NETFLIX AND PRIME MOVIES

EXPLORATORY DATA ANALYSIS

DONE BY: Haritha S

Date: November 2025

Table of content

01 Project overview roadmap

02 Star Journey

03 About The Dataset

04 sample dataset

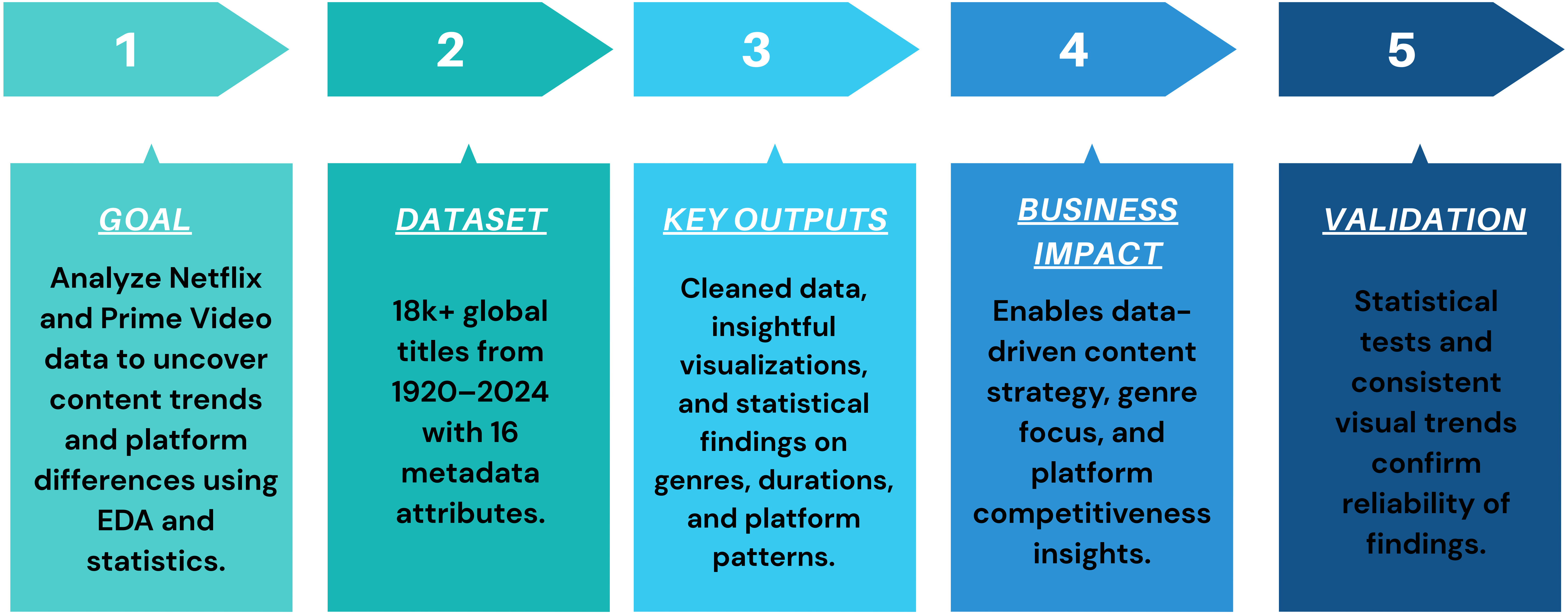
05 Business goal & Who it is useful for?

06 Project Sample work

07 Summary of insights

08 What i learned

PROJECT OVERVIEW ROADMAP



1

GOAL

Analyze Netflix and Prime Video data to uncover content trends and platform differences using EDA and statistics.

2

DATASET

18k+ global titles from 1920–2024 with 16 metadata attributes.

3

KEY OUTPUTS

Cleaned data, insightful visualizations, and statistical findings on genres, durations, and platform patterns.

4

BUSINESS IMPACT

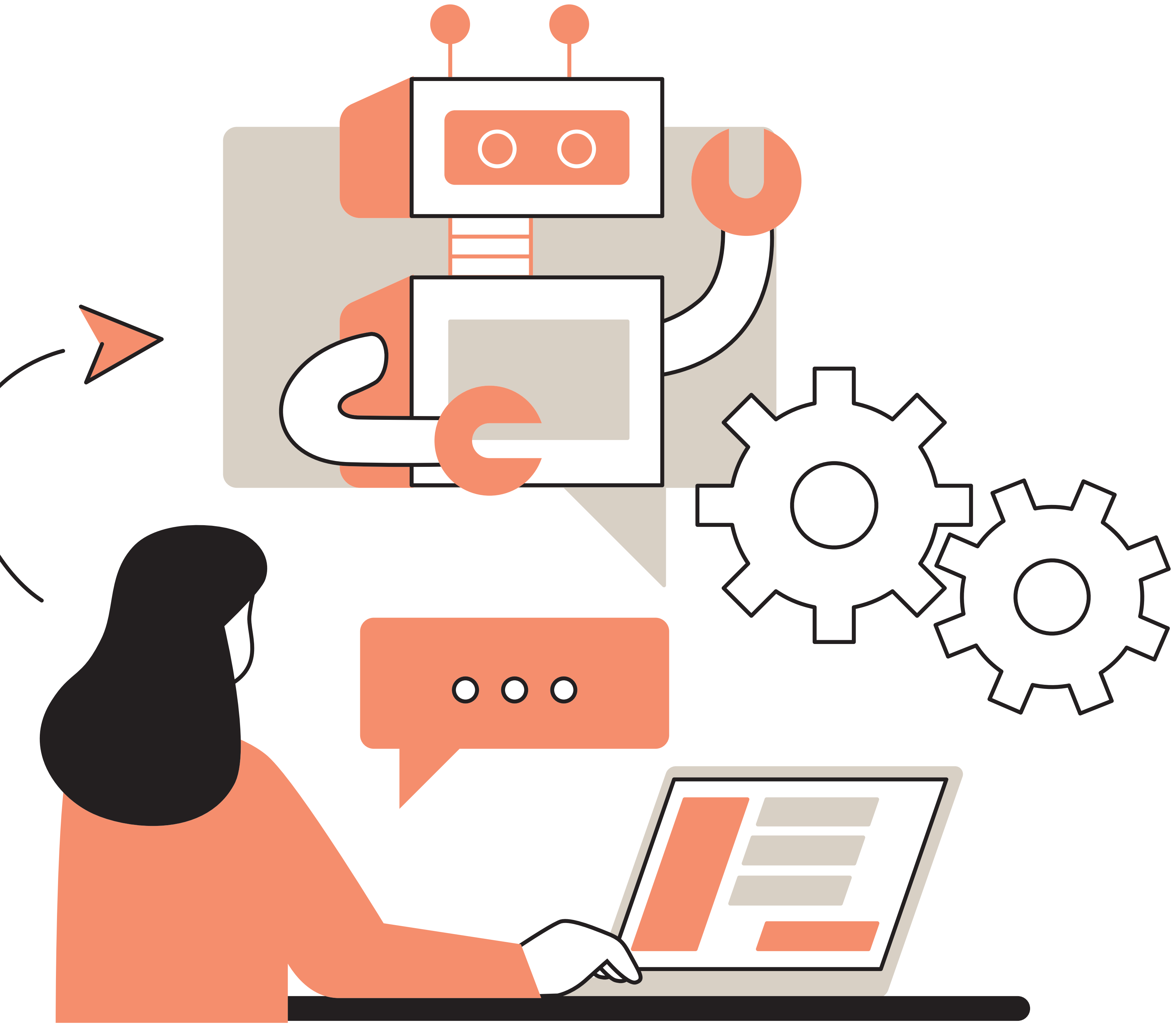
Enables data-driven content strategy, genre focus, and platform competitiveness insights.

5

VALIDATION

Statistical tests and consistent visual trends confirm reliability of findings.

STAR JOURNEY



SITUATION

I began learning Data Science by studying ***statistics and EDA with Python libraries*** and applied those skills inorder to gain practical hands-on experience.

TASK

Use my new skills to ***perform a complete exploratory analysis*** on Netflix and Prime Video content data.

ACTION

I ***cleaned*** and prepared the data, ***handled missing values, engineered features***, and performed full EDA with ***statistical tests*** to compare platform behavior.

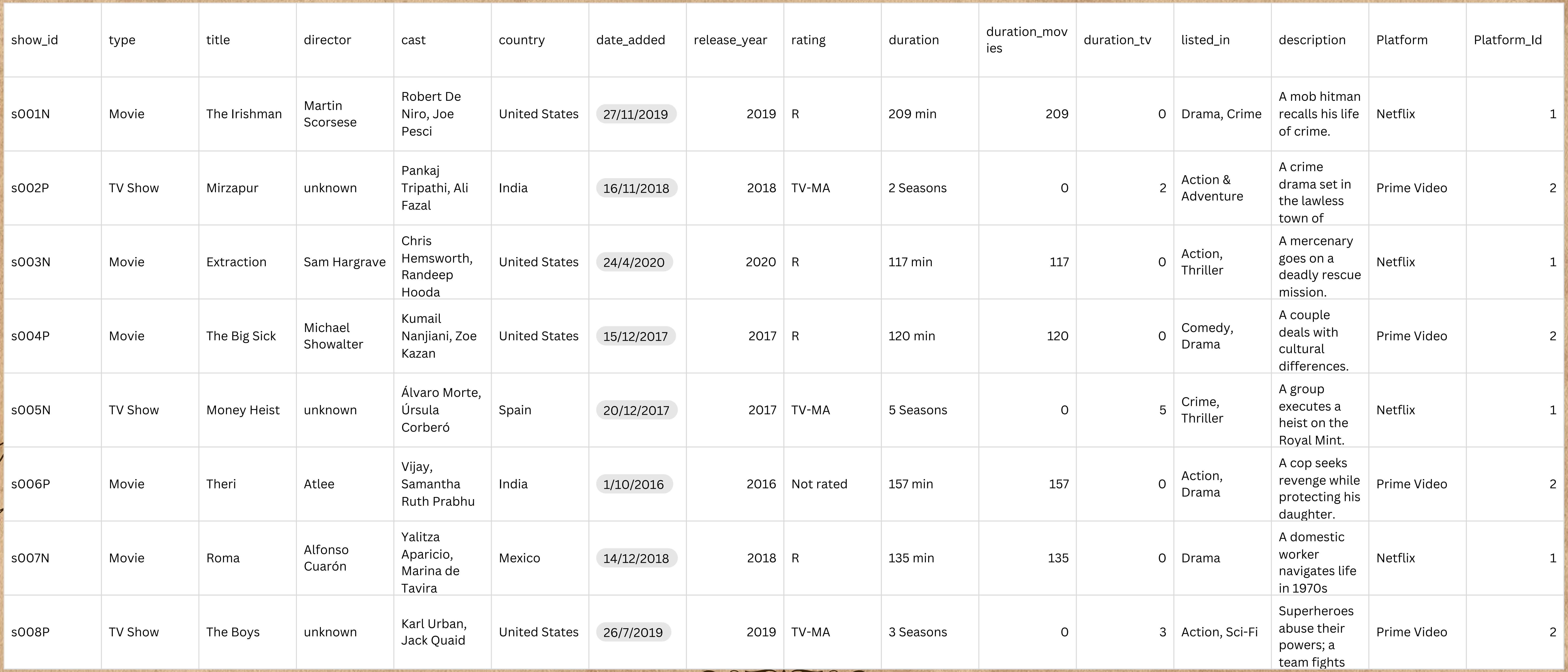
RESULT

Discovered platform trends, genre patterns, duration differences, and insights *useful for content strategy*, while strengthening my understanding of EDA and statistics.

ABOUT THE DATASET

1. ***Dataset Source:*** **Public content catalog for Netflix and Prime Video (Excel file: Content Catalog (Netflix - Prime Video)).**
2. ***Dataset Period:*** **Release years range from 1920–2024; content added dates up to 2021**
3. ***Number of Rows:*** **18,477 titles.**
4. ***Number of Columns:*** **16 attributes**
5. ***Region Focus:*** **Primarily U.S., India, UK, Japan, and other international contributions.**

SAMPLE DATASET



show_id	type	title	director	cast	country	date_added	release_year	rating	duration	duration_movies	duration_tv	listed_in	description	Platform	Platform_Id
s001N	Movie	The Irishman	Martin Scorsese	Robert De Niro, Joe Pesci	United States	27/11/2019	2019	R	209 min	209	0	Drama, Crime	A mob hitman recalls his life of crime.	Netflix	1
s002P	TV Show	Mirzapur	unknown	Pankaj Tripathi, Ali Fazal	India	16/11/2018	2018	TV-MA	2 Seasons	0	2	Action & Adventure	A crime drama set in the lawless town of	Prime Video	2
s003N	Movie	Extraction	Sam Hargrave	Chris Hemsworth, Randeep Hooda	United States	24/4/2020	2020	R	117 min	117	0	Action, Thriller	A mercenary goes on a deadly rescue mission.	Netflix	1
s004P	Movie	The Big Sick	Michael Showalter	Kumail Nanjiani, Zoe Kazan	United States	15/12/2017	2017	R	120 min	120	0	Comedy, Drama	A couple deals with cultural differences.	Prime Video	2
s005N	TV Show	Money Heist	unknown	Álvaro Morte, Úrsula Corberó	Spain	20/12/2017	2017	TV-MA	5 Seasons	0	5	Crime, Thriller	A group executes a heist on the Royal Mint.	Netflix	1
s006P	Movie	Theri	Atlee	Vijay, Samantha Ruth Prabhu	India	1/10/2016	2016	Not rated	157 min	157	0	Action, Drama	A cop seeks revenge while protecting his daughter.	Prime Video	2
s007N	Movie	Roma	Alfonso Cuarón	Yalitza Aparicio, Marina de Tavira	Mexico	14/12/2018	2018	R	135 min	135	0	Drama	A domestic worker navigates life in 1970s	Netflix	1
s008P	TV Show	The Boys	unknown	Karl Urban, Jack Quaid	United States	26/7/2019	2019	TV-MA	3 Seasons	0	3	Action, Sci-Fi	Superheroes abuse their powers; a team fights	Prime Video	2

BUSINESS GOAL

To strengthen viewer engagement and platform competitiveness by optimizing content strategy—focusing on top genres, regional diversity, and balancing movies vs TV shows.

WHO IS IT USEFUL FOR?

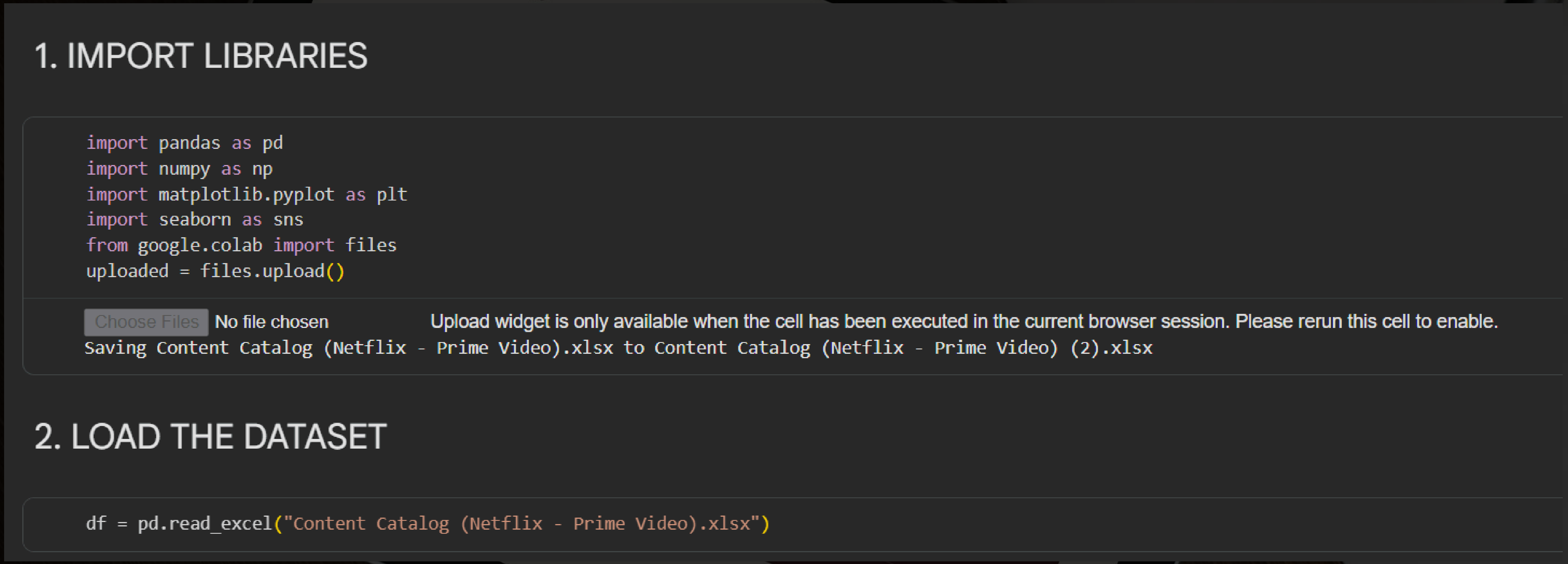
- Streaming platform content strategy teams
- Media analytics professionals
- Product managers and business analysts
- Researchers in entertainment industry trends

INDUSTRY BACKGROUND
WHAT IS THE INDUSTRY'S HISTORY AND WHAT ARE ITS USUAL TRENDS? DO YOU SEE NEW PATTERNS DEVELOPING? GIVE A PREDICTION OR OUTLOOK ABOUT WHERE THE INDUSTRY IS HEADED.
40
20
10
0
2020
2021
2022
2023
2024

SAMPLE WORK OF MY PROJECT

JUPITER NOTEBOOK

IMPORT LIBRARIES, LOAD DATASET



1. IMPORT LIBRARIES

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from google.colab import files
uploaded = files.upload()
```

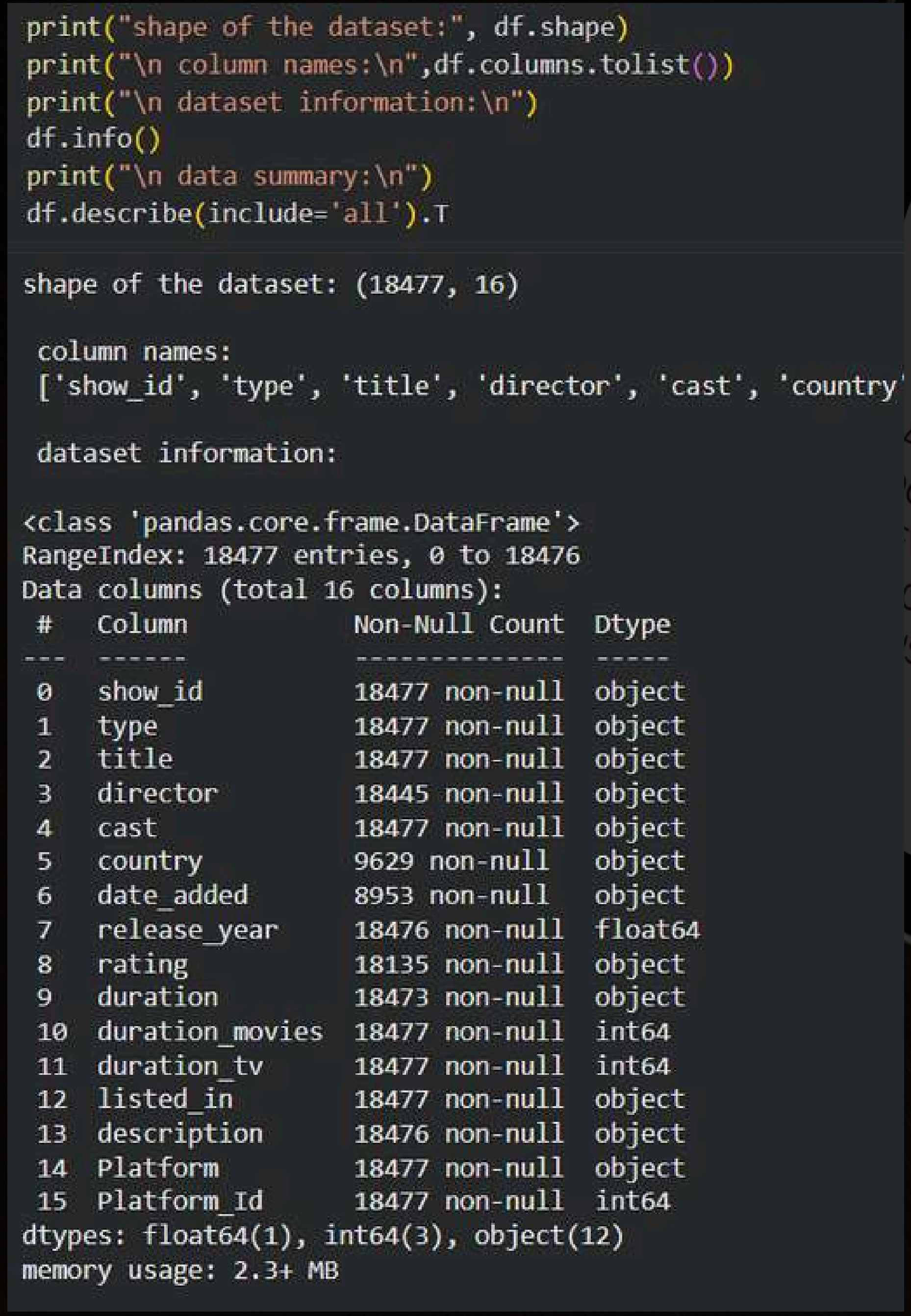
Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

```
Saving Content Catalog (Netflix - Prime Video).xlsx to Content Catalog (Netflix - Prime Video) (2).xlsx
```

2. LOAD THE DATASET

```
df = pd.read_excel("Content Catalog (Netflix - Prime Video).xlsx")
```

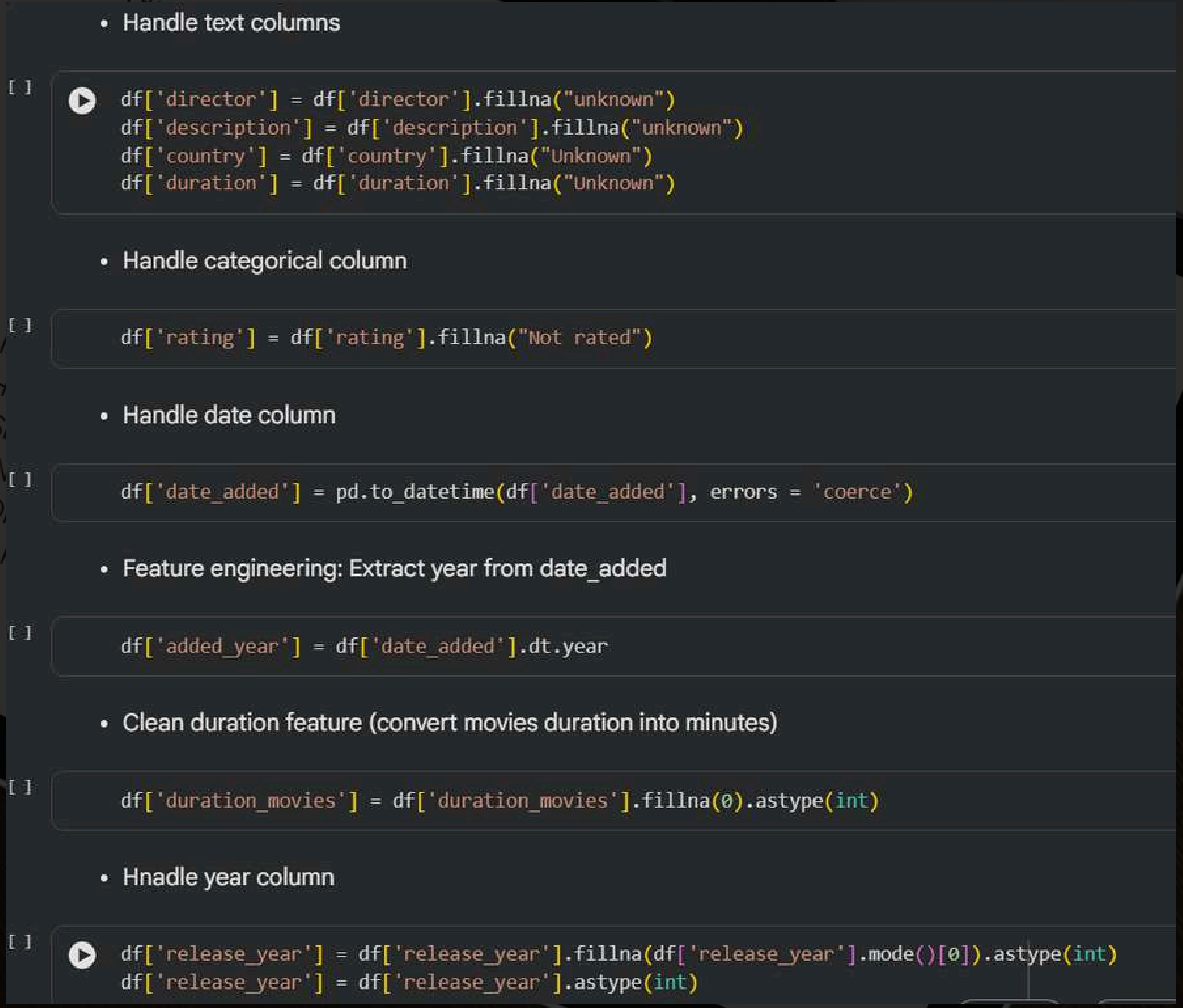
DATA EXPLORATION



```
print("shape of the dataset:", df.shape)
print("\n column names:\n",df.columns.tolist())
print("\n dataset information:\n")
df.info()
print("\n data summary:\n")
df.describe(include='all').T
```

```
shape of the dataset: (18477, 16)

 column names:
 ['show_id', 'type', 'title', 'director', 'cast', 'country'

 dataset information:

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 18477 entries, 0 to 18476
Data columns (total 16 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   show_id         18477 non-null  object
 1   type            18477 non-null  object
 2   title           18477 non-null  object
 3   director        18445 non-null  object
 4   cast            18477 non-null  object
 5   country         9629 non-null   object
 6   date_added      8953 non-null   object
 7   release_year    18476 non-null  float64
 8   rating          18135 non-null  object
 9   duration        18473 non-null  object
 10  duration_movies 18477 non-null  int64
 11  duration_tv     18477 non-null  int64
 12  listed_in       18477 non-null  object
 13  description     18476 non-null  object
 14  Platform        18477 non-null  object
 15  Platform_Id     18477 non-null  int64
dtypes: float64(1), int64(3), object(12)
memory usage: 2.3+ MB
```

DATA CLEANING



- Handle text columns

```
df['director'] = df['director'].fillna("unknown")
df['description'] = df['description'].fillna("unknown")
df['country'] = df['country'].fillna("Unknown")
df['duration'] = df['duration'].fillna("Unknown")
```

- Handle categorical column

```
df['rating'] = df['rating'].fillna("Not rated")
```

- Handle date column

```
df['date_added'] = pd.to_datetime(df['date_added'], errors = 'coerce')
```

- Feature engineering: Extract year from date_added

```
df['added_year'] = df['date_added'].dt.year
```

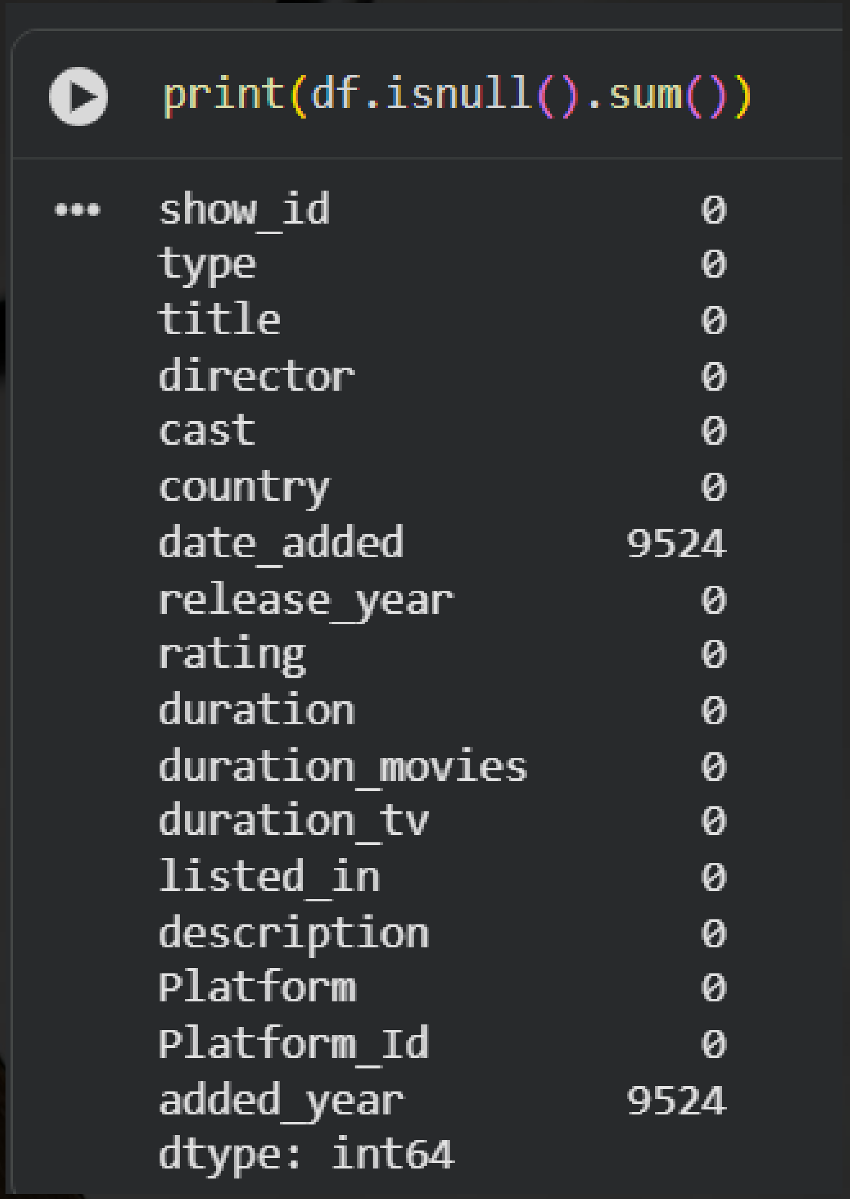
- Clean duration feature (convert movies duration into minutes)

```
df['duration_movies'] = df['duration_movies'].fillna(0).astype(int)
```

- Hnadle year column

```
df['release_year'] = df['release_year'].fillna(df['release_year'].mode()[0]).astype(int)
df['release_year'] = df['release_year'].astype(int)
```

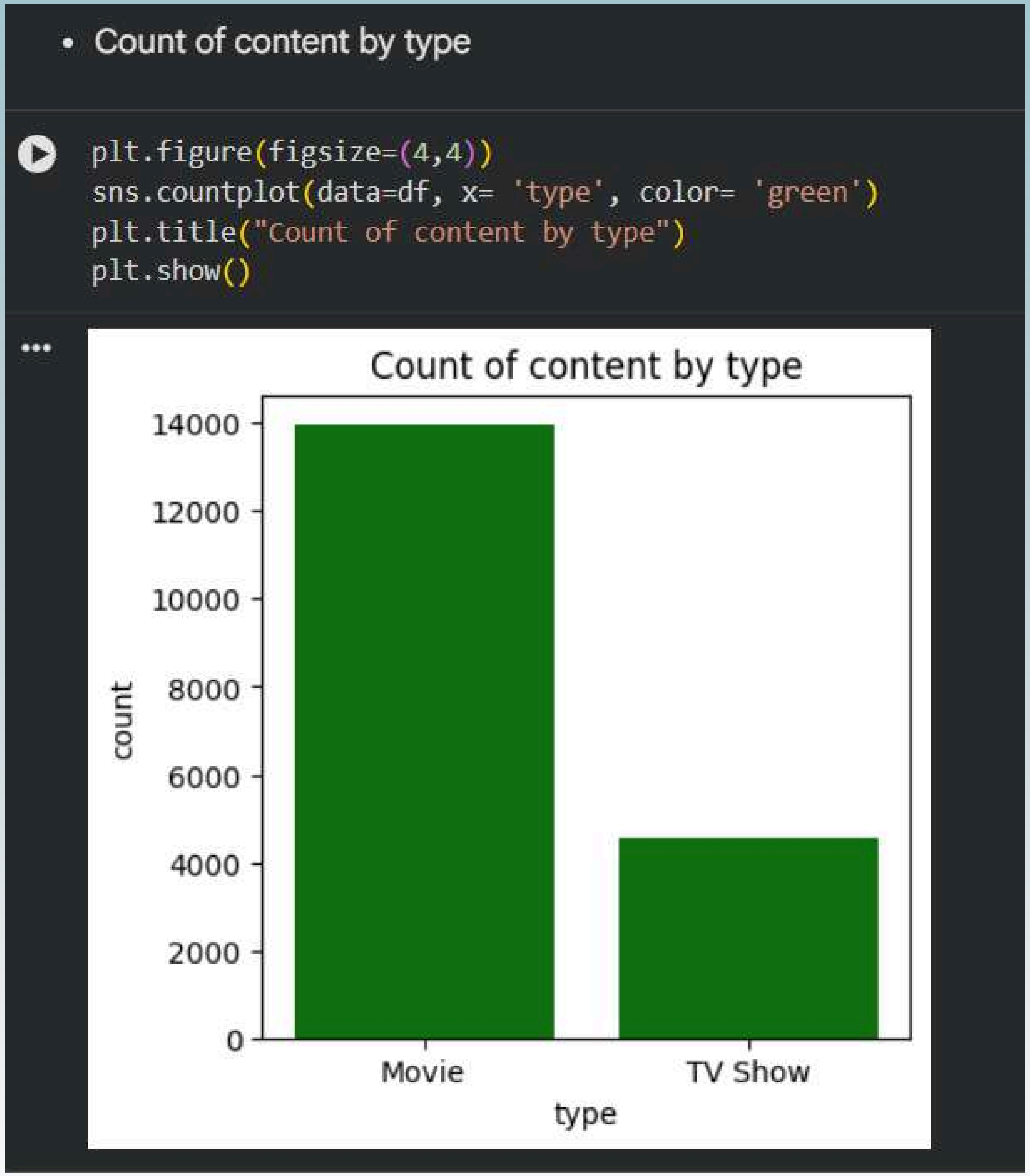
CLEANED DATA



```
print(df.isnull().sum())
```

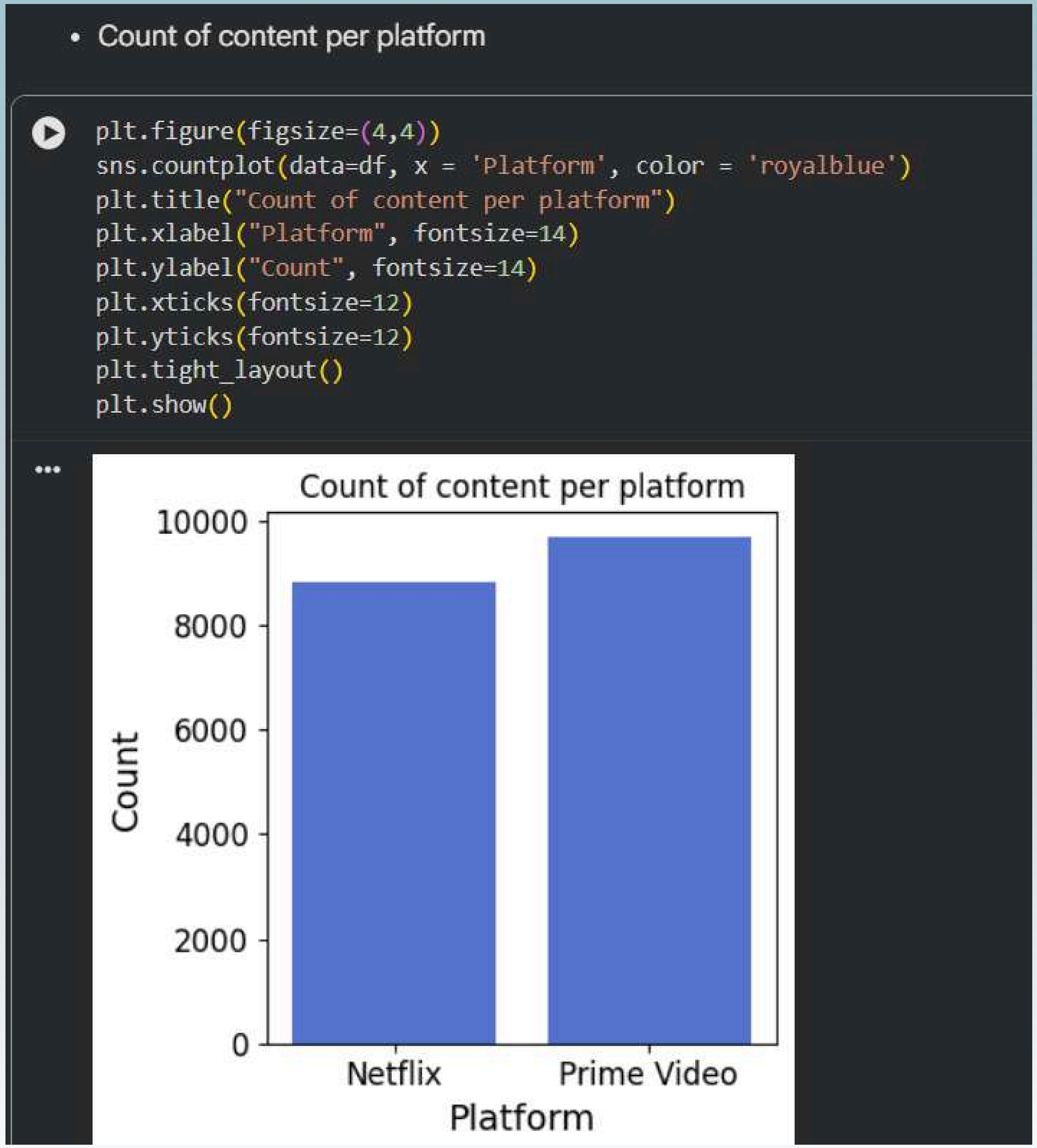
```
show_id            0
type               0
title              0
director           0
cast               0
country            0
date_added      9524
release_year       0
rating             0
duration           0
duration_movies    0
duration_tv        0
listed_in          0
description        0
Platform           0
Platform_Id        0
added_year      9524
dtype: int64
```

UNIVARIATE ANALYSIS



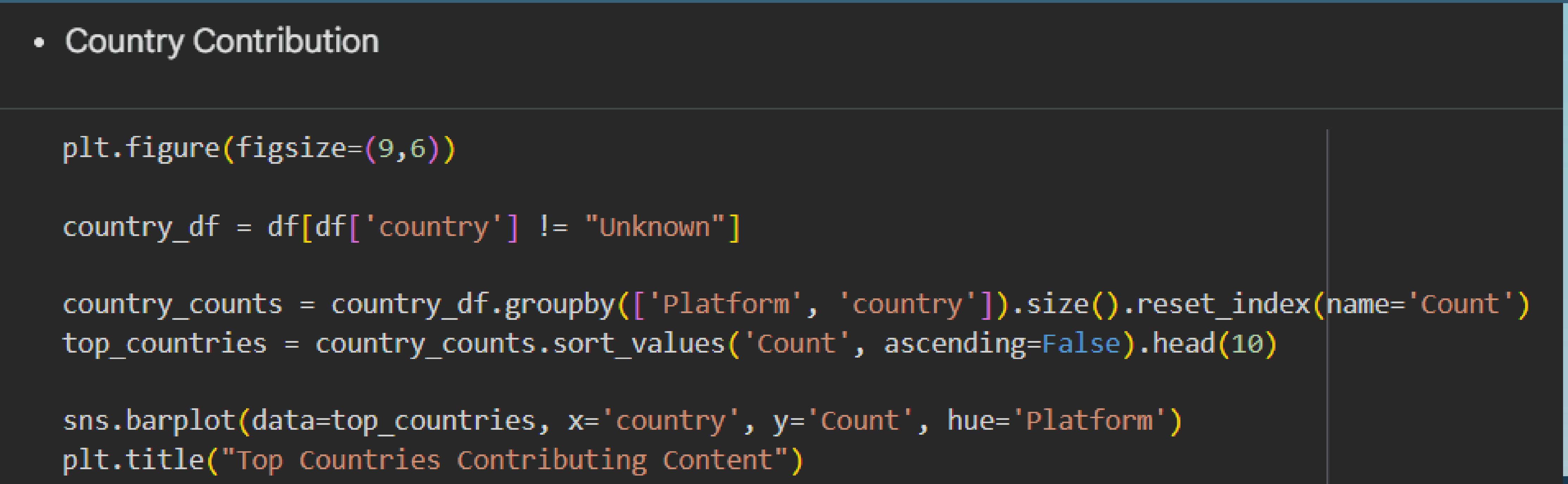
- Count of content by type

```
plt.figure(figsize=(4,4))
sns.countplot(data=df, x= 'type', color= 'green')
plt.title("Count of content by type")
plt.show()
```



- Count of content per platform

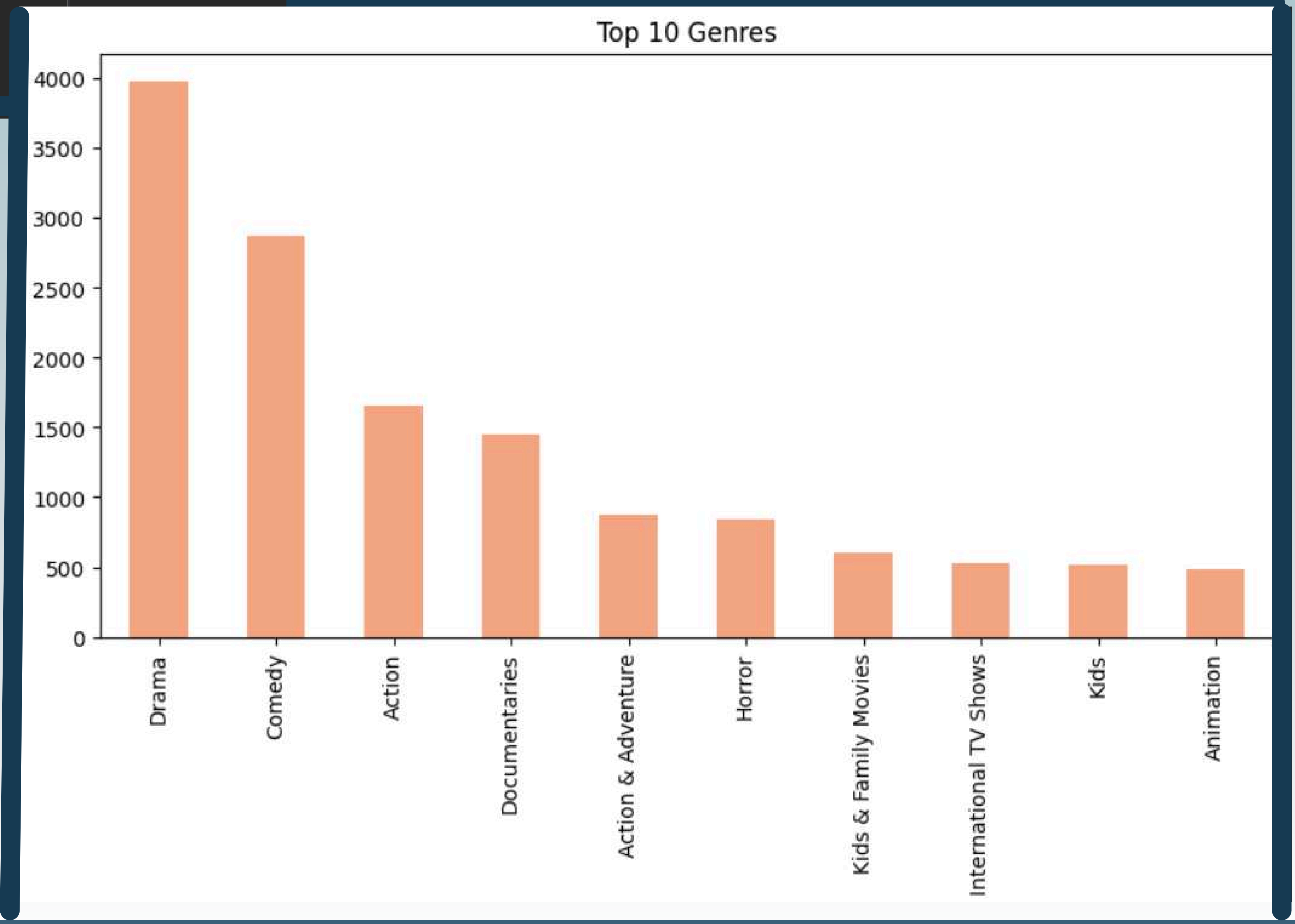
```
plt.figure(figsize=(4,4))
sns.countplot(data=df, x = 'Platform', color = 'royalblue')
plt.title("Count of content per platform")
plt.xlabel("Platform", fontsize=14)
plt.ylabel("Count", fontsize=14)
plt.xticks(fontsize=12)
plt.yticks(fontsize=12)
plt.tight_layout()
plt.show()
```

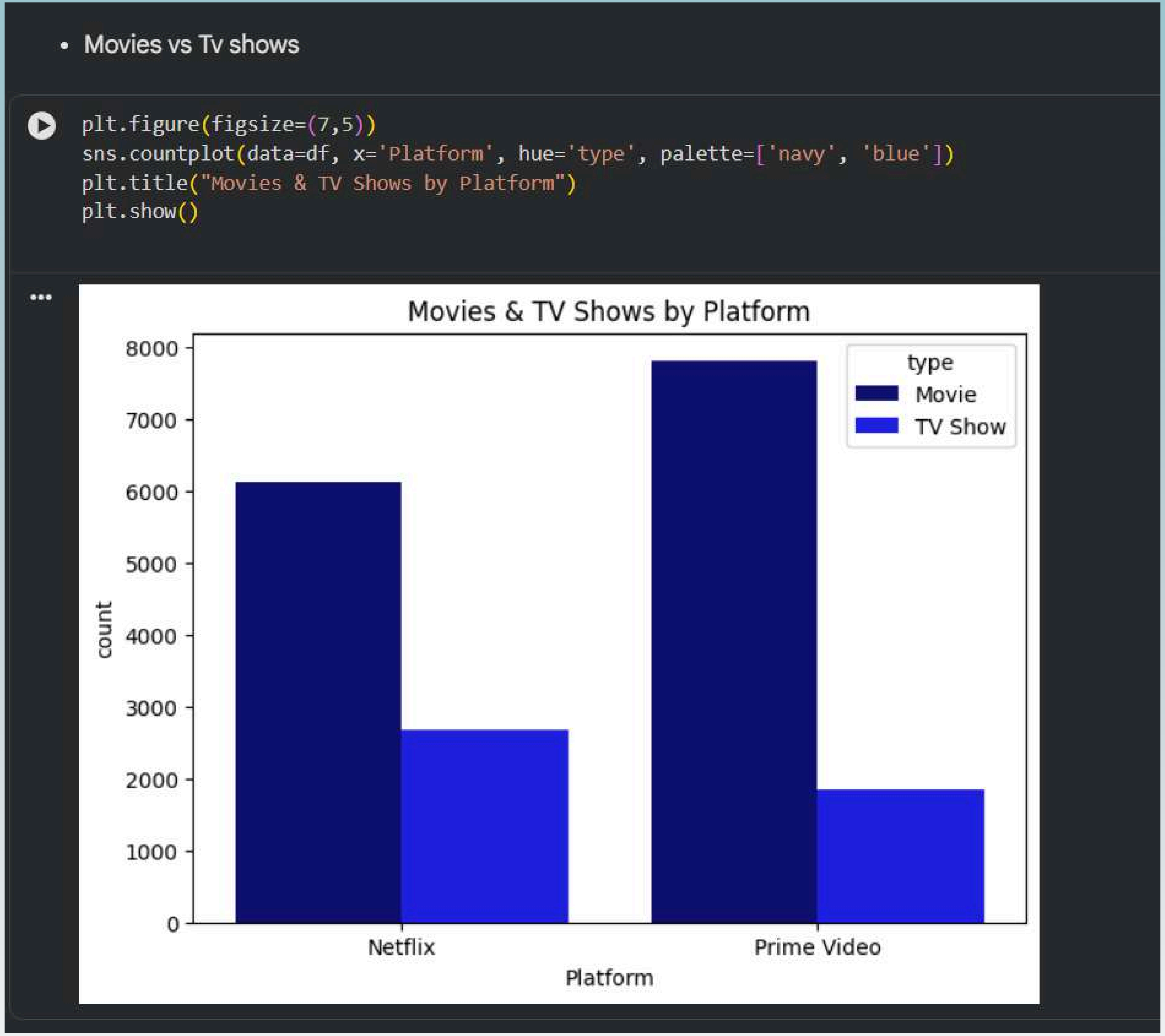


- Country Contribution

```
plt.figure(figsize=(9,6))

country_df = df[df['country'] != "Unknown"]

country_counts = country_df.groupby(['Platform', 'country']).size().reset_index(name='Count')
top_countries = country_counts.sort_values('Count', ascending=False).head(10)

sns.barplot(data=top_countries, x='country', y='Count', hue='Platform')
plt.title("Top Countries Contributing Content")
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

UNIVARIATE ANALYSIS



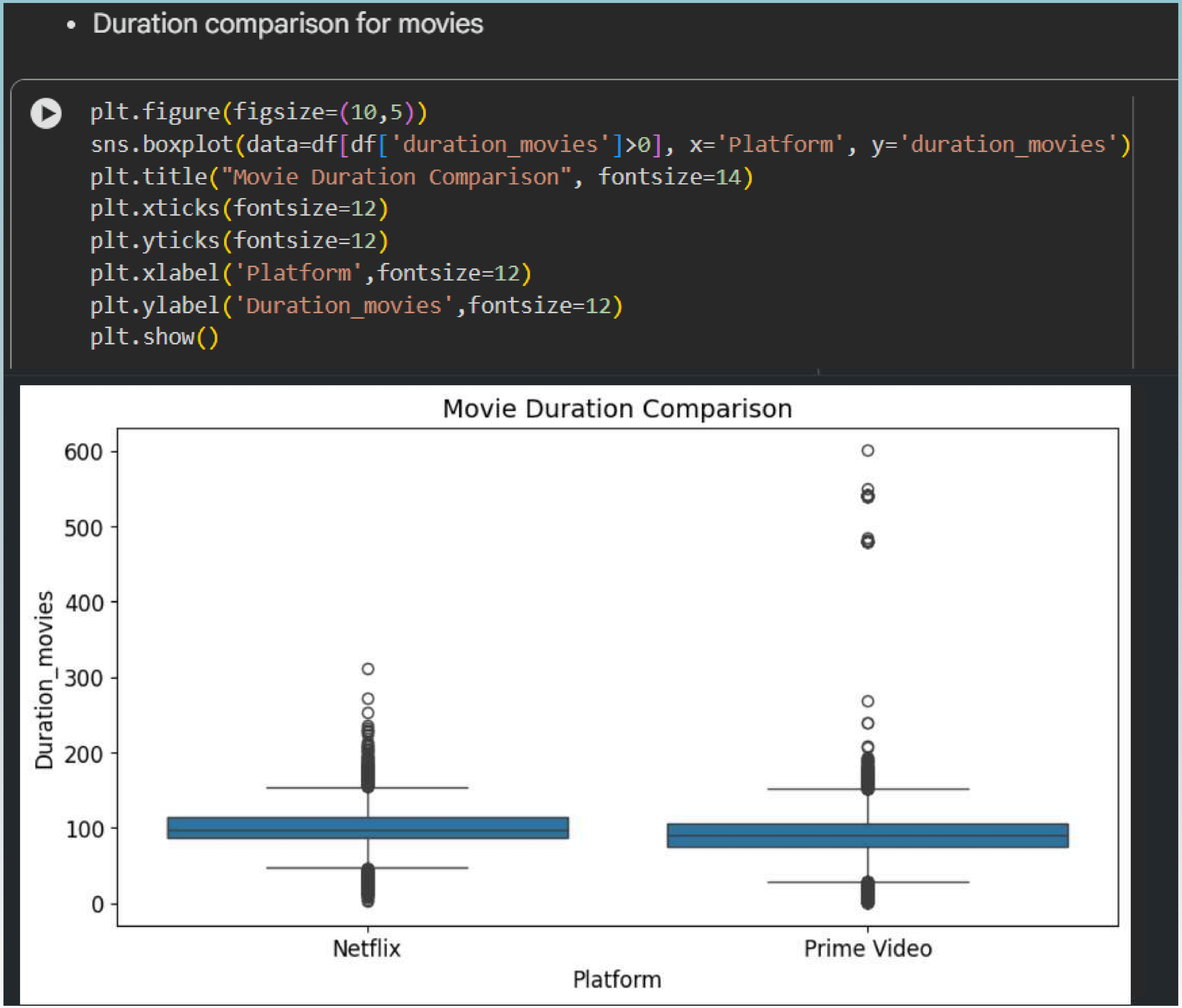


- Movies vs Tv shows

```
plt.figure(figsize=(7,5))
sns.countplot(data=df, x='Platform', hue='type', palette=['navy', 'blue'])
plt.title("Movies & TV Shows by Platform")
plt.show()
```

BIVARIATE ANALYSIS

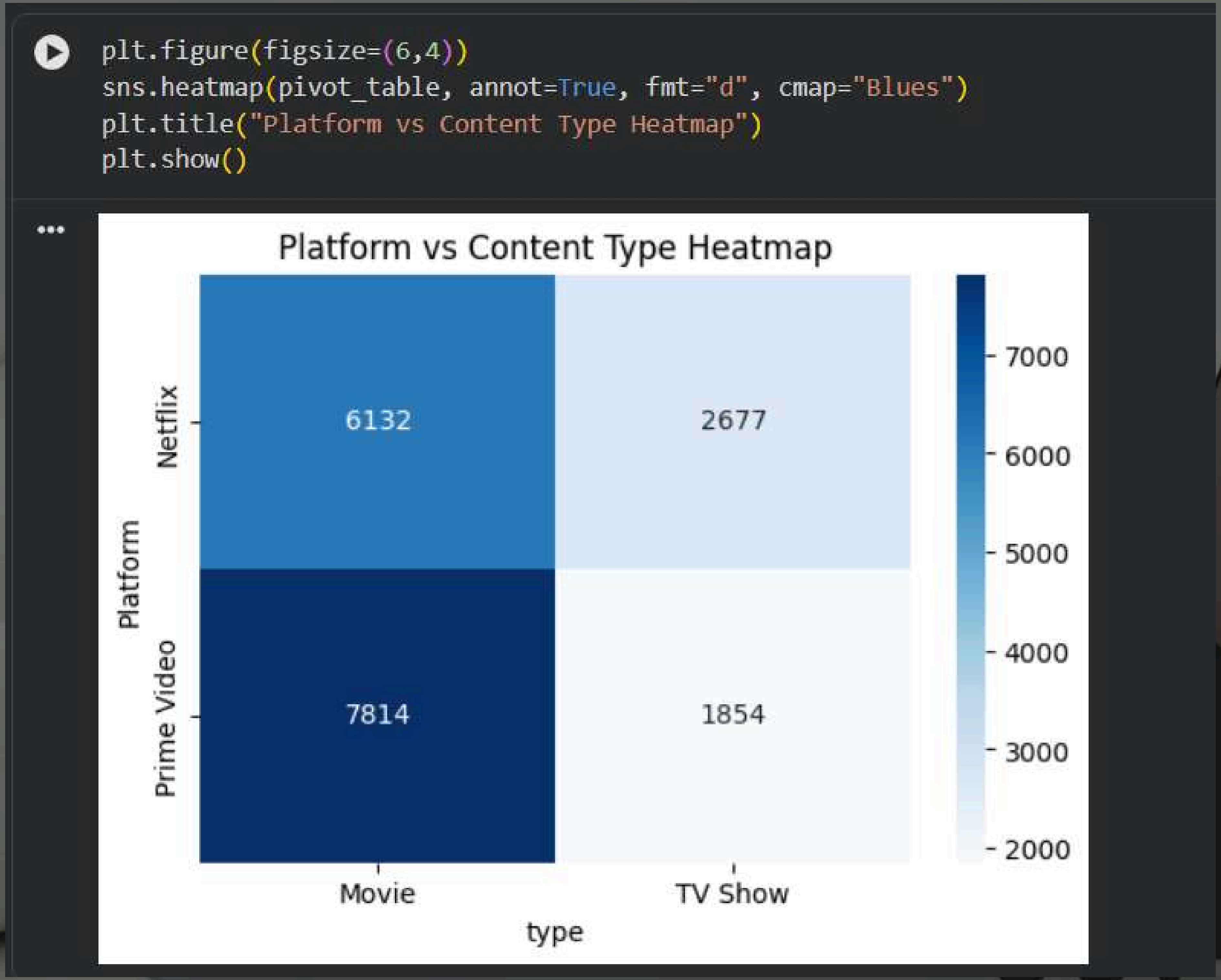
BIVARIATE ANALYSIS



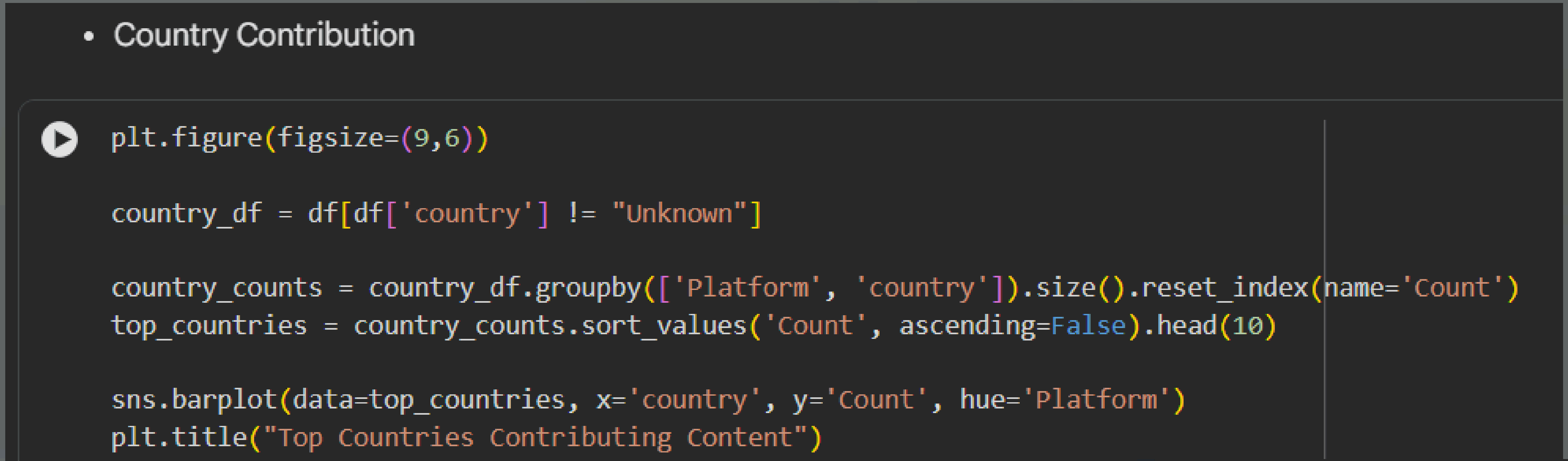
- Duration comparison for movies

```
plt.figure(figsize=(10,5))
sns.boxplot(data=df[df['duration_movies']>0], x='Platform', y='duration_movies')
plt.title("Movie Duration Comparison", fontsize=14)
plt.xticks(fontsize=12)
plt.yticks(fontsize=12)
plt.xlabel('Platform',fontsize=12)
plt.ylabel('Duration_movies',fontsize=12)
plt.show()
```

MULTIVARIATE ANALYSIS

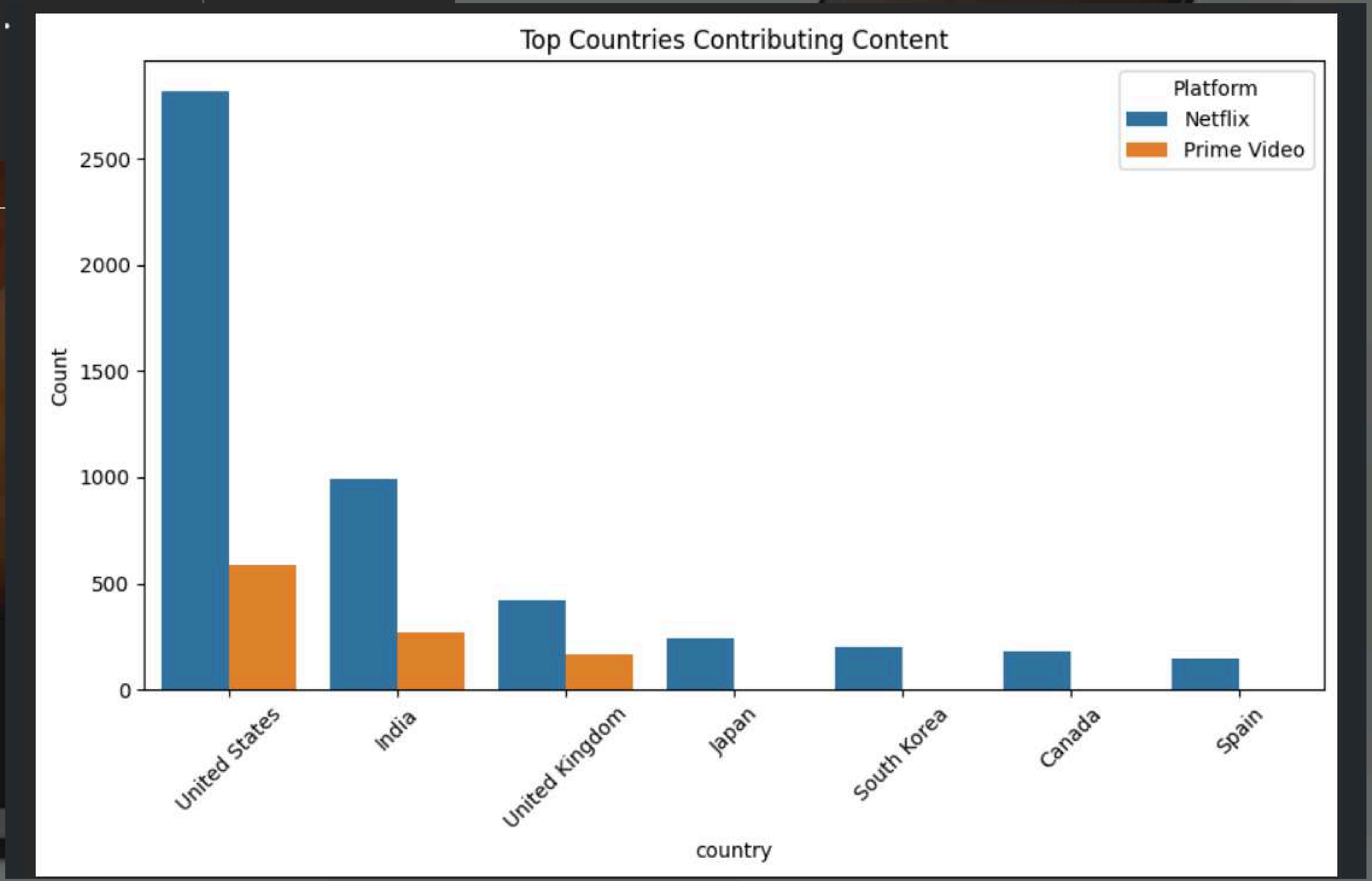


```
plt.figure(figsize=(6,4))
sns.heatmap(pivot_table, annot=True, fmt="d", cmap="Blues")
plt.title("Platform vs Content Type Heatmap")
plt.show()
```



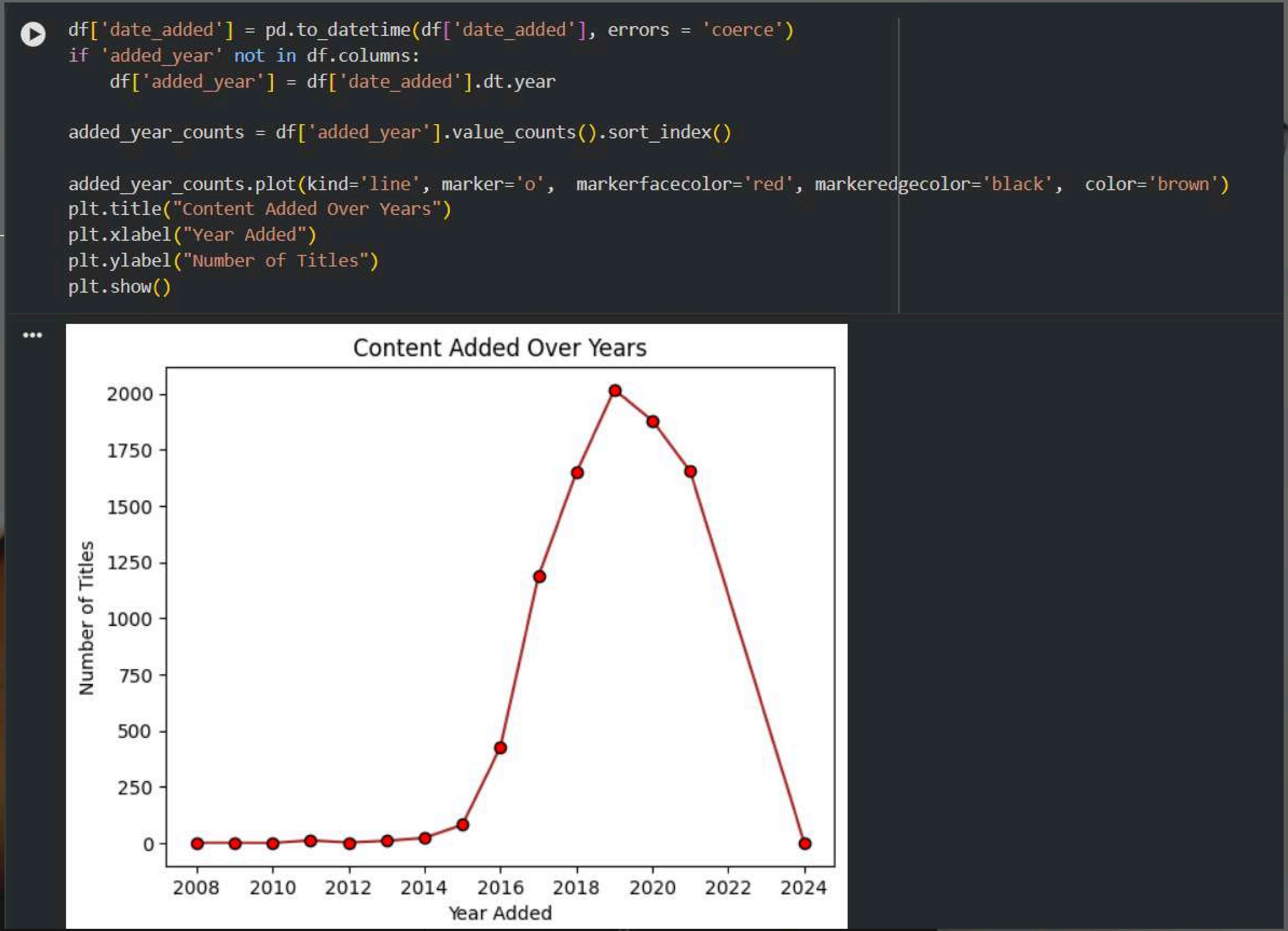
- Country Contribution

```
plt.figure(figsize=(9,6))

country_df = df[df['country'] != "Unknown"]

country_counts = country_df.groupby(['Platform', 'country']).size().reset_index(name='Count')
top_countries = country_counts.sort_values('Count', ascending=False).head(10)

sns.barplot(data=top_countries, x='country', y='Count', hue='Platform')
plt.title("Top Countries Contributing Content")
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

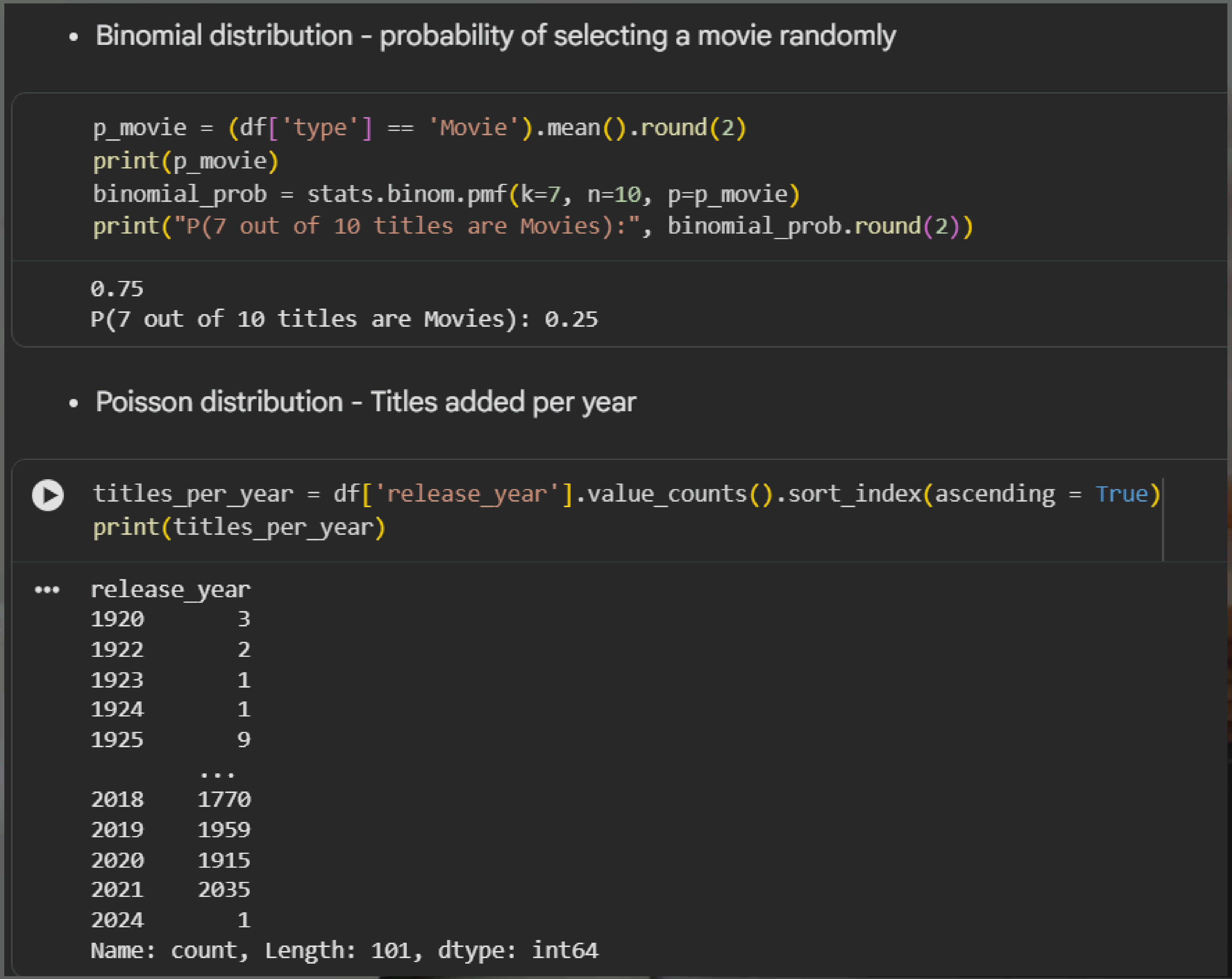


MULTIVARIATE ANALYSIS

TIME-BASED ANALYSIS



```python
df['date_added'] = pd.to_datetime(df['date_added'], errors = 'coerce')
if 'added_year' not in df.columns:
    df['added_year'] = df['date_added'].dt.year

added_year_counts = df['added_year'].value_counts().sort_index()

added_year_counts.plot(kind='line', marker='o',  markerfacecolor='red', markeredgecolor='black',  color='brown')
plt.title("Content Added Over Years")
plt.xlabel("Year Added")
plt.ylabel("Number of Titles")
plt.show()
```



- Binomial distribution - probability of selecting a movie randomly

```
p_movie = (df['type'] == 'Movie').mean().round(2)
print(p_movie)
binomial_prob = stats.binom.pmf(k=7, n=10, p=p_movie)
print("P(7 out of 10 titles are Movies):", binomial_prob.round(2))
```

```
0.75
P(7 out of 10 titles are Movies): 0.25
```

- Poisson distribution - Titles added per year

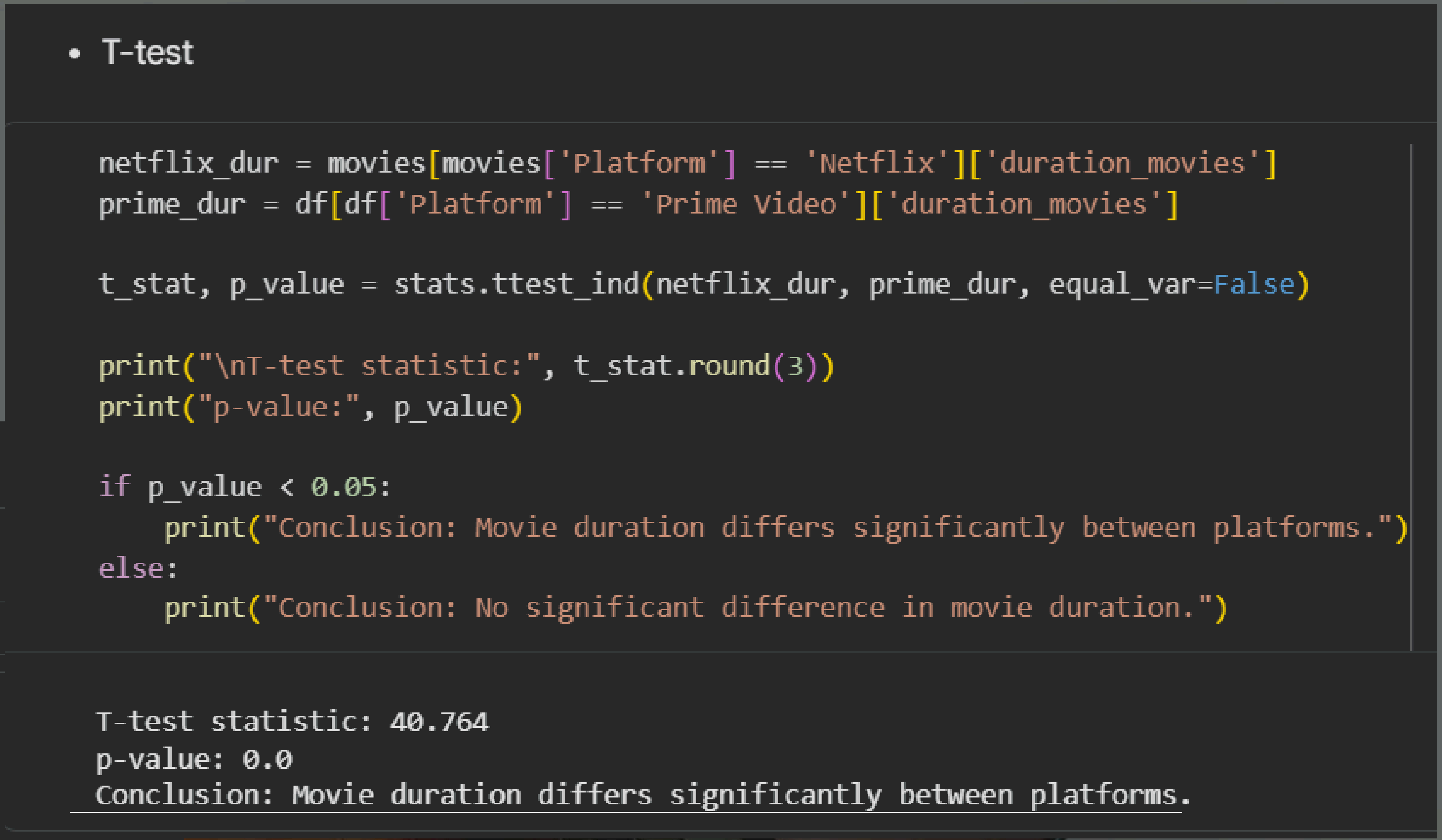
```
titles_per_year = df['release_year'].value_counts().sort_index(ascending = True)
print(titles_per_year)
```

```
release_year
1920       3
1922       2
1923       1
1924       1
1925       9
        ... 
2018    1770
2019    1959
2020    1915
2021    2035
2024       1
Name: count, Length: 101, dtype: int64
```

STATISTICAL ANALYSIS

T-TEST

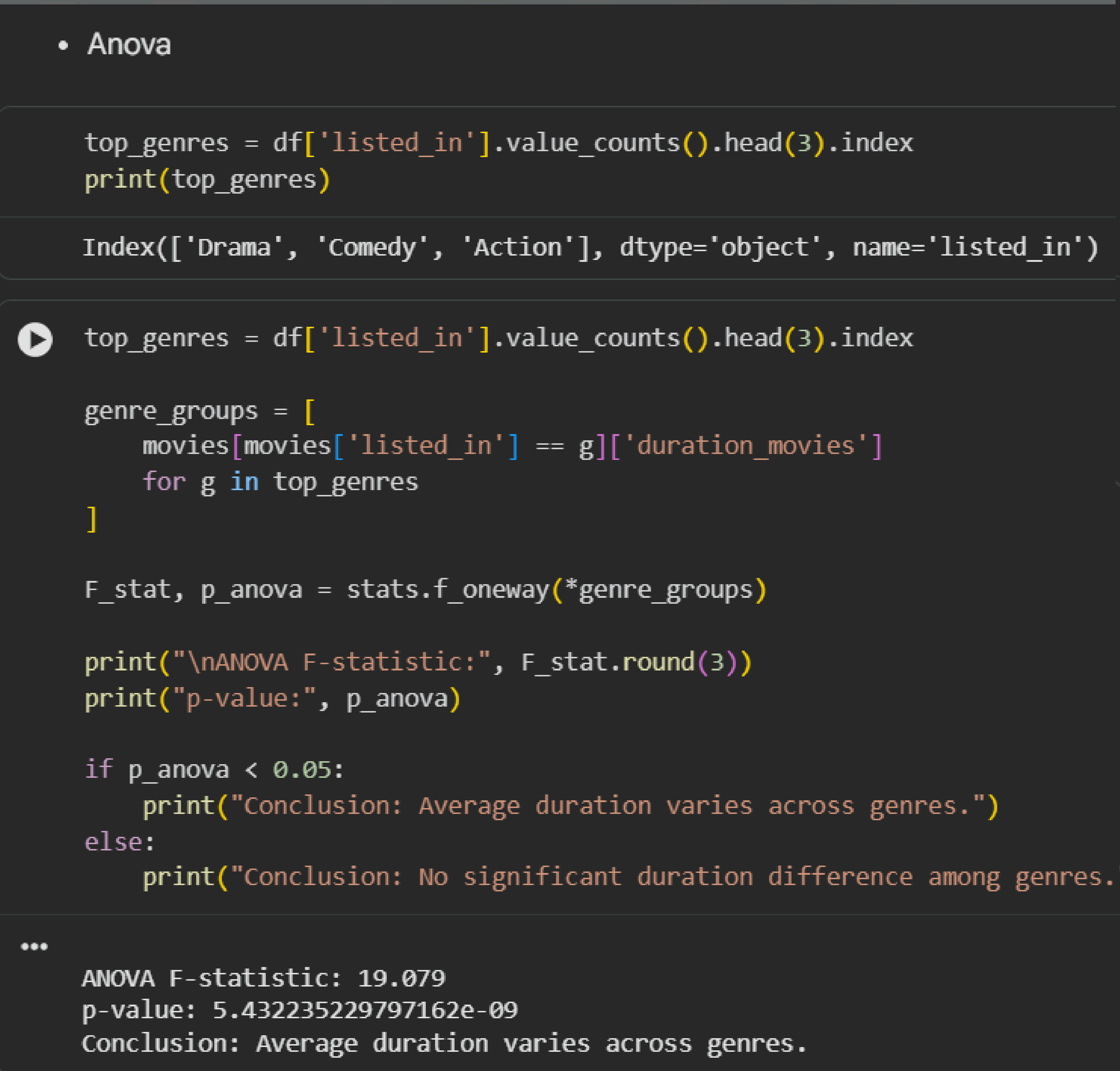
--->



- T-test

```
netflix_dur = movies[movies['Platform'] == 'Netflix']['duration_movies']
prime_dur = df[df['Platform'] == 'Prime Video']['duration_movies']

t_stat, p_value = stats.ttest_ind(netflix_dur, prime_dur, equal_var=False)

print("\nT-test statistic:", t_stat.round(3))
print("p-value:", p_value)

if p_value < 0.05:
    print("Conclusion: Movie duration differs significantly between platforms.")
else:
    print("Conclusion: No significant difference in movie duration.")
```

```
T-test statistic: 40.764
p-value: 0.0
Conclusion: Movie duration differs significantly between platforms.
```



- Anova

```
top_genres = df['listed_in'].value_counts().head(3).index
print(top_genres)
```

```
Index(['Drama', 'Comedy', 'Action'], dtype='object', name='listed_in')
```

```
top_genres = df['listed_in'].value_counts().head(3).index

genre_groups = [
    movies[movies['listed_in'] == g]['duration_movies']
    for g in top_genres
]

F_stat, p_anova = stats.f_oneway(*genre_groups)

print("\nANOVA F-statistic:", F_stat.round(3))
print("p-value:", p_anova)

if p_anova < 0.05:
    print("Conclusion: Average duration varies across genres.")
else:
    print("Conclusion: No significant duration difference among genres.")
```

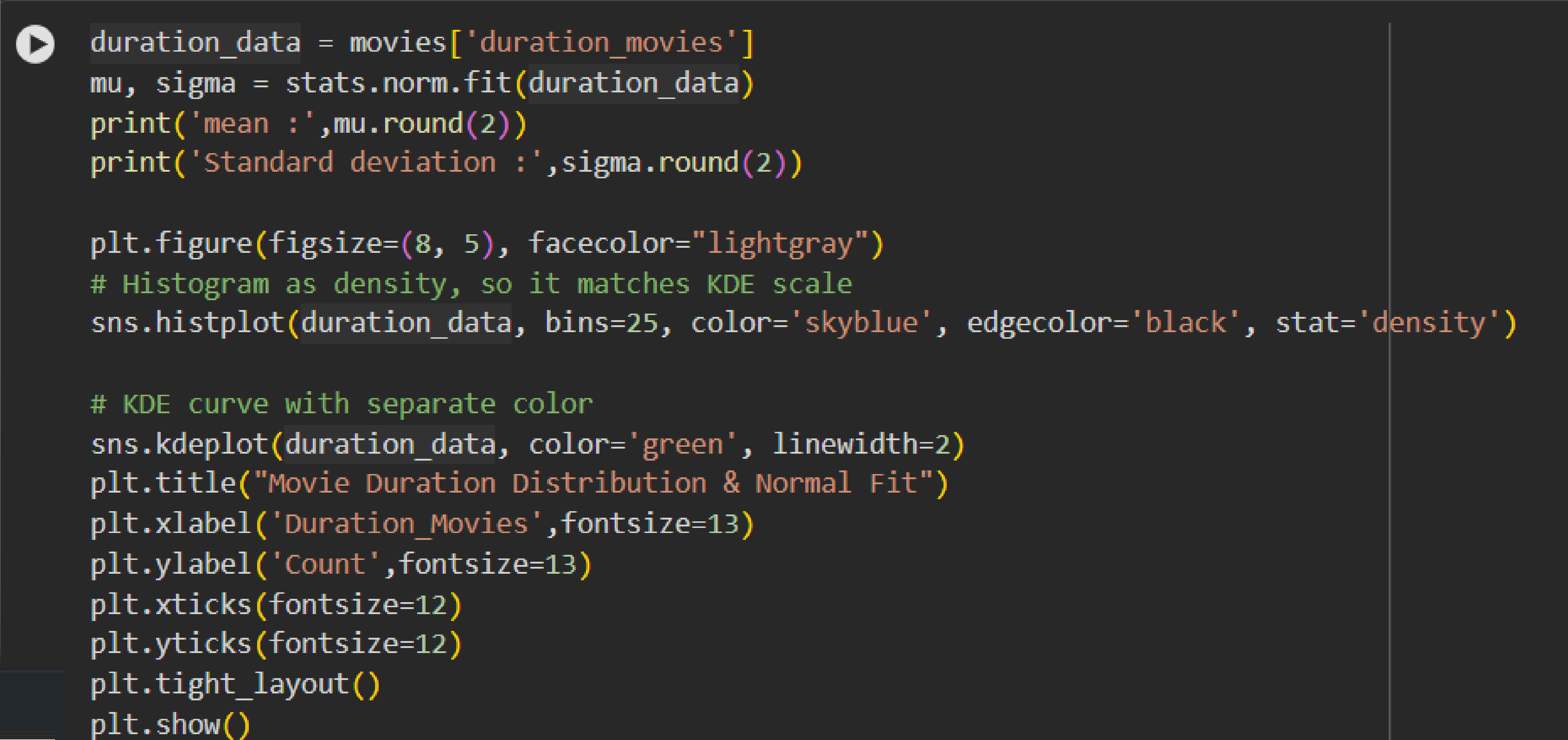
```
ANOVA F-statistic: 19.079
p-value: 5.432235229797162e-09
Conclusion: Average duration varies across genres.
```

<---

ANOVA

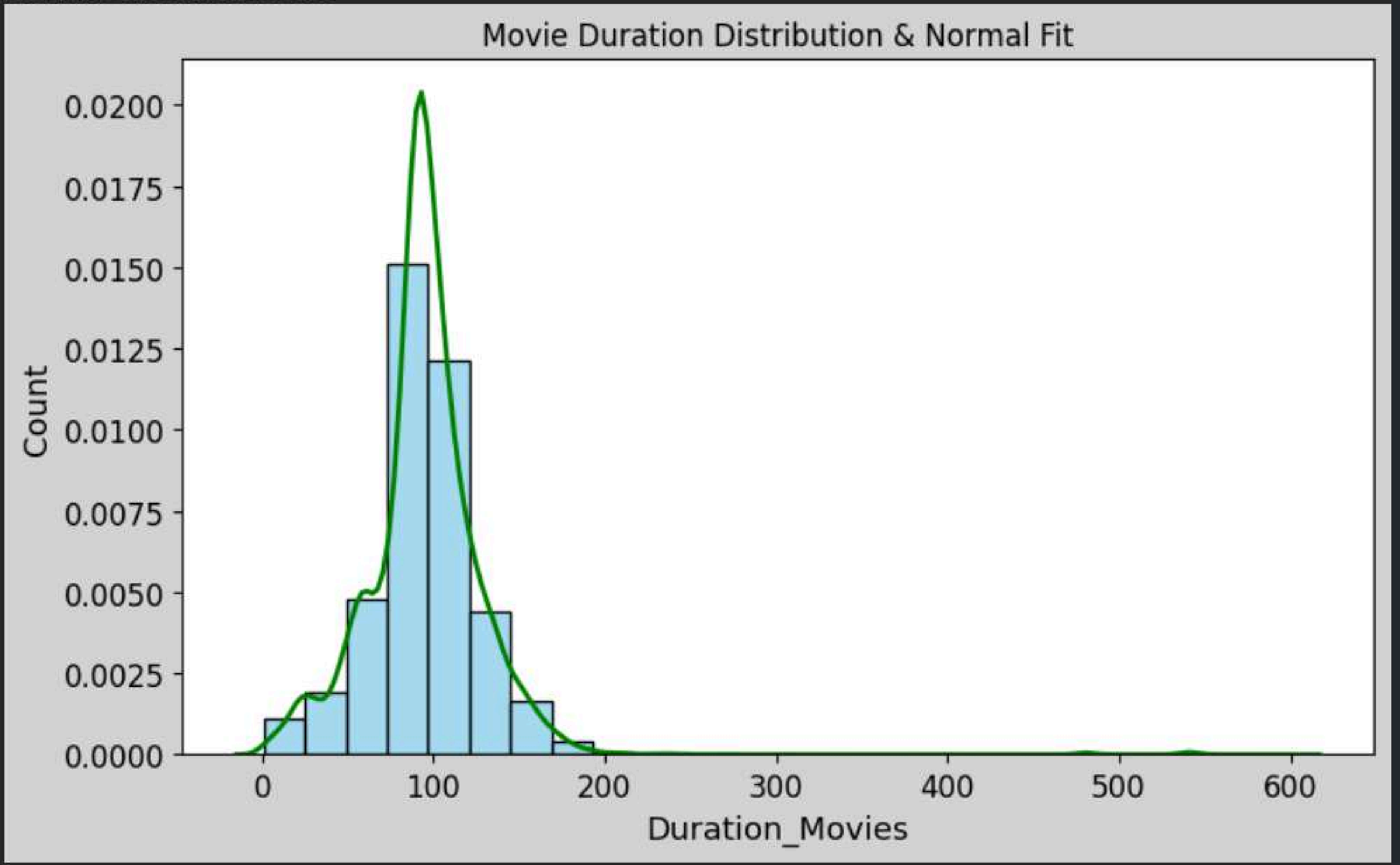
NORMAL DISTRIBUTION

---->



```python
duration_data = movies['duration_movies']
mu, sigma = stats.norm.fit(duration_data)
print('mean :',mu.round(2))
print('Standard deviation :',sigma.round(2))

plt.figure(figsize=(8, 5), facecolor="lightgray")
# Histogram as density, so it matches KDE scale
sns.histplot(duration_data, bins=25, color='skyblue', edgecolor='black', stat='density')

# KDE curve with separate color
sns.kdeplot(duration_data, color='green', linewidth=2)
plt.title("Movie Duration Distribution & Normal Fit")
plt.xlabel('Duration_Movies',fontsize=13)
plt.ylabel('Count',fontsize=13)
plt.xticks(fontsize=12)
plt.yticks(fontsize=12)
plt.tight_layout()
plt.show()
```

```
mean : 95.02
Standard deviation : 35.7
```



<---

VISUAL

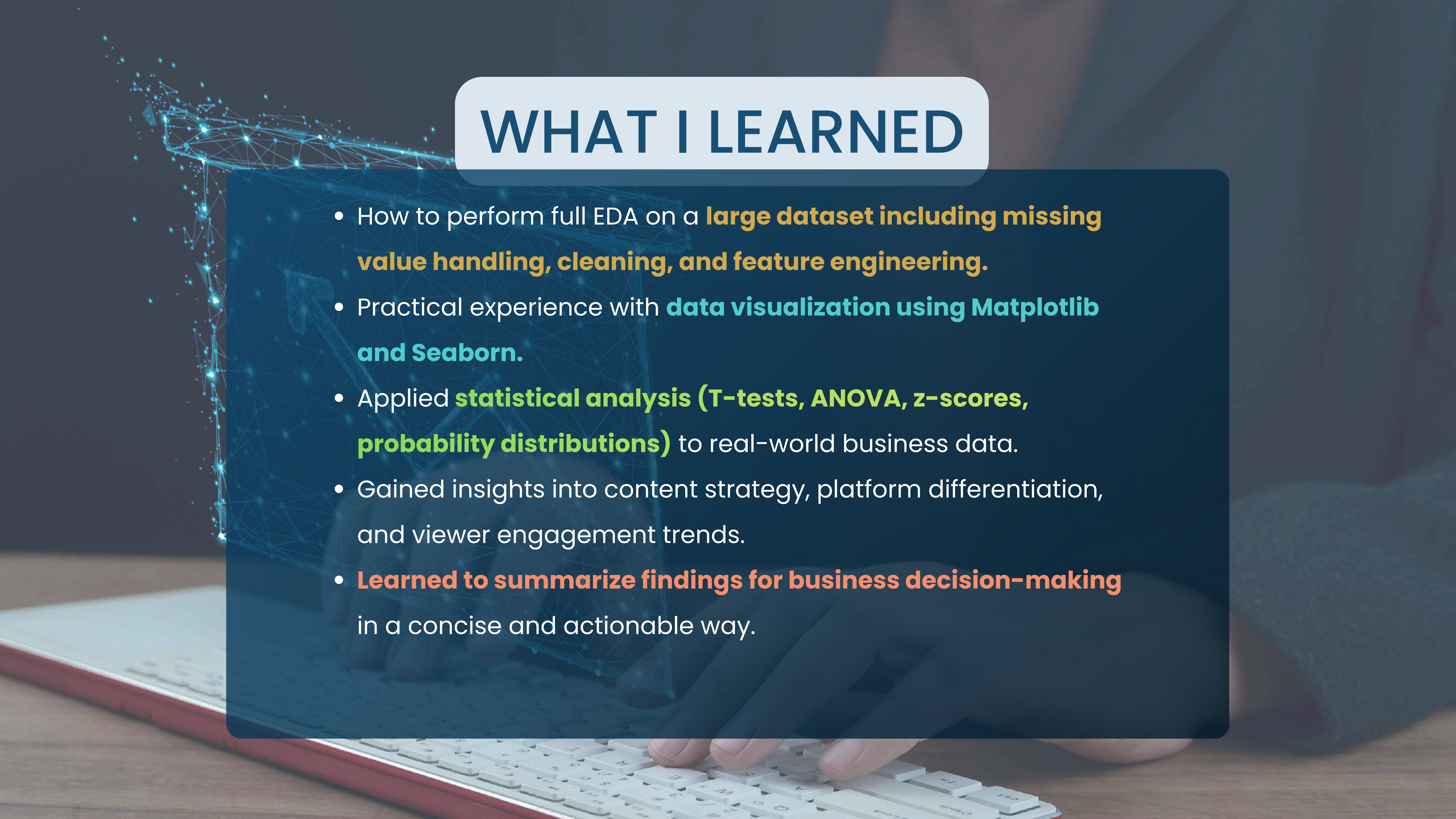
SUMMARY INSIGHTS

- ***Content Type:*** Movies dominate **(75%)**, Netflix favors TV shows, Prime Video favors movies.
- ***Platform Size:*** Prime Video has slightly more content overall.
- ***Genres:*** Drama, Comedy, Action, Documentaries, and Action & Adventure are top genres.
- ***Duration:*** Netflix movies are longer (avg. 99 mins) than Prime Video (avg. 91 mins).
- ***Country Contribution:*** Most content comes from U.S., followed by India, UK, and Japan.
- ***Time Trends:*** Content addition peaked between **2018–2021.**
- ***Statistical Findings:***

1. Significant difference in movie durations across platforms (T-test).

1. *Genre affects average movie duration* (ANOVA).

2. Extreme outliers exist for very long movies.



WHAT I LEARNED

- How to perform full EDA on a **large dataset including missing value handling, cleaning, and feature engineering.**
- Practical experience with **data visualization using Matplotlib and Seaborn.**
- Applied **statistical analysis (T-tests, ANOVA, z-scores, probability distributions)** to real-world business data.
- Gained insights into content strategy, platform differentiation, and viewer engagement trends.
- **Learned to summarize findings for business decision-making** in a concise and actionable way.

Thank You.