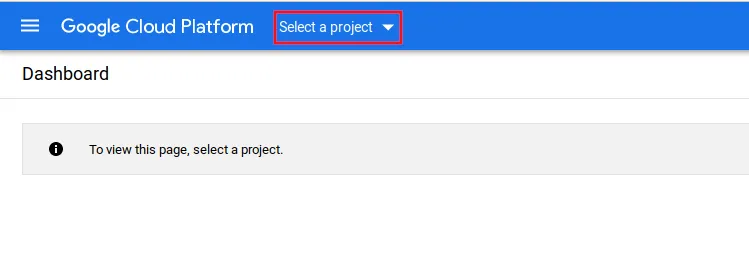
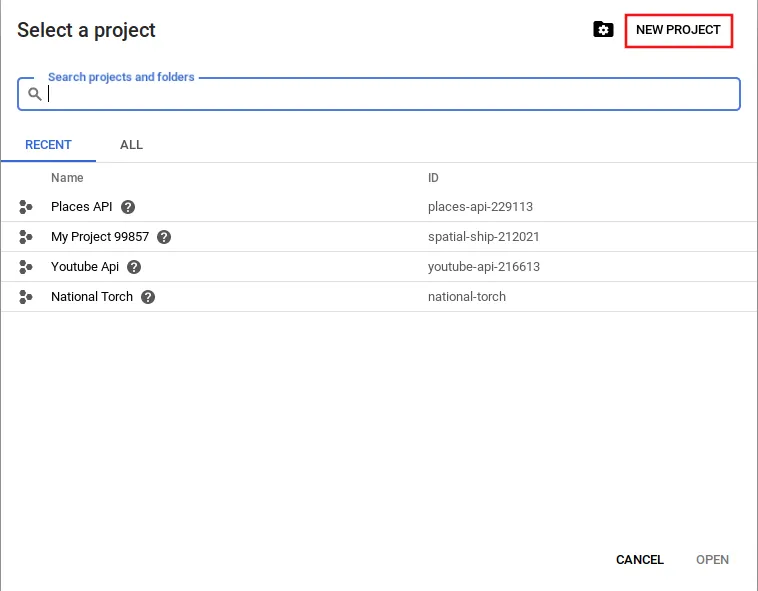
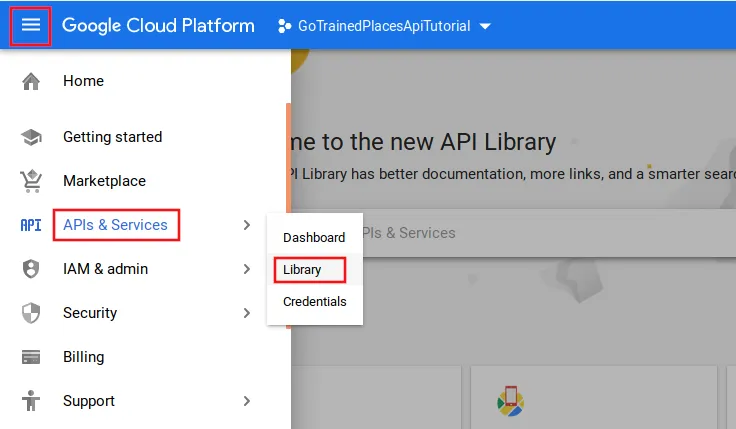
**How to use Google Maps API**

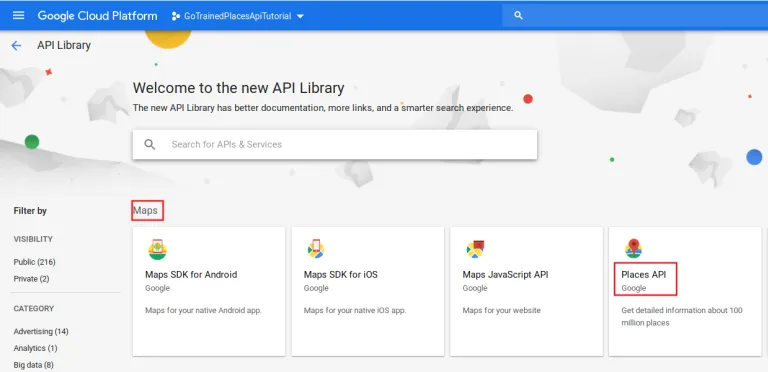
* First, we have to create an account in Google Cloud Console
* Once you are into the google cloud console from top navigation bar click “Select a project”
* In the new window click “New project” and create a new one



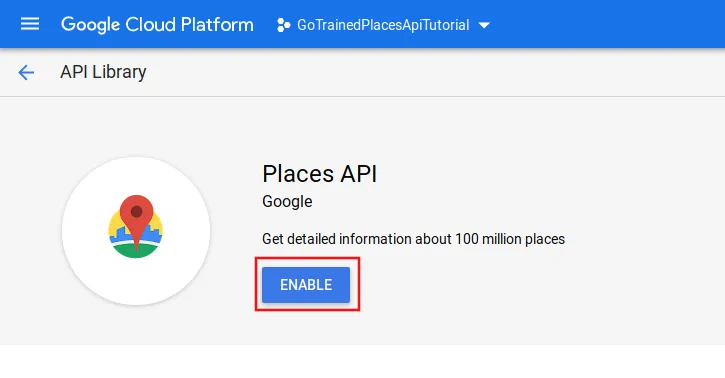
* From left side navigation go to “APIs & Services -> Library”



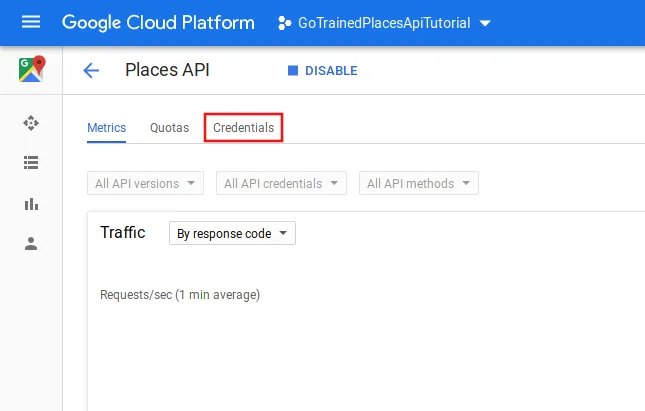
* From Maps section select “Places API” or search for “Places API” in the search box.



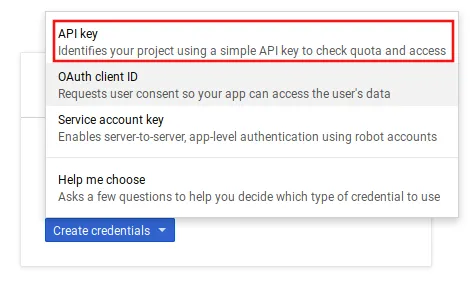
* Click on “Enable”



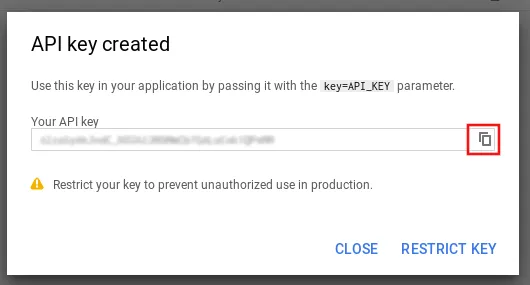
* Go to Credentials tab



* Click on Create credentials and then “API Key”



* So right now you’ve got your Google API key successfully. Now let’s get started with coding.



**Nearby Search Method**

The Places API allows you to query for place information on a variety of categories, such as: establishments, prominent points of interest, geographic locations, and more. You can search for places either by proximity or a text string. A Place Search returns a list of places along with summary information about each place; additional information is available via a Place Details query.

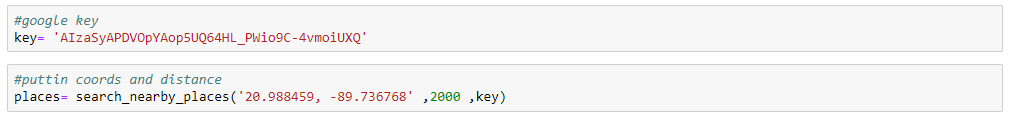
* Before star to code we need to import some libraries



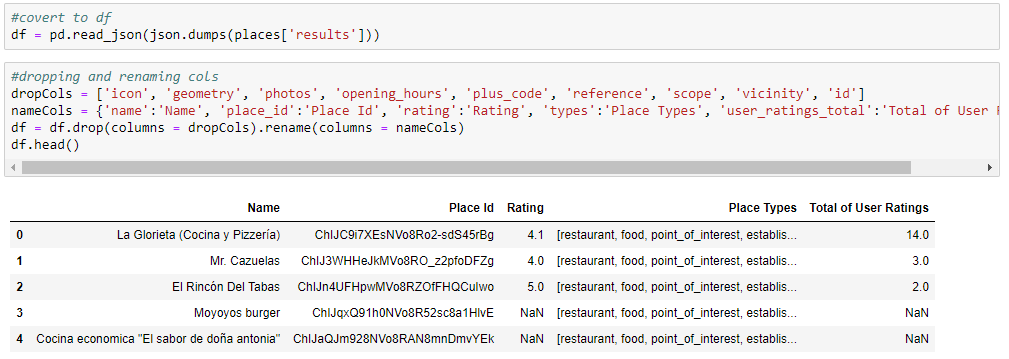
* The code is divided by functions which return a response with the data from the places in a json format, also we have to use the url which is for nearbysearch that we can look for in the documentation of google (<https://developers.google.com/places/web-service/search>).
* This one is a function to find nearby restaurants to the given location in a radius in meters, we introduce some parameters which are: location, radius, key and the type which is restaurant,



* Once we code the function we need to proceed to declare our key, the location of the place and the radius in meters



* Once we have the results, we have to convert them into a data frame, and here I drop and rename some columns for the data frame aesthetic



* The next is a function to find the id from a place when we input the name and location data, the url is changed to findplacefromtext and the parameters needed are name, location, key and the inputtype



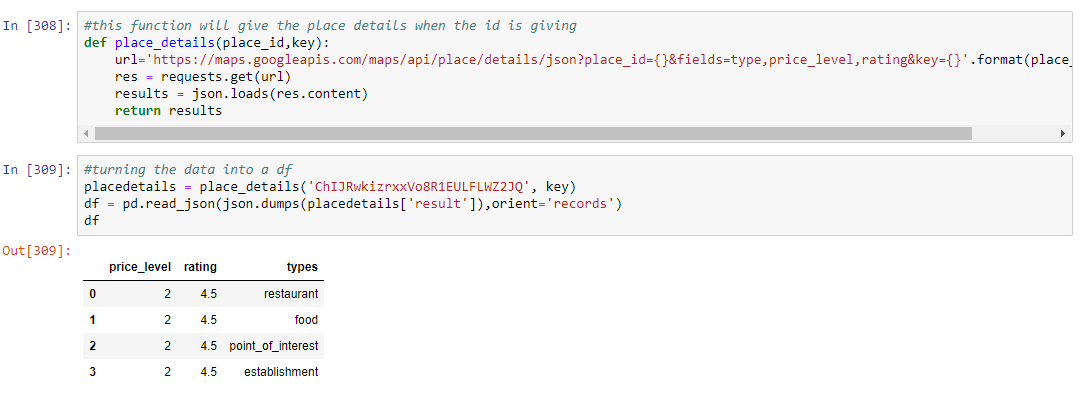
* The following function is to find the details from a place which includes review, rating, etc. Here we have to change in the url to details and put the language default in spinach or English, the parameters we need are the place id that we can get with the last function and our key



-Once we get the data we have to turn the json response into a df for a better visualization, here is the code that shows the example, also if you want you can drop and rename some columns for a better visualization.



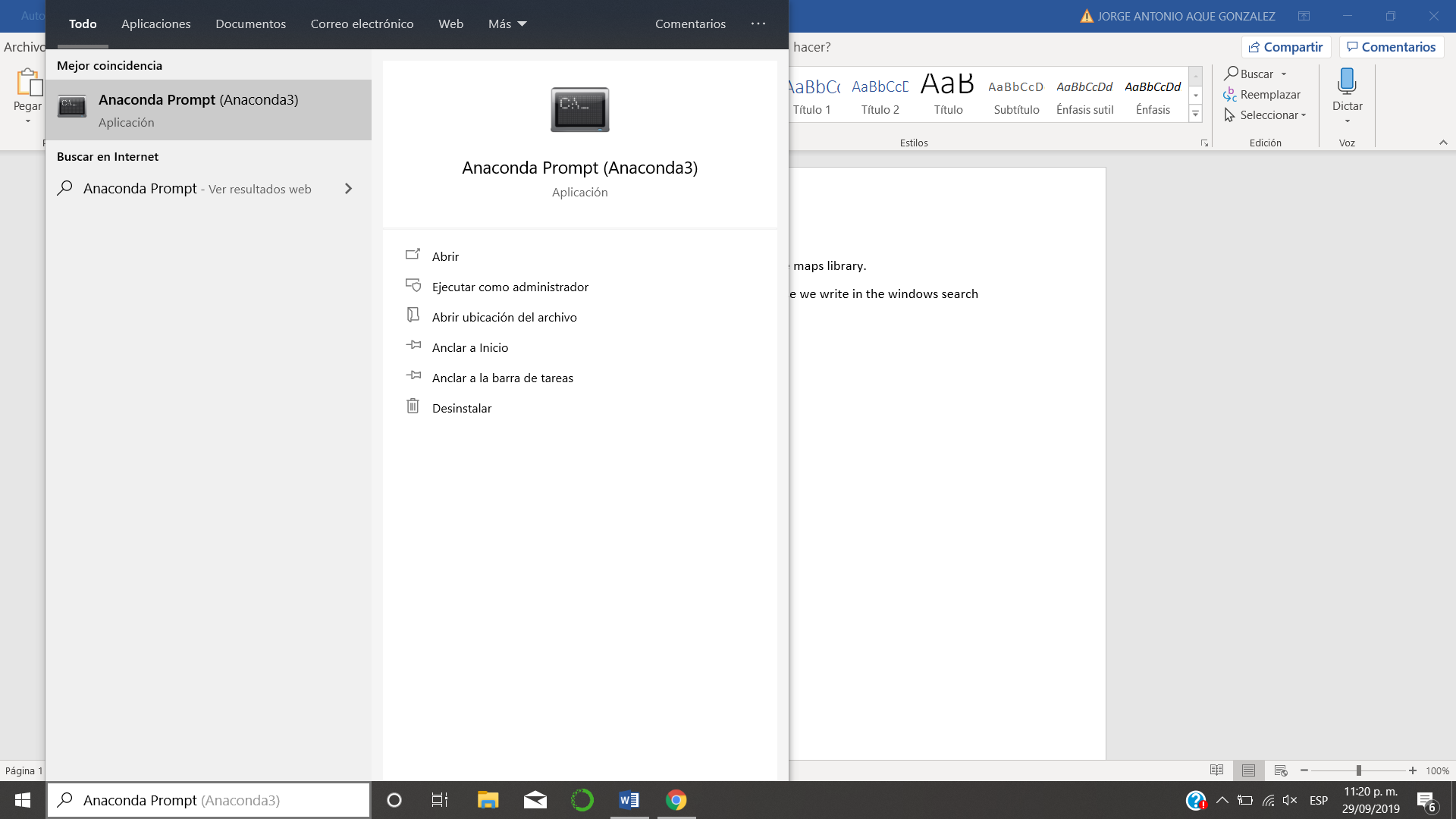
* To continue here there is a very similar function called place\_detail and the difference here is that we add an extra parameter which is the price level, this one it will be returned into the json response and is the same process to convert it into a data frame

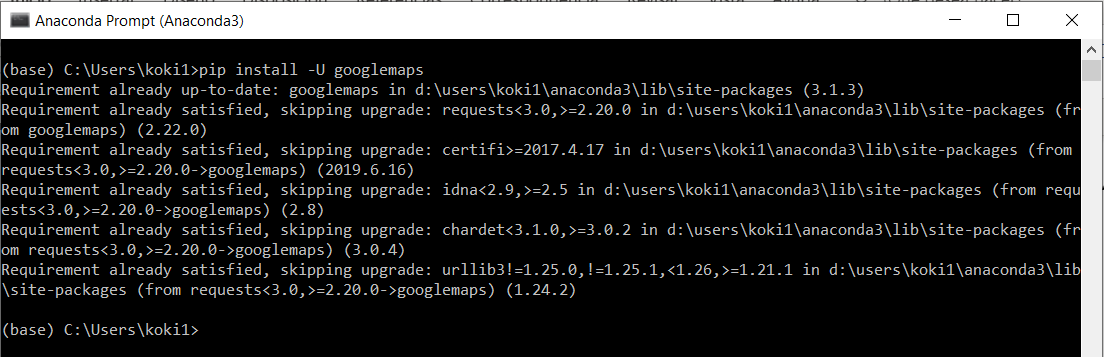


* Applying the last function we can also look for similar places in base to the price level so in this new function we use the price level as a new parameters and the function will return the data from similar places.



* Now, We will work whit the Google´s API but using the google maps library.
* First, we need to install the google maps library, so, in this case we write in the windows search “anaconda prompt”.



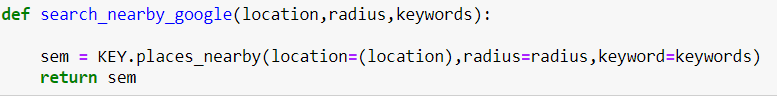
* Then we click it. Once inside we need to write “pip install -U googlemaps” and press enter. And we will have already installed googlemaps library
* Those are the libraries that we need to import, this is the way to use them.



* Here is you will put your key, you will use it (“your key “ = \*\*\*\*)



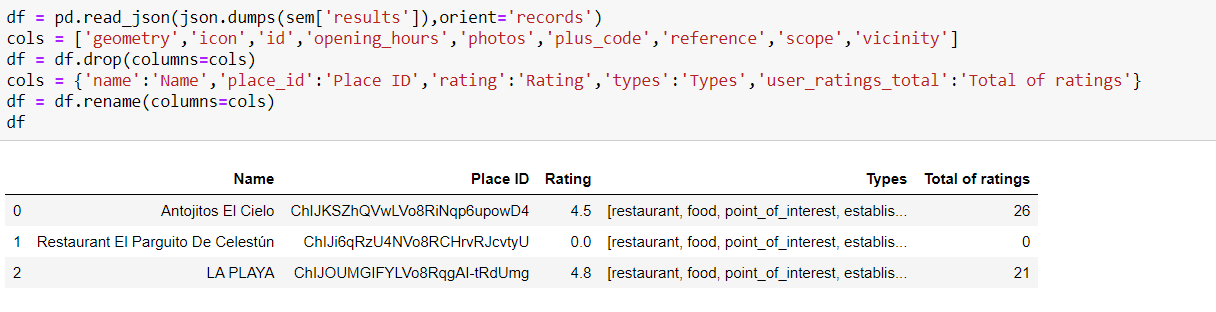
* So, first we create the function and declare the parameter as the google´s documentation



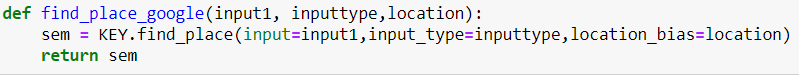
* Here we will create a variable and we will use the function, we need to put the parameters in the same position



* Once done we create a Data Frame to show the results, and here we deleted the columns that we won´t use with .drop()



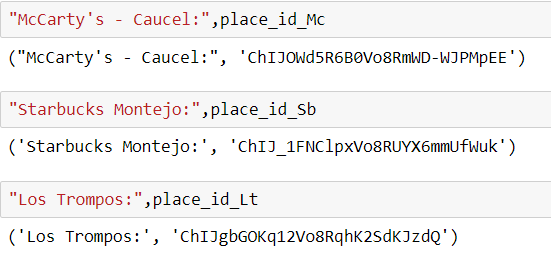
* This are the function using the parameters. Something important is that we always use the API key as you can appreciated.



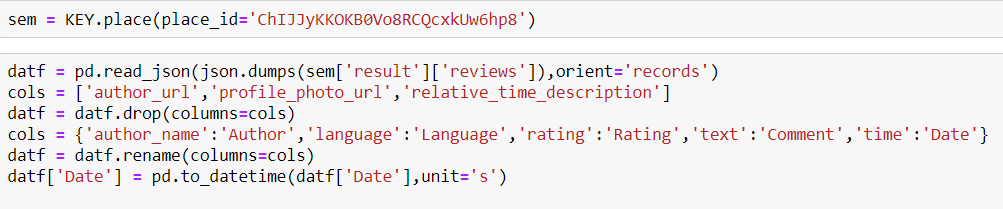
* We Will use the find\_place\_google function to find those places, so we can declare one variable for each place. We need to put the parameters to find the places correctly.



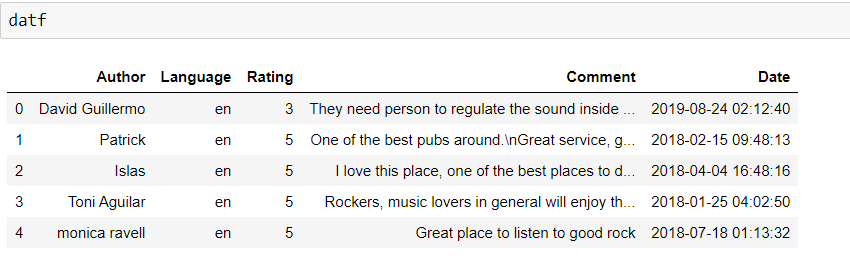
* And this is the form that we can do the output. As we can see, we print the place´s name and the Place ID



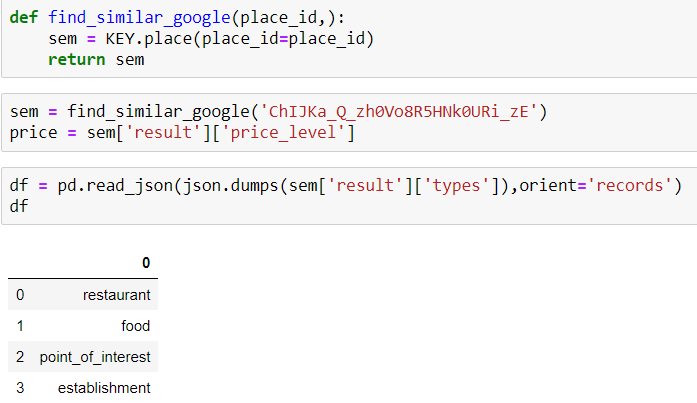
* Here we use a Place ID as an example to do methon Place Detail, Here we deleted the columns that we will no use and make use of the library datetime to change the time format.



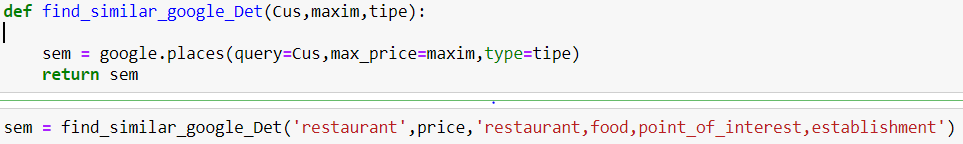
* This is the outputs form, as we can see, the time format is easier to read



* Here we need to create the function and declare the parameter, in this case the place Id. We need to use Los Trompos ID. We fill the parameter and the we create a data frame to watch the results.



* The second point is to create a function that return similar place, we need to declare the parameter to return the correct results that we want.



* Once done we create a data frame to watch the results of the coincidences. This is the way that we can use the google maps library.

