**Procedure:**

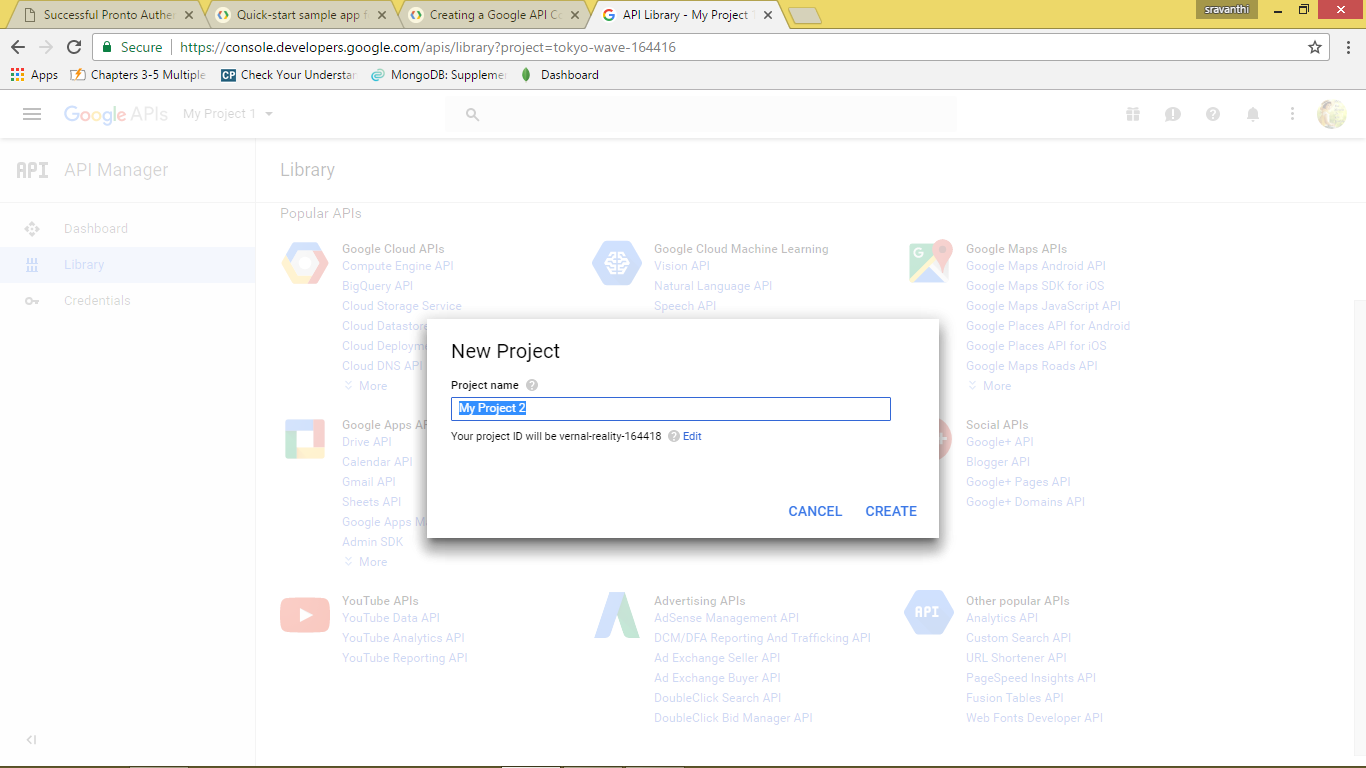
To create a Google API Console project and client ID:

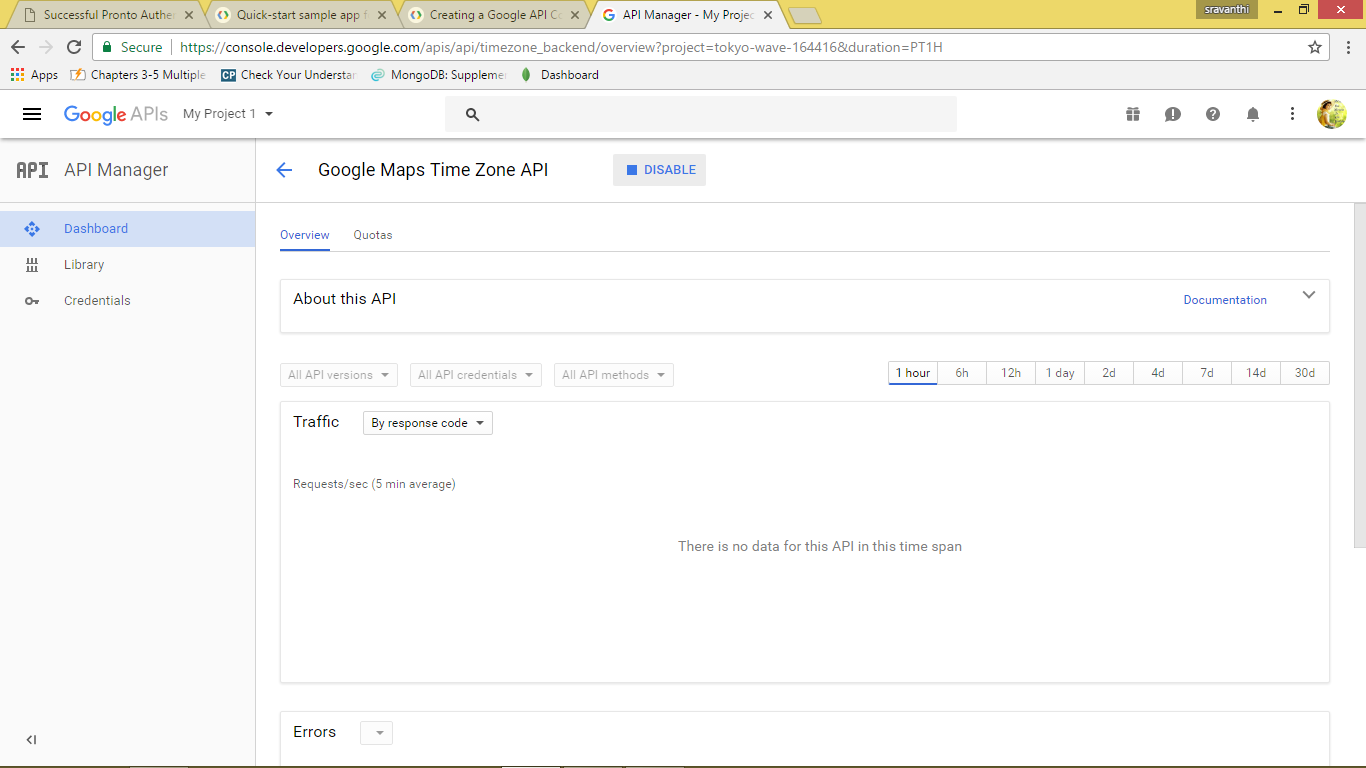
1. Go to the [Google API Console](https://console.developers.google.com/project/_/apiui/apis/library)
2. From the project drop-down, select an existing project or create a new one by selecting Create a new project.
3. In the sidebar under "API Manager", select Credentials, then select the OAuth consent screen tab.
   1. Choose an Email Address, specify a Product Name, and press Save.
4. In the Credentials tab, select the New credentials drop-down list, and choose OAuth client ID.
5. Under Application type, select Web application.   
   Register the origins from which your app is allowed to access the Google APIs, as follows. An origin is a unique combination of protocol, hostname, and port.
   1. In the Authorized JavaScript origins field, enter the origin for your app. You can enter multiple origins to allow for your app to run on different protocols, domains, or subdomains. You cannot use wildcards. In the example below, the second URL could be a production URL.

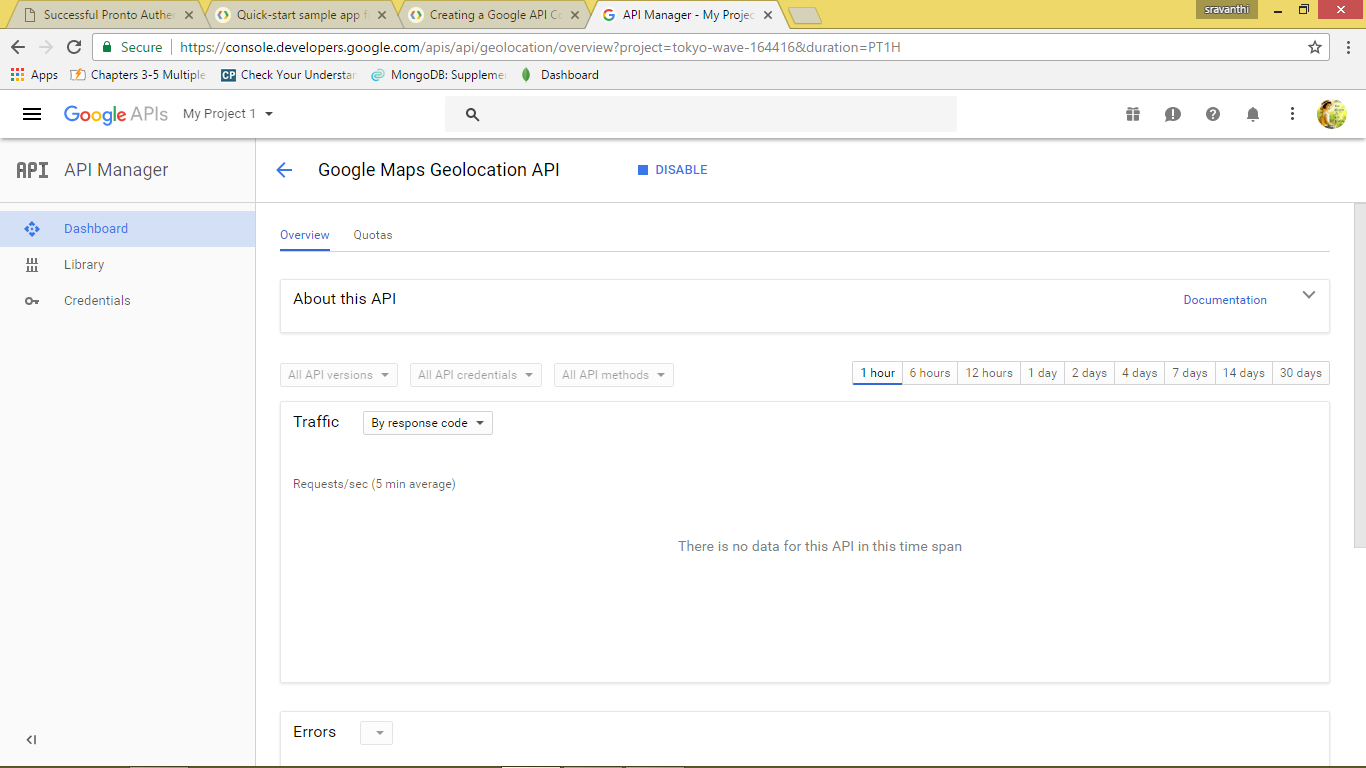
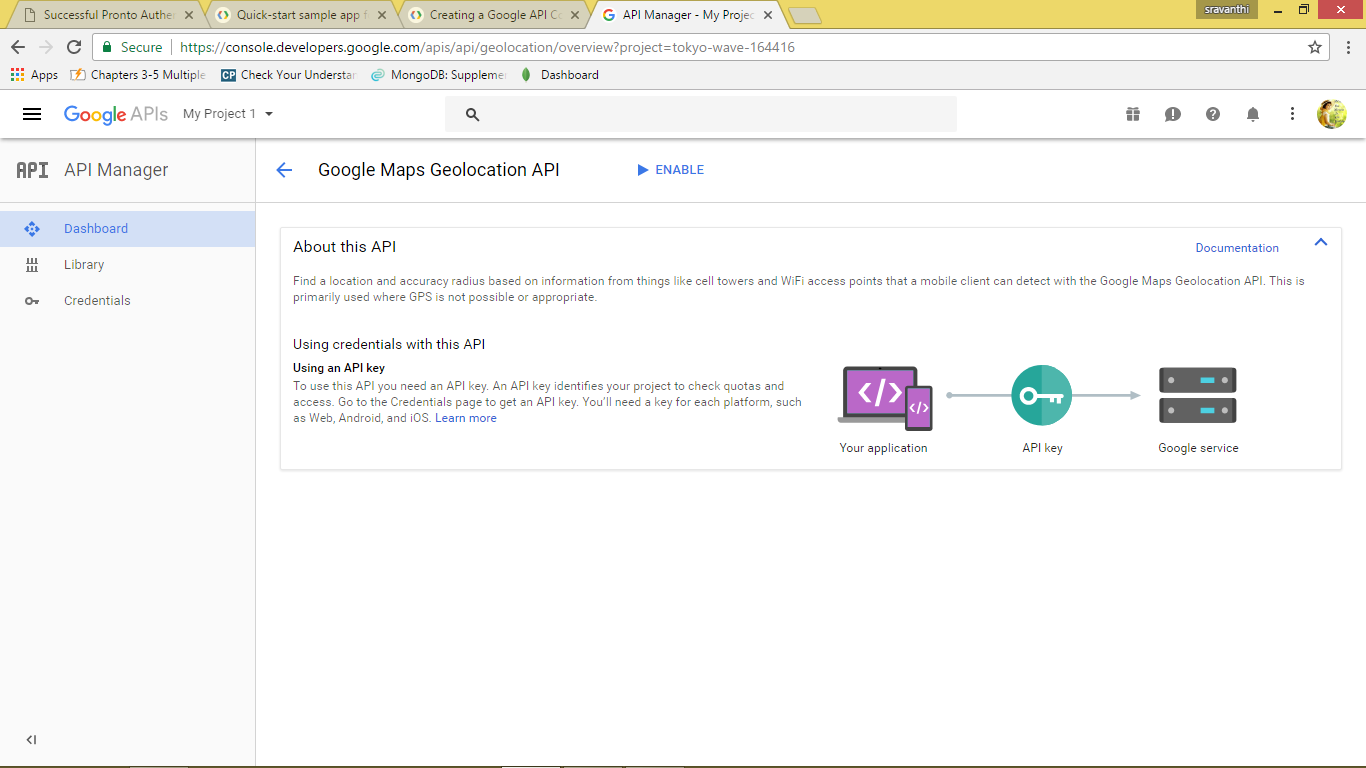
http://localhost:8080  
https://myproductionurl.example.com

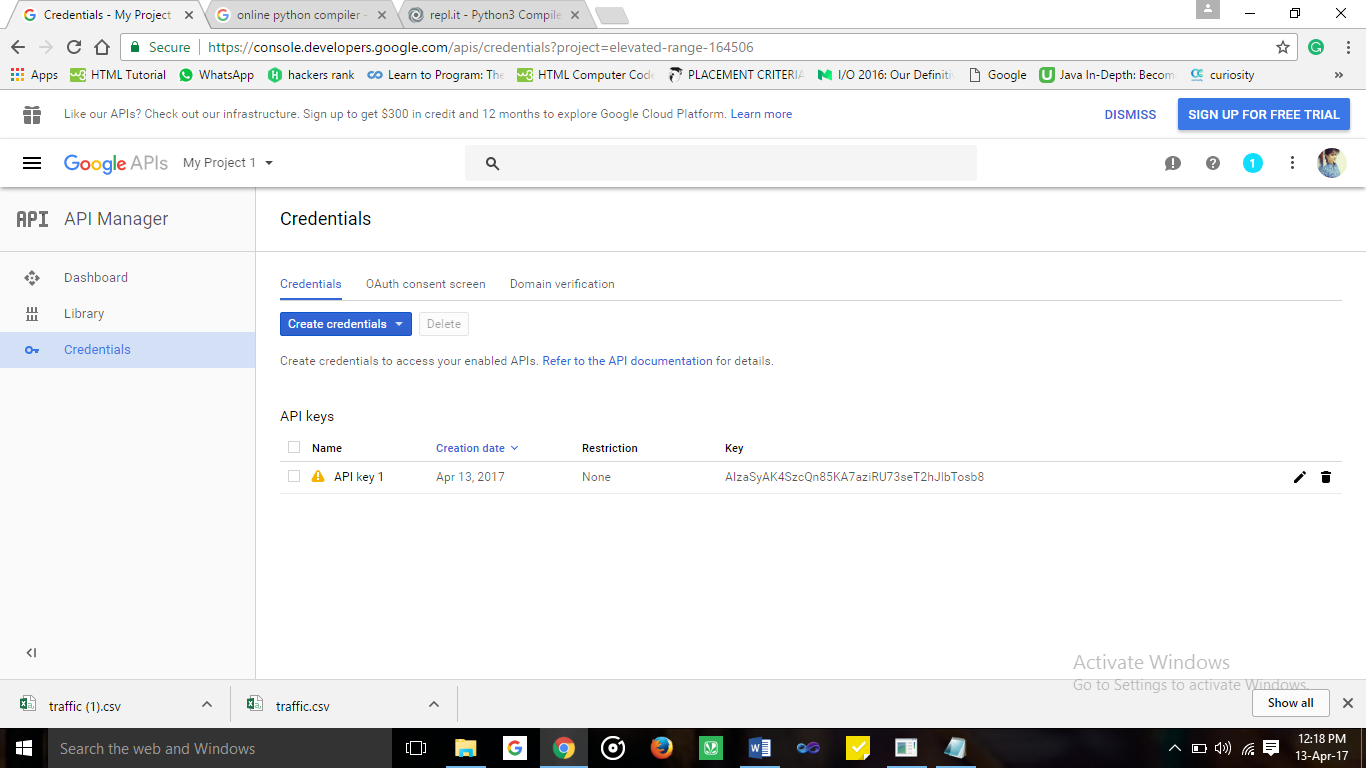
* 1. The Authorized redirect URI field does not require a value. Redirect URIs are not used with JavaScript APIs.
  2. Press the Create button.

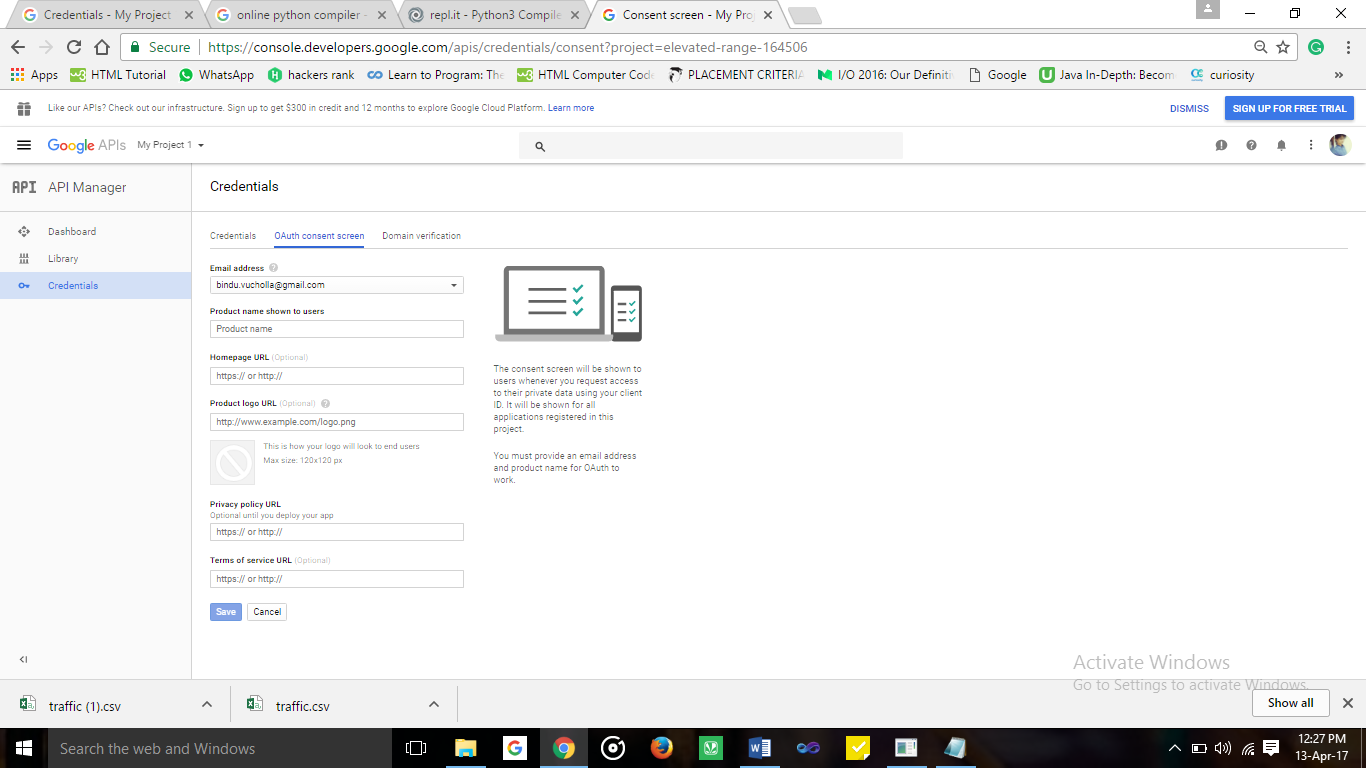
1. From the resulting OAuth client dialog box, copy the **Client ID** . The Client ID lets your app access enabled Google APIs.











Key is generated copy the key.

Code:

import urllib.request

import json

li = "\*"

timezone\_url = "https://maps.googleapis.com/maps/api/timezone/json?"

geocode\_url= "https://maps.googleapis.com/maps/api/geocode/json?"

#################################### FOR GEOCODE #############################################

list=input('Enter address: ')

print(li\*35 + " GEO CODE API-1 " + li\*35)

APIkey=’ AIzaSyACAc6V2kfRKLmVdq3yNXij6ULhk\_OnsRg’

ts=1458000000 #timestamp on 8:58 pm

url = geocode\_url + urllib.parse.urlencode({'address':list,'key':APIkey})

u = urllib.request.urlopen(url)

data = u.read()

j = json.loads(data.decode('utf-8'))

#print(j)

geocode = j['results'][0]['formatted\_address']

print ("Address : "+geocode)

long = j['results'][0]['geometry']['location']['lng']

lat = j['results'][0]['geometry']['location']['lat']

#print(j)

print('latitude =',lat,',longitude =',long)

#################################### FOR TIMEZONE ############################################

print(li\*35 + " TIME ZONE API-2 " + li\*35)

an\_url = timezone\_url + urllib.parse.urlencode({'location':'12.969264,79.15593799999999','timestamp':ts,'key':APIkey})

an\_u = urllib.request.urlopen(an\_url)

an\_data = an\_u.read()

a\_j = json.loads(an\_data.decode('utf-8'))

tz = a\_j['timeZoneName']

tz\_ID = a\_j['timeZoneId']

print ("Time Zone Name : "+tz+ "\nTime Zone : "+tz\_ID)

output:

