Homework #7 Google Cloud Platform

This semester we are allowing all students to explore cloud computing as offered by the Google Cloud Platform. Using the instructions below one can establish a website using Google App Engine. Once established, you will be able to move your PHP program developed for Assignment #6 to your Google App Engine website and have it execute there.

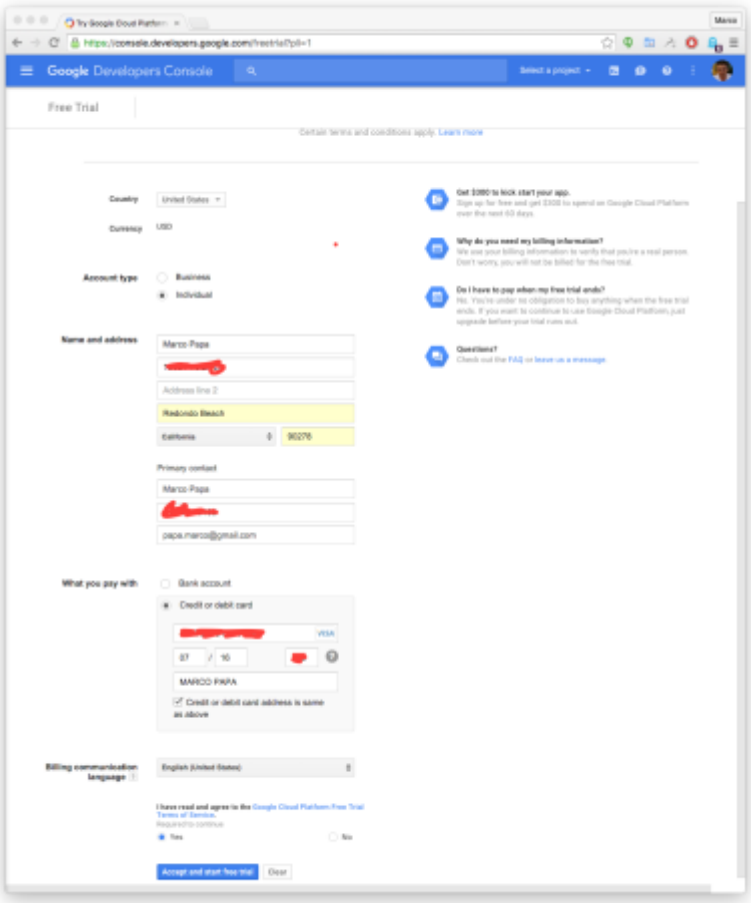
1. Sign up for Google Cloud Platform

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

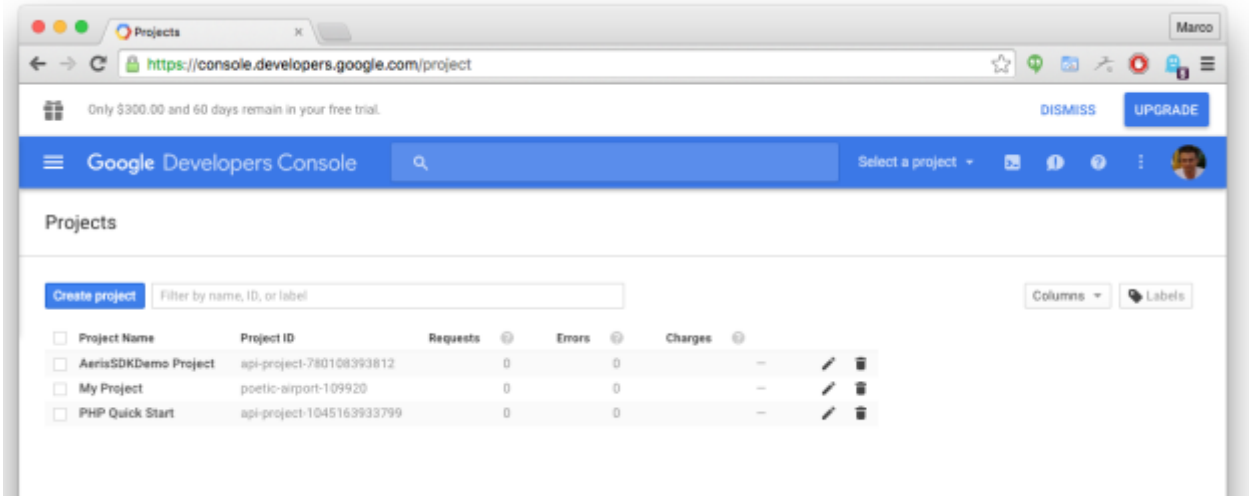
To sign up go to:

https://console.developers.google.com/freetrial?pli=1

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 Accept and start free trial.



After you are signed up, you will be redirected to the Projects section of the Google Developer Console.



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

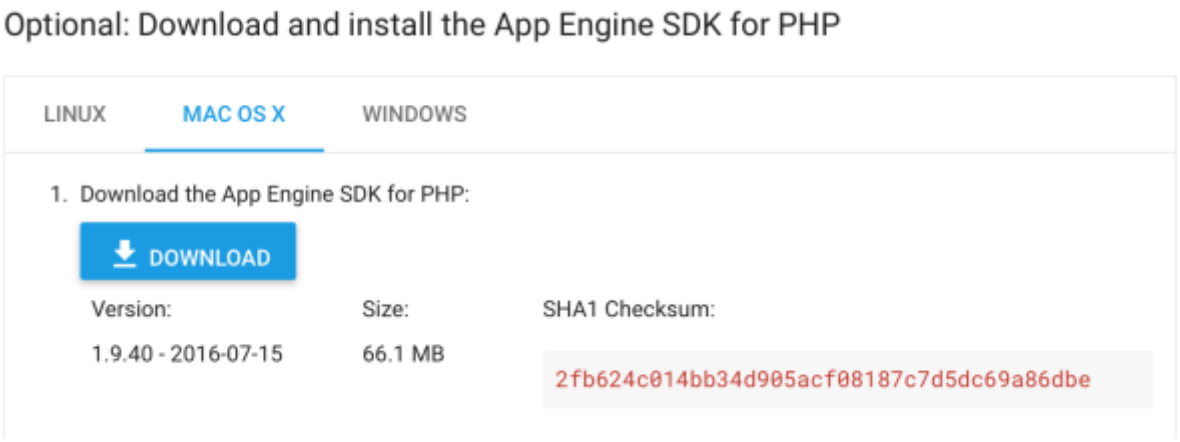
2. Download the App Engine SDK for PHP

The Google App Engine SDK for PHP is available for Windows, Mac OS X and Linux platforms.

Go to the URL:

https://cloud.google.com/appengine/downloads/#Google_App_Engine_SDK_for_PHP

Click on "Optionally, you can also download the original App Engine SDK for PHP."



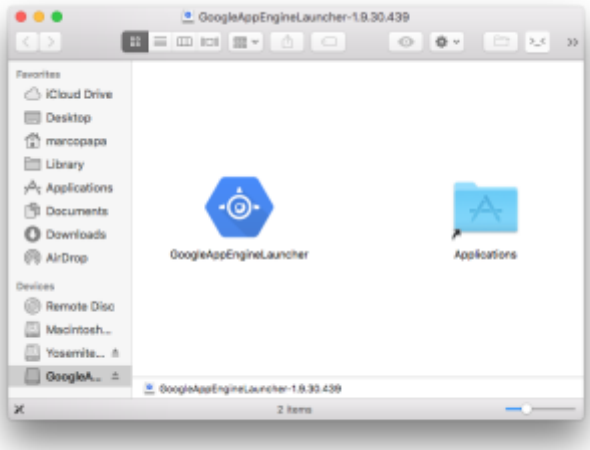
Download the package for your Platform to your local machine. The rest of this installation will show the steps needed for installing the SDK on the Mac OS X and Windows platforms.

3. Install the App Engine SDK for PHP

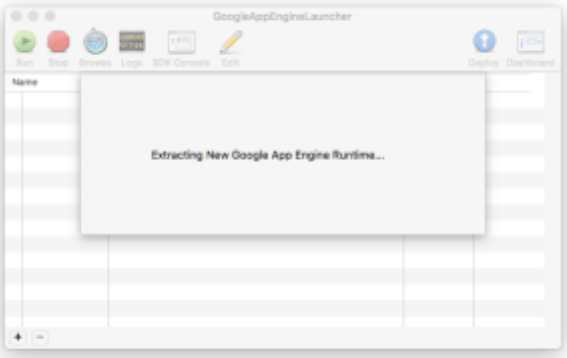
Installing on Mac OS X

To install the SDK on Mac OS X:

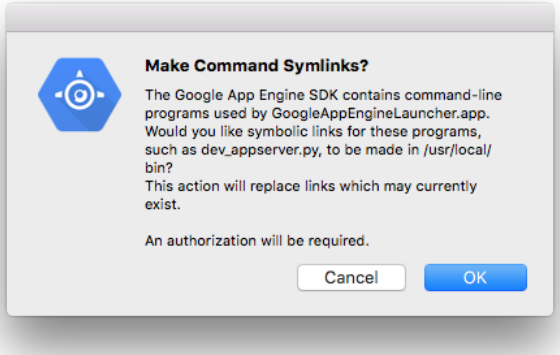
1. In the Finder, click Go > Applications to open the Applications folder.
2. Double click the App Engine SDK file you downloaded (GoogleAppEngineLauncher-1.9.31.dmg) to open it, then drag the GoogleAppEngineLauncher icon over to the Applications folder.



3. Double-click GoogleAppEngineLauncher in the Application folder.



4. When prompted to Make Command Symlinks, click OK. The symlinks allow you to run important SDK command-line tools in any terminal window.



> Important: The Google App Engine Launcher is a convenient UI-based tool for running and deploying App Engine apps, but it does not provide all the features you'll need. You will need to use the command-line equivalent, appcfg.py, for many of the features you'll want to use.

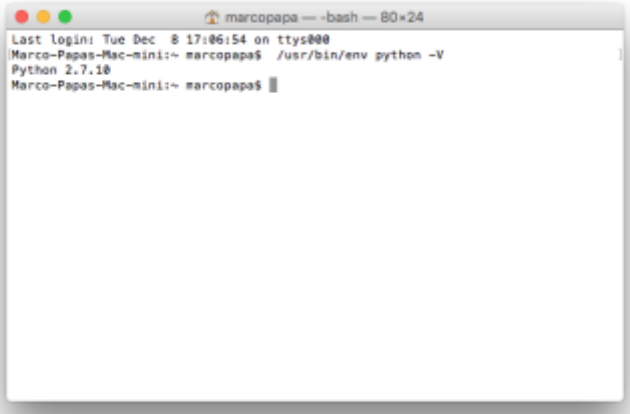
5. Notice that the installation process above unpacks the contents of the App Engine SDK at the location:

```
/usr/local/google_appengine
```

6. The App Engine PHP SDK requires Python 2.7, which is installed by default on Mac OS X 10.6 (Lion) or later. Verify your Mac's Python installation using the following command:

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

   If the output looks like Python 2.7.<number> then you already have the correct Python version installed. Otherwise you can download and install Python 2.7 from the Python web site. If you are using Mac OS X El Capitan, you will have Python 2.7.10, as shown below:



Installing on Windows

To install the SDK on Windows:

1. Double-click the SDK file you downloaded (GoogleAppEngine-1.9.31.msi) and follow the prompts to install the SDK.

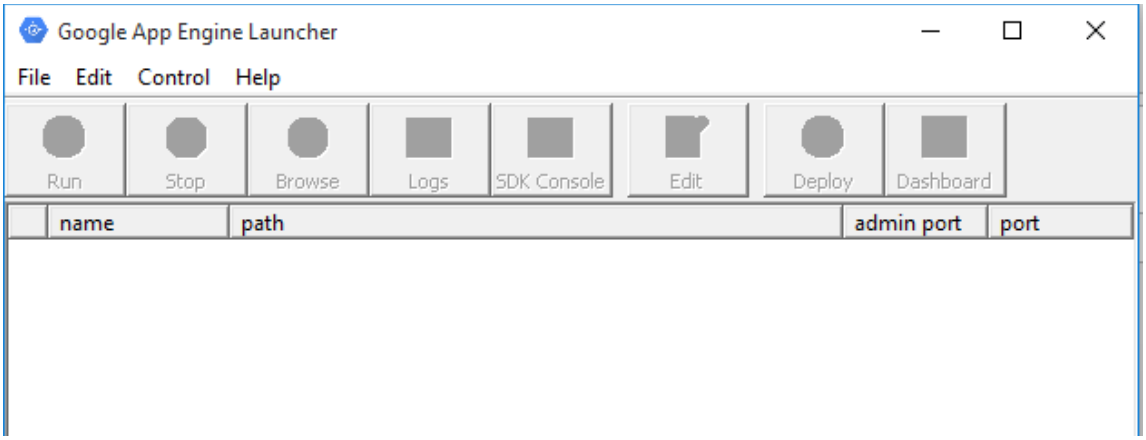
2. You will need Python 2.7 to use the App Engine PHP SDK, because the Development Server is a Python application. Download Python 2.7.X (don't use a higher version) from the Python web site.

> Note: The PHP SDK includes binaries for the PHP 5.4 runtime, including all enabled extensions, so there is no need to download PHP separately for the purposes of developing with App Engine -- you just need Python.

4. Create application using GoogleAppEngineLauncher

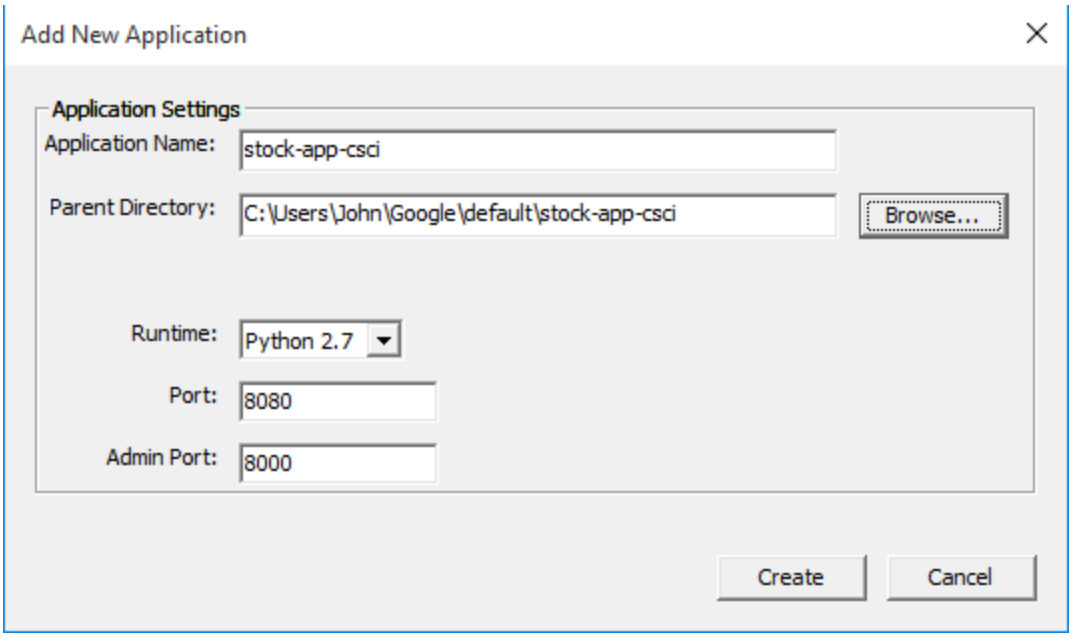
You can use the Google App Engine Launcher, which is installed as part of the App Engine SDK for PHP.

Run the Google App Engine Launcher:

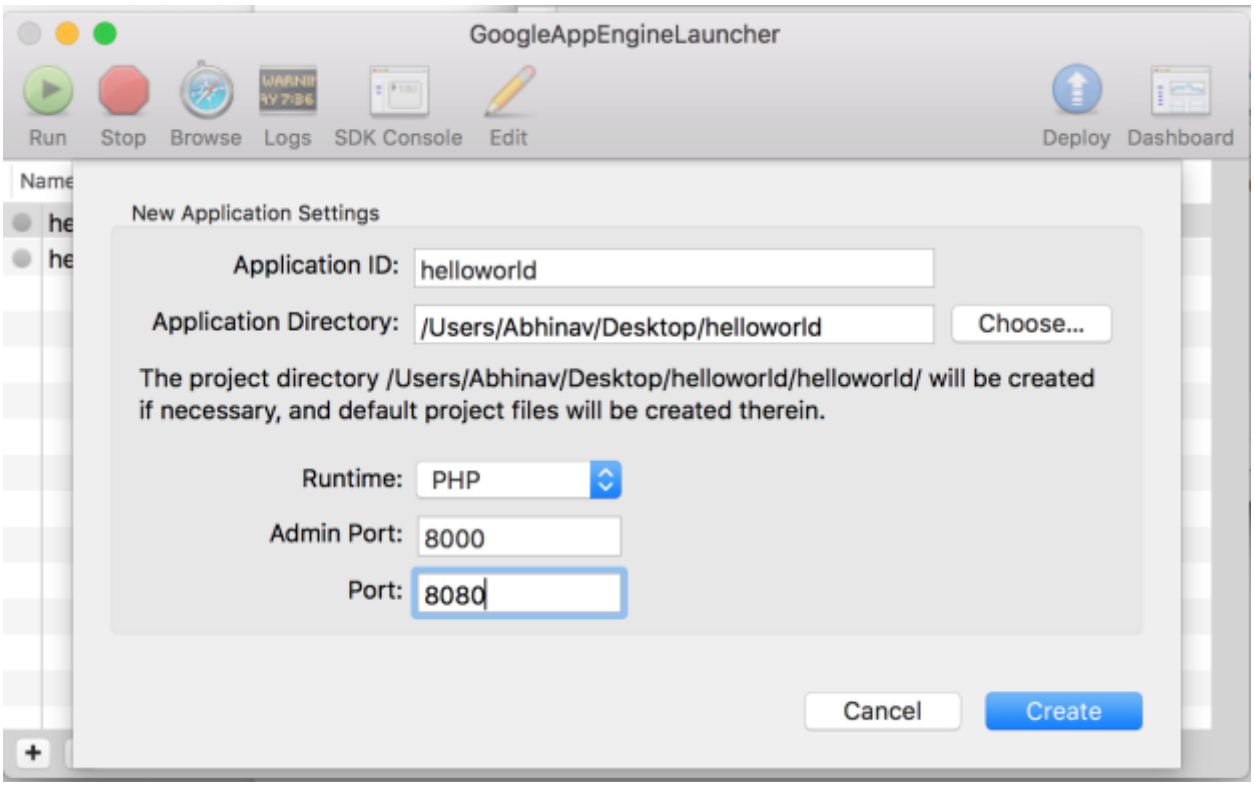


Google App Launcher (Windows)

Invoke File -> Create (Windows) or File -> New Application (OS X). Or use File -> Add Existing Application (OS X) if the app already has been created.



Add New Application (Windows)

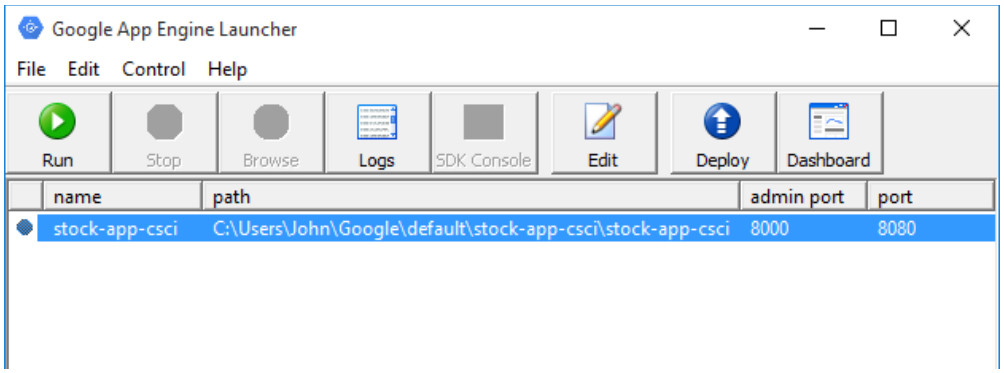


Add New Application (Mac)

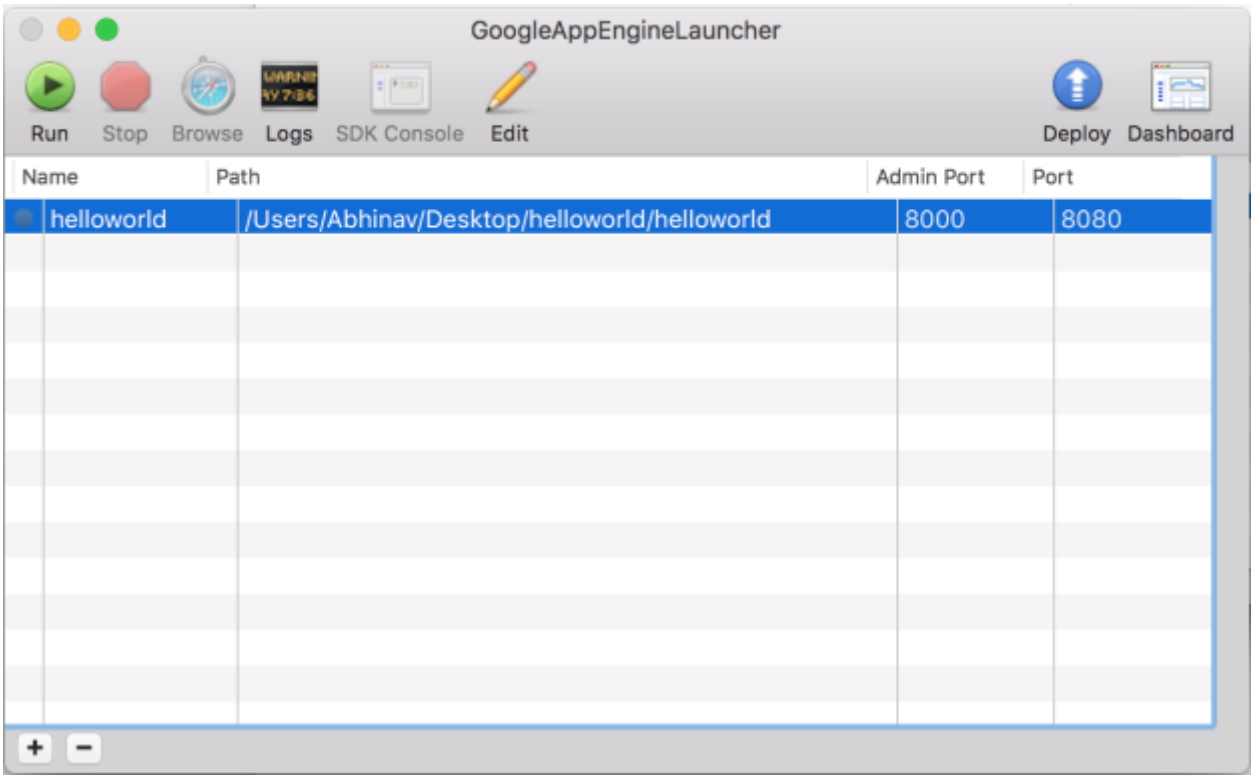
You will need to enter the following fields:

- Application Name or ID: use the app name created in Google Cloud
- Parent of Application Directory: use the full path, for example:
    - /Users/yourname/Desktop/helloworld in OS X
    - C:\Users\yourname\Desktop\helloworld in Windows
- Runtime: should be PHP
- Port: defaults to 8080
- Admin Port: defaults to 8000

Now you can run the application by clicking the Run button:



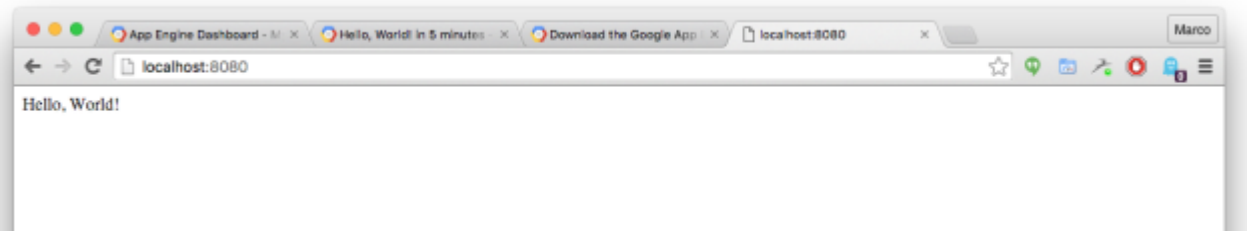
Run application locally (Windows)



Run Application locally (OS X)

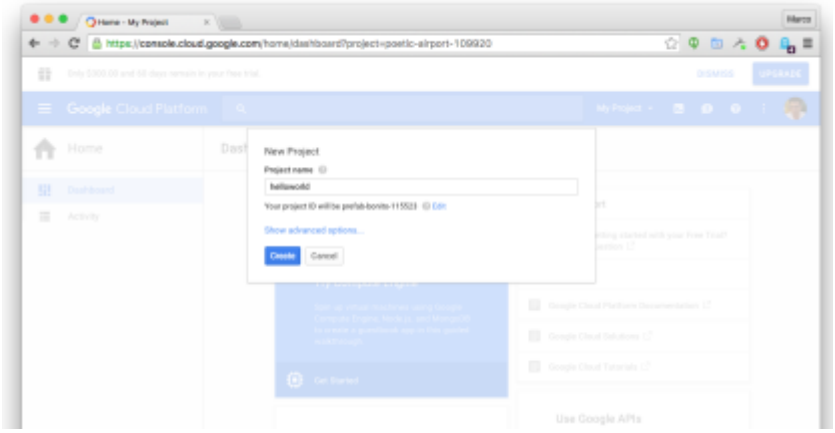
The web server is now running, listening for requests on port 8080. You can test the application by visiting the following URL in your web browser:

http://localhost:8080/

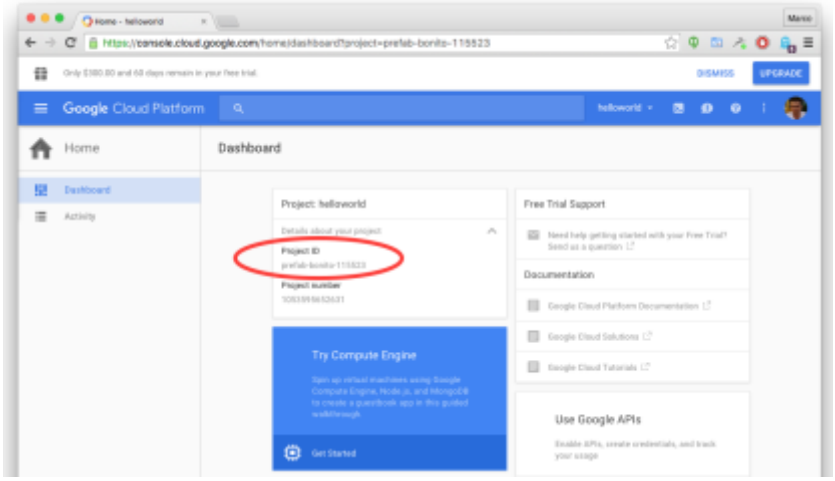


5. Uploading the Application

1. Sign in to App Engine using your Google account. If you do not have a Google account, you can create a Google account with an email address and password.
2. If you haven't already done so, create a project for your App Engine app as follows:
   a. Visit the Google Cloud Platform Console and click Create Project.
   b. Supply the desired project name in the New Project form. It doesn't have to match your app name, but using the same name as your app might make administration easier.



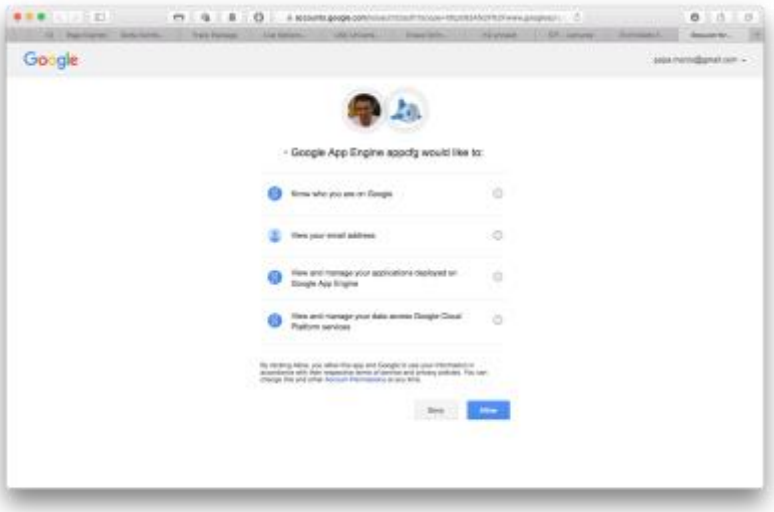
   c. Accept the generated project ID or supply your own ID. This project ID is used as the App Engine application ID. Note that this ID can only be used once: if you subsequently delete your project, you won't be able to re-use the ID in a new project.



3. Note the application ID (project ID) you created above.
4. Upload your finished application to Google App Engine by invoking the following command. This opens a browser window for you to sign in using your Google account. You'll be providing the project ID as the argument for -A.
   Note: Make sure you run this command in the directory where you created helloworld. Otherwise, give the full path to the helloworld folder.

```
sudo /usr/local/google_appengine/appcfg.py -A YOUR_PROJECT_ID update helloworld/
```

5. The first time you execute this script, you will be asked to Allow the appcfg.py script, as shown below:



6. You will receive a message in the browser that "The authentication flow has completed." Your app is now deployed and ready for use!

7. The full URL for your application is http://<YOUR_PROJECT_ID>.appspot.com/.

6. Check PHP Info

To find out the capability and installed options of the PHP component on Google Engine, you need to be able to execute the phpinfo() API. Unfortunately this feature is turned off by default on the Google Cloud Platform.

To enable phpinfo(), you need to create a file named php.ini inside the helloworld folder, containing the following:

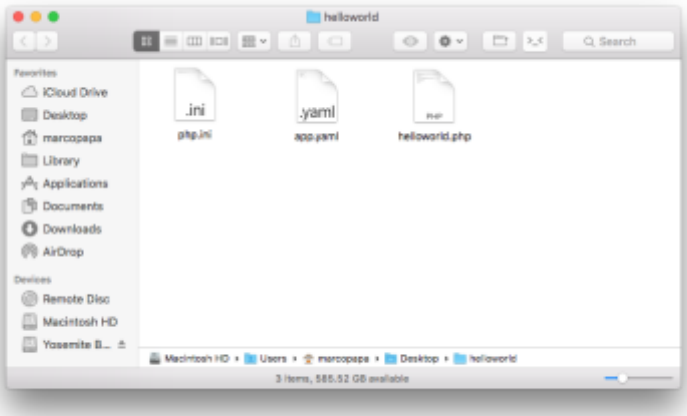
```
google_app_engine.enable_functions = "phpinfo, php_uname, php_sapi_name,
phpversion"

google_app_engine.enable_curl_lite = "1"
```

Then update your helloworld.php script with the code:

```
<html>
 <head>
 <title>PHP Test</title>
 <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
 </head>
 <body>
 <h1>PHP Test</h1>
 <p>
 <b>An Example of PHP in Action</b>
 <?php date_default_timezone_set('America/Los_Angeles');?>
 <?php echo "The Current Date and Time is: "; echo date("g:i A l, F j Y.");?>
 </p>
 <h2>PHP Information</h2>
 <p>
 <?php phpversion(); ?>
 <?php phpinfo(); ?>
 </p>
 </body>
 </html>
```

The helloworld folder will now contain 3 files, as shown below.



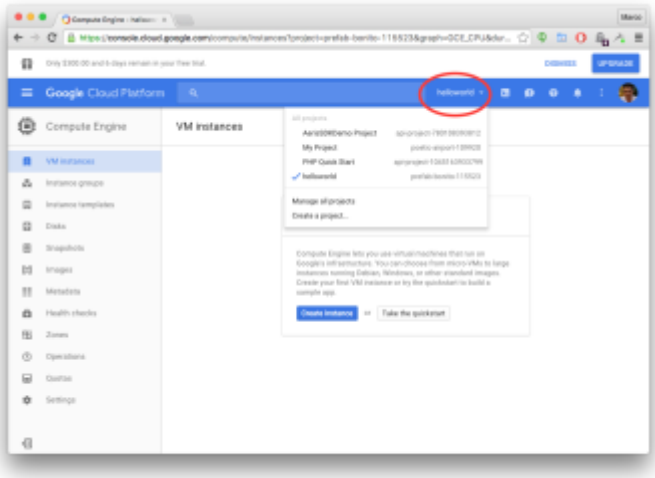
You can now "update" your app by invoking the appcfg.py script from section 5.4 above. If all goes well, you'll see the following new home page:

7. Set up Exploring Your instance (Optional)

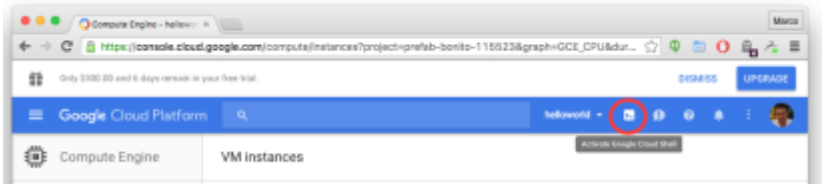
If you want to explore your server instance you can activate the Google Cloud Shell.

https://console.cloud.google.com/compute

Select the helloworld project from the dropdown.

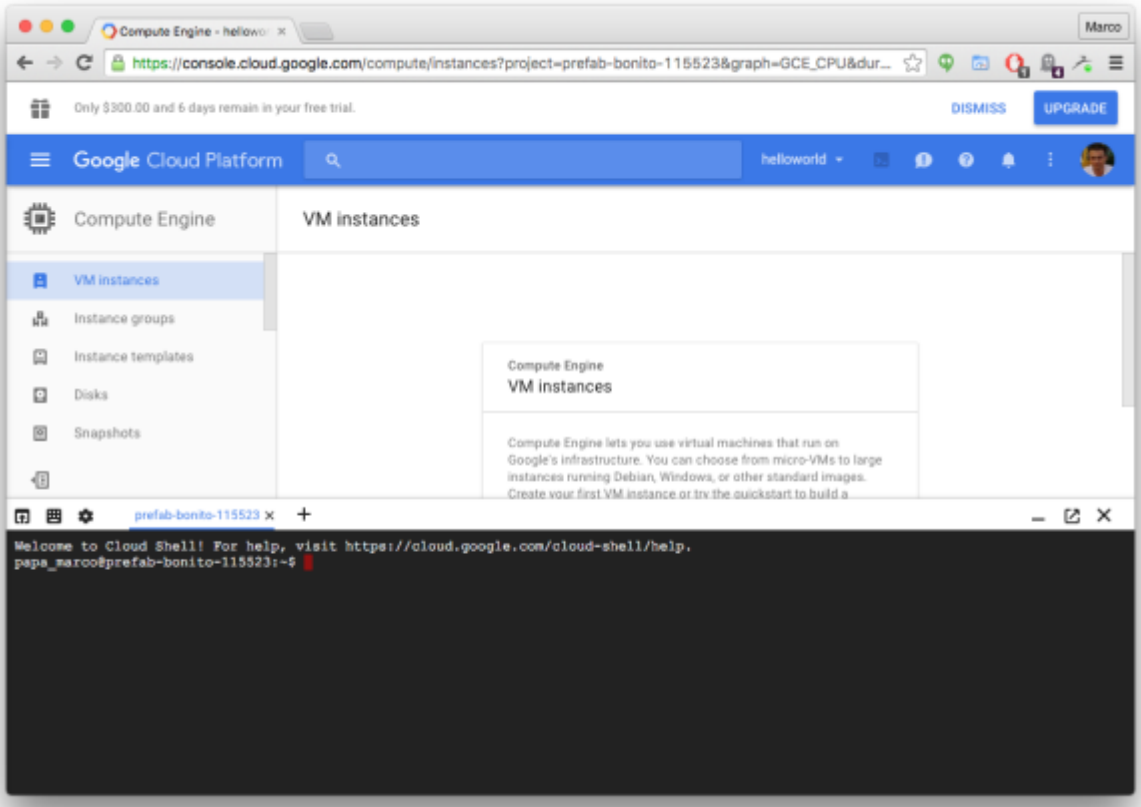


Now click on the Activate Google Cloud Shell icon next to helloworld.



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/

You can use Cloud Shell for free to the end of 2016

Have fun exploring Google Cloud Platform!!