In [1]:

```
import warnings
warnings.filterwarnings("ignore")
import pandas as pd
from wordcloud import WordCloud
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.graph_objects as go
from plotly.offline import init_notebook_mode, iplot
```

In [2]:

```
df = pd.read_csv('Netflix.txt')
```

In [3]:

```
#filling missing values
df['director'].fillna('Unknown Director',inplace=True)
df['duration'].fillna(0,inplace=True)
df['rating'].fillna('Unknown Rating',inplace=True)
df['country'].fillna('Unknown Country',inplace=True)
df['cast'].fillna('Unknown Actor',inplace =True)
```

In [4]:

```
def split_a_str(s):
    return str(s).split(', ')
df["cast"] = df.cast.apply(split_a_str)
df["country"] = df.country.apply(split_a_str)
df["director"] = df.director.apply(split_a_str)
df["listed_in"] = df.listed_in.apply(split_a_str)
df = df.explode("cast")
df = df.explode("country")
df = df.explode("director")
df = df.explode("listed_in")
```

In [5]:

```
#2 (a)
movies_df = df[df['type'] == 'Movie']
movie_counts = movies_df.groupby('country')['title'].nunique().reset_index()
movie_counts = movie_counts.sort_values(by='title', ascending=False).head(10)
movie_counts
#Insights
#USA is the top movie produced in the netflix dataset.USA produced 2731 movies.
```

Out[5]:

	country	title
114	United States	2751
43	India	962
112	United Kingdom	532
116	Unknown Country	440
20	Canada	319
34	France	303
36	Germany	182
100	Spain	171
51	Japan	119
23	China	114

In [6]:

```
#2(b)
tv_df = df[df['type'] == 'TV Show']
tv_counts = tv_df.groupby('country')['title'].nunique().reset_index()
tv_counts = tv_counts.sort_values(by='title', ascending=False).head(10)
tv_counts
#Insights
#USA is the top TV shows produced in the netflix dataset. USA produced 938 TV Shows
```

Out[6]:

	country	title
63	United States	938
64	Unknown Country	391
62	United Kingdom	272
30	Japan	199
52	South Korea	170
8	Canada	126
19	France	90
25	India	84
57	Taiwan	70
2	Australia	66

In [74]:

```
#3(a)
df["date_added"] = pd.to_datetime(df['date_added'])
df['year_added'] = df['date_added'].dt.year
df['month_added'] = df['date_added'].dt.month
df['week_added'] = df['date_added'].dt.week
movies = df[df['type'] == 'Movie']
movies_by_week = movies.groupby('week_added')['title'].count().reset_index()
best_movie_week = movies_by_week[movies_by_week['title'] == movies_by_week['title'].max()]
tv_shows_df = df[df['type'] == 'TV Show']
tv_shows_by_week = tv_shows_df.groupby('week_added')['title'].count().reset_index()
best_tv_show_week = tv_shows_by_week[tv_shows_by_week['title'] == tv_shows_by_week['title'].max()]
print("Best week to release movie : ",best_movie_week.iloc[0][0].astype(int))
print("Best week to release TV Shows : ",best_tv_show_week.iloc[0][0].astype(int))
#Insights
#Every 1st week is the best time to release movies
#Every 2nd week is the best time to release TV shows
```

```
Best week to release movie :  1
Best week to release TV Shows :  27
```

In [66]:

```
#3(b)
movies_by_month = movies.groupby('month_added')['title'].count().reset_index()
tv_shows_by_month = tv_shows_df.groupby('month_added')['title'].count().reset_index()
best_movie_month = movies_by_month[movies_by_month['title'] == movies_by_month['title'].max()]
best_tv_show_month = tv_shows_by_month[tv_shows_by_month['title'] == tv_shows_by_month['title'].max()]
print("Best month to release movie : ",best_movie_month.iloc[0][0].astype(int))
print("Best month to release TV Shows : ",best_tv_show_month.iloc[0][0].astype(int))
#Insights
#Every 7th month is the best time to release movies
#Every 12th month is the best time to release TV shows
```

```
Best week to release movie :  7
Best week to release TV Shows :  12
```

In [70]:

```
#4(a and b)
seti = df[['director', 'cast', 'title', 'type']]
a = seti.groupby("cast")['title'].nunique().sort_values(ascending=False).head(10)
d=seti.groupby("director")['title'].nunique().sort_values(ascending=False).head(10)
print("Top 10 Actor who have appeared in most movies or TV shows")
print(a)
print()
print("Top 10 Directors who have appeared in most movies or TV shows")
print(d)
#Insights
#Anupam Kher is the actor who appeared most movies or TV shows
#Rajiv Chilaka is the director appeared most movies or TV shows
```

```
Top 10 Actor who have appeared in most movies or TV shows
cast
Unknown Actor       825
Anupam Kher          43
Shah Rukh Khan       35
Julie Tejwani        33
Naseeruddin Shah     32
Takahiro Sakurai     32
Rupa Bhimani         31
Om Puri              30
Akshay Kumar         30
Yuki Kaji            29
Name: title, dtype: int64

Top 10 Directors who have appeared in most movies or TV shows
director
Unknown Director       2634
Rajiv Chilaka            22
Jan Suter                21
Raúl Campos              19
Marcus Raboy             16
Suhas Kadav              16
Jay Karas                15
Cathy Garcia-Molina      13
Jay Chapman              12
Martin Scorsese          12
Name: title, dtype: int64
```

In [10]:

```
df.rename(columns={'listed_in':'genre'},inplace=True)
```

In [11]:
```
#5
all_genres = ' '.join(df['genre'].dropna())

# Generate a WordCloud object
wordcloud = WordCloud(width=800, height=400, background_color='white').generate(all_genres)

# Plot the WordCloud image
plt.figure(figsize=(10, 5))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.show()
#Insights
#"International Movies" is most appeared in Netflix dataset
```

International Movies
TV Shows
Dramas International
Movies Comedies
Action Adventure
Movies Dramas
International TV
Family Movies
Children Family
Independent Movies
Shows TV
TV Dramas
Comedies International
Romantic Movies
Movies International
Shows International
Dramas Independent
Comedies Dramas
Music Musicals
Horror Movies
Comedies Children
Fantasy Action
Crime TV
Kids' TV
Sci Fi
Thrillers Dramas
Movies Action
Movies Thrillers
Anime Series
Adventure Comedies
Romantic TV
Shows Spanish
TV Action
Spanish Language
Classic Movies
TV Comedies
Movies Romantic
Dramas Crime
Movies Children

In [73]:
```
import pandas as pd

# Assuming you have a DataFrame named 'df' with columns 'date_added' and 'release_year'
# Convert 'date_added' to datetime format
df['date_added'] = pd.to_datetime(df['date_added'])

# Calculate the time difference in days
df['days_to_addition'] = (df['date_added'] - pd.to_datetime(df['release_year'].astype(str) + '-01-01')).dt.day

# Calculate the mode of the difference
mode_days_to_addition = df['days_to_addition'].mode()[0].astype(int)

print(f"The mode of the difference between date added and release year is {mode_days_to_addition} days.")
#Insights
#After 547 days the movie will be added to Netflix after the release of the movie
```

```
The mode of the difference between date added and release year is 547 days.
```

In [13]:
```
nd=pd.read_csv('Netflix.txt')
```

In [14]:
```
si=nd.show_id.value_counts().sort_values(ascending=True).head(10)
si
#Insights
#There are 8807 show id's are present in netflix data
```

Out[14]:
```
s1     1
s6     1
s7     1
s8     1
s9     1
s10    1
s11    1
s12    1
s13    1
s14    1
Name: show_id, dtype: int64
```

In [15]:
```
ty=nd.type.value_counts()
ty
#Insights
#There are 2676 TV shows and 6131 movies are present in netflix data
```

Out[15]:
```
Movie      6131
TV Show    2676
Name: type, dtype: int64
```

In [16]:

```
ti=nd.title.value_counts().head(10)
ti
#Insights
#There are 8807 TV shows and movies are present in netflix data, for example Iam presenting 10 titles
```

Out[16]:

```
Dick Johnson Is Dead                1
Ip Man 2                            1
Hannibal Buress: Comedy Camisado    1
Turbo FAST                          1
Masha's Tales                       1
Chelsea Does                        1
Ricardo O'Farrill Abrazo Genial     1
Ip Man                              1
Tom Segura: Mostly Stories          1
Team Foxcatcher                     1
Name: title, dtype: int64
```

In [17]:

```
di=nd.director.value_counts()
di
#Insights
#There are 4528 directors are present in netflix data and some directors directed multiple movies and TV shows
```

Out[17]:

```
Rajiv Chilaka                     19
Raúl Campos, Jan Suter            18
Marcus Raboy                      16
Suhas Kadav                       16
Jay Karas                         14
                                  ..
Raymie Muzquiz, Stu Livingston     1
Joe Menendez                       1
Eric Bross                         1
Will Eisenberg                     1
Mozez Singh                        1
Name: director, Length: 4528, dtype: int64
```

In [75]:

```
c=nd.cast.value_counts().head(10)
c
#Insights
#There are 4528 directors are present in netflix data and some directors directed multiple movies and TV shows
```

Out[75]:

```
David Attenborough
19
Vatsal Dubey, Julie Tejwani, Rupa Bhimani, Jigna Bhardwaj, Rajesh Kava, Mousam, Swapnil
14
Samuel West
10
Jeff Dunham
7
David Spade, London Hughes, Fortune Feimster
6
Kevin Hart
6
Craig Sechler
6
Michela Luci, Jamie Watson, Eric Peterson, Anna Claire Bartlam, Nicolas Aqui, Cory Doran, Julie Lemieux, Dere
k McGrath     6
Bill Burr
5
Iliza Shlesinger
5
Name: cast, dtype: int64
```

In [77]:

```
ct=nd.country.value_counts()
ct
#Insights
#There are 748 countries are present in netflix dataset
```

Out[77]:

```
United States                             2818
India                                      972
United Kingdom                             419
Japan                                      245
South Korea                                199
                                          ...
Romania, Bulgaria, Hungary                   1
Uruguay, Guatemala                           1
France, Senegal, Belgium                     1
Mexico, United States, Spain, Colombia       1
United Arab Emirates, Jordan                 1
Name: country, Length: 748, dtype: int64
```

In [20]:

```
d = nd.date_added.value_counts()
d
ct=nd.country.value_counts()
ct
#Insights
#There are 1767 dates are present in netflix dataset
```

Out[20]:

```
January 1, 2020      109
November 1, 2019      89
March 1, 2018         75
December 31, 2019     74
October 1, 2018       71
                    ...
December 4, 2016       1
November 21, 2016      1
November 19, 2016      1
November 17, 2016      1
January 11, 2020       1
Name: date_added, Length: 1767, dtype: int64
```

In [84]:

```
r = nd.release_year.value_counts().head()
r
#Insights
#In the given data From 1925 to 2021 year movies and TV shows are present in the Netflix dataset
```

Out[84]:

```
2018    1147
2017    1032
2019    1030
2020     953
2016     902
Name: release_year, dtype: int64
```

In [83]:

```
ra = nd.rating.value_counts().head()
ra
#Insights
#TV-MA rating is reviewed most Movies and TV Shows
```

Out[83]:

```
TV-MA    3207
TV-14    2160
TV-PG     863
R         799
PG-13     490
Name: rating, dtype: int64
```

In [85]:

```
du = nd.duration.value_counts().head()
du
# In TV shows 1 season TV Show is appeared most times.
```

Out[85]:

```
1 Season     1793
2 Seasons     425
3 Seasons     199
90 min        152
94 min        146
Name: duration, dtype: int64
```

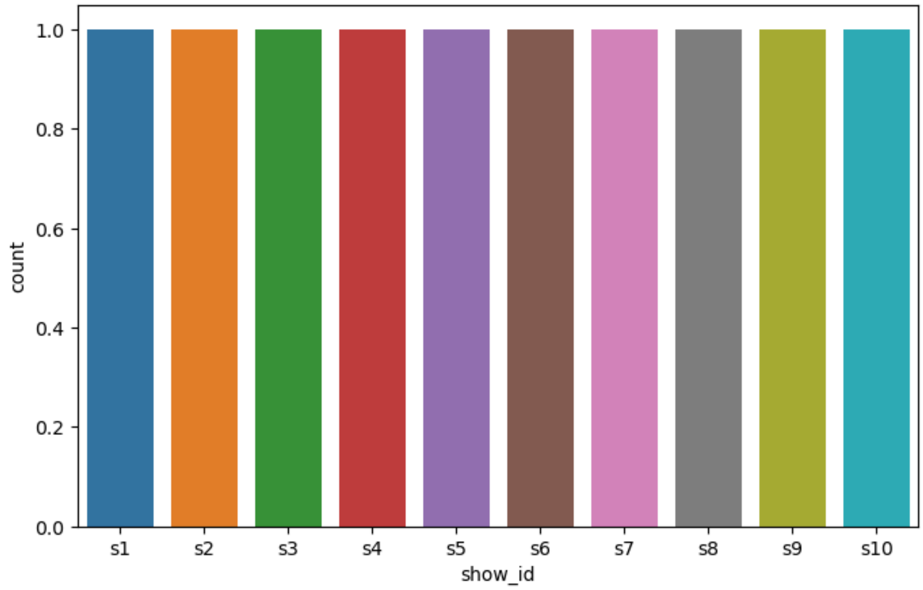
In [24]:

```
li = nd.listed_in.value_counts()
li
#Insights
#International movies appeared most in netflix dataset.
```

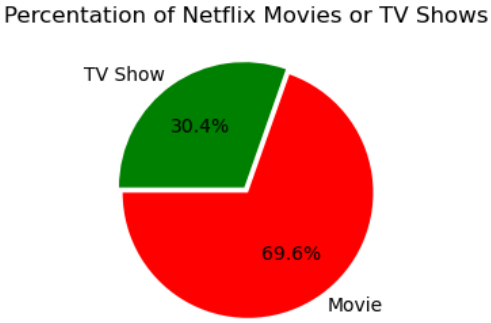
Out[24]:

```
Dramas, International Movies                          362
Documentaries                                         359
Stand-Up Comedy                                       334
Comedies, Dramas, International Movies                274
Dramas, Independent Movies, International Movies      252
                                                     ...
Kids' TV, TV Action & Adventure, TV Dramas              1
TV Comedies, TV Dramas, TV Horror                       1
Children & Family Movies, Comedies, LGBTQ Movies        1
Kids' TV, Spanish-Language TV Shows, Teen TV Shows      1
Cult Movies, Dramas, Thrillers                          1
Name: listed_in, Length: 514, dtype: int64
```

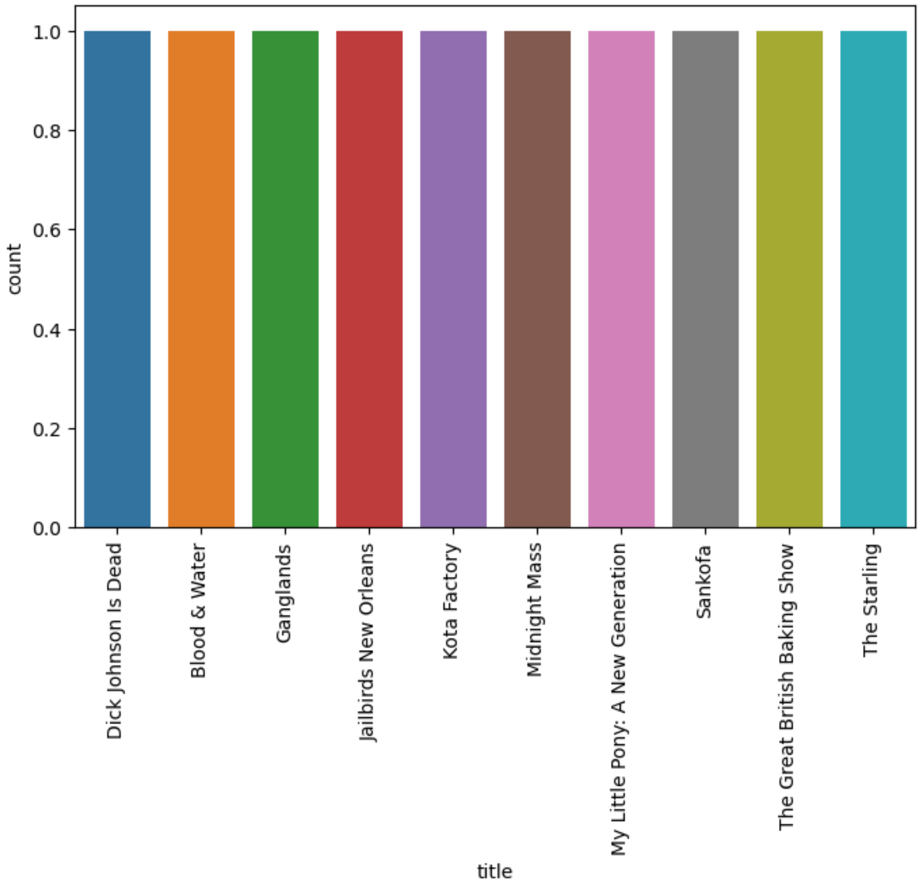
```
In [25]: plt.figure(figsize=(8,5))
         sns.countplot(data=nd.head(10),x='show_id')
         plt.show()
         #Insights
         #There are 8807 show id's are present in netflix data
```



```
In [26]: plt.figure(figsize=(6,3))
         plt.title("Percentation of Netflix Movies or TV Shows")
         g=plt.pie(nd.type.value_counts(),explode=(0.025,0.025),
         labels=nd.type.value_counts().index, colors=['red','green'],autopct='%1.1f%%',
         startangle=180)
         #Insights
         #There are 30.4% TV shows and 69.6% movies are present in netflix data
```



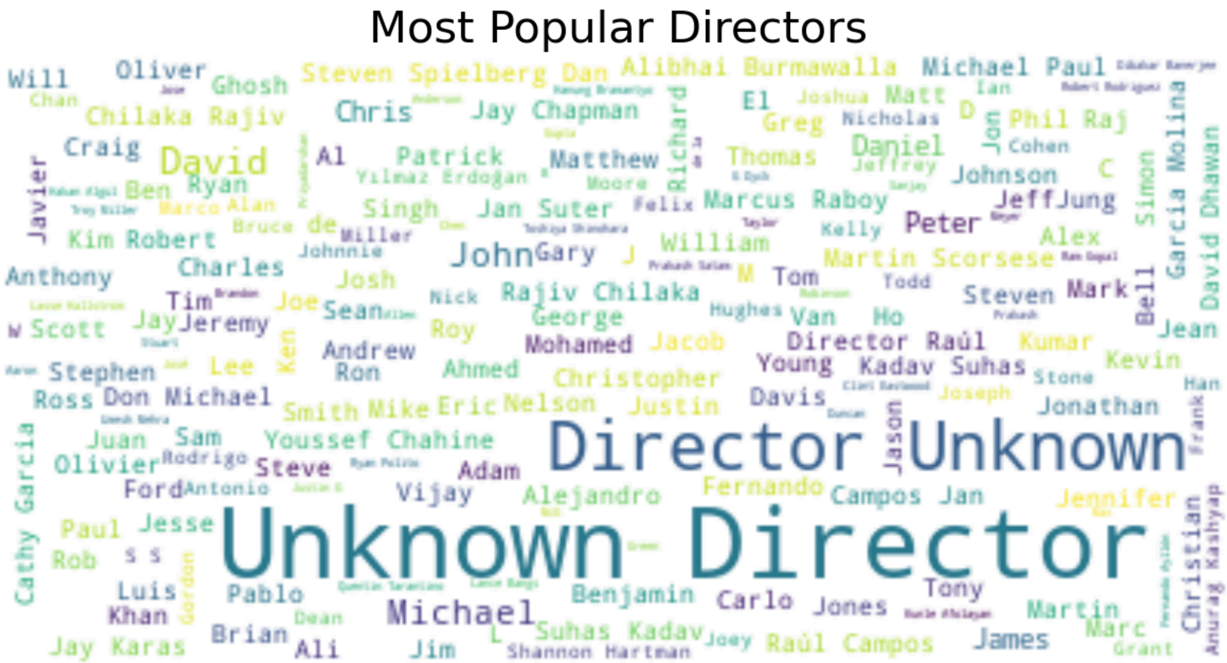
In [27]:
```
#1b
plt.figure(figsize=(8,5))
sns.countplot(data=nd.head(10),x='title')
plt.xticks(rotation=90)
plt.show()
#Insights
#There are 8807 TV shows and movies are present in netflix data, for example Iam presenting 10 titles
```



In [87]:

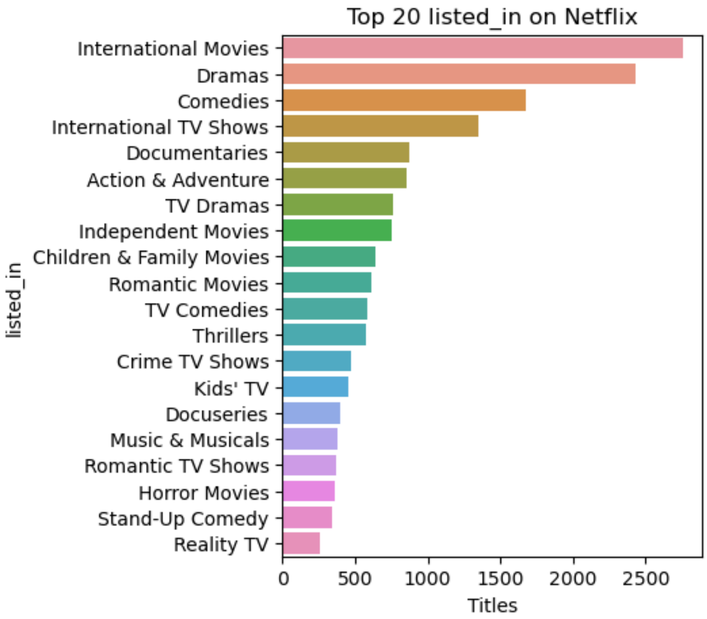
```
#1b
text = " ".join(str(each) for each in nd.director)
nd.director.fillna('Unknown Director',inplace=True)
wordcloud = WordCloud(max_words=200, background_color="white").generate(text)
plt.figure(figsize=(10,6))
plt.figure(figsize=(15,10))
plt.imshow(wordcloud, interpolation='Bilinear')
plt.title('Most Popular Directors',fontsize = 30)
plt.axis("off")
plt.show()
#Insights
#Rajiv Chilaka is the director appeared most movies or TV shows
```

```
<Figure size 1000x600 with 0 Axes>
```



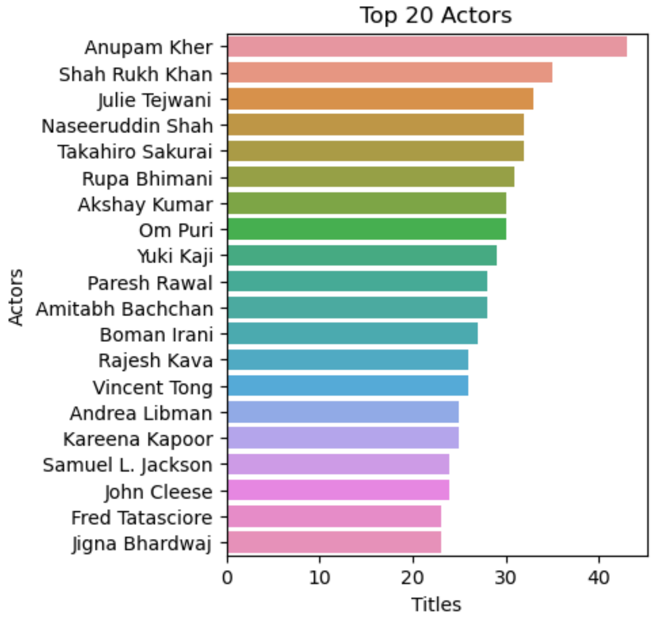
In [29]:

```
#1(B)
filtered_genres = nd.set_index('title').listed_in.str.split(', ',
expand=True).stack().reset_index(level=1, drop=True);
plt.figure(figsize=(4,5))
g = sns.countplot(y = filtered_genres,
order=filtered_genres.value_counts().index[:20])
plt.title('Top 20 listed_in on Netflix')
plt.xlabel('Titles')
plt.ylabel('listed_in')
plt.show()
#Insights
#"International Movies" are appeared most in netflix dataset.
```



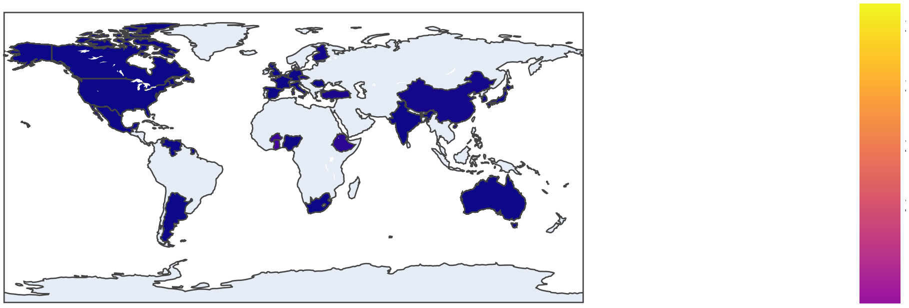
In [30]:

```
#1(B)
filtered_genres = nd.set_index('title').cast.str.split(', ',
expand=True).stack().reset_index(level=1, drop=True);
plt.figure(figsize=(4,5))
g = sns.countplot(y = filtered_genres,
order=filtered_genres.value_counts().index[:20])
plt.title('Top 20 Actors')
plt.xlabel('Titles')
plt.ylabel('Actors')
plt.show()
#Insights
#Anupam Kher is the actor who appeared most movies or TV shows
```

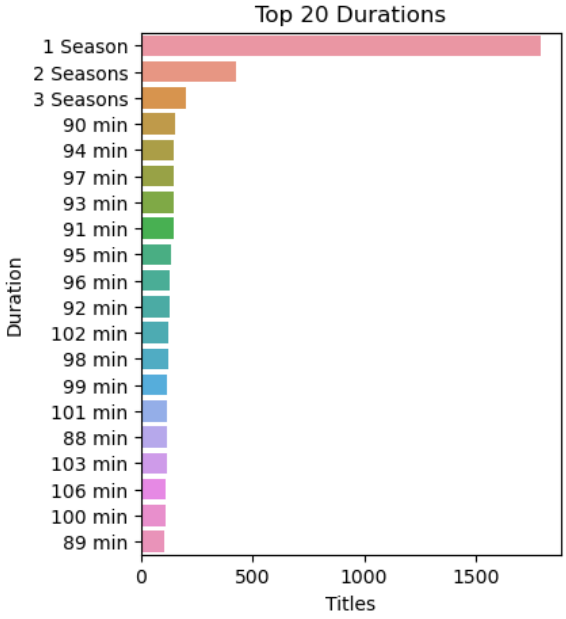


In [36]:

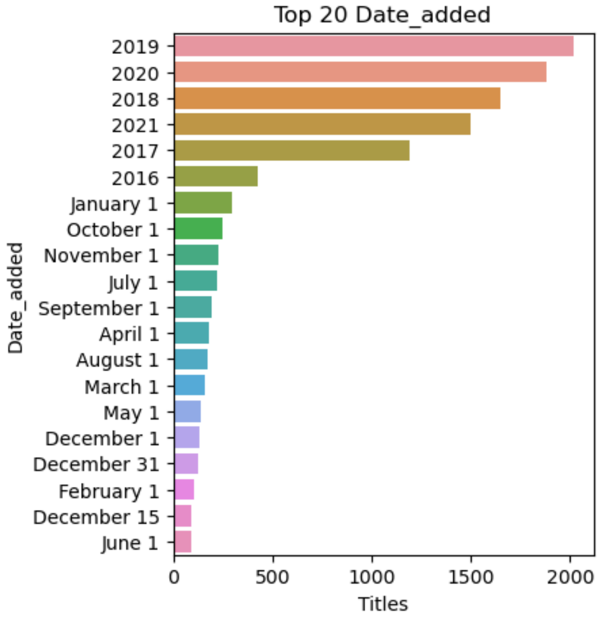
```
iltered_countries = nd.set_index('title').country.str.split(', ',
xpand=True).stack().reset_index(level=1, drop=True);
iltered_countries = filtered_countries[filtered_countries != 'Unknown country']
plot([go.Choropleth(locationmode='country names',locations=filtered_countries,z=filtered_countries.value_counts
Insights
USA is the country that produced most movies and TV Shows are present in netflix dataset
```



In [88]:
```
filtered_genres = nd.set_index('title').duration.str.split(', ', expand=True).stack().reset_index(level=1, drop=True);
plt.figure(figsize=(4,5))
g = sns.countplot(y = filtered_genres, order=filtered_genres.value_counts().index[:20])
plt.title('Top 20 Durations')
plt.xlabel('Titles')
plt.ylabel('Duration')
plt.show()
```

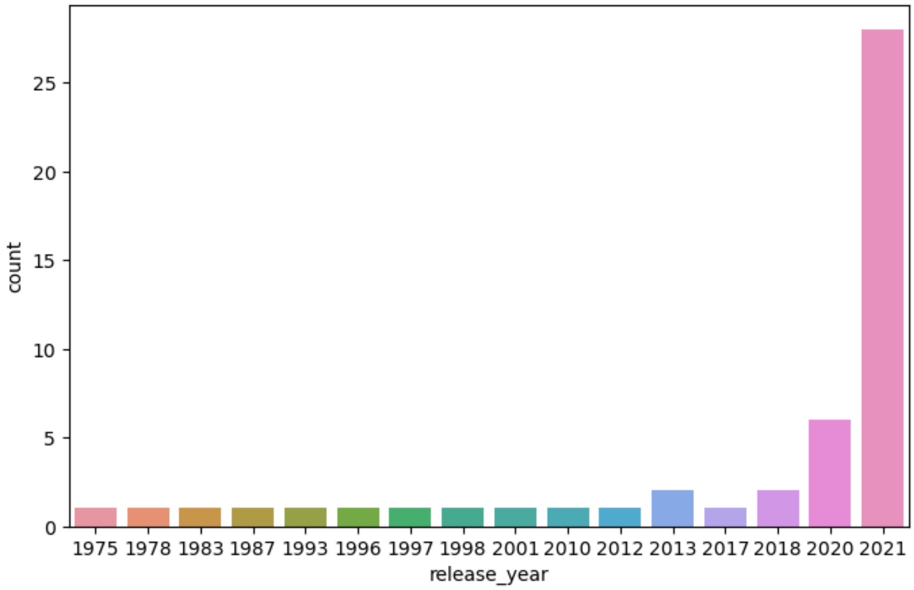


In [47]:
```
filtered_genres = nd.set_index('title').date_added.str.split(', ', expand=True).stack().reset_index(level=1, drop=True);
plt.figure(figsize=(4,5))
g = sns.countplot(y = filtered_genres, order=filtered_genres.value_counts().index[:20])
plt.title('Top 20 Date_added')
plt.xlabel('Titles')
plt.ylabel('Date_added')
plt.show()
#Insights
#In 2019, Most movies or TV shows are added in netflix.
```



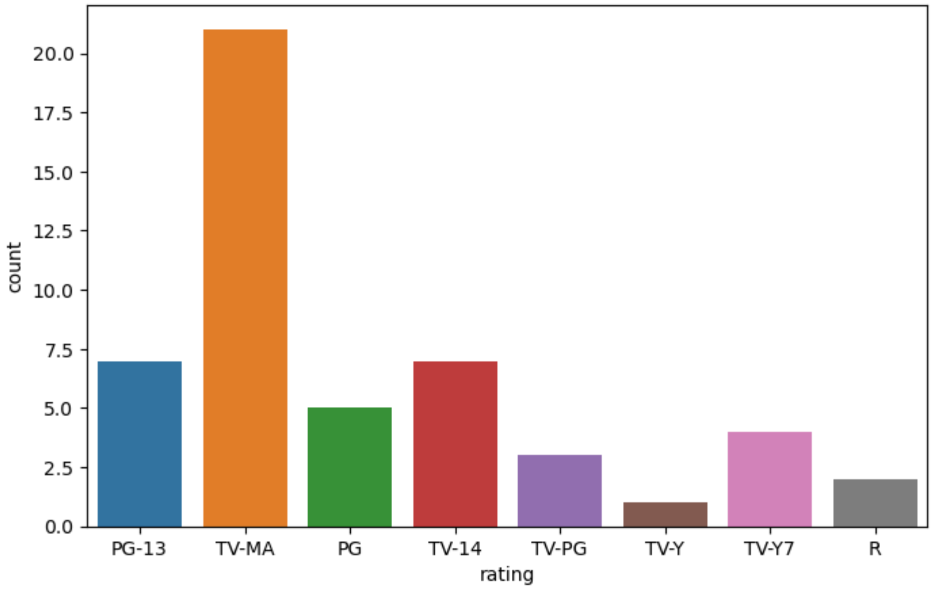
In [92]:

```
plt.figure(figsize=(8,5))
sns.countplot(data=nd.head(50),x='release_year')
plt.show()
#Insights
#In 2021, Most movies or TV shws are added in netflix dataset.
```



In [59]:

```
plt.figure(figsize=(8,5))
sns.countplot(data=nd.head(50),x='rating')
plt.show()
#Insights
#TV-MA rating is reviewed most Movies and TV Shows
```



In [ ]: