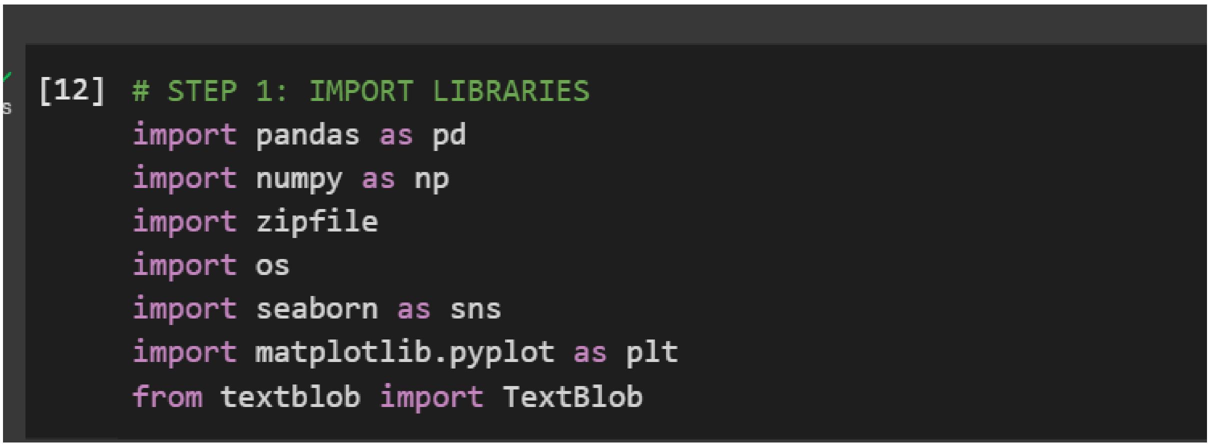
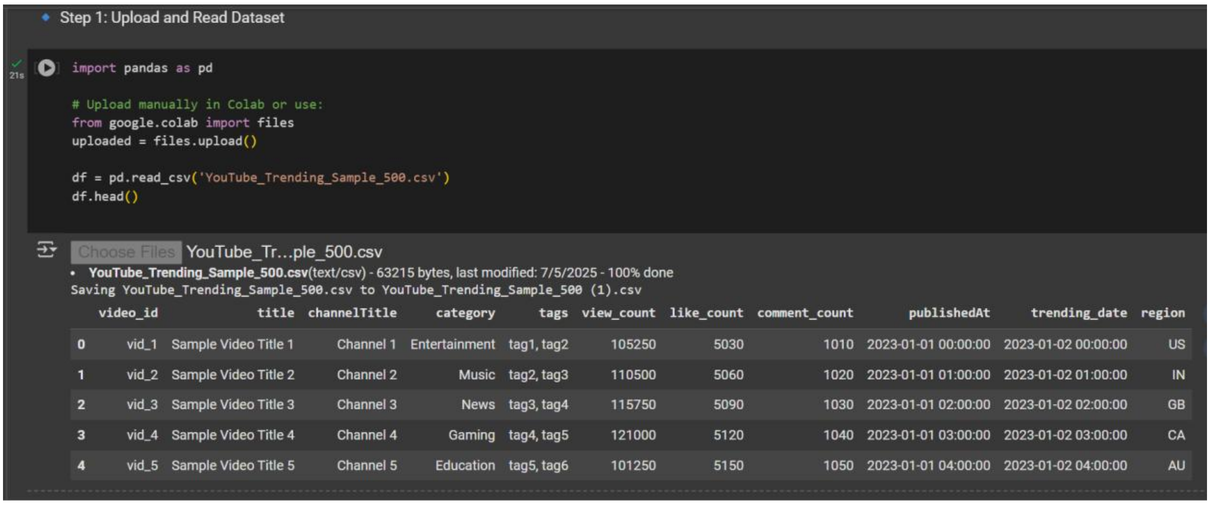
IMPORT LIBRARIES



```python
# STEP 1: IMPORT LIBRARIES
import pandas as pd
import numpy as np
import zipfile
import os
import seaborn as sns
import matplotlib.pyplot as plt
from textblob import TextBlob
```

UPLOAD AND READ DATASET



Step 1: Upload and Read Dataset

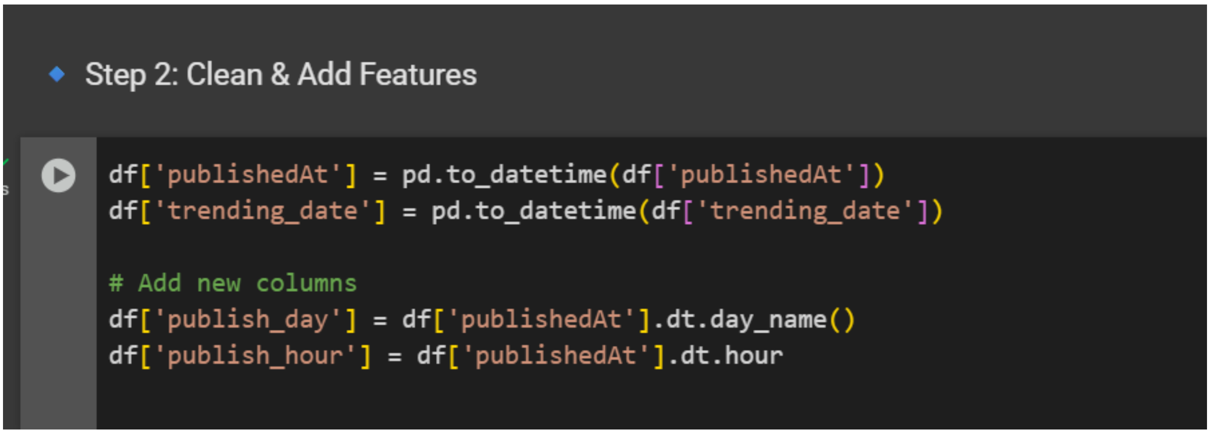
```python
import pandas as pd

# Upload manually in Colab or use:
from google.colab import files
uploaded = files.upload()

df = pd.read_csv('YouTube_Trending_Sample_500.csv')
df.head()
```

- **YouTube_Trending_Sample_500.csv**(text/csv) - 63215 bytes, last modified: 7/5/2025 - 100% done

```
Saving YouTube_Trending_Sample_500.csv to YouTube_Trending_Sample_500 (1).csv
```

	video_id	title	channelTitle	category	tags	view_count	like_count	comment_count	publishedAt	trending_date	region
0	vid_1	Sample Video Title 1	Channel 1	Entertainment	tag1, tag2	105250	5030	1010	2023-01-01 00:00:00	2023-01-02 00:00:00	US
1	vid_2	Sample Video Title 2	Channel 2	Music	tag2, tag3	110500	5060	1020	2023-01-01 01:00:00	2023-01-02 01:00:00	IN
2	vid_3	Sample Video Title 3	Channel 3	News	tag3, tag4	115750	5090	1030	2023-01-01 02:00:00	2023-01-02 02:00:00	GB
3	vid_4	Sample Video Title 4	Channel 4	Gaming	tag4, tag5	121000	5120	1040	2023-01-01 03:00:00	2023-01-02 03:00:00	CA
4	vid_5	Sample Video Title 5	Channel 5	Education	tag5, tag6	101250	5150	1050	2023-01-01 04:00:00	2023-01-02 04:00:00	AU

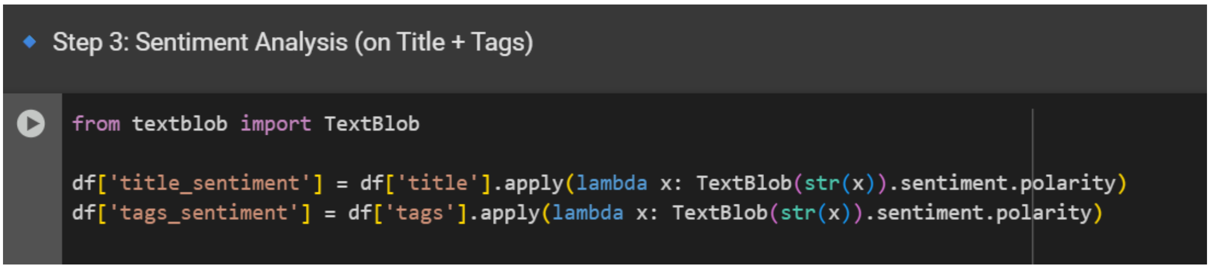
DATA CLEANING



Step 2: Clean & Add Features

```python
df['publishedAt'] = pd.to_datetime(df['publishedAt'])
df['trending_date'] = pd.to_datetime(df['trending_date'])

# Add new columns
df['publish_day'] = df['publishedAt'].dt.day_name()
df['publish_hour'] = df['publishedAt'].dt.hour
```

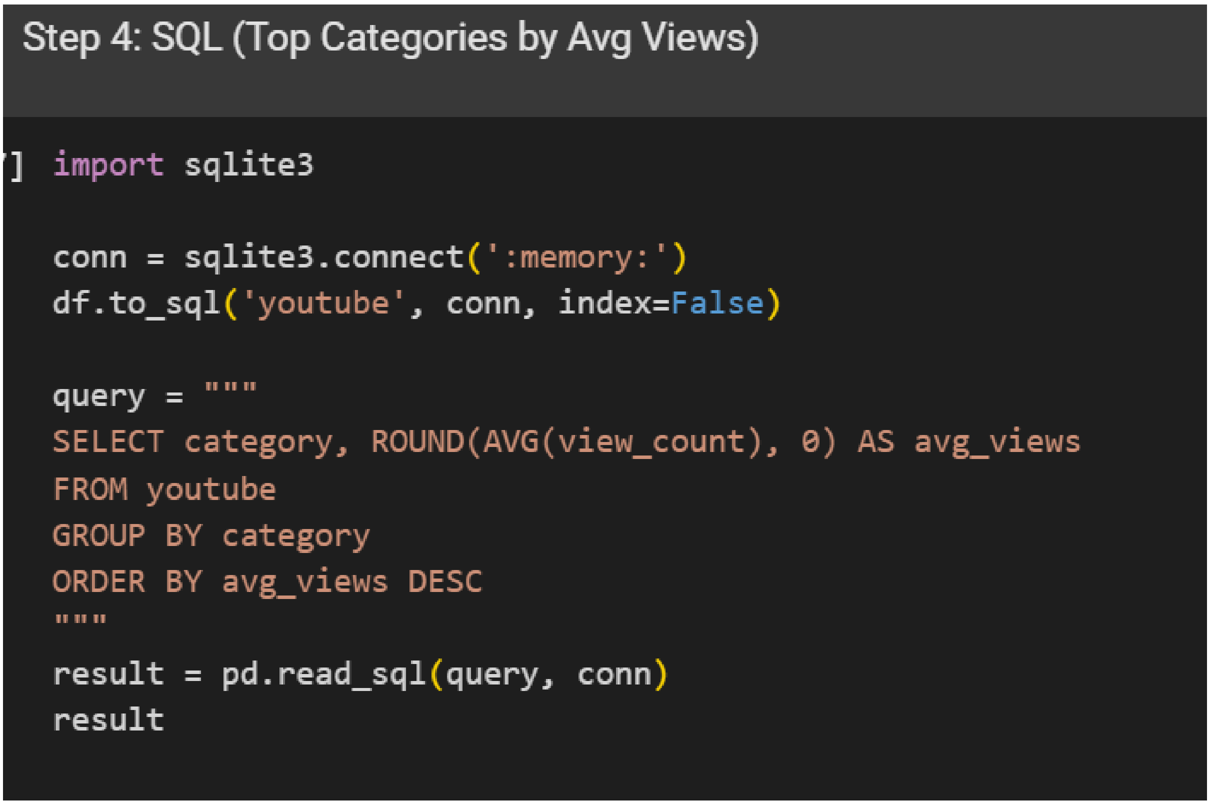
- DATA ANALYSIS



Step 3: Sentiment Analysis (on Title + Tags)

```python
from textblob import TextBlob

df['title_sentiment'] = df['title'].apply(lambda x: TextBlob(str(x)).sentiment.polarity)
df['tags_sentiment'] = df['tags'].apply(lambda x: TextBlob(str(x)).sentiment.polarity)
```

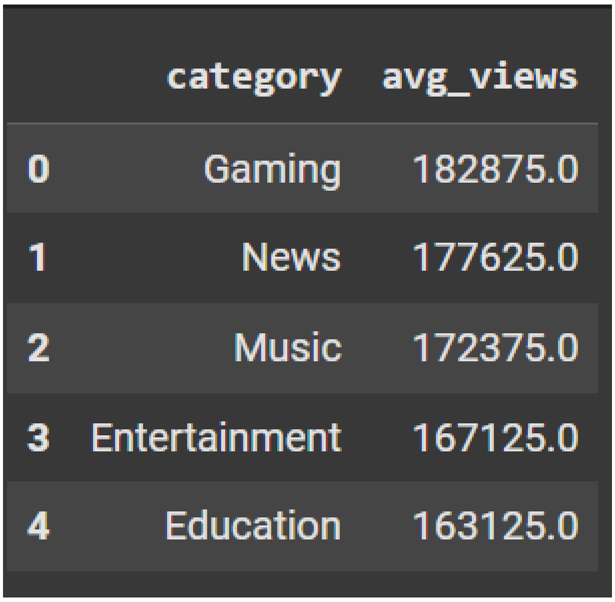
- SQL FOR FINDING TOP CATEGORIES BY AVERAGE VIEWS



Step 4: SQL (Top Categories by Avg Views)

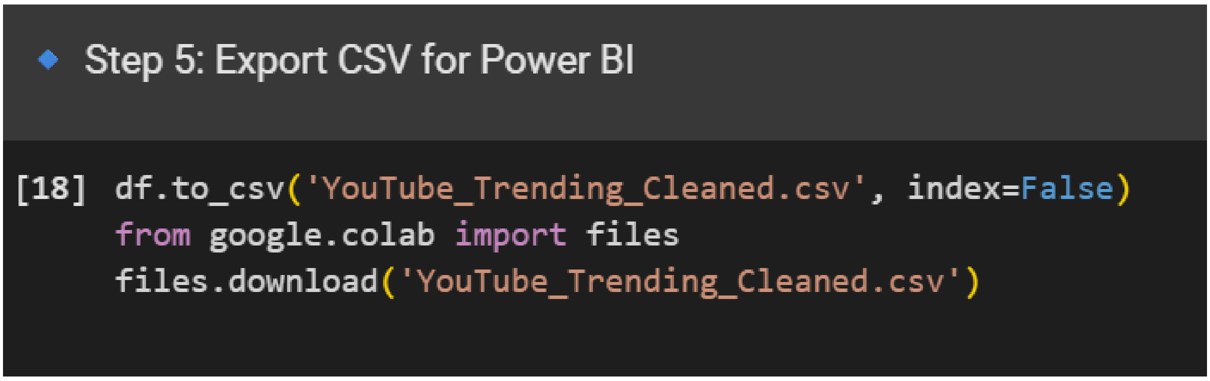
```python
import sqlite3

conn = sqlite3.connect(':memory:')
df.to_sql('youtube', conn, index=False)

query = """
SELECT category, ROUND(AVG(view_count), 0) AS avg_views
FROM youtube
GROUP BY category
ORDER BY avg_views DESC
"""
result = pd.read_sql(query, conn)
result
```

-



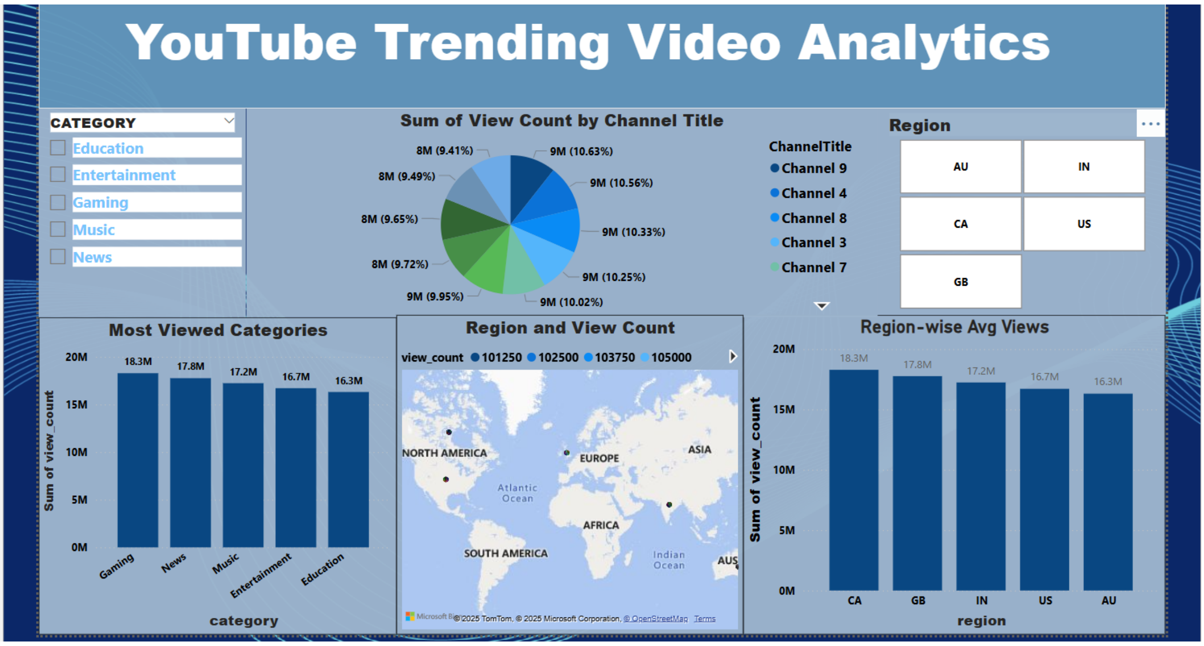
	category	avg_views
0	Gaming	182875.0
1	News	177625.0
2	Music	172375.0
3	Entertainment	167125.0
4	Education	163125.0

- EXPORT CSV FOR POWER BI

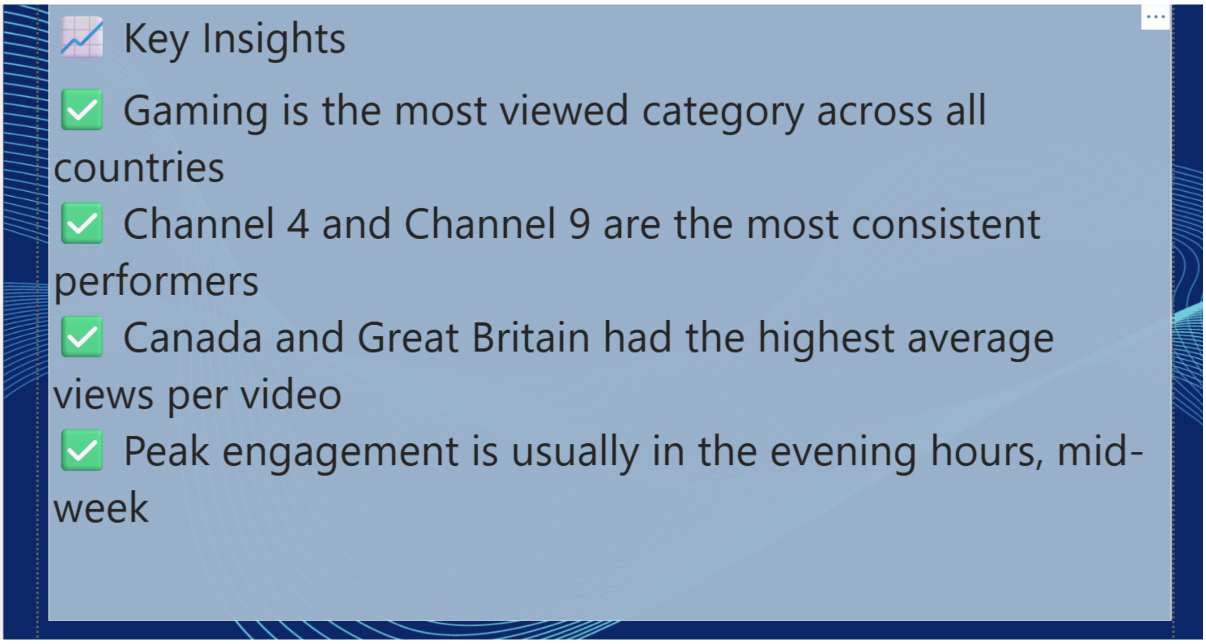


◆ Step 5: Export CSV for Power BI

```
[18] df.to_csv('YouTube_Trending_Cleaned.csv', index=False)
     from google.colab import files
     files.download('YouTube_Trending_Cleaned.csv')
```



YouTube Trending Video Analytics



📈 Key Insights

✅ Gaming is the most viewed category across all countries

✅ Channel 4 and Channel 9 are the most consistent performers

✅ Canada and Great Britain had the highest average views per video

✅ Peak engagement is usually in the evening hours, mid-week