[2CEIT603: CLOUD COMPUTING]

Practical: 9

AIM- Study and Explore Google App Engine

Submitted By: Adeshara Brijesh

Enrollment number: 21012021001

Department of Computer Engineering

Google App Engine

1. INTRODUCTION OF GOOGLE APP ENGINE:

➢ Abstract:

Google App Engine was first released as a beta version in April 2008. It is a platform for developing and hosting web applications in Google-managed data centers. Google's App Engine opens Google's production to any person in the world at no charge. Much like Google gives us all free email with an amazing amount of long term storage, we now have the ability to run the software that we write in Google's data centers.

Google App Engine is cloud computing technology. Google App Engine is software that facilitates the user to run his web applications on Google infrastructure. It is more reliable because failure of any server will not affect either the performance of the end user or the service of the Google.

It virtualizes applications across multiple servers and data centers. Other cloud-based platforms include offerings such as Amazon Web Services and Microsoft's Azure Services Platform.

➢ Introduction:

- Google App Engine lets you run your web applications on Google's infrastructure. App Engine applications are easy to build, easy to maintain, and easy to scale as your traffic and data storage needs grow. With App Engine, there are no servers to maintain: You just upload your application, and it's ready to serve your users.

- You can serve your app from your own domain name (such as http://www.example.com/) using Google Apps. Or, you can serve your app using a free name on the appspot.com domain. You can share your application with the world, or limit access to members of your organization.

- Google App Engine supports apps written in several programming languages. With App Engine's Java runtime environment, you can build your app using standard Java technologies, including the JVM, Java servlets, and the Java

programming language—or any other language using a JVM-based interpreter or compiler, such as JavaScript or Ruby. App Engine also features a dedicated Python runtime environment, which includes a fast Python interpreter and the Python standard library. The Java and Python runtime environments are built to ensure that your application runs quickly, securely, and without interference from other apps on the system.With App Engine, you only pay for what you use. There are no set -up costs and no recurring fees. The resources your application uses, such as storage and bandwidth, are measured by the gigabyte, and billed at competitive rates. You control the maximum amounts of resources your app can consume, so it always stays within your budget.

- App Engine costs nothing to get started. All applications can use up to 500 MB of storage and enough CPU and bandwidth to support an efficient app serving around 5 million page views a month, absolutely free. When you enable billing for your application, your free limits are raised, and you only pay for resources you use above the free levels.

➢ Advantages & Disadvantages:

- But how exactly is App Engine useful in real-life settings and why would it benefit you to begin using this low-cost, time-sensitive solution? Ask around and you'll commonly hear the following points:

- **Cost-effective:** The price - or lack thereof - is the biggest selling point. The free tier is incredibly generous and most small applications don't need anything more. However, if you do find that you need more features, even the paid apps are cost- friendly. You only pay for the resources you use, and they're relatively cheap. Best of all, you only get charged for your application when it handles requests. Talk about maximizing capital!

- **No servers to maintain:** With App Engine, you don't have any reason or need to maintain a server. You simply upload your applications and they're ready to go. Google takes care of the rest for you. This frees up your resources so they can be better utilized in other areas of your business.

- **Fully integrated:** One of the advantages that often flies under the radar is that App Engine is connected to Google's many other products and services. App Engine, Big Query, Compute Engine, and Cloud Storage all work together for your greater good.

- **Risk Free Trial Period**: we can use its services for free for one month with $300 credit initially.

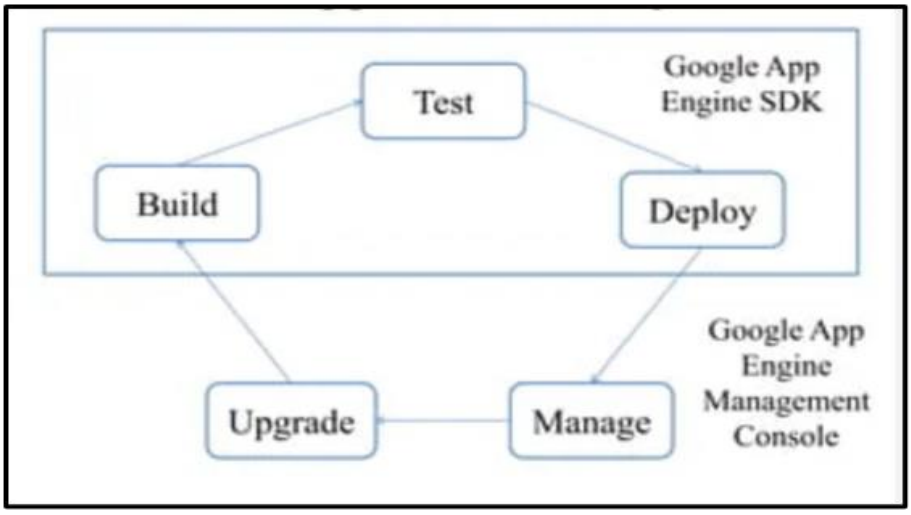
➢ Disadvantages:

- **You are putting your application in Google's hands** Think about that for a minute. You are at the mercy of Google -- if disaster strikes and Google one day disappears, you are done too. Or, more realistically, if the Google App Engine goes down for an hour, you are also down for an hour -- and you will have no idea what happened. Even if you try and get an answer from someone at Google, you won't. Just like Google Apps, it will be impossible to explain things to your end users.

- **Once you are in, you are really in** Using Google's infrastructure is very tempting. But any smart company should have some sort of plan for the future. What if you realized that you didn't want to host your application on Google App Engine anymore? Good luck, almost everything you are given access to is proprietary -- that means all your data is locked into Big Table in a format that isn't like a traditional relational database. It's also very tempting to use the API's Google provides to interface with things like Google accounts.

- **Privacy should not be taken lightly** Google has a very strong privacy policy -- and personally I trust them. However, I'm trusting them with my personal information -- you will be trusting them with all of your company's data. These are two completely different things. If you have a low trust tolerance, you may not want to risk putting everything that belongs to your company behind Google's doors. That said, I personally would still feel comfortable putting company data on their infrastructure -- simply because I know it's proven to be secure, scalable and robust over the last several years with their own services.

➢ Creating/Deploying Application:



- Creating an App Engine application is easy, and only takes a few minutes. And it's free to start: upload your app and share it with users right away, at no charge and with no commitment required.
- Google App Engine applications can be written in either the Java or Python programming languages.
- The Steps for how to create an application and deploy on app engine is shown below.

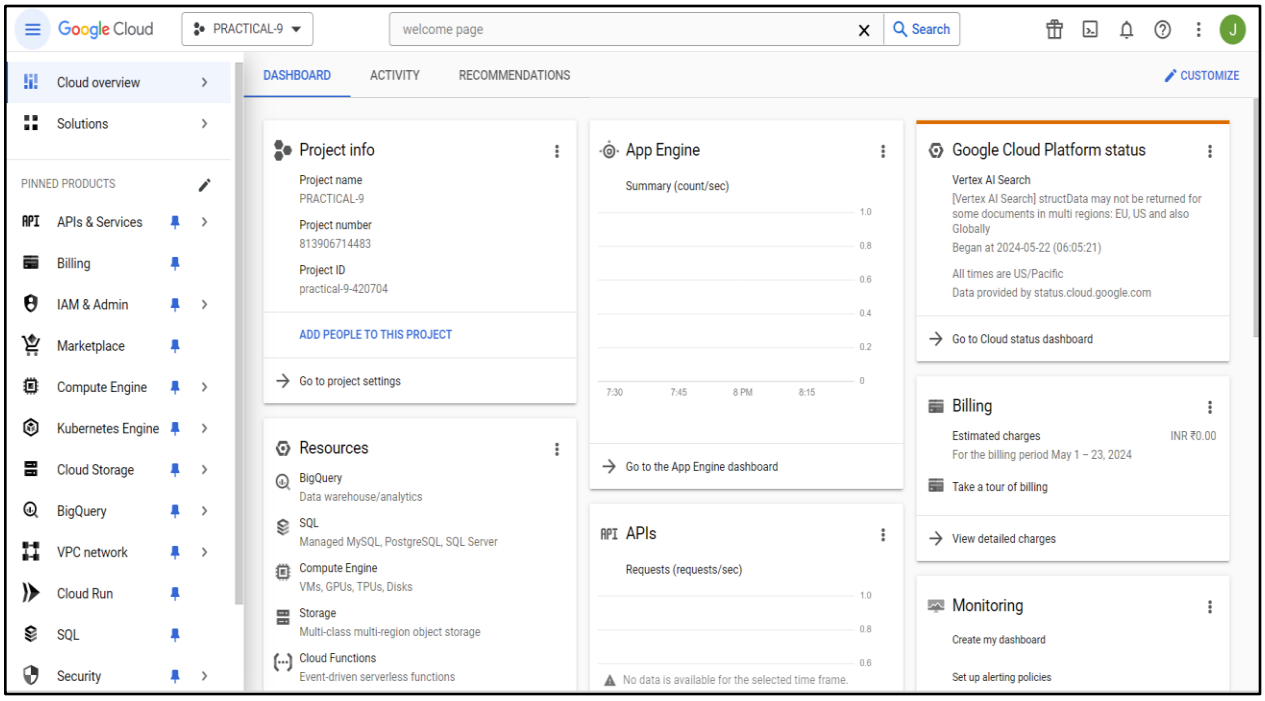
➢ **Deploying a website with database on Google App Engine step by step.**

Pre-Requirements:

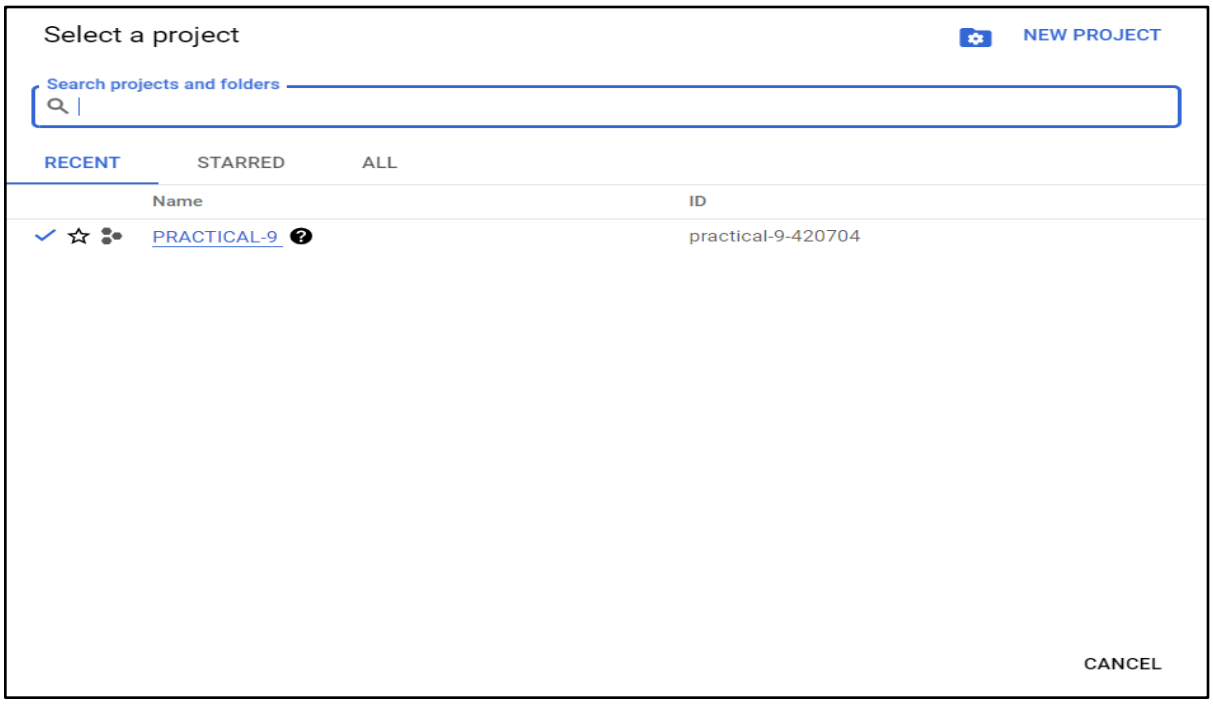
VS Code
Google Cloud SDK
PHP

STEPS:

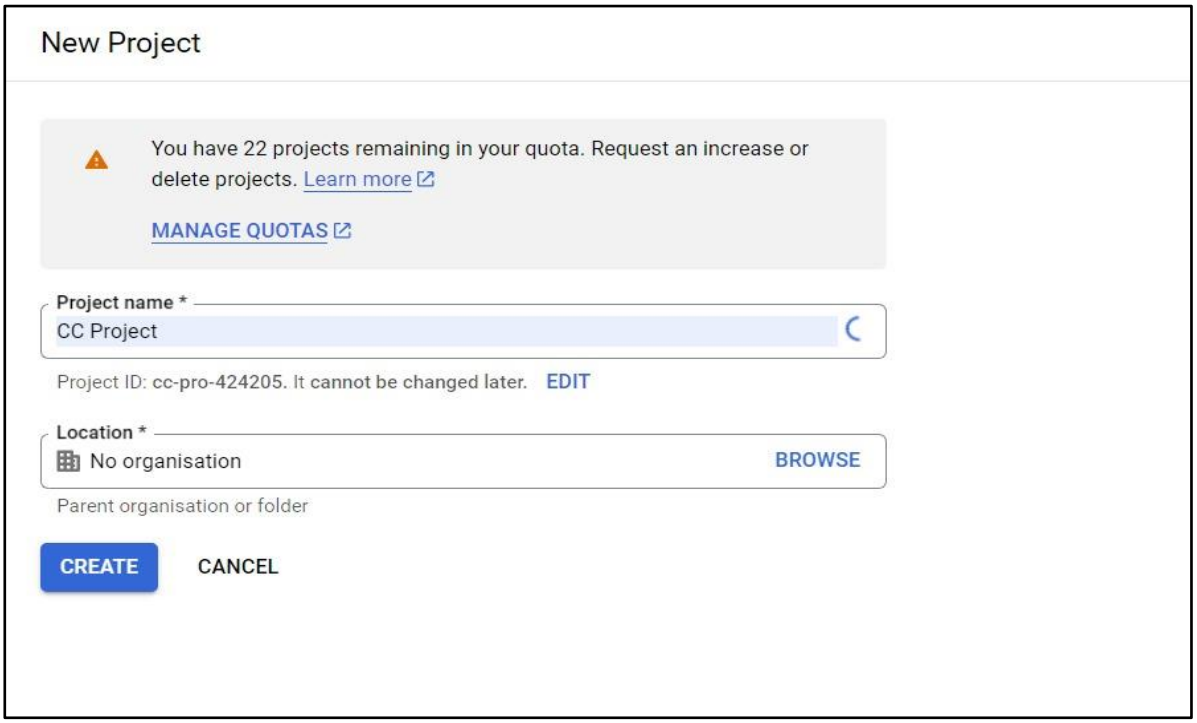
1. Sign up with your existing Google account and you can see this HOME PAGE.



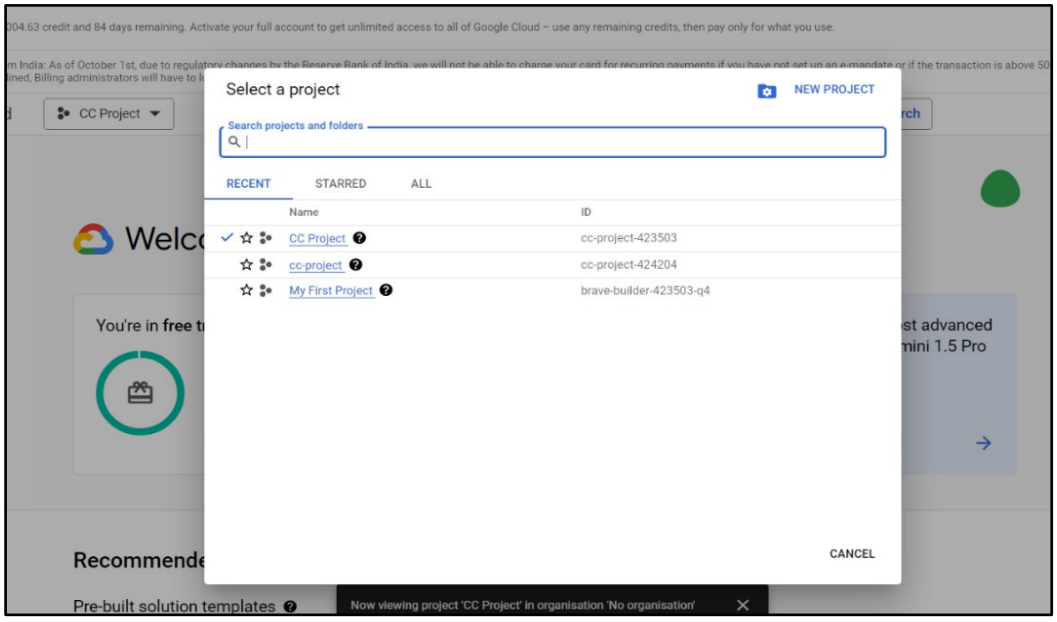
2. Now for create new project click on “Create Project” here.



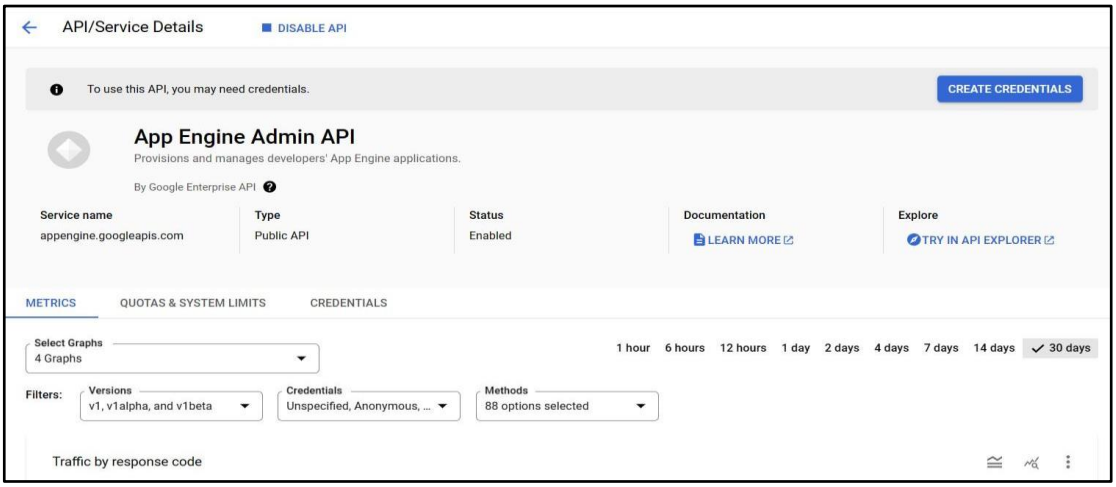
3. Give and Project Name and Create it.



4. From here select the project.



5. Search for Admin API and enable it



1. Create a app in newly make project using terminal

```
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to cc-project-423503.
Use "gcloud config set project [PROJECT_ID]" to change to a different project.
kishanpatel2984@cloudshell:~ (cc-project-423503)$ gcloud app create
```

Selecting a serv.

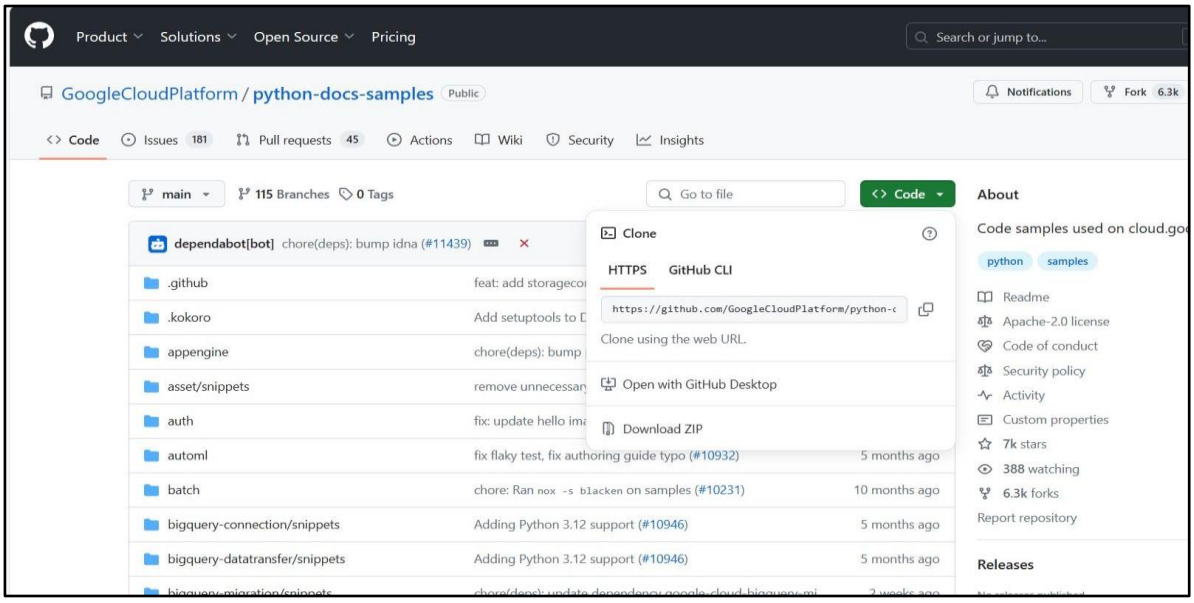
```
kishanpatel2984@cloudshell:~ (cc-project-423503)$ gcloud app create
You are creating an app for project [cc-project-423503].
WARNING: Creating an App Engine application for a project is irreversible and the region
cannot be changed. More information about regions is at
<https://cloud.google.com/appengine/docs/locations>.

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

 [1] asia-east1    (supports standard and flexible)
 [2] asia-east2    (supports standard and flexible and search_api)
 [3] asia-northeast1 (supports standard and flexible and search_api)
 [4] asia-northeast2 (supports standard and flexible and search_api)
 [5] asia-northeast3 (supports standard and flexible and search_api)
 [6] asia-south1   (supports standard and flexible and search_api)
 [7] asia-southeast1 (supports standard and flexible)
 [8] asia-southeast2 (supports standard and flexible and search_api)
 [9] australia-southeast1 (supports standard and flexible and search_api)
 [10] europe-central2 (supports standard and flexible)
 [11] europe-west   (supports standard and flexible and search_api)
 [12] europe-west2  (supports standard and flexible and search_api)
 [13] europe-west3  (supports standard and flexible and search_api)
 [14] europe-west6  (supports standard and flexible and search_api)
 [15] northamerica-northeast1 (supports standard and flexible and search_api)
 [16] southamerica-east1 (supports standard and flexible and search_api)
 [17] us-central    (supports standard and flexible and search_api)
 [18] us-east1      (supports standard and flexible and search_api)
 [19] us-east4      (supports standard and flexible and search_api)
 [20] us-west1      (supports standard and flexible)
 [21] us-west2      (supports standard and flexible and search_api)
 [22] us-west3      (supports standard and flexible and search_api)
 [23] us-west4      (supports standard and flexible and search_api)
 [24] cancel
Please enter your numeric choice:  7

Creating App Engine application in project [cc-project-423503] and region [asia-southeast1]....done.
Success! The app is now created. Please use `gcloud app deploy` to deploy your first app.
```

2. Clone any git repository OR you can make your own project in editor.



Repository link :

python-docs_samples/appengine/standard_python3/hello_world at main · GoogleCloudPlatform/python-docs-samples · GitHub

```
kishanpatel2984@cloudshell:~ (cc-project-423503)$ git clone https://github.com/GoogleCloudPlatform/python-docs-samples.git
Cloning into 'python-docs-samples'...
remote: Enumerating objects: 112065, done.
remote: Counting objects: 100% (180/180), done.
remote: Compressing objects: 100% (98/98), done.
remote: Total 112065 (delta 108), reused 126 (delta 82), pack-reused 111885
Receiving objects: 100% (112065/112065), 241.25 MiB | 14.04 MiB/s, done.
Resolving deltas: 100% (66123/66123), done.
Updating files: 100% (5024/5024), done.
```

3. Go to directory where your files are placed.

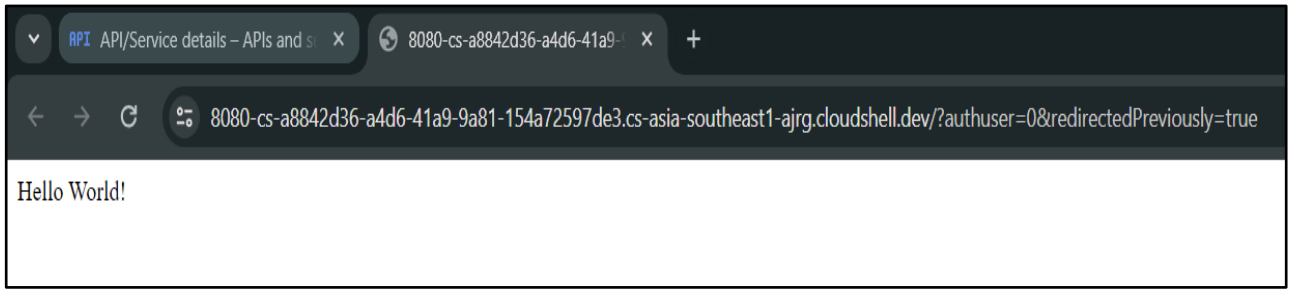
```
kishanpatel2984@cloudshell:~ (cc-project-423503)$ cd python-docs-samples/
kishanpatel2984@cloudshell:~/python-docs-samples (cc-project-423503)$ cd appengine
kishanpatel2984@cloudshell:~/python-docs-samples/appengine (cc-project-423503)$ cd standard_python3/
kishanpatel2984@cloudshell:~/python-docs-samples/appengine/standard_python3 (cc-project-423503)$ cd hello_world
kishanpatel2984@cloudshell:~/python-docs-samples/appengine/standard_python3/hello_world (cc-project-423503)$
```

4. Run your files in terminal for testing purpose.

- 


```
kishanpatel2984@cloudshell:~/python-docs-samples/appengine/standard_python3/hello_world (cc-project-423503)$ python3 main.py
 * Serving Flask app 'main'
 * Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:8080
Press CTRL+C to quit
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 119-313-427
127.0.0.1 - - [16/May/2024 04:07:15] "GET /?authuser=0&redirectedPreviously=true HTTP/1.1" 200 -
127.0.0.1 - - [16/May/2024 04:07:16] "GET /favicon.ico HTTP/1.1" 404 -
^Ckishanpatel2984@cloudshell:~/python-docs-samples/appengine/standard_python3/hello_world (cc-project-423503)$
```

5. Check Locally whether is running or not.



https://8080-cs-a8842d36-a4d6-41a9-9a81-154a72597de3.cs-asia-southeast1-ajrg.cloudshell.dev/?authuser=0&redirectedPreviously=true

6. Deploy your Project on cloud.

```
kishanpatel2984@cloudshell:~/python-docs-samples/appengine/standard_python3/hello_world (cc-project-423503)$ gcloud app deploy
Services to deploy:

descriptor:                  [/home/kishanpatel2984/python-docs-samples/appengine/standard_python3/hello_world/app.yaml]
source:                      [/home/kishanpatel2984/python-docs-samples/appengine/standard_python3/hello_world]
target project:              [cc-project-423503]
target service:              [default]
target version:              [20240516t040820]
target url:                  [https://cc-project-423503.as.r.appspot.com]
target service account:      [cc-project-423503@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
17%
33%
50%
67%
83%
100%
100%
File upload done.
Updating service [default]...done.
Setting traffic split for service [default]...done.
Deployed service [default] to [https://cc-project-423503.as.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
```

7. Now browse service.

```
kishanpatel2984@cloudshell:~/python-docs-samples/appengine/standard_python3/hello_world (cc-project-423503)$ gcloud app browse
Did not detect your browser. Go to this link to view your app:
https://cc-project-423503.as.r.appspot.com
kishanpatel2984@cloudshell:~/python-docs-samples/appengine/standard_python3/hello_world (cc-project-423503)$
```

8. Run service on browse via link.

- Link: https://cc-project-423503.as.r.appspot.com/