

**VPC Networking Fundamentals**

experimentLabschedule1 houruniversal\_currency\_altNo costshow\_chartIntroductory

infoThis lab may incorporate AI tools to support your learning.

**GSP210**



**Overview**

Google Cloud Virtual Private Cloud (VPC) provides networking functionality to Compute Engine virtual machine (VM) instances, Kubernetes Engine containers and App Engine Flex. In other words, without a VPC network you cannot create VM instances, containers or App Engine applications. Therefore, each Google Cloud project has a **default** network to get you started.

You can think of a VPC network the same way you would think of a physical network, except that it is virtualized within Google Cloud. A VPC network is a global resource which consists of a list of regional virtual subnetworks (subnets) in data centers, all connected by a global wide area network (WAN). VPC networks are logically isolated from each other in Google Cloud.

In this lab, you create an auto mode VPC network with firewall rules and two VM instances. Then, you explore the connectivity for the VM instances.

Objectives

In this lab, you learn how to perform the following tasks:

* Explore the default VPC network
* Create an auto mode network with firewall rules
* Create VM instances using Compute Engine
* Explore the connectivity for VM instances

**Setup and requirements**

Before you click the Start Lab button

Read these instructions. Labs are timed and you cannot pause them. The timer, which starts when you click **Start Lab**, shows how long Google Cloud resources will be made available to you.

This hands-on lab lets you do the lab activities yourself in a real cloud environment, not in a simulation or demo environment. It does so by giving you new, temporary credentials that you use to sign in and access Google Cloud for the duration of the lab.

To complete this lab, you need:

* Access to a standard internet browser (Chrome browser recommended).

**Note:** Use an Incognito or private browser window to run this lab. This prevents any conflicts between your personal account and the Student account, which may cause extra charges incurred to your personal account.

* Time to complete the lab---remember, once you start, you cannot pause a lab.

**Note:** If you already have your own personal Google Cloud account or project, do not use it for this lab to avoid extra charges to your account.

How to start your lab and sign in to the Google Cloud console

1. Click the **Start Lab** button. If you need to pay for the lab, a pop-up opens for you to select your payment method. On the left is the **Lab Details** panel with the following:
   * The **Open Google Cloud console** button
   * Time remaining
   * The temporary credentials that you must use for this lab
   * Other information, if needed, to step through this lab
2. Click **Open Google Cloud console** (or right-click and select **Open Link in Incognito Window** if you are running the Chrome browser).

The lab spins up resources, and then opens another tab that shows the **Sign in** page.

***Tip:*** Arrange the tabs in separate windows, side-by-side.

**Note:**If you see the **Choose an account** dialog, click **Use Another Account**.

1. If necessary, copy the **Username** below and paste it into the **Sign in** dialog.

student-04-195dc9ae9109@qwiklabs.net

Copied!

content\_copy

You can also find the **Username** in the **Lab Details** panel.

1. Click **Next**.
2. Copy the **Password** below and paste it into the **Welcome** dialog.

OZ1mUjCngcQM

Copied!

content\_copy

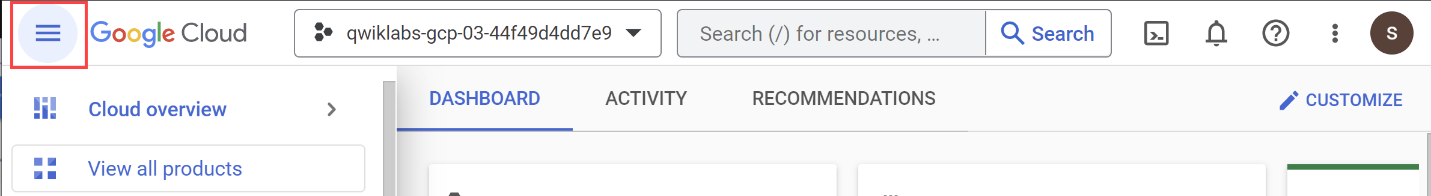
You can also find the **Password** in the **Lab Details** panel.

1. Click **Next**.

**Important:**You must use the credentials the lab provides you. Do not use your Google Cloud account credentials.**Note:**Using your own Google Cloud account for this lab may incur extra charges.

1. Click through the subsequent pages:
   * Accept the terms and conditions.
   * Do not add recovery options or two-factor authentication (because this is a temporary account).
   * Do not sign up for free trials.

After a few moments, the Google Cloud console opens in this tab.

**Note:** To view a menu with a list of Google Cloud products and services, click the **Navigation menu** at the top-left. 

**Task 1. Explore the default network**

Each Google Cloud project has a **default** network with subnets, routes, and firewall rules.

View the subnets

The **default** network has a subnet in [each Google Cloud region](https://cloud.google.com/compute/docs/regions-zones/#available).

1. In the Cloud console, navigate to **Navigation menu** (Navigation menu icon) > **VPC network** > **VPC networks**.
2. Click on the **default** network. Notice the **default** network details and the subnets.

**Note:**Each subnet is associated with a Google Cloud region and a private RFC 1918 CIDR block for its internal **IP addresses range** and a **gateway**.

View the routes

Routes tell VM instances and the VPC network how to send traffic from an instance to a destination, either inside the network or outside of Google Cloud.

Each VPC network comes with some default routes to route traffic among its subnets and send traffic from eligible instances to the Internet.

1. In the left pane, click on **Routes**.
2. In the **Effective Routes** tab, select the default network and the us-central1 region.

Notice that there is a route for each subnet and one for the **Default internet gateway** (0.0.0.0./0).

**Note:**These routes are managed for you but you can create custom static routes to direct some packets to specific destinations. For example, you can create a route that sends all outbound traffic to an instance configured as a NAT gateway.

View the firewall rules

Each VPC network implements a distributed virtual firewall that you can configure. Firewall rules allow you to control which packets are allowed to travel to which destinations.

Every VPC network has two implied firewall rules that block all incoming connections and allow all outgoing connections.

1. In the left pane, click on **Firewall**.

Notice that there are 4 **Ingress** firewall rules for the **default** network:

* default-allow-icmp
* default-allow-internal
* default-allow-rdp
* default-allow-ssh

**Note:**These firewall rules allow **ICMP**, **RDP** and **SSH** ingress traffic from anywhere (0.0.0.0/0) and all **TCP**, **UDP** and **ICMP** traffic within the network (10.128.0.0/9). The **Targets**, **Source filters**, **Protocols/ports** and **Action** columns explain these rules.

Delete the default network

1. Select all firewall rules and click **DELETE**.
2. In the left pane, click on **VPC networks**.
3. Click on the **default** network.
4. Click **Delete VPC network** at the top of the page,
5. Then click **DELETE** to confirm the deletion of the **default** network.

**Note:**Wait for the network to be deleted before moving on.

1. In the left pane, click on **Routes**.

Notice that there are no routes. You may need to click the **Refresh** button at the top of the page.

**Note:**Without a VPC network, there are no routes!

Try to create a VM instance

Verify that you cannot create a VM instance without a VPC network.

1. In the Cloud console, navigate to **Navigation menu** (Navigation menu icon) > **Compute Engine** > **VM instances**.
2. Click **+CREATE INSTANCE** to create a VM instance.
3. Leave all the values at their default and click **Create**.

**Note:**Notice the error.

1. Expand the **Advanced options** section, and then scroll down to **Network interfaces**.

**Note:**Notice the error *No more networks available in this project* under the **Network** box.

1. Click **Cancel**.

**Note:**As expected, you cannot create a VM instance without a VPC network!

**Task 2. Create a VPC network and VM instances**

Create a VPC network so that you can create VM instances.

Create an auto mode VPC network with Firewall rules

Replicate the **default** network by creating an auto mode network.

1. In the console, navigate to **Navigation menu** (Navigation menu icon) > **VPC network** > **VPC networks**, and then click **+CREATE VPC NETWORK**.
2. Set the **Name** to mynetwork.
3. For **Subnet creation mode**, click **Automatic**.

Auto mode networks create subnets in each region automatically.

1. For **Firewall rules**, check all available rules.

These are the same standard firewall rules that the default network had.

**Note:**The **deny-all-ingress** and **allow-all-egress** rules are also displayed, but you cannot check or uncheck them as they are implied. These two rules have a lower **Priority** (higher integers indicate lower priorities) so that the allow ICMP, custom, RDP and SSH rules are considered first.

1. Click **CREATE**, then wait for **mynetwork** to be created.

Notice that a subnet was created for each region.

1. Click on the **mynetwork** name and record the IP address range for the subnets in us-west1 and us-east4. Refer to these in the next steps.

**Note:**If you ever delete the default network, you can quickly re-create it by creating an auto mode network as you just did.

Test completed task

Click **Check my progress** to verify your performed task. If you have completed the task successfully, you are granted an assessment score.

Create a VPC network.

Check my progress

Create a VM instance in us-west1

Create a VM instance in the us-west1 region. Selecting a region and zone determines the subnet and assigns the internal IP address from the subnet's IP address range.

1. In the console, navigate to **Navigation menu** (Navigation menu icon) > **Compute Engine** > **VM instances**,
2. Click **+CREATE INSTANCE**.
3. Set the following values, leaving all others at their defaults:

|  |  |
| --- | --- |
| **Property** | **Value (type value or select option as specified)** |
| Name | mynet-us-vm |
| Region | us-west1 |
| Zone | us-west1-a |
| Series | E2 |
| Machine type | e2-micro |

1. Click **Create**, then wait for the instance to be created.
2. Verify that the **Internal IP** was assigned from the IP address range for the subnet in us-west1 10.138.0.0/20.

Test completed task

Click **Check my progress** to verify your performed task. If you have completed the task successfully, you are granted an assessment score.

Create a VM instance in us-west1.

Check my progress

Create a VM instance in us-east4

Create a VM instance in the us-east4 region.

1. Click **+CREATE INSTANCE**.
2. Set the following values, leaving all others at their defaults:

|  |  |
| --- | --- |
| **Property** | **Value (type value or select option as specified)** |
| Name | mynet-second-vm |
| Region | us-east4 |
| Zone | us-east4-c |
| Series | E2 |
| Machine type | e2-micro |

1. Click **Create**, then wait for the instance to be created.

**Note:**If you receive an error stating that this zone does not have enough resources to fulfill the request, try re-running steps 1-3 with a different zone.

1. Verify that the **Internal IP** was assigned from the IP address range for the subnet in us-east4 10.150.0.0/20.

The **Internal IP** should be 10.150.0.2 as x.x.x.1 is reserved for the gateway and you have not configured any other instances in that subnet.

**Note:**The **External IP addresses** for both VM instances are ephemeral. If an instance is stopped, any ephemeral external IP addresses assigned to the instance are released back into the general Compute Engine pool and become available for use by other projects.

When a stopped instance is started again, a new ephemeral external IP address is assigned to the instance. Alternatively, you can reserve a static external IP address, which assigns the address to your project indefinitely until you explicitly release it.

Test completed task

Click **Check my progress** to verify your performed task. If you have completed the task successfully, you are granted an assessment score.

Create a VM instance in us-east4.

Check my progress

**Task 3. Explore the connectivity for VM instances**

Explore the connectivity for the VM instances. Specifically, SSH to your VM instances using tcp:22 and ping both the internal and external IP addresses of your VM instances using ICMP. Then, explore the effects of the firewall rules on connectivity by removing the firewall rules one-by-one.

**Verify connectivity for the VM instances**

The firewall rules that you created with **mynetwork** allow ingress SSH and ICMP traffic from within **mynetwork** (internal IP) and outside of that network (external IP).

1. In the console, navigate to **Navigation menu** (Navigation menu icon) > **Compute Engine** > **VM instances**.

Note the external and internal IP addresses for **mynet-second-vm**.

1. For **mynet-us-vm**, click **SSH** to launch a terminal and connect. You may have to click **SSH** twice.

You are able to SSH because of the **allow-ssh** firewall rule, which allows incoming traffic from anywhere (0.0.0.0/0) for **tcp:22**.

**Note:**The SSH connection works seamlessly because Compute Engine generates an SSH key for you and stores it in one of the following locations:

* + By default, Compute Engine adds the generated key to project or instance metadata.
  + If your account is configured to use OS Login, Compute Engine stores the generated key with your user account.

Alternatively, you can control access to Linux instances by creating SSH keys and editing public SSH key metadata.

1. To test connectivity to **mynet-second-vm**'s internal IP, run the following command using **mynet-second-vm**'s internal IP:

ping -c 3 <Enter mynet-second-vm's internal IP here>

Copied!

content\_copy

You are able to ping **mynet-second-vm**'s internal IP because of the **allow-custom** firewall rule.

1. To test connectivity to **mynet-second-vm**'s external IP, run the following command using **mynet-second-vm**'s external IP:

ping -c 3 <Enter mynet-second-vm's external IP here>

Copied!

content\_copy

**Task 4. Test your understanding**

Below are a multiple choice questions to reinforce your understanding of this lab's concepts. Answer them to the best of your abilities.

Which firewall rule allows the ping to mynet-second-vm's external IP address?

mynetwork-allow-rdp

mynetwork-allow-icmp

mynetwork-allow-custom

mynetwork-allow-ssh

Submit

Google Cloud firewall rules let you allow or deny traffic to and from your virtual machine (VM) instances based on a configuration.

True

False

Firewall rules can be shared among networks.

True

False

**Note:**You were able to SSH to **mynet-us-vm** and ping **mynet-second-vm**'s internal and external IP address as expected. Alternatively, you could SSH to **mynet-second-vm** and ping **mynet-us-vm**'s internal and external IP address, which also works.

**Task 5. Remove the allow-icmp firewall rules**

Remove the **allow-icmp** firewall rule and try to ping the internal and external IP address of **mynet-second-vm**.

1. In the console, navigate to **Navigation menu** (Navigation menu icon) > **VPC network** > **Firewall**.
2. Check the **mynetwork-allow-icmp** rule.
3. Click **DELETE**.
4. Click **DELETE** to confirm the deletion.

Wait for the firewall rule to be deleted.

1. Return to the **mynet-us-vm** SSH terminal.
2. To test connectivity to **mynet-second-vm**'s internal IP, run the following command using **mynet-second-vm**'s internal IP:

ping -c 3 <Enter mynet-second-vm's internal IP here>

Copied!

content\_copy

You are able to ping **mynet-second-vm**'s internal IP because of the **allow-custom** firewall rule.

1. To test connectivity to **mynet-second-vm**'s external IP, run the following command using **mynet-second-vm**'s external IP:

ping -c 3 <Enter mynet-second-vm's external IP here>

Copied!

content\_copy

**Note:**The **100% packet loss** indicates that you are unable to ping **mynet-second-vm**'s external IP. This is expected because you deleted the **allow-icmp** firewall rule!

**Task 6. Remove the allow-custom firewall rules**

Remove the **allow-custom** firewall rule and try to ping the internal IP address of **mynet-second-vm**.

1. In the console, navigate to **Navigation menu** (Navigation menu icon) > **VPC network** > **Firewall**.
2. Check the **mynetwork-allow-custom** rule and then click **DELETE**.
3. Click **DELETE** to confirm the deletion.

Wait for the firewall rule to be deleted.

1. Return to the **mynet-us-vm** SSH terminal.
2. To test connectivity to **mynet-second-vm**'s internal IP, run the following command using **mynet-second-vm**'s internal IP:

ping -c 3 <Enter mynet-second-vm's internal IP here>

Copied!

content\_copy

**Note:**The **100% packet loss** indicates that you are unable to ping **mynet-second-vm**'s internal IP. This is expected because you deleted the **allow-custom** firewall rule!

1. Close the SSH terminal:

exit

Copied!

content\_copy

**Task 7. Remove the allow-ssh firewall rules**

Remove the **allow-ssh** firewall rule and try to SSH to **mynet-us-vm**.

1. In the console, navigate to **Navigation menu** (Navigation menu icon) > **VPC network** > **Firewall**.
2. Check the **mynetwork-allow-ssh** rule and then click **DELETE**.
3. Click **DELETE** to confirm the deletion.

Wait for the firewall rule to be deleted.

1. In the console, navigate to **Navigation menu** (Navigation menu icon) > **Compute Engine** > **VM instances**.
2. For **mynet-us-vm**, click **SSH** to launch a terminal and connect.

**Note:**The **Connection failed** message indicates that you are unable to SSH to **mynet-us-vm** because you deleted the **allow-ssh** firewall rule!

**Congratulations!**

In this lab, you explored the default network along with its subnets, routes, and firewall rules. You deleted the default network and determined that you cannot create any VM instances without a VPC network. Thus, you created a new auto mode VPC network with subnets, routes, firewall rules and two VM instances. Then, you tested the connectivity for the VM instances and explored the effects of the firewall rules on connectivity.

Next steps / learn more

Learn more about Google VPCs by reading [Virtual Private Cloud (VPC) Network Overview](https://cloud.google.com/vpc/docs/vpc).

**Manual last updated October 18, 2024**

**Lab last tested September 1, 2023**

[student-04-195dc9ae9109@qwiklabs.net](mailto:student-04-195dc9ae9109@qwiklabs.net)

OZ1mUjCngcQM

qwiklabs-gcp-01-f573e00d66cc

**Cloud Storage: Qwik Start - CLI/SDK**

experimentLabschedule30 minutesuniversal\_currency\_altNo costshow\_chartIntroductory

infoThis lab may incorporate AI tools to support your learning.

**GSP074**



**Overview**

Cloud Storage allows world-wide storage and retrieval of any amount of data at any time. You can use Cloud Storage for a range of scenarios including serving website content, storing data for archival and disaster recovery, or distributing large data objects to users via direct download.

In this hands-on lab you will learn how to create a storage bucket, upload objects to it, create folders and subfolders in it, and make objects publicly accessible using the Google Cloud command line.

Throughout this lab you'll be able to verify your work in the console by going to **Navigation menu** > **Cloud Storage**. You'll just need to refresh your browser after each command is run to see the new items you've created.

What you'll do

In this hands-on lab you will learn how to use Google Cloud command line to:

* Create a storage bucket
* Upload objects to the bucket
* Create folders and subfolders in the bucket
* Make objects in a storage bucket publicly accessible

**Setup and requirements**

Before you click the Start Lab button

Read these instructions. Labs are timed and you cannot pause them. The timer, which starts when you click **Start Lab**, shows how long Google Cloud resources will be made available to you.

This hands-on lab lets you do the lab activities yourself in a real cloud environment, not in a simulation or demo environment. It does so by giving you new, temporary credentials that you use to sign in and access Google Cloud for the duration of the lab.

To complete this lab, you need:

* Access to a standard internet browser (Chrome browser recommended).

**Note:** Use an Incognito or private browser window to run this lab. This prevents any conflicts between your personal account and the Student account, which may cause extra charges incurred to your personal account.

* Time to complete the lab---remember, once you start, you cannot pause a lab.

**Note:** If you already have your own personal Google Cloud account or project, do not use it for this lab to avoid extra charges to your account.

How to start your lab and sign in to the Google Cloud console

1. Click the **Start Lab** button. If you need to pay for the lab, a pop-up opens for you to select your payment method. On the left is the **Lab Details** panel with the following:
   * The **Open Google Cloud console** button
   * Time remaining
   * The temporary credentials that you must use for this lab
   * Other information, if needed, to step through this lab
2. Click **Open Google Cloud console** (or right-click and select **Open Link in Incognito Window** if you are running the Chrome browser).

The lab spins up resources, and then opens another tab that shows the **Sign in** page.

***Tip:*** Arrange the tabs in separate windows, side-by-side.

**Note:**If you see the **Choose an account** dialog, click **Use Another Account**.

1. If necessary, copy the **Username** below and paste it into the **Sign in** dialog.

student-03-251a9bc4055b@qwiklabs.net

Copied!

content\_copy

You can also find the **Username** in the **Lab Details** panel.

1. Click **Next**.
2. Copy the **Password** below and paste it into the **Welcome** dialog.

FNCx3QklqAMs

Copied!

content\_copy

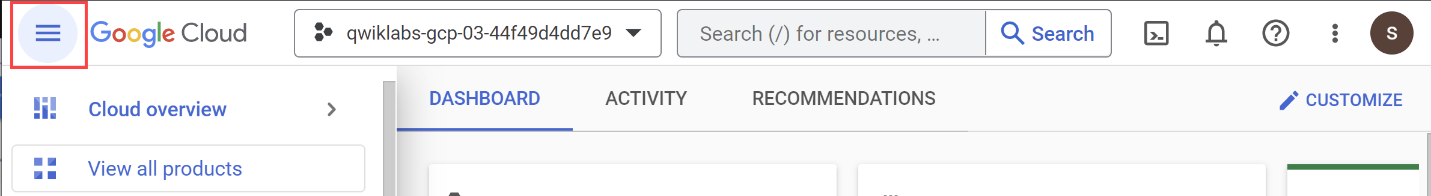
You can also find the **Password** in the **Lab Details** panel.

1. Click **Next**.

**Important:**You must use the credentials the lab provides you. Do not use your Google Cloud account credentials.**Note:**Using your own Google Cloud account for this lab may incur extra charges.

1. Click through the subsequent pages:
   * Accept the terms and conditions.
   * Do not add recovery options or two-factor authentication (because this is a temporary account).
   * Do not sign up for free trials.

After a few moments, the Google Cloud console opens in this tab.

**Note:** To view a menu with a list of Google Cloud products and services, click the **Navigation menu** at the top-left. 

Activate Cloud Shell

Cloud Shell is a virtual machine that is loaded with development tools. It offers a persistent 5GB home directory and runs on the Google Cloud. Cloud Shell provides command-line access to your Google Cloud resources.

1. Click **Activate Cloud Shell** Activate Cloud Shell icon at the top of the Google Cloud console.

When you are connected, you are already authenticated, and the project is set to your **Project\_ID**, qwiklabs-gcp-00-35d31f79dc0e. The output contains a line that declares the **Project\_ID** for this session:

Your Cloud Platform project in this session is set to qwiklabs-gcp-00-35d31f79dc0e

gcloud is the command-line tool for Google Cloud. It comes pre-installed on Cloud Shell and supports tab-completion.

1. (Optional) You can list the active account name with this command:

gcloud auth list

Copied!

content\_copy

1. Click **Authorize**.

**Output:**

ACTIVE: \*

ACCOUNT: student-03-251a9bc4055b@qwiklabs.net

To set the active account, run:

$ gcloud config set account `ACCOUNT`

1. (Optional) You can list the project ID with this command:

gcloud config list project

Copied!

content\_copy

**Output:**

[core]

project = qwiklabs-gcp-00-35d31f79dc0e

**Note:**For full documentation of gcloud, in Google Cloud, refer to [the gcloud CLI overview guide](https://cloud.google.com/sdk/gcloud).

Set the region

Set the project region for this lab:

gcloud config set compute/region us-central1

Copied!

content\_copy

**Task 1. Create a bucket**

In this lab you use [gcloud storage](https://cloud.google.com/sdk/gcloud/reference/storage) and [gsutil](https://cloud.google.com/storage/docs/gsutil) commands.

When you create a bucket you must follow the universal bucket naming rules, below.

**Bucket naming rules**

* Do not include sensitive information in the bucket name, because the bucket namespace is global and publicly visible.
* Bucket names must contain only lowercase letters, numbers, dashes (-), underscores (\_), and dots (.). Names containing dots require [verification](https://cloud.google.com/storage/docs/domain-name-verification).
* Bucket names must start and end with a number or letter.
* Bucket names must contain 3 to 63 characters. Names containing dots can contain up to 222 characters, but each dot-separated component can be no longer than 63 characters.
* Bucket names cannot be represented as an IP address in dotted-decimal notation (for example, 192.168.5.4).
* Bucket names cannot begin with the "goog" prefix.
* Bucket names cannot contain "google" or close misspellings of "google".
* Also, for DNS compliance and future compatibility, you should not use underscores (\_) or have a period adjacent to another period or dash. For example, ".." or "-." or ".-" are not valid in DNS names.

Use the make bucket (buckets create) command to make a bucket, replacing <YOUR\_BUCKET\_NAME> with a unique name that follows the bucket naming rules:

gcloud storage buckets create gs://<YOUR-BUCKET-NAME>

Copied!

content\_copy

This command is creating a bucket with default settings. To see what those default settings are, use the Cloud console **Navigation menu** > **Cloud Storage**, then click on your bucket name, and click on the **Configuration** tab.

That's it — you've just created a Cloud Storage bucket!

**Note:** If the bucket name is already taken, either by you or someone else, the command returns:

Creating gs://YOUR-BUCKET-NAME/...  
ServiceException: 409 Bucket YOUR-BUCKET-NAME already exists.

Try again with a different bucket name.

Test completed task

Click **Check my progress** to verify your performed task. If you've successfully created a Cloud Storage bucket, you'll see an assessment score.

Assessment Completed! Bucket count: 1. Bucket name(s): ["superman2013"]

Create a Cloud Storage bucket.

Check my progress

*Assessment Completed! Bucket count: 1. Bucket name(s): ["superman2013"]*

Test your understanding

Below is a multiple choice question to reinforce your understanding of this lab's concepts. Answer it to the best of your ability.

Each bucket has a default storage class, which you can specify when you create your bucket.

checkTrue

False

**Task 2. Upload an object into your bucket**

Use Cloud Shell to upload an object into a bucket.

1. To download this image (ada.jpg) into your bucket, enter this command into Cloud Shell:

curl https://upload.wikimedia.org/wikipedia/commons/thumb/a/a4/Ada\_Lovelace\_portrait.jpg/800px-Ada\_Lovelace\_portrait.jpg --output ada.jpg

Copied!

content\_copy

1. Use the gcloud storage cp command to upload the image from the location where you saved it to the bucket you created:

gcloud storage cp ada.jpg gs://YOUR-BUCKET-NAME

Copied!

content\_copy

**Note:**When typing your bucket name, you can use the tab key to autocomplete it.

You can see the image load into your bucket from the command line.

You've just stored an object in your bucket!

1. Now remove the downloaded image:

rm ada.jpg

Copied!

content\_copy

**Task 3. Download an object from your bucket**

* Use the gcloud storage cp command to download the image you stored in your bucket to Cloud Shell:

gcloud storage cp -r gs://YOUR-BUCKET-NAME/ada.jpg .

Copied!

content\_copy

If successful, the command returns:

Copying gs://YOUR-BUCKET-NAME/ada.jpg...

/ [1 files][360.1 KiB/2360.1 KiB]

Operation completed over 1 objects/360.1 KiB.

You've just downloaded the image from your bucket.

**Task 4. Copy an object to a folder in the bucket**

* Use the gcloud storage cp command to create a folder called image-folder and copy the image (ada.jpg) into it:

gcloud storage cp gs://YOUR-BUCKET-NAME/ada.jpg gs://YOUR-BUCKET-NAME/image-folder/

Copied!

content\_copy

**Note:** Compared to local file systems, [folders in Cloud Storage](https://cloud.google.com/sdk/gcloud/reference/storage/folders) have limitations, but many of the same operations are supported.

If successful, the command returns:

Copying gs://YOUR-BUCKET-NAME/ada.jpg [Content-Type=image/png]...

- [1 files] [ 360.1 KiB/ 360.1 KiB]

Operation completed over 1 objects/360.1 KiB

The image file has been copied into a new folder in your bucket.

Test completed task

Click **Check my progress** to verify your performed task. If you have successfully uploaded an object into a folder in your Cloud Storage bucket, you'll see an assessment score.

Assessment Completed! Bucket name(s): ["superman2013"] Object count: [2]

Copy an object to a folder in the bucket (ada.jpg).

Check my progress

*Assessment Completed! Bucket name(s): ["superman2013"] Object count: [2]*

**Task 5. List contents of a bucket or folder**

* Use the gcloud storage ls command to list the contents of the bucket:

gcloud storage ls gs://YOUR-BUCKET-NAME

Copied!

content\_copy

If successful, the command returns a message similar to:

gs://YOUR-BUCKET-NAME/ada.jpg

gs://YOUR-BUCKET-NAME/image-folder/

That's everything currently in your bucket.

**Task 6. List details for an object**

* Use the gcloud storage ls command, with the -l flag to get some details about the image file you uploaded to your bucket:

gcloud storage ls -l gs://YOUR-BUCKET-NAME/ada.jpg

Copied!

content\_copy

If successful, the command returns a message similar to:

306768 2017-12-26T16:07:570Z gs://YOUR-BUCKET-NAME/ada.jpg

TOTAL: 1 objects, 30678 bytes (360.1 KiB)

Now you know the image's size and date of creation.

**Task 7. Make your object publicly accessible**

* Use the gsutil acl ch command to grant all users read permission for the object stored in your bucket:

gsutil acl ch -u AllUsers:R gs://YOUR-BUCKET-NAME/ada.jpg

Copied!

content\_copy

If successful, the command returns:

Updated ACL on gs://YOUR-BUCKET-NAME/ada.jpg

Your image is now public, and can be made available to anyone.

Test completed ask

Click **Check my progress** to verify your performed task. If you have successfully shared an object from your storage bucket, you will see an assessment score.

Assessment Completed!

Make your object publicly accessible

Check my progress

*Assessment Completed!*

Validate that your image is publicly available.

* Go to **Navigation menu** > **Cloud Storage**, then click on the name of your bucket.

You should see your image with the **Public link** box. Click the **Copy URL** and open the URL in a new browser tab.

**Note:**Who are you looking at? This is [Ada Lovelace](https://en.wikipedia.org/wiki/Ada_Lovelace), credited with being the first computer programmer. She worked with mathematician and computer pioneer Charles Babbage, who proposed the [Analytical Engine](https://en.wikipedia.org/wiki/Analytical_Engine).

Her interest in the Analytical Engine lead to translating a paper on the machine by Italian mathematician Luigi Menabrea, adding her own extensive annotations. These notes are considered the first computer program - an algorithm designed to be carried out by the machine. She developed a vision of the capability of computers, going beyond number crunching, and examined how individuals and society relate to technology as a collaborative tool.

**Citation:** Ada Lovelace. (2015, October 22). Wikimedia Commons, the free media repository. Retrieved 08:01, May 31, 2022 from https://commons.wikimedia.org/w/index.php?title=Ada\_Lovelace&oldid=176490980, .

Test your understanding

Below is a multiple choice question to reinforce your understanding of this lab's concepts. Answer it to the best of your ability.

An access control list (ACL) is a mechanism you can use to define who has access to your buckets and objects.

checkTrue

False

**Task 8. Remove public access**

1. To remove this permission, use the command:

gsutil acl ch -d AllUsers gs://YOUR-BUCKET-NAME/ada.jpg

Copied!

content\_copy

If successful, the command returns:

Updated ACL on gs://YOUR-BUCKET-NAME/ada.jpg

You have removed public access to this object.

1. Verify that you've removed public access by clicking the **Refresh** button in the console. The checkmark will be removed.

Test your understanding

Below is a multiple choice question to reinforce your understanding of this lab's concepts. Answer it to the best of your ability.

You can stop publicly sharing an object by removing the permission entry that has:

close~~By updating storage class~~

checkallUsers

By removing project owner role

Submit

Delete objects

1. Use the gcloud storage rm command to delete an object - the image file in your bucket:

gcloud storage rm gs://YOUR-BUCKET-NAME/ada.jpg

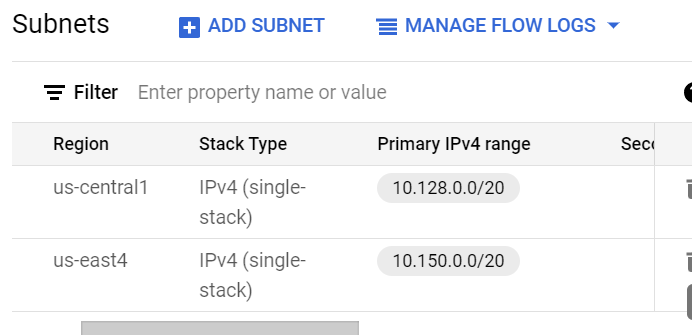
Copied!

content\_copy

If successful, the command returns:

Removing gs://YOUR-BUCKET-NAME/ada.jpg...

1. Refresh the console. The copy of the image file is no longer stored on Cloud Storage (though the copy you made in the image-folder/ folder still exists).



**Cloud SQL for MySQL: Qwik Start**

experimentLabschedule30 minutesuniversal\_currency\_altNo costshow\_chartIntroductory

infoThis lab may incorporate AI tools to support your learning.

**GSP151**



**Overview**

In this lab, you learn how to create and connect to a Cloud SQL for MySQL instance and perform basic SQL operations using the Cloud console and the mysql client.

What you'll do

* Create a Cloud SQL instance
* Connect to the instance in Cloud Shell
* Create a database and upload data

**Setup and requirements**

Before you click the Start Lab button

Read these instructions. Labs are timed and you cannot pause them. The timer, which starts when you click **Start Lab**, shows how long Google Cloud resources will be made available to you.

This hands-on lab lets you do the lab activities yourself in a real cloud environment, not in a simulation or demo environment. It does so by giving you new, temporary credentials that you use to sign in and access Google Cloud for the duration of the lab.

To complete this lab, you need:

* Access to a standard internet browser (Chrome browser recommended).

**Note:** Use an Incognito or private browser window to run this lab. This prevents any conflicts between your personal account and the Student account, which may cause extra charges incurred to your personal account.

* Time to complete the lab---remember, once you start, you cannot pause a lab.

**Note:** If you already have your own personal Google Cloud account or project, do not use it for this lab to avoid extra charges to your account.

How to start your lab and sign in to the Google Cloud console

1. Click the **Start Lab** button. If you need to pay for the lab, a pop-up opens for you to select your payment method. On the left is the **Lab Details** panel with the following:
   * The **Open Google Cloud console** button
   * Time remaining
   * The temporary credentials that you must use for this lab
   * Other information, if needed, to step through this lab
2. Click **Open Google Cloud console** (or right-click and select **Open Link in Incognito Window** if you are running the Chrome browser).

The lab spins up resources, and then opens another tab that shows the **Sign in** page.

***Tip:*** Arrange the tabs in separate windows, side-by-side.

**Note:**If you see the **Choose an account** dialog, click **Use Another Account**.

1. If necessary, copy the **Username** below and paste it into the **Sign in** dialog.

student-03-251a9bc4055b@qwiklabs.net

Copied!

content\_copy

You can also find the **Username** in the **Lab Details** panel.

1. Click **Next**.
2. Copy the **Password** below and paste it into the **Welcome** dialog.

FNCx3QklqAMs

Copied!

content\_copy

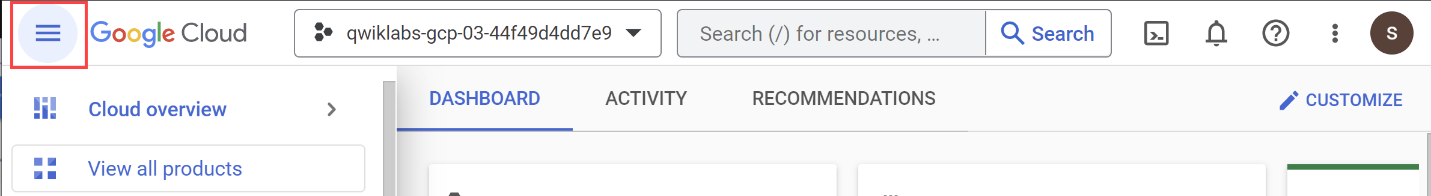
You can also find the **Password** in the **Lab Details** panel.

1. Click **Next**.

**Important:**You must use the credentials the lab provides you. Do not use your Google Cloud account credentials.**Note:**Using your own Google Cloud account for this lab may incur extra charges.

1. Click through the subsequent pages:
   * Accept the terms and conditions.
   * Do not add recovery options or two-factor authentication (because this is a temporary account).
   * Do not sign up for free trials.

After a few moments, the Google Cloud console opens in this tab.

**Note:** To view a menu with a list of Google Cloud products and services, click the **Navigation menu** at the top-left. 

Activate Cloud Shell

Cloud Shell is a virtual machine that is loaded with development tools. It offers a persistent 5GB home directory and runs on the Google Cloud. Cloud Shell provides command-line access to your Google Cloud resources.

1. Click **Activate Cloud Shell** Activate Cloud Shell icon at the top of the Google Cloud console.

When you are connected, you are already authenticated, and the project is set to your **Project\_ID**, qwiklabs-gcp-00-813f044d957a. The output contains a line that declares the **Project\_ID** for this session:

Your Cloud Platform project in this session is set to qwiklabs-gcp-00-813f044d957a

gcloud is the command-line tool for Google Cloud. It comes pre-installed on Cloud Shell and supports tab-completion.

1. (Optional) You can list the active account name with this command:

gcloud auth list

Copied!

content\_copy

1. Click **Authorize**.

**Output:**

ACTIVE: \*

ACCOUNT: student-03-251a9bc4055b@qwiklabs.net

To set the active account, run:

$ gcloud config set account `ACCOUNT`

1. (Optional) You can list the project ID with this command:

gcloud config list project

Copied!

content\_copy

**Output:**

[core]

project = qwiklabs-gcp-00-813f044d957a

**Note:**For full documentation of gcloud, in Google Cloud, refer to [the gcloud CLI overview guide](https://cloud.google.com/sdk/gcloud).

**Task 1. Create a Cloud SQL instance**

1. From the **Navigation menu** (Navigation menu icon) click on **SQL**.
2. Click **Create Instance**.
3. Choose **MySQL** database engine.
4. Enter Instance ID as myinstance.
5. In the password field click on the **Generate** link and the eye icon to see the password. **Save** the password to use in the next section.
6. Select the database version as **MySQL 8**.
7. For **Choose a Cloud SQL edition**, select **Enterprise** edition.
8. For **Preset** choose **Development** (4 vCPU, 16 GB RAM, 100 GB Storage, Single zone).

**Warning:** if you choose a preset larger than Development, your project will be flagged and your lab will be terminated.

1. Set **Region** as us-east4.
2. Set the **Multi zones (Highly available)** > **Primary Zone** field as us-east4-b.
3. Click **CREATE INSTANCE**.

It might take a few minutes for the instance to be created. Once it is, you will see a green checkmark next to the instance name.

1. Click on the Cloud SQL instance. The **SQL Overview** page opens.

**Test Completed Task**

Click **Check my progress** to verify your performed task. If you have successfully created a Cloud SQL instance, you will see an assessment score.

Assessment Completed! Cloud SQL instance created. SQL instance name(s): ["myinstance"]

Create a Cloud SQL instance

Check my progress

*Assessment Completed! Cloud SQL instance created. SQL instance name(s): ["myinstance"]*

**Test your understanding**

Below is a true/false question to reinforce your understanding of this lab's concepts. Answer it to the best of your ability.

Instance ID is used to uniquely identify your instance within the project.

checkTrue

False

**Task 2. Connect to your instance using the mysql client in Cloud Shell**

1. In the Cloud Console, click the **Cloud Shell** icon in the upper right corner.
2. Click **Continue**.
3. At the Cloud Shell prompt, connect to your Cloud SQL instance by running the following:

gcloud sql connect myinstance --user=root

Copied!

content\_copy

Click **Authorize**.

1. Enter your root password when prompted. **Note:** The cursor will not move.
2. Press the **Enter** key when you're done typing.

You should now see the mysql prompt.

**Task 3. Create a database and upload data**

1. Create a SQL database called guestbook on your Cloud SQL instance:

CREATE DATABASE guestbook;

Copied!

content\_copy

**Test completed task**

Click **Check my progress** to verify your performed task. If you have successfully created a custom database on the Cloud SQL instance, you will see an assessment score.

Assessment Completed! Databse created successfully. Database name(s): ["guestbook"]

Create a database.

Check my progress

*Assessment Completed! Databse created successfully. Database name(s): ["guestbook"]*

1. Insert the following sample data into the guestbook database:

USE guestbook;

CREATE TABLE entries (guestName VARCHAR(255), content VARCHAR(255),

entryID INT NOT NULL AUTO\_INCREMENT, PRIMARY KEY(entryID));

INSERT INTO entries (guestName, content) values ("first guest", "I got here!");

INSERT INTO entries (guestName, content) values ("second guest", "Me too!");

Copied!

content\_copy

1. Now retrieve the data:

SELECT \* FROM entries;

Copied!

content\_copy

You should see:

+--------------+-------------------+---------+

| guestName | content | entryID |

+--------------+-------------------+---------+

| first guest | I got here! | 1 |

| second guest | Me too! | 2 |

+--------------+-------------------+---------+

2 rows in set (0.00 sec)

mysql>

**C**

**Cloud SQL for PostgreSQL: Qwik Start**

experimentLabschedule30 minutesuniversal\_currency\_altNo costshow\_chartIntroductory

infoThis lab may incorporate AI tools to support your learning.

**GSP152**



**Overview**

In this lab you'll learn how to create and connect to a Google Cloud SQL PostgreSQL instance and perform basic SQL operations using the Cloud Console and the psql client.

**Setup and requirements**

Before you click the Start Lab button

Read these instructions. Labs are timed and you cannot pause them. The timer, which starts when you click **Start Lab**, shows how long Google Cloud resources will be made available to you.

This hands-on lab lets you do the lab activities yourself in a real cloud environment, not in a simulation or demo environment. It does so by giving you new, temporary credentials that you use to sign in and access Google Cloud for the duration of the lab.

To complete this lab, you need:

* Access to a standard internet browser (Chrome browser recommended).

**Note:** Use an Incognito or private browser window to run this lab. This prevents any conflicts between your personal account and the Student account, which may cause extra charges incurred to your personal account.

* Time to complete the lab---remember, once you start, you cannot pause a lab.

**Note:** If you already have your own personal Google Cloud account or project, do not use it for this lab to avoid extra charges to your account.

How to start your lab and sign in to the Google Cloud console

1. Click the **Start Lab** button. If you need to pay for the lab, a pop-up opens for you to select your payment method. On the left is the **Lab Details** panel with the following:
   * The **Open Google Cloud console** button
   * Time remaining
   * The temporary credentials that you must use for this lab
   * Other information, if needed, to step through this lab
2. Click **Open Google Cloud console** (or right-click and select **Open Link in Incognito Window** if you are running the Chrome browser).

The lab spins up resources, and then opens another tab that shows the **Sign in** page.

***Tip:*** Arrange the tabs in separate windows, side-by-side.

**Note:**If you see the **Choose an account** dialog, click **Use Another Account**.

1. If necessary, copy the **Username** below and paste it into the **Sign in** dialog.

student-03-48de6b3374c6@qwiklabs.net

Copied!

content\_copy

You can also find the **Username** in the **Lab Details** panel.

1. Click **Next**.
2. Copy the **Password** below and paste it into the **Welcome** dialog.

V364D9mlBwUB

Copied!

content\_copy

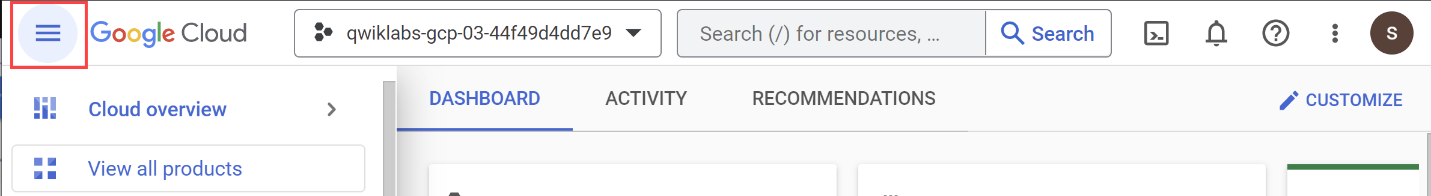
You can also find the **Password** in the **Lab Details** panel.

1. Click **Next**.

**Important:**You must use the credentials the lab provides you. Do not use your Google Cloud account credentials.**Note:**Using your own Google Cloud account for this lab may incur extra charges.

1. Click through the subsequent pages:
   * Accept the terms and conditions.
   * Do not add recovery options or two-factor authentication (because this is a temporary account).
   * Do not sign up for free trials.

After a few moments, the Google Cloud console opens in this tab.

**Note:** To view a menu with a list of Google Cloud products and services, click the **Navigation menu** at the top-left. 

**Task 1. Create a Cloud SQL instance**

1. Click on the menu icon in the top left of the screen to see the **Navigation menu** (Navigation menu icon).
2. In the left menu of the Console, click on **SQL**.
3. Click **Create Instance**.
4. Click **Choose PostgreSQL**.
5. Create your instance with the following settings:

* Enter myinstance for **Instance ID**.
* Enter a password for the postgres user. Save or remember this password, you'll need it in the next section.
* For **Choose a Cloud SQL edition**, select **Enterprise**.
* For **Region** select us-east4.
* Leave the default values for the other fields.

1. Click **Create Instance**.

You are returned to the instances list. Your new instance is greyed out while it initializes and starts.

After a few minutes your instance is created and you can continue to the next section. If it seems to be taking a long time refresh your browser.

Test completed task

Click **Check my progress** to verify your performed task. If you have completed the task successfully you will granted with an assessment score.

Assessment Completed! Cloud SQL instance created. SQL instance names: ["myinstance"]

Create a Cloud SQL instance

Check my progress

*Assessment Completed! Cloud SQL instance created. SQL instance names: ["myinstance"]*

**Task 2. Test your understanding**

Below are multiple choice questions to reinforce your understanding of this lab's concepts. Answer them to the best of your abilities.

What PostgreSQL Database version used in lab to create Cloud SQL instance?

5.6

check16.4

1.0

5.7

Submit

**Task 3. Connect to your instance using the psql client in the Cloud Shell**

1. In the Cloud Console, click **Cloud Shell** (Cloud Shell icon) in the upper right corner.
2. Then click **Continue** if prompted.
3. At the Cloud Shell prompt, connect to your Cloud SQL instance by running:

gcloud sql connect myinstance --user=postgres

Copied!

content\_copy

1. Enter your postgres password.

**Note:**The cursor will not move. Press **Enter** when you're done typing.

You should now see the psql prompt.

**Task 4. Upload data into the postgres database**

1. Insert sample data into the postgres database:

CREATE TABLE guestbook (guestName VARCHAR(255), content VARCHAR(255),

entryID SERIAL PRIMARY KEY);

INSERT INTO guestbook (guestName, content) values ('first guest', 'I got here!');

INSERT INTO guestbook (guestName, content) values ('second guest', 'Me too!');

Copied!

content\_copy

1. Retrieve the data:

SELECT \* FROM guestbook;

Copied!

content\_copy

You should now see:

guestname | content | entryid

--------------+-------------+---------

first guest | I got here! | 1

second guest | Me too! | 2

(2 rows)

postgres=>

You have created a Google Cloud SQL PostgreSQL instance and connected to it.

**Task 5. Test your understanding**

Below are multiple choice questions to reinforce your understanding of this lab's concepts. Answer them to the best of your abilities.

What is the name of default database in Postgres Cloud SQL instance?

close~~guestbook~~

checkpostgres

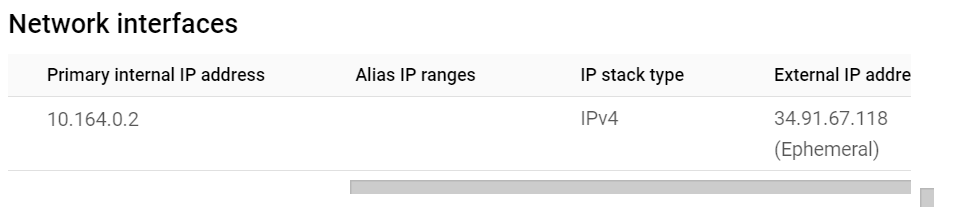
performance\_schema

information\_schema

Submit

**Congratulations!**

34.133.51.66



**Google Cloud Fundamentals: Getting Started with Cloud Storage and Cloud SQL**

experimentLabschedule1 hour 15 minutesuniversal\_currency\_altNo costshow\_chartIntroductory

infoThis lab may incorporate AI tools to support your learning.

**Overview**

In this lab, you create a Cloud Storage bucket and place an image in it. You also configure an application running in Compute Engine to use a database managed by Cloud SQL. For this lab, you configure a web server with PHP, a web development environment that is the basis for popular blogging software. Outside this lab, you will use analogous techniques to configure these packages.

You also configure the web server to reference the image in the Cloud Storage bucket.

**Objectives**

In this lab, you learn how to perform the following tasks:

* Create a Cloud Storage bucket and place an image into it.
* Create a Cloud SQL instance and configure it.
* Connect to the Cloud SQL instance from a web server.
* Use the image in the Cloud Storage bucket on a web page.

**Task 1. Sign in to the Google Cloud Console**

For each lab, you get a new Google Cloud project and set of resources for a fixed time at no cost.

1. Click the **Start Lab** button. If you need to pay for the lab, a pop-up opens for you to select your payment method. On the left is the **Lab Details** panel with the following:
   * The **Open Google Cloud console** button
   * Time remaining
   * The temporary credentials that you must use for this lab
   * Other information, if needed, to step through this lab
2. Click **Open Google Cloud console** (or right-click and select **Open Link in Incognito Window** if you are running the Chrome browser).

The lab spins up resources, and then opens another tab that shows the **Sign in** page.

***Tip:*** Arrange the tabs in separate windows, side-by-side.

**Note:**If you see the **Choose an account** dialog, click **Use Another Account**.

1. If necessary, copy the **Username** below and paste it into the **Sign in** dialog.

student-04-8c9e8f384b4c@qwiklabs.net

Copied!

content\_copy

You can also find the **Username** in the **Lab Details** panel.

1. Click **Next**.
2. Copy the **Password** below and paste it into the **Welcome** dialog.

y9oUDjxoD7yJ

Copied!

content\_copy

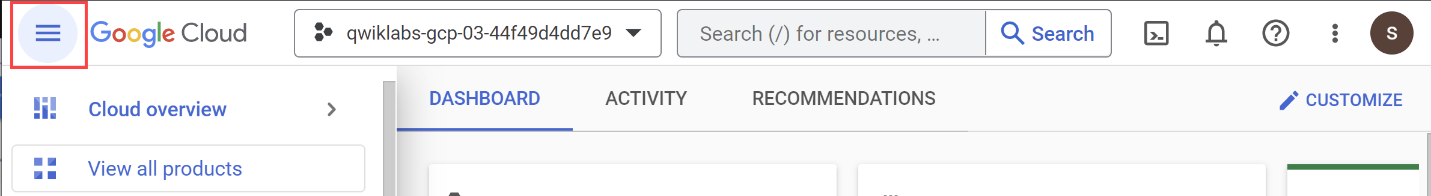
You can also find the **Password** in the **Lab Details** panel.

1. Click **Next**.

**Important:**You must use the credentials the lab provides you. Do not use your Google Cloud account credentials.**Note:**Using your own Google Cloud account for this lab may incur extra charges.

1. Click through the subsequent pages:
   * Accept the terms and conditions.
   * Do not add recovery options or two-factor authentication (because this is a temporary account).
   * Do not sign up for free trials.

After a few moments, the Google Cloud console opens in this tab.

**Note:** To view a menu with a list of Google Cloud products and services, click the **Navigation menu** at the top-left. 

**Task 2. Deploy a web server VM instance**

1. In the Google Cloud console, on the **Navigation menu** (Navigation menu icon), click **Compute Engine** > **VM instances**.
2. Click **Create Instance**.
3. On the **Create an Instance** page, for **Name**, type bloghost
4. For **Region** and **Zone**, select the region and zone assigned by Qwiklabs.
5. For **Machine type**, accept the default.
6. For **Boot disk**, if the **Image** shown is not **Debian GNU/Linux 12**, click **Change** and select **Debian GNU/Linux 11 (bullseye)**.
7. Leave the defaults for **Identity and API access** unmodified.
8. For **Firewall**, click **Allow HTTP traffic**.
9. Click **Advanced options** to open that section of the dialog.
10. Click **Management** to open that section of the dialog
11. Scroll down to the Automation section, and enter the following script as the value for **Startup script**:

apt-get update

apt-get install apache2 php php-mysql -y

service apache2 restart

Copied!

content\_copy

**Note:** Be sure to supply that script as the value of the **Startup script** field. If you accidentally put it into another field, it won't be executed when the VM instance starts.

1. Leave the remaining settings as their defaults, and click **Create**.

**Note:** Instance can take about two minutes to launch and be fully available for use.

1. On the **VM instances** page, copy the **bloghost** VM instance's internal and external IP addresses to a text editor for use later in this lab.

Click *Check my progress* to verify the objective.

Assessment Completed!

Deploy a web server VM instance

Check my progress

*Assessment Completed!*

**Task 3. Create a Cloud Storage bucket using the gcloud storage command line**

All Cloud Storage bucket names must be globally unique. To ensure that your bucket name is unique, these instructions will guide you to give your bucket the same name as your Google Cloud project ID, which is also globally unique.

Cloud Storage buckets can be associated with either a region or a multi-region location: **US**, **EU**, or **ASIA**. In this activity, you associate your bucket with the multi-region closest to the region and zone that Qwiklabs or your instructor assigned you to.

1. On the **Google Cloud console**, on the top right toolbar, click the **Activate Cloud Shell** Activate Cloud Shell icon. If a dialog box appears, click **Continue**.
2. For convenience, enter your chosen location into an environment variable called LOCATION. Enter one of these commands:

export LOCATION=US

Copied!

content\_copy

Or

export LOCATION=EU

Copied!

content\_copy

Or

export LOCATION=ASIA

Copied!

content\_copy

1. In Cloud Shell, the DEVSHELL\_PROJECT\_ID environment variable contains your project ID. Enter this command to make a bucket named after your project ID:

gcloud storage buckets create -l $LOCATION gs://$DEVSHELL\_PROJECT\_ID

Copied!

content\_copy

If prompted, click **Authorize** to continue.

1. Retrieve a banner image from a publicly accessible Cloud Storage location:

gcloud storage cp gs://cloud-training/gcpfci/my-excellent-blog.png my-excellent-blog.png

Copied!

content\_copy

1. Copy the banner image to your newly created Cloud Storage bucket:

gcloud storage cp my-excellent-blog.png gs://$DEVSHELL\_PROJECT\_ID/my-excellent-blog.png

Copied!

content\_copy

1. Modify the Access Control List of the object you just created so that it's readable by everyone:

gsutil acl ch -u allUsers:R gs://$DEVSHELL\_PROJECT\_ID/my-excellent-blog.png

Copied!

content\_copy

Click *Check my progress* to verify the objective.

Assessment Completed!

Create a Cloud Storage bucket using the gcloud storage command line

Check my progress

*Assessment Completed!*

**Task 4. Create the Cloud SQL instance**

1. In the Google Cloud console, on the **Navigation menu** (Navigation menu icon), click **SQL**.
2. Click **Create instance**.
3. For **Choose a database engine**, select **Choose MySQL**.
4. For **Choose a Cloud SQL edition**, click **Enterprise** and then select **Sandbox** from the dropdown.
5. For **Instance ID,** type **blog-db**, and for **Root password** type a password of your choice.

**Note:** Choose a password that you remember. There's no need to obscure the password because you use mechanisms to connect that aren't open access to everyone.

1. Select **Single zone** and set the region and zone assigned by Qwiklabs.

**Note:** This is the same region and zone into which you launched the **bloghost** instance. The best performance is achieved by placing the client and the database close to each other.

1. Click **Create Instance**.

**Note:** Wait for the instance to finish deploying. It will take a few minutes.

1. Click the name of the instance, **blog-db**, to open its details page.
2. From the SQL instances details page, copy the **Public IP address** for your SQL instance to a text editor for use later in this lab.
3. Click **Users** menu on the left-hand side, and then click **Add User Account**.
4. For **User name**, type blogdbuser
5. For **Password**, type a password of your choice. Make a note of it.
6. Click **Add** to add the user account in the database.

**Note:** Wait for the user to be created.

1. Click **Connections** menu on the left-hand side, and then click **Networking** tab.
2. Click **Add a Network**.

**Note:** If you're offered the choice between a **Private IP** connection and a **Public IP** connection, choose **Public IP** for purposes of this lab.**Note:** The **Add network** button may be unavailable if the user account creation is not yet complete.

1. For **Name**, type web front end
2. For **Network**, type the external IP address of your **bloghost** VM instance, followed by /32

The result will look like this:

35.192.208.2/32

**Note:** Be sure to use the external IP address of your VM instance followed by /32. Do not use the VM instance's internal IP address. Do not use the sample IP address shown here.

1. Click **Done** to finish defining the authorized network.
2. Click **Save** to save the configuration change.

**Note**: If the message appears like **Another operation is in progress**, wait for few minutes until you see the green check for **blog-db** to save the configuration.

Click *Check my progress* to verify the objective.

Assessment Completed!

Create the Cloud SQL instance

Check my progress

*Assessment Completed!*

**Task 5. Configure an application in a Compute Engine instance to use Cloud SQL**

1. On the **Navigation menu** (Navigation menu icon), click **Compute Engine** > **VM instances**.
2. In the VM instances list, click **SSH** in the row for your VM instance **bloghost**.
3. In your ssh session on **bloghost**, change your working directory to the document root of the web server:

cd /var/www/html

Copied!

content\_copy

1. Use the **nano** text editor to edit a file called **index.php**:

sudo nano index.php

Copied!

content\_copy

1. Paste the content below into the file:

<html>

<head><title>Welcome to my excellent blog</title></head>

<body>

<h1>Welcome to my excellent blog</h1>

<?php

$dbserver = "CLOUDSQLIP";

$dbuser = "blogdbuser";

$dbpassword = "DBPASSWORD";

// In a production blog, we would not store the MySQL

// password in the document root. Instead, we would store

// it in a Secret Manger. For more information see

// https://cloud.google.com/sql/docs/postgres/use-secret-manager

try {

$conn = new PDO("mysql:host=$dbserver;dbname=mysql", $dbuser, $dbpassword);

// set the PDO error mode to exception

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

echo "Connected successfully";

} catch(PDOException $e) {

echo "Database connection failed:: " . $e->getMessage();

}

?>

</body></html>

Copied!

content\_copy

**Note:** In a later step, you will insert your Cloud SQL instance's IP address and your database password into this file. For now, leave the file unmodified.

1. Press **Ctrl+O**, and then press **Enter** to save your edited file.
2. Press **Ctrl+X** to exit the nano text editor.
3. Restart the web server:

sudo service apache2 restart

Copied!

content\_copy

1. Open a new web browser tab and paste into the address bar your **bloghost** VM instance's external IP address followed by **/index.php**. The URL will look like this:

35.192.208.2/index.php

**Note:** Be sure to use the external IP address of your VM instance followed by /index.php. Do not use the VM instance's internal IP address. Do not use the sample IP address shown here.

When you load the page, you will see that its content includes an error message beginning with the words:

Database connection failed: ...

**Note:** This message occurs because you have not yet configured PHP's connection to your Cloud SQL instance.

1. Return to your ssh session on **bloghost**. Use the **nano** text editor to edit **index.php** again.

sudo nano index.php

Copied!

content\_copy

1. In the **nano** text editor, replace CLOUDSQLIP with the Cloud SQL instance Public IP address that you noted above. Leave the quotation marks around the value in place.
2. In the **nano** text editor, replace DBPASSWORD with the Cloud SQL database password that you defined above. Leave the quotation marks around the value in place.
3. Press **Ctrl+O**, and then press **Enter** to save your edited file.
4. Press **Ctrl+X** to exit the nano text editor.
5. Restart the web server:

sudo service apache2 restart

Copied!

content\_copy

1. Return to the web browser tab in which you opened your **bloghost** VM instance's external IP address. When you load the page, the following message appears:

Database connection succeeded.

**Note:** In an actual blog, the database connection status would not be visible to blog visitors. Instead, the database connection would be managed solely by the administrator.

**Task 6. Configure an application in a Compute Engine instance to use a Cloud Storage object**

1. In the Google Cloud console, click **Cloud Storage > Buckets**.
2. Click the bucket that is named after your Google Cloud project.
3. In this bucket, there is an object called **my-excellent-blog.png**. Copy the URL behind the link icon that appears in that object's **Public access** column, or behind the words "Public link" if shown.

**Note:** If you see neither a link icon nor a "Public link", try refreshing the browser. If you still do not see a link icon, return to Cloud Shell and confirm that your attempt to change the object's Access Control list with the **gsutil acl ch** command was successful.

1. Return to your ssh session on your **bloghost** VM instance.
2. Enter this command to set your working directory to the document root of the web server:

cd /var/www/html

Copied!

content\_copy

1. Use the **nano** text editor to edit **index.php**:

sudo nano index.php

Copied!

content\_copy

1. Use the arrow keys to move the cursor to the line that contains the **h1** element. Press **Enter** to open up a new, blank screen line, and then paste the URL you copied earlier into the line.
2. Paste this HTML markup immediately before the URL:

<img src='

1. Place a closing single quotation mark and a closing angle bracket at the end of the URL:

'>

The resulting line will look like this:

<img src='https://storage.googleapis.com/qwiklabs-gcp-0005e186fa559a09/my-excellent-blog.png'>

The effect of these steps is to place the line containing <img src='...'> immediately before the line containing <h1>...</h1>

**Note:** Do not copy the URL shown here. Instead, copy the URL shown by the Storage browser in your own Cloud Platform project.

1. Press **Ctrl+O**, and then press **Enter** to save your edited file.
2. Press **Ctrl+X** to exit the nano text editor.
3. Restart the web server:

sudo service apache2 restart

Copied!

content\_copy

1. Return to the web browser tab in which you opened your **bloghost** VM instance's external IP address. When you load the page, its content now includes a banner image.