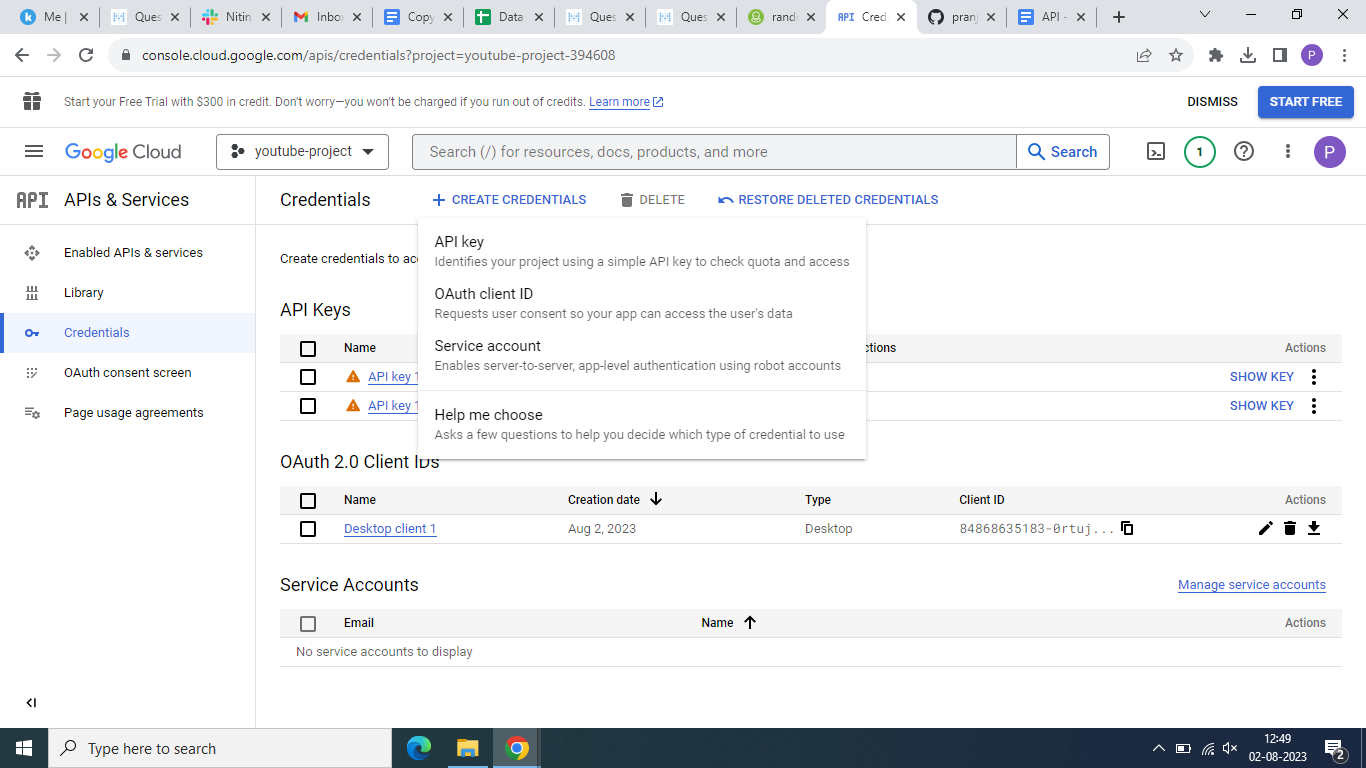
# **Fetch Data From An API**

In this project, we have established a connection between our API and our local system.We will save the received file in csv format on our local system.

1. Youtube -API
2. Random user API
3. IMDB API

# **Youtube API :**

## **Step 1: Set Up the Gmail API and Obtain Credentials**

1. Go to the Google Cloud Console (<https://console.cloud.google.com/>).
2. Create a new project (if you don't have one) or select an existing project.
3. In the left navigation menu, go to "APIs & Services" > "Library."
4. Search for "Gmail API" and enable it for your project.
5. In the left navigation menu, go to "APIs & Services" > "Credentials."
6. Create credentials and only API KEY
7. Copy your API key and you’ll have to paste it in your code.
8. Now to find the channel id, you can select any channel and look for its channel id on this : <https://commentpicker.com/youtube-channel-id.php>
9. Now get the channel id and paste it in the given code.

## **Step 2: Run the Code**

Once you have completed the authentication process, the code will execute. It will fetch the first email from your inbox, check if there is a CSV attachment, load the CSV data into a DataFrame, print the column names and the original DataFrame, drop the first and second rows, and save the modified DataFrame as a new CSV file named 'modified\_dataframe.csv' on your desktop.

## **Step 3: Verify the Output**

After the code execution, check the output in your terminal or command prompt to see the printed column names, original DataFrame, and modified DataFrame. Additionally, you should find the 'modified\_dataframe.csv' file on your desktop containing the modified DataFrame in CSV format.

# **Random User API :**

**Step 1: Install required libraries**

Make sure you have the requests library installed. If you don't have it, you can install it usingpip command:

pip install requests

**Step 2: Import the necessary libraries**

import requests

import json

**Step 3: Make the API request and retrieve the data**

**import pandas as pd**

# Replace 'path/to/your/attachment.csv' with the actual file path of the downloaded attachment

df = pd.read\_csv('path/to/your/attachment.csv')

**Step 4: Save the retrieved data locally**

With these steps, you'll be able to connect to the Random User API, fetch the data, and save it locally as a JSON file named "random\_users.json" (you can choose a different name if you prefer). The data will be formatted with an indentation of 4 spaces for better readability. Remember to handle any potential exceptions and errors that might occur during the process to ensure your code is robust.

# Drop the first two rows and reset the index

df = df.iloc[2:].reset\_index(drop=True)

**Step 5: Verify the DataFrame**

You can check the DataFrame content to ensure the first two rows have been removed:

print(df.head()) # Display the first few rows of the DataFrame

**Step 6: Save the Updated DataFrame**

If you want to save the updated DataFrame as a new CSV file:

# Replace 'path/to/updated\_dataframe.csv' with the desired file path for the new CSV file

df.to\_csv('path/to/updated\_dataframe.csv', index=False)

# **IMDB API :**

* Import the required libraries: You'll need to import the requests library to make API requests and the pandas library to work with DataFrames.
* Set up the API URL and headers: Define the API URL for the IMDB API and the required headers (e.g., X-RapidAPI-Key and X-RapidAPI-Host) to access the API.
* Make the API request: Use the requests.get() method to make the API request and get the response from the IMDB API.
* Extract the data from the API response: Parse the JSON response obtained from the API and extract the relevant data that you want to store.
* Create a DataFrame: Use the pandas.DataFrame() function to create a DataFrame with the extracted data.
* Save the DataFrame: Save the DataFrame to a CSV file or any other desired format.

**Process to obtain the X-RapidAPI-Key and X-RapidAPI-Host for the IMDB API:**

**Step 1: Create an account on RapidAPI**

Visit the RapidAPI website (https://rapidapi.com/) and create a free account if you don't have one already. Click on the "Sign Up" button in the top-right corner to create your account.

**Step 2: Search for the IMDB API**

Once you are logged in, use the search bar at the top to find the IMDB API. You can search for "IMDB" or "Movie Database" to find relevant APIs related to IMDB.

**Step 3: Select the IMDB API**

From the search results, click on the IMDB API that you want to use. This will take you to the API's documentation page.

**Step 4: Subscribe to the API**

On the API's documentation page, you'll find information about the API, endpoints, and pricing details. Click on the "Pricing" or "Subscribe" button to subscribe to the API.

**Step 5: Choose a plan**

Select a plan that suits your needs. RapidAPI usually offers both free and paid plans with different rate limits and features. If you are just testing or using the API for a small project, you can start with the free plan.

**Step 6: Obtain the API Key**

After subscribing to the API, you'll be provided with an API key. This key is unique to your account and will be used to authenticate your requests. Copy the API key from the RapidAPI dashboard or API documentation page.

**Step 7: Use the API Key in your Python Code**

In your Python code, replace "Your-API-Key" with the API key you obtained in Step 6. For example:

python

Copy code

headers = {

"X-RapidAPI-Key": "Your-API-Key",

"X-RapidAPI-Host": "imdb8.p.rapidapi.com"

}

**Step 8: Start Making API Requests**

With the API key and host set in your Python code, you can now start making requests to the IMDB API and use the data retrieved for your application.