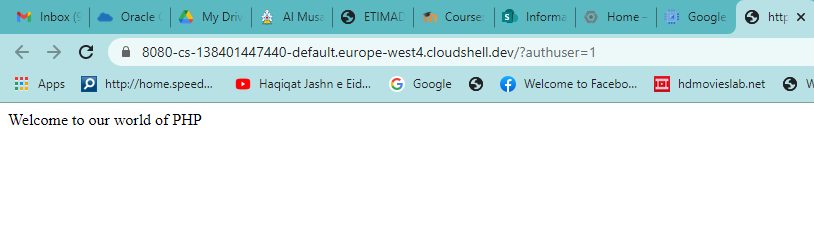
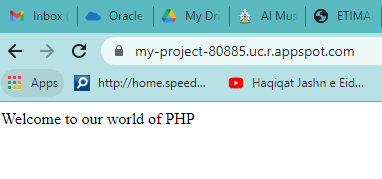
**Exercise 1: a)** In Google Clod Platform, deploy a simple web page (HelloWorld app) on appEngine using php-docs-samples available in github repository.

The web page should show the following output (on local machine and Cloud):

Output: local machine

 Output: GCP

1. Create a VM instance with machine type and zone as **“2 vCPUs, 2 GB”** and **“us-central 1-a**” inside the project.
2. Type appropriate commands in Google cloud shell to clone “HelloWorld” project from github repository (GoogleCloudPlatform/php-docs-samples).
3. Using appropriate command run the project on local web server to see the output in web preview (on localhost, port 8080).
4. Using appropriate command run, the project on GCP’s appEngine & see the output in your browser

**Solution:**

1. Create a project in GCP.
2. Create a VM instance with appropriate settings inside the project.
3. Enable billing (if it is not enabled already).
4. Open <https://github.com/GoogleCloudPlatform/php-docs-samples>.
5. Open Google cloud shell and type the command in command prompt

$ git clone <https://github.com/GoogleCloudPlatform/php-docs-samples>

1. Now type the following command (to reach the desired project directory)

$ cd php-docs-samples/appengine/flexible/helloworld

1. Now do the following to run/test ‘helloworld’ app on your local machine🡪
2. Install dependencies by typing the command🡪

$ composer install

1. Start a local web server by typing the command🡪

$ php –S localhost:8080 –t web

1. Now click on web preview button at the top right corner of shell to see the output in the web browser (web preview>> preview on port 8080).
2. Press ctrl + C to come out.
3. Now after testing the app locally, it’s the time to run your app on GCP’s appEngine.
4. Type the following command on command prompt

$ gcloud app deploy

1. Now, it will take max. 5 minutes in deployment process.
2. After the process is done, just copy the address from

**Deployed Service Default 🡪** [**https://.............appspot.com**](https://.............appspot.com)

1. Now paste this address in your browser’s address bar and press enter…….done.

So, Congratulations, it is done. Your application is deployed and running on GCP.

**Exercise 2: a)** Create a python app (**index0**

**1.py**) for demonstrating the use of library functions (like some mathematical functions- **sqrt, ceil, floor**, etc.) using IDLE 3.8 and save it on your desktop inside <**UPy\_PyLib**> folder.

b) Create and save a config file (**app.yaml**) inside the project folder.

c) Upload your project on github repository.

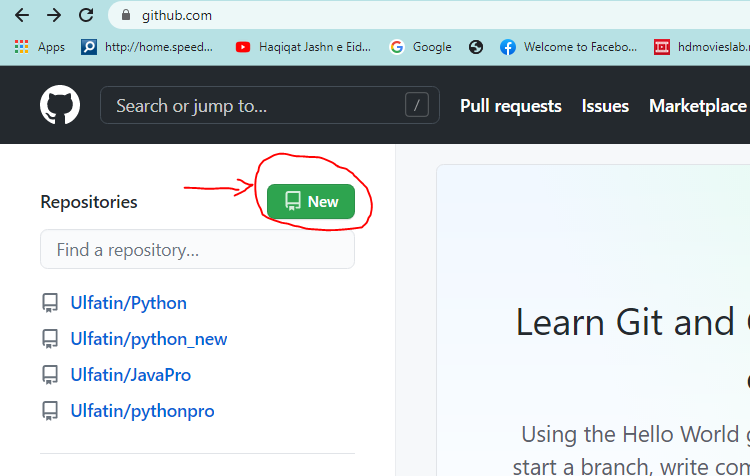
d) Create a new VM instance inside a project of your GCP account by choosing appropriate options.

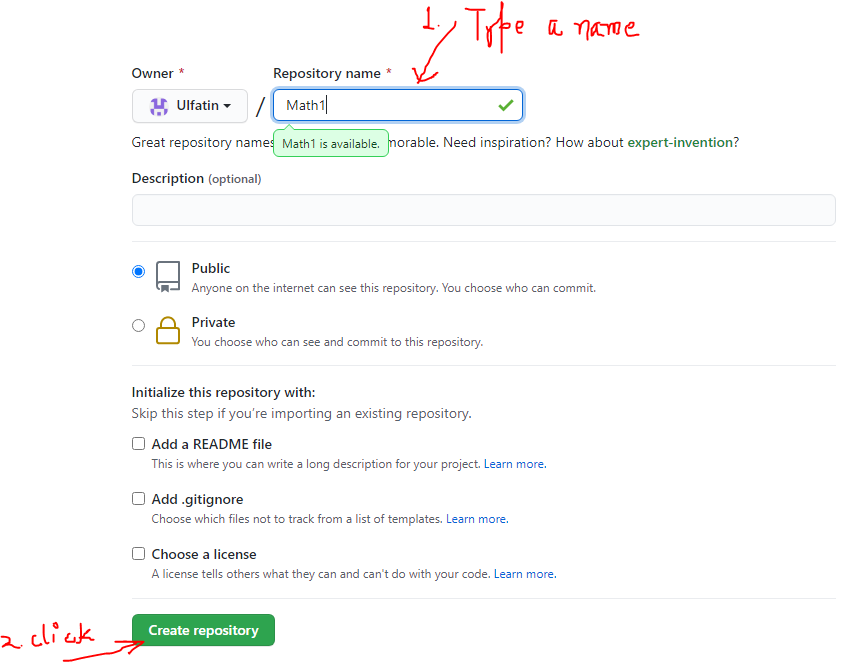
e) Type appropriate commands in Google cloud shell to clone your project that you have already uploaded in github repository.

f) Using appropriate commands locate your project folder and run your python app(index1.py) in Google Cloud shell.

**Solution:**

1. Create a project in GCP.
2. Create a VM instance with appropriate settings inside the project.
3. Enable billing (if it is not enabled already).
4. Create a python app (index01.py) for finding the remainder of two numbers using IDLE 3.8 and save it on your desktop inside <**UPy\_Project**> folder.
5. Create and save a config file (app.yaml) inside the project folder.
6. Open <https://github.com/> and login to your github account (if not already logged in)
7. c) Upload your project on github repository.





1. Crete a new repository MyPy (by clicking >>**new repository** on the top right menu).
2. Just drag and drop this project folder(MyPy) from your desktop to the github repository.
3. Open Google cloud shell and type the command in command prompt (copy the address of your github current page and paste in the command prompt)

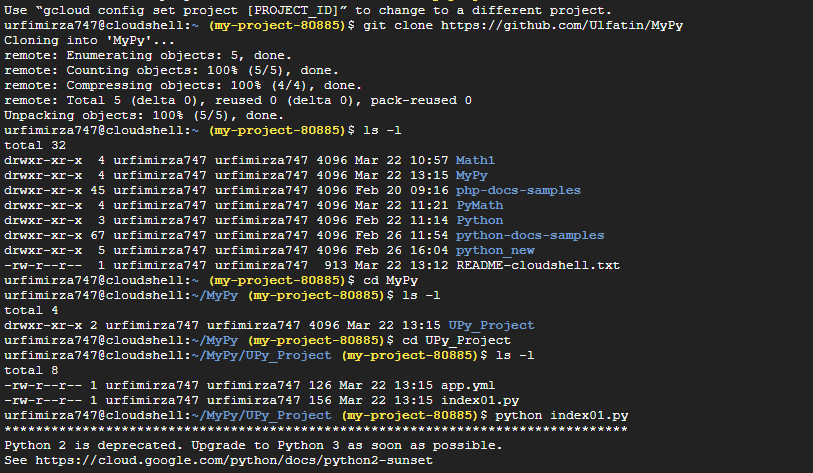
$ git clone <https://github.com/Ulfatin/>MyPy

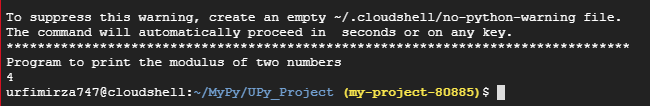
1. Now type the following command (to reach the desired project directory). You can use ls command also to check the directory and file listing.

$ cd Urfi/MyPy

1. Now, finally type the following command to run/test the python project🡪

$ python index01.py





|  |
| --- |
| **Index01.py (Python application file)**  **import math**  **print(" Program to demonstrate the use of Pyton library-")**  **num=121.121**  **print(math.sqrt(num))**  **print(math.ceil(num))**  **print(math.floor(num))**  **app.yaml (config. file)**  **application: UPy\_PyLib**  **version: 1**  **runtime: python**  **api\_version: 1**  **handlers:**  **- url: /.\***  **script: index01.py** |

**Some other Sample Python Programs**

1.

# Relational and Equality Operators

a=30

b=20

if a==b:

  print('a is equal to b')

else:

  print('a and b are not equal')

2.

# Relational and Equality Operators

a=30

b=20

if a==b:

  print('a is equal to b')

else:

  print('a and b are not equal')

3. # Relational and Equality Operators

a=30

b=20

if a!=b:

  print('a is not equal to b')

else:

  print('a and b are equal')

4. # Print the square of first 10 numbers ( 1 to 10)

for i in range(1,10):

  print(i\*i)

5. # Print the series of first 10 even numbers ( 2 to 20)

for i in range(2,20,2):

print(i)

6. # Print the series of first 10 even numbers in reverse ( 2 to 20)

for i in range(2,20,-2):

print(i)

Git clone

Ls

Cd

Ls

Cd

Ls

Python………

ROI [(Gain – Cost) / Cost]\*100 %

Exercise 1: The data of a data science research project is transmitted through a high speed network (fiber-optic

cables) and provide a bandwidth of 56 Gbps.

a) How long does it take to transfer the 35 PB (1 PetaBytes = 2 50 Byte) through a 56 Gbps network?

b) What will be the best choice of network service if you want faster data transfer (Public Cloud, Private

Cloud or Own Network Infrastructure/ Data Centre) and why?

Accessibility, durability, fast data access, flexibility, Availability

Solution:

56 Gbps bandwidth = 56 \* 2 30 b / s = (56/8) \*

= 7 x 2 30 Bytes / s (since, 1 Byte = 8 bits)

Time = Distance / Speed

Duration of transmission(T) = 35 x 2 50 / 7 x 2 30 = 5 x 2 50 – 30 = 5 x 2 20

sec

= 5 \* 1024 \* 1024

= 5,242,880 s = (5242880 / 60) minutes

= 87,381 minutes

= 1456 hours

= 60.66 days

= Approx 61 days

It will take approximately 61 days

1 byte = 8 bits

1 K Byte = 1024 = 210 Byte = 1024 bytes

1 MByte = 210 K byte = 210 \* 210Byte = 2 20 Bytes

1 GB = 210 Mbyte = 210 \* 210 K Byte = 210 \* 210 \* 210 Byte = 2 30 Bytes

1 TB = 240 Bytes

1 Peta Byte = 250 Bytes

1 Zeta Byte = 260 Bytes