In [156]:

```
#import important libraries
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import seaborn as sns
df = pd.read_csv("netflix.csv")
df
```

Out[156]:

	show_id	type	title	director	cast	country	date_added	release_year
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021
...	...	...	...	...	...	...	...	...
8802	s8803	Movie	Zodiac	David Fincher	Mark Ruffalo, Jake Gyllenhaal, Robert Downey J...	United States	November 20, 2019	2007
8803	s8804	TV Show	Zombie Dumb	NaN	NaN	NaN	July 1, 2019	2018

	show_id	type	title	director	cast	country	date_added	release_year
8804	s8805	Movie	Zombieland	Ruben Fleischer	Jesse Eisenberg, Woody Harrelson, Emma Stone, ...	United States	November 1, 2019	2009
8805	s8806	Movie	Zoom	Peter Hewitt	Tim Allen, Courteney Cox, Chevy Chase, Kate Ma...	United States	January 11, 2020	2006
8806	s8807	Movie	Zubaan	Mozez Singh	Vicky Kaushal, Sarah-Jane Dias, Raaghav Chanan...	India	March 2, 2019	2015

8807 rows × 12 columns

In [3]:

```
1 df.head()
```

Out[3]:

	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-13
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

In [4]:

```
1 df.tail()
```

Out[4]:

	show_id	type	title	director	cast	country	date_added	release_year	rat
8802	s8803	Movie	Zodiac	David Fincher	Mark Ruffalo, Jake Gyllenhaal, Robert Downey J...	United States	November 20, 2019	2007	
8803	s8804	TV Show	Zombie Dumb	NaN	NaN	NaN	July 1, 2019	2018	TV·
8804	s8805	Movie	Zombieland	Ruben Fleischer	Jesse Eisenberg, Woody Harrelson, Emma Stone, ...	United States	November 1, 2019	2009	
8805	s8806	Movie	Zoom	Peter Hewitt	Tim Allen, Courteney Cox, Chevy Chase, Kate Ma...	United States	January 11, 2020	2006	
8806	s8807	Movie	Zubaan	Mozez Singh	Vicky Kaushal, Sarah-Jane Dias, Raaghav Chanan...	India	March 2, 2019	2015	TV

In [5]:

```
1 df.info()
2
```

```
<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
```

In [6]:

```
1 df.shape
```

Out[6]:

```
(8807, 12)
```

In [7]:

```
1 df.isna().sum()
```

Out[7]:

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

In [8]:

```
df.describe()
```

Out[8]:

	release_year
count	8807.000000
mean	2014.180198
std	8.819312
min	1925.000000
25%	2013.000000
50%	2017.000000
75%	2019.000000
max	2021.000000

In [9]:

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

Out[9]:

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

In [10]:

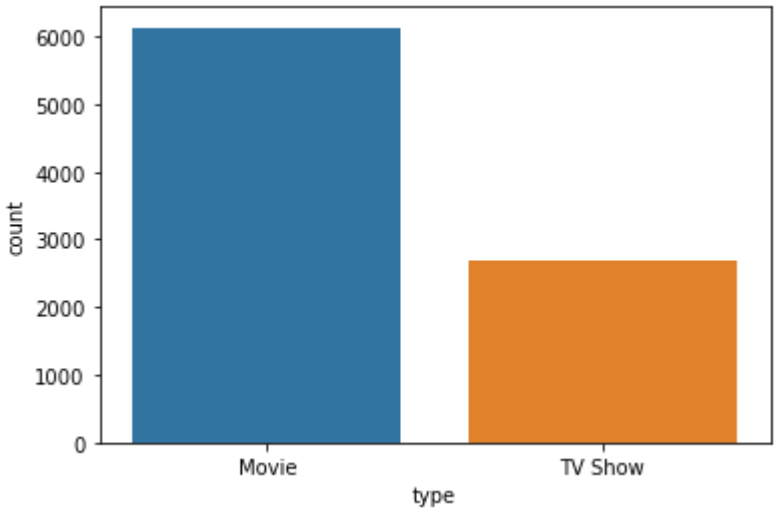
```
df['type'].value_counts(normalize=True)*100
```

Out[10]:

```
Movie      69.615079
TV Show    30.384921
Name: type, dtype: float64
```

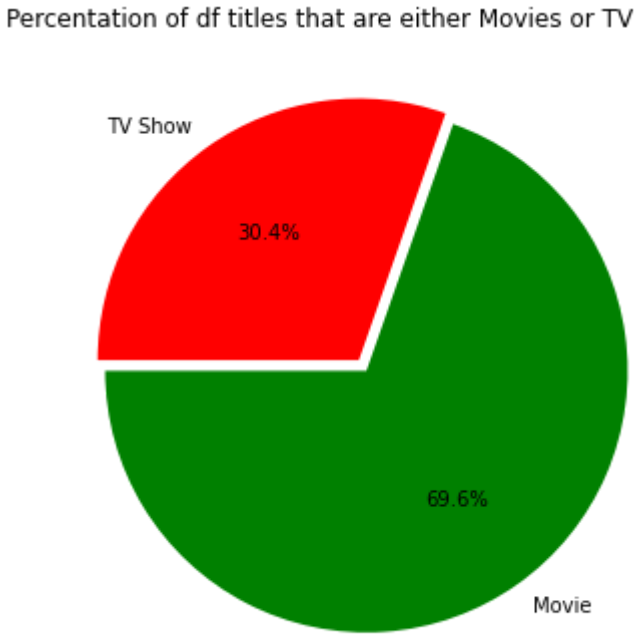
In [11]:

```
sns.countplot(data= df, x= 'type')
plt.show()
```



In [13]:

```
titles that are either Movies or TV Shows")
ts(),explode=(0.025,0.025), labels=df.type.value_counts().index, colors=['green','red'],auto
```



In [14]:

```
1 df['rating'].value_counts(normalize=True)*100
```

Out[14]:

```
TV-MA       36.430762
TV-14       24.537090
TV-PG        9.803476
R            9.076451
PG-13        5.566284
TV-Y7        3.794161
TV-Y         3.487447
PG           3.260252
TV-G         2.499148
NR           0.908781
G            0.465750
TV-Y7-FV     0.068159
NC-17        0.034079
UR           0.034079
74 min       0.011360
84 min       0.011360
66 min       0.011360
Name: rating, dtype: float64
```

In [15]:

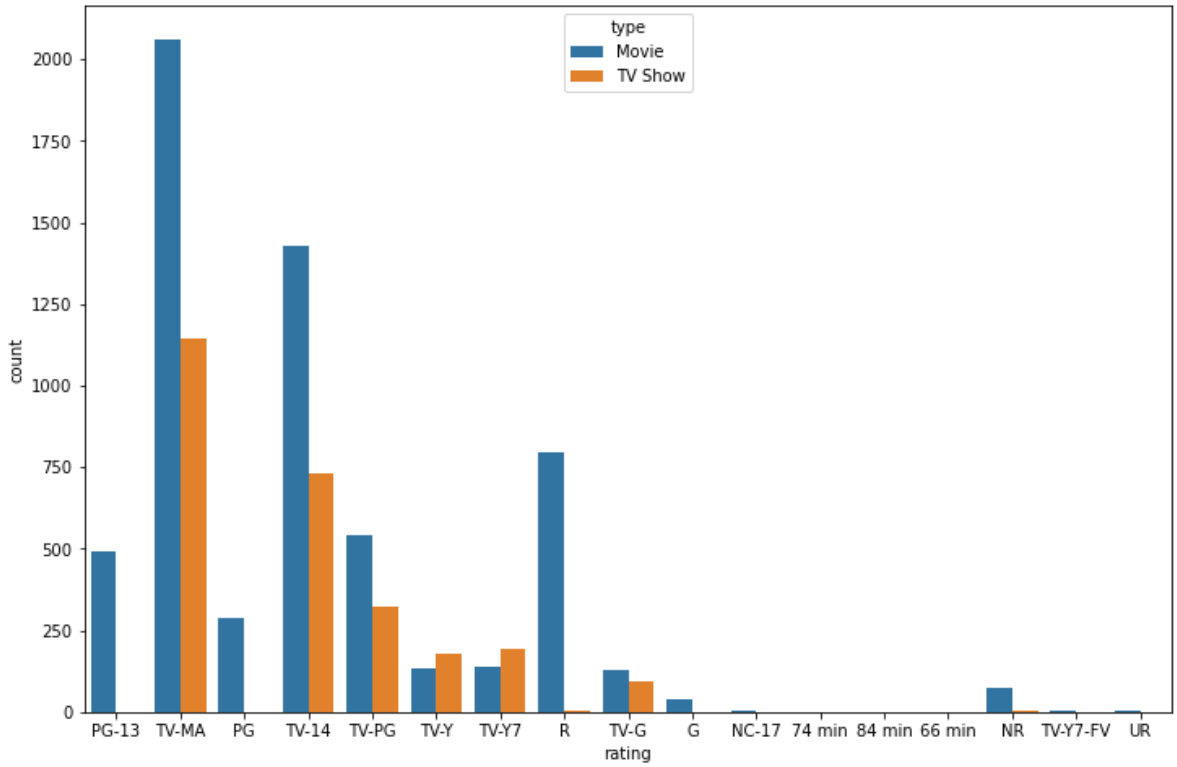
```
df[df['rating']=='R']['type'].value_counts()
```

Out[15]:

```
Movie      797
TV Show      2
Name: type, dtype: int64
```

In [19]:

```
plt.figure(figsize=(12,8))
sns.countplot(data= df, x= 'rating' , hue= 'type')
plt.show()
```



In [20]:

```
df['release_year'].value_counts(normalize=True)*100
```

Out[20]:

```
2018    13.023731
2017    11.717952
2019    11.695242
2020    10.820938
2016    10.241853
          ...
1959     0.011355
1925     0.011355
1961     0.011355
1947     0.011355
1966     0.011355
Name: release_year, Length: 74, dtype: float64
```

In [21]:

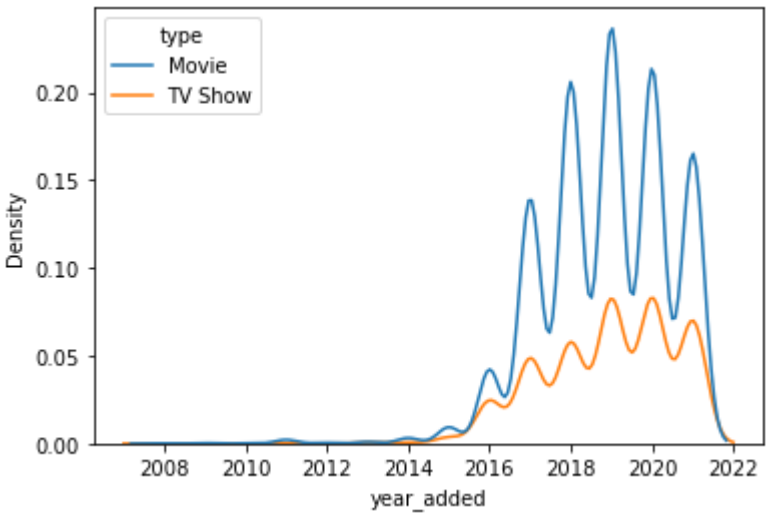
```
df['year_added'] = pd.to_datetime(df['date_added']).dt.year
# df['year_added'] = df['year_added'][df['year_added'].isna()==False].astype(int)
df.head()
```

Out[21]:

	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-13
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

In [22]:

```
1 sns.kdeplot(data=df, x='year_added' , hue='type')
2 plt.show()
3
```



In [23]:

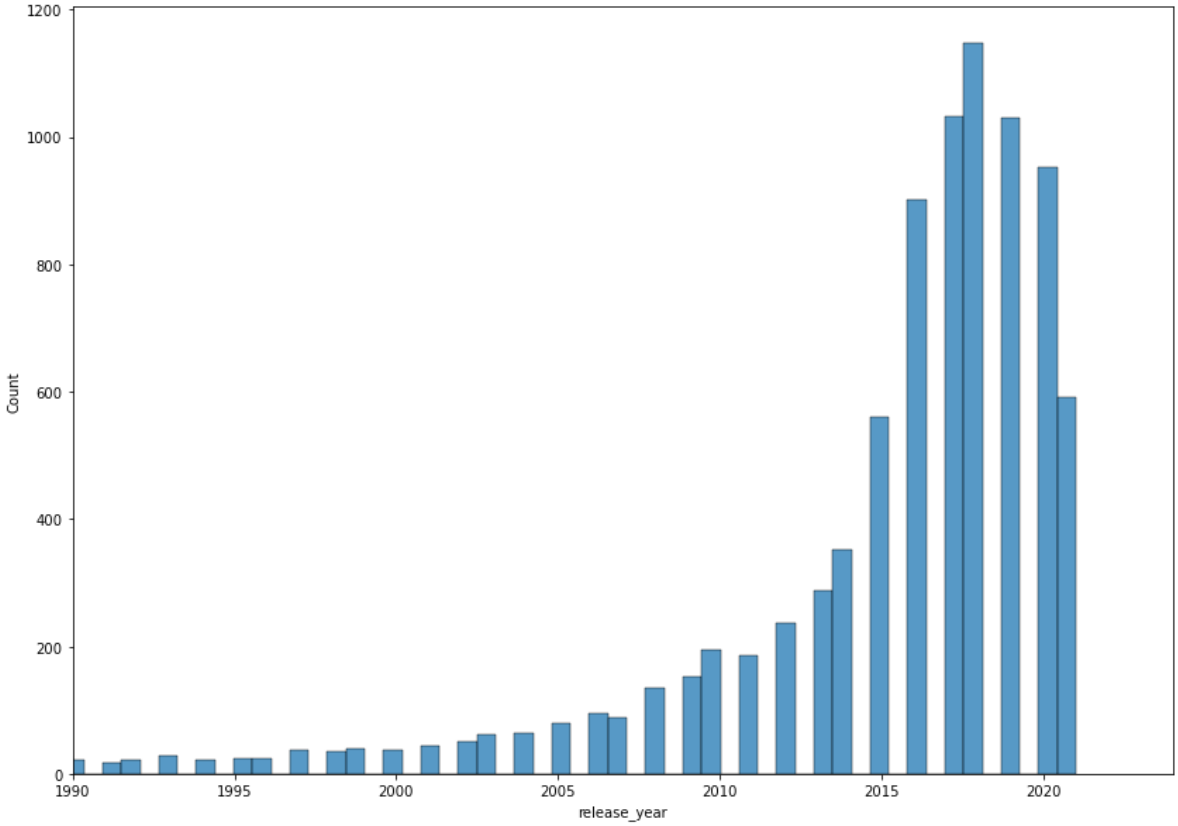
```
1 df[(df['release_year'] >=1990) & (df['release_year'] <= 2022)]['release_year'].value_cc
```

Out[23]:

```
2018    1147
2017    1032
2019    1030
2020     953
2016     902
2021     592
2015     560
2014     352
2013     288
2012     237
2010     194
2011     185
2009     152
2008     136
2006      96
2007      88
2005      80
2004      64
2003      61
2002      51
2001      45
1999      39
1997      38
2000      37
1998      36
1993      28
1995      25
1996      24
1992      23
1990      22
1994      22
1991      17
Name: release_year, dtype: int64
```

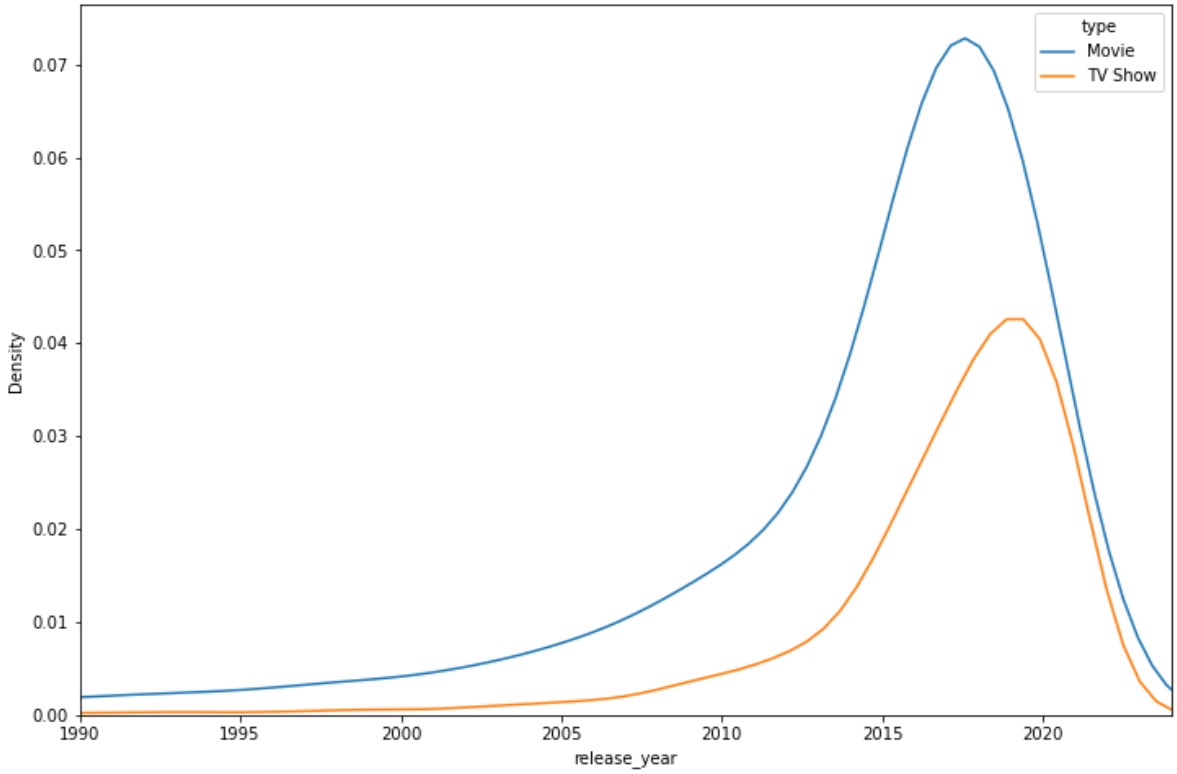
In [24]:

```
plt.figure(figsize=(14,10))
sns.histplot(data= df, x= 'release_year')
plt.xlim(1990,2024)
plt.show()
```



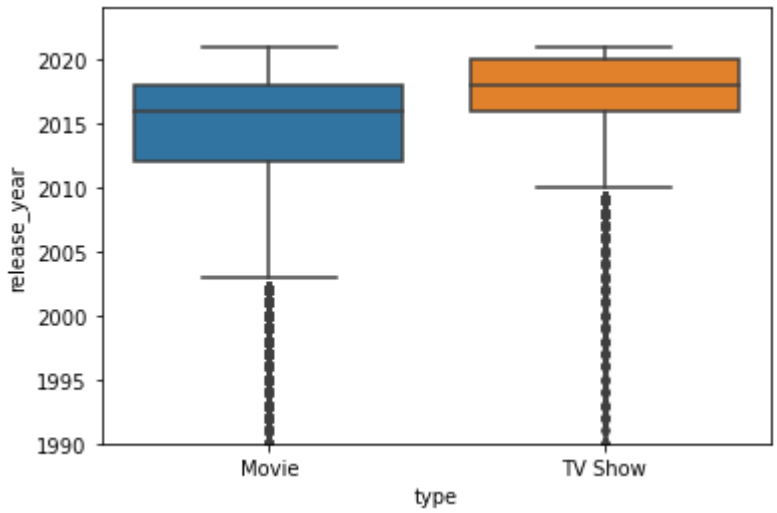
In [25]:

```
plt.figure(figsize=(12,8))
sns.kdeplot(data= df, x= 'release_year', hue = 'type')
plt.xlim(1990,2024)
plt.show()

```



In [26]:

```
sns.boxplot(data= df, y= 'release_year' , x= 'type')
plt.ylim(1990,2024)
plt.show()
```



unnesting the cast column

In [27]:

```
# Seperating the cast members into columns for each title
constraint=df['cast'].apply(lambda x: str(x).split(', ')).tolist()
cast_expand=pd.DataFrame(constraint,index=df['title'])

```

In [28]:

```
#stacking all cast columns into rows for each title
cast_expand = cast_expand.stack()
cast_expand = pd.DataFrame(cast_expand, columns=['cast'])

```

In [29]:

```
1 cast_expand
```

Out[29]:

		cast
title		
Dick Johnson Is Dead	**0**	nan
Blood & Water	**0**	Ama Qamata
	1	Khosi Ngema
	2	Gail Mabalane
	3	Thabang Molaba
...	**...**	...
Zubaan	**3**	Manish Chaudhary
	4	Meghna Malik
	5	Malkeet Rauni
	6	Anita Shabdish
	7	Chittaranjan Tripathy

64951 rows × 1 columns

In [30]:

```
1 # top 10 cast members with most content
2 cast_expand.value_counts().head(11)
```

Out[30]:

```
cast
nan                 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
Amitabh Bachchan     28
dtype: int64
```

In [31]:

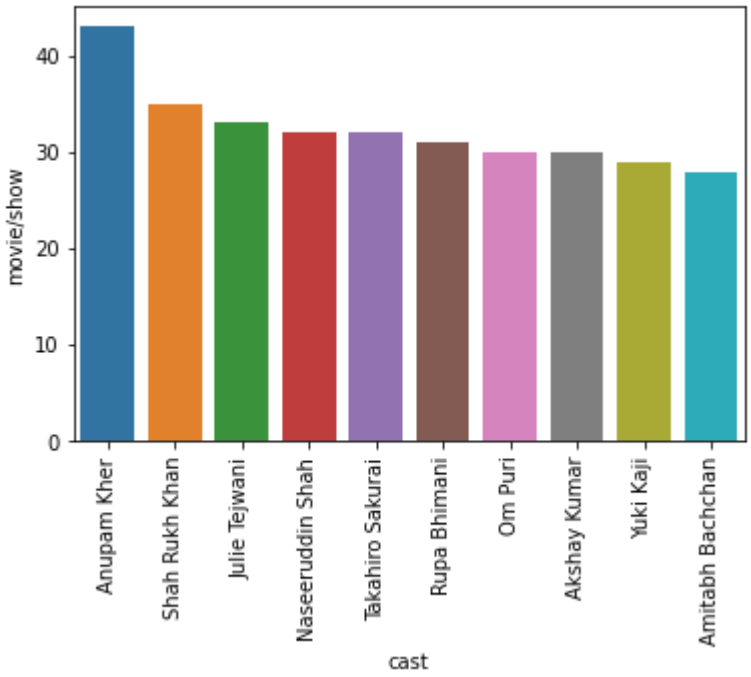
```
top10_cast = cast_expand.value_counts()[1:11]
top10_cast = pd.DataFrame(top10_cast, columns = ['movie/show']).reset_index()
top10_cast
```

Out[31]:

	cast	movie/show
0	Anupam Kher	43
1	Shah Rukh Khan	35
2	Julie Tejwani	33
3	Naseeruddin Shah	32
4	Takahiro Sakurai	32
5	Rupa Bhimani	31
6	Om Puri	30
7	Akshay Kumar	30
8	Yuki Kaji	29
9	Amitabh Bachchan	28

In [32]:

```
sns.barplot(data= top10_cast, x= 'cast' , y='movie/show')
plt.xticks(rotation = 90)
plt.show()
```



In [33]:

```
temp = df[['title','type']].set_index('title')
temp
```

Out[33]:

title	type
Dick Johnson Is Dead	Movie
Blood & Water	TV Show
Ganglands	TV Show
Jailbirds New Orleans	TV Show
Kota Factory	TV Show
...	...
Zodiac	Movie
Zombie Dumb	TV Show
Zombieland	Movie
Zoom	Movie
Zubaan	Movie

8807 rows × 1 columns

In [56]:

```
categorised_cast = pd.merge(cast_expand, temp, left_index=True, right_index= True)
categorised_cast
```

Out[56]:

		cast	type
title			
Dick Johnson Is Dead	**0**	nan	Movie
Blood & Water	**0**	Ama Qamata	TV Show
	1	Khosi Ngema	TV Show
	2	Gail Mabalane	TV Show
	3	Thabang Molaba	TV Show
...	**...**	...	...
Zubaan	**3**	Manish Chaudhary	Movie
	4	Meghna Malik	Movie
	5	Malkeet Rauni	Movie
	6	Anita Shabdish	Movie
	7	Chittaranjan Tripathy	Movie

64951 rows × 2 columns

In [61]:

```
cast = pd.DataFrame(categorised_cast[categorised_cast['type'] == 'Movie'].value_counts().hea
cast
```

Out[61]:

		No. of Movies
cast	**type**	
nan	**Movie**	475
Anupam Kher	**Movie**	42
Shah Rukh Khan	**Movie**	35
Naseeruddin Shah	**Movie**	32
Om Puri	**Movie**	30
Akshay Kumar	**Movie**	30
Amitabh Bachchan	**Movie**	28
Julie Tejwani	**Movie**	28
Paresh Rawal	**Movie**	28
Rupa Bhimani	**Movie**	27

In [62]:

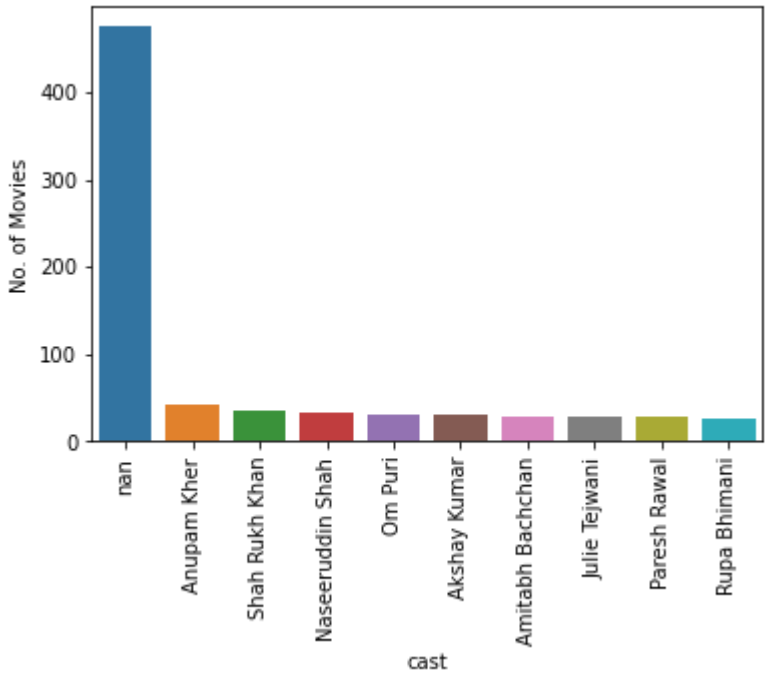
```
top_movie_cast.reset_index(1,drop=True,inplace=True)
top_movie_cast.reset_index(inplace = True)
top_movie_cast
```

Out[62]:

	cast	No. of Movies
0	nan	475
1	Anupam Kher	42
2	Shah Rukh Khan	35
3	Naseeruddin Shah	32
4	Om Puri	30
5	Akshay Kumar	30
6	Amitabh Bachchan	28
7	Julie Tejwani	28
8	Paresh Rawal	28
9	Rupa Bhimani	27

In [63]:

```
sns.barplot(data=top_movie_cast, x= 'cast', y='No. of Movies')
plt.xticks(rotation = 90)
plt.show()
```



In [69]:

```
pd.DataFrame(categorised_cast[categorised_cast['type'] == 'TV Show'].value_counts().head(1
```

Out[69]:

		No. of TV Shows
cast	**type**	
nan	**TV Show**	350
Takahiro Sakurai	**TV Show**	25
Yuki Kaji	**TV Show**	19
Daisuke Ono	**TV Show**	17
Ai Kayano	**TV Show**	17
Junichi Suwabe	**TV Show**	17
Yuichi Nakamura	**TV Show**	16
Yoshimasa Hosoya	**TV Show**	15
Jun Fukuyama	**TV Show**	15
David Attenborough	**TV Show**	14

In [70]:

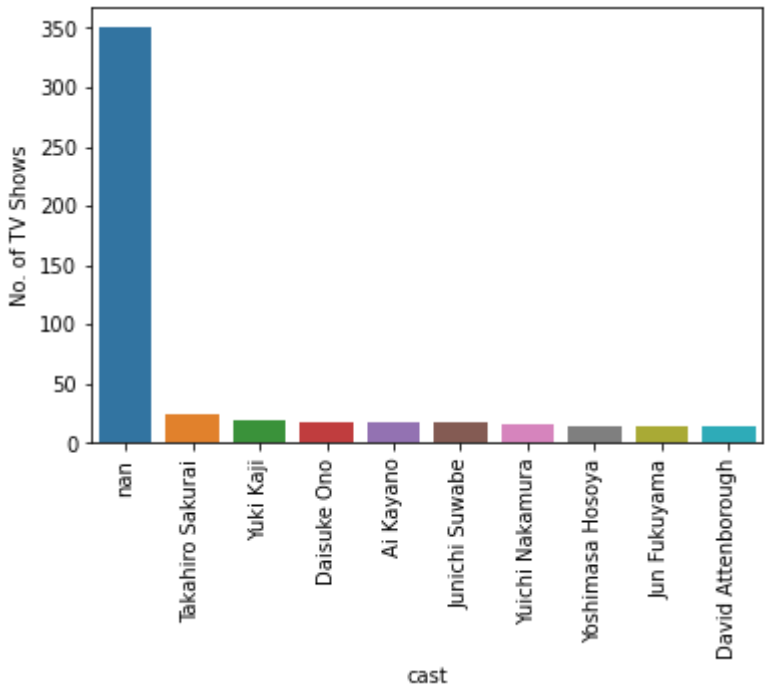
```
top_show_cast.reset_index(1,drop=True,inplace=True)
top_show_cast.reset_index(inplace = True)
top_show_cast
```

Out[70]:

	cast	No. of TV Shows
0	nan	350
1	Takahiro Sakurai	25
2	Yuki Kaji	19
3	Daisuke Ono	17
4	Ai Kayano	17
5	Junichi Suwabe	17
6	Yuichi Nakamura	16
7	Yoshimasa Hosoya	15
8	Jun Fukuyama	15
9	David Attenborough	14

In [71]:

```
sns.barplot(data=top_show_cast, x= 'cast', y='No. of TV Shows')
plt.xticks(rotation = 90)
plt.show()
```



In [72]:

```
temp2 = df[['title','year_added']]
temp2
```

Out[72]:

	title	year_added
0	Dick Johnson Is Dead	2021.0
1	Blood & Water	2021.0
2	Ganglands	2021.0
3	Jailbirds New Orleans	2021.0
4	Kota Factory	2021.0
...	...	...
8802	Zodiac	2019.0
8803	Zombie Dumb	2019.0
8804	Zombieland	2019.0
8805	Zoom	2020.0
8806	Zubaan	2019.0

8807 rows × 2 columns

In [73]:

```
cast = cast_expand.reset_index(1,drop=True)

```

In [74]:

```
df2= pd.merge(cast , temp2 , left_index=True , right_on='title')
df2
```

Out[74]:

	cast	title	year_added
0	nan	Dick Johnson Is Dead	2021.0
1	Ama Qamata	Blood & Water	2021.0
1	Khosi Ngema	Blood & Water	2021.0
1	Gail Mabalane	Blood & Water	2021.0
1	Thabang Molaba	Blood & Water	2021.0
...	...	...	...
8806	Manish Chaudhary	Zubaan	2019.0
8806	Meghna Malik	Zubaan	2019.0
8806	Malkeet Rauni	Zubaan	2019.0
8806	Anita Shabdish	Zubaan	2019.0
8806	Chittaranjan Tripathy	Zubaan	2019.0

64951 rows × 3 columns

In [75]:

```
df2 = df2.sort_index()[['title', 'year_added' , 'cast']]
df2
```

Out[75]:

	title	year_added	cast
0	Dick Johnson Is Dead	2021.0	nan
1	Blood & Water	2021.0	Ama Qamata
1	Blood & Water	2021.0	Khosi Ngema
1	Blood & Water	2021.0	Gail Mabalane
1	Blood & Water	2021.0	Thabang Molaba
...	...	...	...
8806	Zubaan	2019.0	Manish Chaudhary
8806	Zubaan	2019.0	Meghna Malik
8806	Zubaan	2019.0	Malkeet Rauni
8806	Zubaan	2019.0	Anita Shabdish
8806	Zubaan	2019.0	Chittaranjan Tripathy

64951 rows × 3 columns

In [76]:

```
grp1 = df2.groupby(by='cast')['year_added'].value_counts().sort_values(ascending = Fals
```

In [77]:

```
grp1 = pd.DataFrame(grp1)
grp1.head()
```

Out[77]:

		year_added
cast	**year_added**	
nan	**2019.0**	165
	2020.0	155
	2021.0	150
	2018.0	150
	2017.0	140

In [78]:

```
grp1 = grp1.drop(index= 'nan')
```

```
C:\Users\Shelendra\anaconda3\lib\site-packages\pandas\core\generic.py:4150:
PerformanceWarning: dropping on a non-lexsorted multi-index without a level
parameter may impact performance.
  obj = obj._drop_axis(labels, axis, level=level, errors=errors)
```

In [79]:

```
grp1.columns = ['No. of Movies/Shows']
grp1.reset_index(1)
```

Out[79]:

cast	year_added	No. of Movies/Shows
Julie Tejwani	2021.0	22
Rupa Bhimani	2021.0	22
Rajesh Kava	2021.0	21
Anupam Kher	2018.0	19
Jigna Bhardwaj	2021.0	19
...	...	...
Ibrahim Suleiman	2021.0	1
Ibrahima Gueye	2020.0	1
Ibrahima Mbaye	2019.0	1
Ibrahima Traore	2019.0	1
Ṣọpẹ́ Dìrísù	2020.0	1

50860 rows × 2 columns

In [80]:

```
per_year = grp1.loc[top10_cast['cast']]
per_year
```

Out[80]:

		No. of Movies/Shows
cast	**year_added**	
Anupam Kher	**2018.0**	19
	2020.0	10
	2021.0	5
	2017.0	5
	2019.0	4
Shah Rukh Khan	**2017.0**	14
	2018.0	11
	2020.0	5
	2019.0	3
	2021.0	2
Julie Tejwani	**2021.0**	22
	2019.0	9
	2020.0	2
Naseeruddin Shah	**2019.0**	10
	2018.0	9
	2020.0	6
	2017.0	4
	2021.0	3
Takahiro Sakurai	**2019.0**	11
	2020.0	7
	2021.0	4
	2016.0	4
	2017.0	3
	2018.0	3
Rupa Bhimani	**2021.0**	22
	2019.0	9
Om Puri	**2018.0**	14
	2019.0	7
	2020.0	6
	2017.0	3
Akshay Kumar	**2018.0**	11
	2020.0	8
	2019.0	5

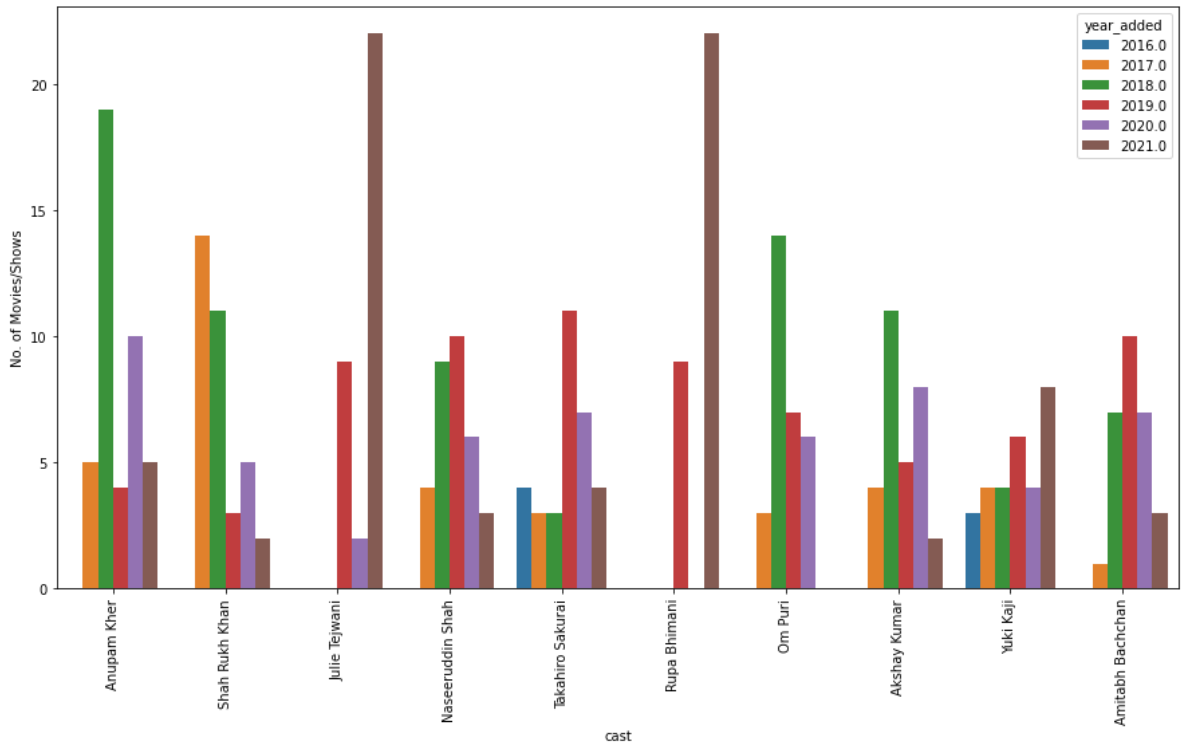
		No. of Movies/Shows
cast	**year_added**	
	2017.0	4
	2021.0	2
Yuki Kaji	**2021.0**	8
	2019.0	6
	2017.0	4
	2020.0	4
	2018.0	4
	2016.0	3
Amitabh Bachchan	**2019.0**	10
	2020.0	7
	2018.0	7
	2021.0	3
	2017.0	1

In [81]:

```
per_year.reset_index(inplace=True)

```

In [82]:

```
plt.figure(figsize=(15,8))
sns.barplot(data=per_year, x ='cast' , y = 'No. of Movies/Shows', hue= 'year_added')
plt.xticks(rotation =90)
plt.show()
```



```
# unnesting the director column
```

In [83]:

```
df.head()
```

Out[83]:

	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-13
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

In [84]:

```
# Seperating the directors into columns for each title
constraint2 = df['director'].apply(lambda x: str(x).split(", ")).to_list()
director_expand = pd.DataFrame(constraint2 , index=df['title'])

```

In [85]:

```
#stacking all director columns into rows for each title
director_expand = director_expand.stack()
director_expand = pd.DataFrame(director_expand, columns=['director'])
```

In [86]:

```
director_expand

```

Out[86]:

		director
title		
Dick Johnson Is Dead	**0**	Kirsten Johnson
Blood & Water	**0**	nan
Ganglands	**0**	Julien Leclercq
Jailbirds New Orleans	**0**	nan
Kota Factory	**0**	nan
...	**...**	...
Zodiac	**0**	David Fincher
Zombie Dumb	**0**	nan
Zombieland	**0**	Ruben Fleischer
Zoom	**0**	Peter Hewitt
Zubaan	**0**	Mozez Singh

9612 rows × 1 columns

In [87]:

```
# top 10 directors with most content
director_expand.value_counts().head(11)
```

Out[87]:

```
director
nan                    2634
Rajiv Chilaka            22
Jan Suter                21
Raúl Campos              19
Suhas Kadav              16
Marcus Raboy             16
Jay Karas                15
Cathy Garcia-Molina      13
Martin Scorsese          12
Youssef Chahine          12
Jay Chapman              12
dtype: int64
```

In [88]:

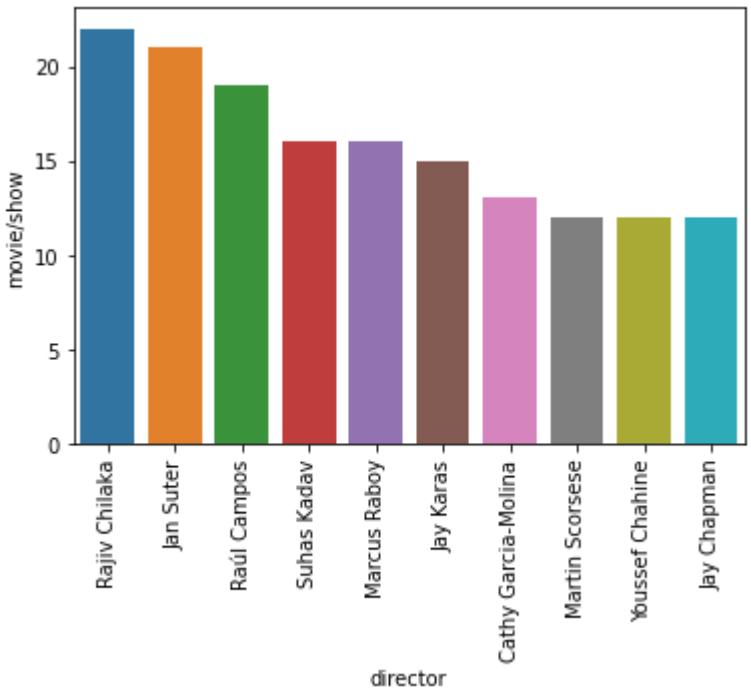
```
top10_director = director_expand.value_counts()[1:11]
top10_director = pd.DataFrame(top10_director, columns = ['movie/show']).reset_index()
top10_director
```

Out[88]:

	director	movie/show
0	Rajiv Chilaka	22
1	Jan Suter	21
2	Raúl Campos	19
3	Suhas Kadav	16
4	Marcus Raboy	16
5	Jay Karas	15
6	Cathy Garcia-Molina	13
7	Martin Scorsese	12
8	Youssef Chahine	12
9	Jay Chapman	12

In [90]:

```
sns.barplot(data= top10_director, x= 'director' , y='movie/show')
plt.xticks(rotation = 90)
plt.show()
```



In [91]:

```
categorised_director = pd.merge(director_expand, temp, left_index=True, right_index= Tr
categorised_director
```

Out[91]:

title		director	type
Dick Johnson Is Dead	**0**	Kirsten Johnson	Movie
Blood & Water	**0**	nan	TV Show
Ganglands	**0**	Julien Leclercq	TV Show
Jailbirds New Orleans	**0**	nan	TV Show
Kota Factory	**0**	nan	TV Show
...	**...**	...	...
Zodiac	**0**	David Fincher	Movie
Zombie Dumb	**0**	nan	TV Show
Zombieland	**0**	Ruben Fleischer	Movie
Zoom	**0**	Peter Hewitt	Movie
Zubaan	**0**	Mozez Singh	Movie

9612 rows × 2 columns

In [97]:

```
top_movie_dir = pd.DataFrame(categorised_director[categorised_director['type'] == 'Movi
top_movie_dir
```

Out[97]:

		No. of Movies
director	**type**	
nan	**Movie**	188
Rajiv Chilaka	**Movie**	22
Jan Suter	**Movie**	21
Raúl Campos	**Movie**	19
Suhas Kadav	**Movie**	16
Marcus Raboy	**Movie**	15
Jay Karas	**Movie**	15
Cathy Garcia-Molina	**Movie**	13
Youssef Chahine	**Movie**	12
Martin Scorsese	**Movie**	12

In [98]:

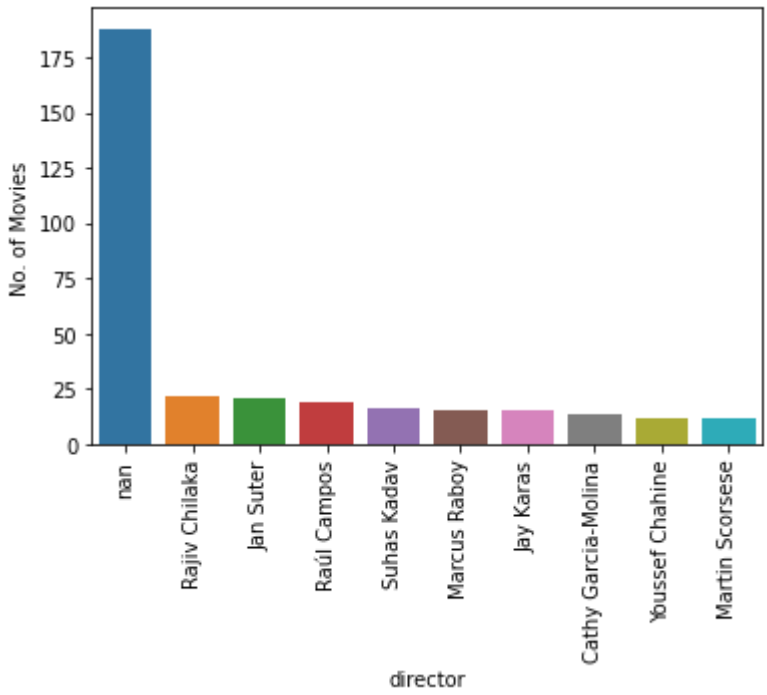
```
top_movie_dir.reset_index(1,drop=True,inplace=True)
top_movie_dir.reset_index(inplace = True)
top_movie_dir

```

Out[98]:

	director	No. of Movies
0	nan	188
1	Rajiv Chilaka	22
2	Jan Suter	21
3	Raúl Campos	19
4	Suhas Kadav	16
5	Marcus Raboy	15
6	Jay Karas	15
7	Cathy Garcia-Molina	13
8	Youssef Chahine	12
9	Martin Scorsese	12

In [99]:

```
sns.barplot(data=top_movie_dir, x= 'director', y='No. of Movies')
plt.xticks(rotation = 90)
plt.show()
```



In [100]:

```
rame(categorised_director[categorised_director['type'] == 'TV Show'].value_counts().head(10
```

Out[100]:

		No. of TV Shows
director	**type**	
nan	**TV Show**	2446
Ken Burns	**TV Show**	3
Alastair Fothergill	**TV Show**	3
Jung-ah Im	**TV Show**	2
Joe Berlinger	**TV Show**	2
Hsu Fu-chun	**TV Show**	2
Stan Lathan	**TV Show**	2
Gautham Vasudev Menon	**TV Show**	2
Lynn Novick	**TV Show**	2
Shin Won-ho	**TV Show**	2

In [101]:

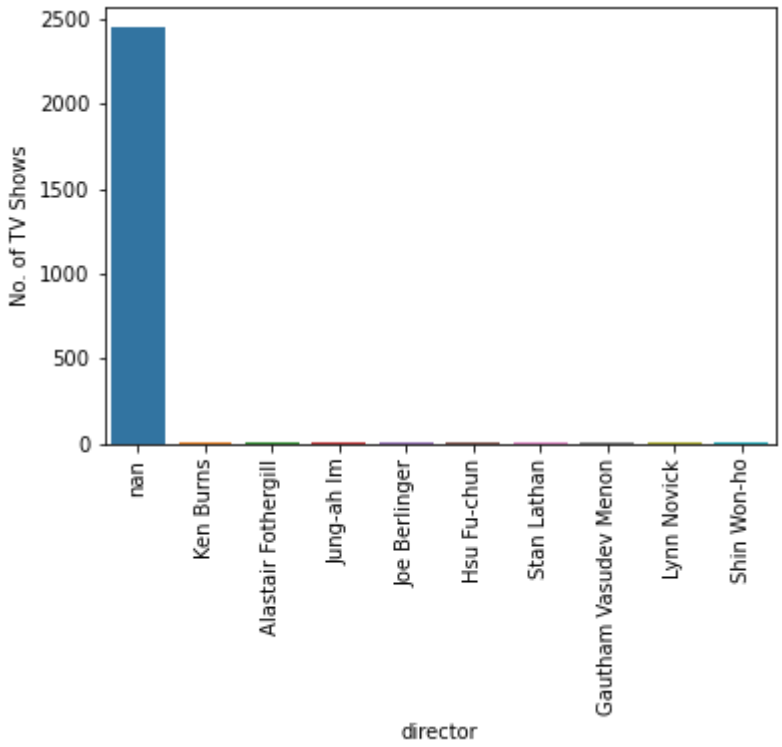
```
top_show_dir.reset_index(1,drop=True,inplace=True)
top_show_dir.reset_index(inplace = True)
top_show_dir
```

Out[101]:

	director	No. of TV Shows
0	nan	2446
1	Ken Burns	3
2	Alastair Fothergill	3
3	Jung-ah Im	2
4	Joe Berlinger	2
5	Hsu Fu-chun	2
6	Stan Lathan	2
7	Gautham Vasudev Menon	2
8	Lynn Novick	2
9	Shin Won-ho	2

In [102]:

```
sns.barplot(data=top_show_dir, x= 'director', y='No. of TV Shows')
plt.xticks(rotation = 90)
plt.show()

```



In [103]:

```
dire = director_expand.reset_index(1,drop=True)
```

In [104]:

```
df3= pd.merge(dire , temp2 , left_index=True , right_on='title')
df3

```

Out[104]:

	director	title	year_added
0	Kirsten Johnson	Dick Johnson Is Dead	2021.0
1	nan	Blood & Water	2021.0
2	Julien Leclercq	Ganglands	2021.0
3	nan	Jailbirds New Orleans	2021.0
4	nan	Kota Factory	2021.0
...	...	...	...
8802	David Fincher	Zodiac	2019.0
8803	nan	Zombie Dumb	2019.0
8804	Ruben Fleischer	Zombieland	2019.0
8805	Peter Hewitt	Zoom	2020.0
8806	Mozez Singh	Zubaan	2019.0

9612 rows × 3 columns

In [105]:

```
df3 = df3.sort_index()[['title', 'year_added' , 'director']]
df3
```

Out[105]:

	title	year_added	director
0	Dick Johnson Is Dead	2021.0	Kirsten Johnson
1	Blood & Water	2021.0	nan
2	Ganglands	2021.0	Julien Leclercq
3	Jailbirds New Orleans	2021.0	nan
4	Kota Factory	2021.0	nan
...	...	...	...
8802	Zodiac	2019.0	David Fincher
8803	Zombie Dumb	2019.0	nan
8804	Zombieland	2019.0	Ruben Fleischer
8805	Zoom	2020.0	Peter Hewitt
8806	Zubaan	2019.0	Mozez Singh

9612 rows × 3 columns

In [106]:

```
grp2 = df3.groupby(by='director')['year_added'].value_counts().sort_values(ascending =
```

In [107]:

```
grp2 = pd.DataFrame(grp2)
grp2.head()

```

Out[107]:

		year_added
director	**year_added**	
nan	**2019.0**	598
	2020.0	564
	2021.0	470
	2018.0	435
	2017.0	334

In [108]:

```
grp2 = grp2.drop(index= 'nan')
```

```
C:\Users\Shelendra\anaconda3\lib\site-packages\pandas\core\generic.py:4150:
PerformanceWarning: dropping on a non-lexsorted multi-index without a level
parameter may impact performance.
  obj = obj._drop_axis(labels, axis, level=level, errors=errors)
```

In [109]:

```
grp2.columns = ['No. of Movies/Shows']
grp2.reset_index(1)
```

Out[109]:

director	year_added	No. of Movies/Shows
Rajiv Chilaka	2021.0	17
Suhas Kadav	2021.0	15
Raúl Campos	2018.0	12
Jan Suter	2018.0	12
Youssef Chahine	2020.0	11
...	...	...
Huang Hsin-Yao	2021.0	1
Hua Shan	2018.0	1
Hsu Chih-yen	2021.0	1
Hsu Chih-yen	2019.0	1
Şenol Sönmez	2021.0	1

5982 rows × 2 columns

In [110]:

```
per_year1 = grp2.loc[top10_director['director']]
per_year1
```

Out[110]:

		No. of Movies/Shows
director	year_added	
Rajiv Chilaka	2021.0	17
	2019.0	3
	2020.0	2
Jan Suter	2018.0	12
	2017.0	5
	2016.0	4
Raúl Campos	2018.0	12
	2017.0	4
	2016.0	3
Suhas Kadav	2021.0	15
	2017.0	1
Marcus Raboy	2017.0	6
	2018.0	4
	2019.0	3
	2020.0	2
	2016.0	1
Jay Karas	2016.0	4
	2018.0	3
	2015.0	2
	2017.0	2
	2019.0	2
	2020.0	1
	2014.0	1
Cathy Garcia-Molina	2019.0	7
	2020.0	6
Martin Scorsese	2019.0	7
	2021.0	3
	2020.0	2
Youssef Chahine	2020.0	11
	2021.0	1
Jay Chapman	2017.0	7
	2019.0	2
	2018.0	1

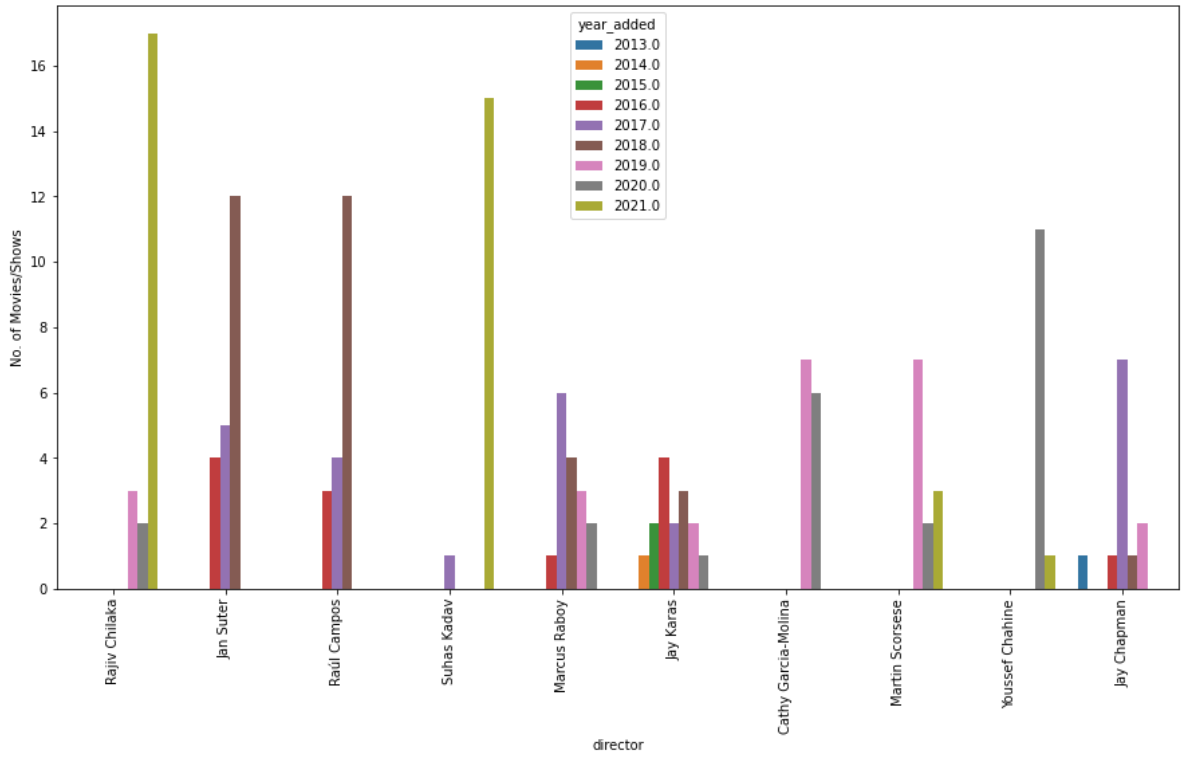
		No. of Movies/Shows
director	**year_added**	
	2016.0	1
	2013.0	1

In [111]:

```
per_year1.reset_index(inplace=True)
```

In [112]:

```
plt.figure(figsize=(15,8))
sns.barplot(data=per_year1, x ='director' , y = 'No. of Movies/Shows', hue= 'year_added
plt.xticks(rotation =90)
plt.show()
```




```
# unnesting lested_in
```

In [113]:

```
df.head()
```

Out[113]:

	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-13
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

In [114]:

```
# Seperating the genres into columns for each title
constraint3 = df['listed_in'].apply(lambda x: str(x).split(", ")).to_list()
genre_expand = pd.DataFrame(constraint3 , index=df['title'])

```

In [115]:

```
#stacking all genre columns into rows for each title
genre_expand = genre_expand.stack()
genre_expand = pd.DataFrame(genre_expand, columns=['genre'])
```

In [116]:

```
1 genre_expand
```

Out[116]:

		genre
title		
Dick Johnson Is Dead	**0**	Documentaries
Blood & Water	**0**	International TV Shows
	1	TV Dramas
	2	TV Mysteries
Ganglands	**0**	Crime TV Shows
...	**...**	...
Zoom	**0**	Children & Family Movies
	1	Comedies
Zubaan	**0**	Dramas
	1	International Movies
	2	Music & Musicals

19323 rows × 1 columns

In [117]:

```
1 genre_expand.value_counts().head(10)
```

Out[117]:

```
genre
International Movies        2752
Dramas                      2427
Comedies                    1674
International TV Shows      1351
Documentaries                869
Action & Adventure           859
TV Dramas                    763
Independent Movies           756
Children & Family Movies     641
Romantic Movies              616
dtype: int64
```

In [118]:

```
(genre_expand.value_counts(normalize=True)*100)[:10]

```

Out[118]:

```
genre
International Movies        14.242095
Dramas                      12.560161
Comedies                     8.663251
International TV Shows       6.991668
Documentaries                4.497231
Action & Adventure           4.445479
TV Dramas                    3.948662
Independent Movies           3.912436
Children & Family Movies     3.317290
Romantic Movies              3.187911
dtype: float64
```

In [119]:

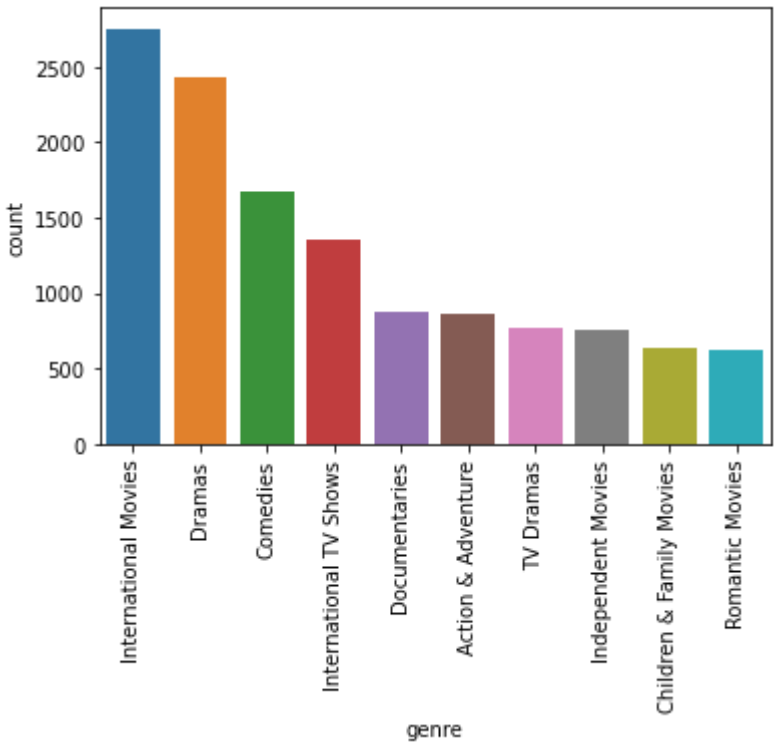
```
top_genre = pd.DataFrame(genre_expand.value_counts()[:10], columns=['count'])
top_genre.reset_index(inplace= True)
top_genre
```

Out[119]:

	genre	count
0	International Movies	2752
1	Dramas	2427
2	Comedies	1674
3	International TV Shows	1351
4	Documentaries	869
5	Action & Adventure	859
6	TV Dramas	763
7	Independent Movies	756
8	Children & Family Movies	641
9	Romantic Movies	616

In [120]:

```
sns.barplot(data=top_genre, x="genre", y ="count")
plt.xticks(rotation = 90)
plt.show()

```



In [121]:

```
genre = genre_expand.reset_index(1,drop=True)
```

In [122]:

```
df4= pd.merge(genre , temp2 , left_index=True , right_on='title')
df4
```

Out[122]:

	genre	title	year_added
0	Documentaries	Dick Johnson Is Dead	2021.0
1	International TV Shows	Blood & Water	2021.0
1	TV Dramas	Blood & Water	2021.0
1	TV Mysteries	Blood & Water	2021.0
2	Crime TV Shows	Ganglands	2021.0
...	...	...	...
8805	Children & Family Movies	Zoom	2020.0
8805	Comedies	Zoom	2020.0
8806	Dramas	Zubaan	2019.0
8806	International Movies	Zubaan	2019.0
8806	Music & Musicals	Zubaan	2019.0

19323 rows × 3 columns

In [123]:

```
df4.head()
```

Out[123]:

	genre	title	year_added
0	Documentaries	Dick Johnson Is Dead	2021.0
1	International TV Shows	Blood & Water	2021.0
1	TV Dramas	Blood & Water	2021.0
1	TV Mysteries	Blood & Water	2021.0
2	Crime TV Shows	Ganglands	2021.0

In [124]:

```
grp3 = df4.groupby(by='genre')['year_added'].value_counts().sort_values(ascending = Fal
grp3

```

Out[124]:

```
genre                     year_added
International Movies      2018.0        668
                          2019.0        610
                          2020.0        575
Dramas                    2019.0        564
                          2020.0        535
                                        ...
Classic & Cult TV         2016.0          1
                          2015.0          1
                          2014.0          1
Children & Family Movies  2012.0          1
Thrillers                 2011.0          1
Name: year_added, Length: 331, dtype: int64
```

In [125]:

```
grp3 = pd.DataFrame(grp3)
grp3.head()
```

Out[125]:

		year_added
genre	year_added	
International Movies	**2018.0**	668
	2019.0	610
	2020.0	575
Dramas	**2019.0**	564
	2020.0	535

In [126]:

```
grp3.columns = ['No. of Movies/Shows']
grp3.reset_index(1)
```

Out[126]:

genre	year_added	No. of Movies/Shows
International Movies	2018.0	668
International Movies	2019.0	610
International Movies	2020.0	575
Dramas	2019.0	564
Dramas	2020.0	535
...	...	...
Classic & Cult TV	2016.0	1
Classic & Cult TV	2015.0	1
Classic & Cult TV	2014.0	1
Children & Family Movies	2012.0	1
Thrillers	2011.0	1

331 rows × 2 columns

In [127]:

```
top_genre
```

Out[127]:

	genre	count
0	International Movies	2752
1	Dramas	2427
2	Comedies	1674
3	International TV Shows	1351
4	Documentaries	869
5	Action & Adventure	859
6	TV Dramas	763
7	Independent Movies	756
8	Children & Family Movies	641
9	Romantic Movies	616

In [128]:

```
per_year2 = grp3.loc[top_genre['genre']]
per_year2
```

Out[128]:

		No. of Movies/Shows
genre	**year_added**	
International Movies	**2018.0**	668
	2019.0	610
	2020.0	575
	2021.0	408
	2017.0	395
...	**...**	...
Romantic Movies	**2021.0**	114
	2018.0	108
	2017.0	63
	2016.0	7
	2015.0	1

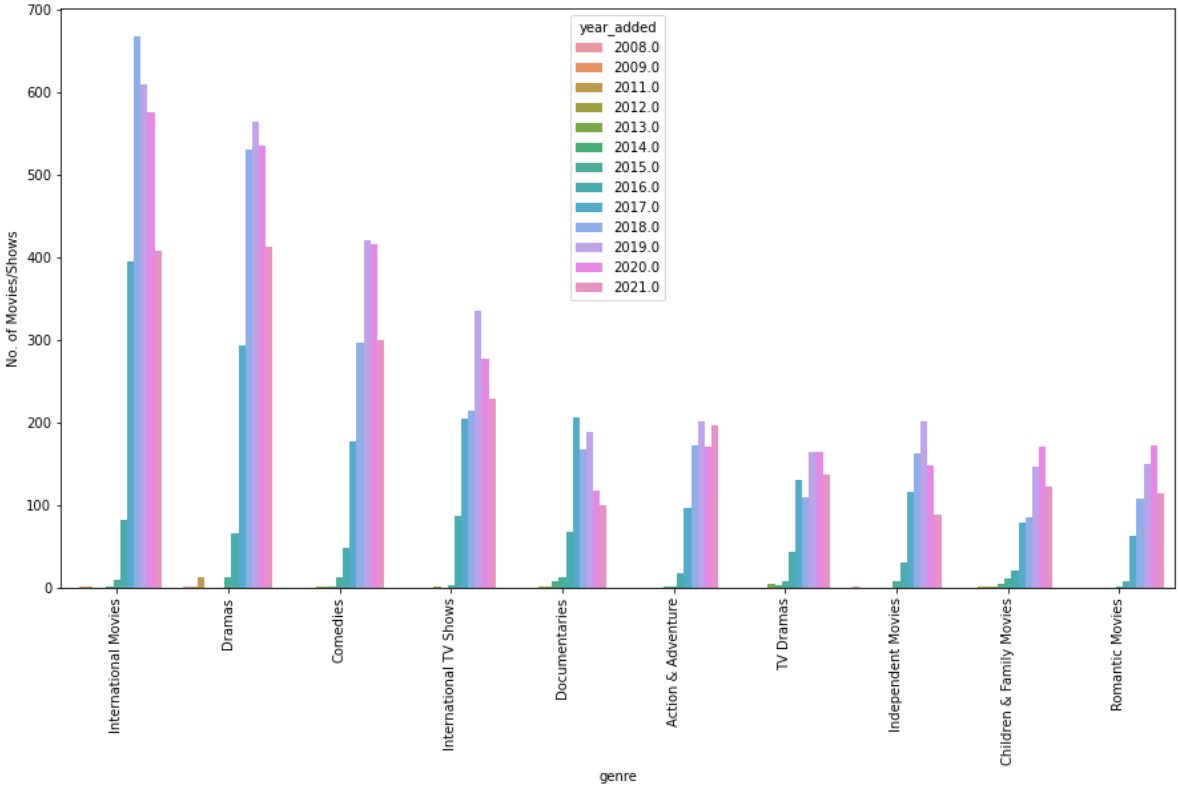
91 rows × 1 columns

In [129]:

```
per_year2.reset_index(inplace=True)
```

In [131]:

```
plt.figure(figsize=(15,8))
sns.barplot(data=per_year2, x ='genre' , y = 'No. of Movies/Shows', hue= 'year_added')
plt.xticks(rotation = 90)
plt.show()
```




```
# unnesting country
```

In [132]:

```
df.head()
```

Out[132]:

	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-13
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

In [133]:

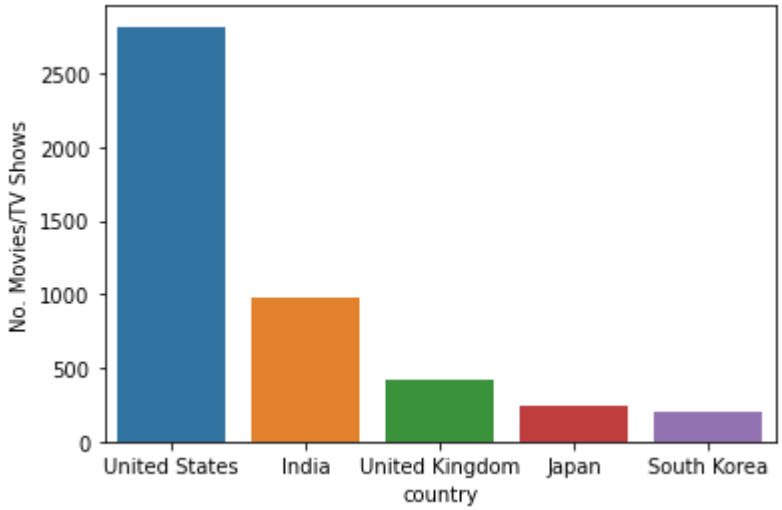
```
# Country with most content on Netflix
country = pd.DataFrame(df['country'].value_counts()[:5])
country.reset_index(inplace = True)
country.columns = ['country', 'No. Movies/TV Shows']
country
```

Out[133]:

	country	No. Movies/TV Shows
0	United States	2818
1	India	972
2	United Kingdom	419
3	Japan	245
4	South Korea	199

In [134]:

```
sns.barplot(data=country, x='country', y = 'No. Movies/TV Shows')
plt.show()
```



datatype cleaning

In [135]:

```
df.dtypes
```

Out[135]:

```
show_id          object
type             object
title            object
director         object
cast             object
country          object
date_added       object
release_year      int64
rating           object
duration         object
listed_in        object
description      object
year_added      float64
dtype: object
```

In [136]:

```
#here duration and date_added are string make it int
df['date_added'] = pd.to_datetime(df.date_added)
df.dtypes
```

Out[136]:

```
show_id                 object
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
year_added             float64
dtype: object
```

In [137]:

```
1 df.head()
```

Out[137]:

	show_id	type	title	director	cast	country	date_added	release_year	rating
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	2021-09-25	2020	PG-13
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	2021-09-24	2021	TV-MA
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	2021-09-24	2021	TV-MA
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	2021-09-24	2021	TV-MA
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	2021-09-24	2021	TV-MA

In [138]:

```
#fil null values of column 'date_added' with the most occuring date in data set
most_occur_date=df['date_added'].max()
df.fillna({'date_added': most_occur_date}, inplace=True)
df
```

Out[138]:

	show_id	type	title	director	cast	country	date_added	release_year
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	2021-09-25	2020
1	s2	TV Show	Blood & Water	NaN	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...	NaN	2021-09-24	2021
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	2021-09-24	2021
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	2021-09-24	2021
...	...	...	...	...	...	...	...	...
8802	s8803	Movie	Zodiac	David Fincher	Mark Ruffalo, Jake Gyllenhaal, Robert Downey J...	United States	2019-11-20	2007
8803	s8804	TV Show	Zombie Dumb	NaN	NaN	NaN	2019-07-01	2018
8804	s8805	Movie	Zombieland	Ruben Fleischer	Jesse Eisenberg, Woody Harrelson, Emma Stone, ...	United States	2019-11-01	2009

	show_id	type	title	director	cast	country	date_added	release_year
8805	s8806	Movie	Zoom	Peter Hewitt	Tim Allen, Courteney Cox, Chevy Chase, Kate Ma...	United States	2020-01-11	2006
8806	s8807	Movie	Zubaan	Mozez Singh	Vicky Kaushal, Sarah-Jane Dias, Raaghav Chanan...	India	2019-03-02	2015

8807 rows × 13 columns

In [139]:

```
# lets check
df[df.show_id == 's6067']
```

Out[139]:

	show_id	type	title	director	cast	country	date_added	release_year	rating
6066	s6067	TV Show	A Young Doctor's Notebook and Other Stories	NaN	Daniel Radcliffe, Jon Hamm, Adam Godley, Chris...	United Kingdom	2021-09-25	2013	TV-MA

In [140]:

```
df['duration_new']=df['duration'].apply(lambda x: str(x).split(' ')[0])
df
```

Out[140]:

	show_id	type	title	director	cast	country	date_added	release_year	rat
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	2021-09-25	2020	F
1	s2	TV Show	Blood & Water	NaN	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...	NaN	2021-09-24	2021	
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	2021-09-24	2021	
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	2021-09-24	2021	
...	...	...	...	...	...	...	...	...	
8802	s8803	Movie	Zodiac	David Fincher	Mark Ruffalo, Jake Gyllenhaal, Robert Downey J...	United States	2019-11-20	2007	
8803	s8804	TV Show	Zombie Dumb	NaN	NaN	NaN	2019-07-01	2018	TV
8804	s8805	Movie	Zombieland	Ruben Fleischer	Jesse Eisenberg, Woody Harrelson, Emma Stone, ...	United States	2019-11-01	2009	

	show_id	type	title	director	cast	country	date_added	release_year	rat
8805	s8806	Movie	Zoom	Peter Hewitt	Tim Allen, Courteney Cox, Chevy Chase, Kate Ma...	United States	2020-01-11	2006	
8806	s8807	Movie	Zubaan	Mozez Singh	Vicky Kaushal, Sarah-Jane Dias, Raaghav Chanan...	India	2019-03-02	2015	TV

8807 rows × 14 columns

In [141]:

```
df = df.drop('duration',axis = 1)
df
```

Out[141]:

	show_id	type	title	director	cast	country	date_added	release_year
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	2021-09-25	2020
1	s2	TV Show	Blood & Water	NaN	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...	NaN	2021-09-24	2021
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	2021-09-24	2021
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	2021-09-24	2021
...	...	...	...	...	...	...	...	...
8802	s8803	Movie	Zodiac	David Fincher	Mark Ruffalo, Jake Gyllenhaal, Robert Downey J...	United States	2019-11-20	2007
8803	s8804	TV Show	Zombie Dumb	NaN	NaN	NaN	2019-07-01	2018
8804	s8805	Movie	Zombieland	Ruben Fleischer	Jesse Eisenberg, Woody Harrelson, Emma Stone, ...	United States	2019-11-01	2009

	show_id	type	title	director	cast	country	date_added	release_year
8805	s8806	Movie	Zoom	Peter Hewitt	Tim Allen, Courteney Cox, Chevy Chase, Kate Ma...	United States	2020-01-11	2006
8806	s8807	Movie	Zubaan	Mozez Singh	Vicky Kaushal, Sarah-Jane Dias, Raaghav Chanan...	India	2019-03-02	2015

8807 rows × 13 columns

In [146]:

```
df['year'] = pd.DatetimeIndex(df['date_added']).year
```

In [147]:

```
df.duration_new.value_counts().head(10)
```

Out[147]:

```
1     1793
2      425
3      200
90     152
94     146
93     146
97     146
91     144
95     137
96     130
Name: duration_new, dtype: int64
```

```
# Number of movies released per year
```

In [148]:

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

Out[148]:

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

In [150]:

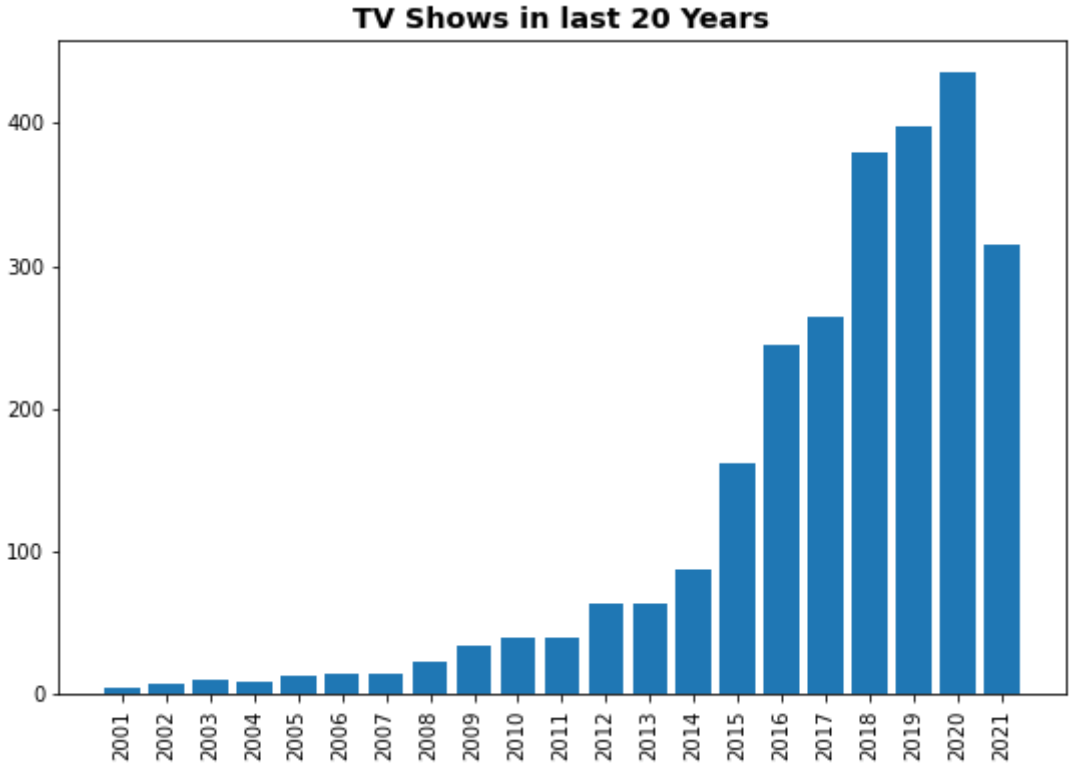
```
df_agg = df.groupby(['type'])[['title','year']].aggregate({'year':['min','max'],'title'
df_agg
```

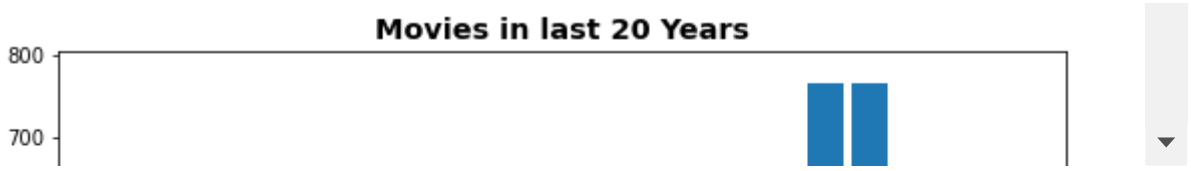
Out[150]:

	year		title
	min	max	count
type			
Movie	2008	2021	6131
TV Show	2008	2021	2676

In [151]:

```
past20_yr = df.loc[(df['release_year']>=2001)].pivot_table(index=['release_year'],colum
past20_yr
#4:58
plt.figure(figsize=(9,6))
plt.bar(x = past20_yr['release_year'].values, height = past20_yr['title']['TV Show'].va
plt.xticks(past20_yr.release_year.values,rotation='90')
plt.title("TV Shows in last 20 Years",fontdict={"weight":"bold","size":"x-large"})
plt.show()
#5:01
#for movies
plt.figure(figsize=(9,6))
plt.bar(x = past20_yr['release_year'].values,height = past20_yr['title']['Movie'].value
plt.xticks(past20_yr.release_year.values,rotation='90')
plt.title("Movies in last 20 Years",fontdict={"weight":"bold","size":"x-large"})
plt.show()
```

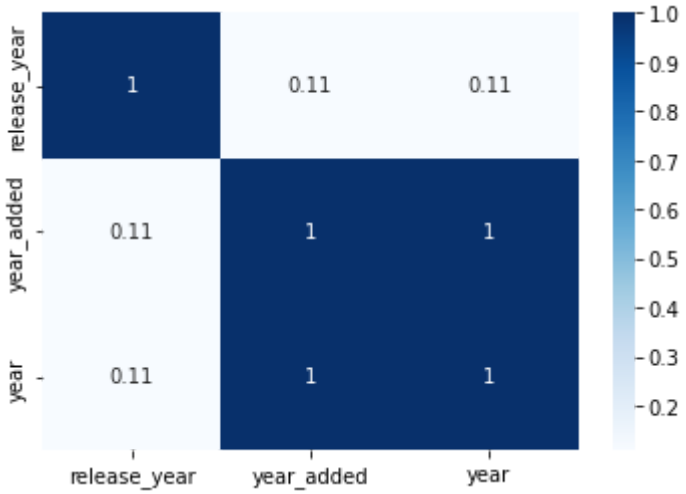





```
1 # Heat map
```

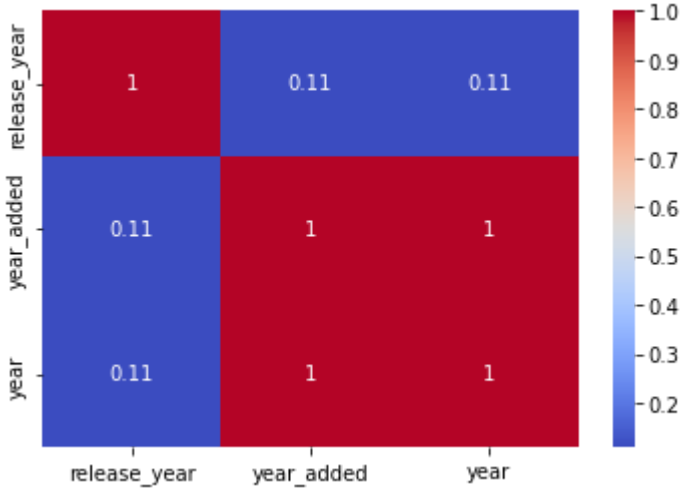
In [152]:

```
1 sns.heatmap(df.corr(), cmap= "Blues", annot=True)
2 plt.show()
```



In [153]:

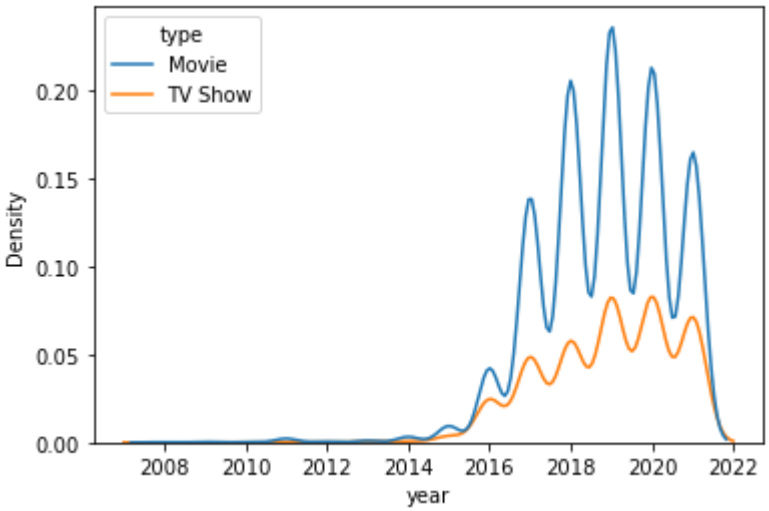
```
1 sns.heatmap(df.corr(), cmap= "coolwarm", annot=True)
2 plt.show()
```




```
1 # KDE PLOT
```

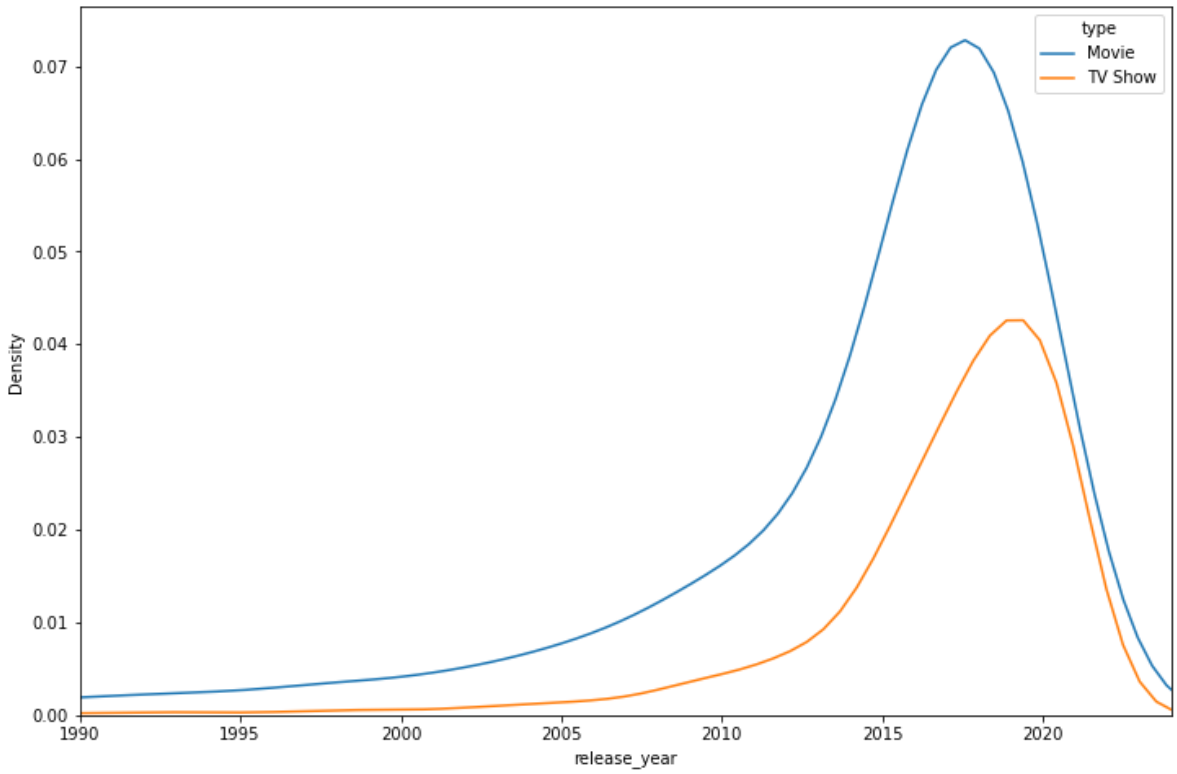
In [154]:

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



In [155]:

```
plt.figure(figsize = (12,8))
sns.kdeplot(data = df , x = 'release_year',hue = 'type')
plt.xlim(1990,2024)
plt.show()
```



In [ ]:

```
1
```