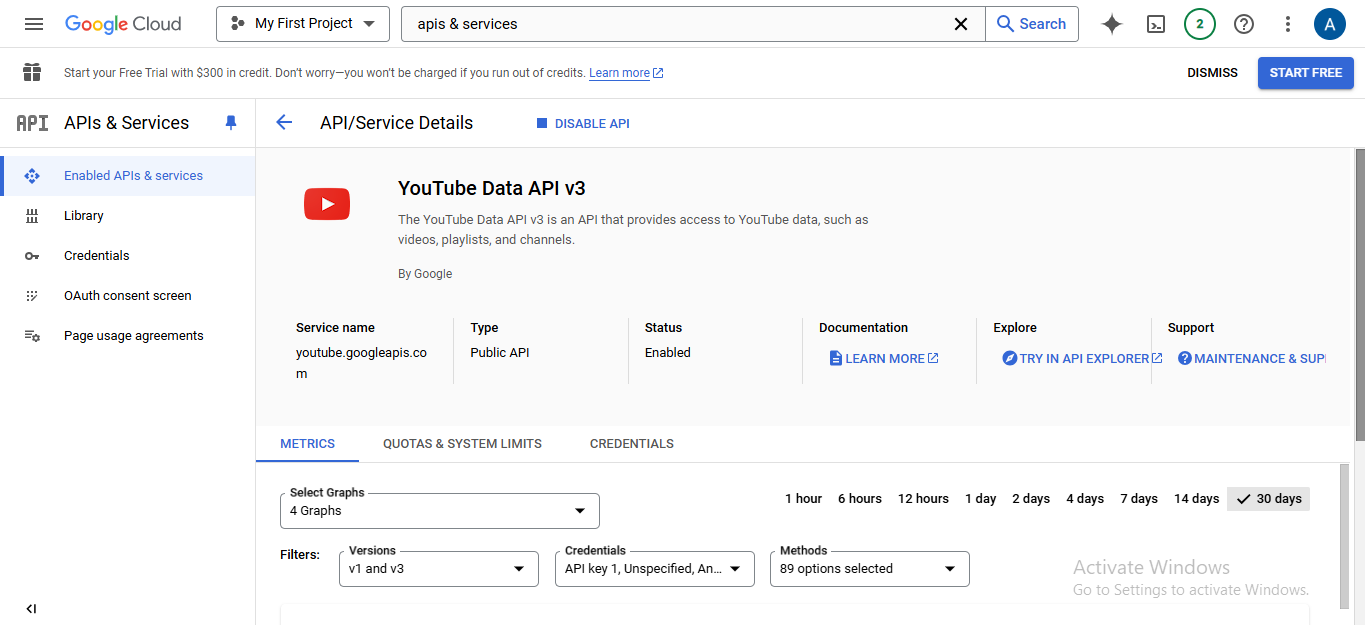
**3. Social Network API**

**Youtube Vedios Dataset**

Src: From Youtube using YouTube Data API v3 from Gogole cloud



**Description:**

The YouTube Video Dataset contains information about various videos collected from YouTube channels. It includes features such as video title, description, publish date, publish time, likes, views, category, and channel ID. This data can be used for analysis and modeling related to video performance, audience engagement, and content trends.

**Data Collection and Understanding:**

1. **Attributes / Feature Description**

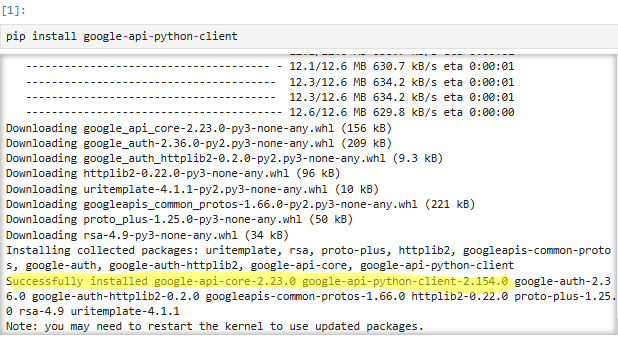
|  |  |  |  |
| --- | --- | --- | --- |
| Column Name | Description | Data Type | Attribute Type |
| index | The index number for each entry. | int64 | Discrete |
| title | The title of the video. | object | Nominal |
| description | The description of the video. | object | Nominal |
| publish\_date | The date when the video was published. | object | Nominal |
| publish\_time | The time when the video was published. | object | Nominal |
| likes | The number of likes the video has received. | int64 | Numeric (Ratio-Scaled) |
| views | The number of views the video has received. | int64 | Numeric (Ratio-Scaled) |
| category | The category the video belongs to [news , sport .. etc] | object | Nominal |
| channel\_id | The ID of the channel that published the video. | object | Nominal |

1. **Dataset Collection :** Collected YouTube video data using the YouTube Data API and saved it to `youtube\_videos.csv`

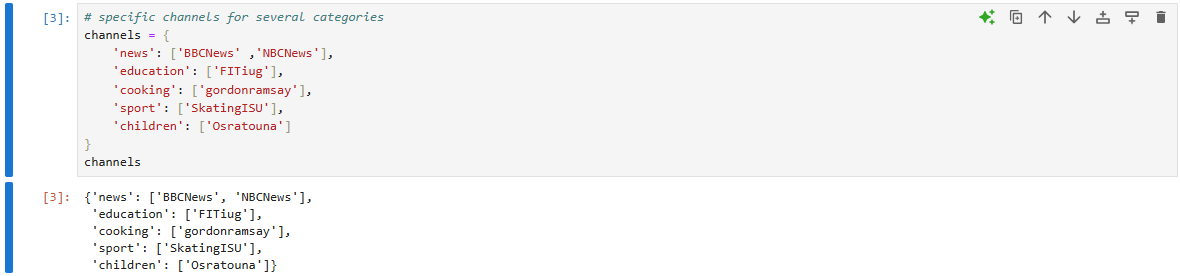
In this Part I collect data using Social Network API and “Python”:

1. Import Libraries used in interact with youtube API

&

 download nedded librarys

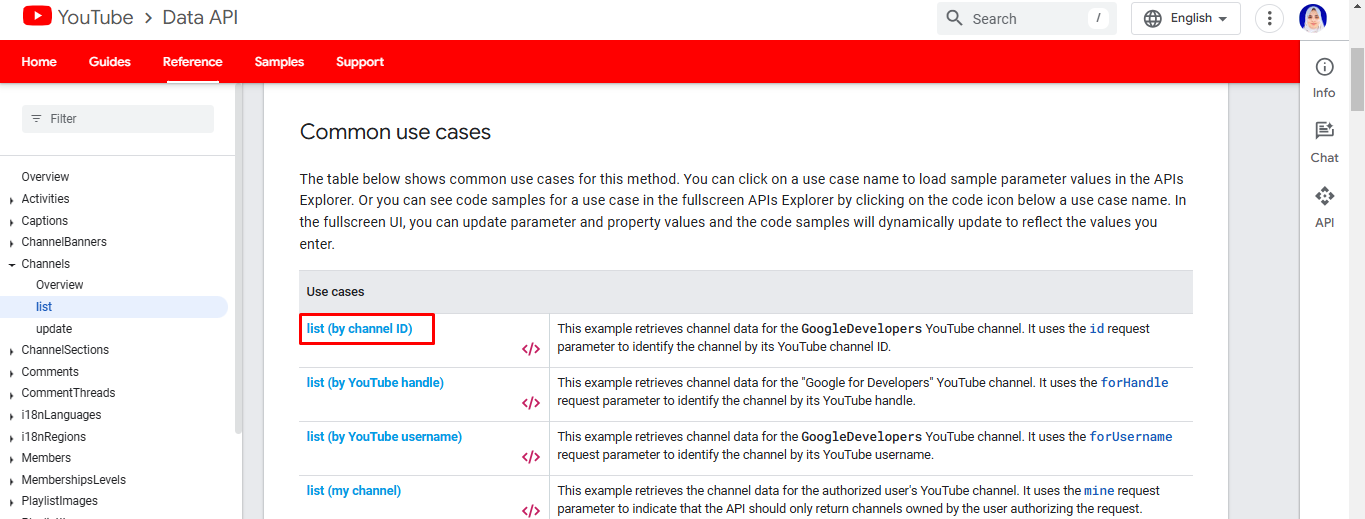
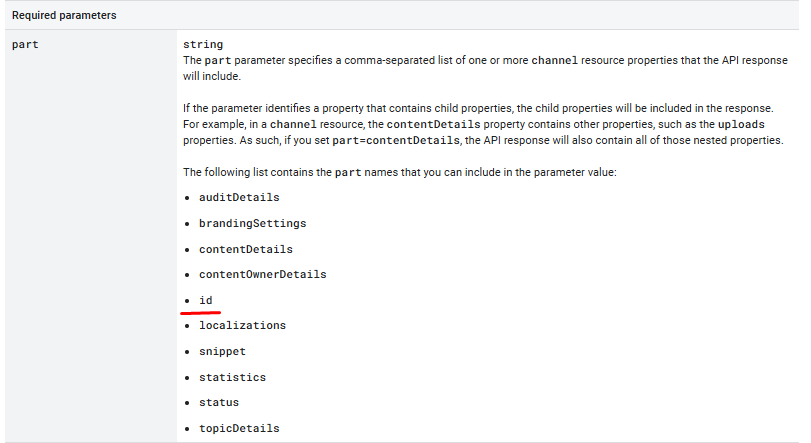
1. Setting YouTube Data API Client: set up and authenticate a YouTube Data API client using an API key generated from the Google Cloud Console. **build** from the googleapiclient.discovery module used to create an instance of the YouTube Data API client.
2. Define channels to read from: specify channels to read vedios data from it in a dictionary {key is categoy : value is channel name}.

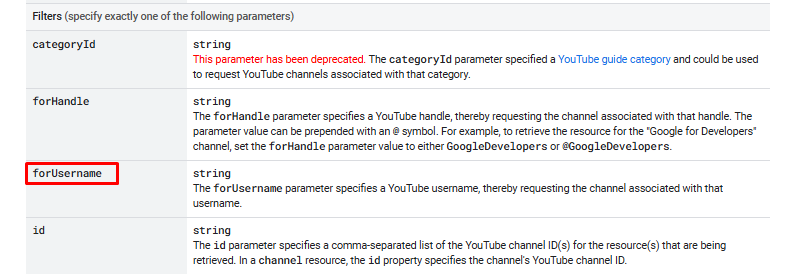


1. Get Channel id: use channel name to get id .. send request to list channel with specific name

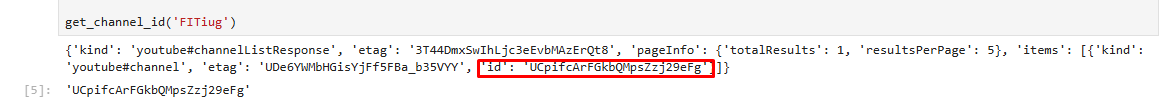


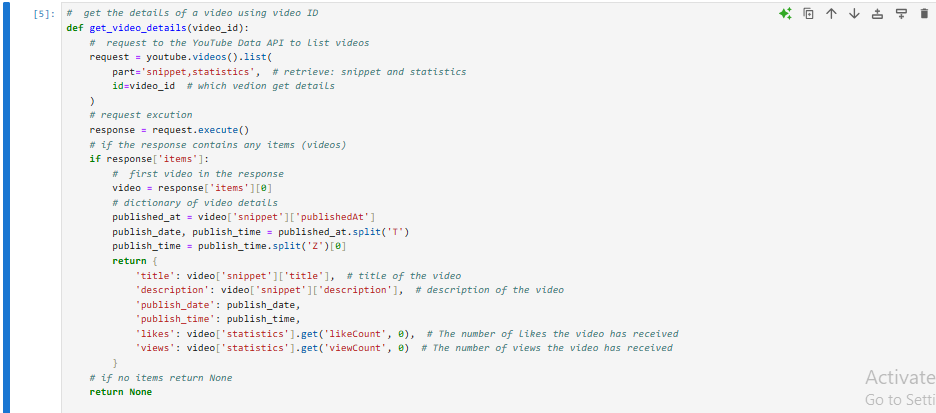
All apis function I used available in **Youtube Data API Documentaition**

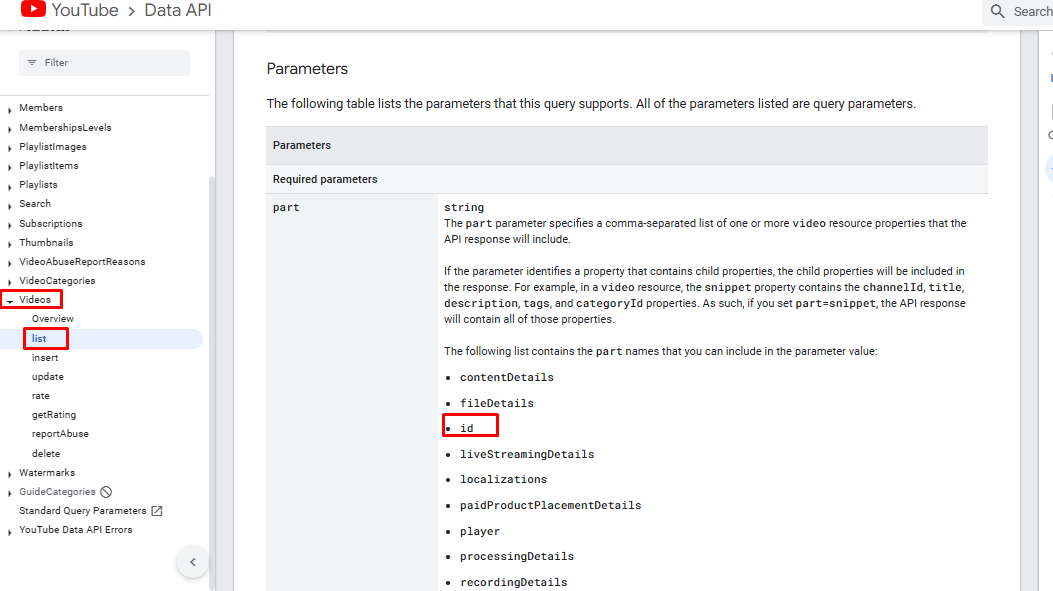




Response return

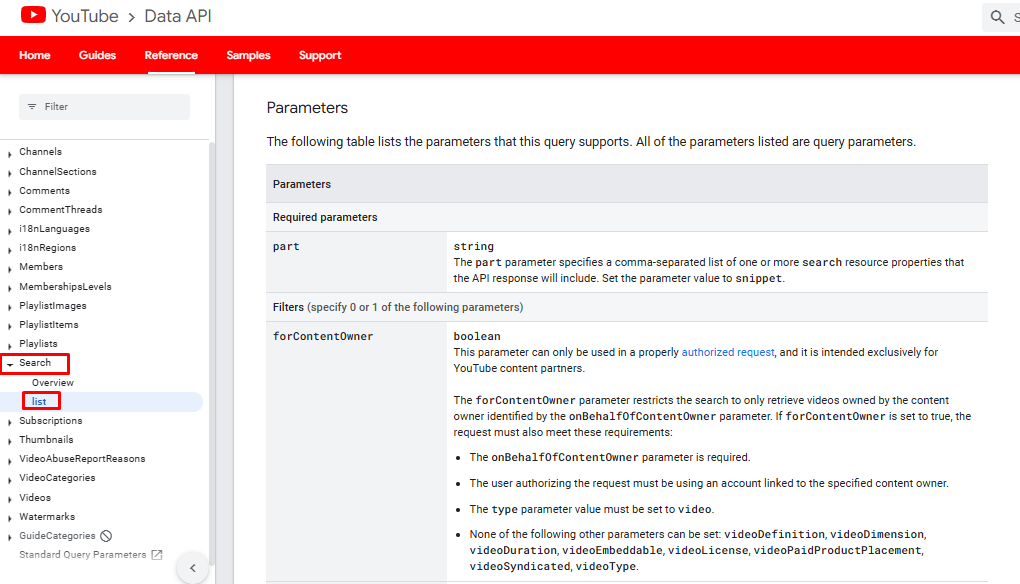


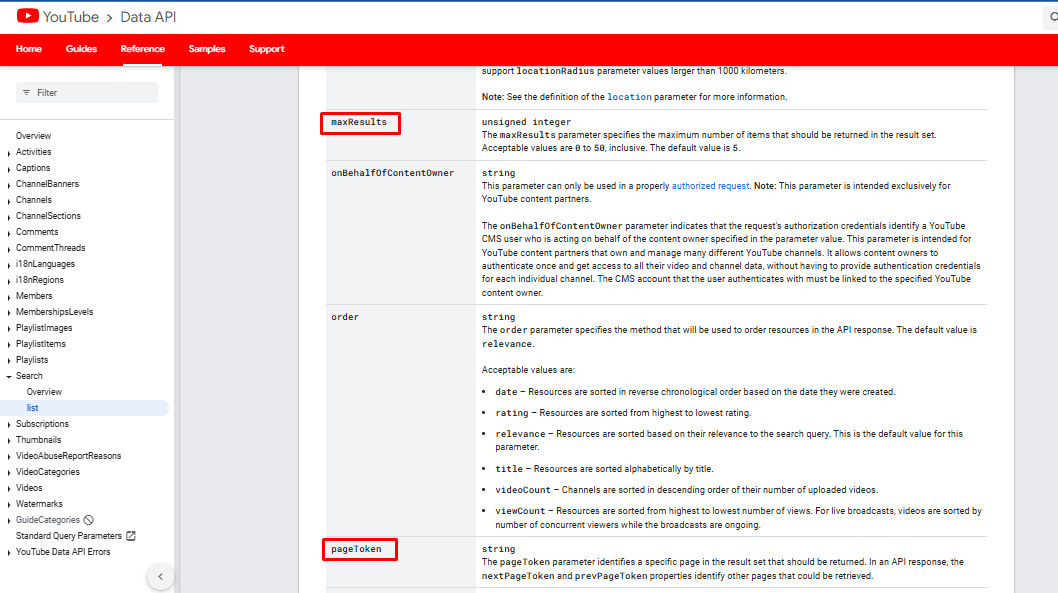
1. Get Vedio Details: for each vedio collected from chnnel list data about it ‘statistics for likes , views’ & ‘snippet for title, description , publiseted details'



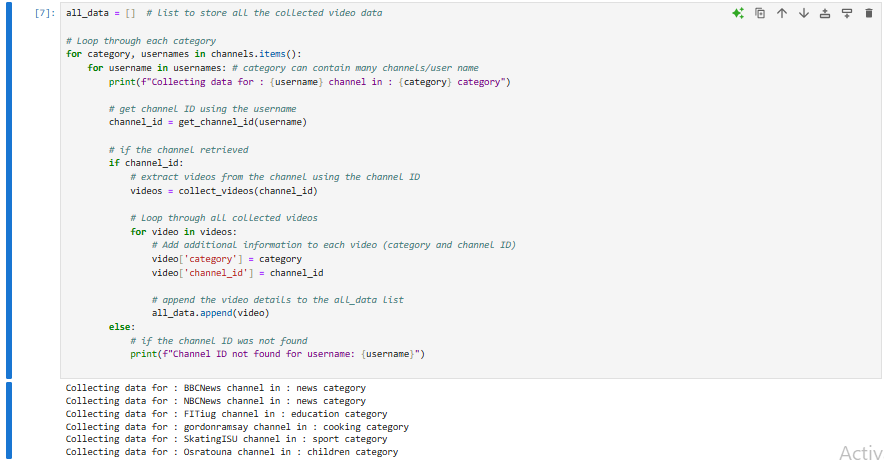
1. Collect Vedio: collect 200 vedio data from each channel using channel id retened using function in 4 step.



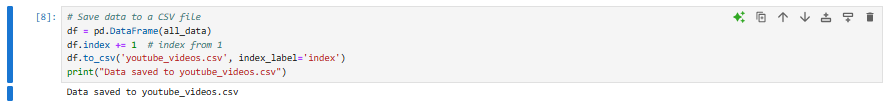


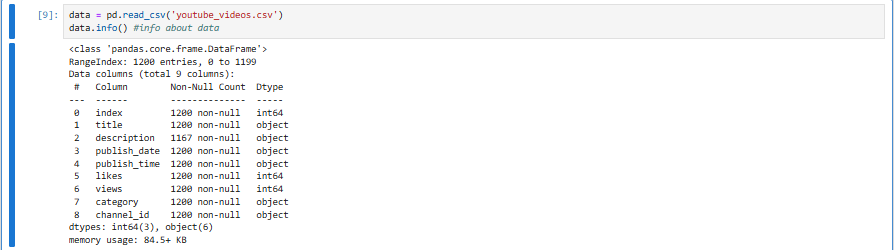


1. Collect data: collect vedios data from diferent channels using all privios functions



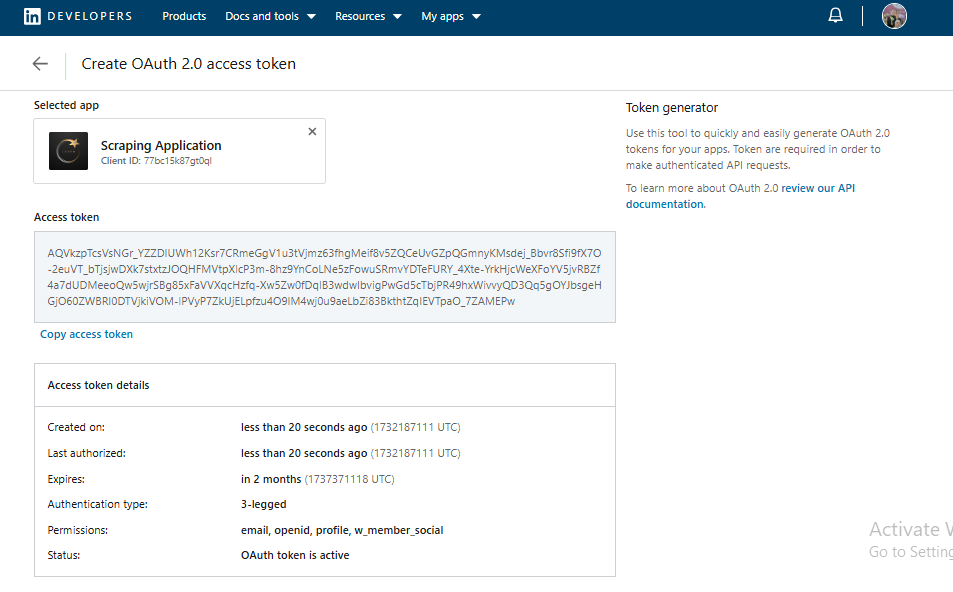
1. Finally store data in csv file using pandas DataFrame and reading info()

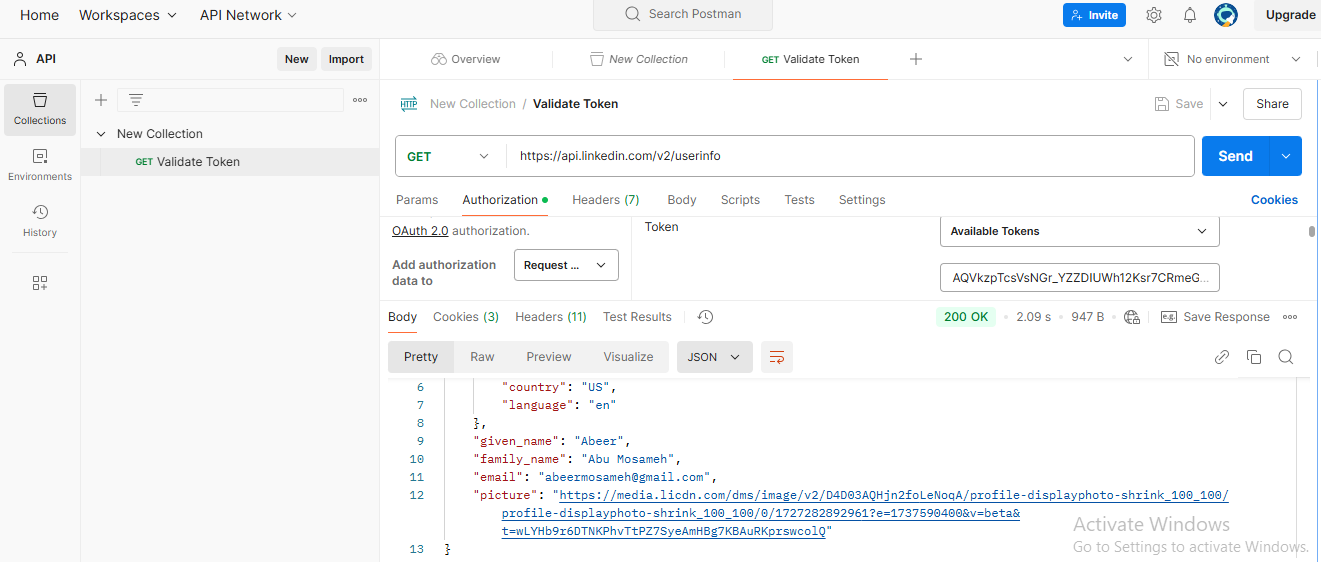




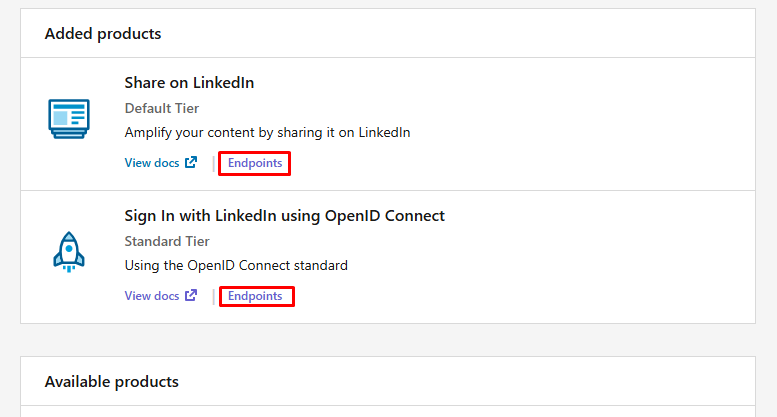
**limitiation of Social API**

I statrt to scap Linked in Jobs “ title , desc , location and so on)



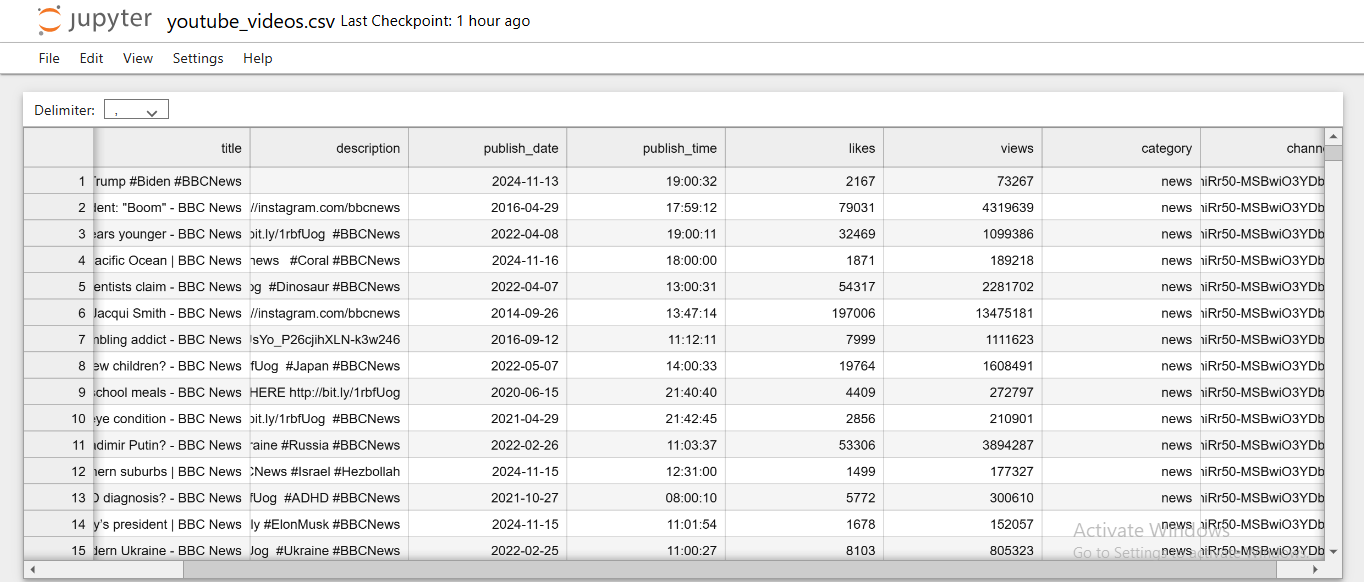


But avalible free end point can’t help in that



I need more products to access or company/busniss formal document to request access to this API’s

And same thing in X “twitter” I can’t read general data just for specific user with unlegal ways becous in 2024 be not free in legal way So finally I choos youtube API to complete assignment



**Data Preparation:**

1. **Data Exploration**

**Explore the dataset to understand its structure and content.**

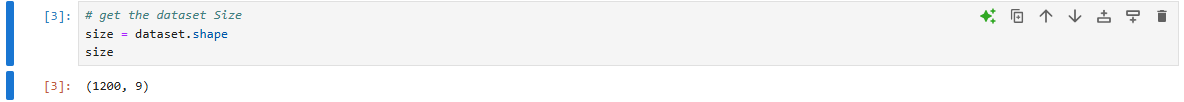
1. **Import Necessary Libraries**



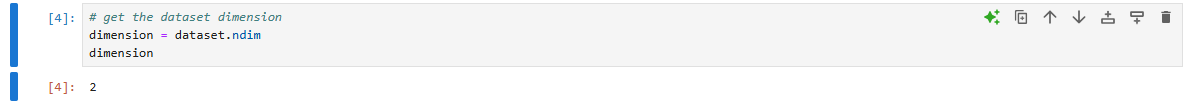
1. **Load dataset & Read it**



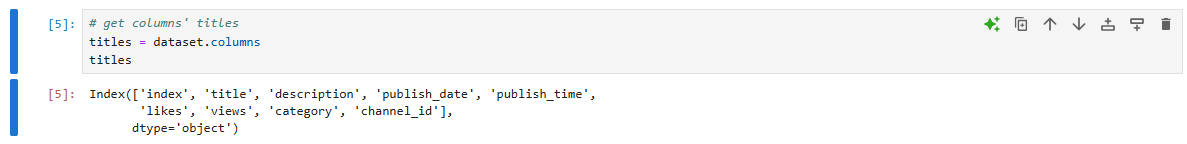
1. **Dataset Size**



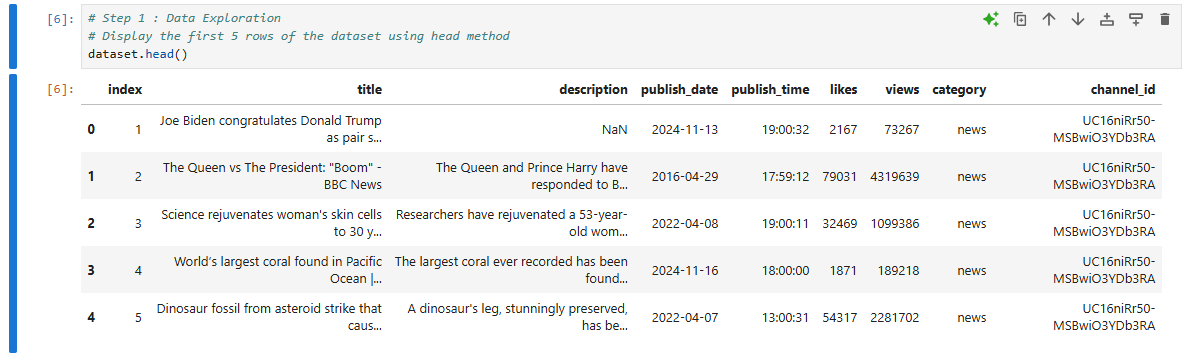
1. **Dataset dimensions**



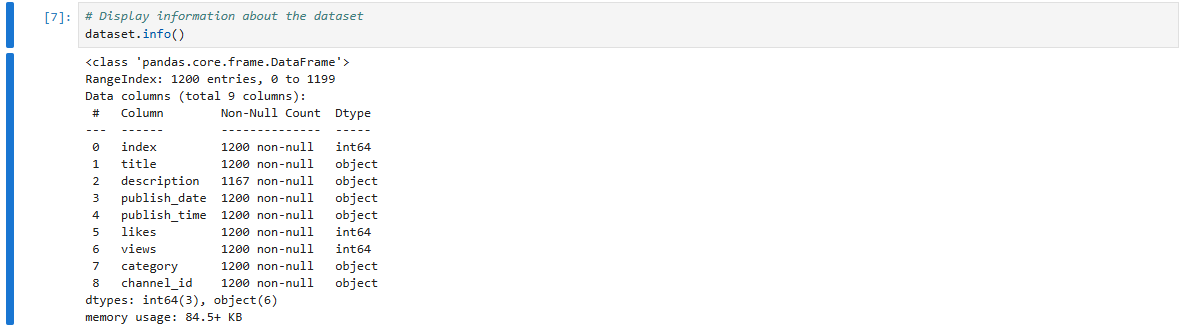
1. **Get dataset columns**



1. **Head of dataset ‘”first rows”**



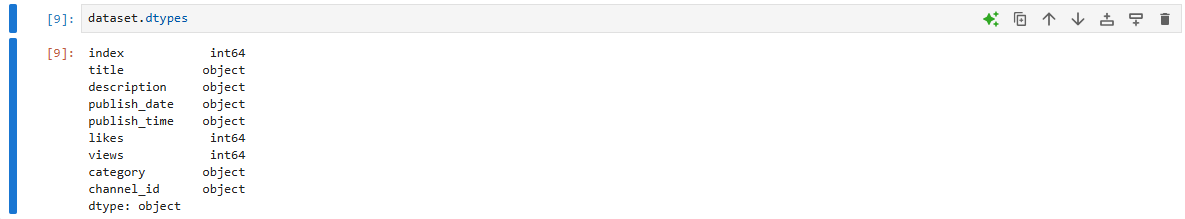
1. **Explore the Structure using info()**



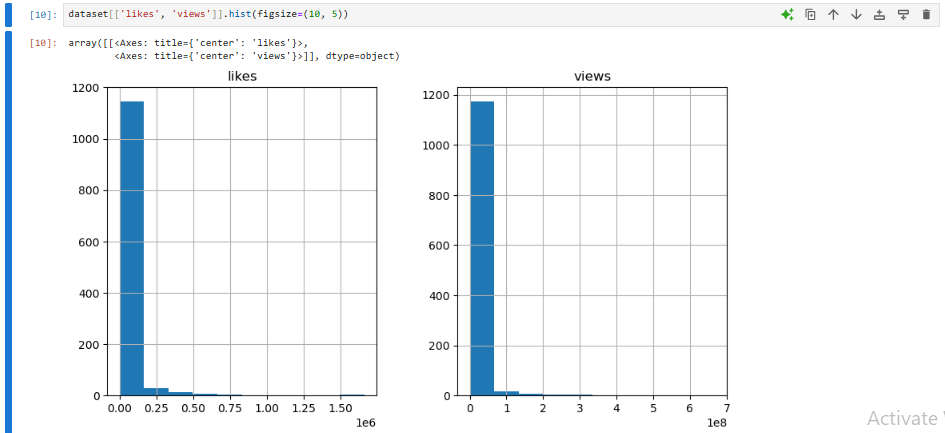
1. **Display descriptive statistic using describe()**

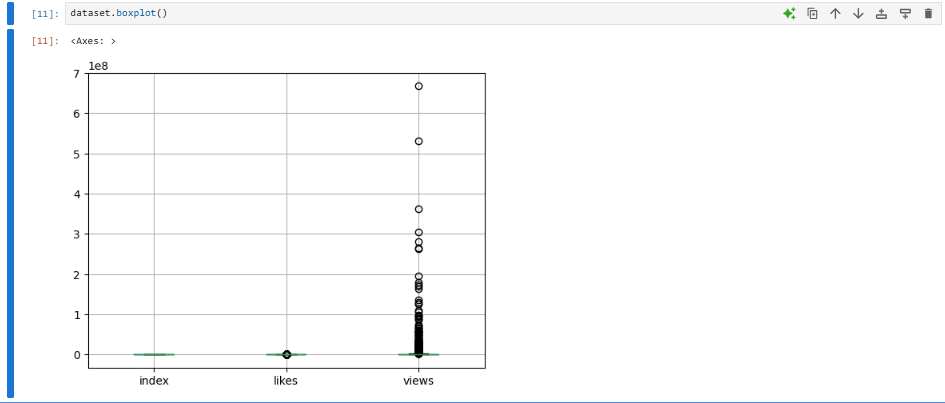


1. **Data Types**

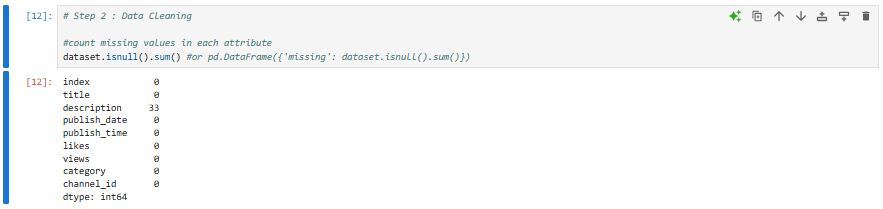


Data Exploration befor cleaning & Transformation Steps





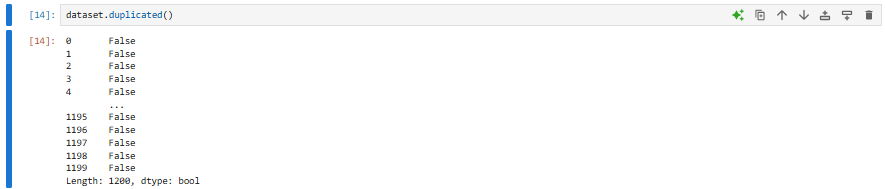
1. **Data Cleaning**
2. **Missing/Incomplete Values**

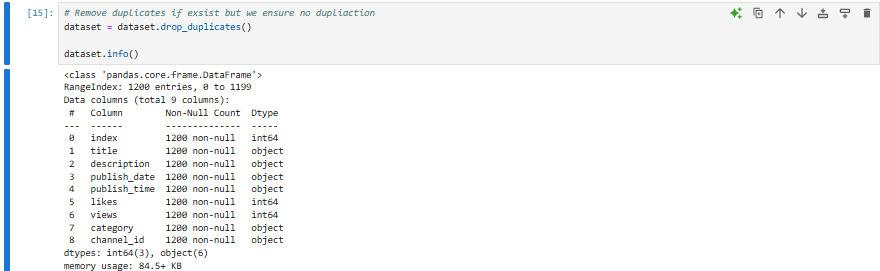


Handel Missing Values



1. **Duplicated Values**

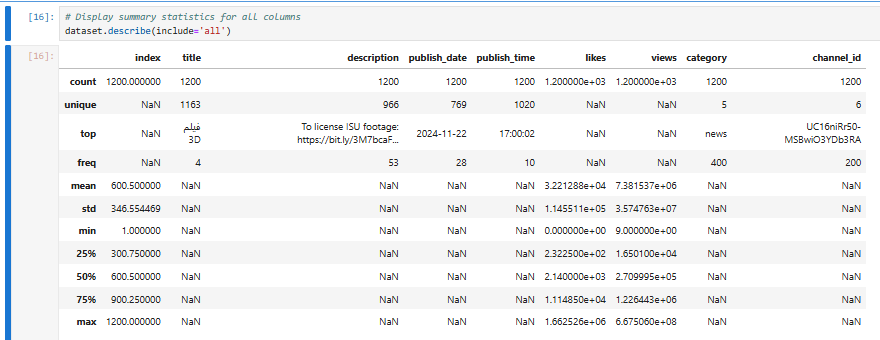




No Duplicated Values

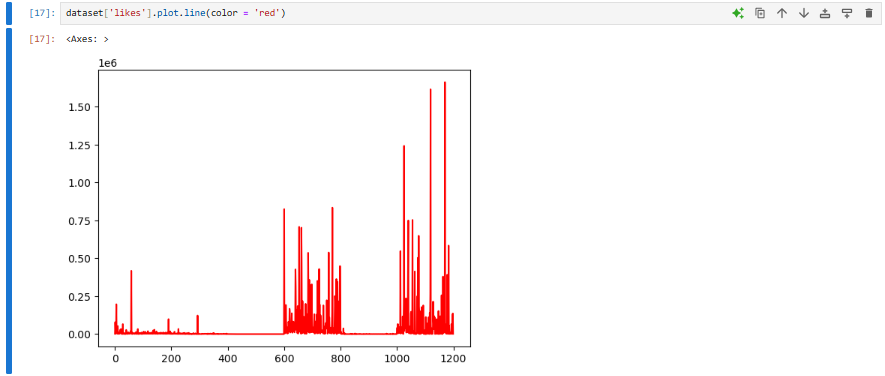
1. **Noisy Values**

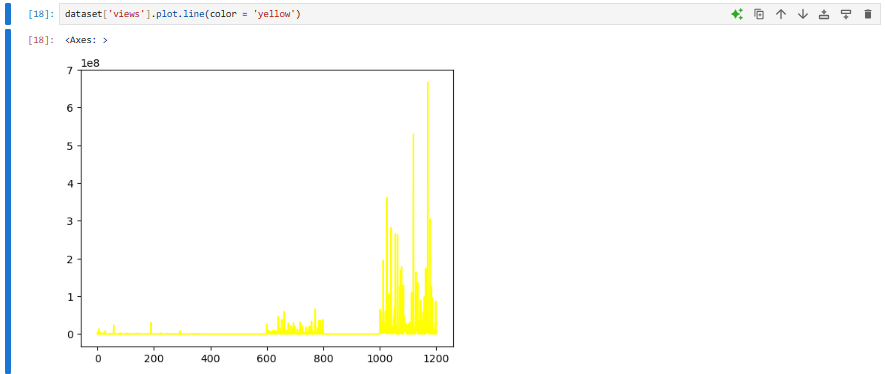
In Image below we notice no error, noisy or outlier “can be found later”

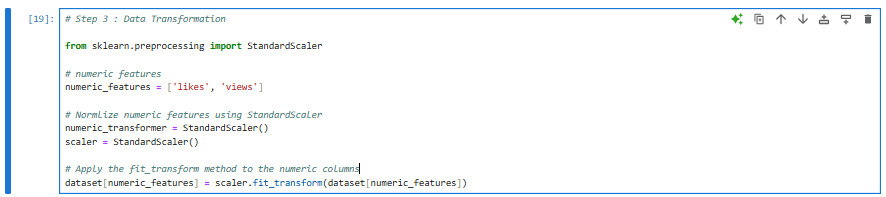
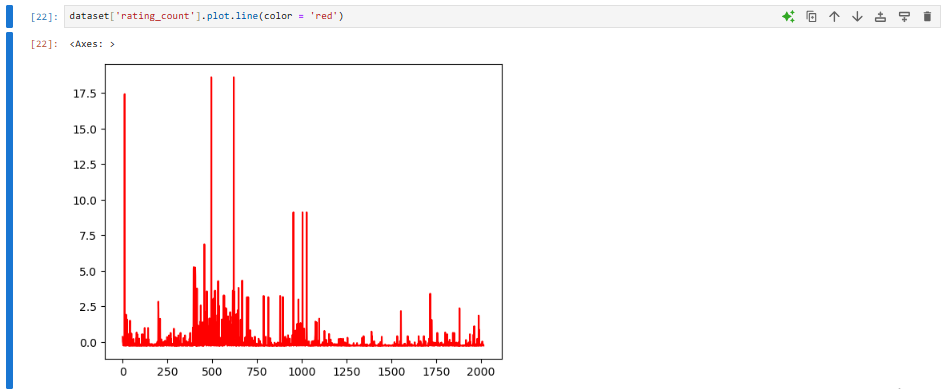


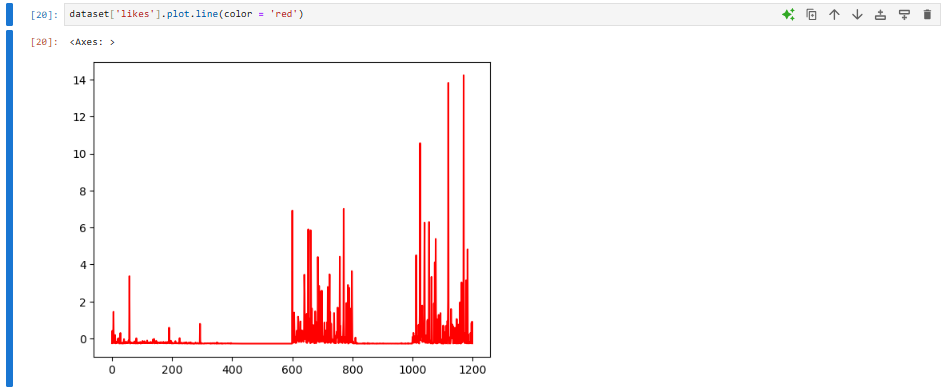
1. **Data Transformation**
2. **Z-score Normalization (StandardScaler)**

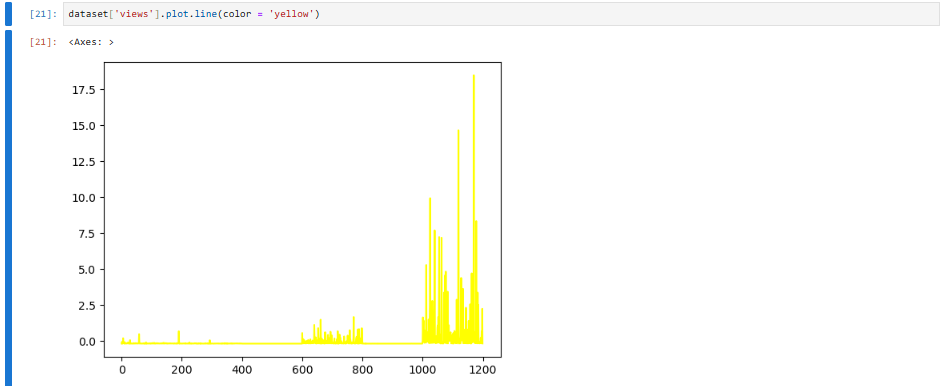
we need do transformation “normalization” in likes , views column using StandardScaler() : depend on mean and standard deviation of column







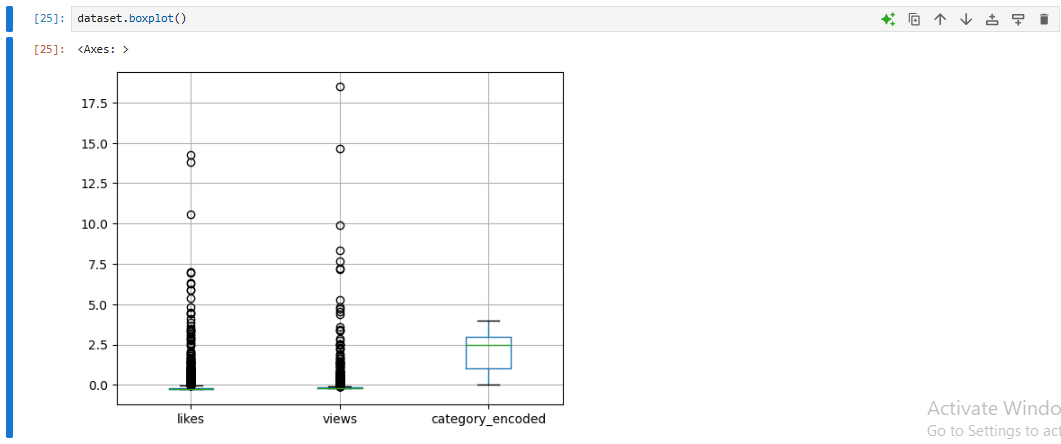




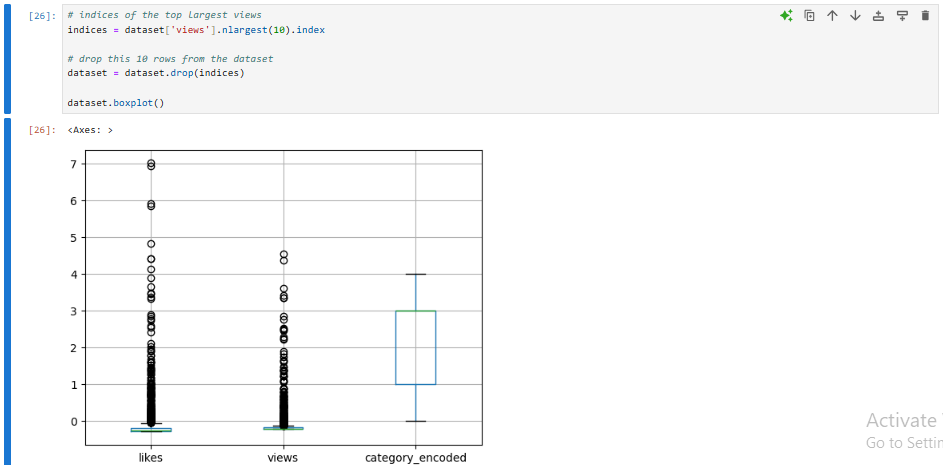
1. **Categorical Encoding - Ordinal/Label Encoder**

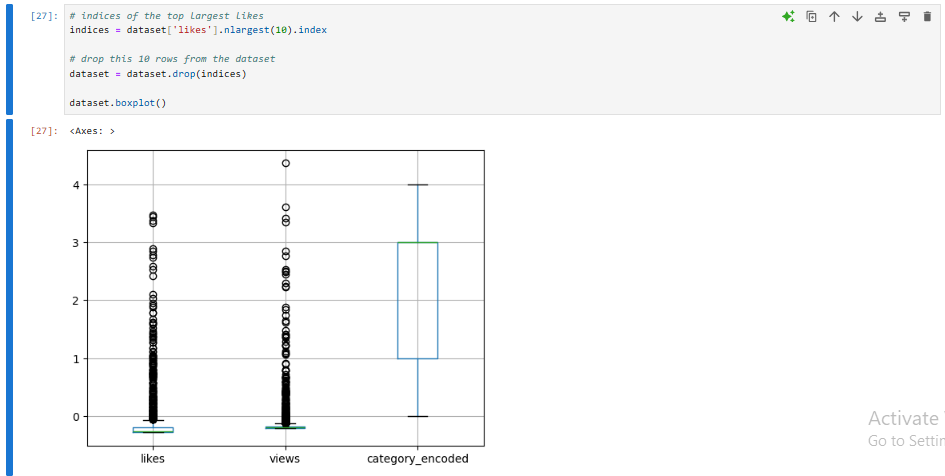






We can renove outlier on views and likes:

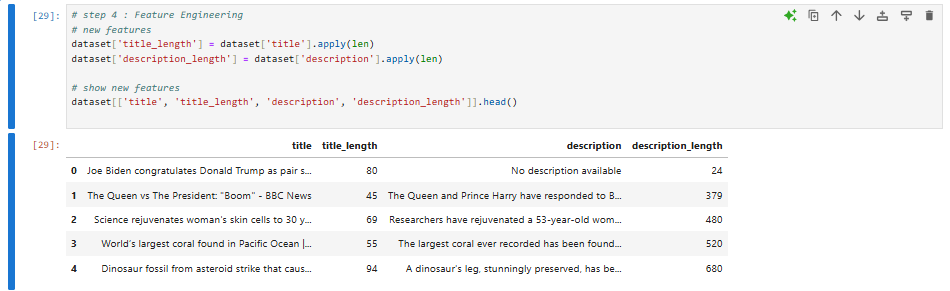




1. **Feature Engineering**

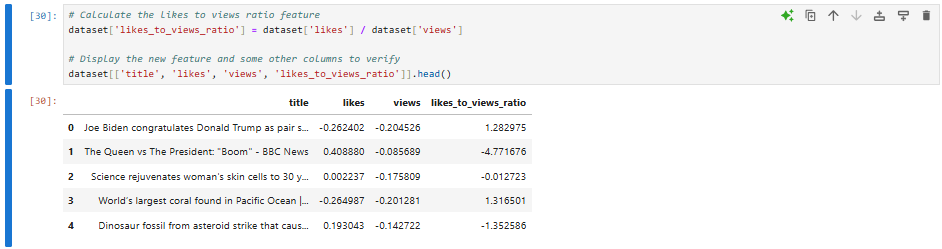
#### **Create a New Feature: likes\_to\_views\_ratio “ratio of likes to views for each video”**

Indicate that higher ratio suggests that the video is well-received by its viewers and used in Comparative Analysis



1. ***Create a New Feature: description\_length and title\_length***

Useful in SEO Optimization “how long of it" .. insights into how these lengths might impact search engine rankings and viewer engagement”



Data Exploration after all **Data Preparation**



