```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
!wget https://d2beiqkhq929f0.cloudfront.net/public_assets/assets/000/000/940/ori
df=pd.read_csv('netflix.csv')
```

```
--2025-01-12 04:38:06--  https://d2beiqkhq929f0.cloudfront.net/public_asset
Resolving d2beiqkhq929f0.cloudfront.net (d2beiqkhq929f0.cloudfront.net)...
Connecting to d2beiqkhq929f0.cloudfront.net (d2beiqkhq929f0.cloudfront.net)
HTTP request sent, awaiting response... 200 OK
Length: 3399671 (3.2M) [text/plain]
Saving to: ‘netflix.csv’

netflix.csv         100%[===================>]   3.24M  2.40MB/s    in 1.3s

2025-01-12 04:38:08 (2.40 MB/s) - ‘netflix.csv’ saved [3399671/3399671]
```

```
df.head()
```

	show_id	type	title	director	cast	country	date_added	release_
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	

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

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8807 entries, 0 to 8806
Data columns (total 12 columns):
 #   Column        Non-Null Count  Dtype 
---  ------        --------------  ----- 
 0   show_id       8807 non-null   object
 1   type          8807 non-null   object
 2   title         8807 non-null   object
 3   director      6173 non-null   object
 4   cast          7982 non-null   object
 5   country       7976 non-null   object
 6   date_added    8797 non-null   object
 7   release_year  8807 non-null   int64 
 8   rating        8803 non-null   object
 9   duration      8804 non-null   object
 10  listed_in     8807 non-null   object
 11  description   8807 non-null   object
dtypes: int64(1), object(11)
memory usage: 825.8+ KB
```

```
# 1. Un-nesting the columns
df['cast'] = df['cast'].str.split(',')
df['country'] = df['country'].str.split(',')
df['director'] = df['director'].str.split(',')
df['listed_in'] = df['listed_in'].str.split(',')

df=df.explode('cast').reset_index(drop=True)
df=df.explode('listed_in').reset_index(drop=True)
df=df.explode('country').reset_index(drop=True)
df=df.explode('director').reset_index(drop=True)

df
```

	show_id	type	title	director	cast	country	date_added	rele
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	
1	s2	TV Show	Blood & Water	NaN	Ama Qamata	South Africa	September 24, 2021	
2	s2	TV Show	Blood & Water	NaN	Ama Qamata	South Africa	September 24, 2021	

3	s2	TV Show	Blood & Water	NaN	Ama Qamata	South Africa	September 24, 2021
4	s2	TV Show	Blood & Water	NaN	Khosi Ngema	South Africa	September 24, 2021
...	...	...	...	...	...	...	...
202060	s8807	Movie	Zubaan	Mozez Singh	Anita Shabdish	India	March 2, 2019
202061	s8807	Movie	Zubaan	Mozez Singh	Anita Shabdish	India	March 2, 2019
			Zubaan	Mozez Singh	Chittaranjan Tripathy	India	March 2, 2019
202063	s8807	Movie	Zubaan	Mozez Singh	Chittaranjan Tripathy	India	March 2, 2019
202064	s8807	Movie	Zubaan	Mozez Singh	Chittaranjan Tripathy	India	March 2, 2019

202065 rows × 12 columns

Run cell (⌘/Ctrl+Enter)
cell executed since last change

executed by Vishal Singh
10:08 AM (0 minutes ago)
executed in 2.603s

```
#2. Handling Null Values


df['director'].fillna('Unknown Director')
df['cast'].fillna('Unknown cast')
df['country'].fillna('Unknown country')
df['date_added'].fillna('Unknown date_added')
df['rating'].fillna('Unknown rating')


df['duration'].fillna(0)
```

	duration
0	90 min
1	2 Seasons
2	2 Seasons
3	2 Seasons
4	2 Seasons
...	...
202060	111 min
202061	111 min
202062	111 min
202063	111 min
202064	111 min

202065 rows × 1 columns

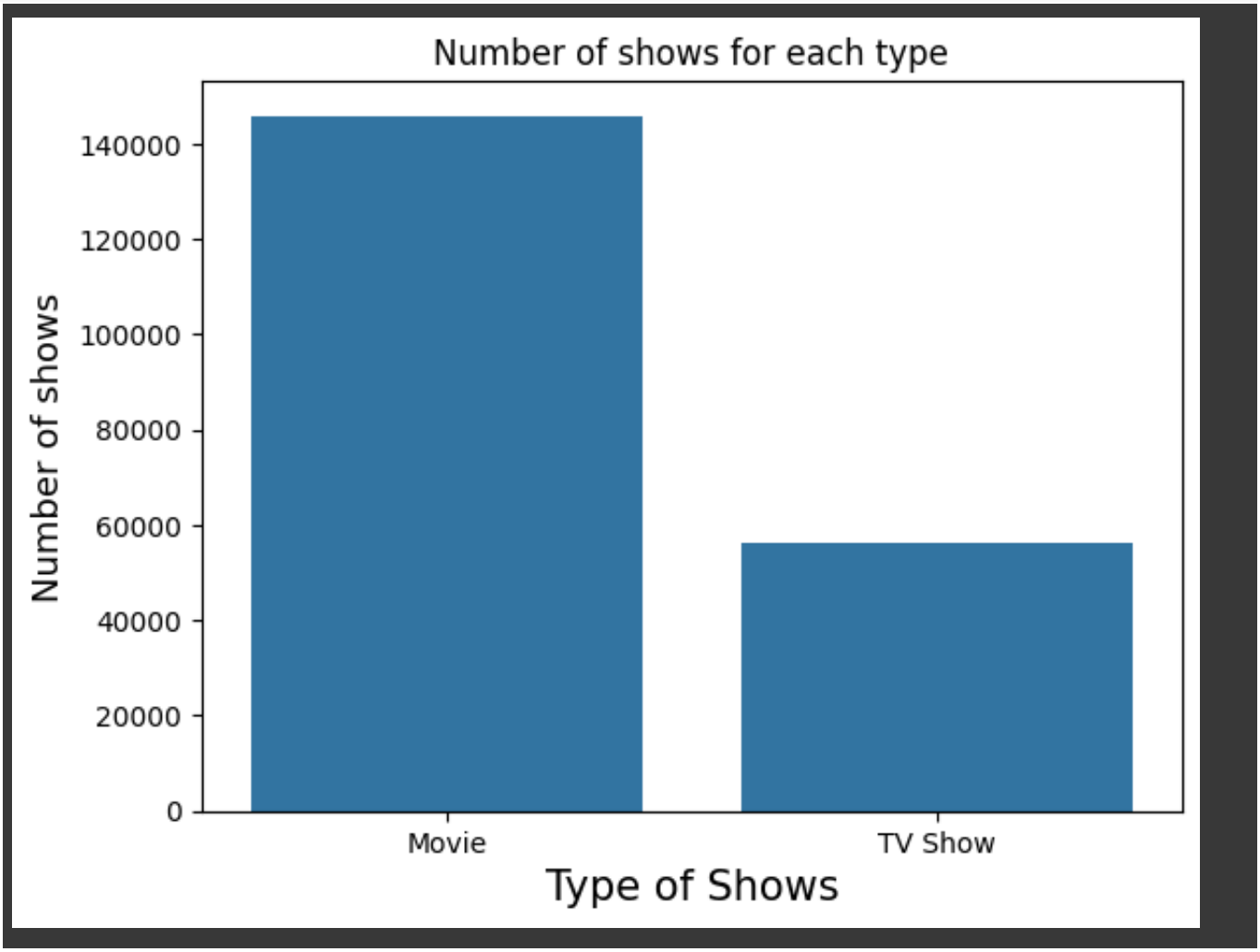
dtype: object

```
#Find the counts of each categorical variable both using graphical and non-
#graphical analysis.

# a. For Non-graphical Analysis:

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

type	count
Movie	145917
TV Show	56148

dtype: int64

```
#b. For graphical analysis:
sns.countplot(x='type',data=df)
plt.title('Number of shows for each type')
plt.xlabel('Type of Shows',fontsize=15)
plt.ylabel('Number of shows',fontsize=13)
plt.show()
```



We can clearly see that Netflix has significantly large number of Movies as compared to TV shows

```
#2. Comparison of tv shows vs. movies.
#a. Find the number of movies produced in each country and pick the top 10
#countries.

df_movies=df[df['type']=='Movie']
df_movies.groupby(['country'])['title'].nunique().sort_values(ascending=False)|
```

country	title
United States	2364
India	927
United States	388
United Kingdom	382
Canada	187
France	155
United Kingdom	152
France	148
Canada	132
Spain	129

dtype: int64

Clearly , United States has the maximum number of movies

```
# b.Find the number of Tv-Shows produced in each country and pick the top 10 cc

df_shows = df[df['type']=='TV Show']
df_shows.groupby(['country'])['title'].nunique().sort_values(ascending=False)[:
```

country	title
United States	847
United Kingdom	246
Japan	174
South Korea	164
United States	91
Canada	84
India	81
Taiwan	70
France	64
Australia	56

dtype: int64

Clearly , United States has maximum number of TV Shows

```
# 3. What is the best time to launch a TV show?
# a.Find which is the best week to release the Tv-show or the movie. Do the ana

#Movies
df['date_added']=df['date_added'].str.strip()
df['datetime']=pd.to_datetime(df['date_added'], format='%B %d, %Y')
df['weeknumber'] = df['datetime'].dt.isocalendar().week

df[df['type']=='Movie'].groupby(['weeknumber'])['title'].nunique().sort_values(
```

	title
weeknumber	
1	316

dtype: int64

Week 1 is the best time to release a Movie , as we see close to 316 movies released during that time

```
#TV SHows
df[df['type']=='TV Show'].groupby(['weeknumber'])['title'].nunique().sort_value
```

	title
weeknumber	
27	86

dtype: int64

Week 27 is the best time to release a TV show , as we see close to 86 TV Shows released during that time

```
# 3.b Find which is the best month to release the Tv-show or the movie. Do the

df[df['type']=='Movie'].groupby(df['datetime'].dt.month)['title'].nunique().sor
```

datetime	title
7.0	565

dtype: int64

7th month , which is July month having the most number of Movies

```
df[df['type']=='TV Show'].groupby(df['datetime'].dt.month)['title'].nunique().s
```

datetime	title
12.0	266

dtype: int64

12th month , which is December month having the most number of TV Shows

```
#4. Analysis of actors/directors of different types of shows/movies.
  #a. Identify the top 10 actors who have appeared in most movies or TV shows.

df.groupby(['cast'])['title'].nunique().sort_values(ascending=False)[0:10]
```

cast	title
Anupam Kher	39
Rupa Bhimani	31
Takahiro Sakurai	30
Julie Tejwani	28
Om Puri	27
Shah Rukh Khan	26
Rajesh Kava	26
Boman Irani	25
Andrea Libman	25
Yuki Kaji	25

dtype: int64

Above are the top 10 actors and We can clearly see Anupam Kher's appearance is highest among all the Tv Shows/Movies

```
#b. Identify the top 10 directors who have appeared in most movies or TV shows.

df.groupby(['director'])['title'].nunique().sort_values(ascending=False)[0:10]
```

director	title
Rajiv Chilaka	22
Raúl Campos	18
Jan Suter	18
Suhas Kadav	16
Marcus Raboy	16
Jay Karas	15
Cathy Garcia-Molina	13
Jay Chapman	12
Martin Scorsese	12
Youssef Chahine	12

dtype: int64

Above are the top 10 directors and We can clearly see Director Rajiv Chilaka has appeared in most of the the movies/Tv shows

```
#5. Which genre movies are more popular or produced more
df[df['type']=='Movie'].groupby(['listed_in'])['show_id'].count().sort_values(a
```

listed_in	show_id
International Movies	27141
Dramas	19657
Comedies	13894
Action & Adventure	12216
Dramas	10149

dtype: int64

We can clearly see that Movies with 'International Movies' Genre are the most popular one.

```
# 6. Find After how many days the movie will be added to Netflix after the relea
df['difference']=df['datetime'].dt.year-df['release_year']

df['difference'].mode()
```

	difference
0	0.0

dtype: float64

we can clearly see that most of the movies/TV shows were added in the netflix platform the same year as they were released