

In [2]: !pip install numpy

Requirement already satisfied: numpy in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (1.26.3)

[notice] A new release of pip is available: 23.2.1 -> 24.0

[notice] To update, run: python.exe -m pip install --upgrade pip

In [3]: !pip install pandas

Requirement already satisfied: pandas in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (2.2.0)

Requirement already satisfied: numpy<2,>=1.26.0 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from pandas) (1.26.3)

Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from pandas) (2.8.2)

Requirement already satisfied: pytz>=2020.1 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from pandas) (2023.3.post1)

Requirement already satisfied: tzdata>=2022.7 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from pandas) (2023.4)

Requirement already satisfied: six>=1.5 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)

[notice] A new release of pip is available: 23.2.1 -> 24.0

[notice] To update, run: python.exe -m pip install --upgrade pip

In [4]: !pip install matplotlib

Requirement already satisfied: matplotlib in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (3.8.2)

Requirement already satisfied: contourpy>=1.0.1 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from matplotlib) (1.2.0)

Requirement already satisfied: cycler>=0.10 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from matplotlib) (0.12.1)

Requirement already satisfied: fonttools>=4.22.0 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from matplotlib) (4.47.2)

Requirement already satisfied: kiwisolver>=1.3.1 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from matplotlib) (1.4.5)

Requirement already satisfied: numpy<2,>=1.21 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from matplotlib) (1.26.3)

Requirement already satisfied: packaging>=20.0 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from matplotlib) (23.2)

Requirement already satisfied: pillow>=8 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from matplotlib) (10.2.0)

Requirement already satisfied: pyparsing>=2.3.1 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from matplotlib) (3.1.1)

Requirement already satisfied: python-dateutil>=2.7 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from matplotlib) (2.8.2)

Requirement already satisfied: six>=1.5 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from python-dateutil>=2.7->matplotlib) (1.16.0)

[notice] A new release of pip is available: 23.2.1 -> 24.0

[notice] To update, run: python.exe -m pip install --upgrade pip

In [5]: !pip install seaborn

Requirement already satisfied: seaborn in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (0.13.2)

Requirement already satisfied: numpy!=1.24.0,>=1.20 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from seaborn) (1.26.3)

Requirement already satisfied: pandas>=1.2 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from seaborn) (2.2.0)

Requirement already satisfied: matplotlib!=3.6.1,>=3.4 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from seaborn) (3.8.2)

Requirement already satisfied: contourpy>=1.0.1 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (1.2.0)

Requirement already satisfied: cycler>=0.10 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (0.12.1)

Requirement already satisfied: fonttools>=4.22.0 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (4.47.2)

Requirement already satisfied: kiwisolver>=1.3.1 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (1.4.5)

Requirement already satisfied: packaging>=20.0 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (23.2)

Requirement already satisfied: pillow>=8 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (10.2.0)

Requirement already satisfied: pyparsing>=2.3.1 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (3.1.1)

Requirement already satisfied: python-dateutil>=2.7 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (2.8.2)

Requirement already satisfied: pytz>=2020.1 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from pandas>=1.2->seaborn) (2023.3.post1)

Requirement already satisfied: tzdata>=2022.7 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from pandas>=1.2->seaborn) (2023.4)

Requirement already satisfied: six>=1.5 in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from python-dateutil>=2.7->matplotlib!=3.6.1,>=3.4->seaborn) (1.16.0)

[notice] A new release of pip is available: 23.2.1 -> 24.0

[notice] To update, run: python.exe -m pip install --upgrade pip

```
In [6]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```

C:\Users\rahul\AppData\Local\Temp\ipykernel_15532\1883991981.py:2: DeprecationWarning:

Pyarrow will become a required dependency of pandas in the next major release of pandas (pandas 3.0),

(to allow more performant data types, such as the Arrow string type, and better interoperability with other libraries)

but was not found to be installed on your system.

If this would cause problems for you,

please provide us feedback at <https://github.com/pandas-dev/pandas/issues/54466>

```
import pandas as pd
```

In [7]: `!pip install openpyxl`

Requirement already satisfied: openpyxl in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (3.1.2)
Requirement already satisfied: et-xmlfile in c:\users\rahul\appdata\local\programs\python\python312\lib\site-packages (from openpyxl) (1.1.0)

[notice] A new release of pip is available: 23.2.1 -> 24.0

[notice] To update, run: python.exe -m pip install --upgrade pip

In [8]: `df=pd.read_excel('netflix_python.xlsx')`
`df`

Out[8]:

	show_id	type	title	director	cast	country	date_added
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021
...
2652	s2653	Movie	Circus of Books	Rachel Mason	NaN	United States	April 22, 2020
2653	s2654	Movie	The Plagues of Breslau	Patryk Vega	Małgorzata Kożuchowska, Daria Widawska, Kata...	Poland	April 22, 2020
2654	s2655	Movie	The Set Up	Niyi Akinmolayan	Adesua Etomi, Jim Iyke, Dakore Akande, Marie H...	Nigeria	April 22, 2020
2655	s2656	Movie	The Silence of the Marsh	Marc Vigil	Pedro Alonso, Nacho Fresneda, Carmina Barrios,...	Spain	April 22, 2020
2656	s2657	Movie	The Willoughbys	Kris Pearn	Will Forte, Maya Rudolph, Ricky Gervais, Aless...	NaN	NaN

2657 rows × 12 columns



Un nesting and filling null values

```
In [9]: df['cast'].fillna('Unknown Cast')
```

```
Out[9]: 0                                Unknown Cast
1      Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...
2      Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...
3                                Unknown Cast
4      Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...
...
2652                                Unknown Cast
2653      MaÅ¸gorzata KoÅ¸uchowska, Daria Widawska, Kata...
2654      Adesua Etomi, Jim Iyke, Dakore Akande, Marie H...
2655      Pedro Alonso, Nacho Fresneda, Carmina Barrios,...
2656      Will Forte, Maya Rudolph, Ricky Gervais, Aless...
Name: cast, Length: 2657, dtype: object
```

```
In [10]: df['cast']=df['cast'].str.split(',')
df=df.explode('cast')
```

```
In [11]: df
```

Out[11]:

	show_id	type	title	director	cast	country	date_added	release_y
--	---------	------	-------	----------	------	---------	------------	-----------

0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	202
1	s2	TV Show	Blood & Water	NaN	Ama Qamata	South Africa	September 24, 2021	202
1	s2	TV Show	Blood & Water	NaN	Khosi Ngema	South Africa	September 24, 2021	202
1	s2	TV Show	Blood & Water	NaN	Gail Mabalane	South Africa	September 24, 2021	202
1	s2	TV Show	Blood & Water	NaN	Thabang Molaba	South Africa	September 24, 2021	202
...
2656	s2657	Movie	The Willoughbys	Kris Pearn	Maya Rudolph	NaN	NaN	N
2656	s2657	Movie	The Willoughbys	Kris Pearn	Ricky Gervais	NaN	NaN	N
2656	s2657	Movie	The Willoughbys	Kris Pearn	Alessia Cara	NaN	NaN	N
2656	s2657	Movie	The Willoughbys	Kris Pearn	Terry Crews	NaN	NaN	N
2656	s2657	Movie	The Willoughbys	Kris Pearn	J	NaN	NaN	N

21126 rows × 12 columns



In [12]: `df.isna().sum()`

```
Out[12]: show_id      0
         type        0
         title       0
         director    6501
         cast        260
         country     2955
         date_added   6
         release_year 6
         rating       6
         duration     6
         listed_in    6
         description   6
         dtype: int64
```

```
In [13]: df['director'].fillna('Unknown Director')
```

```
Out[13]: 0      Kirsten Johnson
         1      Unknown Director
         1      Unknown Director
         1      Unknown Director
         1      Unknown Director
         ...
        2656      Kris Pearn
        2656      Kris Pearn
        2656      Kris Pearn
        2656      Kris Pearn
        2656      Kris Pearn
        Name: director, Length: 21126, dtype: object
```

```
In [14]: df['country'].fillna('Unknown Country')
         df['date_added'].fillna('Unknown date')
         df['release_year'].fillna('Unknown year')
         df['rating'].fillna('0')
         df['listed_in'].fillna('Unknown Genre')
         df['duration'].fillna('0')
         df['description'].fillna('Unknown desc')
```

```
Out[14]: 0      As her father nears the end of his life, filmm...
         1      After crossing paths at a party, a Cape Town t...
         1      After crossing paths at a party, a Cape Town t...
         1      After crossing paths at a party, a Cape Town t...
         1      After crossing paths at a party, a Cape Town t...
         ...
        2656      Unknown desc
        2656      Unknown desc
        2656      Unknown desc
        2656      Unknown desc
        2656      Unknown desc
        Name: description, Length: 21126, dtype: object
```

```
In [15]: df
```

Out[15]:

	show_id	type	title	director	cast	country	date_added	release_yr
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2021
1	s2	TV Show	Blood & Water	NaN	Ama Qamata	South Africa	September 24, 2021	2021
1	s2	TV Show	Blood & Water	NaN	Khosi Ngema	South Africa	September 24, 2021	2021
1	s2	TV Show	Blood & Water	NaN	Gail Mabalane	South Africa	September 24, 2021	2021
1	s2	TV Show	Blood & Water	NaN	Thabang Molaba	South Africa	September 24, 2021	2021
...
2656	s2657	Movie	The Willoughbys	Kris Pearn	Maya Rudolph	NaN	NaN	NaN
2656	s2657	Movie	The Willoughbys	Kris Pearn	Ricky Gervais	NaN	NaN	NaN
2656	s2657	Movie	The Willoughbys	Kris Pearn	Alessia Cara	NaN	NaN	NaN
2656	s2657	Movie	The Willoughbys	Kris Pearn	Terry Crews	NaN	NaN	NaN
2656	s2657	Movie	The Willoughbys	Kris Pearn	J	NaN	NaN	NaN

21126 rows × 12 columns



```
In [16]: df['listed_in']=df['listed_in'].str.split(',')
df=df.explode('listed_in')
```

```
In [17]: df
```


Out[17]:

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

0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020
1	s2	TV Show	Blood & Water	NaN	Ama Qamata	South Africa	September 24, 2021	2021
1	s2	TV Show	Blood & Water	NaN	Ama Qamata	South Africa	September 24, 2021	2021
1	s2	TV Show	Blood & Water	NaN	Ama Qamata	South Africa	September 24, 2021	2021
1	s2	TV Show	Blood & Water	NaN	Khosi Ngema	South Africa	September 24, 2021	2021
...
2656	s2657	Movie	The Willoughbys	Kris Pearn	Maya Rudolph	NaN	NaN	NaN
2656	s2657	Movie	The Willoughbys	Kris Pearn	Ricky Gervais	NaN	NaN	NaN
2656	s2657	Movie	The Willoughbys	Kris Pearn	Alessia Cara	NaN	NaN	NaN
2656	s2657	Movie	The Willoughbys	Kris Pearn	Terry Crews	NaN	NaN	NaN
2656	s2657	Movie	The Willoughbys	Kris Pearn	J	NaN	NaN	NaN

48488 rows × 12 columns

1. Find the counts of each categorical variable both using graphical and non-graphical Analysis

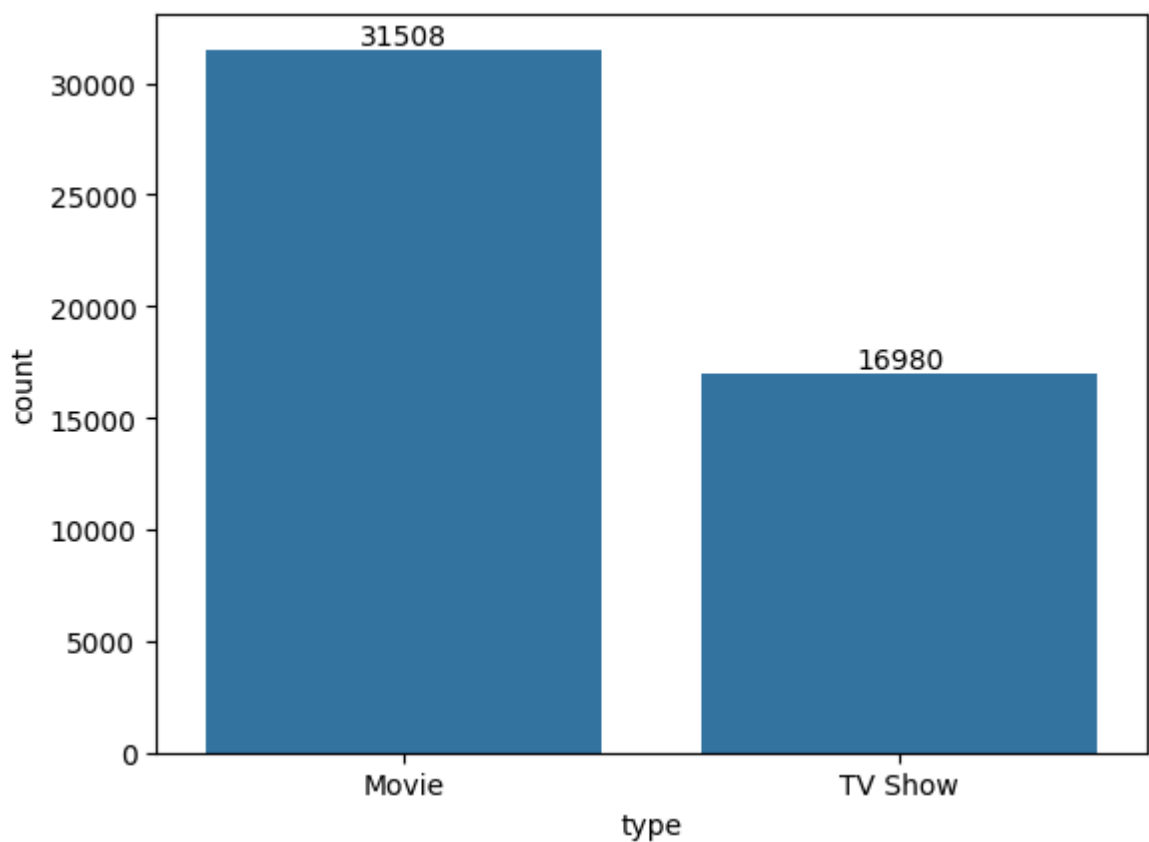
a) Total number of movies and Tv shows

```
In [18]: df['type'].value_counts().reset_index()
```

```
Out[18]:
```

	type	count
0	Movie	31508
1	TV Show	16980

```
In [19]: ax=sns.countplot(x='type',data=df)  
for bars in ax.containers:  
    ax.bar_label(bars)
```



b) Total count of titles

```
In [20]: df['title'].value_counts().reset_index()
```

Out[20]:

	title	count
0	Heartbreak High	141
1	Love, Death & Robots	120
2	American Horror Story	120
3	Social Distance	100
4	Slasher	84
...
2652	Pat a Pat Como	1
2653	Qurious Como	1
2654	Shiva: The Secret World Of Vedas City	1
2655	Shiva: Journey to Plunotaria	1
2656	Dick Johnson Is Dead	1

2657 rows × 2 columns

c) Total count of ratings

```
In [21]: df['rating'].value_counts().reset_index()
```

Out[21]:

	rating	count
0	TV-MA	19324
1	TV-14	12058
2	R	4569
3	PG-13	3484
4	TV-PG	3173
5	PG	1985
6	TV-Y7	1579
7	TV-G	1195
8	TV-Y	1057
9	G	58

d) Total count of countries

```
In [22]: df['country'].value_counts().reset_index()
```

Out[22]:

	country	count
0	United States	11526
1	India	4516
2	Japan	2473
3	South Korea	1316
4	Nigeria	1244
...
285	Singapore	2
286	Thailand, United States	2
287	Switzerland, France, Belgium, United States	2
288	United States, China, United Kingdom	1
289	United States, Brazil, Japan, Spain, India	1

290 rows × 2 columns

e) Total count of Directors

```
In [23]: df['director'].value_counts().reset_index()
```

Out[23]:

	director	count
0	Youssef Chahine	288
1	Hidenori Inoue	177
2	Cathy Garcia-Molina	146
3	Suhas Kadav	129
4	Rajiv Chilaka	117
...
1485	Robert Townsend	1
1486	Jennifer Beamish, Toby Trackman	1
1487	Arnon Manor, Timothy Ware-Hill	1
1488	Eric Notarnicola	1
1489	Kirsten Johnson	1

1490 rows × 2 columns

f) Total count of cast

```
In [24]: df['cast'].value_counts().reset_index()
```

Out[24]:

	cast	count
0	Anupam Kher	29
1	Hassan Hosny	29
2	Ahmed Helmy	28
3	Shaffy Bello	27
4	Tina Mba	27
...
16115	Maggie Q	1
16116	Michael Sirow	1
16117	Stephen Fry	1
16118	Sharon Maughan	1
16119	J	1

16120 rows × 2 columns

g) Total count Genres

```
In [25]: df['listed_in'].value_counts().reset_index()
```

Out[25]:

	listed_in	count
0	International Movies	6374
1	Dramas	4114
2	Comedies	3428
3	Action & Adventure	2702
4	TV Dramas	2478
...
65	Movies	14
66	Documentaries	11
67	Classic & Cult TV	11
68	LGBTQ Movies	5
69	Romantic Movies	4

70 rows × 2 columns

2. Comparison of tv shows vs. movies.

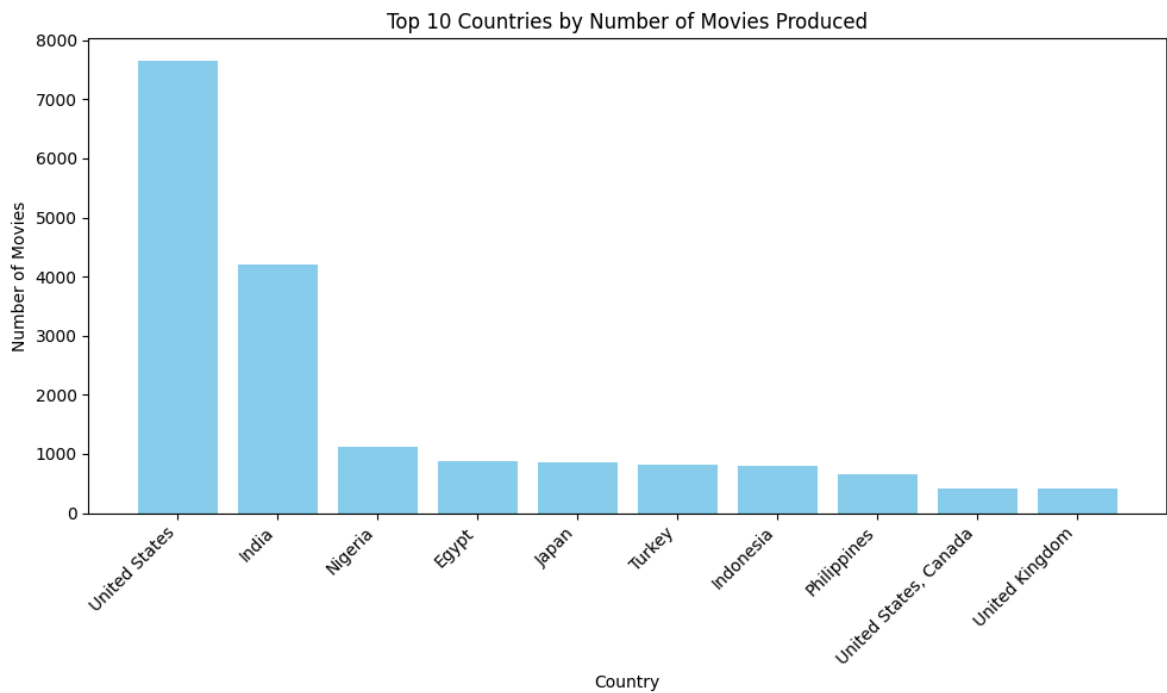
a) Number of movies produced in each country and pick the top 10 countries.

```
In [26]: movies_df=df.loc[df['type']=='Movie',:]  
movies_by_country = movies_df.groupby('country').size().reset_index(name='num_movies')  
top_10_countries = movies_by_country.sort_values(by='num_movies', ascending=False)  
top_10_countries
```

```
Out[26]:
```

	country	num_movies
193	United States	7646
78	India	4200
121	Nigeria	1110
41	Egypt	885
96	Japan	863
162	Turkey	817
85	Indonesia	799
125	Philippines	664
198	United States, Canada	421
168	United Kingdom	416

```
In [27]: plt.figure(figsize=(10, 6))  
plt.bar(top_10_countries['country'], top_10_countries['num_movies'], color='skyblue')  
plt.title('Top 10 Countries by Number of Movies Produced')  
plt.xlabel('Country')  
plt.ylabel('Number of Movies')  
plt.xticks(rotation=45, ha='right')  
plt.tight_layout()  
plt.show()
```



b) Number of Tv-Shows produced in each country and pick the top 10 countries.

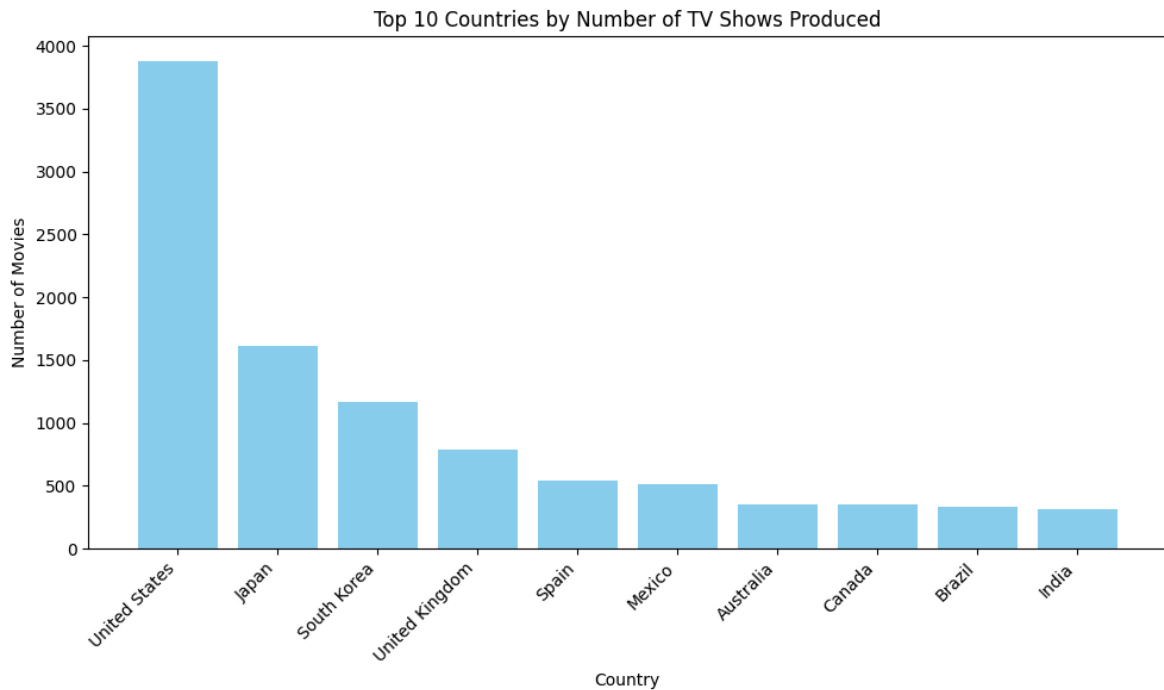
```
In [28]: TV_Show_df=df.loc[df['type']=='TV Show',:]
TV_Show_by_country = TV_Show_df.groupby('country').size().reset_index(name='num_
top_10_countries = TV_Show_by_country.sort_values(by='num_TV_Show', ascending=False)
top_10_countries
```

```
Out[28]:
```

	country	num_TV_Show
87	United States	3880
46	Japan	1610
68	South Korea	1171
76	United Kingdom	785
69	Spain	541
53	Mexico	513
2	Australia	354
12	Canada	348
11	Brazil	337
37	India	316

```
In [29]: plt.figure(figsize=(10, 6))
plt.bar(top_10_countries['country'], top_10_countries['num_TV_Show'], color='sky
plt.title('Top 10 Countries by Number of TV Shows Produced')
plt.xlabel('Country')
plt.ylabel('Number of Movies')
plt.xticks(rotation=45, ha='right')
```

```
plt.tight_layout()
plt.show()
```



From the above graph we can conclude that in both movies and Tv shows United states holds the first place.

c) best week to launch a TV show

```
In [60]: TV_Show_week_df=df.loc[df['type']=='TV Show',:]
TV_Show_week_df['date_added']=pd.to_datetime(TV_Show_df['date_added'])
TV_Show_by_week = TV_Show_week_df.groupby(TV_Show_df['date_added'].dt.isocalendar())
best_week = TV_Show_by_week.sort_values(by='num_TV_Show', ascending=False)
best_week
```

C:\Users\rahul\AppData\Local\Temp\ipykernel_15532\4261123588.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

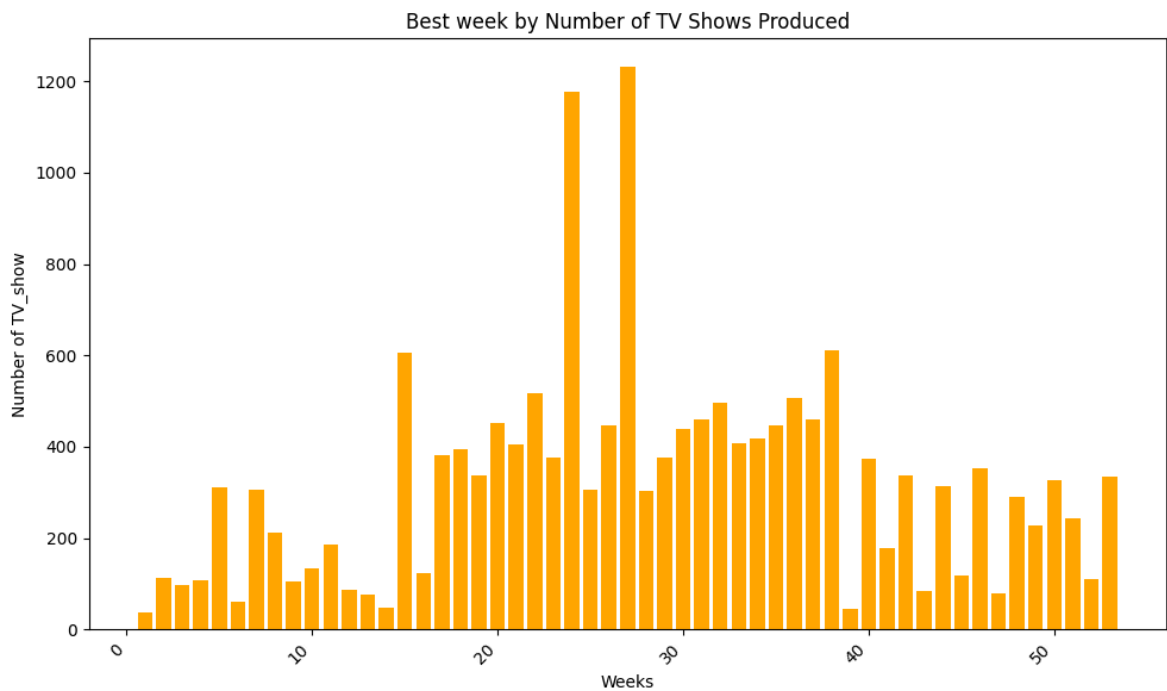
See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
TV_Show_week_df['date_added']=pd.to_datetime(TV_Show_df['date_added'])

Out[60]:

	week	num_TV_Show
26	27	1232
23	24	1178
37	38	610
14	15	606
21	22	517
35	36	507
31	32	497
36	37	461
30	31	459
19	20	452
25	26	448
34	35	446
29	30	439
33	34	418
32	33	408
20	21	406
17	18	394
16	17	381
28	29	377
22	23	377
39	40	373
45	46	354
41	42	337
18	19	337
52	53	334
49	50	327
43	44	313
4	5	310
6	7	307
24	25	306
27	28	304
47	48	290
50	51	243

	week	num_TV_Show
48	49	229
7	8	213
10	11	186
40	41	178
9	10	135
15	16	123
44	45	117
1	2	112
51	52	111
3	4	107
8	9	104
2	3	97
11	12	86
42	43	85
46	47	79
12	13	76
5	6	62
13	14	49
38	39	45
0	1	38

```
In [61]: plt.figure(figsize=(10, 6))
plt.bar(best_week['week'], best_week['num_TV_Show'], color='Orange')
plt.title('Best week by Number of TV Shows Produced')
plt.xlabel('Weeks')
plt.ylabel('Number of TV_show')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```



d) Best week to launch a movie

```
In [32]: Movies_week_df=df.loc[df['type']=='Movie',:]
Movies_week_df['date_added']=pd.to_datetime(Movies_week_df['date_added'])
Movies_by_week = Movies_week_df.groupby(Movies_week_df['date_added'].dt.isocalendar().week)
best_week = Movies_by_week.sort_values(by='num_Movies', ascending=False)
best_week
```

C:\Users\rahul\AppData\Local\Temp\ipykernel_15532\1486471989.py:2: SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame.

Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

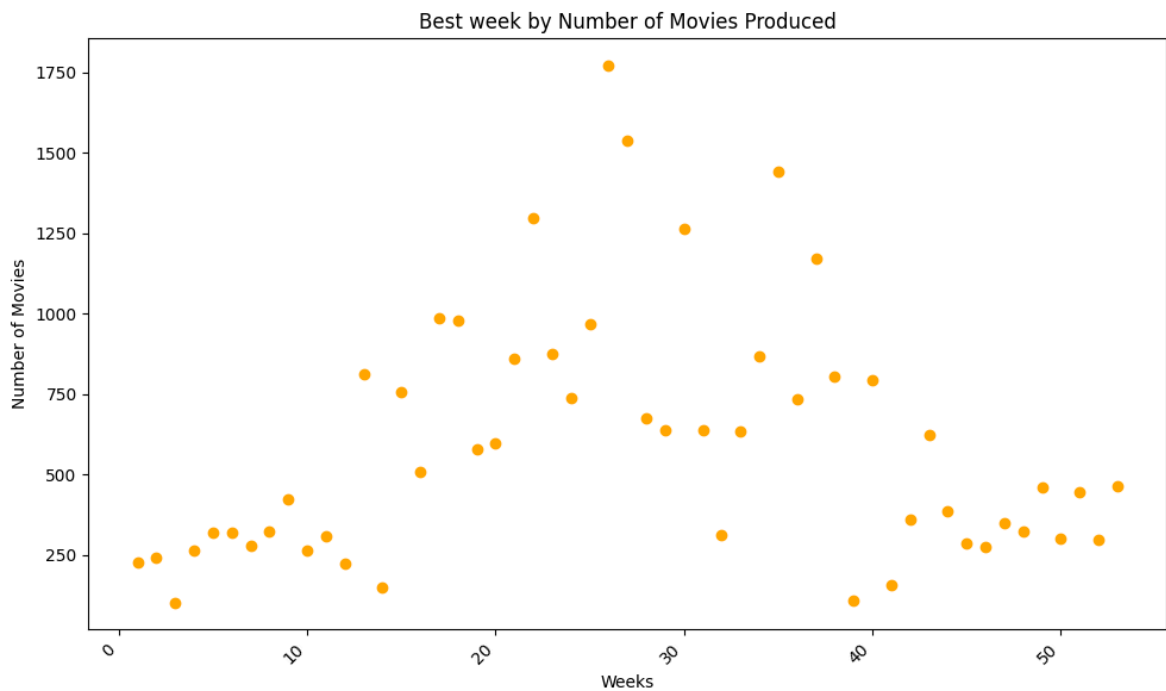
```
Movies_week_df['date_added']=pd.to_datetime(Movies_week_df['date_added'])
```

Out[32]:

	week	num_Movies
25	26	1771
26	27	1538
34	35	1442
21	22	1298
29	30	1265
36	37	1171
16	17	987
17	18	977
24	25	968
22	23	876
33	34	867
20	21	861
12	13	811
37	38	805
39	40	791
14	15	756
23	24	739
35	36	734
27	28	675
28	29	637
30	31	637
32	33	633
42	43	621
19	20	598
18	19	579
15	16	506
52	53	462
48	49	459
50	51	446
8	9	424
43	44	384
41	42	360
46	47	348

	week	num_Movies
47	48	323
7	8	322
5	6	320
4	5	320
31	32	313
10	11	308
49	50	302
51	52	298
44	45	286
6	7	279
45	46	276
3	4	265
9	10	263
1	2	242
0	1	225
11	12	223
40	41	156
13	14	147
38	39	107
2	3	101

```
In [38]: plt.figure(figsize=(10, 6))
plt.scatter(best_week['week'], best_week['num_Movies'], color='Orange')
plt.title('Best week by Number of Movies Produced')
plt.xlabel('Weeks')
plt.ylabel('Number of Movies')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```



e) Best month to launch a TV Show

```
In [41]: df['date_added']=pd.to_datetime(df['date_added'])
```

```
In [40]: TV_Show_df=df.loc[df['type']=='TV Show',:]
TV_Show_df['date_added']=pd.to_datetime(TV_Show_df['date_added'])

TV_Show_by_months = TV_Show_df.groupby(TV_Show_df['date_added'].dt.strftime('%b'))
Best_month = TV_Show_by_months.sort_values(by='num_TV_Show', ascending=False).head(1)
Best_month
```

C:\Users\rahul\AppData\Local\Temp\ipykernel_15532\220259267.py:2: SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

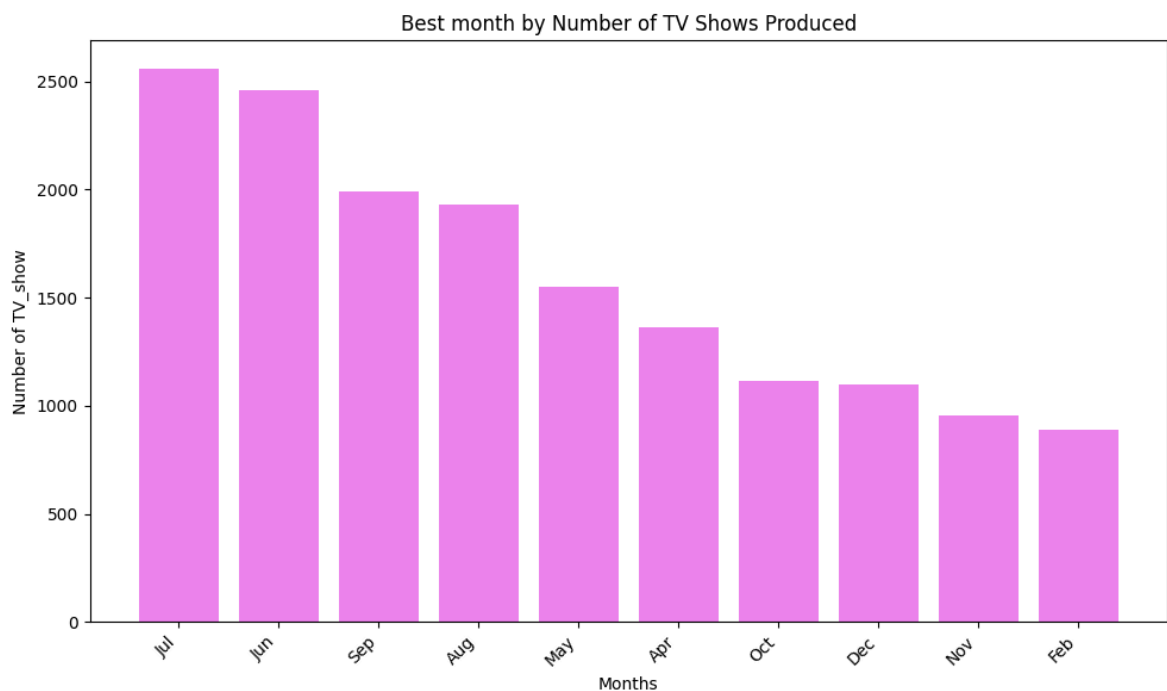
See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

```
TV_Show_df['date_added']=pd.to_datetime(TV_Show_df['date_added'])
```

Out[40]:

	date_added	num_TV_Show
5	Jul	2561
6	Jun	2459
11	Sep	1990
1	Aug	1930
8	May	1549
0	Apr	1364
10	Oct	1113
2	Dec	1097
9	Nov	953
3	Feb	892

```
In [42]: plt.figure(figsize=(10, 6))
plt.bar(Best_month['date_added'], Best_month['num_TV_Show'], color='violet')
plt.title('Best month by Number of TV Shows Produced')
plt.xlabel('Months')
plt.ylabel('Number of TV_show')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```



e) Best Month to launch a Movie

```
In [43]: Movies_df=df.loc[df['type']=='Movie',:]
Movies_df['date_added']=pd.to_datetime(Movies_df['date_added'])
Movies_by_months = Movies_df.groupby(Movies_df['date_added'].dt.strftime('%b')).
best_month = Movies_by_months.sort_values(by='num_Movies', ascending=False)
best_month
```

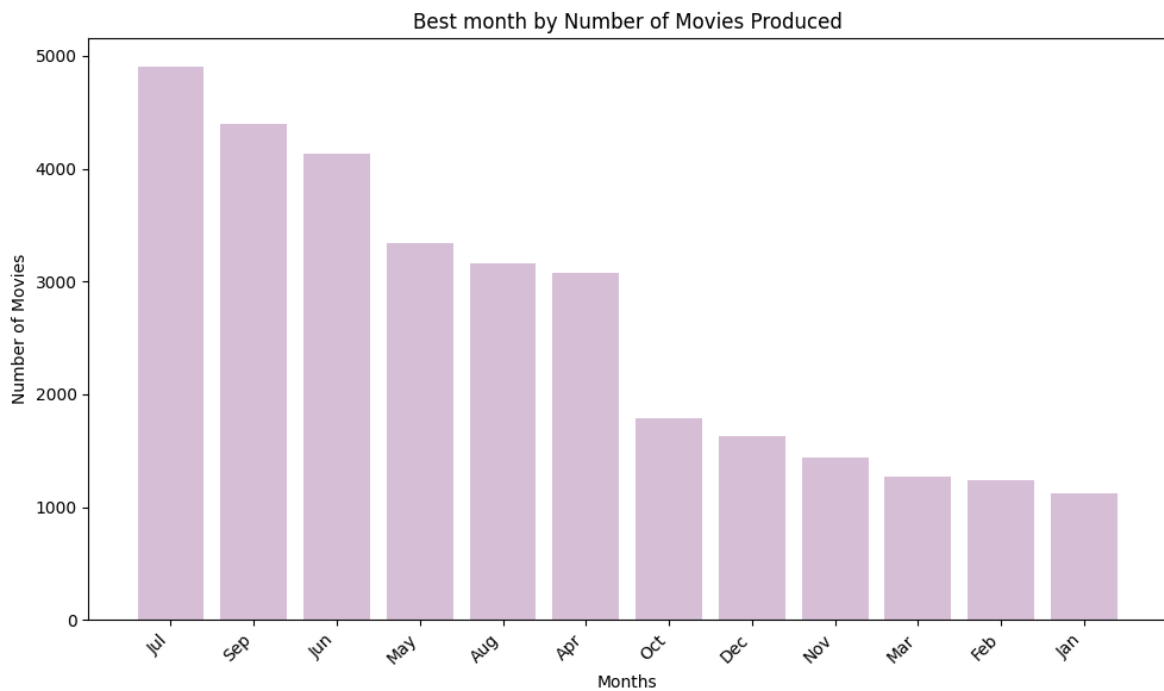
```
C:\Users\rahul\AppData\Local\Temp\ipykernel_15532\1955631701.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
Movies_df['date_added']=pd.to_datetime(Movies_df['date_added'])
```

```
Out[43]:
```

	date_added	num_Movies
5	Jul	4908
11	Sep	4397
6	Jun	4132
8	May	3337
1	Aug	3158
0	Apr	3074
10	Oct	1787
2	Dec	1629
9	Nov	1445
7	Mar	1269
3	Feb	1241
4	Jan	1125

```
In [44]: plt.figure(figsize=(10, 6))
plt.bar(best_month['date_added'], best_month['num_Movies'], color='thistle')
plt.title('Best month by Number of Movies Produced')
plt.xlabel('Months')
plt.ylabel('Number of Movies')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```

4. Top 10 Directors for movies and Tv shows

a) Top 10 Directors of Tv shows

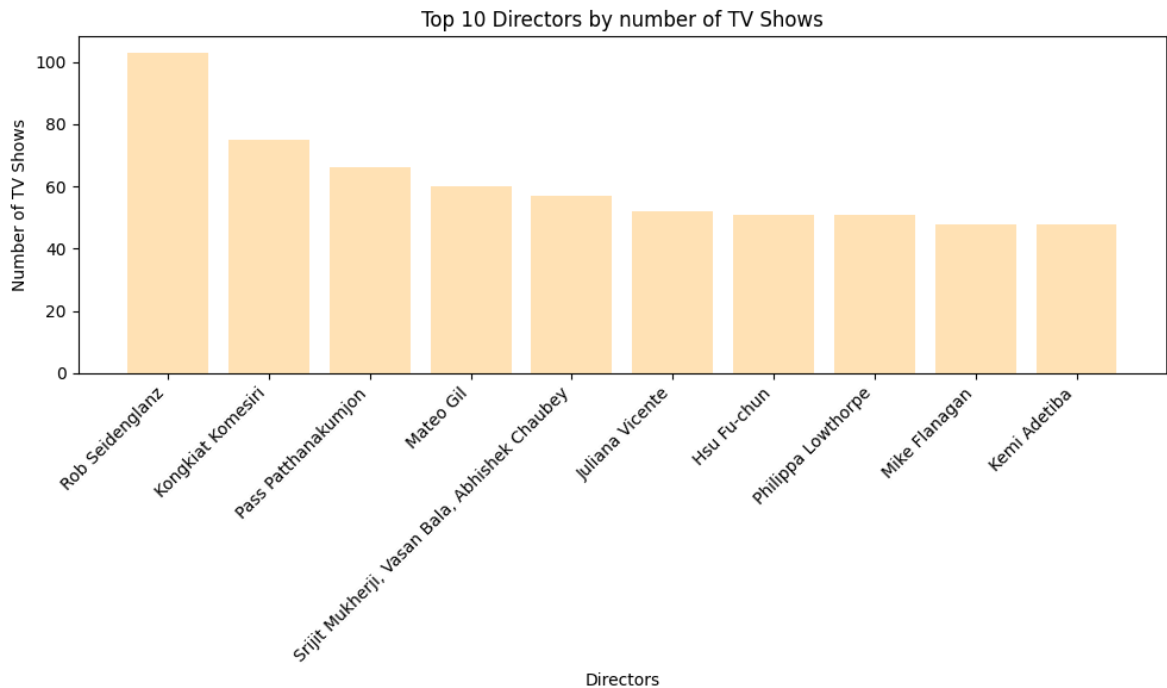
```
In [45]: TV_directors=df.loc[df['type']=='TV Show',:]
Top10 = TV_directors.groupby('director').size().reset_index(name='num_TV_Show')
top_10_directors = Top10.sort_values(by='num_TV_Show', ascending=False).head(10)
top_10_directors
```

```
Out[45]:
```

	director	num_TV_Show
82	Rob Seidenglanz	103
56	Kongkiat Komesiri	75
74	Pass Patthanakumjon	66
65	Mateo Gil	60
91	Srijit Mukherji, Vasan Bala, Abhishek Chaubey	57
49	Juliana Vicente	52
34	Hsu Fu-chun	51
78	Philippa Lowthorpe	51
68	Mike Flanagan	48
53	Kemi Adetiba	48

```
In [46]: plt.figure(figsize=(10, 6))
plt.bar(top_10_directors['director'], top_10_directors['num_TV_Show'], color='mo
plt.title('Top 10 Directors by number of TV Shows')
plt.xlabel('Directors')
plt.ylabel('Number of TV Shows')
```

```
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
```



b) Top 10 Directors of Movies

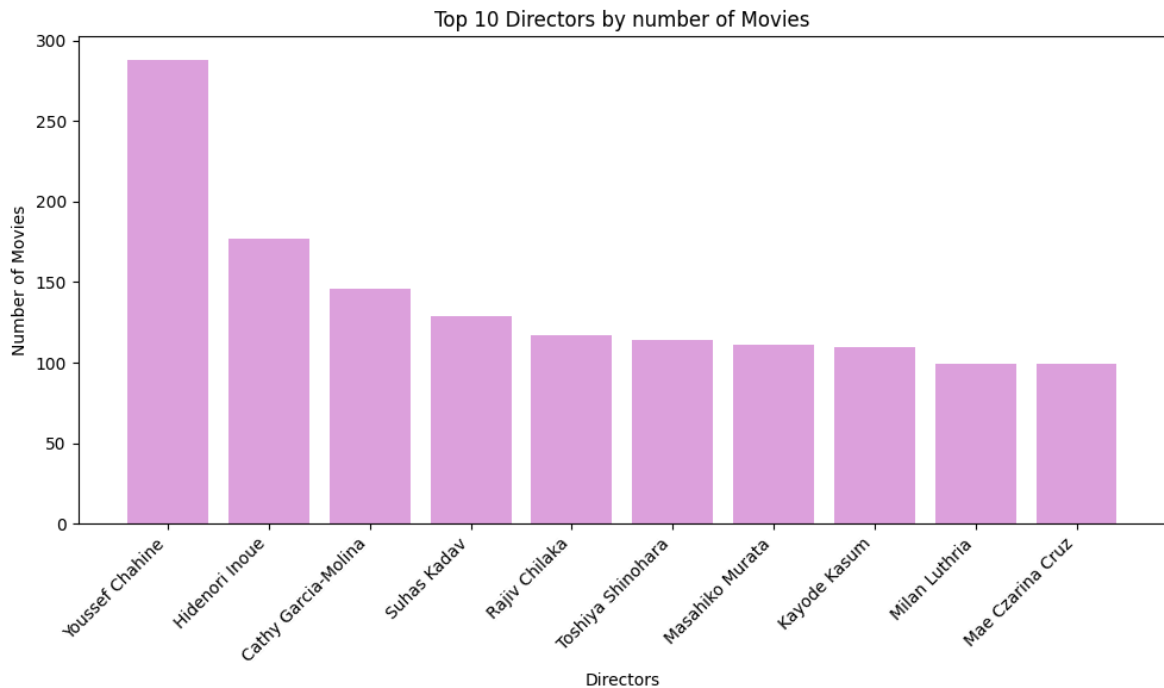
```
In [47]: Movies_directors=df.loc[df['type']=='Movie',:]
Top10 = Movies_directors.groupby('director').size().reset_index(name='num_Movies')
top_10_directors = Top10.sort_values(by='num_Movies', ascending=False).head(10)
top_10_directors
```

```
Out[47]:
```

	director	num_Movies
1373	Youssef Chahine	288
495	Hidenori Inoue	177
211	Cathy Garcia-Molina	146
1243	Suhas Kadav	129
1034	Rajiv Chilaka	117
1304	Toshiya Shinohara	114
804	Masahiko Murata	111
656	Kayode Kasum	110
869	Milan Luthria	99
754	Mae Czarina Cruz	99

```
In [48]: plt.figure(figsize=(10, 6))
plt.bar(top_10_directors['director'], top_10_directors['num_Movies'], color='plu')
plt.title('Top 10 Directors by number of Movies')
plt.xlabel('Directors')
plt.ylabel('Number of Movies')
```

```
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
```



c) Top 10 casts of TV shows

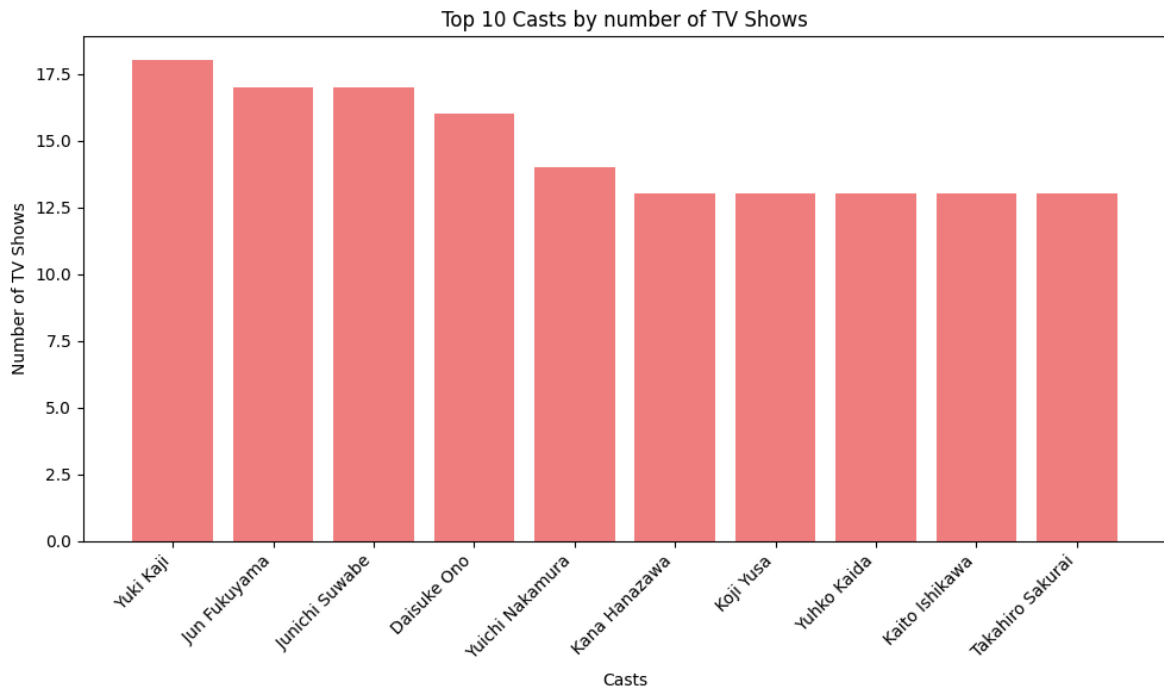
```
In [62]: TV_casts=df.loc[df['type']=='TV Show',:]
Top10 = TV_casts.groupby('cast').size().reset_index(name='num_TV_Show')
top_10_casts = Top10.sort_values(by='num_TV_Show', ascending=False).head(10)
top_10_casts
```

```
Out[62]:
```

	cast	num_TV_Show
5444	Yuki Kaji	18
2487	Jun Fukuyama	17
2503	Junichi Suwabe	17
1071	Daisuke Ono	16
5439	Yuichi Nakamura	14
2549	Kana Hanazawa	13
2837	Koji Yusa	13
5435	Yuhko Kaida	13
2543	Kaito Ishikawa	13
4956	Takahiro Sakurai	13

```
In [65]: plt.figure(figsize=(10, 6))
plt.bar(top_10_casts['cast'], top_10_casts['num_TV_Show'], color='lightcoral')
plt.title('Top 10 Casts by number of TV Shows')
plt.xlabel('Casts')
plt.ylabel('Number of TV Shows')
```

```
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
```



d) Top 10 Casts of Movies

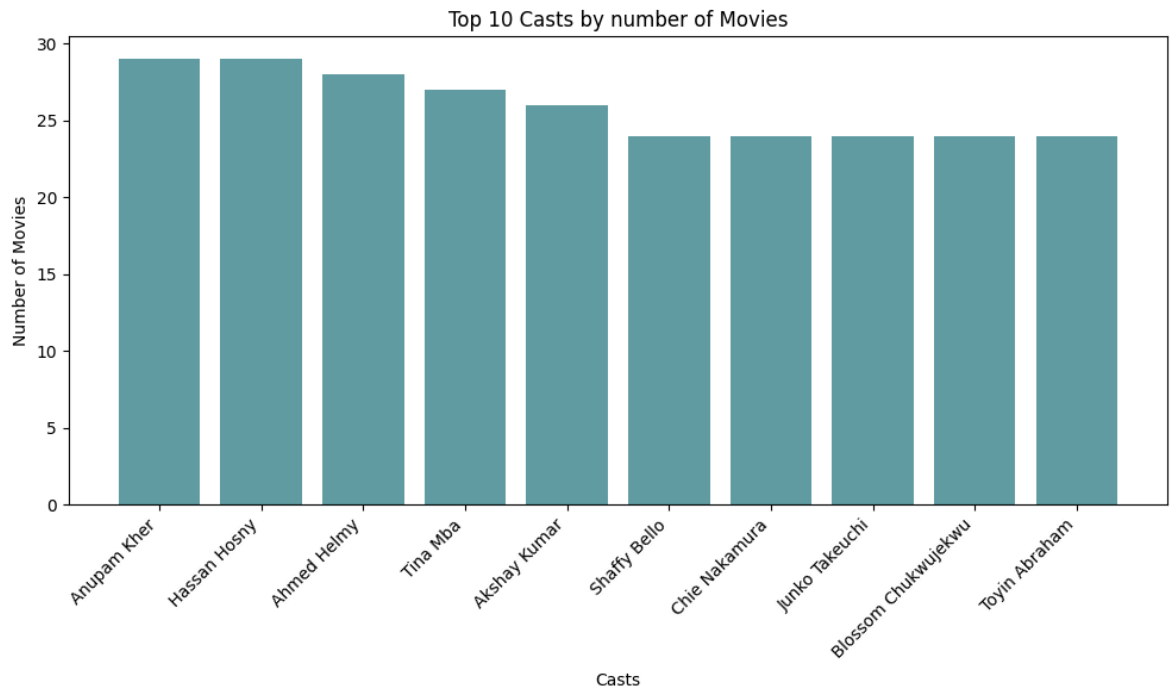
```
In [66]: Movies_casts=df.loc[df['type']=='Movie',:]
Top10 = Movies_casts.groupby('cast').size().reset_index(name='num_Movies')
top_10_casts = Top10.sort_values(by='num_Movies', ascending=False).head(10)
top_10_casts
```

```
Out[66]:
```

	cast	num_Movies
781	Anupam Kher	29
3302	Hassan Hosny	29
9539	Ahmed Helmy	28
8734	Tina Mba	27
9544	Akshay Kumar	26
7993	Shaffy Bello	24
1600	Chie Nakamura	24
10104	Junko Takeuchi	24
1179	Blossom Chukwujekwu	24
8863	Toyin Abraham	24

```
In [67]: plt.figure(figsize=(10, 6))
plt.bar(top_10_casts['cast'], top_10_casts['num_Movies'], color='cadetblue')
plt.title('Top 10 Casts by number of Movies')
plt.xlabel('Casts')
plt.ylabel('Number of Movies')
```

```
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
```



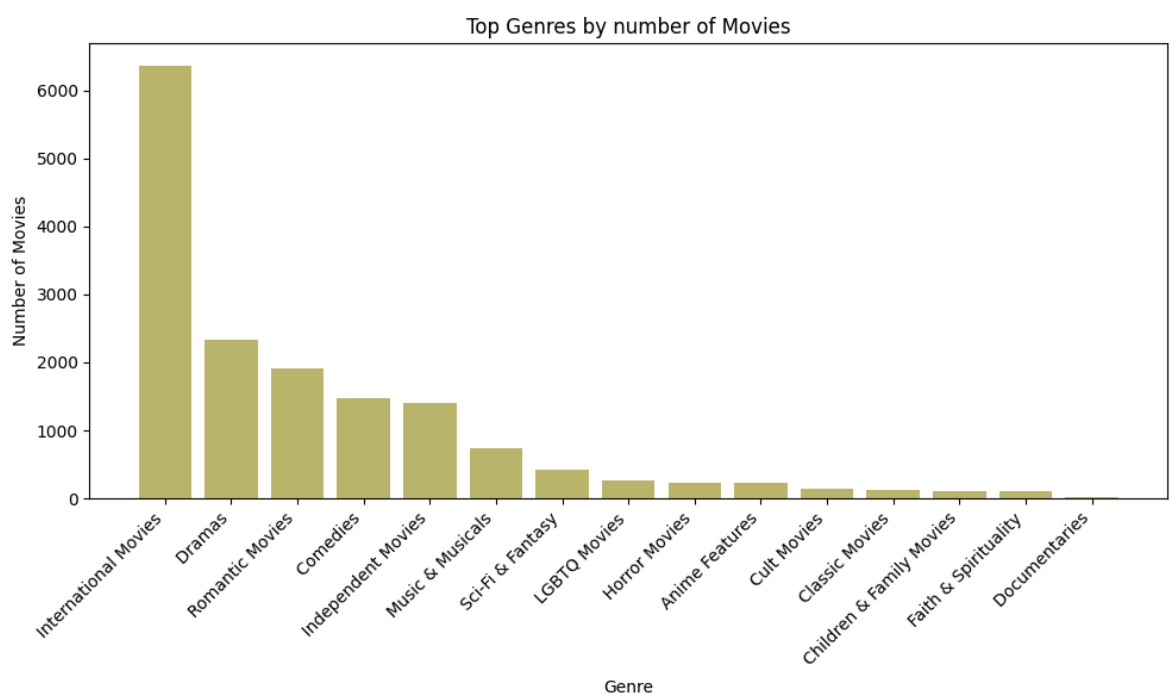
5) a. Genre of movies that are more popular or produced more

```
In [49]: Movies_genre=df.loc[df['type']=='Movie',:]
genre=Movies_genre.groupby('listed_in').size().reset_index(name='num_Movies').he
top_genre = genre.sort_values(by='num_Movies', ascending=False)
top_genre
```

Out[49]:

	listed_in	num_Movies
10	International Movies	6374
6	Dramas	2331
13	Romantic Movies	1914
3	Comedies	1478
9	Independent Movies	1397
12	Music & Musicals	737
14	Sci-Fi & Fantasy	429
11	LGBTQ Movies	262
8	Horror Movies	236
0	Anime Features	234
4	Cult Movies	150
2	Classic Movies	120
1	Children & Family Movies	115
7	Faith & Spirituality	107
5	Documentaries	11

```
In [50]: plt.figure(figsize=(10, 6))
plt.bar(top_genre['listed_in'], top_genre['num_Movies'], color='darkkhaki')
plt.title('Top Genres by number of Movies')
plt.xlabel('Genre')
plt.ylabel('Number of Movies')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
```



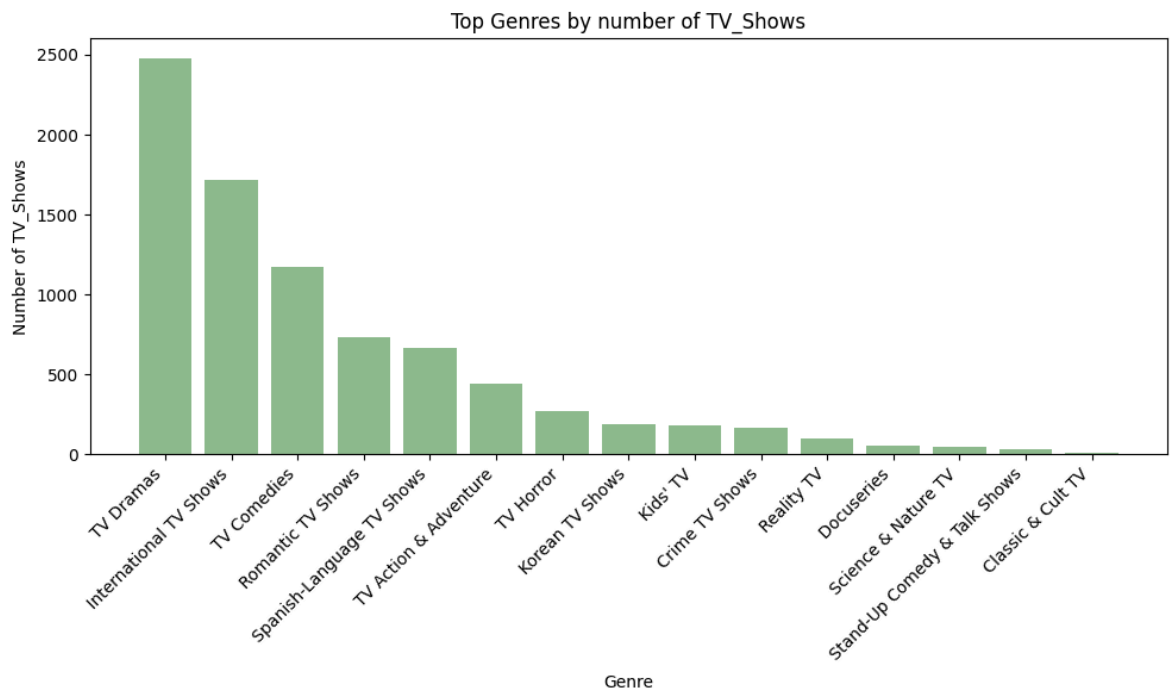
b. Genres of TV Shows that are produced more

```
In [51]: TV_genre=df.loc[df['type']=='TV Show',:]
genre=TV_genre.groupby('listed_in').size().reset_index(name='num_TV_Shows').head
top_genre = genre.sort_values(by='num_TV_Shows', ascending=False)
top_genre
```

```
Out[51]:
```

	listed_in	num_TV_Shows
13	TV Dramas	2478
3	International TV Shows	1717
12	TV Comedies	1173
7	Romantic TV Shows	734
9	Spanish-Language TV Shows	662
11	TV Action & Adventure	442
14	TV Horror	270
5	Korean TV Shows	186
4	Kids' TV	183
1	Crime TV Shows	164
6	Reality TV	100
2	Docuseries	54
8	Science & Nature TV	48
10	Stand-Up Comedy & Talk Shows	35
0	Classic & Cult TV	11

```
In [52]: plt.figure(figsize=(10, 6))
plt.bar(top_genre['listed_in'], top_genre['num_TV_Shows'], color='darkseagreen')
plt.title('Top Genres by number of TV_Shows')
plt.xlabel('Genre')
plt.ylabel('Number of TV_Shows')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
```



6. Find After how many days the movie will be added to Netflix after the release of the movie (you can consider the recent past data)

)

In [53]: df

Out[53]:

	show_id	type	title	director	cast	country	date_added	release_year
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	2021-09-25	2020
1	s2	TV Show	Blood & Water	NaN	Ama Qamata	South Africa	2021-09-24	2021
1	s2	TV Show	Blood & Water	NaN	Ama Qamata	South Africa	2021-09-24	2021
1	s2	TV Show	Blood & Water	NaN	Ama Qamata	South Africa	2021-09-24	2021
1	s2	TV Show	Blood & Water	NaN	Khosi Ngema	South Africa	2021-09-24	2021
...
2656	s2657	Movie	The Willoughbys	Kris Pearn	Maya Rudolph	NaN	NaT	Na
2656	s2657	Movie	The Willoughbys	Kris Pearn	Ricky Gervais	NaN	NaT	Na
2656	s2657	Movie	The Willoughbys	Kris Pearn	Alessia Cara	NaN	NaT	Na
2656	s2657	Movie	The Willoughbys	Kris Pearn	Terry Crews	NaN	NaT	Na
2656	s2657	Movie	The Willoughbys	Kris Pearn	J	NaN	NaT	Na

48488 rows × 12 columns

```
In [54]: movies=df.loc[df['type']=='Movie']
```

```
In [57]: movies['release_year']=pd.to_datetime(movies['release_year'])
movies['added_after_N_days']=(movies['date_added']-movies['release_year']).dt.da
```

```

C:\Users\rahul\AppData\Local\Temp\ipykernel_15532\3098157394.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
    movies['release_year']=pd.to_datetime(movies['release_year'])
C:\Users\rahul\AppData\Local\Temp\ipykernel_15532\3098157394.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
    movies['added_after_N_days']=(movies['date_added']-movies['release_year']).dt.days

```

```
In [59]: movies['added_after_N_days'].reset_index()
```

```
Out[59]:
```

	index	added_after_N_days
0	0	18894.0
1	6	18893.0
2	6	18893.0
3	6	18893.0
4	6	18893.0
...
31503	2656	NaN
31504	2656	NaN
31505	2656	NaN
31506	2656	NaN
31507	2656	NaN

31508 rows × 2 columns

```
In [ ]:
```