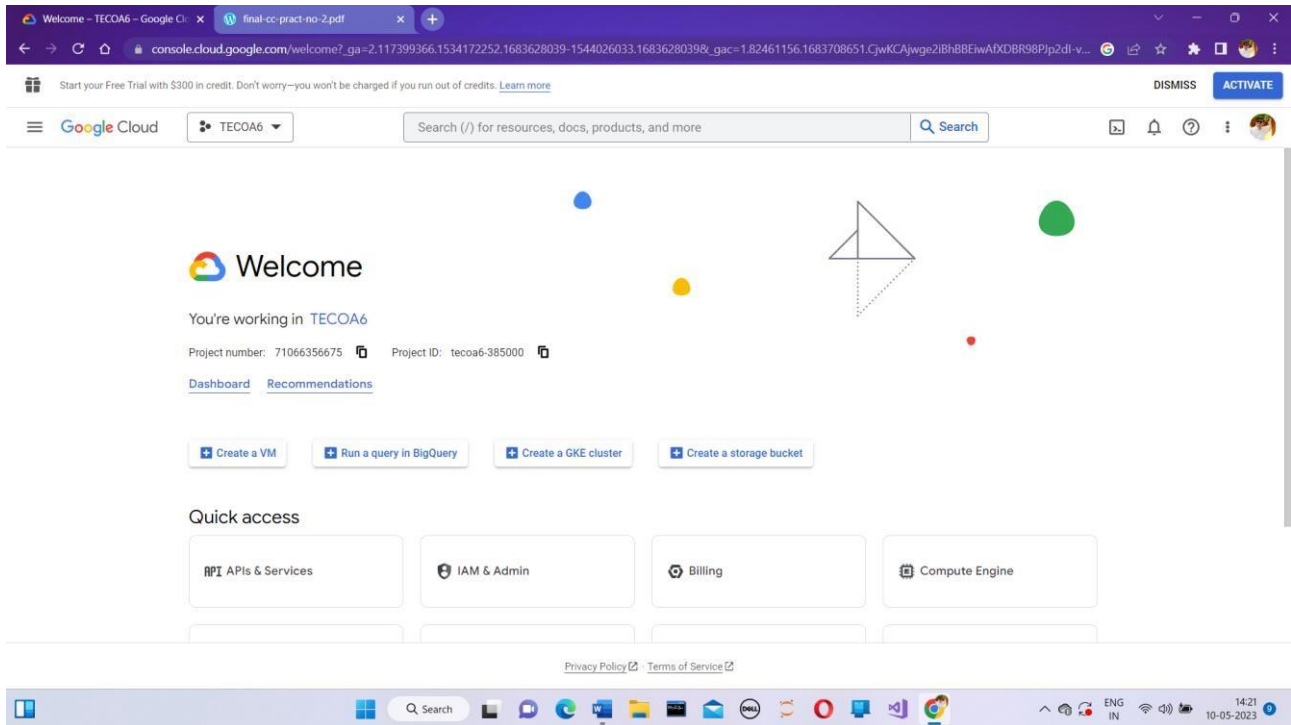
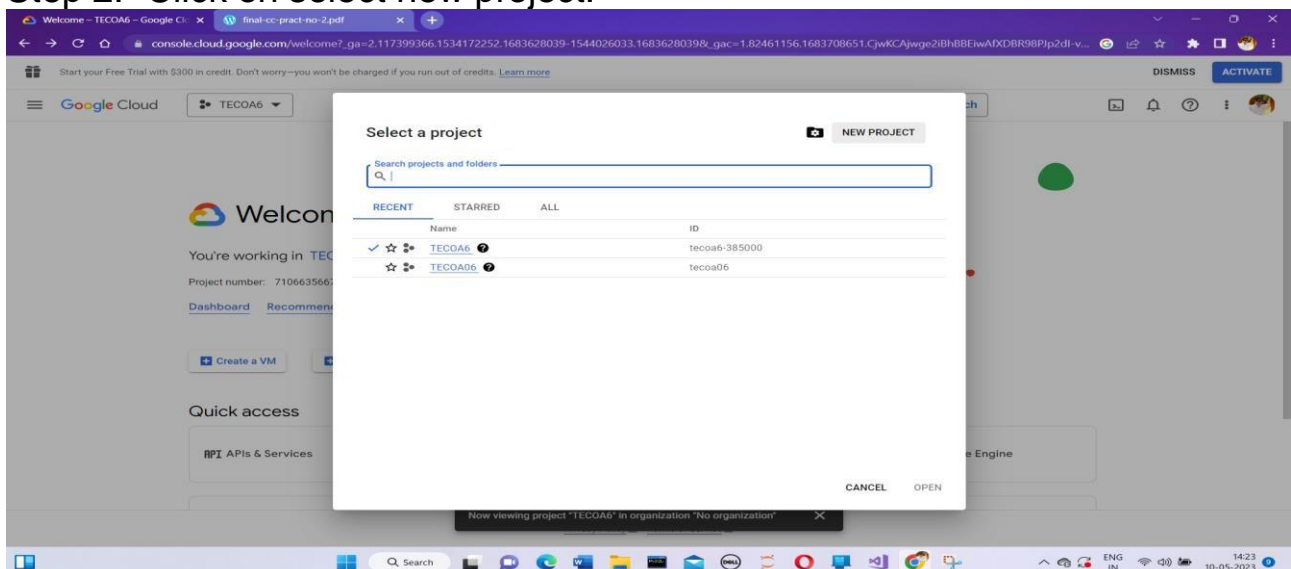


# Title:- Installation and configure Google app engine

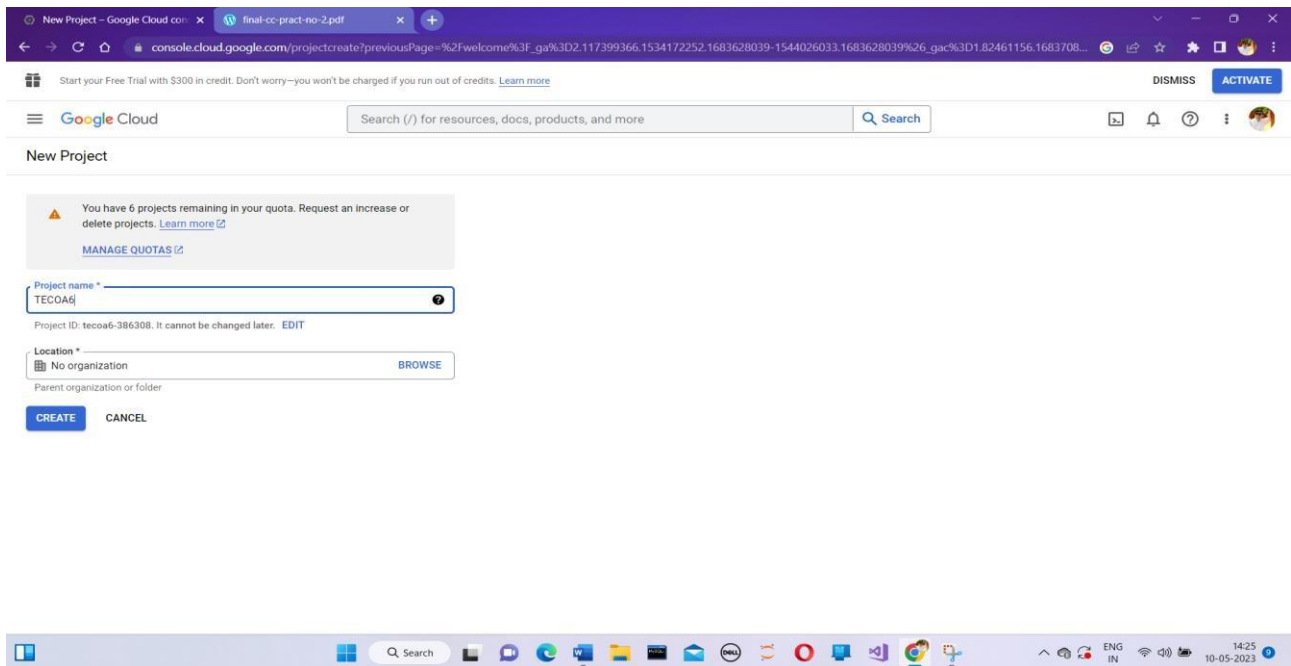
Step 1:- Search Google Cloud Platform in a any search engine& Click on Console.



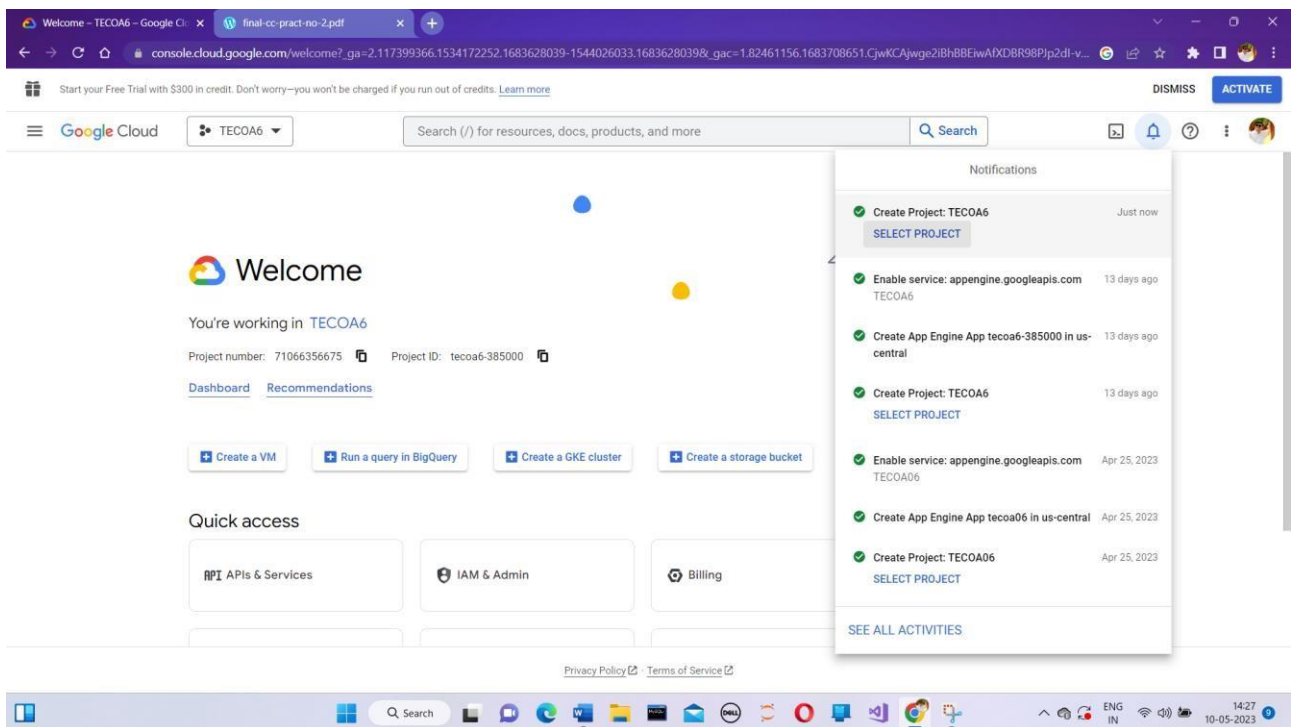
Step 2:- Click on select new project.



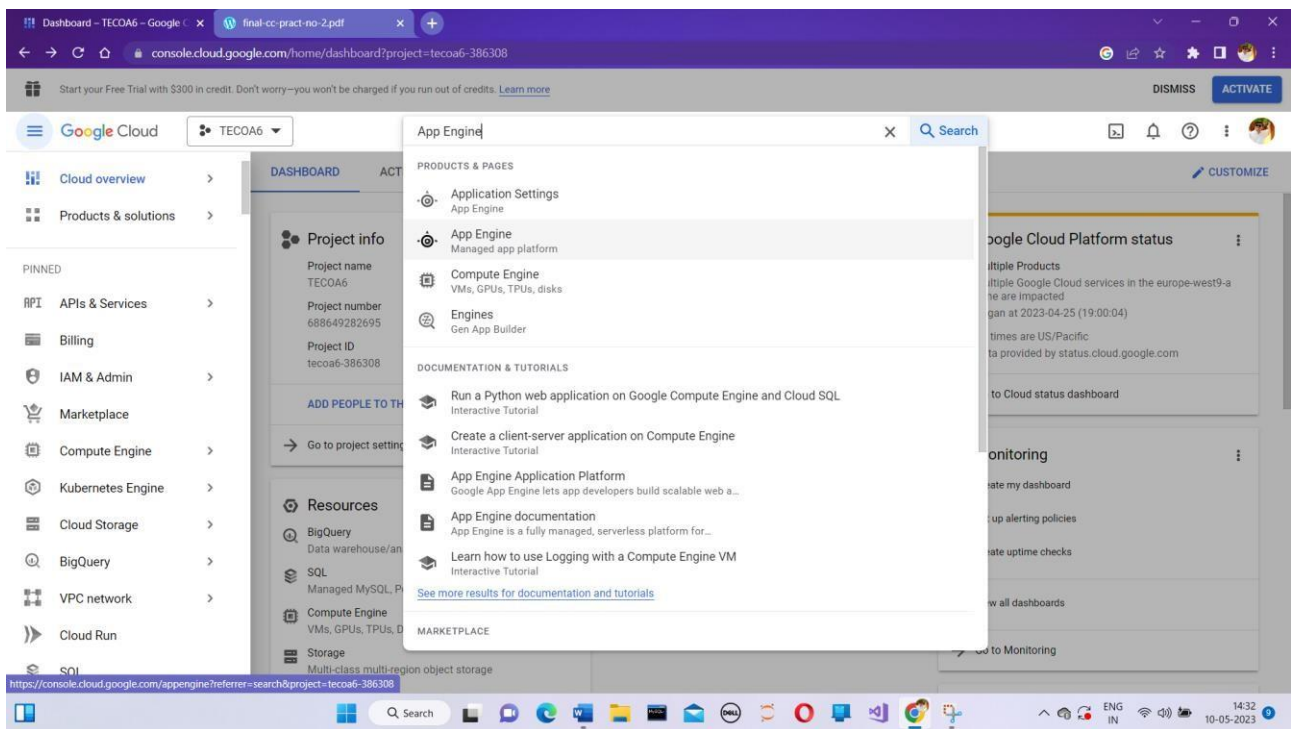
Step3:- Give Project name and click on create.



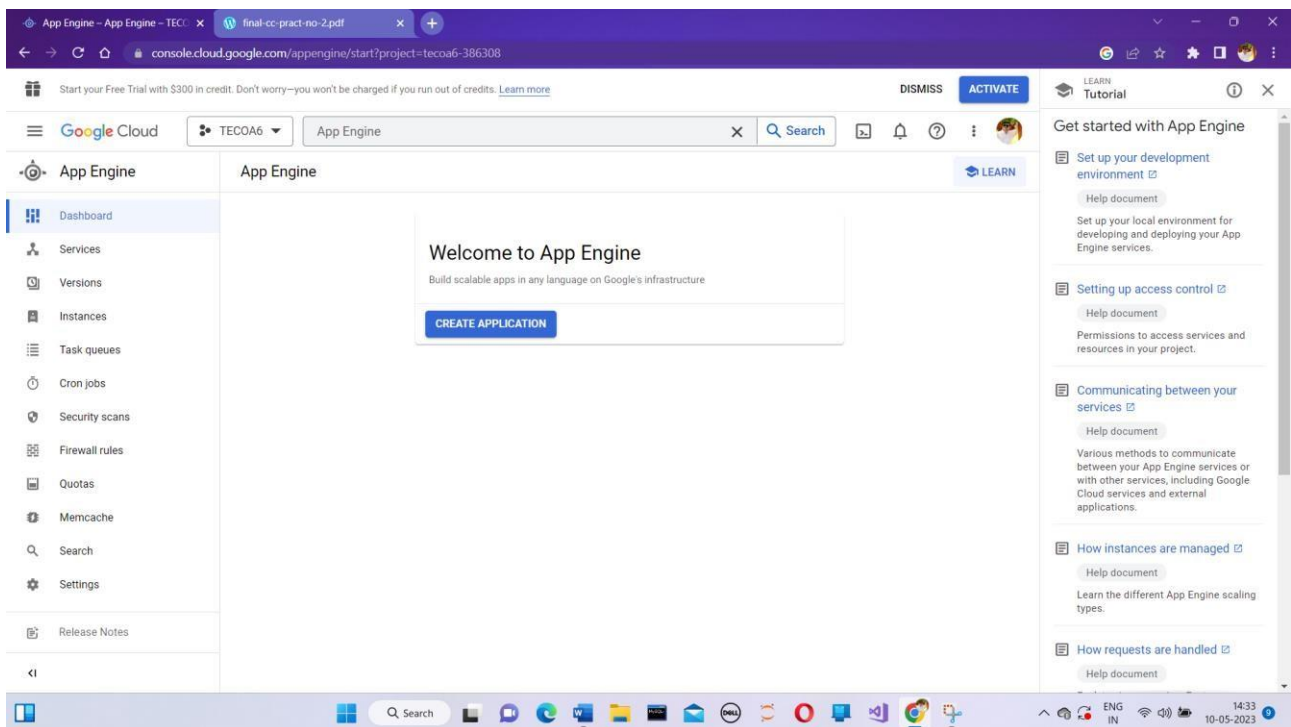
Step4:- Click on select project



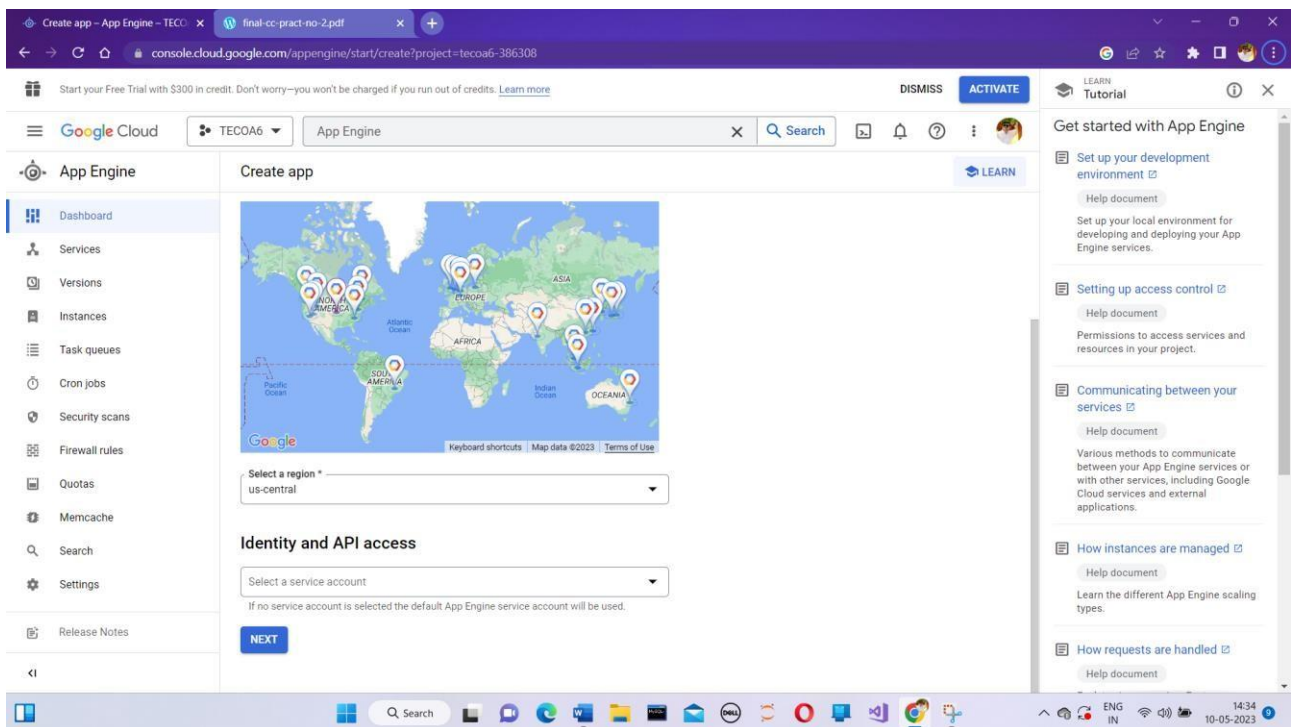
Step5:- In a search bar type search App Engine



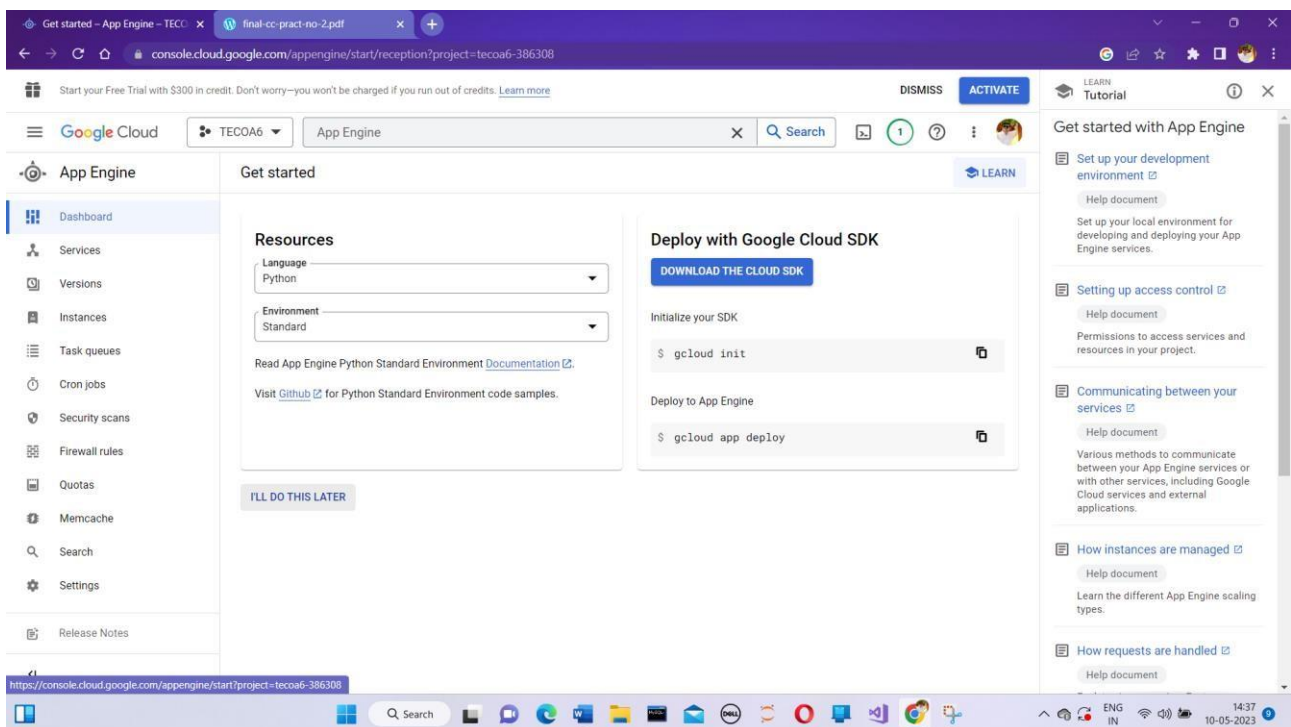
Step 6:-Click on App Engine and Following screen will appear& Click on Create Application.



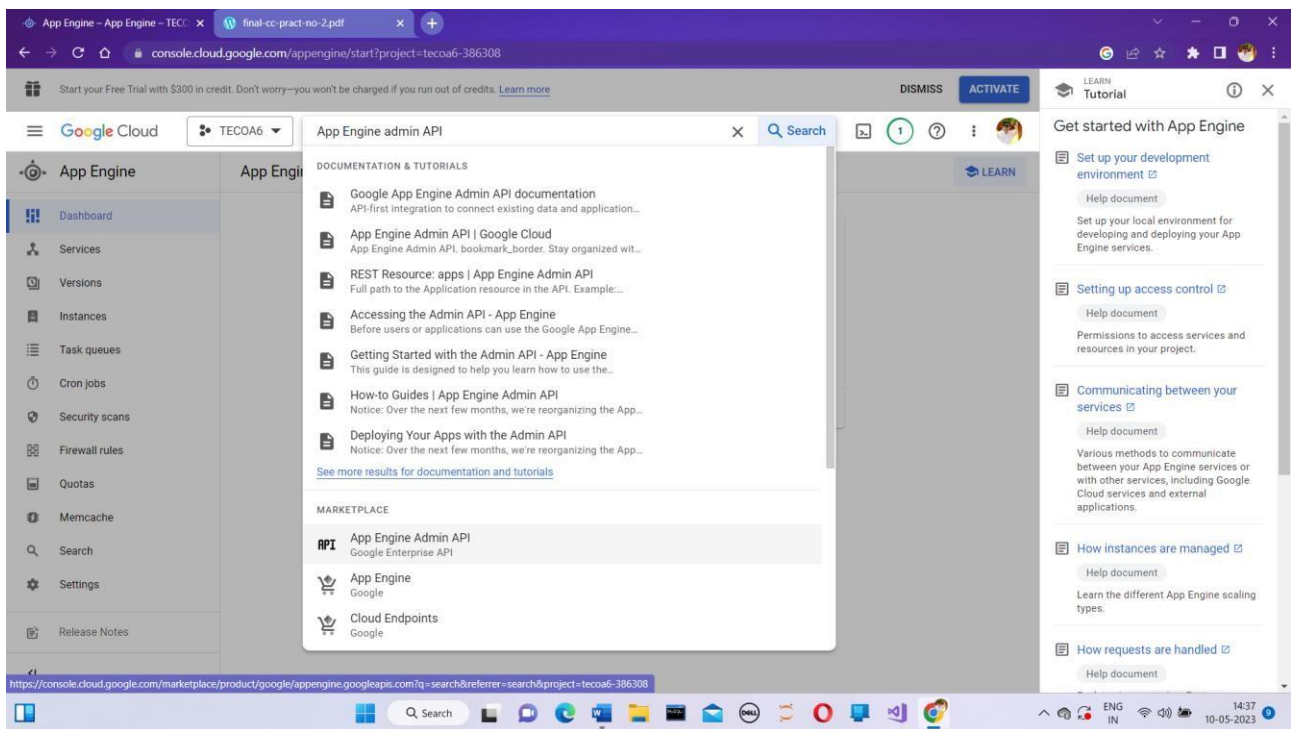
## Step 7:- Click on next



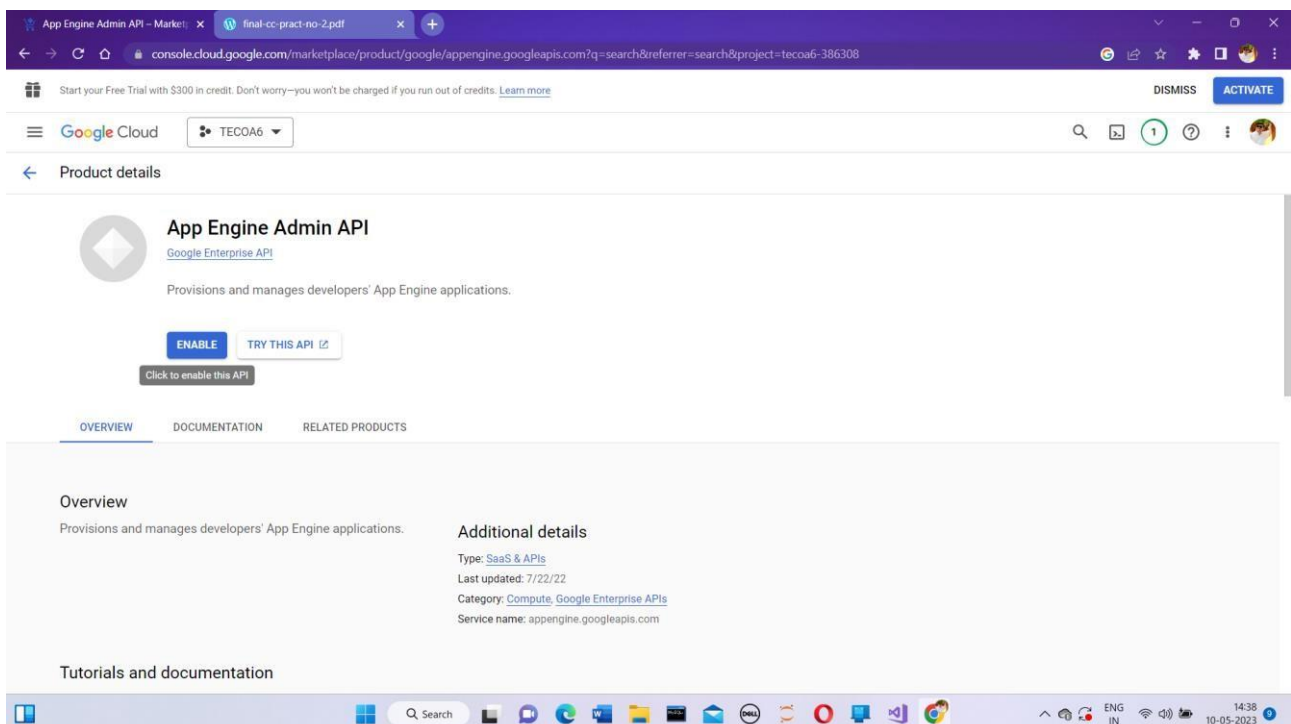
## Step 8:- Scroll down and click on I'll do this later.



Step 9:- In search bar type App Engine Admin API.

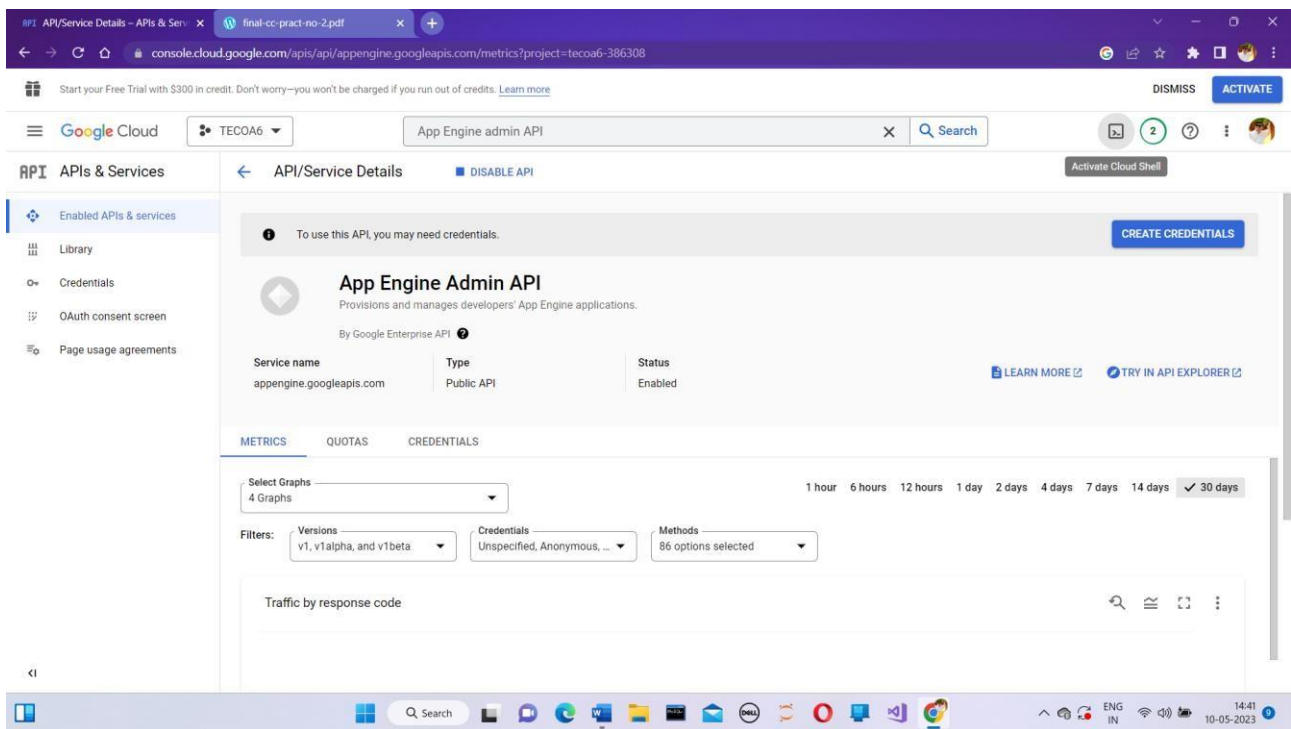


Step 10:- Click Enables

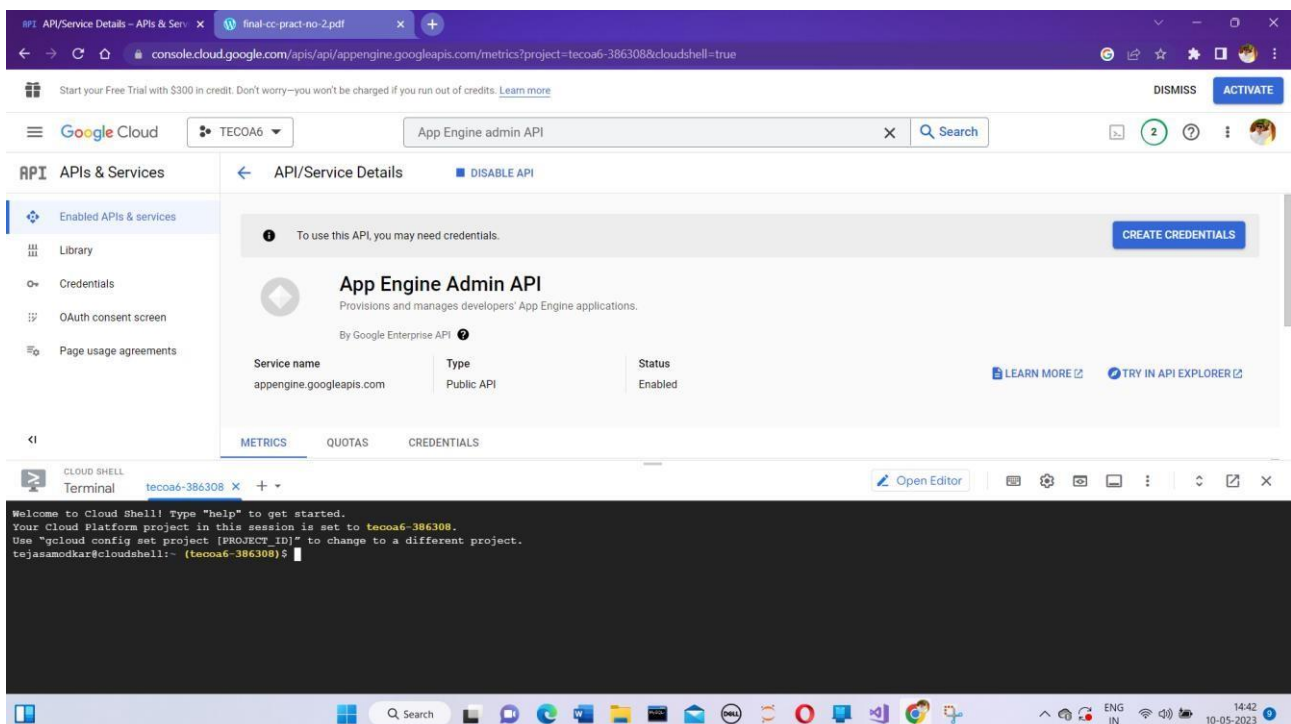




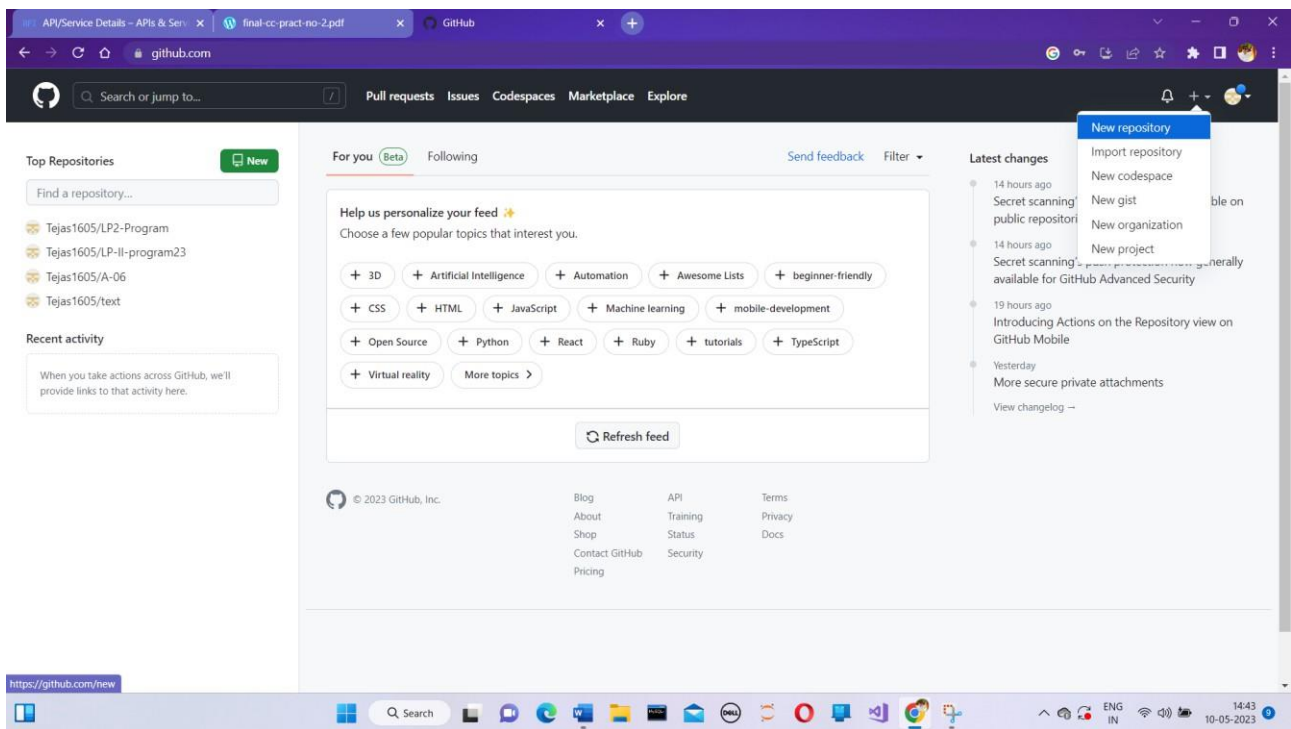
## Step 11:- Click Activate Cloud Shell:-



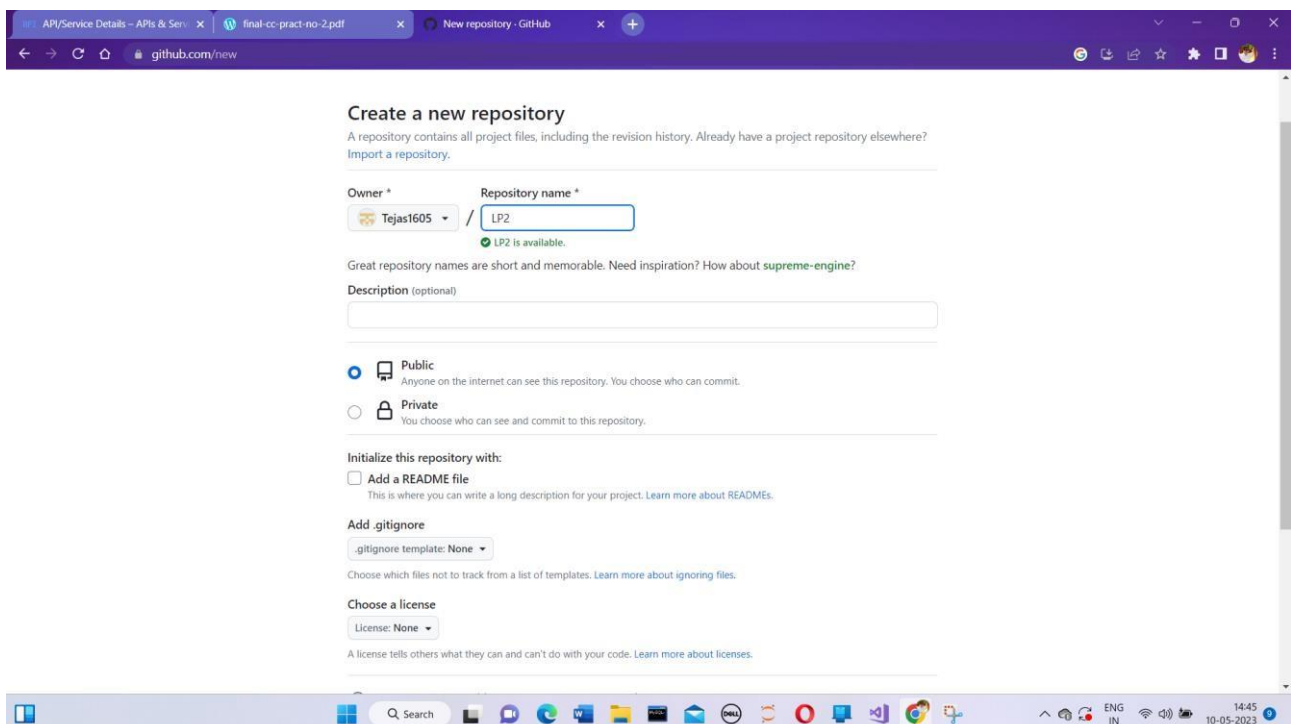
## Step 12 :- Following screen will appear:-



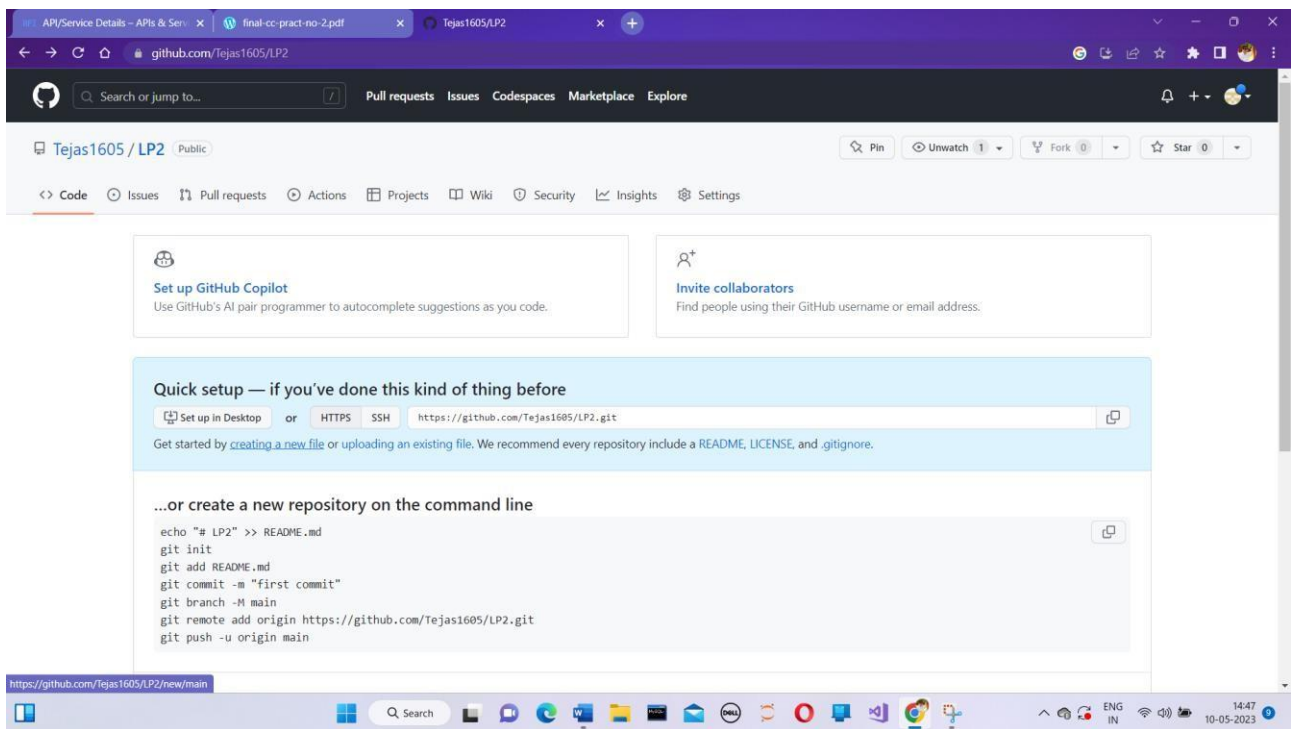
Step 13:- Login into your GitHub account and click on new repository.



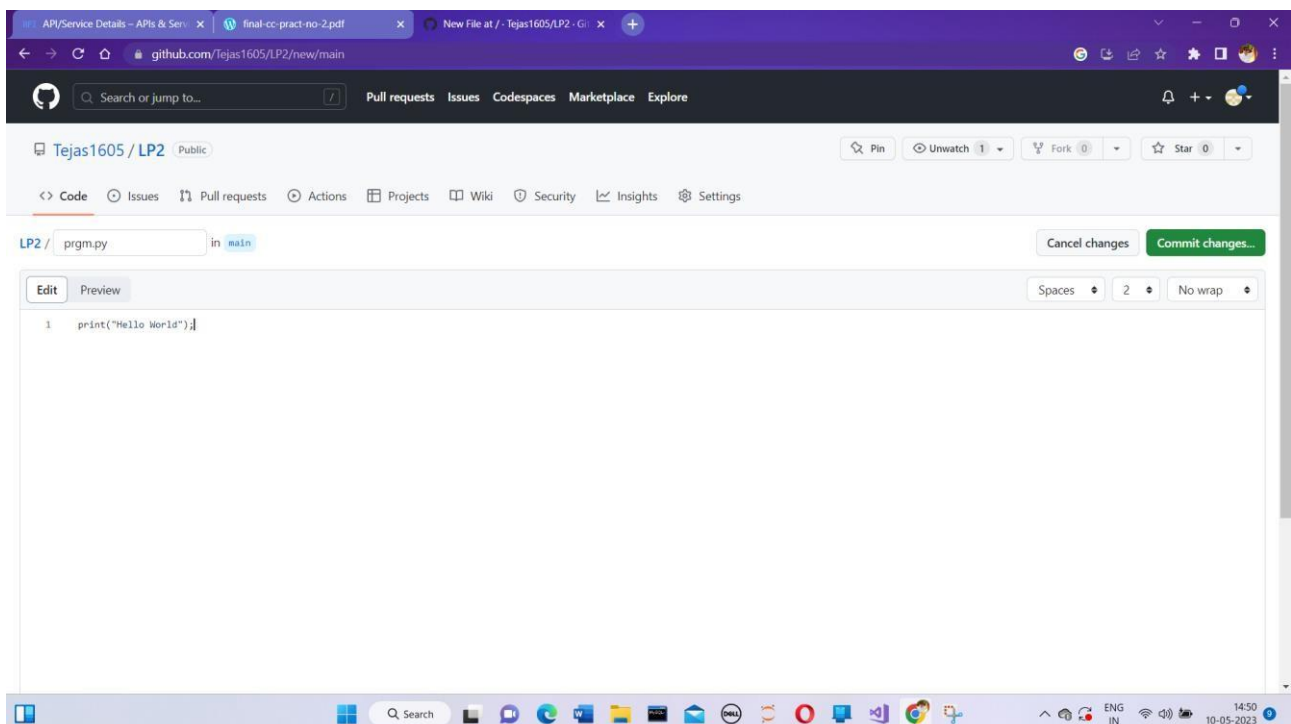
Step 14 :- Give name to your repository and click create.



## Step 15:- Click on creating new file

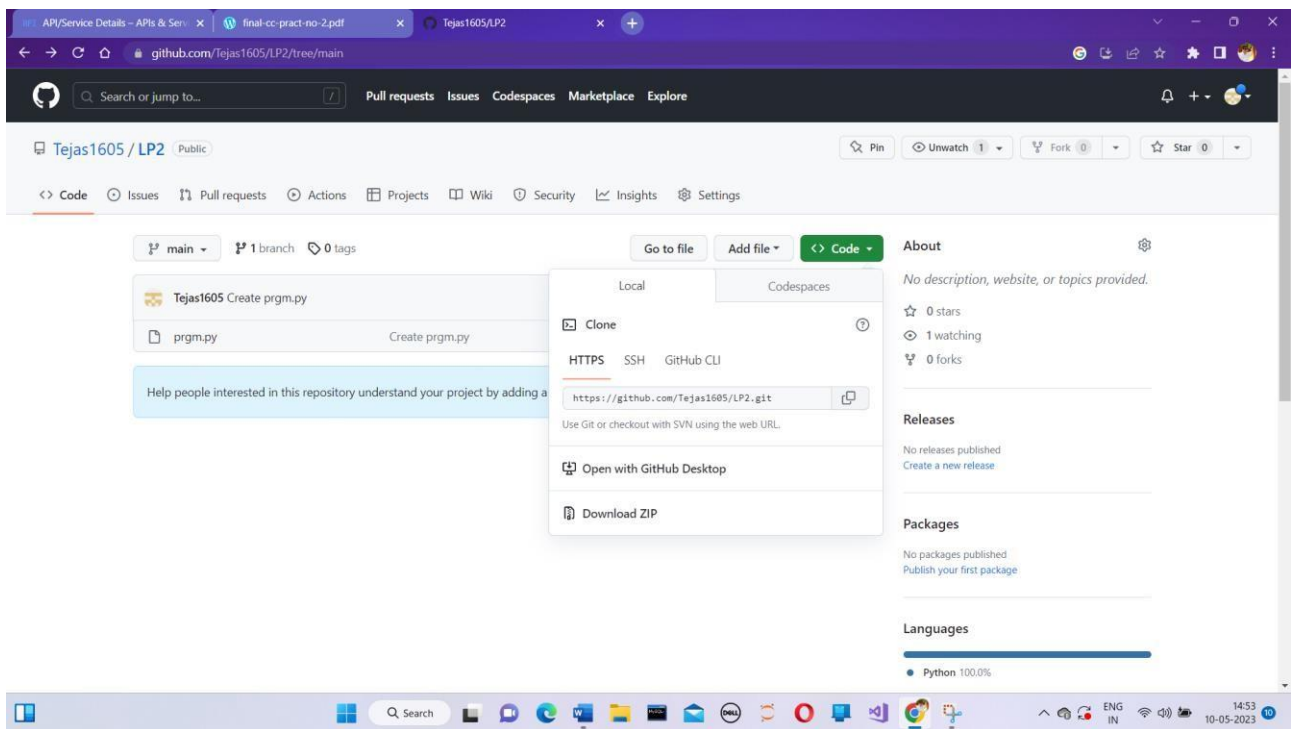


## Step 16 :- Give name to the python file & type your code.

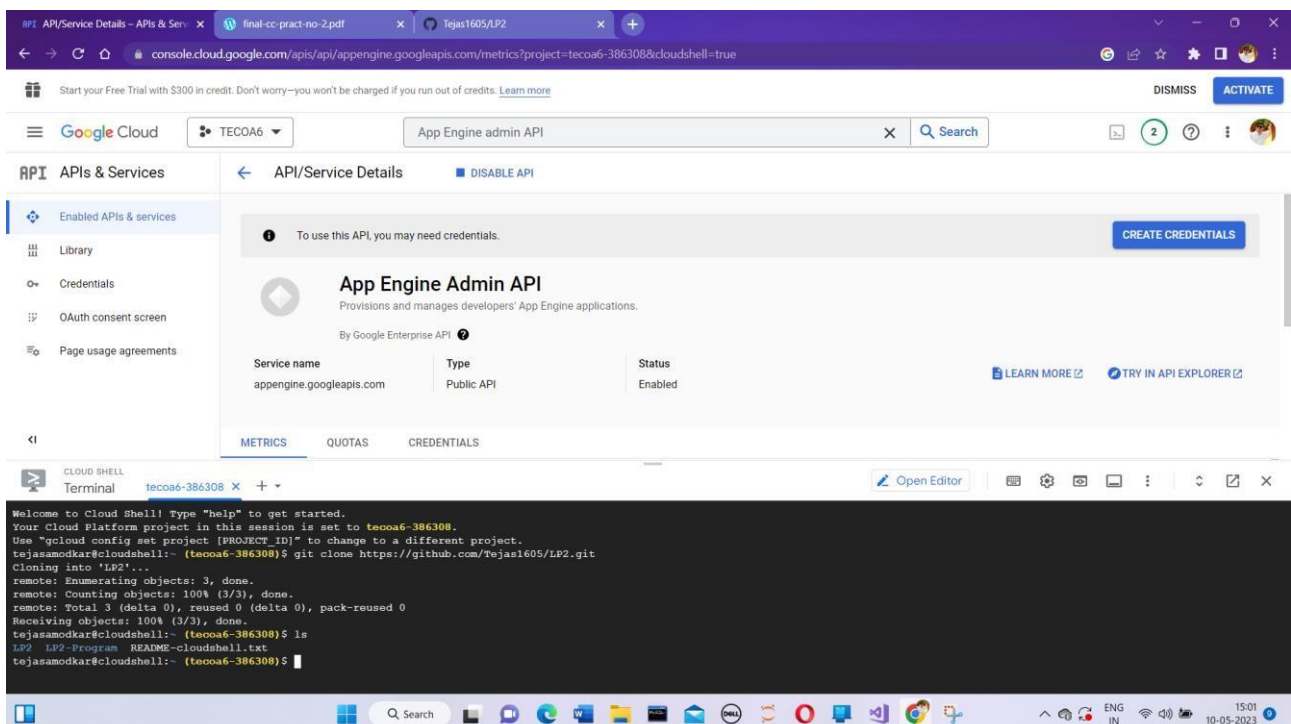




Step 17:- Click on code and copy URL.



Step 18:- Go to cloud and type – git clone and paste url.



## Step 19 :- Type ls

The screenshot shows the Google Cloud Console interface. The top navigation bar includes the Google Cloud logo, a dropdown menu for 'TECOA6', and a search bar. The left sidebar shows the 'APIs & Services' section with 'Enabled APIs & services' selected. The main content area displays the 'App Engine Admin API' details, including its service name 'appengine.googleapis.com', type 'Public API', and status 'Enabled'. Below this, there are tabs for 'METRICS', 'QUOTAS', and 'CREDENTIALS'. At the bottom, a 'CLOUD SHELL' terminal window is open, showing the command prompt 'tecoa6-386308' and the output of the 'ls' command, which lists files in the current directory.

## Step 20 :- Enter cd-repository name.

The screenshot shows the Google Cloud Console interface, similar to the previous one. The 'App Engine Admin API' details are visible. The 'CLOUD SHELL' terminal window at the bottom shows the command prompt 'tecoa6-386308' and the output of the 'cd LP2' command, which changes the directory to 'LP2'. The terminal also shows the output of the 'ls' command, listing files in the 'LP2' directory.

Step 21:- Type ls and to run python code type python-program name.

The screenshot displays the Google Cloud console interface. The top navigation bar shows the Google Cloud logo, the project name 'TECOA6', and the 'App Engine admin API' search bar. The left sidebar lists 'APIs & Services' with options like 'Enabled APIs & services', 'Library', 'Credentials', 'OAuth consent screen', and 'Page usage agreements'. The main content area shows the 'App Engine Admin API' details, including its service name 'appengine.googleapis.com', type 'Public API', and status 'Enabled'. A 'CREATE CREDENTIALS' button is visible. Below this, there are tabs for 'METRICS', 'QUOTAS', and 'CREDENTIALS'. At the bottom, a 'CLOUD SHELL' terminal window is open, showing the following commands and output:

```
Cloning into 'LP2'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
tejasmodkar@cloudshell:~ (tecoa6-386308)$ ls
LP2 prgm.py README-cloudshell.txt
tejasmodkar@cloudshell:~ (tecoa6-386308)$ cd LP2
tejasmodkar@cloudshell:~/LP2 (tecoa6-386308)$ ls
prgm.py
tejasmodkar@cloudshell:~/LP2 (tecoa6-386308)$ python prgm.py
Hello World
tejasmodkar@cloudshell:~/LP2 (tecoa6-386308)$
```