Creating a Virtual Machine

GSP001

Overview

Compute Engine lets you create virtual machines that run different operating systems, including multiple flavors of Linux (Debian, Ubuntu, Suse, Red Hat, CoreOS) and Windows Server, on Google infrastructure. You can run thousands of virtual CPUs on a system that is designed to be fast and to offer strong consistency of performance.

In this hands-on lab, you'll create virtual machine instances of various machine types using the Google Cloud Console and the `gcloud` command line. You'll also learn how to connect an NGINX web server to your virtual machine.

Although you can easily copy and paste commands from the lab to the appropriate place, we recommend that you type the commands yourself to reinforce your understanding of the core concepts.

What you'll do

- Create a virtual machine with the Cloud Console.
- Create a virtual machine with the `gcloud` command line.
- Deploy a web server and connect it to a virtual machine.

Prerequisites

- Familiarity with standard Linux text editors such as `vim`, `emacs`, or `nano` will be helpful.

Setup

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 Qwiklabs 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.

What you need

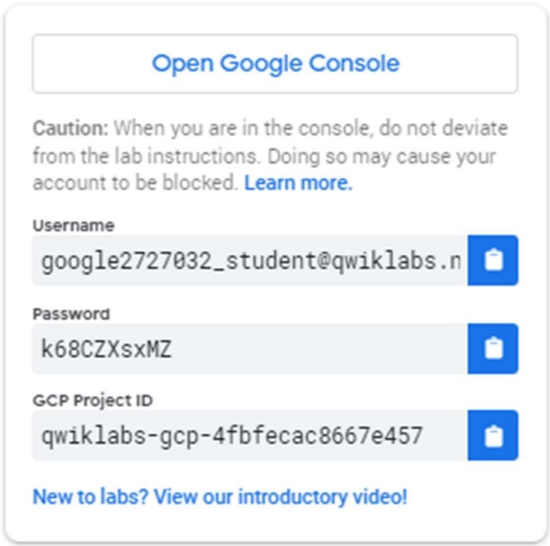
To complete this lab, you need:

- Access to a standard internet browser (Chrome browser recommended).
- Time to complete the lab.
  Note: If you already have your own personal Google Cloud account or project, do not use it for this lab.

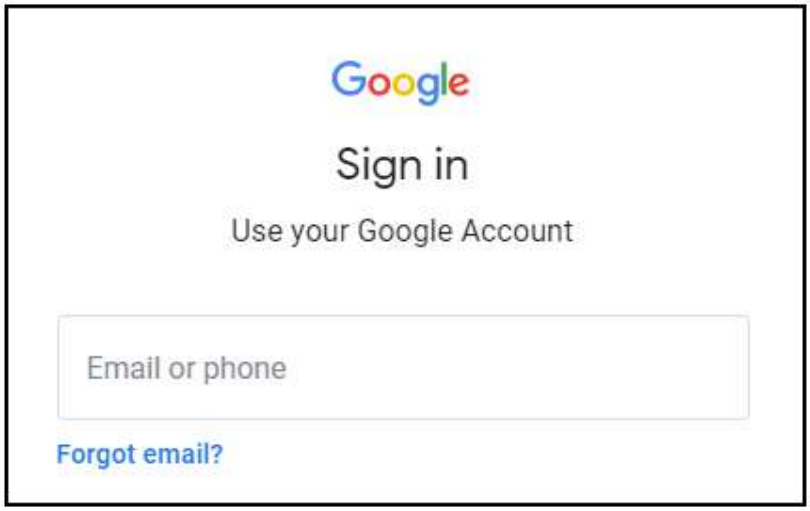
Note: If you are using a Pixelbook, open an Incognito window to run this lab.

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 a panel populated with the temporary credentials that you must use for this lab.

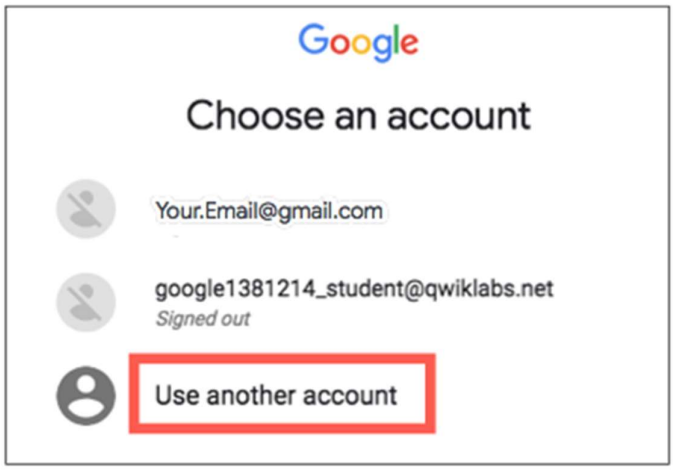


2. Copy the username, and then click **Open Google Console**. The lab spins up resources, and then opens another tab that shows the **Sign in** page.



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

If you see the **Choose an account** page, click **Use Another Account**.



3. In the **Sign in** page, paste the username that you copied from the Connection Details panel. Then copy and paste the password.

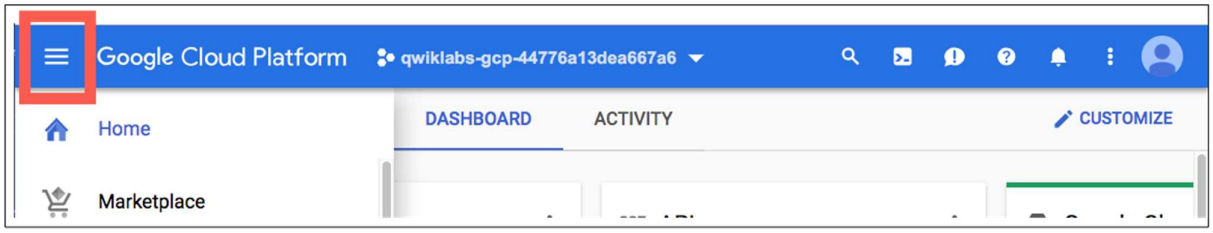
   Important: You must use the credentials from the Connection Details panel. Do not use your Qwiklabs credentials. If you have your own Google Cloud account, do not use it for this lab (avoids incurring charges).

4. 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 Cloud Console opens in this tab.

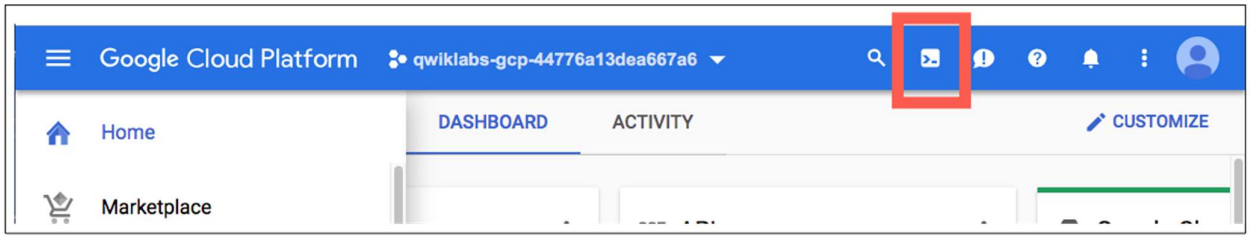
Note: You can view the menu with a list of Google Cloud Products and Services by clicking the **Navigation menu** at the top-left.



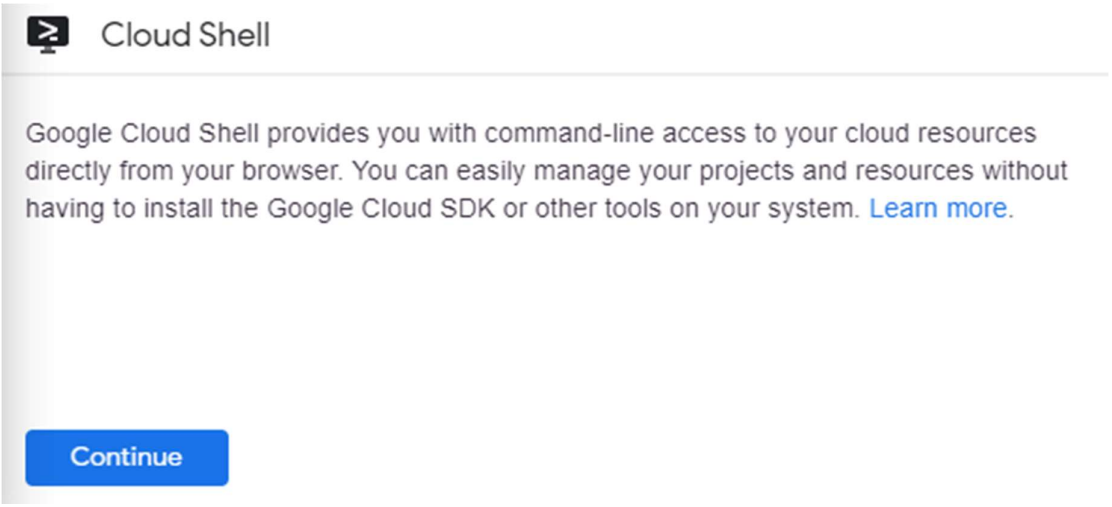
Activate Cloud Shell

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.

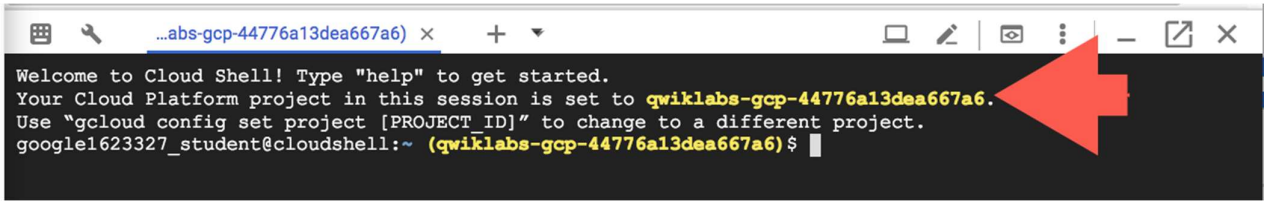
In the Cloud Console, in the top right toolbar, click the **Activate Cloud Shell** button.



Click **Continue**.



It takes a few moments to provision and connect to the environment. When you are connected, you are already authenticated, and the project is set to your *PROJECT_ID*. For example:



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

You can list the active account name with this command:

```
gcloud auth list
```

(Output)

```
Credentialed accounts:
 - <myaccount>@<mydomain>.com (active)
```

(Example output)

```
Credentialed accounts:
 - google1623327_student@qwiklabs.net
```

You can list the project ID with this command:

```
gcloud config list project
```

(Output)

```
[core]
project = <project_ID>
```

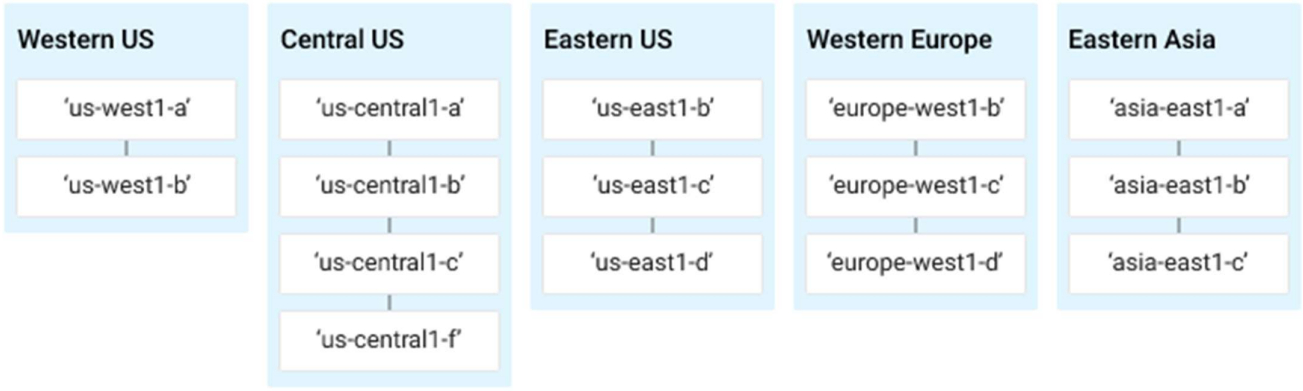
(Example output)

```
[core]
project = qwiklabs-gcp-44776a13dea667a6
```

For full documentation of `gcloud` see the gcloud command-line tool overview.

Understanding Regions and Zones

Certain Compute Engine resources live in regions or zones. A region is a specific geographical location where you can run your resources. Each region has one or more zones. For example, the us-central1 region denotes a region in the Central United States that has zones `us-central1-a`, `us-central1-b`, `us-central1-c`, and `us-central1-f`.



Resources that live in a zone are referred to as zonal resources. Virtual machine Instances and persistent disks live in a zone. To attach a persistent disk to a virtual machine instance, both resources must be in the same zone. Similarly, if you want to assign a static IP address to an instance, the instance must be in the same region as the static IP.

Learn more about regions and zones and see a complete list in Regions & Zones documentation.

Task 1: Create a new instance from the Cloud Console

In this section, you'll learn how to create new pre-defined machine types with Compute Engine from the Cloud Console.

1. In the Cloud Console, on the **Navigation menu** (☰), click **Compute Engine** > **VM Instances**.

   This may take a minute to initialize for the first time.

2. To create a new instance, click **CREATE INSTANCE**.

3. There are many parameters you can configure when creating a **new instance**. Use the following for this lab:

Field	Value	Additional Information
Name	**gcelab**	Name for the VM instance
Region	**us-central1 (Iowa)**	For more information about regions, see Regions and Zones.
Zone	**us-central1-c**	**Note:** Remember the zone that you selected: you'll need it later. For more information about zones, see Regions and Zones.
Series	**N1**	Name of the series
Machine Type	**2 vCPUs**	This is an (n1-standard-2), 2-CPU, 7.5GB RAM instance. Several machine types are available, ranging from micro instance types to 32-core/208GB RAM instance types. For more information, see Machine Types. **Note:** A new project has a default resource quota, which may limit the number of CPU cores. You can request more when you work on projects outside this lab.
Boot Disk	**New 10 GB standard persistent disk OS Image: Debian GNU/Linux 10 (buster)**	Several images are available, including Debian, Ubuntu, CoreOS, and premium images such as Red Hat Enterprise Linux and Windows Server. For more information, see Operating System documentation.
Firewall	**Allow HTTP traffic**	Select this option in order to access a web server that you'll install later. **Note:** This will automatically create a firewall rule to allow HTTP traffic on port 80.

4. Click **Create**.

It should take about a minute for the machine to be created. After that, the new virtual machine is listed on the **VM Instances** page.

5. To use **SSH** to connect to the virtual machine, in the row for your machine, click **SSH**.

   This launches an SSH client directly from your browser.

   Note: For more information, see how can we connect to an instance using ssh.

Task 2: Install an NGINX web server

Now you'll install an NGINX web server, one of the most popular web servers in the world, to connect your virtual machine to something.

1. In the SSH terminal, to get `root` access, run the following command:

```
2. sudo su -
```

3. As the `root` user, update your OS:

```
4. apt-get update
```

Expected output (Do not copy):

```
Get:1 http://security.debian.org stretch/updates InRelease [94.3 kB]
Ign http://deb.debian.org strech InRelease
Get:2 http://deb.debian.org strech-updates InRelease [91.0 kB]
...
```

5. Install **NGINX**:

```
6. apt-get install nginx -y
```

Expected output (Do not copy):

```
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
...
```

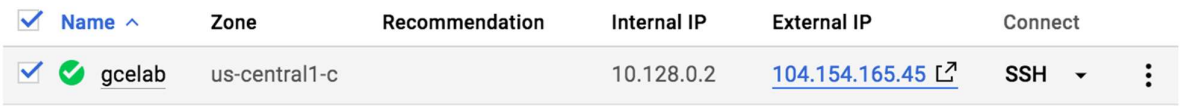
7. Confirm that **NGINX is running**:

```
8. ps auwx | grep nginx
```

Expected output (Do not copy):

```
root      2330  0.0  0.0 159532  1628 ?        Ss   14:06   0:00 nginx: master
process /usr/sbin/nginx -g daemon on; master_process on;
www-data  2331  0.0  0.0 159864  3204 ?        S    14:06   0:00 nginx: worker
process
www-data  2332  0.0  0.0 159864  3204 ?        S    14:06   0:00 nginx: worker
process
root      2342  0.0  0.0  12780   988 pts/0    S+   14:07   0:00 grep nginx
```

9. To see the web page, return to the Cloud Console and click the **External IP** link in the row for your machine, or add the **External IP** value to `http://EXTERNAL_IP/` in a new browser window or tab.



This default web page should open:

To check your progress in this lab, click **Check my progress** below. A checkmark means you're successful.

Create a Compute Engine instance and add an NGINX Server to your instance with necessary firewall rules.

Task 3: Create a new instance with gcloud

Instead of using the Cloud Console to create a virtual machine instance, you can use the command line tool `gcloud`, which is pre-installed in Google Cloud Shell. Cloud Shell is a Debian-based virtual machine loaded with all the development tools you'll need (`gcloud`, `git`, and others) and offers a persistent 5-GB home directory.

Note: If you want to try this on your own machine, read the gcloud command line tool guide.

1. In the Cloud Shell, use `gcloud` to create a new virtual machine instance from the command line:

```
2. gcloud compute instances create gcelab2 --machine-type n1-standard-2 --zone us-
   central1-c
```

Expected output (Do not copy):

```
Created [...gcelab2].
NAME     ZONE           MACHINE_TYPE  ...    STATUS
gcelab2  us-central1-c  n1-standard-2 ...    RUNNING
```

To check your progress in this lab, click **Check my progress** below. A checkmark means you're successful.

The new instance has these default values:

- The latest Debian 10 (buster) image.
- The `n1-standard-2` machine type. In this lab, you can select one of these other machine types: `n1-highmem-4` or `n1-highcpu-4`. When you're working on a project outside Qwiklabs, you can also specify a custom machine type.
- A root persistent disk with the same name as the instance; the disk is automatically attached to the instance.

3. To see all the defaults, run:

```
4. gcloud compute instances create --help
```

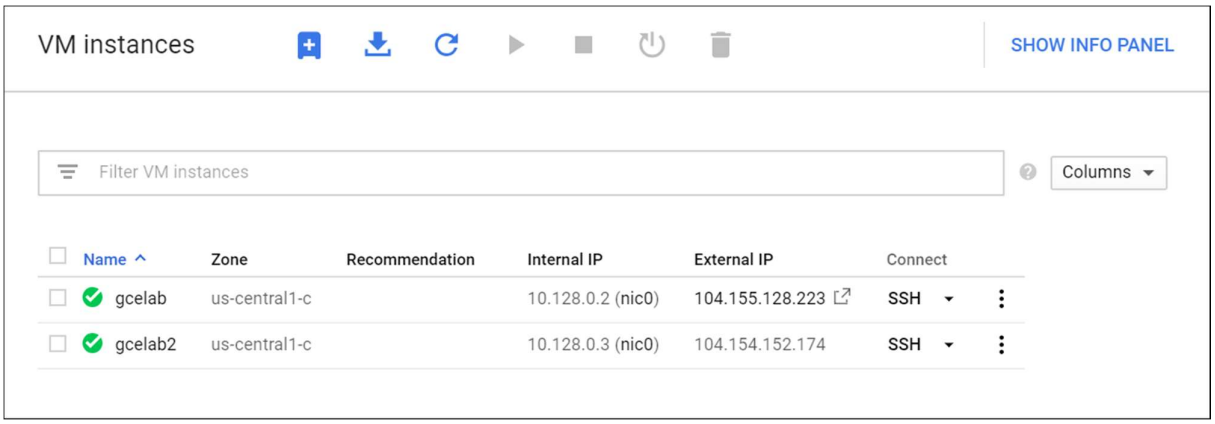
Note: You can set the default region and zones that `gcloud` uses if you are always working within one region/zone and you don't want to append the `--zone` flag every time. To do this, run these commands:

```
gcloud config set compute/zone ...

gcloud config set compute/region ...
```

5. To exit `help`, press **CTRL + C**.

6. In the Cloud Console, on the **Navigation menu**, click **Compute Engine > VM instances**. Your 2 new instances should be listed.



7. You can also use SSH to connect to your instance via `gcloud`. Make sure to add your zone, or omit the `--zone` flag if you've set the option globally:

```
8. gcloud compute ssh gcelab2 --zone us-central1-c
```

Expected output (Do not copy):

```
WARNING: The public SSH key file for gcloud does not exist.
WARNING: The private SSH key file for gcloud does not exist.
WARNING: You do not have an SSH key for gcloud.
WARNING: [/usr/bin/ssh-keygen] will be executed to generate a key.
This tool needs to create the directory
[/home/gcpstaging306_student/.ssh] before being able to generate SSH
Keys.
```

9. Type **Y** to continue.

```
10.  Do you want to continue? (Y/n)
```

11. Press **ENTER** through the passphrase section to leave the passphrase empty.

```
12.  Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase)
```

13. After connecting, disconnect from SSH by exiting from the remote shell:

```
14.  exit
```

Test your knowledge

Test your knowledge about Google Cloud by taking our quiz. (Please select multiple correct options if necessary.)

- [ ] ugh which of the following ways can you create a VM instance in Compute Engine(Compute Engine)?
- [ ] gcloud command line tool

The Cloud Console

Congratulations!

Compute Engine is the foundation of Google Cloud's infrastructure as a service. You created a virtual machine with Compute Engine and can now map your existing server infrastructure, load balancers, and network topology to Google Cloud.

Finish your Quest

This self-paced lab is part of the Qwiklab Google Cloud Essentials Quest. A Quest is a series of related labs that form a learning path. Completing a Quest earns you a badge to recognize your achievement. You can make your badge (or badges) public and link to them in your online resume or social media account. Enroll in this Quest and get immediate completion credit for taking this lab. See other available Qwiklabs Quests.

Take your next lab

Continue your Quest with Getting Started with Cloud Shell and gcloud, or check out these suggestions:

- Getting Started with Cloud Shell and gcloud
- Provision Services with Google Cloud Marketplace

Next Step/learn more

- For an overview of VMs, see Virtual Machine Instances.
- Check out how to migrate VMs to the Google Cloud.
- Learn more about subnetworks and network topology.
- And then be sure to choose the right VM type by reviewing Choosing a VM Machine.

Google Cloud Training & Certification

...helps you make the most of Google Cloud technologies. Our classes include technical skills and best practices to help you get up to speed quickly and continue your learning journey. We offer fundamental to advanced level training, with on-demand, live, and virtual

options to suit your busy schedule. Certifications help you validate and prove your skill and expertise in Google Cloud technologies.
Manual Last Updated: April 01, 2021
Lab Last Tested: April 01, 2021