1. Data Visualization of Netflix Dataset

In [4]:

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

In [5]:

```
# Load dataset
df= pd.read_csv("D:\Desktop\Projects\DataViz\dataset/netflix_titles.csv")
df.head()
```

Out[5]:

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

                                                                                                                                            Before
                                                                                                                                 Children
                              Norm of the   Richard        Alan Marriott,   United States,                                                   planning an
                                                                                            September                                   & Family
0   81145628     Movie       North: King    Finn, TimAndrew Toth, Brian       India, South    2019      TV-PG   90 min   awesome
                                                                                             9, 2019                                     Movies,
                         Sized Adventure      Maltby        Dobson, Cole...    Korea, China                                                       wedding
                                                                                                                                 Comedies
                                                                                                                                              for his gra...

                                                                                                                                               Jandino
                                Jandino:                                                                                                      Asporaat
                                                                                United      September                 TV-                Stand-Up
1   80117401     Movie      Whatever it     NaN      Jandino Asporaat 2016   94 min  riffs on the Kingdom     9, 2016  MA        Comedy
                                   Takes

                                                                                                                                               challen
```

ges of ra...

2	70234439	TV Show	Transformers Prime	NaN	Peter Cullen, Sumalee Montano, Frank Welker, J...	United States	September 8, 2018	2013	TV-Y7-FV	1 Season	Kids' TV	With the help of three human allies, the Autob...
3	80058654	TV Show	Transformers: Robots in Disguise	NaN	Will Friedle, Darren Criss, Constance Zimmer, ...	United States	September 8, 2018	2016	TV-Y7	1 Season	Kids' TV	When a prison ship crash unleashes hundreds of...

show_id

type

title

director

cast

country

date_added

release_year

r

ating

duration

listed_in

description

4	80125979	Movie	#realityhigh	Fernando Lebrija	Nesta Cooper, Kate Walsh, John Michael Higgins...	United States	September 8, 2017	2017	TV-14	99 min	Comedies	When nerdy high schooler Dani finally attracts...

Exploratory Data Analysis

In [6]:

```
# Checking missing values
df.isnull().sum()
```

Out[6]:

```
show_id            0
type
0 title
0 director
1969 cast
570 country
476 date_added
11 release_year
0 rating
10 duration
0 listed_in
0 description
0 dtype: int64
```

In [7]:

```
# Basic information of the data
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6234 entries, 0 to 6233
Data columns (total 12 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   show_id       6234 non-null   int64
1   type          6234 non-null   object
2   title         6234 non-null   object
3   director      4265 non-null   object
4   cast          5664 non-null   object
5   country       5758 non-null   object
6   date_added    6223 non-null   object  7   release_year
    6234 non-null   int64
8   rating        6224 non-null   object
9   duration      6234 non-null   object
10  listed_in     6234 non-null   object
11  description   6234 non-null   object dtypes: int64(2),
    object(10) memory usage: 584.6+ KB
```

In [8]:

```
# unique values of the data
df.nunique()
```

Out[8]:

```
show_id         6234
type               2
title           6172
director        3301
cast            5469
country          554
date_added      1524
release_year      72
rating            14
duration         201
listed_in        461
description     6226
dtype: int64
```

In [9]:

```
# Drop missing values
df=df.dropna()
df.shape
```

Out[9]:

```
(3774, 12)
```

In [10]:

```
# Convert date_added column to datetime format
df['date_added'] = pd.to_datetime(df['date_added'])
df['day_added'] = df['date_added'].dt.day
df['month_added'] = df['date_added'].dt.month
df['year_added'] = df['date_added'].dt.year
```

In [11]:

```
df.dtypes
```

Out[11]:

```
show_id                  int64
type                    object
title                   object
director                object
cast                    object
country                 object
date_added      datetime64[ns]
release_year             int64
rating                  object
duration                object
listed_in               object
description             object
day_added                int64
month_added              int64
year_added               int64
dtype: object
```

In [12]:

```
df_movies=df[df["type"]=="Movie"]
df_tvshows=df[df["type"]=="TV Shows"]
```

In [77]:

```
v = df[['cast', 'director']]
v
```

Out[77]:

	cast	director
0	Alan Marriott, Andrew Toth, Brian Dobson, Cole...	Richard Finn, Tim Maltby
4	Nesta Cooper, Kate Walsh, John Michael Higgins...	Fernando Lebrija
6	Antonio Banderas, Dylan McDermott, Melanie Gri...	Gabe Ibáñez
7	Fabrizio Copano	Rodrigo Toro, Francisco Schultz
9	James Franco, Kate Hudson, Tom Wilkinson, Omar...	Henrik Ruben Genz
...	...	...
6142	Mel Giedroyc, Sue Perkins, Mary Berry, Paul Ho...	Andy Devonshire
6158	Cristina Vee, Bryce Papenbrook, Keith Silverst...	Thomas Astruc
6167	Saif Ali Khan, Nawazuddin Siddiqui, Radhika Ap...	Vikramaditya Motwane, Anurag Kashyap
6182	Ho-dong Kang, Soo-geun Lee, Sang-min Lee, Youn...	Jung-ah Im
6213	Ali Atay, Melis Birkan, Serkan Keskin, Ahmet M...	Onur Ünlü

3774 rows × 2 columns

In [65]:

```
#replacing rating
rating_replace = {
    'TV-PG': 'Older Kids',
    'TV-MA': 'Adults',
    'TV-Y7-FV': 'Older Kids',
    'TV-Y7': 'Older Kids',
    'TV-14': 'Teens',
    'R': 'Adults',
    'TV-Y': 'Kids',
    'NR': 'Adults',
    'PG-13': 'Teens',
    'TV-G': 'Kids',
    'PG': 'Older Kids',
    'G': 'Kids',
    'UR': 'Adults',
    'NC-17': 'Adults'
}
df['rating'] = df['rating'].replace(rating_replace)
```

Data Visualization most watched type on netflix (movies or tv shows)

In [13]:

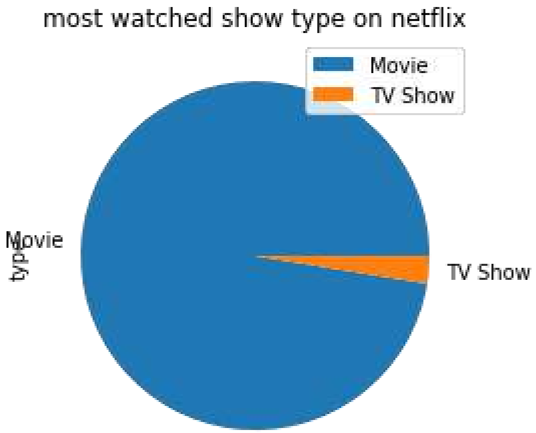
```
#most watched type on netflix
df.type.value_counts()
```

Out[13]:

```
Movie      3678
TV Show      96
Name: type, dtype: int64
```

```
df.type.value_counts().plot(kind='pie')
plt.title("most watched show type on
netflix") plt.legend() plt.show()
```

In [14]:



Movies are watched by maximum audience of Netflix.

Top 10 directors on Netflix

In [15]:

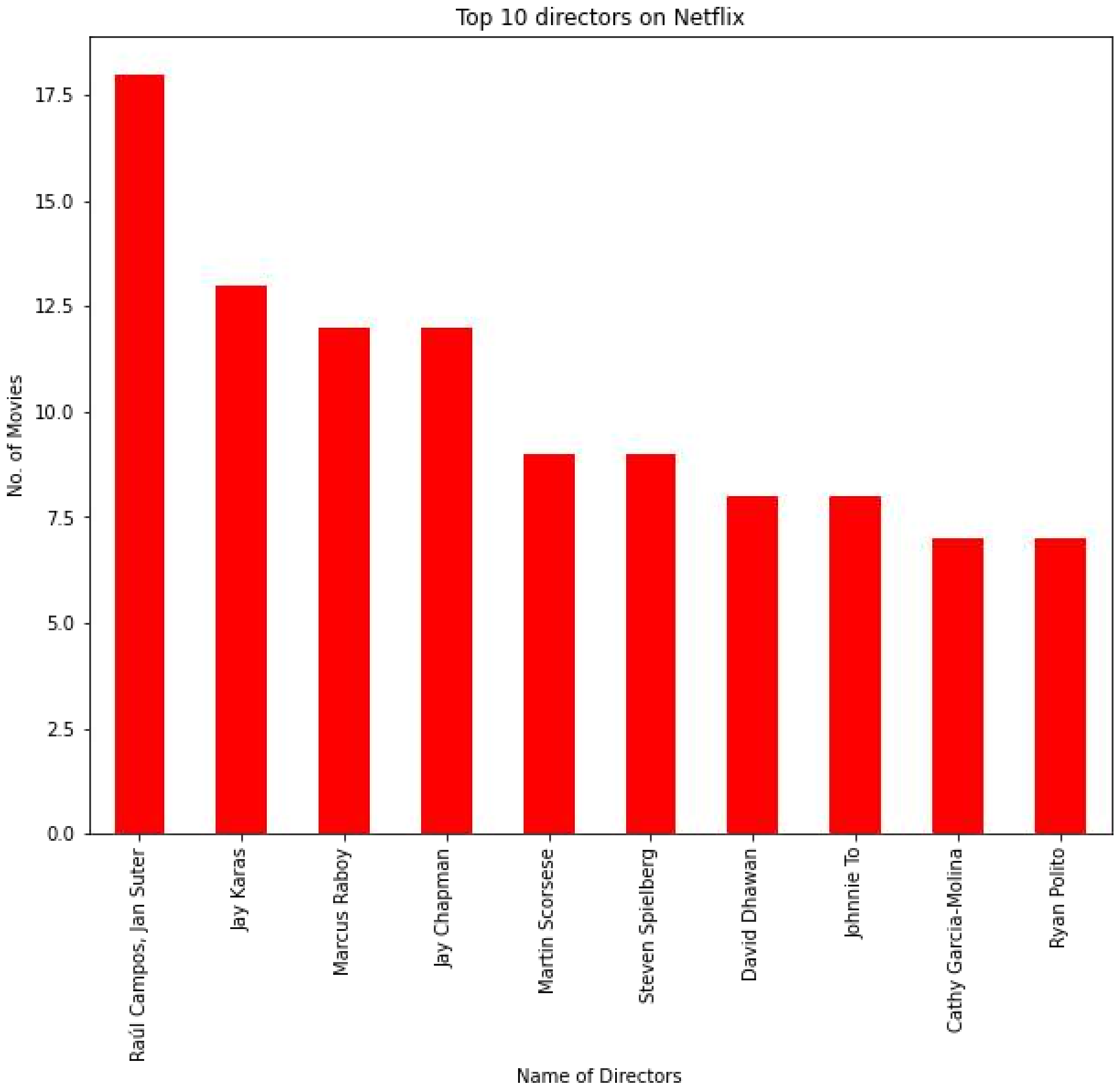
```
# Top 10 directors of netflix
df.director.value_counts().head(10)
```

Out[15]:

```
Raúl Campos, Jan Suter    18
Jay Karas                 13
Marcus Raboy              12
Jay Chapman               12
Martin Scorsese            9
Steven Spielberg           9
David Dhawan               8
Johnnie To                 8
Cathy Garcia-Molina        7
```

```
Ryan Polito                7 Name: director, dtype: int64 In [18]:
```

```
plt.figure(figsize=(10,8))
df.director.value_counts().head(10).plot(kind='bar',color='r
ed') plt.title("Top 10 directors on Netflix")
plt.xlabel("Name of Directors") plt.ylabel("No. of Movies")
plt.show()
```


Top 10 directors on Netflix
No. of Movies
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
Raúl Campos, Jan Suter
Jay Karas
Marcus Raboy
Jay Chapman
Martin Scorsese
Steven Spielberg
David Dhawan
Johnnie To
Cathy Garcia-Molina
Ryan Polito
Name of Directors

Top 10

In [19]:

```
# Which country releases most movies in a year?
df.country.value_counts()
```

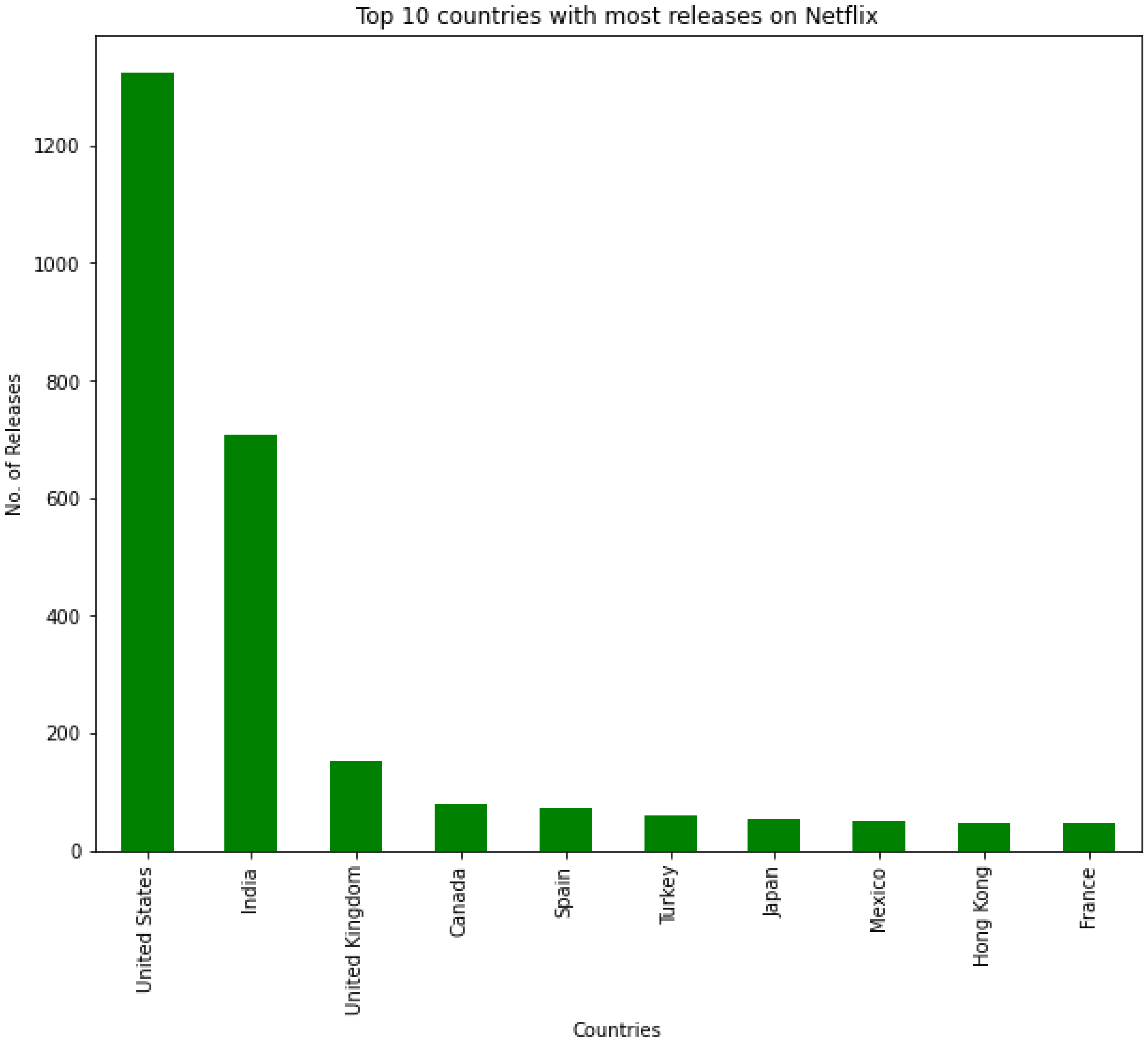
Countries with most releases

Out[19]:

```
United States                     1323
India                              707
United Kingdom                     152
Canada                              78
Spain                               72
...
South Korea, Czech Republic          1
Spain, France, Uruguay               1
Chile, Argentina                     1
Czech Republic, Slovakia             1
United Kingdom, Russia               1
Name: country, Length: 433, dtype: int64
```

In [23]:

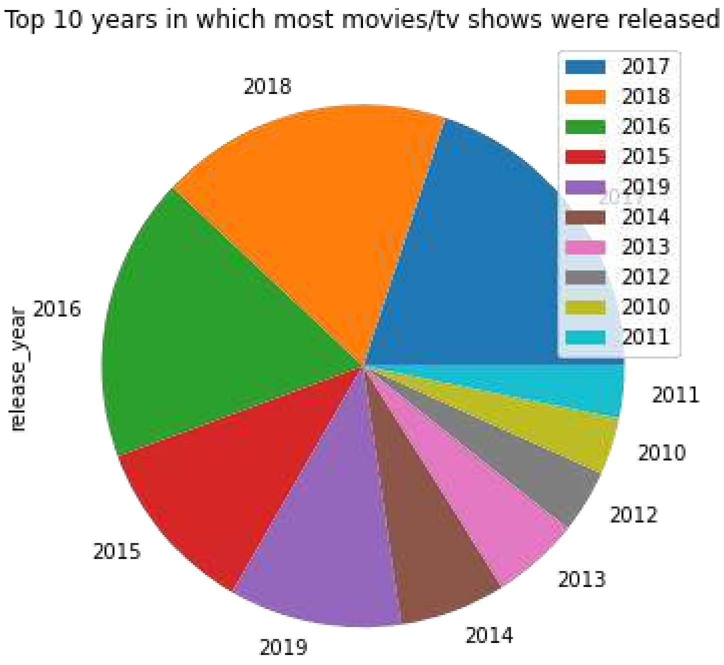
```
# Top 10 countries releases show on Netflix
plt.figure(figsize=(10,8))
df.country.value_counts().head(10).plot(kind='bar',color='green')
plt.title("Top 10 countries with most releases on Netflix")
plt.xlabel("Countries") plt.ylabel("No. of Releases") plt.show()
```


Top 10 countries with most releases on Netflix
No. of Releases
0
200
400
600
800
1000
1200
United States
India
United Kingdom
Canada
Spain
Turkey
Japan
Mexico
Hong Kong
France
Countries

Top 10 years in which most movies/tv shows were released

In [48]:

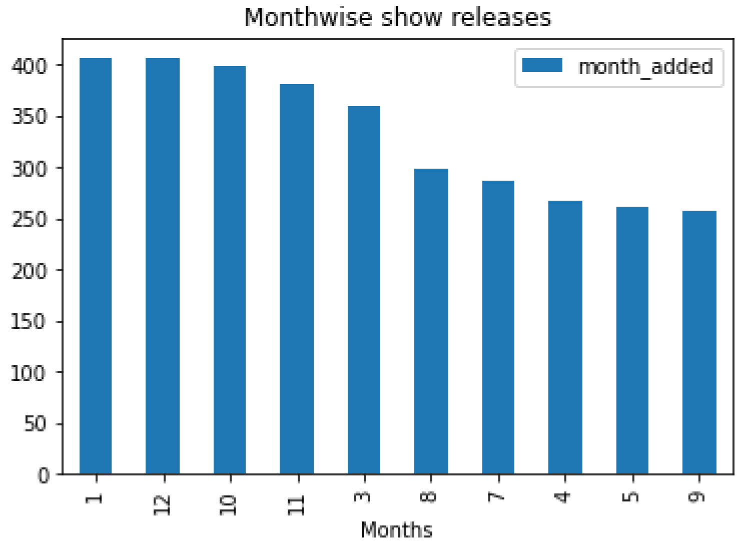
```
# Top 10 years in which most movies/tv shows were released
plt.figure(figsize=(8,6))
df_movies.release_year.value_counts().head(10).plot(kind='pie')
plt.title("Top 10 years in which most movies/tv shows were released")
plt.legend()
plt.show()
```



Month in which most of the shows were released

In [49]:

```
# In which month most of the shows release?
df['month_added'].value_counts().head(10).plot(kind='bar')
plt.title("Monthwise show releases") plt.legend()
plt.xlabel("Months") plt.show()
```

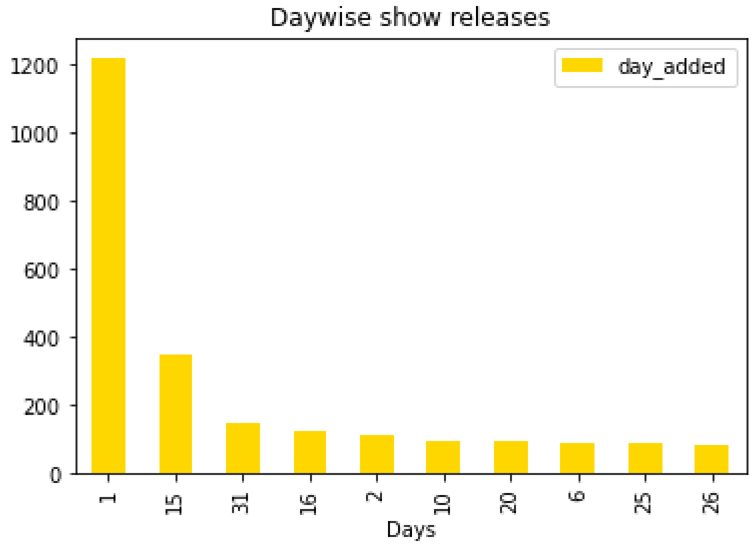


January and December are the two months when most of the shows are released

Date on which most shows were released

```
# On which date most of the shows release?
df['day_added'].value_counts().head(10).plot(kind='bar',
color='gold') plt.title("Daywise show releases") plt.legend()
plt.xlabel("Days") plt.show()
```

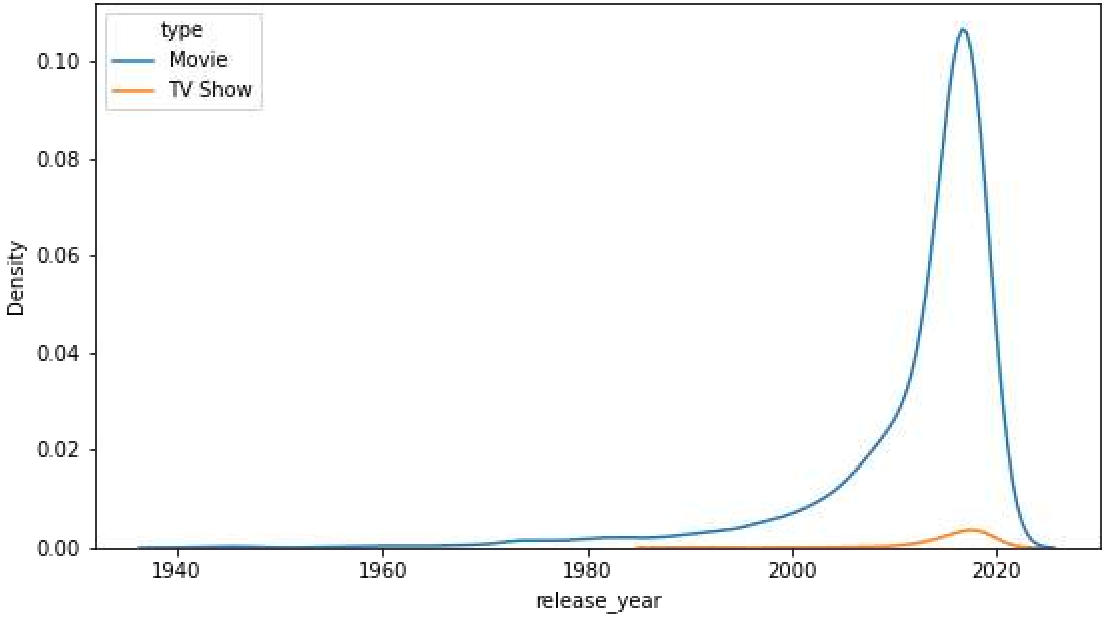
In [51]:



First of every month is the day when most of the shows are released.

In [64]:

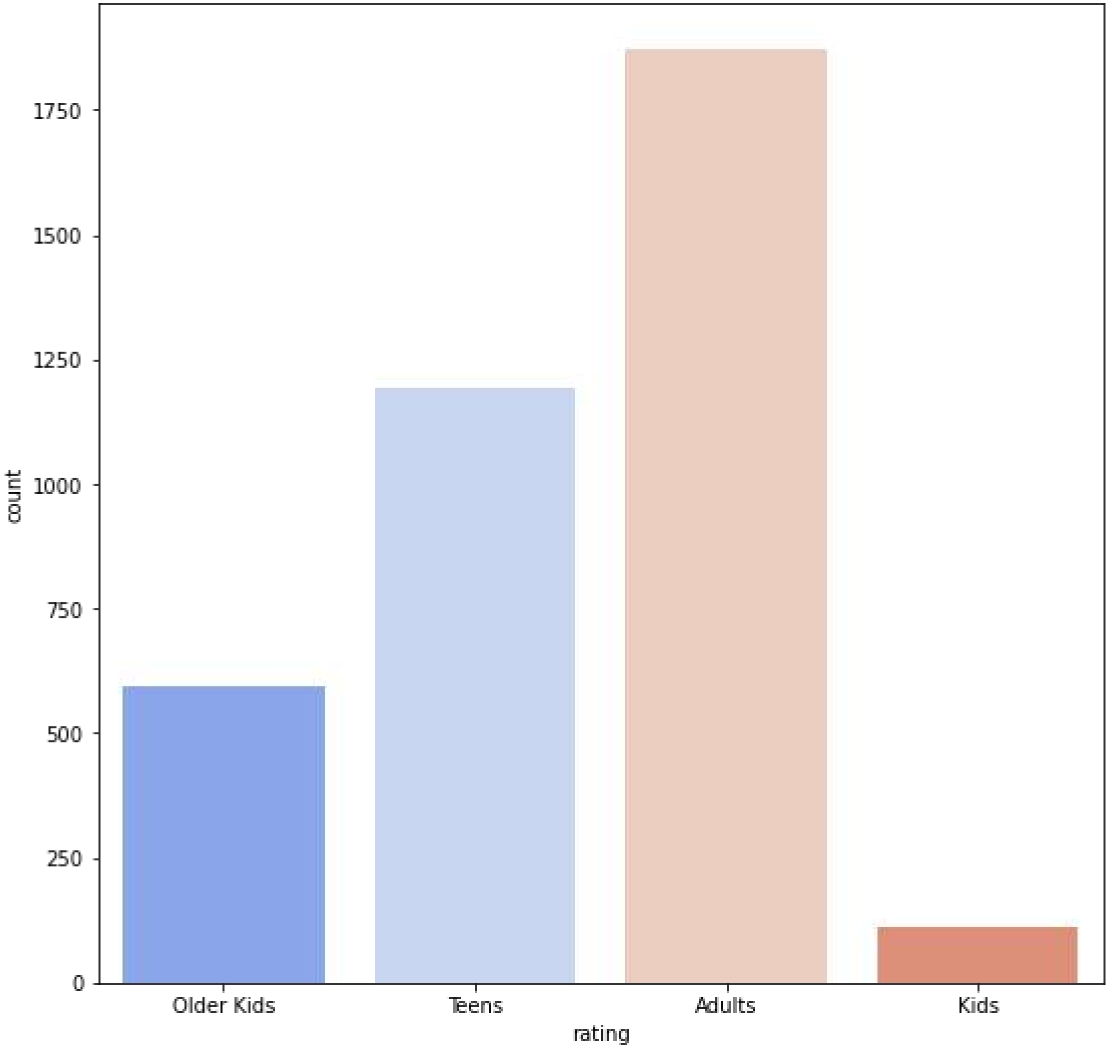
```
## PLOT FOR MAXIMUM RELEASE ACCORDING TO YEAR.
plt.figure(figsize = (9, 5))

sns.kdeplot(data = df, x = df['release_year'], hue = df['type'])
plt.show()
```



IN THIS, YEARS BETWEEN 2015 - 2020 SEEMS TO HAVE A MAXIMUM NUMBER OF RELEASES. THIS ALSO GIVES AN ADDITIONAL INSIGHT THAT, NETFLIX HAS SHOWN INTEREST IN TV SHOWS, WHICH WE CAN SEE IT SLIGHTLY INCREASING AROUND YEAR 2018 - 2020. THOUGH IT DOES HAVE GREATER AMOUNT OF CONTENT IN MOVIES, ALSO SUBTLE AMOUNT IN TV SHOWS.

In [66]:

```
#types of contents plt.figure(figsize = (9, 9))
sns.countplot(x = df['rating'], palette =
'coolwarm') plt.show()
```



‘Adults’ seems to be utmost, followed by ‘Teens’ and ‘Older Kids’.

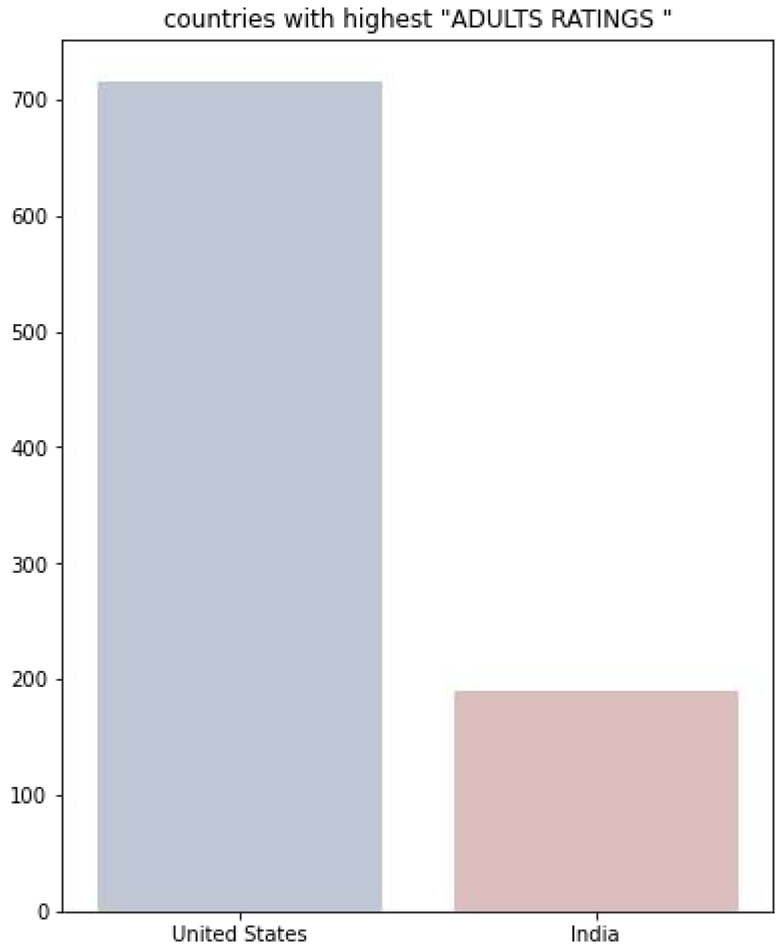
Countries with different highest rated content.

In [68]:

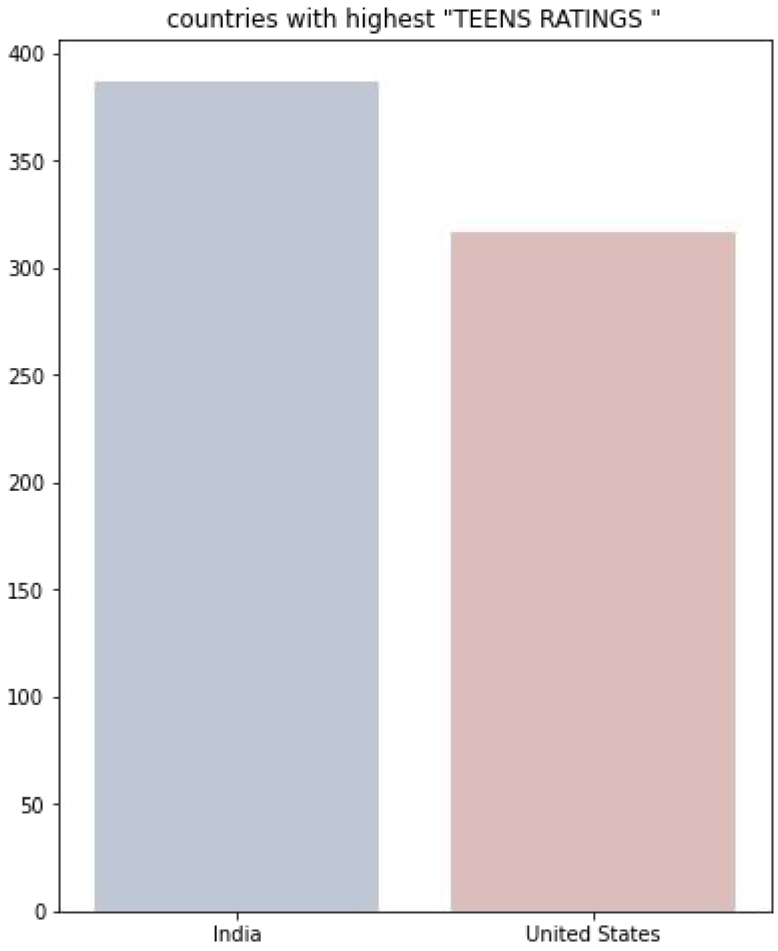
```
plt.figure(figsize = (14, 8))
plt.subplot(1, 2, 1)
e = df[df['rating']== 'Adults']['country'].value_counts().head(2)
sns.barplot(x =e.index, y= e.values, palette = 'vlag')
plt.title('countries with highest "ADULTS RATINGS \"')


plt.subplot(1, 2, 2)
f      = df[df['rating']== 'Teens']['country'].value_counts().head(2)
plt.title('countries with highest "TEENS RATINGS \"') sns.barplot(x =
f.index, y =f.values,  palette = 'vlag')

plt.figure(figsize = (16, 8))
plt.subplot(2, 3, 3 )
g      = df[df['rating']== 'Older Kids']['country'].value_counts().head(2)
plt.title('countries with highest "OLDER KIDS RATINGS\" ')

sns.barplot(x = g.index,y = g.values,  palette = 'vlag')
```

Out[68]:

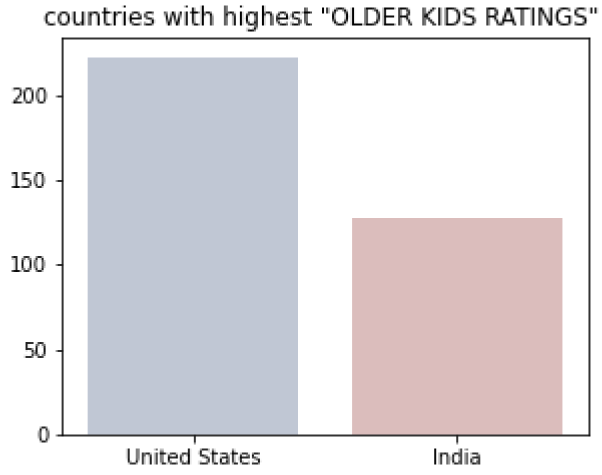
```
<AxesSubplot:title={'center':'countries with highest "OLDER KIDS RATINGS" '}>
```



countries with highest "ADULTS RATINGS "



countries with highest "TEENS RATINGS "



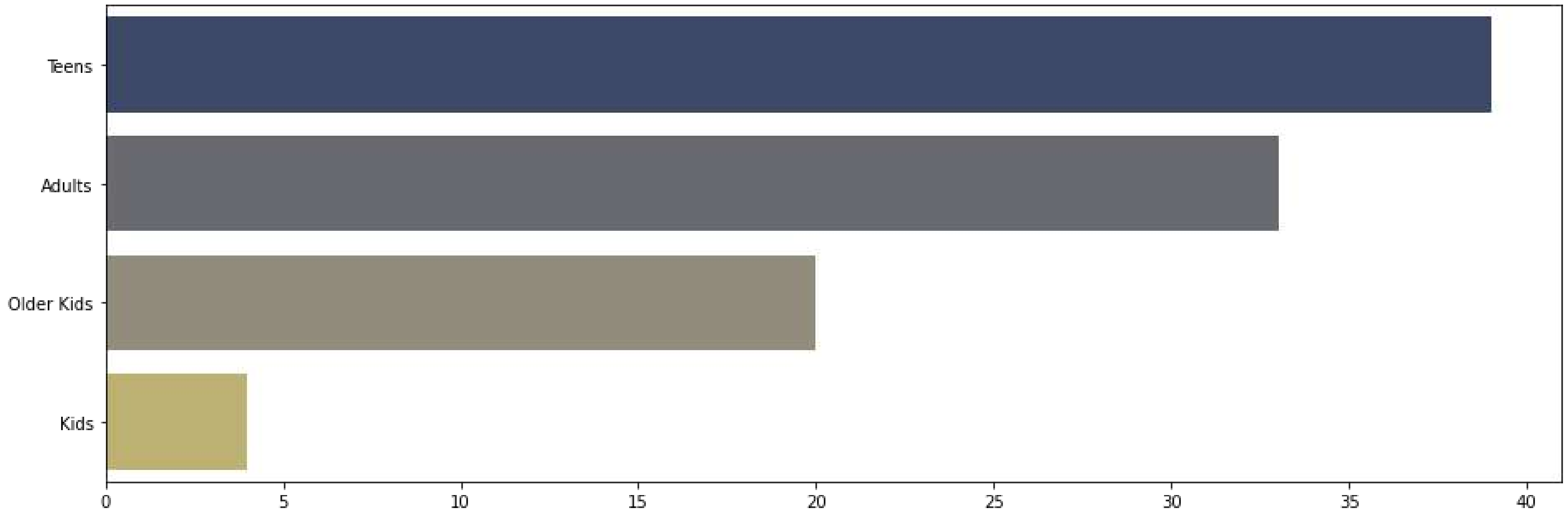
TV Show analysis based on ratings

In [86]:

```
h = df[df['type'] == 'TV Show']['rating'].value_counts()
h
a4_dims = (15.7, 5.27)
plt.figure(figsize= (a4_dims))
sns.barplot(x = h.values, y = h.index, orient = "h", palette = 'cividis')
```

Out[86]:

```
<AxesSubplot:>
```



Tv shows are maximally rated with 'adult' followed by ' teens' and 'older kids'

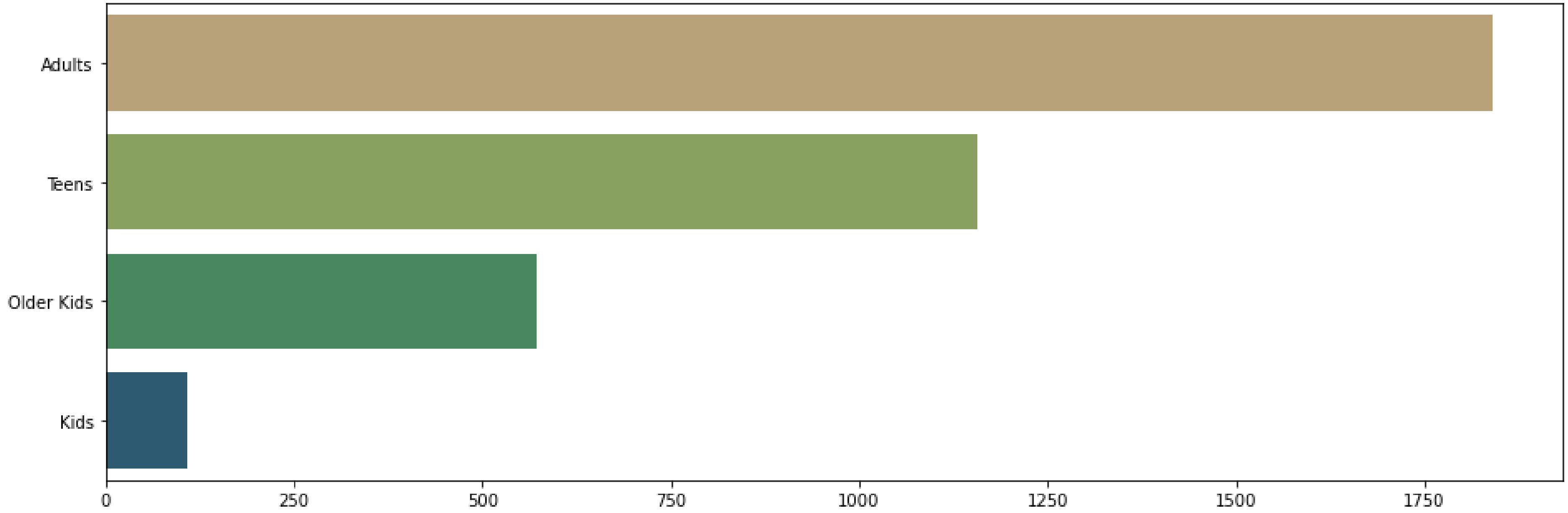
```
h = df[df['type'] == 'Movie']['rating'].value_counts()
h
a4_dims = (15.7, 5.27) plt.figure(figsize= (a4_dims)) sns.barplot(x =
h.values, y = h.index, orient = "h", palette = 'gist_earth_r')
```

Movie analysis based on ratings

In [84]:

```
<AxesSubplot:>
```

Out[84]:



Here also, the movies are rated with 'adults' followed by ' teens' and 'older kids'

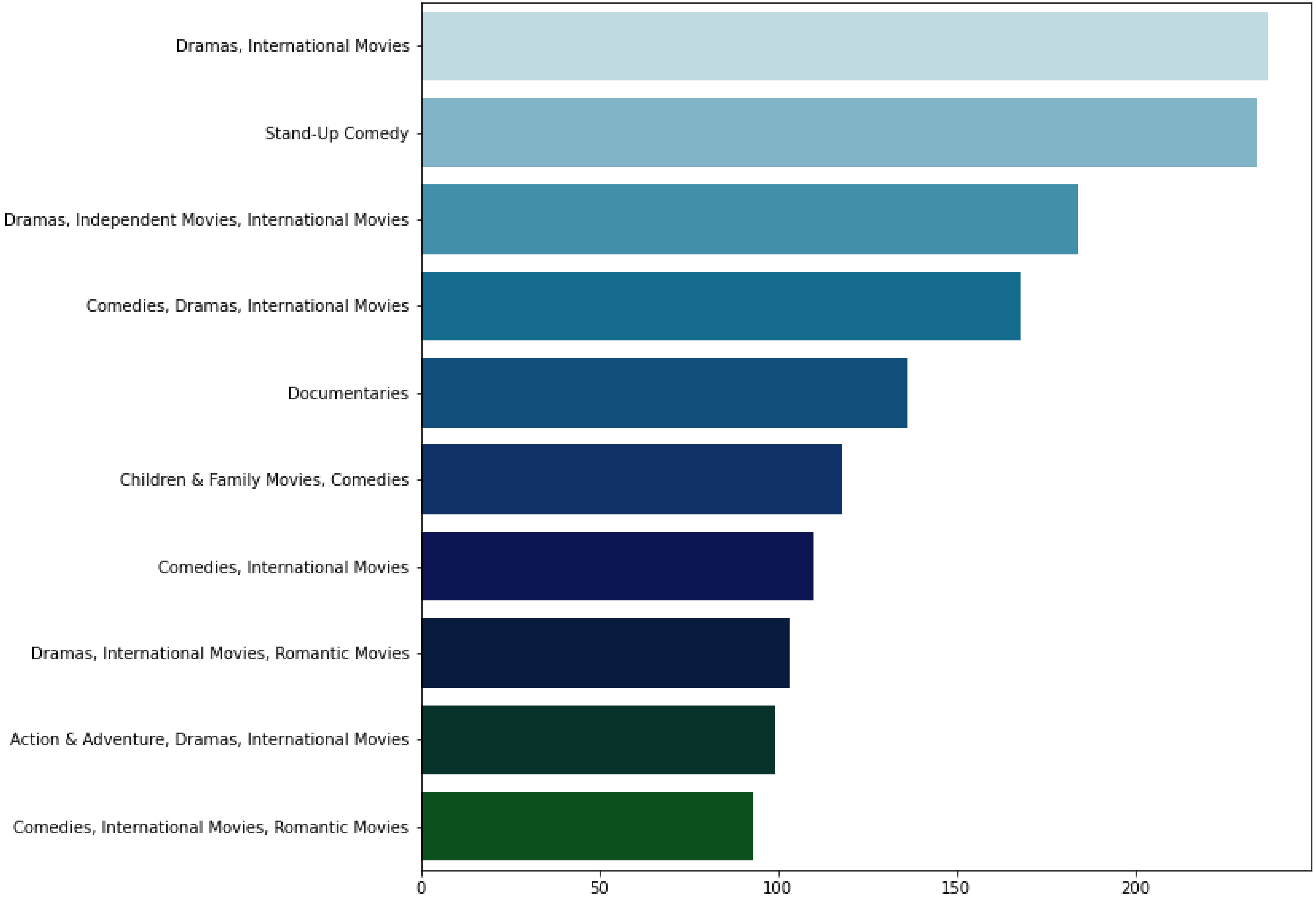
Top 10 Genres.

In [71]:

```
genres = df['listed_in'].value_counts().head(10) plt.figure(figsize =
(10, 10))
sns.barplot(x = genres.values,y = genres.index, palette = 'ocean_r')
genres
```

Out[71]:

```
Dramas, International Movies                       237
Stand-Up Comedy                                    234
Dramas, Independent Movies, International Movies   184
Comedies, Dramas, International Movies             168
Documentaries                                      136
Children & Family Movies, Comedies                 118
Comedies, International Movies                     110
Dramas, International Movies, Romantic Movies      103
Action & Adventure, Dramas, International Movies    99
Comedies, International Movies, Romantic Movies     93
Name: listed_in, dtype: int64
```



These are the top 10 genres which is widely available. Netfix has a merely good collections in dramas , followed by comedies and documentries.