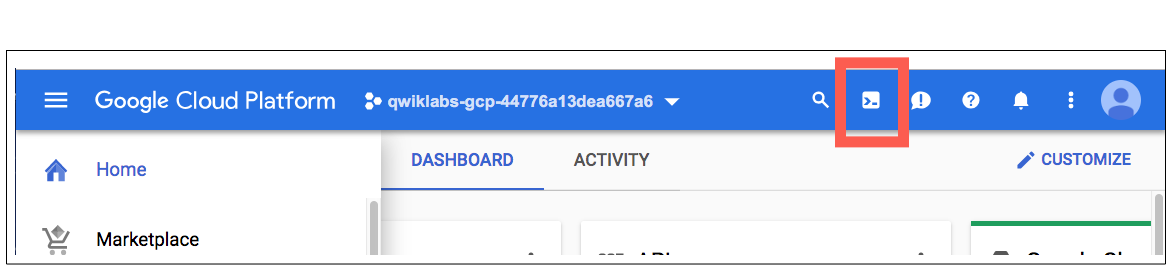
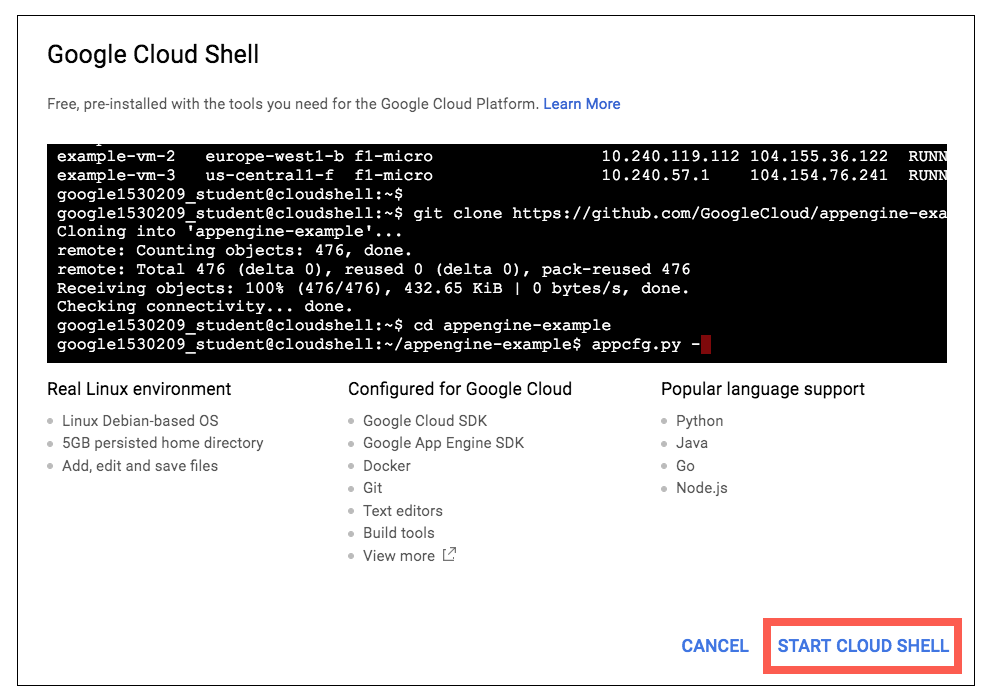
**1) Open Cloud Shell**

* Go to “[https://console.cloud.google.com](https://console.cloud.google.com/)”
* At GCP ribbon, find the symbol to open Cloud Shell



* Click the “START CLOUD SHELL” button to start the shell



* Type the following command:

**gcloud compute firewall-rules create default-allow-http-8080 \**

**--allow tcp:8080 \**

**--source-ranges 0.0.0.0/0 \**

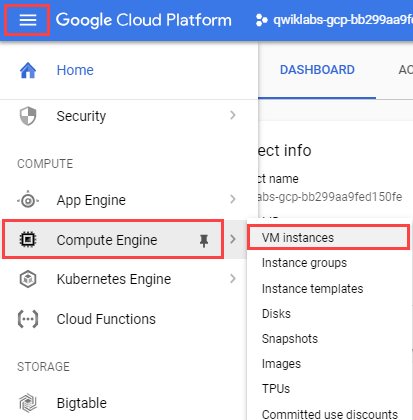
**--target-tags http-server \**

**--description "Allow port 8080 access to http-server"**

### 

### **2) Create a new Compute Engine:**

* On the Google Cloud Platform on the left menu, select “Compute Engine”



* Click “Create Instance”
* Type the name of the instance you would like.
* In Region, select “us-central1”. Leave the default value for zone.
* In Machine type, select “n1-standard-1”
* Make sure Boot disk is “Debian GNU/Linux 9”
* Click “Create”
* Once the machine started, write down the **external IP address** of the compute engine.
* Click “SSH” to open a shell interface to the compute engine.

### **3) Create a new virtual environment and install requirements:**

* Type the following commands:

**sudo apt-get install -y git**

**sudo apt-get install -y virtualenv**

**git clone** [**https://github.com/tadinve/HousePrice**](https://github.com/tadinve/HousePrice)

**cd HousePrice**

**virtualenv --python=python3.5 venv**

**source venv/bin/activate**

**pip install -r requirements.txt**

### **4) Test the environment**

* Edit the following file by typing

**nano controller.py**

* Search for “app.run(debug=True)” and change it to:

**app.run(host=”0.0.0.0”, port=8080, debug=True)**

* Type <ctrl>-O and hit <Enter> key
* Type <ctrl>-X

**python controller.py # it will load the model and start a web-server.**

* Visit http://<IP Address noted in step 2>:8080 with your browser and test. Example:

**http://10.10.10.2:8080**

* Explore the "controller.py" file. Can you figure out what it does?
* Explore the rest of the code. Can you figure out what the other files do?

### **5) Shut down**

* Go to VM and Hit <CTRL>-C to stop the python controller.py
* Type “deactivate”
* Type “exit”
* Go to GCP console and stop the virtual machine.