

Homework - Google Cloud Platform (GCP) with Python

This semester we are allowing all students to explore cloud computing as offered by the Google Cloud Platform using Python. Using the instructions below, you can establish a service using Google App Engine. Once established, you will be able to move your Python program developed for Assignment #2 to your Google App Engine instance and have it executed there.

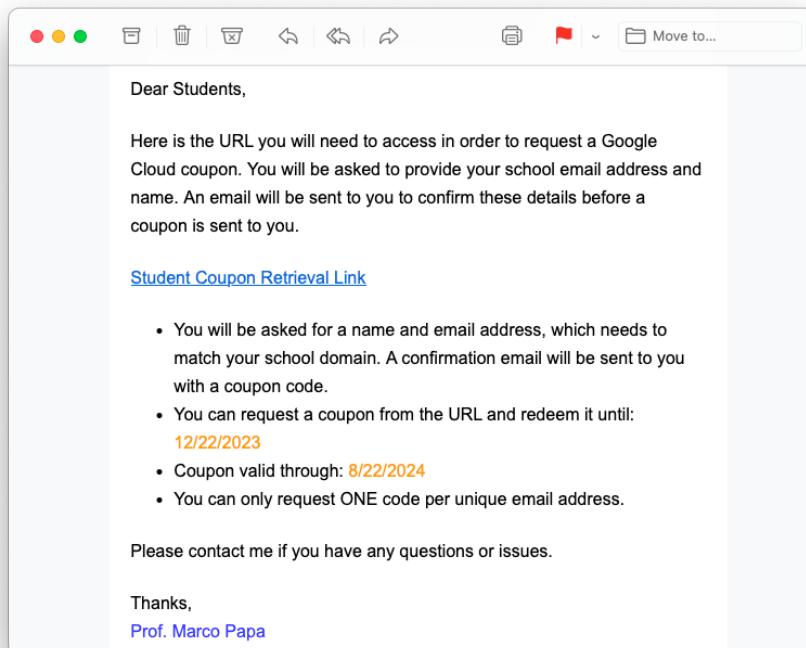
1. Sign up for Google Cloud Platform

If you do not have a credit card, Google provides you with a coupon code via the Google Cloud Platform Education Grants program (see **section 1.1**). If you do have a credit card, you can sign up for the Google Cloud Platform “Free Trial” (see **section 1.2**).

1.1 Get Google Cloud Platform Education Grants credit

On Piazza and by e-mail, you will receive a communication like the one displayed below. The communication contains information on how to request a Google Cloud Platform coupon. Click on the text **Student Coupon Retrieval Link**, or the link provided in the Piazza post.

You will be redirected to a web form as shown below:



The screenshot shows a web browser window with the title "Higher Education Grants - Google Cloud Platform". The URL is https://google.secure.force.com/GCPEDU/?cid=IAoM0... The page has a blue header with the Google Cloud logo and the text "Cloud Platform Education Grants". Below the header, it says "Use credits provided to you via the Google Cloud Platform Education Grants program to access Google Cloud Platform. Get what you need to build and run your apps, websites and services." A central white box contains instructions: "Thank you for your interest in Google Cloud Platform Education Grants. Please fill out the form below to receive a coupon code for credit to use on Google Cloud Platform." It has three input fields: "First Name", "Last Name", and "School Email" (with a ".@usc.edu" suffix). Below the email field is a note: "If you do not see your domain listed, please contact your course instructor: papa@usc.edu". There is a small text block about privacy: "By clicking "Submit" below, you agree that we may share the following information with your educational institution and course instructor (papa@usc.edu): (1) personal information that you provide to us on this form and (2) information regarding your use of the coupon and Google Cloud Platform products." A blue "Submit" button is at the bottom. At the very bottom of the page, there is a link to "Privacy Policy".

Enter your **First Name**, **Last Name** and your **USC e-mail address**. @usc.edu will be pre-filled. **Click on Submit**. If you entered a valid USC e-mail address, an email will be sent to that USC email address to verify that you own such address. A sample email is shown below:

Dear [Laurie](#),

Thank you for your interest in downloading a Google Cloud Platform Coupon Code. Please click on this [link](#) to verify your email address and a code will be sent to your email account.

Notice that anyone with the URL from USC can request a coupon, so please be careful and **do not share the Student Coupon Retrieval Link** or the link to verify your email.

Once your USC email address is “verified”, you will receive a second email with a Google Cloud Platform Coupon Code, as shown below.

Dear [Laurie](#),

Here is your Google Cloud Platform Coupon Code: **4G8B-E0XC-6J2H-65TR**

Click [\[here\]](#) to redeem.

Danger!

Important step: Before clicking on the link labeled **[here]**, you should open your default browser, and **login** to a **Gmail** account. USC no longer provides @gmail.com accounts to students. If you already have a @gmail.com account, you can use it. If not, you can create one by going here:

<https://support.google.com/mail/answer/56256?hl=en>

Once logged into Gmail, you can click on **[here]** in the e-mail, or you can go to this page:

<https://console.cloud.google.com/education>

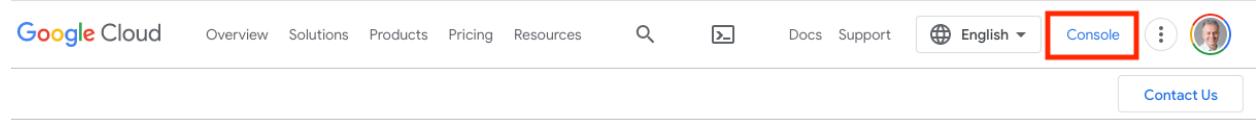
and paste the Coupon Code, to redeem your coupon. The web form below will be displayed.

The screenshot shows a web browser window titled "Education grants" at <https://console.cloud.google.com/education>. The page is part of the Google Cloud Platform interface. At the top right, there is a user profile for "Marco Papa" with the email "papa.marco@gmail.com". Below the profile, there is another section for "Marco Papa" with the email "papa@usc.edu". A "Sign out" button is also visible. The main form area has fields for "Coupon code" (empty), "Country of residence" (set to "United States"), and a checkbox for "Please email me updates regarding future announcements, performance suggestions, feedback surveys and special offers." The checkbox has two options: "Yes" (selected) and "No". Below these fields, there is a detailed "Google Cloud Platform education grants credits terms and conditions" section. It includes a paragraph about the credit being valid for Google Cloud Platform products and subject to acceptance of the Google Cloud Platform License Agreement and other applicable terms of service. It also mentions that the credit is non-transferable and may not be sold or bartered. The credit expires on the date indicated in the message. The text also notes that unused credit may be removed from the account as You use the credit over the period of time that the credit is valid. Offer void where prohibited by law. A note states that You represent that You are accepting the promotional credit on behalf of your educational institution and the credit can only be used on behalf of that educational entity and not for your personal use. You represent, on behalf of such educational entity, that (i) You are authorized to accept this credit; (ii) the credit is being used in accordance with all applicable laws and regulations, including bank rules and laws, and (iii) the use of the credit will not negatively impact Google's current or future ability to do business with such educational entity. A note also states that You agree that You may share the following information with your educational institution and course instructor: (1) personal information that You provide to us during the sign-up and onboarding process and (2) information regarding Your use of the coupon on Google Cloud Platform products. At the bottom of the form are two buttons: "Accept and continue" (highlighted in blue) and "Clear".

You need to paste your coupon into the field labeled **Coupon code**. Select **Yes** or **No** to receive announcements. Make sure that the active profile in the top right is the one associated with your Gmail account. **Click on Accept and continue**. You will now be

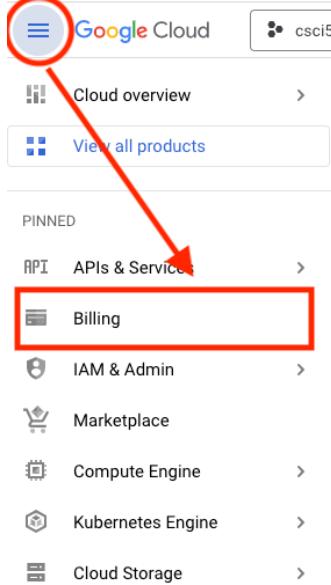
taken to the Google Cloud Platform's **Home** section.

If you see a “**Console**” link next to your picture / avatar, click on it.

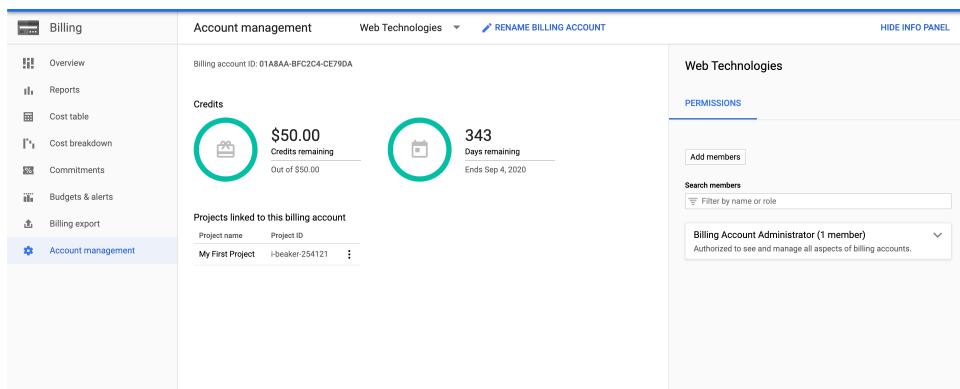


The screenshot shows the top navigation bar of the Google Cloud Home page. On the right side, there is a user profile icon, a "Console" link, and a "Contact Us" button. A red box highlights the "Console" link.

On the next page, click on the “**burger**” in the top left, and then select “**Billing**”.

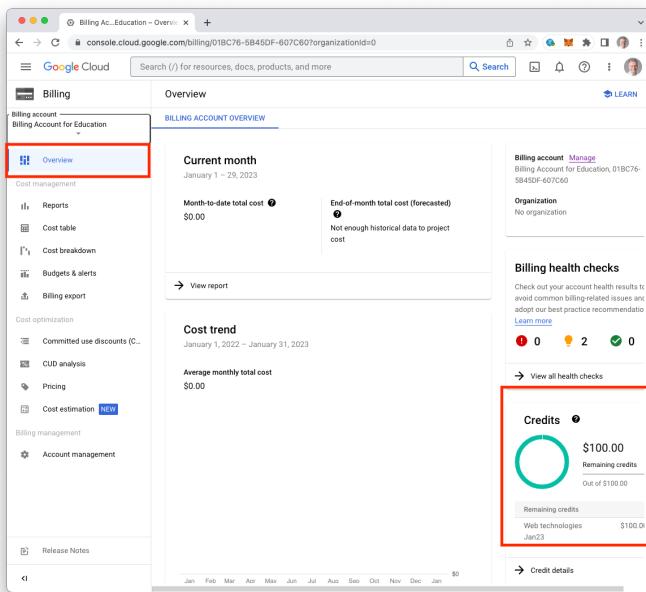


You can navigate to the **Billing** section and navigate to **Account Management** to see the amount of your credit, as shown below.



The screenshot shows the "Account management" page under the "Billing" section. The sidebar on the left has options: Overview, Reports, Cost table, Cost breakdown, Commitments, Budgets & alerts, Billing export, and Account management (which is selected and highlighted with a blue background). The main content area shows "Credits" with a balance of "\$50.00" and "343 Days remaining" (both highlighted with green circles). It also lists "Projects linked to this billing account" with one entry: "My First Project" (Project ID: i-beaker-254121). On the right, there are sections for "Web Technologies" (with a "PERMISSIONS" sub-section showing "Add members" and "Search members") and "BILLING ACCOUNT ADMINISTRATOR (1 member)" (listing "Billing Account Administrator (1 member)" with "Authorized to see and manage all aspects of billing accounts").

If the **Account Management** does not show your credits, select **Overview**. Your \$25 credits should show on the Credits section.



Important Note: if you have redeemed your coupon with your USC e-mail account, instead of your Gmail account, your coupon will not be usable, as the USC G Suite account does not allow the user to create GCP Projects. If you accidentally did this, you can apply the coupon to the correct billing account, by following the steps in this document:

http://csci571.com/hw/hw5/GCP_G_Suite_Workaround.pdf

If you were successful in signing up and obtaining the \$25 CGP Education Credit, skip to section **1.3. How to get additional student Coupons.**

1.2 Sign up for Google Cloud Platform Free Trial

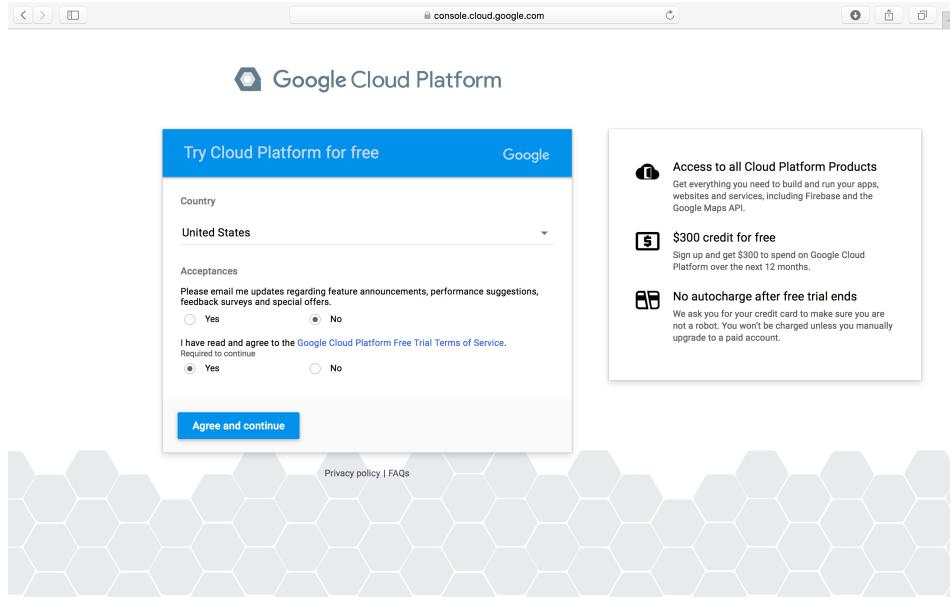
IMPORTANT: you should follow the steps in this section, only of you were unable to obtain the \$50 coupon.

To sign up for the Free Trial, with an additional \$300 credit, you need a credit card. Unfortunately, an American Express or other pre-paid Gift card will not work with Google Cloud.

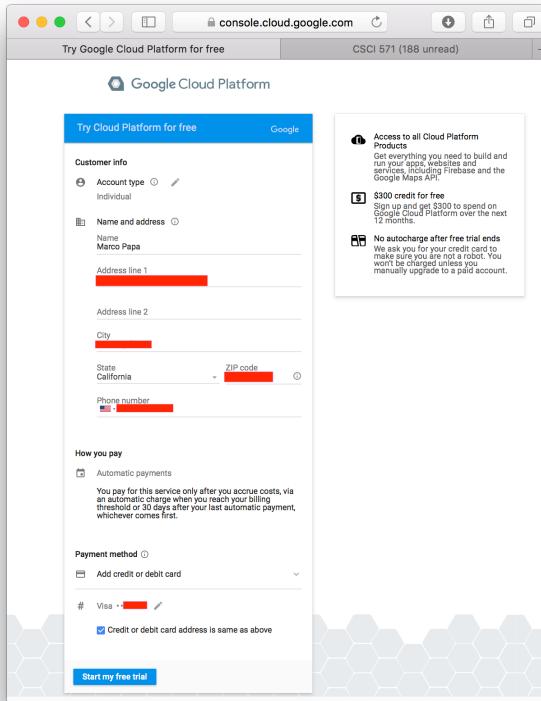
To sign up go to:

<https://console.cloud.google.com/freetrial?pli=1&page=0>

In the Try Cloud Platform for free page, select **Yes** under “I have read and agree to the Google Cloud Platform Free Trial terms of Service” and click on **Agree and continue**.

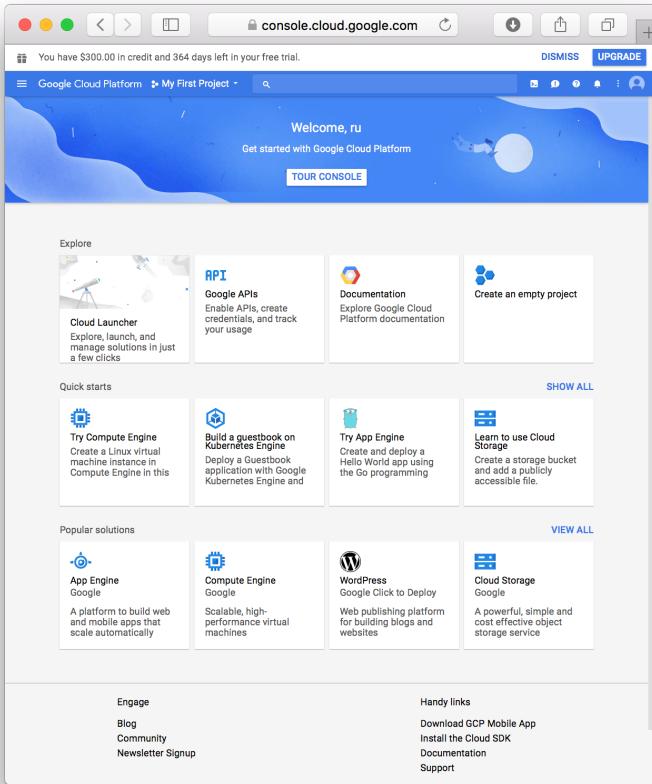


Select **Account type Individual**. Follow the instructions to enter your account data. You should not be using your `@usc.edu` e-mail account for your primary contact e-mail address, but instead use your `@gmail.com` address and finish by clicking **Start my free trial**. You will have to provide a credit or debit card.



After you are signed up, you will see the message “Creating project. This may take a few

moments'.' You will then be redirected to the **Dashboard** of the **Google Developer Console**.



To confirm your credits, navigate to **Billing > Account Management** from the left navigation bar to see a credit value of \$300 valid for 365 days or you can verify it as below.

The screenshot shows the Google Cloud Platform Billing Overview page. The left sidebar has options like Overview, Reports, Cost table, Cost breakdown, Commitments, Budgets & alerts, Billing export, Transactions, Payment settings, Payment method, and Account management. The main area has tabs for "BILLING ACCOUNT OVERVIEW" and "PAYMENT OVERVIEW". It displays a message: "Welcome to the new Billing Overview! Billing permissions, project associations, and credit details can now be found on the account management page." Below this are sections for "Current month" (September 1 – 24, 2019) and "Cost trend" (September 1, 2018 – September 30, 2019). The "Current month" section shows "Month-to-date total cost" (\$0.00) and "End-of-month total cost (forecasted)" (\$0.00). The "Cost trend" section shows "Average monthly total cost" (\$0.00). To the right, there's a "Billing account Manage" section with "My Billing Account, 01F770-5AFF1E-4B1967", "Organization" (No organization), and "Promotional credits" (\$300.00). A circled area highlights the "\$300.00" value.

If you previously developed any projects using Google APIs, you will find them listed.

1.3. How to get additional student Coupons

If you follow our instructions to install Python (and late on Node.js) you will likely never incur charges that exceed the value of your coupons. But there are always students that want to play around and run services all over the place.

When a student exceeds 60% of the value of a coupon, Google sends a notification, by e-mail, to the instructor. The instructor can get additional coupons for the student by filling out the same form listed on page 2, using the instructor's e-mail address that was used to obtain the grant. The instructor will receive the coupon and deliver it to the student by e-mail.

Google limits the additional coupons to 2 for each student account used in each course.

2. Setting up a Python development environment

To set up a Python development environment for GCP to develop Python apps that run on Google Cloud, you should follow the steps from this tutorial:

<https://cloud.google.com/python/docs/setup>

The tutorial covers all the following:

- Install the latest version of Python.
- Use `venv` to isolate dependencies.
- Install an editor (optional).
- Install the Cloud SDK (optional).
- Install the Cloud Client Libraries for Python (optional).
- Install other useful tools.

2.1 Installing Python

The tutorial provides steps to install the latest version of Python 3 on macOS, Windows and Linux.

Note: As of 9/17/2023, App Engine on Google Cloud is compatible with Python 3.7, all the way to 3.12. Quoting from:

<https://cloud.google.com/appengine/docs/standard/python3/runtime>

“The Python 3 runtime supports Python 3.7 through Python 3.12 and uses the latest stable release of the version that is specified in your `app.yaml` file.”

Note: Support for Python 3.5-3.6 was deprecated on August 8th, 2023.

Installing on macOS

Older versions of macOS includes a version of Python by default and uses it for its own purposes (normally version 2.7.X). Verify your Mac's Python installation using the following command:

```
/usr/bin/env python -V
```

Newer version of macOS, no longer include Python 2.7. In April 2022, Apple removed support for Python 2.7 on macOS devices running **Monterey 12.3** and above.

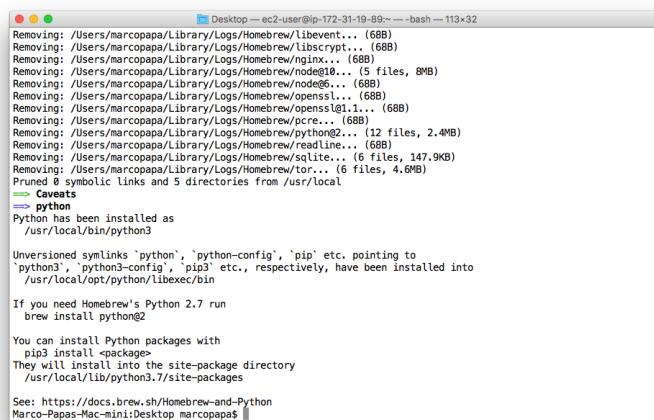
To avoid interfering with macOS, we recommend creating a separate development environment and installing the latest version of Python (version 3.8 or later). To install Python, use **Homebrew**, available at:

<https://brew.sh/>

After installing Homebrew, you can install the latest Python with:

```
brew install python
```

As of this writing, Homebrew will install **Python 3.11**. If all is well, the installation will complete, as shown below.



```
Removing: /Users/marcopapa/Library/Logs/Homebrew/libevent... (68B)
Removing: /Users/marcopapa/Library/Logs/Homebrew/libcrypt... (68B)
Removing: /Users/marcopapa/Library/Logs/Homebrew/nginx... (68B)
Removing: /Users/marcopapa/Library/Logs/Homebrew/nodev10... (5 files, 8MB)
Removing: /Users/marcopapa/Library/Logs/Homebrew/nodev6... (68B)
Removing: /Users/marcopapa/Library/Logs/Homebrew/openssl... (68B)
Removing: /Users/marcopapa/Library/Logs/Homebrew/opensslv11... (68B)
Removing: /Users/marcopapa/Library/Logs/Homebrew/opensslv10... (68B)
Removing: /Users/marcopapa/Library/Logs/Homebrew/pythonv2... (12 files, 2.4MB)
Removing: /Users/marcopapa/Library/Logs/Homebrew/readline... (68B)
Removing: /Users/marcopapa/Library/Logs/Homebrew/sqlite... (6 files, 147.9KB)
Removing: /Users/marcopapa/Library/Logs/Homebrew/tor... (6 files, 4.6MB)
Prune 0 symbolic link and 5 directories from /usr/local
==> Caveats
=> python
Python has been installed as
/usr/local/bin/python3

Unversioned symlinks 'python', 'python-config', 'pip' etc. pointing to
'python3', 'python3-config', 'pip3' etc., respectively, have been installed into
/usr/local/opt/python/libexec/bin

If you need Homebrew's Python 2.7 run
brew install python@2

You can install Python packages with
pip3 install <package>
They will install into the site-package directory
/usr/local/lib/python3.7/site-packages

See: https://docs.brew.sh/Homebrew-and-Python
Marco-Papa-Mac-mini:Desktop marcopapa$
```

Normally Python 3 will be installed in `/usr/local/bin/python3` (a symbolic link to another folder).

If you have a newer Mac with Apple Silicon M Chips, brew will be installed in /opt/homebrew/bin/python3.

The following aliases need only be run on an old Mac with Python 2.7. If you have kept the default Python 2.7, you will have to add aliases to your startup files (for Bash and Zsh) for Python 3 and Pip 3 locations, run the following commands:

```
echo "alias python=/usr/local/bin/python3.9" >> ~/.zshrc
echo "alias python=/usr/local/bin/python3.9" >> ~/.bashrc
echo "alias pip=/usr/local/bin/pip3" >> ~/.zshrc
echo "alias pip=/usr/local/bin/pip3" >> ~/.bashrc
```

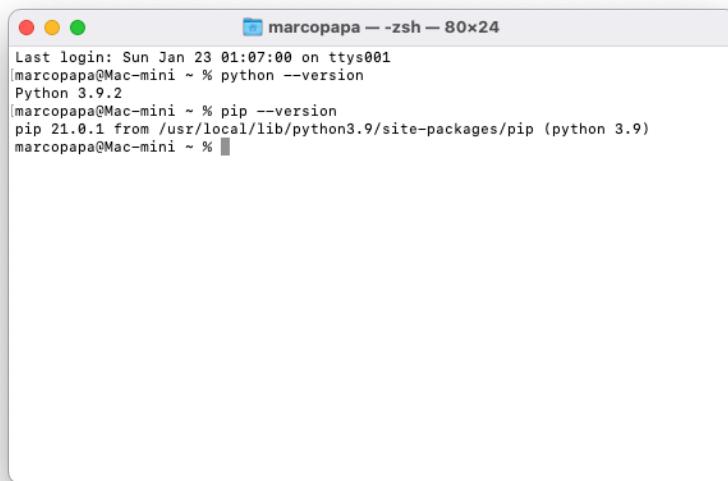
If you are using bash and `~/.bash_profile`, you may have to do this instead:

```
echo "alias python=/usr/local/bin/python3.9" >> ~/.bash_profile
echo "alias pip=/usr/local/bin/pip3" >> ~/.bash_profile.
```

Replace `python3.9` in the script above with your installed version, such as `python3.11`.

You will have to re-start Terminal, so the aliases will take effect. Start a new terminal session and verify Python 3 is available as `python` and `python3`, and that `pip` is also installed, by running the following commands:

```
python --version
pip --version
```



The screenshot shows a terminal window with the title bar 'marcopapa -- zsh -- 80x24'. The window contains the following text:

```
Last login: Sun Jan 23 01:07:00 on ttys001
marcopapa@Mac-mini ~ % python --version
Python 3.9.2
marcopapa@Mac-mini ~ % pip --version
pip 21.0.1 from /usr/local/lib/python3.9/site-packages/pip (python 3.9)
marcopapa@Mac-mini ~ %
```

As an alternative of installing Python and Pip with homebrew, one can install Xcode with

additional commands. When doing so, both Python 3 and Pip 3 are installed as part of the Xcode *command line developer tools*, and there is no need to set the bash and/or zsh aliased paths.

Installing on Windows

Since Windows does not come with Python, download the installers for the latest versions of Python from the Python website at:

<https://www.python.org/downloads/windows/>

as of this writing, we recommend you download **Python 3.7.6**, the same version that we recommend for macOS. Complete the installation by adding the proper PATH and verifying the version of Python 3 and pip installed, as outlined in the tutorial.

2.2 Use venv to isolate dependencies

venv is a tool that creates isolated Python environments. Use the **venv** command to create a virtual copy of the entire Python installation.

Follow the tutorial to do the following:

- Create a virtual copy in a folder named venv
- Set your shell to use the venv paths for Python by activating the virtual environment
- Install packages without affecting other projects or your global Python installation
- If you want to stop using the virtual environment and go back to your global Python, you can deactivate it

2.3 Installing a Python editor

There are several, popular editors for Python. In particular **Sublime Text**, **Atom**, **Visual Studio Code**, and **PyCharm**. We recommend that you use **PyCharm**, as it is free for students from JetBrains, and available at:

<https://www.jetbrains.com/pycharm/>

The free “educational” version of **PyCharm** can be downloaded here:

<https://www.jetbrains.com/education/download/#section=pycharm-edu>

PyCharm is available for macOS, Windows and Linux.

Another good editor is **Visual Studio Code** by Microsoft. It can be downloaded here:

<https://code.visualstudio.com/>

Visual Studio Code is available for macOS, Windows and Linux.

3. Creating a Project and Application using CLI

The Cloud SDK is a set of command-line tools for Google Cloud. It contains **gcloud**, and **gsutil**, which you can use to access App Engine, Compute Engine, Cloud Storage, and other products and services from the command line. The Cloud SDK is available at:

<https://cloud.google.com/sdk/>

The Cloud SDK is available for **Linux**, Ubuntu, CentOS, **macOS** and **Windows**. Quickstarts for each platform are available here:

<https://cloud.google.com/sdk/docs/quickstarts>

1. The “*Quickstart for Python 3 in the App Engine Standard Environment*” page is available at:

<https://cloud.google.com/appengine/docs/standard/python3/quickstart>

2. The QuickStart tutorial provides all the steps needed to do all the following:

- Downloading and installing the Cloud SDK
- Creating a new project
- Initialize App Engine app
- Enable billing for the project
- Downloading and installing Git
- Install the App Engine extension for Python 3
- Download the Hello World app written with **Flask**
- Run Hello World on your local machine
- Deploy and run Hello World on App Engine
- Clean-up to stop billing

3. Download and install the **Google Cloud SDK** “latest” version for your platform (Mac OS, Windows) from:

<https://cloud.google.com/sdk/docs/>

4. Under “Guides” click on **Installing the gcloud CLI**.

5. Select the package for your OS platform and extract the archive file on your local file system.
6. (Optional) Add the Cloud SDK tools to your PATH. Run the script (from the root of the folder you extracted in the last step) using this command:

```
./google-cloud-sdk/install.sh
```

```

marcopapa -- zsh - 121x98
[(env) marcopapa@Mac-mini-2 ~ % ./google-cloud-sdk/install.sh
Welcome to the Google Cloud CLI!

To help improve the quality of this product, we collect anonymized usage data
and anonymized stacktraces when crashes are encountered; additional information
is available at <https://cloud.google.com/sdk/usage-statistics>. This data is
handled in accordance with our privacy policy
<https://cloud.google.com/terms/cloud-privacy-notice>. You may choose to opt in to this
collection now (by choosing 'Y' at the below prompt), or at any time in the
future by running the following command:

    gcloud config set disable_usage_reporting false

Do you want to help improve the Google Cloud CLI (y/N)? N

Your current Google Cloud CLI version is: 434.0.0
The latest available version is: 434.0.0

Components
+-----+-----+-----+
| Status | Name | ID | Size |
+-----+-----+-----+
| Not Installed | App Engine Go Extensions | app-engine-go | 4.4 MiB |
| Not Installed | Appctl | appctl | 18.5 MiB |
| Not Installed | Artifact Registry Go Module Package Helper | package-go-module | < 1 MiB |
| Not Installed | Cloud Bigtable Command Line Tool | cbt | 11.2 MiB |
| Not Installed | Cloud Bigtable Emulator | bigtable | 6.9 MiB |
| Not Installed | Cloud Datastore Emulator | cloud-datastore-emulator | 35.1 MiB |
| Not Installed | Cloud Firestore Emulator | cloud-firestore-emulator | 42.4 MiB |
| Not Installed | Cloud Pub/Sub Emulator | pubsub-emulator | 62.6 MiB |
| Not Installed | Cloud Run Proxy | cloud-run-proxy | 11.8 MiB |
| Not Installed | Cloud SQL Proxy | cloud_sql_proxy | 7.6 MiB |
| Not Installed | Google Container Registry's Docker credential helper | docker-credential-gcr | 2.2 MiB |
| Not Installed | Kustomize | kustomize | 7.6 MiB |
| Not Installed | Log Streaming | log-streaming | 12.3 MiB |
| Not Installed | Minikube | minikube | 33.2 MiB |
| Not Installed | Nomos CLI | nomos | 25.8 MiB |
| Not Installed | On-Demand Scanning API extraction helper | local-extract | 14.0 MiB |
| Not Installed | Skaffold | skaffold | 24.1 MiB |
| Not Installed | Terraform Tools | terraform-tools | 62.1 MiB |
| Not Installed | anthos-auth | anthos-auth | 20.2 MiB |
| Not Installed | config-connector | config-connector | 57.1 MiB |
| Not Installed | enterprise-certificate-proxy | enterprise-certificate-proxy | 6.7 MiB |
| Not Installed | gcloud Alpha Commands | alpha | < 1 MiB |
| Not Installed | gcloud Beta Commands | beta | < 1 MiB |
| Not Installed | gcloud app Java Extensions | app-engine-java | 64.6 MiB |
| Not Installed | gcloud app PHP Extensions | app-engine-php | 21.9 MiB |
| Not Installed | gcloud app Python Extensions | app-engine-python | 8.5 MiB |
| Not Installed | gcloud app Python Extensions (Extra Libraries) | app-engine-python-extras | 27.3 MiB |
| Not Installed | gke-gcloud-auth-plugin | gke-gcloud-auth-plugin | 7.5 MiB |
| Not Installed | kpt | kpt | 14.6 MiB |
| Not Installed | kubectl | kubectl | < 1 MiB |
| Not Installed | kubectl-oidc | kubectl-oidc | 20.2 MiB |
| Not Installed | pkg | pkg | 1.6 MiB |
| Installed | BigQuery Command Line Tool | bq | 11.3 MiB |
| Installed | Cloud Storage Command Line Tool | gsutil | 20.7 MiB |
| Installed | Google Cloud CLI Core Libraries | core | 1.2 MiB |
| Installed | Google Cloud CRC32C Hash Tool | gcloud-crc32c |  |

To install or remove components at your current SDK version [434.0.0], run:
$ gcloud components install COMPONENT_ID
$ gcloud components remove COMPONENT_ID

To update your SDK installation to the latest version [434.0.0], run:
$ gcloud components update

To take a quick anonymous survey, run:
$ gcloud survey

Modify profile to update your $PATH and enable shell command completion?

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

The Google Cloud SDK installer will now prompt you to update an rc file to bring the Google Cloud CLIs into your environment.

Enter a path to an rc file to update, or leave blank to use [/Users/marcopapa/.zshrc]:
Backing up [/Users/marcopapa/.zshrc] to [/Users/marcopapa/.zshrc.backup].
[/Users/marcopapa/.zshrc] has been updated.

==> Start a new shell for the changes to take effect.

Google Cloud CLI works best with Python 3.7 and certain modules.

Download and run Python 3.7 installer? (Y/n)? n

For more information on how to get started, please visit:
https://cloud.google.com/sdk/docs/quickstarts

(env) marcopapa@Mac-mini-2 ~ %

```

7. Run a command to install the cloud component that includes the App Engine extension for Python:

```
gcloud components install app-engine-python
```

The screenshot shows a terminal window titled "marcopapa — zsh — 89x41". The command "(env) marcopapa@Mac-mini-2 ~ % gcloud components install app-engine-python" is entered. The output shows the current Google Cloud CLI version (434.0.0) and the components being installed (Cloud Datastore Emulator, gRPC Python library, gcloud app Python Extensions). A table provides details on the components, including their names, versions, and sizes. The user is prompted to continue (Y/n) and chooses Y. The process then begins, showing steps like creating a staging area, installing components, and performing post-processing. Finally, the update is completed.

```
(env) marcopapa@Mac-mini-2 ~ % gcloud components install app-engine-python

Your current Google Cloud CLI version is: 434.0.0
Installing components from version: 434.0.0

These components will be installed.

+-----+-----+-----+
| Name | Version | Size   |
+-----+-----+-----+
| Cloud Datastore Emulator | 2.3.0   | 35.1 MiB |
| gRPC Python library      | 1.20.0  | 1.9 MiB  |
| gcloud app Python Extensions | 1.9.104 | 8.5 MiB  |
+-----+-----+-----+

For the latest full release notes, please visit:
https://cloud.google.com/sdk/release_notes

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

+-----+-----+-----+
| Creating update staging area |
| Installing: Cloud Datastore Emulator |
| Installing: gRPC Python library |
| Installing: gRPC Python library |
| Installing: gcloud app Python Extensions |
| Creating backup and activating new installation |
+-----+-----+-----+

Performing post processing steps...done.

Update done!
(env) marcopapa@Mac-mini-2 ~ %
```

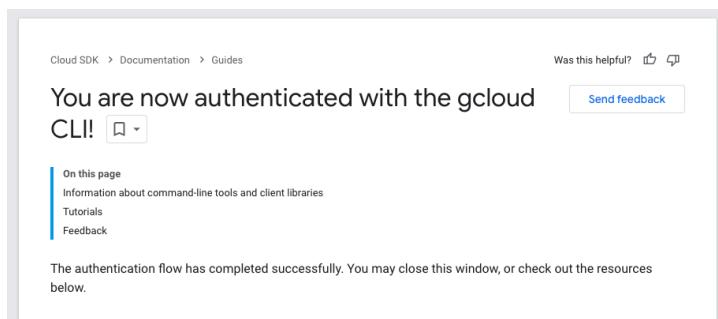
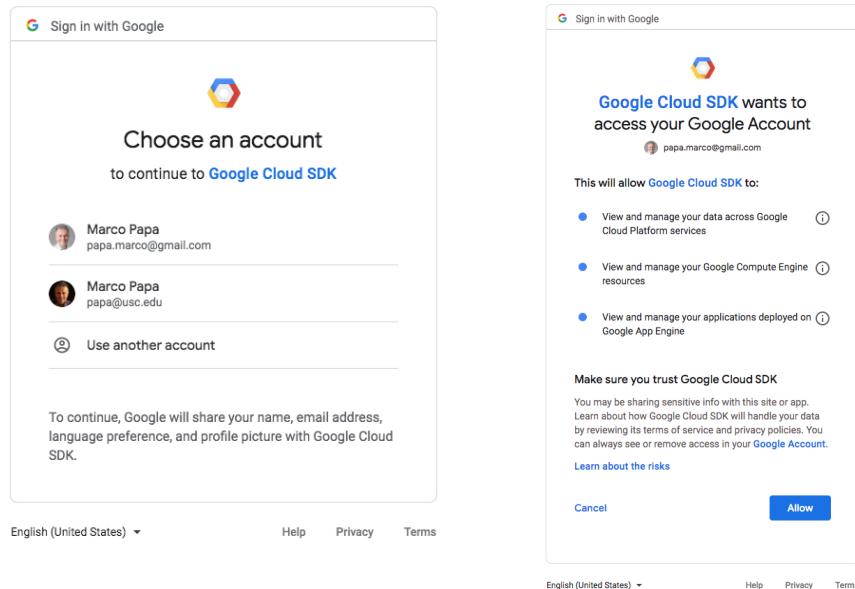
8. Initialize the `gcloud` tool to initialize the SDK:

```
gcloud init
```

You will be asked to “log in (Y/n)?” Answer Y.

You will be asked to “Pick a current project” or “**Create a new project**”. Select “create” and when asked, enter a name as appropriate.

You will be asked to **Choose an account** and **Allow** access as shown below.



```

marcopapa@Mac-mini ~ % gcloud init
Welcome! This command will take you through the configuration of gcloud.

Your current configuration has been set to: [default]

You can skip diagnostics next time by using the following flag:
  gcloud init --skip-diagnostics

Network diagnostic detects and fixes local network connection issues.
Checking network connection...done.
Reachability Check passed.
Network diagnostic passed (1/1 checks passed).

You must log in to continue. Would you like to log in (Y/n)? Y

Your browser has been opened to visit:

  https://accounts.google.com/o/oauth2/auth?response_type=code&client_id=32555940569.appspotusercontent.com&redirect_uri=http://localhost:32808/&scopeopenidhttps://www.googleapis.com/auth/userinfo.email+https://www.googleapis.com/auth/cloud-platform+https://www.googleapis.com/auth/m2m2ffppengine.admin+https://www.googleapis.com/auth/accounts.readonly&state=ZQ8yAWFzoS7VxMKnzrVC1s4SCZejH&access_type=offline&code_challenge=AxQc5-vFvkQkjIUuWzNejac4sv0Wj_V49hiwrjpHAA&code_challenge_method=S256

You are logged in as: [marcopapa@gmail.com].

Pick cloud project to use:
[1] cscis571-lamp
[2] facultyinstitute-174920
[3] lunar-box-281109
[4] my-python-project-94270
[5] myfirstpython-94534
[6] myfunction&project-3888
[7] quixotic-dynamo-165616
[8] Create a new project
Please enter numeric choice or text value (must exactly match list
item): 8

Enter a Project ID. Note that a Project ID CANNOT be changed later.
Project IDs must be 6-30 characters (lowercase ASCII, digits, or
hyphens) in length and start with a lowercase letter. second-python-745309
Waiting for [operations/cp.731480934088499568] to finish.

Your current project has been set to: [second-python-745309].

Not setting default zone/region (this feature makes it easier to use
[gcloud compute] by setting an appropriate default value for the
--zone and --region flags.
See https://cloud.google.com/compute/docs/gcloud-compute section on how to set
default compute region and zone manually. If you would like [gcloud init] to be
able to do this for you the next time you run it, make sure the
Compute Engine API is enabled for your project on the
https://console.developers.google.com/apis page.

Created a default .boto configuration file at [/Users/marcopapa/.boto]. See this file and
[https://cloud.google.com/storage/docs/gsutil/commands/config] for more
information about configuring Google Cloud Storage.
Your Google Cloud SDK is configured and ready to use!

* Commands that require authentication will use marcopapa@gmail.com by default
* Commands will reference project 'second-python-745309' by default
Run 'gcloud help config' to learn how to change individual settings

This gcloud configuration is called [default]. You can create additional configurations if you work with multiple accounts a
nd/or projects.
Run 'gcloud topic configurations' to learn more.

Some things to try next:
* Run 'gcloud --help' to see the Cloud Platform services you can interact with. And run 'gcloud help COMMAND' to get help on
any gcloud command.
* Run 'gcloud topic --help' to learn about advanced features of the SDK like arg files and output formatting
marcopapa@Mac-mini ~ %

```

Notice that Project IDs must start with a lowercase letter and can have lowercase ASCII letters, digits, or hyphens. Project IDs must be between 6 and 30 characters. For example:

myfirstpython-94534

9. Verify the project was created and see its details:

gcloud projects describe myfirstpython-94534

For example, you'll see something like this:

```

createTime: '2020-01-08T18:34:36.846Z'
lifecycleState: ACTIVE
name: myfirstpython-94534
projectId: myfirstpython-94534
projectNumber: '675437181434'

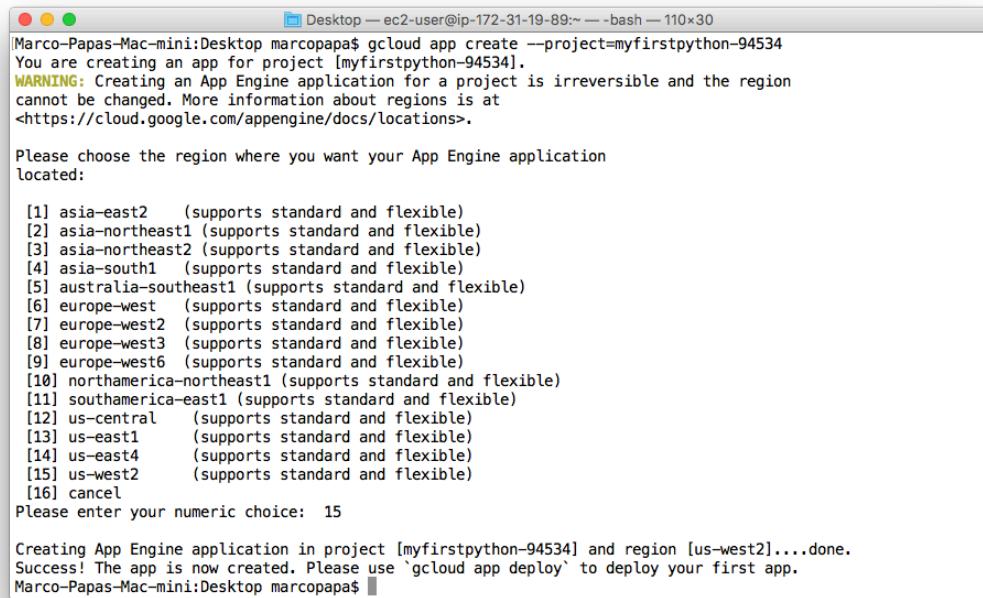
```

10. Initialize the App Engine app with your newly created project and choose its region (for example **us-west2**):

```
gcloud app create --project=[YOUR_PROJECT_ID]
```

for example:

```
gcloud app create --project=myfirstpython-94534
```



The terminal window shows the command `gcloud app create --project=myfirstpython-94534` being run. It displays a warning about the不可逆性 of choosing a region, followed by a list of available regions with their descriptions. The user is prompted to enter a numeric choice, and after selecting `15` (us-west2), the process completes successfully.

```
[Marco-Papas-Mac-mini:Desktop marcopapa$ gcloud app create --project=myfirstpython-94534
You are creating an app for project [myfirstpython-94534].
WARNING: Creating an App Engine application for a project is irreversible and the region
cannot be changed. More information about regions is at
<https://cloud.google.com/appengine/docs/locations>.

Please choose the region where you want your App Engine application
located:

[1] asia-east2      (supports standard and flexible)
[2] asia-northeast1 (supports standard and flexible)
[3] asia-northeast2 (supports standard and flexible)
[4] asia-south1     (supports standard and flexible)
[5] australia-southeast1 (supports standard and flexible)
[6] europe-west     (supports standard and flexible)
[7] europe-west2    (supports standard and flexible)
[8] europe-west3    (supports standard and flexible)
[9] europe-west6    (supports standard and flexible)
[10] northamerica-northeast1 (supports standard and flexible)
[11] southamerica-east1 (supports standard and flexible)
[12] us-central      (supports standard and flexible)
[13] us-east1        (supports standard and flexible)
[14] us-east4        (supports standard and flexible)
[15] us-west2        (supports standard and flexible)
[16] cancel

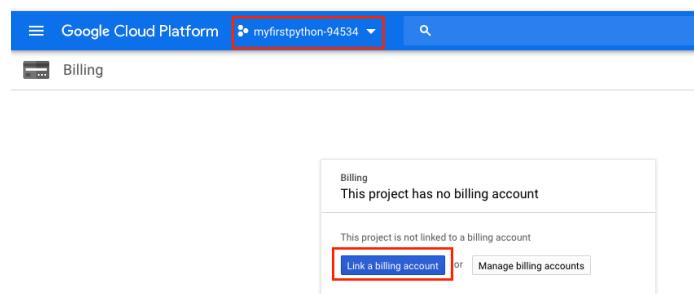
Please enter your numeric choice: 15

Creating App Engine application in project [myfirstpython-94534] and region [us-west2]....done.
Success! The app is now created. Please use `gcloud app deploy` to deploy your first app.
Marco-Papas-Mac-mini:Desktop marcopapa$ ]
```

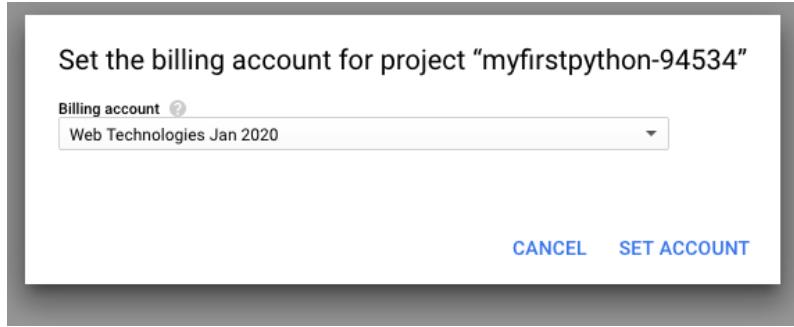
11. **Enable billing** for the project. You will do this in the Cloud console at:

<https://console.cloud.google.com/projectselector/billing?lang=python3>

You will have to select the project and click **Link a billing account**.



Select the billing account you created with your Google credits.



12. Install Git

13. Clone the **Google Cloud Platform Python Samples** repository from Github:

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

```
Marco-Papas-Mac-mini:Desktop marcopapas$ git clone https://github.com/GoogleCloudPlatform/python-docs-samples
Cloning into 'python-docs-samples'...
remote: Enumerating objects: 28, done.
remote: Counting objects: 100% (28/28), done.
remote: Compressing objects: 100% (17/17), done.
remote: Total 25697 (delta 10), reused 15 (delta 10), pack-reused 25669
Receiving objects: 100% (25697/25697) 39.97 MiB | 535.00 Kib/s, done.
Resolving deltas: 100% (13614/13614), done.
Marco-Papas-Mac-mini:Desktop marcopapas$ cd python-docs-samples/appengine/standard_python37/hello_world
Marco-Papas-Mac-mini:hello_world marcopapas$
```

14. Test the **Hello World** app on your local machine:

- Windows ONLY:** download and install **PowerShell** as indicated in the tutorial.
- Create an isolated Python environment.

macOS/Linux:

```
pip install virtualenv
virtualenv <your-env>
source <your-env>/bin/activate
```

Windows:

```
pip install virtualenv
virtualenv <your-env>
<your-env>\Scripts\activate
```

c. Change to the directory that contains the **Hello World** sample code:

```
cd python-docs-
samples/appengine/standard_python3/hello_world
```

d. If the file requirements.txt does not exists, create it with:

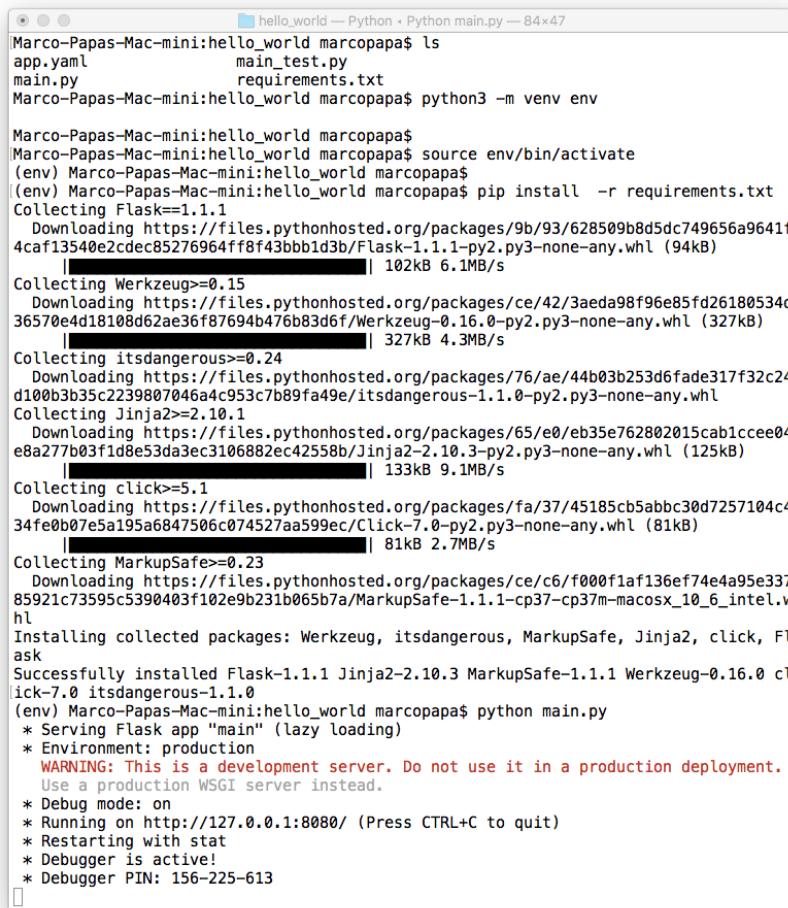
```
pip freeze > requirements.txt
```

e. Install dependencies (this step will install Flask):

```
pip install -r requirements.txt
```

f. Run the application:

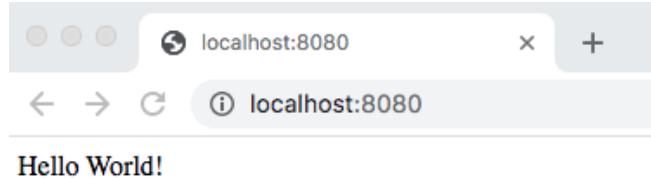
```
python main.py
```



```
Marco-Papas-Mac-mini:hello_world marcopapa$ ls
app.yaml          main_test.py
main.py           requirements.txt
Marco-Papas-Mac-mini:hello_world marcopapa$ python3 -m venv env
Marco-Papas-Mac-mini:hello_world marcopapa$ source env/bin/activate
(env) Marco-Papas-Mac-mini:hello_world marcopapa$
(env) Marco-Papas-Mac-mini:hello_world marcopapa$ pip install -r requirements.txt
Collecting Flask==1.1.1
  Downloading https://files.pythonhosted.org/packages/9b/93/628509b8d5dc749656a9641f
4caf13540e2cdec85276964ff8f43bbb1d3b/Flask-1.1.1-py2.py3-none-any.whl (94kB)
|██████████| 102kB 6.1MB/s
Collecting Werkzeug>=0.15
  Downloading https://files.pythonhosted.org/packages/ce/42/3aeda98f96e85fd26180534d
36570e4d18108d62ae36fb87694b476b83d6f/Werkzeug-0.16.0-py2.py3-none-any.whl (327kB)
|██████████| 327kB 4.3MB/s
Collecting itsdangerous>=0.24
  Downloading https://files.pythonhosted.org/packages/76/ae/44b03b253d6fade317f32c24
d100b3b35c2239807046aa4c953c7b89fa49e/itsdangerous-1.1.0-py2.py3-none-any.whl
Collecting Jinja2>=2.10.1
  Downloading https://files.pythonhosted.org/packages/65/e0/eb35e762802015cab1cce04
e8a277d03f1d8e53da3ec3106882ec42558b/Jinja2-2.10.3-py2.py3-none-any.whl (125kB)
|██████████| 133kB 9.1MB/s
Collecting click>=5.1
  Downloading https://files.pythonhosted.org/packages/fa/37/45185cb5abbc30d7257104c4
34fe0b07e5a195a6847506c074527aa599ec/Click-7.0-py2.py3-none-any.whl (81kB)
|██████████| 81kB 2.7MB/s
Collecting MarkupSafe>=0.23
  Downloading https://files.pythonhosted.org/packages/ce/c6/f000f1af136ef74e4a95e337
85921c73595c5390403f102e9b231b065b7a/MarkupSafe-1.1.1-cp37-cp37m-macosx_10_6_intel.w
hl
Installing collected packages: Werkzeug, itsdangerous, MarkupSafe, Jinja2, click, Fl
ask
Successfully installed Flask-1.1.1 Jinja2-2.10.3 MarkupSafe-1.1.1 Werkzeug-0.16.0 cl
ick-7.0 itsdangerous-1.1.0
(env) Marco-Papas-Mac-mini:hello_world marcopapa$ python main.py
 * Serving Flask app "main" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Running on http://127.0.0.1:8080/ (Press CTRL+C to quit)
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 156-225-613
```

g. Open the app in your browser

<http://localhost:8080>



Type CTRL-C to quit serving locally the Flask app.

15. Deploy and run Hello World on App Engine:

```
gcloud app deploy
```

```

pa@Mac-mini-2 hello_world % gcloud app deploy
Services to deploy:
descriptor: [/Users/marcopapa/python-docs-samples/appengine/standard_python3/hello_world/app.yaml]
source: [/Users/marcopapa/python-docs-samples/appengine/standard_python3/hello_world]
target project: [my-python-project-67167]
target service: [default]
target version: [20230609t213445]
target url: [https://my-python-project-67167.wl.r.appspot.com]
target service account: [my-python-project-67167@appspot.gserviceaccount.com]

Do you want to continue (Y/n)? Y
Beginning deployment of service [default]...
Created .gcloudignore file. See `gcloud topic gcloudignore` for details.
Uploading 6 files to Google Cloud Storage
File upload done.
Updating service [default]...done.
Setting traffic split for service [default]...done.
Deployed service [default] to [https://my-python-project-67167.wl.r.appspot.com]

You can stream logs from the command line by running:
$ gcloud app logs tail -s default

To view your application in the web browser run:
$ gcloud app browse
(env) marcopapa@Mac-mini-2 hello_world %

```

16. View your application in the cloud. Launch your browser with the app at [http://\[YOUR PROJECT ID\].\[REGION ID\].r.appspot.com](http://[YOUR PROJECT ID].[REGION ID].r.appspot.com),

running the command:

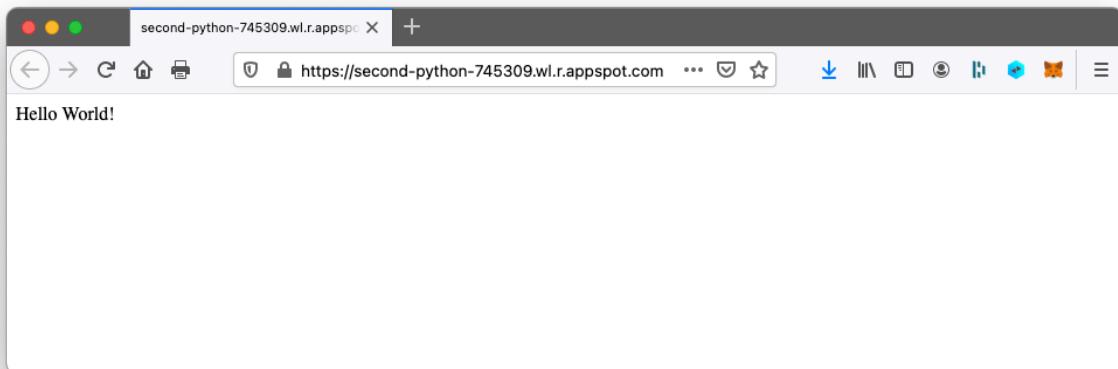
```
gcloud app browse
```

Or type the URL in the browser:

<https://myfirstpython-94534.us-west2.r.appspot.com/>

or

<https://second-python-745309.wl.r.appspot.com/>



17. Clean up. First stop using the virtual environment. Type this to the (env) prompt:

```
deactivate
```

18. To avoid incurring charges, **delete your Cloud Platform project** to stop billing on all resources.

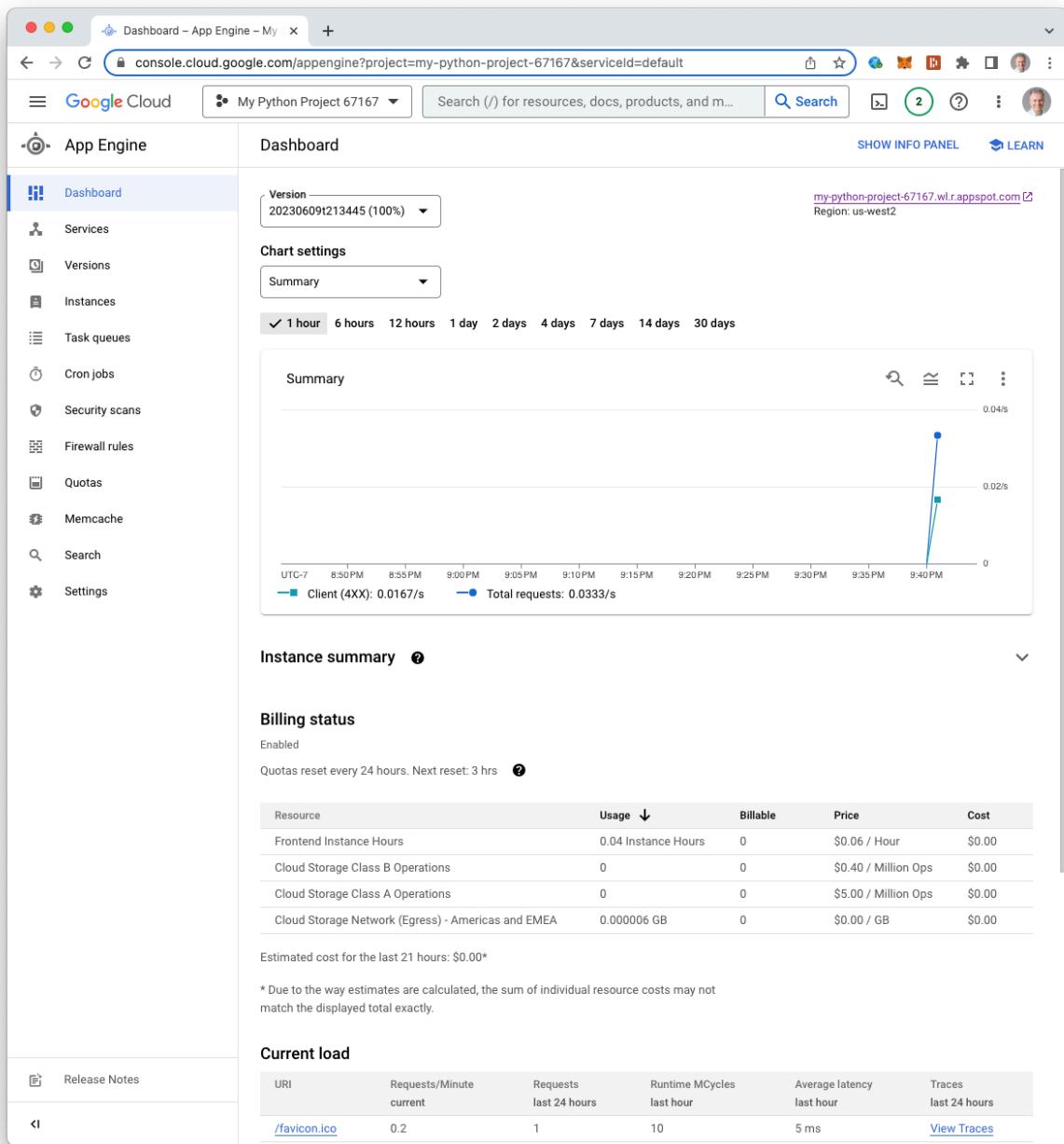
19. The “Hello World” program comes with a file named `requirements.txt`. This file needs to be deployed to GCP. Once you add your code and add some Python libraries, this file needs to be updated. Use pip to install your libraries locally. Then run the following pip command:

```
pip freeze > requirements.txt
```

This pip command will update the `requirements.txt` file with all the needed libraries. The local Python libraries should not be uploaded and deployed to GCP. Instead, the libraries included in the deployed `requirements.txt` file will be automatically downloaded and installed by GCP. Every time you add a new library to your local copy, you need to run “`pip freeze`” before deploying to GCP.

4. Check App Engine Dashboard

Click on “triple bar” on top left of the GCP console. Select App Engine. Select your Project ID.



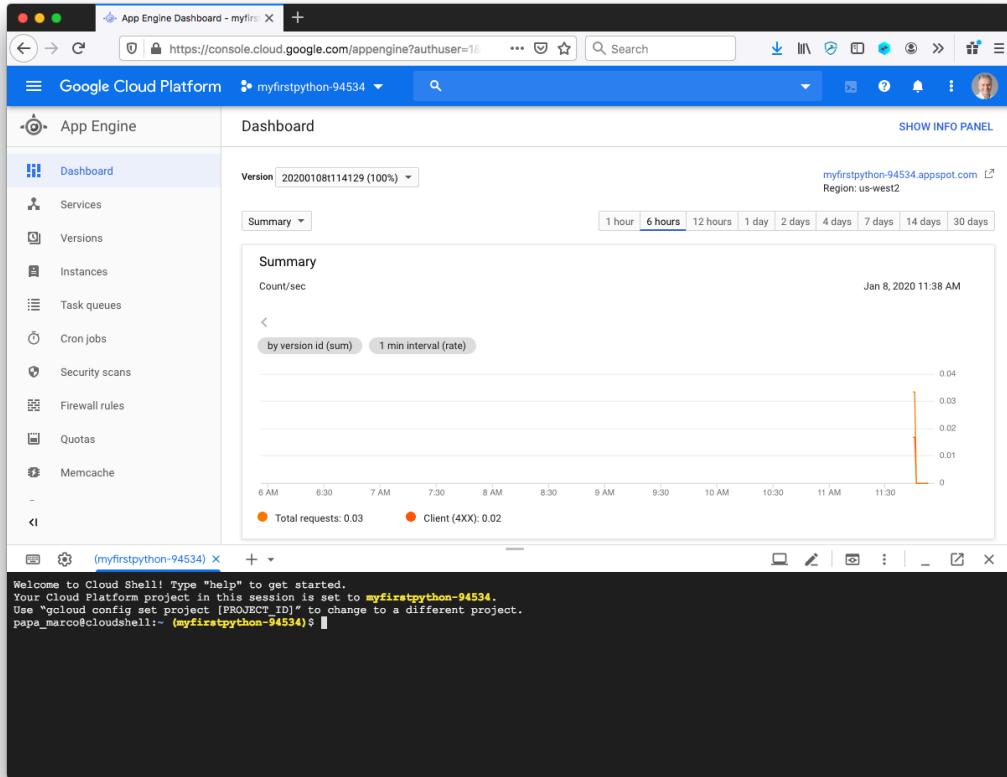
5. Set up Exploring Your instance (Optional)

If you want to explore your server instance you can activate the **Google Cloud Shell**.

Go to the App Engine Dashboard. Select the Hello World project from the dropdown. Now click on the **Activate Cloud Shell** icon in the top toolbar (see picture above).

After waiting a few minutes for Google to establish the connection, you will see the shell appear at the bottom of the browser window. You can now use Linux commands to

manage your Cloud Platform Console projects and resources.



You can read more about the **Google Cloud Shell** here:

<https://cloud.google.com/cloud-shell/docs/>

6. Monitoring your instance and you bill

Select Google Cloud Platform and go to the Dashboard. If you do not see a **Billing** “tile”, click **CUSTOMIZE** in the upper left toolbar. Turn on the billing tile “switch” and click **DONE**. Under **Billing** you will see if you are incurring any charges. You will likely see \$0.00 estimated charges.

The screenshot shows the Google Cloud Platform dashboard for the project 'myfirstpython-94534'. The dashboard is divided into several sections:

- Project info:** Displays the project name (myfirstpython-94534), project ID (myfirstpython-94534), and project number (675437181434). A red box highlights the 'CUSTOMIZE' button at the top right.
- Resources:** Shows App Engine (1 version) and Storage (3 buckets).
- Trace:** Indicates 'No trace data from the past 7 days'.
- Getting Started:** Provides links to explore APIs, deploy solutions, add dynamic logging, monitor errors, take a VM quickstart, create a Cloud Function, and install the Cloud SDK. A red box highlights the 'Billing' section.
- App Engine:** Monitors 'Summary (count/sec)' with a chart showing a sharp peak around 11:45 AM. The chart includes series for 'http/server/response_count: 0' (orange) and 'http/server/response_count: 0' (purple).
- API APIs:** Monitors 'Requests (requests/sec)' with a chart showing fluctuations. A blue dot indicates 'Requests: 0.017'.
- Google Cloud Platform status:** Shows 'All services normal' and a link to the Cloud status dashboard.
- Billing:** Displays 'Estimated charges' in USD \$0.00.
- Error Reporting:** States 'No sign of any errors. Have you set up Error Reporting?' and a link to learn how.
- News:** Features news items like 'Put your archive data on ice with new storage offering' (3 hours ago) and 'Google Cloud and FDA MyStudies: Harnessing real-world data for medical research' (1 day ago).
- Documentation:** Links to Learn about Compute Engine, Cloud Storage, and App Engine.

Have fun exploring Google Cloud Platform!!