In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
df=pd.read_csv(r'D:\netflix.csv')
```

In [3]:
```
data=df
```

In [4]:
```
data.head()
```

Out[4]:

	show_id	type	title	director	cast	country	date_added	release_year	rati
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	P
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	T M
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Iouv	NaN	September 24, 2021	2021	T M

In [5]:
```
cast=data['cast'].apply(lambda x : str(x).split(',')).tolist()
```

In [6]:
```
df_cast=pd.DataFrame(cast, index = data['show_id'])
```

In [7]:
```
df_cast
```

Out[7]:

show_id	0	1	2	3	4	5	6	
s1	nan	None	None	None	None	None	None	
s2	Ama Qamata	Khosi Ngema	Gail Mabalane	Thabang Molaba	Dillon Windvogel	Natasha Thahane	Arno Greeff	Tsha
s3	Sami Bouajila	Tracy Gotoas	Samuel Jouy	Nabiha Akkari	Sofia Lesaffre	Salim Kechiouche	Noureddine Farihi	Ge Ramp
s4	nan	None	None	None	None	None	None	
s5	Mayur More	Jitendra Kumar	Ranjan Raj	Alam Khan	Ahsaas Channa	Revathi Pillai	Urvi Singh	Arun
...	...	...	...	...	...	...	...	
s8803	Mark Ruffalo	Jake Gyllenhaal	Robert Downey Jr.	Anthony Edwards	Brian Cox	Elias Koteas	Donal Logue	John
s8804	nan	None	None	None	None	None	None	
s8805	Jesse Eisenberg	Woody Harrelson	Emma Stone	Abigail Breslin	Amber Heard	Bill Murray	Derek Graf	
s8806	Tim Allen	Courteney Cox	Chevy Chase	Kate Mara	Ryan Newman	Michael Cassidy	Spencer Breslin	R
s8807	Vicky Kaushal	Sarah-Jane Dias	Raaghav Chanana	Manish Chaudhary	Meghna Malik	Malkeet Rauni	Anita Shabdish	Chitta T

8807 rows × 50 columns

In [8]:
```
df_cast = df_cast.stack()
```

In [9]:
```
df_cast = df_cast.reset_index()
```

In [10]:
```
df_cast.drop(columns = 'level_1', inplace = True)
```

In [11]:
```
df_cast.columns = ['show_id', 'cast']
```

In [12]:
```
df_cast_fav = data.merge(df_cast, on = 'show_id')
```

In [13]:
```
df_cast_fav.drop(columns = ['cast_x'], inplace = True)
```

In [14]:
```
df_cast_fav = df_cast_fav.rename({'cast_y':'cast'}, axis=1)
```

In [15]:
```
df_cast_fav['cast']=df_cast_fav['cast'].apply(lambda x : str(x).strip())
```

In [16]:
```
df_cast_fav.drop_duplicates(keep = 'first', inplace = True)
```

In [17]:
```
# df_cast_fav
```

In [18]:
```
#next for director column
```

In [19]:
```
director=df_cast_fav['director'].apply(lambda x : str(x).split(',')).tolist(
```

In [20]:
```
df_director=pd.DataFrame(director, index = df_cast_fav['show_id'])
```

In [21]:
```
df_director = df_director.stack()
```

In [22]:
```
df_director = df_director.reset_index()
```

In [23]:
```
df_director.drop(columns = 'level_1', inplace = True)
```

In [24]:
```
df_director.columns = ['show_id', 'director']
```

In [25]:
```
df_director_fav = df_cast_fav.merge(df_director, on = 'show_id')
```

In [26]:
```
df_director_fav.drop(columns = ['director_x'], inplace = True)
```

In [27]:
```
df_director_fav = df_director_fav.rename({'director_y':'director'}, axis=1)
```

In [28]:
```
df_director_fav['director']=df_director_fav['director'].apply(lambda x : str
```

In [29]:
```
df_director_fav.drop_duplicates(keep = 'first', inplace = True)
```

In [30]:
```
df_director_fav
```

Out[30]:

	show_id	type	title	country	date_added	release_year	rating	duration	li
0	s1	Movie	Dick Johnson Is Dead	United States	September 25, 2021	2020	PG-13	90 min	Docum
1	s2	TV Show	Blood & Water	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	Inter TV Sh Dra M
20	s2	TV Show	Blood & Water	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	Inter TV Sh Dra M
39	s2	TV Show	Blood & Water	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	Inter TV Sh Dra M
58	s2	TV Show	Blood & Water	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	Inter TV Sh Dra M
...	...	...	...	...	...	...	...	...	
751186	s8807	Movie	Zubaan	India	March 2, 2019	2015	TV-14	111 min	I Inter Movie & M
751194	s8807	Movie	Zubaan	India	March 2, 2019	2015	TV-14	111 min	I Inter Movie & M
751202	s8807	Movie	Zubaan	India	March 2, 2019	2015	TV-14	111 min	I Inter Movie & M
751210	s8807	Movie	Zubaan	India	March 2, 2019	2015	TV-14	111 min	I Inter Movie & M
751218	s8807	Movie	Zubaan	India	March 2, 2019	2015	TV-14	111 min	I Inter Movie & M

70802 rows × 12 columns

In [31]:
```
#next for listedin column
```

In [32]:
```
listed_in=df_director_fav['listed_in'].apply(lambda x : str(x).split(',')).t

df_listed_in = pd.DataFrame(listed_in, index = df_director_fav['show_id'])

df_listed_in = df_listed_in.stack()

df_listed_in = df_listed_in.reset_index()

df_listed_in.drop(columns = 'level_1', inplace = True)

df_listed_in.columns = ['show_id', 'listed_in']

df_listed_in_fav = df_director_fav.merge(df_listed_in, on = 'show_id')

df_listed_in_fav.drop(columns = ['listed_in_x'], inplace = True)

df_listed_in_fav = df_listed_in_fav.rename({'listed_in_y':'listed_in'}, axis

df_listed_in_fav['listed_in']=df_listed_in_fav['listed_in'].apply(lambda x :

df_listed_in_fav.drop_duplicates(keep = 'first', inplace = True)
```

In [33]:
```
df_listed_in_fav
```

Out[33]:

	show_id	type	title	country	date_added	release_year	rating	duration	descr
0	s1	Movie	Dick Johnson Is Dead	United States	September 25, 2021	2020	PG-13	90 min	father the h fi
1	s2	TV Show	Blood & Water	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	cr patl p Cape
2	s2	TV Show	Blood & Water	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	cr patl p Cape
3	s2	TV Show	Blood & Water	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	cr patl p Cape
58	s2	TV Show	Blood & Water	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	cr patl p Cape
...	...	...	...	...	...	...	...	...	
2340692	s8807	Movie	Zubaan	India	March 2, 2019	2015	TV-14	111 min	A s bu boy his w
2340693	s8807	Movie	Zubaan	India	March 2, 2019	2015	TV-14	111 min	A s bu boy his w
2340715	s8807	Movie	Zubaan	India	March 2, 2019	2015	TV-14	111 min	A s bu boy his w
2340716	s8807	Movie	Zubaan	India	March 2, 2019	2015	TV-14	111 min	A s bu boy his w
2340717	s8807	Movie	Zubaan	India	March 2, 2019	2015	TV-14	111 min	A s bu boy his w

161189 rows × 12 columns

#next for country column

In [34]:
```
country=df_listed_in_fav['country'].apply(lambda x : str(x).split(',')).toli

df_country = pd.DataFrame(country, index = df_listed_in_fav['show_id'])

df_country = df_country.stack()

df_country = df_country.reset_index()

df_country.drop(columns = 'level_1', inplace = True)

df_country.columns = ['show_id', 'country']

df_country_fav = df_listed_in_fav.merge(df_country, on = 'show_id')

df_country_fav.drop(columns = ['country_x'], inplace = True)

df_country_fav = df_country_fav.rename({'country_y':'country'}, axis=1)

df_country_fav['country']=df_country_fav['country'].apply(lambda x : str(x).

df_country_fav.drop_duplicates(keep = 'first', inplace = True)
```

In [35]:
```
df_country_fav.head()
```

Out[35]:

	show_id	type	title	date_added	release_year	rating	duration	description	cas
0	s1	Movie	Dick Johnson Is Dead	September 25, 2021	2020	PG-13	90 min	As her father nears the end of his life, filmm...	na
1	s2	TV Show	Blood & Water	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Am Qamat
58	s2	TV Show	Blood & Water	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Am Qamat
115	s2	TV Show	Blood & Water	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Am Qamat
172	s2	TV Show	Blood & Water	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Khos Ngem

In [36]:
```
Data2 = df_country_fav
```

In [37]:
```
Data2.reset_index(inplace = True)
```

In [38]:
```
Data2.drop(columns = ['index'], inplace = True)
```

In [39]:
```
Data2
```

Out[39]:

	show_id	type	title	date_added	release_year	rating	duration	description	
0	s1	Movie	Dick Johnson Is Dead	September 25, 2021	2020	PG-13	90 min	As her father nears the end of his life, filmm...	
1	s2	TV Show	Blood & Water	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	
2	s2	TV Show	Blood & Water	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	
3	s2	TV Show	Blood & Water	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	
4	s2	TV Show	Blood & Water	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	
...	...	...	...	...	...	...	...	...	
202005	s8807	Movie	Zubaan	March 2, 2019	2015	TV-14	111 min	A scrappy but poor boy worms his way into a ty...	S
202006	s8807	Movie	Zubaan	March 2, 2019	2015	TV-14	111 min	A scrappy but poor boy worms his way into a ty...	S
202007	s8807	Movie	Zubaan	March 2, 2019	2015	TV-14	111 min	A scrappy but poor boy worms his way into a ty...	Chi
202008	s8807	Movie	Zubaan	March 2, 2019	2015	TV-14	111 min	A scrappy but poor boy worms his way into a ty...	Chi
202009	s8807	Movie	Zubaan	March 2, 2019	2015	TV-14	111 min	A scrappy but poor boy worms his way into a ty...	Chi

202010 rows × 12 columns

In [40]:
```
Data2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 202010 entries, 0 to 202009
Data columns (total 12 columns):
 #   Column        Non-Null Count   Dtype 
---  ------        --------------   ----- 
 0   show_id       202010 non-null  object
 1   type          202010 non-null  object
 2   title         202010 non-null  object
 3   date_added    201852 non-null  object
 4   release_year  202010 non-null  int64 
 5   rating        201943 non-null  object
 6   duration      202007 non-null  object
 7   description   202010 non-null  object
 8   cast          202010 non-null  object
 9   director      202010 non-null  object
 10  listed_in     202010 non-null  object
 11  country       202010 non-null  object
dtypes: int64(1), object(11)
memory usage: 18.5+ MB
```

#Cheking sum of all na column wise

In [41]:
```
Data2.isna().sum()
```

Out[41]:
```
show_id           0
type              0
title             0
date_added      158
release_year      0
rating           67
duration          3
description       0
cast              0
director          0
listed_in         0
country           0
dtype: int64
```

#Checking string 'nan' in all columns

In [42]:
```
len(Data2[Data2['show_id']=='nan'])
```

Out[42]:
```
0
```

In [43]:
```
len(Data2[Data2['type']=='nan'])
```

Out[43]:
```
0
```

In [44]:
```
len(Data2[Data2['title']=='nan'])
```

Out[44]:
```
0
```

In [45]:
```
len(Data2[Data2['date_added']=='nan'])
```
Out[45]: 0

In [46]:
```
len(Data2[Data2['release_year']=='nan'])
```
Out[46]: 0

In [47]:
```
len(Data2[Data2['rating']=='nan'])
```
Out[47]: 0

In [48]:
```
len(Data2[Data2['duration']=='nan'])
```
Out[48]: 0

In [49]:
```
len(Data2[Data2['description']=='nan'])
```
Out[49]: 0

In [50]:
```
len(Data2[Data2['cast']=='nan'])
```
Out[50]: 2149

In [51]:
```
len(Data2[Data2['director']=='nan'])
```
Out[51]: 50643

In [52]:
```
len(Data2[Data2['listed_in']=='nan'])
```
Out[52]: 0

In [53]:
```
len(Data2[Data2['country']=='nan'])
```
Out[53]: 11897

#Filling null values with unkown_columns_names

In [54]:
```
Data2['date_added'].fillna('Unknown_date_added', inplace = True)
```

In [55]:
```
Data2['rating'].fillna('Unknown_rating', inplace = True)
```

In [56]:
```
Data2['duration'].fillna('Unknown_duration', inplace = True)
```

In [57]:
```
Data2.isna().sum()
```

Out[57]:
```
show_id         0
type            0
title           0
date_added      0
release_year    0
rating          0
duration        0
description     0
cast            0
director        0
listed_in       0
country         0
dtype: int64
```

#Filling 'nan' values with unkown_columns_names

In [58]:
```
Data2['cast'].replace('nan','Unkown_cast', inplace = True)
```

In [59]:
```
Data2['director'].replace('nan','Unkown_director', inplace = True)
```

In [60]:
```
Data2['country'].replace('nan','Unkown_country', inplace = True)
```

In [61]:
```
Data2.head()
```

Out[61]:

	show_id	type	title	date_added	release_year	rating	duration	description	
0	s1	Movie	Dick Johnson Is Dead	September 25, 2021	2020	PG-13	90 min	As her father nears the end of his life, filmm...	Unkown_
1	s2	TV Show	Blood & Water	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Ama Qan
2	s2	TV Show	Blood & Water	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Ama Qan
3	s2	TV Show	Blood & Water	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Ama Qan
4	s2	TV Show	Blood & Water	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Khosi Ng

In [62]:
```
Data2[(Data2['rating']=='74 min') | (Data2['rating']=='66 min') | (Data2['ra
```

Out[62]:

	show_id	type	title	date_added	release_year	rating	duration	d
126533	s5542	Movie	Louis C.K. 2017	April 4, 2017	2017	74 min	Unknown_duration	Louis C eterna
131599	s5795	Movie	Louis C.K.: Hilarious	September 16, 2016	2010	84 min	Unknown_duration	Emr cor Louis
131733	s5814	Movie	Louis C.K.: Live at the Comedy Store	August 15, 2016	2015	66 min	Unknown_duration	The his hilariou

In [63]:
```
# sawp the duration given in rating to duration column
Data2.at[126533,'rating']=Data2.at[126533,'duration']
Data2.at[131599,'rating']=Data2.at[131599,'duration']
Data2.at[131733,'rating']=Data2.at[131733,'duration']
```

In [64]:
```
#fill the same row rating cell with unkown rating text
Data2.at[126533,'rating']='Unknown_rating'
Data2.at[131599,'rating']='Unknown_rating'
Data2.at[131733,'rating']='Unknown_rating'
```

In [65]:
```
#mistakenly did not filled the correct values in duration cells
#filling it again manualy

Data2.at[126533,'duration']="74 min"
Data2.at[131599,'duration']="84 min"
Data2.at[131733,'duration']="66 min"
```

1. Find the counts of each categorical variable both using graphical and nongraphical analysis.

a. For Non-graphical Analysis: Hint : We want you to find the values counts of each category for the given column

In [66]:
```
Data2['type'].value_counts()
```

Out[66]:
```
Movie      145862
TV Show     56148
Name: type, dtype: int64
```

In [67]:
```
Data2['rating'].value_counts()
```

Out[67]:
```
TV-MA             73867
TV-14             43951
R                 25859
PG-13             16246
TV-PG             14926
PG                10919
TV-Y7              6304
TV-Y               3665
TV-G               2779
NR                 1573
G                  1530
NC-17               149
TV-Y7-FV             86
UR                   86
Unknown_rating       70
Name: rating, dtype: int64
```

In [68]:
```
Data2['listed_in'].value_counts()
```

Out[68]:
```
Dramas                          29787
International Movies            28224
Comedies                        20829
International TV Shows          12845
Action & Adventure              12216
Independent Movies               9818
Children & Family Movies         9771
TV Dramas                        8942
Thrillers                        7106
Romantic Movies                  6412
TV Comedies                      4963
Crime TV Shows                   4733
Horror Movies                    4571
Kids' TV                         4568
Sci-Fi & Fantasy                 4037
Music & Musicals                 3077
Romantic TV Shows                3049
Documentaries                    2409
Anime Series                     2313
TV Action & Adventure            2288
Spanish-Language TV Shows        2126
British TV Shows                 1808
Sports Movies                    1531
Classic Movies                   1443
TV Mysteries                     1281
Korean TV Shows                  1122
Cult Movies                      1077
TV Sci-Fi & Fantasy              1045
Anime Features                   1045
TV Horror                         941
Docuseries                        845
LGBTQ Movies                      838
TV Thrillers                      768
Teen TV Shows                     742
Reality TV                        735
Faith & Spirituality              719
Stand-Up Comedy                   540
Movies                            412
TV Shows                          337
Classic & Cult TV                 272
Stand-Up Comedy & Talk Shows      268
Science & Nature TV               157
Name: listed_in, dtype: int64
```

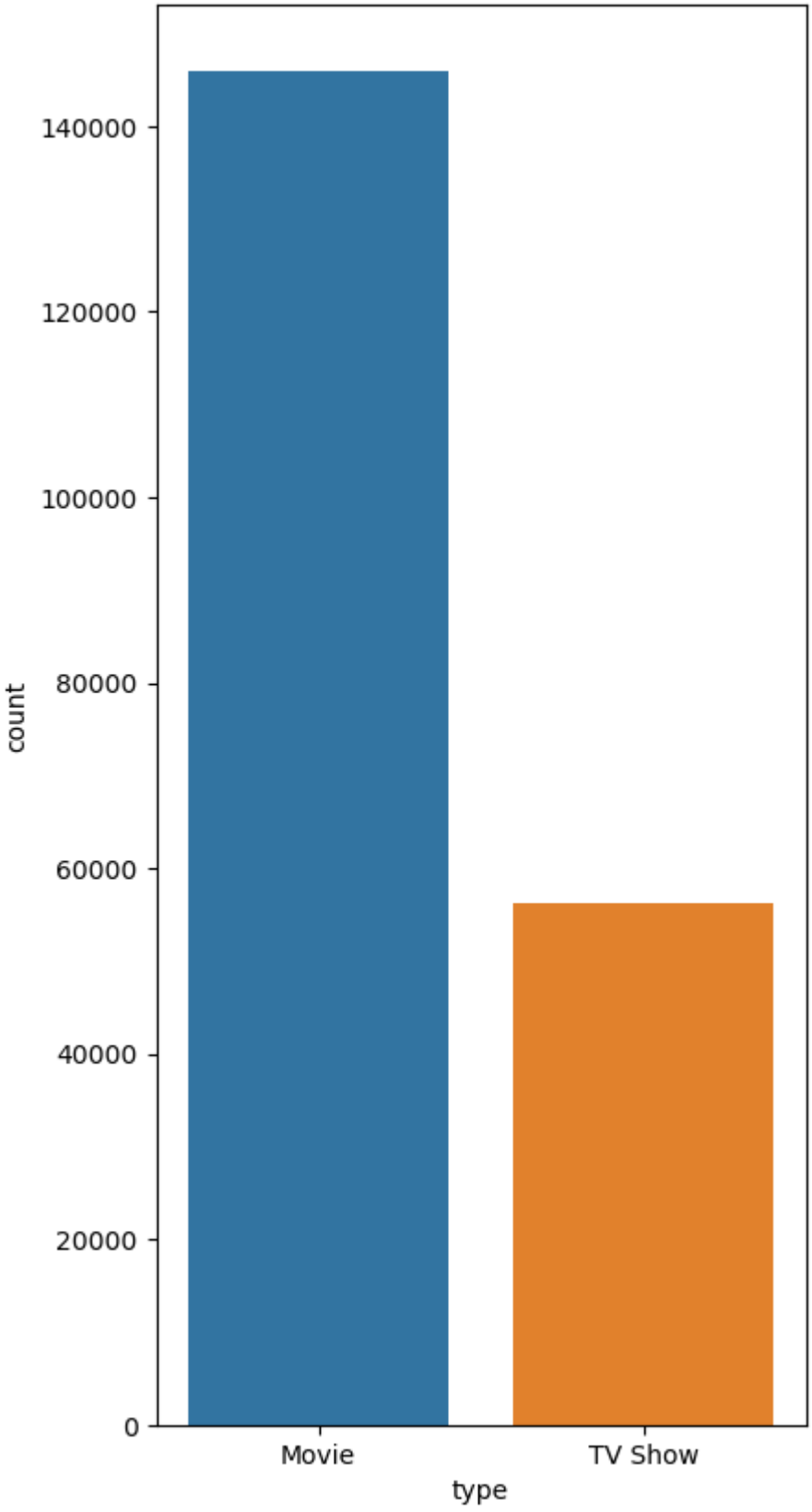
b. For graphical analysis:

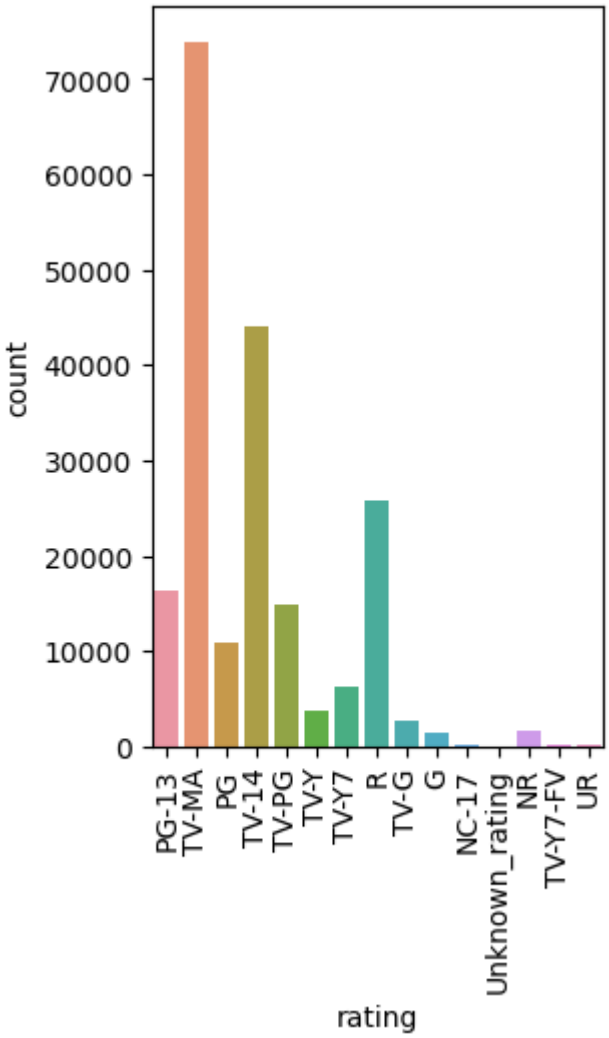
Hint : We can use a count plot to get the counts of each category

In [69]:
```
plt.figure(figsize=(10, 10))
plt.subplot(1, 2, 1)
sns.countplot(data=Data2, x='type')
plt.show()

plt.subplot(1, 2, 2)
sns.countplot(data=Data2, x='rating')
plt.xticks(rotation=90)
plt.show()


# plt.subplot(2, 3, 5)
# sns.countplot(data=Data2, x='listed_in')
```

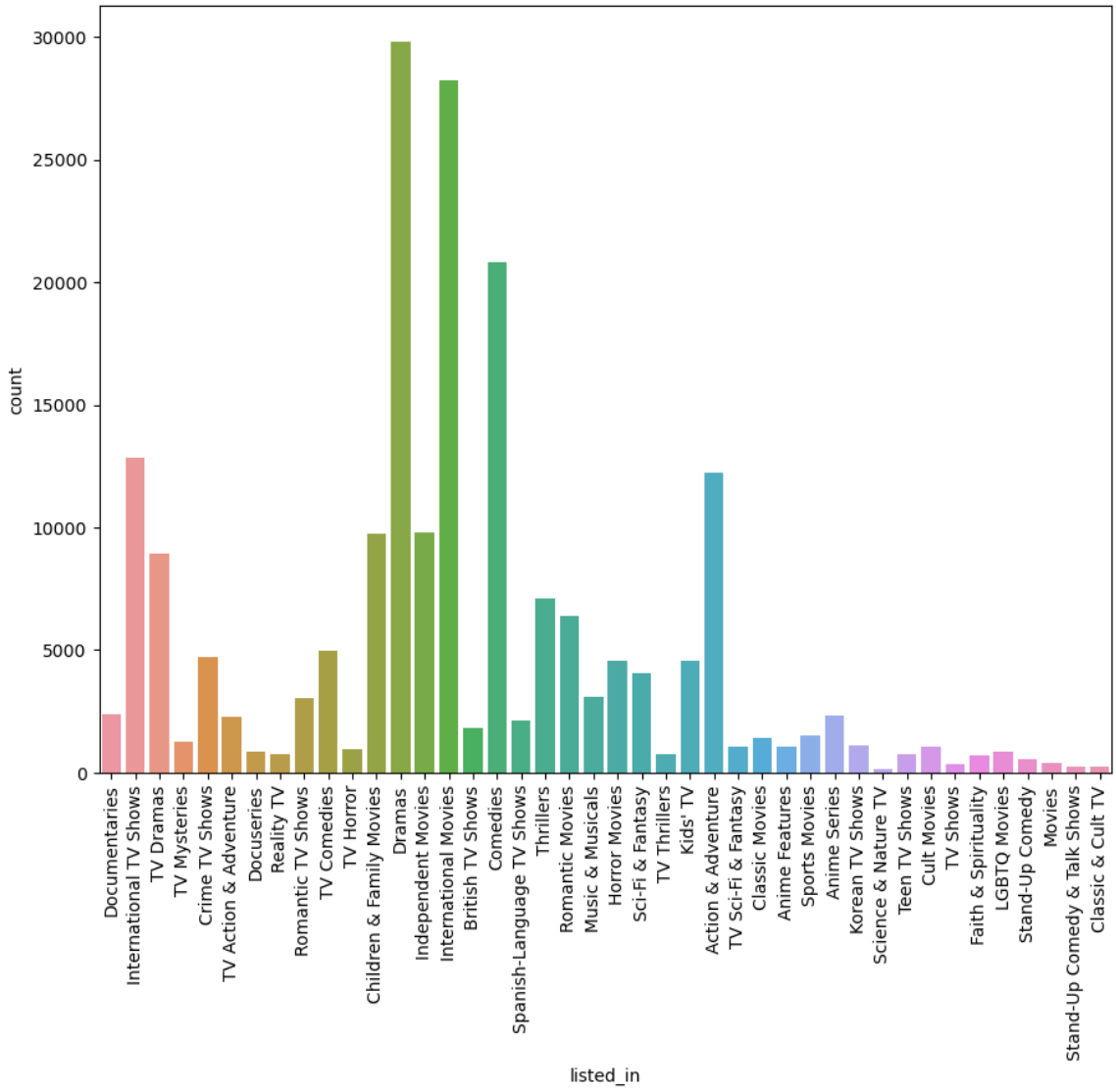

140000
120000
100000
80000
60000
40000
20000
0
count
Movie
TV Show
type



Conclusion:

1. Movies counts are way to more than TV shows or Web series in the netflix platform.
2. TV shows and Movies with rating TV-MA is higest in number followed by TV-14 & R and lowest are with TV-Y7-FV & UR rating.

In [70]:
```
plt.figure(figsize=(10.5,8))
sns.countplot(data=Data2, x='listed_in')
plt.xticks(rotation=90)
plt.show()
```



Conclusion: Children and Family movies is the most in listed in followed by international movies, comedis, internal TV shows and action & adventure. Where as the science and nature TV shows are least in listed in.

2. Comparison of tv shows vs. movies.

a. Find the number of movies produced in each country and pick the top 10 countries.

Hint : We want you to apply group by each country and find the count of unique titles of movies

Non_graphical

In [71]:
```
a2_data = Data2.loc[Data2['type']=='Movie',['title','country']]
```

In [72]:
```
a2_data.groupby("country")[["title"]].aggregate(
    counts = ("title", "count")
).sort_values(['counts'], ascending = False).head(11).reset_index()
```

Out[72]:

	country	counts
0	United States	45792
1	India	21411
2	United Kingdom	8580
3	France	6605
4	Unkown_country	6199
5	Canada	5738
6	Japan	3525
7	Spain	3469
8	Germany	3427
9	China	2377
10	Nigeria	2236

Conclusion : Above are the top 10 countries who produesc movies.

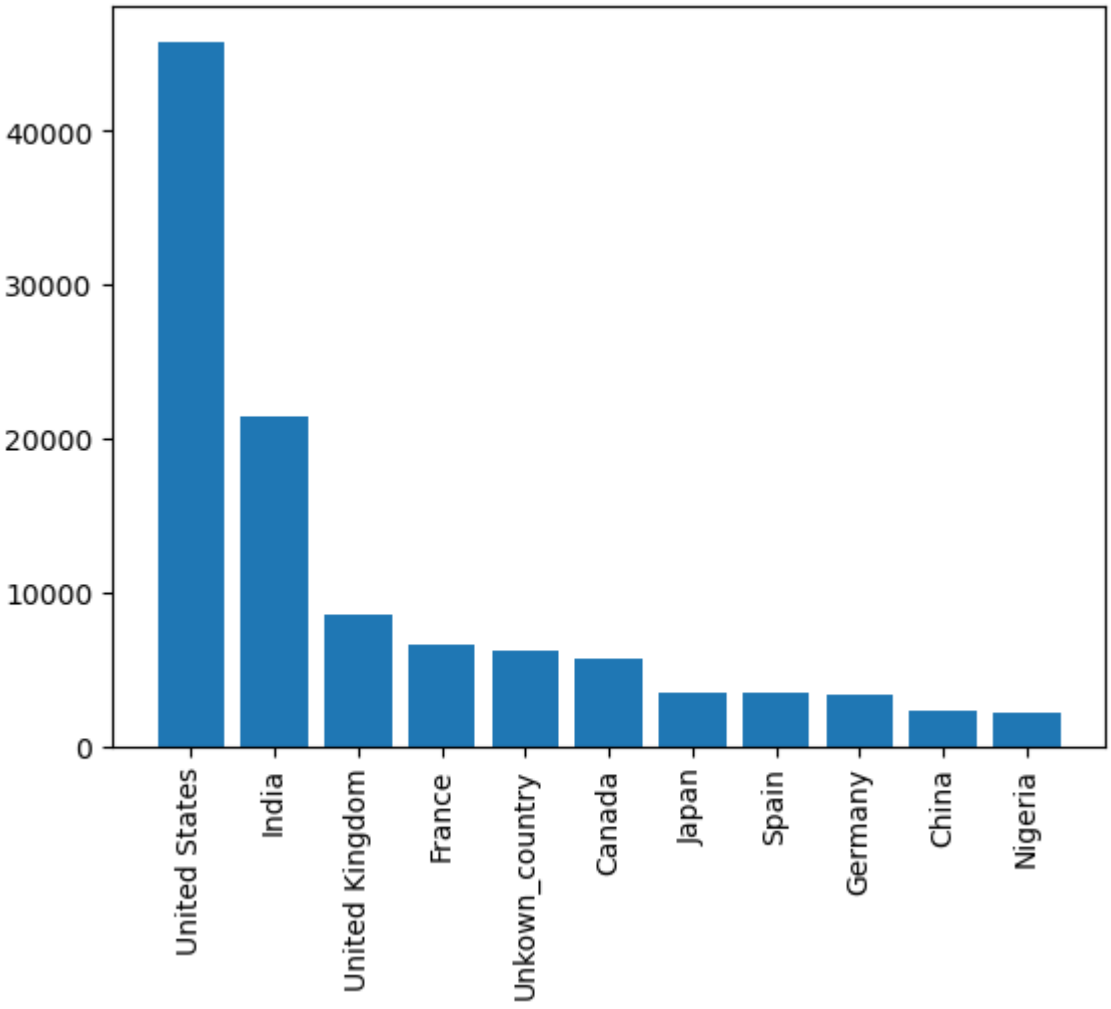
Note: Neglect unknown_country column

Graphical

In [73]:
```
a2_graphical_data = Data2[Data2['type']=='Movie']
```

In [74]:
```
a2_graphical_data = a2_data.groupby("country")[["title"]].aggregate(
    counts = ("title", "count")
).sort_values(['counts'], ascending = False).head(11).reset_index()
```

In [75]:
```
plt.bar(a2_graphical_data['country'].values, a2_graphical_data['counts'].val
plt.xticks(rotation=90)
plt.show()
```



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

Hint : We want you to apply group by each country and find the count of unique titles of Tv-showsb. Find the number of Tv-Shows produced in each country and pick the top 10 countries.

Non graphical

In [76]:
```
b2_data = Data2.loc[Data2['type']=='TV Show',['title','country']]
```

In [77]:
```
b2_data.groupby("country")[["title"]].aggregate(
    counts = ("title", "count")
).sort_values(['counts'], ascending = False).head(11).reset_index()
```

Out[77]:

	country	counts
0	United States	13533
1	Unkown_country	5698
2	Japan	5154
3	United Kingdom	4385
4	South Korea	3754
5	Canada	2177
6	Mexico	2018
7	Spain	1846
8	Taiwan	1719
9	France	1647
10	India	1403

Conclusion : Above are the top 10 countries who produesc TV Show.

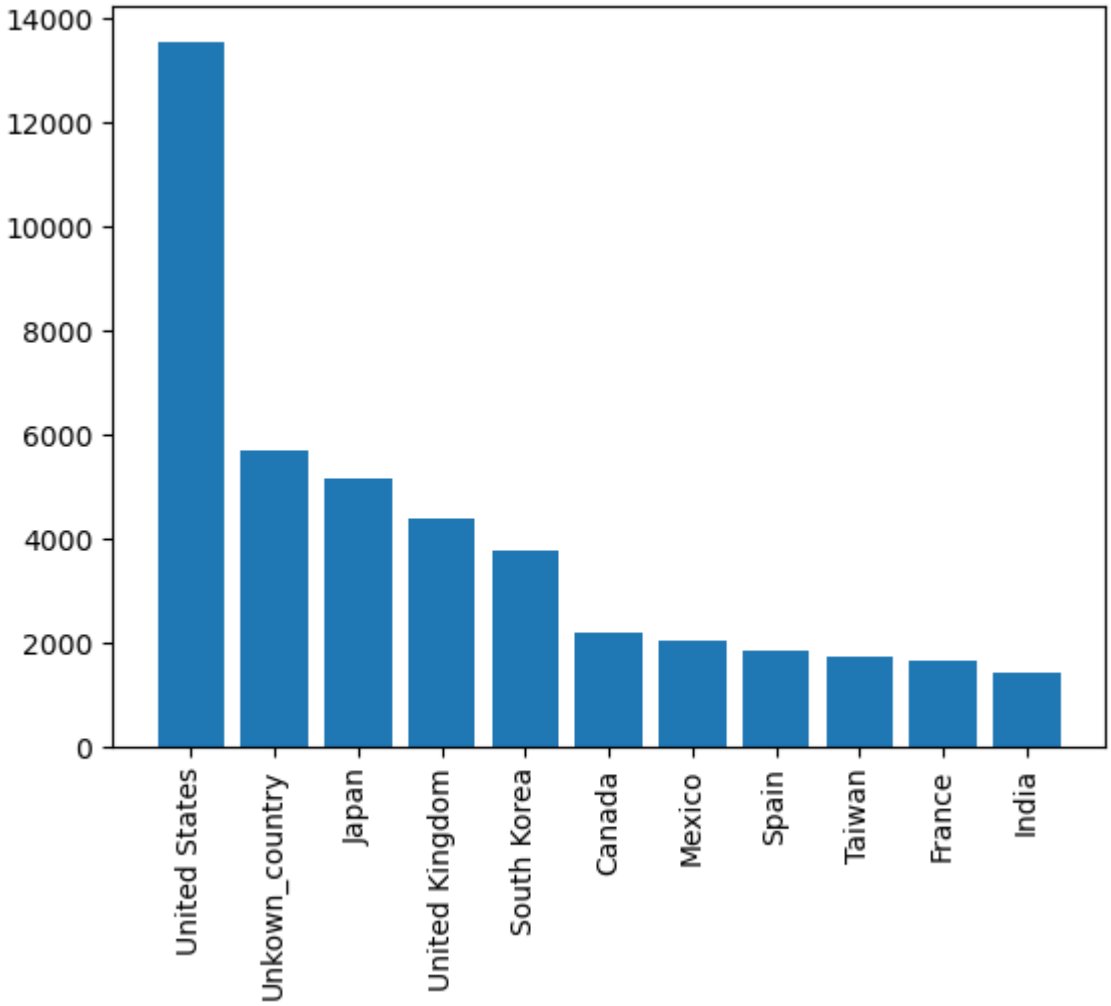
Note: Neglect unknown_country column

Graphical

In [78]:
```
b2_graphical_data = Data2[Data2['type']=='TV Show']
```

In [79]:
```
b2_graphical_data = b2_data.groupby("country")[["title"]].aggregate(
    counts = ("title", "count")
).sort_values(['counts'], ascending = False).head(11).reset_index()
```

In [80]:
```
plt.bar(b2_graphical_data['country'].values, b2_graphical_data['counts'].val
plt.xticks(rotation=90)
plt.show()
```



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 analysis separately for Tv-shows and Movies

Hint : We expect you to create a new column and group by each week and count the total number of movies/ tv shows.

In [81]:
```
Data2.head()
```

Out[81]:

	show_id	type	title	date_added	release_year	rating	duration	description	
0	s1	Movie	Dick Johnson Is Dead	September 25, 2021	2020	PG-13	90 min	As her father nears the end of his life, filmm...	Unkown_
1	s2	TV Show	Blood & Water	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Ama Qan
2	s2	TV Show	Blood & Water	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Ama Qan
3	s2	TV Show	Blood & Water	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Ama Qan
4	s2	TV Show	Blood & Water	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Khosi Ng

replacing unknown_date_added to mode value of that column for analysis

In [82]:
```
Data2['date_added'].mode()
```

Out[82]:
```
0    January 1, 2020
Name: date_added, dtype: object
```

In [83]:
```
Data2['date_added'].replace('Unknown_date_added','January 1, 2020', inplace
```

In [84]:
```
Data2['date_added'] = pd.to_datetime(Data2['date_added'])
```

In [85]:
```
Data2.head()
```

Out[85]:

	show_id	type	title	date_added	release_year	rating	duration	description	
0	s1	Movie	Dick Johnson Is Dead	2021-09-25	2020	PG-13	90 min	As her father nears the end of his life, filmm...	Unkown_
1	s2	TV Show	Blood & Water	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Ama Qan
2	s2	TV Show	Blood & Water	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Ama Qan
3	s2	TV Show	Blood & Water	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Ama Qan
4	s2	TV Show	Blood & Water	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Khosi Ng

In [86]:
```
Data2['Month'] = pd.to_datetime(Data2['date_added']).dt.month
```

In [87]:
```
Data2['week'] = pd.to_datetime(Data2['date_added']).dt.week
```

```
C:\Users\Dhrubo\AppData\Local\Temp\ipykernel_7444\729652704.py:1: FutureWa
rning: Series.dt.weekofyear and Series.dt.week have been deprecated. Pleas
e use Series.dt.isocalendar().week instead.
  Data2['week'] = pd.to_datetime(Data2['date_added']).dt.week
```

In [88]:
```
Data2.head()
```

Out[88]:

	show_id	type	title	date_added	release_year	rating	duration	description	
0	s1	Movie	Dick Johnson Is Dead	2021-09-25	2020	PG-13	90 min	As her father nears the end of his life, filmm...	Unkown_
1	s2	TV Show	Blood & Water	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Ama Qan
2	s2	TV Show	Blood & Water	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Ama Qan
3	s2	TV Show	Blood & Water	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Ama Qan
4	s2	TV Show	Blood & Water	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	Khosi Ng

In [89]:
```
#Best time(week) for releasing TV Show
Data2.loc[Data2['type']=='TV Show'].groupby('week')['title'].count().sort_va
```

Out[89]:
```
week
27    1977
35    1945
24    1702
26    1662
31    1646
Name: title, dtype: int64
```

Hence week 27 is the best time for releasing TV Show

In [90]:
```
#Best time(week) for releasing Movie
Data2.loc[Data2['type']=='Movie'].groupby('week')['title'].count().sort_valu
```

Out[90]:
```
week
1     8456
44    5563
9     5094
35    5048
26    4931
Name: title, dtype: int64
```

Hence week 1 is best time for releasing Movie

In [91]:
```
#Best time(Month) for releasing TV Show
Data2.loc[Data2['type']=='TV Show'].groupby('Month')['title'].count().sort_v
```

Out[91]:
```
Month
12    5498
7     5227
8     5162
6     5043
9     4900
Name: title, dtype: int64
```

Hence month 12 is best time for releasing Movie

In [92]:
```
#Best time(Month) for releasing Movie
Data2.loc[Data2['type']=='Movie'].groupby('Month')['title'].count().sort_val
```

Out[92]:
```
Month
7     15075
1     13947
10    13535
9     13220
12    12768
Name: title, dtype: int64
```

Hence Month 7 is best time for releasing Movie

4. Analysis of actors/directors of different types of shows/movies.

a. Identify the top 10 directors who have appeared in most movies or TV shows.

Hint : We want you to group by each actor and find the count of unique titles of Tv-shows/movies

In [93]:
```
#Top 10 Actors in TV Shows
Data2.loc[Data2['type']=='TV Show'].groupby('cast')['title'].count().sort_va
```

Out[93]:
```
cast
Unkown_cast          818
David Attenborough    82
Takahiro Sakurai      56
Yuki Kaji             45
Ai Kayano             41
Junichi Suwabe        39
Daisuke Ono           38
Jun Fukuyama          38
Yuichi Nakamura       38
Kate Harbour          37
Joanna Kulig          35
Name: title, dtype: int64
```

Ignore Unknown_cast as they missing data.

These are the top 10 Actor worked on maximum TV shows.

In [94]:
```
#Top 10 Actors in Movies.
Data2.loc[Data2['type']=='Movie'].groupby('cast')['title'].count().sort_valu
```

Out[94]:
```
cast
Unkown_cast         1331
Liam Neeson          161
Alfred Molina        157
John Krasinski       138
Salma Hayek          130
Frank Langella       128
Anupam Kher          118
John Rhys-Davies     116
Shah Rukh Khan       108
Naseeruddin Shah     106
James Franco         100
Name: title, dtype: int64
```

Ignore Unknown_cast as they missing data.

These are the top 10 Actor worked on maximum Movies.

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

Hint : We want you to group by each director and find the count of unique titles of Tv-shows/movies

In [95]:
```
#Top 10 Directors in TV Shows
Data2.loc[Data2['type']=='TV Show'].groupby('director')['title'].count().sor
```

Out[95]:
```
director
Unkown_director     49358
Noam Murro            189
Thomas Astruc         160
Damien Chazelle       104
Houda Benyamina       104
Laïla Marrakchi       104
Alan Poul             104
Rob Seidenglanz       103
Alejandro Lozano       90
Jay Oliva              81
Manolo Caro            78
Name: title, dtype: int64
```

Ignore Unknown_director as they missing data.

These are the top 10 director worked on maximum TV shows.

In [96]:
```
#Top 10 Directors in Movie
Data2.loc[Data2['type']=='Movie'].groupby('director')['title'].count().sort_
```

Out[96]:
```
director
Unkown_director        1285
Martin Scorsese         419
Youssef Chahine         409
Cathy Garcia-Molina     356
Steven Spielberg        355
Lars von Trier          336
Raja Gosnell            308
Tom Hooper              306
McG                     293
David Dhawan            270
Wilson Yip              260
Name: title, dtype: int64
```

Ignore Unknown_director as they missing data.

These are the top 10 director worked on maximum Movies.

5. Which genre movies are more popular or produced more

Hint : We want you to apply the word cloud on the genre columns to know which kind of genre is produced

In [130]:
```
#filter only movies type as per question
Data5 = Data2[Data2['type']=='Movie']
```

In [131]:
```
Genre = Data5['listed_in'].str.cat(sep = ', ')
```

In [132]:
```
Genre2 = Genre.split(', ')
```

In [133]:
```
Genre2 = set(Genre2)
Genre2 = list(Genre2)
```

In [134]:
```
Genre2 = ' '.join(Genre2)
```

In [135]:
```
Genre2
```

Out[135]:
```
'Comedies Horror Movies LGBTQ Movies Movies Anime Features Dramas Action &
Adventure Stand-Up Comedy Cult Movies Sci-Fi & Fantasy Faith & Spiritualit
y Romantic Movies Independent Movies Thrillers Music & Musicals Classic Mo
vies Sports Movies Documentaries Children & Family Movies International Mo
vies'
```

In [136]:
```
from wordcloud import WordCloud
import matplotlib.pyplot as plt
text = Genre2  # Replace this with your text or load a file
wordcloud = WordCloud(width=800, height=400, max_words=200, background_color
plt.figure(figsize=(10, 8))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.show()
```

So we can conlclude that International Movies, LGBTQ, Comedies,Horror, Features, Drama etc are the most produced Genre.

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)

Hint : We want you to get the difference between the columns having date added information and release year information and get the mode of difference. This will give an insight into what will be the better time to add in Netflix

We will add a coloumn of year from date added and then subtract it from released year

In [143]:
```
#filter only movies type as per question
Data6 = Data2[Data2['type']=='Movie']
```

In [144]:
```
Data6['added_Year'] = pd.to_datetime(Data6['date_added']).dt.year
```

```
C:\Users\Dhrubo\AppData\Local\Temp\ipykernel_7444\715241312.py:1: SettingW
ithCopyWarning:
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-doc
s/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://
pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-
view-versus-a-copy)
  Data6['added_Year'] = pd.to_datetime(Data6['date_added']).dt.year
```

In [148]:
```
Data6['diff'] = Data6['added_Year']-Data6['release_year']
```

```
C:\Users\Dhrubo\AppData\Local\Temp\ipykernel_7444\3228998302.py:1: Setting
WithCopyWarning:
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-doc
s/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://
pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-
view-versus-a-copy)
  Data6['diff'] = Data6['added_Year']-Data6['release_year']
```

In [149]:
```
Data6['diff'].mode()
```

Out[149]:
```
0    0
Name: diff, dtype: int64
```

Hence, Movies have added in the same year it was/is released.

In [ ]:

In [ ]:

In [ ]:

In [ ]: