

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

```
!gdown "https://d2beiqkhq929f0.cloudfront.net/public_assets/assets/000/000/940/original/netflix.csv" -O netflix.csv
```



Downloading...

From: https://d2beiqkhq929f0.cloudfront.net/public_assets/assets/000/000/940/original/netflix.csv

To: /content/netflix.csv

100% 3.40M/3.40M [00:08<00:00, 409kB/s]

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

```
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	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	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	To protect his

Next steps:

[Generate code with df](#)[View recommended plots](#)[New interactive sheet](#)

```
sr_m=df.loc[(df['release_year']>2000)& (df['type']=='Movie')].groupby('release_year')['show_id'].count()
df_m=sr_m.to_frame()
df_m.head()
```



	show_id
release_year	
2001	40
2002	44
2003	51
2004	55
2005	67

Next steps:

[Generate code with df_m](#)[View recommended plots](#)[New interactive sheet](#)

```
sr_t=df.loc[(df['release_year']>2000)&(df['type']=='TV Show')].groupby('release_year')['show_id'].count()
df_t=sr_t.to_frame()
df_t.head()
```



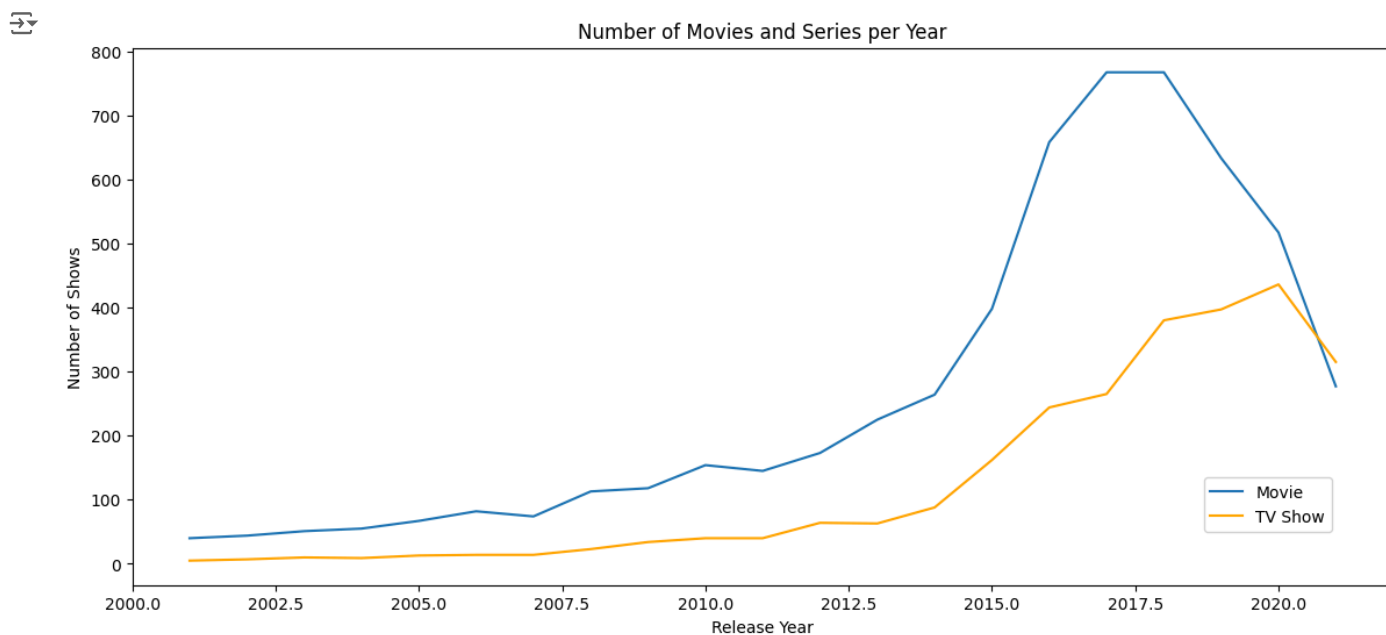
	show_id
release_year	
2001	5
2002	7
2003	10
2004	9
2005	13

Next steps:

[Generate code with df_t](#)[View recommended plots](#)[New interactive sheet](#)

```
plt.figure(figsize=(14,6))
sns.lineplot(x='release_year', y='show_id', data=df_m,label='Movie')
sns.lineplot(x='release_year', y='show_id', data=df_t, color='#FFA500', label='TV Show')
```

```
plt.xlabel('Release Year')
plt.ylabel('Number of Shows')
plt.title('Number of Movies and Series per Year')
#plt.xticks(rotation=45)
plt.legend(loc=(0.85,0.1), ncol=1)
plt.show()
```



```
df.head()
```

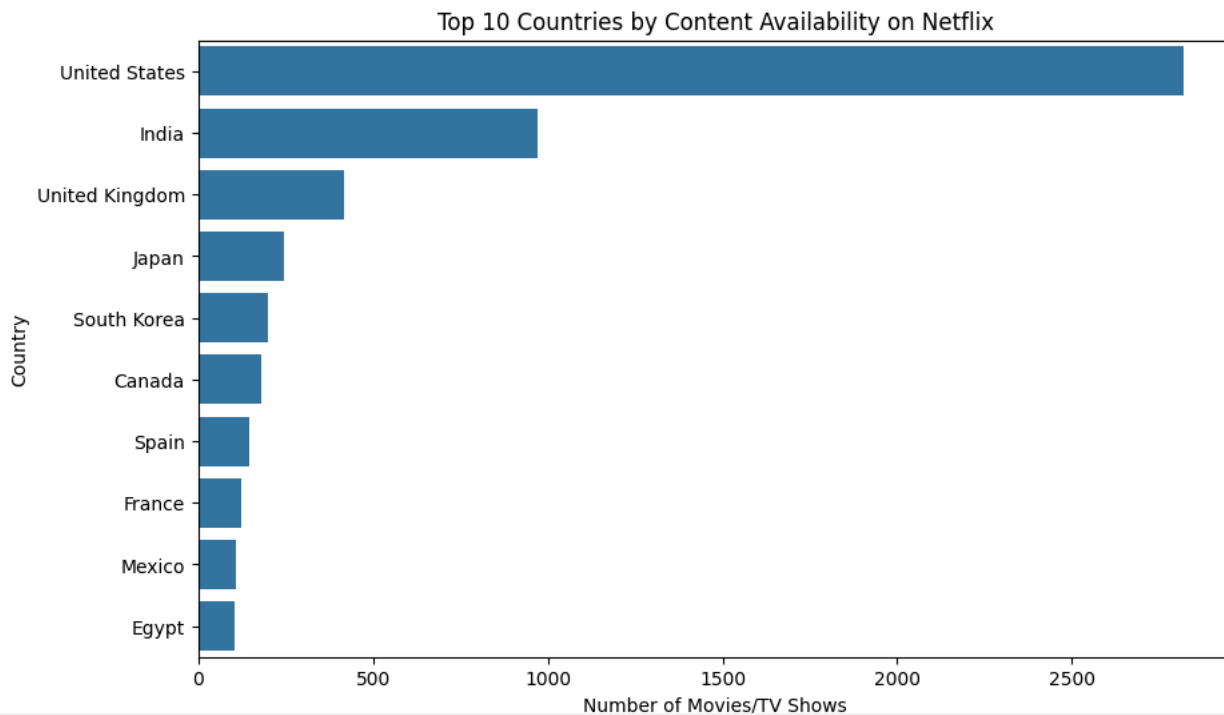
	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	Documentaries	As her father nears the end of his life, filmm...
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	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	To protect his

Next steps:

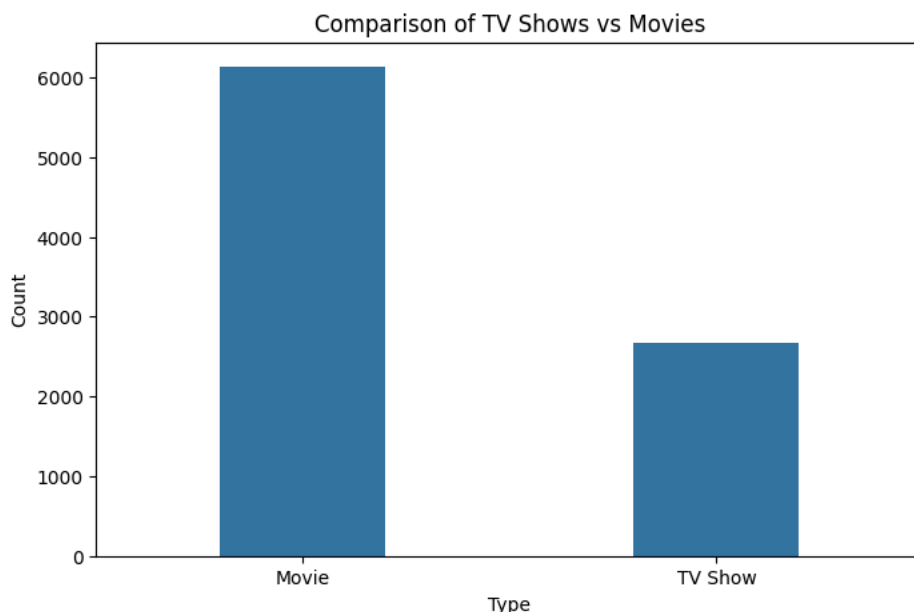
[Generate code with df](#)[View recommended plots](#)[New interactive sheet](#)

```
# Convert 'date_added' to datetime format and extract year
df['date_added'] = pd.to_datetime(df['date_added'], format='%B %d, %Y', errors='coerce')
df['year_added'] = df['date_added'].dt.year
```

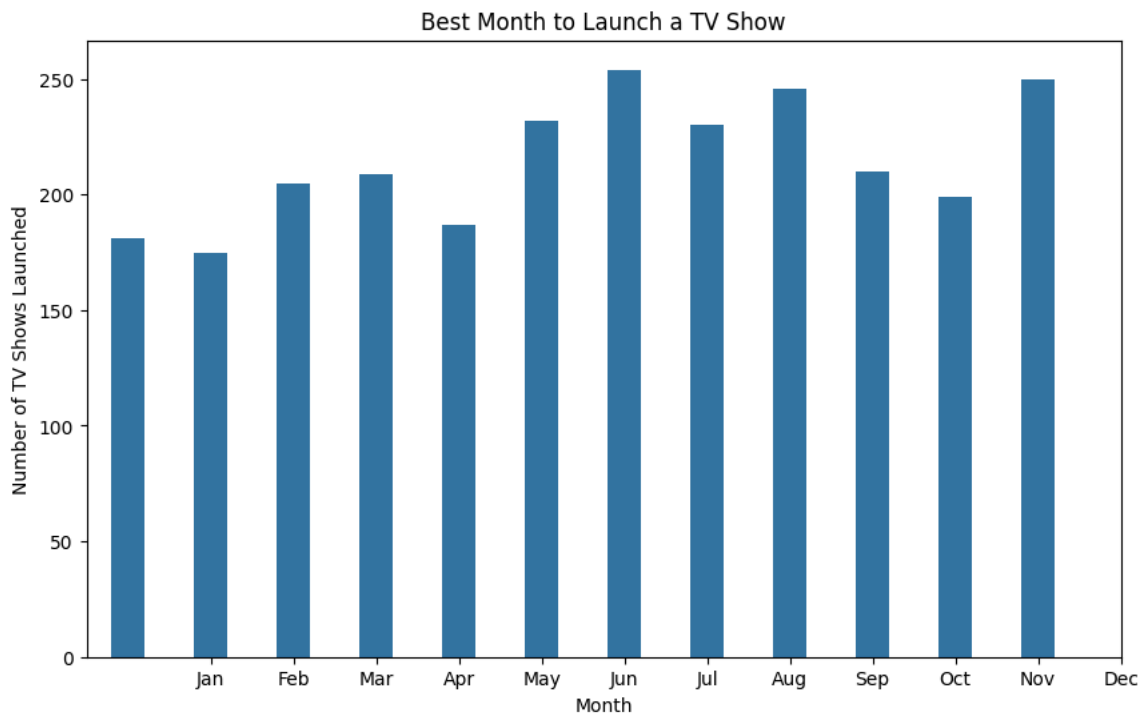
```
country_content = df['country'].value_counts().head(10) # Top 10 countries by content availability
plt.figure(figsize=(10, 6))
sns.barplot(x=country_content.values, y=country_content.index)
plt.title('Top 10 Countries by Content Availability on Netflix')
plt.xlabel('Number of Movies/TV Shows')
plt.ylabel('Country')
plt.show()
```



```
type_counts = df['type'].value_counts()
plt.figure(figsize=(8, 5))
sns.barplot(x=type_counts.index, y=type_counts.values,width=0.4)
plt.title('Comparison of TV Shows vs Movies')
plt.xlabel('Type')
plt.ylabel('Count')
plt.show()
```



```
df['month_added'] = df['date_added'].dt.month
tv_shows = df[df['type'] == 'TV Show']
tv_show_launch_months = tv_shows['month_added'].value_counts().sort_index()
plt.figure(figsize=(10, 6))
sns.barplot(x=tv_show_launch_months.index, y=tv_show_launch_months.values, width=0.4)
plt.title('Best Month to Launch a TV Show')
plt.xlabel('Month')
plt.ylabel('Number of TV Shows Launched')
plt.xticks(ticks=np.arange(1, 13), labels=['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec'])
plt.show()
```



```
# Correlation between Release of Movies and their rating
df = df.dropna(subset=['release_year', 'rating'])
df['release_year'] = df['release_year'].astype(int)
rating_mapping = {
    'G': 0,
    'TV-Y': 1,
    'TV-Y7': 2,
    'TV-G': 3,
    'TV-PG': 4,
    'PG': 5,
    'PG-13': 6,
    'R': 7,
    'NC-17': 8,
    'TV-MA': 9,
    'Unrated': 10
}
df['rating_numeric'] = df['rating'].map(rating_mapping)
```

<ipython-input-30-549639093986>:3: 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

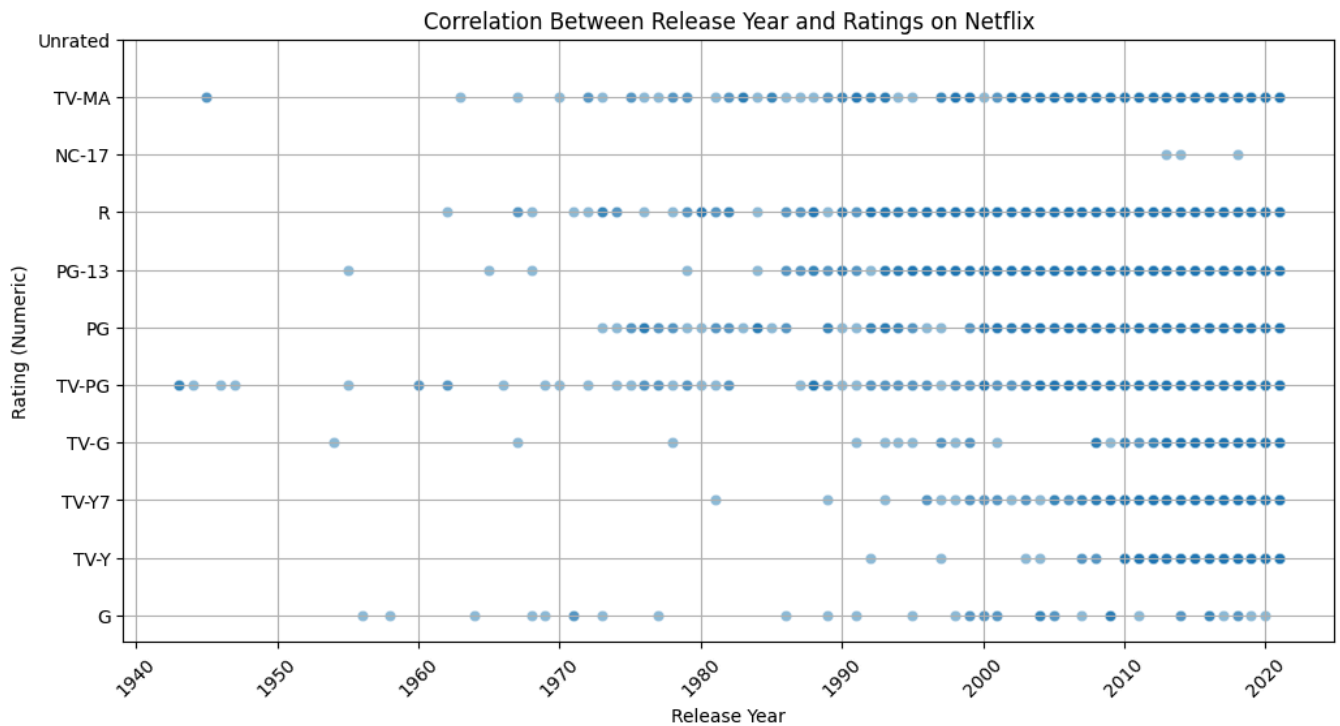
```
df['release_year'] = df['release_year'].astype(int)
```

<ipython-input-30-549639093986>:17: 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

```
df['rating_numeric'] = df['rating'].map(rating_mapping)
```

```
plt.figure(figsize=(12, 6))
sns.scatterplot(data=df, x='release_year', y='rating_numeric', alpha=0.5)
plt.title('Correlation Between Release Year and Ratings on Netflix')
plt.xlabel('Release Year')
plt.ylabel('Rating (Numeric)')
plt.xticks(rotation=45)
plt.yticks(list(rating_mapping.values()), list(rating_mapping.keys()))
plt.grid(True)
plt.show()
```



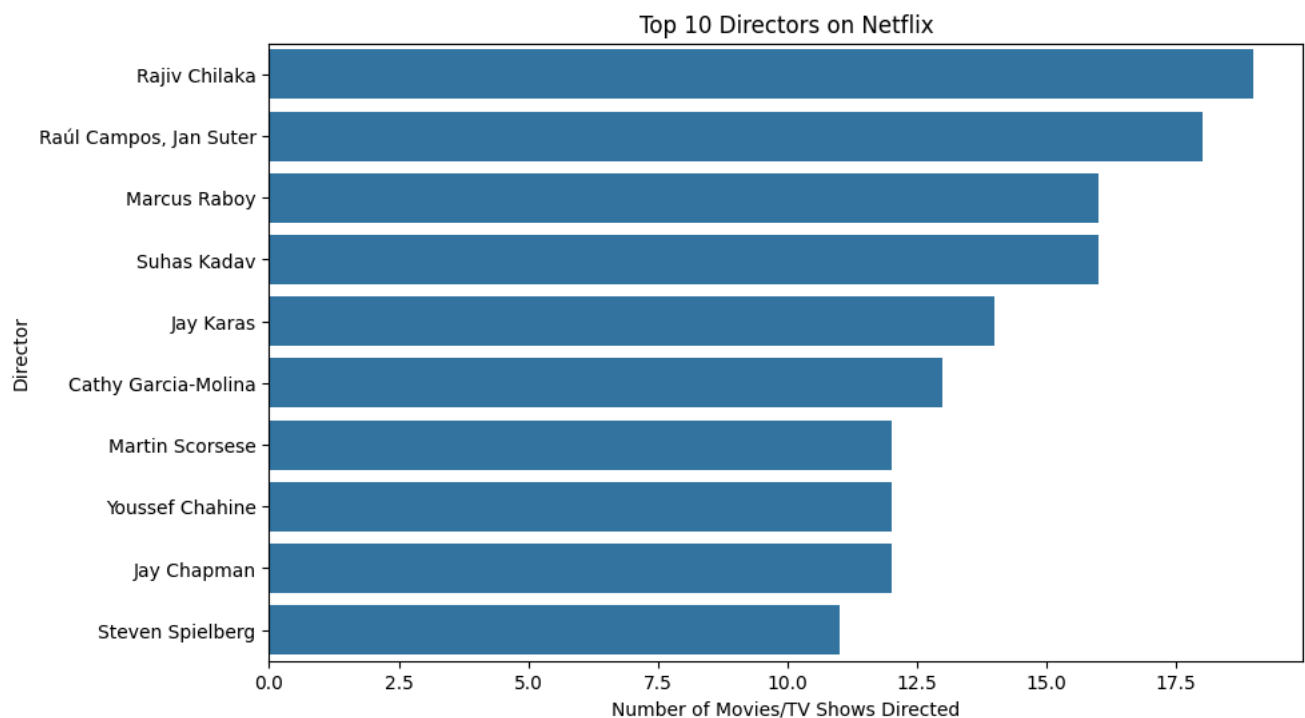
```
correlation = df['release_year'].corr(df['rating_numeric'])
print(f'Correlation coefficient between release year and ratings: {correlation:.2f}')
```



Correlation coefficient between release year and ratings: 0.13

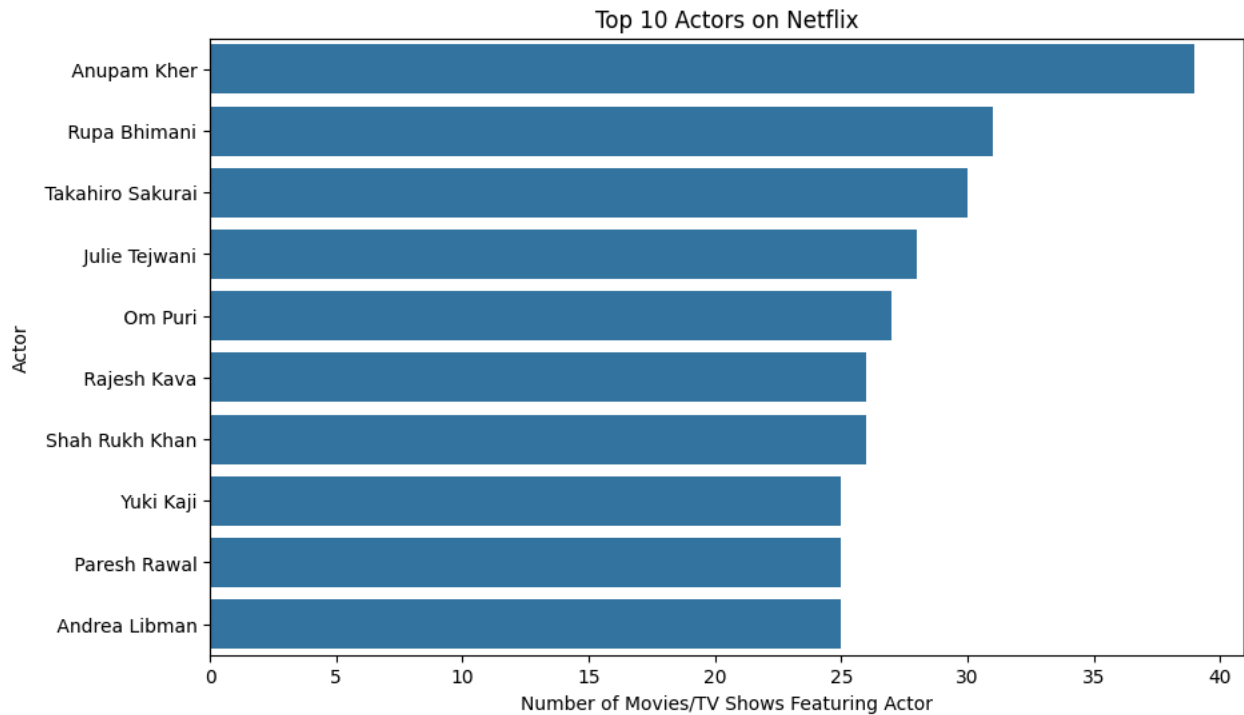
```
top_directors = df['director'].value_counts().head(10)
top_actors = df['cast'].str.split(',').explode().value_counts().head(10)
```

```
plt.figure(figsize=(10, 6))
sns.barplot(x=top_directors.values, y=top_directors.index)
plt.title('Top 10 Directors on Netflix')
plt.xlabel('Number of Movies/TV Shows Directed')
plt.ylabel('Director')
plt.show()
```

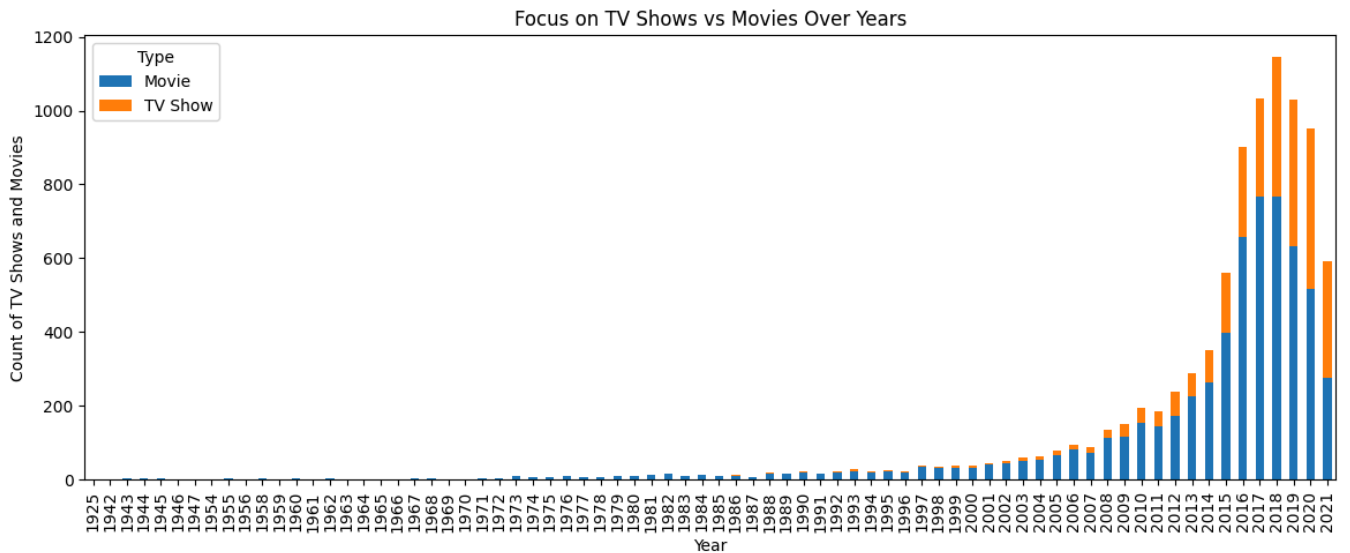


```
plt.figure(figsize=(10, 6))
sns.barplot(x=top_actors.values, y=top_actors.index)
```

```
plt.title('Top 10 Actors on Netflix')
plt.xlabel('Number of Movies/TV Shows Featuring Actor')
plt.ylabel('Actor')
plt.show()
```



```
recent_focus = df.groupby(['release_year', 'type']).size().unstack(fill_value=0)
recent_focus.plot(kind='bar', stacked=True, figsize=(14, 5))
plt.title('Focus on TV Shows vs Movies Over Years')
plt.xlabel('Year')
plt.ylabel('Count of TV Shows and Movies')
plt.legend(title='Type')
plt.show()
```



Start coding or [generate](#) with AI.