```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv('/content/netflix.csv')
```

```
df.head()
```

	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, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021

```
df.shape
```

```
(8807, 12)
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5398 entries, 0 to 5397
Data columns (total 12 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   show_id       5398 non-null   object
 1   type          5398 non-null   object
 2   title         5397 non-null   object
 3   director      3515 non-null   object
 4   cast          4903 non-null   object
 5   country       4735 non-null   object
 6   date_added    5397 non-null   object
 7   release_year  5397 non-null   float64
 8   rating        5397 non-null   object
 9   duration      5397 non-null   object
 10  listed_in     5397 non-null   object
 11  description   5397 non-null   object
dtypes: float64(1), object(11)
memory usage: 506.2+ KB
```

```
df.isnull().sum()*100/len(df)
```

	0
show_id	0.000000
type	0.000000
title	0.018525
director	34.883290
cast	9.170063
country	12.282327
date_added	0.018525
release_year	0.018525
rating	0.018525
duration	0.018525
listed_in	0.018525
description	0.018525

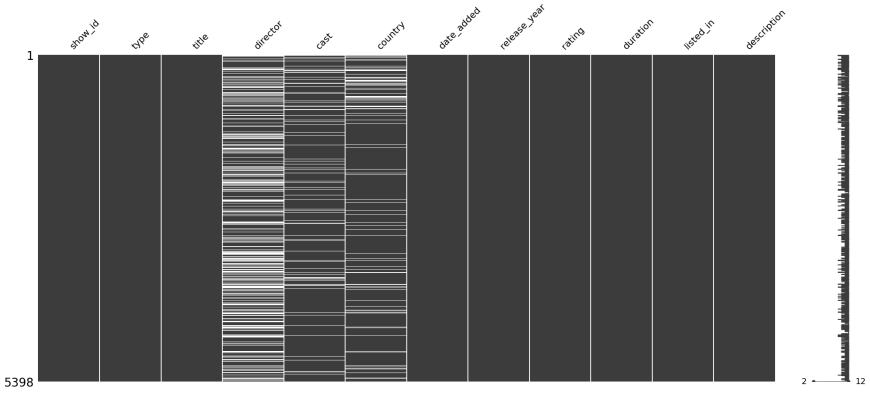
dtype: float64

```
df[df.isnull().all(axis = 1)]
```

show_id	type	title	director	cast	country	date_added	release_year	rating	dur

```
import missingno as msno
msno.matrix(df)
```

<Axes: >



we can observe most of the missing values are present in director followed by cast and country.

```
# Exploratory Data Analysis (EDA)
```

Non Graphical Analysis

```
# unique values
df.nunique()
```

	0
show_id	5397
type	2
title	5397
director	2721
cast	4749
country	475
date_added	1177
release_year	62
rating	11
duration	208
listed_in	430
description	5381

dtype: int64

```
df['type'].unique()
```

```
array(['Movie', 'TV Show', 'TV Sh'], dtype=object)
```

```
df
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	descrip
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020.0	PG-13	90 min	Documentaries	As her f nea end life, fil
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021.0	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	cro path party, a Tov
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021.0	TV-MA	1 Season	Crime TV Shows, International TV Shows, TV Act...	To prote family f pov drug
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021.0	TV-MA	1 Season	Docuseries, Reality TV	F flirtation toilet ta down a
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar,	India	September 24, 2021	2021.0	TV-MA	2 Seasons	International TV Shows, Romantic TV	In a coa ce

```
# row 5397 is almost empty so lets remove it from the dataset
df = df.drop(5397)

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

	0
show_id	0
type	0
title	0
director	2633
cast	825
country	831
date_added	10
release_year	0
rating	4
duration	3
listed_in	0
description	0

dtype: int64

```
df.nunique()
```

	0
show_id	5397
type	2
title	5397
director	2721
cast	4749
country	475
date_added	1177
release_year	62
rating	11
duration	208
listed_in	430
description	5381

dtype: int64

```
df['type'].unique()          # now it's showing only 2 unique values
```

```
array(['Movie', 'TV Show'], dtype=object)
```

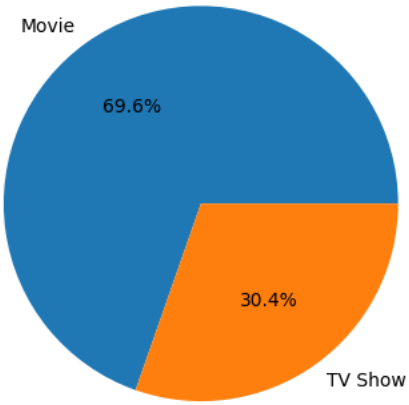
```
df['type'].value_counts()
```

type	count
Movie	6131
TV Show	2675

dtype: int64

based on this data more movies are released by netflix compare to tv shows.

```
plt.pie(df['type'].value_counts().values, labels = df['type'].value_counts().index, autopct = '
plt.show()
```




```
# null value
```

Null value imputing strategy for "director" column:

"

1. Fill with "Unknown"
2. Max/Mode per country, listed_in # who directed the most number of movies in a country for a particular genre?

```
df.columns
```

```
Index(['show_id', 'type', 'title', 'director', 'cast', 'country', 'date_added',
       'release_year', 'rating', 'duration', 'listed_in', 'description'],
      dtype='object')
```

```
df.head()
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020.0	PG-13	90 min	Documentaries	As her father nears the end of his life, filmm...
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021.0	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	After crossing paths at a party, a Cape Town t...
					Sami Bouajila						Crime TV	

```
df['director'].value_counts()
```

director	count
Rajiv Chilaka	17
Suhas Kadav	15
Raúl Campos, Jan Suter	14
Marcus Raboy	13
Youssef Chahine	12
...	...
Kasper Collin	1
Fazila Allana	1
Eric Abrams	1
Agustí Villaronga	1
Harry Chaskin	1

2721 rows × 1 columns

dtype: int64

```
type_director_count = df[['type','director']].value_counts().reset_index(name = 'count')
type_director_count
```

	type	director	count
0	Movie	Rajiv Chilaka	19
1	Movie	Raúl Campos, Jan Suter	18
2	Movie	Suhas Kadav	16
3	Movie	Marcus Raboy	15
4	Movie	Jay Karas	14
...	...	...	...
4571	Movie	Jan-Peter Horns	1
4572	Movie	Jan Suter, Raúl Campos Delgado	1
4573	Movie	Jan Suter, Raúl Campos	1
4574	Movie	Jan Suter	1
4575	Movie	Jared Stern	1

4576 rows × 3 columns

Next steps: Generate code with type_director_count | View recommended plots | New interactive sheet

```
top_5_per_type = type_director_count.groupby('type').apply( lambda x : x.nlargest(5, 'count')).
top_5_per_type
```

```
<ipython-input-8-37953b6c7aff>:1: DeprecationWarning: DataFrameGroupBy.apply operated on the grouping columns. This behavior is depr
  top_5_per_type = type_director_count.groupby('type').apply( lambda x : x.nlargest(5, 'count')).reset_index(drop = True)
```

	type	director	count
0	Movie	Rajiv Chilaka	19
1	Movie	Raúl Campos, Jan Suter	18
2	Movie	Suhas Kadav	16
3	Movie	Marcus Raboy	15
4	Movie	Jay Karas	14
5	TV Show	Alastair Fothergill	3
6	TV Show	Hsu Fu-chun	2
7	TV Show	Iginio Straffi	2
8	TV Show	Ken Burns	2
9	TV Show	Shin Won-ho	2

Next steps: Generate code with top_5_per_type | View recommended plots | New interactive sheet

in movies most productive director is 'Rajiv Chilaka' who produced maximum number of movies and in tv shows most productive director is 'Alastair Fothergill'. so we can fill null values by using this director names.

```
# for simplicy currently i'm replacing all nan values in director column with unknown
df['director'].fillna('unknown').isnull().sum().item()
```

```
0
```

```
df['director'] = df['director'].fillna('unknown')

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

	0
show_id	0
type	0
title	0
director	0
cast	825
country	831
date_added	10
release_year	0
rating	4
duration	3
listed_in	0
description	0

dtype: int64

```
df['cast']
```

	cast
0	NaN
1	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...
2	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...
3	NaN
4	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...
...	...
5392	Kate Higgins, Sean Hankinson, Haviland Stillwe...
5393	Jeanette Aw, Elvin Ng, Zhou Ying, Christopher ...
5394	Hans Teeuwen
5395	İbrahim Çelikkol, Belçim Bilgin, Alican Yüceso...
5396	Raaj Kumar, Hema Malini, Rakhee Gulzar, Vinod ...

5397 rows × 1 columns

dtype: object

```
temp = df[['type','cast']].set_index('type')
temp
```

type	cast
Movie	NaN
TV Show	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...
TV Show	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...
TV Show	NaN
TV Show	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...
...	...
Movie	Mark Ruffalo, Jake Gyllenhaal, Robert Downey J...
TV Show	NaN
Movie	Jesse Eisenberg, Woody Harrelson, Emma Stone, ...
Movie	Tim Allen, Courteney Cox, Chevy Chase, Kate Ma...
Movie	Vicky Kaushal, Sarah-Jane Dias, Raaghav Chanan...

8806 rows × 1 columns

Next steps: Generate code with temp | View recommended plots | New interactive sheet

```
temp['cast']
```

type	cast
Movie	NaN
TV Show	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...
TV Show	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...
TV Show	NaN
TV Show	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...
...	...
Movie	Mark Ruffalo, Jake Gyllenhaal, Robert Downey J...
TV Show	NaN
Movie	Jesse Eisenberg, Woody Harrelson, Emma Stone, ...
Movie	Tim Allen, Courteney Cox, Chevy Chase, Kate Ma...
Movie	Vicky Kaushal, Sarah-Jane Dias, Raaghav Chanan...

8806 rows × 1 columns

dtype: object

```
x = temp['cast'].explode().reset_index()
x
```

	type	cast
0	Movie	NaN
1	TV Show	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...
2	TV Show	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...
3	TV Show	NaN
4	TV Show	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...
...	...	...
8801	Movie	Mark Ruffalo, Jake Gyllenhaal, Robert Downey J...
8802	TV Show	NaN
8803	Movie	Jesse Eisenberg, Woody Harrelson, Emma Stone, ...
8804	Movie	Tim Allen, Courteney Cox, Chevy Chase, Kate Ma...
8805	Movie	Vicky Kaushal, Sarah-Jane Dias, Raaghav Chanan...

8806 rows × 2 columns

Next steps: Generate code with x | View recommended plots | New interactive sheet

```
x['cast'].dropna().value_counts()
```

cast	count
David Attenborough	19
Vatsal Dubey, Julie Tejwani, Rupa Bhimani, Jigna Bhardwaj, Rajesh Kava, Mousam, Swapnil	14
Samuel West	10
Jeff Dunham	7
Michela Luci, Jamie Watson, Eric Peterson, Anna Claire Bartlam, Nicolas Aqui, Cory Doran, Julie Lemieux, Derek McGrath	6
...	...
Toyin Abraham, Sambasa Nzeribe, Chioma Chukwuka Akpotha, Chioma Omeruah, Chiwetalu Agu, Dele Odule, Femi Adebayo, Bayray McNwizu, Biodun Stephen	1
Neeraj Kabi, Geetanjali Kulkarni, Danish Husain, Sheeba Chaddha, Paras Priyadarshan, Anshul Chauhan, Anud Singh Dhaka, Shirin Sewani, Mihir Ahuja, Vasundhara Rajput	1
Sanjay Dutt, Arjun Kapoor, Kriti Sanon, Zeenat Aman, Mohnish Bahl, Padmini Kolhapure, Kunal Kapoor, Suhasini Mulay	1
Lika Berning, Bobby van Jaarsveld, Marlee van der Merwe, Sonja Herholdt, Elize Cawood, Rouel Beukes, Kevin Leo, Paul du Toit, Sylvaine Strike	1
Della Dartyan, Adipati Dolken, Ratna Riantiarno, Ariyo Wahab, Bastian Steel, Gading Marten, Putri Ayudya, Taskya Namya, Egi Fedly, Yayu Unru, Abdurrahman Arif	1

```
x = x.dropna()
```

```
x.isnull().sum()
```

	0
type	0
cast	0

dtype: int64

```
x
```

	type	cast
1	TV Show	Ama Qamata
2	TV Show	Khosi Ngema
3	TV Show	Gail Mabalane
4	TV Show	Thabang Molaba
5	TV Show	Dillon Windvogel
...	...	...
64943	Movie	Manish Chaudhary
64944	Movie	Meghna Malik
64945	Movie	Malkeet Rauni
64946	Movie	Anita Shabdish
64947	Movie	Chittaranjan Tripathy

64123 rows × 2 columns

```
type_cast_count = x[['type','cast']].value_counts().reset_index(name = 'count')
type_cast_count
```

	type	cast	count
0	TV Show	David Attenborough	14
1	Movie	Vatsal Dubey, Julie Tejwani, Rupa Bhimani, Jig...	13
2	Movie	Samuel West	10
3	Movie	Jeff Dunham	7
4	Movie	Kevin Hart	6
...	...	...	...
7723	Movie	Justin Bieber, Ludacris, Usher Raymond, Jaden ...	1
7724	Movie	Junko Takeuchi, Noriaki Sugiyama, Chie Nakamur...	1
7725	Movie	Junko Takeuchi, Gamon Kaai, Chie Nakamura, Sho...	1
7726	Movie	Junko Takeuchi, Chie Nakamura, Yoichi Masukawa...	1
7727	Movie	Józef Pawłowski, Zofia Domalik, Szymon Bobrows...	1

7728 rows × 3 columns

Next steps: Generate code with type_cast_count View recommended plots New interactive sheet

```
top_by_type = type_cast_count.groupby('type').apply(lambda x : x.nlargest(5, 'count')).reset_in
top_by_type
```

```
<ipython-input-17-3da5d335083a>:1: DeprecationWarning: DataFrameGroupBy.apply operated on the grouping columns. This behavior is dep
  top_by_type = type_cast_count.groupby('type').apply(lambda x : x.nlargest(5, 'count')).reset_index(drop = True)
```

	type	cast	count
0	Movie	Vatsal Dubey, Julie Tejwani, Rupa Bhimani, Jig...	13
1	Movie	Samuel West	10
2	Movie	Jeff Dunham	7
3	Movie	Kevin Hart	6
4	Movie	Craig Sechler	6
5	TV Show	David Attenborough	14
6	TV Show	Michela Luci, Jamie Watson, Anna Claire Bartla...	4
7	TV Show	Dave Chappelle	3
8	TV Show	Marie Kondo	2
9	TV Show	Mattea Conforti, Kobi Frumer	2

Next steps: Generate code with top_by_type View recommended plots New interactive sheet

in movies most frequent cast is 'Anupam Kher' and in tv shows most frequent cast is 'Takahiro Sakurai' so we can replace null values with them.

```
# for simplicy currently i'm replacing all nan values in director column with unknown
df['cast'] = df['cast'].fillna('unknown')
```

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

	0
show_id	0
type	0
title	0
director	0
cast	0
country	831
date_added	10
release_year	0
rating	4
duration	3
listed_in	0
description	0

dtype: int64

```
df['country'].value_counts()
```

country	count
United States	2818
India	972
United Kingdom	418
Japan	245
South Korea	199
...	...
Mexico, United States, Spain, Colombia	1
Canada, Norway	1
Finland, Germany, Belgium	1
Argentina, United States, Mexico	1
United Kingdom, United States, Germany, Denmark, Belgium, Japan	1

748 rows × 1 columns

dtype: int64

```
# us is the most frequent country so lets fill all null values in country column with us
df['country'] = df['country'].fillna('United States')

df['date_added'].value_counts()
```

date_added	count
January 1, 2020	109
November 1, 2019	89
March 1, 2018	75
December 31, 2019	74
October 1, 2018	71
...	...
February 2, 2017	1
September 11, 2019	1
May 17, 2015	1
June 5, 2018	1
October 14, 2017	1

1767 rows × 1 columns

dtype: int64

```
df['date_added'] = df['date_added'].fillna('January 1, 2020')       # fillin null values in dat

df[['type','rating']].value_counts()
```

type	rating	count
Movie	TV-MA	2062
	TV-14	1427
TV Show	TV-MA	1145
Movie	R	797
TV Show	TV-14	733
Movie	TV-PG	540
	PG-13	490
TV Show	TV-PG	323
Movie	PG	287
TV Show	TV-Y7	195
	TV-Y	175
Movie	TV-Y7	139
	TV-Y	131
	TV-G	126
TV Show	TV-G	94
Movie	NR	75
	G	41
	TV-Y7-FV	5
TV Show	NR	5
Movie	UR	3
	NC-17	3
TV Show	R	2
Movie	84 min	1
	74 min	1
	66 min	1
TV Show	TV-Y7-FV	1

dtype: int64

```
# for both movie and tv shows most frequent rating is TV-MA  lets fill all null values with tha
df['rating'] = df['rating'].fillna('TV-MA')
```

```
df['duration'].value_counts()
```

duration	count
1 Season	1792
2 Seasons	425
3 Seasons	199
90 min	152
97 min	146
...	...
228 min	1
18 min	1
205 min	1
201 min	1
191 min	1

220 rows × 1 columns

dtype: int64

```
# 1 season is the most frequent duration lets fill all null values in duration column with that
df['duration'] = df['duration'].fillna('1 Season')
```

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

	0
show_id	0
type	0
title	0
director	0
cast	0
country	0
date_added	0
release_year	0
rating	0
duration	0
listed_in	0
description	0

dtype: int64

now there is no null values present in the data.

```
df.head()
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	unknown	United States	September 25, 2021	2020	PG-13	90 min	Documentaries	As her father nears the end of his life, filmm...
1	s2	TV Show	Blood & Water	unknown	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	After crossing paths at a party, a Cape Town t...
					Sami Bouajila						Crime TV	

```
df['date_added'] = df['date_added'].str.strip()

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

	date_added
0	2021-09-25
1	2021-09-24
2	2021-09-24
3	2021-09-24
4	2021-09-24
...	...
8802	2019-11-20
8803	2019-07-01
8804	2019-11-01
8805	2020-01-11
8806	2019-03-02

8806 rows × 1 columns

dtype: datetime64[ns]

```
df.head()
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	unknown	United States	2021-09-25	2020	PG-13	90 min	Documentaries	As her father nears the end of his life, filmm...
1	s2	TV Show	Blood & Water	unknown	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	2021-09-24	2021	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	After crossing paths at a party, a Cape Town t...
					Sami Bouajila						Crime TV	

```
Start coding or generate with AI.

Start coding or generate with AI.

#How has the number of movies released per year changed over the last 20-30 years?

df['release_year'].min(), df['release_year'].max()
```

```
(1925, 2021)
```

```
movies = df[df['type'] == 'Movie']
tv_shows = df[df['type'] == 'TV Show']

movies.head()
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	descriptio
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	unknown	United States	2021-09-25	2020	PG-13	90 min	Documentaries	As her fathe nears th end of hi life, filmm.
6	s7	Movie	My Little Pony: A New Generation	Robert Cullen, José Luis Ucha	Vanessa Hudgens, Kimiko Glenn, James Marsden, ...	United States	2021-09-24	2021	PG	91 min	Children & Family Movies	Equestria' divided. Bu a bright-eye hero be.
					Kofi	United					Dramas	On a phot

```
movies.shape
```

```
(6131, 12)
```

```
tv_shows.head()
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
1	s2	TV Show	Blood & Water	unknown	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	2021-09-24	2021	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	After crossing paths at a party, a Cape Town t...
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi	United States	2021-09-24	2021	TV-MA	1 Season	Crime TV Shows, International TV Shows, TV Act...	To protect his family from a powerful drug lor...

```
tv_shows.shape
```

```
(2675, 12)
```

```
movies['release_year'].max()
```

```
2021
```

```
past_30_years_movie_data = movies[movies['release_year'] >= 1991]
past_30_years_movie_data
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	descript
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	unknown	United States	2021-09-25	2020	PG-13	90 min	Documentaries	As her fa nears end o life, film
6	s7	Movie	My Little Pony: A New Generation	Robert Cullen, José Luis Ucha	Vanessa Hudgens, Kimiko Glenn, James Marsden, ...	United States	2021-09-24	2021	PG	91 min	Children & Family Movies	Equest divided. a bright-e hero
7	s8	Movie	Sankofa	Haile Gerima	Kofi Ghanaba, Oyafunmike Ogunlano, Alexandra D...	United States, Ghana, Burkina Faso, United Kin...	2021-09-24	1993	TV-MA	125 min	Dramas, Independent Movies, International Movies	On a p sho Ghana Amer mode
9	s10	Movie	The Starling	Theodore Melfi	Melissa McCarthy, Chris O'Dowd, Kevin Kline, T...	United States	2021-09-24	2021	PG-13	104 min	Comedies, Dramas	A wo adjustir life aft conte
12	s13	Movie	Je Suis Karl	Christian Schwochow	Luna Wedler, Jannis Niewöhner,	Germany, Czech Republic	2021-09-23	2021	TV-MA	127 min	Dramas, International Movies	After mo her fam murdere

Next steps: Generate code with past_30_years_movie_data | View recommended plots | New interactive sheet

```
movie_counts = past_30_years_movie_data.groupby('release_year').size().reset_index(name = 'coun
movie_counts
```

	release_year	count
0	1991	16
1	1992	20
2	1993	24
3	1994	20
4	1995	23
5	1996	21
6	1997	34
7	1998	32
8	1999	32
9	2000	33
10	2001	40
11	2002	44
12	2003	51
13	2004	55
14	2005	67
15	2006	82
16	2007	74
17	2008	113
18	2009	118
19	2010	154
20	2011	145
21	2012	173
22	2013	225
23	2014	264
24	2015	398
25	2016	658
26	2017	767
27	2018	767
28	2019	633
29	2020	517
30	2021	277

Next steps: Generate code with movie_counts | View recommended plots | New interactive sheet

```
tv_shows['release_year'].max()
```

```
2021
```

```
past_30_years_tv_data = tv_shows[tv_shows['release_year'] >= 1991]
past_30_years_tv_data
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
1	s2	TV Show	Blood & Water	unknown	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	2021-09-24	2021	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	After crossing paths at a party, a Cape Town t...
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	United States	2021-09-24	2021	TV-MA	1 Season	Crime TV Shows, International TV Shows, TV Act...	To protect his family from a powerful drug lor...
3	s4	TV Show	Jailbirds New Orleans	unknown	unknown	United States	2021-09-24	2021	TV-MA	1 Season	Docuseries, Reality TV	Feuds, flirtations and toilet talk go down amo...
4	s5	TV Show	Kota Factory	unknown	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	2021-09-24	2021	TV-MA	2 Seasons	International TV Shows, Romantic TV Shows, TV ...	In a city of coaching centers known to train I...
5	s6	TV	Midnight	Mike	Kate Siegel, Zach Gilford,	United	2021-09-24	2021	TV-MA	1 Season	TV Dramas, TV Horror,	The arrival of a charismatic

Next steps: Generate code with past_30_years_tv_data | View recommended plots | New interactive sheet

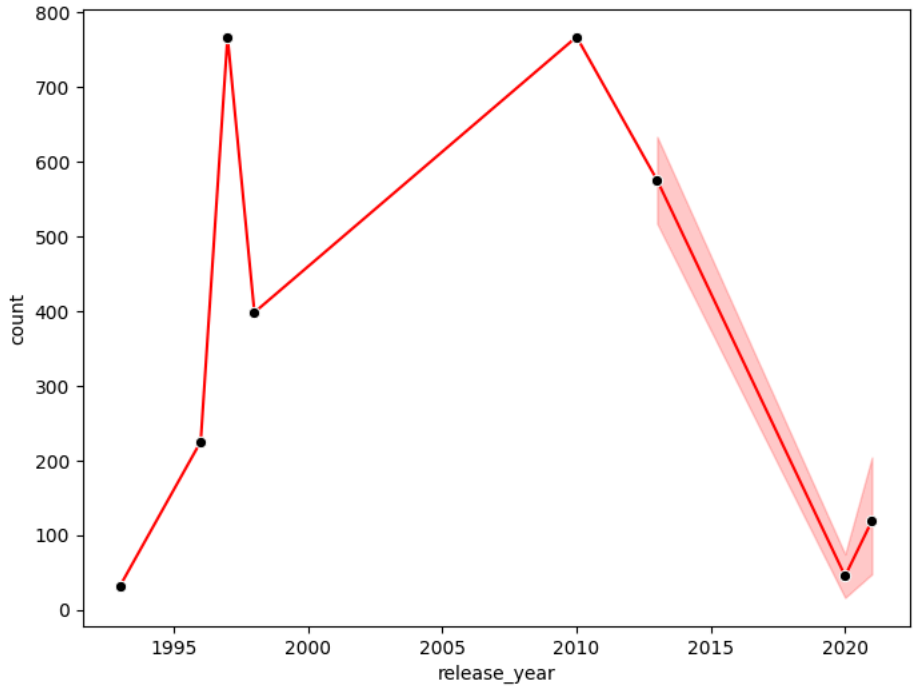
```
tv_counts = past_30_years_tv_data.groupby('release_year').size().reset_index(name='count')
tv_counts
```

	release_year	count
0	1991	1
1	1992	3
2	1993	4
3	1994	2
4	1995	2
5	1996	3
6	1997	4
7	1998	4
8	1999	7
9	2000	4
10	2001	5
11	2002	7
12	2003	10
13	2004	9
14	2005	13
15	2006	14
16	2007	14
17	2008	23
18	2009	34
19	2010	40
20	2011	40
21	2012	64
22	2013	63
23	2014	88
24	2015	161
25	2016	244
26	2017	265
27	2018	380
28	2019	397
29	2020	436
30	2021	315

Next steps: Generate code with tv_counts | View recommended plots | New interactive sheet

```
plt.figure(figsize = (8,6))
sns.lineplot(x = past_30_years_movie_data['release_year'], y = movie_counts['count'], marker =
```

```
<Axes: xlabel='release_year', ylabel='count'>
```




```
overall_past_30_years_data = df[df['release_year'] >= 1991]
overall_past_30_years_data
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	unknown	United States	2021-09-25	2020	PG-13	90 min	Documentaries	As her father nears the end of his life, filmm...
1	s2	TV Show	Blood & Water	unknown	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	2021-09-24	2021	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	After crossing paths at a party, a Cape Town t...
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	United States	2021-09-24	2021	TV-MA	1 Season	Crime TV Shows, International TV Shows, TV Act...	To protect his family from a powerful drug lor...
3	s4	TV Show	Jailbirds New Orleans	unknown	unknown	United States	2021-09-24	2021	TV-MA	1 Season	Docuseries, Reality TV	Feuds, flirtations and toilet talk go down amo...
4	s5	TV Show	Kota Factory	unknown	Mayur More, Jitendra Kumar, Ranjan	India	2021-09-24	2021	TV-MA	2 Seasons	International TV Shows, Romantic TV Shows, TV ...	In a city of coaching centers known to

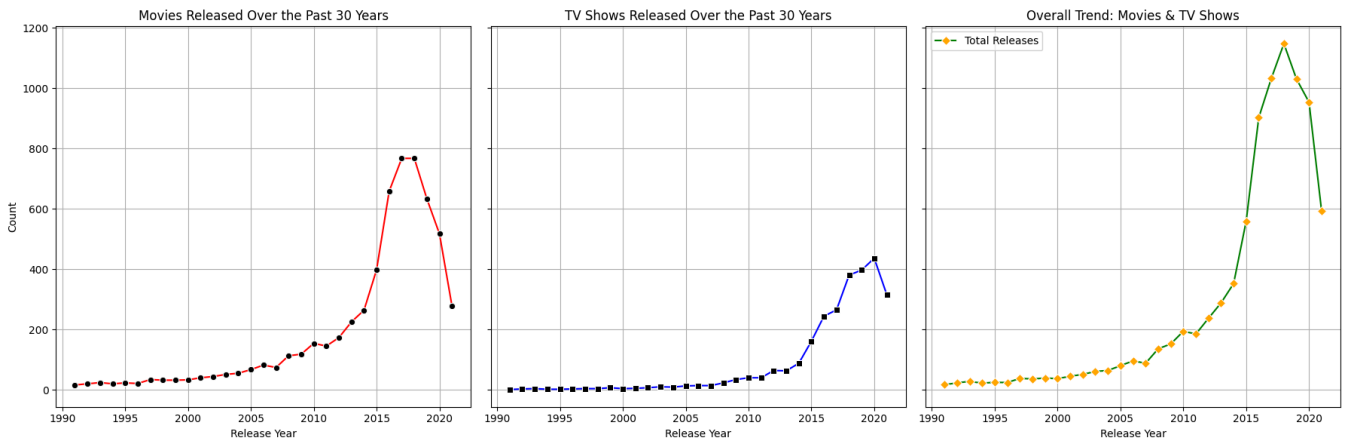
Next steps: Generate code with overall_past_30_years_data | View recommended plots | New interactive sheet

```
overall_data_count = overall_past_30_years_data.groupby('release_year').size().reset_index(name
overall_data_count
```

	release_year	total
0	1991	17
1	1992	23
2	1993	28
3	1994	22
4	1995	25
5	1996	24
6	1997	38
7	1998	36
8	1999	39
9	2000	37
10	2001	45
11	2002	51
12	2003	61
13	2004	64
14	2005	80
15	2006	96
16	2007	88
17	2008	136
18	2009	152
19	2010	194
20	2011	185
21	2012	237
22	2013	288
23	2014	352
24	2015	559
25	2016	902
26	2017	1032
27	2018	1147
28	2019	1030
29	2020	953
30	2021	592

Next steps: Generate code with overall_data_count | View recommended plots | New interactive sheet

```
fig, ax = plt.subplots(1, 3, figsize=(18, 6), sharex=True, sharey=True)

# Plot movies on the first axis
sns.lineplot(
    x=movie_counts['release_year'],
    y=movie_counts['count'],
    marker='o',
    color='r',
    markerfacecolor='black',
    ax=ax[0]
)
ax[0].set_title("Movies Released Over the Past 30 Years")
ax[0].set_xlabel("Release Year")
ax[0].set_ylabel("Count")
ax[0].grid(True)

# Plot TV shows on the second axis
sns.lineplot(
    x=tv_counts['release_year'],
    y=tv_counts['count'],
    marker='s',  # Square marker for TV shows
    color='b',
    markerfacecolor='black',
    ax=ax[1]
)
ax[1].set_title("TV Shows Released Over the Past 30 Years")
ax[1].set_xlabel("Release Year")
ax[1].grid(True)

# Plot overall trend on the third axis
sns.lineplot(
    x=overall_data_count['release_year'],
    y=overall_data_count['total'],
    marker='D',  # Diamond marker for total count
    color='g',
    markerfacecolor='orange',
    label="Total Releases",
    ax=ax[2]
)

ax[2].set_title("Overall Trend: Movies & TV Shows")
ax[2].set_xlabel("Release Year")
ax[2].legend()
ax[2].grid(True)

# Adjust layout for better spacing
plt.tight_layout()

# Show the plot
plt.show()
```



- in movie releasing trend we can observe that from 1991 to 2015 movie release increased gradually but from 2010 to 2017 movie release increased rapidly the after 2018 it's showing rapidly decreasing trend.

- in tv shows releasing trend we can observe from 1991 to 2006 - 07 its almost constant but after 2010 - 2020 its showing increasing trend and after 2020 it's again showing decreasing trend.

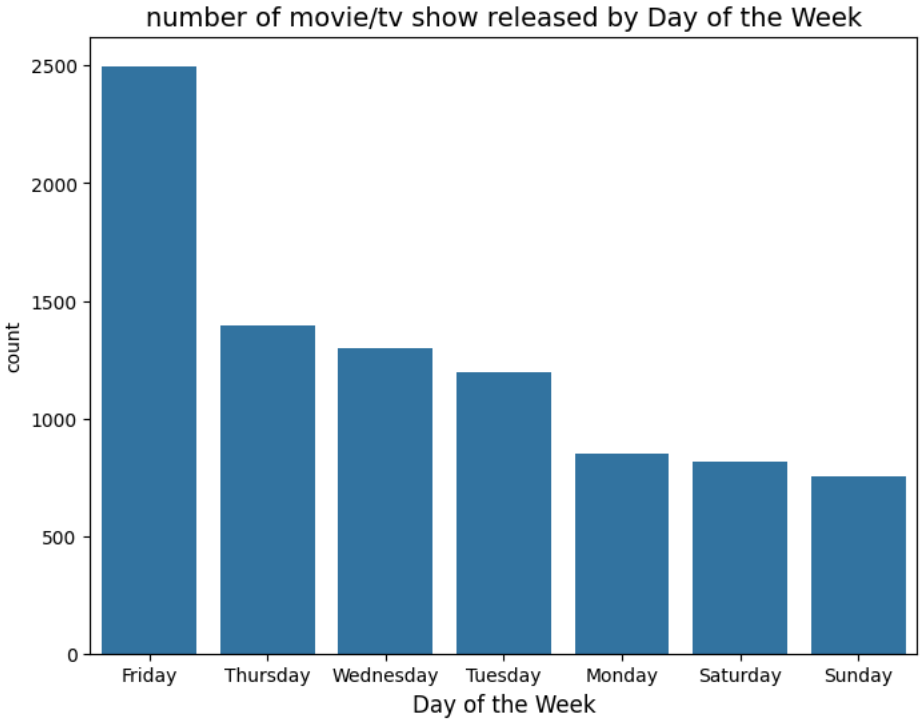
- overall trend is gradually increasing from 1991 to 2010 then it started increasing rapidly till 2018 and after then it shows sudden fall.

```
Start coding or generate with AI.
```

```
#What is the best time to launch a TV show?

df['date_added'].dt.day_name().value_counts().index
```

```
Index(['Friday', 'Thursday', 'Wednesday', 'Tuesday', 'Monday', 'Saturday',
       'Sunday'],
      dtype='object', name='date_added')
```

```
plt.figure(figsize = (8,6))
sns.countplot(x = df['date_added'].dt.day_name(), order = df['date_added'].dt.day_name().value_

plt.title('number of movie/tv show released by Day of the Week', fontsize=14)
plt.xlabel('Day of the Week', fontsize=12)
plt.show()
```

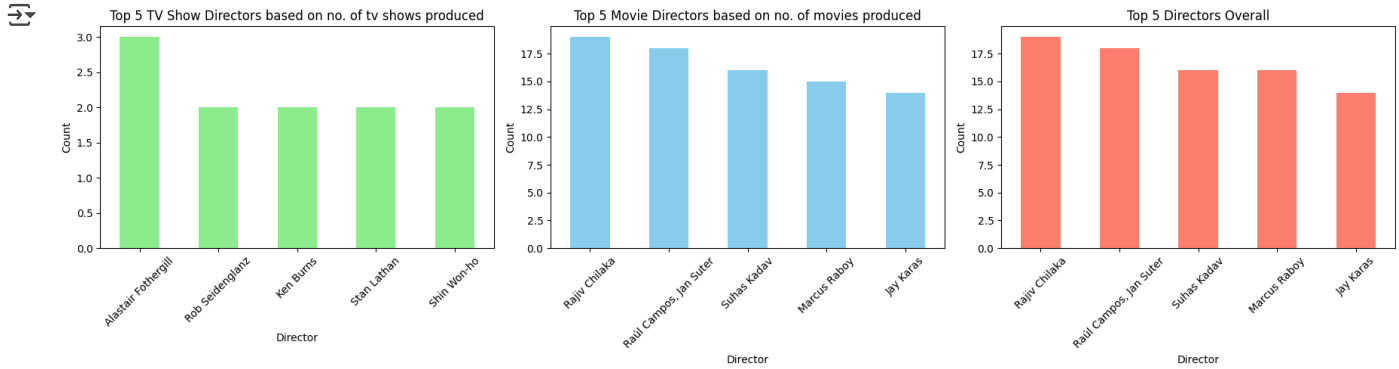


maximum movies and tv shows are released on friday so friday is the best to release any new movie ot tv show.

```
Start coding or generate with AI.
```

```
#Analysis of actors/directors of different types of shows/movies.
```

```
movie_directors = movies['director'].value_counts().iloc[1:6]
tv_directors = tv_shows['director'].value_counts().iloc[1:6]
overall_directors = df['director'].value_counts().iloc[1:6]

# Create subplots
fig, axes = plt.subplots(1, 3, figsize=(18, 5))

# TV Show Directors
tv_directors.plot(kind='bar', ax=axes[0], color='lightgreen')
axes[0].set_title('Top 5 TV Show Directors based on no. of tv shows produced')
axes[0].set_xlabel('Director')
axes[0].set_ylabel('Count')
axes[0].tick_params(axis='x', rotation=45)

# Movie Directors
movie_directors.plot(kind='bar', ax=axes[1], color='skyblue')
axes[1].set_title('Top 5 Movie Directors based on no. of movies produced')
axes[1].set_xlabel('Director')
axes[1].set_ylabel('Count')
axes[1].tick_params(axis='x', rotation=45)

# Overall Directors
overall_directors.plot(kind='bar', ax=axes[2], color='salmon')
axes[2].set_title('Top 5 Directors Overall')
axes[2].set_xlabel('Director')
axes[2].set_ylabel('Count')
axes[2].tick_params(axis='x', rotation=45)

# Final touches
plt.tight_layout()
plt.show()
```




```
movie_cast = movies['cast'].str.split(',').map(lambda x : [i.strip() for i in x]).explode().res
movie_cast
```

	cast
0	unknown
1	Vanessa Hudgens
2	Kimiko Glenn
3	James Marsden
4	Sofia Carson
...	...
44945	Manish Chaudhary
44946	Meghna Malik
44947	Malkeet Rauni
44948	Anita Shabdish
44949	Chittaranjan Tripathy

44950 rows × 1 columns

dtype: object

```
tv_cast = tv_shows['cast'].str.split(',').map(lambda x : [i.strip() for i in x]).explode().rese
tv_cast
```

	cast
0	Ama Qamata
1	Khosi Ngema
2	Gail Mabalane
3	Thabang Molaba
4	Dillon Windvogel
...	...
19993	Samina Peerzada
19994	Waseem Abbas
19995	Javed Sheikh
19996	Hina Khawaja Bayat
19997	unknown

19998 rows × 1 columns

dtype: object

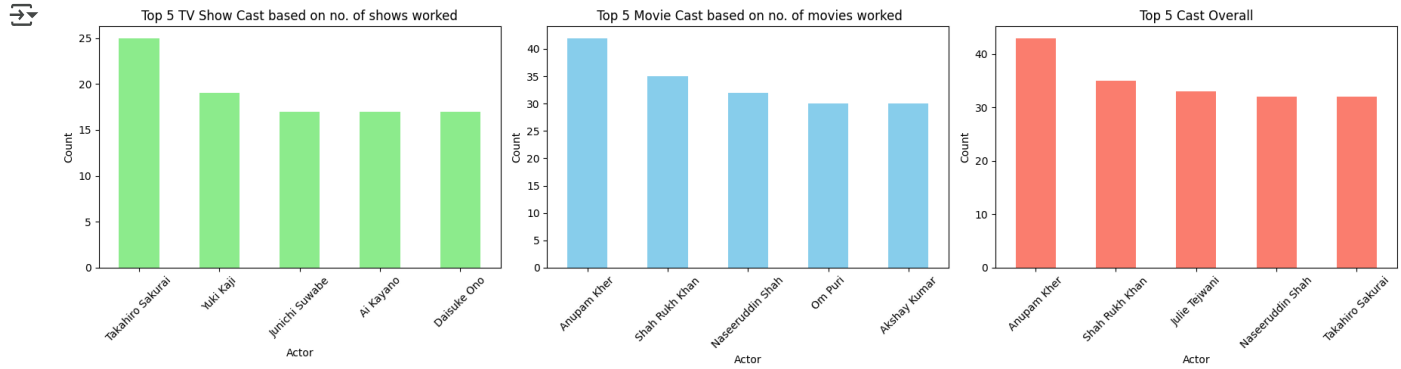
```
overall_cast = df['cast'].str.split(',').map(lambda x : [i.strip() for i in x]).explode().reset
overall_cast
```

	cast
0	unknown
1	Ama Qamata
2	Khosi Ngema
3	Gail Mabalane
4	Thabang Molaba
...	...
64943	Manish Chaudhary
64944	Meghna Malik
64945	Malkeet Rauni
64946	Anita Shabdish
64947	Chittaranjan Tripathy

64948 rows × 1 columns

dtype: object

```
top_movie_cast = movie_cast.value_counts().iloc[1:6]
top_tv_cast = tv_cast.value_counts().iloc[1:6]
top_overall_cast = overall_cast.value_counts().iloc[1:6]

# Step 3: Plotting subplots
fig, axes = plt.subplots(1, 3, figsize=(18, 5))

# TV Shows
top_tv_cast.plot(kind='bar', ax=axes[0], color='lightgreen')
axes[0].set_title('Top 5 TV Show Cast based on no. of shows worked')
axes[0].set_xlabel('Actor')
axes[0].set_ylabel('Count')
axes[0].tick_params(axis='x', rotation=45)

# Movies
top_movie_cast.plot(kind='bar', ax=axes[1], color='skyblue')
axes[1].set_title(' Top 5 Movie Cast based on no. of movies worked')
axes[1].set_xlabel('Actor')
axes[1].set_ylabel('Count')
axes[1].tick_params(axis='x', rotation=45)

# Overall
top_overall_cast.plot(kind='bar', ax=axes[2], color='salmon')
axes[2].set_title('Top 5 Cast Overall')
axes[2].set_xlabel('Actor')
axes[2].set_ylabel('Count')
axes[2].tick_params(axis='x', rotation=45)

# Layout adjustment
plt.tight_layout()
plt.show()
```



'Anupam Kher' is casted on most of the movies. and 'Takahiro Sakurai' is casted in most of the tv shows.

Start coding or generate with AI.

#Does Netflix has more focus on TV Shows than movies in recent years

```
#count of movie released in recent 10 years
movie_count = movies[movies['release_year'] > 2010]
movie_count.shape[0]
```

```
4824
```

```
#count of tv shows released in recent 10 years
tv_show_count = tv_shows[tv_shows['release_year'] > 2010]
tv_show_count.shape[0]
```

```
2453
```

in recent 10 years no. of movies produce is more then no. of tv shows produce so netflix has more focuse on movies rather then tv shows.

```
Start coding or generate with AI.
```

```
#Understanding what content is available in different countries

df['country'].unique()
```

```
        'Norway, Denmark', 'Syria, France, Lebanon, Qatar',
```

```
countries = df['country'].str.split(',').map( lambda x : [i.strip() for i in x]).explode().rese
countries
```

	country
0	United States
1	South Africa
2	United States
3	United States
4	India
...	...
10844	United States
10845	United States
10846	United States
10847	United States
10848	India

10849 rows × 1 columns

dtype: object

```
countries.value_counts()
```

country	count
United States	4521
India	1046
United Kingdom	805
Canada	445
France	393
...	...
Sudan	1
Panama	1
Uganda	1
East Germany	1
Montenegro	1

123 rows × 1 columns

dtype: int64

```
top_movie_release_country = movies['country'].str.split(',').map( lambda x : [i.strip() for i i
top_movie_release_country
```

	country
0	United States
1	United States
2	United States
3	Ghana
4	Burkina Faso
...	...
7814	Jordan
7815	United States
7816	United States
7817	United States
7818	India

7819 rows × 1 columns

dtype: object

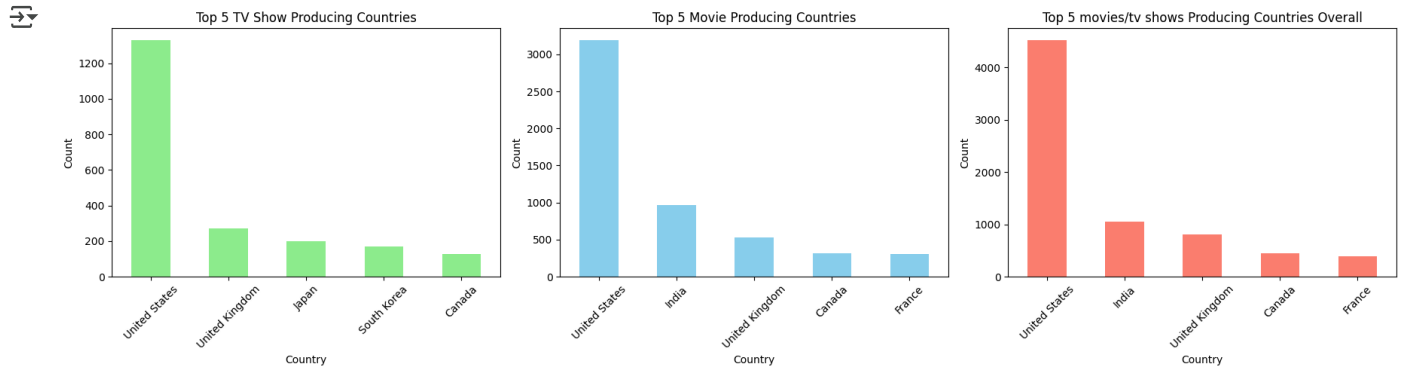
```
top_tv_shows_release_country = tv_shows['country'].str.split(',').map( lambda x : [i.strip() fo
top_tv_shows_release_country
```

	country
0	South Africa
1	United States
2	United States
3	India
4	United States
...	...
3025	France
3026	South Korea
3027	Indonesia
3028	Pakistan
3029	United States

3030 rows × 1 columns

dtype: object

```
tv_country_counts = top_tv_shows_release_country.value_counts().head(5)
movie_country_counts = top_movie_release_country.value_counts().head(5)
overall_country_counts = countries.value_counts().head(5)

# Step 3: Create subplots
fig, axes = plt.subplots(1, 3, figsize=(18, 5))

# TV Show Countries
tv_country_counts.plot(kind='bar', ax=axes[0], color='lightgreen')
axes[0].set_title('Top 5 TV Show Producing Countries')
axes[0].set_xlabel('Country')
axes[0].set_ylabel('Count')
axes[0].tick_params(axis='x', rotation=45)

# Movie Countries
movie_country_counts.plot(kind='bar', ax=axes[1], color='skyblue')
axes[1].set_title('Top 5 Movie Producing Countries')
axes[1].set_xlabel('Country')
axes[1].set_ylabel('Count')
axes[1].tick_params(axis='x', rotation=45)

# Overall Countries
overall_country_counts.plot(kind='bar', ax=axes[2], color='salmon')
axes[2].set_title('Top 5 movies/tv shows Producing Countries Overall')
axes[2].set_xlabel('Country')
axes[2].set_ylabel('Count')
axes[2].tick_params(axis='x', rotation=45)

# Layout fix
plt.tight_layout()
plt.show()
```



united states is the top movies/ tv show releasing country.

```
Start coding or generate with AI.
```

```
# what kin of movies eople like the most?
top_listed_movie = movies['listed_in'].str.split(',').explode().reset_index(drop = True)
top_listed_movie
```

	listed_in
0	Documentaries
1	Children & Family Movies
2	Dramas
3	Independent Movies
4	International Movies
...	...
13185	Children & Family Movies
13186	Comedies
13187	Dramas
13188	International Movies
13189	Music & Musicals

13190 rows × 1 columns

dtype: object

```
top_listed_tv_shows = tv_shows['listed_in'].str.split(',').explode().reset_index(drop = True)
top_listed_tv_shows
```

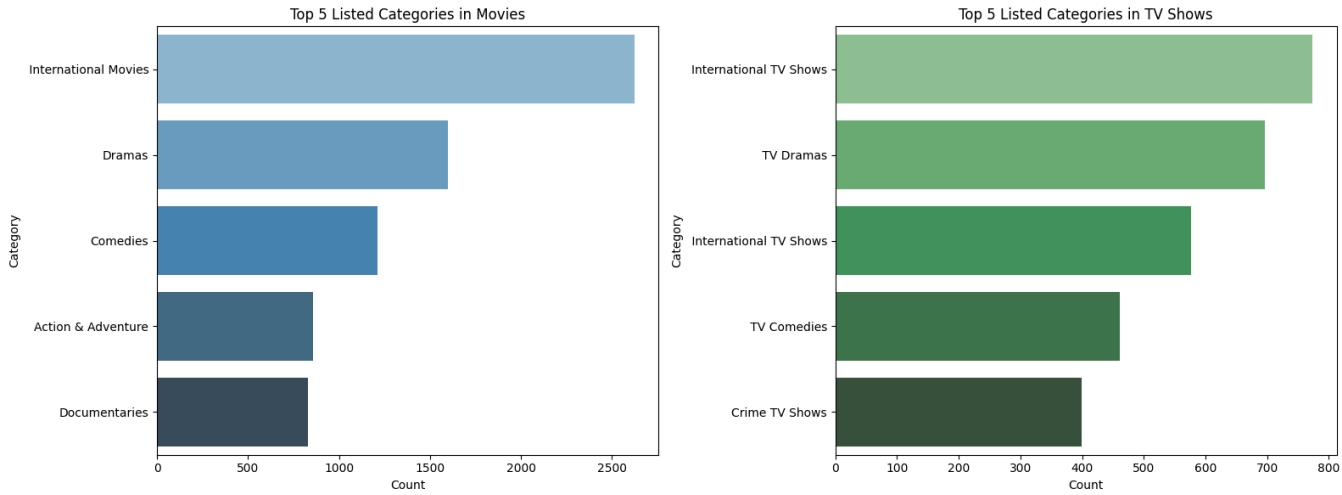
	listed_in
0	International TV Shows
1	TV Dramas
2	TV Mysteries
3	Crime TV Shows
4	International TV Shows
...	...
6126	Romantic TV Shows
6127	TV Dramas
6128	Kids' TV
6129	Korean TV Shows
6130	TV Comedies

6131 rows × 1 columns

dtype: object

```
top_5_movie = top_listed_movie.value_counts().head(5)
top_5_tv = top_listed_tv_shows.value_counts().head(5)

# Step 3: Plot using subplots
fig, axs = plt.subplots(1, 2, figsize=(16, 6))

# Movie plot
sns.barplot(x=top_5_movie.values, y=top_5_movie.index, ax=axs[0], palette='Blues_d')
axs[0].set_title("Top 5 Listed Categories in Movies")
axs[0].set_xlabel("Count")
axs[0].set_ylabel("Category")

# TV Show plot
sns.barplot(x=top_5_tv.values, y=top_5_tv.index, ax=axs[1], palette='Greens_d')
axs[1].set_title("Top 5 Listed Categories in TV Shows")
axs[1].set_xlabel("Count")
axs[1].set_ylabel("Category")

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

```
<ipython-input-56-fc5806e632f4>:8: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `y` variable to `hue` and set `le

  sns.barplot(x=top_5_movie.values, y=top_5_movie.index, ax=axs[0], palette='Blues_d')
<ipython-input-56-fc5806e632f4>:14: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `y` variable to `hue` and set `le

  sns.barplot(x=top_5_tv.values, y=top_5_tv.index, ax=axs[1], palette='Greens_d')
```



most released movies or tv_shows are international movies or international tv shows.

Start coding or generate with AI.

```
movies['duration'].str.extract('(\d+)').astype(int).mean()
```

	0
0	99.528951

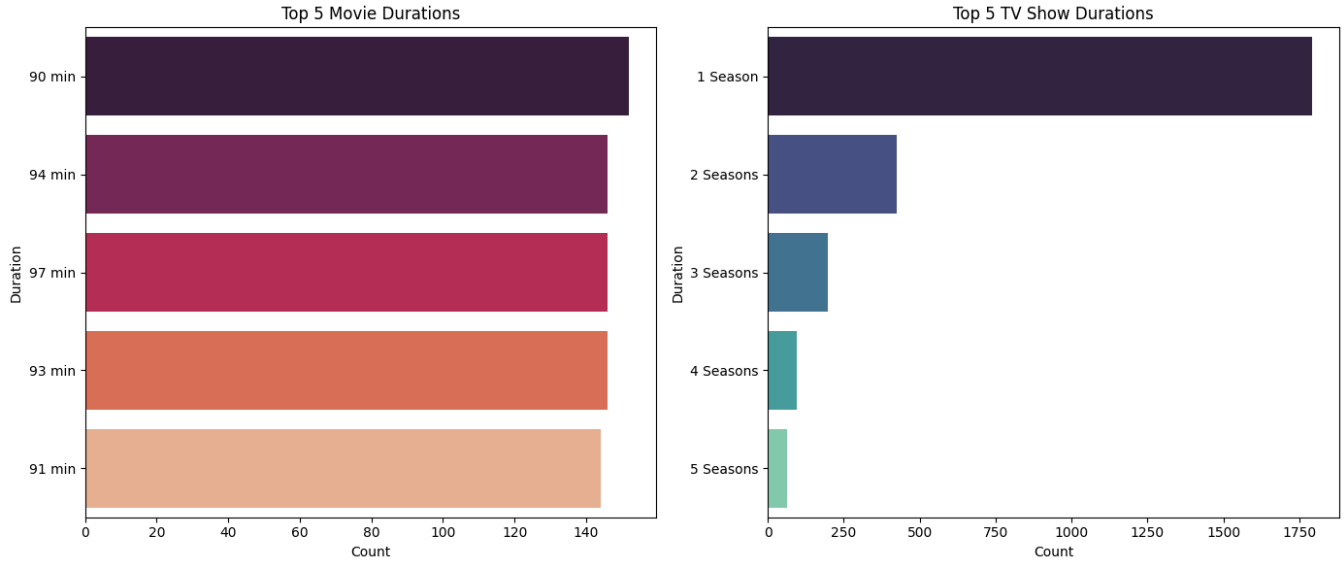
dtype: float64

Average movie duration is 100 min

```
movie_duration = movies['duration'].value_counts().head()
tv_show_duration = tv_shows['duration'].value_counts().head()
```

```
fig, axs = plt.subplots(1, 2, figsize=(14, 6))

# Bar plot for Movies
sns.barplot(x=movie_duration.values, y=movie_duration.index, ax=axs[0], palette='rocket')
axs[0].set_title('Top 5 Movie Durations')
axs[0].set_xlabel('Count')
axs[0].set_ylabel('Duration')

# Bar plot for TV Shows
sns.barplot(x=tv_show_duration.values, y=tv_show_duration.index, ax=axs[1], palette='mako')
axs[1].set_title('Top 5 TV Show Durations')
axs[1].set_xlabel('Count')
axs[1].set_ylabel('Duration')

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

```
<ipython-input-67-567ad3a8f774>:4: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `y` variable to `hue` and set `le

  sns.barplot(x=movie_duration.values, y=movie_duration.index, ax=axs[0], palette='rocket')
<ipython-input-67-567ad3a8f774>:10: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `y` variable to `hue` and set `le

  sns.barplot(x=tv_show_duration.values, y=tv_show_duration.index, ax=axs[1], palette='mako')
```



most of the movie released are of 90 min and most of tv shows released are of duration season 1.

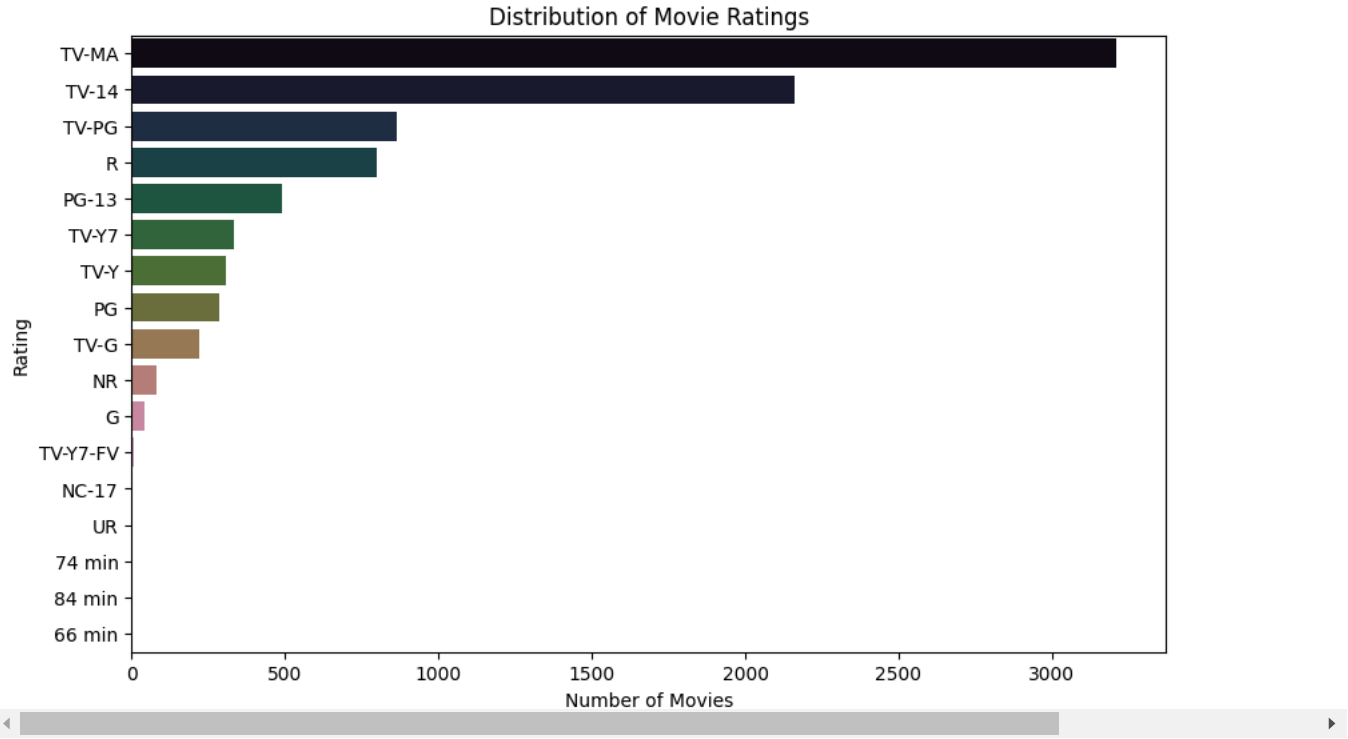
```
Start coding or generate with AI.
```

```
df['rating'].unique()
```

```
array(['PG-13', 'TV-MA', 'PG', 'TV-14', 'TV-PG', 'TV-Y', 'TV-Y7', 'R',
       'TV-G', 'G', 'NC-17', '74 min', '84 min', '66 min', 'NR',
       'TV-Y7-FV', 'UR'], dtype=object)
```

```
rating_counts = df['rating'].value_counts().sort_values(ascending=False)

# Plot
plt.figure(figsize=(10,6))
sns.barplot(x=rating_counts.values, y=rating_counts.index, palette='cubehelix')
plt.title("Distribution of Movie Ratings")
plt.xlabel("Number of Movies")
plt.ylabel("Rating")
plt.show()
```

```
<ipython-input-68-f5315142387e>:5: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `y` variable to `hue` and set `le

  sns.barplot(x=rating_counts.values, y=rating_counts.index, palette='cubehelix')
```



most of movies and tv shows got 'TV-MA' rating.

Start coding or generate with AI.

Netflix Movies & TV Shows Analysis Report

- Content Volume

Netflix has released more movies than TV shows overall.

- Most Productive Directors

Movies: Rajiv Chilaka has directed the highest number of movies.

TV Shows: Alastair Fothergill leads with the most TV show productions.

- Content Release Trend

Movies:

Gradual increase in releases from 1991 to 2015.

Rapid surge from 2010 to 2017.

Sharp decline in releases after 2018.

TV Shows:

Nearly constant release trend from 1991 to 2006-07.

Significant increase from 2010 to 2020.

Decline observed after 2020.

Overall: Consistent growth from 1991 to 2010, rapid increase until 2018, then sudden drop.

- Best Day to Release

Friday is the most common day for releasing both movies and TV shows — indicating it's likely the most effective release day for audience engagement.

- Most Featured Cast

Movies: Anupam Kher appears in the most movies.

TV Shows: Takahiro Sakurai is featured in the most TV shows.

- Recent Decade Focus

In the last 10 years, Netflix has focused more on movies than on TV shows.

- Top Country by Production

United States leads as the top country for producing both movies and TV shows on Netflix.

- Dominant Content Type

A significant portion of content is categorized as International Movies or International TV Shows.

- Typical Duration

Movies: Most movies have a duration of 90 minutes.

TV Shows: Most TV shows start with 1 Season.

- Content Rating

The most common rating across both movies and TV shows is 'TV-MA', indicating mature content dominates the platform.

Recommendations

- Rebalance Production Focus

Consider increasing investment in TV shows to balance content variety, especially as binge-watching culture grows.

- Revive Release Momentum

Reverse the post-2018 decline in releases with a strong comeback strategy — possibly with exclusive or regional content.

- Leverage Friday Releases

Continue to prioritize Friday releases for new content, and experiment with limited series drops mid-week to test engagement.

- Content for All Age Groups

With a majority of content rated TV-MA, introduce more family-friendly or teen-rated content to widen viewership demographics.

- Highlight Productive Talent

Promote popular directors (e.g., Rajiv Chilaka, Alastair Fothergill) and actors (e.g., Anupam Kher, Takahiro Sakurai) in marketing to boost viewer trust and anticipation.

- Duration-Based Curation

Curate special collections for short-duration content (e.g., "90-Minute Binge Night") and seasonal TV shows for weekend viewing.

- Strengthen International Presence

Double down on international content production, while also focusing on localization and dubbed/subtitled releases.

```
Start coding or generate with AI.
```

```
Start coding or generate with AI.
```

```
Start coding or generate with AI.
```

```
Start coding or generate with AI.
```

```
Start coding or generate with AI.
```

+ Code + Text

```
Start coding or generate with AI.
```

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.

Start coding or generate with AI.