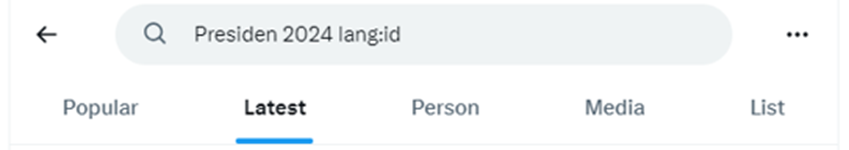
1. **CRAWLING DATA TWITTER (1 tanpaa batas waktu)**
2. **Twitter Authentikasi**
3. Pastikan bahwa kamu sudah login pada twitter.
4. Ketikkan judul yang ingin kamu cari (disini kami contohkan: presiden 2024 lang:id).



1. Lalu pilih latest untuk jumlah data crawling yang lebih maksimal

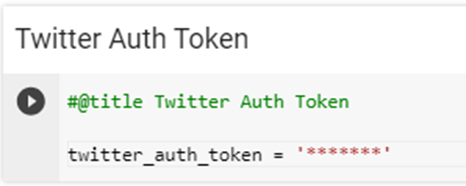


1. Seteleh itu klik kanan lalu pilih inspect
2. Pilihlah application(1), pilih cookies (2), kemudian pilih<https://twitter.com> (3), dan pilihlah auth\_token(4). Jika sudah dipilih maka nanti akan muncul seperti ini, copy saja token autentikasinya(5) (ini bersifat rahasia jadi gunakanlah seefektif mungkin.)





1. Buka google colabs lalu ketikkan kode seperti dibawah ini. Jika jangan lupa untuk menginputkan autentikasi token yang kita dapat dari twitter yang sudaah kita copy tadi. Lalu run (ganti (\*) dengan token kalian masing-masing)



1. **Import dan instalasi beberapa shell yang dieksekusi dalam lingkungan Python menggunakan sintaksis khusus (dimulai dengan !).**
2. Lakukan Pemasangan Paket Python Pandas:



1. lakukan instalasi Node.js

# Install Node.js (because tweet-harvest built using Node.js)

!sudo apt-get update

!sudo apt-get install -y ca-certificates curl gnupg

!sudo mkdir -p /etc/apt/keyrings

!curl -fsSL https://deb.nodesource.com/gpgkey/nodesource-repo.gpg.key | sudo gpg --dearmor -o /etc/apt/keyrings/nodesource.gpg

!NODE\_MAJOR=20 && echo "deb [signed-by=/etc/apt/keyrings/nodesource.gpg] https://deb.nodesource.com/node\_$NODE\_MAJOR.x nodistro main" | sudo tee /etc/apt/sources.list.d/nodesource.list

!sudo apt-get update

!sudo apt-get install nodejs -y

!node -v

1. **Crawling Data dari Twitter API.**

lakukan proses crawling dengan menggunakan kode seperti

# Crawl Data

filename = 'Presiden2024.csv'*(sesuaikan dengan nama file yang kalian mau)*

search\_keyword = 'Presiden2024 lang :id'*(sesuaikan dengan keyword masing-masing)*

limit = 1500

!npx --yes tweet-harvest@2.2.8 -o "{filename}" -s "{search\_keyword}" -l {limit} --token {twitter\_auth\_token}

1. **Membaca dan Menampilkan Data yang Sebelumnya Telah Dikumpulkan dari Twitter dan Disimpan dalam Format CSV.**

lakukan lah dengan menuliskan kode seperti berikut:

**import pandas as pd**

**# Specify the path to your CSV file**

**file\_path = f"tweets-data/{filename}"**

**# Read the CSV file into a pandas DataFrame**

**df = pd.read\_csv(file\_path, delimiter=";")**

**# Display the DataFrame**

**display(df)**

1. **Cek Data Yang diDapatkan**

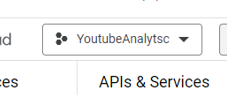
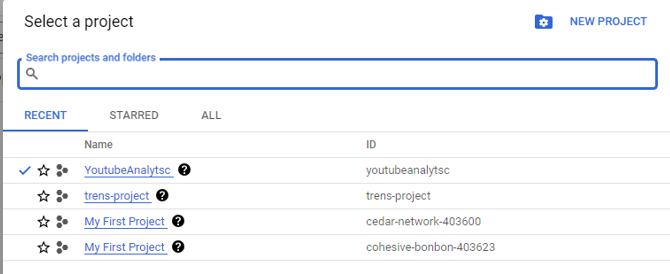
Lakukaan dengan menjalankan kode seperti dibawah ini:

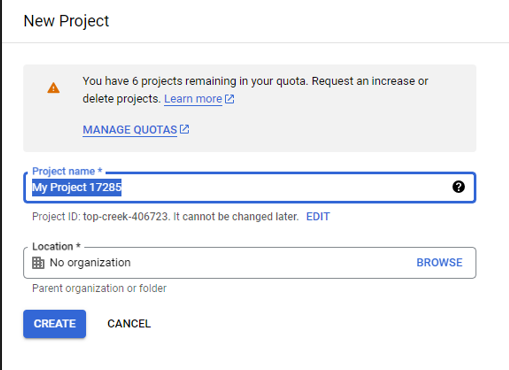
**# Cek jumlah data yang didapatkan**

**num\_tweets = len(df)**

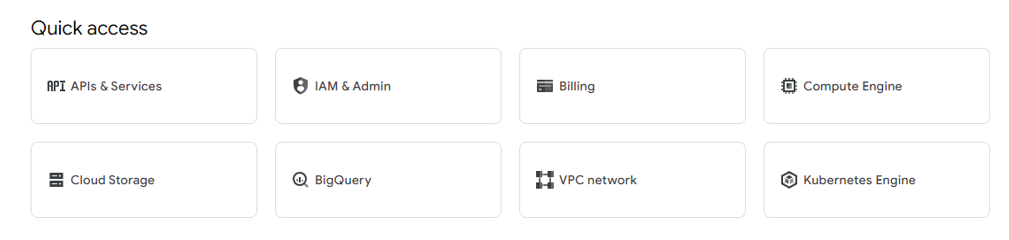
**print(f"Jumlah tweet dalam dataframe adalah {num\_tweets}.")**

1. **CRAWLING DATA YOUTUBE**
2. **Menyiapkan google cloud function**
3. Lakukan login atau registrasi google cloud function
4. buat New Project

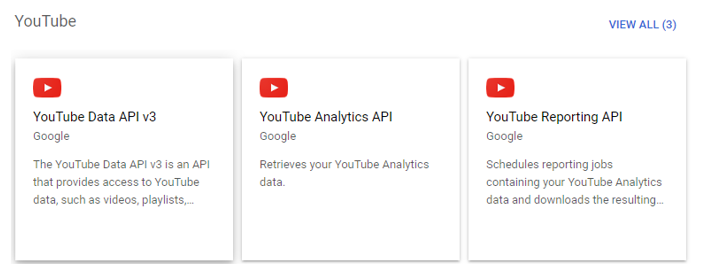
 



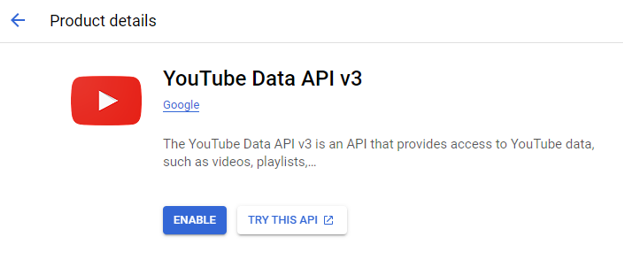
1. Jika sudah berhaasil create new project, masuk dan cari Quick Access. kemudian pilih APIs & Services



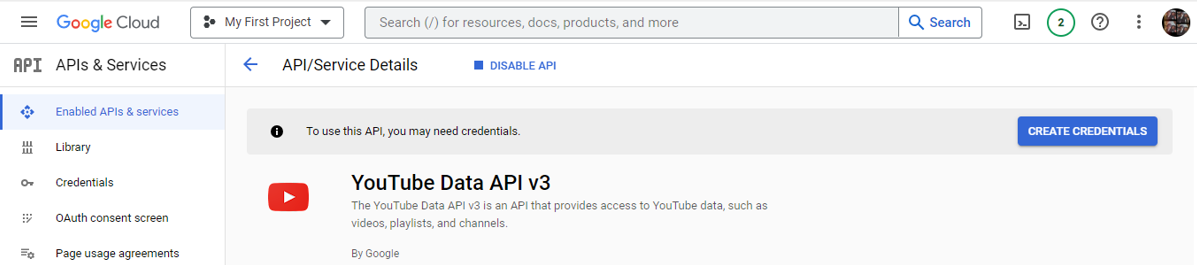
1. scrool ke bawah hingga menemukan Youtube seperti dibawah, kemudian pilih Youtube Data API v3



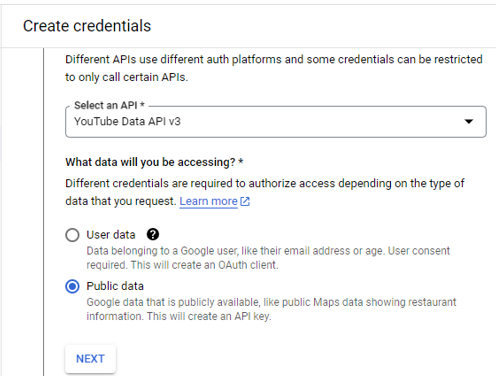
1. Pilih ENABLE



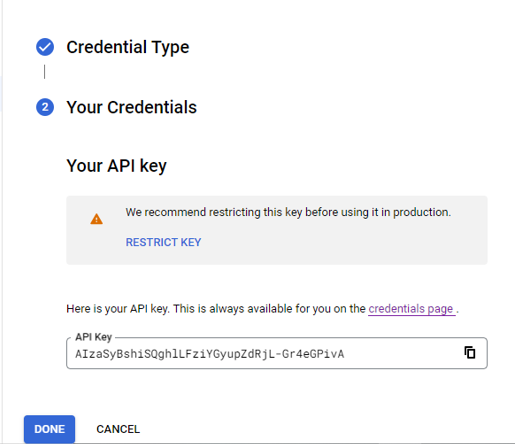
1. Lakukan Create credentials, Ikuti prosesnya, serta isilah apa yang diminta sesuai dengan data kamu.



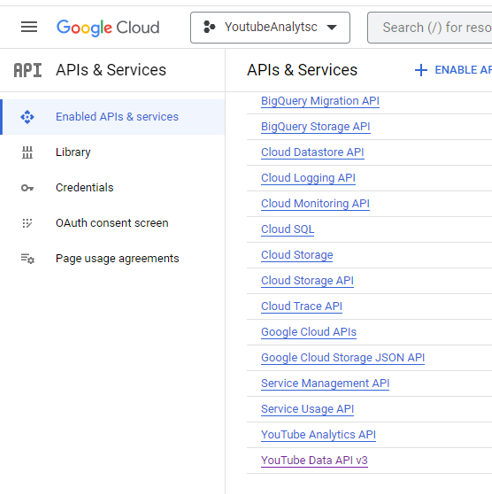
Pilih Next,



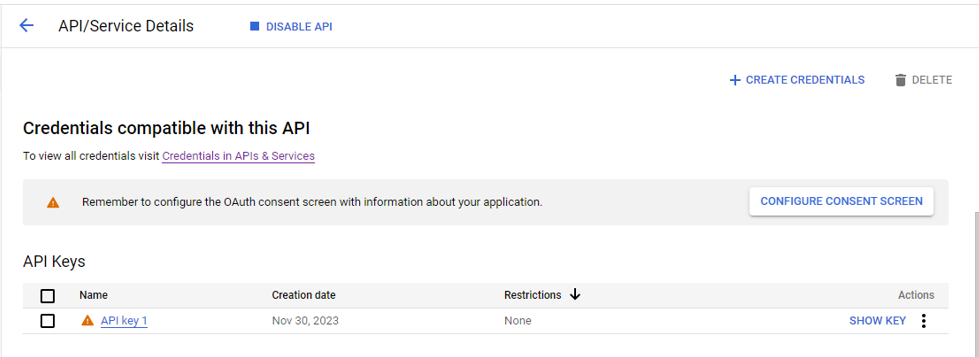
Api Key diatas yang nantinya akan anda gunakan dalam proses crawling data. jangan lupa pilih button Done.



1. Jika kredensial yang kamu buat sudah benar maka akan muncul pada APIs & SERVICES seperti dibawah ini.



Double klik pada kredensial yang kamu buat maka akan diarahkan ke halaman API Keys. lalu bisa pilih show Key untuk melihat API Key kamu. API ini yang nanti akan digunakan dalam proses crawling data komentar youtube



1. **Proses Crawling Data**
2. Mengimport Library pandas dan Koneksi ke layanan API google, menggunakan kode seperti berikut:

import pandas as pd

from googleapiclient.discovery import build

1. Kode berikut ini merupakan kode Python yang gunanya untuk mengambil komentar serta balasan komentar dari video Youtube yang dipilih menggunakan Youtube API v3 tadi.

def video\_comments(video\_id):

# empty list for storing reply

replies = []

# creating youtube resource object

youtube = build('youtube', 'v3', developerKey=api\_key)

# retrieve youtube video results

video\_response = youtube.commentThreads().list(part='snippet,replies', videoId=video\_id).execute()

# iterate video response

while video\_response:

# extracting required info

# from each result object

for item in video\_response['items']:

# Extracting comments ()

published = item['snippet']['topLevelComment']['snippet']['publishedAt']

user = item['snippet']['topLevelComment']['snippet']['authorDisplayName']

# Extracting comments

comment = item['snippet']['topLevelComment']['snippet']['textDisplay']

likeCount = item['snippet']['topLevelComment']['snippet']['likeCount']

replies.append([published, user, comment, likeCount])

# counting number of reply of comment

replycount = item['snippet']['totalReplyCount']

# if reply is there

if replycount>0:

# iterate through all reply

for reply in item['replies']['comments']:

# Extract reply

published = reply['snippet']['publishedAt']

user = reply['snippet']['authorDisplayName']

repl = reply['snippet']['textDisplay']

likeCount = reply['snippet']['likeCount']

# Store reply is list

#replies.append(reply)

replies.append([published, user, repl, likeCount])

# print comment with list of reply

#print(comment, replies, end = '\n\n')

# empty reply list

#replies = []

# Again repeat

if 'nextPageToken' in video\_response:

video\_response = youtube.commentThreads().list(

part = 'snippet,replies',

pageToken = video\_response['nextPageToken'],

videoId = video\_id

).execute()

else:

break

#endwhile

return replies

1. Berikut ini merupakan proses memfungsikan kode yang telah dideklarasikan pada kode sebelumnya. Lebih spesifiknya mengambil komentar youtube menggunakan API sesuai dengan video yang dipilih. Berikut kode yang di maksud:

# isikan dengan api key Anda

api\_key = '\*\*\*\*\*\*\*\*'

# Enter video id

# contoh url video =https://www.youtube.com/watch?v=2iZdW\_\_BR5w

video\_id = "2iZdW\_\_BR5w" #isikan dengan kode / ID video

# Call function

comments = video\_comments(video\_id)

comments

1. Selanjutnya adalah memfungsikan pandas, agar komen youtube yang sudah kita ambil tadi bisa lebih rapi dan mudah dimengerti, seperti berikut:

df = pd.DataFrame(comments, columns=['publishedAt', 'authorDisplayName', 'textDisplay', 'likeCount'])

df

1. Yang terakhir adalah mengekspor data frame ke dalam bentuk csv. Serta menghindari menyimpan index data frame ke dalam file csv.

df.to\_csv('youtube-comments.csv', index=False)

1. **CLEANING DATA TWITTER DAN YOUTUBE**