Code File

Project Title: YouTube Video Performance Analytics

Team Members: –

Shubham Maurya (Roll No.: 202410101150072),

Ansh Singh (Roll No.: 202410101150093)

Department: – Computer Science and Engineering

Course: – B. Tech, Branch: - CS(DS+AI)

Subject: – Data Analytics and Reporting

Academic Year: –2025–2026

Here we use video of channel of **getsetflySCIENCE** who creator name **Mr. Gaurav Thakur** from is where first we take the channel id from his channel and them by that id we create an API id, which is: -

API ID: - *AIzaSyDFJ_THPq-MbhyLiQ3nDCUl-Otvy79RAZ8*

Channel Id: - *UC00ifCvU8YOOzbL3RdiSTDw*

[2] ✓ 2s

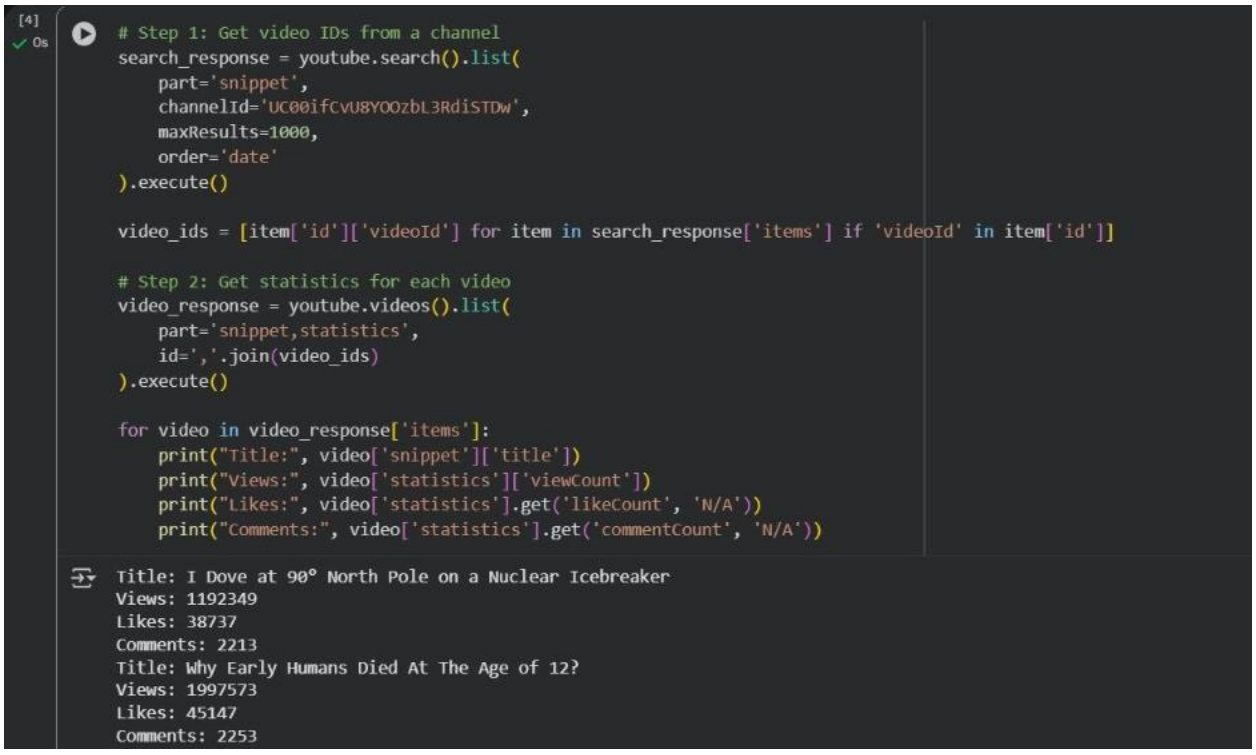
```
from googleapiclient.discovery import build

api_key = 'AIzaSyDFJ_THPq-MbhyLiQ3nDCUl-Otvy79RAZ8'
youtube = build('youtube', 'v3', developerKey=api_key)
```

[3] ✓ 0s

```
request = youtube.channels().list(
    part='snippet,statistics',
   id = 'UC00ifCvU8YOOzbL3RdiSTDw'

  )
response = request.execute()

for item in response['items']:
    print("Channel Title:", item['snippet']['title'])
    print("Subscribers:", item['statistics']['subscriberCount'])
    print("Views:", item['statistics']['viewCount'])
    print("Videos:", item['statistics']['videoCount'])
```

```
Channel Title: GetsetflySCIENCE by Gaurav Thakur
Subscribers: 9560000
Views: 1364189536
Videos: 502
```

- Find all the detail (like, comment, and views also) about the videos that are uploaded on the channel by the help of channel id on the google colab.

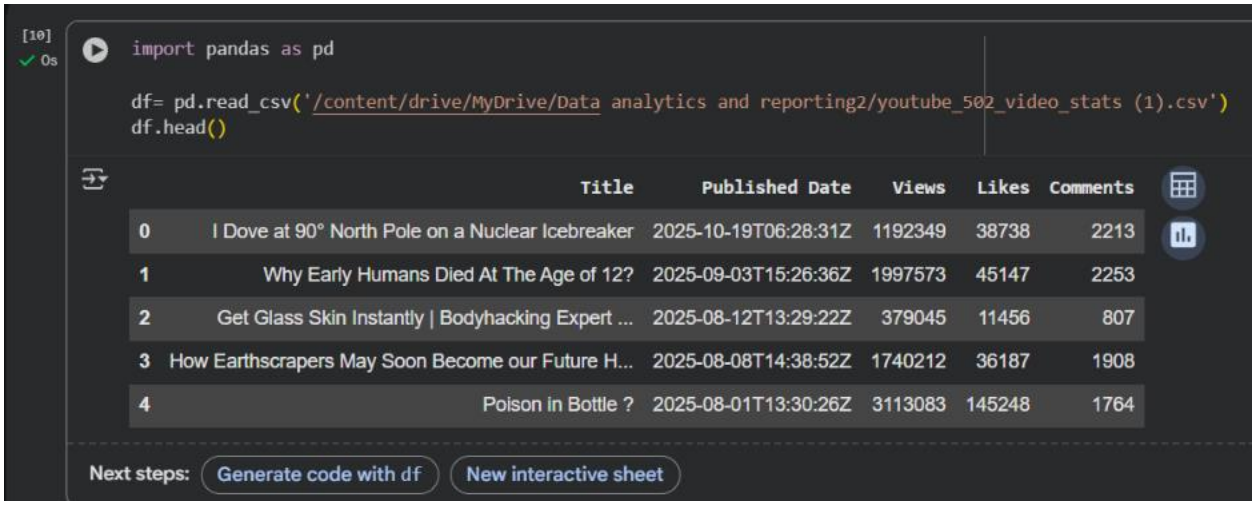


```
[4]
✓ 0s
# Step 1: Get video IDs from a channel
search_response = youtube.search().list(
    part='snippet',
    channelId='UC00ifCvU8YOOzbL3RdiSTDw',
    maxResults=1000,
    order='date'
).execute()

video_ids = [item['id']['videoId'] for item in search_response['items'] if 'videoId' in item['id']]

# Step 2: Get statistics for each video
video_response = youtube.videos().list(
    part='snippet,statistics',
    id=','.join(video_ids)
).execute()

for video in video_response['items']:
    print("Title:", video['snippet']['title'])
    print("Views:", video['statistics']['viewCount'])
    print("Likes:", video['statistics'].get('likeCount', 'N/A'))
    print("Comments:", video['statistics'].get('commentCount', 'N/A'))

Title: I Dove at 90° North Pole on a Nuclear Icebreaker
Views: 1192349
Likes: 38737
Comments: 2213
Title: Why Early Humans Died At The Age of 12?
Views: 1997573
Likes: 45147
Comments: 2253
```

- Store data in CSV for analysis

```
[6]
✓ 0s
from google.colab import files
files.download('youtube_502_video_stats.csv')
```

- Perform descriptive analysis using Pandas

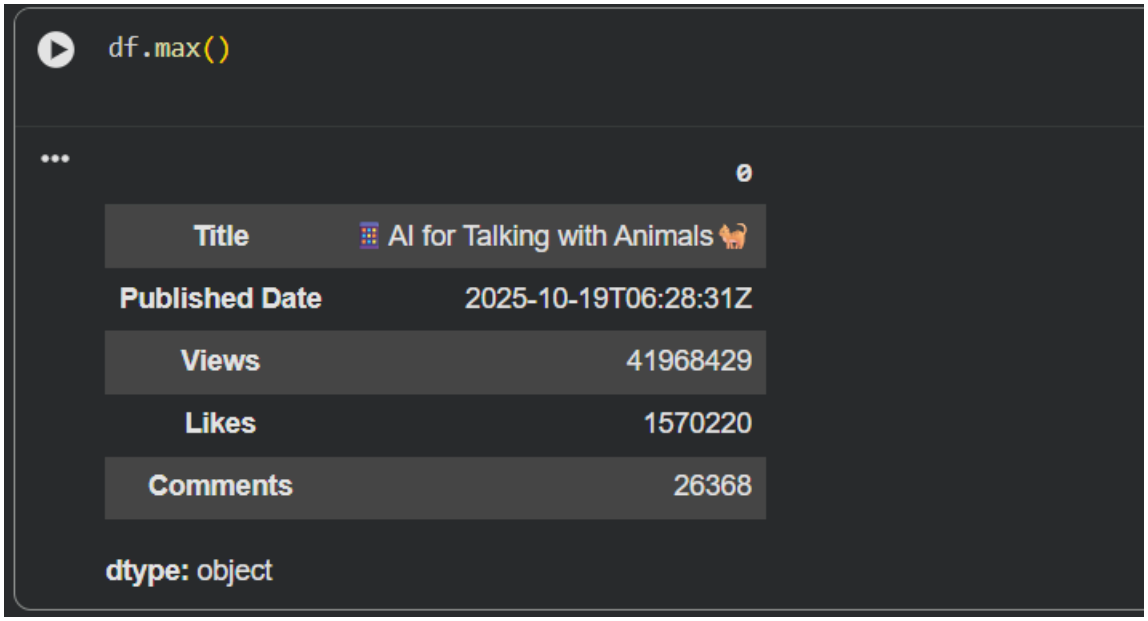


```
[10]
✓ 0s
import pandas as pd

df= pd.read_csv('/content/drive/MyDrive/Data analytics and reporting2/youtube_502_video_stats (1).csv')
df.head()
```

	Title	Published Date	Views	Likes	Comments
0	I Dove at 90° North Pole on a Nuclear Icebreaker	2025-10-19T06:28:31Z	1192349	38738	2213
1	Why Early Humans Died At The Age of 12?	2025-09-03T15:26:36Z	1997573	45147	2253
2	Get Glass Skin Instantly | Bodyhacking Expert ...	2025-08-12T13:29:22Z	379045	11456	807
3	How Earthscrapers May Soon Become our Future H...	2025-08-08T14:38:52Z	1740212	36187	1908
4	Poison in Bottle ?	2025-08-01T13:30:26Z	3113083	145248	1764

Next steps: Generate code with df | New interactive sheet

- Finding the video that have maximum views by the help of max () function in the google colab.



```
df.max()
```

	0
Title	AI for Talking with Animals
Published Date	2025-10-19T06:28:31Z
Views	41968429
Likes	1570220
Comments	26368

dtype: object

- Finding the video that have minimum views by the help of min () function in the google colab.

```
df.min()
```

	0
Title	2 SUPER MASSIVE BLACKHOLES are 99% Close. What...
Published Date	2017-05-31T08:46:09Z
Views	143298
Likes	4377
Comments	135

dtype: object

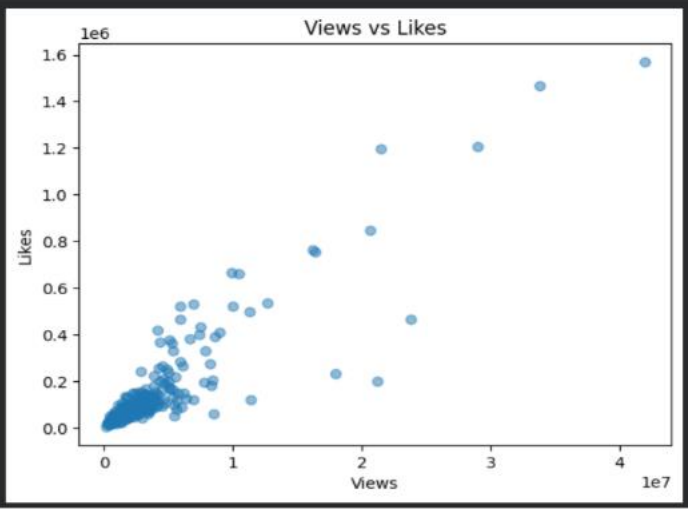
- Distribution Analysis through we can find out the total number of videos, views, average likes and maximum comments.

```
print("Total Videos:", len(df))
print("Total Views:", df['Views'].astype(int).sum())
print("Average Likes:", df['Likes'].astype(int).mean())
print("Max Comments:", df['Comments'].astype(int).max())
```

```
Total Videos: 358
Total Views: 1079969884
Average Likes: 116795.3407821229
Max Comments: 26368
```

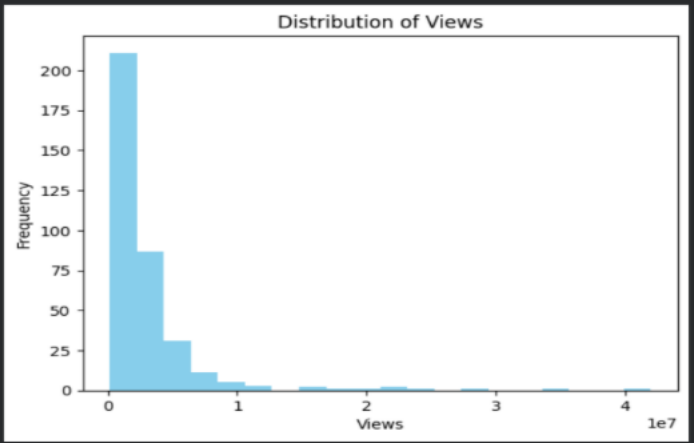
- Graphical representation of relationships between likes and views by the help of a scatter graph.

```
[7]
✓ 0s
plt.scatter(df['Views'], df['Likes'], alpha=0.5)
plt.title('Views vs Likes')
plt.xlabel('Views')
plt.ylabel('Likes')
plt.show()
```

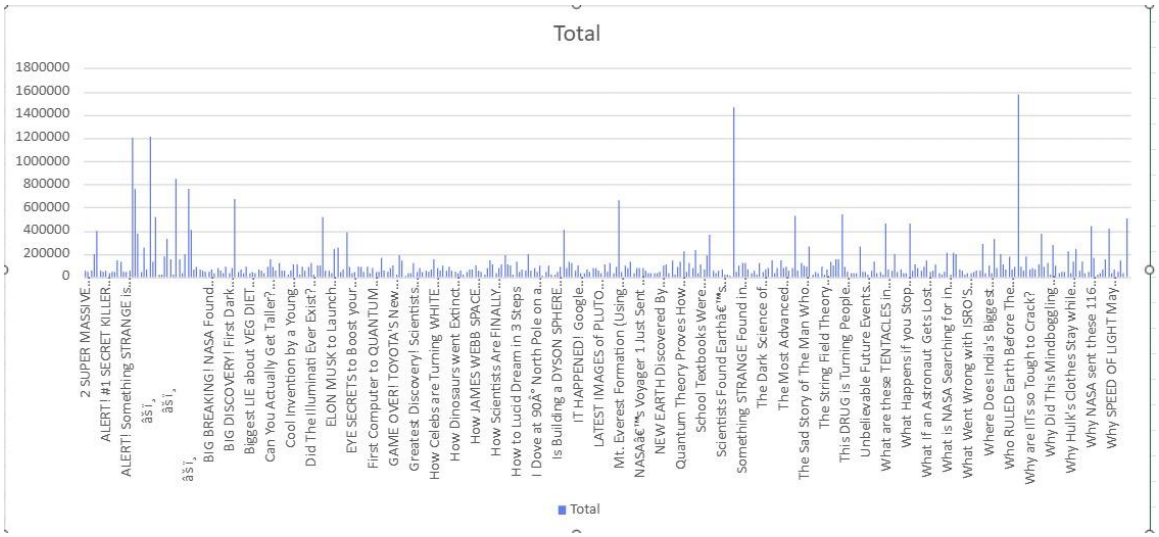


- Graphical representation of frequency of views by the help of a histogram.

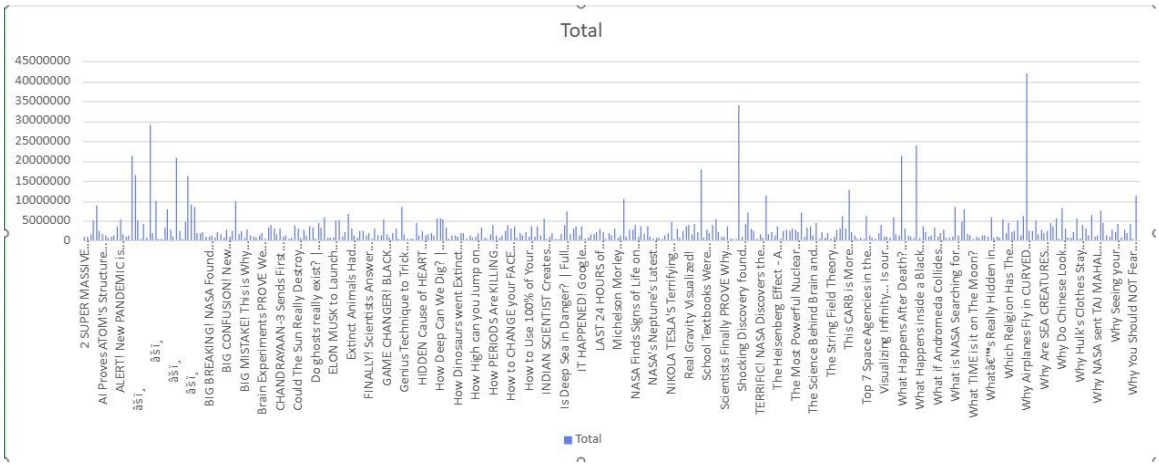
```
[6]
✓ 0s
import matplotlib.pyplot as plt

plt.hist(df['Views'], bins=20, color='skyblue')
plt.title('Distribution of Views')
plt.xlabel('Views')
plt.ylabel('Frequency')
plt.show()
```



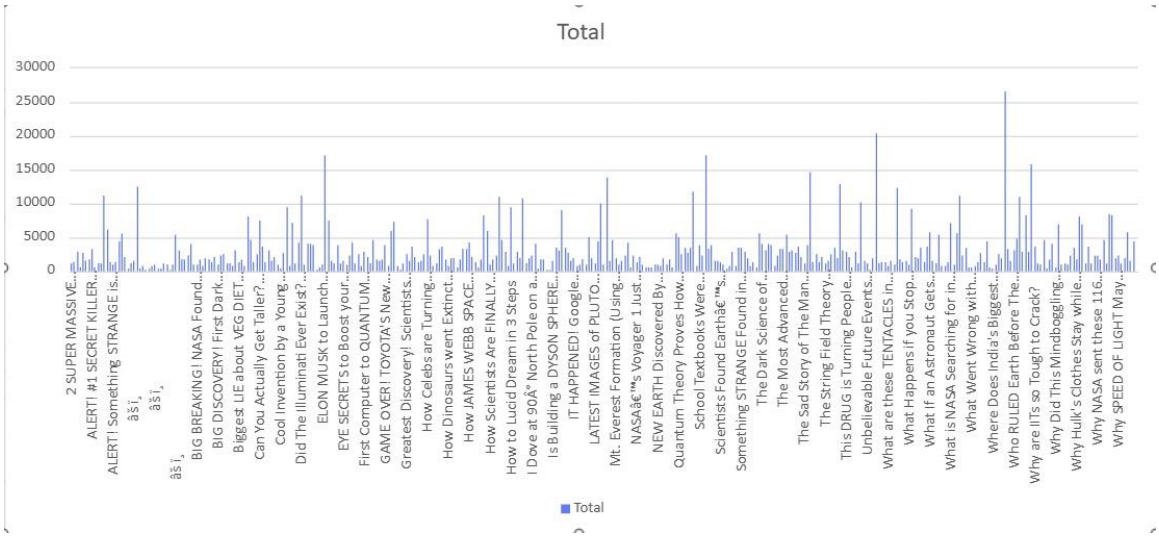
- Graphical representation of views of all videos by the help of a bar chart in the excel sheet.



- Graphical representation of likes of all videos by the help of a bar chart in the excel sheet.



- Graphical representation of likes of all videos by the help of a bar chart in the excel sheet.



- Generate summarized reports of all the videos of the YouTube channel.

```
summary = df.describe()
summary.to_csv('summary_stats.csv')
from google.colab import files
files.download('summary_stats.csv')
```

- Output: -

	A	B	C	D	E
1	Title	Views	Likes	Comments	Month
2	count	358	358	358	358
3	mean	3016676	116795.3	2921.19	6.527933
4	std	4373886	180081.1	3232.147	3.462978
5	min	143298	4377	135	1
6	25%	978983.3	39570.75	1068.75	4
7	50%	1788826	64618.5	1811.5	6
8	75%	3379717	111888.3	3510.5	10
9	max	41968429	1570220	26368	12