Google Cloud Fundamentals: Getting Started with App Engine

20 minutes

Free

Rate Lab

Overview

In this lab, you create and deploy a simple App Engine application using a virtual environment in the Google Cloud Shell.

Objectives

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

Initialize App Engine.

Preview an App Engine application running locally in Cloud Shell.

Deploy an App Engine application, so that others can reach it.

Disable an App Engine application, when you no longer want it to be visible.

Set up your lab environment

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

Make sure you signed into Qwiklabs using an incognito window.

Note the lab's access time (for example, img/time.png and make sure you can finish in that time block.

There is no pause feature. You can restart if needed, but you have to start at the beginning.

When ready, click img/start\_lab.png.

Note your lab credentials. You will use them to sign in to Cloud Platform Console. img/open\_google\_console.png

Click Open Google Console.

Click Use another account and copy/paste credentials for this lab into the prompts.

If you use other credentials, you'll get errors or incur charges.

Accept the terms and skip the recovery resource page.

Do not click End Lab unless you are finished with the lab or want to restart it. This clears your work and removes the project.

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.

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

Cloud Shell icon

Click Continue. cloudshell\_continue.png

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:

Cloud Shell Terminal

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 .

Task 1: Initialize App Engine

Initialize your App Engine app with your project and choose its region:

gcloud app create --project=$DEVSHELL\_PROJECT\_ID

When prompted, select the region where you want your App Engine application located.

Clone the source code repository for a sample application in the hello\_world directory:

git clone https://github.com/GoogleCloudPlatform/python-docs-samples

Navigate to the source directory:

cd python-docs-samples/appengine/standard\_python3/hello\_world

Task 2: Run Hello World application locally

In this task, you run the Hello World application in a local, virtual environment in Cloud Shell.

Ensure that you are at the Cloud Shell command prompt.

Execute the following command to download and update the packages list.

sudo apt-get update

Set up a virtual environment in which you will run your application. Python virtual environments are used to isolate package installations from the system.

sudo apt-get install virtualenv

If prompted [Y/n], press Y and then Enter.

virtualenv -p python3 venv

Activate the virtual environment.

source venv/bin/activate

Navigate to your project directory and install dependencies.

pip install -r requirements.txt

Run the application:

python main.py

Please ignore the warning if any.

In Cloud Shell, click Web preview (Web Preview) > Preview on port 8080 to preview the application.

To access the Web preview icon, you may need to collapse the Navigation menu.

Result:

hello\_world.png

To end the test, return to Cloud Shell and press Ctrl+C to abort the deployed service.

Using the Cloud Console, verify that the app is not deployed. In the Cloud Console, on the Navigation menu (Navigation menu), click App Engine > Dashboard.

Notice that no resources are deployed.

Task 3: Deploy and run Hello World on App Engine

To deploy your application to the App Engine Standard environment:

Navigate to the source directory:

cd ~/python-docs-samples/appengine/standard\_python3/hello\_world

Deploy your Hello World application.

gcloud app deploy

If prompted "Do you want to continue (Y/n)?", press Y and then Enter.

This app deploy command uses the app.yaml file to identify project configuration.

Launch your browser to view the app at http://YOUR\_PROJECT\_ID.appspot.com

gcloud app browse

Copy and paste the URL into a new browser window.

Result:

disable-app.png

Congratulations! You created your first application using App Engine.

Click Check my progress to verify the objective.

Deploy the Hello World application to App Engine

Task 4: Disable the application

App Engine offers no option to Undeploy an application. After an application is deployed, it remains deployed, although you could instead replace the application with a simple page that says something like "not in service."

However, you can disable the application, which causes it to no longer be accessible to users.

In the Cloud Console, on the Navigation menu (Navigation menu), click App Engine > Settings.

Click Disable application.

Read the dialog message. Enter the App ID and click DISABLE.

If you refresh the browser window you used to view to the application site, you'll get a 404 error.

f17c85cf862ddae3.png

Congratulations!

You created your first application using App Engine!

End your lab