent project switching. This approach is recommended in Google Cloud documentation for managing multiple configurations, streamlining project management within the gcloud CLI.
https://cloud.google.com/sdk/docs/configurations,
https://cloud.google.com/sdk/gcloud/reference/config/configurations/activate

**Question: 121** CertyIQ

Your managed instance group raised an alert stating that new instance creation has failed to create new instances. You need to maintain the number of running instances specified by the template to be able to process expected application traffic. What should you do?

A. Create an instance template that contains valid syntax which will be used by the instance group. Delete any persistent disks with the same name as instance names.

B. Create an instance template that contains valid syntax that will be used by the instance group. Verify that the instance name and persistent disk name values are not the same in the template.

C. Verify that the instance template being used by the instance group contains valid syntax. Delete any persistent disks with the same name as instance names. Set the disks.autoDelete property to true in the instance template.

D. Delete the current instance template and replace it with a new instance template. Verify that the instance name and persistent disk name values are not the same in the template. Set the disks.autoDelete property to true in the instance template.

**Answer: A**

**Explanation:**

The correct answer is A because creating a new instance template with valid syntax directly addresses the potential issue of a corrupted or incorrectly configured template that is causing instance creation failures in the Managed Instance Group (MIG). Ensuring the template has valid syntax is crucial for successful instance launches. Deleting persistent disks with names that clash with instance names resolves potential naming conflicts that can also prevent new instances from being created within the group. This approach is more proactive in fixing the root cause compared to simply verifying the existing template or performing more drastic actions like deleting the current template without creating a new valid one. Addressing both template validity and naming conflicts ensures that the MIG can resume creating instances and maintain the desired application traffic capacity. You can research more about instance templates and troubleshooting MIGs using these links: Instance Templates and Troubleshooting Managed Instance Groups.

**Question: 122** CertyIQ

Your company is moving from an on-premises environment to Google Cloud. You have multiple development teams that use Cassandra environments as backend databases. They all need a development environment that is isolated from other Cassandra instances. You want to move to Google Cloud quickly and with minimal support effort. What should you do?

A. 1. Build an instruction guide to install Cassandra on Google Cloud. 2. Make the instruction guide accessible to your developers.

B. 1. Advise your developers to go to Cloud Marketplace. 2. Ask the developers to launch a Cassandra image for their development work.

C. 1. Build a Cassandra Compute Engine instance and take a snapshot of it. 2. Use the snapshot to create instances for your developers.

D. 1. Build a Cassandra Compute Engine instance and take a snapshot of it. 2. Upload the snapshot to Cloud Storage and make it accessible to your developers. 3. Build instructions to create a Compute Engine instance from the snapshot so that developers can do it themselves.

**Answer: B**

**Explanation:**

Option B is the most suitable solution because Cloud Marketplace (link: https://cloud.google.com/marketplace) offers pre-configured Cassandra images, enabling developers to quickly launch isolated instances for development. This approach minimizes central IT support effort as

developers can self-serve their environments, and leverages ready-made solutions optimized for Google Cloud, facilitating a fast migration with minimal overhead compared to building custom solutions or managing snapshots. Using marketplace images reduces the complexity and time associated with manual installation or image management, aligning with the requirements of speed and minimal support.

**Question: 123** CertyIQ

You have a Compute Engine instance hosting a production application. You want to receive an email if the instance consumes more than 90% of its CPU resources for more than 15 minutes. You want to use Google services. What should you do?

A. 1. Create a consumer Gmail account. 2. Write a script that monitors the CPU usage. 3. When the CPU usage exceeds the threshold, have that script send an email using the Gmail account and smtp.gmail.com on port 25 as SMTP server.

B. 1. Create a Stackdriver Workspace, and associate your Google Cloud Platform (GCP) project with it. 2. Create an Alerting Policy in Stackdriver that uses the threshold as a trigger condition. 3. Configure your email address in the notification channel.

C. 1. Create a Stackdriver Workspace, and associate your GCP project with it. 2. Write a script that monitors the CPU usage and sends it as a custom metric to Stackdriver. 3. Create an uptime check for the instance in Stackdriver.

D. 1. In Stackdriver Logging, create a logs-based metric to extract the CPU usage by using this regular expression: CPU Usage: ([0-9] 1,3 )% 2. In Stackdriver Monitoring, create an Alerting Policy based on this metric. 3. Configure your email address in the notification channel.

**Answer: B**

**Explanation:**

**Rationale:**

Option B is the most suitable solution because it directly utilizes Google Cloud's native monitoring service, Cloud Monitoring (formerly Stackdriver Monitoring), to achieve the desired outcome. Cloud Monitoring is designed for performance monitoring and alerting. Creating a Stackdriver Workspace and associating the GCP project allows access to monitoring data for the Compute Engine instance. Setting up an Alerting Policy in Stackdriver with a threshold of 90% CPU usage for 15 minutes triggers notifications when this condition is met. Configuring an email address as a notification channel ensures that alerts are sent via email. This approach is efficient, scalable, and fully managed by Google Cloud.

**Justification Sentence:**

Option B is the correct answer because it leverages Cloud Monitoring's built-in features to create an alert policy based on CPU utilization metrics for the Compute Engine instance, set a threshold of 90% for 15 minutes, and configure email notifications, providing a direct and managed solution for monitoring and alerting within Google Cloud.

**Supporting Links:**

**Cloud Monitoring Alerting Policies:** https://cloud.google.com/monitoring/alerts
**Create an Alerting Policy:** https://cloud.google.com/monitoring/alerts/create-policy
**Notification channels:** https://cloud.google.com/monitoring/support/notification-options

**Question: 124** CertyIQ

You have an application that uses Cloud Spanner as a backend database. The application has a very predictable

traffic pattern. You want to automatically scale up or down the number of Spanner nodes depending on traffic. What should you do?

A. Create a cron job that runs on a scheduled basis to review Cloud Monitoring metrics, and then resize the Spanner instance accordingly.

B. Create a Cloud Monitoring alerting policy to send an alert to oncall SRE emails when Cloud Spanner CPU exceeds the threshold. SREs would scale resources up or down accordingly.

C. Create a Cloud Monitoring alerting policy to send an alert to Google Cloud Support email when Cloud Spanner CPU exceeds your threshold. Google support would scale resources up or down accordingly.

D. Create a Cloud Monitoring alerting policy to send an alert to webhook when Cloud Spanner CPU is over or under your threshold. Create a Cloud Function that listens to HTTP and resizes Spanner resources accordingly.

**Answer: D**

**Explanation:**

Rationale: Option D is the most suitable solution because it leverages Cloud Monitoring's alerting capabilities to trigger automated scaling actions. Cloud Monitoring can send alerts to a webhook when Cloud Spanner CPU usage crosses predefined thresholds. This webhook can then invoke a Cloud Function, which contains code to automatically resize the Spanner instance by adjusting the number of nodes based on the alert condition. This approach enables automatic, metric-driven scaling, ensuring the Spanner instance adapts to predictable traffic patterns without manual intervention. Options A, B, and C are less ideal because they either rely on scheduled, less responsive scaling (A), manual intervention (B), or involve Google Support in customer infrastructure management (C), which is not their responsibility for scaling.

Supporting Links:

**Cloud Monitoring Alerting:** https://cloud.google.com/monitoring/alerts
**Cloud Functions HTTP Triggers:** https://cloud.google.com/functions/http-triggers
**Scaling Cloud Spanner instances:** https://cloud.google.com/spanner/docs/scaling-instances

**Question: 125** CertyIQ

Your company publishes large files on an Apache web server that runs on a Compute Engine instance. The Apache web server is not the only application running in the project. You want to receive an email when the egress network costs for the server exceed 100 dollars for the current month as measured by Google Cloud.
What should you do?

A. Set up a budget alert on the project with an amount of 100 dollars, a threshold of 100%, and notification type of email.

B. Set up a budget alert on the billing account with an amount of 100 dollars, a threshold of 100%, and notification type of email.

C. Export the billing data to BigQuery. Create a Cloud Function that uses BigQuery to sum the egress network costs of the exported billing data for the Apache web server for the current month and sends an email if it is over 100 dollars. Schedule the Cloud Function using Cloud Scheduler to run hourly.

D. Use the Cloud Logging Agent to export the Apache web server logs to Cloud Logging. Create a Cloud Function that uses BigQuery to parse the HTTP response log data in Cloud Logging for the current month and sends an email if the size of all HTTP responses, multiplied by current Google Cloud egress prices, totals over 100 dollars. Schedule the Cloud Function using Cloud Scheduler to run hourly.

**Answer: C**

**Explanation:**

The most effective solution is to **export billing data to BigQuery and utilize a Cloud Function with Cloud Scheduler to monitor egress costs and send email alerts.** This method allows for granular analysis of billing data, enabling you to filter and sum egress network costs specifically for the Apache web server. By

scheduling a Cloud Function to run hourly, you can regularly check if the $100 threshold is exceeded and receive timely email notifications. This approach provides accurate cost tracking and targeted alerting compared to broader budget alerts or log-based estimations.

**Supporting Links:**

**Export Billing Data to BigQuery:** https://cloud.google.com/billing/docs/how-to/export-data-bigquery
**Cloud Functions:** https://cloud.google.com/functions/docs
**Cloud Scheduler:** https://cloud.google.com/scheduler/docs
**BigQuery:** https://cloud.google.com/bigquery/docs

**Question: 126** CertyIQ

You have designed a solution on Google Cloud that uses multiple Google Cloud products. Your company has asked you to estimate the costs of the solution. You need to provide estimates for the monthly total cost. What should you do?

A. For each Google Cloud product in the solution, review the pricing details on the products pricing page. Use the pricing calculator to total the monthly costs for each Google Cloud product.

B. For each Google Cloud product in the solution, review the pricing details on the products pricing page. Create a Google Sheet that summarizes the expected monthly costs for each product.

C. Provision the solution on Google Cloud. Leave the solution provisioned for 1 week. Navigate to the Billing Report page in the Cloud Console. Multiply the 1 week cost to determine the monthly costs.

D. Provision the solution on Google Cloud. Leave the solution provisioned for 1 week. Use Cloud Monitoring to determine the provisioned and used resource amounts. Multiply the 1 week cost to determine the monthly costs.

**Answer: A**

**Explanation:**

Option A is the most accurate and efficient method because it utilizes Google Cloud's official pricing resources, the product pricing pages, and the Pricing Calculator, which are specifically designed for cost estimation. Reviewing pricing pages provides detailed pricing structures for each product, while the Pricing Calculator allows you to input your expected usage and generates a comprehensive monthly cost estimate. This approach avoids incurring unnecessary costs by provisioning resources and provides a proactive and informed estimation before deployment. Options C and D are inefficient and inaccurate as they involve provisioning resources and extrapolating costs from a short period, which might not reflect actual monthly usage and incur unnecessary expenses. Option B is less efficient than A as it involves manual calculation instead of leveraging the Pricing Calculator's automated estimation capabilities.

**Supporting Links:**

**Google Cloud Pricing Overview:** https://cloud.google.com/pricing - Provides a general overview of Google Cloud pricing principles and links to individual product pricing pages.
**Google Cloud Pricing Calculator:** https://cloud.google.com/products/calculator - Direct link to the official Google Cloud Pricing Calculator tool.
**Google Cloud Products:** https://cloud.google.com/products - Lists all Google Cloud products, from which you can navigate to specific product pricing pages.

**Question: 127** CertyIQ

You have an application that receives SSL-encrypted TCP traffic on port 443. Clients for this application are located all over the world. You want to minimize latency for the clients. Which load balancing option should you

use?

A. HTTPS Load Balancer

B. Network Load Balancer

C. SSL Proxy Load Balancer

D. Internal TCP/UDP Load Balancer. Add a firewall rule allowing ingress traffic from 0.0.0.0/0 on the target instances.

**Answer: C**

**Explanation:**

SSL Proxy Load Balancer is the most suitable option because it's a global load balancer that terminates SSL connections close to the clients, minimizing latency by utilizing Google's global network. It's designed to handle SSL-encrypted TCP traffic on any port, including 443, and distributes traffic to the closest backend instances. This global distribution ensures that clients worldwide connect to the nearest point of presence, reducing round trip time. Unlike Network Load Balancer which is regional, SSL Proxy Load Balancer offers global reach which is crucial for minimizing latency for clients across the globe. HTTPS Load Balancer is also global, but it is specifically designed for HTTP(S) traffic and adds HTTP-aware features, which might not be necessary and could introduce slightly more overhead compared to the more streamlined SSL Proxy for generic SSL traffic. Internal TCP/UDP Load Balancer is for internal traffic and not applicable for global clients.

Supporting links:

**Choosing a load balancer:** https://cloud.google.com/load-balancing/docs/load-balancing-overview
**SSL Proxy Load Balancer Overview:** https://cloud.google.com/load-balancing/docs/ssl-proxy
**Global load balancing:** https://cloud.google.com/load-balancing/docs/global-load-balancing

**Question: 128** CertyIQ

You have an application on a general-purpose Compute Engine instance that is experiencing excessive disk read throttling on its Zonal SSD Persistent Disk. The application primarily reads large files from disk. The disk size is currently 350 GB. You want to provide the maximum amount of throughput while minimizing costs.
What should you do?

A. Increase the size of the disk to 1 TB.

B. Increase the allocated CPU to the instance.

C. Migrate to use a Local SSD on the instance.

D. Migrate to use a Regional SSD on the instance.

**Answer: C**

**Explanation:**

Migrating to Local SSDs provides significantly higher read/write IOPS and throughput compared to Zonal SSD Persistent Disks, directly addressing disk read throttling issues for applications reading large files. Local SSDs are physically attached to the server, offering lower latency and higher performance, which maximizes throughput. While Local SSDs are ephemeral, their superior performance for read-intensive workloads makes them a suitable choice for maximizing throughput, potentially offsetting increased cost with improved application efficiency. Increasing Persistent Disk size (option A) improves throughput, but Local SSDs offer a greater performance boost. Regional SSDs (option D) prioritize availability over performance and are more expensive than Zonal SSDs. Increasing CPU (option B) is unlikely to directly resolve disk read throttling.

**Supporting Links:**

**Persistent Disk vs Local SSD:** https://cloud.google.com/compute/docs/disks/performance (This documentation compares performance characteristics of Persistent Disk and Local SSD.)
**Local SSD Overview:** https://cloud.google.com/compute/docs/disks/local-ssd (This provides details on Local SSD features and use cases.)
**Persistent Disk Overview:** https://cloud.google.com/compute/docs/disks/persistent-disks (This documentation explains Persistent Disk types and their characteristics, including Zonal and Regional SSD.)

**Question: 129** CertyIQ

Your Dataproc cluster runs in a single Virtual Private Cloud (VPC) network in a single subnet with range 172.16.20.128/25. There are no private IP addresses available in the VPC network. You want to add new VMs to communicate with your cluster using the minimum number of steps. What should you do?

A. Modify the existing subnet range to 172.16.20.0/24.

B. Create a new Secondary IP Range in the VPC and configure the VMs to use that range.

C. Create a new VPC network for the VMs. Enable VPC Peering between the VMs' VPC network and the Dataproc cluster VPC network.

D. Create a new VPC network for the VMs with a subnet of 172.32.0.0/16. Enable VPC network Peering between the Dataproc VPC network and the VMs VPC network. Configure a custom Route exchange.

**Answer: A**

**Explanation:**

Answer: **A**

Justification: Modifying the subnet range to 172.16.20.0/24 directly expands the available IP addresses within the existing VPC and subnet, providing space for new VMs to communicate with the Dataproc cluster, and this is the simplest approach with minimal steps as it avoids creating new VPCs, secondary IP ranges, or configuring VPC peering. Expanding the subnet is a straightforward way to increase IP address availability within the existing network infrastructure, thus requiring the fewest configuration changes. This solution directly addresses the problem of insufficient IP addresses in the current subnet by increasing its capacity.

Supporting Links:

Expanding subnet CIDR ranges

**Question: 130** CertyIQ

You manage an App Engine Service that aggregates and visualizes data from BigQuery. The application is deployed with the default App Engine Service account.
The data that needs to be visualized resides in a different project managed by another team. You do not have access to this project, but you want your application to be able to read data from the BigQuery dataset. What should you do?

A. Ask the other team to grant your default App Engine Service account the role of BigQuery Job User.

B. Ask the other team to grant your default App Engine Service account the role of BigQuery Data Viewer.

C. In Cloud IAM of your project, ensure that the default App Engine service account has the role of BigQuery Data Viewer.

D. In Cloud IAM of your project, grant a newly created service account from the other team the role of BigQuery Job User in your project.

**Answer: B**

**Explanation:**

The correct answer is **B**. Here's the justification: To allow your App Engine application to read data from a BigQuery dataset in another project, you need to grant the default App Engine service account the **BigQuery Data Viewer** role on the BigQuery dataset's project. This role specifically provides permissions to read BigQuery data without the ability to modify it. Option A, BigQuery Job User, grants broader permissions than necessary, including running jobs, which is not required for just viewing data. Options C and D are incorrect because they suggest modifying IAM settings in your project instead of the project containing the BigQuery dataset, and option D incorrectly proposes creating a new service account instead of utilizing the default App Engine service account. For further information, refer to Granting dataset access and Predefined roles and permissions.

**Question: 131** CertyIQ

You need to create a copy of a custom Compute Engine virtual machine (VM) to facilitate an expected increase in application traffic due to a business acquisition.
What should you do?

A. Create a Compute Engine snapshot of your base VM. Create your images from that snapshot.
B. Create a Compute Engine snapshot of your base VM. Create your instances from that snapshot.
C. Create a custom Compute Engine image from a snapshot. Create your images from that image.
D. Create a custom Compute Engine image from a snapshot. Create your instances from that image.

**Answer: D**

**Explanation:**

To copy a custom Compute Engine VM for scaling, first create a snapshot of the base VM's disk to capture its current state, then create a custom image from this snapshot which acts as a template containing the VM's configuration, and finally, launch new VM instances from this custom image to efficiently deploy identical copies for handling increased application traffic.

**Supporting Links:**

**Creating snapshots:** https://cloud.google.com/compute/docs/disks/create-snapshots
**Creating custom images:** https://cloud.google.com/compute/docs/images/create-custom-images
**Creating instances from images:** https://cloud.google.com/compute/docs/instances/create-start-instance

**Question: 132** CertyIQ

You have deployed an application on a single Compute Engine instance. The application writes logs to disk. Users start reporting errors with the application. You want to diagnose the problem. What should you do?

A. Navigate to Cloud Logging and view the application logs.
B. Connect to the instance's serial console and read the application logs.
C. Configure a Health Check on the instance and set a Low Healthy Threshold value.
D. Install and configure the Cloud Logging Agent and view the logs from Cloud Logging.

**Answer: D**

**Explanation:**

To diagnose application errors on a Compute Engine instance where logs are written to disk, you should install

and configure the Cloud Logging agent on the instance to forward these logs to Cloud Logging (formerly Stackdriver Logging), then view them within Cloud Logging for analysis and troubleshooting. This approach centralizes logs for easier inspection and allows you to efficiently identify the root cause of the reported application errors. The Cloud Logging agent is designed specifically for collecting logs from Compute Engine instances and is the recommended method for this scenario. Options A and B are less efficient or practical: Option A assumes logs are already in Cloud Logging without explicit agent configuration, while Option B, using the serial console, is not designed for application log retrieval and is inefficient for log analysis. Option C, health checks, monitor instance availability but do not provide application log details for error diagnosis.

**Supporting Links:**

**Cloud Logging Agent Installation:** https://cloud.google.com/logging/docs/agent/installation
**Cloud Logging Overview:** https://cloud.google.com/logging/docs/overview
**Compute Engine Logging:** https://cloud.google.com/logging/docs/setup/platform-logs/compute-engine

**Question: 133** CertyIQ

An application generates daily reports in a Compute Engine virtual machine (VM). The VM is in the project corp-iot-insights. Your team operates only in the project corp-aggregate-reports and needs a copy of the daily exports in the bucket corp-aggregate-reports-storage. You want to configure access so that the daily reports from the VM are available in the bucket corp-aggregate-reports-storage and use as few steps as possible while following Google-recommended practices. What should you do?

A. Move both projects under the same folder.

B. Grant the VM Service Account the role Storage Object Creator on corp-aggregate-reports-storage.

C. Create a Shared VPC network between both projects. Grant the VM Service Account the role Storage Object Creator on corp-iot-insights.

D. Make corp-aggregate-reports-storage public and create a folder with a pseudo-randomized suffix name. Share the folder with the IoT team.

**Answer: B**

**Explanation:**

Option B, granting the VM's service account the "Storage Object Creator" role on the corp-aggregate-reports-storage bucket, is the most efficient and secure solution. This approach adheres to the principle of least privilege, granting only the necessary permissions for the VM to write objects to the target bucket. The VM, by virtue of its service account, is then authorized to upload the reports directly to the bucket in the corp-aggregate-reports project without any complex configurations. No changes within the project structure or network topology are required. This method also leverages the identity and access management (IAM) controls, which is a best practice for managing permissions within Google Cloud. Alternative approaches like making the bucket public (Option D) introduce severe security risks and are unsuitable for production systems. Moving projects (Option A) is an overly complex and unnecessary step for data transfer. Using Shared VPC (Option C) adds unnecessary complexity, as this use case doesn't require network-level sharing, but rather, a simple data transfer. The focus should be on granting the correct IAM permissions directly.

Relevant Google Cloud documentation:

Service accounts: Provides information on how service accounts are used for authentication and authorization.
IAM roles for Storage: Outlines the different IAM roles available for Google Cloud Storage, such as Storage Object Creator.
Principle of least privilege: Explains the importance of granting minimum required permissions for security.

**Question: 134** CertyIQ

You built an application on your development laptop that uses Google Cloud services. Your application uses Application Default Credentials for authentication and works fine on your development laptop. You want to migrate this application to a Compute Engine virtual machine (VM) and set up authentication using Google-recommended practices and minimal changes. What should you do?

A. Assign appropriate access for Google services to the service account used by the Compute Engine VM.

B. Create a service account with appropriate access for Google services, and configure the application to use this account.

C. Store credentials for service accounts with appropriate access for Google services in a config file, and deploy this config file with your application.

D. Store credentials for your user account with appropriate access for Google services in a config file, and deploy this config file with your application.

**Answer: B**

**Explanation:**

The correct answer is **B. Create a service account with appropriate access for Google services, and configure the application to use this account.**

Here's why:

**Principle of Least Privilege:** Option B aligns with the principle of least privilege, which dictates granting only the necessary permissions to perform a task. Service accounts are designed for applications, allowing for fine-grained control over resource access.
**Security Best Practice:** Storing user credentials (options C and D) is a major security risk and not recommended. User credentials have broader permissions than needed for an application.
**Compute Engine Service Account:** Compute Engine VMs have an associated service account, but it might not have the permissions required by your application. Option A suggests simply using this existing service account, but it might be overly permissive or lack specific roles.
**Application Default Credentials (ADC):** ADC on your local machine works based on your user account and development setup, which won't translate to a VM. Options C and D also aim to recreate a similar mechanism, but using service accounts is the best practice.
**Service Account Management:** By creating a dedicated service account (Option B), you can assign the specific Identity and Access Management (IAM) roles your application needs to access Google Cloud resources, avoiding the risks of overly broad permissions.
**Clean and Maintainable:** A service account dedicated to your application makes the environment cleaner and easier to manage. It allows you to audit, revoke, or modify permissions separately from individual user accounts.
**IAM Roles:** You grant specific permissions to service accounts using IAM roles, which offer a granular way to control access based on functions.

**In summary, using a dedicated service account with the appropriate IAM roles is the recommended method to authenticate applications running on Compute Engine VMs, providing both security and fine-grained control.**

**Relevant Links:**

**Google Cloud IAM documentation:** https://cloud.google.com/iam/docs/
**Service Accounts:** https://cloud.google.com/iam/docs/service-accounts
**Compute Engine service accounts:** https://cloud.google.com/compute/docs/access/service-accounts
**Application Default Credentials:** https://cloud.google.com/docs/authentication/application-default-credentials

**Question: 135** CertyIQ

You need to create a Compute Engine instance in a new project that doesn't exist yet. What should you do?

A. Using the Cloud SDK, create a new project, enable the Compute Engine API in that project, and then create the instance specifying your new project.

B. Enable the Compute Engine API in the Cloud Console, use the Cloud SDK to create the instance, and then use the --project flag to specify a new project.

C. Using the Cloud SDK, create the new instance, and use the --project flag to specify the new project. Answer yes when prompted by Cloud SDK to enable the Compute Engine API.

D. Enable the Compute Engine API in the Cloud Console. Go to the Compute Engine section of the Console to create a new instance, and look for the Create In A New Project option in the creation form.

**Answer: A**

**Explanation:**

Option A is the correct approach because it outlines the necessary steps for creating a Compute Engine instance within a completely new Google Cloud Project. Before any resources can be deployed, a project must exist. Therefore, the process must begin with project creation, which the Cloud SDK facilitates through commands like gcloud projects create. Once the project is established, enabling the relevant API is crucial. In this case, it's the Compute Engine API, as it provides the necessary services to launch and manage virtual machines. This is done programmatically via gcloud services enable compute.googleapis.com. Finally, after ensuring the foundation is in place, the instance creation can proceed using the Cloud SDK and specifying the new project via the --project flag. This ensures the instance is placed in the intended project.

Option B is incorrect because it incorrectly tries to use the --project flag for creation, while a project is not already created. Option C is incorrect because you cannot create an instance in a non-existing project. While the Cloud SDK can prompt for API enabling, relying on implicit activation is less robust and less explicit. Option D is incorrect because, although the Cloud Console can create instances, it doesn't automatically create a new project via an instance creation process. It generally requires manual project creation.

The core concepts involved are Project Management and API enabling within GCP. A project is the fundamental organizational unit in GCP, and APIs provide the specific functionality offered by different services. The Cloud SDK provides a command-line interface to interact with Google Cloud resources and allows for more automation and scripting of resource creation. This approach aligns with best practices, including explicitly creating the project and enabling APIs before resource deployment.

**Relevant Links:**

**Creating and Managing Projects:** https://cloud.google.com/resource-manager/docs/creating-managing-projects
**Enabling and Disabling Services:** https://cloud.google.com/service-usage/docs/enable-disable
**gcloud compute instances create:** https://cloud.google.com/sdk/gcloud/reference/compute/instances/create

**Question: 136** CertyIQ

Your company runs one batch process in an on-premises server that takes around 30 hours to complete. The task runs monthly, can be performed offline, and must be restarted if interrupted. You want to migrate this workload to the cloud while minimizing cost. What should you do?

A. Migrate the workload to a Compute Engine Preemptible VM.

B. Migrate the workload to a Google Kubernetes Engine cluster with Preemptible nodes.

C. Migrate the workload to a Compute Engine VM. Start and stop the instance as needed.

D. Create an Instance Template with Preemptible VMs On. Create a Managed Instance Group from the template

and adjust Target CPU Utilization. Migrate the workload.

**Answer: C**

**Explanation:**

The optimal solution is to migrate the workload to a regular Compute Engine VM, starting and stopping it as needed (Option C). This approach best addresses the requirements of a monthly, interruptible, long-running batch process while minimizing cost. Preemptible VMs (Options A, B, and D) are significantly cheaper but can be terminated by Google with short notice, making them unsuitable for a 30-hour process that must complete. If a preemptible VM were terminated mid-process, the work would be lost, and the process would need to restart, negating cost savings. Additionally, managing a Kubernetes cluster (Option B) or a Managed Instance Group (Option D) adds unnecessary complexity and overhead for a single, offline batch job. Starting and stopping a regular Compute Engine VM ensures the process has the necessary compute resources only when needed, directly controlling costs. This simple approach is the most cost-effective and reliable option for a non-time-sensitive, interruptible batch workload requiring 30 hours of runtime. The VM will only incur costs when it is running.

Relevant links for further research:

**Compute Engine Pricing:** https://cloud.google.com/compute/pricing
**Preemptible VMs:** https://cloud.google.com/compute/docs/instances/preemptible
**Managing VM instances:** https://cloud.google.com/compute/docs/instances/start-stop-instance

**Question: 137** CertyIQ

You are developing a new application and are looking for a Jenkins installation to build and deploy your source code. You want to automate the installation as quickly and easily as possible. What should you do?

A. Deploy Jenkins through the Google Cloud Marketplace.

B. Create a new Compute Engine instance. Run the Jenkins executable.

C. Create a new Kubernetes Engine cluster. Create a deployment for the Jenkins image.

D. Create an instance template with the Jenkins executable. Create a managed instance group with this template.

**Answer: A**

**Explanation:**

The most efficient and rapid way to deploy Jenkins for a new application, as described in the scenario, is by utilizing the Google Cloud Marketplace (option A). The Cloud Marketplace provides pre-configured, ready-to-deploy solutions, including Jenkins, eliminating the need for manual installation and configuration. This approach significantly reduces setup time and complexity compared to the other options. Creating a Compute Engine instance and manually installing Jenkins (option B) requires more manual effort and time for installation, security hardening, and ongoing maintenance. Option C, deploying Jenkins on a Kubernetes Engine cluster, is more complex and overkill for an initial quick setup, involving cluster management, pod deployments, and potentially more resource management. Similarly, using instance templates and managed instance groups (option D) is unnecessary complexity for a simple and quick Jenkins deployment. The Cloud Marketplace solution offers a streamlined, pre-packaged, and tested deployment, ensuring a faster time to value for the application development workflow. It allows developers to focus on their code instead of infrastructure management. This leverages the principle of infrastructure as code and pre-built deployments, aligning with best practices for cloud adoption.

Relevant Links:

**Google Cloud Marketplace:** https://cloud.google.com/marketplace
**Jenkins on Google Cloud Marketplace:** Search for "Jenkins" within the marketplace to find available options.

**Question: 138** CertyIQ

You have downloaded and installed the gcloud command line interface (CLI) and have authenticated with your Google Account. Most of your Compute Engine instances in your project run in the europe-west1-d zone. You want to avoid having to specify this zone with each CLI command when managing these instances.
What should you do?

A. Set the europe-west1-d zone as the default zone using the gcloud config subcommand.
B. In the Settings page for Compute Engine under Default location, set the zone to europe"west1-d.
C. In the CLI installation directory, create a file called default.conf containing zone=europe"west1"d.
D. Create a Metadata entry on the Compute Engine page with key compute/zone and value europe"west1"d.

**Answer: A**

**Explanation:**

The correct answer is **A. Set the europe-west1-d zone as the default zone using the gcloud config subcommand.**

Here's the justification:

The gcloud CLI is a powerful tool for managing Google Cloud resources. To streamline workflows and avoid repetitive typing, it allows users to set default configurations. Specifying a default zone eliminates the need to explicitly include the --zone flag in every gcloud command that interacts with resources in that zone. This significantly enhances efficiency and reduces the potential for errors.

Option A directly addresses this need by using the gcloud config command, which is specifically designed for configuring the gcloud CLI's behavior. The command would be something like: gcloud config set compute/zone europe-west1-d. This command modifies the gcloud configuration, making 'europe-west1-d' the default zone for subsequent operations.

Options B, C, and D are incorrect. Option B, modifying settings in the Compute Engine console's settings page, does not control the gcloud CLI's default zone setting. Option C suggests modifying a file in the installation directory, which is not the documented method and could easily break with updates. Option D, which deals with metadata, is for instance-specific configurations and not relevant to the CLI's default behavior.

In summary, option A is the best approach because it uses the appropriate gcloud command for configuring the default zone, adhering to Google Cloud's best practices for CLI usage. This approach enhances efficiency and prevents typing errors when managing Compute Engine resources in a specific zone.

**Authoritative Links for Further Research:**

**Setting Cloud SDK Properties:** https://cloud.google.com/sdk/docs/properties
**gcloud config set:** https://cloud.google.com/sdk/gcloud/reference/config/set
**gcloud Command-Line Tool Overview:** https://cloud.google.com/sdk/docs/overview

**Question: 139** CertyIQ

The core business of your company is to rent out construction equipment at large scale. All the equipment that is being rented out has been equipped with multiple sensors that send event information every few seconds. These signals can vary from engine status, distance traveled, fuel level, and more. Customers are billed based on the

consumption monitored by these sensors. You expect high throughput `" up to thousands of events per hour per device `" and need to retrieve consistent data based on the time of the event. Storing and retrieving individual signals should be atomic. What should you do?

A. Create a file in Cloud Storage per device and append new data to that file.
B. Create a file in Cloud Filestore per device and append new data to that file.
C. Ingest the data into Datastore. Store data in an entity group based on the device.
D. Ingest the data into Cloud Bigtable. Create a row key based on the event timestamp.

**Answer: D**

**Explanation:**

The correct answer is D, ingesting data into Cloud Bigtable with a row key based on the event timestamp. Here's why:

**High Throughput & Scalability:** Bigtable is a fully managed, scalable NoSQL database designed for high throughput and low latency workloads, perfectly suited for ingesting thousands of events per hour per device. This aligns with the problem's requirement for handling massive sensor data influx.
https://cloud.google.com/bigtable/docs/overview

**Time Series Data:** Sensor data is inherently time-series data, and Bigtable's row key design, based on the event timestamp, allows for efficient retrieval of data based on time ranges. This design enables fast reads and scans for billing calculations, which are crucial for the company.
https://cloud.google.com/bigtable/docs/schema-design

**Atomic Operations:** Bigtable ensures atomic operations at the row level. By storing each event as a single row, retrieving and storing individual sensor readings is guaranteed to be an atomic operation.
https://cloud.google.com/bigtable/docs/writes

**Other Options Are Less Suitable:**

**Option A (Cloud Storage):** While Cloud Storage is great for storing files, appending data to files isn't designed for high throughput or atomic reads and writes. Retrieval and processing would also require extra work to read and parse the data which is not optimal.
**Option B (Cloud Filestore):** Cloud Filestore is a network file system not suitable for high throughput ingestion from thousands of devices and atomic operations. It is designed for shared file access across VM's.
**Option C (Datastore):** Although Datastore allows grouping by entity, it is not designed for the scale of this problem. Datastore does not scale as well as Bigtable, especially for the given high throughput. The "strong consistency" requirement needed for billing in conjunction with scale may require you to use a multi-region deployment that would further impact Datastore's price point and performance.
https://cloud.google.com/datastore/docs/concepts/overview

In summary, Bigtable's ability to handle high throughput, its suitability for time-series data using timestamp row keys, and support for atomic operations makes it the ideal choice for this scenario.

**Question: 140** CertyIQ

You are asked to set up application performance monitoring on Google Cloud projects A, B, and C as a single pane of glass. You want to monitor CPU, memory, and disk. What should you do?

A. Enable API and then share charts from project A, B, and C.
B. Enable API and then give the metrics.reader role to projects A, B, and C.
C. Enable API and then use default dashboards to view all projects in sequence.

D. Enable API, create a workspace under project A, and then add projects B and C.

**Answer: D**

**Explanation:**

The correct answer is **D: Enable API, create a workspace under project A, and then add projects B and C.** Here's why:

Google Cloud's monitoring solution, Cloud Monitoring, relies on the concept of "monitoring scopes" and "workspaces." A workspace serves as the single pane of glass to aggregate metrics from multiple projects. To effectively monitor projects A, B, and C together, you need a centralized workspace. Option D achieves this by creating a workspace in one of the projects (project A) and then adding the other projects (B and C) to that scope. This allows a single view of aggregated metrics like CPU, memory, and disk across all projects.

Options A, B, and C are inadequate. Sharing charts (A) is a manual and cumbersome approach, not providing a single, dynamic view. Granting the metrics.reader role (B) allows project A to read the metrics of other projects, but doesn't combine them into a central dashboard. Using default dashboards in sequence (C) doesn't create a unified view but requires jumping between project dashboards. Thus, they fail to provide the required single-pane-of-glass monitoring experience.

The workspace approach in option D leverages the power of Cloud Monitoring's scoping feature, providing a comprehensive monitoring solution. It's more efficient and practical for managing multiple projects and their metrics in a consolidated way.

For further research, refer to these official Google Cloud documentation links:

Managing workspaces: This page explains how workspaces function and their significance for multi-project monitoring.
Monitoring multiple projects: This page provides guidance on setting up monitoring for multi-project environments.
Cloud Monitoring overview: This page offers a general overview of the entire Google Cloud Monitoring service.

**Question: 141** CertyIQ

You created several resources in multiple Google Cloud projects. All projects are linked to different billing accounts. To better estimate future charges, you want to have a single visual representation of all costs incurred. You want to include new cost data as soon as possible. What should you do?

A. Configure Billing Data Export to BigQuery and visualize the data in Data Studio.
B. Visit the Cost Table page to get a CSV export and visualize it using Data Studio.
C. Fill all resources in the Pricing Calculator to get an estimate of the monthly cost.
D. Use the Reports view in the Cloud Billing Console to view the desired cost information.

**Answer: A**

**Explanation:**

Option A is the correct solution because it leverages Google Cloud's robust data export and visualization capabilities for detailed cost analysis. Configuring Billing Data Export to BigQuery automatically streams billing data into a BigQuery dataset. This ensures that new cost data is available almost in real-time, fulfilling the requirement of getting data "as soon as possible." BigQuery allows for querying and aggregating large datasets, enabling cost breakdowns by project, resource, or other desired dimensions. Then, Data Studio can connect directly to this BigQuery data to create custom dashboards and visualizations. This provides a single, dynamic view of all costs across multiple projects and billing accounts, which meets the user's need for a

"single visual representation of all costs." Option B, using a CSV export from the Cost Table page, requires manual intervention to export and upload the data, which is not suitable for frequent updates. Option C, using the Pricing Calculator, only provides estimates, not actual costs. Option D, using the Reports view, might provide aggregated cost views, but it lacks the flexibility and customization of a Data Studio dashboard based on exported data. Option A is the most scalable, automated, and customizable method for ongoing cost monitoring.

https://cloud.google.com/billing/docs/how-to/export-data-bigqueryhttps://cloud.google.com/bigquery/docs/https://datastudio.google.com/

**Question: 142** CertyIQ

Your company has workloads running on Compute Engine and on-premises. The Google Cloud Virtual Private Cloud (VPC) is connected to your WAN over a
Virtual Private Network (VPN). You need to deploy a new Compute Engine instance and ensure that no public Internet traffic can be routed to it. What should you do?

A. Create the instance without a public IP address.
B. Create the instance with Private Google Access enabled.
C. Create a deny-all egress firewall rule on the VPC network.
D. Create a route on the VPC to route all traffic to the instance over the VPN tunnel.

**Answer: A**

**Explanation:**

The correct answer is **A. Create the instance without a public IP address.**

Here's why:

The core requirement is to prevent public internet traffic from reaching the new Compute Engine instance. A public IP address directly exposes the instance to the internet. By omitting the external IP assignment during instance creation, the instance becomes inherently private, reachable only through the internal VPC network. This approach ensures no public route exists by default, fulfilling the requirement.

Private Google Access (option B) allows instances without public IPs to access Google APIs and services but doesn't inherently block inbound public traffic. Egress firewall rules (option C) control outbound traffic, not inbound. And routing traffic over the VPN tunnel (option D) ensures traffic from on-premises can reach it, but does not inherently restrict public internet access.

Therefore, creating the instance without a public IP address provides the most direct and effective method to satisfy the given condition. It aligns with the principle of least privilege by not exposing the instance to the internet in the first place. This avoids the complexity of relying on firewall rules and is the most straightforward method to achieve the desired outcome.

**Supporting Concepts:**

**Private IP Address:** An IP address within the VPC network, used for internal communication.
**Public IP Address:** An IP address that is routable over the internet.
**VPC Network:** A logically isolated network within Google Cloud that provides private communication between resources.

**Authoritative Links:**

**Google Cloud Documentation - IP Addresses:** https://cloud.google.com/vpc/docs/vpc

**Google Cloud Documentation - Compute Engine Instances:** https://cloud.google.com/compute/docs/instances
**Google Cloud Documentation - Private Google Access:** https://cloud.google.com/vpc/docs/private-google-access

**Question: 143** CertyIQ

Your team maintains the infrastructure for your organization. The current infrastructure requires changes. You need to share your proposed changes with the rest of the team. You want to follow Google's recommended best practices. What should you do?

A. Use Deployment Manager templates to describe the proposed changes and store them in a Cloud Storage bucket.

B. Use Deployment Manager templates to describe the proposed changes and store them in Cloud Source Repositories.

C. Apply the changes in a development environment, run gcloud compute instances list, and then save the output in a shared Storage bucket.

D. Apply the changes in a development environment, run gcloud compute instances list, and then save the output in Cloud Source Repositories.

**Answer: B**

**Explanation:**

The correct answer is **B. Use Deployment Manager templates to describe the proposed changes and store them in Cloud Source Repositories.**

Here's why:

**Infrastructure as Code (IaC):** Google recommends managing infrastructure using IaC principles. This involves defining infrastructure through code, allowing for version control, collaboration, and repeatability. Deployment Manager templates facilitate this by letting you describe your cloud resources in a declarative way.
**Version Control:** Cloud Source Repositories provides a centralized and version-controlled environment for storing your IaC configurations (Deployment Manager templates). This enables tracking changes, reverting to previous states, and facilitating collaboration among team members. This is crucial for managing infrastructure safely.
**Collaboration:** Storing templates in a repository allows multiple team members to access, review, and contribute to the infrastructure configuration. This collaborative approach aligns with DevOps best practices.
**Deployment Manager Templates:** These templates are written in YAML or Python and define the resources needed for your infrastructure. They enable you to consistently create, update, or delete resources, thus promoting consistency and automation.
**Why not other options?**

Option A, while using Deployment Manager, stores templates in a Cloud Storage bucket. Cloud Storage is meant for object storage, not version control or code management, and would be less suitable for tracking infrastructure changes.
Options C and D suggest capturing the output of gcloud compute instances list after applying changes. This approach is not IaC because the desired state of the infrastructure is not captured declaratively. The output list represents the current state of the infrastructure, not the desired state. Also, it doesn't allow for changes. Additionally, it's difficult to manage the changes and doesn't promote easy collaboration.

**In summary, storing Deployment Manager templates in Cloud Source Repositories is the preferred approach because it aligns with IaC best practices, allows for version control, and promotes collaboration among team members.**

**Authoritative Links:**

**Deployment Manager Documentation:** https://cloud.google.com/deployment-manager/docs
**Cloud Source Repositories Documentation:** https://cloud.google.com/source-repositories/docs
**Infrastructure as Code:** https://cloud.google.com/solutions/infrastructure-as-code

**Question: 144** CertyIQ

You have a Compute Engine instance hosting an application used between 9 AM and 6 PM on weekdays. You want to back up this instance daily for disaster recovery purposes. You want to keep the backups for 30 days. You want the Google-recommended solution with the least management overhead and the least number of services. What should you do?

A. 1. Update your instances' metadata to add the following value: snapshot"schedule: 0 1 * * * 2. Update your instances' metadata to add the following value: snapshot"retention: 30

B. 1. In the Cloud Console, go to the Compute Engine Disks page and select your instance's disk. 2. In the Snapshot Schedule section, select Create Schedule and configure the following parameters: - Schedule frequency: Daily - Start time: 1:00 AM " 2:00 AM - Autodelete snapshots after: 30 days

C. 1. Create a Cloud Function that creates a snapshot of your instance's disk. 2. Create a Cloud Function that deletes snapshots that are older than 30 days. 3. Use Cloud Scheduler to trigger both Cloud Functions daily at 1:00 AM.

D. 1. Create a bash script in the instance that copies the content of the disk to Cloud Storage. 2. Create a bash script in the instance that deletes data older than 30 days in the backup Cloud Storage bucket. 3. Configure the instance's crontab to execute these scripts daily at 1:00 AM.

**Answer: B**

**Explanation:**

Option B is the correct solution because it leverages Google Compute Engine's built-in snapshot scheduling feature, aligning with the requirements of least management overhead and minimal services. This feature is specifically designed for automated disk backups, making it a Google-recommended practice for disaster recovery. By creating a snapshot schedule directly on the disk, you eliminate the need for custom scripts, Cloud Functions, or Cloud Scheduler, simplifying the backup process significantly. The "Autodelete snapshots after 30 days" parameter automatically handles snapshot retention, further reducing management effort. Option A attempts to use metadata for snapshot scheduling, which is not how snapshot schedules are configured in Google Cloud. Option C introduces two Cloud Functions and Cloud Scheduler, which adds unnecessary complexity and more services, violating the least overhead requirement. Option D involves managing scripts within the instance itself, making it less reliable and more prone to errors than the managed snapshot service. Furthermore, storing backups in Cloud Storage via scripts is not the recommended method for Compute Engine disk backups. The integrated snapshot service in Compute Engine is optimized for disk backups and provides better performance and consistency than custom scripts. Therefore, using the built-in snapshot schedule feature is the optimal solution, offering simplicity, reliability, and minimal management overhead while adhering to Google's best practices.

For further information, refer to the following resources:

**Google Cloud Documentation on Snapshot Schedules:**
https://cloud.google.com/compute/docs/disks/scheduled-snapshots
**Google Cloud Documentation on Creating Disk Snapshots:**
https://cloud.google.com/compute/docs/disks/create-snapshots
**Google Cloud Best Practices for Disaster Recovery:** https://cloud.google.com/solutions/disaster-recovery

**Question: 145** CertyIQ

Your existing application running in Google Kubernetes Engine (GKE) consists of multiple pods running on four GKE n1`"standard`"2 nodes. You need to deploy additional pods requiring n2`"highmem`"16 nodes without any downtime. What should you do?

A. Use gcloud container clusters upgrade. Deploy the new services.

B. Create a new Node Pool and specify machine type n2"highmem"16. Deploy the new pods.

C. Create a new cluster with n2"highmem"16 nodes. Redeploy the pods and delete the old cluster.

D. Create a new cluster with both n1"standard"2 and n2"highmem"16 nodes. Redeploy the pods and delete the old cluster.

**Answer: B**

**Explanation:**

Option B is the correct solution because it leverages Kubernetes' ability to manage workloads across different node pools within the same cluster. A node pool is a subset of nodes within a GKE cluster that share the same configuration, such as machine type and disk size. By creating a new node pool specifically for the n2-highmem-16 machines, you can introduce these larger nodes into your existing cluster without disrupting the already running pods on the n1-standard-2 nodes. Once the new node pool is ready, you can deploy the new pods requiring more memory and processing power directly to the new node pool, leveraging Kubernetes' scheduling capabilities to place the pods on the correct node type. This method avoids any downtime as your existing application remains functional on the original nodes. gcloud container clusters upgrade (Option A) is for upgrading Kubernetes versions, not for adding new node types. Creating an entirely new cluster (Options C and D) and redeploying introduces unnecessary complexity and downtime as you would need to migrate the entire application. Option D is also problematic because it unnecessarily mixes node types in a single pool, making resource management less efficient. To learn more about node pools, refer to: https://cloud.google.com/kubernetes-engine/docs/concepts/node-pools and https://cloud.google.com/kubernetes-engine/docs/how-to/node-pools. These resources will enhance your comprehension of the practical use of node pools in GKE.

**Question: 146** CertyIQ

You have an application that uses Cloud Spanner as a database backend to keep current state information about users. Cloud Bigtable logs all events triggered by users. You export Cloud Spanner data to Cloud Storage during daily backups. One of your analysts asks you to join data from Cloud Spanner and Cloud Bigtable for specific users. You want to complete this ad hoc request as efficiently as possible. What should you do?

A. Create a dataflow job that copies data from Cloud Bigtable and Cloud Storage for specific users.

B. Create a dataflow job that copies data from Cloud Bigtable and Cloud Spanner for specific users.

C. Create a Cloud Dataproc cluster that runs a Spark job to extract data from Cloud Bigtable and Cloud Storage for specific users.

D. Create two separate BigQuery external tables on Cloud Storage and Cloud Bigtable. Use the BigQuery console to join these tables through user fields, and apply appropriate filters.

**Answer: D**

**Explanation:**

The most efficient solution for joining data from Cloud Spanner (exported to Cloud Storage) and Cloud Bigtable for ad-hoc analysis is option D: creating BigQuery external tables and joining them in BigQuery. This approach leverages BigQuery's powerful query engine and avoids unnecessary data movement or complex

processing pipelines.

Here's why:

**BigQuery External Tables:** BigQuery allows you to query data stored outside of BigQuery using external tables. You can create external tables directly on Cloud Storage (where your Spanner backups reside) and Cloud Bigtable without needing to ingest the data into BigQuery storage. This is much faster than copying the data using dataflow or other methods. https://cloud.google.com/bigquery/docs/external-data-sources
**BigQuery SQL for Joining:** BigQuery is designed for large-scale data analysis and provides SQL capabilities to join data from different tables, including external tables. You can use standard SQL syntax with WHERE clauses and JOIN operations to combine data based on user IDs and apply filters as needed. https://cloud.google.com/bigquery/docs/query-syntax
**Efficiency:** By avoiding copying data with Dataflow or processing with Dataproc, you minimize the time and resources required for the analysis. External tables are cost-effective, and BigQuery only processes data relevant to your query.
**Ad-Hoc Nature:** This approach perfectly suits ad-hoc requests. You can quickly create external tables and run queries without writing and deploying complex pipelines.
**Data Formats:** BigQuery supports many formats for external tables, including formats typically used by Cloud Spanner backups. Bigtable data can also be directly queried via a BigQuery external table.

Options A and B using Dataflow involve more complex pipeline development, deployment, and data copying which are not efficient for an ad-hoc analysis. Option C using Dataproc with Spark involves cluster management overhead and more complex data processing coding compared to a direct SQL query on external tables in BigQuery.

**Question: 147** CertyIQ

You are hosting an application from Compute Engine virtual machines (VMs) in us`"central1`"a. You want to adjust your design to support the failure of a single
Compute Engine zone, eliminate downtime, and minimize cost. What should you do?

A. " Create Compute Engine resources in us"central1"b. " Balance the load across both us"central1"a and us"central1"b.

B. " Create a Managed Instance Group and specify us"central1"a as the zone. " Configure the Health Check with a short Health Interval.

C. " Create an HTTP(S) Load Balancer. " Create one or more global forwarding rules to direct traffic to your VMs.

D. " Perform regular backups of your application. " Create a Cloud Monitoring Alert and be notified if your application becomes unavailable. " Restore from backups when notified.

**Answer: A**

**Explanation:**

Okay, let's break down why option A is the correct approach for achieving high availability, minimizing downtime, and optimizing cost when hosting an application on Compute Engine.

The core issue is zone failure. Option A directly addresses this by suggesting the creation of identical resources in another zone within the same region (us-central1b). This provides redundancy; if us-central1a goes down, the application can continue running from us-central1b. Load balancing across both zones is crucial for ensuring even distribution of traffic and automatic failover if a zone becomes unavailable. This ensures no single point of failure exists. Furthermore, staying within the same region minimizes latency, and avoid cross-region data transfer charges.

Option B is inadequate because it confines the resources to a single zone (us-central1a) and doesn't solve for zone failure, which is the core requirement of the question. Short health interval just helps detect a fault faster, it does not provide high availability. Option C focuses on load balancing, but it doesn't create the underlying infrastructure needed for high availability. It's a necessary component, but insufficient on its own. Option D focuses on backup and restore strategy, which leads to application downtime if not combined with redundancy in place, and does not solve the immediate issue of zone failure.

Therefore, the combination of redundancy through a second zone and load balancing (option A) is the best approach to achieve the requirements. This provides a cost-effective solution, minimizes downtime by enabling seamless failover, and provides better resilience.

**Authoritative Links:**

**Google Cloud Documentation on Regions and Zones:** https://cloud.google.com/compute/docs/regions-zones
**Google Cloud Documentation on Load Balancing:** https://cloud.google.com/load-balancing/docs
**Google Cloud Documentation on Managed Instance Groups:**
https://cloud.google.com/compute/docs/instance-groups

**Question: 148** CertyIQ

A colleague handed over a Google Cloud Platform project for you to maintain. As part of a security checkup, you want to review who has been granted the Project
Owner role. What should you do?

A. In the console, validate which SSH keys have been stored as project-wide keys.
B. Navigate to Identity-Aware Proxy and check the permissions for these resources.
C. Enable Audit Logs on the IAM & admin page for all resources, and validate the results.
D. Use the command gcloud projects get"iam"policy to view the current role assignments.

**Answer: D**

**Explanation:**

The correct answer is **D. Use the command gcloud projects get-iam-policy to view the current role assignments.** This is because the gcloud projects get-iam-policy command is the primary method for programmatically retrieving the IAM policy associated with a Google Cloud Project, including which users have been granted the Project Owner role. IAM (Identity and Access Management) is the core service for controlling access to Google Cloud resources, and understanding the IAM policy is fundamental for security audits. This command provides a clear and auditable view of all role assignments, making it ideal for identifying who has the Project Owner role. Option A is incorrect because SSH keys are not directly related to IAM role assignments; they primarily control access to virtual machines. Option B is incorrect as Identity-Aware Proxy (IAP) manages application access, not project-level IAM roles. Option C is also incorrect; while Audit Logs are useful for tracking changes, they don't provide the current IAM policy view; they only show a history of events. The gcloud command provides a snapshot of current policy. The use of gcloud is the most efficient and accurate way to directly view project IAM permissions compared to the other options.

For further research, consult the following official Google Cloud documentation:

**Managing access using IAM:** https://cloud.google.com/iam/docs/overview
**gcloud projects get-iam-policy documentation:**
https://cloud.google.com/sdk/gcloud/reference/projects/get-iam-policy

**Question: 149** CertyIQ

You are running multiple VPC-native Google Kubernetes Engine clusters in the same subnet. The IPs available for the nodes are exhausted, and you want to ensure that the clusters can grow in nodes when needed. What should you do?

A. Create a new subnet in the same region as the subnet being used.

B. Add an alias IP range to the subnet used by the GKE clusters.

C. Create a new VPC, and set up VPC peering with the existing VPC.

D. Expand the CIDR range of the relevant subnet for the cluster.

**Answer: D**

**Explanation:**

The correct answer is **D. Expand the CIDR range of the relevant subnet for the cluster.**

Here's why:

The problem states that IP addresses within the existing subnet are exhausted, preventing GKE clusters from scaling by adding more nodes. VPC-native GKE clusters directly use IP addresses from the VPC subnet for their nodes. Therefore, the solution involves increasing the available IP address space within the subnet.

Expanding the CIDR range of the subnet achieves this. By increasing the subnet's address space, more IP addresses become available for allocation to GKE nodes. Other options are not ideal solutions: creating a new subnet (A) or VPC (C) and VPC peering would introduce unnecessary complexity and management overhead. Adding alias IP ranges (B) primarily addresses pod IP allocation, not node IP allocation and will not directly fix the IP exhaustion for nodes. Expanding the existing subnet is a direct and efficient way to address the core problem.

Subnet expansion is generally less disruptive than creating new networks. Google Cloud allows expanding subnets without significant downtime.

**Further Research:**

**Google Cloud documentation on Expanding subnets:** https://cloud.google.com/vpc/docs/expand-subnet
**Google Cloud documentation on VPC networks:** https://cloud.google.com/vpc/docs/
**Google Cloud documentation on GKE IP addressing:** https://cloud.google.com/kubernetes-engine/docs/concepts/alias-ips

In summary, expanding the CIDR range of the subnet is the most straightforward and efficient way to resolve the node IP exhaustion issue for GKE clusters operating within the same subnet.

**Question: 150** CertyIQ

You have a batch workload that runs every night and uses a large number of virtual machines (VMs). It is fault-tolerant and can tolerate some of the VMs being terminated. The current cost of VMs is too high. What should you do?

A. Run a test using simulated maintenance events. If the test is successful, use preemptible N1 Standard VMs when running future jobs.

B. Run a test using simulated maintenance events. If the test is successful, use N1 Standard VMs when running future jobs.

C. Run a test using a managed instance group. If the test is successful, use N1 Standard VMs in the managed instance group when running future jobs.

D. Run a test using N1 standard VMs instead of N2. If the test is successful, use N1 Standard VMs when running future jobs.

**Answer: A**

**Explanation:**

The correct answer is A. Preemptible VMs offer significantly lower costs compared to regular VMs (like N1 standard) in exchange for the possibility of being terminated with short notice (24 hours max). Since the batch workload is fault-tolerant, it can withstand these interruptions. Before deploying preemptible VMs widely, it's crucial to simulate maintenance events to verify the application's ability to handle disruptions gracefully. This ensures that the cost savings from preemptible VMs don't compromise the job completion. Option B is incorrect as standard VMs do not offer cost savings. Option C introduces managed instance groups, which offer benefits like scaling, but don't inherently reduce cost and do not address the use of preemptible vs. regular instances; therefore, it's not the most direct approach to cost optimization in this scenario. Option D compares different VM series, but the core issue is about the cost model (preemptible vs. standard) not the specific series. Therefore, the initial simulation of the fault tolerance with preemptible instances is vital. The recommended approach focuses on minimizing cost through using preemptible VMs and validating the ability of the application to withstand maintenance events before implementation.

**Further Research:**

**Preemptible VMs:** https://cloud.google.com/compute/docs/instances/preemptible
**Instance Types:** https://cloud.google.com/compute/docs/machine-types

**Question: 151** CertyIQ

You are working with a user to set up an application in a new VPC behind a firewall. The user is concerned about data egress. You want to configure the fewest open egress ports. What should you do?

A. Set up a low-priority (65534) rule that blocks all egress and a high-priority rule (1000) that allows only the appropriate ports.

B. Set up a high-priority (1000) rule that pairs both ingress and egress ports.

C. Set up a high-priority (1000) rule that blocks all egress and a low-priority (65534) rule that allows only the appropriate ports.

D. Set up a high-priority (1000) rule to allow the appropriate ports.

**Answer: A**

**Explanation:**

The correct approach to minimize egress ports while maximizing security is to implement a "default deny" strategy. This means that by default, all egress traffic is blocked, and then specific exceptions are made to allow only necessary communication. Option A achieves this by establishing a low-priority (65534) rule that blocks all outgoing traffic. Then, a higher priority rule (1000), which is evaluated first due to its lower number, allows only the required egress ports. This ensures that no unexpected outbound connections can occur, minimizing the risk of data exfiltration. Network firewall rules are processed in order of priority, with lower numbers taking precedence. Option B's focus on pairing ingress and egress ports is not suitable for minimizing egress. Option C is incorrect as higher priority rules are evaluated first, so a higher priority block would supersede the lower priority allow rule. Option D is insufficient as it only allows specified ports without blocking others, meaning that other unnecessary outbound connections might still exist. Option A, therefore, aligns with the principle of least privilege and offers the best solution for secure egress.

Here are some authoritative resources that elaborate on these concepts:

**Google Cloud VPC Firewall rules overview:** https://cloud.google.com/vpc/docs/firewalls - Explains how firewall rules are processed in VPC networks.
**Principle of Least Privilege:** https://en.wikipedia.org/wiki/Principle_of_least_privilege - Describes the security concept of granting the minimum necessary access.
**Network Security Best Practices on Google Cloud:** https://cloud.google.com/security/docs/security-best-practices - Offers a wide overview of Google's security best practices.

**Question: 152** CertyIQ

Your company runs its Linux workloads on Compute Engine instances. Your company will be working with a new operations partner that does not use Google
Accounts. You need to grant access to the instances to your operations partner so they can maintain the installed tooling. What should you do?

A. Enable Cloud IAP for the Compute Engine instances, and add the operations partner as a Cloud IAP Tunnel User.

B. Tag all the instances with the same network tag. Create a firewall rule in the VPC to grant TCP access on port 22 for traffic from the operations partner to instances with the network tag.

C. Set up Cloud VPN between your Google Cloud VPC and the internal network of the operations partner.

D. Ask the operations partner to generate SSH key pairs, and add the public keys to the VM instances.

**Answer: A**

**Explanation:**

The correct answer is **A. Enable Cloud IAP for the Compute Engine instances, and add the operations partner as a Cloud IAP Tunnel User.**

Here's why: Cloud Identity-Aware Proxy (IAP) provides a secure way to access applications and resources hosted on Google Cloud Platform (GCP), like Compute Engine instances. It acts as a central authentication and authorization point, verifying users' identities before granting access. IAP integrates with Google Accounts (or Google Workspace accounts) but crucially, it also allows access to be granted to users not using Google Accounts by leveraging their identities within a separate identity provider as long as that provider can provide authentication. By adding the operations partner as a Cloud IAP Tunnel User, you are enabling them to connect to the instances securely through the IAP tunnel, using their own credentials without needing a Google account. The tunnel provides secure connectivity.

Option B is incorrect because opening port 22 directly to the internet via firewall rules is a security risk. While it would grant access, it bypasses proper authentication mechanisms and potentially exposes the instances to brute-force attacks. Option C, setting up Cloud VPN, is overkill and creates unnecessary complexity and cost since the operations partner does not need general access to the whole VPC, and is not required in order to give access. Option D, requesting SSH keys, is also cumbersome, requiring manual key management and distribution, and is difficult to centrally revoke access when needed. It's also a common security risk. IAP offers a much more managed and secure way to control access.

**Authoritative links:**

**Cloud IAP Overview:** https://cloud.google.com/iap/docs/overview
**Using IAP for TCP forwarding:** https://cloud.google.com/iap/docs/using-tcp-forwarding
**IAP Access Control:** https://cloud.google.com/iap/docs/access-control

**Question: 153** CertyIQ

You have created a code snippet that should be triggered whenever a new file is uploaded to a Cloud Storage

bucket. You want to deploy this code snippet. What should you do?

A. Use App Engine and configure Cloud Scheduler to trigger the application using Pub/Sub.
B. Use Cloud Functions and configure the bucket as a trigger resource.
C. Use Google Kubernetes Engine and configure a CronJob to trigger the application using Pub/Sub.
D. Use Dataflow as a batch job, and configure the bucket as a data source.

**Answer: B**

**Explanation:**

The correct answer is **B. Use Cloud Functions and configure the bucket as a trigger resource.**

Cloud Functions are Google Cloud's serverless execution environment for running event-driven code. They are ideal for scenarios like reacting to file uploads in Cloud Storage. Configuring the bucket as a trigger resource means the function automatically executes whenever a new object is created within that bucket. This eliminates the need for manual scheduling or polling.

Option A is incorrect because App Engine is a platform-as-a-service (PaaS) for web applications. While it can be triggered via Pub/Sub, it's not the best fit for this simple event-driven task. It would require more infrastructure setup and management. Using Cloud Scheduler to trigger via Pub/Sub adds unnecessary complexity compared to Cloud Functions. Option C is wrong because Google Kubernetes Engine (GKE) is a container orchestration service, which is overkill for this use case. Using CronJobs to trigger via Pub/Sub adds unnecessary complexity and overhead. Option D is incorrect because Dataflow is a batch and stream processing service designed for large datasets. It's not the appropriate tool for triggering code based on individual file uploads. The process of setting up a Dataflow pipeline with the bucket as a data source is unnecessarily complex for the described task.

Cloud Functions, therefore, provides a simpler, more efficient, and cost-effective solution for handling events triggered by changes in Cloud Storage buckets. It requires minimal configuration, scales automatically, and charges only when the function is executed, aligning perfectly with the use case.

**Further research:**

**Cloud Functions:** https://cloud.google.com/functions/docs
**Cloud Storage Triggers:** https://cloud.google.com/functions/docs/calling/storage
**App Engine:** https://cloud.google.com/appengine/docs
**Google Kubernetes Engine:** https://cloud.google.com/kubernetes-engine/docs
**Dataflow:** https://cloud.google.com/dataflow/docs

**Question: 154** CertyIQ

You have been asked to set up Object Lifecycle Management for objects stored in storage buckets. The objects are written once and accessed frequently for 30 days. After 30 days, the objects are not read again unless there is a special need. The objects should be kept for three years, and you need to minimize cost.
What should you do?

A. Set up a policy that uses Nearline storage for 30 days and then moves to Archive storage for three years.
B. Set up a policy that uses Standard storage for 30 days and then moves to Archive storage for three years.
C. Set up a policy that uses Nearline storage for 30 days, then moves the Coldline for one year, and then moves to Archive storage for two years.
D. Set up a policy that uses Standard storage for 30 days, then moves to Coldline for one year, and then moves to Archive storage for two years.

**Answer: B**

**Explanation:**

The correct answer is **B**. Here's why:

The scenario requires balancing cost optimization with data access patterns. The objects are frequently accessed for 30 days, then infrequently accessed for the remaining three years. Google Cloud Storage offers different storage classes optimized for varied use cases.

**Standard Storage:** Best for frequently accessed "hot" data. It has the highest storage cost but the lowest access latency. Since the objects are frequently accessed for the initial 30 days, Standard is a suitable choice.

**Nearline Storage:** Designed for data accessed less than once a month. While it's cheaper than Standard for storage, it incurs retrieval costs and has slightly higher access latency. It's not suitable for the first 30 days of frequent access.

**Coldline Storage:** Ideal for data accessed less than once a quarter. Storage costs are lower than Nearline, but retrieval costs are higher, along with more latency. This isn't a cost-effective choice for the infrequent access scenario after 30 days, and also adds an unnecessary transition to a slightly more expensive tier than archive.

**Archive Storage:** Meant for data accessed less than once a year. It has the lowest storage cost but the highest retrieval latency and costs. Given the long-term storage requirement with infrequent access after the initial 30 days, Archive is the best option for cost optimization.

Therefore, setting a lifecycle policy that transitions objects from **Standard storage for the first 30 days** (to accommodate frequent access) to **Archive storage for the remaining three years** (for long-term, rarely accessed data) is the most cost-effective solution. This approach minimizes storage costs while providing the required performance for the initial frequent access period. Options C and D are unnecessarily complex and introduce extra transition steps which will cost more. Option A is not suitable as the first 30 days will be costly for frequent access with Nearline.

**Authoritative Links:**

**Google Cloud Storage Storage Classes:** https://cloud.google.com/storage/docs/storage-classes
**Object Lifecycle Management:** https://cloud.google.com/storage/docs/lifecycle

**Question: 155** CertyIQ

You are storing sensitive information in a Cloud Storage bucket. For legal reasons, you need to be able to record all requests that read any of the stored data. You want to make sure you comply with these requirements. What should you do?

A. Enable the Identity Aware Proxy API on the project.
B. Scan the bucket using the Data Loss Prevention API.
C. Allow only a single Service Account access to read the data.
D. Enable Data Access audit logs for the Cloud Storage API.

**Answer: D**

**Explanation:**

The correct answer is **D. Enable Data Access audit logs for the Cloud Storage API.** This is because audit logs, specifically Data Access logs, record API calls that read, modify, or write data. In the context of Cloud Storage, this means logging all requests that retrieve data from your bucket, which directly addresses the requirement to track all read requests for legal compliance.

Option A, enabling the Identity Aware Proxy (IAP) API, is primarily used to manage access control for

applications and resources based on user identity. While it can enhance security, it doesn't inherently provide logging of data access. Option B, scanning the bucket using the Data Loss Prevention (DLP) API, focuses on identifying sensitive data within content, not on logging data access requests. Option C, allowing only a single Service Account access, limits the number of potential accessors, but it doesn't log the actual data reads, making it insufficient for compliance requirements.

Data Access audit logs capture granular details about API calls, including the user or service account that made the request, the timestamp, and the specific data accessed. These logs provide a detailed audit trail, essential for demonstrating compliance with data handling regulations. By enabling Data Access audit logs for the Cloud Storage API, you are ensuring comprehensive record-keeping of all data read requests, fulfilling the stated legal requirements.

For further information on Cloud Storage audit logs, refer to Google Cloud documentation:

Cloud Storage Audit Logs
Cloud Logging Audit Logs
Understanding audit logs

**Question: 156** CertyIQ

You are the team lead of a group of 10 developers. You provided each developer with an individual Google Cloud Project that they can use as their personal sandbox to experiment with different Google Cloud solutions. You want to be notified if any of the developers are spending above $500 per month on their sandbox environment. What should you do?

A. Create a single budget for all projects and configure budget alerts on this budget.

B. Create a separate billing account per sandbox project and enable BigQuery billing exports. Create a Data Studio dashboard to plot the spending per billing account.

C. Create a budget per project and configure budget alerts on all of these budgets.

D. Create a single billing account for all sandbox projects and enable BigQuery billing exports. Create a Data Studio dashboard to plot the spending per project.

**Answer: C**

**Explanation:**

The correct answer is C because it directly addresses the requirement of monitoring individual developer spending. Here's a detailed justification:

Option C suggests creating a budget for each project and configuring alerts on those budgets. This approach is ideal for granular cost control because each developer's spending is tracked separately. If any developer exceeds the $500 threshold, an alert will be triggered specifically for that project, ensuring timely notification. This method allows for targeted intervention, enabling you to directly address the overspending with the individual developer. Creating separate budgets avoids the situation where the total spending is monitored while masking individual overages, which would be a problem with option A.

Options A, B, and D are less suitable. Option A creates a single budget across all projects. This would only alert you if the total spending exceeds a limit, not if individual developers are overspending. Option B introduces the complexity of creating separate billing accounts which is not typically recommended for sandbox environments, plus it requires setting up BigQuery exports and dashboards which is over-engineering for the stated requirement. Option D similarly attempts to use a single billing account which can mask individual overspend, even with dashboarding, because the alert mechanism will not function as desired. Furthermore, option D also adds unnecessary overhead with BigQuery and Data Studio.

Therefore, option C offers the most straightforward, effective, and least complex solution by providing

granular budget control and alerts for each individual developer's sandbox project, aligning precisely with the specified need to be notified about each developer exceeding a $500 monthly spend.

**Authoritative Links:**

**Creating budgets:** https://cloud.google.com/billing/docs/how-to/budgets
**Budget alerts:** https://cloud.google.com/billing/docs/how-to/budget-alerts
**Understanding billing accounts:** https://cloud.google.com/billing/docs/concepts/billing-account

**Question: 157** CertyIQ

You are deploying a production application on Compute Engine. You want to prevent anyone from accidentally destroying the instance by clicking the wrong button. What should you do?

A. Disable the flag Delete boot disk when instance is deleted.
B. Enable delete protection on the instance.
C. Disable Automatic restart on the instance.
D. Enable Preemptibility on the instance.

**Answer: B**

**Explanation:**

The correct answer is **B. Enable delete protection on the instance.**

Here's why: Delete protection is a specific feature offered by Compute Engine designed to prevent accidental deletion of critical resources. When enabled, it adds an extra layer of security, requiring users to explicitly disable the protection before deleting the instance. This safeguard significantly reduces the risk of unintended data loss or service disruption due to accidental clicks or misoperations.

Option A, "Disable the flag Delete boot disk when instance is deleted," only addresses the boot disk's deletion behavior and doesn't prevent the instance itself from being deleted. Option C, "Disable Automatic restart on the instance," focuses on how the instance behaves after a failure, not preventing deletion. Option D, "Enable Preemptibility on the instance," makes the instance more vulnerable to being terminated, the opposite of what is desired for accidental deletion prevention.

Delete protection is a best practice for managing production instances, especially those running critical services. It's a simple yet effective way to enforce a more careful approach to instance management. This security feature aligns with principles of operational excellence and risk mitigation by safeguarding important compute resources from human error.

Google Cloud documentation clearly describes how to enable and manage delete protection through the console or command-line tools. Using this feature reinforces an environment where accidental deletions require intentional action and provides a crucial safety net for production deployments.

**Authoritative Links:**

**Google Cloud Documentation on Instance Delete Protection:**
https://cloud.google.com/compute/docs/instances/prevent-instance-deletion

**Question: 158** CertyIQ

Your company uses a large number of Google Cloud services centralized in a single project. All teams have specific projects for testing and development. The

DevOps team needs access to all of the production services in order to perform their job. You want to prevent Google Cloud product changes from broadening their permissions in the future. You want to follow Google-recommended practices. What should you do?

A. Grant all members of the DevOps team the role of Project Editor on the organization level.

B. Grant all members of the DevOps team the role of Project Editor on the production project.

C. Create a custom role that combines the required permissions. Grant the DevOps team the custom role on the production project.

D. Create a custom role that combines the required permissions. Grant the DevOps team the custom role on the organization level.

**Answer: C**

**Explanation:**

Option C is the best choice because it adheres to the principle of least privilege and Google's recommended practices for managing IAM (Identity and Access Management). Granting the DevOps team a custom role with only the necessary permissions for their tasks on the production project limits their potential impact if their accounts are compromised or if they accidentally make unwanted changes. Using a custom role prevents Google Cloud product updates from inadvertently granting broader permissions to the DevOps team than intended, a concern explicitly raised in the problem. Options A and B violate the principle of least privilege by granting the overly permissive "Project Editor" role. This role grants more access than the DevOps team requires, potentially leading to accidental or malicious misuse. Project Editor grants permission to manage all Google Cloud resources within a project. Option D, while using a custom role, incorrectly assigns it at the organization level. This grants the DevOps team the custom role across all projects, which contradicts the need to restrict access to only production resources. This also violates the principle of least privilege by giving access to resources which should be restricted for security and best practice reasons. Custom roles allow for granular control over permissions, ensuring that each team member only has the access required to perform their specific tasks. Best practices advocate for using custom roles instead of broad predefined roles for enhanced security and reduced blast radius. This approach offers a secure and tailored permission strategy for the specific requirements of the DevOps team.

Further Research:

**Google Cloud IAM Documentation:** https://cloud.google.com/iam/docs
**IAM Best Practices:** https://cloud.google.com/iam/docs/best-practices
**Creating and Managing Custom Roles:** https://cloud.google.com/iam/docs/understanding-custom-roles

**Question: 159** CertyIQ

You are building an application that processes data files uploaded from thousands of suppliers. Your primary goals for the application are data security and the expiration of aged data. You need to design the application to:
* Restrict access so that suppliers can access only their own data.
* Give suppliers write access to data only for 30 minutes.
* Delete data that is over 45 days old.
You have a very short development cycle, and you need to make sure that the application requires minimal maintenance. Which two strategies should you use?
(Choose two.)

A. Build a lifecycle policy to delete Cloud Storage objects after 45 days.

B. Use signed URLs to allow suppliers limited time access to store their objects.

C. Set up an SFTP server for your application, and create a separate user for each supplier.

D. Build a Cloud function that triggers a timer of 45 days to delete objects that have expired.

E. Develop a script that loops through all Cloud Storage buckets and deletes any buckets that are older than 45 days.

**Answer: AB**

**Explanation:**

The correct strategies are A and B because they directly address the requirements with minimal maintenance overhead. Option A, using Cloud Storage lifecycle policies, allows for automated deletion of objects older than 45 days. This is a built-in feature that requires no coding or custom maintenance. It leverages object metadata (specifically creation date) to trigger deletion actions, fulfilling the data expiration requirement. [https://cloud.google.com/storage/docs/lifecycle] Option B, utilizing signed URLs, provides time-limited access for suppliers to upload their data. A signed URL grants temporary write access without requiring long-term credentials, thereby satisfying the 30-minute write access window. This method enhances security by restricting access to only the necessary window. [https://cloud.google.com/storage/docs/access-control/signed-urls]

Option C, setting up an SFTP server, adds unnecessary complexity, requiring server maintenance, user management, and potentially firewall configuration. It doesn't inherently handle time-limited access and isn't well-suited to a short development cycle. Option D, building a Cloud Function, adds custom code and maintenance overhead. While functional, it's less efficient than using built-in lifecycle policies. Finally, option E, deleting buckets, is incorrect; we need to delete individual objects, not entire buckets, and it is inefficient to loop through all buckets. It could lead to data loss and service disruptions by deleting unrelated data if not carefully handled. The chosen answers, A and B, optimally balance security, functionality, and ease of implementation, making them the best solutions. They leverage existing Google Cloud functionalities, minimizing custom development and maintenance.

**Question: 160** CertyIQ

Your company wants to standardize the creation and management of multiple Google Cloud resources using Infrastructure as Code. You want to minimize the amount of repetitive code needed to manage the environment. What should you do?

A. Develop templates for the environment using Cloud Deployment Manager.
B. Use curl in a terminal to send a REST request to the relevant Google API for each individual resource.
C. Use the Cloud Console interface to provision and manage all related resources.
D. Create a bash script that contains all requirement steps as gcloud commands.

**Answer: A**

**Explanation:**

The correct answer is **A. Develop templates for the environment using Cloud Deployment Manager.** This choice aligns with the principles of Infrastructure as Code (IaC), which aims to manage and provision infrastructure through machine-readable definition files, rather than manual configuration. Cloud Deployment Manager (CDM) is Google Cloud's native infrastructure deployment service that allows users to define their infrastructure as templates using YAML or Python. These templates can then be reused to deploy consistent environments. This approach significantly reduces repetitive code and potential human errors inherent in manual provisioning.

Option B, using curl to send REST requests, while possible, requires crafting specific API calls for each resource, making it cumbersome and error-prone for managing multiple resources. This approach lacks the structure and repeatability of IaC. Option C, using the Cloud Console, involves manual point-and-click actions that don't scale well, leading to inconsistencies and difficulties in tracking changes. Option D, a bash script, is better than manual configuration, but it lacks the robust templating, state management, and dependency resolution that a dedicated tool like CDM provides. Furthermore, bash scripts often become complex and hard

to maintain.

CDM, on the other hand, provides features like parameterization, allowing for customization of deployments, and built-in dependency management, ensuring that resources are created in the correct order. By employing CDM, users can effectively version, share, and consistently reproduce infrastructure deployments. Utilizing templates allows for the creation of reusable modules, further reducing the amount of repetitive code and simplifying management. Ultimately, the template-driven approach offered by Cloud Deployment Manager allows for a scalable, reliable, and maintainable infrastructure management strategy.

For further research, please refer to the official Google Cloud documentation on Cloud Deployment Manager: https://cloud.google.com/deployment-manager/docs and on Infrastructure as Code: https://cloud.google.com/solutions/infrastructure-as-code.

**Question: 161** CertyIQ

You are performing a monthly security check of your Google Cloud environment and want to know who has access to view data stored in your Google Cloud
Project. What should you?

A. Enable Audit Logs for all APIs that are related to data storage.
B. Review the IAM permissions for any role that allows for data access.
C. Review the Identity-Aware Proxy settings for each resource.
D. Create a Data Loss Prevention job.

**Answer: B**

**Explanation:**

The correct answer is **B. Review the IAM permissions for any role that allows for data access.** Here's why:

Identity and Access Management (IAM) is the cornerstone of controlling who has access to your Google Cloud resources, including data. IAM uses roles, which are collections of permissions, to grant specific actions users can perform. To identify who has access to view data, you must examine these roles and the users or groups assigned to them.

Option A, enabling Audit Logs, while useful for tracking what actions have been taken, doesn't directly reveal who has the initial access. Audit logs record actions after they happen; they do not show the existing authorization structure. Option C, reviewing Identity-Aware Proxy (IAP) settings, concerns access to applications through HTTP(S) and does not govern access to data within a project's storage services. Option D, creating a Data Loss Prevention (DLP) job, is for identifying and protecting sensitive data, not for determining who has access to it.

Therefore, reviewing IAM roles that include permissions like storage.objects.get, storage.objects.list, or equivalent for databases or other data services is the most direct and accurate approach. This involves checking the roles granted at project, folder, or resource levels and examining the principals (users, groups, service accounts) that are members of those roles. You're looking for any role with permissions that allow reading data.

In summary, while audit logs are valuable for monitoring activity, IAM is the system that enforces access. By scrutinizing the IAM roles and bindings, you can definitively ascertain who has permission to view data.

**Authoritative Links for Further Research:**

**IAM Overview:** https://cloud.google.com/iam/docs/overview
**IAM Roles:** https://cloud.google.com/iam/docs/understanding-roles

**IAM Permissions:** https://cloud.google.com/iam/docs/permissions-overview

**Question: 162** CertyIQ

Your company has embraced a hybrid cloud strategy where some of the applications are deployed on Google Cloud. A Virtual Private Network (VPN) tunnel connects your Virtual Private Cloud (VPC) in Google Cloud with your company's on-premises network. Multiple applications in Google Cloud need to connect to an on-premises database server, and you want to avoid having to change the IP configuration in all of your applications when the IP of the database changes.
What should you do?

A. Configure Cloud NAT for all subnets of your VPC to be used when egressing from the VM instances.
B. Create a private zone on Cloud DNS, and configure the applications with the DNS name.
C. Configure the IP of the database as custom metadata for each instance, and query the metadata server.
D. Query the Compute Engine internal DNS from the applications to retrieve the IP of the database.

**Answer: B**

**Explanation:**

The correct answer is **B. Create a private zone on Cloud DNS, and configure the applications with the DNS name.** Here's why:

Option B leverages the power of Domain Name System (DNS) for dynamic resolution of the database server's IP address. By creating a private zone within Cloud DNS, you can define a hostname (e.g., database.internal) that maps to the current IP address of the on-premises database. Your applications then simply need to query this hostname, and Cloud DNS will provide the correct IP. If the database server's IP changes, you only need to update the DNS record, eliminating the need to modify each application's configuration. This decouples the applications from the underlying infrastructure, promoting flexibility and maintainability.

Option A, Cloud NAT, addresses egress traffic from Google Cloud to the internet, not to an on-premises server via a VPN. Option C, metadata server, could be used but requires individual updates for each instance. Option D, Compute Engine's internal DNS, is for instances within your VPC, not on-premises resources. Cloud DNS provides a centralized, easily updated resolution mechanism.

Cloud DNS is a scalable and reliable DNS service offered by Google Cloud. Private zones allow you to manage DNS records for internal use within your network without exposing them to the public internet. It facilitates consistent name resolution and makes the database service easier to use by utilizing a symbolic name.

**Authoritative Links:**

**Google Cloud DNS Documentation:** https://cloud.google.com/dns/docs
**Private DNS Zones:** https://cloud.google.com/dns/docs/zones/private-zones
**Hybrid Networking with Google Cloud:** https://cloud.google.com/solutions/hybrid-networking

**Question: 163** CertyIQ

You have developed a containerized web application that will serve internal colleagues during business hours. You want to ensure that no costs are incurred outside of the hours the application is used. You have just created a new Google Cloud project and want to deploy the application. What should you do?

A. Deploy the container on Cloud Run for Anthos, and set the minimum number of instances to zero.
B. Deploy the container on Cloud Run (fully managed), and set the minimum number of instances to zero.
C. Deploy the container on App Engine flexible environment with autoscaling, and set the value min_instances

to zero in the app.yaml.

D. Deploy the container on App Engine flexible environment with manual scaling, and set the value instances to zero in the app.yaml.

**Answer: B**

**Explanation:**

Option B, deploying the container on Cloud Run (fully managed) with a minimum instance count of zero, is the most suitable approach for the scenario. Cloud Run is a serverless compute platform designed for containerized applications. Setting the minimum instances to zero ensures that no resources are consumed and no costs are incurred when the application is not actively handling requests, effectively scaling down to zero. This aligns perfectly with the requirement to avoid costs outside of business hours. Cloud Run's fully managed nature handles infrastructure concerns, simplifying deployment and maintenance. Options A, C, and D are less ideal. Cloud Run for Anthos (A) requires a Kubernetes cluster and is an overkill for this simple use case and might increase costs. App Engine flexible environment (C & D) while autoscalable may have warm-up times for new instances leading to increased latency for initial users and manual scaling (D) would require explicit manual intervention to scale instances up and down and doesn't align with the requirement to auto-scale to zero. With Cloud Run, the scaling to zero happens automatically and quickly, making it the most cost-effective and efficient option for the given constraints. Cloud Run fully managed offers pay-per-use billing, maximizing cost-efficiency. This makes option B the optimal choice for the described needs.

Cloud Run DocumentationCloud Run Scaling

**Question: 164** CertyIQ

You have experimented with Google Cloud using your own credit card and expensed the costs to your company. Your company wants to streamline the billing process and charge the costs of your projects to their monthly invoice. What should you do?

A. Grant the financial team the IAM role of Billing Account User on the billing account linked to your credit card.

B. Set up BigQuery billing export and grant your financial department IAM access to query the data.

C. Create a ticket with Google Billing Support to ask them to send the invoice to your company.

D. Change the billing account of your projects to the billing account of your company.

**Answer: D**

**Explanation:**

The correct answer is **D. Change the billing account of your projects to the billing account of your company.** Here's why:

The core issue is transferring billing responsibility from your personal credit card to your company's centralized billing. Option D directly addresses this by reassigning your projects to use the company's billing account. This ensures all costs associated with those projects are billed to the company's monthly invoice.

Option A is incorrect because granting the financial team "Billing Account User" on your personal billing account does not shift billing responsibility. They would only see your expenses, not direct them to the company's invoice. Option B, while useful for cost analysis, doesn't change billing responsibility either. BigQuery export just provides cost data, not a payment mechanism. Option C is also unsuitable. Google Billing Support cannot change billing assignments on your behalf; it requires your own action within the GCP console or API. The correct approach involves modifying project settings to reflect your organization's billing account. This process involves going to the project's "Billing" settings and selecting the appropriate billing account provided by your company. This ensures a streamlined, centralized billing approach, which is the primary goal

of the request.

Essentially, changing the billing account is a fundamental step for organizational cost management and alignment with company procedures.

Further research:

**Google Cloud Billing Documentation:** https://cloud.google.com/billing/docs - provides comprehensive information about Google Cloud Billing, including managing billing accounts, changing billing assignments, and cost management.
**Changing a project's billing account:** https://cloud.google.com/billing/docs/how-to/modify-project - Directly explains the steps on changing project billing accounts.
**Cloud Billing overview:** https://cloud.google.com/billing/docs/concepts - Offers foundational understanding of key concepts.

**Question: 165** CertyIQ

You are running a data warehouse on BigQuery. A partner company is offering a recommendation engine based on the data in your data warehouse. The partner company is also running their application on Google Cloud. They manage the resources in their own project, but they need access to the BigQuery dataset in your project. You want to provide the partner company with access to the dataset. What should you do?

A. Create a Service Account in your own project, and grant this Service Account access to BigQuery in your project.

B. Create a Service Account in your own project, and ask the partner to grant this Service Account access to BigQuery in their project.

C. Ask the partner to create a Service Account in their project, and have them give the Service Account access to BigQuery in their project.

D. Ask the partner to create a Service Account in their project, and grant their Service Account access to the BigQuery dataset in your project.

**Answer: D**

**Explanation:**

The correct approach, option D, aligns with Google Cloud's principle of least privilege and emphasizes granting access to resources where they reside. The partner company needs access to your BigQuery dataset. Therefore, their service account should be granted permissions within your project, specifically on the dataset in question. Option A is incorrect because it creates a service account in your project, which the partner would need to manage. This is less secure and less clear responsibility. Option B is incorrect because it asks the partner to grant permissions in their project which is not logical because they need access to your resource. Option C reverses the permissions logic, asking them to give access in their project which does not grant access to your dataset. Instead of granting broad access, a service account created and owned by the partner in their project is used. The partner then grants that service account permission to the specific BigQuery dataset in your project. This adheres to the best practice of using service accounts to enable cross-project access, maintaining clear ownership and accountability. The service account becomes the identity the partner's application uses when interacting with your dataset. This is the most secure and manageable solution in a cloud environment. Using IAM and service account ensures granular permissions and easy auditing.

**Authoritative Links for further research:**

**Google Cloud IAM Overview:** https://cloud.google.com/iam/docs/overview
**Understanding Service Accounts:** https://cloud.google.com/iam/docs/service-accounts
**Granting Access to BigQuery datasets:** https://cloud.google.com/bigquery/docs/share-access-datasets

**Best practices for using IAM:** https://cloud.google.com/iam/docs/best-practices

**Question: 166** CertyIQ

Your web application has been running successfully on Cloud Run for Anthos. You want to evaluate an updated version of the application with a specific percentage of your production users (canary deployment). What should you do?

A. Create a new service with the new version of the application. Split traffic between this version and the version that is currently running.

B. Create a new revision with the new version of the application. Split traffic between this version and the version that is currently running.

C. Create a new service with the new version of the application. Add an HTTP Load Balancer in front of both services.

D. Create a new revision with the new version of the application. Add an HTTP Load Balancer in front of both revisions.

**Answer: B**

**Explanation:**

The correct answer is B, creating a new revision with the updated application and splitting traffic. Cloud Run for Anthos natively supports traffic splitting between revisions of the same service, a key feature for canary deployments. A revision in Cloud Run represents an immutable snapshot of a container image and its configuration. By creating a new revision with the updated application, you maintain the existing production environment while introducing the new version. This allows for controlled testing and minimizes risk. Traffic splitting is configured directly within the Cloud Run service, allowing you to gradually shift traffic towards the new revision based on a percentage, enabling you to monitor its performance and behavior with a small portion of real users. Option A is incorrect because creating a new service would mean managing two distinct services, complicating routing and traffic management for a canary deployment. Options C and D are also incorrect because while HTTP Load Balancers can distribute traffic, they are not needed for a canary deployment within Cloud Run, since traffic splitting is built-in. Using a load balancer introduces unnecessary complexity. Cloud Run's traffic management features are designed for rolling out updates, including canary deployments, efficiently. Traffic splitting is a core feature of Cloud Run for managing updates safely and effectively.

Here are some authoritative links for further research:

**Google Cloud Run documentation on traffic splitting:** https://cloud.google.com/run/docs/rollouts-rollbacks-traffic#traffic-splitting
**Google Cloud Run documentation on Revisions:** https://cloud.google.com/run/docs/revisions
**Google Cloud Run Concepts:** https://cloud.google.com/run/docs/concepts

**Question: 167** CertyIQ

Your company developed a mobile game that is deployed on Google Cloud. Gamers are connecting to the game with their personal phones over the Internet. The game sends UDP packets to update the servers about the gamers' actions while they are playing in multiplayer mode. Your game backend can scale over multiple virtual machines (VMs), and you want to expose the VMs over a single IP address. What should you do?

A. Configure an SSL Proxy load balancer in front of the application servers.

B. Configure an Internal UDP load balancer in front of the application servers.

C. Configure an External HTTP(s) load balancer in front of the application servers.

D. Configure an External Network load balancer in front of the application servers.

**Answer: D**

**Explanation:**

The correct answer is **D. Configure an External Network load balancer in front of the application servers.** Here's why:

The scenario requires load balancing UDP traffic from gamers to multiple backend VMs. An External Network load balancer is specifically designed to handle this type of traffic. It operates at the network layer (Layer 4) and can forward UDP packets based on IP address and port, without needing to understand the contents of the packets. This is crucial because the game uses UDP for real-time updates, which prioritizes speed and low latency over connection reliability.

Option A, the SSL Proxy load balancer, is for TCP-based traffic and specifically handles SSL/TLS termination. It's unsuitable for UDP. Similarly, option C, the External HTTP(s) load balancer, is designed for HTTP/HTTPS traffic and isn't applicable to UDP. Option B, the Internal UDP load balancer, is used for internal load balancing within a VPC, not for internet-facing traffic from gamers.

Therefore, an External Network load balancer is the ideal choice for exposing the game's backend VMs to gamers via a single IP address while correctly handling the UDP protocol. It provides high performance, low latency, and operates at the necessary network layer to achieve the objective. It can also scale automatically based on the load.

Authoritative Links for Further Research:

1. **Google Cloud Network Load Balancer Overview:** https://cloud.google.com/load-balancing/docs/network
2. **Google Cloud Load Balancing Documentation:** https://cloud.google.com/load-balancing/docs
3. **Choosing a load balancer:** https://cloud.google.com/load-balancing/docs/choosing-load-balancer

**Question: 168** CertyIQ

You are working for a hospital that stores its medical images in an on-premises data room. The hospital wants to use Cloud Storage for archival storage of these images. The hospital wants an automated process to upload any new medical images to Cloud Storage. You need to design and implement a solution. What should you do?

A. Create a Pub/Sub topic, and enable a Cloud Storage trigger for the Pub/Sub topic. Create an application that sends all medical images to the Pub/Sub topic.

B. Deploy a Dataflow job from the batch template, Datastore to Cloud Storage. Schedule the batch job on the desired interval.

C. Create a script that uses the gsutil command line interface to synchronize the on-premises storage with Cloud Storage. Schedule the script as a cron job.

D. In the Cloud Console, go to Cloud Storage. Upload the relevant images to the appropriate bucket.

**Answer: C**

**Explanation:**

The most suitable approach is to use gsutil for synchronizing on-premises storage with Cloud Storage, scheduled as a cron job (option C). Here's why:

**Automation:** The requirement is for an automated process. Using gsutil within a cron job allows for regular, hands-off uploads, eliminating manual intervention.
**Synchronization:** gsutil rsync command efficiently mirrors the source directory to the destination Cloud

Storage bucket. Only new or modified files are uploaded, saving time and bandwidth. This fulfills the need to upload new images.
**Simplicity and Efficiency:** gsutil is a command-line utility optimized for interacting with Cloud Storage, making it a straightforward solution for this task.
**On-Premises to Cloud Storage:** The scenario specifically involves moving data from an on-premises environment to Cloud Storage, a key capability of gsutil.
**Cost-Effective:** gsutil is a free utility, making this a cost-effective approach.

Option A is less appropriate because it involves setting up Pub/Sub, which adds unnecessary complexity for a basic file transfer use case. Pub/Sub is typically used for asynchronous message processing, not direct file transfers. Option B, using Dataflow, is overly complicated for a simple synchronization task. Dataflow is designed for large-scale data processing and transformation, not file transfers. Option D is manual, requiring consistent human interaction, which goes against the automated requirement.

**Authoritative Links:**

**gsutil documentation:** https://cloud.google.com/storage/docs/gsutil
**gsutil rsync command:** https://cloud.google.com/storage/docs/gsutil/commands/rsync
**Cron Job documentation (for Linux/Unix systems):** https://en.wikipedia.org/wiki/Cron

**Question: 169** CertyIQ

Your auditor wants to view your organization's use of data in Google Cloud. The auditor is most interested in auditing who accessed data in Cloud Storage buckets. You need to help the auditor access the data they need. What should you do?

A. Turn on Data Access Logs for the buckets they want to audit, and then build a query in the log viewer that filters on Cloud Storage.
B. Assign the appropriate permissions, and then create a Data Studio report on Admin Activity Audit Logs.
C. Assign the appropriate permissions, and then use Cloud Monitoring to review metrics.
D. Use the export logs API to provide the Admin Activity Audit Logs in the format they want.

**Answer: A**

**Explanation:**

The correct answer is A because it directly addresses the auditor's requirement of tracking data access within Cloud Storage buckets. Data Access Logs specifically record when users or service accounts read or modify data, which is exactly what the auditor wants to examine. Enabling these logs at the bucket level is necessary to capture the relevant events. The logs are then accessible through the log viewer, allowing for querying and filtering based on Cloud Storage to isolate the relevant data access events. Option B is incorrect because Admin Activity Audit Logs focus on changes made to the configuration of your Google Cloud project, not data access events within Cloud Storage. Option C is also incorrect; Cloud Monitoring primarily focuses on performance metrics and resource utilization, not data access auditing. Option D, using the export logs API, is a valid method for getting the audit logs, but it's an additional step. Using the log viewer with a query is the quickest way for the auditor to start examining data access logs. The recommended approach uses the UI for the auditor and addresses the specific audit requirement more directly.

Authoritative links for further research:

**Google Cloud Logging:** https://cloud.google.com/logging/docs/audit/
**Data Access Audit Logs:** https://cloud.google.com/logging/docs/audit/data-access
**Cloud Storage Audit Logging:** https://cloud.google.com/storage/docs/audit-logging

**Question: 170** CertyIQ

You received a JSON file that contained a private key of a Service Account in order to get access to several resources in a Google Cloud project. You downloaded and installed the Cloud SDK and want to use this private key for authentication and authorization when performing gcloud commands. What should you do?

A. Use the command gcloud auth login and point it to the private key.

B. Use the command gcloud auth activate-service-account and point it to the private key.

C. Place the private key file in the installation directory of the Cloud SDK and rename it to credentials.json.

D. Place the private key file in your home directory and rename it to GOOGLE_APPLICATION_CREDENTIALS.

**Answer: B**

**Explanation:**

The correct answer is **B. Use the command gcloud auth activate-service-account and point it to the private key.**

Here's a detailed justification:

Service accounts in Google Cloud Platform (GCP) are non-human accounts used by applications or services to interact with GCP resources. They are authenticated using private keys. When you obtain a JSON file containing a service account's private key, you're essentially provided with the necessary credentials to impersonate that account. The gcloud command-line tool, part of the Cloud SDK, requires you to explicitly declare these credentials for any command that needs them. Option A, gcloud auth login, is designed for authenticating a user account, not a service account. Options C and D incorrectly suggest modifying the Cloud SDK installation or relying on environment variables, which are not the standard practice.

gcloud auth activate-service-account is the designated command to load and activate credentials from a service account key file. It takes the path to the JSON key file as an argument. This command essentially configures the gcloud tool to use the service account's permissions for subsequent commands. Once activated, the CLI will perform actions as if they were initiated by the service account. This is crucial for automating tasks, building CI/CD pipelines, or running services with specific access rights. This approach is secure, maintains proper security practices, and aligns with Google's recommended procedures for using service accounts. Improper usage of credentials can lead to unauthorized access or security breaches. Therefore, adhering to the correct method is essential for maintaining control over your GCP resources. The activated service account remains the active credentials until a different account or user is activated, or the current session closes.

**Authoritative Links for Further Research:**

**gcloud auth activate-service-account command:**
https://cloud.google.com/sdk/gcloud/reference/auth/activate-service-account
**Understanding Service Accounts:** https://cloud.google.com/iam/docs/service-accounts
**Authentication Overview:** https://cloud.google.com/docs/authentication/

**Question: 171** CertyIQ

You are working with a Cloud SQL MySQL database at your company. You need to retain a month-end copy of the database for three years for audit purposes.
What should you do?

A. Set up an export job for the first of the month. Write the export file to an Archive class Cloud Storage bucket.

B. Save the automatic first-of-the-month backup for three years. Store the backup file in an Archive class Cloud Storage bucket.

C. Set up an on-demand backup for the first of the month. Write the backup to an Archive class Cloud Storage bucket.

D. Convert the automatic first-of-the-month backup to an export file. Write the export file to a Coldline class Cloud Storage bucket.

**Answer: A**

**Explanation:**

Option A is the most appropriate solution for long-term archival of a month-end Cloud SQL MySQL database copy due to its cost-effectiveness and flexibility. Setting up an export job allows you to create a point-in-time consistent copy of the database in a portable format, such as SQL dump or CSV. This exported data is independent of the database instance, making it resilient to instance-level failures or changes. Writing the export file to an Archive class Cloud Storage bucket is the best strategy for long-term storage because this class is designed for infrequently accessed data, offering significant cost savings compared to other storage classes. Backups, on the other hand, are tightly coupled to the database instance and cannot be easily exported or independently managed for archival. While automatic backups can be retained, they are not ideal for long-term, cost-effective archival and may be removed to save costs if they are kept as automated backups. Using on-demand backups for this purpose (Option C) incurs higher storage costs than exporting to Archive storage, and backup files are not readily portable. Converting automatic backups to export files (Option D) adds unnecessary complexity as direct export is simpler and more efficient. Archiving to Coldline storage in Option D is less optimal than using Archive storage because Archive offers lower storage costs for infrequently accessed data while being suitable for the 3 year requirement. Option A provides the most cost-effective and flexible method of achieving the stated requirement of retaining the monthly database copy for audit purposes.

**Relevant Links:**

**Cloud SQL Exports:** https://cloud.google.com/sql/docs/mysql/export
**Cloud Storage Storage Classes:** https://cloud.google.com/storage/docs/storage-classes
**Cloud SQL Backups:** https://cloud.google.com/sql/docs/mysql/backup-recovery/backups

**Question: 172** CertyIQ

You are monitoring an application and receive user feedback that a specific error is spiking. You notice that the error is caused by a Service Account having insufficient permissions. You are able to solve the problem but want to be notified if the problem recurs. What should you do?

A. In the Log Viewer, filter the logs on severity 'Error' and the name of the Service Account.

B. Create a sink to BigQuery to export all the logs. Create a Data Studio dashboard on the exported logs.

C. Create a custom log-based metric for the specific error to be used in an Alerting Policy.

D. Grant Project Owner access to the Service Account.

**Answer: C**

**Explanation:**

The correct answer is **C. Create a custom log-based metric for the specific error to be used in an Alerting Policy.**

Here's why:

Option C directly addresses the need for proactive notification when the specific error recurs. By creating a custom log-based metric, you define a specific pattern within your logs that corresponds to the error caused by the service account's insufficient permissions. This metric acts as a counter, increasing each time the error

occurs. You can then create an alerting policy that triggers a notification (e.g., email, Slack message, PagerDuty alert) when the metric surpasses a defined threshold, enabling you to react swiftly to recurrence. This solution is targeted and efficient.

Option A, while helpful for troubleshooting, does not provide proactive alerting. Filtering logs manually is reactive and doesn't inform you automatically of the issue's reappearance. Option B is overly complex and resource-intensive for this scenario. Exporting all logs to BigQuery and creating dashboards is valuable for long-term analysis but not necessary for simple error recurrence monitoring, also requiring more maintenance and cost. Option D is a security risk and not best practice. Granting Project Owner access to a service account is excessive and should be avoided. It would give the account far more permissions than needed and could open a large security hole.

Log-based metrics are a fundamental component of monitoring in Google Cloud Platform (GCP). They allow you to extract insights from log data and act upon them, making them ideal for this use case.

**Relevant GCP Concepts:**

**Log-Based Metrics:** Enables you to create metrics from log data patterns and use them for alerting and monitoring. https://cloud.google.com/logging/docs/metrics/
**Alerting Policies:** Configure notifications based on metrics, enabling proactive monitoring of systems. https://cloud.google.com/monitoring/alerts
**Service Accounts:** Non-user accounts used by applications to access resources with specific permissions. https://cloud.google.com/iam/docs/service-accounts
**Least Privilege:** The practice of granting only the necessary permissions to each service account or user to reduce security risks, in contrast to Option D. https://cloud.google.com/iam/docs/principles-of-least-privilege

**Question: 173** CertyIQ

You are developing a financial trading application that will be used globally. Data is stored and queried using a relational structure, and clients from all over the world should get the exact identical state of the data. The application will be deployed in multiple regions to provide the lowest latency to end users. You need to select a storage option for the application data while minimizing latency. What should you do?

A. Use Cloud Bigtable for data storage.
B. Use Cloud SQL for data storage.
C. Use Cloud Spanner for data storage.
D. Use Firestore for data storage.

**Answer: C**

**Explanation:**

The correct answer is C, using Cloud Spanner for data storage. This choice best addresses the requirements of a globally distributed, strongly consistent relational database with low latency. Cloud Spanner is a globally distributed, scalable, and strongly consistent database service, making it ideal for applications requiring the same view of data across different regions. It uses a unique architecture that allows for horizontal scaling and offers transactional consistency, which are crucial for financial trading applications. Unlike Cloud SQL (B), which is region-specific and requires complex replication setups for multi-region deployments, Spanner offers built-in multi-region support and ensures strong consistency via TrueTime, a highly accurate clock. Cloud Bigtable (A), while globally scalable, is a NoSQL database and does not natively support relational structures and transactional consistency required here. Firestore (D), a NoSQL document database, also lacks the relational capabilities and strongly consistent distributed transactions required. Spanner's globally distributed nature, combined with strong consistency, ensures that users anywhere will see the same, up-to-date data. The performance characteristics of Spanner are suited for low-latency access globally, aligning

with the application's requirement. Therefore, Spanner stands as the most suitable solution for a global financial trading application demanding consistent and low-latency data access across regions.

**Supporting links:**

**Cloud Spanner Overview:** https://cloud.google.com/spanner/docs/overview
**Cloud Spanner Concepts:** https://cloud.google.com/spanner/docs/concepts
**Cloud Spanner Regions and Instances:** https://cloud.google.com/spanner/docs/instances
**Cloud SQL Overview:** https://cloud.google.com/sql/docs/overview
**Cloud Bigtable Overview:** https://cloud.google.com/bigtable/docs/overview
**Firestore Overview:** https://firebase.google.com/docs/firestore

**Question: 174** CertyIQ

You are about to deploy a new Enterprise Resource Planning (ERP) system on Google Cloud. The application holds the full database in-memory for fast data access, and you need to configure the most appropriate resources on Google Cloud for this application. What should you do?

A. Provision preemptible Compute Engine instances.
B. Provision Compute Engine instances with GPUs attached.
C. Provision Compute Engine instances with local SSDs attached.
D. Provision Compute Engine instances with M1 machine type.

**Answer: D**

**Explanation:**

The correct answer is **D. Provision Compute Engine instances with M1 machine type.** Here's why:

The ERP system's requirement for in-memory data processing necessitates a machine type optimized for large memory capacity. M1 machine types, specifically designed for memory-intensive workloads, excel in this area, offering significantly more RAM than general-purpose instance types. This ensures the entire database can reside in memory for rapid access. Preemptible instances (Option A) are unsuitable due to their potential for unpredictable termination, which could lead to data loss or downtime. GPUs (Option B) are beneficial for parallel processing tasks like machine learning, not in-memory database storage. Local SSDs (Option C), while offering fast storage, don't address the primary need for high RAM. The M1 family of machines, with its emphasis on memory, directly fulfills the ERP system's needs. Choosing M1 instances provides the balance of performance and memory capacity essential for this scenario.

Google Cloud Machine TypesGoogle Cloud M1 Machine TypesUnderstanding Memory-Optimized Machine Types

**Question: 175** CertyIQ

You have developed an application that consists of multiple microservices, with each microservice packaged in its own Docker container image. You want to deploy the entire application on Google Kubernetes Engine so that each microservice can be scaled individually. What should you do?

A. Create and deploy a Custom Resource Definition per microservice.
B. Create and deploy a Docker Compose File.
C. Create and deploy a Job per microservice.
D. Create and deploy a Deployment per microservice.

**Answer: D**

**Explanation:**

The correct answer is D, "Create and deploy a Deployment per microservice." Deployments in Kubernetes are the ideal way to manage stateless application instances, like microservices. They declaratively specify the desired state of your application, ensuring the specified number of replicas are running and healthy. This aligns perfectly with the need to scale each microservice independently.

Option A, creating a Custom Resource Definition (CRD), is not suitable for directly deploying microservices. CRDs are used to extend the Kubernetes API with custom objects, but they don't manage the deployment and scaling of applications. Option B, using Docker Compose, is primarily for development and local environments, and not well-suited for managing deployments in a production Kubernetes cluster like Google Kubernetes Engine (GKE). While Docker Compose can define multiple services, it lacks the robust scaling, rolling updates, and self-healing features of Kubernetes Deployments. Option C, using Kubernetes Jobs, is for finite, batch-oriented tasks, rather than for long-running applications like microservices. Jobs execute pods to completion, and are not appropriate for a service that should continuously run.

Deployments provide powerful capabilities, including rolling updates (allowing new versions to be released with minimal downtime), rollback mechanisms (to revert to previous versions), and automated scaling (using horizontal pod autoscaling). Therefore, deploying individual microservices as separate Kubernetes Deployments is the most appropriate practice for achieving scalable, resilient and manageable application on GKE. This allows each microservice to be scaled independently based on its resource needs and traffic demands. The flexibility offered by deployments, combined with the features provided by Kubernetes, makes it the best choice for this scenario.

**Relevant Links:**

Kubernetes Deployments: https://kubernetes.io/docs/concepts/workloads/controllers/deployment/
Kubernetes Jobs: https://kubernetes.io/docs/concepts/workloads/controllers/job/
Kubernetes Custom Resources: https://kubernetes.io/docs/concepts/extend-kubernetes/api-extension/custom-resources/
Docker Compose overview: https://docs.docker.com/compose/

**Question: 176** CertyIQ

You will have several applications running on different Compute Engine instances in the same project. You want to specify at a more granular level the service account each instance uses when calling Google Cloud APIs. What should you do?

A. When creating the instances, specify a Service Account for each instance.

B. When creating the instances, assign the name of each Service Account as instance metadata.

C. After starting the instances, use gcloud compute instances update to specify a Service Account for each instance.

D. After starting the instances, use gcloud compute instances update to assign the name of the relevant Service Account as instance metadata.

**Answer: A**

**Explanation:**

The correct answer is A: "When creating the instances, specify a Service Account for each instance." This is the most direct and secure method for granting fine-grained access to Google Cloud APIs. Service accounts are identities that applications use to authenticate with Google Cloud services. By associating a service account with a Compute Engine instance during its creation, you're explicitly defining the permissions that

applications running on that instance will have. This approach adheres to the principle of least privilege, granting each application only the access it needs. Option B, while technically possible, is not the standard method for setting instance-level service accounts; metadata is better suited for configuration variables. Options C and D suggest updating instances post-creation, which is a less efficient workflow. Modifying a running instance's service account requires restarting the instance for changes to take full effect. Therefore, specifying the service account during instance creation is the optimal and recommended way for fine-grained access control. Furthermore, this enables you to leverage Identity and Access Management (IAM) roles effectively for each service account, granting specific permissions to different instances. [https://cloud.google.com/iam/docs/service-accounts] and [https://cloud.google.com/compute/docs/access/create-enable-service-accounts-for-instances]

**Question: 177** CertyIQ

You are creating an application that will run on Google Kubernetes Engine. You have identified MongoDB as the most suitable database system for your application and want to deploy a managed MongoDB environment that provides a support SLA. What should you do?

A. Create a Cloud Bigtable cluster, and use the HBase API.
B. Deploy MongoDB Atlas from the Google Cloud Marketplace.
C. Download a MongoDB installation package, and run it on Compute Engine instances.
D. Download a MongoDB installation package, and run it on a Managed Instance Group.

**Answer: B**

**Explanation:**

The correct answer is **B. Deploy MongoDB Atlas from the Google Cloud Marketplace.** Here's why:

The scenario requires a managed MongoDB environment with a support SLA. This implies the need for a fully managed service where infrastructure management and support are handled by a third party, freeing the user to focus on the application.

Option B, deploying MongoDB Atlas from the Google Cloud Marketplace, directly addresses this requirement. MongoDB Atlas is a fully managed database service provided by MongoDB, Inc. It offers a production-ready MongoDB environment with features like automatic backups, monitoring, scaling, and crucially, an SLA for uptime and support. This aligns perfectly with the need for a managed and supported MongoDB instance.

Option A, using Cloud Bigtable with the HBase API, is incorrect because Bigtable is a NoSQL wide-column store, different from MongoDB's document-oriented model. It is not a suitable substitute for a MongoDB database.

Options C and D involve manually installing and managing MongoDB on either Compute Engine instances or a Managed Instance Group. This means the user is responsible for all operational aspects including setup, maintenance, patching, backups, and ensuring high availability, which contradicts the need for a managed service with a support SLA.

Therefore, MongoDB Atlas from the Google Cloud Marketplace is the ideal choice for a managed MongoDB deployment with an SLA, simplifying database operations and providing enterprise-grade support.

**Authoritative Links for Further Research:**

**MongoDB Atlas:** https://www.mongodb.com/atlas
**Google Cloud Marketplace:** https://cloud.google.com/marketplace
**Google Cloud Bigtable:** https://cloud.google.com/bigtable

**Question: 178** CertyIQ

You are managing a project for the Business Intelligence (BI) department in your company. A data pipeline ingests data into BigQuery via streaming. You want the users in the BI department to be able to run the custom SQL queries against the latest data in BigQuery. What should you do?

A. Create a Data Studio dashboard that uses the related BigQuery tables as a source and give the BI team view access to the Data Studio dashboard.

B. Create a Service Account for the BI team and distribute a new private key to each member of the BI team.

C. Use Cloud Scheduler to schedule a batch Dataflow job to copy the data from BigQuery to the BI team's internal data warehouse.

D. Assign the IAM role of BigQuery User to a Google Group that contains the members of the BI team.

**Answer: D**

**Explanation:**

The correct answer is **D**, assigning the IAM role of BigQuery User to a Google Group containing the BI team members. This approach directly addresses the requirement of enabling BI users to run custom SQL queries against the latest data in BigQuery. IAM roles in Google Cloud Platform (GCP) control access to resources, and the BigQuery User role grants the necessary permissions to query BigQuery datasets. Using a Google Group simplifies management by allowing you to grant permissions to a group rather than individual users, making it easier to add or remove team members. This approach also aligns with best practices for least privilege, granting only the necessary permissions to the BI team.

Option A, creating a Data Studio dashboard, limits the BI team to pre-defined visualizations and doesn't allow for custom SQL queries directly. Option B, using Service Accounts, is not suitable for end-user access and is generally used for applications, not for individual users within a business. Option C, using Cloud Scheduler to copy data, adds unnecessary complexity and latency; the BI team should directly access the data in BigQuery. Therefore, option D provides the most efficient and secure method of enabling the BI team to perform custom SQL queries on the latest BigQuery data.

**Supporting Links:**

**Google Cloud IAM:** https://cloud.google.com/iam/docs/overview
**BigQuery IAM Roles:** https://cloud.google.com/bigquery/docs/access-control
**Google Groups:** https://support.google.com/groups/answer/2464926?hl=en

**Question: 179** CertyIQ

Your company is moving its entire workload to Compute Engine. Some servers should be accessible through the Internet, and other servers should only be accessible over the internal network. All servers need to be able to talk to each other over specific ports and protocols. The current on-premises network relies on a demilitarized zone (DMZ) for the public servers and a Local Area Network (LAN) for the private servers. You need to design the networking infrastructure on
Google Cloud to match these requirements. What should you do?

A. 1. Create a single VPC with a subnet for the DMZ and a subnet for the LAN. 2. Set up firewall rules to open up relevant traffic between the DMZ and the LAN subnets, and another firewall rule to allow public ingress traffic for the DMZ.

B. 1. Create a single VPC with a subnet for the DMZ and a subnet for the LAN. 2. Set up firewall rules to open up relevant traffic between the DMZ and the LAN subnets, and another firewall rule to allow public egress traffic for the DMZ.

C. 1. Create a VPC with a subnet for the DMZ and another VPC with a subnet for the LAN. 2. Set up firewall rules to open up relevant traffic between the DMZ and the LAN subnets, and another firewall rule to allow public

ingress traffic for the DMZ.

D. 1. Create a VPC with a subnet for the DMZ and another VPC with a subnet for the LAN. 2. Set up firewall rules to open up relevant traffic between the DMZ and the LAN subnets, and another firewall rule to allow public egress traffic for the DMZ.

**Answer: A**

**Explanation:**

The correct answer is A because it provides the most efficient and secure solution for mirroring the described on-premises network structure within Google Cloud. Creating a single Virtual Private Cloud (VPC) is best practice for managing a unified network, allowing simplified routing and resource management. Subnet creation within this VPC separates the DMZ (public-facing) and LAN (private) server groups, providing logical isolation. Crucially, firewall rules are then used to control traffic flow. Specifically, rules allow the necessary communication between DMZ and LAN subnets on specific ports/protocols, while a separate rule allows traffic from the public internet into the DMZ. This mimics the function of an on-premises DMZ. Option B is incorrect because it tries to allow public egress traffic from the DMZ, which doesn't make sense for a publicly accessible DMZ. Options C and D incorrectly create two separate VPCs, which would make internal communication more complex (requiring VPC peering or other mechanisms). Option A uses a single VPC, which provides better management, security (using firewalls), and simplicity. This is a standard approach to segregating networks on Google Cloud.

Relevant links for further reading:

**VPC Network Overview:** https://cloud.google.com/vpc/docs/vpc
**Firewall Rules Overview:** https://cloud.google.com/vpc/docs/firewalls
**Subnets:** https://cloud.google.com/vpc/docs/subnets

**Question: 180** CertyIQ

You have just created a new project which will be used to deploy a globally distributed application. You will use Cloud Spanner for data storage. You want to create a Cloud Spanner instance. You want to perform the first step in preparation of creating the instance. What should you do?

A. Enable the Cloud Spanner API.

B. Configure your Cloud Spanner instance to be multi-regional.

C. Create a new VPC network with subnetworks in all desired regions.

D. Grant yourself the IAM role of Cloud Spanner Admin.

**Answer: A**

**Explanation:**

The correct answer is **A. Enable the Cloud Spanner API.** Before you can use any Google Cloud service, including Cloud Spanner, you need to explicitly enable its corresponding API for your project. This is a fundamental prerequisite for interacting with the service and accessing its resources. The Cloud Spanner API provides the necessary endpoints and functionalities to create, manage, and utilize Spanner instances. Without enabling the API, any attempt to create a Spanner instance will fail. Configuring a Spanner instance to be multi-regional (B) or creating new VPC networks (C) are actions taken after the API has been enabled. Similarly, granting yourself the Cloud Spanner Admin role (D) is also a step that occurs after the API is active because you need the API to manage IAM roles within the service's scope. Think of it like building a house - you need to clear the land (enable the API) before laying the foundation (creating the instance), wiring it (setting up networks), or assigning roles to residents (IAM). Enabling the API is the initial, foundational step that makes subsequent actions possible.

**Authoritative Links:**

**Enabling APIs:** https://cloud.google.com/apis/docs/enable-disable-apis - Google Cloud documentation on enabling APIs.
**Cloud Spanner Quickstart:** https://cloud.google.com/spanner/docs/quickstart-console - The Quickstart highlights that enabling the API is a prerequisite.

**Question: 181** CertyIQ

You have created a new project in Google Cloud through the gcloud command line interface (CLI) and linked a billing account. You need to create a new Compute
Engine instance using the CLI. You need to perform the prerequisite steps. What should you do?

A. Create a Cloud Monitoring Workspace.
B. Create a VPC network in the project.
C. Enable the compute googleapis.com API.
D. Grant yourself the IAM role of Computer Admin.

**Answer: C**

**Explanation:**

The correct answer is **C. Enable the compute.googleapis.com API.**

Before creating a Compute Engine instance, the Compute Engine API must be enabled within the target project. This API allows you to interact with Google's infrastructure to provision and manage virtual machines. The gcloud command-line tool uses this API for its operations. Without enabling the API, the CLI will not be authorized to make necessary requests to create instances, resulting in failures. Option A is incorrect because Cloud Monitoring Workspaces are used to monitor metrics, not a prerequisite for creating a VM. Option B, creating a VPC, while essential for a functional network, isn't immediately required before the instance can be created; the default network could be used initially. While having the Computer Admin IAM role is vital for performing tasks like instance creation, it's not a prerequisite step before using gcloud to launch a VM. The API needs to be enabled first, before access controls can be validated. The correct workflow is to first enable the API, then grant necessary permissions, then create the network if needed, and finally create the compute engine instance.

**Authoritative Links:**

**Enabling APIs:** https://cloud.google.com/apis/docs/enable-disable-apis
**Compute Engine API:** https://cloud.google.com/compute/docs/reference/rest/v1/
**gcloud compute instances create:** https://cloud.google.com/sdk/gcloud/reference/compute/instances/create

**Question: 182** CertyIQ

Your company has developed a new application that consists of multiple microservices. You want to deploy the application to Google Kubernetes Engine (GKE), and you want to ensure that the cluster can scale as more applications are deployed in the future. You want to avoid manual intervention when each new application is deployed. What should you do?

A. Deploy the application on GKE, and add a HorizontalPodAutoscaler to the deployment.
B. Deploy the application on GKE, and add a VerticalPodAutoscaler to the deployment.
C. Create a GKE cluster with autoscaling enabled on the node pool. Set a minimum and maximum for the size of the node pool.

D. Create a separate node pool for each application, and deploy each application to its dedicated node pool.

**Answer: C**

**Explanation:**

The correct answer is **C: Create a GKE cluster with autoscaling enabled on the node pool. Set a minimum and maximum for the size of the node pool.** This approach directly addresses the requirement for scalability and automated handling of future deployments. Node pool autoscaling in GKE allows the cluster to dynamically adjust the number of nodes based on resource demands. By setting a minimum and maximum, you establish boundaries for the scaling process, preventing runaway resource consumption or insufficient capacity. This avoids manual intervention for each new application deployment, as the cluster itself adapts to the increased workload. Option A, using a HorizontalPodAutoscaler (HPA), only scales the number of pods within the existing node capacity, not the underlying infrastructure. Option B, a VerticalPodAutoscaler (VPA), adjusts resources within the existing pod, which does not satisfy the requirement for cluster scalability and has limitations. Option D, creating separate node pools per application, is an inefficient way to manage resources and is more difficult to administer, adding to operational overhead. Node autoscaling ensures that the cluster can handle increased resource requirements of new applications.

**Here's why C is the best fit, using Cloud concepts:**

**Scalability:** Node pool autoscaling directly addresses the requirement for scaling the cluster to handle more applications in the future, a key principle of cloud architecture.
**Automation:** This approach minimizes manual intervention as the cluster responds to changes in demand.
**Efficiency:** It uses resources more effectively by dynamically adding/removing nodes only when needed.
**Cost Optimization:** Autoscaling can help reduce costs by only using the resources required at any given time.
**Resource Management:** It avoids over-provisioning and under-provisioning of resources, providing a balanced approach.

**Authoritative Links:**

**GKE Node Autoscaling:** https://cloud.google.com/kubernetes-engine/docs/how-to/node-autoscaling
**Understanding Autoscaling in GKE:** https://cloud.google.com/blog/products/containers-kubernetes/kubernetes-autoscaling-101

**Question: 183** CertyIQ

You need to manage a third-party application that will run on a Compute Engine instance. Other Compute Engine instances are already running with default configuration. Application installation files are hosted on Cloud Storage. You need to access these files from the new instance without allowing other virtual machines (VMs) to access these files. What should you do?

A. Create the instance with the default Compute Engine service account. Grant the service account permissions on Cloud Storage.

B. Create the instance with the default Compute Engine service account. Add metadata to the objects on Cloud Storage that matches the metadata on the new instance.

C. Create a new service account and assign this service account to the new instance. Grant the service account permissions on Cloud Storage.

D. Create a new service account and assign this service account to the new instance. Add metadata to the objects on Cloud Storage that matches the metadata on the new instance.

**Answer: C**

**Explanation:**

The correct answer is **C. Create a new service account and assign this service account to the new instance. Grant the service account permissions on Cloud Storage.**

Here's the detailed justification:

The core requirement is to grant access to Cloud Storage files only to the new Compute Engine instance, preventing access from other VMs. Using the default Compute Engine service account (option A and B) would grant permissions to all VMs using that account, violating the security requirement.

Service accounts in Google Cloud provide a way to manage permissions for applications running on resources like Compute Engine. By creating a new, dedicated service account for the new instance (option C and D), you isolate its permissions. This adheres to the principle of least privilege, granting only the necessary access. This ensures the new VM can access Cloud Storage files while other VMs using other service accounts cannot.

Granting the dedicated service account specific permissions on the Cloud Storage bucket containing the installation files via IAM (Identity and Access Management) is crucial (option C). This gives the service account the authorization to fetch the required files. Option D incorrectly suggests using metadata, which is not a proper authorization mechanism for this use case and does not provide granular access control like IAM. Metadata is primarily used for providing configuration information to the VM.

Therefore, creating a new service account, assigning it to the new instance, and granting this account the necessary Cloud Storage permissions ensures the required secure and isolated access.

**Key Concepts:**

**Service Accounts:** Non-user accounts used by applications and services to interact with Google Cloud resources.
**Principle of Least Privilege:** Granting only the necessary permissions to perform a specific task.
**IAM (Identity and Access Management):** Google Cloud's system for managing permissions.

**Authoritative Links:**

Understanding service accounts
Granting service account permissions
Best practices for service accounts

**Question: 184** CertyIQ

You need to configure optimal data storage for files stored in Cloud Storage for minimal cost. The files are used in a mission-critical analytics pipeline that is used continually. The users are in Boston, MA (United States). What should you do?

A. Configure regional storage for the region closest to the users. Configure a Nearline storage class.
B. Configure regional storage for the region closest to the users. Configure a Standard storage class.
C. Configure dual-regional storage for the dual region closest to the users. Configure a Nearline storage class.
D. Configure dual-regional storage for the dual region closest to the users. Configure a Standard storage class.

**Answer: D**

**Explanation:**

The correct answer is D, configuring dual-regional storage with a Standard storage class. Here's why:

The question prioritizes cost minimization while maintaining optimal performance for a mission-critical, continuously used analytics pipeline. This means we need to balance cost with availability and latency. The

key factors driving the choice are:

1. **Mission-Critical & Continuous Use:** The continuous usage implies high access frequency. Standard storage, while slightly more expensive per GB than Nearline or Coldline, is optimized for frequent access and delivers the lowest latency. Using Nearline, designed for infrequent access, would result in higher access fees and latency penalties for the analytics pipeline.
2. **Geographic Proximity & Availability:** The users are in Boston, MA. While a single regional bucket could offer lower cost, the mission-critical nature of the pipeline necessitates higher availability and resilience. Dual-regional storage provides redundancy by storing data in two geographically separate locations within the same dual-region, ensuring business continuity if one region experiences an outage.
3. **Cost-Effectiveness Trade-off:** Although dual-regional storage has a higher storage cost than regional storage, this is offset by the mission-critical nature of the application, and the latency improvement compared to a multi-region bucket. Further, a single regional bucket could have outages, causing interruptions to the analytics pipeline and financial losses. Therefore, the slight increase in storage costs for dual-region storage is a worthwhile trade-off for the improved availability.
4. **Standard Storage Class Appropriateness:** Standard storage is the appropriate class given that the data is accessed regularly, which justifies the higher storage price to avoid retrieval fees associated with cold storage classes like Nearline, Coldline, or Archive.

Therefore, **dual-regional Standard storage** provides the necessary availability for mission-critical applications, the low latency needed for continuous use, and optimal cost-efficiency for the requirements of the scenario.

**Authoritative Links:**

**Cloud Storage Classes:** https://cloud.google.com/storage/docs/storage-classes
**Cloud Storage Locations:** https://cloud.google.com/storage/docs/locations
**Choosing a Cloud Storage Class**: https://cloud.google.com/blog/products/storage/choosing-the-right-google-cloud-storage-class

**Question: 185** CertyIQ

You are developing a new web application that will be deployed on Google Cloud Platform. As part of your release cycle, you want to test updates to your application on a small portion of real user traffic. The majority of the users should still be directed towards a stable version of your application. What should you do?

A. Deploy the application on App Engine. For each update, create a new version of the same service. Configure traffic splitting to send a small percentage of traffic to the new version.

B. Deploy the application on App Engine. For each update, create a new service. Configure traffic splitting to send a small percentage of traffic to the new service.

C. Deploy the application on Kubernetes Engine. For a new release, update the deployment to use the new version.

D. Deploy the application on Kubernetes Engine. For a new release, create a new deployment for the new version. Update the service to use the new deployment.

**Answer: A**

**Explanation:**

Option A is the correct approach for canary deployments using App Engine. App Engine's versioning system is designed for this purpose. By deploying each update as a new version within the same service, you leverage

App Engine's built-in traffic splitting capabilities. This allows you to gradually shift traffic from the stable version to the new version, effectively performing a canary test. You can precisely control the percentage of traffic routed to each version, ensuring only a small portion of users experience the update initially. This minimizes risk and allows for observation of the new version's behavior in a real-world setting. Using the same service simplifies management and avoids unnecessary resource overhead.

Option B is incorrect because creating new services for each update complicates the overall architecture and resource management. While traffic splitting can be done between services, it's less efficient than using App Engine's versioning within a single service for canary deployments.

Option C is incorrect because updating the existing deployment in Kubernetes Engine would be a blue/green deployment approach, not a canary deployment. This would shift all traffic simultaneously, not gradually.

Option D is incorrect as creating a new deployment and changing the service to use it would be a blue/green deployment approach, not a canary. This approach does not facilitate a phased rollout and the associated monitoring and verification of a new version with a small segment of traffic.

Relevant Links:

**App Engine Traffic Splitting:** https://cloud.google.com/appengine/docs/standard/python/how-instances-are-managed#traffic_splitting
**Canary Deployments:** https://martinfowler.com/bliki/CanaryRelease.html
**App Engine Versions:** https://cloud.google.com/appengine/docs/standard/python/how-instances-are-managed

**Question: 186** CertyIQ

You need to add a group of new users to Cloud Identity. Some of the users already have existing Google accounts. You want to follow one of Google's recommended practices and avoid conflicting accounts. What should you do?

A. Invite the user to transfer their existing account.
B. Invite the user to use an email alias to resolve the conflict.
C. Tell the user that they must delete their existing account.
D. Tell the user to remove all personal email from the existing account.

**Answer: A**

**Explanation:**

The correct answer is **A. Invite the user to transfer their existing account.** This approach aligns with Google's best practices for managing users in Cloud Identity, particularly when dealing with existing Google accounts (personal accounts). Here's why:

When users have existing Google accounts (e.g., Gmail accounts), inviting them to transfer their account to your Cloud Identity organization prevents the creation of conflicting accounts, where a user has one personal account and one managed account with the same email address. Such conflicts can lead to confusion and management challenges. The transfer process migrates the user's existing Google account (including its data, if opted for) into the organization, thus consolidating their identity under a single managed account. It also ensures the user continues to access their services with the same credentials, thereby minimizing disruption. This method respects the principle of least disruption and provides a smoother user experience. Options B, C, and D are not recommended. Email aliases (B) don't solve the fundamental conflict of identity and can complicate account access. Forcing users to delete their existing accounts (C) is impractical and causes data loss and user frustration. Removing personal emails (D) from their account doesn't integrate them into the organization and doesn't address the duplicate identity issue.

**Authoritative Links:**

**Google Cloud Documentation on User Accounts:** https://cloud.google.com/identity/docs/manage-accounts (This provides a general overview of managing user accounts in Google Cloud Identity)
**Google Cloud Documentation on Transferring Accounts:** https://support.google.com/a/answer/6355147?hl=en (This document specifically explains the process of transferring existing Google accounts into your managed organization.)

**Question: 187** CertyIQ

You need to manage a Cloud Spanner instance for best query performance. Your instance in production runs in a single Google Cloud region. You need to improve performance in the shortest amount of time. You want to follow Google best practices for service configuration. What should you do?

A. Create an alert in Cloud Monitoring to alert when the percentage of high priority CPU utilization reaches 45%. If you exceed this threshold, add nodes to your instance.

B. Create an alert in Cloud Monitoring to alert when the percentage of high priority CPU utilization reaches 45%. Use database query statistics to identify queries that result in high CPU usage, and then rewrite those queries to optimize their resource usage.

C. Create an alert in Cloud Monitoring to alert when the percentage of high priority CPU utilization reaches 65%. If you exceed this threshold, add nodes to your instance.

D. Create an alert in Cloud Monitoring to alert when the percentage of high priority CPU utilization reaches 65%. Use database query statistics to identify queries that result in high CPU usage, and then rewrite those queries to optimize their resource usage.

**Answer: C**

**Explanation:**

The correct answer is C because it aligns with Google Cloud's best practices for managing Cloud Spanner performance and addresses the need for rapid improvement. Monitoring CPU utilization is crucial for understanding Spanner's load. A threshold of 65% for high-priority CPU utilization, as suggested in Google's documentation, is a more appropriate trigger point for scaling than 45%. While identifying and optimizing poorly performing queries (as mentioned in options B and D) is essential for long-term optimization, simply adding nodes (option C) provides the quickest relief when a performance bottleneck is identified. This immediate action addresses performance issues rapidly. Option A and B recommend a less optimal threshold of 45%, leading to potentially unnecessary scaling events. Option D is not the best immediate response. The best strategy combines both scaling up and query optimization, but option C is the best immediate response for the shortest time to performance improvement. While query tuning (as in B and D) is important, it's a process that takes longer. Therefore, the fastest path to improved performance when experiencing high CPU is to add nodes once the utilization has reached a certain threshold.

Authoritative Links:

**Cloud Spanner Performance:** https://cloud.google.com/spanner/docs/performance
**Monitoring Spanner:** https://cloud.google.com/spanner/docs/monitoring
**Scaling Spanner:** https://cloud.google.com/spanner/docs/instance-config
**High-Priority CPU Utilization:** While there isn't one single document stating 65% precisely, it's the typical advised threshold to signal scaling actions, and it is implied in the general performance documentation for Spanner

**Question: 188** CertyIQ

Your company has an internal application for managing transactional orders. The application is used exclusively by

employees in a single physical location. The application requires strong consistency, fast queries, and ACID guarantees for multi-table transactional updates. The first version of the application is implemented in PostgreSQL, and you want to deploy it to the cloud with minimal code changes. Which database is most appropriate for this application?

A. BigQuery
B. Cloud SQL
C. Cloud Spanner
D. Cloud Datastore

**Answer: B**

**Explanation:**

The correct answer is **B. Cloud SQL**.

Cloud SQL is a fully managed database service offered by Google Cloud Platform (GCP) that supports relational databases like PostgreSQL, MySQL, and SQL Server. This aligns directly with the requirement of migrating an existing PostgreSQL application with minimal code changes. Cloud SQL provides the necessary infrastructure, including provisioning, patching, backups, and scaling, allowing developers to focus on the application logic rather than database management tasks. Furthermore, it natively supports ACID properties and transactional updates across multiple tables, satisfying the need for strong consistency and transactional guarantees.

BigQuery (option A) is a data warehouse service designed for analytical queries on large datasets; it's not optimized for transactional workloads requiring strong consistency and frequent updates. Cloud Spanner (option C) provides globally distributed scalability and strong consistency but introduces more complexity and might require significant code refactoring. Cloud Datastore (option D) is a NoSQL, schema-less datastore, therefore, unsuitable for relational data migration. Given the minimal code change requirement and PostgreSQL compatibility, Cloud SQL is the most suitable choice for a straightforward migration of the existing application.

In summary, Cloud SQL provides a managed, relational database service that best matches the requirements of migrating a PostgreSQL-based transactional application that requires strong consistency, ACID properties, and minimal code changes.

**Authoritative Links:**

**Cloud SQL Overview:** https://cloud.google.com/sql/docs/
**Cloud SQL for PostgreSQL:** https://cloud.google.com/sql/docs/postgres
**BigQuery Overview:** https://cloud.google.com/bigquery/docs
**Cloud Spanner Overview:** https://cloud.google.com/spanner/docs
**Cloud Datastore Overview:** https://cloud.google.com/datastore/docs

**Question: 189** CertyIQ

You are assigned to maintain a Google Kubernetes Engine (GKE) cluster named 'dev' that was deployed on Google Cloud. You want to manage the GKE configuration using the command line interface (CLI). You have just downloaded and installed the Cloud SDK. You want to ensure that future CLI commands by default address this specific cluster What should you do?

A. Use the command gcloud config set container/cluster dev.
B. Use the command gcloud container clusters update dev.
C. Create a file called gke.default in the ~/.gcloud folder that contains the cluster name.
D. Create a file called defaults.json in the ~/.gcloud folder that contains the cluster name.

**Answer: A**

**Explanation:**

The correct answer is **A: Use the command gcloud config set container/cluster dev**. Here's why:

The gcloud command-line tool is the primary interface for interacting with Google Cloud services, including GKE. To avoid repeatedly specifying the target cluster for each command, you can configure a default cluster using the gcloud config set command. This command allows you to set various configuration properties.

Specifically, container/cluster is the property that defines the default GKE cluster. By setting it to dev, you're telling gcloud to automatically target the dev cluster for subsequent commands. This simplifies your workflow and reduces the chance of errors.

Option B, gcloud container clusters update dev, is used for modifying the configuration of the cluster itself, not for setting the default for the CLI. Options C and D suggest creating configuration files, which isn't the standard way to manage default cluster settings using gcloud. While custom configuration can be done, gcloud config set provides a more direct and intended approach for setting a default cluster.

Therefore, gcloud config set container/cluster dev is the recommended and most efficient method to ensure that future CLI commands default to the specified 'dev' cluster, aligning with the best practices for gcloud usage.

**Authoritative Links for Further Research:**

**gcloud config set documentation:** https://cloud.google.com/sdk/gcloud/reference/config/set
**gcloud command-line tool overview:** https://cloud.google.com/sdk/gcloud
**Working with clusters in GKE:** https://cloud.google.com/kubernetes-engine/docs/how-to/working-with-clusters

**Question: 190** CertyIQ

The sales team has a project named Sales Data Digest that has the ID acme-data-digest. You need to set up similar Google Cloud resources for the marketing team but their resources must be organized independently of the sales team. What should you do?

A. Grant the Project Editor role to the Marketing team for acme-data-digest.

B. Create a Project Lien on acme-data-digest and then grant the Project Editor role to the Marketing team.

C. Create another project with the ID acme-marketing-data-digest for the Marketing team and deploy the resources there.

D. Create a new project named Marketing Data Digest and use the ID acme-data-digest. Grant the Project Editor role to the Marketing team.

**Answer: C**

**Explanation:**

The correct answer is **C. Create another project with the ID acme-marketing-data-digest for the Marketing team and deploy the resources there.**

Here's why:

Projects in Google Cloud are fundamental for resource organization and isolation. They provide a distinct namespace for resources, enabling clear separation and independent management. Option A, granting the Marketing team the Project Editor role on the existing Sales project, would not achieve the goal of independent resource organization. Instead, it would give the Marketing team the ability to manage and potentially interfere with the Sales team's resources, violating the requirement. Project Liens, as suggested in

Option B, are used to prevent accidental deletion of projects, not for isolating teams. Option D is incorrect because project IDs must be unique within an organization. You cannot have two projects with the same ID (acme-data-digest), even if they have different names.

By creating a new project specifically for the Marketing team (acme-marketing-data-digest), you establish a clear boundary for their resources. This ensures they are independent, have their own identity and billing, and are free from potential conflicts with the Sales team's infrastructure. Each team can then manage their own resources, permissions, and configurations without affecting the other. This approach aligns with best practices for multi-team deployments in Google Cloud, leveraging the fundamental isolation provided by projects.

Further Research:

**Google Cloud Resource Hierarchy:** https://cloud.google.com/resource-manager/docs/cloud-platform-resource-hierarchy
**Projects:** https://cloud.google.com/resource-manager/docs/creating-managing-projects
**Project Identifiers:** https://cloud.google.com/resource-manager/docs/project-identifiers

**Question: 191** CertyIQ

You have deployed multiple Linux instances on Compute Engine. You plan on adding more instances in the coming weeks. You want to be able to access all of these instances through your SSH client over the internet without having to configure specific access on the existing and new instances. You do not want the Compute Engine instances to have a public IP. What should you do?

A. Configure Cloud Identity-Aware Proxy for HTTPS resources.
B. Configure Cloud Identity-Aware Proxy for SSH and TCP resources
C. Create an SSH keypair and store the public key as a project-wide SSH Key.
D. Create an SSH keypair and store the private key as a project-wide SSH Key.

**Answer: B**

**Explanation:**

The correct answer is **B. Configure Cloud Identity-Aware Proxy for SSH and TCP resources.**

Here's why:

The primary requirement is to access multiple Compute Engine instances without public IPs via SSH over the internet, without configuring each instance individually. This necessitates a secure intermediary that can handle authentication and routing.

Cloud Identity-Aware Proxy (IAP) fulfills this role perfectly for SSH and TCP. IAP acts as a reverse proxy, verifying user identity and access permissions before allowing connections to backend resources, in this case, the Compute Engine instances. This removes the need for public IPs on the instances, enhancing security. Instead of directly exposing the instances to the internet, you connect to IAP, which then forwards the secure SSH connection.

Option A is incorrect because IAP for HTTPS focuses on securing web applications, not SSH connections. Options C and D are incorrect because managing SSH keys project-wide does not, on its own, provide secure internet access to instances without public IPs. These options would still require a mechanism to connect over the internet, for example, through a public IP on the instance itself. Option B leverages the IAP's functionality to handle authentication and routing, addressing the question's core requirement. IAP securely bridges the internet to internal compute resources by verifying the identity before establishing connections. The instances themselves are not exposed to the outside network, enhancing security and reducing the

complexity of network management and security configurations.

**Authoritative Links:**

**Cloud Identity-Aware Proxy Overview:** https://cloud.google.com/iap/docs/overview
**Accessing VM instances with IAP TCP forwarding:** https://cloud.google.com/iap/docs/tcp-forwarding

**Question: 192** CertyIQ

You have created an application that is packaged into a Docker image. You want to deploy the Docker image as a workload on Google Kubernetes Engine. What should you do?

A. Upload the image to Cloud Storage and create a Kubernetes Service referencing the image.
B. Upload the image to Cloud Storage and create a Kubernetes Deployment referencing the image.
C. Upload the image to Container Registry and create a Kubernetes Service referencing the image.
D. Upload the image to Container Registry and create a Kubernetes Deployment referencing the image.

**Answer: D**

**Explanation:**

The correct answer is **D. Upload the image to Container Registry and create a Kubernetes Deployment referencing the image.**

Here's a detailed justification:

1. **Container Images Need a Registry:** Docker images are not directly deployable to Kubernetes. They need to be stored in a container registry. Google Cloud offers Container Registry (now Artifact Registry), a private and secure place to store and manage your container images. Alternatives like Cloud Storage are not designed for container image storage and management.
2. **Kubernetes Deployments for Scalable Applications:** Kubernetes provides various object types to manage applications, and Deployment is the recommended object for deploying stateless applications. Deployments declaratively describe the desired state of your application and manage replica sets, ensuring high availability and automatic updates. Services, on the other hand, are used for exposing applications to the network.
3. **Why not Cloud Storage:** Cloud Storage is designed for storing generic objects like documents, videos, etc. While you could technically store a Docker image there, it lacks the necessary tooling for container image management, such as versioning, tagging, and direct integration with Kubernetes.
4. **Service vs. Deployment:** A Kubernetes Service acts as an abstraction layer, defining a logical set of pods (containers) and a policy by which to access them. A Service isn't for deploying the image itself; it's for making the deployed application accessible. A Deployment is specifically responsible for pulling the image from the registry and ensuring the desired number of replicas is running.
5. **Workflow Summary:** The standard workflow for deploying a containerized application on GKE involves these steps: build the Docker image, push the image to Container Registry (or Artifact Registry), create a Kubernetes Deployment manifest, and then apply it to your GKE cluster. The Deployment will then pull the image from the registry and launch the containers within pods.

In conclusion, storing the image in Container Registry (or Artifact Registry) enables proper version control and retrieval by Kubernetes. Using a Kubernetes Deployment object provides a declarative and scalable way to manage your containerized application.

**Authoritative links:**

**Google Cloud Container Registry Documentation:** https://cloud.google.com/container-registry/docs
**Google Cloud Artifact Registry Documentation:** https://cloud.google.com/artifact-registry/docs
**Kubernetes Deployments Documentation:** https://kubernetes.io/docs/concepts/workloads/controllers/deployment/
**Kubernetes Services Documentation:** https://kubernetes.io/docs/concepts/services-networking/service/

**Question: 193** CertyIQ

You are using Data Studio to visualize a table from your data warehouse that is built on top of BigQuery. Data is appended to the data warehouse during the day.
At night, the daily summary is recalculated by overwriting the table. You just noticed that the charts in Data Studio are broken, and you want to analyze the problem. What should you do?

A. Review the Error Reporting page in the Cloud Console to find any errors.
B. Use the BigQuery interface to review the nightly job and look for any errors.
C. Use Cloud Debugger to find out why the data was not refreshed correctly.
D. In Cloud Logging, create a filter for your Data Studio report.

**Answer: B**

**Explanation:**

The correct answer is **B. Use the BigQuery interface to review the nightly job and look for any errors.**

Here's why: The core issue stems from the nightly overwriting of the BigQuery table that Data Studio relies on. When charts in Data Studio break, it often indicates that the underlying data source is either unavailable, has a schema change, or has no data at all. Since the table is overwritten nightly, the most likely cause of the breakage is an issue with this nightly job. Examining the job directly within BigQuery provides the most immediate insights.

Option A (Error Reporting) is less likely to be helpful in this situation because it primarily captures application-level errors, not necessarily issues arising from batch data processing within BigQuery. Option C (Cloud Debugger) is designed for debugging code execution, and wouldn't be applicable to investigating a data warehouse refresh job. Option D (Cloud Logging) could provide some information, but directly reviewing the BigQuery job is more focused and efficient in pinpointing the problem.

By checking the BigQuery interface, you can specifically examine the following for clues:

1. **Job Status:** Verify if the nightly job completed successfully or failed.
2. **Error Messages:** If the job failed, the error messages often provide details on the reasons, such as incorrect queries or data issues.
3. **Query Details:** Ensure that the query used to create the table is correct and has not been inadvertently altered.
4. **Data Volume:** See if any data was written to the table after the job completed.

These insights directly correlate to the nightly process and allow for targeted debugging of the Data Studio breakage. Cloud Logging might be a useful secondary check for BigQuery logs, but starting with the BigQuery interface is a more precise approach.

Here are some relevant links for more information:

**BigQuery Documentation:** https://cloud.google.com/bigquery/docs
**Data Studio Documentation:** https://support.google.com/datastudio/?hl=en#topic=7647269

**Cloud Logging:** https://cloud.google.com/logging/docs

**Question: 194** CertyIQ

You have been asked to set up the billing configuration for a new Google Cloud customer. Your customer wants to group resources that share common IAM policies. What should you do?

A. Use labels to group resources that share common IAM policies.

B. Use folders to group resources that share common IAM policies.

C. Set up a proper billing account structure to group IAM policies.

D. Set up a proper project naming structure to group IAM policies.

**Answer: B**

**Explanation:**

The correct answer is **B. Use folders to group resources that share common IAM policies.** Folders in Google Cloud are hierarchical organizational units that sit between the organization node and projects. They are designed specifically for grouping projects that share similar characteristics, including IAM policies. By assigning IAM policies at the folder level, these policies are inherited by all projects within that folder, ensuring consistent access control for resources that share a common security posture. This simplifies policy management and reduces the risk of inconsistencies. Using labels (A) is useful for metadata tagging but doesn't provide hierarchical IAM inheritance. Billing accounts (C) are for payment management, not IAM. Project naming conventions (D) are helpful for organization but don't affect policy inheritance. Folders provide the right level of hierarchy and access control needed in this scenario.

**Authoritative Links:**

**Google Cloud Resource Hierarchy:** https://cloud.google.com/resource-manager/docs/cloud-platform-resource-hierarchy
**IAM with Folders:** https://cloud.google.com/resource-manager/docs/access-control-folders

**Question: 195** CertyIQ

You have been asked to create robust Virtual Private Network (VPN) connectivity between a new Virtual Private Cloud (VPC) and a remote site. Key requirements include dynamic routing, a shared address space of 10.19.0.1/22, and no overprovisioning of tunnels during a failover event. You want to follow Google- recommended practices to set up a high availability Cloud VPN. What should you do?

A. Use a custom mode VPC network, configure static routes, and use active/passive routing.

B. Use an automatic mode VPC network, configure static routes, and use active/active routing.

C. Use a custom mode VPC network, use Cloud Router border gateway protocol (BGP) routes, and use active/passive routing.

D. Use an automatic mode VPC network, use Cloud Router border gateway protocol (BGP) routes, and configure policy-based routing.

**Answer: C**

**Explanation:**

The correct answer is C because it aligns with Google's best practices for highly available Cloud VPN setups using dynamic routing. A custom mode VPC network (as opposed to an automatic mode VPC network) is necessary because it gives you full control over subnet creation and IP address ranges. This is crucial for avoiding overlapping IP ranges when connecting to external networks via VPN, especially given the

requirement of a specific shared address space (10.19.0.1/22) which requires manual configuration. Static routes (options A & B) are not ideal for dynamic, resilient connectivity. BGP, facilitated by Cloud Router, is essential for dynamic routing, enabling automatic route updates and failover between tunnels, fulfilling the "no overprovisioning of tunnels" requirement. In an active/passive setup (option C), one tunnel is primarily active, and the other becomes active only upon failure of the primary. This prevents unnecessary data transfer and keeps cost optimal and is better suited for the requirement of dynamic routing than active/active. Policy-based routing (option D) is not used for dynamic routing with BGP in Cloud VPN. Therefore, option C, using a custom mode VPC, Cloud Router with BGP, and active/passive routing is the recommended solution by Google for the outlined requirements.

**Authoritative Links:**

**Cloud VPN Overview:** https://cloud.google.com/vpn/docs/concepts/overview
**Choosing the Right VPN Architecture:** https://cloud.google.com/vpn/docs/how-to/choosing-a-vpn-architecture
**Cloud Router Overview:** https://cloud.google.com/router/docs/concepts/overview
**Custom vs. Auto Mode Networks:** https://cloud.google.com/vpc/docs/vpc
**High-availability VPN configurations using Cloud Router:** https://cloud.google.com/network-connectivity/docs/vpn/how-to/high-availability

**Question: 196** CertyIQ

You are running multiple microservices in a Kubernetes Engine cluster. One microservice is rendering images. The microservice responsible for the image rendering requires a large amount of CPU time compared to the memory it requires. The other microservices are workloads that are optimized for n1-standard machine types. You need to optimize your cluster so that all workloads are using resources as efficiently as possible. What should you do?

A. Assign the pods of the image rendering microservice a higher pod priority than the other microservices.

B. Create a node pool with compute-optimized machine type nodes for the image rendering microservice. Use the node pool with general-purpose machine type nodes for the other microservices.

C. Use the node pool with general-purpose machine type nodes for the image rendering microservice. Create a node pool with compute-optimized machine type nodes for the other microservices.

D. Configure the required amount of CPU and memory in the resource requests specification of the image rendering microservice deployment. Keep the resource requests for the other microservices at the default.

**Answer: B**

**Explanation:**

**Justification for Answer B:**

Option B, creating a separate node pool with compute-optimized machines for the image rendering microservice, is the most efficient solution for optimizing resource utilization. This approach leverages the concept of node pools within Kubernetes Engine to cater to diverse workload requirements. The image rendering microservice, being CPU-intensive, benefits significantly from compute-optimized machines that offer higher CPU-to-memory ratios, leading to better performance and resource utilization for that specific task. Simultaneously, the other microservices, being optimized for n1-standard (general-purpose) machines, can continue running efficiently in their designated node pool, preventing resource contention and ensuring cost-effectiveness.

This strategy aligns with best practices for resource management in Kubernetes, emphasizing workload isolation and tailored infrastructure. By isolating the CPU-heavy workload, it prevents resource contention with other services. Assigning higher pod priority (Option A) only influences scheduling during resource scarcity, not optimizing for the workload's inherent need for high CPU. Options C and D are incorrect; Option C reverses the correct node placement, and Option D only specifies requests, not optimizing the machine type

on which to run the workload. Using compute optimized machines means running the compute heavy workload on more efficient hardware.

**Supporting Concepts and Links:**

**Kubernetes Node Pools:** A node pool is a subset of nodes within a cluster that have the same machine type, enabling grouping of similar workload requirements. https://cloud.google.com/kubernetes-engine/docs/concepts/node-pools
**Compute-Optimized Machine Types:** These are machine types optimized for CPU-intensive workloads, offering higher CPU cores per memory and enhanced performance.
https://cloud.google.com/compute/docs/machine-types
**Resource Requests and Limits:** Resource requests and limits ensure efficient usage of the cluster. However, setting these without the correct machine type is insufficient for optimum use of compute resources.
https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
**Workload Optimization:** Tailoring infrastructure based on workload characteristics is crucial for cost-effectiveness and performance in cloud environments.
https://cloud.google.com/architecture/framework/optimize-cost

**Question: 197** CertyIQ

Your organization has three existing Google Cloud projects. You need to bill the Marketing department for only their Google Cloud services for a new initiative within their group. What should you do?

A. 1. Verify that you are assigned the Billing Administrator IAM role for your organization's Google Cloud Project for the Marketing department. 2. Link the new project to a Marketing Billing Account.

B. 1. Verify that you are assigned the Billing Administrator IAM role for your organization's Google Cloud account. 2. Create a new Google Cloud Project for the Marketing department. 3. Set the default key-value project labels to department:marketing for all services in this project.

C. 1. Verify that you are assigned the Organization Administrator IAM role for your organization's Google Cloud account. 2. Create a new Google Cloud Project for the Marketing department. 3. Link the new project to a Marketing Billing Account.

D. 1. Verify that you are assigned the Organization Administrator IAM role for your organization's Google Cloud account. 2. Create a new Google Cloud Project for the Marketing department. 3. Set the default key-value project labels to department:marketing for all services in this project.

**Answer: B**

**Explanation:**

The correct answer is **B**. Here's why:

The core requirement is to bill the Marketing department separately for their new initiative within Google Cloud. This necessitates segregating their resource usage at the billing level.

Option B achieves this through the following steps:

1. **Billing Administrator Role (Organization Level):** The Billing Administrator role at the organization level (not project level as suggested in option A) is essential to manage billing accounts, link projects to them, and define how billing is handled across the entire organization. This is a prerequisite to creating a new billing account and linking a project to it.
2. **New Google Cloud Project:** Creating a dedicated Google Cloud Project isolates the Marketing initiative's resources. This segregation prevents the mixing of costs with other projects and allows for specific budget tracking for the Marketing department. Each project has its own set of resources, making it easier to manage and track resource usage by department.
3. **Default Key-Value Project Labels:** Project labels help categorize and filter project resources but **do not affect billing**. They are useful for organizational and management purposes but not for billing

separation directly. Project labels are used for filtering and reporting after billing data is exported, not during billing calculations itself.

Option A is incorrect because it incorrectly focuses on project-level Billing Administrator rights. Also, it is missing the creation of new project. Options C and D both involve Organization Administrator role, which is higher level than Billing Administrator and is unnecessary for this task. Option D also suffers from the problem of incorrect label usage for billing separation, like in Option B.

In essence, each Google Cloud project is associated with a single billing account. By creating a new project for Marketing and linking it to a dedicated billing account (which is assumed to exist), their costs can be accurately tracked separately.

**Authoritative Links:**

**Cloud Billing Overview:** https://cloud.google.com/billing/docs/concepts (Explains billing accounts, projects, and how they relate)
**IAM for Cloud Billing:** https://cloud.google.com/billing/docs/concepts/access-control (Details necessary roles for billing management)
**Organizing Cloud Resources:** https://cloud.google.com/resource-manager/docs/cloud-platform-resource-hierarchy (Illustrates the hierarchical nature of Google Cloud resources, including projects)

**Question: 198** CertyIQ

You deployed an application on a managed instance group in Compute Engine. The application accepts Transmission Control Protocol (TCP) traffic on port 389 and requires you to preserve the IP address of the client who is making a request. You want to expose the application to the internet by using a load balancer. What should you do?

A.Expose the application by using an external TCP Network Load Balancer.
B.Expose the application by using a TCP Proxy Load Balancer.
C.Expose the application by using an SSL Proxy Load Balancer.
D.Expose the application by using an internal TCP Network Load Balancer.

**Answer: B**

**Explanation:**

The correct answer is **B. Expose the application by using a TCP Proxy Load Balancer.**

Here's why:

**Preserving Client IP:** The core requirement is to preserve the client's original IP address. Network Load Balancers (both internal and external) do not preserve the original client IP for TCP traffic. They forward traffic using their own IP, so the backend instances see the load balancer's IP as the source. A TCP Proxy Load Balancer, on the other hand, preserves the original client IP by using the TCP Proxy Protocol. This sends client information in a header before forwarding the request, making it available at the application server.
**TCP Protocol:** The application uses TCP on port 389, so an appropriate load balancer needs to handle TCP traffic. The TCP Proxy Load Balancer is specifically designed for this.
**External Exposure:** The requirement is to expose the application to the internet, which calls for an external-facing load balancer.
**SSL Proxy Inappropriateness:** An SSL Proxy Load Balancer is not needed as the question does not specify any requirement for handling encrypted traffic. Therefore, that choice would be overkill.
**Internal Load Balancer Restriction:** An internal load balancer is not suitable for exposing the application to the internet, as it routes traffic within the internal network only.

Therefore, the TCP Proxy Load Balancer is the only suitable option because it correctly handles TCP traffic, exposes the application to the internet, and preserves the original client IP, as required.

**Authoritative Links for further research:**

**Google Cloud Load Balancing Overview:** https://cloud.google.com/load-balancing/docs/load-balancing-overview
**TCP Proxy Load Balancer:** https://cloud.google.com/load-balancing/docs/tcp/
**Preserving Client IPs:** https://cloud.google.com/load-balancing/docs/preserving-client-ip

**Question: 199** CertyIQ

You are building a multi-player gaming application that will store game information in a database. As the popularity of the application increases, you are concerned about delivering consistent performance. You need to ensure an optimal gaming performance for global users, without increasing the management complexity. What should you do?

A.Use Cloud SQL database with cross-region replication to store game statistics in the EU, US, and APAC regions.

B.Use Cloud Spanner to store user data mapped to the game statistics.

C.Use BigQuery to store game statistics with a Redis on Memorystore instance in the front to provide global consistency.

D.Store game statistics in a Bigtable database partitioned by username.

**Answer: B**

**Explanation:**

The correct answer is **B. Use Cloud Spanner to store user data mapped to the game statistics.** Here's why:

Cloud Spanner is a globally distributed, scalable, and strongly consistent database service. Its key strength lies in providing transactional consistency across geographical locations, crucial for real-time gaming applications where data integrity is paramount. This global distribution ensures low-latency access for users worldwide, addressing the need for optimal gaming performance. Unlike regional replication (Option A), Spanner's global nature avoids the eventual consistency issues that can arise with cross-region data synchronization. Option A's regional replication may lead to inconsistent data views for different players, disrupting the gaming experience.

BigQuery (Option C) is designed for analytical workloads and isn't suitable for real-time transactions required in a gaming application. Furthermore, relying on Redis for global consistency alongside BigQuery adds management complexity and potential points of failure. Option D, using Bigtable, although scalable, lacks Spanner's strong transactional consistency, making it inappropriate for real-time game state updates that need to occur across many regions. Cloud Spanner's automatic replication and fault tolerance greatly reduces management overhead compared to other solutions, aligning with the requirement to avoid increased complexity. For more information, see Google Cloud's documentation on Cloud Spanner: https://cloud.google.com/spanner/docs and its benefits in gaming https://cloud.google.com/customers/unity-technologies.

**Question: 200** CertyIQ

You are building an application that stores relational data from users. Users across the globe will use this application. Your CTO is concerned about the scaling requirements because the size of the user base is unknown. You need to implement a database solution that can scale with your user growth with minimum configuration changes. Which storage solution should you use?

A.Cloud SQL

B.Firestore

C.Cloud Spanner

D.Bigtable

**Answer: C**

**Explanation:**

The correct answer is **C. Cloud Spanner**. Here's a detailed justification:

Cloud Spanner is Google's globally distributed, scalable, and strongly consistent database service. It's designed to handle large-scale applications with transactional consistency, making it ideal for relational data needing global availability and scalability. Unlike Cloud SQL, which has limitations in horizontal scaling and can be constrained by a single instance or region, Cloud Spanner is built for automatic scaling both vertically (within an instance) and horizontally (across instances) as data and traffic grow. Firestore, while scalable, is a NoSQL document database, not a relational one, making it unsuitable for the given scenario. Bigtable is also a highly scalable NoSQL database designed for massive analytical and operational workloads but is not relational, hence not the best choice for user data requiring SQL queries and relationships. Cloud Spanner's automatic scaling and strong consistency make it the most appropriate solution for a user base with unknown growth and demanding transactional needs, requiring minimal configuration changes to keep up with increasing demands. It eliminates the need for manual sharding and database management overhead, aligning perfectly with the requirement of minimal configuration changes during growth.

**Supporting Links:**

**Google Cloud Spanner Overview:** https://cloud.google.com/spanner/docs/overview
**Cloud Spanner Scalability:** https://cloud.google.com/spanner/docs/scaling
**Cloud Spanner vs. Other Databases:** https://cloud.google.com/spanner/docs/compare
**Cloud SQL Overview:** https://cloud.google.com/sql/docs/overview
**Firestore Overview:** https://firebase.google.com/docs/firestore
**Bigtable Overview:** https://cloud.google.com/bigtable/docs/overview

**Question: 201** CertyIQ

Your company has multiple projects linked to a single billing account in Google Cloud. You need to visualize the costs with specific metrics that should be dynamically calculated based on company-specific criteria. You want to automate the process. What should you do?

A.In the Google Cloud console, visualize the costs related to the projects in the Reports section.

B.In the Google Cloud console, visualize the costs related to the projects in the Cost breakdown section.

C.In the Google Cloud console, use the export functionality of the Cost table. Create a Looker Studio dashboard on top of the CSV export.

D.Configure Cloud Billing data export to BigQuery for the billing account. Create a Looker Studio dashboard on top of the BigQuery export.

**Answer: D**

**Explanation:**

The correct answer is **D: Configure Cloud Billing data export to BigQuery for the billing account. Create a Looker Studio dashboard on top of the BigQuery export.**

Here's why:

**Granular Data:** Direct export of Cloud Billing data to BigQuery provides access to detailed, raw cost and usage data. This level of granularity is crucial for calculating custom metrics based on company-specific criteria. The native Google Cloud console reports (options A and B) offer pre-defined visualizations and lack the flexibility needed for custom calculations.

**Automation:** BigQuery export can be automated, ensuring data is consistently updated. This enables automated dashboard updates, eliminating manual data processing and reporting.

**Scalability:** BigQuery is a scalable data warehouse, capable of handling large volumes of billing data from multiple projects. It efficiently processes complex queries required for custom metrics.

**Custom Calculations:** BigQuery's SQL capabilities allow you to perform intricate calculations and transformations on the exported billing data, implementing your company-specific metrics. This is a crucial aspect that other options don't accommodate.

**Looker Studio Integration:** Looker Studio can directly connect to BigQuery, allowing you to create interactive and dynamic dashboards based on the custom metrics derived from the exported billing data. The dynamic calculation requirement of the question aligns perfectly with Looker Studio's reporting and visualization tools.

**Why other options are not suitable**: Exporting the Cost table as a CSV (Option C) is not ideal for automation and scalability. It would require manual processing, making the process prone to errors and inefficient. Google Cloud console reports lack the ability to create custom metrics, making option A and B unsuitable.

**Authoritative links for further research:**

**Export billing data to BigQuery:** https://cloud.google.com/billing/docs/how-to/export-data-bigquery
**BigQuery Overview:** https://cloud.google.com/bigquery/docs/introduction
**Looker Studio Overview:** https://lookerstudio.google.com/overview

In summary, exporting to BigQuery provides the data granularity, automation capabilities, scalability, and flexibility required for custom calculations and dynamic visualization through Looker Studio, making it the most appropriate solution.

**Question: 202** CertyIQ

You have an application that runs on Compute Engine VM instances in a custom Virtual Private Cloud (VPC). Your company's security policies only allow the use of internal IP addresses on VM instances and do not let VM instances connect to the internet. You need to ensure that the application can access a file hosted in a Cloud Storage bucket within your project. What should you do?

A.Enable Private Service Access on the Cloud Storage Bucket.

B.Add storage.googleapis.com to the list of restricted services in a VPC Service Controls perimeter and add your project to the list of protected projects.

C.Enable Private Google Access on the subnet within the custom VPC.

D.Deploy a Cloud NAT instance and route the traffic to the dedicated IP address of the Cloud Storage bucket.

**Answer: C**

**Explanation:**

The correct answer is **C. Enable Private Google Access on the subnet within the custom VPC.** Here's why:

The scenario describes a need for Compute Engine VMs, residing within a custom VPC with no internet access, to reach a Cloud Storage bucket, also within the same project. These VMs are restricted to internal IP addresses, precluding direct access to Google services over the public internet.

Option C, enabling Private Google Access on the subnet, allows these VMs to communicate with Google APIs and services, including Cloud Storage, using private IP addresses. This feature routes traffic destined for Google services through Google's private network, keeping the data within Google's infrastructure and avoiding the public internet. This satisfies the security requirement of not allowing internet access for the VMs.

Option A, enabling Private Service Access, is used to access services such as Cloud SQL or Memorystore, not Cloud Storage. Option B, while related to security, uses VPC Service Controls to restrict access based on perimeters, not enabling private access. Option D, deploying Cloud NAT, allows internet access for VMs that don't have external IP addresses, but this contradicts the requirement that VMs cannot connect to the internet. Private Google Access provides a secure, private path for communication with Google services as required by the scenario.

**Authoritative Links for further research:**

**Private Google Access:** https://cloud.google.com/vpc/docs/private-google-access
**VPC Service Controls:** https://cloud.google.com/vpc-service-controls
**Cloud NAT:** https://cloud.google.com/nat/docs

**Question: 203** CertyIQ

Your company completed the acquisition of a startup and is now merging the IT systems of both companies. The startup had a production Google Cloud project in their organization. You need to move this project into your organization and ensure that the project is billed to your organization. You want to accomplish this task with minimal effort. What should you do?

A.Use the projects.move method to move the project to your organization. Update the billing account of the project to that of your organization.

B.Ensure that you have an Organization Administrator Identity and Access Management (IAM) role assigned to you in both organizations. Navigate to the Resource Manager in the startup's Google Cloud organization, and drag the project to your company's organization.

C.Create a Private Catalog for the Google Cloud Marketplace, and upload the resources of the startup's production project to the Catalog. Share the Catalog with your organization, and deploy the resources in your company's project.

D.Create an infrastructure-as-code template for all resources in the project by using Terraform, and deploy that template to a new project in your organization. Delete the project from the startup's Google Cloud organization.

**Answer: A**

**Explanation:**

Option A is the correct solution for migrating a Google Cloud project to a new organization with minimal effort. The projects.move method within the Resource Manager API is specifically designed for transferring projects between organizations. This method allows for a direct transfer of the project while maintaining its existing configurations and resources. After the move, updating the billing account to the acquiring organization ensures all project costs are appropriately allocated. Options B, C, and D offer more complex and time-consuming approaches. Option B suggests a drag-and-drop method, which is not a standard feature for project transfers within the Google Cloud Console. Option C involves creating a Private Catalog, a solution meant for sharing reusable assets, not for complete project migrations. Option D involves a complete re-deployment of resources, which is cumbersome and introduces unnecessary risk compared to a direct project move. The projects.move method minimizes disruption and ensures a smoother transition.

Relevant documentation:

Moving projects

projects.move Method
Change a project's billing account

**Question: 204** CertyIQ

All development (dev) teams in your organization are located in the United States. Each dev team has its own Google Cloud project. You want to restrict access so that each dev team can only create cloud resources in the United States (US). What should you do?

A.Create a folder to contain all the dev projects. Create an organization policy to limit resources in US locations.

B.Create an organization to contain all the dev projects. Create an Identity and Access Management (IAM) policy to limit the resources in US regions.

C.Create an Identity and Access Management (IAM) policy to restrict the resources locations in the US. Apply the policy to all dev projects.

D.Create an Identity and Access Management (IAM) policy to restrict the resources locations in all dev projects. Apply the policy to all dev roles.

**Answer: A**

**Explanation:**

**Justification:**

Option A is the correct approach because it leverages Google Cloud's organizational structure and policy enforcement mechanisms for centralized control. Creating a folder to group all development projects allows for the application of organizational policies at a higher level, affecting all projects within that folder. The core component is the organization policy, which is specifically designed to enforce constraints across an organization, including limiting resource locations. This approach enables consistent and scalable restriction of resource creation to the US region. IAM policies, in contrast, focus on controlling access and permissions rather than enforcing resource limitations. While IAM policies can manage who can do what, they don't inherently restrict where resources can be created. Applying an IAM policy to restrict resource locations would require more complex configurations, likely involving conditional roles and permissions, which is not the recommended practice for this specific scenario. Option D is also incorrect as it attempts to use IAM, but applies the restriction to the user instead of the resources. IAM doesn't govern resource location restrictions. Therefore, the combination of a folder for organization and a location-restricting organization policy provides the most efficient and effective means to meet the stated requirement.

**Authoritative Links:**

**Organization Policies:** https://cloud.google.com/resource-manager/docs/organization-policy/overview
**Resource Hierarchy:** https://cloud.google.com/resource-manager/docs/cloud-platform-resource-hierarchy
**IAM Policies vs. Organization Policies:** https://cloud.google.com/resource-manager/docs/access-control-comparison

**Question: 205** CertyIQ

You are configuring Cloud DNS. You want to create DNS records to point home.mydomain.com, mydomain.com, and www.mydomain.com to the IP address of your Google Cloud load balancer. What should you do?

A.Create one CNAME record to point mydomain.com to the load balancer, and create two A records to point WWW and HOME to mydomain.com respectively.

B.Create one CNAME record to point mydomain.com to the load balancer, and create two AAAA records to point WWW and HOME to mydomain.com respectively.

C.Create one A record to point mydomain.com to the load balancer, and create two CNAME records to point

WWW and HOME to mydomain.com respectively.

D.Create one A record to point mydomain.com to the load balancer, and create two NS records to point WWW and HOME to mydomain.com respectively.

**Answer: C**

**Explanation:**

The correct answer is **C**. Here's why:

**A Records:** A records directly map a domain name to an IP address. Since mydomain.com needs to point to the load balancer's IP, we must use an A record for it.
**CNAME Records:** CNAME records create an alias, pointing one domain name to another. For www.mydomain.com and home.mydomain.com to inherit the IP address of the load balancer, we point them to mydomain.com using CNAME records. This establishes a redirection, making both subdomains resolve to the same IP as the main domain.
**Why other options are wrong:** Option A is incorrect because it uses an A record for the subdomains, requiring manual IP address updates should the load balancer IP change, whereas CNAME will follow whatever address is set for the base domain. Option B is incorrect for the same reason as A, and because it mistakenly uses AAAA records which are for IPv6 addresses, not IPv4 addresses. Option D uses NS records which are used for delegating DNS zones not for mapping hostnames to IP addresses.

In essence, this setup ensures efficient DNS management. Only the A record for mydomain.com needs to be modified if the load balancer's IP changes. The CNAME records for the subdomains will automatically update due to following the base record. This approach is scalable and reduces administrative overhead.

**Authoritative Links:**

**Google Cloud DNS Overview:** https://cloud.google.com/dns/docs/overview
**Cloud DNS Records Types:** https://cloud.google.com/dns/docs/records
**Understanding A Records:** https://www.cloudflare.com/learning/dns/dns-records/dns-a-record/
**Understanding CNAME Records:** https://www.cloudflare.com/learning/dns/dns-records/dns-cname-record/

**Question: 206** CertyIQ

You have two subnets (subnet-a and subnet-b) in the default VPC. Your database servers are running in subnet-a. Your application servers and web servers are running in subnet-b. You want to configure a firewall rule that only allows database traffic from the application servers to the database servers. What should you do?

A.•Create service accounts sa-app and sa-db.
•Associate service account sa-app with the application servers and the service account sa-db with the database servers.
•Create an ingress firewall rule to allow network traffic from source service account sa-app to target service account sa-db.

B.•Create network tags app-server and db-server.
•Add the app-server tag to the application servers and the db-server tag to the database servers.
•Create an egress firewall rule to allow network traffic from source network tag app-server to target network tag db-server.

C.•Create a service account sa-app and a network tag db-server.
•Associate the service account sa-app with the application servers and the network tag db-server with the database servers.
•Create an ingress firewall rule to allow network traffic from source VPC IP addresses and target the subnet-a IP addresses.

D.•Create a network tag app-server and service account sa-db.
•Add the tag to the application servers and associate the service account with the database servers.
•Create an egress firewall rule to allow network traffic from source network tag app-server to target service account sa-db.

**Answer: A**

**Explanation:**

The correct answer is **A**. Let's break down why.

The primary goal is to restrict database traffic to only originate from the application servers within a VPC. Options B, C, and D have fundamental issues in how they approach this restriction. Firewall rules in Google Cloud are directional - they are either ingress (incoming) or egress (outgoing). To control traffic to the database servers, we need an ingress rule. B & D attempt to use egress rules, which are for traffic leaving the instances. Option C tries to allow traffic from the entire VPC, and uses a subnet target, making it far too broad. It doesn't target specific instances and doesn't focus on only application server traffic.

Option A correctly utilizes service accounts as the identity for network traffic control. Service accounts provide a more granular level of security than network tags, which are generally used for grouping instances rather than identifying traffic sources. It creates sa-app for the application servers and sa-db for the database servers, enabling precise traffic control. The ingress rule is configured to allow traffic from the sa-app (source) to sa-db (target). This ensures only traffic coming from servers running with the sa-app identity will be allowed to reach database servers running with the sa-db identity. It's a best practice for securing communication within your cloud environment.

Therefore, option A is the only option that correctly implements an ingress rule to filter traffic based on the identity of the source and target using service accounts and provides the most secure way to implement the requested access restriction.

**Further research:**

**Google Cloud Firewall Rules:** https://cloud.google.com/vpc/docs/firewalls
**Service Accounts:** https://cloud.google.com/iam/docs/service-accounts
**Using service accounts with firewall rules:** https://cloud.google.com/vpc/docs/firewalls#sa-target

**Question: 207** CertyIQ

Your team wants to deploy a specific content management system (CMS) solution to Google Cloud. You need a quick and easy way to deploy and install the solution. What should you do?

A.Search for the CMS solution in Google Cloud Marketplace. Use gcloud CLI to deploy the solution.

B.Search for the CMS solution in Google Cloud Marketplace. Deploy the solution directly from Cloud Marketplace.

C.Search for the CMS solution in Google Cloud Marketplace. Use Terraform and the Cloud Marketplace ID to deploy the solution with the appropriate parameters.

D.Use the installation guide of the CMS provider. Perform the installation through your configuration management system.

**Answer: B**

**Explanation:**

Option B is the most appropriate solution because it aligns with Google Cloud's goal of providing a simplified and expedited deployment process for pre-configured applications. Google Cloud Marketplace offers a curated selection of software solutions, including content management systems (CMS), that are packaged for easy deployment. This approach eliminates the need for manual configuration or complex scripting.

Specifically, the Marketplace allows users to directly deploy pre-built images of CMS solutions, often with

recommended settings, through a few clicks. This user-friendly method drastically reduces the effort and time required to get a CMS up and running compared to manually setting up virtual machines and installing the software. The integrated nature of the Marketplace also handles basic setup tasks, such as network configurations.

Option A is incorrect because while the gcloud CLI can be used for deployments, direct deployment from the Cloud Marketplace UI is the more convenient, direct, and intended way to deploy marketplace offerings. Option C, while using Terraform can provide infrastructure as code and potentially integrate with a Cloud Marketplace, it introduces an unnecessary level of complexity for a scenario where the goal is "quick and easy" deployment. Finally, option D introduces more complexity and manual effort than is required, going against the aim of simple and fast deployment. It also assumes the existence of an external configuration management system which might not be present or easy to use.

Therefore, directly deploying from Cloud Marketplace offers the best balance of ease, speed, and reliability for a typical CMS deployment as it eliminates manual configuration and potential errors associated with manual setups.

**Authoritative Links:**

**Google Cloud Marketplace:** https://cloud.google.com/marketplace
**Google Cloud Marketplace Documentation:** https://cloud.google.com/marketplace/docs

**Question: 208** CertyIQ

You are working for a startup that was officially registered as a business 6 months ago. As your customer base grows, your use of Google Cloud increases. You want to allow all engineers to create new projects without asking them for their credit card information. What should you do?

A.Create a Billing account, associate a payment method with it, and provide all project creators with permission to associate that billing account with their projects.

B.Grant all engineers permission to create their own billing accounts for each new project.

C.Apply for monthly invoiced billing, and have a single invoice for the project paid by the finance team.

D.Create a billing account, associate it with a monthly purchase order (PO), and send the PO to Google Cloud.

**Answer: A**

**Explanation:**

The correct answer is **A. Create a Billing account, associate a payment method with it, and provide all project creators with permission to associate that billing account with their projects.**

Here's the justification:

To allow engineers to create projects without needing individual credit card information, the organization needs a centralized billing method. Google Cloud projects must be linked to a billing account to incur charges. Option A fulfills this by establishing a single billing account, which acts as a central repository for payment information. By associating a payment method (like a credit card) to this account, the startup ensures that all cloud usage is billed under a single financial entity. Granting engineers permission to associate their new projects with this pre-configured billing account removes the need for them to enter individual payment details, streamlining project creation and maintaining financial control. This approach adheres to best practices for cloud resource management and financial accountability. Option B is unsuitable because granting individual billing account creation permission is inefficient and lacks financial oversight. Option C is not suitable as it requires a pre-existing billing account linked to a payment method before Google can issue monthly invoices. Similarly, Option D is not suitable as the PO mechanism relies on a pre-existing billing account.

Essentially, Option A achieves centralized billing and management with a single payment method and controlled access to link projects, while the other options are either unsuitable or require additional steps not necessary in the context of the question. Centralized billing offers better cost tracking, budgeting, and financial planning for the startup.

**Authoritative Links:**

**Google Cloud Billing Overview:** https://cloud.google.com/billing/docs/concepts
**Manage Billing Accounts and Payments:** https://cloud.google.com/billing/docs/how-to/manage-billing-account
**Billing Access Control:** https://cloud.google.com/billing/docs/how-to/grant-access

**Question: 209** CertyIQ

Your continuous integration and delivery (CI/CD) server can't execute Google Cloud actions in a specific project because of permission issues. You need to validate whether the used service account has the appropriate roles in the specific project.

What should you do?

A.Open the Google Cloud console, and check the Identity and Access Management (IAM) roles assigned to the service account at the project or inherited from the folder or organization levels.

B.Open the Google Cloud console, and check the organization policies.

C.Open the Google Cloud console, and run a query to determine which resources this service account can access.

D.Open the Google Cloud console, and run a query of the audit logs to find permission denied errors for this service account.

**Answer: A**

**Explanation:**

The correct answer is **A** because it directly addresses the problem of verifying service account permissions. To resolve permission issues with a CI/CD server, you need to identify the specific roles assigned to the service account in the target project. Option A, checking IAM roles, allows you to see the explicitly granted roles to the service account at the project level and those inherited from folders or the organization. IAM (Identity and Access Management) is the fundamental Google Cloud service for managing access control and permissions.

Option B, checking organization policies, is incorrect because organization policies enforce constraints on resource usage and configurations, but they do not directly show the permissions granted to individual service accounts. Option C, running a query to determine resource access, is also incorrect. While useful for understanding what resources a service account can access, this is a reactive step. First, the correct permissions need to be determined. Option D, checking audit logs for permission denied errors, is a reactive method useful for diagnosing issues, but it doesn't show what permissions are actually granted. A is the best validation step.

Therefore, examining the IAM roles via the Google Cloud Console, as described in option A, is the most efficient way to determine if the service account has the correct permissions required for executing Google Cloud actions within the specified project. This approach allows for direct inspection of assigned roles and quick identification of missing or insufficient permissions, aiding in rapid troubleshooting and remediation of CI/CD pipeline issues. By looking directly at the IAM settings, one can quickly determine if the service account is lacking the necessary privileges.

For further information on IAM and service accounts, refer to the official Google Cloud documentation:

**IAM Overview:** https://cloud.google.com/iam/docs/overview
**Service Accounts:** https://cloud.google.com/iam/docs/service-accounts

**Question: 210** CertyIQ

Your team is using Linux instances on Google Cloud. You need to ensure that your team logs in to these instances in the most secure and cost efficient way. What should you do?

A.Attach a public IP to the instances and allow incoming connections from the internet on port 22 for SSH.

B.Use the gcloud compute ssh command with the --tunnel-through-iap flag. Allow ingress traffic from the IP range 35.235.240.0/20 on port 22.

C.Use a third party tool to provide remote access to the instances.

D.Create a bastion host with public internet access. Create the SSH tunnel to the instance through the bastion host.

**Answer: B**

**Explanation:**

The correct answer is **B: Use the gcloud compute ssh command with the --tunnel-through-iap flag. Allow ingress traffic from the IP range 35.235.240.0/20 on port 22.**

Here's why:

Option B leverages Identity-Aware Proxy (IAP) for secure SSH access. IAP enforces access policies before allowing connections, ensuring only authorized users can reach the instances. The gcloud compute ssh --tunnel-through-iap command automatically sets up this secure tunnel. The IP range 35.235.240.0/20 is Google's IAP range, allowing IAP to establish connections to the instances. This approach avoids exposing instances directly to the public internet, enhancing security. It's also cost-effective as it doesn't require maintaining a separate bastion host. Option A is highly insecure, as opening port 22 directly to the internet exposes the instances to potential attacks. Option C relies on a third-party tool which adds unnecessary complexity and potentially higher costs. Option D introduces a bastion host which while a common approach, adds complexity and costs for managing an additional instance. IAP provides a simpler, more secure, and cost-effective solution. Therefore, Option B is the most suitable approach given the requirement for security and cost efficiency.

**Authoritative links:**

**Google Cloud IAP Documentation:** https://cloud.google.com/iap/docs/
**gcloud compute ssh command:** https://cloud.google.com/sdk/gcloud/reference/compute/ssh
**Using IAP for SSH:** https://cloud.google.com/iap/docs/using-tcp-forwarding
**IAP IP Ranges:** https://cloud.google.com/iap/docs/using-tcp-forwarding#firewall

**Question: 211** CertyIQ

An external member of your team needs list access to compute images and disks in one of your projects. You want to follow Google-recommended practices when you grant the required permissions to this user. What should you do?

A.Create a custom role, and add all the required compute.disks.list and compute.images.list permissions as includedPermissions. Grant the custom role to the user at the project level.

B.Create a custom role based on the Compute Image User role. Add the compute.disks.list to the includedPermissions field. Grant the custom role to the user at the project level.

C.Create a custom role based on the Compute Storage Admin role. Exclude unnecessary permissions from the

custom role. Grant the custom role to the user at the project level.

D.Grant the Compute Storage Admin role at the project level.

**Answer: A**

**Explanation:**

**Justification for Answer A:**

Option A is the most secure and adheres to the principle of least privilege, a core security concept in cloud computing. Creating a custom role and explicitly granting only compute.disks.list and compute.images.list permissions provides the user with the exact access needed — nothing more, nothing less. This minimizes the potential attack surface and reduces the risk of unauthorized actions.

Option B is problematic because it starts from the Compute Image User role, which might include permissions beyond just listing images (e.g., using them to create instances). By building from a broader predefined role you're not minimizing permissions. While adding compute.disks.list would seemingly satisfy the requirements, starting from a more limited base and adding only the needed permissions, as done in option A, is a better approach.

Option C and D propose granting much broader permissions. Compute Storage Admin is a very powerful role, which grants wide ranging abilities. Option C's suggestion to remove unnecessary permissions is complex and prone to human error. Directly granting the full Compute Storage Admin role in option D is a major security risk. It grants the external user far more power than they need, such as the ability to create, delete and modify disks and images, which is not required for mere listing.

Therefore, A provides a minimal and specific permission set, meeting the requirements while following security best practices of granting only necessary permissions.

**Authoritative Links:**

**IAM Overview:** https://cloud.google.com/iam/docs/overview - This document explains core IAM concepts, including roles, permissions, and the principle of least privilege.
**Understanding Roles:** https://cloud.google.com/iam/docs/understanding-roles - This explains predefined and custom roles in greater detail.
**Creating and Managing Custom Roles:** https://cloud.google.com/iam/docs/creating-custom-roles - Here you can explore how to create custom roles in GCP.
**Compute Engine Permissions:** https://cloud.google.com/compute/docs/access/iam - This article outlines the various permissions available for Compute Engine resources, specifically disk and image permissions.

**Question: 212** CertyIQ

You are running a web application on Cloud Run for a few hundred users. Some of your users complain that the initial web page of the application takes much longer to load than the following pages. You want to follow Google's recommendations to mitigate the issue. What should you do?

A.Set the minimum number of instances for your Cloud Run service to 3.

B.Set the concurrency number to 1 for your Cloud Run service.

C.Set the maximum number of instances for your Cloud Run service to 100.

D.Update your web application to use the protocol HTTP/2 instead of HTTP/1.1.

**Answer: A**

**Explanation:**

The correct answer is A, setting the minimum number of Cloud Run instances to 3. This addresses the "cold start" latency issue reported by users when initially accessing the web application. Cloud Run, a serverless platform, scales instances based on incoming traffic. When no requests are active, instances can scale down to zero. This process of scaling up from zero, termed a "cold start", incurs a delay. By setting a minimum number of instances, Cloud Run keeps those instances active even when traffic is low or nonexistent. These "warm" instances can immediately serve requests, eliminating the startup delay and providing faster initial page load times. Option B, setting concurrency to 1, would actually worsen performance by processing only one request per instance at a time. Option C, setting a maximum, does not address the cold start. Option D, while HTTP/2 can improve performance, is not the primary issue in a cold start scenario and is not Google's recommended solution for this specific problem. Maintaining a minimum number of instances is a key strategy for optimizing Cloud Run application performance, particularly for the initial user experience. Google's documentation emphasizes leveraging minimum instances to minimize cold starts, aligning with the provided answer.

https://cloud.google.com/run/docs/configuring/min-instanceshttps://cloud.google.com/blog/products/serverless/understanding-cold-starts-on-cloud-run

**Question: 213** CertyIQ

You are building a data lake on Google Cloud for your Internet of Things (IoT) application. The IoT application has millions of sensors that are constantly streaming structured and unstructured data to your backend in the cloud. You want to build a highly available and resilient architecture based on Google-recommended practices. What should you do?

A.Stream data to Pub/Sub, and use Dataflow to send data to Cloud Storage.

B.Stream data to Pub/Sub, and use Storage Transfer Service to send data to BigQuery.

C.Stream data to Dataflow, and use Dataprep by Trifacta to send data to Bigtable.

D.Stream data to Dataflow, and use Storage Transfer Service to send data to BigQuery.

**Answer: A**

**Explanation:**

The correct answer is A. Streaming data to Pub/Sub and using Dataflow to send it to Cloud Storage is the most appropriate approach for building a highly available and resilient IoT data lake. Pub/Sub acts as a reliable, asynchronous message bus, decoupling data producers (IoT sensors) from consumers (processing pipelines). This decoupling provides fault tolerance; if a consumer fails, messages are still safely stored in Pub/Sub until they can be processed. Dataflow is a fully managed, serverless stream processing service ideally suited for transforming and loading large volumes of streaming data. It handles scaling and fault tolerance automatically.

Cloud Storage is a scalable and durable object storage service, perfect for storing the raw data coming from the IoT devices, forming the basis of the data lake. Options B, C, and D are not ideal. While Pub/Sub is a good choice, Storage Transfer Service is used for batch data transfers, unsuitable for real-time streaming (B). Dataflow is a good choice, Dataprep is an interactive data preparation service not designed for continuous streaming (C), and neither is Storage Transfer Service (D). BigQuery, while excellent for analysis, is not a suitable initial storage location for raw data in a data lake (B, D). Bigtable is a NoSQL database useful for operational workloads, not an ideal place for general data lake storage (C). Thus, option A leverages the strengths of Pub/Sub, Dataflow, and Cloud Storage to establish a resilient, scalable, and highly available data pipeline for an IoT data lake.

**Authoritative Links:**

**Pub/Sub:** https://cloud.google.com/pubsub

**Dataflow:** https://cloud.google.com/dataflow
**Cloud Storage:** https://cloud.google.com/storage
**Data Lake Overview:** https://cloud.google.com/solutions/data-lake
**IoT on Google Cloud:** https://cloud.google.com/solutions/iot

**Question: 214** CertyIQ

You are running out of primary internal IP addresses in a subnet for a custom mode VPC. The subnet has the IP range 10.0.0.0/20, and the IP addresses are primarily used by virtual machines in the project. You need to provide more IP addresses for the virtual machines. What should you do?

A.Add a secondary IP range 10.1.0.0/20 to the subnet.
B.Change the subnet IP range from 10.0.0.0/20 to 10.0.0.0/18.
C.Change the subnet IP range from 10.0.0.0/20 to 10.0.0.0/22.
D.Convert the subnet IP range from IPv4 to IPv6.

**Answer: B**

**Explanation:**

The correct solution is **B. Change the subnet IP range from 10.0.0.0/20 to 10.0.0.0/18.**

Here's why: A /20 subnet provides 4096 IP addresses (2^12), while a /18 subnet provides 16,384 IP addresses (2^14). Subnet masks determine the number of available IP addresses. The smaller the mask number (e.g., /18 versus /20), the larger the IP range and hence the more addresses available. When running out of primary internal IP addresses within a subnet, you need to expand the subnet's address range. Modifying the subnet mask to a smaller value effectively increases the address space the subnet can use.

Option A, adding a secondary IP range (10.1.0.0/20), is not the best practice. While technically feasible, it introduces complexity and might not address the core issue of primary IP exhaustion efficiently. Secondary IP ranges are better suited for specific use cases like alias IPs and not meant to act as primary IP expansion mechanisms.

Option C, reducing the subnet IP range to /22, is incorrect because it decreases the available IP addresses further. A smaller subnet mask number means more IP addresses, not fewer.

Option D, converting from IPv4 to IPv6, is also not the correct approach in this specific scenario. While IPv6 provides an enormous address space, it requires a more complex transition that is unnecessary given that the need is only to increase primary IP addresses, not to change the IP protocol. Also, the question specifies a need for "primary" IP addresses, which are still IPv4 addresses in a hybrid IPv4/IPv6 environment for Google Cloud VMs.

By changing the subnet's IP range to a larger range (from /20 to /18), it ensures that the existing resources can remain in the same subnet without the complications of a secondary range or protocol switch, while simultaneously providing the needed additional IP addresses.

**Authoritative links:**

**Google Cloud VPC Subnets:** https://cloud.google.com/vpc/docs/vpc
**Google Cloud VPC Subnet Ranges:** https://cloud.google.com/vpc/docs/configure-subnets
**CIDR Notation:** https://en.wikipedia.org/wiki/Classless_Inter-Domain_Routing

**Question: 215** CertyIQ

Your company requires all developers to have the same permissions, regardless of the Google Cloud project they are working on. Your company's security policy also restricts developer permissions to Compute Engine, Cloud Functions, and Cloud SQL. You want to implement the security policy with minimal effort. What should you do?

A.•Create a custom role with Compute Engine, Cloud Functions, and Cloud SQL permissions in one project within the Google Cloud organization.
•Copy the role across all projects created within the organization with the gcloud iam roles copy command.
•Assign the role to developers in those projects.

B.•Add all developers to a Google group in Google Groups for Workspace.
•Assign the predefined role of Compute Admin to the Google group at the Google Cloud organization level.

C.•Add all developers to a Google group in Cloud Identity.
•Assign predefined roles for Compute Engine, Cloud Functions, and Cloud SQL permissions to the Google group for each project in the Google Cloud organization.

D.•Add all developers to a Google group in Cloud Identity.
•Create a custom role with Compute Engine, Cloud Functions, and Cloud SQL permissions at the Google Cloud organization level.
•Assign the custom role to the Google group.

**Answer: D**

**Explanation:**

Here's a detailed justification for why option D is the correct approach:

Option D offers the most efficient and scalable method to manage developer permissions consistently across all Google Cloud projects. By using a Google group in Cloud Identity, we centralize user management, allowing us to easily add or remove developers without modifying individual project permissions. Creating a custom role at the organization level ensures that this role is available and consistent across all projects. The custom role encapsulates the specific permissions required (Compute Engine, Cloud Functions, and Cloud SQL), adhering to the company's security policy. Finally, assigning this custom role to the Google group grants all group members, i.e., all developers, the defined permissions across every project under the organization. This avoids manual permission management for each project and aligns with best practices for centralized IAM (Identity and Access Management). This method is far less cumbersome than copying roles between projects (Option A) and is more secure than assigning overly broad predefined roles (Option B). Option C is also less scalable since it requires defining roles for each project individually.

Using a Google Group simplifies adding and removing developers; adding or removing them from the group automatically manages their permissions. A custom role ensures that only the required permissions are granted, adhering to the principle of least privilege. Centralizing the role at the organization level promotes consistency and avoids discrepancies. This approach minimizes effort and potential errors and maintains a uniform permission model throughout the Google Cloud environment.

**Authoritative Links for Further Research:**

**IAM Overview:** https://cloud.google.com/iam/docs/overview
**Creating and Managing Custom Roles:** https://cloud.google.com/iam/docs/understanding-custom-roles
**Granting, Changing, and Revoking Access to Resources:** https://cloud.google.com/iam/docs/granting-changing-revoking-access
**Groups in Cloud Identity:** https://cloud.google.com/identity/docs/groups

**Question: 216** CertyIQ

You are working for a hospital that stores its medical images in an on-premises data room. The hospital wants to use Cloud Storage for archival storage of these images. The hospital wants an automated process to upload any new medical images to Cloud Storage. You need to design and implement a solution. What should you do?

A.Create a Pub/Sub topic, and enable a Cloud Storage trigger for the Pub/Sub topic. Create an application that

sends all medical images to the Pub/Sub topic.

B.Create a script that uses the gcloud storage command to synchronize the on-premises storage with Cloud Storage, Schedule the script as a cron job.

C.Create a Pub/Sub topic, and create a Cloud Function connected to the topic that writes data to Cloud Storage. Create an application that sends all medical images to the Pub/Sub topic.

D.In the Google Cloud console, go to Cloud Storage. Upload the relevant images to the appropriate bucket.

**Answer: B**

**Explanation:**

Here's a detailed justification for why option B is the most suitable answer, along with supporting concepts and links:

**Justification:**

Option B, using gcloud storage for synchronization and scheduling with a cron job, is the most straightforward and efficient approach for this scenario. The core requirement is to automatically upload new medical images to Cloud Storage for archival, and gcloud storage provides a simple and direct way to achieve this. The gcloud storage sync command can mirror the on-premises directory to a Cloud Storage bucket, only uploading new or modified files. Scheduling this script as a cron job ensures that the synchronization process happens regularly and automatically. This avoids the complexity of implementing pub/sub, cloud functions, and manual console uploads, making it a highly practical and cost-effective choice.

**Why other options are less ideal:**

**Option A:** Using Pub/Sub as a trigger for Cloud Storage is counterintuitive. Pub/Sub is primarily for asynchronous message passing, not direct file transfers. This option adds an unnecessary layer of complexity, and is not efficient for a bulk transfer of image files.
**Option C:** While Cloud Functions can handle data manipulation, using them solely to transfer files from Pub/Sub to Cloud Storage is overkill. Similar to option A, it introduces extra layers of complexity and potential failure points without significant benefit. It also adds costs associated with Cloud Functions execution.
**Option D:** Manually uploading via the Google Cloud console is not an automated solution and is impractical for a continuous flow of new images. This would require manual intervention for every upload, defeating the requirement for automation.

**Key Cloud Computing Concepts:**

**Cloud Storage:** Object storage service for storing unstructured data, well-suited for images and archives.
**gcloud storage:** Google Cloud CLI tool to manage Cloud Storage resources, including copying and synchronizing files.
**Cron job:** Scheduling utility to execute commands or scripts at specific intervals.
**Synchronization:** Keeping two storage locations consistent by automatically copying changes from one to the other.

**Authoritative Links:**

**gcloud storage documentation:** https://cloud.google.com/sdk/gcloud/reference/storage
**Cloud Storage documentation:** https://cloud.google.com/storage/docs
**Cron jobs on Compute Engine:** https://cloud.google.com/compute/docs/instances/schedule-instance-start-stop (While this is for VM scheduling, it illustrates the principle of cron job usage in GCP)

In summary, option B provides a simple, automated, cost-effective and reliable solution for this specific scenario.

**Question: 217** CertyIQ

Your company has an internal application for managing transactional orders. The application is used exclusively by employees in a single physical location. The application requires strong consistency, fast queries, and ACID guarantees for multi-table transactional updates. The first version of the application is implemented in PostgreSQL, and you want to deploy it to the cloud with minimal code changes. Which database is most appropriate for this application?

A.Bigtable

B.BigQuery

C.Cloud SQL

D.Firestore

**Answer: C**

**Explanation:**

The most appropriate database for this application is **C. Cloud SQL**. Cloud SQL is Google Cloud's fully managed relational database service, offering managed instances of popular RDBMS engines like PostgreSQL, MySQL, and SQL Server. Given the requirement for strong consistency, fast queries, and ACID guarantees for multi-table transactions, a traditional relational database is best suited. This is because RDBMS are specifically designed to handle such workloads efficiently, providing transactional integrity and the ability to perform complex joins across multiple tables. The application already uses PostgreSQL, so Cloud SQL with PostgreSQL minimizes code changes, as it's a direct compatible service. Options A (Bigtable) and D (Firestore) are NoSQL databases, unsuitable for transactional workloads requiring ACID properties. Bigtable is optimized for high throughput and scalability for large datasets, but lacks robust support for transactions. Similarly, Firestore excels at document-based data storage and is well-suited for applications needing real-time synchronization, but isn't optimized for multi-table transactions and complex queries. BigQuery (option B), is for analytical workloads and large data processing, not for operational systems needing real-time strong consistency and transactional guarantees.

**Further Research:**

**Google Cloud SQL:** https://cloud.google.com/sql
**ACID Properties:** https://en.wikipedia.org/wiki/ACID
**Relational vs. NoSQL Databases:** https://www.mongodb.com/nosql-explained

**Question: 218** CertyIQ

Your company runs one batch process in an on-premises server that takes around 30 hours to complete. The task runs monthly, can be performed offline, and must be restarted if interrupted. You want to migrate this workload to the cloud while minimizing cost. What should you do?

A.Create an Instance Template with Spot VMs On. Create a Managed Instance Group from the template and adjust Target CPU Utilization. Migrate the workload.

B.Migrate the workload to a Compute Engine VM. Start and stop the instance as needed.

C.Migrate the workload to a Google Kubernetes Engine cluster with Spot nodes.

D.Migrate the workload to a Compute Engine Spot VM.

**Answer: B**

**Explanation:**

The correct answer is **B. Migrate the workload to a Compute Engine VM. Start and stop the instance as needed.** This is the most cost-effective approach for a batch process with the given constraints.

Here's why:

**Batch Processing Nature:** The workload is described as a monthly batch process, meaning it runs to completion once a month without needing continuous operation. This makes it ideal for temporary compute resources.
**Cost Minimization:** Option B utilizes the pay-per-use model of Compute Engine. Starting the VM only when needed (for 30 hours monthly) and stopping it immediately after completion avoids incurring costs during the remaining time. This offers significant cost savings compared to running VMs continuously.
**Simplicity:** This approach is straightforward to implement. It avoids the complexities of managing instance groups, Kubernetes clusters, and spot VMs (which can be preempted).
**Restartability:** Since the VM can be started and stopped, the process can be restarted manually or through simple scripting if it gets interrupted.
**Spot VMs (Options A, C, D):** Spot VMs, while cost-effective, are not suitable for this case. Spot VMs are prone to preemption (being terminated by Google with short notice if demand rises). Since the task takes 30 hours to complete, the likelihood of being preempted before completion is high, necessitating a restart. Managed Instance Groups (option A) are not needed, since only one instance is needed. Kubernetes (option C) also brings unnecessary management overhead, since it is designed for microservices.

In summary, option B offers the perfect blend of cost-effectiveness and simplicity for a monthly, interruptible batch processing job by leveraging the on-demand nature of compute engine VMs and is a more appropriate solution than using spot VMs or container services.**Authoritative Links:**

**Google Cloud Compute Engine Pricing:** https://cloud.google.com/compute/pricing
**Compute Engine start/stop Instances:** https://cloud.google.com/compute/docs/instances/start-stop-instance
**Spot VMs:**https://cloud.google.com/compute/docs/instances/spot

**Question: 219** CertyIQ

You are planning to migrate the following on-premises data management solutions to Google Cloud:

•One MySQL cluster for your main database
•Apache Kafka for your event streaming platform
•One Cloud SQL for PostgreSQL database for your analytical and reporting needs

You want to implement Google-recommended solutions for the migration. You need to ensure that the new solutions provide global scalability and require minimal operational and infrastructure management. What should you do?

A.Migrate from MySQL to Cloud SQL, from Kafka to Pub/Sub, and from Cloud SQL for PostgreSQL to BigQuery.

B.Migrate from MySQL to Cloud Spanner, from Kafka to Pub/Sub, and from Cloud SQL for PostgreSQL to BigQuery.

C.Migrate from MySQL to Cloud Spanner, from Kafka to Memorystore, and from Cloud SQL for PostgreSQL to Cloud SQL.

D.Migrate from MySQL to Cloud SQL, from Kafka to Memorystore, and from Cloud SQL for PostgreSQL to Cloud SQL.

**Answer: B**

**Explanation:**

The correct answer is B. Here's why:

**MySQL to Cloud Spanner:** Cloud Spanner is Google Cloud's globally distributed, scalable, and strongly consistent database service. It's designed for mission-critical applications needing high availability and scalability, aligning with the requirement for global scalability and minimal management, making it a superior choice over Cloud SQL for MySQL which is regional. https://cloud.google.com/spanner/docs/overview

**Apache Kafka to Pub/Sub:** Pub/Sub is Google Cloud's fully managed, globally scalable messaging service. It's a serverless solution that eliminates the operational burden of managing Kafka infrastructure. Pub/Sub is ideal for real-time event streaming, matching the functionality of Kafka while simplifying operations, addressing the requirement for minimal infrastructure management.
https://cloud.google.com/pubsub/docs/overview

**Cloud SQL for PostgreSQL to BigQuery:** BigQuery is Google Cloud's fully managed, serverless data warehouse service. It's built for analytical workloads and reporting, which is the primary purpose of the PostgreSQL database. BigQuery provides massive scalability and performance for analytics, while requiring no infrastructure management on the user's part, perfectly aligning with the prompt's requirements.
https://cloud.google.com/bigquery/docs/introduction

Option A is incorrect because while Cloud SQL is simpler than managing MySQL instances, it's not designed for global scalability the way Cloud Spanner is. Option C is incorrect because Memorystore is a caching service, not an event streaming service like Kafka. Option D is also incorrect for similar reasons as A and C; it fails to suggest solutions that meet the scalability and minimal management demands of the scenario.

**Question: 220** CertyIQ

During a recent audit of your existing Google Cloud resources, you discovered several users with email addresses outside of your Google Workspace domain. You want to ensure that your resources are only shared with users whose email addresses match your domain. You need to remove any mismatched users, and you want to avoid having to audit your resources to identify mismatched users. What should you do?

A.Create a Cloud Scheduler task to regularly scan your projects and delete mismatched users.

B.Create a Cloud Scheduler task to regularly scan your resources and delete mismatched users.

C.Set an organizational policy constraint to limit identities by domain to automatically remove mismatched users.

D.Set an organizational policy constraint to limit identities by domain, and then retroactively remove the existing mismatched users

**Answer: D**

**Explanation:**

The correct answer is **D: Set an organizational policy constraint to limit identities by domain, and then retroactively remove the existing mismatched users.**

Here's the justification:

Organizational policies in Google Cloud provide centralized control over resource usage and configurations. The "constraints/iam.allowedPolicyMemberDomains" constraint specifically addresses the issue of limiting identities by domain. By setting this constraint at the organization level, you ensure that only users with email addresses matching your defined domain can be granted permissions on any resources within the organization, including existing and future ones. This action satisfies the requirement of restricting access to users within your Google Workspace domain.

The key part of option D that makes it superior is the "retroactively remove the existing mismatched users." The constraint, when applied, does not automatically remove existing users with mismatched email addresses. While the policy will prevent new mismatched users from gaining access, the previously granted access must be addressed to maintain compliance. Therefore, a secondary action is required to identify and remove previously granted access to those users. Options A and B are incorrect because these involve complex workarounds using Cloud Scheduler to continuously check for and remove such users and are not the most effective solution given the available organizational policy. Option C is incorrect because the domain limit

constraint, by itself, does not remove existing mismatched users.

Therefore, setting the organizational policy and performing retroactive cleanup of existing users is the optimal approach. It provides both proactive prevention and remediation of the current situation as requested in the question and fulfills the requirement to avoid future audits.

Relevant Links:

Organizational policy constraints
Understanding identity access management
iam.allowedPolicyMemberDomains constraint

**Question: 221** CertyIQ

Your application is running on Google Cloud in a managed instance group (MIG). You see errors in Cloud Logging for one VM that one of the processes is not responsive. You want to replace this VM in the MIG quickly. What should you do?

A.Use the gcloud compute instances update command with a REFRESH action for the VM.
B.Use the gcloud compute instance-groups managed recreate-instances command to recreate the VM.
C.Select the MIG from the Compute Engine console and, in the menu, select Replace VMs.
D.Update and apply the instance template of the MIG.

**Answer: B**

**Explanation:**

The correct answer is **B. Use the gcloud compute instance-groups managed recreate-instances command to recreate the VM.** This is the most efficient way to replace a single faulty VM within a managed instance group (MIG) while minimizing disruption. MIGs are designed for high availability and self-healing capabilities, automatically managing instances according to defined policies. Directly targeting a specific VM for recreation, as option B does, allows for a focused replacement of the problematic instance. Option A, using gcloud compute instances update with a REFRESH action, does not initiate a full recreation of the VM and may not resolve the underlying issue. Option C, "Replace VMs" in the Compute Engine console, is not a standard option provided in Google Cloud for directly targeting a single instance for recreation; it's often linked to a rolling update process. While Option D, updating the instance template, is necessary for broad changes across the MIG, it would trigger a rolling update, affecting all instances and not providing an immediate solution for a single faulty VM. Recreating an individual instance through gcloud compute instance-groups managed recreate-instances leverages the MIG's ability to quickly bring up a new instance based on the current template while decommissioning the faulty one, ensuring application resilience and quick recovery. This operation respects the group's defined configuration, ensuring the new instance is configured correctly.

Relevant Documentation:

**Recreating instances in a MIG:** https://cloud.google.com/compute/docs/instance-groups/recreating-instances-in-managed-instance-groups
**gcloud compute instance-groups managed recreate-instances:**
https://cloud.google.com/sdk/gcloud/reference/compute/instance-groups/managed/recreate-instances

**Question: 222** CertyIQ

You want to permanently delete a Pub/Sub topic managed by Config Connector in your Google Cloud project. What should you do?

A.Use kubectl to create the label deleted-by-cnrm and to change its value to true for the topic resource.

B.Use kubectl to delete the topic resource.

C.Use gcloud CLI to delete the topic.

D.Use gcloud CLI to update the topic label managed-by-cnrm to false.

**Answer: B**

**Explanation:**

The correct answer is **B. Use kubectl to delete the topic resource.** Here's why:

Config Connector manages Google Cloud resources as Kubernetes resources. When a Pub/Sub topic is managed by Config Connector, it is represented by a Kubernetes custom resource definition (CRD). Therefore, to delete the topic, you must interact with Kubernetes using kubectl.

Deleting a resource managed by Config Connector through kubectl will, in turn, instruct Config Connector to delete the corresponding Google Cloud resource. Option A is incorrect because adding a label doesn't trigger deletion. Option C is incorrect because gcloud CLI operations won't directly affect resources managed by Config Connector. Option D is also incorrect because changing a Config Connector management label will only change whether or not it is managed but not delete the underlying resource.

Using kubectl delete on the relevant Pub/Sub topic resource, therefore, is the intended method for permanently deleting a resource that is under Config Connector management. This adheres to the principle of managing infrastructure as code, where Kubernetes and kubectl become the single point of interaction for resource lifecycle management. Deleting the resource in the google cloud console would not remove the kubernetes definition. Using kubectl ensures both the Kubernetes definition and the underlying Pub/Sub topic are removed.

**Further Research:**

**Config Connector Documentation:** https://cloud.google.com/config-connector/docs - The official documentation for understanding how Config Connector manages Google Cloud resources.
**Kubectl Documentation:** https://kubernetes.io/docs/reference/kubectl/ - Details about the kubectl command-line tool.
**Config Connector Deletion:** https://cloud.google.com/config-connector/docs/how-to/delete-resources - Specifically covers the deletion of resources managed by Config Connector.

**Question: 223** CertyIQ

Your company is using Google Workspace to manage employee accounts. Anticipated growth will increase the number of personnel from 100 employees to 1,000 employees within 2 years. Most employees will need access to your company's Google Cloud account. The systems and processes will need to support 10x growth without performance degradation, unnecessary complexity, or security issues. What should you do?

A.Migrate the users to Active Directory. Connect the Human Resources system to Active Directory. Turn on Google Cloud Directory Sync (GCDS) for Cloud Identity. Turn on Identity Federation from Cloud Identity to Active Directory.

B.Organize the users in Cloud Identity into groups. Enforce multi-factor authentication in Cloud Identity.

C.Turn on identity federation between Cloud Identity and Google Workspace. Enforce multi-factor authentication for domain wide delegation.

D.Use a third-party identity provider service through federation. Synchronize the users from Google Workplace to the third-party provider in real time.

**Answer: C**

**Explanation:**

The correct answer is **C. Turn on identity federation between Cloud Identity and Google Workspace. Enforce multi-factor authentication for domain wide delegation.**

Here's a detailed justification:

Option C leverages the existing Google Workspace setup and extends it to Google Cloud Platform (GCP) through Cloud Identity. This approach avoids the unnecessary complexity and potential migration issues associated with moving to a different directory service like Active Directory (Option A). Identity federation allows users to utilize their existing Google Workspace credentials for GCP access, streamlining the login process and reducing administrative overhead. This directly addresses the scalability needs of the company's projected growth since authentication is handled by the federated system.

Enforcing multi-factor authentication (MFA) for domain-wide delegation adds a crucial security layer, safeguarding access to sensitive resources. Domain-wide delegation allows specific service accounts to act on behalf of all users in the domain. This means that a compromised delegated service account could access any user data in the domain. Therefore, enforcing MFA provides necessary protection. Option B, while focusing on groups and MFA, doesn't directly address how existing Google Workspace accounts gain access to GCP. Option D adds the overhead of introducing a third-party system, which doesn't align with the goal of minimizing complexity. Additionally, synchronizing users in real-time between Google Workspace and a third-party provider can be complex, error-prone, and not the optimal solution when Cloud Identity is available. Cloud Identity is designed to be used directly with Google Workspace accounts for a seamless integration.

By utilizing federation and enforcing MFA, option C offers a scalable, secure, and efficient approach to managing user access across both Google Workspace and Google Cloud, which perfectly aligns with the requirements outlined in the scenario.

**Authoritative Links for Further Research:**

**Google Cloud Identity Federation:** https://cloud.google.com/identity/docs/federation
**Google Cloud Identity and Access Management (IAM):** https://cloud.google.com/iam/docs
**Multi-Factor Authentication in Google Workspace:** https://support.google.com/a/answer/6262289
**Domain-wide Delegation:** https://developers.google.com/identity/protocols/oauth2/service-account#delegatingauthority

**Question: 224** CertyIQ

You want to host your video encoding software on Compute Engine. Your user base is growing rapidly, and users need to be able to encode their videos at any time without interruption or CPU limitations. You must ensure that your encoding solution is highly available, and you want to follow Google-recommended practices to automate operations. What should you do?

A.Deploy your solution on multiple standalone Compute Engine instances, and increase the number of existing instances when CPU utilization on Cloud Monitoring reaches a certain threshold.

B.Deploy your solution on multiple standalone Compute Engine instances, and replace existing instances with high-CPU instances when CPU utilization on Cloud Monitoring reaches a certain threshold.

C.Deploy your solution to an instance group, and increase the number of available instances whenever you see high CPU utilization in Cloud Monitoring.

D.Deploy your solution to an instance group, and set the autoscaling based on CPU utilization.

**Answer: D**

**Explanation:**

The correct answer is **D: Deploy your solution to an instance group, and set the autoscaling based on CPU**

**utilization.** Here's why:

**Instance Groups for High Availability:** Instance groups (both managed and unmanaged) are designed for managing multiple identical VMs, improving availability and resilience. This aligns with the requirement of ensuring the encoding solution is highly available.
**Autoscaling for Dynamic Scaling:** Autoscaling automatically adjusts the number of VM instances in a group based on defined metrics, such as CPU utilization. This directly addresses the need to handle rapid user growth without manual intervention or interruptions. When CPU utilization increases, more instances are added, and when utilization decreases, instances are removed.
**Google-Recommended Practice:** Utilizing instance groups and autoscaling is a core best practice recommended by Google for building scalable and resilient applications on Compute Engine. This enables automatic scaling based on demand, aligning with the prompt's requirement for automation.
**Inefficiency of Standalone Instances (A & B):** Options A and B suggest standalone instances, which are not ideal for scaling or high availability. They involve manual scaling or replacement, which is not automated and does not provide resilience against a single instance failure. Scaling based on standalone instances is cumbersome and inefficient, especially when rapid scaling is needed.
**Dynamic Resource Allocation:** Unlike option B, which proposes replacing existing instances with higher-CPU instances when usage is high, autoscaling dynamically adds and removes instances based on actual utilization without requiring pre-configured size upgrades. This avoids paying for unnecessary compute resources.

Therefore, deploying the video encoding software to an instance group with autoscaling based on CPU utilization is the most suitable approach, fulfilling the requirements of high availability, automated operations, and scalability according to Google's recommended practices.

**Further Research:**

**Google Cloud Documentation on Instance Groups:** https://cloud.google.com/compute/docs/instance-groups
**Google Cloud Documentation on Autoscaling:** https://cloud.google.com/compute/docs/autoscaling
**Google Cloud Documentation on Best practices for Compute Engine:** https://cloud.google.com/architecture/best-practices-compute-engine

**Question: 225** CertyIQ

Your managed instance group raised an alert stating that new instance creation has failed to create new instances. You need to solve the instance creation problem. What should you do?

A.Create an instance template that contains valid syntax which will be used by the instance group. Delete any persistent disks with the same name as instance names.

B.Create an instance template that contains valid syntax that will be used by the instance group. Verify that the instance name and persistent disk name values are not the same in the template.

C.Verify that the instance template being used by the instance group contains valid syntax. Delete any persistent disks with the same name as instance names. Set the disks.autoDelete property to true in the instance template.

D.Delete the current instance template and replace it with a new instance template. Verify that the instance name and persistent disk name values are not the same in the template. Set the disks.autoDelete property to true in the instance template.

**Answer: A**

**Explanation:**

The correct answer is **A**. Here's a detailed justification:

Managed Instance Groups (MIGs) rely on instance templates to define the configuration of the VMs they create. A primary reason for instance creation failures in a MIG is an issue with the instance template itself.

Option A directly addresses this by suggesting a review of the template for "valid syntax." This implies that the template might have configuration errors, misspellings, or incorrect resource specifications, preventing new instances from launching. Additionally, option A correctly identifies a common conflict: when a persistent disk shares the same name as the instance. This is problematic as the instance creation process may not be able to distinguish the disk from the instance leading to creation failures. By deleting disks with conflicting names the creation issue will be resolved.

Options B, C, and D also touch on important aspects, but they either introduce unnecessary steps or combine them incorrectly. While Option B correctly mentions the name conflict issue, creating an instance template is already mentioned in the question and does not need to be reiterated. Option C suggests deleting persistent disks with the same name as instances and sets the autoDelete property, while beneficial in other situations, it doesn't resolve the underlying syntax issue in the instance template that is causing the problem in the first place. Option D suggests deleting the current template, which is an extreme measure, and while it's fine to replace a faulty template, you should always attempt to repair the existing template first. Moreover, setting the autoDelete property is not related to fixing creation issues. The core problem is an instance template with invalid syntax that can be quickly remediated by creating and using a new template. Option A directly addresses both the invalid syntax and the naming conflict using the most efficient and correct method to solve the problem.

**Key Concepts and Links:**

**Managed Instance Groups (MIGs):** https://cloud.google.com/compute/docs/instance-groups/
**Instance Templates:** https://cloud.google.com/compute/docs/instance-templates/
**Persistent Disks:** https://cloud.google.com/compute/docs/disks
**Troubleshooting Instance Creation Issues:** While there's no single document detailing this specific scenario, general debugging steps for MIGs and templates are available across various pages linked above. For example, check the logs and alerts which can help you identify invalid syntax or resource conflicts.

**Question: 226** CertyIQ

You have created an application that is packaged into a Docker image. You want to deploy the Docker image as a workload on Google Kubernetes Engine. What should you do?

A.Upload the image to Cloud Storage and create a Kubernetes Service referencing the image.
B.Upload the image to Cloud Storage and create a Kubernetes Deployment referencing the image.
C.Upload the image to Artifact Registry and create a Kubernetes Service referencing the image.
D.Upload the image to Artifact Registry and create a Kubernetes Deployment referencing the image.

**Answer: D**

**Explanation:**

The correct answer is **D. Upload the image to Artifact Registry and create a Kubernetes Deployment referencing the image.** Here's why:

**Docker Image Storage:** Docker images are container images that need to be stored in a registry. While Cloud Storage can store various files, it's not designed for managing container images. Artifact Registry is Google Cloud's managed service specifically for storing container images and other build artifacts. This makes it the correct choice for securely storing your Docker image before deployment.

**Kubernetes Deployments:** Kubernetes Deployments are a higher-level abstraction than Services. Deployments manage and update replicated sets of Pods (which are the smallest deployable units in Kubernetes, typically representing one container). They handle rolling updates, rollbacks, and scaling of your application. Services are used to expose your application to the network, usually after a deployment has been

set up.

**Why not Cloud Storage?** While Cloud Storage is versatile, it lacks the specific features of a container registry like Artifact Registry. It doesn't manage image versions, indexing, or the specific metadata needed by Kubernetes when pulling container images.

**Why not Kubernetes Service alone?** A Kubernetes Service does not, in itself, deploy application instances. It exposes the application, but needs a backing deployment. Deployments are what manages the actual containers that are running the application.

**Workflow:** The correct workflow involves first pushing the built Docker image to Artifact Registry using tools like docker push or gcloud container images push. Then, a Kubernetes Deployment manifest is created, specifying the image location in Artifact Registry. When the Deployment is applied to the Kubernetes cluster, Kubernetes will pull the image from Artifact Registry, create the necessary Pods, and then monitor them, allowing for easy scaling and rollouts.

**Security:** Using Artifact Registry ensures proper authentication and authorization for pulling the image, enhancing security within your cluster.

**Authoritative Links:**

**Artifact Registry Documentation:** https://cloud.google.com/artifact-registry/docs
**Kubernetes Deployments Documentation:** https://kubernetes.io/docs/concepts/workloads/controllers/deployment/
**Kubernetes Services Documentation:** https://kubernetes.io/docs/concepts/services-networking/service/
**Container Registry vs. Artifact Registry:** https://cloud.google.com/container-registry/docs/transition/comparison

In conclusion, Artifact Registry provides the correct storage solution for Docker images, while Deployments are the appropriate Kubernetes resource for managing the desired containerized application within your cluster, making option D the only correct choice.

**Question: 227** CertyIQ

You are using Looker Studio to visualize a table from your data warehouse that is built on top of BigQuery. Data is appended to the data warehouse during the day. At night, the daily summary is recalculated by overwriting the table. You just noticed that the charts in Looker Studio are broken, and you want to analyze the problem. What should you do?

A.In Cloud Logging, create a filter for your Looker Studio report.
B.Use the open source CLI tool, Snapshot Debugger, to find out why the data was not refreshed correctly.
C.Review the Error Reporting page in the Google Cloud console to find any errors.
D.Use the BigQuery interface to review the nightly job and look for any errors.

**Answer: D**

**Explanation:**

The correct answer is **D. Use the BigQuery interface to review the nightly job and look for any errors.** Here's why:

The core issue lies in the fact that the Looker Studio charts are broken after the nightly summary calculation overwrites the BigQuery table. This strongly suggests a problem with the overnight BigQuery job itself. Looker Studio, as a data visualization tool, simply displays the data it receives from BigQuery. If that data is incorrect or unavailable due to a failed job, Looker Studio will reflect this.

Option A (Cloud Logging filter) is not as targeted. While useful for general monitoring, logs for Looker Studio would primarily capture activity from Looker Studio, not the underlying data source issue. Option B (Snapshot Debugger) is irrelevant, as it is designed for debugging application code, not data pipeline failures. Option C (Error Reporting) might catch high-level errors but lacks the granularity needed to understand the specifics of a failed BigQuery job.

BigQuery's interface provides the direct means to inspect the details of the nightly job, including any error messages, query results, and resource utilization. This allows direct troubleshooting of the data transformation process, which is essential to resolving the Looker Studio chart problem. This aligns with the principle of "checking the source" before examining downstream tools. Debugging BigQuery queries is a common task for those who use Looker studio with BigQuery as the data source. By using the BigQuery UI, it becomes more transparent what exactly is going wrong during the job execution.

**Authoritative Links for Further Research:**

**BigQuery Job Monitoring:** https://cloud.google.com/bigquery/docs/monitoring
**Looker Studio Troubleshooting:** https://support.google.com/looker-studio/answer/7020514?hl=en
**Debugging BigQuery Queries:** https://cloud.google.com/bigquery/docs/troubleshoot-queries

**Question: 228** CertyIQ

You have a batch workload that runs every night and uses a large number of virtual machines (VMs). It is fault-tolerant and can tolerate some of the VMs being terminated. The current cost of VMs is too high. What should you do?

A.Run a test using simulated maintenance events. If the test is successful, use Spot N2 Standard VMs when running future jobs.

B.Run a test using simulated maintenance events. If the test is successful, use N2 Standard VMs when running future jobs.

C.Run a test using a managed instance group. If the test is successful, use N2 Standard VMs in the managed instance group when running future jobs.

D.Run a test using N1 standard VMs instead of N2. If the test is successful, use N1 Standard VMs when running future jobs.

**Answer: A**

**Explanation:**

The correct answer is A. Spot VMs are a cost-effective option for fault-tolerant batch workloads because they utilize spare Compute Engine capacity, offering significant discounts compared to regular VMs. However, Spot VMs can be preempted, meaning Google may terminate them with short notice if the capacity is needed elsewhere. Therefore, it's crucial to test the workload's resilience to such interruptions before relying on Spot VMs in production. Option A proposes running a test using simulated maintenance events, which effectively mimics Spot VM preemption, allowing verification of the workload's tolerance. If the test is successful, utilizing Spot N2 Standard VMs (which offer a balance of performance and cost) becomes a cost-reducing strategy. Option B suggests using regular N2 Standard VMs, which while efficient, do not address the cost issue. Option C incorrectly focuses on Managed Instance Groups (MIGs), which provide high availability and scaling but are not primarily for cost reduction like Spot VMs. Option D proposes switching to N1 Standard VMs which are older generation and not cost-efficient than the current N2 standard VMs with spot pricing. Therefore, the only option that is both cost-effective and robust is using spot instances with proper testing.

Further research:

**Google Cloud Spot VMs:** https://cloud.google.com/compute/docs/instances/spot - This documentation provides detailed information about Spot VMs, including their characteristics, limitations, and use cases.

**Google Cloud Preemptible VMs:** https://cloud.google.com/compute/docs/instances/preemptible - This page explains how preemptible VMs (now called Spot VMs) work and how they differ from regular VMs.
**Google Cloud Instance Types:** https://cloud.google.com/compute/docs/machine-types - Provides an overview of different machine types (like N1, N2), their performance capabilities and price.

**Question: 229** CertyIQ

You created several resources in multiple Google Cloud projects. All projects are linked to different billing accounts. To better estimate future charges, you want to have a single visual representation of all costs incurred. You want to include new cost data as soon as possible. What should you do?

A.Fill all resources in the Pricing Calculator to get an estimate of the monthly cost.
B.Use the Reports view in the Cloud Billing Console to view the desired cost information.
C.Visit the Cost Table page to get a CSV export and visualize it using Looker Studio.
D.Configure Billing Data Export to BigQuery and visualize the data in Looker Studio.

**Answer: D**

**Explanation:**

The correct answer is **D. Configure Billing Data Export to BigQuery and visualize the data in Looker Studio.**

Here's why:

**Centralized Data:** Billing data export to BigQuery provides a centralized repository for all cost data from multiple projects, regardless of their associated billing accounts. This addresses the need for a single view of all costs.
**Near Real-Time Data:** BigQuery receives billing data updates regularly (typically within a few hours), allowing for timely analysis of costs. This aligns with the requirement to include new cost data as soon as possible.
**Data Analysis Capabilities:** BigQuery offers powerful query capabilities, making it easy to analyze cost data based on different dimensions (project, resource type, service, etc.).
**Visualization with Looker Studio:** Looker Studio is a robust business intelligence platform that integrates seamlessly with BigQuery. It can be used to create interactive dashboards and visualizations that effectively communicate cost information.
**Scalability:** Both BigQuery and Looker Studio are scalable solutions capable of handling large volumes of billing data.
**Cost Transparency and Optimization:** This approach provides detailed insights into spending patterns, enabling cost optimization efforts.

**Why other options are incorrect:**

**A. Fill all resources in the Pricing Calculator:** The Pricing Calculator is for estimating costs before deployment. It doesn't provide actual, historical cost data. It's not suitable for analyzing existing expenses across multiple projects and billing accounts.
**B. Use the Reports view in the Cloud Billing Console:** While the Cloud Billing Console provides cost reports, it is not as flexible or powerful for deep analysis as BigQuery. It also does not directly provide a single view of cost across different billing accounts.
**C. Visit the Cost Table page to get a CSV export and visualize it using Looker Studio:** CSV exports require manual effort for each project and are not automated. Manually managing CSV files is prone to error and doesn't meet the requirement for near real-time updates.

**Authoritative links for further research:**

**Google Cloud Billing Export to BigQuery:** https://cloud.google.com/billing/docs/how-to/export-data-bigquery

**Looker Studio:** https://lookerstudio.google.com/
**Google Cloud Billing Concepts:** https://cloud.google.com/billing/docs/concepts

In summary, exporting billing data to BigQuery and then visualizing it with Looker Studio is the most efficient and robust solution for gaining a single, near-real-time view of costs across multiple projects and billing accounts within Google Cloud.

**Question: 230** CertyIQ

Your company has a large quantity of unstructured data in different file formats. You want to perform ETL transformations on the data. You need to make the data accessible on Google Cloud so it can be processed by a Dataflow job. What should you do?

A.Upload the data to BigQuery using the bq command line tool.
B.Upload the data to Cloud Storage using the gcloud storage command.
C.Upload the data into Cloud SQL using the import function in the Google Cloud console.
D.Upload the data into Cloud Spanner using the import function in the Google Cloud console.

**Answer: B**

**Explanation:**

The correct answer is B, uploading the data to Cloud Storage using the gcloud storage command. Here's why:

Cloud Storage is Google Cloud's object storage service, designed for storing large amounts of unstructured data, including various file formats. This makes it the ideal first landing point for data destined for ETL processing. Dataflow, Google Cloud's managed data processing service, can easily read from and write to Cloud Storage. This allows for a seamless integration of storage and processing for ETL pipelines. Options A, C, and D are incorrect because they involve databases (BigQuery, Cloud SQL, and Cloud Spanner), which are better suited for structured data. Directly loading unstructured data into these databases is typically not efficient, and they are not the appropriate source for a Dataflow job in this scenario. The gcloud storage command provides a command-line interface to interact with Cloud Storage, allowing for easy uploads of data. Using Cloud Storage simplifies the initial data ingest for subsequent processing by Dataflow. Cloud Storage acts as a staging area, enabling data transformations in Dataflow without directly impacting production databases. For further research on Cloud Storage and its uses, refer to the official documentation: https://cloud.google.com/storage/docs. You can explore Dataflow's capabilities via their documentation here: https://cloud.google.com/dataflow/docs

**Question: 231** CertyIQ

You have deployed an application on a single Compute Engine instance. The application writes logs to disk. Users start reporting errors with the application. You want to diagnose the problem. What should you do?

A.Navigate to Cloud Logging and view the application logs.
B.Configure a health check on the instance and set a “consecutive successes” Healthy threshold value of 1.
C.Connect to the instance’s serial console and read the application logs.
D.Install and configure the Ops agent and view the logs from Cloud Logging.

**Answer: D**

**Explanation:**

Here's a detailed justification for why the correct answer is D, and why the other options are less suitable for diagnosing application errors in this scenario:

**Justification for Option D: Install and configure the Ops agent and view the logs from Cloud Logging.**

Option D is the most appropriate approach because it leverages Google Cloud's robust logging and monitoring capabilities. The Ops Agent is specifically designed to collect logs and metrics from Compute Engine instances and forward them to Cloud Logging. This offers a centralized and searchable repository for log data, making it easy to analyze and identify the root cause of the errors. Unlike simply accessing logs locally on the machine, Cloud Logging offers powerful filtering, querying, and alerting capabilities. This is crucial for efficiently identifying patterns and anomalies in the application's log output, especially during periods of user-reported errors. The Ops Agent provides a consistent and standardized method of log collection, ensuring that the data is in a format that Cloud Logging can easily process and analyze. This allows for efficient correlation of error messages and other relevant log events to troubleshoot the application.

**Why other options are less suitable:**

**Option A: Navigate to Cloud Logging and view the application logs.** This option is incorrect because without prior configuration and the installation of the Ops Agent or a similar log forwarding method, the application's logs residing on the instance's disk will not be available in Cloud Logging. Cloud Logging does not automatically scrape application logs from the disk without such setup.

**Option B: Configure a health check on the instance and set a "consecutive successes" Healthy threshold value of 1.** While health checks are critical for application availability and auto-healing, they only provide information about the instance's operational status and won't help in debugging application-level errors. A healthy instance may still be running an application that is generating errors. Health checks are for high-level availability not low-level application debugging.

**Option C: Connect to the instance's serial console and read the application logs.** The serial console is primarily used for debugging instance-level issues and not application-specific logs. While you can access the console for some basic troubleshooting, it is not efficient for reviewing a large amount of application log data and does not offer the robust filtering and analysis capabilities of Cloud Logging. It also requires direct console interaction and not easily scalable and auditable.

**Authoritative Links for Further Research:**

**Google Cloud Ops Agent:** https://cloud.google.com/stackdriver/docs/solutions/ops-agent
**Cloud Logging:** https://cloud.google.com/logging
**Compute Engine Health Checks:** https://cloud.google.com/compute/docs/load-balancing/health-checks
**Compute Engine Serial Console:** https://cloud.google.com/compute/docs/troubleshooting/serial-console

In conclusion, the most efficient and recommended approach for diagnosing application errors in this scenario is to leverage the Ops Agent to collect the application's logs and analyze them within Cloud Logging due to its centralized, searchable, and analyzable nature.

**Question: 232** CertyIQ

You recently received a new Google Cloud project with an attached billing account where you will work. You need to create instances, set firewalls, and store data in Cloud Storage. You want to follow Google-recommended practices. What should you do?

A.Use the gcloud CLI services enable cloudresourcemanager.googleapis.com command to enable all resources.

B.Use the gcloud services enable compute.googleapis.com command to enable Compute Engine and the gcloud services enable storage-api.googleapis.com command to enable the Cloud Storage APIs.

C.Open the Google Cloud console and enable all Google Cloud APIs from the API dashboard.

D.Open the Google Cloud console and run gcloud init --project in a Cloud Shell.

**Answer: B**

**Explanation:**

The correct answer is **B**, focusing on enabling specific APIs necessary for the tasks described in the scenario. Google Cloud projects, by default, have no APIs enabled. You must explicitly enable the APIs required for your project to function. Choice **B** correctly identifies that to create instances (virtual machines), you need the Compute Engine API (compute.googleapis.com), and to store data in Cloud Storage, you need the Cloud Storage API (storage-api.googleapis.com). The gcloud services enable command is the appropriate tool for enabling APIs via the command line. Option **A** is incorrect because it attempts to enable the Cloud Resource Manager API. While the Resource Manager API is fundamental, it is not the API required for the use cases described. It is the API to organize the project rather than use its resources. Option **C** is incorrect because enabling all APIs is an unnecessary and insecure practice; it unnecessarily expands your attack surface and can incur unexpected costs. Option **D** is incorrect; while running gcloud init --project sets the active project in your Cloud Shell environment, it doesn't enable any APIs. You need to explicitly enable the APIs you want to use. Enabling APIs should be done on a need-to-use basis. Choosing B aligns with the principle of least privilege and best practices for Google Cloud.

**Authoritative Links for further research:**

**gcloud services enable:** https://cloud.google.com/sdk/gcloud/reference/services/enable
**Enabling and disabling services:** https://cloud.google.com/service-usage/docs/enable-disable
**Compute Engine API:** https://cloud.google.com/compute/docs/reference/rest/v1
**Cloud Storage API:** https://cloud.google.com/storage/docs/apis

**Question: 233** CertyIQ

Your application development team has created Docker images for an application that will be deployed on Google Cloud. Your team does not want to manage the infrastructure associated with this application. You need to ensure that the application can scale automatically as it gains popularity. What should you do?

A.Create an instance template with the container image, and deploy a Managed Instance Group with Autoscaling.

B.Upload Docker images to Artifact Registry, and deploy the application on Google Kubernetes Engine using Standard mode.

C.Upload Docker images to the Cloud Storage, and deploy the application on Google Kubernetes Engine using Standard mode.

D.Upload Docker images to Artifact Registry, and deploy the application on Cloud Run.

**Answer: D**

**Explanation:**

The correct answer is **D. Upload Docker images to Artifact Registry, and deploy the application on Cloud Run.** Here's why:

Cloud Run is a fully managed serverless platform designed for deploying containerized applications. It abstracts away the underlying infrastructure management, including scaling, which aligns perfectly with the requirement of not managing infrastructure. By uploading Docker images to Artifact Registry (a private registry for storing container images), Cloud Run can easily access and deploy the application. Cloud Run automatically scales the application based on demand, ensuring optimal performance during periods of high traffic without manual intervention.

Option A, using Managed Instance Groups (MIGs), while providing autoscaling, still requires managing the underlying VMs and instances, which is against the stated requirement. Option B and C, although utilize Google Kubernetes Engine (GKE), necessitate more infrastructure configuration and management compared to Cloud Run. GKE also requires cluster management which falls outside of the team's preference to avoid managing infrastructure. GKE's auto-scaling needs to be configured specifically, where cloud run automatically scales as necessary. Cloud Storage is also unsuitable for storing application container images; Artifact Registry is the specific Google Cloud service for this. Therefore, Cloud Run offers the most streamlined and hands-off approach for deploying and scaling containerized applications, meeting the application team's needs perfectly.

**Key Concepts:**

**Serverless:** Cloud Run is a serverless platform, meaning you don't manage the underlying servers.
**Containerization:** Docker images package your application and its dependencies.
**Artifact Registry:** A secure and private repository for storing container images.
**Autoscaling:** Cloud Run automatically adjusts the number of instances based on demand.

**Authoritative Links for Further Research:**

**Cloud Run:** https://cloud.google.com/run
**Artifact Registry:** https://cloud.google.com/artifact-registry
**Serverless Computing:** https://cloud.google.com/serverless
**Managed Instance Groups:** https://cloud.google.com/compute/docs/instance-groups
**Google Kubernetes Engine:** https://cloud.google.com/kubernetes-engine

**Question: 234** CertyIQ

You are migrating a business critical application from your local data center into Google Cloud. As part of your high-availability strategy, you want to ensure that any data used by the application will be immediately available if a zonal failure occurs. What should you do?

A.Store the application data on a zonal persistent disk. Create a snapshot schedule for the disk. If an outage occurs, create a new disk from the most recent snapshot and attach it to a new VM in another zone.

B.Store the application data on a zonal persistent disk. If an outage occurs, create an instance in another zone with this disk attached.

C.Store the application data on a regional persistent disk. Create a snapshot schedule for the disk. If an outage occurs, create a new disk from the most recent snapshot and attach it to a new VM in another zone.

D.Store the application data on a regional persistent disk. If an outage occurs, create an instance in another zone with this disk attached.

**Answer: D**

**Explanation:**

The correct answer is **D**. Here's why:

High availability in Google Cloud Platform (GCP) necessitates minimizing downtime during zonal failures. A critical component is data storage. Zonal persistent disks (options A and B) are tied to a single zone. If that zone fails, the disk and its data become unavailable. This violates the requirement for immediate data availability. To mitigate this, we use regional persistent disks (options C and D). Regional disks are synchronously replicated across two zones within a region, ensuring data availability even if one zone goes down.

Option A, despite using snapshots, which create backups at specific points, doesn't offer immediate recovery in case of a zonal failure. Restoring from snapshots adds delay in making the data available. Option B using a

zonal persistent disk is tied to a single zone, which is exactly what needs to be avoided to ensure immediate data availability when a zonal failure occurs. Option C creates a regional persistent disk, which is correct, however, it suggests using a snapshot instead of immediately attaching the already replicated data in the other zone, which is incorrect.

Option D correctly addresses the requirement by using a regional persistent disk. Because regional disks are replicated, data remains accessible in another zone within the same region if a zone failure occurs. By creating a new instance in an available zone and attaching the regional persistent disk to it, the application can resume quickly. The data will already exist and be available on the disk in the other zone without the need to restore from a snapshot, which ensures the desired immediate availability in the case of a zonal failure.

Therefore, using regional persistent disks is crucial for building high-availability applications in GCP where immediate data access during zonal failures is a requirement.

**Authoritative links:**

**Regional persistent disks:** https://cloud.google.com/compute/docs/disks#regional_disks
**Persistent disk concepts:** https://cloud.google.com/compute/docs/disks/
**High availability in Compute Engine:** https://cloud.google.com/compute/docs/high-availability

**Question: 235** CertyIQ

The DevOps group in your organization needs full control of Compute Engine resources in your development project. However, they should not have permission to create or update any other resources in the project. You want to follow Google's recommendations for setting permissions for the DevOps group. What should you do?

A.Grant the basic role roles/viewer and the predefined role roles/compute.admin to the DevOps group.

B.Create an IAM policy and grant all compute.instanceAdmin.* permissions to the policy. Attach the policy to the DevOps group.

C.Create a custom role at the folder level and grant all compute.instanceAdmin.* permissions to the role. Grant the custom role to the DevOps group.

D.Grant the basic role roles/editor to the DevOps group.

**Answer: C**

**Explanation:**

Here's a breakdown of why option C is the best approach and why the others fall short:

**Justification for Option C (Correct Answer):**

Option C, creating a custom role at the folder level with compute.instanceAdmin. permissions and granting it to the DevOps group, aligns with Google Cloud's principle of least privilege. Custom roles allow precise control over permissions, granting only the necessary access and preventing over-provisioning which is crucial for security best practices. By granting *compute.instanceAdmin.*, the DevOps team gains complete control over Compute Engine instances (creation, modification, deletion, etc.) without access to other resources, satisfying the requirement. Creating the custom role at the folder level allows the permission to apply to all subprojects of that folder and improves scalability and administration.

**Why Other Options Are Incorrect:**

**Option A:** Granting roles/viewer is insufficient for managing Compute Engine resources. While roles/compute.admin allows for comprehensive access to Compute Engine resources including networking which was not required for this scenario. This does not restrict access to other non compute engine resources. This is why this is not the best answer.

**Option B:** Creating an IAM policy with specific permissions is not the best approach because it is very specific to the project, making it very difficult to manage and implement in the future. Custom roles at folder level are reusable across all projects in the folder.
**Option D:** Granting roles/editor provides very broad permissions that go beyond Compute Engine and include resource creation/modification, potentially violating the constraint of not being able to access other resources. This approach violates the principle of least privilege.

**Key Cloud Computing Concepts:**

**Principle of Least Privilege:** Granting only the minimum permissions necessary for a user or service to perform its tasks, which reduces the attack surface and limits potential damage from security breaches.
**IAM (Identity and Access Management):** Google Cloud's service for managing who (identities) has what access (roles) to Google Cloud resources.
**Predefined Roles:** Ready-made roles that group commonly used permissions.
**Custom Roles:** User-defined roles that offer granular control over permissions.
**Folder Level IAM:** Applying IAM settings to a folder which cascades to all the projects within that folder.

**Authoritative Links:**

**Understanding IAM:** https://cloud.google.com/iam/docs/overview
**Creating and managing custom roles:** https://cloud.google.com/iam/docs/creating-custom-roles
**Compute Engine IAM Roles:** https://cloud.google.com/compute/docs/access/iam

In summary, option C provides a secure and scalable solution by implementing the principle of least privilege using a custom role tailored for Compute Engine access within the specified project while also promoting reusability through folder level IAM.

**Question: 236** CertyIQ

Your team is running an on-premises ecommerce application. The application contains a complex set of microservices written in Python, and each microservice is running on Docker containers. Configurations are injected by using environment variables. You need to deploy your current application to a serverless Google Cloud cloud solution. What should you do?

A.Use your existing CI/CD pipeline. Use the generated Docker images and deploy them to Cloud Run. Update the configurations and the required endpoints.

B.Use your existing continuous integration and delivery (CI/CD) pipeline. Use the generated Docker images and deploy them to Cloud Function. Use the same configuration as on-premises.

C.Use the existing codebase and deploy each service as a separate Cloud Function. Update the configurations and the required endpoints.

D.Use your existing codebase and deploy each service as a separate Cloud Run. Use the same configurations as on-premises.

**Answer: A**

**Explanation:**

The correct answer is **A**. Here's why:

The scenario describes an existing microservices application packaged in Docker containers with configurations managed by environment variables. Cloud Run is the ideal Google Cloud serverless compute platform for deploying containerized applications. Option A leverages the existing Docker images, minimizing changes to the application's build process. Cloud Run allows developers to deploy the pre-built containers directly. Cloud Run also facilitates easy management of environment variables, allowing for updating the configurations and endpoints required for the cloud environment. Using existing CI/CD pipelines also

streamlines the transition. Option B suggests Cloud Functions, which are designed for single-purpose, event-driven functions, not full applications like the stated complex microservices. While feasible to break it down, it adds unecessary overhead. Option C also suggests Cloud Functions and would involve significant refactoring of the existing microservices. Option D, although utilizing the correct product Cloud Run, proposes using the existing code, when the question is set on dockerized images, which would also force some refactoring. Hence option A, which utilizes the containerization and existing pipeline is the best option.

**Authoritative Links for Further Research:**

**Cloud Run Documentation:** https://cloud.google.com/run/docs
**Cloud Functions Documentation:** https://cloud.google.com/functions/docs
**Containers Overview:** https://cloud.google.com/containers/docs/overview
**CI/CD on Google Cloud:** https://cloud.google.com/solutions/continuous-delivery

**Question: 237** CertyIQ

You are running multiple microservices in a Kubernetes Engine cluster. One microservice is rendering images. The microservice responsible for the image rendering requires a large amount of CPU time compared to the memory it requires. The other microservices are workloads that are optimized for n2-standard machine types. You need to optimize your cluster so that all workloads are using resources as efficiently as possible. What should you do?

A.Assign the pods of the image rendering microservice a higher pod priority than the other microservices.

B.Create a node pool with compute-optimized machine type nodes for the image rendering microservice. Use the node pool with general-purpose machine type nodes for the other microservices.

C.Use the node pool with general-purpose machine type nodes for the image rendering microservice. Create a node pool with compute-optimized machine type nodes for the other microservices.

D.Configure the required amount of CPU and memory in the resource requests specification of the image rendering microservice deployment. Keep the resource requests for the other microservices at the default.

**Answer: B**

**Explanation:**

The optimal solution is **B. Create a node pool with compute-optimized machine type nodes for the image rendering microservice. Use the node pool with general-purpose machine type nodes for the other microservices.**

Here's why:

The core issue is resource optimization. The image rendering service is CPU-bound, requiring significant processing power but relatively little memory. General-purpose machine types, like the n2-standard, offer a balance of CPU and memory, which isn't ideal for this workload. Conversely, compute-optimized machine types are designed for CPU-intensive tasks, offering more CPU cores per unit of memory.

By segregating the image rendering service onto a node pool with compute-optimized nodes, you allow it to effectively utilize the available CPU resources without paying for excess memory. Simultaneously, the other microservices, already optimized for n2-standard machines, continue to run efficiently on the general-purpose node pool. This approach maximizes resource utilization across the cluster, reducing costs and improving performance.

Option A, prioritizing pods, influences scheduling but does not address the underlying need for suitable hardware. Options C and D are not ideal as they either place the CPU-intensive task on the general-purpose machines (C) or simply define the request resources without providing the proper node for the optimized allocation (D). Specifically, option C puts the workload that requires more CPU to run on general-purpose nodes which is against the requirement. Option D, while important for Kubernetes scheduling, doesn't address

the underlying hardware mismatch, as Kubernetes will still schedule pods based on available resources, leading to sub-optimal usage on the general purpose node. Segregating workloads based on resource requirements is a core concept in cloud resource management.

**Authoritative Links:**

**Google Cloud Documentation on Machine Types:** https://cloud.google.com/compute/docs/machine-types - Provides an overview of different machine types and their suitability for various workloads.
**Google Cloud Documentation on Node Pools:** https://cloud.google.com/kubernetes-engine/docs/how-to/node-pools - Explains how to create and manage node pools for workload segregation.
**Kubernetes Documentation on Resource Management:** https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ - Provides more details on how to manage resource allocation in Kubernetes.

**Question: 238** CertyIQ

You are working in a team that has developed a new application that needs to be deployed on Kubernetes. The production application is business critical and should be optimized for reliability. You need to provision a Kubernetes cluster and want to follow Google-recommended practices. What should you do?

A.Create a GKE Autopilot cluster. Enroll the cluster in the rapid release channel.
B.Create a GKE Autopilot cluster. Enroll the cluster in the stable release channel.
C.Create a zonal GKE standard cluster. Enroll the cluster in the stable release channel.
D.Create a regional GKE standard cluster. Enroll the cluster in the rapid release channel.

**Answer: B**

**Explanation:**

Here's a justification for why option B is the correct choice:

The primary requirement is to deploy a business-critical application on Kubernetes with a focus on reliability, while adhering to Google-recommended practices. GKE Autopilot clusters are designed to simplify cluster management and optimize for reliability and cost, making them a natural fit for production workloads. They abstract away much of the underlying infrastructure management, such as node provisioning and scaling, which Google handles automatically, leading to higher reliability.

While GKE Standard clusters offer more control, this comes with an added burden of manual configuration and maintenance, potentially introducing human error and affecting reliability. Therefore, GKE Autopilot is preferable for a reliability-focused deployment.

Release channels dictate the frequency with which new GKE features and bug fixes are rolled out. The Stable release channel is specifically recommended for production environments due to its focus on thoroughly tested and validated releases, which is crucial for business-critical applications requiring stability. The rapid release channel might introduce newer features sooner but carries a higher risk of encountering unexpected issues.

Therefore, creating a GKE Autopilot cluster and enrolling it in the stable release channel (option B) aligns with Google's best practices for reliability and provides a robust platform for the production application. Option A contradicts best practices by opting for the rapid release channel for business-critical applications. Options C and D utilize the GKE Standard cluster which requires more management on the user end and is not optimal for reliability. Option C incorrectly suggests the zonal cluster, which offers less availability than the regional alternative. Lastly option D combines a less reliable release channel with the less preferable cluster type.

**Authoritative Links for Further Research:**

**GKE Autopilot:** https://cloud.google.com/kubernetes-engine/docs/concepts/autopilot
**GKE Release Channels:** https://cloud.google.com/kubernetes-engine/docs/concepts/release-channels
**Choosing a cluster type** https://cloud.google.com/kubernetes-engine/docs/how-to/choosing-a-cluster-type

**Question: 239** CertyIQ

You are responsible for a web application on Compute Engine. You want your support team to be notified automatically if users experience high latency for at least 5 minutes. You need a Google-recommended solution with no development cost. What should you do?

A.Export Cloud Monitoring metrics to BigQuery and use a Looker Studio dashboard to monitor your web application's latency.

B.Create an alert policy to send a notification when the HTTP response latency exceeds the specified threshold.

C.Implement an App Engine service which invokes the Cloud Monitoring API and sends a notification in case of anomalies.

D.Use the Cloud Monitoring dashboard to observe latency and take the necessary actions when the response latency exceeds the specified threshold.

**Answer: B**

**Explanation:**

The correct answer is B: Create an alert policy to send a notification when the HTTP response latency exceeds the specified threshold. This approach directly addresses the problem by leveraging Cloud Monitoring's built-in alerting capabilities. It requires no code development, aligning with the requirement of "no development cost." Cloud Monitoring collects metrics from Compute Engine, including HTTP response latency. You can define an alert policy that triggers a notification (via email, SMS, or other channels) when this metric surpasses a specified threshold for a given duration (e.g., 5 minutes). This is a standard practice for monitoring critical applications and ensuring timely incident response. Option A, using BigQuery and Looker Studio, provides reporting and visualization, but not the proactive, automated notification required. Option C necessitates developing a custom App Engine service, which contradicts the "no development cost" constraint. Option D involves manual observation, which is inefficient and not automatic. Alerting policies are fundamental for operational excellence in cloud environments and ensure immediate attention to performance issues. For further information, refer to the official Google Cloud documentation: https://cloud.google.com/monitoring/alerts and https://cloud.google.com/monitoring/docs/alerting/policies.

**Question: 240** CertyIQ

You have an on-premises data analytics set of binaries that processes data files in memory for about 45 minutes every midnight. The sizes of those data files range from 1 gigabyte to 16 gigabytes. You want to migrate this application to Google Cloud with minimal effort and cost. What should you do?

A.Create a container for the set of binaries. Use Cloud Scheduler to start a Cloud Run job for the container.

B.Create a container for the set of binaries. Deploy the container to Google Kubernetes Engine (GKE) and use the Kubernetes scheduler to start the application.

C.Upload the code to Cloud Functions. Use Cloud Scheduler to start the application.

D.Lift and shift to a VM on Compute Engine. Use an instance schedule to start and stop the instance.

**Answer: D**

**Explanation:**

The most suitable solution for migrating the on-premises data analytics application with minimal effort and

cost is option D: "Lift and shift to a VM on Compute Engine. Use an instance schedule to start and stop the instance." This approach aligns best with the requirements by directly mirroring the existing environment.

Here's a detailed justification:

**Minimal Effort:** Lift and shift involves minimal code refactoring or architectural changes. The existing binaries can be directly deployed on a Compute Engine VM, similar to how they operate on-premises. This significantly reduces migration complexity and development time.
**Cost-Effectiveness:** For a processing workload that runs only for 45 minutes per day, a Compute Engine VM can be started and stopped on a schedule. This prevents unnecessary resource usage, reducing costs significantly compared to continuously running containerized services or serverless functions.
**Data Handling:** The application processes large data files (1-16 GB). A Compute Engine VM offers flexibility to mount storage for these data files and perform in-memory processing as required, accommodating the large file sizes effectively.
**Avoiding Over-Engineering:** Options A, B, and C involve more complex setup and management. Containerization (A & B) requires Dockerizing the binaries and managing container orchestration, while Cloud Functions (C) is limited by runtime and memory restrictions, making it less appropriate for large in-memory data processing.

Compute Engine's instance scheduling provides granular control over when the VM starts and stops, perfectly fitting the 45-minute daily processing window. This optimizes resource utilization and reduces expenses, which is in line with the requirements of minimal effort and cost. Cloud Run jobs (A), GKE (B), and Cloud Functions (C) introduce unnecessary complexities and costs for this particular use case.

**Authoritative Links:**

**Google Compute Engine:** https://cloud.google.com/compute
**Compute Engine Instance Scheduling:** https://cloud.google.com/compute/docs/instances/schedule-instance-start-stop
**Lift and Shift Migration:** https://cloud.google.com/migration/modernize/lift-and-shift

**Question: 241** CertyIQ

You used the gcloud container clusters command to create two Google Cloud Kubernetes (GKE) clusters: prod-cluster and dev-cluster.

•prod-cluster is a standard cluster.
•dev-cluster is an auto-pilot cluster.

When you run the kubectl get nodes command, you only see the nodes from prod-cluster. Which commands should you run to check the node status for dev-cluster?

A.gcloud container clusters get-credentials dev-cluster
kubectl get nodes

B.gcloud container clusters update -generate-password dev-cluster kubectl get nodes

C.kubectl config set-context dev-cluster
kubectl cluster-info

D.kubectl config set-credentials dev-cluster
kubectl cluster-info

**Answer: A**

**Explanation:**

The correct answer is **A. gcloud container clusters get-credentials dev-cluster; kubectl get nodes**. Here's why:

Kubernetes clusters, including GKE, require specific authentication credentials for kubectl to interact with them. The kubectl command-line tool uses a configuration file, typically ~/.kube/config, to store these credentials. When you create a cluster using gcloud container clusters create, the credentials are not automatically added to this configuration for direct access using kubectl.

To access a specific cluster, you must first obtain its credentials using gcloud container clusters get-credentials. This command retrieves the necessary certificates and tokens and updates your kubectl configuration to include the specified cluster (in this case, dev-cluster). Only after this step can you successfully run kubectl commands against that cluster.

Option A does precisely this: it uses gcloud container clusters get-credentials dev-cluster to fetch the credentials for dev-cluster and add them to the kubectl configuration. Then, kubectl get nodes can successfully connect to and retrieve node information from dev-cluster.

Options B, C, and D are incorrect because they do not properly obtain and configure kubectl to connect to the dev-cluster. Option B is specifically for generating passwords (not relevant here), while C and D attempt to set contexts/credentials directly with kubectl, which will not resolve credentials unless they are already fetched with gcloud tools. Option C focuses on cluster information and does not retrieve or use the right authentication.

Autopilot clusters, while managed by Google, still require proper credentials to be configured for user interaction using kubectl. The core process of obtaining credentials with gcloud then using kubectl to interact with the cluster remains the same for both standard and autopilot GKE clusters.

Relevant links:

**Getting cluster credentials**: https://cloud.google.com/kubernetes-engine/docs/how-to/connect-to-cluster
**gcloud container clusters get-credentials documentation**: https://cloud.google.com/sdk/gcloud/reference/container/clusters/get-credentials
**kubectl documentation**: https://kubernetes.io/docs/reference/kubectl/

**Question: 242** CertyIQ

You recently discovered that your developers are using many service account keys during their development process. While you work on a long term improvement, you need to quickly implement a process to enforce short-lived service account credentials in your company. You have the following requirements:

•All service accounts that require a key should be created in a centralized project called pj-sa.
•Service account keys should only be valid for one day.

You need a Google-recommended solution that minimizes cost. What should you do?

A.Implement a Cloud Run job to rotate all service account keys periodically in pj-sa. Enforce an org policy to deny service account key creation with an exception to pj-sa.

B.Implement a Kubernetes CronJob to rotate all service account keys periodically. Disable attachment of service accounts to resources in all projects with an exception to pj-sa.

C.Enforce an org policy constraint allowing the lifetime of service account keys to be 24 hours. Enforce an org policy constraint denying service account key creation with an exception on pj-sa.

D.Enforce a DENY org policy constraint over the lifetime of service account keys for 24 hours. Disable attachment of service accounts to resources in all projects with an exception to pj-sa.

**Answer: C**

**Explanation:**

The correct answer is **C**. Here's why:

Option C effectively addresses the requirements using Google Cloud's recommended best practices for managing service account keys, minimizing cost, and offering a robust solution. Enforcing an organization policy constraint to limit the lifetime of service account keys to 24 hours directly addresses the need for short-lived credentials, reducing the risk associated with long-lived keys being compromised. Furthermore, restricting key creation via another organizational policy, except in the pj-sa project where keys are created, centralizes key management as required. This approach leverages the inherent features of Google Cloud's Identity and Access Management (IAM) framework and is a cost-effective solution as it uses existing policy enforcement mechanisms. No new compute resources or services are necessary. The policy enforcement acts as a guardrail, automatically implementing the desired behavior.

Options A and B suggest actively rotating keys using Cloud Run and Kubernetes CronJob, respectively. While rotation is a good practice, these approaches are more complex, introduce additional management overhead, and incur costs associated with running the compute resources. In addition to the complexity, these options don't directly enforce short lifespans. They just automate rotations and would require a second layer of enforcement to ensure that only short-lived keys are used. Option D incorrectly suggests using a DENY policy over key lifetimes. Policies limit the creation/usage, not the lifetime. Also option D incorrectly suggests the need to disable the attachment of service accounts to resources as the issue is about key management not the service account.

Therefore, option C provides the most cost-effective and efficient solution, directly using policy enforcement for managing the key lifetime while ensuring that key creation is performed within the project designated for it.

**Authoritative Links:**

**Organization Policies:** https://cloud.google.com/resource-manager/docs/organization-policy/overview
**Service Account Key Management:** https://cloud.google.com/iam/docs/understanding-service-accounts#managing_service_account_keys
**Constrain Key Duration:** https://cloud.google.com/iam/docs/understanding-service-accounts#key-expiration

**Question: 243** CertyIQ

Your company is running a three-tier web application on virtual machines that use a MySQL database. You need to create an estimated total cost of cloud infrastructure to run this application on Google Cloud instances and Cloud SQL. What should you do?

A.Create a Google spreadsheet with multiple Google Cloud resource combinations. On a separate sheet, import the current Google Cloud prices and use these prices for the calculations within formulas.

B.Use the Google Cloud Pricing Calculator and select the Cloud Operations template to define your web application with as much detail as possible.

C.Implement a similar architecture on Google Cloud, and run a reasonable load test on a smaller scale. Check the billing information, and calculate the estimated costs based on the real load your system usually handles.

D.Use the Google Cloud Pricing Calculator to determine the cost of every Google Cloud resource you expect to use. Use similar size instances for the web server, and use your current on-premises machines as a comparison for Cloud SQL.

**Answer: D**

**Explanation:**

The correct answer is **D**. Here's why:

Option D directly addresses the need for an estimated cost by utilizing the Google Cloud Pricing Calculator. This tool is specifically designed to provide cost estimates for Google Cloud resources. You can input the

specific resources you plan to use, such as Compute Engine instances for the web servers and Cloud SQL for the database. By selecting appropriate instance types and sizes that closely match your current on-premises setup, you can achieve a reasonable cost projection. The approach is granular and allows for tailored estimates based on specific resource choices.

Option A, while technically feasible, is overly manual and inefficient. Manually tracking Google Cloud prices and building a spreadsheet introduces potential errors and requires constant price updates. It doesn't use Google's dedicated tool for this purpose.

Option B suggests using the "Cloud Operations template," which is not relevant to cost estimation. This template is more about operational management, not cost projection.

Option C, while useful for real-world performance insights, is not a cost estimation method. Running a load test, even at a smaller scale, will generate actual bills rather than an estimated cost which is the goal. Additionally, this approach would involve creating resources which would add unwanted costs when only a cost projection is desired.

The Google Cloud Pricing Calculator (option D) is the recommended method to achieve the required estimate as it is designed for this exact purpose.

**Authoritative Links:**

**Google Cloud Pricing Calculator:** https://cloud.google.com/products/calculator
**Google Cloud Pricing Overview:** https://cloud.google.com/pricing

**Question: 244** CertyIQ

You have a Bigtable instance that consists of three nodes that store personally identifiable information (PII) data. You need to log all read or write operations, including any metadata or configuration reads of this database table, in your company's Security Information and Event Management (SIEM) system. What should you do?

A.•Navigate to Cloud Monitoring in the Google Cloud console, and create a custom monitoring job for the Bigtable instance to track all changes.
•Create an alert by using webhook endpoints, with the SIEM endpoint as a receiver.

B.•Navigate to the Audit Logs page in the Google Cloud console, and enable Admin Write logs for the Bigtable instance.
•Create a Cloud Functions instance to export logs from Cloud Logging to your SIEM.

C.•Navigate to the Audit Logs page in the Google Cloud console, and enable Data Read, Data Write and Admin Read logs for the Bigtable instance.
•Create a Pub/Sub topic as a Cloud Logging sink destination, and add your SIEM as a subscriber to the topic.

D.•Install the Ops Agent on the Bigtable instance during configuration.
•Create a service account with read permissions for the Bigtable instance.
•Create a custom Dataflow job with this service account to export logs to the company's SIEM system.

**Answer: C**

**Explanation:**

The correct answer is C. Here's why:

Option C provides the most complete solution for logging Bigtable operations, including data access and administrative actions, and sending them to a SIEM system. Google Cloud Audit Logs are the primary mechanism for recording administrative and data access activities within Google Cloud services, including Bigtable.

To capture all relevant operations:

**Data Read logs** are essential to log when PII data is accessed within the table.
**Data Write logs** are needed to log when PII data is modified.
**Admin Read logs** capture any administrative actions involving the Bigtable instance, such as configuration or metadata reads.

By enabling all three, you ensure comprehensive logging coverage. Cloud Logging acts as the central repository for these audit logs.

To get the logs into the SIEM, a common approach is to use Pub/Sub as an intermediary. Cloud Logging can be configured to send logs matching a specific query (in this case, Bigtable-related logs) to a Pub/Sub topic. Then, the SIEM system, which likely supports Pub/Sub subscriptions, can consume the logs from that topic. This approach decouples the logging system from the SIEM and provides a reliable and scalable mechanism for log delivery.

Let's examine why the other options are incorrect:

**Option A:** Cloud Monitoring primarily tracks performance metrics, not audit trails. It wouldn't capture the detail required for SIEM analysis of data and configuration changes and would be far more complex than using Audit Logs.
**Option B:** While Admin Write logs are important for tracking changes in Bigtable configuration, this option doesn't address the need to track data reads and writes containing PII, limiting visibility and making it incomplete for audit purposes. Cloud Function would also add unnecessary complexity compared to a sink.
**Option D:** Installing the Ops Agent on a Bigtable instance is not the appropriate way to capture audit logs. The Ops Agent is primarily for metrics and logs of the underlying VM (not relevant in Bigtable's managed case). Dataflow should be reserved for more complicated ETL operations than simply exporting logs.

**Authoritative Links:**

**Cloud Audit Logs Overview:** https://cloud.google.com/logging/docs/audit
**Bigtable Audit Logging:** https://cloud.google.com/bigtable/docs/audit-logging
**Cloud Logging Sinks:** https://cloud.google.com/logging/docs/export/configure_export
**Pub/Sub Overview:** https://cloud.google.com/pubsub/docs/overview

**Question: 245** CertyIQ

You want to set up a Google Kubernetes Engine cluster. Verifiable node identity and integrity are required for the cluster, and nodes cannot be accessed from the internet. You want to reduce the operational cost of managing your cluster, and you want to follow Google-recommended practices. What should you do?

A.Deploy a private autopilot cluster.
B.Deploy a public autopilot cluster.
C.Deploy a standard public cluster and enable shielded nodes.
D.Deploy a standard private cluster and enable shielded nodes.

**Answer: A**

**Explanation:**

The correct answer is A: Deploy a private autopilot cluster. Let's break down why. Autopilot clusters in Google Kubernetes Engine (GKE) are designed for managed operations, significantly reducing the operational burden by automating node management, scaling, and upgrades. This directly addresses the requirement to reduce operational cost. Furthermore, private clusters, as opposed to public, restrict node access from the internet, fulfilling the need for nodes that cannot be accessed from the internet. Shielded nodes offer verifiable node identity and integrity through Secure Boot, Measured Boot, and virtual Trusted Platform Module (vTPM), ensuring nodes are legitimate. While option D also utilizes shielded nodes and a private cluster, it relies on a

standard cluster. Standard clusters require more hands-on management of the underlying infrastructure, going against the need to reduce operational cost. Autopilot handles node configuration and management, meaning that there is no manual setup required for the shielded nodes. Given these considerations, a private autopilot cluster effectively aligns with all requirements of secure, manageable, and cost-effective cluster.

Authoritative Links:

**GKE Autopilot:** https://cloud.google.com/kubernetes-engine/docs/concepts/autopilot
**GKE Private Clusters:** https://cloud.google.com/kubernetes-engine/docs/how-to/private-clusters
**GKE Shielded Nodes:** https://cloud.google.com/kubernetes-engine/docs/how-to/shielded-nodes

**Question: 246** CertyIQ

Your company wants to migrate their on-premises workloads to Google Cloud. The current on-premises workloads consist of:

•A Flask web application
•A backend API
•A scheduled long-running background job for ETL and reporting

You need to keep operational costs low. You want to follow Google-recommended practices to migrate these workloads to serverless solutions on Google Cloud. What should you do?

A.Migrate the web application to App Engine and the backend API to Cloud Run. Use Cloud Tasks to run your background job on Compute Engine.

B.Migrate the web application to App Engine and the backend API to Cloud Run. Use Cloud Tasks to run your background job on Cloud Run.

C.Run the web application on a Cloud Storage bucket and the backend API on Cloud Run. Use Cloud Tasks to run your background job on Cloud Run.

D.Run the web application on a Cloud Storage bucket and the backend API on Cloud Run. Use Cloud Tasks to run your background job on Compute Engine.

**Answer: B**

**Explanation:**

The correct answer is **B**. Here's why:

Option B best aligns with the principle of leveraging serverless solutions for cost optimization and ease of management. Google Cloud's recommended practice for migrating workloads involves using serverless services whenever possible.

**App Engine:** It's a fully managed platform-as-a-service (PaaS) ideal for hosting web applications, offering automatic scaling and no infrastructure management overhead. This suits the Flask web application well. https://cloud.google.com/appengine
**Cloud Run:** It's a managed compute platform that allows you to run stateless containers, making it a great fit for the backend API. It also scales automatically, handles traffic management, and doesn't require server management. https://cloud.google.com/run
**Cloud Tasks:** It's a fully managed service for asynchronous task execution, perfectly suited to handling scheduled long-running jobs like ETL and reporting. By using Cloud Run to execute the tasks queued by Cloud Tasks, you maintain serverless consistency and don't need to manage VMs. https://cloud.google.com/tasks

**Why other options are incorrect:**

**Option A:** While App Engine and Cloud Run are good choices, using Compute Engine for the background job contradicts the serverless goal. Compute Engine involves VM management, increasing operational overhead

and costs.
**Option C and D:** Cloud Storage is primarily for object storage and unsuitable for directly serving a dynamic web application. It's best practice to host web apps on platforms like App Engine. While Cloud Run is suitable for the API and Cloud Tasks is great for scheduling jobs, the use of Compute Engine in D, and Cloud Storage for the application in C and D, make them unsuitable and not serverless focused.

In summary, Option B provides a fully serverless solution, maximizing the benefits of Google Cloud services, minimizing operational costs and management overhead, and aligning with Google's recommended best practices for workload migration.

**Question: 247** CertyIQ

Your company is moving its continuous integration and delivery (CI/CD) pipeline to Compute Engine instances. The pipeline will manage the entire cloud infrastructure through code. How can you ensure that the pipeline has appropriate permissions while your system is following security best practices?

A.•Attach a single service account to the compute instances.
•Add minimal rights to the service account.
•Allow the service account to impersonate a Cloud Identity user with elevated permissions to create, update, or delete resources.

B.•Add a step for human approval to the CI/CD pipeline before the execution of the infrastructure provisioning.
•Use the human approvals IAM account for the provisioning.

C.•Attach a single service account to the compute instances.
•Add all required Identity and Access Management (IAM) permissions to this service account to create, update, or delete resources.

D.•Create multiple service accounts, one for each pipeline with the appropriate minimal Identity and Access Management (IAM) permissions.
•Use a secret manager service to store the key files of the service accounts.
•Allow the CI/CD pipeline to request the appropriate secrets during the execution of the pipeline.

**Answer: D**

**Explanation:**

Option D is the most secure and aligned with cloud security best practices. It advocates for the principle of least privilege by creating multiple, narrowly scoped service accounts, each dedicated to a specific pipeline. This limits the potential damage if one service account is compromised, as its permissions are restricted. Storing service account keys in a secret manager, such as Google Cloud Secret Manager, is a crucial security measure to avoid hardcoding credentials in the code or environment variables. This practice mitigates the risk of accidental exposure. The CI/CD pipeline requesting secrets dynamically only when required minimizes exposure time, further enhancing security. Options A, B, and C are less secure. Option A suggests impersonation, which can lead to unnecessary complexity and potential privilege escalation issues. Option B relies on human intervention and an IAM user, which isn't scalable or conducive to automation, and is vulnerable to human error. Option C violates the principle of least privilege by granting excessive permissions to a single service account.

**Key Concepts:**

**Principle of Least Privilege:** Granting only the necessary permissions to perform a task.
**Service Accounts:** Special accounts used by applications and services to interact with Google Cloud APIs.
**Secret Manager:** A service to securely store and manage sensitive data like API keys and passwords.
**Identity and Access Management (IAM):** A service to manage access to Google Cloud resources.

**Relevant Links:**

Google Cloud IAM Documentation
Google Cloud Service Accounts Documentation
Google Cloud Secret Manager Documentation
Principle of Least Privilege

**Question: 248** CertyIQ

Your application stores files on Cloud Storage by using the Standard Storage class. The application only requires access to files created in the last 30 days. You want to automatically save costs on files that are no longer accessed by the application. What should you do?

A.Create an object lifecycle on the storage bucket to change the storage class to Archive Storage for objects with an age over 30 days.

B.Create a cron job in Cloud Scheduler to call a Cloud Functions instance every day to delete files older than 30 days.

C.Create a retention policy on the storage bucket of 30 days, and lock the bucket by using a retention policy lock.

D.Enable object versioning on the storage bucket and add lifecycle rules to expire non-current versions after 30 days.

**Answer: A**

**Explanation:**

Option A is the most suitable solution because it leverages Cloud Storage's built-in object lifecycle management capabilities to automatically transition objects to a more cost-effective storage class (Archive Storage) after 30 days of age, fulfilling the requirement of cost optimization for older, infrequently accessed files. This process is serverless and doesn't require manual intervention or external scheduling mechanisms. Object lifecycle management allows the configuration of rules based on object age, creation time, and other properties to perform actions like storage class changes, deletion, and archival, making it ideal for this scenario. Archive Storage is the lowest-cost storage class within Google Cloud Storage, intended for data that is accessed infrequently, often once a year or less, thus reducing overall storage expenditure.

Option B is less efficient, involving a cron job in Cloud Scheduler calling a Cloud Function to delete files. This requires managing a Cloud Function, scheduling infrastructure, and the deletion of data, which might not be desirable if the data is intended for long-term preservation, only requiring less frequent access. Option C utilizes a retention policy, which is more oriented towards data compliance and preservation for a specified duration, not for storage class transitions. Once locked, the retention policy cannot be modified, and data would be inaccessible/deleted after 30 days, which isn't the desired outcome. Option D, enabling versioning and expiring non-current versions, is better suited for data version control and recovery, not cost optimization of older data. Expired versions still consume storage.

Therefore, Object Lifecycle Management provides the most cost-effective and efficient method for transitioning older files to a less expensive storage tier.

**Authoritative Links:**

**Cloud Storage Object Lifecycle Management:** https://cloud.google.com/storage/docs/lifecycle
**Cloud Storage Storage Classes:** https://cloud.google.com/storage/docs/storage-classes
**Cloud Storage Archive Storage:** https://cloud.google.com/storage/docs/storage-classes#archive

**Question: 249** CertyIQ

Your manager asks you to deploy a workload to a Kubernetes cluster. You are not sure of the workload's resource

requirements or how the requirements might vary depending on usage patterns, external dependencies, or other factors. You need a solution that makes cost-effective recommendations regarding CPU and memory requirements, and allows the workload to function consistently in any situation. You want to follow Google-recommended practices. What should you do?

A.Configure the Horizontal Pod Autoscaler for availability, and configure the cluster autoscaler for suggestions.

B.Configure the Horizontal Pod Autoscaler for availability, and configure the Vertical Pod Autoscaler recommendations for suggestions.

C.Configure the Vertical Pod Autoscaler recommendations for availability, and configure the Cluster autoscaler for suggestions.

D.Configure the Vertical Pod Autoscaler recommendations for availability, and configure the Horizontal Pod Autoscaler for suggestions.

**Answer: B**

**Explanation:**

The correct answer is **B: Configure the Horizontal Pod Autoscaler for availability, and configure the Vertical Pod Autoscaler recommendations for suggestions.**

Here's the justification:

The core challenge is determining the optimal CPU and memory resources for a workload with unknown and potentially varying resource needs. Google-recommended practices favor automatic scaling to handle this situation effectively.

**Vertical Pod Autoscaler (VPA):** VPA is designed to analyze the resource usage of pods and provide recommendations for appropriate CPU and memory limits and requests. In recommendation mode, it doesn't automatically adjust pod resources but suggests optimal values. This aligns perfectly with the requirement of making cost-effective recommendations for resource usage and understanding the workload's needs before fully automating resource changes. Using VPA's recommendation feature helps identify resource requirements without disrupting the current running pod.

**Horizontal Pod Autoscaler (HPA):** The HPA automatically adjusts the number of pod replicas based on observed metrics (e.g., CPU utilization, memory usage). This ensures consistent workload functionality in any situation by scaling out the application when demand increases and scaling in when demand decreases, ensuring enough compute resources to maintain availability.

Combining these two autoscalers provides a robust solution: VPA first offers suggestions on pod-level resource needs which you can then act upon. HPA keeps your service available to handle any load by scaling the number of pods up or down. Option A incorrectly mentions that cluster autoscaler can offer resource suggestions and option C and D are incorrect because HPA is intended for availability not suggestions, and VPA is intended for resource recommendations not availability.

Therefore, **option B** optimally addresses both the cost-effective resource recommendation (VPA recommendation mode) and the consistent workload availability (HPA).

**Authoritative Links for further research:**

**Vertical Pod Autoscaler:** https://github.com/kubernetes/autoscaler/tree/master/vertical-pod-autoscaler
**Horizontal Pod Autoscaler:** https://kubernetes.io/docs/tasks/run-application/horizontal-pod-autoscale/
**Kubernetes Autoscaling:** https://kubernetes.io/docs/concepts/cluster-administration/autoscaling/

**Question: 250** CertyIQ

You need to migrate invoice documents stored on-premises to Cloud Storage. The documents have the following

storage requirements:

•Documents must be kept for five years.
•Up to five revisions of the same invoice document must be stored, to allow for corrections.
•Documents older than 365 days should be moved to lower cost storage tiers.

You want to follow Google-recommended practices to minimize your operational and development costs. What should you do?

A.Enable retention policies on the bucket, and use Cloud Scheduler to invoke a Cloud Function to move or delete your documents based on their metadata.

B.Enable retention policies on the bucket, use lifecycle rules to change the storage classes of the objects, set the number of versions, and delete old files.

C.Enable object versioning on the bucket, and use Cloud Scheduler to invoke a Cloud Functions instance to move or delete your documents based on their metadata.

D.Enable object versioning on the bucket, use lifecycle conditions to change the storage class of the objects, set the number of versions, and delete old files.

**Answer: D**

**Explanation:**

Here's a breakdown of why option D is the correct approach for managing invoice documents in Cloud Storage, based on the given requirements, along with supporting concepts:

Option D, "Enable object versioning on the bucket, use lifecycle conditions to change the storage class of the objects, set the number of versions, and delete old files," aligns perfectly with Google Cloud's recommended practices for cost optimization and data management.

**Object Versioning:** Enabling object versioning addresses the requirement of storing up to five revisions of each invoice document. This allows for tracking changes and recovery if needed. When a file is overwritten, the previous version isn't lost, but instead, it's stored as an older version.
**Lifecycle Rules:** Lifecycle rules are a core feature of Cloud Storage to automatically manage objects based on predefined conditions. These conditions can include object age and storage class. In this case, we can configure a lifecycle rule to move documents older than 365 days to a lower-cost storage tier (e.g., from Standard to Nearline, Coldline, or Archive), directly fulfilling the cost-saving requirement.
**Version Control:** Cloud Storage's version control isn't just about retaining deleted items. It can also be configured to maintain a limited number of older versions of a file, addressing the "up to five revisions" requirement while preventing runaway storage costs associated with unlimited versioning.
**Cost Optimization:** By combining versioning and lifecycle rules, we automatically manage storage costs by transitioning older, less frequently accessed data to cheaper storage tiers. This approach avoids manual intervention and is consistent with Google's best practices.
**Simplicity and Scalability:** This method is serverless and doesn't involve external compute resources (like Cloud Functions for this particular task), hence lowering the operational overhead and ensuring scalability without requiring dedicated computing infrastructure.

Option A is incorrect because it relies on Cloud Scheduler and Cloud Functions, adding unnecessary complexity and operational overhead, particularly since this specific task can be accomplished using built-in storage management features. Option B utilizes retention policies, which are primarily for compliance rather than version control and storage tiering based on age. Option C incorrectly uses Cloud Scheduler for a task that can be directly handled by Storage lifecycle rules.

**Authoritative links:**

**Cloud Storage Object Versioning:** https://cloud.google.com/storage/docs/versioning
**Cloud Storage Lifecycle Management:** https://cloud.google.com/storage/docs/lifecycle
**Cloud Storage Storage Classes:** https://cloud.google.com/storage/docs/storage-classes

**Question: 251** CertyIQ

You installed the Google Cloud CLI on your workstation and set the proxy configuration. However, you are worried that your proxy credentials will be recorded in the gcloud CLI logs. You want to prevent your proxy credential from being logged. What should you do?

A.Configure username and password by using gcloud config set proxy/username and gcloud config set proxy/password commands.

B.Encode username and password in sha256 encoding, and save in to a text file. Use filename as a value in the gcloud config set core/custom_ca_certs_file command.

C.Provide values for CLOUDSDK_PROXY_USERNAME and CLOUDSDK_PROXY_PASSWORD in the gcloud CLI tool configuration file.

D.Set the CLOUDSDK_PROXY_USERNAME and CLOUDSDK_PROXY_PASSWORD properties by using environment variables in your command line tool.

**Answer: D**

**Explanation:**

The correct answer is **D. Set the CLOUDSDK_PROXY_USERNAME and CLOUDSDK_PROXY_PASSWORD properties by using environment variables in your command line tool.**

Here's the justification:

The core issue is preventing sensitive proxy credentials from being stored directly within gcloud CLI configuration files or logs. Environment variables offer a secure way to pass credentials to applications without hardcoding them into configuration files. When you set CLOUDSDK_PROXY_USERNAME and CLOUDSDK_PROXY_PASSWORD as environment variables, the gcloud CLI will pick these up at runtime when connecting through the proxy server. These environment variables are typically not logged by the shell or the gcloud CLI itself. This contrasts sharply with options A and C. Option A would store credentials directly in the gcloud CLI config, making them vulnerable if the config file is exposed. Option C attempts to configure proxy settings within a dedicated configuration file, and this would still expose the credentials to log or config file access. Option B is irrelevant to the proxy configuration but relates to using self-signed certificate authorities.

Environment variables provide a transient and more secure method of handling sensitive information. Each time you open a new terminal session or start a new process, the environment variables have to be set. This ensures that your credentials are not persisted in any easily accessible location. Utilizing environment variables is a best practice for managing credentials in various cloud computing tools and programming environments. The gcloud CLI respects these variables, allowing for secure proxy configuration without logging the sensitive credentials directly.

For more details on using environment variables with gcloud CLI, you can consult Google Cloud's official documentation:

gcloud CLI Configuration (Search for "Environment variables")
Using proxies and firewalls (Look for sections on using environment variables for proxy settings)

**Question: 252** CertyIQ

Your company developed an application to deploy on Google Kubernetes Engine. Certain parts of the application are not fault-tolerant and are allowed to have downtime. Other parts of the application are critical and must always be available. You need to configure a Google Kubernetes Engine cluster while optimizing for cost. What should you do?

A.Create a cluster with a single node-pool by using standard VMs. Label he fault-tolerant Deployments as spot_true.

B.Create a cluster with a single node-pool by using Spot VMs. Label the critical Deployments as spot_false.

C.Create a cluster with both a Spot VM node pool and a node pool by using standard VMs. Deploy the critical deployments on the Spot VM node pool and the fault-tolerant deployments on the node pool by using standard VMs.

D.Create a cluster with both a Spot VM node pool and a nods pool by using standard VMs. Deploy the critical deployments on the node pool by using standard VMs and the fault-tolerant deployments on the Spot VM node pool.

**Answer: D**

**Explanation:**

The correct answer is **D**. The key to understanding this problem lies in the differing availability requirements of the application's components and the cost-effectiveness of Spot VMs. Spot VMs are significantly cheaper than standard VMs but can be preempted (terminated) with little notice. This makes them unsuitable for critical components requiring high availability. Therefore, a mixed-node pool strategy is required.

Option D correctly places critical components, which demand continuous operation, on the standard VM node pool, guaranteeing their availability. Simultaneously, it utilizes the cost-effective Spot VM node pool for the fault-tolerant components. These components are designed to withstand interruptions and can tolerate the transient nature of Spot VMs, making option D the most cost-optimized solution that adheres to the availability requirements.

Option A is incorrect because it suggests using labels for different deployments on a single standard VM pool. This does not leverage the cost savings of spot VMs and does not separate deployment based on criticality. Option B's logic is completely reversed, placing critical components on spot VMs which would cause unacceptable downtime. Option C is incorrect as it recommends deploying critical components on Spot VMs, which violates the availability requirements.

**Key Concepts:**

**Spot VMs:** Cost-effective but preemptible virtual machines suitable for fault-tolerant workloads.
**Standard VMs:** Stable virtual machines with guaranteed availability.
**Node Pools:** Groups of nodes with the same configuration within a GKE cluster.
**Deployment:** A Kubernetes object to create multiple replicas of application containers.

**Authoritative Links for Further Research:**

**GKE Node Pools:** https://cloud.google.com/kubernetes-engine/docs/concepts/node-pools
**Spot VMs on GKE:** https://cloud.google.com/kubernetes-engine/docs/how-to/using-spot-vms
**Kubernetes Deployments:** https://kubernetes.io/docs/concepts/workloads/controllers/deployment/

**Question: 253** CertyIQ

You need to deploy an application in Google Cloud using serverless technology. You want to test a new version of the application with a small percentage of production traffic. What should you do?

A.Deploy the application to Cloud Run. Use gradual rollouts for traffic splitting.

B.Deploy the application to Google Kubernetes Engine. Use Anthos Service Mash for traffic splitting.

C.Deploy the application to Cloud Functions. Specify the version number in the functions name.

D.Deploy the application to App Engine. For each new version, create a new service.

**Answer: A**

**Explanation:**

Here's a detailed justification for choosing option A, deploying to Cloud Run with gradual rollouts, as the correct approach for testing a new application version with a small percentage of production traffic using serverless technology on Google Cloud:

Cloud Run is a fully managed serverless platform that excels at running containerized applications. Its built-in features directly support the requirement of traffic splitting. Gradual rollouts in Cloud Run allow you to progressively shift production traffic to a new revision of your application while monitoring its performance. This controlled deployment method minimizes the risk of impacting the entire user base with potential issues from the new version. You can precisely specify the percentage of traffic each version receives, ensuring the new version only receives a small portion initially, allowing for thorough testing. Option B, while utilizing Kubernetes and its associated service mesh capabilities for traffic management is valid, it introduces unnecessary complexity for a serverless deployment scenario. GKE is not a serverless solution. Option C, Cloud Functions, is designed for event-driven functions and lacks native support for gradual rollouts and traffic splitting across different function versions. Though versions can be named, routing is not intended for production-level A/B testing. Option D, App Engine, does allow traffic splitting across different versions but it requires the creation of new services for each deployment which leads to unnecessary resource management. Cloud Run's gradual rollout simplifies this with seamless version updates within the same service. Therefore, utilizing Cloud Run with gradual rollouts directly addresses the prompt's requirements, providing an efficient and less complex solution compared to the other options, in line with serverless best practices and specific platform features.

**Authoritative Links:**

**Cloud Run documentation on gradual rollouts:** https://cloud.google.com/run/docs/rollouts-rollbacks
**Cloud Run overview:** https://cloud.google.com/run/docs
**Serverless on Google Cloud:** https://cloud.google.com/serverless

**Question: 254** CertyIQ

Your company's security vulnerability management policy wants a member of the security team to have visibility into vulnerabilities and other OS metadata for a specific Compute Engine instance. This Compute Engine instance hosts a critical application in your Google Cloud project. You need to implement your company's security vulnerability management policy. What should you do?

A.•Ensure that the Ops Agent is installed on the Compute Engine instance.
•Create a custom metric in the Cloud Monitoring dashboard.
•Provide the security team member with access to this dashboard.

B.•Ensure that the Ops Agent is installed on the Compute Engine instance.
•Provide the security team member roles/osconfig.inventoryViewer permission.

C.•Ensure that the OS Config agent is installed on the Compute Engine instance.
•Provide the security team member roles/osconfig.vulnerabilityReportViewer permission.

D.•Ensure that the OS Config agent is installed on the Compute Engine instance.
•Create a log sink to BigQuery dataset.
•Provide the security team member with access to this dataset.

**Answer: C**

**Explanation:**

The correct answer is **C**, which focuses on using the OS Config agent and granting specific permissions. The company requires visibility into vulnerabilities and OS metadata for a critical Compute Engine instance, managed through its security vulnerability management policy. To achieve this, the OS Config agent needs to be installed. This agent is specifically designed to manage and monitor OS configurations, including

vulnerability scans, and can report this data back to Google Cloud. Option A utilizes the Ops Agent, which is primarily for collecting logs and metrics, not for in-depth vulnerability scanning and reporting. Option B proposes using the OS Config agent but provides the incorrect role (inventoryViewer), which only grants access to basic OS inventory information, not vulnerability reports. Option D involves creating a log sink to BigQuery which, while providing data access, is not the most efficient way for the security team to directly view vulnerability information within the platform. The roles/osconfig.vulnerabilityReportViewer permission, mentioned in option C, is the precise role needed to grant access to the vulnerability reports generated by the OS Config agent, as stated in Google's documentation. This role allows the security team to view the vulnerabilities reported for the specific Compute Engine instance. Therefore, option C provides the correct agent and the appropriate access control, making it the most effective solution for meeting the company's security vulnerability management policy requirements.

**Authoritative Links:**

**OS Config overview:** https://cloud.google.com/compute/docs/osconfig/osconfig-overview
**OS Config roles:** https://cloud.google.com/iam/docs/understanding-roles#osconfig
**Vulnerability Scanning with OS Config:** https://cloud.google.com/compute/docs/osconfig/vulnerability-scanning

**Question: 255** CertyIQ

You want to enable your development team to deploy new features to an existing Cloud Run service in production. To minimize the risk associated with a new revision, you want to reduce the number of customers who might be affected by an outage without introducing any development or operational costs to your customers. You want to follow Google-recommended practices for managing revisions to a service. What should you do?

A.Ask your customers to retry access to your service with exponential backoff to mitigate any potential problems after the new revision is deployed.

B.Gradually roll out the new revision and split customer traffic between the revisions to allow rollback in case a problem occurs.

C.Send all customer traffic to the new revision, and roll back to a previous revision if you witness any problems in production.

D.Deploy your application to a second Cloud Run service, and ask your customers to use the second Cloud Run service.

**Answer: B**

**Explanation:**

The correct answer is **B. Gradually roll out the new revision and split customer traffic between the revisions to allow rollback in case a problem occurs.**

Here's why:

Option B aligns with the best practices for deploying new revisions to a Cloud Run service in production, specifically aiming for risk mitigation. Cloud Run supports traffic splitting, allowing you to direct a percentage of incoming requests to a new revision while the rest continues to be served by the existing, stable revision. This gradual rollout provides a safety net. If issues arise with the new revision, only a small portion of users are affected, enabling a quick rollback to the previous healthy revision without causing widespread outages. This is a common strategy for minimizing the blast radius of potential problems.

Option A is not ideal because while retries with exponential backoff can mitigate some temporary errors, it doesn't actively address the core risk of introducing a flawed revision. Asking clients to retry adds burden and doesn't solve the problem if the issue is within the new release itself. Option C presents a high-risk approach, immediately exposing all users to the new revision. If an issue exists, the outage would be immediately

widespread. Option D is not practical because it introduces unnecessary operational overhead and complicates access for customers by using multiple services. Cloud Run's built-in traffic splitting capability is designed to avoid such complexity.

Traffic splitting allows for canary deployments or A/B testing, all common practices for releasing software safely. This allows monitoring the performance and stability of the new revision with a smaller user base before fully committing to the rollout, adhering to the Google recommended practices.

**Supporting Concepts:**

**Canary Deployments:** Releasing a new version to a small subset of users before making it available to everyone.
**A/B Testing:** Testing different versions of an application on different groups of users.
**Traffic Splitting:** Directing a specified percentage of user traffic to different versions of an application.
**Rollback Strategy:** Having a plan to revert to a previous stable version in case a new release encounters issues.

**Authoritative Links:**

**Cloud Run Traffic Management:** https://cloud.google.com/run/docs/traffic
**Google Cloud Deployment Strategies:** https://cloud.google.com/architecture/devops/devops-deployment-strategies

**Question: 256** CertyIQ

You have deployed an application on a Compute Engine instance. An external consultant needs to access the Linux-based instance. The consultant is connected to your corporate network through a VPN connection, but the consultant has no Google account. What should you do?

A.Instruct the external consultant to use the gcloud compute ssh command line tool by using Identity-Aware Proxy to access the instance.

B.Instruct the external consultant to use the gcloud compute ssh command line tool by using the public IP address of the instance to access it.

C.Instruct the external consultant to generate an SSH key pair, and request the public key from the consultant. Add the public key to the instance yourself, and have the consultant access the instance through SSH with their private key.

D.Instruct the external consultant to generate an SSH key pair, and request the private key from the consultant. Add the private key to the instance yourself, and have the consultant access the instance through SSH with their public key.

**Answer: C**

**Explanation:**

The correct answer is C. Here's why:

Option A is incorrect because Identity-Aware Proxy (IAP) requires a Google account for authentication, which the consultant lacks. IAP is designed to control access based on user identity and would be unsuitable for this scenario.

Option B is incorrect as directly exposing a VM's public IP address for SSH access is a security risk. This opens the instance up to potential attacks from the internet, and should be avoided in most scenarios.

Option C provides the most secure and appropriate method. Here's why it's the correct approach:

**SSH Key-Based Authentication:** Using SSH keys (a public and private key pair) is the standard and recommended way to securely access Linux-based servers. This is more secure than password-based

authentication.
**Public Key Distribution:** The consultant creates the key pair. They keep their private key secret and provide the public key. The public key is then copied and added to the ~/.ssh/authorized_keys file on the Compute Engine instance.
**Private Key Access:** The consultant uses their corresponding private key to authenticate when connecting to the instance via SSH. This is secure because only the private key can decrypt the encrypted connection.
**No Google Account Dependency:** This method does not require a Google account on the consultant's part, allowing them to connect successfully while using their own key pair.
**Principle of Least Privilege:** This approach grants the consultant only the access they require (SSH access), and avoids providing them with more permissions than necessary.

Option D is incorrect because sharing the private key is extremely insecure, like sharing a password. It defeats the purpose of using key-based authentication. The private key must remain secret and only accessible to the consultant.

**Authoritative Links:**

**Google Cloud Documentation on Connecting to Linux VMs using SSH:**
https://cloud.google.com/compute/docs/connect/ssh
**General SSH Best Practices:** A quick search online will provide numerous sources highlighting the importance of private key protection when using SSH key pairs.

**Question: 257** CertyIQ

After a recent security incident, your startup company wants better insight into what is happening in the Google Cloud environment. You need to monitor unexpected firewall changes and instance creation. Your company prefers simple solutions. What should you do?

A.Create a log sink to forward Cloud Audit Logs filtered for firewalls and compute instances to Cloud Storage. Use BigQuery to periodically analyze log events in the storage bucket.

B.Use Cloud Logging filters to create log-based metrics for firewall and instance actions. Monitor the changes and set up reasonable alerts.

C.Install Kibana on a compute instance. Create a log sink to forward Cloud Audit Logs filtered for firewalls and compute instances to Pub/Sub. Target the Pub/Sub topic to push messages to the Kibana instance. Analyze the logs on Kibana in real time.

D.Turn on Google Cloud firewall rules logging, and set up alerts for any insert, update, or delete events.

**Answer: B**

**Explanation:**

The most suitable solution is **B. Use Cloud Logging filters to create log-based metrics for firewall and instance actions. Monitor the changes and set up reasonable alerts.** This approach directly addresses the requirement for simple monitoring of unexpected firewall and instance changes. Cloud Logging allows filtering audit logs to focus on specific events like firewall modifications and instance creation. By creating log-based metrics, you can aggregate these events into numerical data, enabling trend analysis and proactive alerting. This method avoids the complexity of setting up additional services like Cloud Storage, BigQuery, Pub/Sub, or Kibana, aligning with the company's preference for simplicity.

Option A, while feasible, introduces unnecessary complexity. Exporting logs to Cloud Storage and then querying them with BigQuery adds more steps and infrastructure to manage, making it less straightforward. Option C is also more complex by requiring deployment and maintenance of a Kibana instance and utilizing Pub/Sub for log forwarding. Option D only focuses on firewall rules logging and misses instance actions, not fulfilling the complete requirements, while also implying manual setup of alerts rather than using the provided alerting features of Cloud Logging. Therefore, leveraging Cloud Logging's built-in capabilities to

create metrics and alerts is the most streamlined and effective method for this scenario, ensuring simplicity and ease of monitoring without adding management overhead.

Relevant documentation:

**Cloud Logging Overview:** https://cloud.google.com/logging/docs/overview
**Log-based Metrics:** https://cloud.google.com/logging/docs/metrics/
**Creating Alerts from Logs:** https://cloud.google.com/logging/docs/alerting/

**Question: 258** CertyIQ

You are configuring service accounts for an application that spans multiple projects. Virtual machines (VMs) running in the web-applications project need access to BigQuery datasets in the crm-databases project. You want to follow Google-recommended practices to grant access to the service account in the web-applications project. What should you do?

A.Grant "project owner" for web-applications appropriate roles to crm-databases.
B.Grant "project owner" role to crm-databases and the web-applications project.
C.Grant "project owner" role to crm-databases and roles/bigquery.dataViewer role to web-applications.
D.Grant roles/bigquery.dataViewer role to crm-databases and appropriate roles to web-applications.

**Answer: D**

**Explanation:**

The correct approach to grant access between projects using service accounts follows the principle of least privilege. Option D aligns with this principle by granting specific BigQuery data viewing permission (roles/bigquery.dataViewer) directly to the service account in the crm-databases project, enabling VMs in web-applications to access the datasets. It's important to grant roles at the resource level (datasets in this case) within the project where the resource resides, as it's a core best practice in GCP IAM. The service account used by the VMs in web-applications would then be granted the necessary permissions to act on resources in crm-databases, rather than granting overly broad project-level permissions. Option A is incorrect because it grants excessive permissions to the web-applications project at the crm-databases project level. Options B and C are also incorrect because they propose giving broad "project owner" roles, which grant excessive rights beyond what's needed for viewing BigQuery data. This violates the principle of least privilege and poses a security risk. By granting specific roles to the service account at the crm-databases project level, you limit the potential impact of compromised credentials.

Further research:

**Understanding service accounts:** https://cloud.google.com/iam/docs/service-accounts
**IAM best practices:** https://cloud.google.com/iam/docs/best-practices
**BigQuery IAM roles:** https://cloud.google.com/bigquery/docs/access-control
**Principle of least privilege:** https://en.wikipedia.org/wiki/Principle_of_least_privilege

**Question: 259** CertyIQ

Your Dataproc cluster runs in a single Virtual Private Cloud (VPC) network in a single subnetwork with range 172.16.20.128/25. There are no private IP addresses available in the subnetwork. You want to add new VMs to communicate with your cluster using the minimum number of steps. What should you do?

A.Modify the existing subnet range to 172.16.20.0/24.
B.Create a new Secondary IP Range in the VPC and configure the VMs to use that range.

C.Create a new VPC network for the VMs. Enable VPC Peering between the VMs'VPC network and the Dataproc cluster VPC network.

D.Create a new VPC network for the VMs with a subnet of 172.32.0.0/16. Enable VPC network Peering between the Dataproc VPC network and the VMs VPC network. Configure a custom Route exchange.

**Answer: A**

**Explanation:**

The correct answer is A. Modifying the existing subnet range to 172.16.20.0/24 is the most efficient solution to accommodate new VMs within the same VPC network as the Dataproc cluster. Currently, the subnet 172.16.20.128/25 provides only 126 usable IP addresses, which are insufficient if the Dataproc cluster and associated resources have consumed all available private addresses. Expanding the subnet to 172.16.20.0/24 provides 254 usable addresses, offering ample space for new VMs without requiring more complex network configurations. Option B involves configuring secondary IP ranges, which is a good option for a long-term strategy but is less efficient than expanding the current range for this specific problem. Option C and D require creating a new VPC and configuring VPC peering. These solutions involve more steps and management overhead, which contradicts the goal of minimizing steps. Moreover, VPC peering adds complexity, including potentially managing route exchange. Expanding the existing subnet is the most direct solution to the immediate problem of insufficient IP addresses. It avoids the need for new resource creation and simplifies network management within the constraints of the existing infrastructure. This is also a common practice when initially setting up VPC resources. Therefore, expanding the existing subnet (A) is the best solution.

Relevant links:

**Google Cloud VPC Subnet Ranges:** https://cloud.google.com/vpc/docs/vpc
**Google Cloud VPC Secondary IP Ranges:** https://cloud.google.com/vpc/docs/configure-ip-aliases
**Google Cloud VPC Peering:** https://cloud.google.com/vpc/docs/vpc-peering

**Question: 260** CertyIQ

You are building a backend service for an ecommerce platform that will persist transaction data from mobile and web clients. After the platform is launched, you expect a large volume of global transactions. Your business team wants to run SQL queries to analyze the data. You need to build a highly available and scalable data store for the platform. What should you do?

A.Create a multi-region Cloud Spanner instance with an optimized schema.

B.Create a multi-region Firestore database with aggregation query enabled.

C.Create a multi-region Cloud SQL for PostgreSQL database with optimized indexes.

D.Create a multi-region BigQuery dataset with optimized tables.

**Answer: A**

**Explanation:**

The correct answer is A: Create a multi-region Cloud Spanner instance with an optimized schema. Here's why:

Cloud Spanner is a globally distributed, scalable, and strongly consistent database service, making it ideal for handling high volumes of transactional data with strong consistency requirements. Ecommerce transactions demand atomicity, consistency, isolation, and durability (ACID properties), which Spanner guarantees. Multi-region deployment in Spanner ensures high availability and resilience against regional failures, a critical requirement for a globally operating e-commerce platform. While other options have merits, they don't fully satisfy all requirements. Firestore, while scalable, is a NoSQL database which isn't ideal for complex SQL analysis. Cloud SQL, although relational, isn't designed for global scale and high availability as seamlessly as

Spanner. BigQuery is primarily an analytical data warehouse, not a transactional database. It is suitable for running analytical queries after the data is extracted from transactional store, but not for the original storage. An optimized schema within Spanner further enhances query performance and storage efficiency, crucial for handling large transaction volumes. The ability to run SQL queries directly on Spanner simplifies analytics efforts, as per the business team's requirement. Therefore, for a globally scaled e-commerce platform requiring transactional integrity, availability, and SQL query access, Cloud Spanner is the most suitable solution.

**Authoritative Links:**

**Cloud Spanner Overview:** https://cloud.google.com/spanner/docs/overview
**Cloud Spanner Concepts:** https://cloud.google.com/spanner/docs/concepts
**Cloud Spanner Multi-Region Configurations:** https://cloud.google.com/spanner/docs/instance-configurations#multi-region

**Question: 261** CertyIQ

You are in charge of provisioning access for all Google Cloud users in your organization. Your company recently acquired a startup company that has their own Google Cloud organization. You need to ensure that your Site Reliability Engineers (SREs) have the same project permissions in the startup company's organization as in your own organization. What should you do?

A.In the Google Cloud console for your organization, select Create role from selection, and choose destination as the startup company's organization.

B.In the Google Cloud console for the startup company, select Create role from selection and choose source as the startup company's Google Cloud organization.

C.Use the gcloud iam roles copy command, and provide the Organization ID of the startup company's Google Cloud Organization as the destination.

D.Use the gcloud iam roles copy command, and provide the project IDs of all projects in the startup company's organization as the destination.

**Answer: C**

**Explanation:**

The correct answer is **C: Use the gcloud iam roles copy command, and provide the Organization ID of the startup company's Google Cloud Organization as the destination.**

Here's why:

The core requirement is to replicate existing IAM roles (specifically, SRE project permissions) from your organization to the newly acquired startup's organization. Option C directly addresses this using the gcloud iam roles copy command. This command is specifically designed to copy custom IAM roles from a source to a destination, which is perfect for this scenario. You would use the startup organization's ID as the destination, which copies the role to the organization level, making it available across all projects within that organization. This aligns with the question's goal of providing consistent access for the SREs.

Options A and B are incorrect because the "Create role from selection" method, typically used in the console, works at the project level, not the organization level. This approach would require you to repeat the process for every project in the startup company’s organization, which is inefficient and error-prone.

Option D is also incorrect because it attempts to use project IDs instead of the organizational ID with the gcloud iam roles copy command. While this might seem like it could apply to all projects, copying roles to specific projects individually can quickly become unmanageable. More importantly, using the organization ID in the destination enables inheriting the role automatically to any new project under the org. We also need to

ensure that the roles can be inherited. Project-level permissions do not inherit.

Furthermore, it's a best practice to manage permissions consistently across an organization at the organization level itself, when applicable, to ensure consistency and reduce administrative overhead. Using the organizational ID as the destination aligns with this principle, allowing for proper inheritance. The gcloud tool is the preferred way for infrastructure-as-code practices, and ensures replicability and can be easily updated.

**Authoritative Links:**

**gcloud iam roles copy documentation:** https://cloud.google.com/sdk/gcloud/reference/iam/roles/copy
**IAM Overview:** https://cloud.google.com/iam/docs/overview
**Organizations in Google Cloud:** https://cloud.google.com/resource-manager/docs/organizations-overview

**Question: 262** CertyIQ

You need to extract text from audio files by using the Speech-to-Text API. The audio files are pushed to a Cloud Storage bucket. You need to implement a fully managed, serverless compute solution that requires authentication and aligns with Google-recommended practices. You want to automate the call to the API by submitting each file to the API as the audio file arrives in the bucket. What should you do?

A.Create an App Engine standard environment triggered by Cloud Storage bucket events to submit the file URI to the Google Speech-to-TextAPI.

B.Run a Kubernetes job to scan the bucket regularly for incoming files, and call the Speech-to-Text API for each unprocessed file.

C.Run a Python script by using a Linux cron job in Compute Engine to scan the bucket regularly for incoming files, and call the Speech-to-Text API for each unprocessed file.

D.Create a Cloud Function triggered by Cloud Storage bucket events to submit the file URI to the Google Speech-to-Text API.

**Answer: D**

**Explanation:**

The correct answer is **D. Create a Cloud Function triggered by Cloud Storage bucket events to submit the file URI to the Google Speech-to-Text API.**

Here's why:

Cloud Functions are ideal for event-driven, serverless compute tasks. They automatically execute code in response to events, like a new file appearing in a Cloud Storage bucket. This aligns perfectly with the requirement of automating API calls upon file arrival. Cloud Functions are fully managed, meaning Google handles the underlying infrastructure, allowing you to focus solely on the code.

Option A, App Engine, while serverless, is generally better suited for web applications or backend services, not simple event triggers. It requires more configuration and overhead than a Cloud Function.

Options B and C both rely on scheduled polling for changes. Kubernetes jobs and cron jobs in Compute Engine require more manual management and aren't triggered immediately by events. This defeats the goal of a fully managed, event-driven solution. They also aren't cost effective as the instance has to be running constantly just to check the bucket periodically. Polling adds complexity and delay as it won't react immediately upon a new file being added to the Cloud Storage Bucket.

Cloud Functions inherently handle authentication within the Google Cloud environment through service accounts. This makes it easy to securely access the Speech-to-Text API. They are also designed to be scalable, automatically adjusting to handle varying workloads. Cloud Functions are the Google recommended

best practice for these kind of event-triggered tasks.

**Authoritative Links:**

**Cloud Functions:** https://cloud.google.com/functions/docs
**Cloud Storage Triggers for Cloud Functions:** https://cloud.google.com/functions/docs/calling/storage
**Speech-to-Text API:** https://cloud.google.com/speech-to-text/docs
**Serverless Computing on Google Cloud:** https://cloud.google.com/serverless

**Question: 263** CertyIQ

Your customer wants you to create a secure website with autoscaling based on the compute instance CPU load. You want to enhance performance by storing static content in Cloud Storage. Which resources are needed to distribute the user traffic?

A.An external HTTP(S) load balancer with a managed SSL certificate to distribute the load and a URL map to target the requests for the static content to the Cloud Storage backend.

B.An external network load balancer pointing to the backend instances to distribute the load evenly. The web servers will forward the request to the Cloud Storage as needed.

C.An internal HTTP(S) load balancer together with Identity-Aware Proxy to allow only HTTPS traffic.

D.An external HTTP(S) load balancer to distribute the load and a URL map to target the requests for the static content to the Cloud Storage backend. Install the HTTPS certificates on the instance.

**Answer: A**

**Explanation:**

The correct answer is A because it provides the most comprehensive solution for a secure, scalable website with static content offloading. An external HTTP(S) load balancer is essential for distributing incoming user traffic across multiple backend instances, ensuring high availability and responsiveness. This load balancer operates at the application layer (Layer 7) and is capable of routing traffic based on URL paths. This ability allows us to effectively direct requests for static content, such as images and CSS files, to a Cloud Storage bucket via a URL map. This offloads the serving of static assets from the compute instances, reducing their load and improving overall application performance. Furthermore, using a managed SSL certificate with the load balancer automatically handles the complexities of SSL/TLS encryption, enhancing the site's security by securing all traffic to and from the website. Option B is incorrect because using an external network load balancer (Layer 4) would require the web servers to handle both static and dynamic content and is not the most efficient way to load balance a web application that has both kinds of content. Option C is incorrect because an internal load balancer is not designed for external users and is not the right fit for public facing website; It would also require an additional component for handling external traffic. Option D is incorrect because while it uses an external HTTP(S) load balancer and URL maps, it requires manual installation and management of SSL certificates which is less secure and complex compared to managed certificates.

**Key Concepts:**

**External HTTP(S) Load Balancer:** Distributes traffic at the application layer (Layer 7) based on request characteristics such as URL paths. https://cloud.google.com/load-balancing/docs/https
**URL Map:** Directs incoming requests based on URL patterns to different backends, enabling the efficient handling of static content by Cloud Storage. https://cloud.google.com/load-balancing/docs/url-map
**Managed SSL Certificates:** Simplifies the process of securing the website, automatically handling the complexities of certificate management, increasing security and efficiency. https://cloud.google.com/load-balancing/docs/ssl-certificates
**Cloud Storage:** Ideal for storing and serving static assets, reducing the load on compute instances and improving performance and scalability. https://cloud.google.com/storage

**Question: 264** CertyIQ

The core business of your company is to rent out construction equipment at large scale. All the equipment that is being rented out has been equipped with multiple sensors that send event information every few seconds. These signals can vary from engine status, distance traveled, fuel level, and more. Customers are billed based on the consumption monitored by these sensors. You expect high throughput – up to thousands of events per hour per device – and need to retrieve consistent data based on the time of the event. Storing and retrieving individual signals should be atomic. What should you do?

A.Create files in Cloud Storage as data comes in.

B.Create a file in Filestore per device, and append new data to that file.

C.Ingest the data into Cloud SQL. Use multiple read replicas to match the throughput.

D.Ingest the data into Bigtable. Create a row key based on the event timestamp.

**Answer: D**

**Explanation:**

The correct answer is **D. Ingest the data into Bigtable. Create a row key based on the event timestamp.**

Here's why:

Bigtable is a fully managed, scalable NoSQL database service ideal for high-throughput, low-latency applications. It's optimized for time-series data, making it a perfect fit for sensor readings arriving frequently. Creating row keys based on timestamps allows for efficient retrieval of data within a specific timeframe, which is a requirement stated in the question for consistent data based on the time of the event. Bigtable's column-family structure allows atomic updates on individual signals, adhering to the requirement that storing and retrieving individual signals should be atomic.

Options A and B, utilizing Cloud Storage or Filestore, are not suited for this scenario due to lack of indexing and atomic update capabilities and high throughput would be difficult to manage. Cloud SQL (option C) is a relational database, which struggles with high write volumes and may not be efficient for time-series data storage like sensor readings. Additionally, scaling read replicas won't address the write throughput issue which is key for the requirements stated in the question. Bigtable is designed to handle high write volumes with efficient storage and retrieval patterns based on the row keys.

In contrast, Bigtable excels at handling the described data stream. Its ability to store and retrieve time-series data at scale with low latency and atomic operations makes it the ideal choice for this use case.

Further reading:

**Bigtable Overview:** https://cloud.google.com/bigtable/docs/overview
**Time Series Data with Bigtable:** https://cloud.google.com/blog/products/databases/using-bigtable-for-time-series-data
**Choosing a Storage Option:** https://cloud.google.com/storage/docs/choosing-storage-options

**Question: 265** CertyIQ

You just installed the Google Cloud CLI on your new corporate laptop. You need to list the existing instances of your company on Google Cloud. What must you do before you run the gcloud compute instances list command? (Choose two.)

A.Run gcloud auth login, enter your login credentials in the dialog window, and paste the received login token to gcloud CLI.

B.Create a Google Cloud service account, and download the service account key. Place the key file in a folder

on your machine where gcloud CLI can find it.

C.Download your Cloud Identity user account key. Place the key file in a folder on your machine where gcloud CLI can find it.

D.Run gcloud config set compute/zone $my_zone to set the default zone for gcloud CLI.

E.Run gcloud config set project $my_project to set the default project for gcloud CLI.

**Answer: AE**

**Explanation:**

Here's a detailed justification for why options A and E are the correct steps before running gcloud compute instances list:

Option A, running gcloud auth login, is crucial for establishing your identity and granting the gcloud CLI permission to interact with your Google Cloud resources. It initiates an authentication flow, typically opening a web browser where you log in with your Google Cloud user account. Upon successful login, a token is generated, allowing gcloud CLI to access your account's resources on your behalf. This is fundamental as gcloud needs to know who is making the requests to enforce proper authorization and access control. Without this, the CLI would be anonymous and unable to view any project resources.

Option E, running gcloud config set project $my_project, is necessary to specify which project you want to work with. Google Cloud organizes resources within projects, providing a logical grouping and isolation. Before listing instances, the gcloud CLI needs to know which project's instances to retrieve. By setting a default project, you avoid having to specify the project with each command, streamlining your workflow. Without setting a default project, the gcloud compute instances list command would not know where to look for instance data.

Option B is incorrect because service accounts are not generally used for interactive, human-driven use of the gcloud CLI. Service accounts are primarily used by applications and services, not individual users. Option C is wrong because there's no such concept as a "Cloud Identity user account key" for CLI authentication. User authentication relies on tokens obtained via the gcloud auth login process. Option D, while useful for defining a default zone, is not a prerequisite to just listing instances from any zone in any project. The list command without a zone parameter shows all instances across all zones within the project. It is helpful for later commands that operate on specific zone.

In summary, authentication (A) and specifying the project (E) are the two essential initial steps before running any gcloud command that interacts with Google Cloud resources.

Here are some helpful resources for further exploration:

**gcloud auth login documentation:** https://cloud.google.com/sdk/gcloud/reference/auth/login
**gcloud config set documentation:** https://cloud.google.com/sdk/gcloud/reference/config/set
**Google Cloud Projects Overview:** https://cloud.google.com/resource-manager/docs/cloud-platform-resource-hierarchy
**gcloud compute instances list documentation:** https://cloud.google.com/sdk/gcloud/reference/compute/instances/list

**Question: 266** CertyIQ

You are planning to migrate your on-premises data to Google Cloud. The data includes:

•200 TB of video files in SAN storage
•Data warehouse data stored on Amazon Redshift
•20 GB of PNG files stored on an S3 bucket

You need to load the video files into a Cloud Storage bucket, transfer the data warehouse data into BigQuery, and load the PNG files into a second Cloud Storage bucket. You want to follow Google-recommended practices and avoid writing any code for the migration. What should you do?

A.Use gcloud storage for the video files, Dataflow for the data warehouse data, and Storage Transfer Service for the PNG files.

B.Use Transfer Appliance for the videos, BigQuery Data Transfer Service for the data warehouse data, and Storage Transfer Service for the PNG files.

C.Use Storage Transfer Service for the video files, BigQuery Data Transfer Service for the data warehouse data, and Storage Transfer Service for the PNG files.

D.Use Cloud Data Fusion for the video files, Dataflow for the data warehouse data, and Storage Transfer Service for the PNG files.

**Answer: B**

**Explanation:**

The correct answer is **B**. Here's a detailed justification:

**Video Files (200 TB):** Transfer Appliance is the most suitable solution for migrating large datasets like 200 TB of video files from an on-premises SAN storage. It's a physical device that you ship to Google, loaded with your data, bypassing network bandwidth limitations. This is preferable to using gcloud storage due to the size and would be faster. Using Storage Transfer Service is possible, but would likely be slower and more complex for large on-premise data transfers. https://cloud.google.com/transfer-appliance

**Data Warehouse Data (Amazon Redshift):** BigQuery Data Transfer Service is specifically designed for transferring data from various sources, including Amazon Redshift, directly into BigQuery, without code. It handles schema conversion and scheduling of data transfers. Using Dataflow for this task would require code development and is more complex than needed. https://cloud.google.com/bigquery/docs/transfer-service

**PNG Files (20 GB):** Storage Transfer Service is the recommended option to copy data from an S3 bucket to Cloud Storage. It's a managed, fully serverless service that is easy to configure and doesn't require any code to transfer objects between cloud storage services. https://cloud.google.com/storage-transfer-service

Option A is incorrect because gcloud storage is for smaller scale transfers and using Dataflow for Redshift migration is more complex than necessary. Option C is incorrect because the large video file transfer would be more efficient with the Transfer Appliance. Option D is incorrect because Cloud Data Fusion is primarily for building data pipelines with more complex transformations and is unnecessary for a simple large file transfer and using Dataflow for Redshift migration is more complex than necessary.

Therefore, **Transfer Appliance, BigQuery Data Transfer Service, and Storage Transfer Service,** respectively, provide the most direct and Google-recommended way to achieve the migration without coding.

**Question: 267** CertyIQ

You want to deploy a new containerized application into Google Cloud by using a Kubernetes manifest. You want to have full control over the Kubernetes deployment, and at the same time, you want to minimize configuring infrastructure. What should you do?

A.Deploy the application on GKE Autopilot.

B.Deploy the application on Cloud Run.

C.Deploy the application on GKE Standard.

D.Deploy the application on Cloud Functions.

**Answer: A**

**Explanation:**

The correct answer is A, deploying the application on GKE Autopilot. GKE Autopilot is a managed Kubernetes service that abstracts away much of the underlying infrastructure management. This aligns with the requirement to minimize infrastructure configuration while retaining full control over the Kubernetes deployment through manifests. Options B, C, and D are less suitable. Cloud Run (B) is a serverless platform that doesn't provide the same level of direct control over Kubernetes as GKE Autopilot. It's more suitable for stateless applications where you don't need granular control over the Kubernetes infrastructure. GKE Standard (C) provides more flexibility, but also places the burden of managing node infrastructure on the user, which contradicts the desire to minimize configuration. Cloud Functions (D) is a serverless function-as-a-service offering and is not suitable for deploying a containerized application using Kubernetes manifests. GKE Autopilot manages the control plane and nodes, ensuring high availability and allowing users to focus on the application's manifests. This greatly simplifies infrastructure operations. Therefore, GKE Autopilot is the best option when you require full Kubernetes control via manifest deployment with minimal infrastructure configuration overhead. Autopilot handles scaling, patching, and other routine operations, simplifying deployment.

Authoritative Links for further research:

**Google Kubernetes Engine (GKE):** https://cloud.google.com/kubernetes-engine
**GKE Autopilot:** https://cloud.google.com/kubernetes-engine/docs/concepts/autopilot
**Cloud Run:** https://cloud.google.com/run
**GKE Standard:** https://cloud.google.com/kubernetes-engine/docs/concepts/kubernetes-engine-overview#standard_clusters
**Cloud Functions:** https://cloud.google.com/functions

**Question: 268** CertyIQ

Your team is building a website that handles votes from a large user population. The incoming votes will arrive at various rates. You want to optimize the storage and processing of the votes. What should you do?

A.Save the incoming votes to Firestore. Use Cloud Scheduler to trigger a Cloud Functions instance to periodically process the votes.

B.Use a dedicated instance to process the incoming votes. Send the votes directly to this instance.

C.Save the incoming votes to a JSON file on Cloud Storage. Process the votes in a batch at the end of the day.

D.Save the incoming votes to Pub/Sub. Use the Pub/Sub topic to trigger a Cloud Functions instance to process the votes.

**Answer: D**

**Explanation:**

Option D, saving votes to Pub/Sub and triggering Cloud Functions, is the most suitable solution for handling variable vote rates due to its inherent decoupling and scalability. Pub/Sub acts as a message queue, allowing votes to be ingested regardless of processing speed. This decouples the vote submission from processing, preventing backpressure on the client application when vote rates spike. Cloud Functions, triggered by the Pub/Sub topic, can scale automatically based on message volume, enabling efficient processing even during high traffic. Firestore (Option A) is a NoSQL database primarily for storage and retrieval; using it for direct vote ingestion might not be as performant for variable rates as a message queue. Dedicated instances (Option B) can be costly for handling variable loads, often resulting in over-provisioning or bottlenecks. Similarly, storing votes in a JSON file on Cloud Storage (Option C) and processing them in batches introduces unnecessary latency and limits real-time insights. Pub/Sub's asynchronous message passing and Cloud

Functions' serverless nature make Option D highly responsive, scalable, and cost-effective. This allows for near real-time processing of votes as they are received, providing better data analysis and insights. Furthermore, the managed nature of both Pub/Sub and Cloud Functions reduces operational overhead.

**Supporting Links:**

**Pub/Sub:** https://cloud.google.com/pubsub/docs/overview
**Cloud Functions:** https://cloud.google.com/functions/docs/concepts/overview
**Serverless Computing:** https://cloud.google.com/serverless
**Message Queues:** https://en.wikipedia.org/wiki/Message_queue

**Question: 269** CertyIQ

You are deploying an application on Google Cloud that requires a relational database for storage. To satisfy your company's security policies, your application must connect to your database through an encrypted and authenticated connection that requires minimal management and integrates with Identity and Access Management (IAM). What should you do?

A.Deploy a Cloud SQL database with the SSL mode set to encrypted only, configure SSL/TLS client certificates, and configure a database user and password.

B.Deploy a Cloud SQL database with the SSL mode set to encrypted only, configure SSL/TLS client certificates, and configure IAM database authentication.

C.Deploy a Cloud SQL database and configure IAM database authentication. Access the database through the Cloud SQL Auth Proxy.

D.Deploy a Cloud SQL database and configure a database user and password. Access the database through the Cloud SQL Auth Proxy.

**Answer: C**

**Explanation:**

Option C is the correct approach because it leverages Google Cloud's managed services and security features to meet the specified requirements. Cloud SQL provides a managed relational database service, reducing administrative overhead. IAM database authentication ensures that access to the database is controlled through Google Cloud's IAM system, aligning with the company's security policies and allowing for granular permission management using existing IAM roles. The Cloud SQL Auth Proxy facilitates secure, encrypted connections to the database without requiring client-side certificates, thus simplifying management. The proxy handles authentication and encryption, ensuring a secure channel and minimizing the need to manage complex client-side configurations. This method also integrates well with Google Cloud's ecosystem.

Option A, while utilizing SSL for encryption, requires managing client certificates, which is cumbersome and adds administrative burden. Option B includes IAM database authentication but also client certificate management. Option D does not use IAM, it still requires database user and password management and thus fails to leverage the more secure and manageable approach provided by IAM. Using a database user and password, although simpler in initial setup, doesn't offer the centralized access control provided by IAM.

Therefore, utilizing Cloud SQL with IAM database authentication, accessed through the Cloud SQL Auth Proxy is the ideal solution, providing a secure, scalable, and easily manageable solution that aligns with modern security best practices for cloud environments.

**Authoritative Links:**

**Cloud SQL IAM database authentication:** https://cloud.google.com/sql/docs/mysql/iam-authentication
**Cloud SQL Auth Proxy:** https://cloud.google.com/sql/docs/mysql/connect-auth-proxy

**Question: 270** CertyIQ

You have two Google Cloud projects: project-a with VPC vpc-a (10.0.0.0/16) and project-b with VPC vpc-b (10.8.0.0/16). Your frontend application resides in vpc-a and the backend API services are deployed in vpc-b. You need to efficiently and cost-effectively enable communication between these Google Cloud projects. You also want to follow Google-recommended practices. What should you do?

A.Create an OpenVPN connection between vpc-a and vpc-b.

B.Create VPC Network Peering between vpc-a and vpc-b.

C.Configure a Cloud Router in vpc-a and another Cloud Router in vpc-b.

D.Configure a Cloud Interconnect connection between vpc-a and vpc-b.

**Answer: B**

**Explanation:**

The correct answer is **B. Create VPC Network Peering between vpc-a and vpc-b.** Here's why:

VPC Network Peering allows private IP address connectivity across two VPC networks, regardless of whether they reside in the same organization or different Google Cloud projects. This aligns perfectly with the requirement of enabling communication between the frontend in vpc-a and the backend in vpc-b. Peering provides a secure and efficient communication pathway without traversing the public internet. It's also a cost-effective solution as it doesn't involve the complexities and expenses associated with options like Cloud Interconnect or VPN tunnels. VPC Network Peering is a fully managed service, minimizing administrative overhead. It's also a recommended practice by Google for connecting VPC networks within Google Cloud.

Option A, OpenVPN, introduces unnecessary complexity and operational overhead compared to native peering. It requires managing and maintaining VPN gateways and tunnels, which adds to the cost and potential failure points. Option C, Cloud Routers, are typically used for dynamic routing over VPN or interconnect connections, not for direct peering between VPCs. While Cloud Routers would be used to establish connectivity via VPN or Interconnect, in this scenario such complexity is not needed. Option D, Cloud Interconnect, is primarily used for connecting on-premises networks to Google Cloud, which is not a requirement here. Using interconnect for just communication between two Google Cloud VPCs would be overkill and inefficient. Therefore, VPC Network Peering is the most suitable solution because of its simplicity, security, cost-effectiveness, and alignment with Google's recommended practices for inter-VPC communication.

**Authoritative Links:**

**VPC Network Peering:** https://cloud.google.com/vpc/docs/vpc-peering
**Choosing a Network Connectivity Solution:** https://cloud.google.com/architecture/network-connectivity

**Question: 271** CertyIQ

Your company is running a critical workload on a single Compute Engine VM instance. Your company's disaster recovery policies require you to back up the entire instance's disk data every day. The backups must be retained for 7 days. You must configure a backup solution that complies with your company's security policies and requires minimal setup and configuration. What should you do?

A.Configure the instance to use persistent disk asynchronous replication.

B.Configure daily scheduled persistent disk snapshots with a retention period of 7 days.

C.Configure Cloud Scheduler to trigger a Cloud Function each day that creates a new machine image and deletes machine images that are older than 7 days.

D.Configure a bash script using gsutil to run daily through a cron job. Copy the disk's files to a Cloud Storage bucket with archive storage class and an object lifecycle rule to delete the objects after 7 days.

**Answer: B**

**Explanation:**

The correct answer is **B: Configure daily scheduled persistent disk snapshots with a retention period of 7 days.**

Here's why:

Option B directly addresses the requirements with minimal setup. Persistent disk snapshots are a native Google Cloud feature specifically designed for backing up disk data. Scheduling these snapshots daily ensures regular backups, and setting a 7-day retention period automatically manages the lifecycle, adhering to the policy. Snapshots are incremental, meaning they only store the changes since the last snapshot, making them efficient in terms of both storage and time. This solution requires minimal scripting or custom code, aligning with the requirement of minimal setup and configuration.

Option A, using asynchronous replication, is primarily for high availability and disaster recovery where a secondary instance is kept in sync. It doesn't fulfill the requirement of a 7-day retention. Option C, using machine images, is primarily for creating new VMs from a template; it is not as efficient for data backups, and managing the lifecycle through a Cloud Function adds unnecessary complexity. Option D, a custom script using gsutil, involves more manual configuration, introduces potential security vulnerabilities, and isn't as efficient as Google's managed snapshot service. Also, copying the disk's files is not a good approach for a disk that is in use and may result in data inconsistencies.

Therefore, option B is the most efficient, secure, and straightforward method for daily backups of the VM instance's disk data with a 7-day retention, using built-in Google Cloud functionalities, minimizing setup and configuration, which fulfills the problem statement.

**Authoritative Links for Further Research:**

**Google Cloud Persistent Disk Snapshots:** https://cloud.google.com/compute/docs/disks/snapshots
**Scheduling Snapshots:** https://cloud.google.com/compute/docs/disks/scheduled-snapshots

**Question: 272** CertyIQ

Your company requires that Google Cloud products are created with a specific configuration to comply with your company's security policies. You need to implement a mechanism that will allow software engineers at your company to deploy and update Google Cloud products in a preconfigured and approved manner. What should you do?

A.Create Java packages that utilize the Google Cloud Client Libraries for Java to configure Google Cloud products. Store and share the packages in a source code repository.

B.Create bash scripts that utilize the Google Cloud CLI to configure Google Cloud products. Store and share the bash scripts in a source code repository.

C.Use the Google Cloud APIs by using curl to configure Google Cloud products. Store and share the curl commands in a source code repository.

D.Create Terraform modules that utilize the Google Cloud Terraform Provider to configure Google Cloud products. Store and share the modules in a source code repository.

**Answer: D**

**Explanation:**

The correct answer is **D. Create Terraform modules that utilize the Google Cloud Terraform Provider to configure Google Cloud products. Store and share the modules in a source code repository.**

Here's why:

Terraform, an Infrastructure as Code (IaC) tool, allows you to define and manage your infrastructure using declarative configuration files. This approach perfectly fits the requirement of deploying and updating Google Cloud products in a pre-configured and approved manner. Terraform modules encapsulate reusable infrastructure components, meaning the company's security policies can be codified into these modules. Software engineers can then utilize these pre-approved modules without needing to individually configure each Google Cloud product, guaranteeing compliance. Storing these modules in a source code repository enables version control, collaborative development, and consistent deployments across the organization.

Options A, B, and C are less suitable. While Java packages (A) and bash scripts (B) can achieve configuration, they lack the declarative nature and infrastructure management capabilities that Terraform provides. They require procedural coding which makes it more complex to maintain consistency and less reliable compared to Terraform. Using curl with Google Cloud APIs (C) is even more manual, error-prone, and doesn't address state management effectively, leading to a lack of idempotency and making it difficult to ensure that configuration is consistent and repeatable across different deployments and environments.

Terraform modules provide a consistent, repeatable, and auditable method for managing Google Cloud resources, aligning with best practices for infrastructure management. It also enables easy changes to the configurations, while keeping track of the configuration history.

**Authoritative Links:**

**Terraform:** https://www.terraform.io/
**Google Cloud Provider for Terraform:** https://registry.terraform.io/providers/hashicorp/google/latest/docs
**Infrastructure as Code (IaC):** https://aws.amazon.com/what-is/infrastructure-as-code/

In summary, Terraform modules offer the most effective solution for creating and sharing pre-configured infrastructure while adhering to security policies and enabling software engineers to deploy resources in a controlled and consistent manner.

**Question: 273** CertyIQ

You are a Google Cloud organization administrator. You need to configure organization policies and log sinks on Google Cloud projects that cannot be removed by project users to comply with your company's security policies. The security policies are different for each company department. Each company department has a user with the Project Owner role assigned to their projects. What should you do?

A.Use a standard naming convention for projects that includes the department name. Configure organization policies on the organization and log sinks on the projects.

B.Use a standard naming convention for projects that includes the department name. Configure both organization policies and log sinks on the projects.

C.Organize projects under folders for each department. Configure both organization policies and log sinks on the folders.

D.Organize projects under folders for each department. Configure organization policies on the organization and log sinks on the folders.

**Answer: C**

**Explanation:**

Here's a detailed justification for why option C is the correct answer:

Option C, organizing projects under folders for each department and configuring both organization policies and log sinks on the folders, is the most effective approach for several reasons. Folders in Google Cloud

Platform (GCP) provide a hierarchical structure that mirrors organizational structures. By placing projects of each department under their respective folders, you can apply consistent policies and logging configurations to the entire department at once. This eliminates the need to configure each project individually, which would be time-consuming and error-prone.

Organization policies are inherited down the resource hierarchy. When set at the folder level, these policies automatically apply to all projects within that folder. This ensures that security policies are enforced consistently across each department's projects, even if project owners have different levels of technical expertise or focus. Similarly, log sinks configured at the folder level collect logs from all resources within that folder, including all contained projects. This provides a centralized location for security monitoring and analysis for the entire department. Crucially, neither organization policies nor log sinks are directly modifiable by project owners if applied at a level above them in the hierarchy. This achieves the requirement that project users cannot remove these configurations.

Option A and B are inadequate because directly managing organization policies and log sinks on the projects themselves would be problematic. Project owners would have the ability to remove these configurations because of their Project Owner role which grants permissions on project resource management. Option D is incorrect, while it employs folders for organization, the organization policies should be applied at the folder level to ensure consistency, and that each department can have their own defined policies. It incorrectly states that organization policies should be applied at the organization level which does not fulfill the use case that different policies should be applied to different departments.In summary, Option C's use of folders for departmental grouping coupled with folder-level policy and log sink configuration is the only answer that properly meets the need for consistent, centrally managed, and non-removable security and logging policies, fulfilling all of the specific requirement of the use case.

**Authoritative Links:**

**Resource hierarchy:** https://cloud.google.com/resource-manager/docs/cloud-platform-resource-hierarchy
**Organization policies:** https://cloud.google.com/resource-manager/docs/organization-policy/overview
**Cloud Logging sinks:** https://cloud.google.com/logging/docs/reference/v2/rest/v2/sinks
**Understanding IAM hierarchy:** https://cloud.google.com/iam/docs/overview#hierarchy

**Question: 274** CertyIQ

You are deploying a web application using Compute Engine. You created a managed instance group (MIG) to host the application. You want to follow Google-recommended practices to implement a secure and highly available solution. What should you do?

A.Use SSL proxy load balancing for the MIG and an A record in your DNS private zone with the load balancer's IP address.

B.Use SSL proxy load balancing for the MIG and a CNAME record in your DNS public zone with the load balancer’s IP address.

C.Use HTTP(S) load balancing for the MIG and a CNAME record in your DNS private zone with the load balancer’s IP address.

D.Use HTTP(S) load balancing for the MIG and an A record in your DNS public zone with the load balancer’s IP address.

**Answer: D**

**Explanation:**

The correct answer is D: Use HTTP(S) load balancing for the MIG and an A record in your DNS public zone with the load balancer’s IP address.

Here's the justification:

**HTTP(S) Load Balancing:** For web applications, HTTP(S) load balancing is the recommended approach. It provides features like SSL/TLS termination, content-based routing, and session affinity, which are crucial for modern web applications. SSL proxy load balancing is generally used for TCP-based applications, not HTTP(S).
**Managed Instance Groups (MIGs):** MIGs provide scalability and high availability. Using a load balancer in front of a MIG allows traffic to be distributed evenly across the instances, ensuring that the application remains available even if some instances fail.
**Public DNS Zone:** Web applications intended for public access require a DNS record in a public zone. This allows users from anywhere on the internet to resolve the application's domain name to the load balancer's IP address. Private zones are used for internal networks.
**A Record:** An A record maps a domain name to the IP address of the load balancer. It is the appropriate DNS record type when the resource (the load balancer) has a fixed IP address. CNAME records, on the other hand, alias one domain name to another, which is not directly applicable for a load balancer's IP.

Options A and B are incorrect because they use SSL proxy load balancing, which is not optimal for web traffic. Option C uses a private DNS zone, which would make the web application inaccessible to public users. Therefore, option D aligns with best practices for web application deployments by ensuring scalability, availability, security through TLS encryption, and public accessibility.

**Authoritative Links:**

Google Cloud Load Balancing: https://cloud.google.com/load-balancing/docs/
Managed Instance Groups: https://cloud.google.com/compute/docs/instance-groups/
Cloud DNS Overview: https://cloud.google.com/dns/docs/overview

**Question: 275** CertyIQ

You have several hundred microservice applications running in a Google Kubernetes Engine (GKE) cluster. Each microservice is a deployment with resource limits configured for each container in the deployment. You've observed that the resource limits for memory and CPU are not appropriately set for many of the microservices. You want to ensure that each microservice has right sized limits for memory and CPU. What should you do?

A.Configure a Vertical Pod Autoscaler for each microservice.
B.Modify the cluster's node pool machine type and choose a machine type with more memory and CPU.
C.Configure a Horizontal Pod Autoscaler for each microservice.
D.Configure GKE cluster autoscaling.

**Answer: A**

**Explanation:**

The correct answer is A, configuring a Vertical Pod Autoscaler (VPA) for each microservice. Here's why: VPAs automatically adjust the CPU and memory requests and limits of pods within a Kubernetes deployment based on their observed resource consumption. This directly addresses the problem of incorrectly sized resource limits. Option B, changing the node pool machine type, increases the cluster's overall capacity but doesn't dynamically resize individual microservice pods. Option C, using Horizontal Pod Autoscaler (HPA), scales the number of pod replicas based on CPU or memory utilization, not the resource allocation of each pod. Option D, GKE cluster autoscaling, adds or removes nodes based on resource demand, not on per-pod adjustments. The VPA analyzes each pod's performance over time, recommending and potentially applying appropriate resource limits. This ensures each microservice receives the necessary resources without over-provisioning, which can waste cluster resources. Unlike manual adjustments, the VPA provides an automated and continuous process for right-sizing resources. Therefore, option A is the only solution that directly addresses the need to ensure each microservice has right-sized resource limits for memory and CPU. It's a core Kubernetes feature designed for this very problem.

Further reading:

**Kubernetes Vertical Pod Autoscaler:** https://github.com/kubernetes/autoscaler/tree/master/vertical-pod-autoscaler
**GKE Documentation on VPA:** https://cloud.google.com/kubernetes-engine/docs/how-to/vertical-pod-autoscaling

**Question: 276** CertyIQ

Your company uses BigQuery to store and analyze data. Upon submitting your query in BigQuery, the query fails with a quotaExceeded error. You need to diagnose the issue causing the error. What should you do? (Choose two.)

A.Use BigQuery BI Engine to analyze the issue.
B.Use the INFORMATION_SCHEMA views to analyze the underlying issue.
C.Configure Cloud Trace to analyze the issue.
D.Search errors in Cloud Audit Logs to analyze the issue.
E.View errors in Cloud Monitoring to analyze the issue.

**Answer: BD**

**Explanation:**

Okay, let's break down why options B and D are the correct choices for diagnosing a BigQuery quotaExceeded error, and why the others aren't.

When a BigQuery query fails due to exceeding a quota, the core issue is that the query is attempting to use more resources than are currently allowed within the project. Option B, using **INFORMATION_SCHEMA views**, directly addresses this. These views provide metadata about BigQuery jobs, including resource consumption, query history, and errors encountered. By querying INFORMATION_SCHEMA.JOBS, you can examine which queries are consuming the most resources and if previous queries contributed to hitting the quota limit. This is a direct way to pinpoint where the resource usage is happening within BigQuery itself. https://cloud.google.com/bigquery/docs/information-schema-jobs

Option D, searching for errors in **Cloud Audit Logs**, is also highly relevant. Cloud Audit Logs records administrative activities and access to Google Cloud services. While not solely focused on resource utilization, it captures log entries for operations like query submissions and failures, including quota errors. These logs provide vital context around the time of the error, the user who initiated the job, and specific details about the quota exceeded. This helps in auditing and understanding patterns that lead to these kinds of issues. https://cloud.google.com/logging/docs/audit

Option A, using **BigQuery BI Engine**, is incorrect. BI Engine is designed for interactive analysis with sub-second query response time. While it could speed up data analysis once the underlying quota issue is resolved, it doesn't directly help diagnose the root cause of a quotaExceeded error. Option C, **Cloud Trace**, is primarily used for tracking latency and timing within applications. It doesn't provide insight into quota usage in BigQuery. And finally, while Cloud Monitoring(option E) can monitor BigQuery resource consumption, the logs in option D provide more immediate and relevant information for quota error analysis. Cloud Monitoring would be more for setting up alerts for exceeding quotas.

Therefore, the combination of using INFORMATION_SCHEMA to analyze BigQuery-specific job data and using Cloud Audit Logs to see administrative activity associated with the error provides the most effective method to diagnose a quotaExceeded error in BigQuery.

**Question: 277** CertyIQ

Your team has developed a stateless application which requires it to be run directly on virtual machines. The application is expected to receive a fluctuating amount of traffic and needs to scale automatically. You need to deploy the application. What should you do?

A.Deploy the application on a managed instance group and configure autoscaling.

B.Deploy the application on a Kubernetes Engine cluster and configure node pool autoscaling.

C.Deploy the application on Cloud Functions and configure the maximum number instances.

D.Deploy the application on Cloud Run and configure autoscaling.

**Answer: A**

**Explanation:**

The correct answer is **A. Deploy the application on a managed instance group and configure autoscaling.** Here's why:

Managed Instance Groups (MIGs) are ideal for deploying applications on virtual machines (VMs) that require automatic scaling and high availability. They allow you to manage a collection of identical VMs as a single entity. Autoscaling within a MIG automatically adjusts the number of VMs based on demand, ensuring your application can handle fluctuating traffic loads. This feature addresses the requirement of the application needing to scale automatically. Since the application needs to run on VMs, a MIG is a direct fit.

Option B, Kubernetes Engine, introduces a layer of orchestration which isn't needed if the application can run directly on VMs. While Kubernetes allows node pool autoscaling, it's more complex than necessary for this use case and adds overhead. Option C, Cloud Functions, is for event-driven serverless functions, not for deploying applications directly onto virtual machines. Cloud Run (Option D) is designed for containerized applications, and it also abstracts away the VM layer, not fulfilling the requirement for running the app on virtual machines. Therefore, the most efficient and fitting solution for deploying a stateless application on VMs with automatic scaling is by using a managed instance group with configured autoscaling.

For further research, refer to the official Google Cloud documentation:

**Managed Instance Groups:** https://cloud.google.com/compute/docs/instance-groups
**Autoscaling:** https://cloud.google.com/compute/docs/autoscaling

**Question: 278** CertyIQ

Your web application is hosted on Cloud Run and needs to query a Cloud SQL database. Every morning during a traffic spike, you notice API quota errors in Cloud SQL logs. The project has already reached the maximum API quota. You want to make a configuration change to mitigate the issue. What should you do?

A.Modify the minimum number of Cloud Run instances.

B.Use traffic splitting.

C.Modify the maximum number of Cloud Run instances.

D.Set a minimum concurrent requests environment variable for the application.

**Answer: A**

**Explanation:**

The correct answer is A, modifying the minimum number of Cloud Run instances. Cloud Run scales automatically based on incoming requests. However, during a traffic spike, if the minimum number of instances is insufficient, it can take time for Cloud Run to scale up, resulting in a surge of new connections to

the Cloud SQL database. This sudden influx of connections can overwhelm the database, leading to API quota errors. By increasing the minimum number of instances, you ensure that a pool of ready-to-serve instances is already available, mitigating the need for rapid scaling and reducing the pressure on the Cloud SQL API. This proactive approach avoids the sudden burst of connection requests during traffic spikes, allowing Cloud SQL to handle a more gradual increase in demand. Options B, C, and D, while potentially useful for other situations, do not directly address the issue of initial connection overload during traffic spikes. Traffic splitting (B) distributes traffic across different versions of a service but doesn't address the immediate scaling issues. Modifying the maximum number of Cloud Run instances (C) limits scaling during high traffic, exacerbating the problem. Setting a minimum concurrent requests environment variable (D) controls concurrency within a single instance but doesn't help in scaling out enough instances initially.

Here are some resources to understand more about Cloud Run and its scaling behavior:

**Cloud Run Scaling:** https://cloud.google.com/run/docs/configuring/min-max-instances
**Cloud Run Autoscaling:** https://cloud.google.com/run/docs/configuring/autoscaling
**Cloud SQL Connection Management:** https://cloud.google.com/sql/docs/postgres/manage-connections

**Question: 279** CertyIQ

You need to deploy a single stateless web application with a web interface and multiple endpoints. For security reasons, the web application must be reachable from an internal IP address from your company's private VPC and on-premises network. You also need to update the web application multiple times per day with minimal effort and want to manage a minimal amount of cloud infrastructure. What should you do?

A.Deploy the web application on Google Kubernetes Engine standard edition with an internal ingress.
B.Deploy the web application on Cloud Run with Private Google Access configured.
C.Deploy the web application on Cloud Run with Private Service Connect configured.
D.Deploy the web application to GKE Autopilot with Private Google Access configured.

**Answer: B**

**Explanation:**

The correct answer is **B. Deploy the web application on Cloud Run with Private Google Access configured.** Here's why:

Cloud Run is a fully managed serverless platform, perfectly aligning with the requirement of managing minimal cloud infrastructure and enabling frequent, effortless deployments. It abstracts away the underlying server management, allowing developers to focus on code. This satisfies the need for multiple daily updates with minimal effort. Cloud Run also natively supports containerized applications, simplifying deployments.

Private Google Access enables Cloud Run services to access Google APIs and services over Google's internal network, addressing the need for internal IP address access. This ensures the application is reachable from the company's private VPC and on-premises network when configured correctly with Cloud VPN or Cloud Interconnect.

Option A, GKE standard edition, requires more infrastructure management and introduces unnecessary complexity for a single stateless application. Option C, Cloud Run with Private Service Connect, is an option for different use cases and introduces unnecessary complexity for accessing a internal private VPC. Option D, GKE Autopilot, while less complex than standard GKE, still involves managing Kubernetes constructs when the serverless approach offered by Cloud Run is more efficient for this scenario.

In conclusion, Cloud Run with Private Google Access provides the best balance of managed infrastructure, easy deployments, and secure internal network access, making it the optimal choice for this use case.

**Authoritative Links:**

**Cloud Run:** https://cloud.google.com/run
**Private Google Access:** https://cloud.google.com/vpc/docs/private-google-access
**Cloud Run Networking:** https://cloud.google.com/run/docs/configuring/vpc

**Question: 280** CertyIQ

You use Cloud Logging to capture application logs. You now need to use SQL to analyze the application logs in Cloud Logging, and you want to follow Google-recommended practices. What should you do?

A.Develop SQL queries by using Gemini for Google Cloud.
B.Enable Log Analytics for the log bucket and create a linked dataset in BigQuery.
C.Create a schema for the storage bucket and run SQL queries for the data in the bucket.
D.Export logs to a storage bucket and create an external view in BigQuery.

**Answer: B**

**Explanation:**

The correct answer is **B: Enable Log Analytics for the log bucket and create a linked dataset in BigQuery.** This approach aligns with Google's recommended practices for analyzing Cloud Logging data with SQL.

Here's why:

**Log Analytics:** This feature transforms log data into a structured, queryable format within Cloud Logging itself. It allows you to run SQL-like queries directly on your logs without needing to export them first. https://cloud.google.com/logging/docs/log-analytics

**BigQuery Linked Datasets:** Log Analytics can create "linked datasets" in BigQuery. These datasets don't physically move the log data. Instead, they act as views that allow you to use BigQuery's robust SQL capabilities on the underlying log data managed by Cloud Logging. This avoids data duplication and minimizes operational overhead. https://cloud.google.com/logging/docs/log-analytics/bigquery-linked-datasets

**Google Recommendation:** This method is explicitly recommended by Google as a best practice for SQL-based log analysis. It leverages the tight integration between Cloud Logging and BigQuery, providing a performant, cost-effective, and manageable solution.

Let's examine why the other options are less suitable:

**A: Develop SQL queries by using Gemini for Google Cloud.** While Gemini for Google Cloud can assist in generating queries, it doesn't address the underlying data pipeline for enabling SQL analysis of logs. Gemini is an AI assistant; it doesn't create linked datasets.
**C: Create a schema for the storage bucket and run SQL queries for the data in the bucket.** While you can export logs to Cloud Storage, querying them directly from storage is less efficient and more difficult than using linked datasets. You would need a more complex setup and manual schema management. This isn't the recommended path for SQL-based analysis from Google.
**D: Export logs to a storage bucket and create an external view in BigQuery.** This approach is a valid alternative but less efficient. It requires a manual export, consumes storage space, and might require more overhead for schema management in BigQuery. Log Analytics with linked datasets offers a more direct and integrated approach and is preferred over this method.

In summary, option B utilizes the best features of Cloud Logging and BigQuery to provide efficient SQL querying of logs, aligns with Google's recommended practices, and minimizes operational complexity.

**Question: 281** CertyIQ

You need to deploy a third-party software application onto a single Compute Engine VM instance. The application requires the highest speed read and write disk access for the internal database. You need to ensure the instance will recover on failure. What should you do?

A.Create an instance template. Set the disk type to be an SSD Persistent Disk. Launch the instance template as part of a stateful managed instance group.

B.Create an instance template. Set the disk type to be an SSD Persistent Disk. Launch the instance template as part of a stateless managed instance group.

C.Create an instance template. Set the disk type to be Hyperdisk Extreme. Launch the instance template as part of a stateful managed instance group.

D.Create an instance template. Set the disk type to be Hyperdisk Extreme. Launch the instance template as part of a stateless managed instance group.

**Answer: C**

**Explanation:**

Here's a detailed justification for why option C is the correct answer:

The core requirement is high-speed disk access for an internal database combined with instance recovery on failure. Let's break down each component:

**High-Speed Disk Access:** Standard Persistent Disks, even SSD versions, have limitations on IOPS (Input/Output Operations Per Second). For applications requiring the highest possible speed and low latency, **Hyperdisk Extreme** is the appropriate choice. It's specifically designed for high-performance, demanding workloads. https://cloud.google.com/compute/docs/disks/hyperdisk

**Instance Recovery:** To ensure automatic recovery in case of failure, the Compute Engine instance needs to be part of a managed instance group (MIG). MIGs automatically recreate instances if they become unhealthy or fail. Stateful MIGs are crucial here, as they preserve instance identity and attached storage, which is important for database consistency. A stateless MIG would recreate the instance with a fresh disk, potentially leading to data loss. https://cloud.google.com/compute/docs/instance-groups

**Instance Templates:** Both stateful and stateless MIGs require instance templates which define the configuration of instances within the group. The critical configuration here is the usage of Hyperdisk Extreme. https://cloud.google.com/compute/docs/instance-templates

Therefore:

Options A and B incorrectly use SSD Persistent Disk which does not deliver the highest IOPS.
Option B uses a stateless MIG, which would not preserve state required by the database.
Option D uses a stateless MIG, which would not preserve state required by the database.

Only option C accurately combines the correct disk type (Hyperdisk Extreme) with a suitable deployment strategy for resilience (Stateful MIG), aligning with both requirements of the question: "highest speed read and write disk access" and "ensure the instance will recover on failure". This makes it the optimal solution.

**Question: 282** CertyIQ

You have a VM instance running in a VPC with single-stack subnets. You need to ensure that the VM instance has a fixed IP address so that other services hosted in the same VPC can communicate with the VM. You want to follow Google-recommended practices while minimizing cost. What should you do?

A.Promote the existing IP address of the VM to become a static external IP address.

B.Promote the existing IP address of the VM to become a static internal IP address.

C.Reserve a new static external IPv6 address and assign the new IP address to the VM.

D.Reserve a new static external IP address and assign the new IP address to the VM.

**Answer: B**

**Explanation:**

The correct answer is **B. Promote the existing IP address of the VM to become a static internal IP address.** This option aligns with Google's best practices for internal communication within a Virtual Private Cloud (VPC) while also minimizing costs. Here's a detailed justification:

**Internal IP Addresses for Internal Communication:** VMs within the same VPC should ideally communicate using private (internal) IP addresses. This approach avoids unnecessary traffic leaving the VPC network, reducing latency and costs associated with traversing the internet and external network paths.
**Static Internal IP Addresses for Predictability:** Using a static internal IP address ensures that the VM maintains the same IP address, allowing other services within the VPC to reliably communicate with it. Without a static IP, the VM's IP could change upon reboot or replacement.
**Promoting an Existing IP:** The question mentions that the VM already has an IP address assigned. Promoting this to static means you are not reserving and paying for another IP address, thus optimizing cost.
**Single-Stack Subnets:** The single-stack subnet implies only IPv4 addresses are in use, thus, options using IPv6 addresses are incorrect.
**Cost Minimization:** Static external IP addresses incur costs even when not in use and should only be reserved when publicly reachable via the internet. This scenario is for internal communication where there is no need for external connectivity. By promoting the existing IP to an internal static one, you are also saving on cost.

Options A, C, and D involve external IP addresses, which are unnecessary for internal communication within a VPC. External IPs also incur additional costs, making them inappropriate when internal communication is required. Option C also uses IPv6 addresses which are not relevant considering the question's context of single-stack subnet which means IPv4.

**Relevant Google Cloud Documentation:**

**Static Internal IP Addresses:** https://cloud.google.com/vpc/docs/configure-private-static-ip-address
**VPC Networks Overview:** https://cloud.google.com/vpc/docs/vpc
**IP Addressing in Google Cloud:** https://cloud.google.com/vpc/docs/ip-addresses

**Question: 283** CertyIQ

Your preview application, deployed on a single-zone Google Kubernetes Engine (GKE) cluster in us-central1, has gained popularity. You are now ready to make the application generally available. You need to deploy the application to production while ensuring high availability and resilience. You also want to follow Google-recommended practices. What should you do?

A.Use the gcloud container clusters create command with the options --enable-multi-networking and --enable-autoscaling to create an autoscaling zonal cluster and deploy the application to it.

B.Use the gcloud container clusters create-auto command to create an autopilot cluster and deploy the application to it.

C.Use the gcloud container clusters update command with the option --region us-central1 to update the cluster and deploy the application to it.

D.Use the gcloud container clusters update command with the option --node-locations us-central1-a,us-central1-b to update the cluster and deploy the application to the nodes.

**Answer: B**

**Explanation:**

The correct answer is **B. Use the gcloud container clusters create-auto command to create an autopilot cluster and deploy the application to it.** Here's the justification:

The primary goal is to achieve high availability and resilience for a production application while adhering to Google-recommended practices. Option B, creating an Autopilot cluster, directly addresses these requirements. Autopilot is a fully managed GKE mode that handles node provisioning, scaling, and upgrades, abstracting away much of the underlying infrastructure management. This aligns with Google's recommended best practices of leveraging managed services to reduce operational overhead. Autopilot clusters inherently provide multi-zonal deployments for high availability, spreading workloads across multiple availability zones within a region. This ensures that if one zone experiences an issue, the application remains available in other zones. Furthermore, Autopilot automatically manages resource scaling based on the application's needs, leading to improved resilience and resource efficiency.

Option A is incorrect because while it mentions autoscaling, it creates a zonal cluster with the responsibility to manage node locations explicitly, and it lacks the holistic management provided by Autopilot, creating higher operational overhead for the end user. Option C is incorrect because updating a single-zone cluster to a regional one using --region will not enable high availability; it will likely only change the control plane configuration. Option D is also incorrect because updating a cluster with --node-locations doesn't inherently bring high availability capabilities but will cause node migration that might cause disruption without correct configurations.

In essence, Autopilot mode simplifies Kubernetes management, automatically implements best practices for high availability and resilience, and aligns with the prompt's requirements.

**Authoritative Links:**

**Google Kubernetes Engine Autopilot:** https://cloud.google.com/kubernetes-engine/docs/autopilot
**Best practices for running Kubernetes on Google Cloud:** https://cloud.google.com/kubernetes-engine/docs/best-practices
**GKE Cluster Types:** https://cloud.google.com/kubernetes-engine/docs/concepts/cluster-types

**Question: 284** CertyIQ

You are developing an application that will be deployed on Google Cloud. The application will use a service account to retrieve data from BigQuery. Before you deploy your application, you want to test the permissions of this service account from your local machine to ensure there will be no authentication issues. You want to ensure that you use the most secure method while following Google-recommended practices. What should you do?

A.Generate a service account key, and configure the gcloud CLI to use this key. Issue a relevant BigQuery request through the gdoud CLI to test the access.

B.Grant the service account the BigQuery Administrator IAM role to ensure the service account has all required access.

C.Configure the gcloud CLI to use service account impersonation. Issue a relevant BigQuery request through the gcloud CLI to test the access.

D.Configure the gcloud CLI with Application Default Credentials using your user account. Issue a relevant BigQuery request through the gcloud CLI to test the access.

**Answer: C**

**Explanation:**

Google-recommended practices advise against distributing or storing long-lived service account keys

because they can be a security risk. Instead, service account impersonation allows you to use your own credentials to "borrow" the identity of the service account without needing to download a key file.

**Question: 285** CertyIQ

You are the Organization Administrator for your company's Google Cloud resources. Your company has strict compliance rules that require you to be notified about any modifications to files and documents hosted on Cloud Storage. In a recent incident, one of your team members was able to modify files and you did not receive any notifications, causing other production jobs to fail. You must ensure that you receive notifications for all changes to files and documents in Cloud Storage while minimizing management overhead. What should you do?

A.View Cloud Audit logs for all Cloud Storage files in Logs Explorer. Filter by Admin Activity logs.

B.Enable Cloud Storage object versioning on your bucket. Configure Pub/Sub notifications for your Cloud Storage buckets.

C.Enable versioning on the Cloud Storage bucket. Set up a custom script that scans versions of Cloud Storage objects being modified and alert the admin by using the script.

D.Configure Object change notifications on the Cloud Storage buckets. Send the events to Pub/Sub.

**Answer: B**

**Question: 286** CertyIQ

You need to manage multiple Google Cloud Platform (GCP) projects in the fewest steps possible. You want to configure the Google Cloud SDK command line interface (CLI) so that you can easily manage multiple GCP projects. What should you?

A. 1. Create a configuration for each project you need to manage. 2. Activate the appropriate configuration when you work with each of your assigned GCP projects.

B. 1. Create a configuration for each project you need to manage. 2. Use gcloud init to update the configuration values when you need to work with a non-default project

C. 1. Use the default configuration for one project you need to manage. 2. Activate the appropriate configuration when you work with each of your assigned GCP projects.

D. 1. Use the default configuration for one project you need to manage. 2. Use gcloud init to update the configuration values when you need to work with a non-default project.

**Answer: A**

**Explanation:**

The correct answer is **A**, which advocates for using named configurations within the Google Cloud SDK (gcloud CLI) to manage multiple GCP projects efficiently. Here's why:

Option A aligns perfectly with the intended use of gcloud configurations. By creating a separate configuration for each project, you establish distinct sets of settings (including the project ID) that are isolated from each other. This avoids the need to re-configure the gcloud settings every time you switch between projects. To operate on a specific project, you simply activate its corresponding configuration. This allows for rapid project context switching without needing to invoke gcloud init repeatedly. gcloud config configurations commands facilitate the management of these configurations.

Option B is incorrect because although creating separate configurations is correct, gcloud init is used primarily for initial setup or updating global SDK settings, not for switching between project configurations. While it can switch the active project, it’s more cumbersome than simply activating a named configuration, which is a quicker, less interruptive way of switching. Option C is incorrect as it suggests that only one project would use

the default settings. This would be very difficult to manage, as changing your project would require reconfiguring the SDK. Option D suffers from the same problem as Option B, by suggesting a cumbersome method of using gcloud init for switching.

Using named configurations is the most efficient and recommended method for managing multiple projects with the gcloud CLI. It promotes clarity, organization, and avoids the need for repetitive project switching procedures.

**Authoritative Links for further research:**

**Managing gcloud CLI configurations:** https://cloud.google.com/sdk/docs/configurations
**gcloud config configurations:** https://cloud.google.com/sdk/gcloud/reference/config/configurations

**Question: 287** CertyIQ

You have an application that uses Cloud Spanner as a backend database. The application has a very predictable traffic pattern. You want to automatically scale up or down the number of Spanner nodes depending on traffic. What should you do?

A. Create a cron job that runs on a scheduled basis to review stackdriver monitoring metrics, and then resize the Spanner instance accordingly.

B. Create a Stackdriver alerting policy to send an alert to oncall SRE emails when Cloud Spanner CPU exceeds the threshold. SREs would scale resources up or down accordingly.

C. Create a Stackdriver alerting policy to send an alert to Google Cloud Support email when Cloud Spanner CPU exceeds your threshold. Google support would scale resources up or down accordingly.

D. Create a Stackdriver alerting policy to send an alert to webhook when Cloud Spanner CPU is over or under your threshold. Create a Cloud Function that listens to HTTP and resizes Spanner resources accordingly.

**Answer: D**

**Explanation:**

Here's a detailed justification for why option D is the correct approach for automatically scaling Cloud Spanner based on predictable traffic patterns, along with explanations of why the other options are unsuitable:

Option D utilizes the power of serverless and event-driven architecture for automatic scaling. Stackdriver (now Cloud Monitoring) can trigger an alert when Spanner CPU usage breaches predefined thresholds. This alert is sent to a webhook, which acts as a trigger for a Cloud Function. This function, written in code, is then responsible for programmatically resizing the Spanner instance (adding or removing nodes). This automated approach aligns perfectly with the need for dynamic, traffic-based adjustments.

Option A, relying on a cron job, is less flexible and less reactive. A fixed schedule might not align with actual traffic changes, leading to either over-provisioning (wasted resources) or under-provisioning (performance issues). Option B involves human intervention, which is undesirable for automated scaling. Relying on SREs to manually adjust resources introduces delays and reduces the ability to quickly respond to changes. Option C incorrectly suggests involving Google Support for routine scaling. Google Support is for resolving technical issues, not for routine operational tasks such as resource scaling.

Cloud Functions are excellent for this task because they execute only when needed, reducing costs, and they provide the logic to interact with the Spanner API for scaling operations. Webhooks act as real-time connectors to trigger actions based on the metrics alerts. This system ensures efficient and responsive scaling that is well-suited to the use case.

Relevant Links for further research:

**Cloud Monitoring Alerting:** https://cloud.google.com/monitoring/alerts
**Cloud Functions:** https://cloud.google.com/functions
**Cloud Spanner API:** https://cloud.google.com/spanner/docs/apis
**Automated Scaling for Spanner**: https://cloud.google.com/spanner/docs/instance-config#autoscaler

**Question: 288** CertyIQ

Your company publishes large files on an Apache web server that runs on a Compute Engine instance. The Apache web server is not the only application running in the project. You want to receive an email when the egress network costs for the server exceed 100 dollars for the current month as measured by Google Cloud Platform (GCP). What should you do?

A. Set up a budget alert on the project with an amount of 100 dollars, a threshold of 100%, and notification type of "email."

B. Set up a budget alert on the billing account with an amount of 100 dollars, a threshold of 100%, and notification type of "email."

C. Export the billing data to BigQuery. Create a Cloud Function that uses BigQuery to sum the egress network costs of the exported billing data for the Apache web server for the current month and sends an email if it is over 100 dollars. Schedule the Cloud Function using Cloud Scheduler to run hourly.

D. Use the Stackdriver Logging Agent to export the Apache web server logs to Stackdriver Logging. Create a Cloud Function that uses BigQuery to parse the HTTP response log data in Stackdriver for the current month and sends an email if the size of all HTTP responses, multiplied by current GCP egress prices, totals over 100 dollars. Schedule the Cloud Function using Cloud Scheduler to run hourly.

**Answer: C**

**Explanation:**

The correct answer is C because it provides the most precise and granular method for tracking egress costs associated specifically with the Apache web server. Standard budget alerts (options A and B) operate at the project or billing account level, making it impossible to isolate the egress cost of a specific resource like the Apache web server. Option A would trigger an alert based on total project costs, which are not solely associated with the Apache web server's egress. Option B, at the billing account level, would be even more broad.

Option C leverages BigQuery, which is a powerful data warehousing service suitable for analyzing detailed billing data. By exporting GCP billing data to BigQuery, we gain access to granular information that includes resource-specific costs. A Cloud Function, triggered by Cloud Scheduler, can then perform calculations to specifically sum the egress network costs for the Apache server. The Cloud Function's query filters the BigQuery data to only consider egress costs and then checks whether the total exceeds the $100 threshold for the current month. If the threshold is surpassed, the Cloud Function sends out an email notification, ensuring timely alerts about the costs related to the Apache server.

Option D's approach relies on parsing Apache server logs, which does not provide accurate cost data as GCP costs are not directly logged by the server; rather, they're calculated internally by Google. Also, calculating network usage from HTTP response sizes can be complex, imprecise, and does not consider internal egress traffic. Therefore, option C is the most accurate, targeted, and efficient method for this scenario.

Here are some helpful links:

**BigQuery Billing Export:** https://cloud.google.com/billing/docs/how-to/export-data-bigquery
**Cloud Functions:** https://cloud.google.com/functions
**Cloud Scheduler:** https://cloud.google.com/scheduler
**Cost Management with Budget Alerts:** https://cloud.google.com/billing/docs/how-to/budgets

**Question: 289** CertyIQ

You have designed a solution on Google Cloud Platform (GCP) that uses multiple GCP products. Your company has asked you to estimate the costs of the solution. You need to provide estimates for the monthly total cost. What should you do?

A. For each GCP product in the solution, review the pricing details on the products pricing page. Use the pricing calculator to total the monthly costs for each GCP product.

B. For each GCP product in the solution, review the pricing details on the products pricing page. Create a Google Sheet that summarizes the expected monthly costs for each product.

C. Provision the solution on GCP. Leave the solution provisioned for 1 week. Navigate to the Billing Report page in the Google Cloud Platform Console. Multiply the 1 week cost to determine the monthly costs.

D. Provision the solution on GCP. Leave the solution provisioned for 1 week. Use Stackdriver to determine the provisioned and used resource amounts. Multiply the 1 week cost to determine the monthly costs.

**Answer: A**

**Explanation:**

Option A is the most accurate and practical approach for estimating monthly costs. Google Cloud Platform (GCP) offers detailed pricing information for each of its products, accessible through their respective pricing pages. Using the GCP Pricing Calculator is the recommended method for obtaining estimated costs. The calculator allows you to specify the usage parameters for each service, such as instance types, storage volume, and network traffic, to generate a customized cost estimate. This approach avoids incurring unnecessary costs and provides a reliable estimation based on your specific needs before actual deployment. Options B, C, and D are not ideal. While creating a spreadsheet (B) can be useful for summarizing, the core of the estimation lies in accurately calculating the costs, for which the pricing calculator is the better tool. Provisioning the solution (C & D) to estimate costs is inefficient and can result in unexpected expenses, especially as a week can not be representative of long term use. Additionally, monitoring and multiplying by 4, like C and D propose, does not account for various discounts that may apply and variability of usage patterns. Option A ensures you understand pricing components and tailor the estimate to your specific needs. Therefore, Option A provides the best method for preliminary cost estimation prior to deploying resources.

GCP Pricing CalculatorGCP Pricing Overview

**Question: 290** CertyIQ

Your existing application running in Google Kubernetes Engine (GKE) consists of multiple pods running on four GKE n1""standard""2 nodes. You need to deploy additional pods requiring n2""highmem""16 nodes without any downtime. What should you do?

A. Use gcloud container clusters upgrade. Deploy the new services.

B. Create a new Node Pool and specify machine type n2""highmem""16. Deploy the new pods.

C. Create a new cluster with n2""highmem""16 nodes. Redeploy the pods and delete the old cluster.

D. Create a new cluster with both n1""standard""2 and n2""highmem""16 nodes. Redeploy the pods and delete the old cluster.

**Answer: B**

**Explanation:**

The correct answer is **B: Create a new Node Pool and specify machine type n2""highmem""16. Deploy the new pods.** This approach allows for a seamless, zero-downtime deployment of pods requiring different node resources within the existing GKE cluster. Node Pools in GKE enable the cluster to accommodate diverse

workload requirements by utilizing separate groups of nodes with distinct configurations. By creating a new node pool with the specified n2-highmem-16 machine type, you introduce nodes suitable for the new pods while maintaining the original nodes and pods on the n1-standard-2 nodes. Once the new node pool is ready, the pods can be deployed with node affinity or node selectors to direct them to the appropriate nodes. The existing pods will continue to function undisturbed on their original nodes. This incremental scaling strategy prevents downtime and ensures a controlled roll-out of new resources, aligning with best practices for Kubernetes deployments. Using a gcloud container clusters upgrade command (option A) is typically used for Kubernetes upgrades, not machine type changes. Options C and D involve creating a new cluster, which results in significant downtime and unnecessary complexity for this specific scenario.

**Authoritative Links for Further Research:**

1. **GKE Node Pools Documentation:** https://cloud.google.com/kubernetes-engine/docs/how-to/node-pools
2. **GKE Node Selectors and Affinity:** https://kubernetes.io/docs/concepts/scheduling-eviction/assign-pod-node/
3. **Kubernetes Rolling Updates:** https://kubernetes.io/docs/tutorials/kubernetes-basics/update/update-intro/

**Question: 291** CertyIQ

You are hosting an application from Compute Engine virtual machines (VMs) in us""central1""a. You want to adjust your design to support the failure of a single
Compute Engine zone, eliminate downtime, and minimize cost. What should you do?

A. "" Create Compute Engine resources in us""central1""b. "" Balance the load across both us""central1""a and us""central1""b.

B. "" Create a Managed Instance Group and specify us""central1""a as the zone. "" Configure the Health Check with a short Health Interval.

C. "" Create an HTTP(S) Load Balancer. "" Create one or more global forwarding rules to direct traffic to your VMs.

D. "" Perform regular backups of your application. "" Create a Cloud Monitoring Alert and be notified if your application becomes unavailable. "" Restore from backups when notified.

**Answer: A**

**Explanation:**

The correct answer is **A**. Here's a detailed justification:

**A. " Create Compute Engine resources in us""central1""b. "" Balance the load across both us""central1""a and us""central1""b."**

This approach directly addresses the requirements of zone failure tolerance, zero downtime, and cost minimization. By deploying Compute Engine instances in a second zone (us-central1-b) within the same region (us-central1), we achieve redundancy. If us-central1-a fails, traffic can automatically be routed to the instances in us-central1-b, ensuring application availability. Balancing the load between the two zones using a regional load balancer (implied by the action) distributes traffic evenly, improving performance and preventing overload on instances in a single zone. This minimizes costs compared to having instances running in multiple regions while still providing HA, as it leverages resources within the same geographical area. This follows best practices for high availability within a region and is the most efficient way to handle zonal failures.

**Why other options are incorrect:**

**B:** A Managed Instance Group (MIG) in a single zone (us-central1-a) does not provide zone failure tolerance. The short health check interval will detect problems but will not prevent downtime during an outage of the whole zone.
**C:** An HTTP(S) load balancer directs traffic but on its own it does not guarantee HA. It still needs VMs in different zones to fail over, but the option in the question statement doesnt create those.
**D:** Regular backups and alerts are essential for disaster recovery but do not address the requirement of minimizing downtime. Restoring from backups after a failure leads to service disruption, failing the zero downtime objective.

**Relevant Concepts:**

**High Availability (HA):** Ensuring systems remain operational with minimal downtime.
**Zone Redundancy:** Distributing resources across different availability zones within a region to withstand a zone failure.
**Load Balancing:** Distributing incoming traffic across multiple servers to prevent any single server from being overwhelmed.
**Regional Resources:** Resources like Regional Load balancers that operate across zones within a region.
**Cost Optimization:** Implementing solutions that minimize operational costs.

**Authoritative Links:**

Google Cloud Documentation on Regions and Zones: Provides details on how Google Cloud organizes infrastructure.
Google Cloud Documentation on Load Balancing: Explains the various load balancing options on GCP.
Google Cloud Documentation on Managed Instance Groups: Describes how to use MIGs for scaling and high availability.

# Thank you

Thank you for being so interested in the premium exam material.
I'm glad to hear that you found it informative and helpful.

If you have any feedback or thoughts on the bumps, I would love to hear them.
Your insights can help me improve our writing and better understand our readers.

## Best of Luck

You have worked hard to get to this point, and you are well-prepared for the exam
Keep your head up, stay positive, and go show that exam what you're made of!

Feedback

More Papers