Importing the libraries

In [469]:

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

Question 1

Problem Statement & Goal

- Netflix is one of the most popular media and video streaming platforms. They have over 10000 movies or tv shows available on their platform, as of mid-2021, they have over 222M Subscribers globally
- Our goal is to analyze the data and generate insights that could help Netflix in deciding which type of shows/movies to produce and how they can grow the business in different countries

Analysing basic metrics

In [470]:

```
##loading the dataset
df = pd.read_csv("netflix.csv")
##top 5 rows
df.head()
```

Out[470]:

	show_id	type	title	director	cast	country	date_added	release_year	rating
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG 1
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV MA
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV MA
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV MA
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV MA

Question 2

Observations on the shape of data, data types of all the attributes

In [471]:

```
print(df.info())
print("Shape:",df.shape)
print("Dimension" ,df.ndim)
print("Size",df.size)
```

```
<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
None
Shape: (8807, 12)
Dimension 2
Size 105684
```

The data contains total 12 columns, only release year(numerical) is type int rest are object(categorical)

In [472]:

```
df.count() ##no of records in each column , including NaN,null values
```

Out[472]:

```
show_id         8807
type            8807
title           8807
director        6173
cast            7982
country         7976
date_added      8797
release_year    8807
rating          8803
duration        8804
listed_in       8807
description     8807
dtype: int64
```

In [473]:

```
df.nunique() ##unique values present in each column
```

Out[473]:

```
show_id         8807
type               2
title           8807
director        4528
cast            7692
country          748
date_added      1767
release_year      74
rating            17
duration         220
listed_in        514
description     8775
dtype: int64
```

Conversion of categorical attributes to 'category'

In [474]:

```
df.head(5)
```

Out[474]:

	show_id	type	title	director	cast	country	date_added	release_year	rating
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG 1
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV MA
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV MA
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV MA
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV MA

Columns type , ratings and listed_in can be put under category data type

In [475]:

```
df["type"]=df["type"].astype("category")
df["rating"]=df["rating"].astype("category")
df["listed_in"]=df["listed_in"].astype("category")
```

In [476]:

```
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   category
 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   category
 9   duration      8804 non-null   object  
 10  listed_in     8807 non-null   category
 11  description   8807 non-null   object  
dtypes: category(3), int64(1), object(8)
memory usage: 674.7+ KB
```

Missing value detection

In [477]:

```
df.isna().sum().sort_values(ascending=False) ##missing values in each column
```

Out[477]:

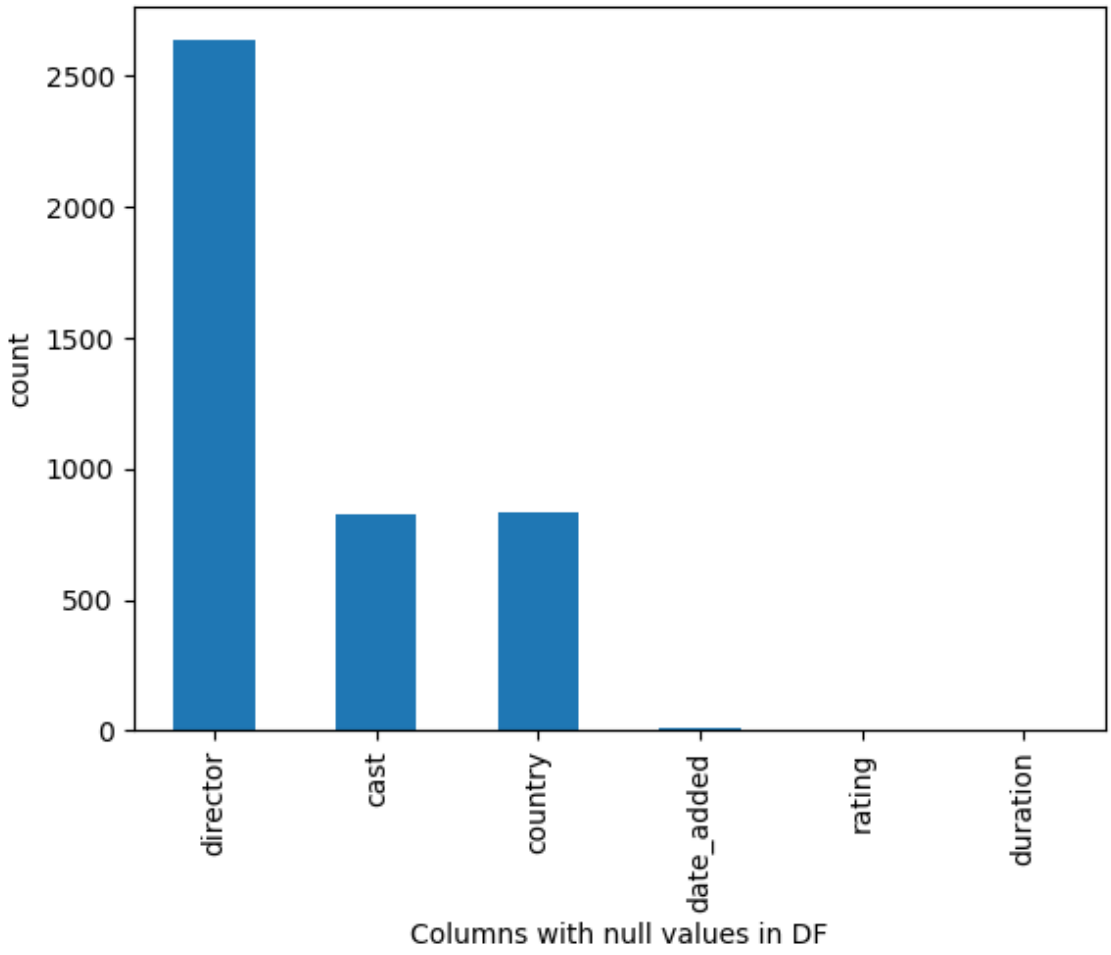
```
director        2634
country          831
cast             825
date_added        10
rating             4
duration           3
show_id            0
type               0
title              0
release_year       0
listed_in          0
description        0
dtype: int64
```

In [478]:

```
df.isna().sum()[df.isna().sum()>0].plot(kind='bar',ylabel="count",xlabel="Columns with r
```

Out[478]:

```
<Axes: xlabel='Columns with null values in DF', ylabel='count'>
```



Statistical summary

In [479]:

```
df.describe(include='all') ##statiscal summary of all columns
```

Out[479]:

	show_id	type	title	director	cast	country	date_added	release_year
count	8807	8807	8807	6173	7982	7976	8797	8807.000000
unique	8807	2	8807	4528	7692	748	1767	NaN
top	s1	Movie	Dick Johnson Is Dead	Rajiv Chilaka	David Attenborough	United States	January 1, 2020	NaN
freq	1	6131	1	19	19	2818	109	NaN
mean	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2014.180198
std	NaN	NaN	NaN	NaN	NaN	NaN	NaN	8.819312
min	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1925.000000
25%	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2013.000000
50%	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2017.000000
75%	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2019.000000
max	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2021.000000

Question 3

Value counts and unique attributes

In [480]:

```
df.nunique() ##unique values in each columns
```

Out[480]:

```
show_id         8807
type               2
title           8807
director        4528
cast            7692
country          748
date_added      1767
release_year      74
rating            17
duration         220
listed_in        514
description     8775
dtype: int64
```

In [481]:

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

Out[481]:

```
Movie      6131
TV Show    2676
Name: type, dtype: int64
```

In [482]:

```
df["country"].value_counts()
```

Out[482]:

```
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 [483]:

```
df["rating"].value_counts()
```

Out[483]:

```
TV-MA       3207
TV-14       2160
TV-PG        863
R            799
PG-13        490
TV-Y7        334
TV-Y         307
PG           287
TV-G         220
NR            80
G             41
TV-Y7-FV       6
UR             3
NC-17          3
74 min         1
84 min         1
66 min         1
Name: rating, dtype: int64
```

In [484]:

```
df["listed_in"].value_counts()
```

Out[484]:

```
Dramas, International Movies                                362
Documentaries                                               359
Stand-Up Comedy                                             334
Comedies, Dramas, International Movies                      274
Dramas, Independent Movies, International Movies            252
                                                           ...
Cult Movies, Dramas, International Movies                     1
Cult Movies, Dramas, Music & Musicals                         1
Cult Movies, Dramas, Thrillers                                1
Cult Movies, Horror Movies, Thrillers                         1
Crime TV Shows, TV Action & Adventure, TV Sci-Fi & Fantasy    1
Name: listed_in, Length: 514, dtype: int64
```

Pre-processing of the data

- Pre-processing involves unnesting of the data in columns like cast,director,country
- Also we will be filling the null/missing/NaN values
- Less significant NaN/null counts will be dropped, date_added,rating,duration

Replace blank directors and cast with "Anonymous"

In [485]:

```
df["director"]=df["director"].fillna("Anonymous")
df["cast"]=df["cast"].fillna("Cast unavailable")
```

Replace blank countries with the mode (most common) country

In [486]:

```
df['country'] = df['country'].fillna(df['country'].mode()[0])
```

Dropping other na values with lesser na count

In [487]:

```
df.dropna(inplace=True) ##total 17 rows are dropped as part of this reducing the number
```

Changing the dtype of column date_added from object to datetime and adding more columnd derived from date_added

In [488]:

```
df["date_added"] = pd.to_datetime(df['date_added'])

df['month_added']=df['date_added'].dt.month.astype('int64')
df['month_name_added']=df['date_added'].dt.month_name()
df['year_added'] = df['date_added'].dt.year.astype('int64')
df['day_added'] = df['date_added'].dt.day_name()

df.head(3)
```

Out[488]:

	show_id	type	title	director	cast	country	date_added	release_year	r
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	Cast unavailable	United States	2021-09-25	2020	
1	s2	TV Show	Blood & Water	Anonymous	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	2021-09-24	2021	
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	United States	2021-09-24	2021	

In [489]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 8790 entries, 0 to 8806
Data columns (total 16 columns):
 #   Column            Non-Null Count  Dtype         
---  ------            --------------  -----         
 0   show_id           8790 non-null   object        
 1   type              8790 non-null   category      
 2   title             8790 non-null   object        
 3   director          8790 non-null   object        
 4   cast              8790 non-null   object        
 5   country           8790 non-null   object        
 6   date_added        8790 non-null   datetime64[ns]
 7   release_year      8790 non-null   int64         
 8   rating            8790 non-null   category      
 9   duration          8790 non-null   object        
 10  listed_in         8790 non-null   category      
 11  description       8790 non-null   object        
 12  month_added       8790 non-null   int64         
 13  month_name_added  8790 non-null   object        
 14  year_added        8790 non-null   int64         
 15  day_added         8790 non-null   object        
dtypes: category(3), datetime64[ns](1), int64(3), object(9)
memory usage: 1016.7+ KB
```

In [490]:

```
df.drop_duplicates(inplace= True)
```

Unnesting/exploding the columns

Explode country

Creating a copy of orginal dataframe

In [491]:

```
df_country=df.copy()
df_country["country"] = df_country["country"].map(str)
df_country["country"]=df_country["country"].str.split(",").apply(lambda x: [e.strip() fo
df_country=df_country.explode("country",ignore_index=True)
df_country.head(3)
```

Out[491]:

	show_id	type	title	director	cast	country	date_added	release_year	ra
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	Cast unavailable	United States	2021-09-25	2020	
1	s2	TV Show	Blood & Water	Anonymous	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	2021-09-24	2021	
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	United States	2021-09-24	2021	

Explode cast

In [492]:

```
df_cast=df_country.copy()
df_cast["cast"] = df_cast["cast"].map(str)
df_cast["cast"]=df_cast["cast"].str.split(",").apply(lambda x: [e.strip() for e in x])
df_cast=df_cast.explode("cast",ignore_index=True)
df_cast.head(3)
```

Out[492]:

	show_id	type	title	director	cast	country	date_added	release_year	rati
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	Cast unavailable	United States	2021-09-25	2020	P
1	s2	TV Show	Blood & Water	Anonymous	Ama Qamata	South Africa	2021-09-24	2021	T M
2	s2	TV Show	Blood & Water	Anonymous	Khosi Ngema	South Africa	2021-09-24	2021	T M

Explode directors

In [493]:

```
df_director=df_cast.copy()
df_director["director"] = df_director["director"].map(str)
df_director["director"]=df_director["director"].str.split(",").apply(lambda x: [e.strip(
df_director=df_director.explode("director",ignore_index=True)
df_director.head(3)
```

Out[493]:

	show_id	type	title	director	cast	country	date_added	release_year	rati
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	Cast unavailable	United States	2021-09-25	2020	P
1	s2	TV Show	Blood & Water	Anonymous	Ama Qamata	South Africa	2021-09-24	2021	T M
2	s2	TV Show	Blood & Water	Anonymous	Khosi Ngema	South Africa	2021-09-24	2021	T M

Explode listed_in as Genre

In [494]:

```
df_new = df_director.copy()
df_genre = df_new.copy()
df_genre["listed_in"] = df_genre["listed_in"].map(str)
df_genre["listed_in"]=df_genre["listed_in"].str.split(",").apply(lambda x: [e.strip() fo
df_genre=df_genre.explode("listed_in",ignore_index=True)
df_genre.head(3)
```

Out[494]:

	show_id	type	title	director	cast	country	date_added	release_year	rati
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	Cast unavailable	United States	2021-09-25	2020	P
1	s2	TV Show	Blood & Water	Anonymous	Ama Qamata	South Africa	2021-09-24	2021	T M
2	s2	TV Show	Blood & Water	Anonymous	Ama Qamata	South Africa	2021-09-24	2021	T M

Question 4

Visual Analysis

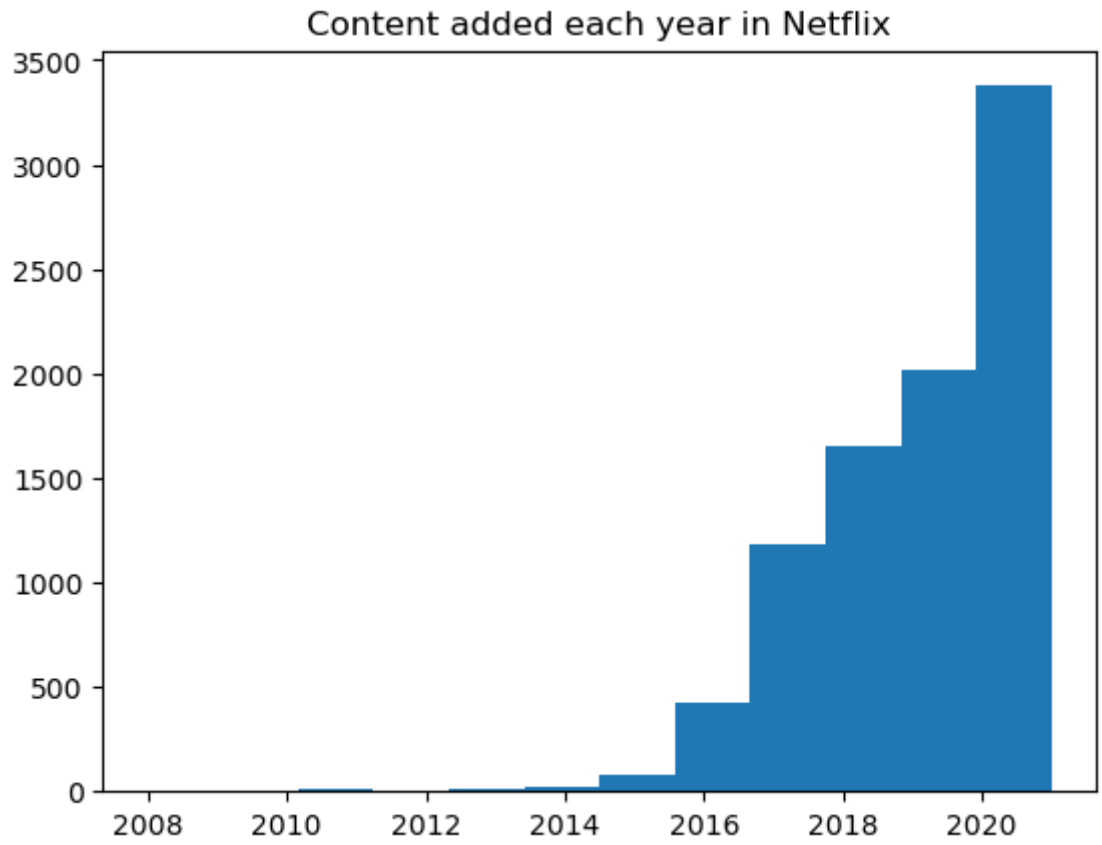
Univariate

4.1 For continuous variable(s): Distplot, countplot, histogram for univariate analysis

Content added each year in Netflix using histogram

In [495]:

```
plt.hist(df['year_added'],bins=12)
plt.title("Content added each year in Netflix")
plt.show()
```

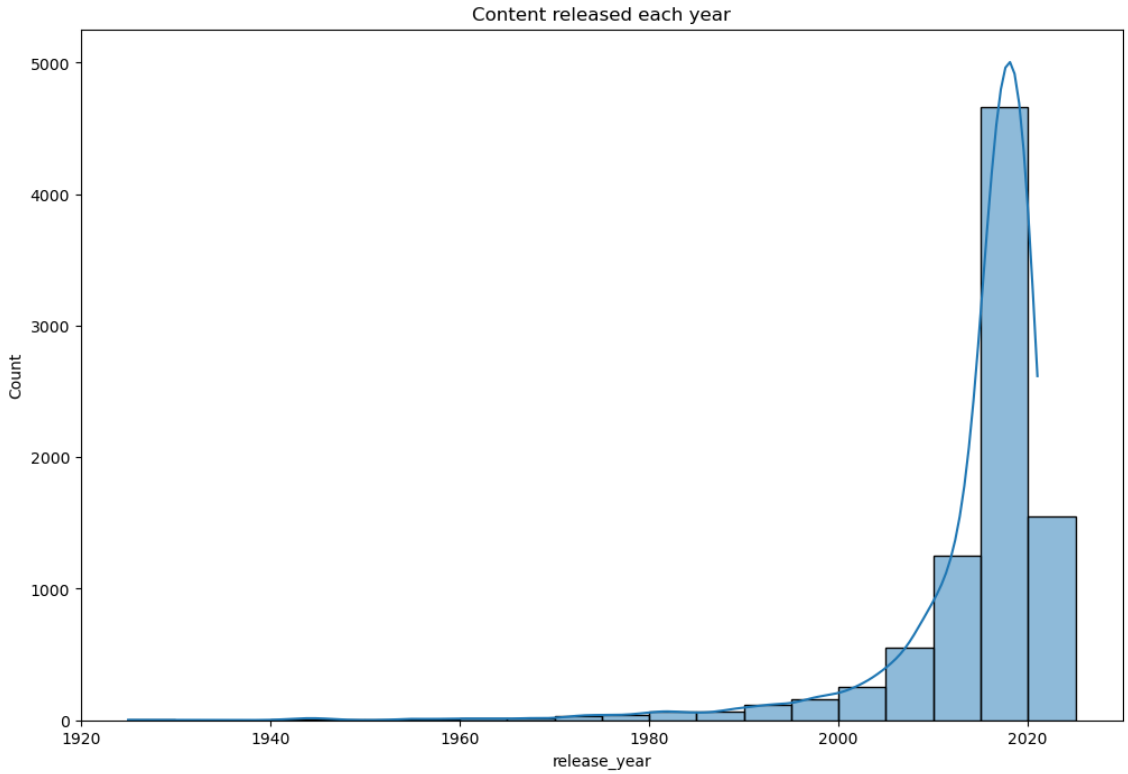


- Most of the content is added 2019 onwards, and dropped during 2020,this coincides with peak-time of COVID-19

Content released each year using histplot

In [496]:

```
plt.figure(figsize=(12,8))
sns.histplot(df['release_year'],binwidth=5,kde=True)
plt.title("Content released each year")
plt.show()
```

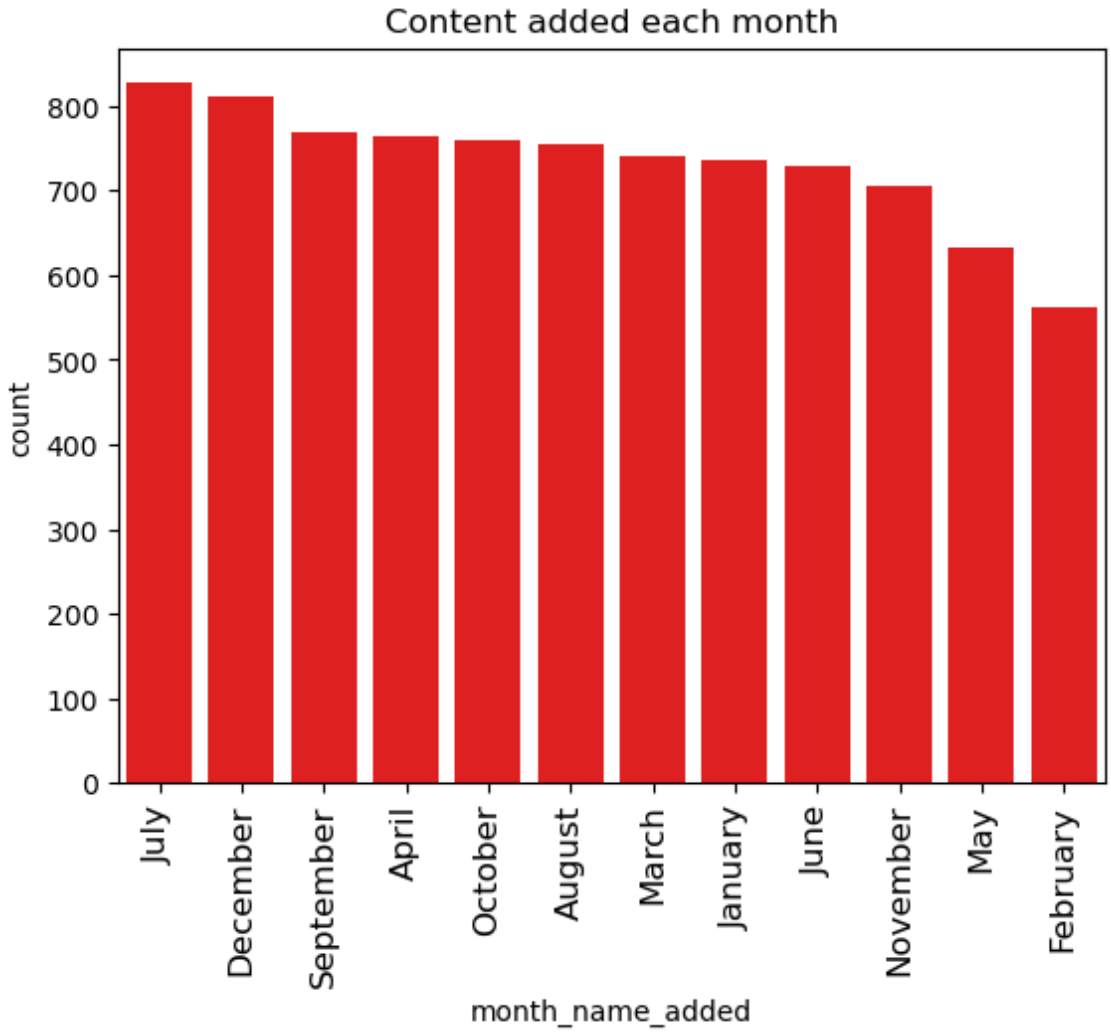


- Most of the content is relesed between 2015-2020 onwards, there is a growing trend in content released/produced from year 2000 onwards,however there is a spike during 2015-2020, this trend correlates with onset of 4G networks and social media like Instagram , Snapchat etc

Content added each month using countplot

In [497]:

```
sns.countplot(data=df,x="month_name_added",order = df['month_name_added'].value_counts()
plt.xticks(rotation=90,fontsize=12)
plt.title("Content added each month")
plt.show()
```



- Most of the content is added during July and December(Festival season in US)
- More content should be added considering the holidays/festival in each country , For eg in India appropriate content should be added around Diwali

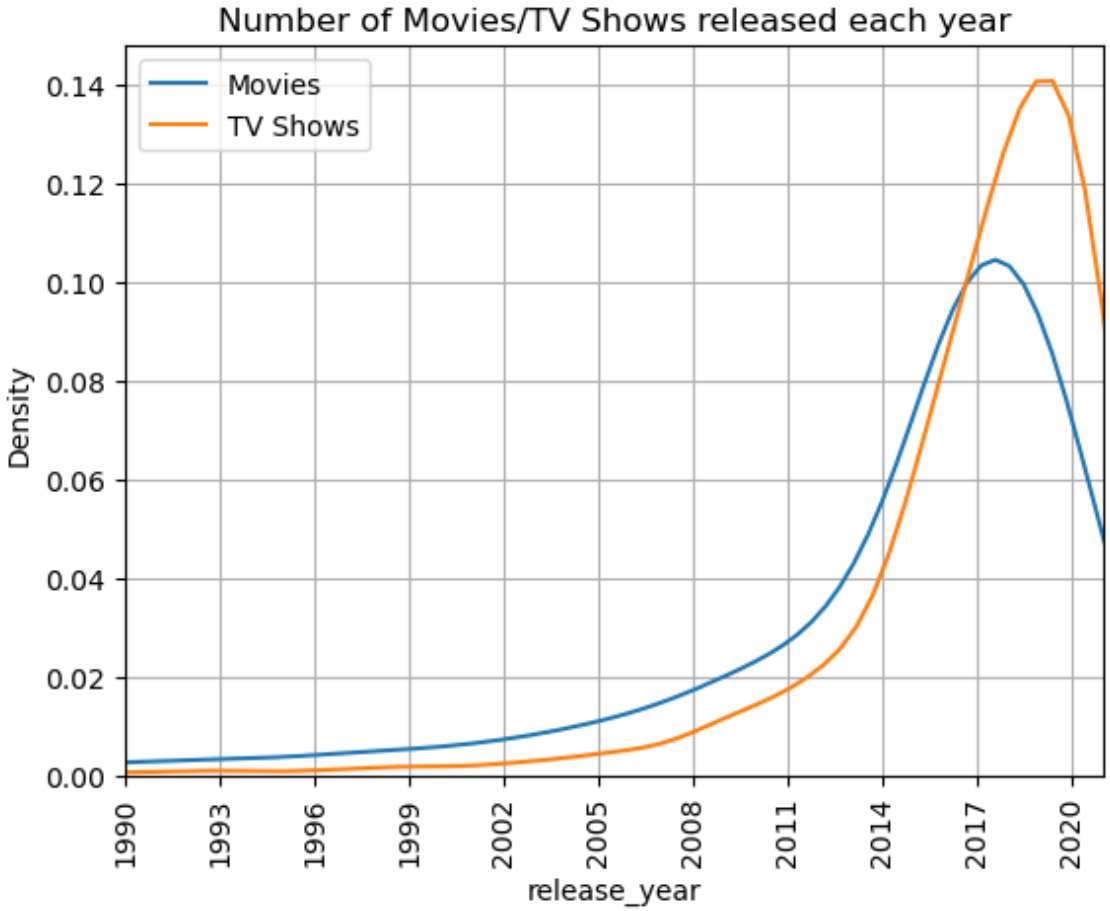
How has the number of movies released per year changed over the last 20-30 years?

In [498]:

```
movies=df.loc[df["type"]=="Movie"]
tv_shows=df.loc[df["type"]=="TV Show"]
```

In [499]:

```
sns.kdeplot(movies["release_year"])
sns.kdeplot(tv_shows["release_year"])
plt.xlim(left=1990,right=2021)
plt.xticks(np.arange(1990, 2021, 3),rotation=90)
plt.title("Number of Movies/TV Shows released each year")
plt.legend(['Movies','TV Shows'],loc='upper left')
plt.grid()
plt.show()
```



- We see a slow start for Netflix over several years.
- Things begin to pick up in 2014 and then there is a rapid increase from 2016.
- It looks like content additions have slowed down in 2020, likely due to the COVID-19 pandemic

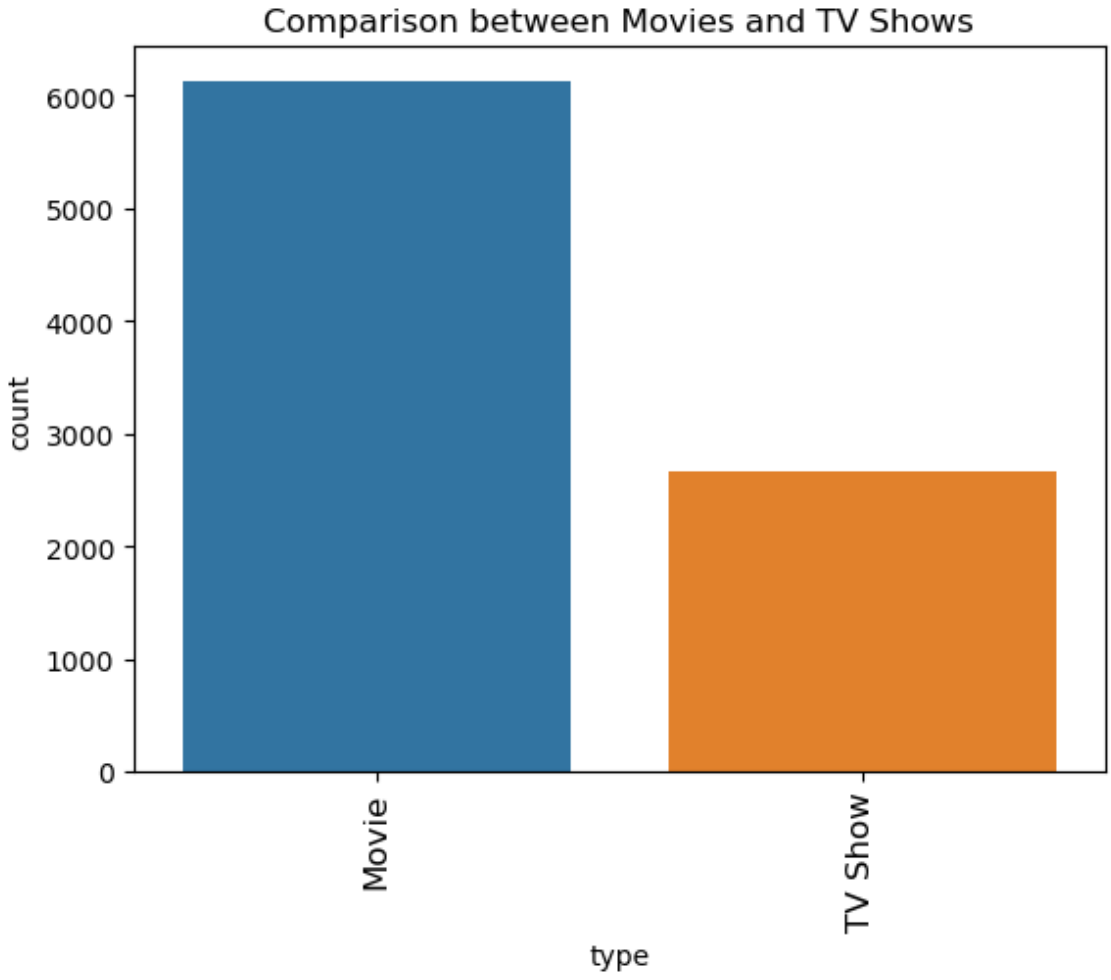
4.2 For categorical variable(s):

Comparison of tv shows vs. movies.

Categorical variables-Content Type using countplot

In [500]:

```
sns.countplot(data=df,x="type",order = df["type"].value_counts().index)
plt.xticks(rotation=90,fontsize=12)
plt.title("Comparison between Movies and TV Shows")
plt.show()
```

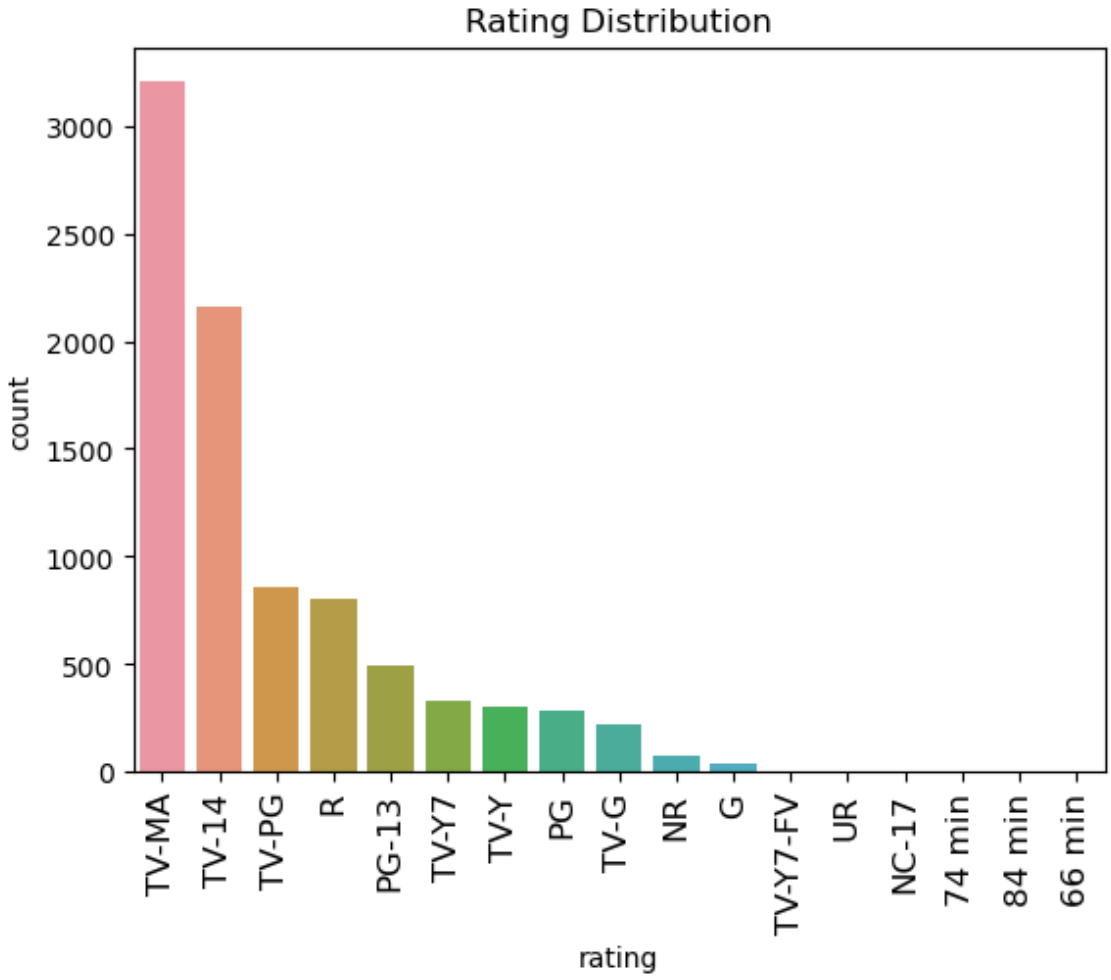


- More number of movies compared to TV Show are present in Netflix
- More number of TV Shows should be focussed on

Rating distribution

In [501]:

```
sns.countplot(data=df,x="rating",order = df["rating"].value_counts().index)
plt.xticks(rotation=90,fontsize=12)
plt.title("Rating Distribution")
plt.show()
```



- Most of the content on Amazon is for Adults(TV-MA) and Older Kids(TV-PG)
- Netflix should focus on putting more content for kids and teens with offers on subsrciption simlar to **Amazon's back to school/student campaign**

Genre/listed_in distribution

In [502]:

```
data=df_genre.loc[df_genre["listed_in"].value_counts()]
```

In [503]:

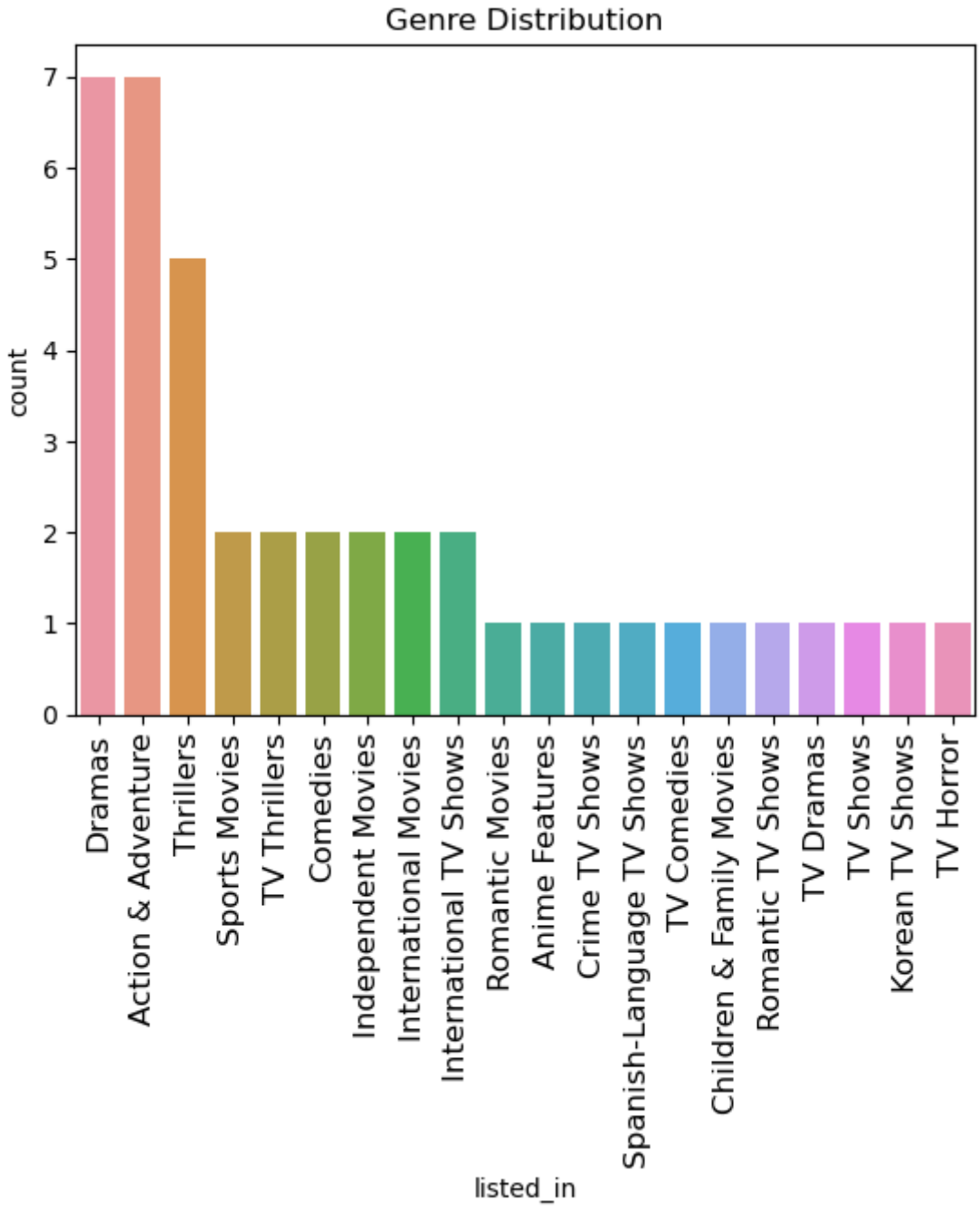
```
data.head(3)
```

Out[503]:

	show_id	type	title	director	cast	country	date_added	release_year	ratin
29799	s1216	Movie	Romantik Komedi	Ketche	Burcu Kara	Turkey	2021-03-12	2010	TV M
28243	s1141	Movie	Universal Soldier: The Return	Mic Rodgers	Kiana Tom	United States	2021-04-01	1999	I
20829	s829	Movie	Collateral Beauty	David Frankel	Kate Winslet	United States	2021-05-28	2016	PG 1

In [504]:

```
sns.countplot(data=data,x="listed_in",order = data["listed_in"].value_counts().index)
plt.xticks(rotation=90,fontsize=12)
plt.title("Genre Distribution")
plt.show()
```



- Dramas, Action&Adventure , Thrillers are most watched genres
- More focus should be given on Comdied and Romantic genre or , preferred genre across each country

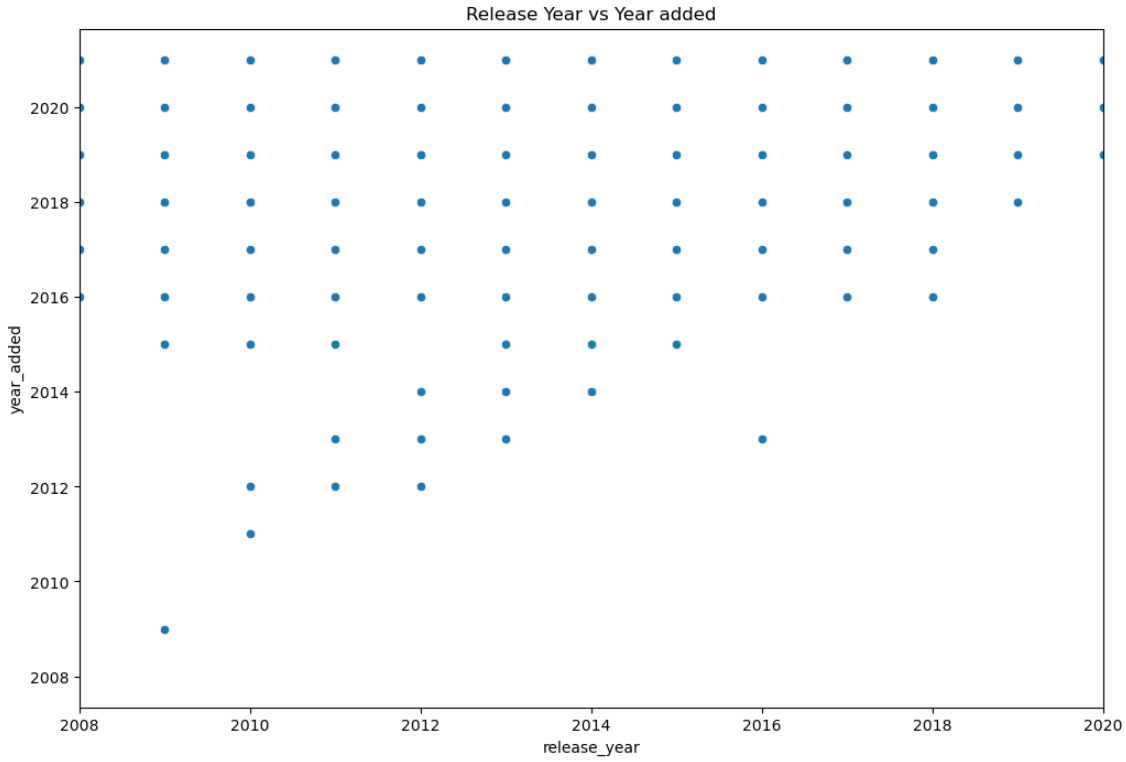
Bivariate

4.1 For continuous variable(s): Distplot, countplot, histogram for univariate analysis (10 Points)

Here we will be considering the N-N or continous-continous type of variables

In [505]:

```
plt.figure(figsize=(12,8))
sns.scatterplot(data=df,x="release_year",y="year_added")
plt.xlim(left=2008,right=2020)
plt.title("Release Year vs Year added")
plt.show()
```

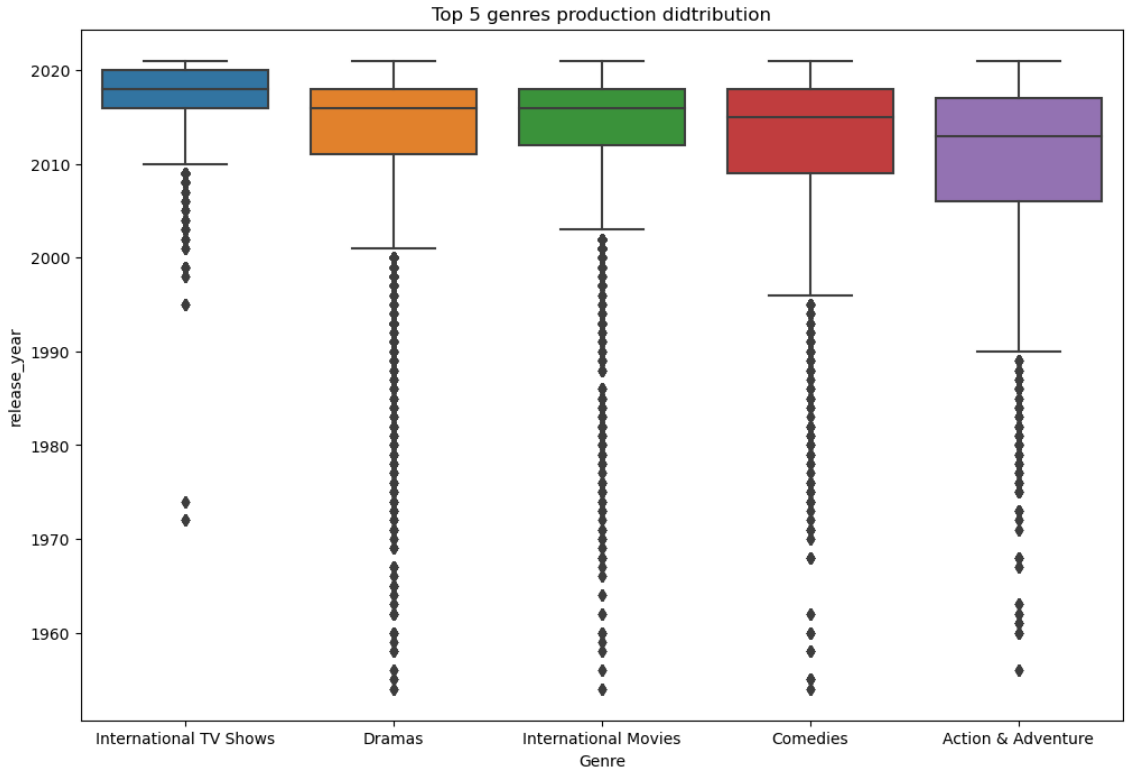


- From 2016 onwards only , content was started adding in Netflix, there on content has been added regularly as it was released

4.2 For categorical variable(s): Boxplot (10 Points)

In [506]:

```
top5_genre=df_genre.loc[df_genre['listed_in'].isin(df_genre["listed_in"].value_counts().
plt.figure(figsize=(12,8))
sns.boxplot(data=top5_genre,x='listed_in',y='release_year')
plt.title("Top 5 genres production didtribution")
plt.xlabel("Genre")
plt.show()
```



- Action and Adventure movies took longer to be produced compared to Interenational TV Shows

What type of content is available in different countries? Understanding what content is available in different countries

Content by country

Here we will be considering bivariate analysis for two catergorical variable content and country

In [507]:

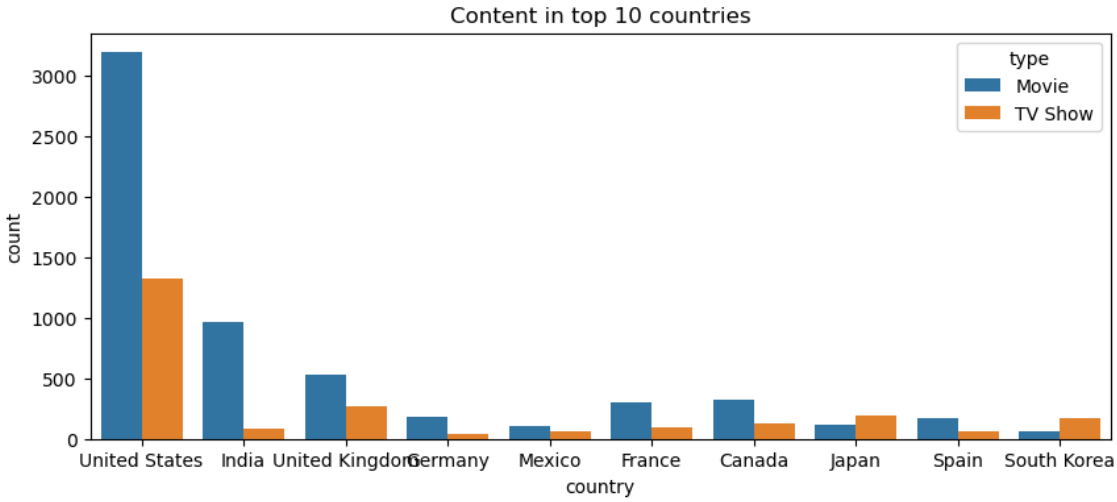
```
top10_countries=df_country["country"].value_counts().index[:10]
top10_countries_data = df_country[(df_country["country"].isin(top10_countries))]
top10_countries_data.head(3)
```

Out[507]:

	show_id	type	title	director	cast	country	date_added	release_year	r
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	Cast unavailable	United States	2021-09-25	2020	
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	United States	2021-09-24	2021	
3	s4	TV Show	Jailbirds New Orleans	Anonymous	Cast unavailable	United States	2021-09-24	2021	

In [508]:

```
plt.figure(figsize=(10,4))
sns.countplot(data=top10_countries_data,x='country',hue='type')
plt.title("Content in top 10 countries")
plt.show()
```



- United States and India are the leading countries for Movies
- However TV shows are very less comapred to movies in these two leading countries
- More content should be released in United Kingdom and the other remaining countries

4.3 For correlation: Heatmaps, Pairplots

In [509]:

```
sns.heatmap(top10_countries_data.corr(),annot=True,cmap='Purples')
```

```
C:\Users\krama\AppData\Local\Temp\ipykernel_7196\982102484.py:1: FutureWa
rning: The default value of numeric_only in DataFrame.corr is deprecated.
In a future version, it will default to False. Select only valid columns
or specify the value of numeric_only to silence this warning.
  sns.heatmap(top10_countries_data.corr(),annot=True,cmap='Purples')
```

Out[509]:

```
<Axes: >
```

Release year and year added are lightly correlated

What is the best time to launch a TV show?

In [510]:

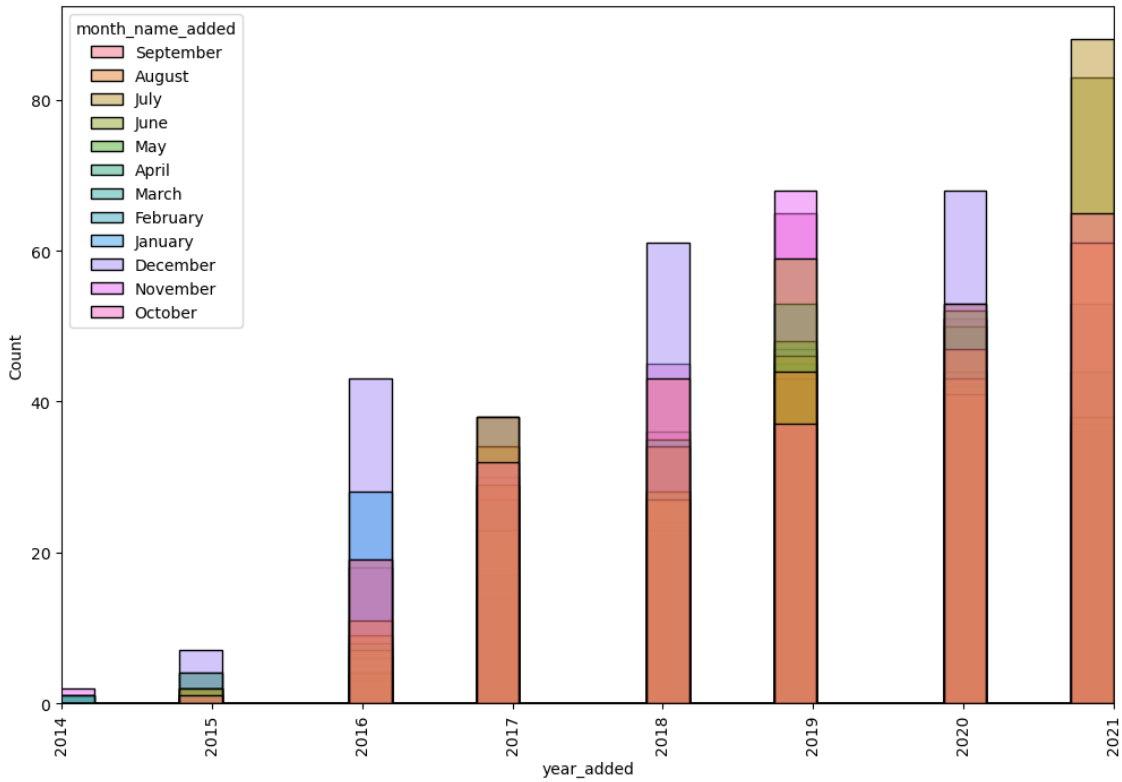
```
tv_shows.head(3)
```

Out[510]:

	show_id	type	title	director	cast	country	date_added	release_year	ra
1	s2	TV Show	Blood & Water	Anonymous	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	2021-09-24	2021	
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	United States	2021-09-24	2021	
3	s4	TV Show	Jailbirds New Orleans	Anonymous	Cast unavailable	United States	2021-09-24	2021	

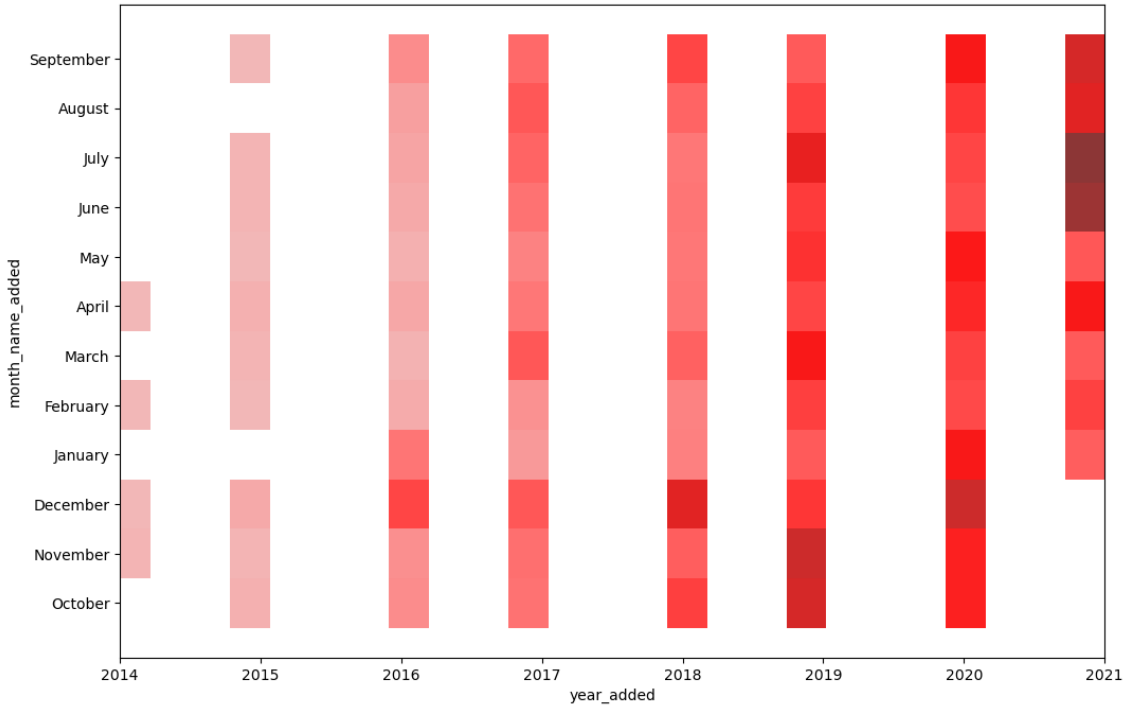
In [511]:

```
plt.figure(figsize=(12,8))
sns.histplot(data=tv_shows , x='year_added',hue="month_name_added")
plt.xlim(left=2014,right=2021)
plt.xticks(np.arange(2014, 2022, 1),rotation=90)
plt.show()
```



In [512]:

```
plt.figure(figsize=(12,8))
sns.histplot(data=tv_shows , x='year_added',y="month_name_added",color="red")
plt.xlim(left=2014,right=2021)
plt.show()
```



If the latest year 2019 is considered, January and December were the months when comparatively much less content was released.Therefore, these months may be a good choice for the success of a new release!

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

In [513]:

```
df_cast.head(2)
```

Out[513]:

	show_id	type	title	director	cast	country	date_added	release_year	rati
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	Cast unavailable	United States	2021-09-25	2020	P
1	s2	TV Show	Blood & Water	Anonymous	Ama Qamata	South Africa	2021-09-24	2021	T M

In [514]:

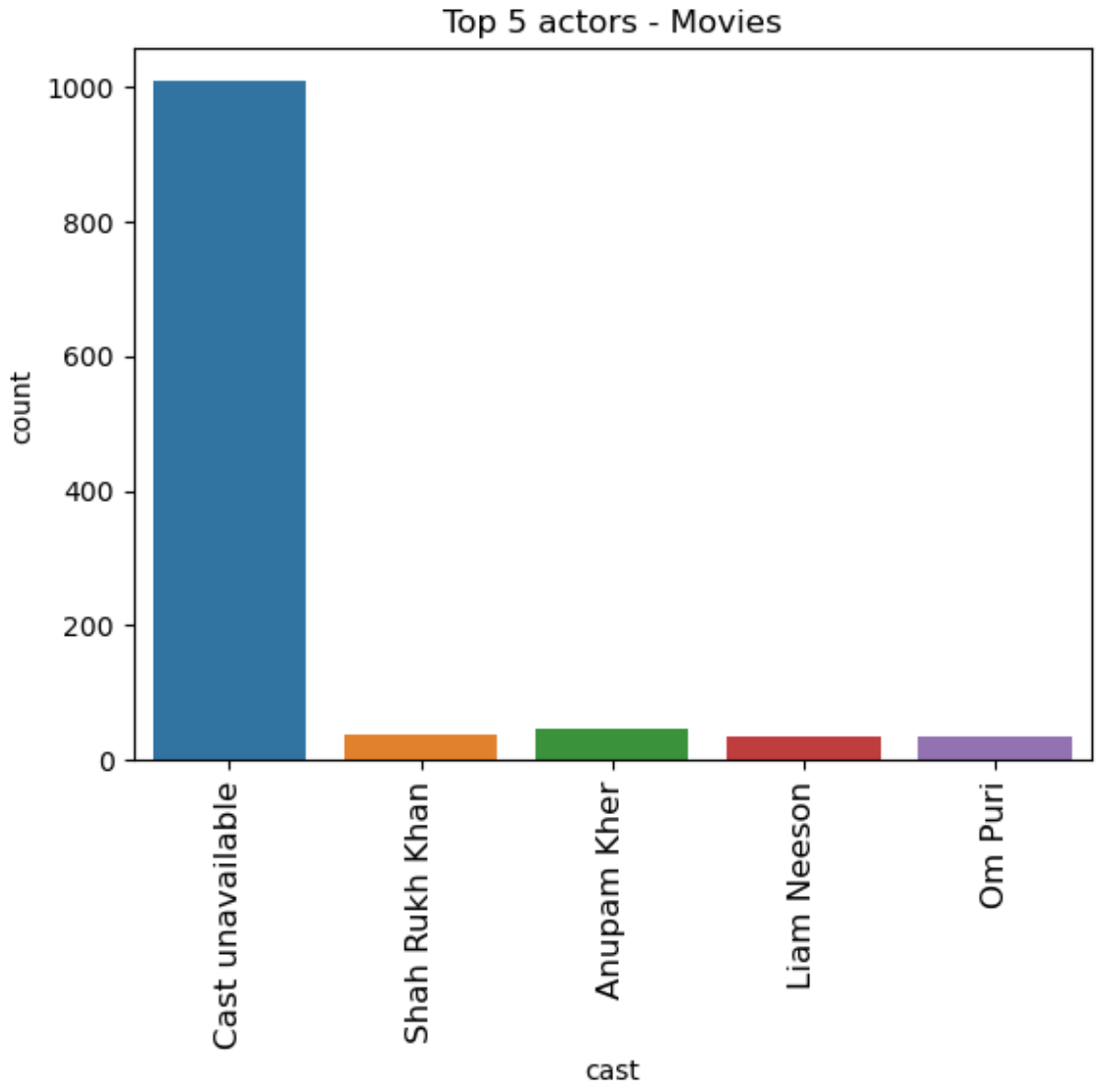
```
cast_movie=df_cast.loc[df_cast["type"]=="Movie"]
cast_tv_show=df_cast.loc[df_cast["type"]=="TV Show"]
top20actorsmovies=cast_movie["cast"].value_counts().index[:5]
top20actorsshow=cast_tv_show["cast"].value_counts().index[:5]
```

In [515]:

```
top20actorsmoviesdata=df_cast.loc[df_cast["cast"].isin(top20actorsmovies)]
top20actorsshowsdata=df_cast.loc[df_cast["cast"].isin(top20actorsshow)]
```

In [516]:

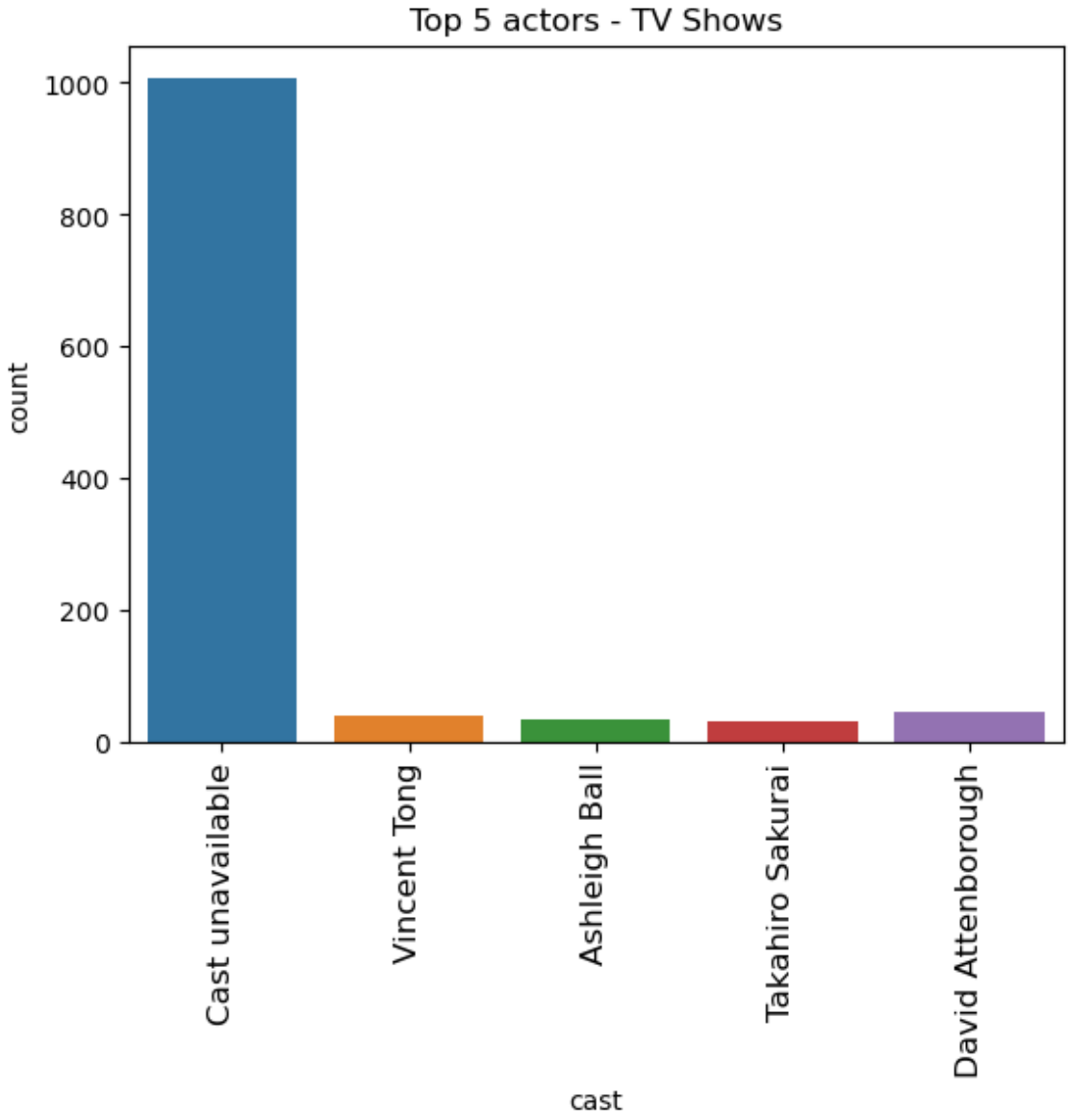
```
sns.countplot(data=top20actorsmoviesdata,x='cast')
plt.xticks(rotation=90,fontsize=12) ## to avoid overlapping labels
plt.title("Top 5 actors - Movies")
plt.show()
```



Bollywood actor Shahrukh Khan is the most popular

In [517]:

```
sns.countplot(data=top20actorsshowsdata,x='cast')
plt.xticks(rotation=90,fontsize=12) ## to avoid overlapping labels
plt.title("Top 5 actors - TV Shows")
plt.show()
```



Top 5 directors TV Shows/Movies

In [518]:

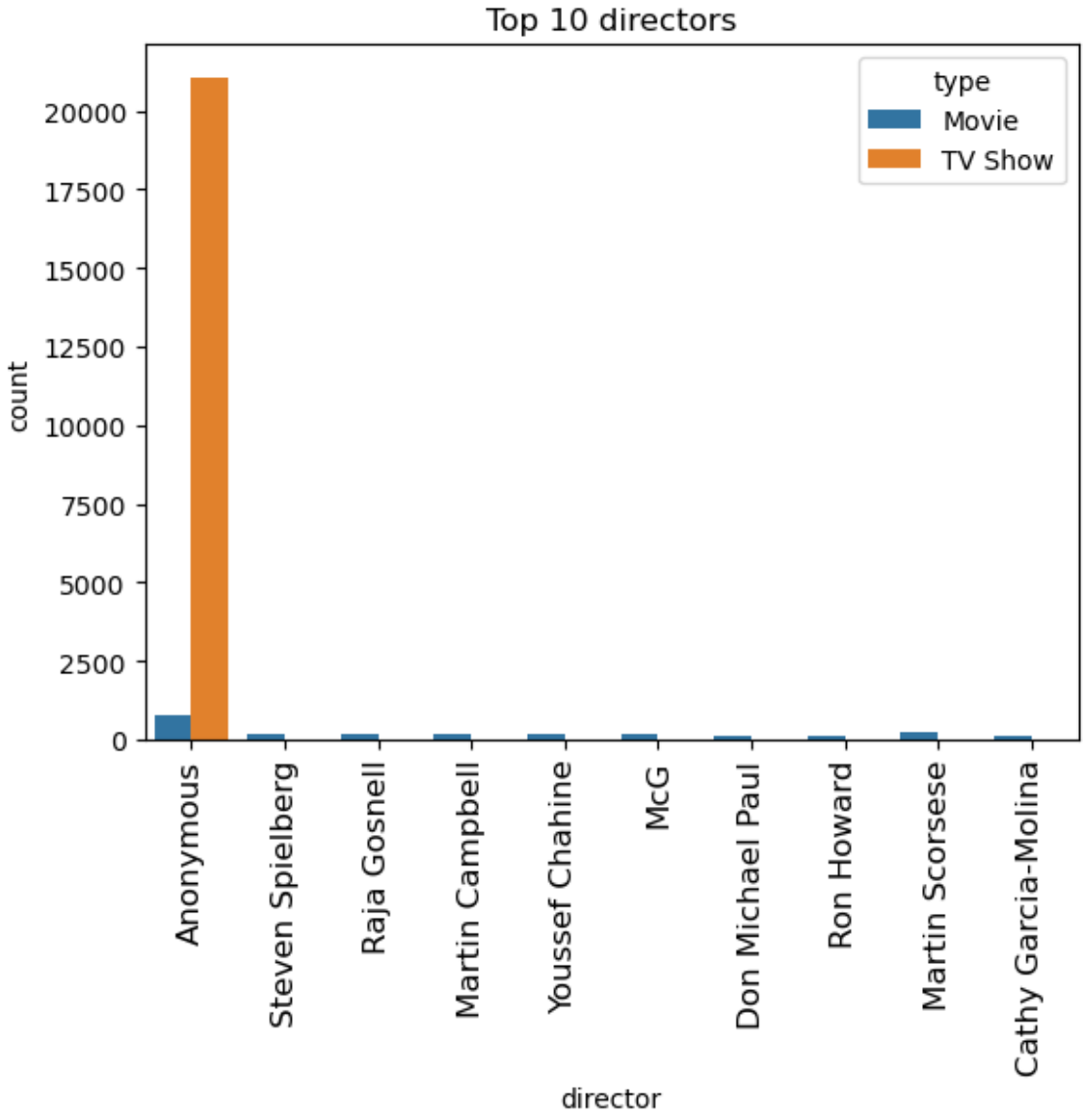
```
top10_dir=df_cast["director"].value_counts().index[:10]
```

In [519]:

```
top10_dir_data=df_cast[df_cast["director"].isin(top10_dir)]
```

In [520]:

```
sns.countplot(data=top10_dir_data,x='director',hue="type")
plt.xticks(rotation=90,fontsize=12) ## to avoid overlapping labels
plt.title("Top 10 directors")
plt.show()
```

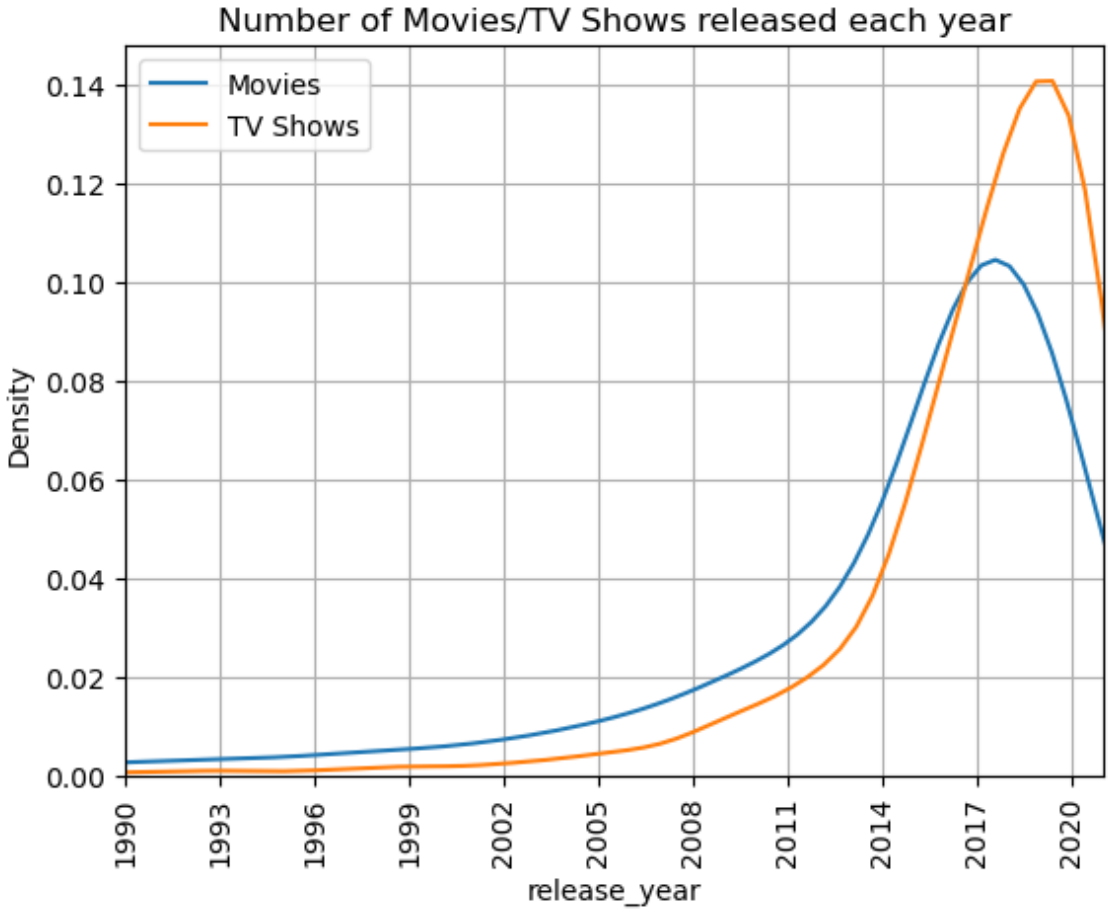


Steven Speilberg is the most popular/productive director across all content type

Does Netflix has more focus on TV Shows than movies in recent years

In [521]:

```
sns.kdeplot(movies["release_year"])
sns.kdeplot(tv_shows["release_year"])
plt.xlim(left=1990,right=2021)
plt.xticks(np.arange(1990, 2021, 3),rotation=90)
plt.title("Number of Movies/TV Shows released each year")
plt.legend(['Movies','TV Shows'],loc='upper left')
plt.grid()
plt.show()
```



We can clearly see that from year 2017 onwards there is an increase in number of tv show compared to movies