

File Edit Selection View Go Run Terminal Help ← → ⚡ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > Read the csv file 'Disney Plus Titles'

+ Code + Markdown | Run All Restart Clear All Outputs Variables Outline ... Python 3.7.2

import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt

[1] ✓ 0.9s Python

Read the csv file 'Disney Plus Titles'

df = pd.read\_csv('disney\_plus\_titles.csv')  
data1=df  
df.head()

[50] ✓ 0.0s Python

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	s1	Movie	Duck the Halls: A Mickey Mouse Christmas Special	Alonso Ramirez Ramos, Dave Wasson	Chris Diamantopoulos, Tony Anselmo, Tress MacN...	NaN	November 26, 2021	2016	TV-G	23 min	Animation, Family	Join Mickey and the gang as they duck the halls!
1	s2	Movie	Ernest Saves Christmas	John Cherry	Jim Varney, Noelle Parker, Douglas Seale	NaN	November 26, 2021	1988	PG	91 min	Comedy	Santa Claus passes his magic bag to a new St. ...
2	s3	Movie	Ice Age: A Mammoth Christmas	Karen Fisher	Raymond Albert Romano, John Leguizamo, Denis L...	United States	November 26, 2021	2011	TV-G	23 min	Animation, Comedy, Family	Sid the Sloth is on Santa's naughty list.
3	s4	Movie	The Queen Family Singalong	Hamish Hamilton	Darren Criss, Adam Lambert, Derek Hough, Alexa...	NaN	November 26, 2021	2021	TV-PG	41 min	Musical	This is real life, not just fantasy!
4	s5	TV Show	The Beatles: Get Back	NaN	John Lennon, Paul McCartney, George Harrison, ...	NaN	November 25, 2021	2021	NaN	1 Season	Docuseries, Historical, Music	A three-part documentary from Peter Jackson ca...

5 8 0 Type here to search 99+ zoom P Microsoft Edge Microsoft Teams 32°C ENG 14:49 07-07-2024 9

File Edit Selection View Go Run Terminal Help ← → ⚙ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > Read the csv file 'Disney Plus Titles'

+ Code + Markdown | Run All Restart Clear All Outputs Variables Outline ... Python 3.7.2

## Check shape of Dataframe

```
[3] df.shape ✓ 0.0s Python
... (1450, 12)
```

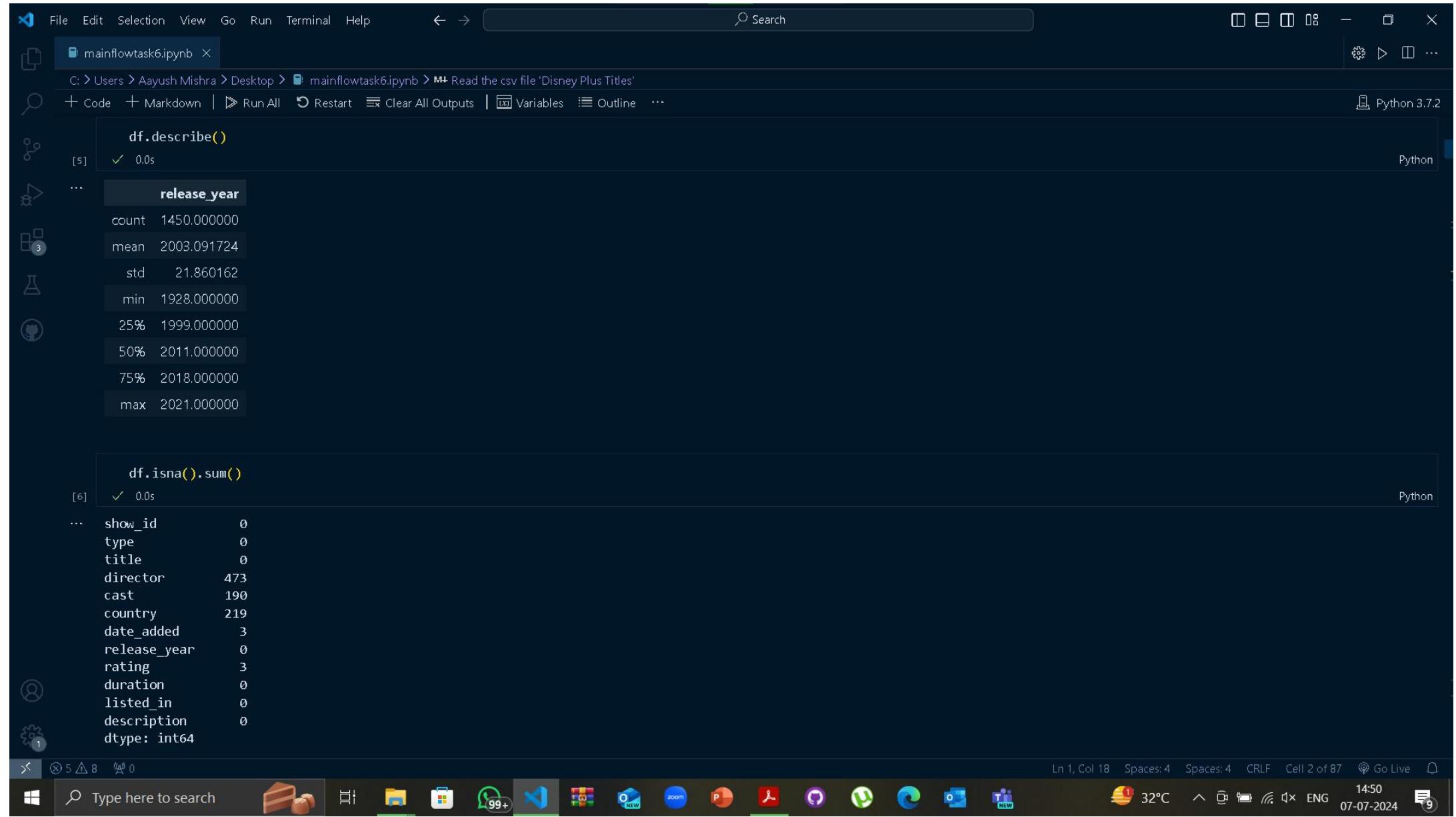
```
[4] df.info() ✓ 0.0s Python
... <class 'pandas.core.frame.DataFrame'>
RangeIndex: 1450 entries, 0 to 1449
Data columns (total 12 columns):
 #   Column      Non-Null Count  Dtype  
--- 
 0   show_id     1450 non-null   object 
 1   type        1450 non-null   object 
 2   title       1450 non-null   object 
 3   director    977 non-null   object 
 4   cast        1260 non-null   object 
 5   country     1231 non-null   object 
 6   date_added  1447 non-null   object 
 7   release_year 1450 non-null   int64  
 8   rating      1447 non-null   object 
 9   duration    1450 non-null   object 
 10  listed_in   1450 non-null   object 
 11  description 1450 non-null   object 
dtypes: int64(1), object(11)
memory usage: 136.1+ KB
```

+ Code + Markdown

Ln 1, Col 18 Spaces: 4 Spaces: 4 CRLF Cell 2 of 87 Go Live

Type here to search

14:49 32°C ENG 07-07-2024



File Edit Selection View Go Run Terminal Help ⏪ ⏩ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > Read the csv file 'Disney Plus Titles'

+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ... Python 3.7.2

df.drop\_duplicates(inplace=True)  
df.shape  
[7] ✓ 0.0s Python

... (1450, 12)

Fill NULL Values

df['director'].fillna('None', inplace=True)  
df['cast'].fillna('None', inplace=True)  
df['country'].fillna('None', inplace=True)  
[8] ✓ 0.0s Python

df.dropna(inplace=True)  
[9] ✓ 0.0s Python

df.isnull().sum()  
[10] ✓ 0.0s Python

show_id	0
type	0
title	0
director	0
cast	0
country	0
date_added	0
release_year	0

Ln 1, Col 18 Spaces: 4 Spaces: 4 CRLF Cell 2 of 87 Go Live

Type here to search

Windows Taskbar icons: File Explorer, Edge, Microsoft Teams, OneDrive, Zoom, Microsoft Word, Microsoft Excel, Microsoft Powerpoint, Microsoft Word, Microsoft Excel, Microsoft Word, Microsoft Word, Microsoft Word, Microsoft Word.

System tray: Battery (14%), 32°C, Wi-Fi, ENG, 14:50, 07-07-2024, Notifications (9).

File Edit Selection View Go Run Terminal Help ← → ⚡ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > Read the csv file 'Disney Plus Titles'

+ Code + Markdown | Run All Restart Clear All Outputs Variables Outline ... Python 3.7.2

## Break into Year ,Month and Day

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

[12] ✓ 0.0s Python

```
[13] ... show_id          object
      type            object
      title           object
      director        object
      cast             object
      country          object
      date_added      datetime64[ns]
      release_year    int64
      rating           object
      duration          object
      listed_in        object
      description       object
      dtype: object
```

```
[13] # Extract year and month for trend analysis
      df['year_month'] = df['date_added'].dt.to_period('M')
```

[13] ✓ 0.0s Python

```
[14] # Aggregate the number of titles added each month
      monthly_titles = df.groupby('year_month').size()
```

[14] ✓ 0.0s Python

x 5 8 0 Type here to search

Ln 2, Col 30 Spaces: 4 Spaces: 4 CRLF Cell 2 of 87 Go Live

14:50 07-07-2024

File Edit Selection View Go Run Terminal Help ← → ⚡ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > Read the csv file 'Disney Plus Titles'

+ Code + Markdown | Run All Restart Clear All Outputs Variables Outline ... Python 3.7.2

[15] ✓ 0.0s Python

```
data = df
```

[16] ✓ 1.3s Python

```
import nltk
from nltk.sentiment.vader import SentimentIntensityAnalyzer
nltk.download('vader_lexicon')
# Initialize the sentiment analyzer
sid = SentimentIntensityAnalyzer()
# Function to calculate sentiment scores
def get_sentiment(description):
    if isinstance(description, str):
        scores = sid.polarity_scores(description)
        return scores['compound']
    else:
        return None
df['sentiment'] = df['description'].apply(get_sentiment)
df[['title', 'description', 'sentiment']].head()
```

[nltk\_data] Downloading package vader\_lexicon to c:\users\Aayush  
[nltk\_data] Mishra\AppData\Roaming\nltk\_data...  
[nltk\_data] Package vader\_lexicon is already up-to-date!

	title	description	sentiment
0	Duck the Halls: A Mickey Mouse Christmas Special	Join Mickey and the gang as they duck the halls!	0.3595
1	Ernest Saves Christmas	Santa Claus passes his magic bag to a new St. ...	0.0000
2	Ice Age: A Mammoth Christmas	Sid the Sloth is on Santa's naughty list.	0.0000
3	The Queen Family Singalong	This is real life, not just fantasy!	0.0000
5	Becoming Cousteau	An inside look at the legendary life of advent...	0.2960

Ln 2, Col 30 Spaces: 4 Spaces: 4 CRLF Cell 2 of 87 Go Live

Type here to search

14:50 07-07-2024

File Edit Selection View Go Run Terminal Help ← → ⌂ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > Read the csv file 'Disney Plus Titles'

+ Code + Markdown | Run All Restart Clear All Outputs Variables Outline ... Python 3.7.2

# Name of the Columns present in the Dataframe

```
df.columns
[17] ✓ 0.0s Python
... Index(['show_id', 'type', 'title', 'director', 'cast', 'country', 'date_added',
       'release_year', 'rating', 'duration', 'listed_in', 'description',
       'year_month', 'sentiment'],
       dtype='object')
```

[18] ✓ 0.0s Python

```
df = pd.DataFrame(df['release_year'])
df
```

	release_year
0	2016
1	1988
2	2011
3	2021
5	2021
...	...
1445	2009
1446	2009
1447	2016
1448	2003
1449	2012

Ln 1, Col 11 Spaces: 4 Spaces: 4 CRLF Cell 2 of 87 Go Live

Type here to search

5 8 0

14:51 07-07-2024

File Edit Selection View Go Run Terminal Help ← → ⌂ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > Read the csv file 'Disney Plus Titles'

+ Code + Markdown | Run All Restart Clear All Outputs Variables Outline ... Python 3.7.2

[19] ✓ 0.2s Python

```
df['release_year'].plot(xlabel='No.of shows released', ylabel='Year of release')
plt.title('No.of shows released each year')
plt.show()
```

No.of shows released each year

!pip install statsmodels

[20] ✓ 1.5s Python

Type here to search

Ln 1, Col 25 Spaces: 4 Spaces: 4 CRLF Cell 2 of 87 Go Live

32°C 14:51 07-07-2024

File Edit Selection View Go Run Terminal Help ← → ⌂ Search

mainflowtask6.ipynb X

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > Read the csv file 'Disney Plus Titles'

+ Code + Markdown | Run All Restart Clear All Outputs Variables Outline ...

[21] ✓ 0.0s Python 3.7.2 Python

release\_cycle

[22] ✓ 0.0s Python

```
... 0      4.822442
    1     -23.774797
    2     -1.376055
    3      8.034462
    5      7.473694
    ...
1445    8.808760
1446    6.885097
1447   11.946330
1448   -2.996700
1449    4.056888
Name: release_year_cycle, Length: 1444, dtype: float64
```

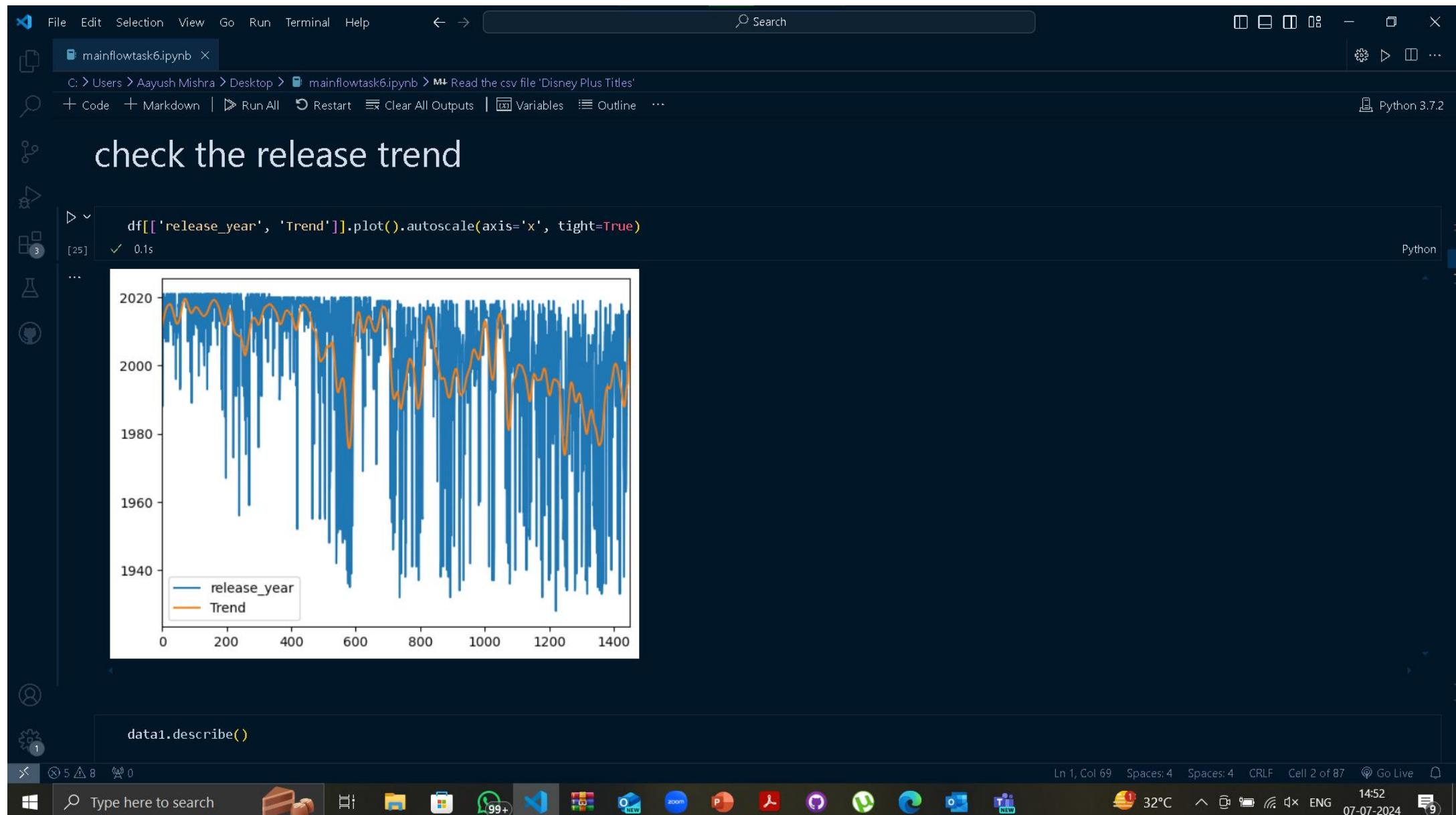
release\_trend

[23] ✓ 0.0s Python

```
... 0      2011.177558
    1     2011.774797
    2     2012.376055
    3     2012.965538
    5     2013.526306
    ...
1445   2000.191240
1446   2002.114903
1447   2004.053670
1448   2005.996700
1449   2007.943112
Name: release_year_trend, Length: 1444, dtype: float64
```

Ln 3, Col 17 Spaces: 4 Spaces: 4 CRLF Cell 2 of 87 Go Live

X 5 8 0 Type here to search 🍰 ⌂ 99+ VS Code zoom P Microsoft Edge Microsoft Teams 32°C ENG 14:51 07-07-2024



File Edit Selection View Go Run Terminal Help ← → ⚡ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > Read the csv file 'Disney Plus Titles'

+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ... Python 3.7.2

[51] ✓ 0.0s Python

```
data1.describe()
```

... release\_year

	release_year
count	1450.000000
mean	2003.091724
std	21.860162
min	1928.000000
25%	1999.000000
50%	2011.000000
75%	2018.000000
max	2021.000000

[52] ✓ 0.0s Python

```
data1.isnull().sum()
```

... show\_id type title director cast country date\_added release\_year rating duration listed\_in description dtype: int64

x 5 8 0 Type here to search

Ln 1, Col 69 Spaces: 4 Spaces: 4 CRLF Cell 2 of 87 Go Live

32°C 14:52 07-07-2024

File Edit Selection View Go Run Terminal Help ⏪ ⏩ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > Read the csv file 'Disney Plus Titles'

+ Code + Markdown | Run All Restart Clear All Outputs Variables Outline ... Python 3.7.2

import missingno as msno

msno.matrix(data1)  
plt.show()

[53] ✓ 0.3s Python

show\_id type title director cast country date\_added release\_year rating duration listed\_in description

1

Ln 1, Col 22 Spaces: 4 Spaces: 4 CRLF Cell 2 of 87 Go Live

Type here to search

14:52 07-07-2024

File Edit Selection View Go Run Terminal Help ← → ⌂ Search

mainflowtask6.ipynb X C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > Read the csv file 'Disney Plus Titles'

+ Code + Markdown | Run All Restart Clear All Outputs Variables Outline ... Python 3.7.2

## let's check the % of null values

```
[54] ✓ 0.0s Python
data1.isnull().sum()/len(data1)*100
```

... show_id	0.000000
type	0.000000
title	0.000000
director	32.620690
cast	13.103448
country	15.103448
date_added	0.206897
release_year	0.000000
rating	0.206897
duration	0.000000
listed_in	0.000000
description	0.000000
dtype: float64	

## Let's remove the NAN values

```
[55] ✓ 0.0s Python
data1['director'].fillna('None', inplace=True)
data1['cast'].fillna('None', inplace=True)
data1['country'].fillna('None', inplace=True)
```

Ln 2, Col 27 Spaces: 4 Spaces: 4 CRLF Cell 2 of 87 Go Live

Type here to search

14:52 07-07-2024

File Edit Selection View Go Run Terminal Help ← → ⚡ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > Read the csv file 'Disney Plus Titles'

+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ... Python 3.7.2

```
data1.duplicated().sum()
```

[64] ✓ 0.0s Python

... 0

```
#(date_added) to (data_time)
data1['date_added'] = pd.to_datetime(data1['date_added'], errors="coerce")
print(f'a. 'date_added' column is in{data1['date_added'].dtype} format")
print(f'a. 'release_year' column is in{data1['release_year'].dtype} format")
```

[65] ✓ 0.0s Python

... a. 'date\_added' column is indatetime64[ns] format
a. 'release\_year' column is inint64format

```
data1.head()
```

[58] ✓ 0.0s Python

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	s1	Movie	Duck the Halls: A Mickey Mouse Christmas Special	Alonso Ramirez Ramos, Dave Wasson	Chris Diamantopoulos, Tony Anselmo, Tress MacN...	None	November 26, 2021	2016	TV-G	23 min	Animation, Family	Join Mickey and the gang as they duck the halls!
1	s2	Movie	Ernest Saves Christmas	John Cherry	Jim Varney, Noelle Parker, Douglas Seale	None	November 26, 2021	1988	PG	91 min	Comedy	Santa Claus passes his magic bag to a new St ...
2	s3	Movie	Ice Age: A Mammoth Christmas	Karen Disher	Raymond Albert Romano, John Leguizamo, Denis L...	United States	November 26, 2021	2011	TV-G	23 min	Animation, Comedy, Family	Sid the Sloth is on Santa's naughty list.

✖ 5 ▲ 8 ❌ 0 Ln 2, Col 1 Spaces: 4 Spaces: 4 CRLF Cell 2 of 87 ⚡ Go Live

Type here to search 🍰 📁 99+ 💬 NEW zoom P 💬 NEW Microsoft Edge Microsoft Teams 📈 32°C 🔍 ENG 14:53 07-07-2024 📅 9

File Edit Selection View Go Run Terminal Help ← → ⌂ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > Read the csv file 'Disney Plus Titles'

+ Code + Markdown | Run All Restart Clear All Outputs Variables Outline ... Python 3.7.2

# Let's count Movies and TV shows

```
[40] ✓ 0.0s Python
data1['type'].value_counts().reset_index()
```

index	type	
0	Movie	1051
1	TV Show	393

```
[41] ✓ 0.4s Python
#Visualisation
fig = px.pie(data1, values=data1["type"].value_counts().values, names=data1["type"].value_counts().index)

fig.update_layout(title = "Content Types Over the Years", plot_bgcolor="rgba(255,255,255,0.5)",paper_bgcolor='grey',font=dict(color='white'))
fig.show()
```

Content Types Over the Years

Movie  
TV Show

27.2%

Ln 1, Col 6 Spaces: 4 Spaces: 4 CRLF Cell 2 of 87 Go Live

Type here to search

32°C 14:53 07-07-2024

### Content Types Over the Years



File Edit Selection View Go Run Terminal Help ⏪ ⏩ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > ...

+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ... Python 3.7.2

```
fig = px.bar(x=data1["type"].value_counts().index,
              y=data1["type"].value_counts().values,
              color=data1["type"].value_counts().index)
## fig.update_layout(
#     title={
#         'text': "Content Types Over the Years",
#         'font': {'color': 'yellow'}
#     },
#
#     xaxis_title={
#         'text': "Types of Content",
#         'font': {'color': 'yellow'}
#     },
#
#     yaxis_title={
#         'text': "Counts",
#         'font': {'color': 'yellow'}
#     },
#
#     plot_bgcolor='rgba(0,0,0,0)',
#     paper_bgcolor='grey',
#     font=dict(color='white')
# )
#
# fig.show()
```

[42] ✓ 0.0s Python

...

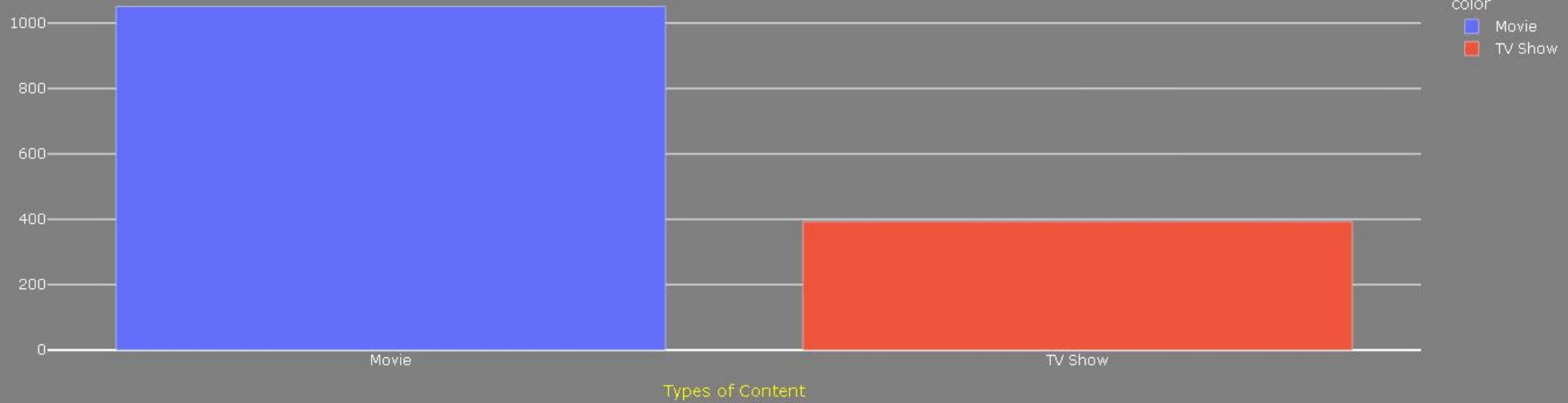
Content Types Over the Years

Spaces: 4 CRLF Cell 50 of 87 Go Live Prettier

x 5 8 0 Type here to search

14:54 07-07-2024

### Content Types Over the Years



File Edit Selection View Go Run Terminal Help ← → ⌂ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > ...

+ Code + Markdown | Run All Restart Clear All Outputs Variables Outline ... Python 3.7.2

## Let's see the release years of movies and shows

```
data1['release_year'].unique()  
[43] ✓ 0.0s Python  
array([2016, 1988, 2011, 2021, 2015, 2019, 2008, 2020, 2007, 2013, 2018,  
       2014, 2012, 2006, 2010, 1996, 2009, 2017, 1993, 1994, 1998, 1989,  
       1997, 2005, 2000, 2004, 1987, 1985, 1967, 1973, 1991, 1956, 1995,  
       1984, 1974, 1959, 2003, 1976, 2001, 1990, 1992, 1952, 1955, 1977,  
       1957, 1999, 1948, 1964, 1969, 1942, 1950, 1951, 1953, 1949, 1940,  
       1946, 1954, 1936, 1944, 1935, 1939, 1975, 1978, 2002, 1971, 1961,  
       1962, 1981, 1932, 1938, 1941, 1986, 1947, 1937, 1966, 1943, 1934,  
       1980, 1960, 1983, 1972, 1982, 1979, 1928, 1965, 1970, 1963, 1933,  
       1945, 1968], dtype=int64)
```

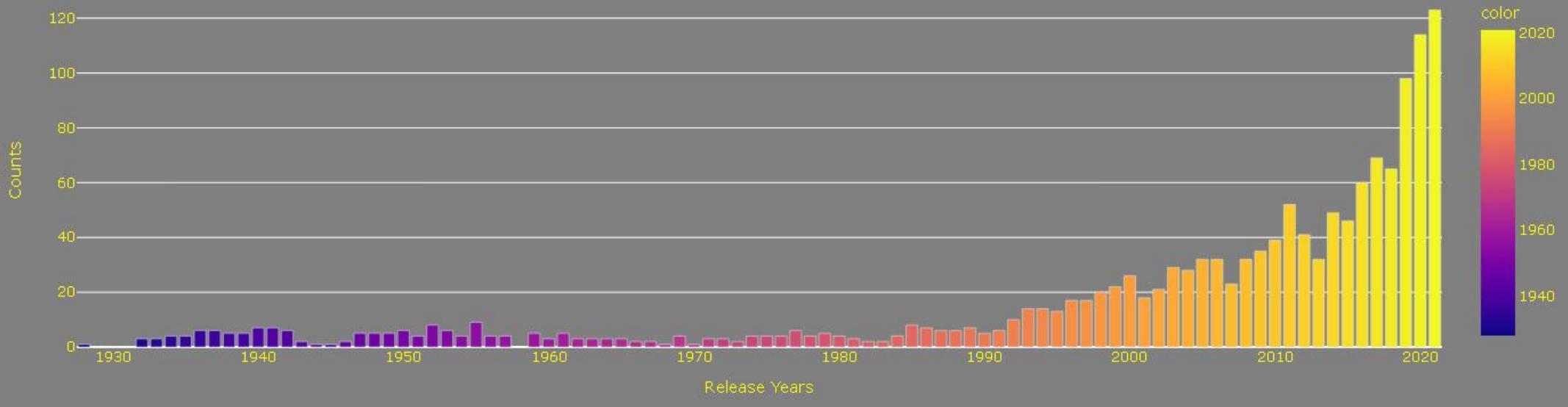
```
#Visualisation  
fig = px.bar(x=data1["release_year"].value_counts().index,  
              y=data1["release_year"].value_counts().values,  
              color=data1["release_year"].value_counts().index)  
##  
fig.update_layout(  
    title={  
        'text': "Most Release Years for Movies and TV Shows",  
        'font': {'color': 'yellow'}  
    },  
  
    xaxis_title={  
        'text': "Release Years",  
        'font': {'color': 'yellow'}  
    },  
  
    yaxis_title=
```

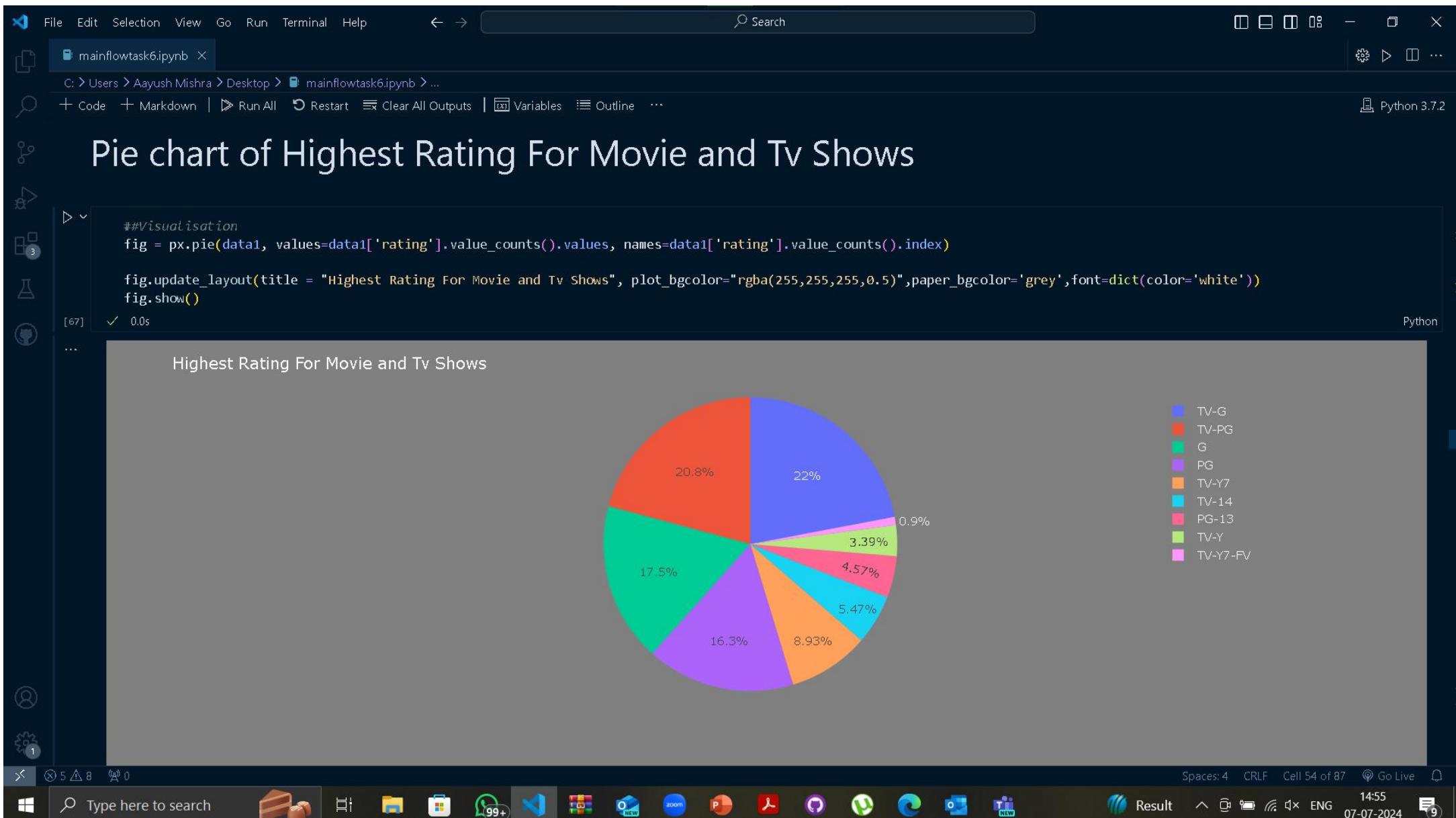
Spaces: 4 CRLF Cell 51 of 87 Go Live

Type here to search

14:55 07-07-2024

### Most Release Years for Movies and TV Shows





File Edit Selection View Go Run Terminal Help ⏪ ⏩ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > ...

+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ... Python 3.7.2

```
fig = px.bar(x=data1["rating"].value_counts().index,
              y=data1["rating"].value_counts().values,
              color=data1["rating"].value_counts().index)
## fig.update_layout(
#     title={
#         'text': "Highest Rating For Movie and Tv Shows",
#         'font': {'color': 'yellow'}
#     },
#     xaxis_title={
#         'text': "Highest Rating",
#         'font': {'color': 'yellow'}
#     },
#     yaxis_title={
#         'text': "Counts",
#         'font': {'color': 'yellow'}
#     },
#     plot_bgcolor='rgba(0,0,0,0)',
#     paper_bgcolor='grey',
#     font=dict(color='white')
# )
fig.show()
```

[47] ✓ 0.1s Python

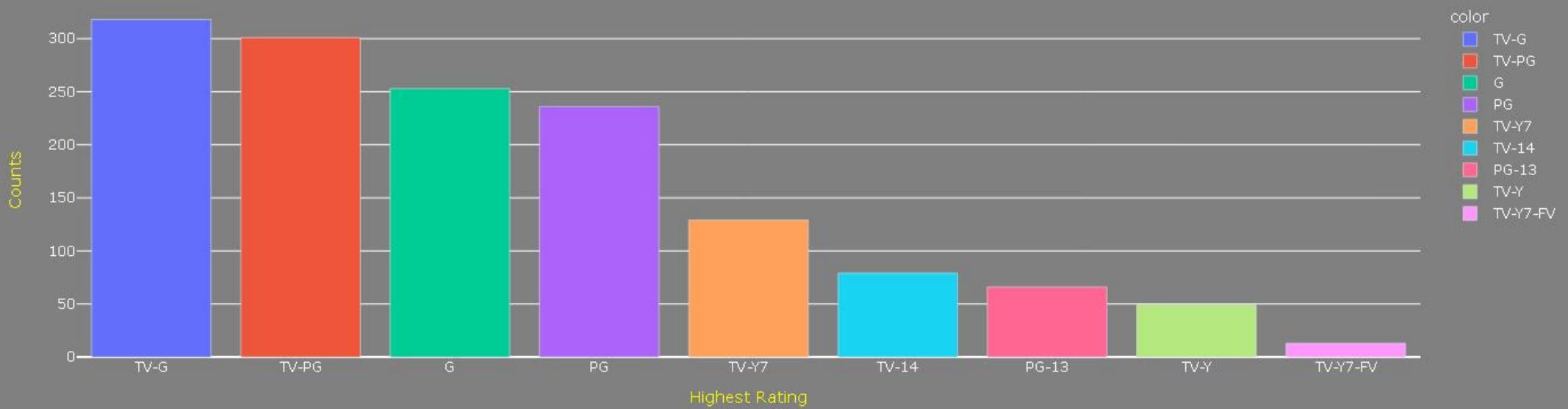
Highest Rating For Movie and Tv Shows

Spaces: 4 CRLF Cell 54 of 87 Go Live

Type here to search

14:55 07-07-2024

### Highest Rating For Movie and Tv Shows



File Edit Selection View Go Run Terminal Help ← → ⌂ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > ...

+ Code + Markdown | Run All Restart Clear All Outputs Variables Outline ... Python 3.7.2

## Let's see the numbers of directors

```
directors = data1.groupby(['director', 'type'])['director'].value_counts()
directors
```

[49] ✓ 0.0s Python

```
... director type director
Aaron Blaise, Robert Walker Movie Aaron Blaise, Robert Walker 1
Adam Shankman Movie Adam Shankman 3
Adam Stein, Zach Lipovsky Movie Adam Stein, Zach Lipovsky 1
Alan Barillaro Movie Alan Barillaro 1
Alan Shapiro Movie Alan Shapiro 1
...
Zach A. Parrish Movie Zach A. Parrish 1
Zak Knutson Movie Zak Knutson 1
Zhong Yu Movie Zhong Yu 1
Érik Canuel Movie Érik Canuel 1
朱家欣 , 钟智行 Movie 朱家欣 , 钟智行 1
Name: director, Length: 611, dtype: int64
```

```
#Let's see the top 10 directors
data1[['director', 'type']].groupby('director')['type'].count().nlargest(10)
```

[ ] Python

```
... director
None 468
Jack Hannah 17
John Lasseter 16
Paul Hoen 16
Charles Nichols 12
Robert Schwartz 12
```

Spaces: 4 CRLF Cell 58 of 87 Go Live

Type here to search

Windows taskbar icons: File Explorer, Edge, Microsoft Store, Zoom, Microsoft Teams, Microsoft Word, Microsoft Excel, Microsoft Powerpoint, Microsoft Word, Microsoft Word, Microsoft Word, Microsoft Word, Microsoft Word, Microsoft Word.

System tray: 32°C, ENG, 14:56, 07-07-2024, battery icon.

File Edit Selection View Go Run Terminal Help ← → ⌂ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > ...

+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ... Python 3.7.2

```
##Visualisation
import seaborn as sns
plt.figure(figsize=(12, 6))
top_directors = df["director"].value_counts().head(10)
sns.barplot(x=top_directors.values, y=top_directors.index, palette='Paired')
plt.title('Top 10 Directors with Best Content')
plt.xlabel('Number of Titles')
plt.ylabel('Director')
plt.gca().set_facecolor('grey')
plt.show()
```

[ ] Python

Top 10 Directors with Best Content

Director	Number of Titles
None	~100
Jack Hannah	~5
John Lasseter	~5
Paul Hoen	~5
Charles Nichols	~5
Robert Stevenson	~5
Bob Peterson	~5
Vincent McEveety	~5
James Algar	~5

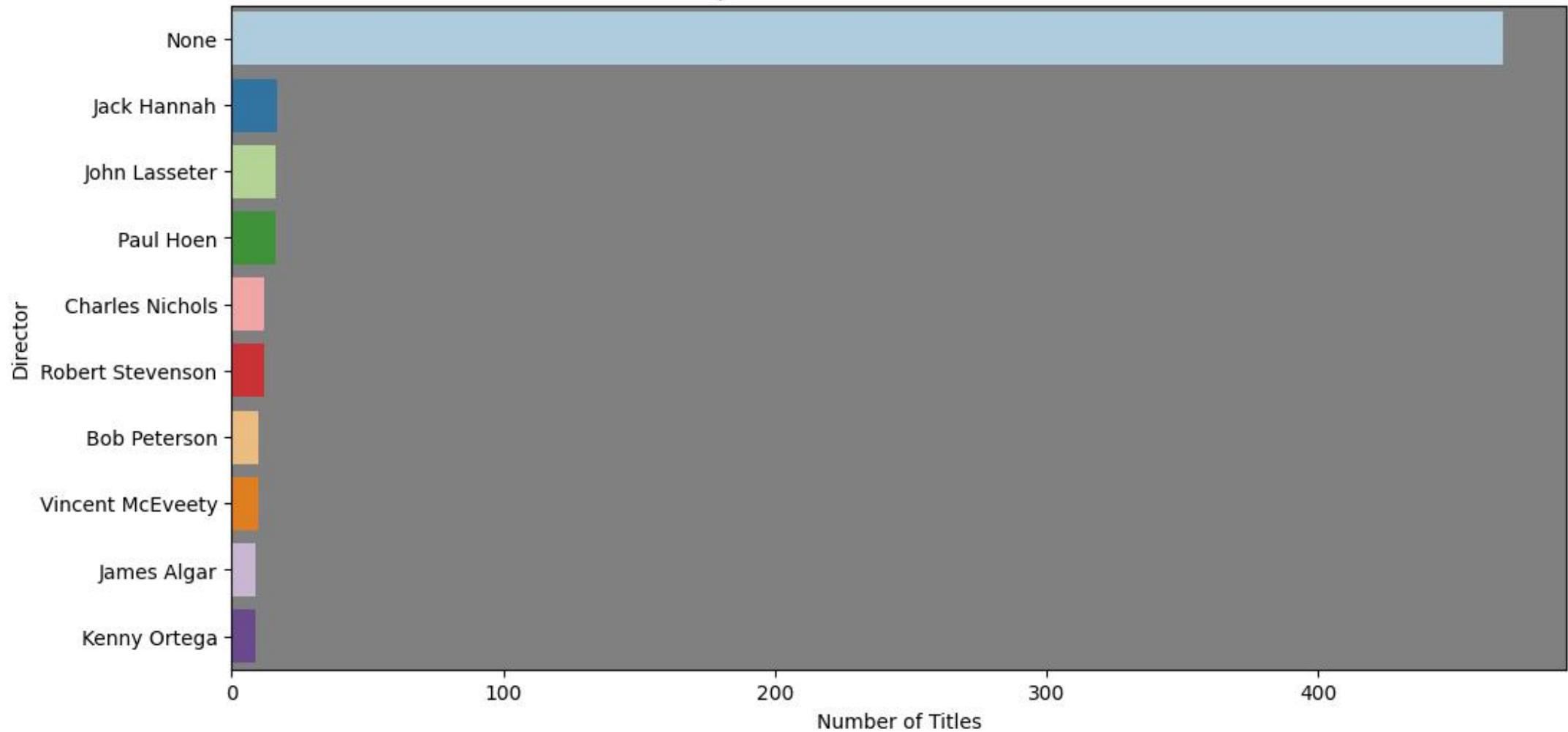
Ln 2, Col 22 Spaces: 4 Spaces: 4 CRLF Cell 58 of 87 Go Live

Type here to search

Windows Taskbar icons: File Explorer, Edge, Microsoft Store, Zoom, OneDrive, Microsoft Teams, Microsoft Word, Microsoft Excel, Microsoft Powerpoint, Microsoft Word, Microsoft Excel, Microsoft Word, Microsoft Word.

System tray icons: Volume, Battery (14:56, 07-07-2024), Network, Weather (32°C), Language (ENG), Notifications.

Top 10 Directors with Best Content



File Edit Selection View Go Run Terminal Help ⌘ ⌘ ⌘ ⌘ ⌘ ⌘ ⌘ ⌘ ⌘

Search

mainflowtask6.ipynb X

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > ...

+ Code + Markdown | Run All Restart Clear All Outputs Variables Outline ... Python 3.7.2

## Let's see, which country produce's the most number of content on Disney

```
grouped_data = data1.groupby(['type', 'country']).size().reset_index(name='count')
grouped_data
```

	type	country	count
0	Movie	Australia, United States	3
1	Movie	Austria, South Africa	1
2	Movie	Austria, United Kingdom, United States	1
3	Movie	Canada	3
4	Movie	Canada, Malaysia, United States	1
...	...	...	...
95	TV Show	United States, South Korea, China, Taiwan	1
96	TV Show	United States, South Korea, France	1
97	TV Show	United States, Taiwan, South Korea, China, Jap...	1
98	TV Show	United States, United Kingdom	2
99	TV Show	United States, United Kingdom, South Korea	1

100 rows × 3 columns

```
grouped_data = data1.groupby(['type', 'country']).size().reset_index(name='count')
grouped_data = grouped_data.sort_values(by='count', ascending=False)
grouped_data = grouped_data.head(20)
```

Ln 2, Col 22 Spaces: 4 Spaces: 4 CRLF Cell 62 of 87 Go Live Prettier

Type here to search

14:57 07-07-2024

## Countris Produces Highest level of Content

File Edit Selection View Go Run Terminal Help ← → ⌂ Search

mainflowtask6.ipynb X C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > ...

+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ... Python 3.7.2

## let's see the duration of Movies and TV shows

```
grouped_data = data1.groupby(['type', 'duration']).size().reset_index(name='count')

grouped_data = grouped_data.sort_values(by='count', ascending=False)
grouped_data
```

	type	duration	count
145	TV Show	1 Season	217
149	TV Show	2 Seasons	81
150	TV Show	3 Seasons	43
112	Movie	7 min	42
85	Movie	44 min	42
...	...	...	...
61	Movie	175 min	1
98	Movie	56 min	1
59	Movie	169 min	1
58	Movie	162 min	1
60	Movie	170 min	1

158 rows × 3 columns

#Visualisation

5 8 0

Type here to search

Spaces: 4 CRLF Cell 65 of 87 Go Live Prettier

14:57 07-07-2024

File Edit Selection View Go Run Terminal Help ← → ⌂ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > ...

+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ...

Python 3.7.2

#Visualisation

```
fig = px.bar(grouped_data, x='duration', y='count', color='type',
              title="length of Movies and Tv Show",
              labels={'duration': "Duration", 'count': "Count"},
              color_discrete_map={'short' : 'red', 'long': 'blue'})
fig.update_layout(
    plot_bgcolor='rgba(0,0,0,0)',
    paper_bgcolor='grey',
    font=dict(color='yellow'))
```

length of Movies and Tv Show

type

TV Show

Movie

Count

Duration

Spaces: 4 CRLF Cell 65 of 87 Go Live

Type here to search

14:58 07-07-2024

File Edit Selection View Go Run Terminal Help ← → ⌂ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > ...

+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ...

Python 3.7.2

## let's see the cast

```
data1['cast'].value_counts().reset_index()
```

	index	cast
0	None	189
1	Winston Hibler	10
2	Dan Nachtrab	7
3	Larry the Cable Guy , Keith Ferguson	6
4	Russell Boulter	6
...	...	...
1184	Tony Goldwyn, Minnie Driver, Glenn Close, Alex...	1
1185	Robert Downey Jr., Chris Hemsworth, Mark Ruffa...	1
1186	Walt Disney, Paul Frees, Dr. Ernst Stuhlinger...	1
1187	Walt Disney, Dick Tufeld, Ward Kimball, Willy ...	1
1188	Charlie Tahan	1

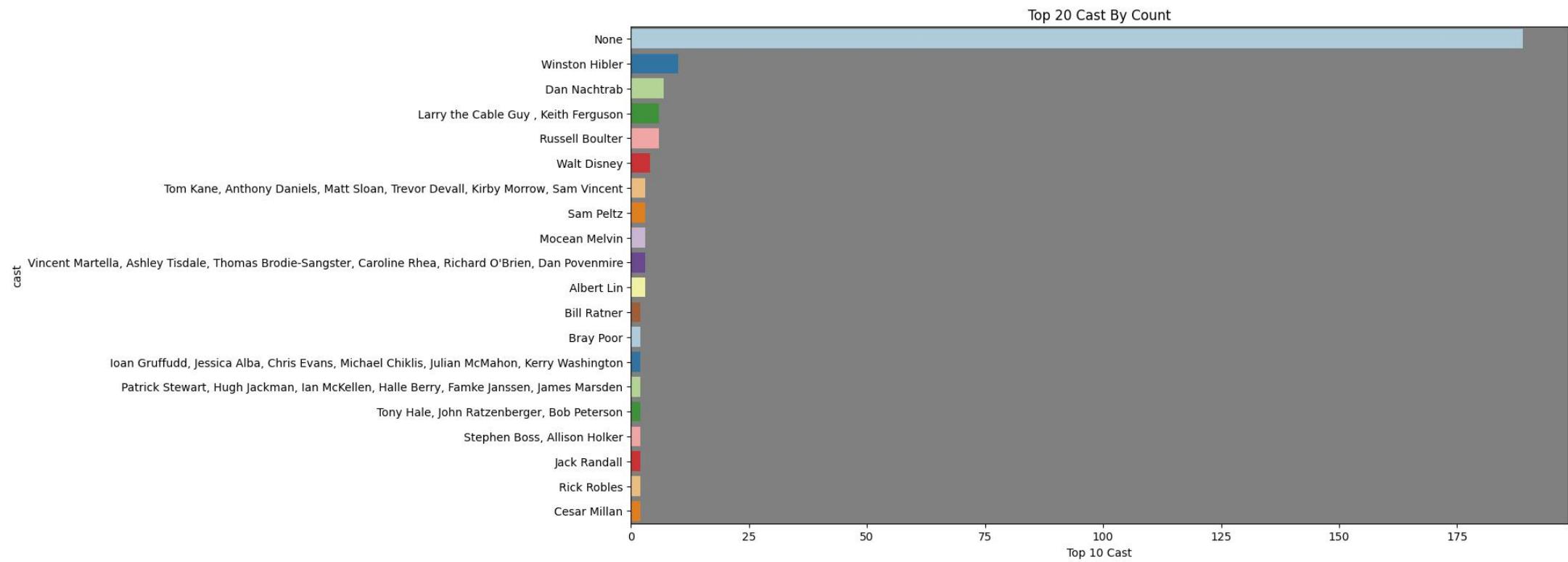
1189 rows × 2 columns

```
##Visualisation
plt.figure(figsize=(15, 8))
top_directors = data1["cast"].value_counts().head(20)
sns.barplot(x=top_directors.values, y=top_directors.index, palette='Paired')
```

Ln 3, Col 22 Spaces: 4 Spaces: 4 CRLF Cell 65 of 87 ⌂ Go Live

Type here to search

14:58 07-07-2024



File Edit Selection View Go Run Terminal Help ← → ⌂ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > ...

+ Code + Markdown | Run All Restart Clear All Outputs Variables Outline ... Python 3.7.2

## Let's see Listed in columns

```
data1['listed_in'].value_counts().reset_index()
```

	index	listed_in
0	Animation, Comedy, Family	124
1	Action-Adventure, Animation, Comedy	77
2	Action-Adventure, Animation, Kids	45
3	Animals & Nature, Documentary, Family	40
4	Action-Adventure, Animation, Family	40
...	...	...
323	Biographical, Coming of Age, Drama	1
324	Docuseries, Family, Lifestyle	1
325	Comedy, Fantasy, Romance	1
326	Buddy, Comedy, Drama	1
327	Action-Adventure, Animals & Nature, Animation	1

328 rows × 2 columns

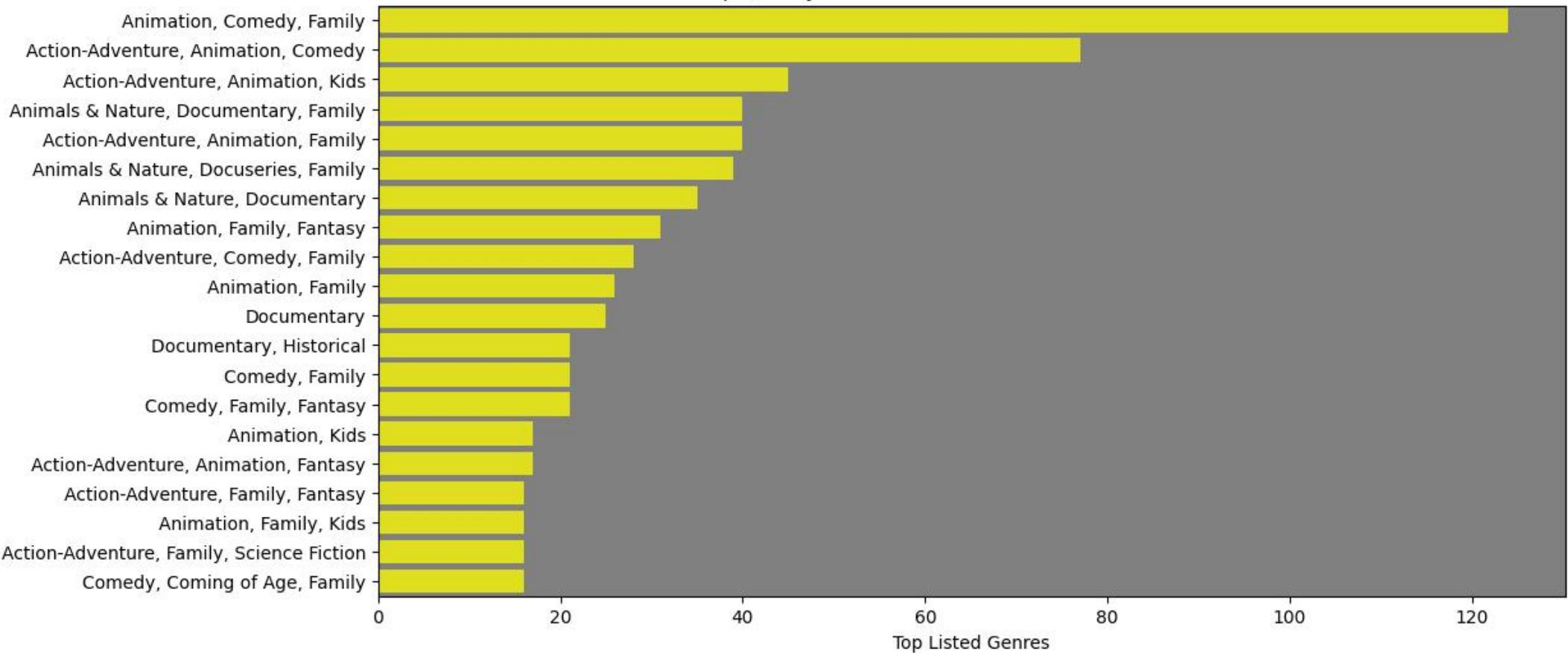
```
#Visualisation
plt.figure(figsize=(12,6))
top_genres = data1['listed_in'].value_counts().head(20)
sns.barplot(x=top_genres.values, y=top_genres.index, color='yellow')
plt.title('Top Quality Listed Genres for Movies and TV Shows')
```

Ln 3, Col 22 Spaces: 4 CRLF Cell 71 of 87 Go Live

Type here to search

14:58 07-07-2024

## Top Quality Listed Genres for Movies and TV Shows



File Edit Selection View Go Run Terminal Help ← → ⌂ Search

C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > ...

+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ... Python 3.7.2

```
data1.hist(bins = 20, figsize = (20, 20), color = 'green')
plt.show()
```

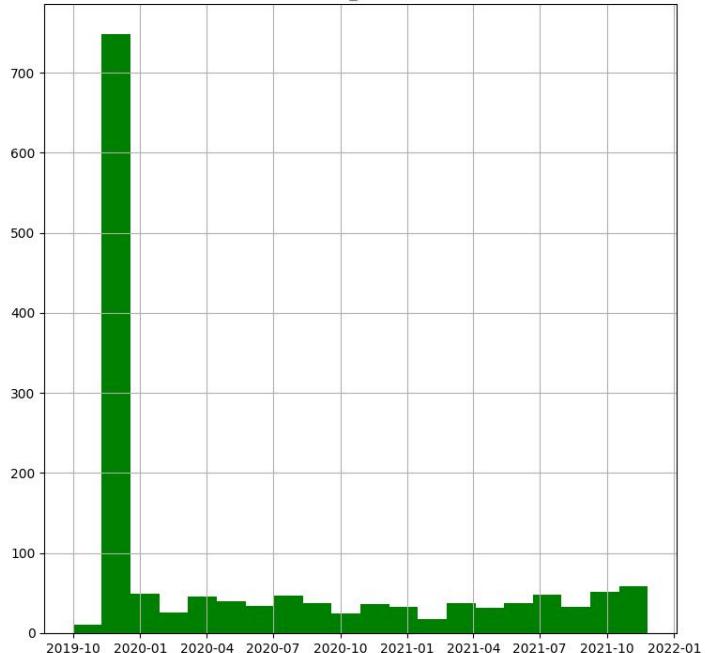
Python

date\_added

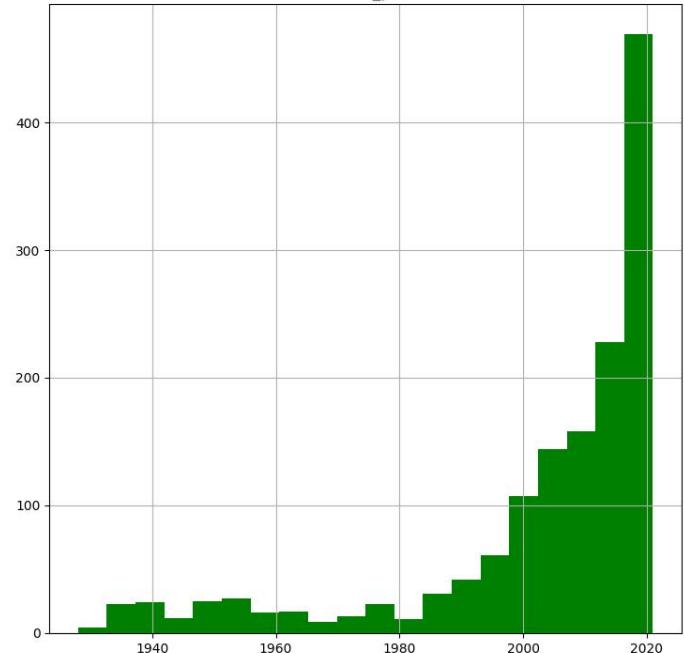
release\_year

5 8 0 0 Type here to search 99+ 32°C 14:59 07-07-2024 9

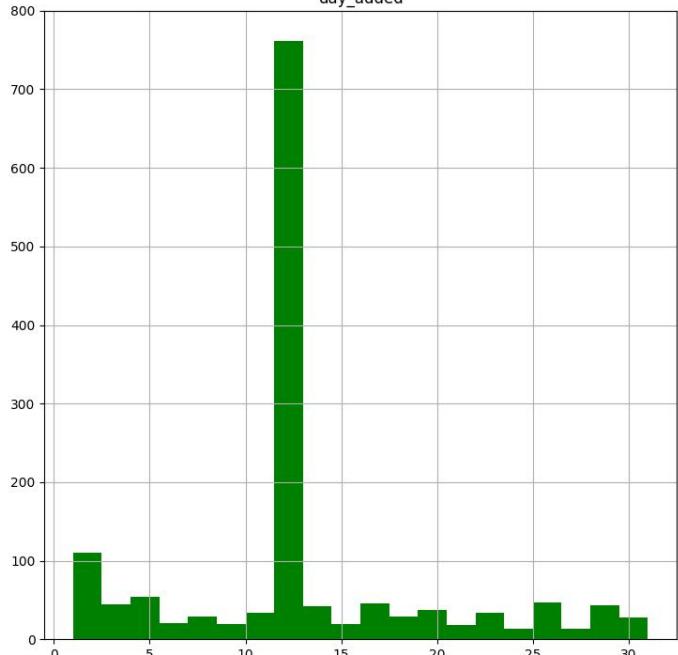
*date\_added*



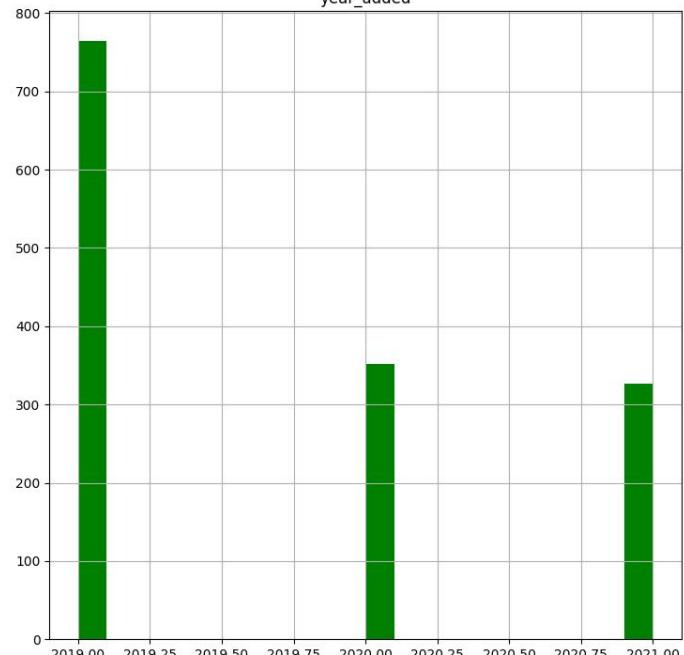
*release\_year*

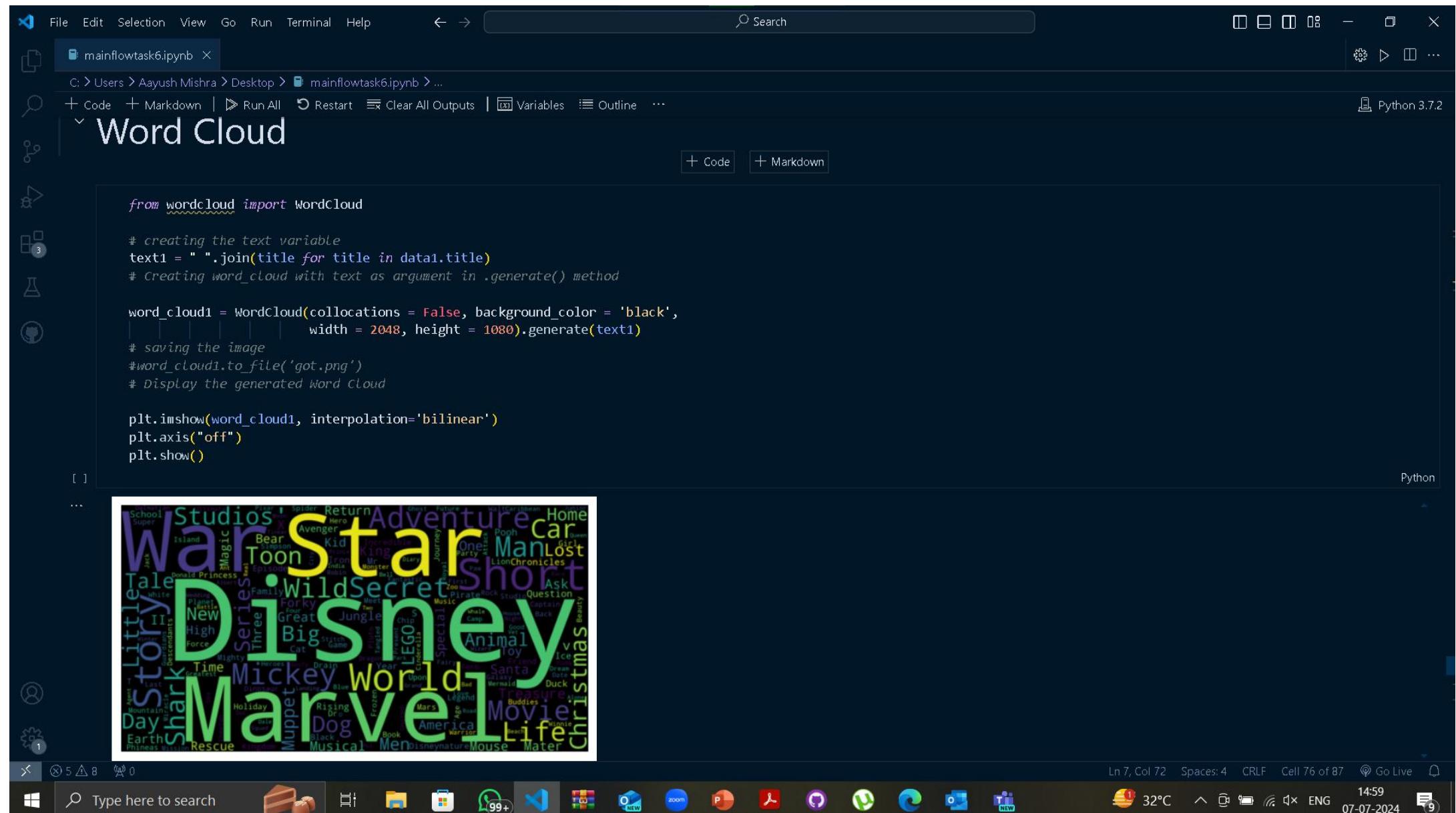


*day\_added*



*year\_added*





File Edit Selection View Go Run Terminal Help ← → ⌂ Search

mainflowtask6.ipynb X C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > ...

+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ... Python 3.7.2

## Generes

```
##Listed_in
from wordcloud import WordCloud
reviews_text = ' '.join(listed_in for listed_in in data1.listed_in)
wordcloud = WordCloud(width=800, height=400, background_color='white', colormap='viridis').generate(reviews_text)
plt.figure(figsize=(15,7))
plt.imshow(wordcloud, interpolation='bilinear')
plt.title('Most Listed Genres')
plt.axis('off')
```

[ ] Python

... (-0.5, 799.5, 399.5, -0.5)

Most Listed Genres

...

Ln 7, Col 72 Spaces: 4 CRLF Cell 76 of 87 ⌂ Go Live

Type here to search

Most Listed Genres



File Edit Selection View Go Run Terminal Help ← → Search

mainflowtask6.ipynb X C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > ...

+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ... Python 3.7.2

# Director's

```
reviews_text = ' '.join(directors for directors in data1.director)
wordcloud = WordCloud(width=800, height=400, background_color='silver', colormap='Spectral').generate(reviews_text)
plt.figure(figsize=(15,7))
plt.imshow(wordcloud, interpolation='bilinear')
plt.title('Directors')
plt.axis('off')
```

[ ] Python

(-0.5, 799.5, 399.5, -0.5)

...

Directors

...

5 8 0

Type here to search

CRLF Cell 81 of 87 Go Live Prettier

32°C ENG 15:00 07-07-2024

## Directors

A word cloud visualization where the size and color of the text represent the frequency and possibly the popularity or significance of the director's name. The most prominent names include James, Robert, John, and Mark, all appearing in large, bold, and varied colors (blue, green, yellow, red). Other notable names include Jack, Kinney, Geronimi, Lasseter, Dave, Andrew, Brian, Gary, Peter, Charles, Nichols, Jack, Hannah, Paul, Hoen, Foster, Steven, Bill, Ron, Clements, Simon, Marshall, Ken, David, Hamilton, Luske, Kenny, Ortega, Stuart, Gillard, and Bob. Smaller names scattered throughout include Chris, Henry, Jon, Turteltaub, Ben, Sharpsteen, Cameron, Scott, Bryan, Singer, David, Silverman, Johnston, Kinney, Geronimi, Steve, Kevin, Deters, Anthony, Russo, Johnson, William, Nathan, Greno, Mattew, Jay, Kenneth, Ward, Peggy, Holmes, Riley, Michael, Hegner, Wermers, Skelton, Stephen, Skelton, and Lee.

File Edit Selection View Go Run Terminal Help ← → ⌂ Search

mainflowtask6.ipynb X C: > Users > Aayush Mishra > Desktop > mainflowtask6.ipynb > ...

+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ... Python 3.7.2

# Country

```
reviews_text = ' '.join(country for country in data1.country)
wordcloud = WordCloud(width=800, height=400, background_color='teal', colormap='plasma').generate(reviews_text)
plt.figure(figsize=(15,7))
plt.imshow(wordcloud, interpolation='bilinear')
plt.title('Most Movies&Tv Shows Releases Countries')
plt.axis('off')
```

[ ] Python

... (-0.5, 799.5, 399.5, -0.5)

... Most Movies&Tv Shows Releases Countries

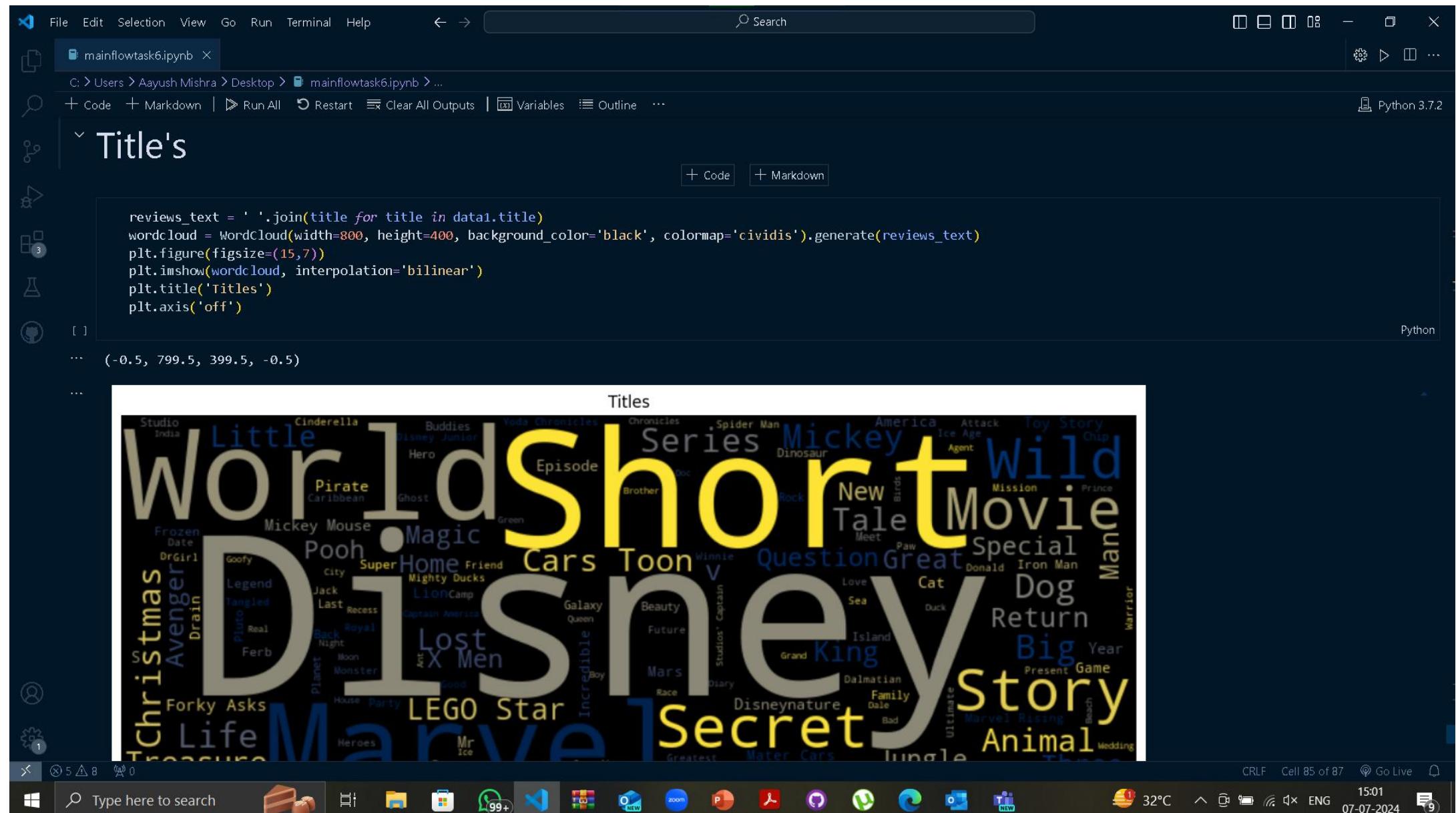
5 8 0 CRLF Cell 83 of 87 Go Live

Type here to search

32°C 15:00 07-07-2024

## Most Movies&Tv Shows Releases Countries

The image features a world map with a teal background. Overlaid on the map are the words "United States" in large orange letters and "United Kingdom" in large purple letters. Numerous country names are scattered across the map, including Australia, Switzerland, South Korea, Argentina, France, Belgium, Spain, Hungary, Malaysia, Kazakhstan, Thailand, Botswana, Arab Emirates, Slovenia, New Zealand, Mexico, Germany, Panama, Japan, Syria, Lebanon, Iraq, Turkey, Russia, Guatemala, Brazil, Philippines, India, Taiwan, China, and others.



## Titles