

A Tour of Qwiklabs and the Google Cloud Platform

45 minutesFree

Rate Lab

GSP282



Google Cloud Self-Paced Labs

Overview

The [Google Cloud Platform](#) (GCP) is a suite of cloud services hosted on Google's infrastructure. From computing and storage, to data analytics, machine learning, and networking, GCP offers a wide variety of services and

APIs that can be integrated with any cloud-computing application or project—be it personal or enterprise-grade.

In this introductory-level lab, you will take your first steps with GCP by getting hands-on practice with the [Google Cloud Platform console](#)—an in-browser UI that lets you access and manage Google Cloud services. You will identify key features of GCP and also learn the ins and outs of the Qwiklabs environment. If you are new to cloud computing or looking for an overview of GCP and Qwiklabs, you are in the right place. Read on to learn about the specifics of this lab and areas that you will get hands-on practice with.

What you will learn

In this lab, you will do the following:

- Learn about the Qwiklabs platform and identify key features of a lab environment.
- Learn about (and possibly purchase) Qwiklabs credits and launch an instance of a lab.
- Learn how to access the GCP console with specific credentials.
- Learn about GCP projects and identify common misconceptions that surround them.
- Learn how to use the GCP navigation menu to identify types of GCP services.
- Learn about primitive roles and use the Cloud IAM service to inspect actions available to specific users.
- Learn about Cloud Shell and run commands that use the `gcloud` toolkit.
- Learn about the API library and examine its chief features.
- Use tools that are pre-installed in Cloud Shell and run commands like `touch`, `nano`, and `cat` to create, edit, and output the content of files.

Prerequisites

This is an **introductory level** lab and the first lab you should take if you're unfamiliar with GCP or Qwiklabs. If you are already experienced with Qwiklabs and the Google Cloud Platform Console, check out the following labs (if you decide to take one, make sure you **end your lab now**):

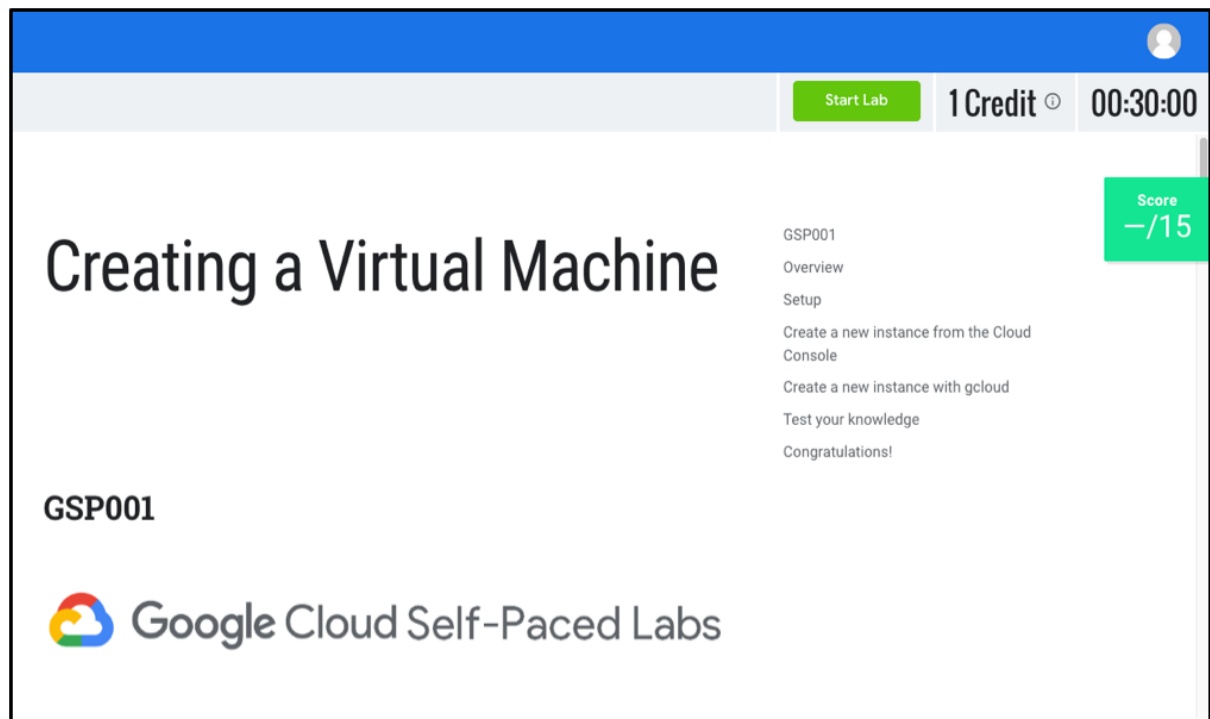
- [Getting Started with Cloud Shell and gcloud](#)
- [Creating a Virtual Machine](#)

Once you're ready, scroll down to get your lab set up.

Qwiklabs Fundamentals

Features and components

Regardless of topic or expertise level, all Qwiklabs share a common interface. This lab that you're taking should look quite similar to the following:



Note: You are not taking the "Creating a Virtual Machine" lab shown above, it is used merely as an example to highlight common features across Qwiklabs.

Read the following lab component definitions and then locate them in the top-right corner of the interface.

Start Lab (button)

Clicking this will create a temporary GCP environment with all the necessary services and credentials enabled so you can dive in and get hands-on practice with the lab's material. Once clicked, you will start a countdown timer that gives you a set amount of time to complete the lab's steps. Once the timer hits 00:00:00, your lab environment and temporary credentials will be deleted.

Credit

The price of a lab. 1 Credit is *usually* equivalent to 1 US dollar (the more credits you purchase, the bigger the discount you receive.) Introductory level

labs (like this one) cost 1 credit, fundamental level cost 5, advanced 7, and expert cost 9 credits. The more specialized labs cost more because they involve heavier computing tasks and demand more GCP resources.

Time

Specifies the amount of time you have to complete a lab. As mentioned earlier, once you click the "Start Lab" button the timer will count down until it reaches 00:00:00. When it does, your temporary GCP environment and resources will be deleted. Ample time is given to complete a lab, but make sure you don't work on something else while a lab is running—you risk losing all of your hard work!

Score

Most, but not all labs include a score. This feature is called "activity tracking" and ensures you complete specified steps in a lab. To pass a lab with activity tracking, you need to complete all the steps *in order* (only then will you receive completion credit.)

Purchasing Credits

If you haven't already, click on the **Start Lab** button now. If you have an "Access Code" (a coupon that covers the cost of a single lab regardless of expertise level) or have already purchased credits, you can skip down to the next section "Starting a Lab". Otherwise, click **BUY CREDITS**. This will take you to a new page that offers different credit packages:

The screenshot shows the Google Cloud Self-Paced Labs interface for the lab titled "Creating a Virtual Machine". The interface is divided into several sections:










- Header:** A blue bar at the top contains a hamburger menu, a back arrow, the lab title "Creating a Virtual Machine", a user profile icon, a "Start Lab" button, "1 Credit" with a refresh icon, and a timer showing "00:30:00".
- Left Sidebar:**
 - 30m access - 30m completion:** Includes a star rating and links for "Rate Lab" and "Lab Details".
 - Student Resources:** A link to "Create a Virtual Machine, GCP Essentials".
 - Connection Details:** A section with a link to "Open Google Console" and three input fields for "Username", "Password", and "GCP Project ID", each with a "Show" button.
 - Footer:** A link that says "New to labs? View our introductory video!".
- Main Content Area:**
 - Title:** "Creating a Virtual Machine" in large black font.
 - Lab ID:** "GSP001" in bold black font.
 - Logo:** The Google Cloud logo followed by the text "Google Cloud Self-Paced Labs".
 - Overview Section:**
 - Header:** "Overview" in bold.
 - Text:** "Google Compute Engine lets you create virtual machines running 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 has been designed to be fast and to offer strong consistency of performance."
 - Text:** "In this hands-on lab you'll learn how to create virtual machine instances of various machine types using the Google Cloud Platform (GCP) Console and using the `gcloud` command line. You'll also learn how to connect an NGINX web server to your virtual machine."
 - Text:** "Although you can easily copy and paste commands from the lab to the appropriate place, students should type the commands themselves to reinforce their understanding of the core concepts."
- Right Sidebar:**
 - Score:** A green box showing "Score --/15".
 - Table of Contents:** A list of links: "GSP001", "Overview", "Setup", "Create a new instance from the Cloud Console", "Create a new instance with gcloud", "Test your knowledge", and "Congratulations!".

Enter in the amount of credits you want to buy and click **Pay with Credit Card**. This will take you to a secure panel, prompting you to enter in your credit card information:

Buy Credits

Labs typically cost 0 to 15 credits. Minimum purchase is 5 credits. Unused credits expire after 6 months. **Or you can buy a monthly Advantage Subscription and get unlimited credits and take as many labs as you like.** The subscription renews automatically each month until you cancel.

If you wish to purchase more than 20,000 lab credits, please contact enterprise sales at sales@qwiklabs.com.

<input checked="" type="radio"/>	 Advantage Subscription	\$55.00
<input type="radio"/>	 8 Credits	\$8.00
<input type="radio"/>	 10 Credits	\$10.00
<input type="radio"/>	 15 Credits	\$15.00
<input type="radio"/>	 40 Credits (5% Discount)	\$38.00
<input type="radio"/>	 400 Credits (10% Discount)	\$360.00
<input type="radio"/>	 1000 Credits (15% Discount)	\$850.00
<input type="radio"/>	 4000 Credits (20% Discount)	\$3,200.00
<input type="radio"/>	 Other <input type="text" value="8"/>	

[Have a Promo Code?](#)

Total:\$55.00

[Pay with Credit Card](#)

Enter in your credit card information and click **Pay**. You're all set to take this and any other Qwiklabs! You can always buy more credits by clicking on the profile icon in the top right corner and selecting "Buy Credits". You will also be prompted to purchase more if you run out and try to start a new lab.

Starting a Lab


Now that we understand the key features and components of a lab, click on the **Start Lab** button in the top-right corner. If you have an Access Code, enter it in now and click **Launch with an Access code**. Otherwise, click **Launch with 1 Credit**:

Start Lab

1 Credit ⓘ

00:30:00

GSP001

 Google Cloud Self-Paced Labs

Score
—/15

GSP001

Overview

Setup

Create a new instance from the Cloud Console

Create a new instance with gcloud

Test your knowledge

Congratulations!

Overview

Google Compute Engine lets you create virtual machines running 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 has been designed to be fast and to offer strong consistency of performance.

In this hands-on lab you'll learn how to create virtual machine instances of various machine types using the Google Cloud Platform (GCP) Console and using 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, students should type the commands themselves to reinforce their understanding of the core concepts

What you'll do

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

←

Prerequisites

It may take a moment for the GCP environment and credentials to spin up—please be patient. Once the timer in the top right corner starts ticking and the **Start Lab** button turns into a red **End Lab** button, everything is in place and you're all set to log in to the Google Cloud Platform Console. Make sure that you do not click the **End Lab** button until you have completed all the tasks required of you. Once clicked, your temporary credentials will be invalidated and you will no longer be able to access the work you've done throughout the lab. Only when you reach the end and have made sure that you've completed all the necessary steps, click the **End Lab** button. You must click this button when you finish—if you do not, you won't be able to take

another lab afterwards (Qwiklabs has protections in place that prevent concurrent enrollment.)

Test your understanding

Answer the following multiple choice questions to reinforce your understanding of the concepts we've covered so far.

This builds a temporary environment in GCP.



Score



Start lab (button)



Credit



Time

Submit

Once the timer reaches 00:00:00, you will lose access to your temporary GCP environment.



False



True

Submit

Some labs have tracking, which scores your completion of hands-on lab activities.



True



False

Submit

In order to receive completion credit for a lab that has tracking, you must complete the required hands-on lab activities.



False



True

Submit


Accessing the GCP Console


Key Terms


Now that your lab instance is up and running, take a look at the Connection Details panel on the left-hand side. It should be populated with an Open Google Console button and Username, Password, and GCP Project ID fields.

Open Google Console

Caution: When you are in the console, do not deviate from the lab instructions. Doing so may cause your account to be blocked. [Learn more.](#)

Username
googlefree160101_student@qwiklabs 

Password
9P49dbgW5TvG 

GCP Project ID
qwiklabs-gcp-56a6b4e51533ac69 

Note: Your credentials will resemble, but not match the above—every lab instance generates new temporary credentials.

Before we go any further, let's examine each of these components.

Open Google Console

This is a button that takes you to the [Google Cloud Platform Console](#)—the web console and central development hub for GCP. As you start working in GCP, you will be doing the majority of your work from this interface. All of the GCP Qwiklabs use the console in some form or another.

GCP Project ID

A [GCP Project](#) is an organizing entity for your Google Cloud resources. It often times contains resources and services—for example, it may hold a pool of virtual machines, a set of databases, and a network that connects them with one another. Projects also contain settings and permissions, which specify security rules and who has access to what resources.

A *GCP Project ID* is a unique identifier that is used to link GCP resources and APIs to your specific project. Project IDs are unique across GCP, meaning that there can only be one **qwiklabs-gcp-xxx....**, which makes it globally identifiable.

Username and Password

These are credentials representing an identity in the GCP Identity and Access Management (IAM) service that has access permissions (a role or roles), which allow you to work with GCP resources in the project you've been allocated. These credentials are *temporary*, and will only work for the access time of the lab. This means that once the timer hits 0, you will no longer be able to access your GCP project with those credentials.

Signing in to GCP

Now that you have a better understanding of the Connection Details panel, let's use the details it contains to sign in to the GCP Console. Click on the **Open Google Console** button. This will open the GCP sign in page in a new browser tab.

You should now be on a page that resembles the following:



Sign in

with your Google Account

Email or phone

[Forgot email?](#)

Not your computer? Use Guest mode to sign in privately.

[Learn more](#)

[Create account](#)

Next

If you've ever signed in to a Google application like Gmail, this page should look familiar. To sign in to the console, copy the **Username** from the Connection Details and paste it in to the "Email or phone" field and hit enter. *Wait! Make sure that you used the `googlexxxxxx_student@qwiklabs.net` email to sign in, NOT your personal or company email address!*

Go back and copy the **Password** from the provisioned credentials on the Qwiklabs lab page and paste it in to the GCP sign in "Password" field and hit enter.

The username that resembles `googlexxxxxx_student@qwiklabs.net` is a Google account that has been created for your use as a Qwiklabs student. It has a specific domain name, which is "qwiklabs.net", and has been assigned IAM roles that allow you to access the GCP Project that you have been provisioned.

If you successfully logged in, your page should resemble the following:



Welcome to your new account

Welcome to your new account: gcpstaging23131_student@qwiklabs.net. Your account is compatible with many [Google services](#), but your qwiklabs.net administrator decides which services you may access using your account. For tips about using your new account, visit the Google [Help Center](#).

When you use Google services, your domain administrator will have access to your gcpstaging23131_student@qwiklabs.net account information, including any data you store with this account in Google services. You can learn more [here](#), or by consulting your organization's privacy policy, if one exists. You can choose to maintain a separate account for your personal use of any Google services, including email. If you have multiple Google accounts, you can [manage which account you use](#) with Google services and [switch between them](#) whenever you choose. Your username and profile picture can help you ensure that you're using the intended account.

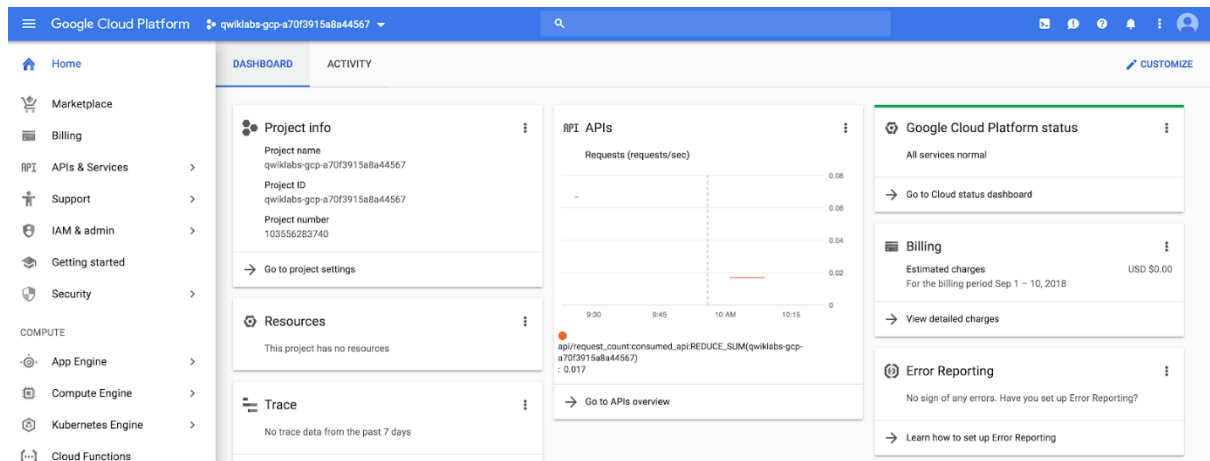
If your organization provides you access to the G Suite [core services](#), your use of those services is governed by your organization's G Suite agreement. Any other Google services your administrator enables ("Additional Services") are available to you under the [Google Terms of Service](#) and the [Google Privacy Policy](#). Certain Additional Services may also have [service-specific terms](#). Your use of any services your administrator allows you to access constitutes acceptance of applicable service-specific terms.

Click "Accept" below to indicate that you understand this description of how your gcpstaging23131_student@qwiklabs.net account works and agree to the [Google Terms of Service](#) and the [Google Privacy Policy](#).

Accept

Go ahead and click **Accept** to indicate your acknowledgement of Google's terms of service and privacy policy. You will then be brought to a "Protect your account" page. Since this is a temporary account, don't worry about updating recovery phone numbers or emails. Click **Done**.

You will now be brought to the "Updates to Terms of Service" page—for email updates regarding future announcements check the **No** box. Check the **Yes** box to agree to the Google Cloud Platform's terms of service. And just like that, you've successfully accessed the Google Cloud Platform Console with your Qwiklabs credentials! Your page should now resemble the following:



Test your understanding

Answer the following multiple choice questions to reinforce your understanding of the concepts we've covered so far.

What field is NOT found in the Connection Details panel?



Open Google Console



Password



GCP Project ID



System admin

Submit

The username in the Connection Details panel, which resembles googlexxxxxx_student@quwiklabs.net is a GCP IAM identity.



False



True

Submit

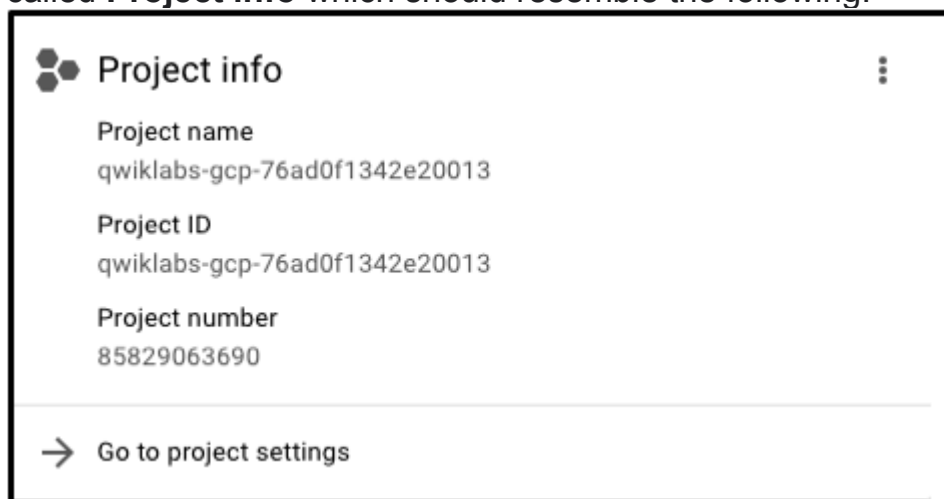
Now that we're signed in to our GCP Console and understand the basics of our credentials, let's learn a little bit more about GCP projects.

Projects in the GCP Console

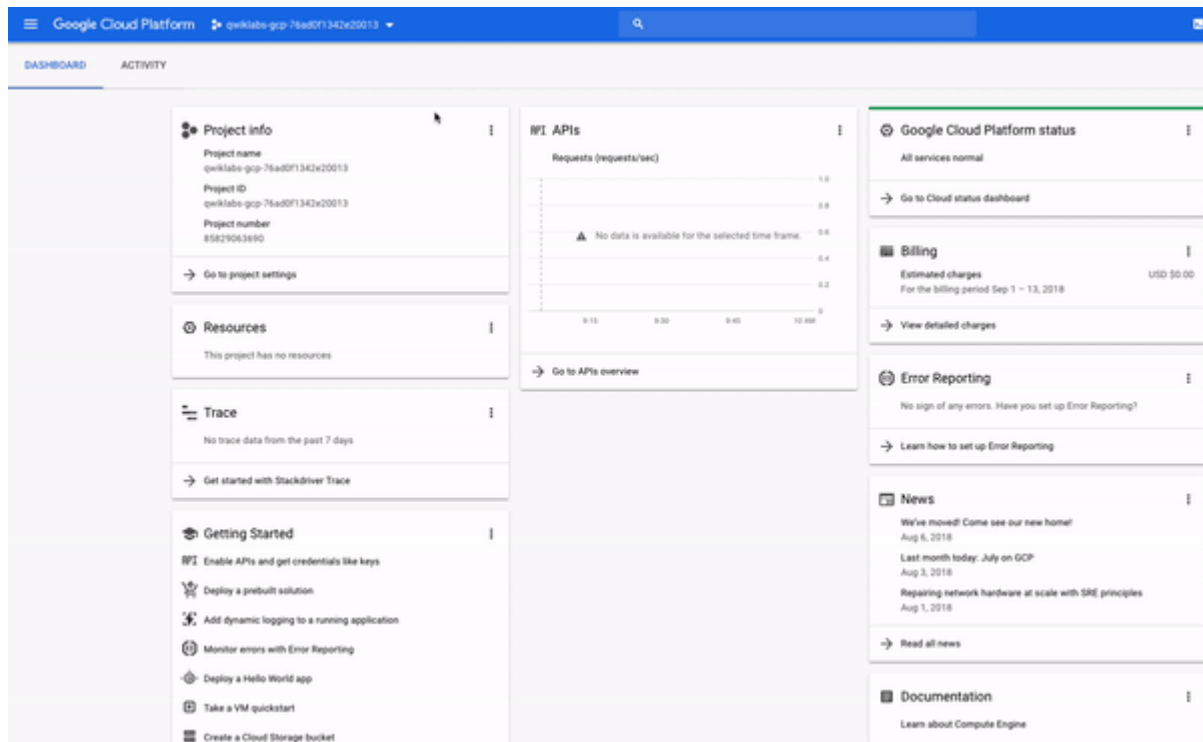
We touched on GCP projects earlier when we examined the components of the "Connection Details" panel. Here's the definition once again:

A [GCP Project](#) is an organizing entity for your Google Cloud resources. It often times contains resources and services—for example, it may hold a pool of virtual machines, a set of databases, and a network that connects them with one another. Projects also contain settings and permissions, which specify security rules and who has access to what resources.

If you look in the top-left side of the console, you will see a panel called **Project info** which should resemble the following:



As you see, your project has a **name**, **ID**, and **number**. These identifiers are frequently used when interacting with GCP services. You are working out of one project so you can get practice with a specific service or feature of GCP. You probably haven't noticed it, but you actually have access to more than one GCP project. In fact, in some labs you may be provisioned more than one project to accomplish the tasks assigned. If you click on the drop-down menu with your project name and select **ALL**, you will see that there is a "Qwiklabs Resources" project visible as well:



Do not switch over to the Qwiklabs Resources Project at this point! However, you may use it later in other labs.

It's not uncommon for large enterprises or experienced users of GCP to have dozens to thousands of GCP projects. Organizations use GCP in different ways, so projects are a good way to separate cloud-computing services (by team or product for example.)

"Qwiklabs Resources" is a project that contains files, datasets, and machine images for certain labs and can be accessed from every GCP lab environment. It's important to note that "Qwiklabs Resources" is shared (read only) with all Qwiklabs users, meaning you won't be able to delete or modify it.

The GCP project that you are working out of and whose name resembles `qwiklabs-gcp-xxx...` is *temporary*, meaning the project and everything it contains will be deleted once the lab ends. Whenever you start a new lab, you will be given access to one or more new GCP project(s), and there (not "Qwiklabs Resources") is where you will run all of the lab steps.

Test your understanding

Answer the following multiple choice questions to reinforce your understanding of the concepts we've covered so far.

An organizing entity for anything you build with the Google Cloud Platform.



Password



Cloud Storage bucket



Username



GCP Project

Submit

Qwiklabs Resources is shared (read only) with all Qwiklabs users, meaning you won't be able to delete or modify it.



False



True

Submit

Qwiklabs Resources is the project where you run all of your lab steps.



False

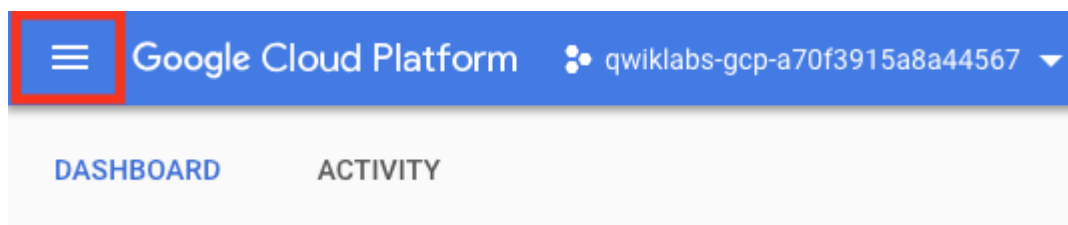


True

Submit

Navigation Menu and Services

In the top-left corner, you will notice a three-line icon that resembles the following:



Clicking on this will reveal (or hide) a **navigation menu** that points to GCP's core services. If the menu isn't exposed, click on the icon now and scroll through to see the types of services offered:

Project info

Project name
quwiklabs-gcp-76ad0f1342e20013

Project ID
quwiklabs-gcp-76ad0f1342e20013

Project number
85829063690

→ Go to project settings

Resources


This project has no resources

Trace

No trace data from the past 7 days


→ Get started with Stackdriver Trace


Getting Started


 Enable APIs and get credentials like keys

 Deploy a prebuilt solution


 Add dynamic logging to a running application


 Monitor errors with Error Reporting

 Deploy a Hello World app

 Take a VM quickstart

 Create a Cloud Storage bucket

 Create a Cloud Function

 Install the Cloud SDK

→ Explore all tutorials

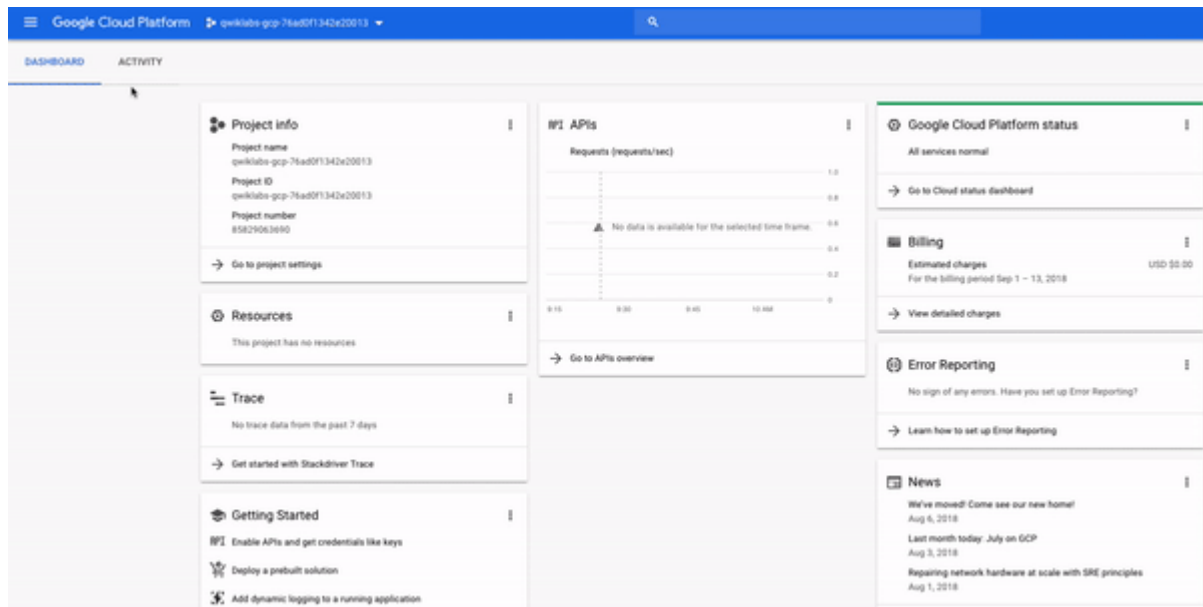
The navigation menu is an important component of the GCP console—it offers quick access to the platform's services and also outlines its offerings. If you scroll through the menu, you will see that there are seven categories of GCP services:

- **Compute:** houses a variety of machine types that support any type of workload. The different computing options let you decide how involved you want to be with operational details and infrastructure amongst other things.
 - **Storage:** data storage and database options for structured or unstructured, relational or non relational data.
 - **Networking:** services that balance application traffic and provision security rules amongst other things.
 - **Stackdriver:** a suite of cross-cloud logging, monitoring, trace, and other service reliability tools.
 - **Tools:** services for developers managing deployments and application build pipelines.
 - **Big Data:** services that allow you to process and analyze large datasets.
 - **Artificial Intelligence:** a suite of APIs that run specific artificial intelligence and machine learning tasks on the Google Cloud platform.
- [This link](#) will take you to documentation that covers each of these categories in more depth if you are interested.

Roles and Permissions

Earlier we mentioned that besides cloud computing services, GCP also houses a collection of permissions and roles that define who has access to what resources. We can use the [Cloud Identity and Access Management \(IAM\)](#) service to inspect and modify such roles and permissions.

If closed, open up the navigation menu. Then near the top click **IAM & admin**. This will take you to a page that contains a list of users, which specifies permissions and roles granted to certain accounts. Try to sift through these and find the "@qwiklabs" username you signed in with:



You should find something similar:

Filter table			
<input type="checkbox"/> Type	Member ↑	Name	Role
<input type="checkbox"/>	85829063690-compute@developer.gserviceaccount.com	Compute Engine default service account	Editor
<input type="checkbox"/>	85829063690@cloudbuild.gserviceaccount.com		Cloud Build Service Account
<input type="checkbox"/>	85829063690@cloudservices.gserviceaccount.com	Google APIs Service Agent	Editor
<input type="checkbox"/>	936076353769-dcb7hgk8cpl26aetf99c7min706qfrr@developer.gserviceaccount.com		Owner
<input type="checkbox"/>	gcpstaging23396_student@qwklabs.net	gcpstaging23396_student@qwklabs.net student	Editor
<input type="checkbox"/>	qwklabs-gcp-76ad0f1342e20013@appspot.gserviceaccount.com	App Engine default service account	Editor
<input type="checkbox"/>	qwklabs-gcp-76ad0f1342e20013@qwklabs-gcp-76ad0f1342e20013.iam.gserviceaccount.com	ql-api	Editor
<input type="checkbox"/>	service-85829063690@cloud-ml.google.com.iam.gserviceaccount.com	Google Cloud ML Engine Service Agent	Cloud ML Service Agent
<input type="checkbox"/>	service-85829063690@cloudcomposer-accounts.iam.gserviceaccount.com	Cloud Composer Service Agent	Cloud Composer API Service Agent

The **member** field is set to "google23396_student@qwklabs.net" (which matches the username you logged in with) and the **name** field is set to "google23396_student@qwklabs.net student". You will see that the **Role** field is set to "Editor", which is one of three *primitive roles* offered by GCP.

Primitive roles set project-level permissions and unless otherwise specified, they control access and management to all GCP services.

The following table pulls definitions from the [roles documentation](#), which gives a brief overview of viewer, editor, and owner role permissions:

Role Name	Permissions
roles/viewer	Permissions for read-only actions that do not affect state, such as viewing (but not modifying) existing resources or data.
roles/editor	All viewer permissions, plus permissions for actions that modify state, such as changing existing resources.
roles/owner	All editor permissions and permissions for the following actions: <ul style="list-style-type: none"> Manage roles and permissions for a project and all resources within the project.

- Set up billing for a project.

So as an editor you will be able to create, modify, and delete GCP resources. However, you won't be able to add or delete members from GCP projects.

Test your understanding

Answer the following multiple choice questions to reinforce your understanding of the concepts we've covered so far.

Offers quick access to the platform's services and also outlines its offerings.



Compute



Stackdriver



Networking



Navigation menu

Submit

Primitive roles set project-level permissions and unless otherwise specified, they control access and management to all GCP services.



True



False

Submit

Provides all viewer permissions, plus permissions for actions that modify state, such as changing existing resources.



Editor role



GCP project



Viewer role



Owner role

Submit

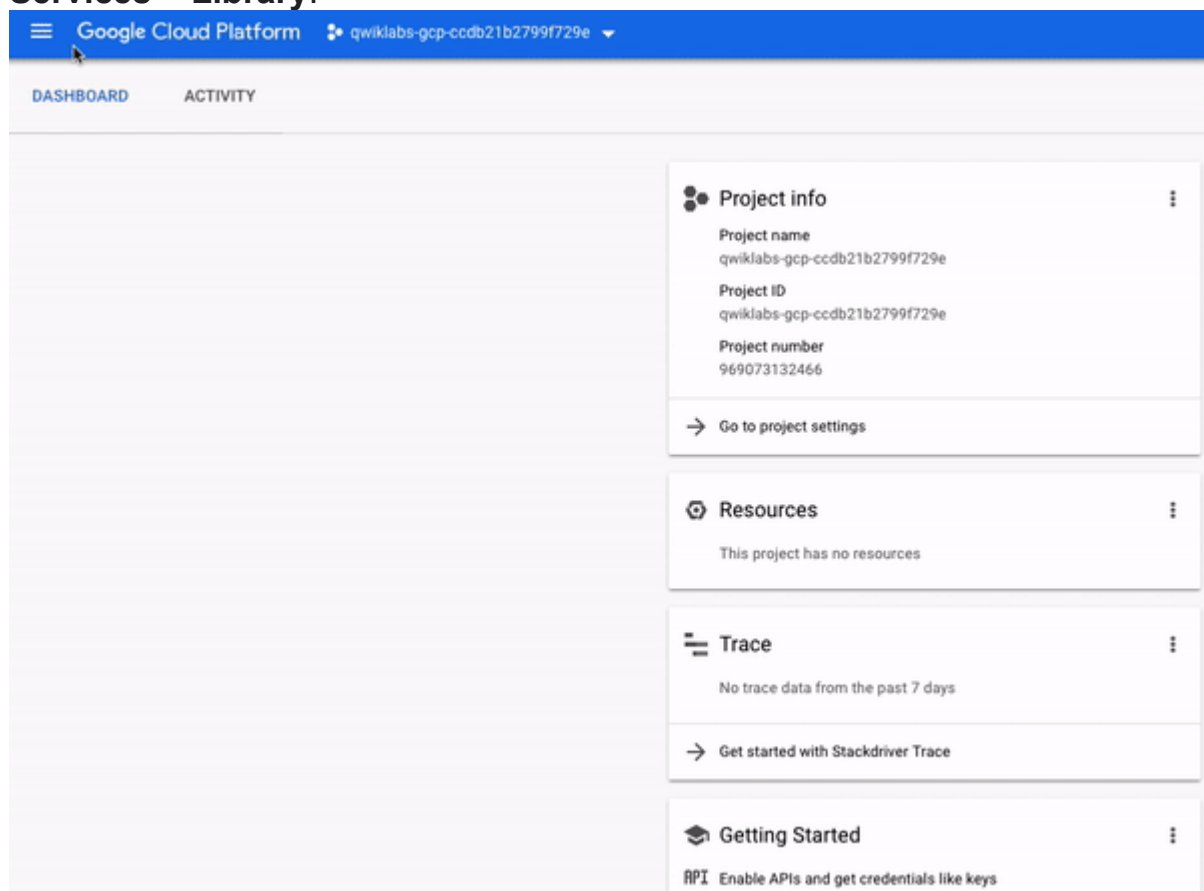
APIs and Services

Google Cloud APIs are a key part of the Google Cloud Platform. Like services, the 200+ APIs in areas that range from business administration to machine learning all easily integrate with GCP projects and applications.

APIs are "Application Programming Interfaces" that you can call directly or via our client libraries. Cloud APIs use resource-oriented design principles as described in our [Google API Design Guide](#).

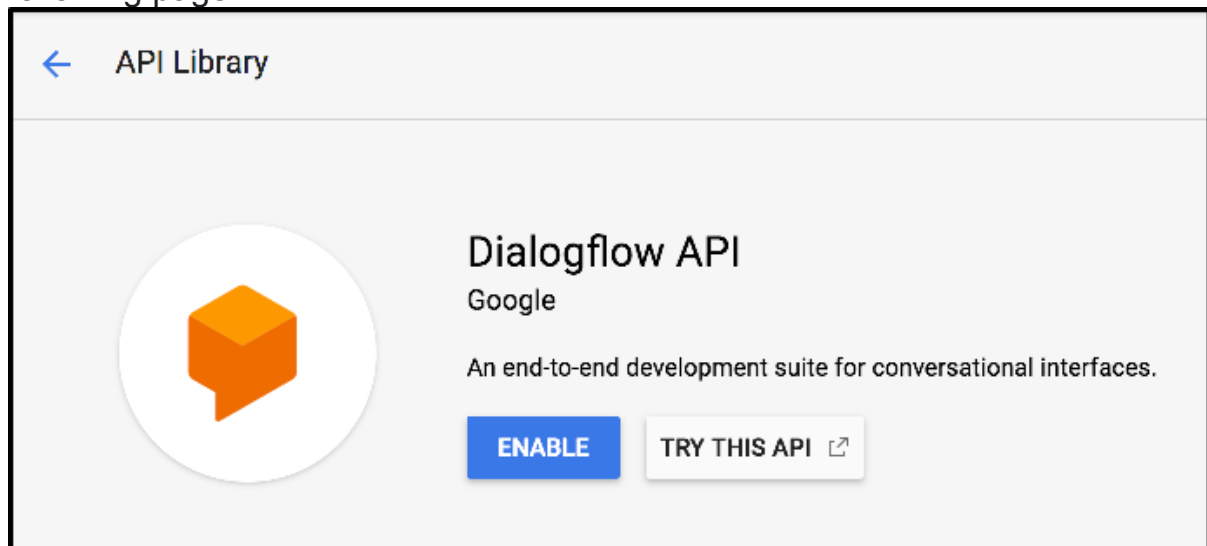
When Qwiklabs provisions a new GCP Project for a lab instance, it enables most APIs behind the scenes so you can work on the lab's tasks right away. Be aware that when you create your own GCP projects outside of Qwiklabs, you will have to enable certain APIs yourself.

Most Cloud APIs provide you with detailed information on your project's usage of that API, including traffic levels, error rates, and even latencies, helping you to quickly triage problems with applications that use Google services. You can view this information by opening the navigation menu and clicking on **APIs & Services > Library**:



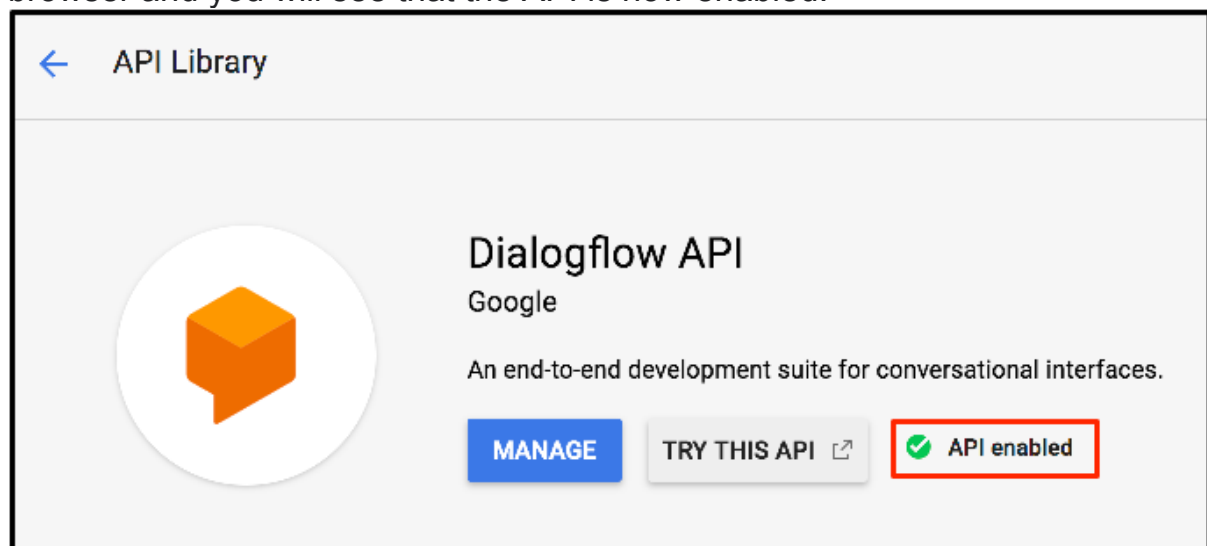
If you take a look at the left-hand menu with the header "CATEGORY", you will see all the different type of categories offered. In the API search bar, type

in Dialogflow and select the **Dialogflow API**. You should now be on the following page:



The Dialogflow API allows you to build rich conversational applications (e.g. for Google Assistant) without having to worry about the underlying machine learning and natural language understanding schema.

Now click **Enable**. This will take you to a new page, hit the back button in your browser and you will see that the API is now enabled:



Now click on **Try this API**. This will open a new tab that will reveal the documentation for the Dialogflow API and will specify the methods available to you. Examine some of these and close the tab when you're finished. If you're interested in learning more about APIs, see the new hands-on tool in Google Cloud called [APIs Explorer](#). We've also created a lab [APIs Explorer: Qwik Start](#) that will give you hands-on experience with the tool using a simple example.

Open the navigation menu and click **Home** to go back to main page of the GCP console.

Test your understanding

Answer the following multiple choice question to reinforce your understanding of the concepts we've covered so far.

When you start a lab in Qwiklabs, you need to enable APIs in your GCP project to start working with GCP.



True



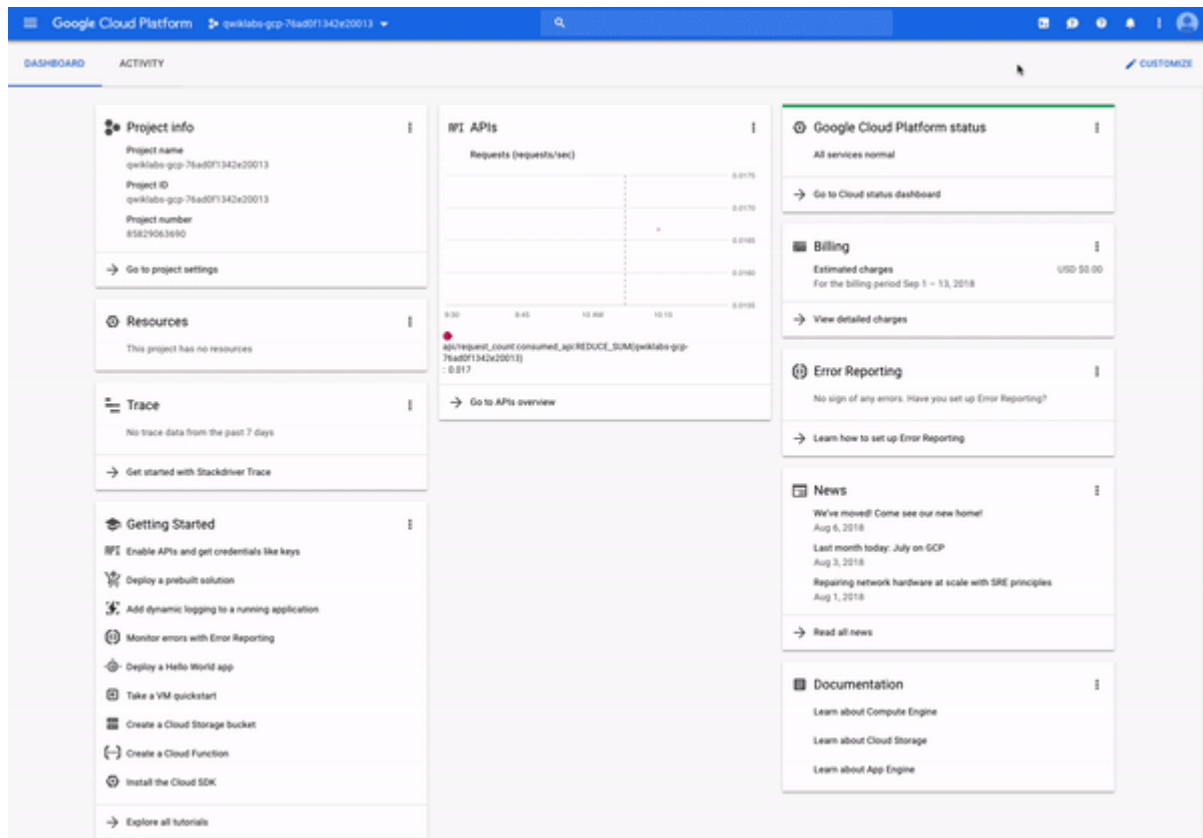
False

Submit

Cloud Shell

Now that you understand the key features of GCP and the console, you will get hands-on practice with [Cloud Shell](#). Cloud Shell is an in-browser command prompt execution environment that allows you to enter commands at a terminal prompt to manage resources and services in your GCP project. Cloud Shell lets you run all of your shell commands without leaving the console and comes with pre-installed command line tools.

In the top-right corner of the console, click on the **Activate Cloud Shell** button and then click **Start Cloud Shell** if prompted:



A new, black window should appear at the bottom of the console with messages and prompts that resemble the following:

```
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to qwiklabs-gcp-
76ad0f1342e20013.
Use "gcloud config set project [PROJECT_ID]" to change to a different
project.
gcpstaging23396_student@cloudshell:~ (quiklabs-gcp-76ad0f1342e20013)$
```

You now have a Cloud Shell session up and running. Copy and paste (or type in) the following into Cloud Shell and hit enter:

```
gcloud auth list
```

You should receive see a similar output, where `ACTIVE ACCOUNT` is set to your GCP IAM identity (`gcpstagingxxxxx_student@quiklabs.net`):

```
Credentialed Accounts
ACTIVE ACCOUNT
*      gcpstaging23396_student@quiklabs.net
To set the active account, run:
$ gcloud config set account `ACCOUNT`
```

As mentioned earlier, Cloud Shell comes preinstalled with specific command line tools. The main GCP toolkit is [gcloud](#), which is used for many tasks on the platform, like resource management and user authentication.

You just ran a `gcloud` command—[auth list](#)—which lists the credentialed account(s) in your GCP project. This account name matches the Qwiklabs username you signed in to the console with earlier.

Besides pre-installed toolkits, Cloud Shell also comes with the standard unix command line interface (CLI) tools and text editors like [nano](#). We can use these to create and edit files right inside Cloud Shell.

Run the following [touch command](#) to create a file called `test.txt`:

```
touch test.txt
```

`touch` won't generate any output. Run the unix command [ls](#) to list the files in our current directory:

```
ls
```

You should receive the following output:

```
README-cloudshell.txt  test.txt
```

We see our new `test.txt` file added to our working directory. Let's make some edits to it using the nano text editor that we referenced earlier. To edit a file, type in `nano` followed by the filename you want to edit in Cloud Shell:

```
nano test.txt
```

This will open the blank file with the Nano text editor:



```
GNU nano 2.7.4
```

```
^G Get Help    ^O Write Out   ^W Where Is
^X Exit        ^R Read File   ^\ Replace
```

Go ahead and type in a message, like the following:

```
GCP and Qwiklabs are the best!
```

Once you have that written in, hold the **CNTRL+X** keys. Then type in **Y** followed by the **Enter key** to save the file with the new message. Another helpful command is [cat](#), which will output the contents of a file. Run the following command to make sure that our file was updated correctly:

```
cat test.txt
```

You should see the following output in your Cloud Shell session:


```
GCP and Qwiklabs are the best!
```

And just like that, you were able to create, edit, and output the contents of a file all in Cloud Shell (all without having to leave your browser.)

Ending your lab

Now that you're finished with the lab, click on the **End Lab** button at the top of this page.

The screenshot shows the Google Cloud Self-Paced Labs interface. At the top, there is a blue header bar with a user profile icon. Below it, a status bar indicates 'Lab Running' with a green icon, a red 'End Lab' button, 'Free' with an information icon, and a timer showing '00:29:28'. The main content area has a large title 'Creating a Virtual Machine' and a sub-label 'GSP001'. To the right, a sidebar lists the lab steps: 'Overview', 'Setup', 'Create a new instance from the Cloud Console', 'Create a new instance with gcloud', 'Test your knowledge', and 'Congratulations!'. A green box in the top right corner shows the 'Score 0/15'. At the bottom, the Google Cloud logo and the text 'Google Cloud Self-Paced Labs' are visible.

Please rate each lab you take in Qwiklabs. Rate it  if you were satisfied—anything less if you weren't. Leave comments about your experiences in the "Comment" window; Google always appreciates thoughtful feedback.

Ending a lab will remove your access to the GCP project as well as the services and work you've done in it. If you go back to the GCP console, you will see that you've been signed out automatically. You can close that tab now.

Congratulations!



In just 30 minutes you developed a solid understanding of the GCP console and the platform's key features. You learned about projects, roles, and the types of services the platform offers. You also practiced with Cloud IAM, the APIs library, and Cloud Shell. You are now ready to take more GCP labs in Qwiklabs.

Finish Your Quest

Continue your Quest with [GCP Essentials](#). A Quest is a series of related labs that form a learning path. Completing this Quest earns you the badge above, 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 if you've taken this lab. [See other available Qwiklabs Quests](#).

Next Steps / Learn More

Be sure to check out the following labs for more practice with essential GCP services:

- [Creating a Virtual Machine](#)
- [Getting Started with Cloud Shell and gcloud](#)

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 November 18, 2019

Lab Last Tested July 15, 2019

Copyright 2020 Google LLC All rights reserved. Google and the Google logo are trademarks of Google LLC. All other company and product names may be trademarks of the respective companies with which they are associated.

Creating a Virtual Machine

40 minutes1 Credit

Rate Lab

GSP001



Google Cloud Self-Paced Labs

Overview

Google Compute Engine lets you create virtual machines running 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 has been designed to be fast and to offer strong consistency of performance.

In this hands-on lab you'll learn how to create virtual machine instances of various machine types using the Google Cloud Platform (GCP) Console and using 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, students should type the commands themselves to reinforce their understanding of the core concepts

What you'll do

- Create a virtual machine with the GCP Console
- Create a virtual machine with `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 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 the Google Cloud Platform for the duration of the lab.

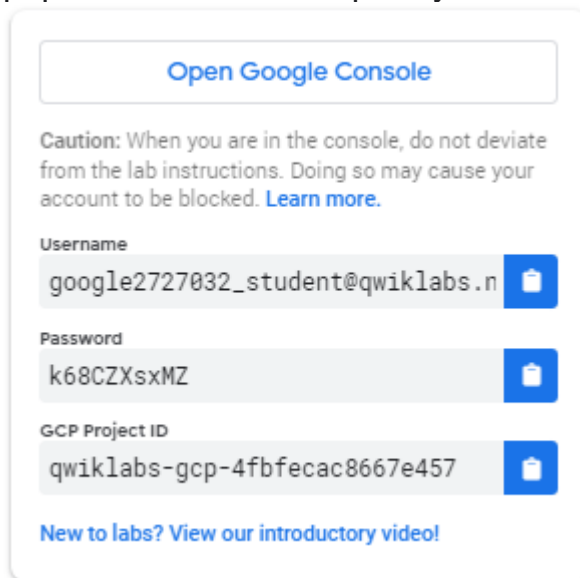
What you need

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 GCP account or project, do not use it for this lab.
- Note:** If you are using a Pixelbook please open an Incognito window to run this lab.

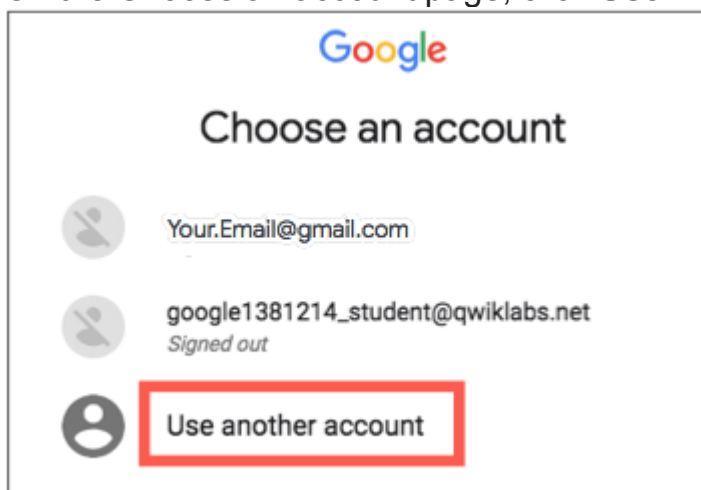
How to start your lab and sign in to the Console

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.



A screenshot of a web panel titled "Open Google Console". At the top is a button labeled "Open Google Console". Below it is a caution message: "Caution: When you are in the console, do not deviate from the lab instructions. Doing so may cause your account to be blocked. [Learn more.](#)". The panel contains three input fields, each with a copy icon to its right: "Username" with the value "google2727032_student@qwiklabs.n", "Password" with the value "k68CZXsxMZ", and "GCP Project ID" with the value "qwiklabs-gcp-4fbfecac8667e457". At the bottom is a link: "New to labs? View our introductory video!"

2. Copy the username, and then click **Open Google Console**. The lab spins up resources, and then opens another tab that shows the **Choose an account** page.
Tip: Open the tabs in separate windows, side-by-side.
3. On the Choose an account page, click **Use Another Account**.



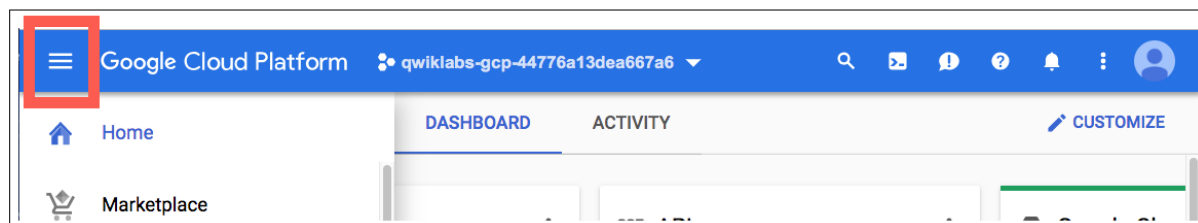
4. The Sign in page opens. Paste the username that you copied from the Connection Details panel. Then copy and paste the password.

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

5. 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 GCP console opens in this tab.

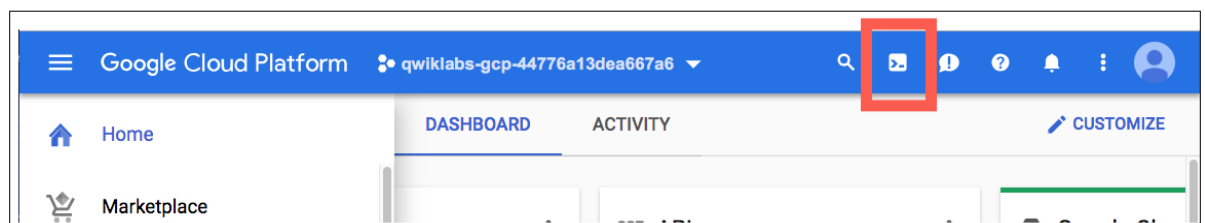
Note: You can view the menu with a list of GCP Products and Services by clicking the **Navigation menu** at the top-left, next to “Google Cloud Platform”.



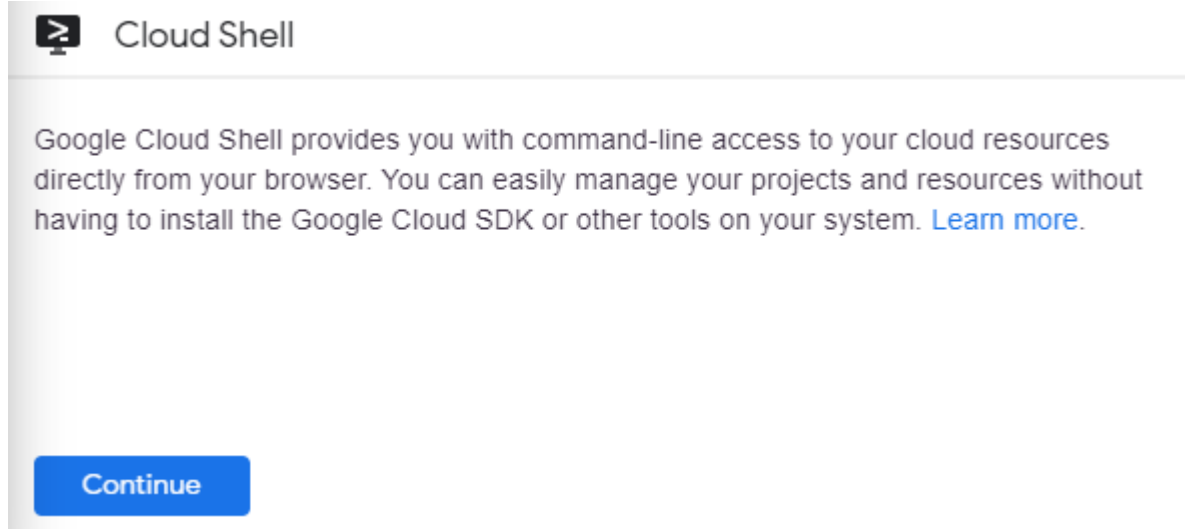
Activate Google Cloud Shell

Google 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. Google Cloud Shell provides command-line access to your GCP resources.

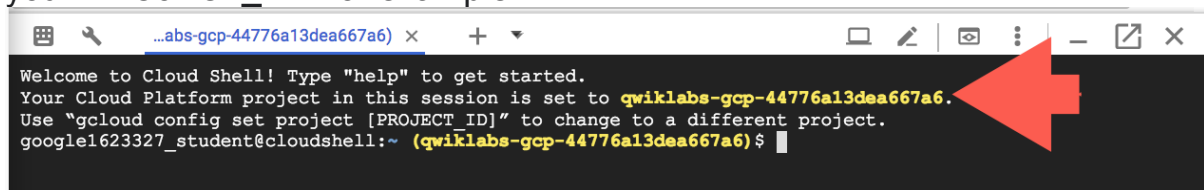
1. In GCP console, on the top right toolbar, click the Open Cloud Shell button.



2. 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:

A screenshot of a web browser window showing the Cloud Shell terminal. The browser's address bar shows a URL with a project ID. The terminal text reads: "Welcome to Cloud Shell! Type 'help' to get started. Your Cloud Platform project in this session is set to qwiklabs-gcp-44776a13dea667a6. Use 'gcloud config set project [PROJECT_ID]' to change to a different project. google1623327_student@cloudshell:~ (quiklabs-gcp-44776a13dea667a6) \$". A large red arrow points to the project ID in the terminal output.

```
...abs-gcp-44776a13dea667a6) x + v
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to qwiklabs-gcp-44776a13dea667a6.
Use "gcloud config set project [PROJECT_ID]" to change to a different project.
google1623327_student@cloudshell:~ (quiklabs-gcp-44776a13dea667a6) $
```

My login details:

Open Google Console

Caution: When you are in the console, do not deviate from the lab instructions. Doing so may cause your account to be blocked. [Learn more.](#)

Username

student-03-8952e4bd582f@quwiklabs

Password

Z9Rrs5q6nV

GCP Project ID

quwiklabs-gcp-03-d3dc11f929c5

gcloud is the command-line tool for Google Cloud Platform. 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
```

MY OUTPUT:

```
student_03_8952e4bd582f@cloudshell:~$ gcloud auth list
Credentialed Accounts
ACTIVE  ACCOUNT
*       student-03-8952e4bd582f@qwiklabs.net

To set the active account, run:
$ gcloud config set account `ACCOUNT`
```

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
```

my output:

```
student_03_8952e4bd582f@cloudshell:~$ gcloud config list project
[core]
project = qwiklabs-gcp-03-d3dc11f929c5

Your active configuration is: [cloudshell-16401]
```

Full documentation of **gcloud** is available on [Google Cloud gcloud 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`.



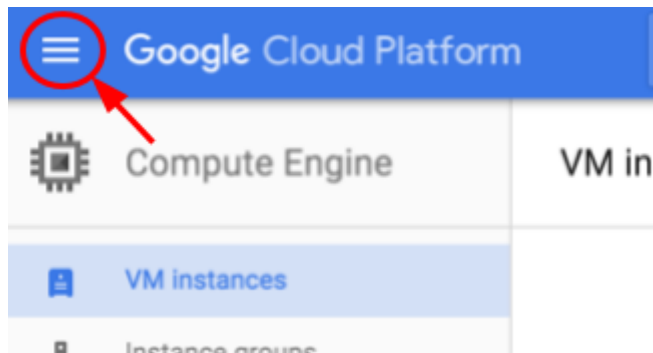
Resources that live in a zone are referred to as zonal resources. Virtual machine Instances and persistent disks live in a zone. To attach a persistent disk to a virtual machine instance, both resources must be in the same zone. Similarly, if you want to assign a static IP address to an instance, the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

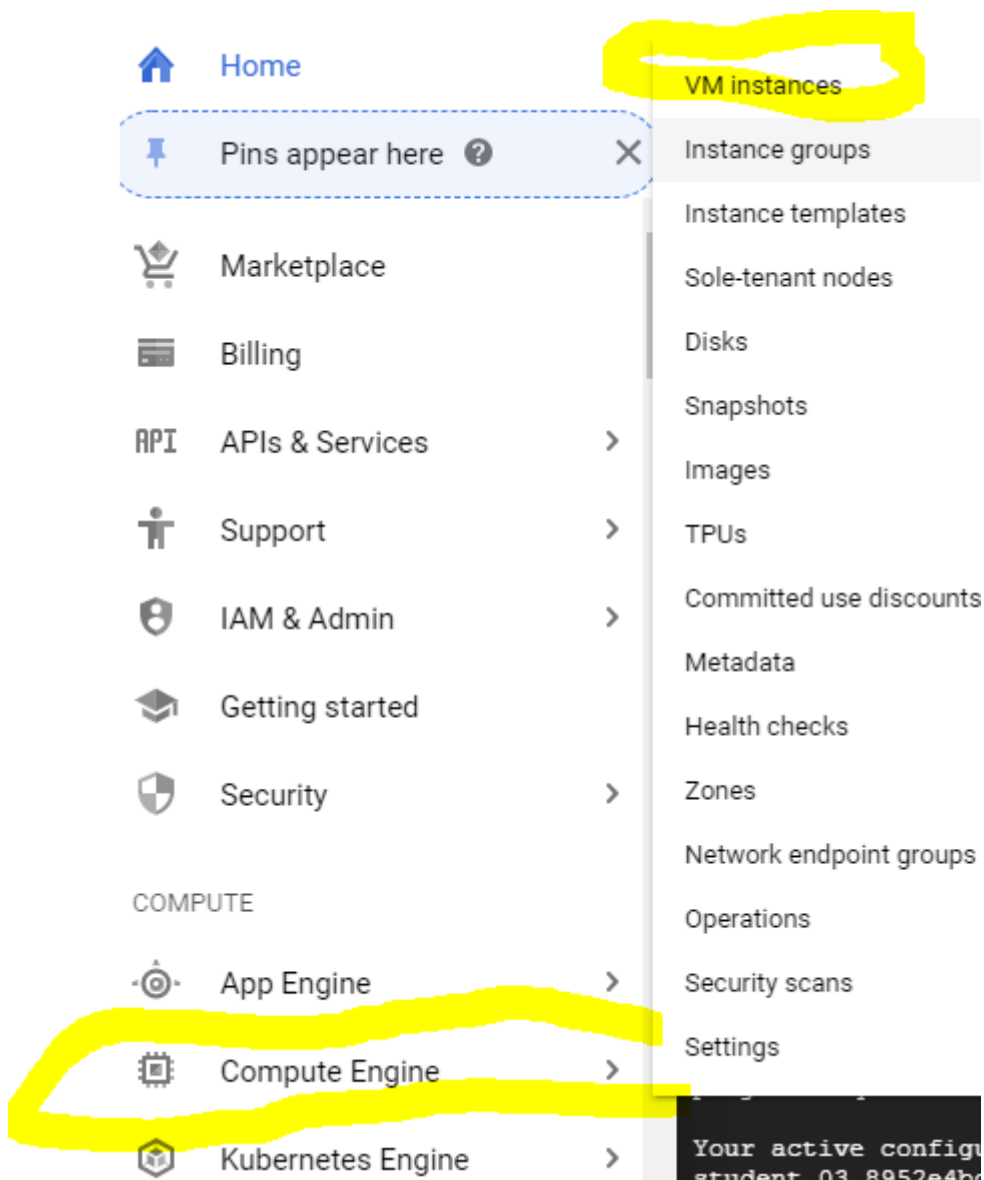
Create a new instance from the Cloud Console

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

In the GCP Console, on the top left of the screen, select **Navigation menu > Compute Engine > VM Instances**:



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



To create a new instance, click **Create**.

Compute Engine VM instances

Compute Engine lets you use virtual machines that run on Google's infrastructure. You can choose from micro-VMs to large instances running Debian, Windows, or other standard images. Create your first VM instance, import it by CloudEndure migration service or try the quickstart to build a sample app.

Create

or

Import

or

Take the quickstart

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) or asia-south1 (Mumbai)	Learn more about regions in Regions & Zones documentation .
Zone	us-central1-c or asia-south1-c Note: remember the zone that you selected, you'll need it later.	Learn more about zones in Regions & Zones documentation .
Machine Type	2 vCPUs This is a (n1-standard-2), 2-CPU, 7.5GB RAM instance. There are a number of machine types, ranging from micro instance types to 32-core/208GB RAM instance types. Learn more in the Machine Types documentation .	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 of this lab.

Boot Disk	New 10 GB standard persistent disk OS Image: Debian GNU/Linux 9 (stretch)	There are a number of images to choose from, including: Debian, Ubuntu, CoreOS as well as premium images such as Red Hat Enterprise Linux and Windows Server. See Operating System documentation for more detail.
Firewall	Check Allow HTTP traffic Check this option so to access a webserver that you'll install later.	Note: This will automatically create firewall rule to allow HTTP traffic on port 80.

Click **Create**.

Wait for it to finish - it shouldn't take more than a minute.

Machine types for common workloads, optimized for cost and flexibility


Series

N1

Powered by Intel Skylake CPU platform or one of its predecessors

Machine type

n1-standard-2 (2 vCPU, 7.5 GB memory)

 vCPU 2 Memory 7.5 GB

⌵ CPU platform and GPU


Container ?

☐ Deploy a container image to this VM instance. [Learn more](#)

Boot disk ?

Once finished, you should see the new virtual machine in the **VM Instances** page.

To SSH into the virtual machine, click on **SSH** on the right hand side. This launches a SSH client directly from your browser.

<input type="checkbox"/> Name ^	Zone	Recommendation	Internal IP	External IP	Connect
<input type="checkbox"/>  gcelab	us-central1-c		10.128.0.2 (nic0)	35.194.40.76 ↗	SSH ▾

Note: For more information, see the [Connect to an instance using ssh documentation](#).

Install a NGINX web server

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

Once SSH'ed, get `root` access using `sudo`:

```
sudo su -
```

As the `root` user, update your OS:

```
apt-get update
```

(Output)

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

Install NGINX:

```
apt-get install nginx -y
```

(Output)

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

Check that NGINX is running:

```
ps auxx | grep nginx
```

(Output)

```
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
```

Awesome! To see the web page, go to the Cloud Console and click the `External IP` link of the virtual machine instance. You can also see the web page by adding the `External IP` to `http://EXTERNAL_IP/` in a new browser window or tab.

VM instances

 CREATE INSTANCE



 IMPORT VM

 REFRESH

 START

Filter by label or name

Columns ▾

<input checked="" type="checkbox"/> Name ^	Zone	Recommendation	Internal IP	External IP	Connect
<input checked="" type="checkbox"/>  gcelab	us-central1-c		10.128.0.2	104.154.165.45 	SSH ▾ ⋮

You should see this default web page:

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

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

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

Check my progress

Create a new instance with gcloud

Rather than using the GCP 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 5GB home directory.

If you want to try this on your own machine in the future, read the [gcloud command line tool guide](#).

In the Cloud Shell, create a new virtual machine instance from the command line using `gcloud`, replacing `[YOUR_ZONE]` with one of the zone choices given earlier:

```
gcloud compute instances create gcelab2 --machine-type n1-standard-2 --zone [your_zone]
```

(Output)

```
Created [...gcelab2].
```

NAME	ZONE	MACHINE_TYPE	...	STATUS
gcelab2	us-central1-c	n1-standard-2	...	RUNNING

Click **Check my progress** below to verify you're on track in this lab.

Create a new instance with gcloud.

Check my progress

The instance created has these default values:

- The latest [Debian 9 \(stretch\)](#) image.
- The `n1-standard-2` [machine type](#). In this lab you can select one of these other machine types if you'd like: `n1-highmem-4` or `n1-highcpu-4`. When you're working on a project outside of 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.

Run `gcloud compute instances create --help` to see all the defaults.

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. Do this by running these commands :

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

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

To exit help, press **Ctrl+c**.

Check out your instances. Select **Navigation menu > Compute Engine > VM instances**. You should see the 2 instances you created in this lab.

VM instances							SHOW INFO PANEL	
Filter VM instances							Columns	
<input type="checkbox"/> Name ^	Zone	Recommendation	Internal IP	External IP	Connect			
<input type="checkbox"/> gcelab	us-central1-c		10.128.0.2 (nic0)	104.155.128.223 ↗	SSH	▼	⋮	
<input type="checkbox"/> gcelab2	us-central1-c		10.128.0.3 (nic0)	104.154.152.174	SSH	▼	⋮	

Finally, you can SSH into your instance using `gcloud` as well. Make sure you add your zone, or omit the `--zone` flag if you've set the option globally:

```
gcloud compute ssh gcelab2 --zone [YOUR_ZONE]
```

(Output)

```
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.
```

Now you'll type **Y** to continue.

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

Enter through the passphrase section to leave the passphrase empty.

```
Generating public/private rsa key pair.
```

```
Enter passphrase (empty for no passphrase)
```

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

exit

Test your knowledge

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

Through which of the following ways you can create a VM instance in Google Compute Engine(GCE)?

☐ Through web console.

☐ The gcloud command line tool.

Submit

Congratulations!

Google Compute Engine is the foundation to GCP'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 GCP.



Finish Your Quest

This self-paced lab is part of the Qwiklab [GCP 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 if you've taken this lab. [See other available Qwiklabs Quests](#).

Take Your Next Lab

Continue your Quest with [Getting Started with Cloud Shell & gcloud](#), or check out these suggestions:

- [Getting Started with Cloud Shell & gcloud](#)
- [Provision Services with GCP Marketplace](#)

Next Steps / Learn More

- For an overview of VMs, see [Virtual Machine Instances](#).
- Check out how to [migrate VMs to the GCP](#).
- 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: July 5, 2019

Lab Last Tested: July 5, 2019

Copyright 2020 Google LLC All rights reserved. Google and the Google logo are trademarks of Google LLC. All other company and product names may be trademarks of the respective companies with which they are associated.

Compute Engine: Qwik Start - Windows

40 minutes1 Credit

Rate Lab

GSP093



Google Cloud Self-Paced Labs

Google Compute Engine lets you create and run virtual machines on Google infrastructure. Compute Engine offers scale, performance, and value that allows you to easily launch large compute clusters on Google's infrastructure.

You can run your Windows applications on Google Compute Engine and take advantage of many benefits available to virtual machine instances such as reliable [storage options](#), the speed of the [Google network](#), and [Autoscaling](#). In this hands-on lab, you will learn how to launch a Windows Server instance in Google Compute Engine, and connect to it using the Remote Desktop Protocol.

If you aren't using Windows on your local machine, install a third-party RDP client such as [Chrome RDP](#) by FusionLabs.

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 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 the Google Cloud Platform 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 GCP account or project, do not use it for this lab.

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

How to start your lab and sign in to the 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.

Open Google Console

Caution: When you are in the console, do not deviate from the lab instructions. Doing so may cause your account to be blocked. [Learn more.](#)

Username

google2727032_student@qwiklabs.n

Password

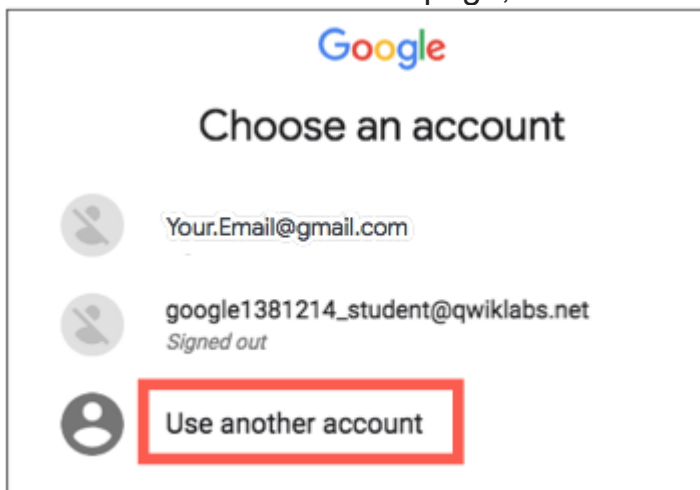
k68CZxsxMZ

GCP Project ID

qwiklabs-gcp-4fbfecac8667e457

New to labs? [View our introductory video!](#)

- Copy the username, and then click **Open Google Console**. The lab spins up resources, and then opens another tab that shows the **Choose an account** page.
Tip: Open the tabs in separate windows, side-by-side.
- On the Choose an account page, click **Use Another Account**.



- The Sign in page opens. Paste the username that you copied from the Connection Details panel. Then copy and paste the password.

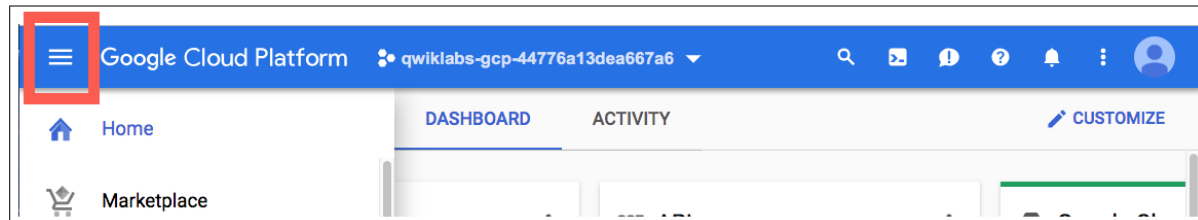
Important: You must use the credentials from the Connection Details panel. Do not use your Qwiklabs credentials. If you have your own GCP account, do not use it for this lab (avoids incurring charges).

- 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.

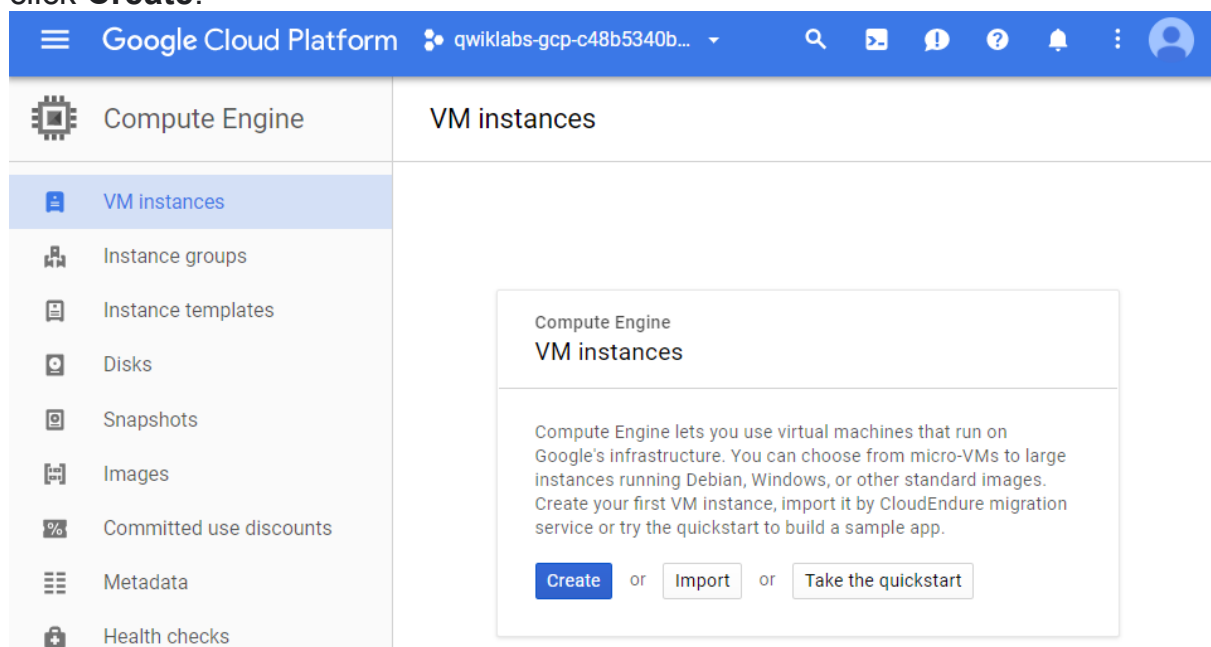
After a few moments, the GCP console opens in this tab.

Note: You can view the menu with a list of GCP Products and Services by clicking the **Navigation menu** at the top-left, next to “Google Cloud Platform”.



Create a virtual machine instance

In the GCP Console, go to **Compute Engine > VM instances**, then click **Create**.



In the **Boot disk** section, click **Change** to begin configuring your boot disk.



Choose **Windows Server 2012 R2 Datacenter**, then **Select**. Leave all other settings at their defaults.



Create an instance



Name ?

instance-1

Zone ?

us-central1-a

Machine type

Customize to select cores, memory and GPUs.

1 vCPU

3.75 GB memory

[Customize](#)

Container ?

☐ Deploy a container image to this VM instance. [Learn more](#)

Boot disk ?

New 50 GB standard persistent disk
Image

Windows Server 2012 R2 Datacenter

[Change](#)

If you are using Windows and intend to run additional Microsoft software, please fill out the [License Verification Form](#)

[Learn more](#) about Microsoft license mobility requirements

Identity and API access ?

Service account ?

Compute Engine default service account

Access scopes ?

- ☒ Allow default access
- ☐ Allow full access to all Cloud APIs
- ☐ Set access for each API

Firewall ?

Add tags and firewall rules to allow specific network traffic from the Internet

- ☐ Allow HTTP traffic
- ☐ Allow HTTPS traffic

⌵ [Management, disks, networking, SSH keys](#)

You will be billed for this instance. [Learn more](#)

[Create](#)[Cancel](#)

Equivalent [REST](#) or [command line](#)

Click the **Create** button to create the instance.

Test Completed Task

Click **Check my progress** to verify your performed task.

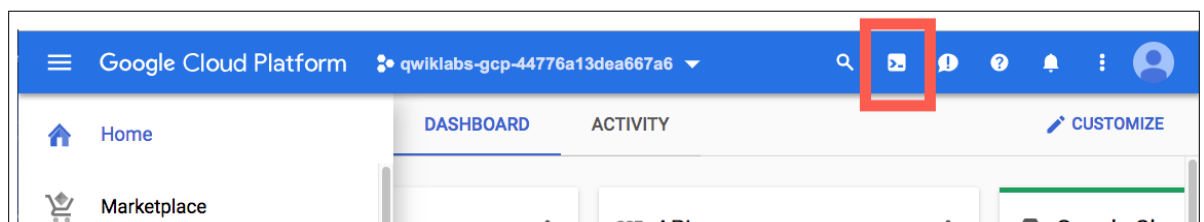
Create a virtual machine instance (zone: us-central1-a).

Check my progress

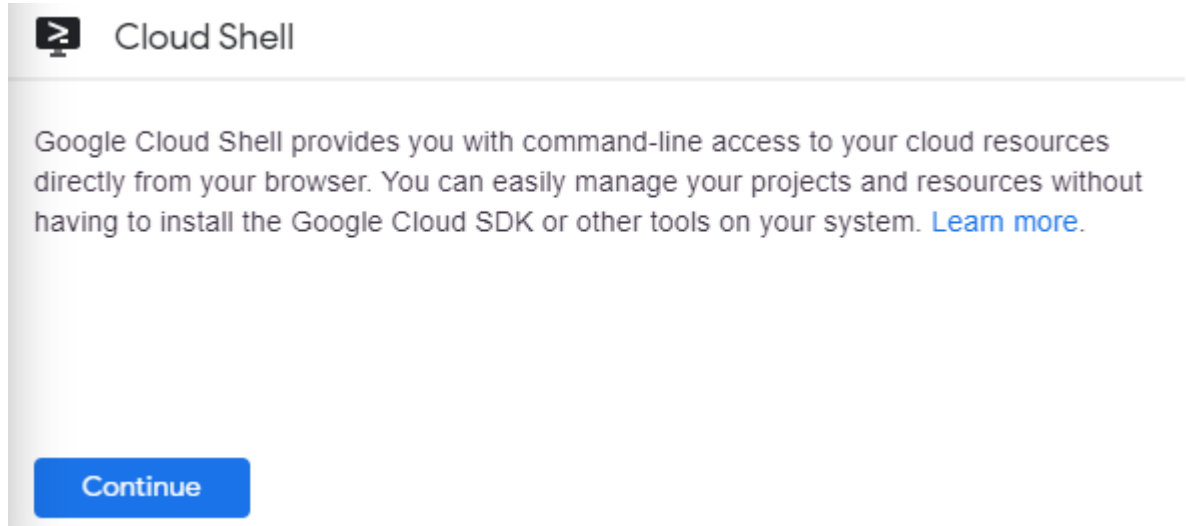
Activate Google Cloud Shell

Google 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. Google Cloud Shell provides command-line access to your GCP resources.

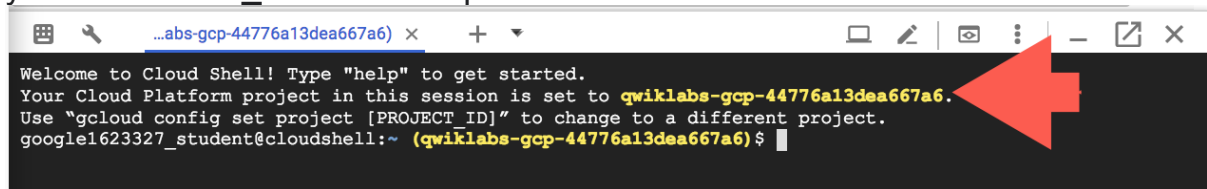
1. In GCP console, on the top right toolbar, click the Open Cloud Shell button.



2. 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:



```
...abs-gcp-44776a13dea667a6) x + v
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to qwiklabs-gcp-44776a13dea667a6.
Use "gcloud config set project [PROJECT ID]" to change to a different project.
google1623327_student@cloudshell:~ (qwiklabs-gcp-44776a13dea667a6) $
```

gcloud is the command-line tool for Google Cloud Platform. 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 = quiklabs-gcp-44776a13dea667a6
```

Full documentation of **gcloud** is available on [Google Cloud gcloud Overview](#).

Test the status of Windows Startup

Allow a short time for the Windows Server instance to start up. Once it has been provisioned, it will be listed on the VM Instances page with a green status icon.

However it may not yet be ready to accept RDP connections, as it takes a while for all the OS components to initialize.

To see whether the server is ready for an RDP connection, run the following command at your Cloud Shell terminal command line:

```
gcloud compute instances get-serial-port-output instance-1 --zone us-central1-a
```

Repeat the command until you see the following in the command output, which tells you that the OS components have initialized and the Windows Server is ready to accept your RDP connection (attempt in the next step).

```
Finished running startup scripts.
```

Connect to your instance

Click the name of your virtual machine:

VM instances						
<div>CREATE INSTANCE</div> <div>IMPORT VM</div> <div>REFRESH</div> <div>START</div> <div>STOP</div>						
<div>Filter VM instances</div>						
<input type="checkbox"/>	Name ^	Zone	Recommendation	Internal IP	External IP	Connect
<input type="checkbox"/>	instance-1	us-central1-a		10.128.0.2	35.225.110.176	RDP

Under the **Remote Access** section, click the **Set Windows Password** button.

Remote access

RDP

Set Windows password

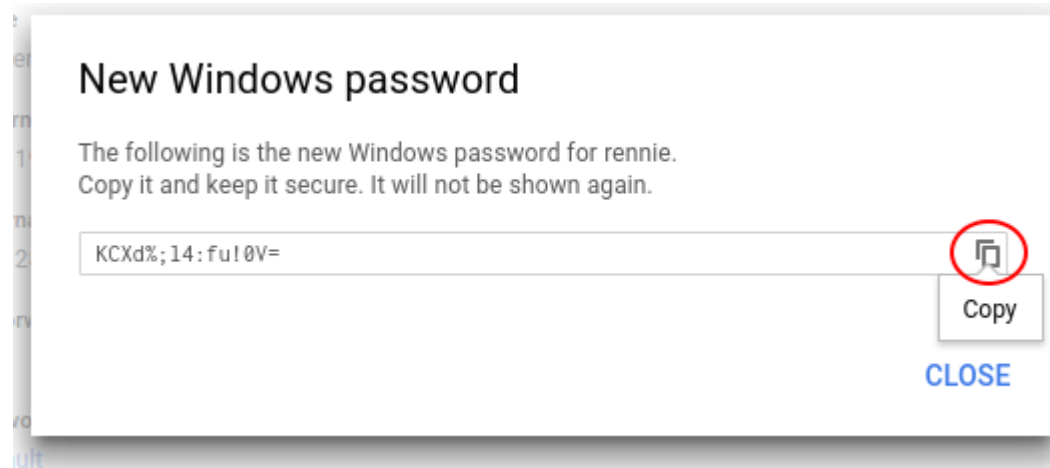
Connect to serial console

☐ Enable connecting to serial ports

A username will be generated.

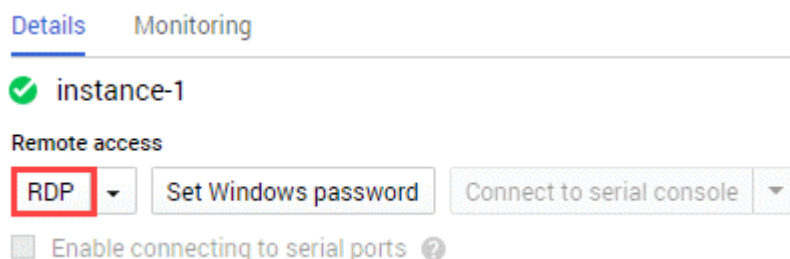
Click **Set** to generate a password for this Windows instance. This may take several minutes to complete.

Copy the password and save it so you can log into the instance.



Remote desktop (RDP) into the Windows Server

It's time to RDP into the Windows Server. You can RDP directly from the browser using the [Chrome RDP for Google Cloud Platform](#) extension. Click on **RDP** to connect.



This prompts you to install the RDP Extension. Once installed, GCP opens up a login page where you use your Windows user and password to log in. Paste in the password you saved earlier.

Enter your credentials

These credentials will be used to connect to 35.202.109.52

Domain:

Username:

Password:

Click **Continue** to confirm you want to connect.

The server certificate returned for 35.193.73.22:3389 cannot be verified. Do you want to connect anyway?

This problem can occur if the remote computer is running a version of Windows that is earlier than Windows Vista, or if the remote computer is not configured to support server authentication.

When Server Manager opens you are connected to instance-1, the VM instance on the Windows Server.

Copy and pasting with the RDP client

Once you are securely logged in to your instance, you may find yourself copying and pasting commands from the lab manual.

To paste, hold the **CTRL-V** keys (if you are a Mac user, using CMND-V will not work.) If you are in a Powershell window, be sure that you have clicked in to the window or else the paste shortcut won't work.

If you are pasting into putty, **right click**.

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.

We can create Windows instance in GCP by changing its ____ in VM instance console.



Boot disk to Windows image



Machine Type



Firewall rules



API Access

Submit

Which command is used to check whether the server is ready for an RDP connection?



gcloud compute ssh



gcloud compute instances get-serial-port-output



gcloud compute instances list



gcloud compute instances create

Submit

Congratulations!

Getting Started with Cloud Shell & gcloud

45 minutes1 Credit

Rate Lab

GSP002



Google Cloud Self-Paced Labs

Overview

Google Cloud Shell provides you with `gcloud` command-line access to computing resources hosted on the Google Cloud Platform. Cloud Shell is a Debian-based virtual machine with a persistent 5GB home directory, which makes it easy for you to manage your GCP projects and resources. The Cloud SDK `gcloud` and other utilities you need come pre-installed in Cloud Shell, which allows you to get up and running quickly.

In this hands-on lab you will learn how to connect to computing resources hosted on the Google Cloud Platform via Cloud Shell with the `gcloud` command-line.

Students are encouraged to type the commands themselves, which reinforces the core concepts. Many labs will include a code block that contains the required commands. You can easily copy and paste the commands from the code block into the appropriate places during the lab.

What you'll do

- Practice using `gcloud` commands.
- Connect to compute services hosted on the Google Cloud Platform.

Prerequisites

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

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 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 the Google Cloud Platform 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 GCP account or project, do not use it for this lab.

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

How to start your lab and sign in to the 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.

Open Google Console

Caution: When you are in the console, do not deviate from the lab instructions. Doing so may cause your account to be blocked. [Learn more.](#)

Username
google2727032_student@qwiklabs.n

Password
k68CZXsxMZ

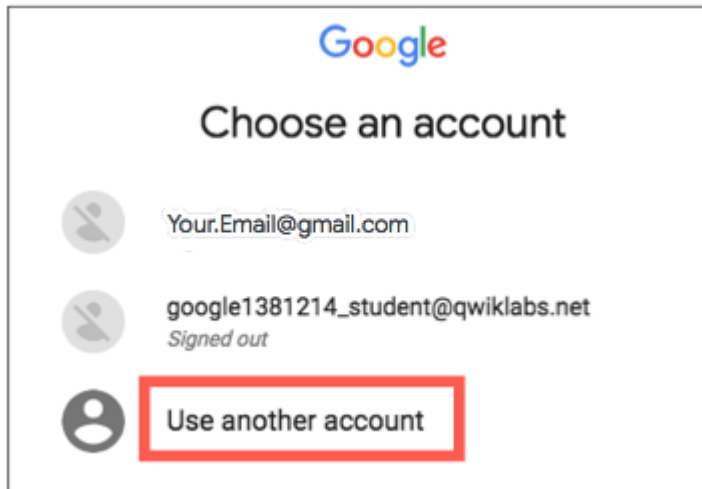
GCP Project ID
qwiklabs-gcp-4fbfecac8667e457

[New to labs? View our introductory video!](#)

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

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

3. On the Choose an account page, click **Use Another Account**.



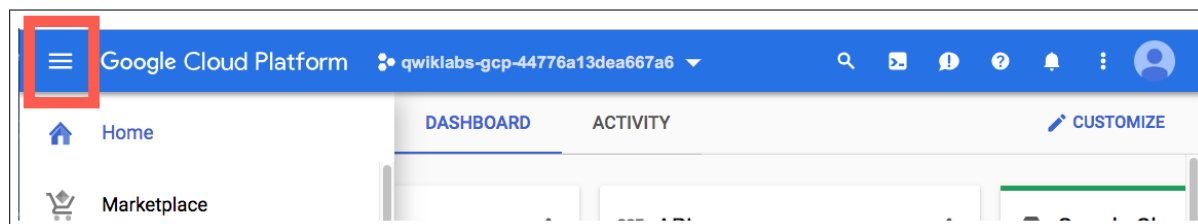
4. The Sign in page opens. Paste the username that you copied from the Connection Details panel. Then copy and paste the password.

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

5. 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.

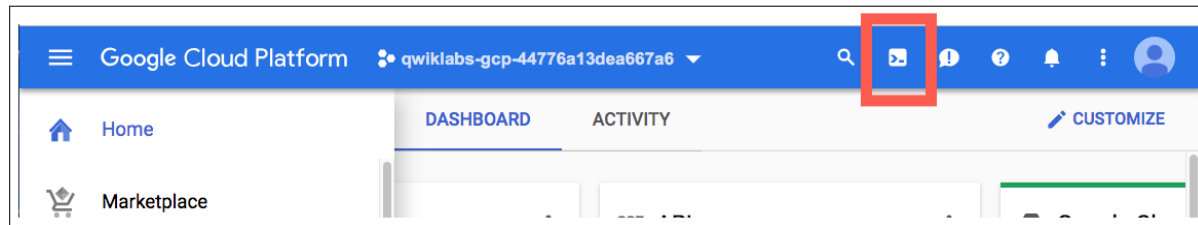
After a few moments, the GCP console opens in this tab.

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

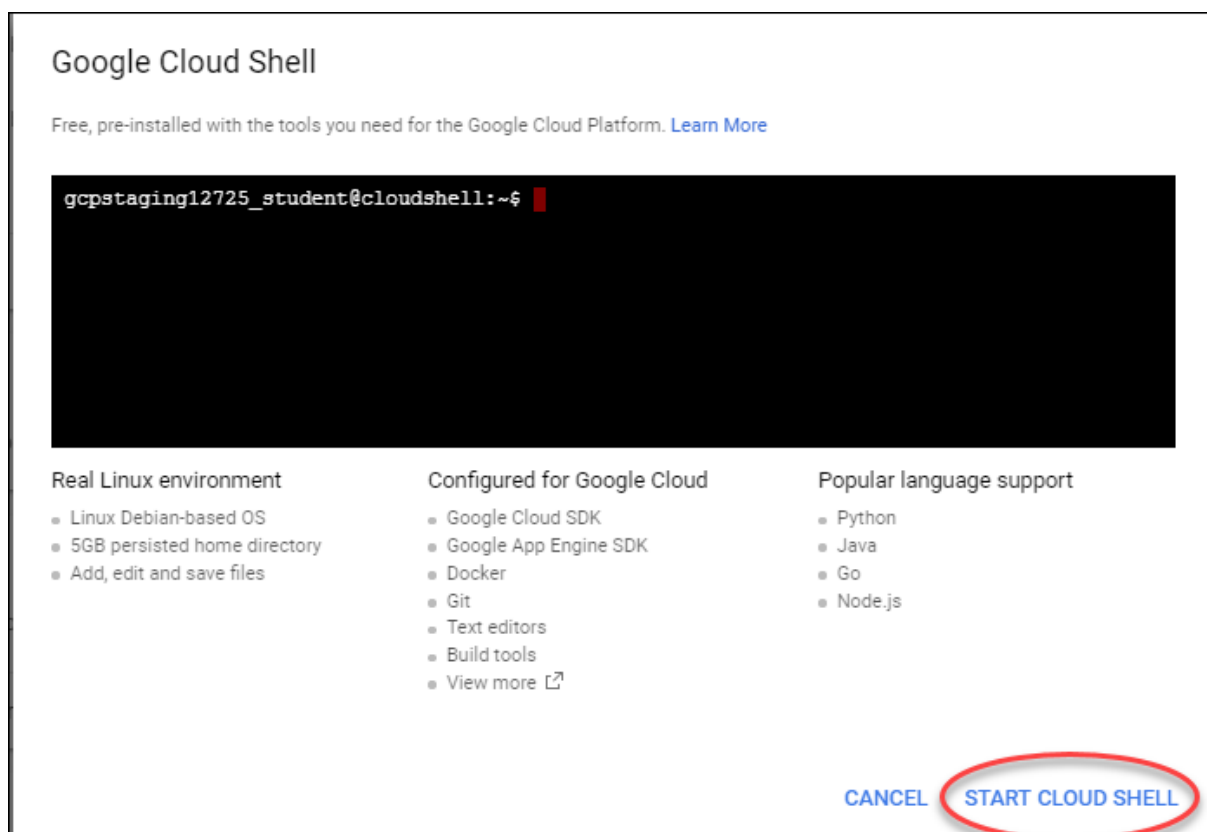


Start Cloud Shell

Open a Cloud Shell session by clicking on the icon in the top right corner of the GCP Console:



Then Start Cloud Shell:



After Cloud Shell is activated you can use the command line to invoke the Cloud SDK `gcloud` command or other tools available on the virtual machine instance. Later in the lab you will use your `$HOME` directory, which is used in persistent disk storage to store files across projects and between Cloud Shell sessions. Your `$HOME` directory is private to you and cannot be accessed by other users.

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](#).

Default regions and zones are set by using the following values:

```
google-compute-default-zone google-compute-default-region
```

To see what your default region and zone settings are, run the following `gcloud` command, replacing `<your_project_id>` which you can see on the Home page in the Console or look in the Qwiklabs tab where you started this lab, with your Project ID:

```
gcloud compute project-info describe --project <your_project_ID>
```

You'll use the zone (`google-compute-default-zone`) from the output later in this lab.

Look for the default zone and region metadata values in the response. If the `google-compute-default-region` and `google-compute-default-zone` keys and values are missing from the response, that means no default zone or region is set.

Initializing Cloud SDK

The `gcloud` CLI is a part of the Google Cloud SDK. You need to download and install the SDK on your own system and initialize it (by running `gcloud init`) before you can use the `gcloud` command-line tool.

The `gcloud` CLI is automatically available in Cloud Shell. Since you're using Cloud Shell for this lab, you don't need to install `gcloud` manually.

Setting environment variables

Environment variables are variables that define your environment. Define your own variables and save yourself time when writing scripts that contain APIs or executables.

Make a couple of environment variables:

```
export PROJECT_ID=<your_project_ID>
```

Set your `ZONE` environment variable (use the value for zone from the earlier command):

```
export ZONE=<your_zone>
```

Verify that your variables were set properly:

```
echo $PROJECT_ID
echo $ZONE
```

Create a virtual machine with `gcloud`

Create a new virtual machine instance using `gcloud`. In the following command you'll use:

- `gcloud compute` which enables you to easily manage your Google Compute Engine resources in a friendlier format than using the Compute Engine API.
- `instances create` creates a new instance.

Run the following to create your vm:

```
gcloud compute instances create gcelab2 --machine-type n1-standard-2 --zone $ZONE
```

- The name of the vm is "gcelab2",
- You're using the `--machine-type` flag to specify the machine type as "n1-standard-2"
- You're using the `--zone` flag to specify that it gets created in the zone you defined with your environment variable.

(Output)

NAME	ZONE	MACHINE TYPE	PREEMPTIBLE	INTERNAL IP	EXTERNAL IP	STATUS
gcelab2	us-central1-a	n1-standard-2		10.128.0.2	35.184.139.176	RUNNING

If you omit the `--zone` flag, `gcloud` can infer your desired zone based on your default properties. Other required instance settings, like `machine type` and `image`, if not specified in the create command, are set to default values.

Test Completed Task

Click **Check my progress** to verify your performed task. If you have successfully created a virtual machine with `gcloud`, you will see an assessment score.

Create a virtual machine with `gcloud`

Check my progress

You can see the default values by displaying help for the `create` command:

```
gcloud compute instances create --help
```

Using `gcloud` commands

`gcloud` offers simple usage guidelines that are available by adding the `-h` flag (for help) onto the end of any `gcloud` invocation.

Run the following command in Cloud Shell:


```
gcloud -h
```

More verbose help can be obtained by appending `--help` flag, or executing `gcloud help` command. Run the following in Cloud Shell:

```
gcloud config --help
```

Use the **Enter** key or the **Spacebar** to scroll through the help content. Type `q` to exit the content.

Now run the following command:

```
gcloud help config
```

You can see that the `gcloud config --help` and `gcloud help config` commands are equivalent. Both give long, detailed help. [gcloud Global Flags](#) govern the behavior of commands on a per-invocation level. Flags override any values set in SDK properties. View the list of configurations in your environment:

```
gcloud config list
```

To check how other properties are set, see all properties by calling:

```
gcloud config list --all
```

List your components:

```
gcloud components list
```

Here you will see what components are ready for you to use in this lab. Next you'll install a new component.

Auto-completion

`gcloud interactive` has auto prompting for commands and flags, and displays inline help snippets in the lower section as the command is typed. Static information, like command and sub-command names, and flag names and enumerated flag values, are auto-completed using dropdown menus.

Install the beta components:

```
gcloud components install beta
```

Enter the `gcloud interactive` mode:

```
gcloud beta interactive
```

When using the interactive mode, click on the **Tab** key to complete file path and resource arguments. If a dropdown menu appears, use the **Tab** key to move through the list, and the **Space bar** to select your choice.

Try it out! Start typing the following command, using auto-complete to finish the command:

```
gcloud compute instances describe <your_vm>
```

Across the bottom of Cloud Shell you can see the shortcut to toggle this feature. Try out the F2 toggle:

F2:help:STATE Toggles the active help section, ON when enabled, OFF when disabled.

SSH into your vm instance

`gcloud compute` makes connecting to your instances easy. The `gcloud compute ssh` command provides a wrapper around SSH, which takes care of authentication and the mapping of instance name to IP address.

Use `gcloud compute ssh` to SSH into your vm:

```
gcloud compute ssh gcelab2 --zone $ZONE
```

(Output)

```
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.
```

Type "Y" to continue:

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

Press the **Enter** key through the passphrase section to leave the passphrase empty.

```
Generating public/private rsa key pair.
```

```
Enter passphrase (empty for no passphrase)
```

You don't need to do anything here, so disconnect from SSH by exiting from the remote shell by typing "exit":

```
exit
```

You should be back at your project's command prompt.

Use the Home directory

Now try out your Home directory. The contents of your Cloud Shell Home directory persists across projects between all Cloud Shell sessions, even after the virtual machine terminates and is restarted.

Change your current working directory:

```
cd $HOME
```

Open your `.bashrc` configuration file using `vi` text editor:

```
vi ~/.bashrc
```

The editor opens and displays the contents of the file. Press the `ESC` key and then `:wq` to exit the editor.

Test your Understanding

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

Three basic ways to interact with the GCP services and resources:

☐ Command-line interface

☐ GCP Console

☐ GStreamer

☐ GLib

☐ Client libraries

Congratulations!

You learned how to launch Cloud Shell and ran some sample `gcloud` commands.



Finish Your Quest

This self-paced lab is part of the Qwiklabs [GCP Essentials](#) and [Using the Cloud SDK Command Line](#) Quests. A Quest is a series of related labs that form a learning path. Completing this Quest earns you the badge above, 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 a Quest and get immediate completion credit if you've taken this lab. [See other available Qwiklabs Quests](#).

Take Your Next Lab

Continue your Quest with [Provision Services with GCP Marketplace](#), or check out these suggestions:

- [Creating a Persistent Disk](#)
- [Configuring Networks via gcloud](#)

Next steps / Learn More

- Cloud Shell [Documentation](#) and [tutorial video](#).
- `gcloud` [Documentation](#) and [tutorial video](#).

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 October 16, 2019

Lab Last Tested August 1, 2019

Copyright 2020 Google LLC All rights reserved. Google and the Google logo are trademarks of Google LLC. All other company and product names may be trademarks of the respective companies with which they are associated.

Kubernetes Engine: Qwik Start

30 minutes1 Credit

Rate Lab

GSP100



Google Cloud Self-Paced Labs

Overview

[Google Kubernetes Engine](#) (GKE) provides a managed environment for deploying, managing, and scaling your containerized applications using Google infrastructure. The Kubernetes Engine environment consists of multiple machines (specifically [Google Compute Engine](#) instances) grouped together to form a [container cluster](#). In this lab, you will get hands on practice with container creation and application deployment with GKE.

Cluster orchestration with Kubernetes Engine

Kubernetes Engine clusters are powered by the [Kubernetes](#) open source cluster management system. Kubernetes provides the mechanisms through which you interact with your container cluster. You use Kubernetes commands and resources to deploy and manage your applications, perform administration tasks and set policies, and monitor the health of your deployed workloads.

Kubernetes draws on the same design principles that run popular Google services and provides the same benefits: automatic management, monitoring and liveness probes for application containers, automatic scaling, rolling updates, and more. When you run your applications on a container cluster, you're using technology based on Google's 10+ years of experience running production workloads in containers.

Kubernetes on Google Cloud Platform

When you run a Kubernetes Engine cluster, you also gain the benefit of advanced cluster management features that Google Cloud Platform provides. These include:

- [Load-balancing](#) for Compute Engine instances.
- [Node Pools](#) to designate subsets of nodes within a cluster for additional flexibility.
- [Automatic scaling](#) of your cluster's node instance count.
- [Automatic upgrades](#) for your cluster's node software.
- [Node auto-repair](#) to maintain node health and availability.
- [Logging and Monitoring](#) with Stackdriver for visibility into your cluster.

Now that you have a basic understanding of Kubernetes, you will learn how to deploy a containerized application with Kubernetes Engine in less than 30 minutes. Scroll down and follow the steps below to get your lab environment set up.

Setup and Requirements

Qwiklabs 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 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 the Google Cloud Platform 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 GCP account or project, do not use it for this lab.

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

Google Cloud Platform Console

How to start your lab and sign in to the 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.

Open Google Console

Caution: When you are in the console, do not deviate from the lab instructions. Doing so may cause your account to be blocked. [Learn more.](#)

Username

google2727032_student@qwiklabs.n

Password

k68CZXsxMZ

GCP Project ID

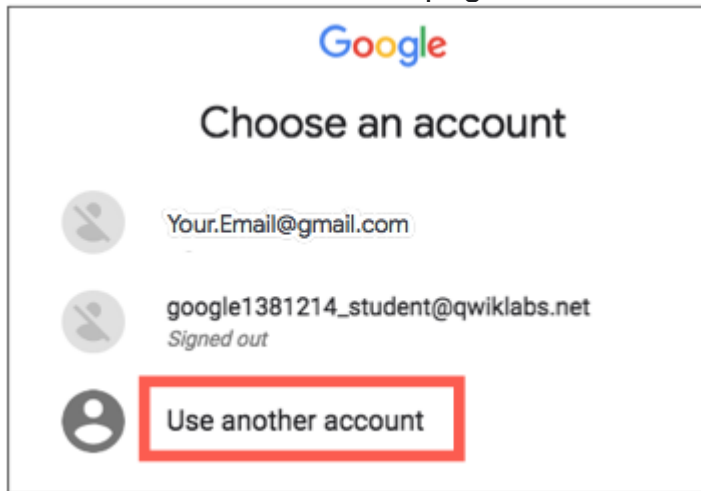
qwiklabs-gcp-4fbfecac8667e457

[New to labs? View our introductory video!](#)

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

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

3. On the Choose an account page, click **Use Another Account**.



4. The Sign in page opens. Paste the username that you copied from the Connection Details panel. Then copy and paste the password.

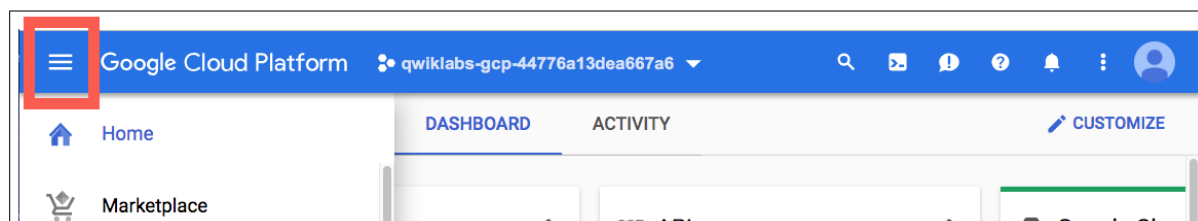
Important: You must use the credentials from the Connection Details panel. Do not use your Qwiklabs credentials. If you have your own GCP account, do not use it for this lab (avoids incurring charges).

5. 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 GCP console opens in this tab.

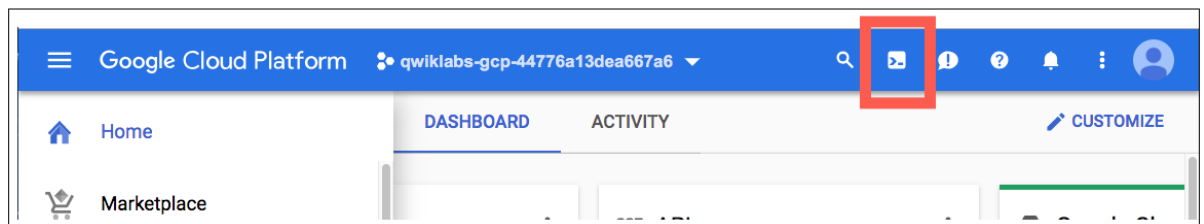
Note: You can view the menu with a list of GCP Products and Services by clicking the **Navigation menu** at the top-left, next to “Google Cloud Platform”.



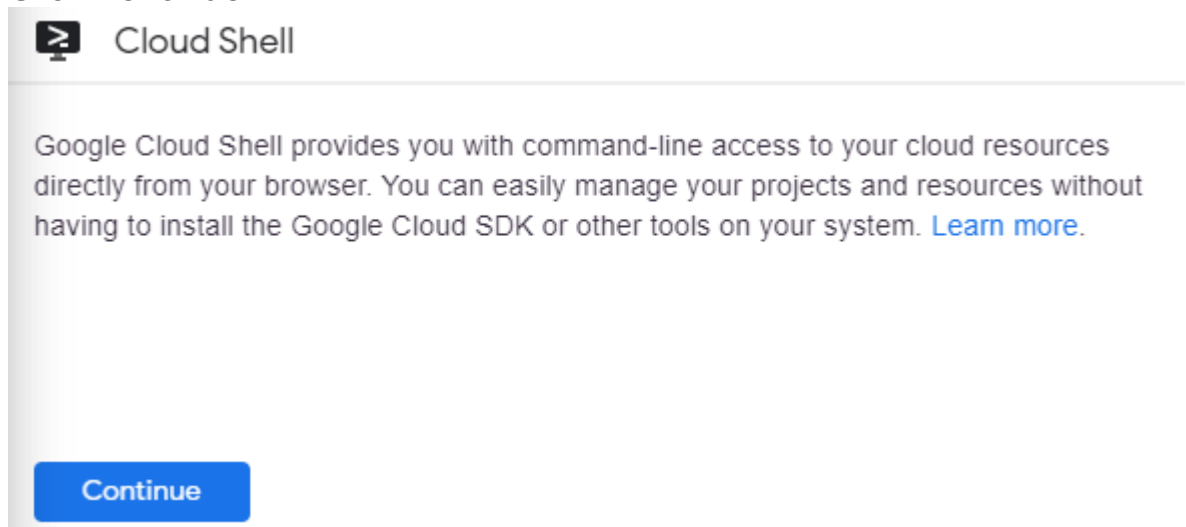
Activate Google Cloud Shell

Google 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. Google Cloud Shell provides command-line access to your GCP resources.

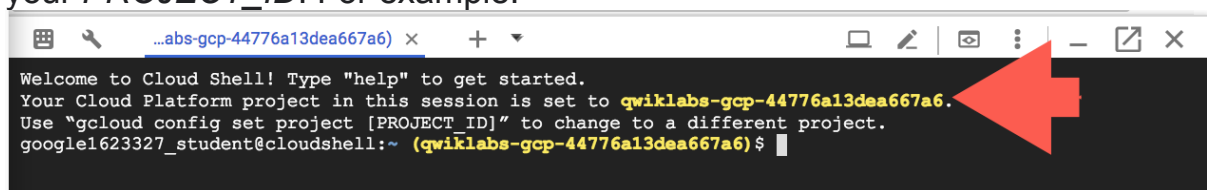
1. In GCP console, on the top right toolbar, click the Open Cloud Shell button.



2. 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 Platform. 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
```

Full documentation of **gcloud** is available on [Google Cloud gcloud Overview](#).

Setting a default compute zone

Your [compute zone](#) is an approximate regional location in which your clusters and their resources live. For example, `us-central1-a` is a zone in the `us-central1` region.

Start a new session in Cloud Shell and run the following command to set your default compute zone to `us-central1-a`:

```
gcloud config set compute/zone us-central1-a
```

You will receive the following output:

```
Updated property [compute/zone].
```

Creating a Kubernetes Engine cluster

A [cluster](#) consists of at least one *cluster master* machine and multiple worker machines called *nodes*. Nodes are [Compute Engine virtual machine \(VM\) instances](#) that run the Kubernetes processes necessary to make them part of the cluster.

To create a cluster, run the following command, replacing [CLUSTER-NAME] with the name you choose for the cluster (for example `my-cluster`). Cluster names must start with a letter, end with an alphanumeric, and cannot be longer than 40 characters.

```
gcloud container clusters create [CLUSTER-NAME]
```

You can ignore any warnings in the output. It might take several minutes to finish creating the cluster. Soon after you should receive a similar output:

NAME	LOCATION	...	NODE_VERSION	NUM_NODES	STATUS
my-cluster	us-central1-a	...	1.13.11-gke.9	3	RUNNING

Click **Check my progress** to verify the objective.

Create a Kubernetes Engine cluster

Check my progress

Get authentication credentials for the cluster

After creating your cluster, you need to get authentication credentials to interact with the cluster.

To authenticate the cluster run the following command, replacing [CLUSTER-NAME] with the name of your cluster:

```
gcloud container clusters get-credentials [CLUSTER-NAME]
```

You should receive a similar output:

```
Fetching cluster endpoint and auth data.  
kubeconfig entry generated for my-cluster.
```

Deploying an application to the cluster

Now that you have created a cluster, you can deploy a [containerized application](#) to it. For this lab you'll run `hello-app` in your cluster.

Kubernetes Engine uses Kubernetes objects to create and manage your cluster's resources. Kubernetes provides the [Deployment](#) object for deploying stateless applications like web servers. [Service](#) objects define rules and load balancing for accessing your application from the Internet.

Run the following [kubectl create](#) command in Cloud Shell to create a new [Deployment](#) `hello-server` from the `hello-app` container image:

```
kubectl create deployment hello-server --image=gcr.io/google-samples/hello-app:1.0
```

You should receive the following output:

```
deployment.apps/hello-server created
```

This Kubernetes command creates a Deployment object that represents `hello-server`. In this case, `--image` specifies a container image to deploy. The command pulls the example image from a [Google Container Registry](#) bucket. `gcr.io/google-samples/hello-app:1.0` indicates the specific image version to pull. If a version is not specified, the latest version is used.

Now create a Kubernetes Service, which is a Kubernetes resource that lets you expose your application to external traffic, by running the following [kubectl expose](#) command:

```
kubectl expose deployment hello-server --type=LoadBalancer --port 8080
```

In this command:

- `--port` specifies the port that the container exposes.
- `type="LoadBalancer"` creates a Compute Engine load balancer for your container.

You should receive the following output:

```
service/hello-server exposed
```

Click **Check my progress** to verify the objective.

Create a new Deployment - hello-server

Check my progress

Inspect the `hello-server` Service by running [kubectl get](#):

```
kubectl get service
```

You should receive a similar output:

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
hello-server	LoadBalancer	10.39.244.36	35.202.234.26	8080:31991/TCP	65s
kubernetes	ClusterIP	10.39.240.1	<none>	443/TCP	5m13s

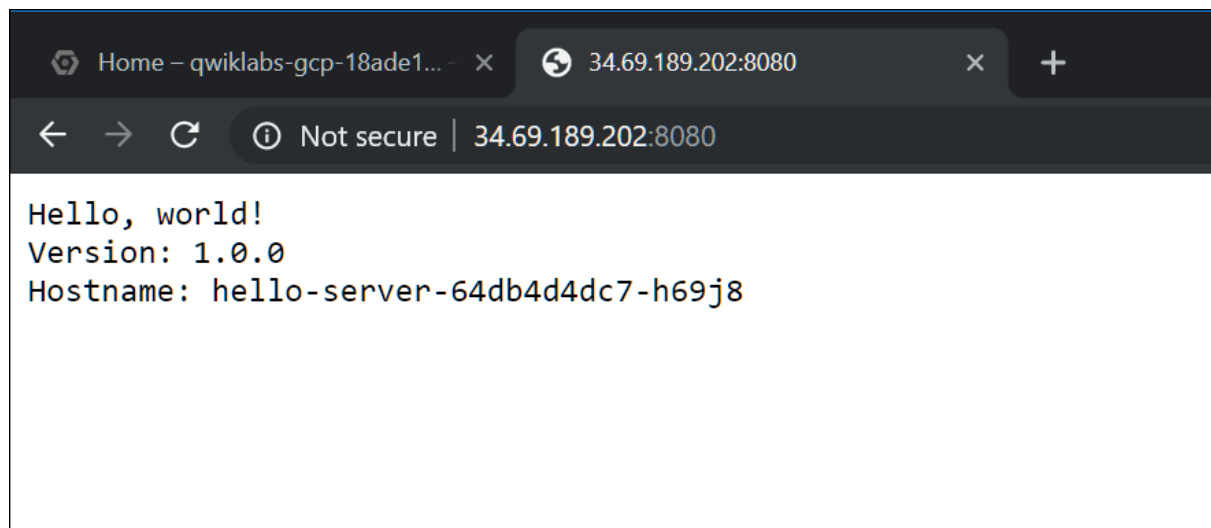
Note: It might take a minute for an external IP address to be generated. Run the above command again if the `EXTERNAL-IP` column is in "pending" status.

From this command's output, copy the Service's external IP address from the `EXTERNAL_IP` column.

View the application from your web browser using the external IP address with the exposed port:

```
http://[EXTERNAL-IP]:8080
```

Your page should resemble the following:



Click **Check my progress** to verify the objective.

Create a Kubernetes Service

Check my progress

Clean Up

Run the following to delete the cluster:

```
gcloud container clusters delete [CLUSTER-NAME]
```

When prompted, type **Y** to confirm. Deleting the cluster can take a few minutes. For more information on deleted Google Kubernetes Engine clusters, view the [documentation](#).

Click **Check my progress** to verify the objective.

Clean up: Delete the cluster

Check my progress

Congratulations!

You have just deployed a containerized application to Kubernetes Engine!

Finish Your Quest



Continue your [Baseline: Infrastructure](#), [GCP Essentials](#), or [VM Migration](#) Quest. A Quest is a series of related labs that form a learning path. Completing this Quest earns you the badge above, 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 a Quest and get immediate completion credit if you've taken this lab. [See other available Qwiklabs Quests](#).

Next Steps / Learn More

This lab is part of a series of labs called Qwik Starts. These labs are designed to give you a little taste of the many features available with Google Cloud. Search for "Qwik Starts" in the [lab catalog](#) to find the next lab you'd like to take!

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 November 11, 2019

Lab Last Tested November 11, 2019

Copyright 2020 Google LLC All rights reserved. Google and the Google logo are trademarks of Google LLC. All other company and product names may be trademarks of the respective companies with which they are associated.

Set Up Network and HTTP Load Balancers

40 minutes5 Credits

Rate Lab

GSP007



Google Cloud Self-Paced Labs

Overview

In this hands-on lab, you'll learn the differences between a network load balancer and a HTTP load balancer, and how to set them up for your applications running on Google Compute Engine virtual machines.

There are several ways you can [load balance in Google Cloud Platform](#). This lab takes you through the setup of the following load balancers.:

- L3 [Network Load Balancer](#)
- L7 [HTTP\(s\) Load Balancer](#)

Students are encouraged to type the commands themselves, which helps in learning the core concepts. Many labs include a code block that contains the required commands. You can easily copy and paste the commands from the code block into the appropriate places during the lab.

What you'll do

- Setup a network load balancer.
- Setup a HTTP(s) load balancer.
- Get hands-on experience learning the differences between network load balancers and HTTP load balancers.

Prerequisites

Familiarity with standard Linux text editors such as `vim`, `emacs`, or `nano` is 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 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 the Google Cloud Platform for the duration of the lab.

What you need

To complete this lab, you need:

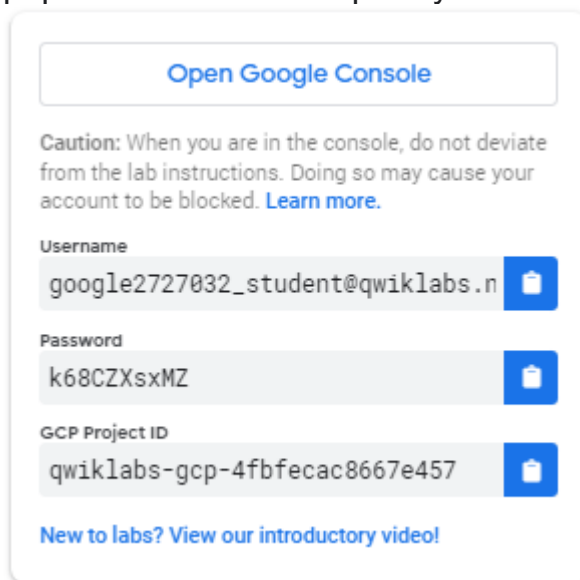
- Access to a standard internet browser (Chrome browser recommended).
- Time to complete the lab.

Note: If you already have your own personal GCP account or project, do not use it for this lab.

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

How to start your lab and sign in to the 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.



Open Google Console

Caution: When you are in the console, do not deviate from the lab instructions. Doing so may cause your account to be blocked. [Learn more.](#)

Username
google2727032_student@qwiklabs.n

Password
k68CZXsxMZ

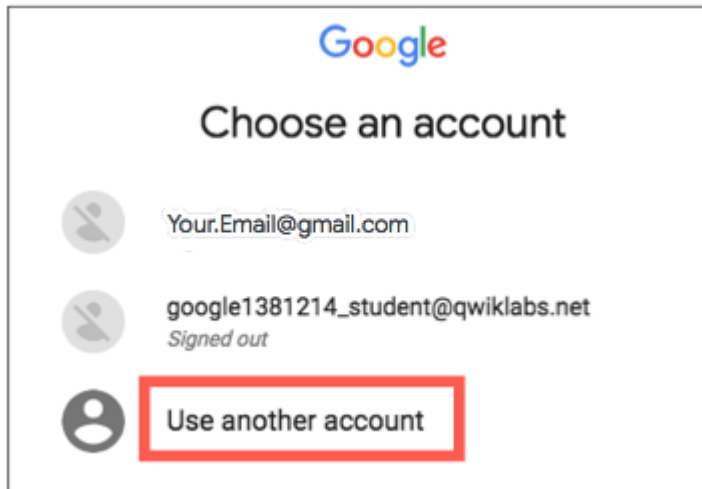
GCP Project ID
qwiklabs-gcp-4fbfecac8667e457

[New to labs? View our introductory video!](#)

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

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

3. On the Choose an account page, click **Use Another Account**.



4. The Sign in page opens. Paste the username that you copied from the Connection Details panel. Then copy and paste the password.

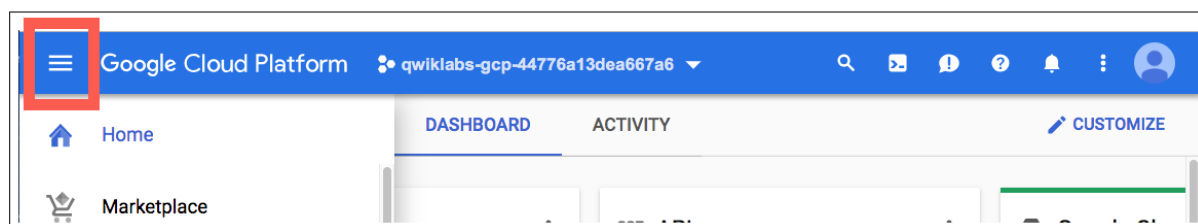
Important: You must use the credentials from the Connection Details panel. Do not use your Qwiklabs credentials. If you have your own GCP account, do not use it for this lab (avoids incurring charges).

5. 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 GCP console opens in this tab.

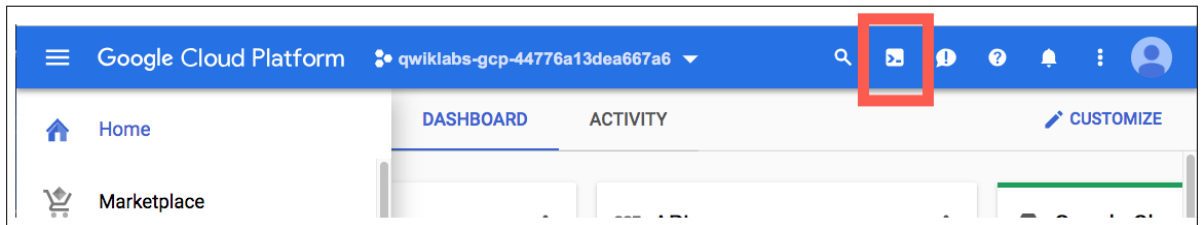
Note: You can view the menu with a list of GCP Products and Services by clicking the **Navigation menu** at the top-left, next to “Google Cloud Platform”.



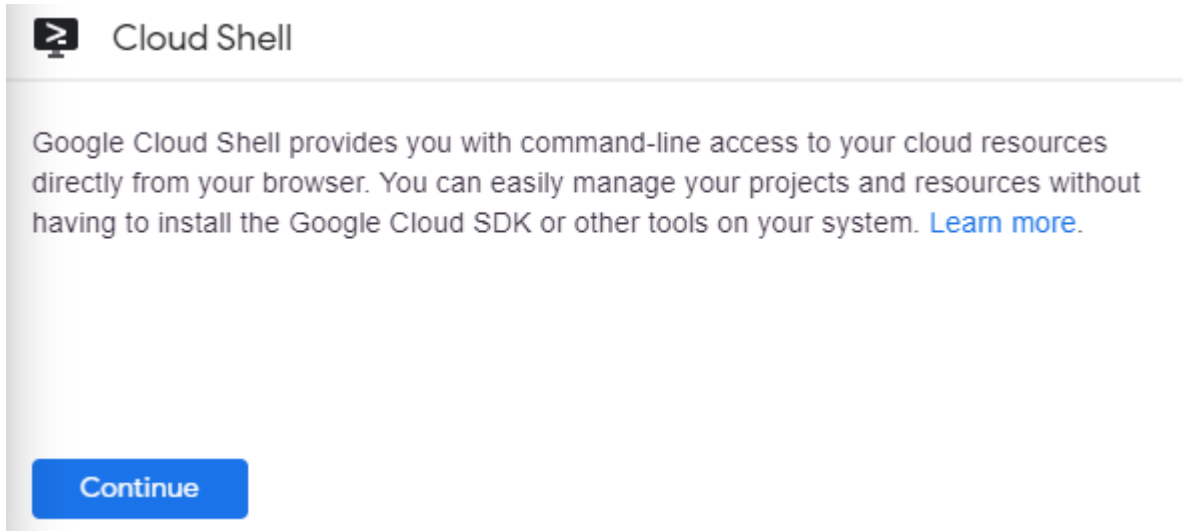
Activate Google Cloud Shell

Google 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. Google Cloud Shell provides command-line access to your GCP resources.

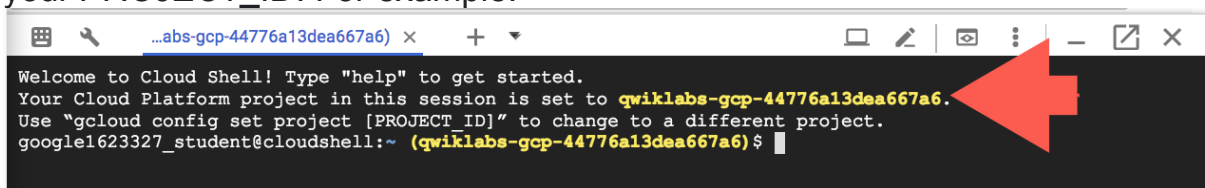
1. In GCP console, on the top right toolbar, click the Open Cloud Shell button.



2. 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 Platform. 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@gwiklabs.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
```

Full documentation of **gcloud** is available on [Google Cloud gcloud Overview](#).

Set the default region and zone for all resources

In Cloud Shell, set the default zone:

```
gcloud config set compute/zone us-central1-a
```

Set the default region:

```
gcloud config set compute/region us-central1
```

Learn more about choosing zones and regions here: [Regions & Zones documentation](#).

Note: When you run `gcloud` on your own machine, the `config` settings persist across sessions. In Cloud Shell you need to set this for every new session or reconnection.

Create multiple web server instances

To simulate serving from a cluster of machines, create a simple cluster of Nginx web servers to serve static content using [Instance Templates](#) and [Managed Instance Groups](#). Instance Templates define the look of every virtual machine in the cluster (disk, CPUs, memory, etc). Managed Instance Groups instantiate a number of virtual machine instances using the Instance Template.

To create the Nginx web server clusters, create the following:

- A startup script to be used by every virtual machine instance to setup Nginx server upon startup
 - An instance template to use the startup script
 - A target pool
 - A managed instance group using the instance template
- Still in Cloud Shell, create a startup script to be used by every virtual machine instance. This script sets up the Nginx server upon startup:

```
cat << EOF > startup.sh
#!/bin/bash
apt-get update
apt-get install -y nginx
service nginx start
sed -i -- 's/nginx/Google Cloud Platform - '"$HOSTNAME"'/'
/var/www/html/index.nginx-debian.html
EOF
```

Create an instance template, which uses the startup script:

```
gcloud compute instance-templates create nginx-template \
--metadata-from-file startup-script=startup.sh
```

(Output)

```
Created [...].
NAME                                MACHINE_TYPE  PREEMPTIBLE  CREATION_TIMESTAMP
nginx-template  n1-standard-1                2015-11-09T08:44:59.007-08:00
```

Create a target pool. A target pool allows a single access point to all the instances in a group and is necessary for load balancing in the future steps.

```
gcloud compute target-pools create nginx-pool
```

(Output)

```
Created [...].
NAME            REGION    SESSION_AFFINITY  BACKUP  HEALTH_CHECKS
nginx-pool  us-central1
```

Create a managed instance group using the instance template:

```
gcloud compute instance-groups managed create nginx-group \
--base-instance-name nginx \
--size 2 \
--template nginx-template \
--target-pool nginx-pool
```

(Output)

```
Created [...].
NAME          LOCATION      SCOPE  BASE_INSTANCE_NAME  SIZE  TARGET_SIZE
INSTANCE_TEMPLATE  AUTOSCALED
nginx-group    us-central1-a  zone   nginx               0     2
nginx-template  no
```

This creates 2 virtual machine instances with names that are prefixed with `nginx-`. This may take a couple of minutes.

List the compute engine instances and you should see all of the instances created:

```
gcloud compute instances list
```

(Output)

```
NAME          ZONE          MACHINE_TYPE  PREEMPTIBLE  INTERNAL_IP  EXTERNAL_IP
STATUS
nginx-7wvi    us-central1-a  n1-standard-1  10.240.X.X   X.X.X.X
RUNNING
nginx-9mwd    us-central1-a  n1-standard-1  10.240.X.X   X.X.X.X
RUNNING
```

Now configure a firewall so that you can connect to the machines on port 80 via the `EXTERNAL_IP` addresses:

```
gcloud compute firewall-rules create www-firewall --allow tcp:80
```

You should be able to connect to each of the instances via their external IP addresses via `http://EXTERNAL_IP/` shown as the result of running the previous command.

Check your lab progress. Click **Check my progress** below to verify that you've created a group of web servers.

Create a group of web servers

Check my progress

Create a Network Load Balancer

Network load balancing allows you to balance the load of your systems based on incoming IP protocol data, such as address, port, and protocol type. You also get some options that are not available, with HTTP(S) load balancing. For example, you can load balance additional TCP/UDP-based protocols such as SMTP traffic. And if your application is interested in TCP-connection-related characteristics, network load balancing allows your app to inspect the packets, where HTTP(S) load balancing does not.

For more information, see [Setting Up Network Load Balancing](#).
Create an L3 network load balancer targeting your instance group:

```
gcloud compute forwarding-rules create nginx-lb \  
  --region us-central1 \  
  --ports=80 \  
  --target-pool nginx-pool
```

(Output)

[Created \[https://www.googleapis.com/compute/v1/projects/...\]](https://www.googleapis.com/compute/v1/projects/...).
List all Google Compute Engine forwarding rules in your project.

```
gcloud compute forwarding-rules list
```

(Output)

NAME	REGION	IP_ADDRESS	IP_PROTOCOL	TARGET
nginx-lb	us-central1	X.X.X.X	TCP	us-central1/targetPools/nginx-pool

You can then visit the load balancer from the browser `http://IP_ADDRESS/` where `IP_ADDRESS` is the address shown as the result of running the previous command.

Check your lab progress. Click **Check my progress** below to verify that you've created an L3 Network Load Balancer that points to the webserver.

Create an L3 Network Load Balancer that points to the webserver

Check my progress

Create a HTTP(s) Load Balancer

HTTP(S) load balancing provides global load balancing for HTTP(S) requests destined for your instances. You can configure URL rules that route some URLs to one set of instances and route other URLs to other instances.

Requests are always routed to the instance group that is closest to the user, provided that group has enough capacity and is appropriate for the request. If the closest group does not have enough capacity, the request is sent to the closest group that does have capacity.

Learn more about the [HTTP\(s\) Load Balancer in the documentation](#).
First, create a [health check](#). Health checks verify that the instance is responding to HTTP or HTTPS traffic:

```
gcloud compute http-health-checks create http-basic-check
```

(Output)

```
Created [https://www.googleapis.com/compute/v1/projects/...].
```

NAME	HOST	PORT	REQUEST_PATH
http-basic-check		80	/

Define an HTTP service and map a port name to the relevant port for the instance group. Now the load balancing service can forward traffic to the named port:

```
gcloud compute instance-groups managed \
  set-named-ports nginx-group \
  --named-ports http:80
```

(Output)

```
Updated [https://www.googleapis.com/compute/v1/projects/...].
```

Create a [backend service](#):

```
gcloud compute backend-services create nginx-backend \
  --protocol HTTP --http-health-checks http-basic-check --global
```

(Output)

```
Created [https://www.googleapis.com/compute/v1/projects/...].
```

NAME	BACKENDS	PROTOCOL
nginx-backend		HTTP

Add the instance group into the backend service:

```
gcloud compute backend-services add-backend nginx-backend \
  --instance-group nginx-group \
  --instance-group-zone us-central1-a \
  --global
```

(Output)

```
Updated [https://www.googleapis.com/compute/v1/projects/...].
```

Create a default URL map that directs all incoming requests to all your instances:

```
gcloud compute url-maps create web-map \
  --default-service nginx-backend
```

(Output)

```
Created [https://www.googleapis.com/compute/v1/projects/...].
```

NAME	DEFAULT_SERVICE
Web-map	nginx-backend

To direct traffic to different instances based on the URL being requested, see [content-based routing](#).

Create a target HTTP proxy to route requests to your URL map:

```
gcloud compute target-http-proxies create http-lb-proxy \
  --url-map web-map
```

(Output)

```
Created [https://www.googleapis.com/compute/v1/projects/...].  
NAME          URL_MAP  
http-lb-proxy web-map
```

Create a global forwarding rule to handle and route incoming requests. A forwarding rule sends traffic to a specific target HTTP or HTTPS proxy depending on the IP address, IP protocol, and port specified. The global forwarding rule does not support multiple ports.

```
gcloud compute forwarding-rules create http-content-rule \  
  --global \  
  --target-http-proxy http-lb-proxy \  
  --ports 80
```

(Output)

```
Created [https://www.googleapis.com/compute/v1/projects/...].
```

After creating the global forwarding rule, it can take several minutes for your configuration to propagate.

```
gcloud compute forwarding-rules list
```

(Output)

NAME	REGION	IP_ADDRESS	IP_PROTOCOL	TARGET
http-content-rule		X.X.X.X	TCP	http-lb-proxy
nginx-lb	us-central1	X.X.X.X	TCP	us-central1/....

Take note of the http-content-rule IP_ADDRESS for the forwarding rule.

From the browser, you should be able to connect to `http://IP_ADDRESS/`. It may take three to five minutes. If you do not connect, wait a minute then reload the browser.

Check your lab progress. Click **Check my progress** below to verify that you've created an L7 HTTP(S) Load Balancer.

Create an L7 HTTP(S) Load Balancer

Check my progress

Test your knowledge

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

Network Load Balancing is a regional, non-proxied load balancer.



True



False

Submit

Congratulations!

You built a network load balancer and a HTTP(s) load balancer. You practiced with Instance templates and Managed Instance Groups. You are well on your way to having your Google Cloud Platform project monitored with Cloud Monitoring.

Finish Your Quest



This self-paced lab is part of the Qwiklabs [GCP Essentials](#) Quest. A Quest is a series of related labs that form a learning path. Completing this Quest earns you the badge above, 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 if you've taken this lab. [See other available Qwiklabs Quests](#).

Take Your Next Lab

Continue your Quest with [Hello Node Kubernetes](#), or check out these suggestions:

- [Provision Services with GCP Marketplace](#)
- [Stackdriver: Qwik Start](#)

Next Steps / Learn More

- Deploy more resources to your project, and see them get monitored
- Add your shiny new GCP Essentials badge to your resume!

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 July 15, 2019

Lab Last Tested July 15, 2019

Copyright 2020 Google LLC All rights reserved. Google and the Google logo are trademarks of Google LLC. All other company and product names may be trademarks of the respective companies with which they are associated.

Google Cloud Essentials: Challenge Lab

1 hour5 Credits

Rate Lab

GSP313



Google Cloud Self-Paced Labs

Overview

You must complete a series of tasks within the allocated time period. Instead of following step-by-step instructions, you'll be given a scenario and a set of tasks - you figure out how to complete it on your own! An automated scoring

system (shown on this page) will provide feedback on whether you have completed your tasks correctly.

To score 100% you must complete all tasks within the time period!

When you take a Challenge Lab, you will not be taught GCP concepts. To build the solution to the challenge presented, use skills learned from the labs in the quest this challenge lab is part of. You will be expected to extend your learned skills; you will be expected to change default values, but new concepts will not be introduced.

This lab is only recommended for students who have completed the labs in the [GCP Essentials](#) Quest. Are you up for the challenge?

Please make sure you review the labs in the GCP Essentials quest before starting this lab!

Topics tested:

- Create an instance.
- Create a 3 node Kubernetes cluster and run a simple service.
- Create an HTTP(s) Load Balancer in front of two web servers.

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 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 the Google Cloud Platform 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 GCP account or project, do not use it for this lab.

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

Challenge scenario

You have started a new role as a Junior Cloud Engineer for Jooli Inc. You are expected to help manage the infrastructure at Jooli. Common tasks include provisioning resources for projects.

You are expected to have the skills and knowledge for these tasks, so don't expect step-by-step guides to be provided.

Some Jooli Inc. standards you should follow:

- Create all resources in the default region or zone, unless otherwise directed.
- Naming is normally *team-resource*, e.g. an instance could be named **nucleus-webserver1**
- Allocate cost effective resource sizes. Projects are monitored and excessive resource use will result in the containing project's termination (and possibly yours), so beware. This is the guidance the monitoring team is willing to share; unless directed use **f1-micro** for small Linux VMs and **n1-standard-1** for Windows or other applications such as Kubernetes nodes.

Your challenge

As soon as you sit down at your desk and open your new laptop you receive several requests from the Nucleus team. Read through each description, then create the resources.

Task 1: Create a project jumphost instance

We will use this instance to perform maintenance for the project.

Make sure you:

- name the instance `nucleus-jumphost`
 - use the machine type of `f1-micro`
 - use the default image type (Debian Linux)
- Click *Check my progress* to verify the objective.

Create a project jumphost instance

Check my progress

If you don't get a green check mark, please click on the Score fly-out on the top right and click Run Step on the relevant step. You will see a hint pop up giving you advice.

Task 2: Create a Kubernetes service cluster

The team is building an application that will use a service. This service will run on Kubernetes. You need to:

- Create a cluster to host the service
- Use the Docker container `hello-app` (`gcr.io/google-samples/hello-app:2.0`) as a place holder, the team will replace the container with their own work later
- Expose the app on port 8080

Click *Check my progress* to verify the objective.

Create a Kubernetes cluster

Check my progress

If you don't get a green check mark, please click on the Score fly-out on the top right and click Run Step on the relevant step. You will see a hint pop up giving you advice.

Task 3: Setup an HTTP load balancer

We will serve the site via nginx web servers, but we want to ensure we have a fault tolerant environment, so please create an HTTP load balancer with a managed instance group of **two nginx web servers**. Use the following to configure the web servers, the team will replace this with their own configuration later.

```
cat << EOF > startup.sh
#!/bin/bash
apt-get update
apt-get install -y nginx
service nginx start
sed -i -- 's/nginx/Google Cloud Platform - '"$HOSTNAME"'/'
/var/www/html/index.nginx-debian.html
EOF
```

You need to:

- Create an instance template
- Create a target pool
- Create a managed instance group
- Create a firewall rule to allow traffic (80/tcp)
- Create a health check
- Create a backend service and attach the managed instance group
- Create a URL map and target HTTP proxy to route requests to your URL map
- Create a forwarding rule

Click *Check my progress* to verify the objective.

Create the website behind the HTTP load balancer

Check my progress

If you don't get a green check mark, please click on the Score fly-out on the top right and click Run Step on the relevant step. You will see a hint pop up giving you advice.

Congratulations!