

# Kubernetes Engine: Qwik Start

**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 [Compute Engine](#) instances) grouped to form a [container cluster](#). In this lab, you get hands-on practice with container creation and application deployment with GKE.

## Cluster orchestration with Google Kubernetes Engine

Google Kubernetes Engine (GKE) 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 administrative tasks, 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 with running production workloads in containers.

## Kubernetes on Google Cloud

When you run a GKE cluster, you also gain the benefit of advanced cluster management features that Google Cloud 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 Cloud Monitoring for visibility into your cluster

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

# Setup and requirements

## Before you click the Start Lab button

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

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

## What you need

To complete this lab, you need:

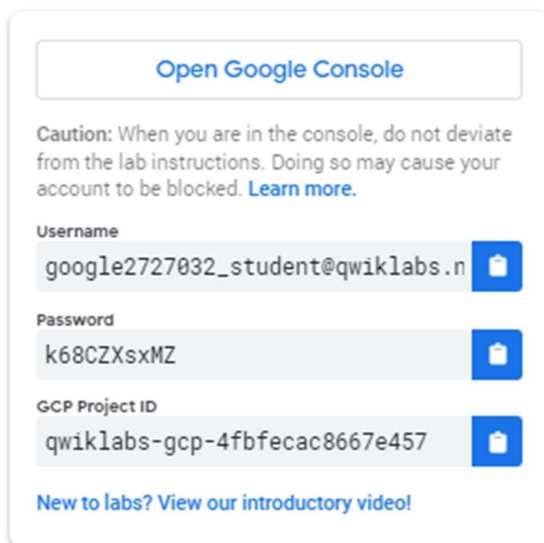
- Access to a standard internet browser (Chrome browser recommended).
- Time to complete the lab.

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

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

## How to start your lab and sign in to the Google Cloud Console

1. Click the **Start Lab** button. If you need to pay for the lab, a pop-up opens for you to select your payment method. On the left is a panel populated with the temporary credentials that you must use for this lab.



The screenshot shows a sign-in panel with the following elements:

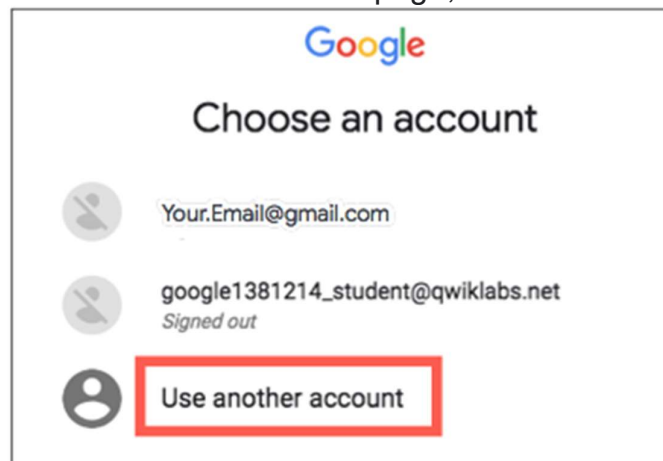
- A button at the top labeled "Open Google Console".
- A caution message: "Caution: When you are in the console, do not deviate from the lab instructions. Doing so may cause your account to be blocked. [Learn more.](#)"
- Three input fields, each with a copy icon to its right:
  - Username:** google2727032\_student@qwiklabs.n
  - Password:** k68CZXsxMZ
  - GCP Project ID:** qwiklabs-gcp-4fbfecac8667e457
- A link at the bottom: "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 **Sign in** page.



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

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



**Account.**

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

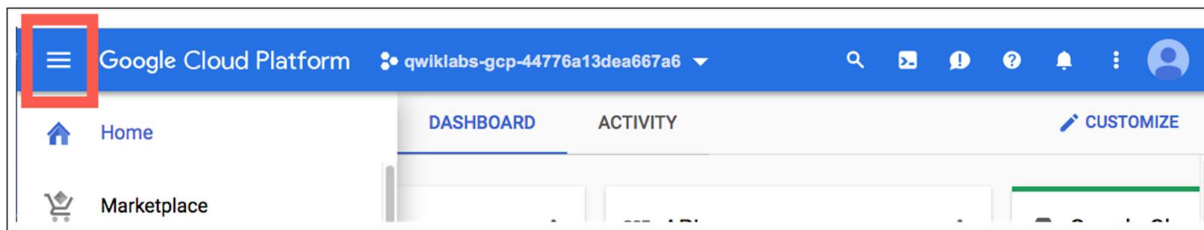
**Important:** You must use the credentials from the Connection Details panel. Do not use your Qwiklabs credentials. If you have your own Google Cloud account, do not use it for this lab (avoids incurring charges).

4. Click through the subsequent pages:

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

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

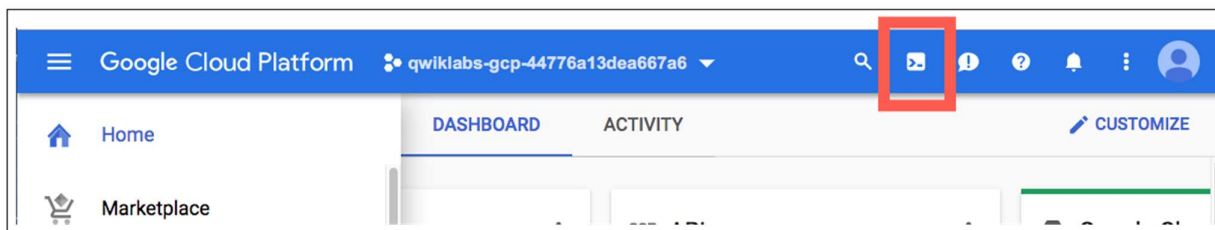
**Note:** You can view the menu with a list of Google Cloud Products and Services by clicking the **Navigation menu** at the top-left.



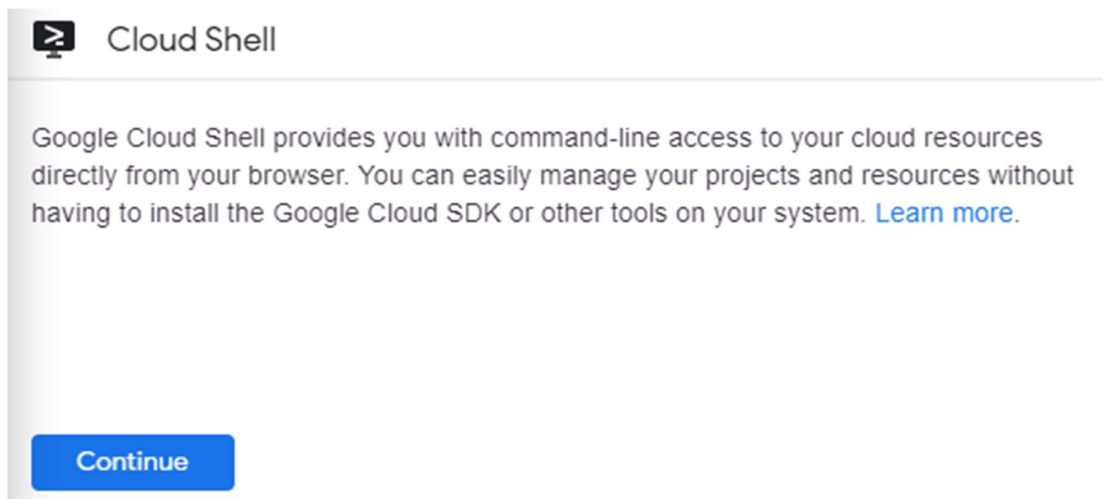
## Activate Cloud Shell

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

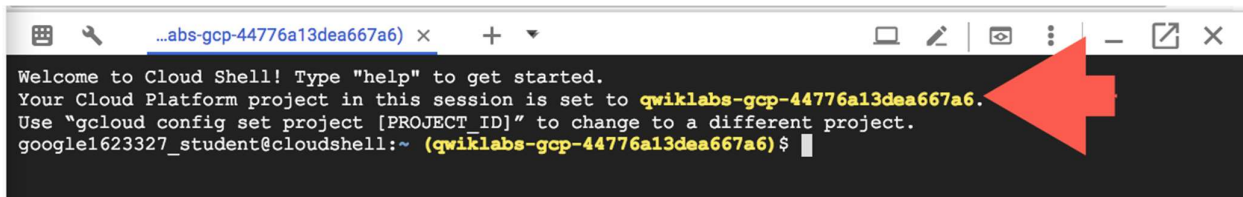
In the Cloud Console, in the top right toolbar, click the **Activate Cloud Shell** button.



Click **Continue**.



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



```
...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. It comes pre-installed on Cloud Shell and supports tab-completion.

You can list the active account name with this command:

```
gcloud auth list
```

(Output)

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

(Example output)

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

You can list the project ID with this command:

```
gcloud config list project
```

(Output)

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

(Example output)

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

For full documentation of `gcloud` see the [gcloud command-line tool overview](#).

# Task 1: Set 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.

1. To **set your default compute zone** to `us-central1-a`, start a new session in Cloud Shell, and run the following command:

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

**Expected output** (Do not copy):

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

# Task 2: Create a GKE 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.

**Note:** Cluster names must start with a letter and end with an alphanumeric, and cannot be longer than 40 characters.

1. To **create a cluster**, run the following command, replacing `[CLUSTER-NAME]` with the name you choose for the cluster (**for example**:`my-cluster`).

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

You can ignore any warnings in the output. It might take several minutes to finish creating the cluster.

**Expected output** (Do not copy):

| NAME                    | LOCATION                   | ... | NODE VERSION                 | NUM NODES      | STATUS               |
|-------------------------|----------------------------|-----|------------------------------|----------------|----------------------|
| <code>my-cluster</code> | <code>us-central1-a</code> | ... | <code>1.16.13-gke.401</code> | <code>3</code> | <code>RUNNING</code> |

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

## Task 3: Get authentication credentials for the cluster

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

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

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

**Expected output** (Do not copy):

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

## Task 4: Deploy an application to the cluster

You can now deploy a [containerized application](#) to the cluster. For this lab, you'll run `hello-app` in your cluster.

GKE 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.

1. To **create a new Deployment** `hello-server` from the `hello-app` container image, run the following [kubectl create](#) command:

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

**Expected output** (Do not copy):

```
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 [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.

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

3. To **create a Kubernetes Service**, which is a Kubernetes resource that lets you expose your application to external traffic, run the following [kubectl expose](#) command:



```
4. 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.

**Expected output** (Do not copy):

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

5. To **inspect** the `hello-server` Service, run [kubectl get](#):

```
6. kubectl get service
```

**Expected output** (Do not copy):

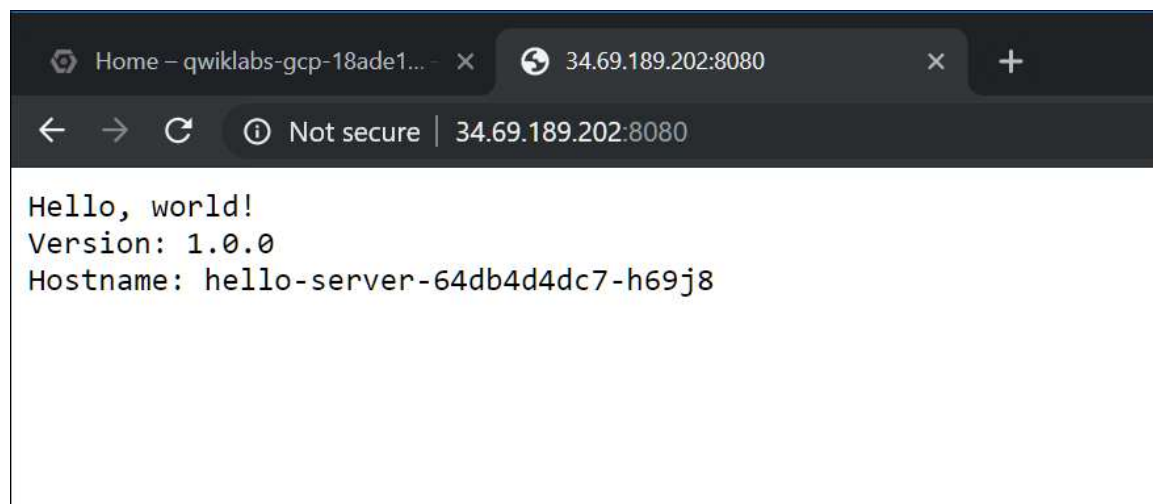
| NAME         | TYPE         | CLUSTER-IP   | EXTERNAL-IP   | PORT(S)        |
|--------------|--------------|--------------|---------------|----------------|
| hello-server | loadBalancer | 10.39.244.36 | 35.202.234.26 | 8080:31991/TCP |
| kubernetes   | ClusterIP    | 10.39.240.1  | <none>        | 433/TCP        |

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

7. To view the application from your web browser, open a new tab and enter the following address, replacing `[EXTERNAL IP]` with the `EXTERNAL-IP` for `hello-server`.

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

**Expected output:**



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

## Task 5: Deleting the cluster

1. To **delete** the cluster, run the following command:

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

3. When prompted, type **Y** to confirm.

Deleting the cluster can take a few minutes. For more information on deleted GKE clusters, view the [documentation](#).

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

# Congratulations!

You have just deployed a containerized application to Kubernetes Engine!

## Finish Your Quest



Continue your [Baseline: Infrastructure](#), [Google Cloud Essentials](#), [Kubernetes in Google Cloud](#), and [VM Migration](#) 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 for taking 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 some experience with 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 3, 2020

Lab Last Tested November 3, 2020

Copyright 2021 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.