

Purpose

The purpose of this exercise is to control your instance using a Python program from the Cloud Shell

Preparation

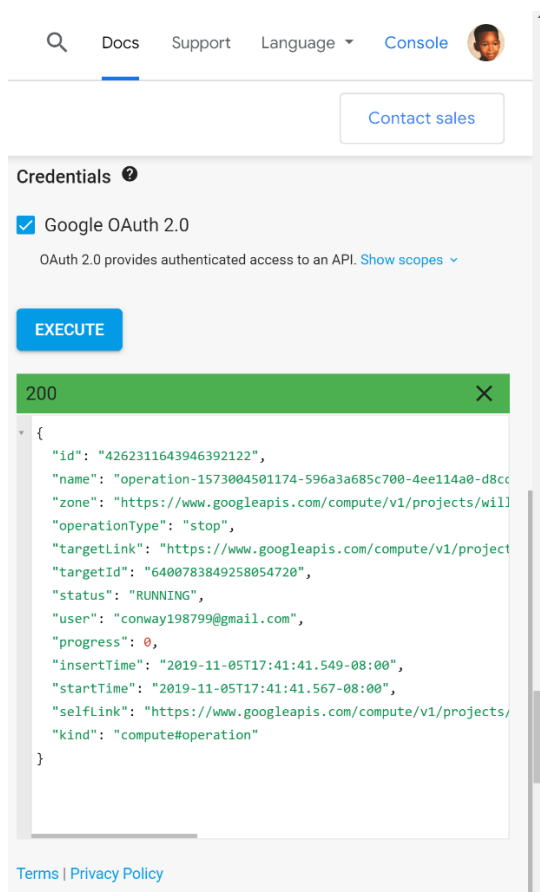
Read through this entire lab so you have a sense of what is expected of you before you start.

Read the page [Methods: instances.stop \(Links to an external site.\)](#), which describes the process for stopping a running instance using an API. You do not need to worry about installing any packages or authorization keys as long you run your code in the Cloud Shell.

Assignment

To document the steps below, create a Word document that contains screenshots and brief explanations of what you did. This serves as a notebook for you and a summary for the instructor to grade.

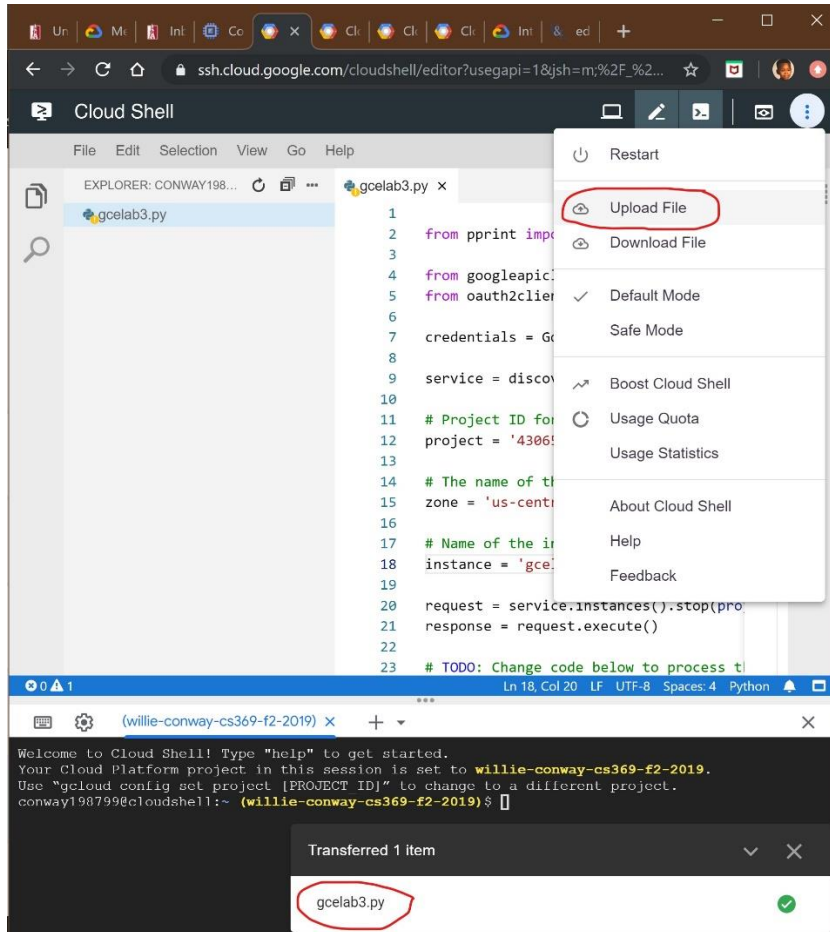
1. On the page [Methods: instances.stop \(Links to an external site.\)](#), use the API Explorer in section “Try this API” to test stopping your instance using an API. The three fields you need to set are project (**id** not name), zone, and instance name. Run this tool until you get the parameters correct, then take a screenshot.



(This is an output of my response dictionary after utilizing the API explorer in the section “Try this

API,” to test stopping my instance. To do this I entered the three fields of my project ID number, zone, and instance.)

2. Copy the python code from the section “Examples” and put it in a file on the Cloud Shell. You can either save the file on your local computer and upload the file, or you can open the Cloud Shell code editor.



(Uploading my gcelab3.py file to the Cloud Shell Editor after writing the file in my SSH through nano editor.)

3. Edit your python program to change the sections marked TODO, to use the parameters you found in step 1.

```
Unit 3: Homev... x  Method: insta... x  Compute Engi... x  Cloud Shell x  Cloud Shell x  Introducing G... x  editors - How... x  Python range... x  +
<  →  ↺  ⌂  ssh.google.com/cloudshell/editor?usegapi=1&jsch-mc%2F.%2Fscs%2Fapps-static%2F.%2Fjs%2Fk%3Dz.gapi.en.7kWSr24wXfC.O%2Fam%3DwQE%2Fd%3D1%2Fct%3Dzgm%2Fr%3DAG...  ☆  📄  🗂  📱  📺
Cloud Shell
File  Edit  Selection  View  Go  Help
EXPLORER: CONWAY198...  gcelab3.py x
gcelab3.py
2
3
4 from googleapiclient import discovery
5 from oauth2client.client import GoogleCredentials
6
7 credentials = GoogleCredentials.get_application_default()
8
9 service = discovery.build('compute', 'v1', credentials=credentials)
10
11 # Project ID for this request.
12 project = '30659292604' # TODO: Update placeholder value.
13
14 # The name of the zone for this request.
15 zone = 'us-central1-c' # TODO: Update placeholder value.
16
17 # Name of the instance resource to stop.
18 instance = 'gcelab2' # TODO: Update placeholder value.
19
20 request = service.instances().stop(project=project, zone=zone, instance=instance)
21 response = request.execute()
22
23 # TODO: Change code below to process the 'response' dict:
24 pprint(response)
25
```

```
{
  'name': 'operation-1b73014901898-596a612743b32-69b9b720-80798c65',
  'operationType': 'stop',
  'progress': 0,
  'selfLink': 'https://www.googleapis.com/compute/v1/projects/willie-conway-cs369-f2-2019/zones/us-central1-c/operations/operation-1573014901898-596a612743b32-69b9b720-80798c65',
  'startTime': '2019-11-05T20:35:02.402-08:00',
  'status': 'RUNNING',
  'targetId': '6400785849258054720',
  'targetLink': 'https://www.googleapis.com/compute/v1/projects/willie-conway-cs369-f2-2019/zones/us-central1-c/instances/gcelab2',
  'user': 'conway198799@gmail.com',
  'zone': 'https://www.googleapis.com/compute/v1/projects/willie-conway-cs369-f2-2019/zones/us-central1-c'
}
conway198799@cloudshell:~ (willie-conway-cs369-f2-2019) $
```

The screenshot displays the Google Cloud Platform console for the project 'Willie Conway CS369 F2 2019'. The 'Compute Engine' tab is active, showing a table of VM instances. A tooltip 'This instance is running' is visible over the first instance. Below the table, the Cloud Shell interface is open, showing a terminal window with the project ID 'willie-conway-cs369-f2-2019'.

Name	Zone	Recommendation	In use by	Internal IP	External IP	Connect
us-central1-c	us-central1-c			10.128.0.2 (nic0)	35.194.25.250	SSH

```
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to willie-conway-cs369-f2-2019.
Use "gcloud config set project [PROJECT_ID]" to change to a different project.
conway198799@cloudshell:~ (willie-conway-cs369-f2-2019) $
```

(I started my instance again so that I can test the code one more time, to make sure my fields were entered in correct, for my program to function.)

The screenshot shows the Google Cloud Platform interface. The top navigation bar includes 'Compute Engine' and 'VM instances'. Below this, there's a table of VM instances. A red arrow points to the instance named 'gcelab2'. Below the table, the Cloud Shell terminal is open, showing the output of a Python script that stops the VM instance.

Name	Zone	Recommendation	In use by	Internal IP	External IP	Connect
gcelab2	us-central1-c			10.128.0.2 (nic0)	None	SSH

```

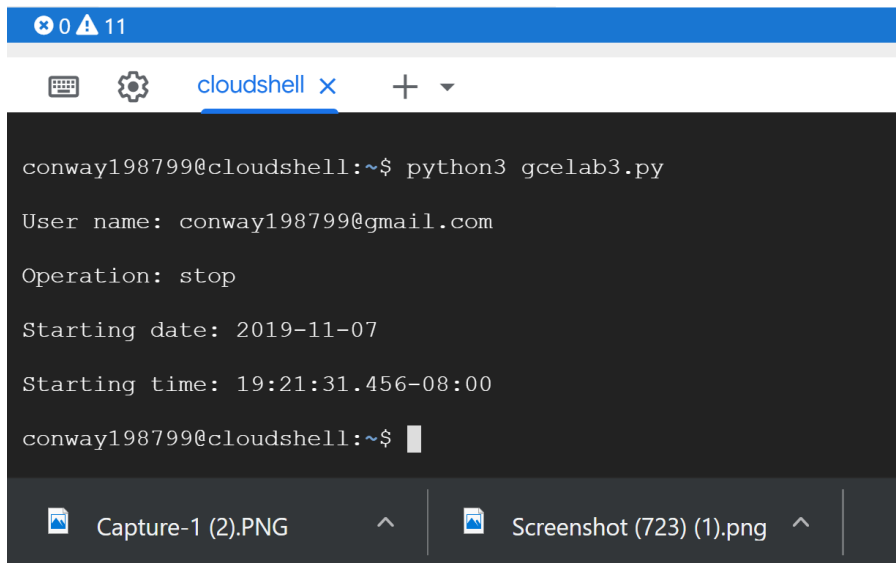
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to willie-conway-cs369-f2-2019.
Use "gcloud config set project [PROJECT ID]" to change to a different project.
conway1987998@cloudshell:~ (willie-conway-cs369-f2-2019) $ python3 --version
Python 3.7.3
conway1987998@cloudshell:~ (willie-conway-cs369-f2-2019) $ python3 gcelab3.py
{'id': '341925612548207093',
 'insertTime': '2019-11-05T20:35:02.362-08:00',
 'kind': 'compute#operation',
 'name': 'operation-1573014901898-596a612743b32-69b9b720-80798c65',
 'operationType': 'stop',
 'progress': 0,
 'selfLink': 'https://www.googleapis.com/compute/v1/projects/willie-conway-cs369-f2-2019/zones/us-central1-c/operations/operation-1573014901898-596a612743b32-69b9b720-80798c65',
 'startTime': '2019-11-05T20:35:02.402-08:00',
 'status': 'RUNNING',
 'targetId': '6400783849258054720',
 'targetLink': 'https://www.googleapis.com/compute/v1/projects/willie-conway-cs369-f2-2019/zones/us-central1-c/instances/gcelab2',
 'user': 'conway1987998@gmail.com',
 'zone': 'https://www.googleapis.com/compute/v1/projects/willie-conway-cs369-f2-2019/zones/us-central1-c'}
conway1987998@cloudshell:~ (willie-conway-cs369-f2-2019) $

```

(This is an output to show that I entered all of the following items correctly to get the same output from the website, of the complete response dictionary. It also showed that my instance gcelab2 has completely stopped. To activate this command, I entered python3 gcelab3.py. This tells the Cloud Shell that I'm using python version 3 and that I want to open the file gcelab3.py.)

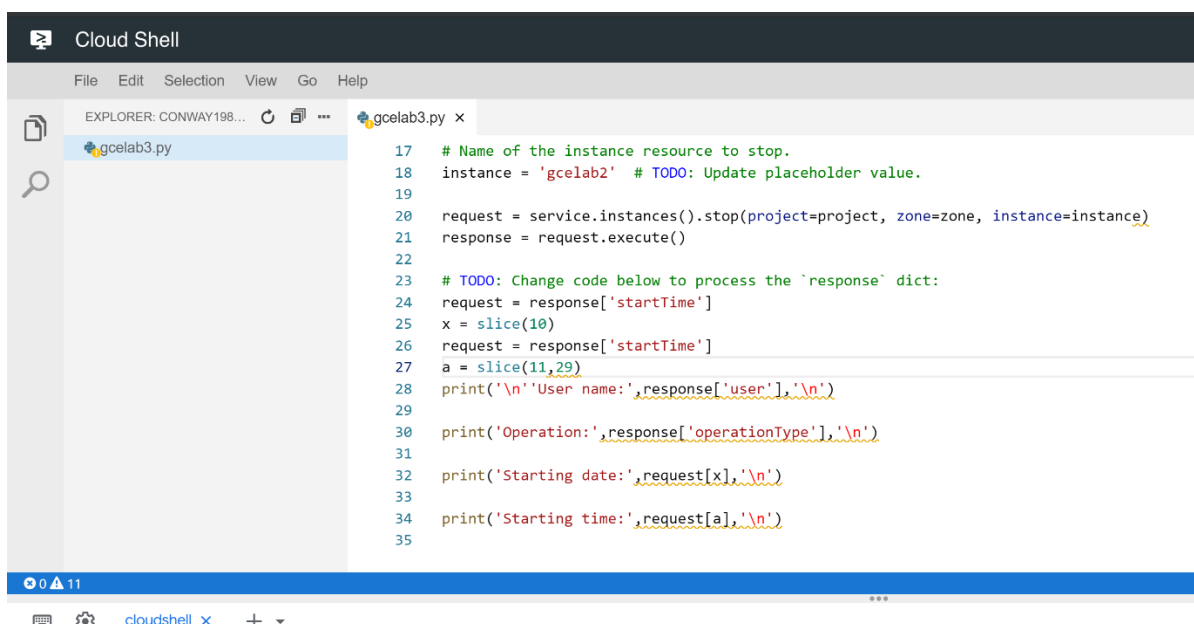
5. Modify your program so that instead of printing the entire dictionary, response, it only prints the following, each on a separate line: user, operation type, start date, start time

Since response is a dictionary, you access the elements like `response['user']`. Take a screenshot of the output of your program.



```
conway198799@cloudshell:~$ python3 gcelab3.py
User name: conway198799@gmail.com
Operation: stop
Starting date: 2019-11-07
Starting time: 19:21:31.456-08:00
conway198799@cloudshell:~$
```

(Output of the User name, Operation(stop), Starting date, and Starting time. I was able to obtain this by selecting specific elements from the response dictionary. See code below.)



```
Cloud Shell
File Edit Selection View Go Help
EXPLORER: CONWAY198... gcelab3.py x
gcelab3.py
17 # Name of the instance resource to stop.
18 instance = 'gcelab2' # TODO: Update placeholder value.
19
20 request = service.instances().stop(project=project, zone=zone, instance=instance)
21 response = request.execute()
22
23 # TODO: Change code below to process the `response` dict:
24 request = response['startTime']
25 x = slice(10)
26 request = response['startTime']
27 a = slice(11,29)
28 print('\n'+'User name:',response['user'],'\n')
29
30 print('Operation:',response['operationType'],'\n')
31
32 print('Starting date:',request[x],'\\n')
33
34 print('Starting time:',request[a],'\\n')
35
```

(The following code above displays the following output of the screenshot above the code. To obtain elements from the response dictionary, I had to go back to the previous output to see the selected elements and how they were typed, because that would be the only way to get my code to react to the previous code that was copied. In the last #TODO section I used request as a string to process the response['startTime']. Due to the response['startTime'] having one line of the date and time, I needed to use a function to split both, so that each would print on a separate line. To do this I used the slice() function, which allows me to slice characters from an object and return the slice object. So in the first string I use slice(10) to slice the time off of response['startTime'], then I used the variable (x) to return the first sliced object. For the second string I used slice(11,29) to slice the date from response['startTime'], then used the variable (a) to return the sliced object. For the remaining elements,

response['user'] and response['operationType'], all I had to do was print out the following elements. This then gave me the first screenshot of exercise 5, after running the program gcelab3.py.)

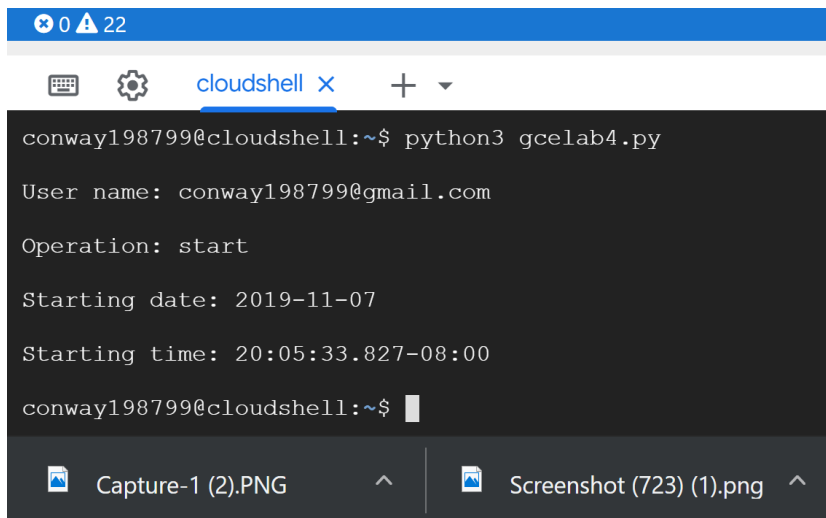
6. In addition to stopping your instance, you should now be able to write a python program to start your instance. See the page [Method: instances.start \(Links to an external site.\)](#) for all the details, though you should find things are very similar to what you have just done. This program should print the same type of information as in step 5 of this assignment. Take a screenshot of its output.

The screenshot displays the Google Cloud Platform console interface. The top navigation bar shows the project 'Willie Conway CS369 F2 2019'. The left sidebar lists various resources, with 'VM instances' selected. The main panel shows a table of VM instances. A red arrow points to the instance 'gcelab2' in the 'us-central1-c' zone. Below the table, the 'cloudshell' terminal window is open, showing the output of a command that prints a JSON dictionary. The dictionary contains details about the instance's operation, including its ID, name, type, and links to the instance and zone. The terminal prompt shows the user is 'conway1987998@cloudshell'.

Name	Zone	Recommendation	In use by	Internal IP	External IP	Connect
gcelab2	us-central1-c			10.128.0.2 (nic0)	35.232.118.180	SSH

```
{
  'id': '7641436103910184631',
  'insertTime': '2019-11-07T20:00:24.349-08:00',
  'kind': 'compute#operation',
  'name': 'operation-1573185623937-596cd4247d10d-7902151c-100b5264',
  'operationType': 'start',
  'progress': 100,
  'selfLink': 'https://www.googleapis.com/compute/v1/projects/willie-conway-cs369-f2-2019/zones/us-central1-c/operations/operation-1573185623937-596cd4247d10d-7902151c-100b5264',
  'startTime': '2019-11-07T20:00:24.352-08:00',
  'status': 'DONE',
  'targetId': '6400783849258054720',
  'targetLink': 'https://www.googleapis.com/compute/v1/projects/willie-conway-cs369-f2-2019/zones/us-central1-c/instances/gcelab2',
  'user': 'conway1987998@gmail.com',
  'zone': 'https://www.googleapis.com/compute/v1/projects/willie-conway-cs369-f2-2019/zones/us-central1-c'
}
```

(This is an output to show that I entered all the following items correctly to get the same output from the website, of the complete response dictionary. It also showed that my instance gcelab2 has completely started. To activate this command, I entered python3 gcelab4.py. This tells the Cloud Shell that I'm using python version 3 and that I want to open the file gcelab4.py.)



```
conway198799@cloudshell:~$ python3 gcelab4.py

User name: conway198799@gmail.com

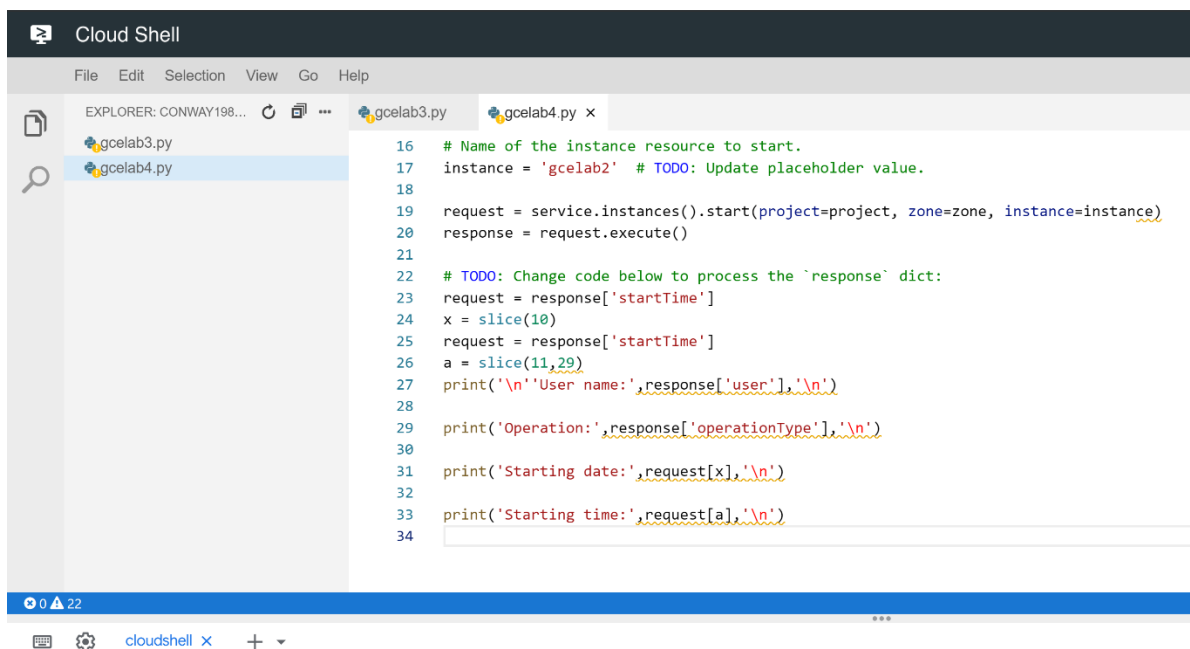
Operation: start

Starting date: 2019-11-07

Starting time: 20:05:33.827-08:00

conway198799@cloudshell:~$
```

(Output of the User name, Operation(start), Starting date, and Starting time. I was able to obtain this by selecting specific elements from the response dictionary. See code below.)



```
Cloud Shell
File Edit Selection View Go Help

EXPLORER: CONWAY198...
gcelab3.py
gcelab4.py

16 # Name of the instance resource to start.
17 instance = 'gcelab2' # TODO: Update placeholder value.
18
19 request = service.instances().start(project=project, zone=zone, instance=instance)
20 response = request.execute()
21
22 # TODO: Change code below to process the `response` dict:
23 request = response['startTime']
24 x = slice(10)
25 request = response['startTime']
26 a = slice(11,29)
27 print('\n'User name:',response['user'],'\n')
28
29 print('Operation:',response['operationType'],'\n')
30
31 print('Starting date:',request[x],'\n')
32
33 print('Starting time:',request[a],'\n')
34
```

(The same fundamentals that were applied in step 5 are applied in step 6. So, I applied the same code to get the similar output. Then processed the program using python gcelab4.py.)

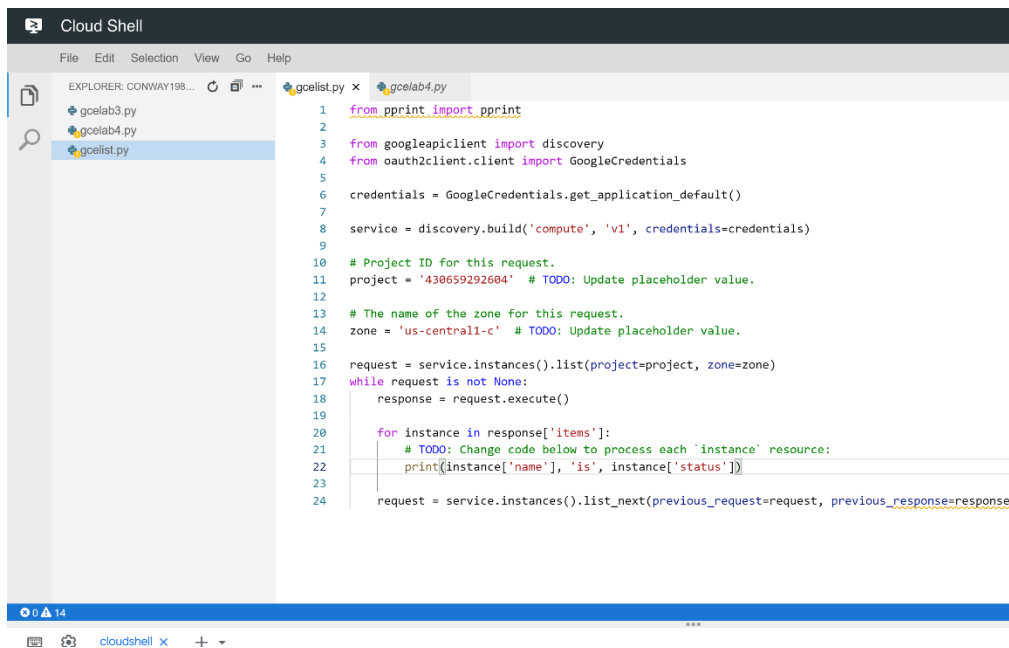
7. In your Word document, write a paragraph about what you did to write the program in step 6. Include any Linux commands you used.

(see Summary2.docx)

- See the page [Method: instances.list \(Links to an external site.\)](#), for generating a list of your instances. Copy the python program and upload it to your Cloud Shell. Edit the program to specify your project and zone, then replace the print statement with

`print(instance['name'], 'is', instance['status'])`

Run your program to get a list of your instances and their current status.



```
Cloud Shell
File Edit Selection View Go Help
EXPLORER: CONWAY198...
gcelab3.py
gcelab4.py
gcelist.py
gcelist.py
1 from pprint import pprint
2
3 from googleapiclient import discovery
4 from oauth2client.client import GoogleCredentials
5
6 credentials = GoogleCredentials.get_application_default()
7
8 service = discovery.build('compute', 'v1', credentials=credentials)
9
10 # Project ID for this request.
11 project = '430659292604' # TODO: Update placeholder value.
12
13 # The name of the zone for this request.
14 zone = 'us-central1-c' # TODO: Update placeholder value.
15
16 request = service.instances().list(project=project, zone=zone)
17 while request is not None:
18     response = request.execute()
19
20     for instance in response['items']:
21         # TODO: Change code below to process each 'instance' resource:
22         print(instance['name'], 'is', instance['status'])
23
24     request = service.instances().list_next(previous_request=request, previous_response=response)
```

(Created file gcelist.py and copy pasted the code from [Method: instances.list \(Links to an external site.\)](#) "Examples" section. Then I updated pprint(instance) to print(instance['name'], 'is', instance['status']). This code is used for generating a list of instances.)

The screenshot shows the Google Cloud Platform console for the project 'Willie Conway CS369 F2 2019'. The 'VM instances' tab is selected, displaying a table with one instance named 'gcelab2' in the 'us-central1-c' zone. The instance is in a 'RUNNING' state. Below the table, the Cloud Shell terminal shows the output of the 'gcloud compute instances list' command, confirming the instance is running.

Name	Zone	Recommendation	In use by	Internal IP	External IP	Connect
gcelab2	us-central1-c			10.128.0.2 (nic0)	35.232.118.180	SSH

```
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to willie-conway-cs369-f2-2019.
Use "gcloud config set project [PROJECT_ID]" to change to a different project.
conway198799@cloudshell:~ (willie-conway-cs369-f2-2019)$ python3 gcelist.py
gcelab2 is RUNNING
conway198799@cloudshell:~ (willie-conway-cs369-f2-2019)$
```

(Output of showing the list of instances that are their current status. Since I only have one instance, gcelab2 is in the current running state.)

The screenshot shows the Google Cloud Platform console for the project 'Willie Conway CS369 F2 2019'. The 'VM instances' tab is selected, displaying a table with one instance named 'gcelab2' in the 'us-central1-c' zone. The instance is in a 'STOPPING' state. Below the table, the Cloud Shell terminal shows the output of the 'gcloud compute instances list' command, confirming the instance is stopping.

Name	Zone	Recommendation	In use by	Internal IP	External IP	Connect
gcelab2	us-central1-c			10.128.0.2 (nic0)	35.232.118.180	SSH

```
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to willie-conway-cs369-f2-2019.
Use "gcloud config set project [PROJECT_ID]" to change to a different project.
conway198799@cloudshell:~ (willie-conway-cs369-f2-2019)$ python3 gcelist.py
gcelab2 is STOPPING
conway198799@cloudshell:~ (willie-conway-cs369-f2-2019)$
```

(Output of showing the list of instances that are their current status. Since I only have one instance, gcelab2 is in the current stopping state.)

The screenshot shows the Google Cloud Platform console for the project 'Willie Conway CS369 F2 2019'. The 'VM instances' page is active, displaying a table with one instance named 'gcelab2' in the 'us-central1-c' zone. The instance is in a 'TERMINATED' state. Below the table, a terminal window shows the output of the 'gcloud compute instances list' command, confirming the instance's status.

Name	Zone	Recommendation	In use by	Internal IP	External IP	Connect
gcelab2	us-central1-c			10.128.0.2 (nic0)	None	SSH

```

Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to willie-conway-cs369-f2-2019.
Use "gcloud config set project [PROJECT ID]" to change to a different project.
conway1987998@cloudshell:~ (willie-conway-cs369-f2-2019)$ python3 gcelist.py
gcelab2 is RUNNING
conway1987998@cloudshell:~ (willie-conway-cs369-f2-2019)$ python3 gcelist.py
gcelab2 is STOPPING
conway1987998@cloudshell:~ (willie-conway-cs369-f2-2019)$ python3 gcelist.py
gcelab2 is TERMINATED
conway1987998@cloudshell:~ (willie-conway-cs369-f2-2019)$
  
```

(Output of showing the list of instances that are their current status. Since I only have one instance, gcelab2 is in the current terminated state.)

9. Stop your instance, any way that you like. Then run your programs in this order

list, start (wait for it to finish), list, stop, list

Take a screenshot of the output, including the command prompt, so that it shows the following information. Be sure it is a screenshot image.

Compute Engine - Willie Conway x Method: instances.list | Comput x +

console.cloud.google.com/compute/instances?project=willie-conway-cs369-f2-20... ☆ | 2

Google Cloud Platform Willie Conway CS369 F2 2019

Compute Engine VM instances + SHOW INFO PANEL

VM instances

Filter VM instances

<input type="checkbox"/>	Name ^	Zone	Recommendation	In use by	Internal IP	External IP	C
<input type="checkbox"/>	gcelab2	us-central1-			10.128.0.2 (nic0)	None	S

(willie-conway-cs369-f2-2019) x

```
Use "gcloud config set project [PROJECT_ID]" to change to a different project.
conway198799@cloudshell:~ (willie-conway-cs369-f2-2019) $ python3 gcelist.py
gcelab2 is TERMINATED
conway198799@cloudshell:~ (willie-conway-cs369-f2-2019) $ python3 gcelab4.py

User name: conway198799@gmail.com

Operation: start

Starting date: 2019-11-08

Starting time: 22:03:30.734-08:00

conway198799@cloudshell:~ (willie-conway-cs369-f2-2019) $ python3 gcelist.py
gcelab2 is RUNNING
conway198799@cloudshell:~ (willie-conway-cs369-f2-2019) $ python3 gcelab3.py

User name: conway198799@gmail.com

Operation: stop

Starting date: 2019-11-08

Starting time: 22:06:32.604-08:00

conway198799@cloudshell:~ (willie-conway-cs369-f2-2019) $ python3 gcelist.py
gcelab2 is STOPPING
conway198799@cloudshell:~ (willie-conway-cs369-f2-2019) $
```

startprogram ss.PNG ^ Capture-1 (2).PNG ^ Screenshot (723) (1).png ^ Show all x

(Outputs the programs in the order of list, start (wait for it to finish), list, stop, list in functioning the gcelab2 instance.)