In [ ]:

```
#Defining Problem Statement and Analysing basic metrics:

understand the data

identify insights/patterns in the data . downtrend,uptrend etc.
provide recommendations to netflex to grow their business or where to concen
assuming this data is for liked/interested data. more movies in perticular c
figure out how to compute missing data
figure out wrong data
figure out relationship between features
```

In [1]:

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

In [2]:

```
raw=pd.read_csv("/Users/naresh6/Downloads/netflix.csv")
```

In [3]:

```
#2. Observations on the shape of data, data types of all the attributes, con
#'category' (If required), missing value detection, statistical summary (10
```

In [4]:

```
raw.shape
#total 8807 shows and 12 features for each show discribes
```

Out[4]:

```
(8807, 12)
```

In [5]:

```
raw.dtypes
#release year is int.
#all other columns are chars,strings or floats and bydefault they are treate
```

Out[5]:

```
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
dtype: object
```

In [6]:

```
raw.head()
#no float . only interger and string columns
```

Out [6]:

	show_id	type	title	director	cast	country	date_added	release_year	ratin
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 M.
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV M.
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV M.
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV M.

In [8]:

```
#converted all categorical columns to category dtypes
for column in raw.columns[~raw.columns.isin(["description","duration","relea

    raw[column]=raw[column].astype("category")
raw.dtypes
```

Out[8]:

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

In [9]:

```
#missing value detection
raw.isna().sum()
#director,cast,country,date_added has null values in few observations . dire
```

Out[9]:
```
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 [10]:
```
raw.describe()  #release_year summary
```

Out[10]:

	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 [14]:
```
raw.describe(include = ['O',"category"])  # all category & objectcolumns sum
```

Out[14]:

	show_id	type	title	director	cast	country	date_added	rating	duratio
count	8807	8807	8807	6173	7982	7976	8797	8803	880
unique	8807	2	8807	4528	7692	748	1767	17	22
top	s1	Movie	#Alive	Rajiv Chilaka	David Attenborough	United States	January 1, 2020	TV-MA	1 Seaso
freq	1	6131	1	19	19	2818	109	3207	179

In [16]:
```
#3. Non-Graphical Analysis: Value counts and unique attributes (10 Points)
```

In [17]:
```
#value counts for categories:
raw.describe(include = 'category')
#type has only 2 unique values . majority is movies than tv shows
#majority of shows comes from United states
#majority of hows has only. 1 season duration
#majority shows listed in Dramas, International Movies
```

Out[17]:

	show_id	type	title	director	cast	country	date_added	rating	duratio
count	8807	8807	8807	6173	7982	7976	8797	8803	880
unique	8807	2	8807	4528	7692	748	1767	17	22
top	s1	Movie	#Alive	Rajiv Chilaka	David Attenborough	United States	January 1, 2020	TV-MA	1 Seaso
freq	1	6131	1	19	19	2818	109	3207	179

In [15]:

```
raw["date_added"]=pd.to_datetime(raw["date_added"])
raw["day_added"]=raw["date_added"].dt.day
raw["month_added"]=raw["date_added"].dt.month
raw["year_added"]=raw["date_added"].dt.year
raw["weekday_added"]=raw["date_added"].dt.weekday
```

In [16]:

```
#value counts for integer:
raw.describe()
#releasing year is left skewed: mean<median
```

Out[16]:

	release_year	day_added	month_added	year_added	weekday_added
count	8807.000000	8797.000000	8797.000000	8797.000000	8797.000000
mean	2014.180198	12.497329	6.654996	2018.871888	3.016824
std	8.819312	9.887551	3.436554	1.574243	1.727631
min	1925.000000	1.000000	1.000000	2008.000000	0.000000
25%	2013.000000	1.000000	4.000000	2018.000000	2.000000
50%	2017.000000	13.000000	7.000000	2019.000000	3.000000
75%	2019.000000	20.000000	10.000000	2020.000000	4.000000
max	2021.000000	31.000000	12.000000	2021.000000	6.000000

In [17]:

```
print(raw.release_year.astype("category").describe())
#majority of movies are released in 2018 year. arround 1147 shows released i

print(raw.release_year.astype("category").describe().freq/raw.shape[0])
#arround 13% of shows are released in this year
```

```
count     8807
unique      74
top       2018
freq      1147
Name: release_year, dtype: int64
0.13023731122970364
```

In [18]:

```
raw.loc[raw["duration"].isna(),["duration"]]=raw[raw["duration"].isna()]["ra
#filling na values of duration from rating
```

In [19]:

```
def unpivot(df,column,pattern):
    colarr=df[column].apply(lambda x:str(x).split(", ")).to_list()

    coldf=pd.DataFrame(colarr,index=df["title"]).reset_index()
    unpivot=pd.melt(coldf, id_vars="title", var_name="count", value_name=col
    unpivot=unpivot[~unpivot[column].isna()]
    unpivot[column]=unpivot[column].apply(lambda s:s.strip())
    return unpivot



directors=unpivot(raw[["title","director"]],"director",", ")
casts=unpivot(raw[["title","cast"]],"cast",", ")
listed_ins=unpivot(raw[["title","listed_in"]],"listed_in",",")
countries=unpivot(raw[["title","country"]],"country",", ")

nest_cats_mapper={"director":directors,"cast":casts,"listed_in":listed_ins,"
```

In [20]:

```
directors.head()
```

Out[20]:

	title	count	director
0	Dick Johnson Is Dead	0	Kirsten Johnson
1	Blood & Water	0	nan
2	Ganglands	0	Julien Leclercq
3	Jailbirds New Orleans	0	nan
4	Kota Factory	0	nan

In [22]:

```
dir_freq=directors["director"].value_counts()
```

In [23]:

```
dir_freq
```

Out[23]:

```
nan                   2634
Rajiv Chilaka           22
Jan Suter               21
RaÃºl Campos            19
Suhas Kadav             16
                      ...
Lee Kyoungmi             1
John R. Leonetti         1
Jeremiah Jones           1
Brie Larson              1
Mark Henn                1
Name: director, Length: 4994, dtype: int64
```

In [24]:

```
pd.DataFrame(dir_freq)
```

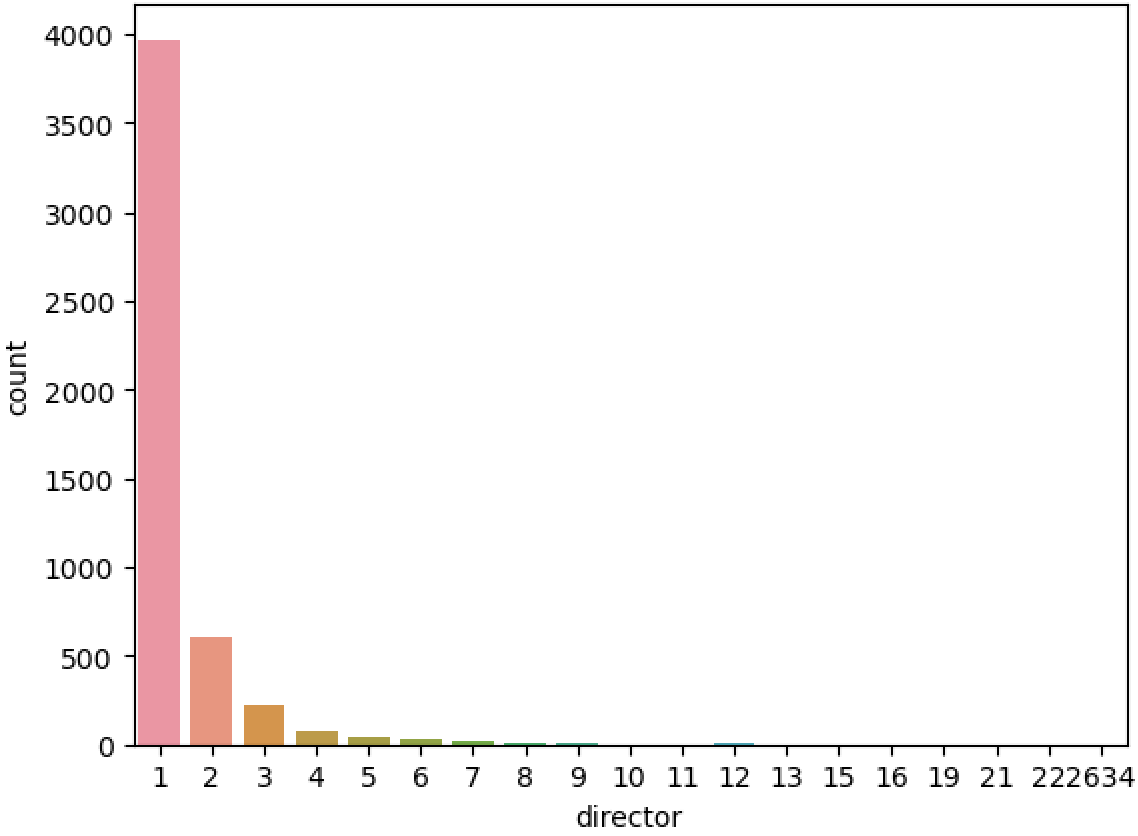
Out[24]:

	director
nan	2634
Rajiv Chilaka	22
Jan Suter	21
RaÃºl Campos	19
Suhas Kadav	16
...	...
Lee Kyoungmi	1
John R. Leonetti	1
Jeremiah Jones	1
Brie Larson	1
Mark Henn	1

4994 rows × 1 columns

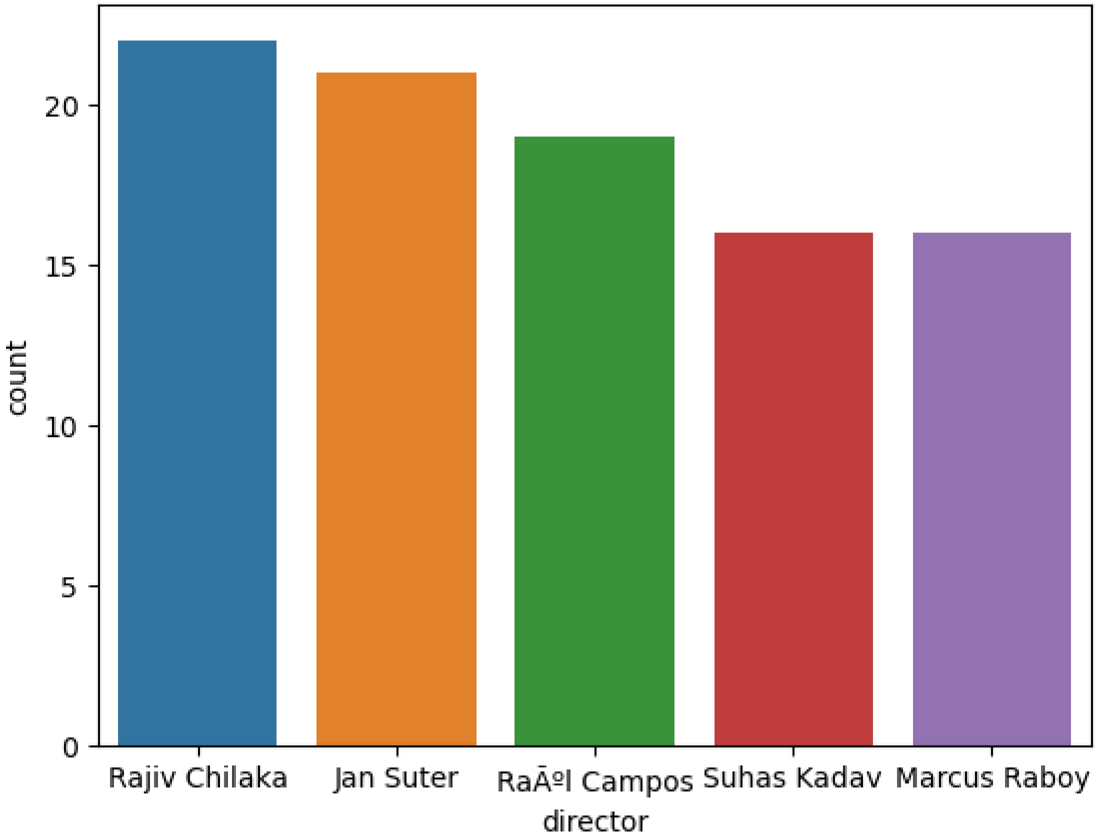
In [25]:

```
sns.countplot(x="director",data =pd.DataFrame(dir_freq))
plt.show()

#majority of directors did only 1 movie
```



In [28]:

```
ntile_dir=directors["director"].value_counts().quantile(q=0.999,interpolatic

top_dir=directors["director"].value_counts()[directors["director"].value_cou
top_dir
top_dirdf=directors[directors["director"].isin(top_dir.index[1:])]
sns.countplot(x="director",data =top_dirdf,order=top_dir.index[1:])
plt.show()
#most frequent directoers. netflix can give more attention to these director
```



In [29]:

```
casts["cast"].value_counts()
```

Out[29]:

```
nan                   825
Anupam Kher            43
Shah Rukh Khan         35
Julie Tejwani          33
Naseeruddin Shah       32
                      ...
Kim Hyun-wook           1
Ã‰lodie Bouchez         1
Anthony Quayle          1
Elva Josephson          1
Ayize Ma'at             1
Name: cast, Length: 36440, dtype: int64
```

In [30]:

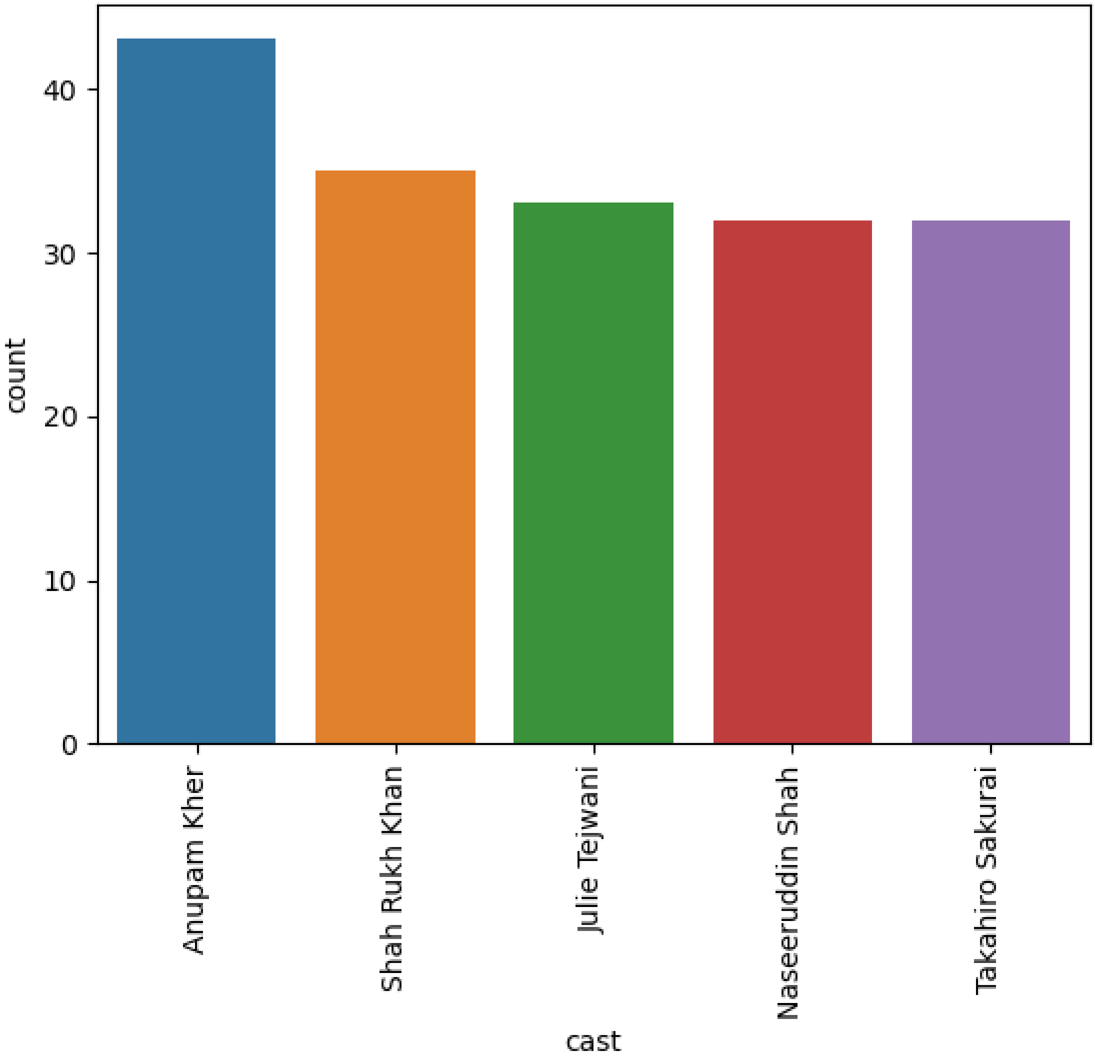
```
casts["cast"].value_counts().quantile(q=0.9999,interpolation="lower")
```

Out[30]:

```
32
```

In [32]:

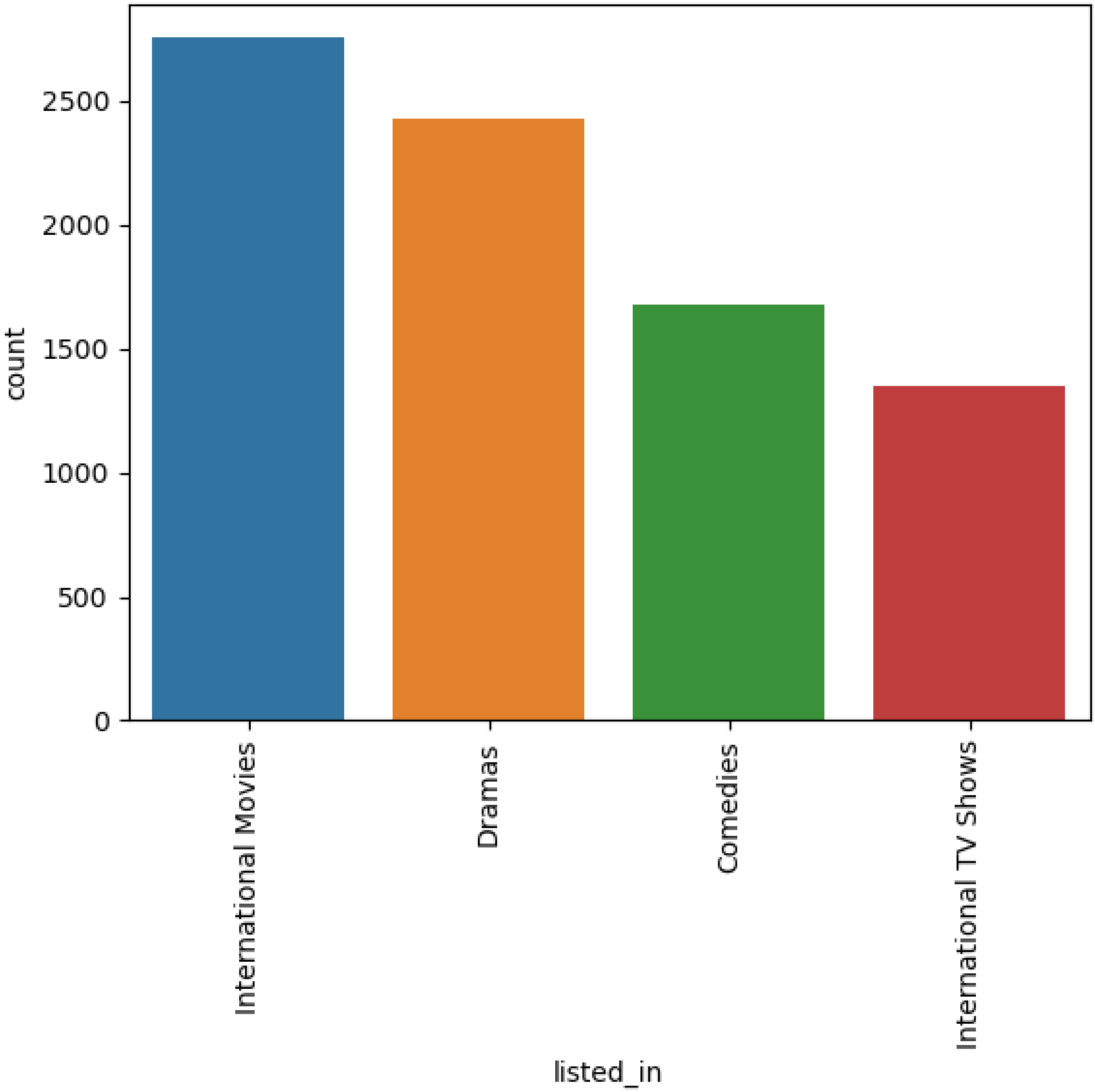
```
ntile_cast=casts["cast"].value_counts().quantile(q=0.9999,interpolation="low

top_cast=casts["cast"].value_counts()[casts["cast"].value_counts()>=ntile_ca
top_cast
top_castdf=casts[casts["cast"].isin(top_cast.index[1:])]
sns.countplot(x="cast",data =top_castdf,order=top_cast.index[1:])
plt.xticks(rotation=90)
plt.show()
#most frequent cast. netflix can give more attention to these cast especiall
```



In [33]:
```
listed_ins["listed_in"].value_counts().quantile(q=0.95,interpolation="lower"
```

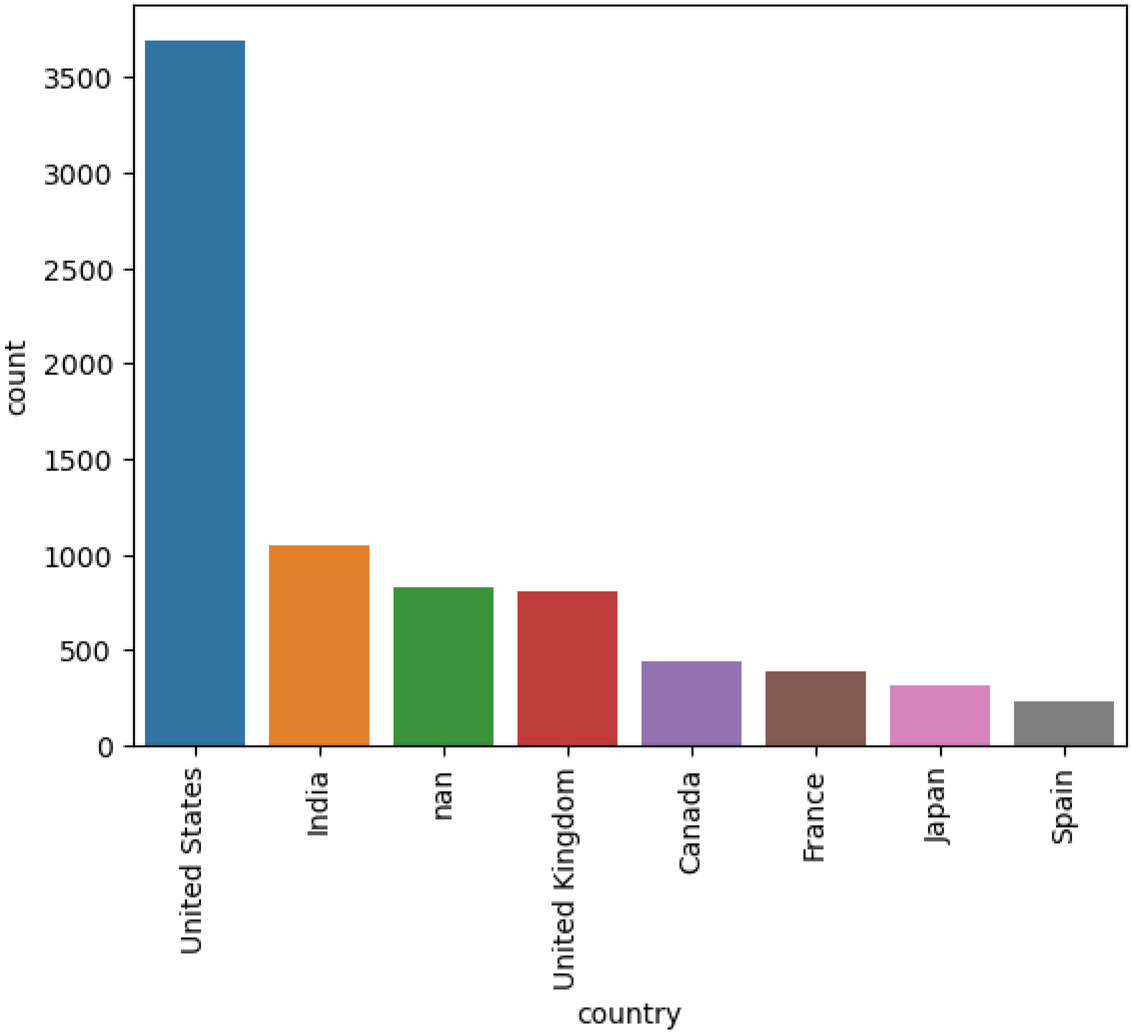
Out[33]:
```
1351
```

In [34]:
```
ntile_list=listed_ins["listed_in"].value_counts().quantile(q=0.95,interpolat

top_list=listed_ins["listed_in"].value_counts()[listed_ins["listed_in"].valu
top_list
top_listdf=listed_ins[listed_ins["listed_in"].isin(top_list.index)]
sns.countplot(x="listed_in",data =top_listdf,order=top_list.index)
plt.xticks(rotation=90)
plt.show()
#most frequent listed in is international movies. netflix can give more atte
```



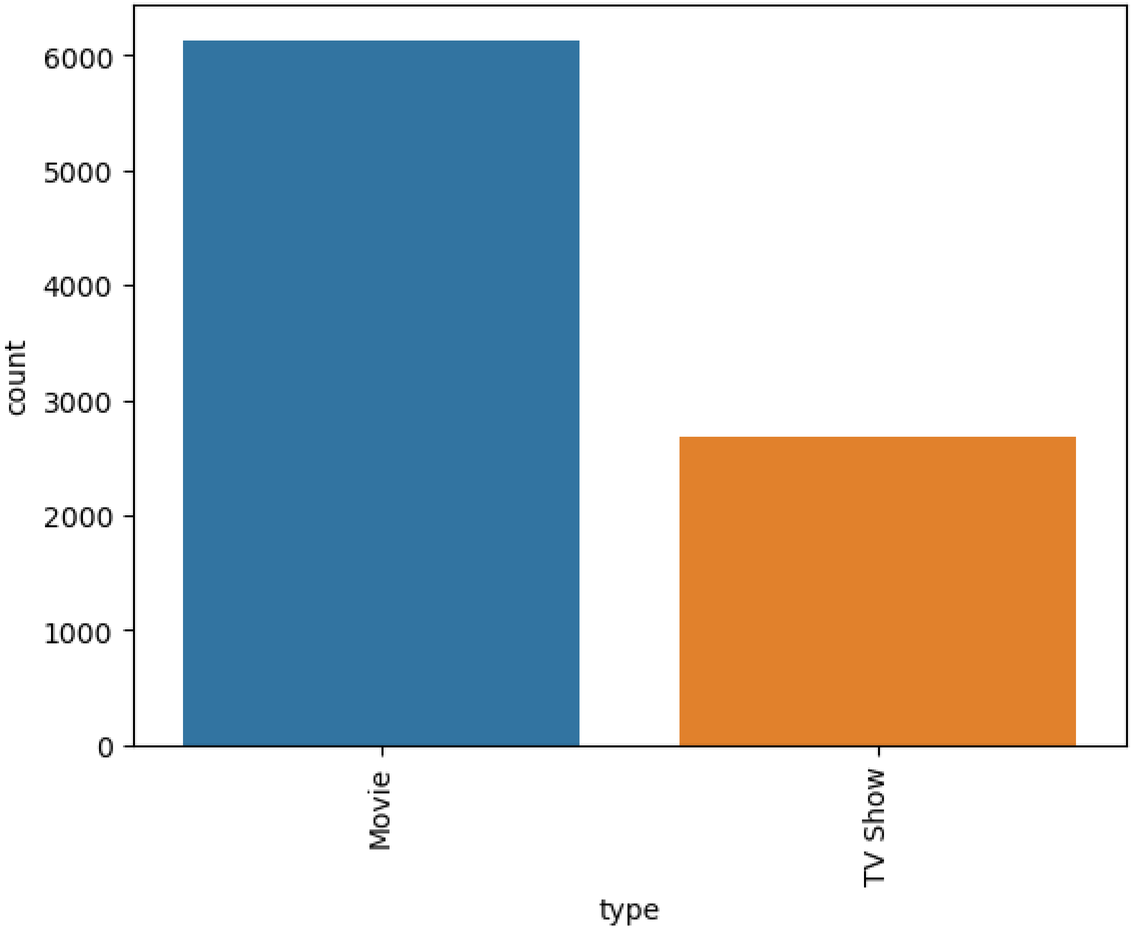
In [35]:

```
ntile_country=countries["country"].value_counts().quantile(q=0.95,interpolat

top_country=countries["country"].value_counts()[countries["country"].value_c
top_country
top_countrydf=countries[countries["country"].isin(top_country.index)]
sns.countplot(x="country",data =top_countrydf,order=top_country.index)
plt.xticks(rotation=90)
plt.show()
#most movies comes from united states. netflix can give more attention to th
```



In [36]:

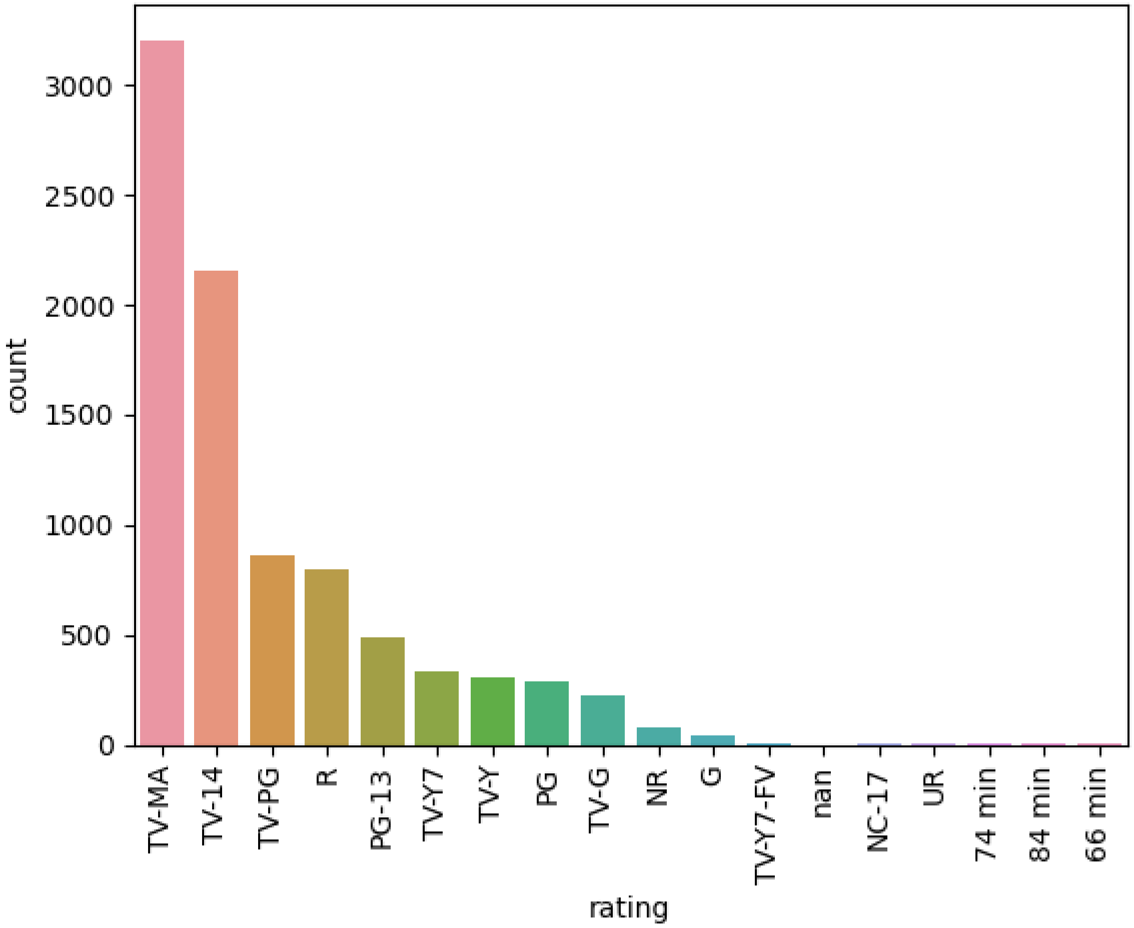
```
sns.countplot(x="type",data =raw)
plt.xticks(rotation=90)
plt.show()
#netflix can concentrate more on getting movies
```



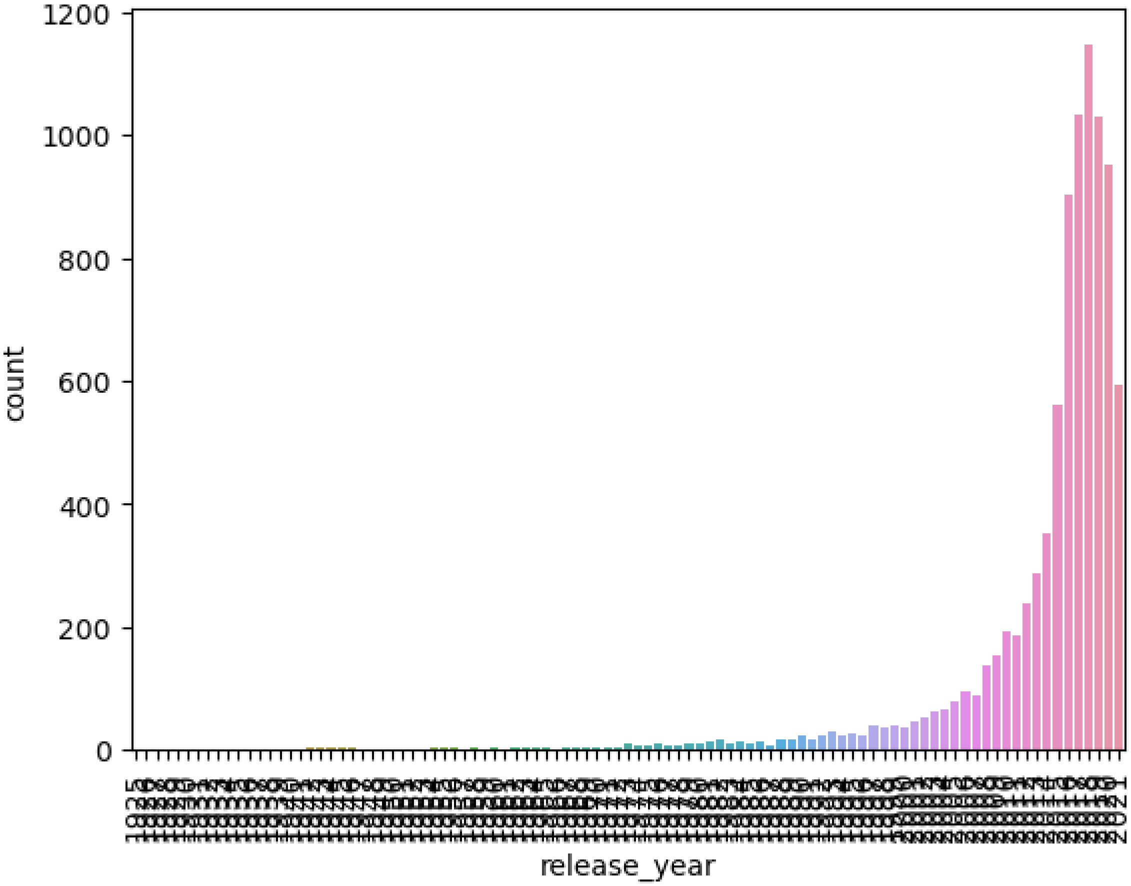
In [37]:
```
raw["rating"].value_counts(dropna=False)
```

Out[37]:
```
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
NaN             4
NC-17           3
UR              3
74 min          1
84 min          1
66 min          1
Name: rating, dtype: int64
```

In [38]:
```
sns.countplot(x="rating",data =raw,order=raw["rating"].value_counts(dropna=F
plt.xticks(rotation=90)
plt.show()
#netflix can concentrate more on getting shows having rating TV-MA
```

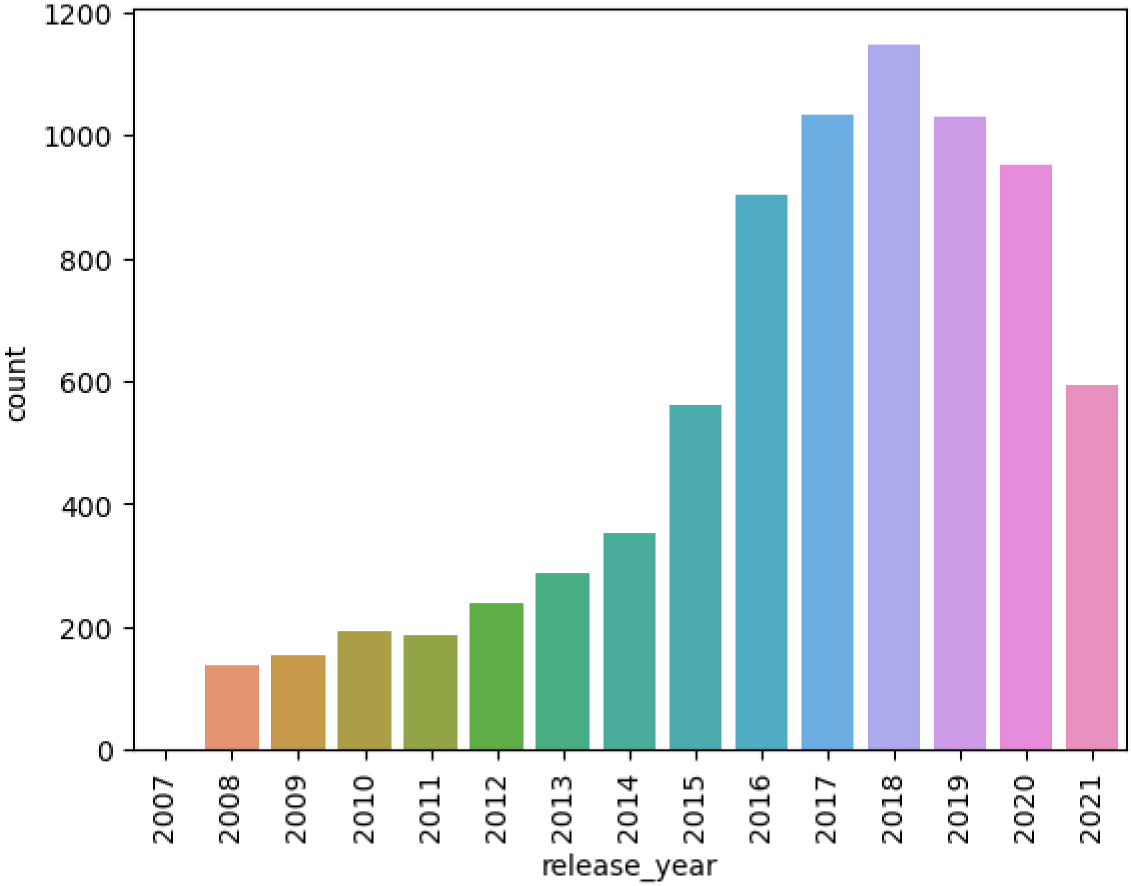


In [39]:
```
years=list(np.arange(raw["release_year"].min(),raw["release_year"].max()+1))
sns.countplot(x="release_year",data =raw,order=years)
plt.xticks(rotation=90)

plt.show()
```

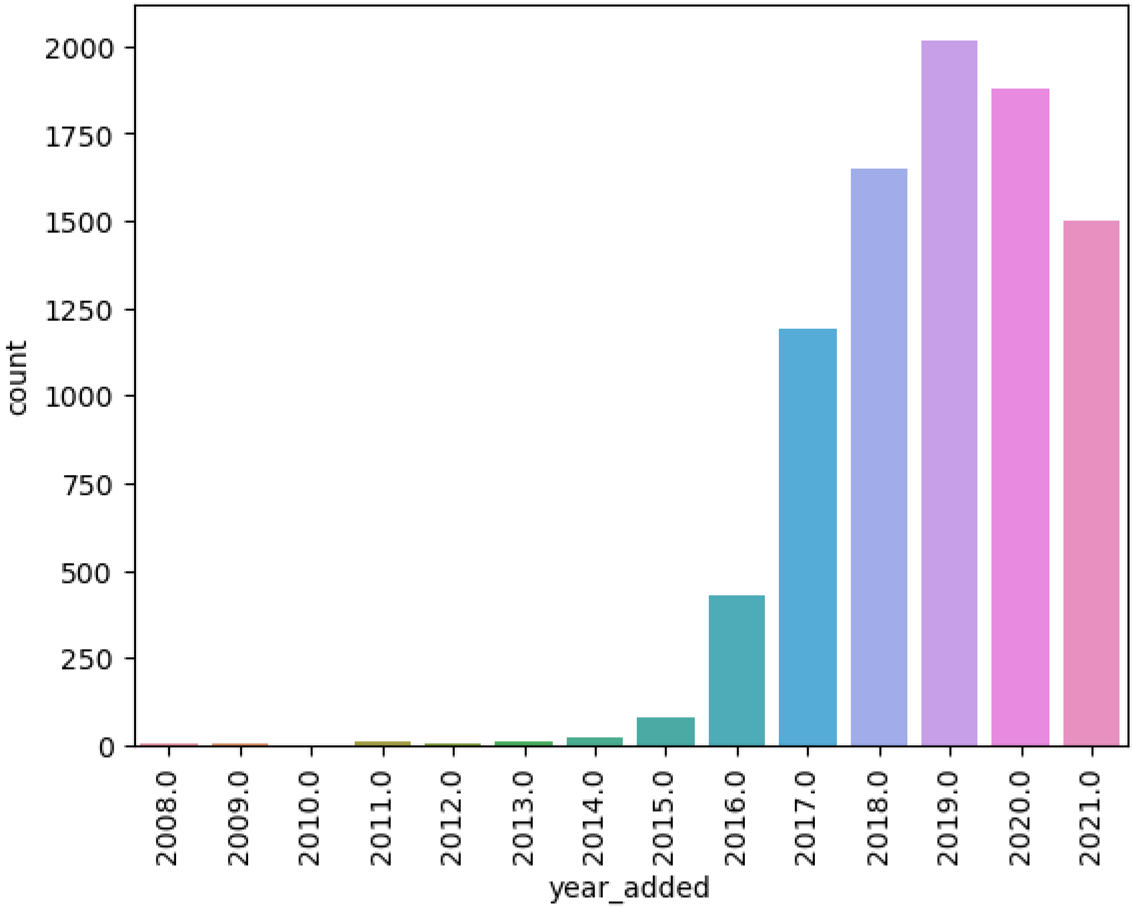


In [40]:

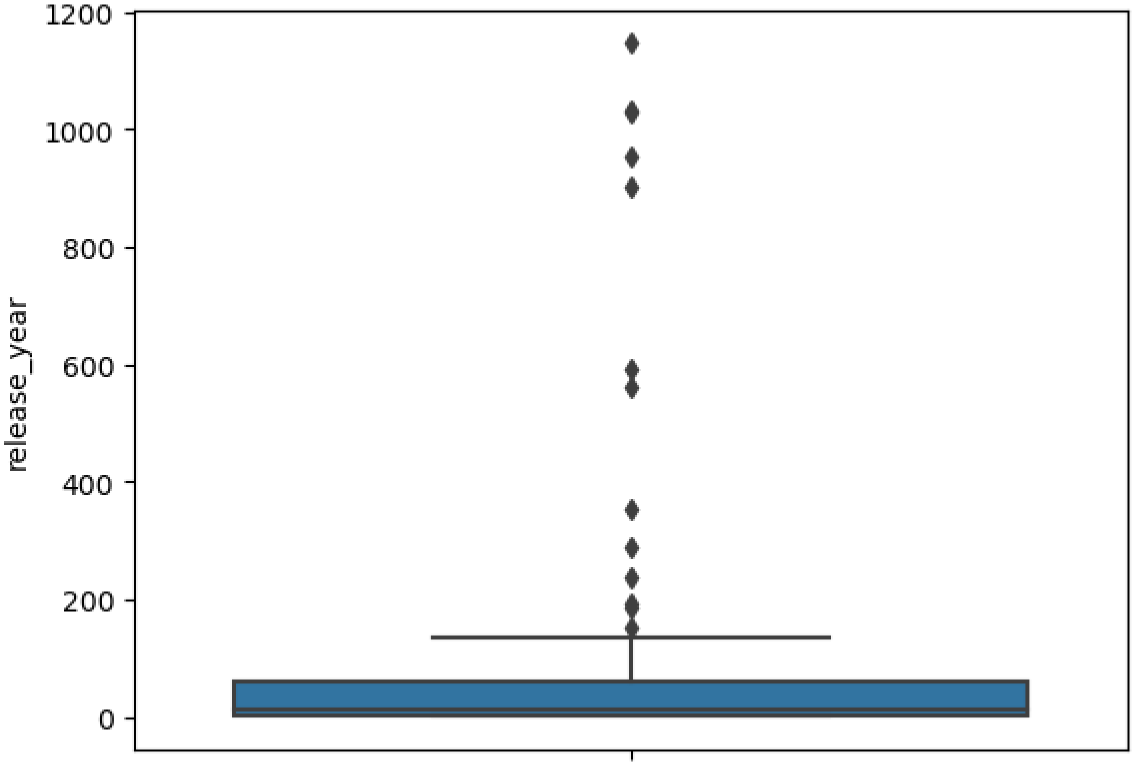
```
years=list(np.arange(2007,raw["release_year"].max()+1))
sns.countplot(x="release_year",data =raw[raw["release_year"]>2007],order=yea
plt.xticks(rotation=90)

plt.show()
#netflix should focus more on latest movies
```

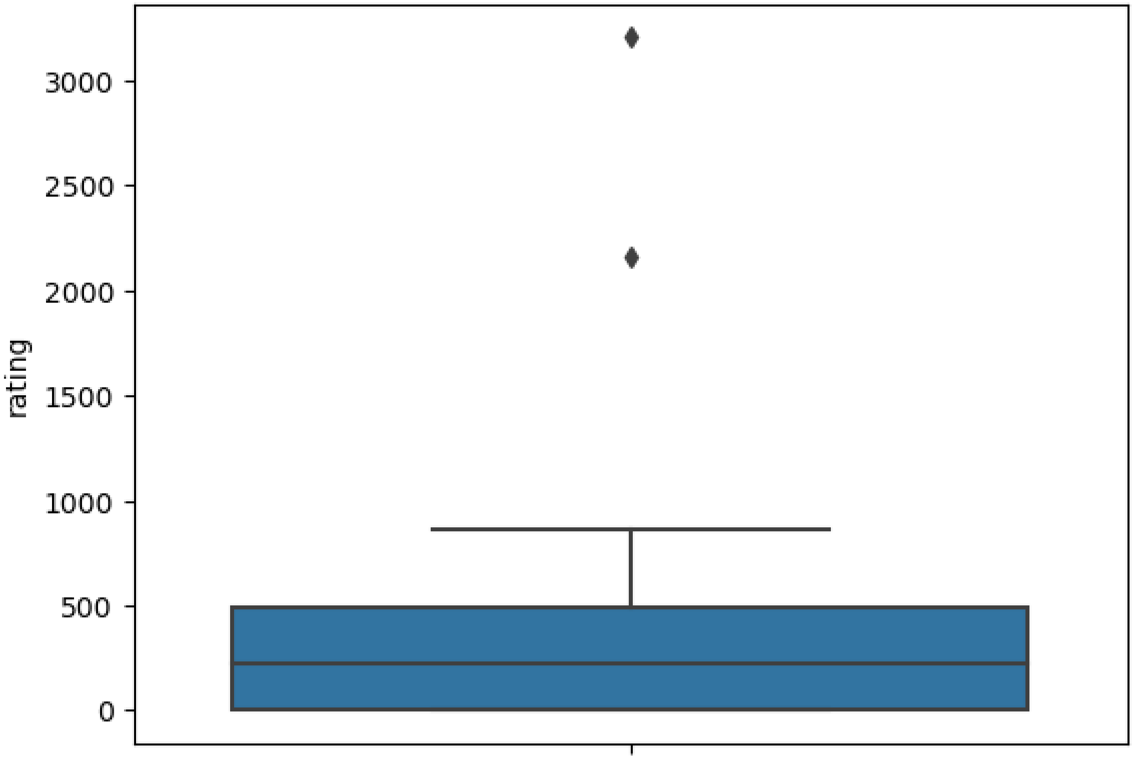


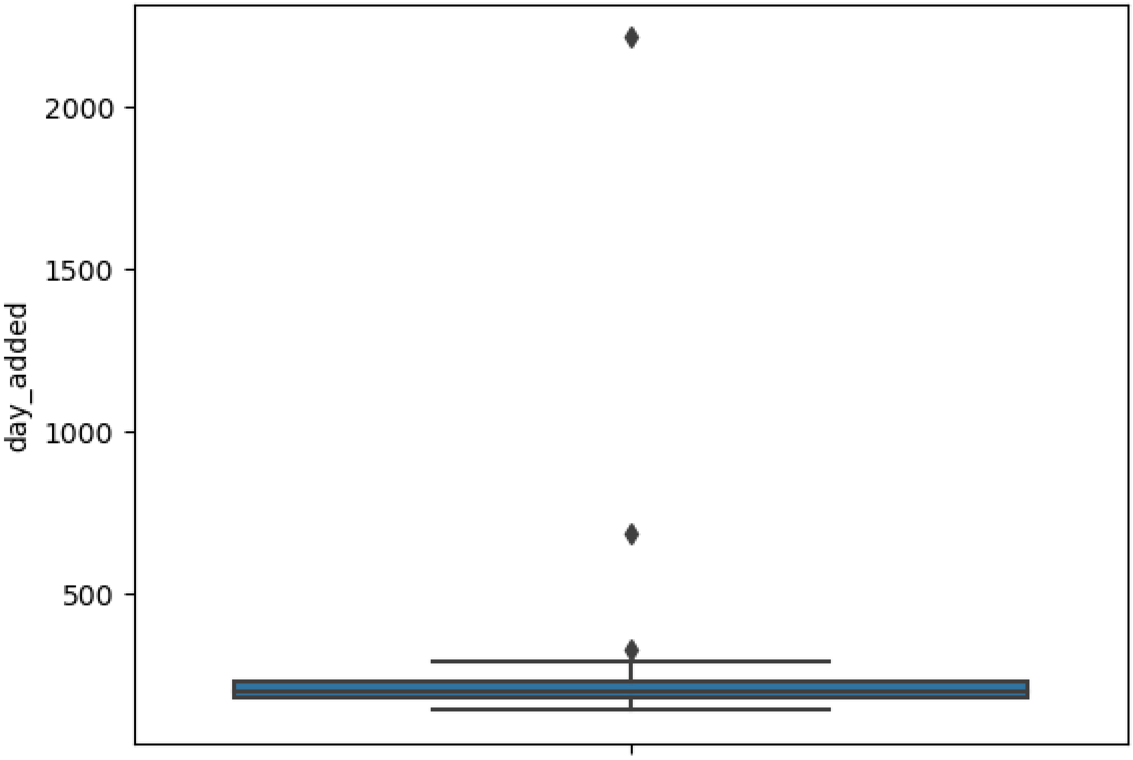
In [41]:
```
years=list(np.arange(raw["year_added"].min(),raw["year_added"].max()+1))
sns.countplot(x="year_added",data =raw,order=years)
plt.xticks(rotation=90)
plt.show()

#all movies are added on or after 2008 . maybe netflex started on that year.
```

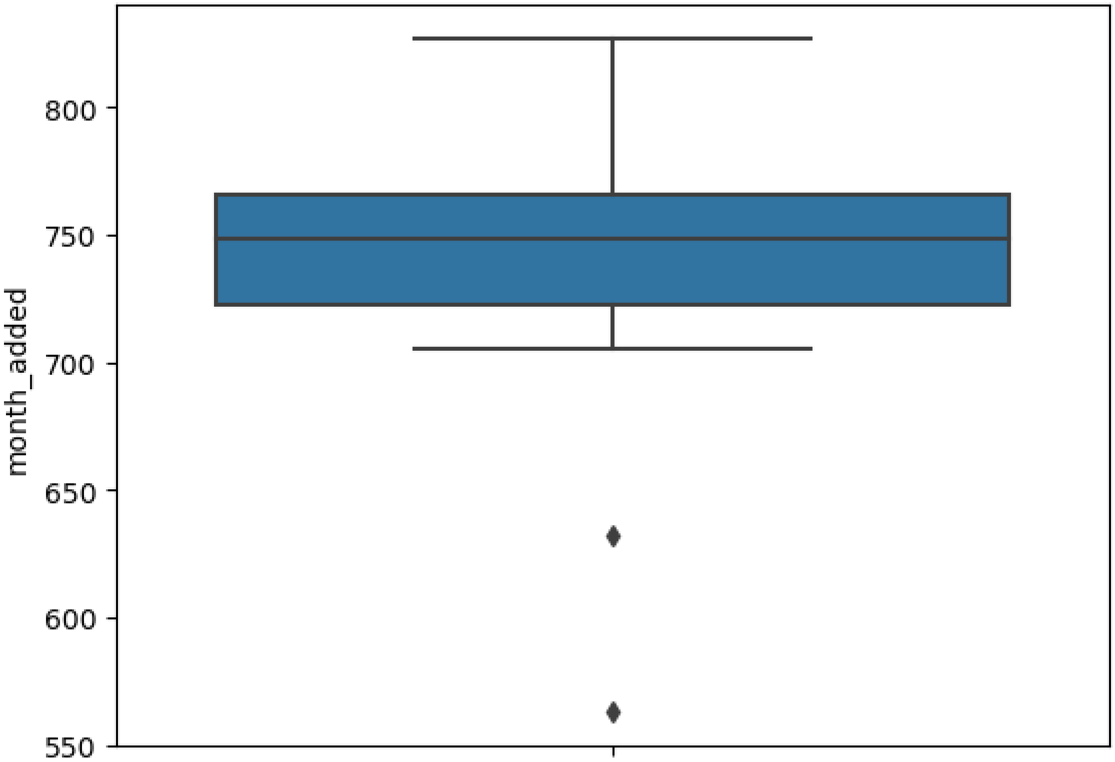


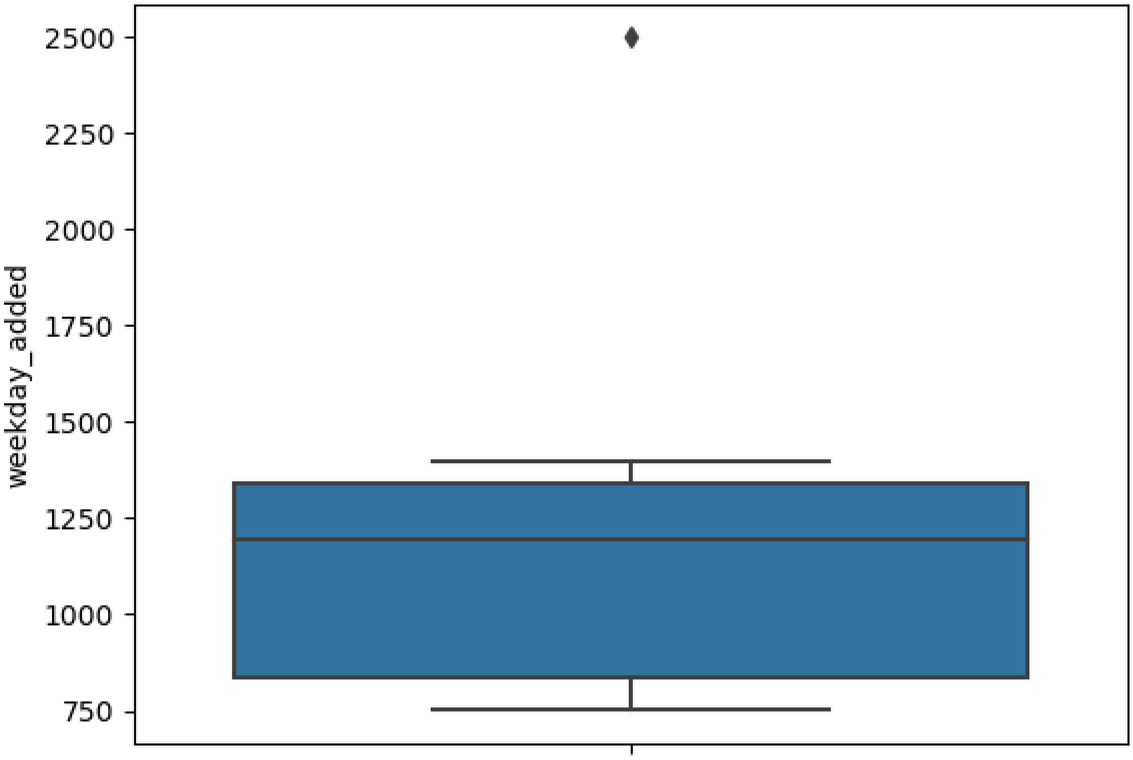
```
In [44]: #4.2 For categorical variable(s): Boxplot (10 Points)
         raw_cat=["release_year","rating","day_added","month_added","weekday_added","
         for cat in raw_cat:
             catdf=raw.groupby(cat)[cat].count()
             sns.boxplot(y = catdf)
             plt.show()
```

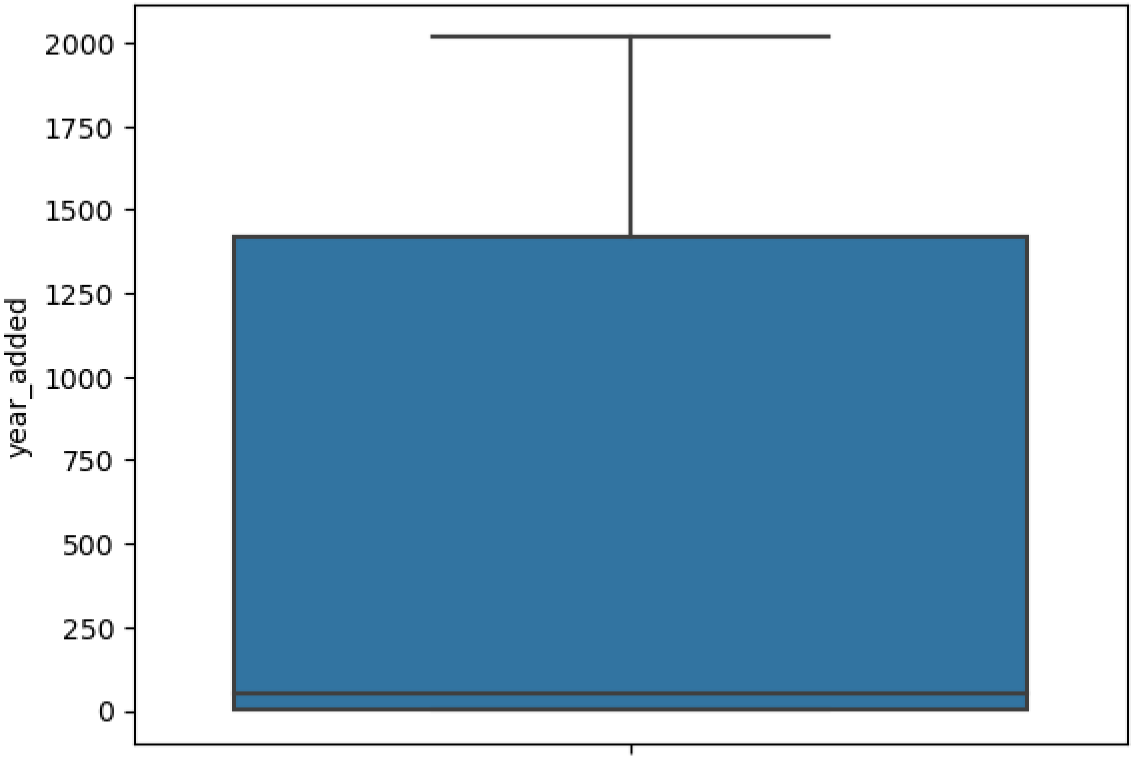



rating
3000
2500
2000
1500
1000
500
0

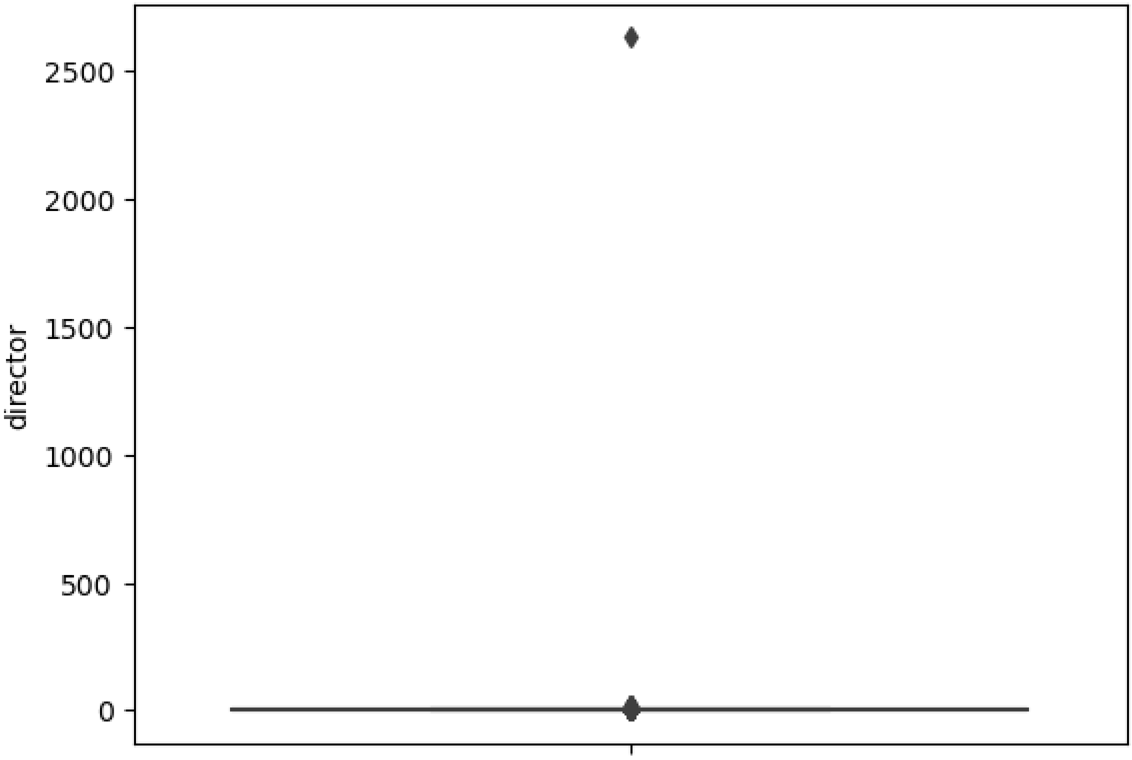

day_added
2000
1500
1000
500

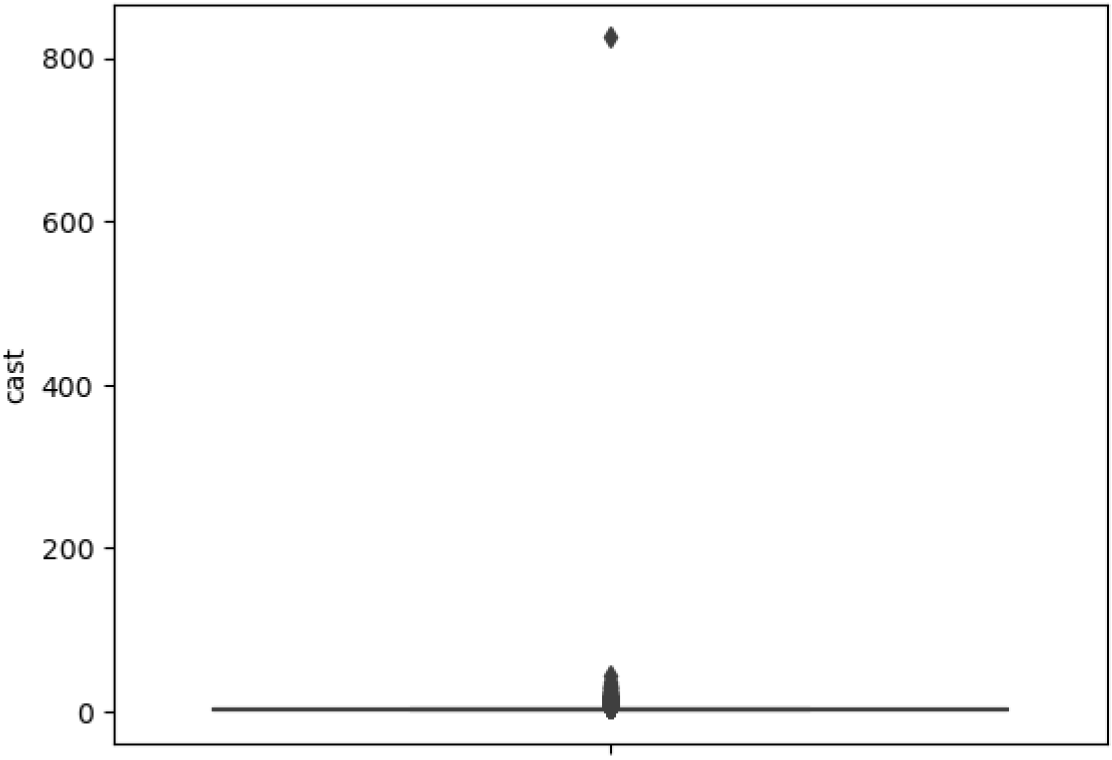

800
750
700
650
600
550
month_added

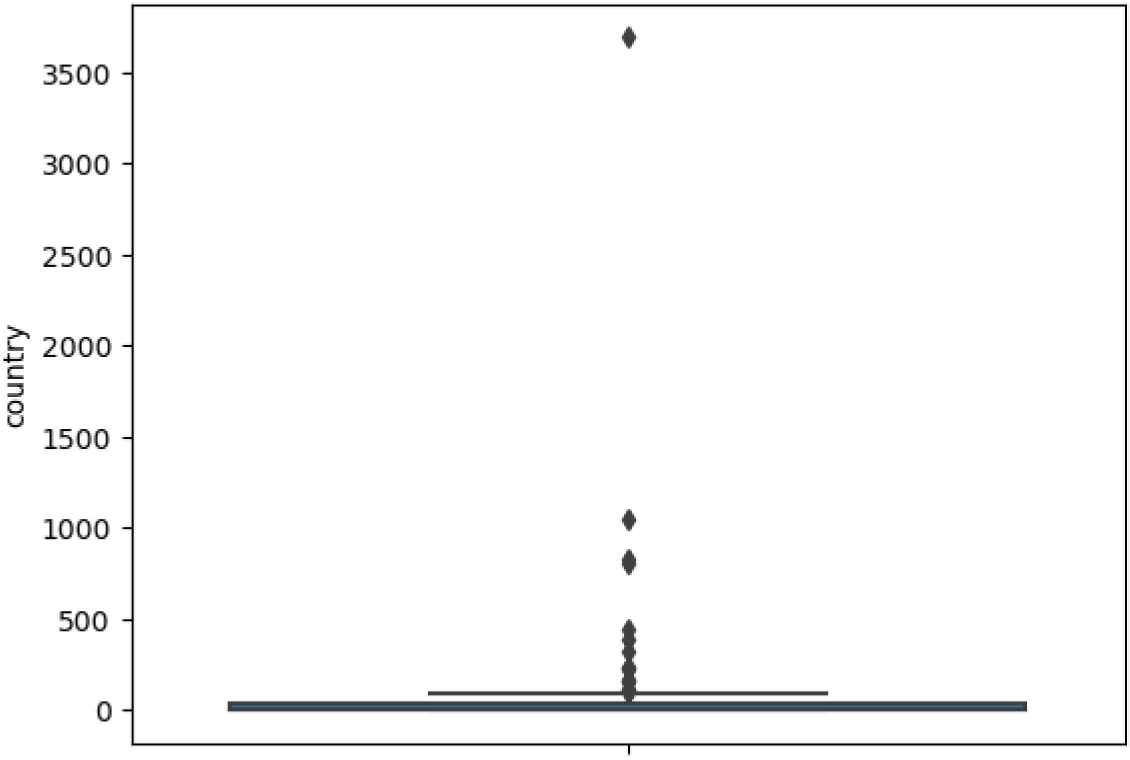

2500
2250
2000
1750
1500
1250
1000
750
weekday_added

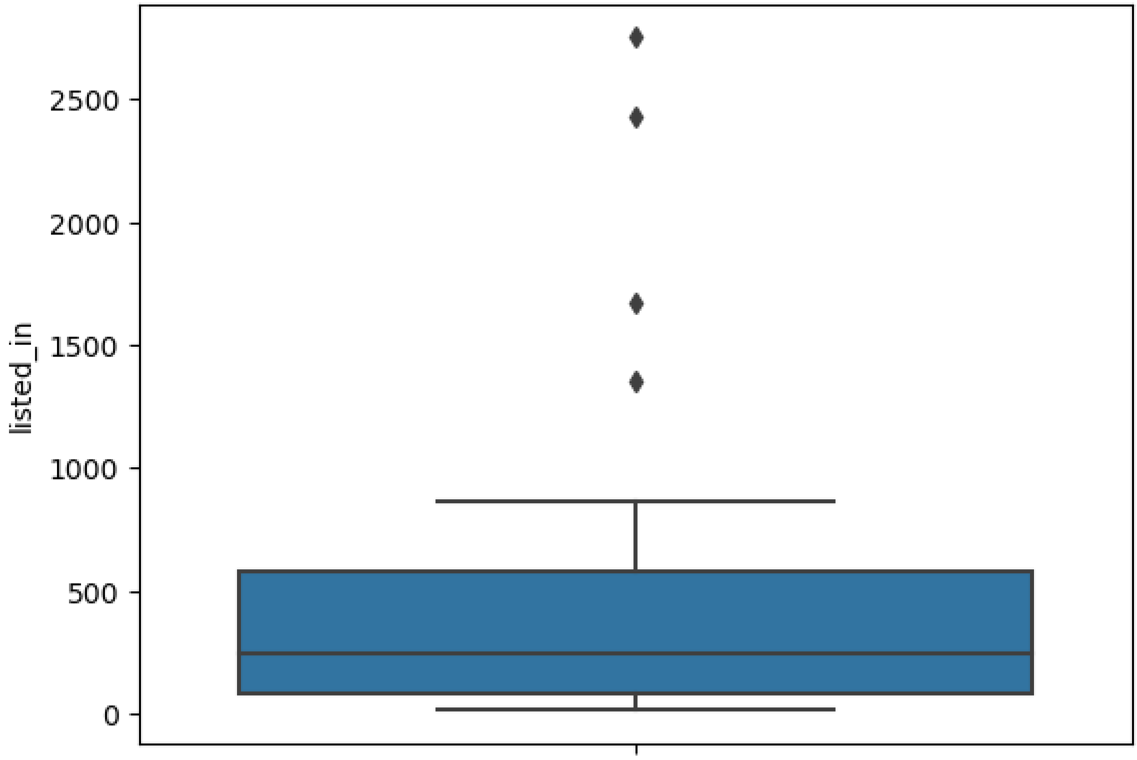


In [45]:
```
nest_cats=["director","cast","country","listed_in"]
for cat in nest_cats:
    catdf= nest_cats_mapper[cat]
    catdf=catdf.groupby(cat)[cat].count()
    sns.boxplot(y = catdf)
    plt.show()
```




800
600
cast
400
200
0


3500
3000
2500
country
2000
1500
1000
500
0



In [47]:

```
def unpivotcnt(df,column,pattern):
    df[column+"cnt"]=df[column].apply(lambda x:len(str(x).split(", ")))




unpivotcnt(raw,"director",", ")
unpivotcnt(raw,"cast",", ")
unpivotcnt(raw,"listed_in",",")
unpivotcnt(raw,"country",", ")
raw["duration_0"]=raw["duration"].apply(lambda x:str(x).split()[0])
#raw["duration_0"].astype('int64')
raw.head()
```

Out[47]:

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

5 rows × 21 columns

In [48]:

```
shows=raw[raw["type"]=="TV Show"]
movies=raw[raw["type"]=="Movie"]
```

In [49]:

```
shows["duration_0"]=shows["duration_0"].astype("int64")
movies["duration_0"]=movies["duration_0"].astype("int64")
```

```
/var/folders/4q/h408ttzx43n2tyts5rkyb668ky0h13/T/ipykernel_26306/2079896317.
py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/
stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  shows["duration_0"]=shows["duration_0"].astype("int64")
/var/folders/4q/h408ttzx43n2tyts5rkyb668ky0h13/T/ipykernel_26306/2079896317.
py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/
stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  movies["duration_0"]=movies["duration_0"].astype("int64")
```

In [50]:

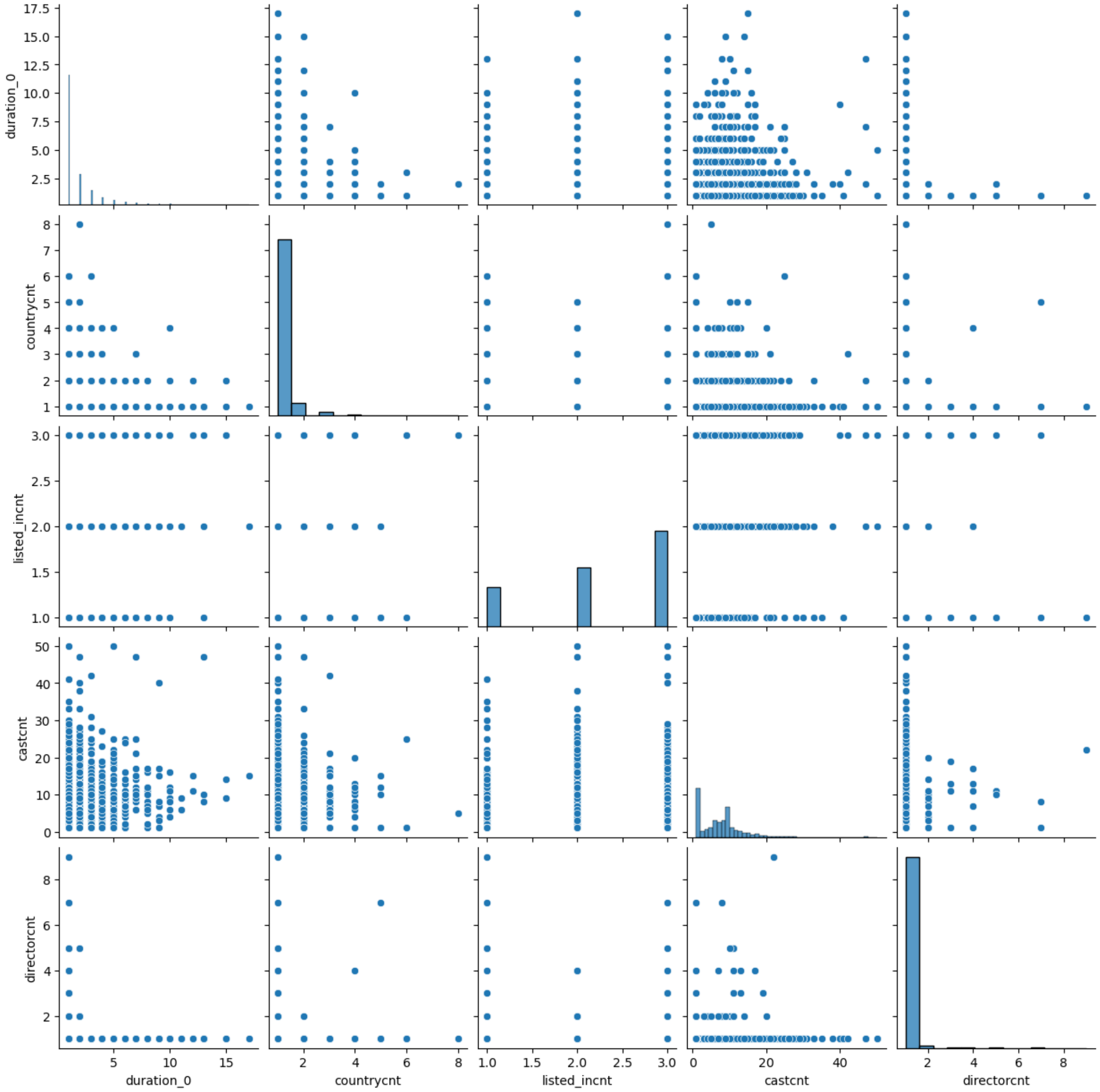
```
shows.dtypes
```

Out[50]:

```
show_id                category
type                   category
title                  category
director                 object
cast                     object
country                  object
date_added       datetime64[ns]
release_year              int64
rating                 category
duration                 object
listed_in                object
description              object
day_added               float64
month_added             float64
year_added              float64
weekday_added           float64
directorcnt               int64
castcnt                   int64
listed_incnt              int64
countrycnt                int64
duration_0                int64
dtype: object
```

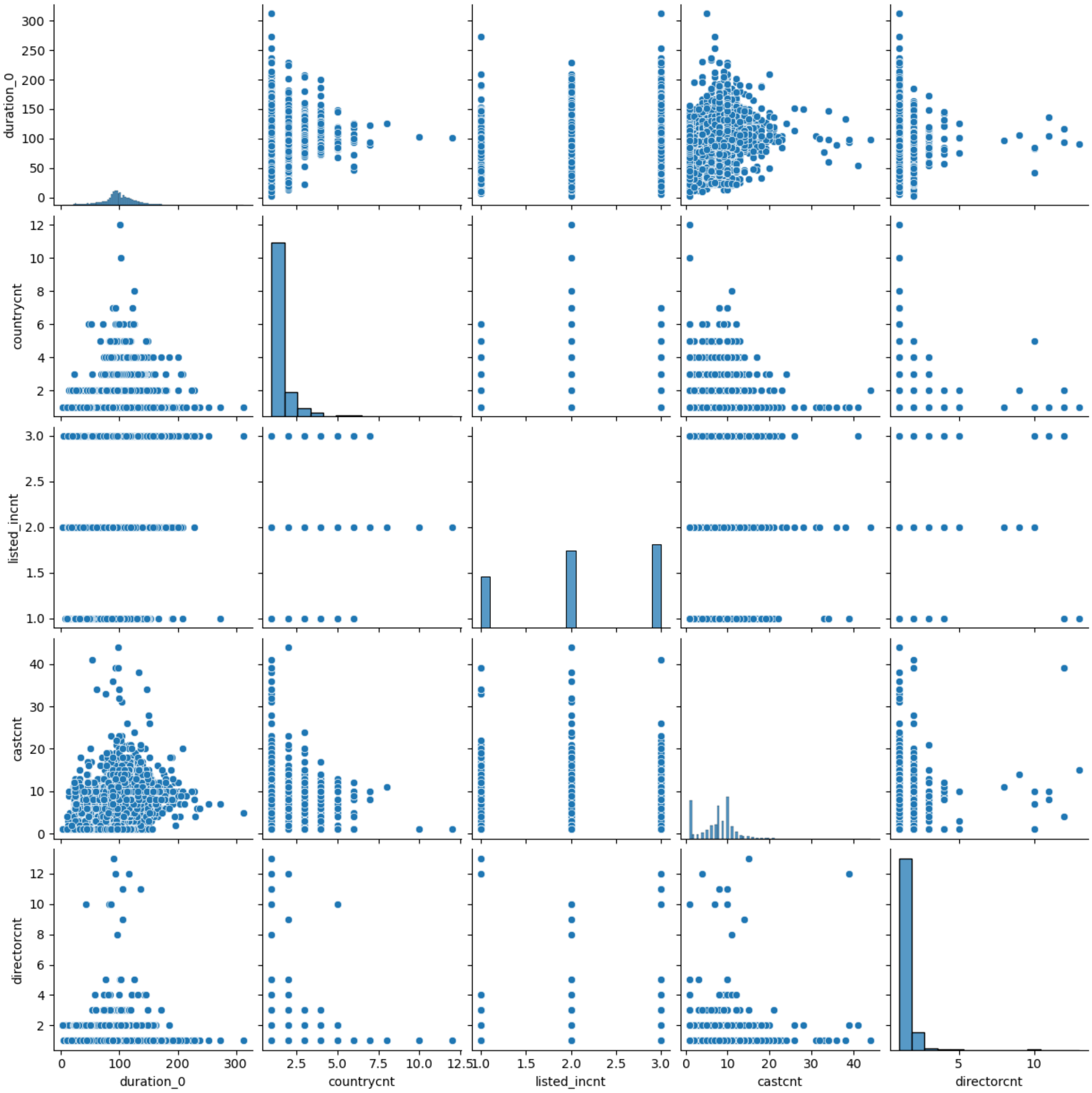
In [51]:

```
sns.pairplot(data = shows[["duration_0","countrycnt","listed_incnt","castcnt
plt.show()
```



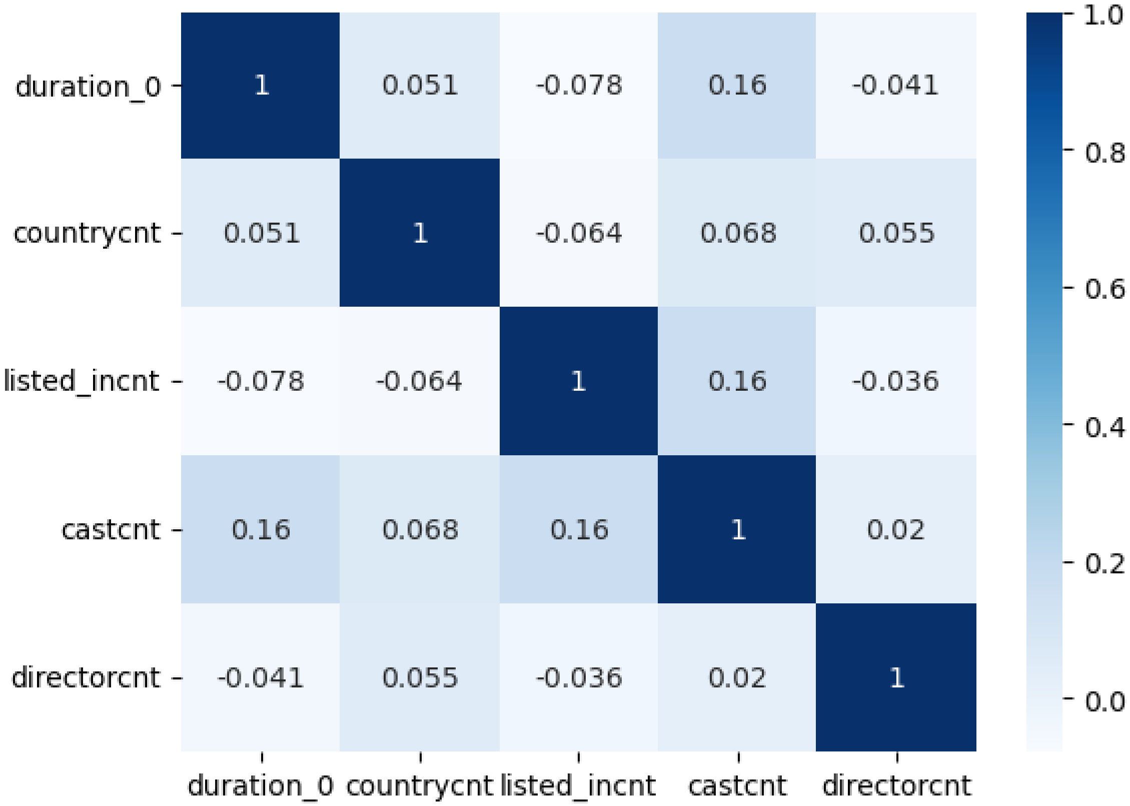
In [52]:

```
sns.pairplot(data = movies[["duration_0","countrycnt","listed_incnt","castcn
plt.show()

#when no of cast increases , there is duration of show increase.
```

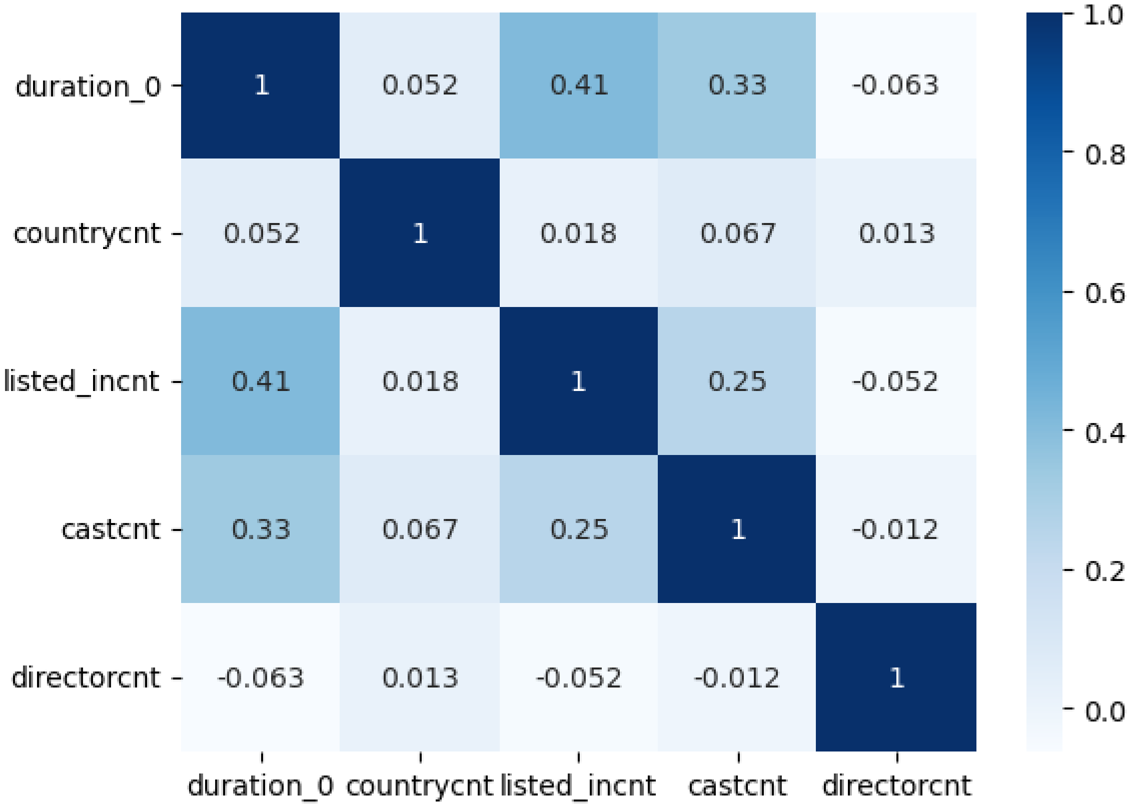


In [53]:

```
sns.heatmap(shows[["duration_0","countrycnt","listed_incnt","castcnt","direc
plt.show()
```



In [54]:
```
sns.heatmap(movies[["duration_0","countrycnt","listed_incnt","castcnt","dire
plt.show()
#this observation is for movies
#more listed_in , more duration
#more casts , more duration
```



In [55]:
```
#5.outlier check

show_quantiles=shows[["duration_0","countrycnt","listed_incnt","castcnt","di
```

```
movie_quantiles=movies[["duration_0","countrycnt","listed_incnt","castcnt","

show_IQR=show_quantiles.loc[0.75]-show_quantiles.loc[0.25]
movie_IQR=movie_quantiles.loc[0.75]-movie_quantiles.loc[0.25]
show_upper=show_quantiles.loc[0.75]+1.5*show_IQR
movie_upper=movie_quantiles.loc[0.75]+1.5*movie_IQR
```

In [56]:

```
show_upper
```

Out[56]:

```
duration_0        3.5
countrycnt        1.0
listed_incnt      4.5
castcnt          22.0
directorcnt       1.0
dtype: float64
```

In [57]:

```
for psudo,feature in zip(["duration_0","countrycnt","listed_incnt","castcnt"

    print("{} has {} outliers in shows type".format(feature,np.sum(shows[psu
```

```
duration has 259 outliers in shows type
country has 245 outliers in shows type
listed_in has 0 outliers in shows type
cast has 51 outliers in shows type
director has 42 outliers in shows type
```

In [58]:

```
for psudo,feature in zip(["duration_0","countrycnt","listed_incnt","castcnt"

    print("{} has {} outliers in movies type".format(feature,np.sum(movies[p
```

```
duration has 201 outliers in movies type
country has 1071 outliers in movies type
listed_in has 0 outliers in movies type
cast has 58 outliers in movies type
director has 572 outliers in movies type
```

In [59]:

```
raw.head()
```

Out[59]:

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

5 rows × 21 columns

In [60]:

```
#6. Insights based on Non-Graphical and Visual Analysis (10 Points)

#6.1 Comments on the range of attributes
#hear
print(raw[["release_year"]].describe())
print("mode: ",raw["release_year"].mode())

#released movies are in range from 1925 to 2021 years. it is left skewed . m
#show_id,type,title,rating,
#date_added
```

```
       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
mode:  0    2018
Name: release_year, dtype: int64
```

In [61]:

```
raw[["date_added"]].describe()

#movies are added into netflix from 2008-01-01 to 2021-09-25 .
```

```
#may be business is started from 2008.
#majority of movies are added on 2020-01-01
```

```
/var/folders/4q/h408ttzx43n2tyts5rkyb668ky0h13/T/ipykernel_26306/3079866777.
py:1: FutureWarning: Treating datetime data as categorical rather than numer
ic in `.describe` is deprecated and will be removed in a future version of p
andas. Specify `datetime_is_numeric=True` to silence this warning and adopt
the future behavior now.
  raw[["date_added"]].describe()
```

Out[61]:

	date_added
count	8797
unique	1714
top	2020-01-01 00:00:00
freq	110
first	2008-01-01 00:00:00
last	2021-09-25 00:00:00

In [62]:

```
print(raw["type"].describe())
print(raw["type"].value_counts())
print("missing values",np.sum(raw["type"].isna()))
# 2 types of shows. movie & TV show
# no missing values
#majority of shows are movies
```

```
count      8807
unique        2
top       Movie
freq       6131
Name: type, dtype: object
Movie      6131
TV Show    2676
Name: type, dtype: int64
missing values 0
```

In [63]:

```
print(raw["title"].describe())

print("missing values",np.sum(raw["title"].isna()))
## all values are unique. this says title of the show
# no missing values
```

```
count       8807
unique      8807
top       #Alive
freq           1
Name: title, dtype: object
missing values 0
```

In [64]:

```
print(raw["rating"].describe())
print(raw["rating"].value_counts())
print("missing values",np.sum(raw["rating"].isna()))
# 4 missing values. 3 wrong entries. 74,84,66 are in min and representing du

#majority of rarting is TV-MA and 3207 shows are present with this rating
```

```
count      8803
unique       17
top       TV-MA
freq       3207
Name: rating, dtype: object
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
missing values 4
```

In [65]:

```
print(raw["director"].describe())
print(raw["director"].value_counts())
print("missing values",np.sum(raw["director"].isna()))

#there are 2634 missing values
#somw shows have multiple directors.
```

```
count              6173
unique             4528
top       Rajiv Chilaka
freq                 19
Name: director, dtype: object
Rajiv Chilaka                     19
RaÃºl 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
missing values 2634
```

In [66]:

```
print(raw["cast"].describe())
print(raw["cast"].value_counts())
print("missing values",np.sum(raw["cast"].isna()))

#there are 825 missing values
#some shows have multiple casts.
```

```
count                      7982
unique                     7692
top         David Attenborough
freq                         19
Name: cast, dtype: object
David Attenborough
19
Vatsal Dubey, Julie Tejwani, Rupa Bhimani, Jigna Bhardwaj, Rajesh Kava, Mous
am, Swapnil
14
Samuel West
10
Jeff Dunham
7
David Spade, London Hughes, Fortune Feimster
6

..
Michael PeÃ±a, Diego Luna, Tenoch Huerta, Joaquin Cosio, JosÃ© MarÃa Yazpik,
Matt Letscher, Alyssa Diaz
1
Nick Lachey, Vanessa Lachey
1
Takeru Sato, Kasumi Arimura, Haru, Kentaro Sakaguchi, Takayuki Yamada, Kendo
Kobayashi, Ken Yasuda, Arata Furuta, Suzuki Matsuo, Koichi Yamadera, Arata I
ura, Chikako Kaku, Kotaro Yoshida     1
Toyin Abraham, Sambasa Nzeribe, Chioma Chukwuka Akpotha, Chioma Omeruah, Chi
wetalu Agu, Dele Odule, Femi Adebayo, Bayray McNwizu, Biodun Stephen
1
Vicky Kaushal, Sarah-Jane Dias, Raaghav Chanana, Manish Chaudhary, Meghna Ma
lik, Malkeet Rauni, Anita Shabdish, Chittaranjan Tripathy
1
Name: cast, Length: 7692, dtype: int64
missing values 825
```

In [67]:

```
print(raw["country"].describe())
print(raw["country"].value_counts())
print("missing values",np.sum(raw["country"].isna()))

#there are 831 missing values
#some shows have multiple countries.majority of movies are from united state
```

```
count                 7976
unique                 748
top         United States
freq                  2818
Name: country, dtype: object
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
missing values 831
```

In [68]:

```
print(raw["duration"].describe())
print(raw["duration"].value_counts())
```

```
#there are seasons & min.
#3 missing values and filled from rating column
```

```
count          8807
unique          220
top        1 Season
freq           1793
Name: duration, dtype: object
1 Season     1793
2 Seasons     425
3 Seasons     199
90 min        152
94 min        146
             ... 
16 min          1
186 min         1
193 min         1
189 min         1
191 min         1
Name: duration, Length: 220, dtype: int64
```

In [69]:

```
print(shows["duration_0"].describe())
print("mode",shows["duration_0"].mode())
#TV shows are from 1 season to 17 seasons
#most of the shows have only 1 season
#right skewed
```

```
count    2676.000000
mean        1.764948
std         1.582752
min         1.000000
25%         1.000000
50%         1.000000
75%         2.000000
max        17.000000
Name: duration_0, dtype: float64
mode 0    1
Name: duration_0, dtype: int64
```

In [70]:

```
print( movies["duration_0"].describe())
print("mode",movies["duration_0"].mode())
#movies are from 3 min to 312 min range
#most of the movies have  90 min length
#right skewed
#here mean==mode==median (approx)
```

```
count    6131.000000
mean       99.564998
std        28.289504
min         3.000000
25%        87.000000
50%        98.000000
75%       114.000000
max       312.000000
Name: duration_0, dtype: float64
mode 0    90
Name: duration_0, dtype: int64
```

In [71]:

```
print(raw["description"].describe())
#few shows description is repeated. but titles are not repeated.
```

```
count                                                   8807
unique                                                  8775
top       Paranormal activity at a lush, abandoned prope...
freq                                                       4
Name: description, dtype: object
```

In [72]:

```
dup_movies=raw[raw["description"].duplicated(keep=False)]
```

In [73]:

```
dup_movies.sort_values(["description"])

#same movie but dubbed into different language have same discription.
#can use this info to fill country missing values. can also fill any other f
```

Out[73]:

	show_id	type	title	director	cast	country	date
78	s79	Movie	Tughlaq Durbar	Delhiprasad Deenadayalan	Vijay Sethupathi, Parthiban, Raashi Khanna	NaN	202
79	s80	Movie	Tughlaq Durbar (Telugu)	Delhiprasad Deenadayalan	Vijay Sethupathi, Parthiban, Raashi Khanna	NaN	202
7022	s7023	Movie	Hum Saath-Saath Hain	Sooraj R. Barjatya	Salman Khan, Karisma Kapoor, Saif Ali Khan, Ta...	India	201
2969	s2970	Movie	Together For Eternity	Sooraj R. Barjatya	Salman Khan, Karisma Kapoor, Saif Ali Khan, Ta...	India	202
3492	s3493	Movie	Oh! Baby (Malayalam)	B. V. Nandini Reddy	Samantha Ruth Prabhu, Lakshmi, Rajendraprasad,...	NaN	2019
3516	s3517	Movie	Oh! Baby	B. V. Nandini Reddy	Samantha Ruth Prabhu, Lakshmi, Rajendraprasad,...	India	201
3493	s3494	Movie	Oh! Baby (Tamil)	B. V. Nandini Reddy	Samantha Ruth Prabhu, Lakshmi, Rajendraprasad,...	NaN	2019
8051	s8052	Movie	Solo: A Star Wars Story	Ron Howard	Alden Ehrenreich, Woody Harrelson, Emilia Clar...	United States	201
8052	s8053	Movie	Solo: A Star Wars Story (Spanish Version)	Ron Howard	Alden Ehrenreich, Woody Harrelson, Emilia Clar...	United States	201
5965	s5966	Movie	22-Jul	Paul Greengrass	Anders Danielsen Lie, Jon Ã˜igarden, Jonas Str...	Norway, Iceland, United States	201
4522	s4523	Movie	22 July	Paul Greengrass	Anders Danielsen Lie, Jon Ã˜igarden, Jonas Str...	Norway, Iceland, United States	201
2840	s2841	Movie	Angu Vaikuntapurathu (Malayalam)	Trivikram Srinivas	Allu Arjun, Pooja Hegde, Tabu, Sushanth, Nivet...	NaN	2020
2873	s2874	Movie	Ala Vaikunthapurramuloo	Trivikram Srinivas	Allu Arjun, Pooja Hegde, Tabu, Sushanth, Nivet...	India	202
3933	s3934	Movie	Petta (Telugu Version)	Karthik Subbaraj	Rajnikanth, Vijay Sethupathi, M. Sasikumar, Na...	NaN	2019

	show_id	type	title	director	cast	country	date
3939	s3940	Movie	Petta	Karthik Subbaraj	Rajnikanth, Vijay Sethupathi, M. Sasikumar, Na...	India	201
3932	s3933	Movie	Sarvam Thaala Mayam (Telugu Version)	Rajiv Menon	G.V. Prakash Kumar, Nedumudi Venu, Aparna Bala...	India	201
4061	s4062	Movie	Sarvam Thaala Mayam (Tamil Version)	Rajiv Menon	G.V. Prakash Kumar, Nedumudi Venu, Aparna Bala...	NaN	201
1652	s1653	Movie	Andhaghaaram	V Vignarajan	Vinoth Kishan, Arjun Das, Pooja Ramachandran, ...	NaN	202
1653	s1654	Movie	Andhakaaram	V Vignarajan	Vinoth Kishan, Arjun Das, Pooja Ramachandran, ...	India	202
3996	s3997	TV Show	February 9	NaN	Shahd El Yaseen, Shaila Sabt, Hala, Hanadi Al-...	NaN	201
5964	s5965	TV Show	Feb-09	NaN	Shahd El Yaseen, Shaila Sabt, Hala, Hanadi Al-...	NaN	201
3569	s3570	Movie	Game Over (Tamil Version)	Ashwin Saravanan	Taapsee Pannu, Vinodhini, Parvathi T, Ramya Su...	India, Turkey	201
3568	s3569	Movie	Game Over (Hindi Version)	Ashwin Saravanan	Taapsee Pannu, Vinodhini, Parvathi T, Ramya Su...	NaN	201
1986	s1987	Movie	Nee Enge En Anbe	Sekhar Kammula	Nayantara, Vaibhav Reddy, Pasupathy, Harshvard...	NaN	202
1982	s1983	Movie	Anaamika	Sekhar Kammula	Nayantara, Vaibhav Reddy, Pasupathy, Harshvard...	India	202
1270	s1271	TV Show	Sin senos sÃ hay paraÃso	NaN	Catherine Siachoque, FabiÃ¡n RÃos, Carolina G...	United States, Colombia	202

	show_id	type	title	director	cast	country	date
8022	s8023	TV Show	Sin Senos sÃ Hay ParaÃso	NaN	Majida Issa, FabiÃ¡n RÃos, Catherine Siachoqu...	United States, Colombia	201
852	s853	Movie	99 Songs (Telugu)	NaN	NaN	NaN	202
851	s852	Movie	99 Songs (Tamil)	NaN	NaN	NaN	202
850	s851	Movie	99 Songs	Vishwesh Krishnamoorthy	Ehan Bhat, Edilsy Vargas, Manisha Koirala, Lis...	India	202
303	s304	Movie	Esperando la carroza	Alejandro Doria	Luis Brandoni, China Zorrilla, Antonio Gasalla...	Argentina	202
6705	s6706	Movie	Esperando La Carroza	Alejandro Doria	Luis Brandoni, China Zorrilla, Antonio Gasalla...	Argentina	201
8173	s8174	TV Show	Thackeray	NaN	NaN	India	201
2336	s2337	Movie	Thackeray (Hindi)	Abhijit Panse	Nawazuddin Siddiqui, Amrita Rao, Rajeev Panday...	India	202
7559	s7560	Movie	Naruto Shippuden : Blood Prison	Masahiko Murata	Junko Takeuchi, Chie Nakamura, Rikiya Koyama, ...	Japan	201
56	s57	Movie	Naruto Shippuden the Movie: Blood Prison	Masahiko Murata	Junko Takeuchi, Chie Nakamura, Rikiya Koyama, ...	Japan	202
2335	s2336	Movie	Seven (Telugu)	NaN	NaN	NaN	202
2334	s2335	Movie	Seven (Tamil)	NaN	NaN	India	202
6024	s6025	TV Show	7 (Seven)	Nizar Shafi	Rahman, Havish, Regina Cassandra, Nandita Swet...	India	201
5966	s5967	Movie	15-Aug	Swapnaneel Jayakar	Rahul Pethe, Mrunmayee	India	201

	show_id	type	title	director	cast	country	date
					Deshpande, Adinath Koth...		
3962	s3963	Movie	15 August	Swapnaneel Jayakar	Rahul Pethe, Mrunmayee Deshpande, Adinath Koth...	India	2019
236	s237	Movie	Boomika	Rathindran R Prasad	Aishwarya Rajesh, Vidhu, Surya Ganapathy, Madh...	NaN	202
239	s240	Movie	Boomika (Telugu)	Rathindran R Prasad	Aishwarya Rajesh, Vidhu, Surya Ganapathy, Madh...	NaN	202
238	s239	Movie	Boomika (Malayalam)	Rathindran R Prasad	Aishwarya Rajesh, Vidhu, Surya Ganapathy, Madh...	NaN	202
237	s238	Movie	Boomika (Hindi)	Rathindran R Prasad	Aishwarya Rajesh, Vidhu, Surya Ganapathy, Madh...	NaN	202
6529	s6530	Movie	ConsequencesÂ	Ozan AÃ§Ä±ktan	Nehir ErdoÄŸan, Tardu Flordun, Ä°lker Kaleli, ...	Turkey	201
3371	s3372	Movie	Consequences	Ozan AÃ§Ä±ktan	Nehir ErdoÄŸan, Tardu Flordun, Ä°lker Kaleli, ...	Turkey	201
52	s53	Movie	InuYasha the Movie 3: Swords of an Honorable R...	Toshiya Shinohara	Kappei Yamaguchi, Satsuki Yukino, Koji Tsujita...	Japan	202
7089	s7090	Movie	Inuyasha the Movie - La spada del dominatore d...	Toshiya Shinohara	Kappei Yamaguchi, Satsuki Yukino, Koji Tsujita...	Japan	201
2664	s2665	TV Show	ChuChu TV Nursery Rhymes & Kids Songs (Hindi)	NaN	NaN	India	202
3393	s3394	TV Show	ChuChu TV Kids Songs, Learning Videos & Bedtim...	NaN	NaN	NaN	201
7649	s7650	TV Show	Ollie & Moon	NaN	Mattea Conforti, Kobi Frumer	France	2018

	show_id	type	title	director	cast	country	date
3028	s3029	TV Show	The Ollie & Moon Show	NaN	Mattea Conforti, Kobi Frumer	France	202
8360	s8361	Movie	The Incredibles 2	Brad Bird	Craig T. Nelson, Holly Hunter, Samuel L. Jacks...	United States	201
7067	s7068	Movie	Incredibles 2 (Spanish Version)	Brad Bird	VÃctor Trujillo, Consuelo Duval, DarÃo T. Pi...	United States	201
6449	s6450	Movie	Chashme Buddoor	David Dhawan	Ali Zafar, Siddharth, Divyendu Sharma, Tapsee ...	India	201
2270	s2271	Movie	Chashme Baddoor	David Dhawan	Rishi Kapoor, Ali Zafar, Taapsee Pannu, Siddha...	India	202
7090	s7091	Movie	InuYasha: The Movie 2: The Castle Beyond the L...	Toshiya Shinohara	Kappei Yamaguchi, Satsuki Yukino, Koji Tsujita...	Japan	201
51	s52	Movie	InuYasha the Movie 2: The Castle Beyond the Lo...	Toshiya Shinohara	Kappei Yamaguchi, Satsuki Yukino, Mieko Harada...	Japan	202

59 rows × 21 columns

In [74]:

```
#7. Business Insights (10 Points) - Should include patterns observed in the



#majority of directors did only 1 movie
#all movies are added on or after 2008 . maybe netflex started on that year.

#when no of cast increases , there is duration of show increase.  there is c
#missing country can be computed from directoer/casts. eg, find same cast sh
```

In [75]:

```
#Recommendations  - Actionable items for business. No technical jargon. No c



#most frequent directoers,top 1%. netflix can give more attention to these c
Rajiv Chilaka      22
Jan Suter          21
RaÃºl Campos       19
Suhas Kadav        16
Marcus Raboy       16



#most frequent directoers. netflix can give more attention to these director
#most frequent cast. netflix can give more attention to these cast especiall
#most frequent listed in is international movies. netflix can give more atte

#most movies comes from united states. netflix can give more attention to th
```

```

#netflix can concentrate more on getting movies
#netflix can concentrate more on getting shows having rating TV-MA
#netflix should focus more on latest movies
```

```
  File "/var/folders/4q/h408ttzx43n2tyts5rkyb668ky0h13/T/ipykernel_26306/161
9957570.py", line 6
    Rajiv Chilaka      22
          ^
SyntaxError: invalid syntax
```

In [76]:

```
#unique attribues analysis:




for column in [column for column in raw.columns if column not in ["show_id",
    print(raw[column].value_counts())

#no of movies >no of TV shows
#more shows are directed by Rajiv Chilaka
# more shows casted by. David Attenborough
# more movies came from United States followed by india
# more shows added on January 1, 2020
#more shows released on 2018 year
#mode of ratings TV-MA  . majority of shows have TV-MA rating
#majority of shows have only 1 Season
#more shows are listed in Dramas, International Movies
```

```
Movie      6131
TV Show    2676
Name: type, dtype: int64
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
2020-01-01    110
2019-11-01     91
2018-03-01     75
2019-12-31     74
2018-10-01     71
             ...
2017-02-21      1
2017-02-07      1
2017-01-29      1
2017-01-25      1
2020-01-11      1
Name: date_added, Length: 1714, dtype: int64
2018    1147
2017    1032
2019    1030
2020     953
2016     902
        ...
1959       1
1925       1
1961       1
1947       1
1966       1
Name: release_year, Length: 74, dtype: int64
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
1 Season     1793
2 Seasons     425
3 Seasons     199
90 min        152
94 min        146
             ...
16 min          1
```

```
186 min         1
193 min         1
189 min         1
191 min         1
Name: duration, Length: 220, dtype: int64
1.0     2212
15.0     687
2.0      325
16.0     289
31.0     274
20.0     249
19.0     243
5.0      231
22.0     230
10.0     214
30.0     210
6.0      210
18.0     207
26.0     206
8.0      201
14.0     198
25.0     197
27.0     195
7.0      194
21.0     193
28.0     190
23.0     184
12.0     181
17.0     180
4.0      175
13.0     175
24.0     159
3.0      151
11.0     149
9.0      147
29.0     141
Name: day_added, dtype: int64
7.0     827
12.0    813
9.0     770
4.0     764
10.0    760
8.0     755
3.0     742
1.0     738
6.0     728
11.0    705
5.0     632
2.0     563
Name: month_added, dtype: int64
2019.0    2016
2020.0    1879
2018.0    1649
2021.0    1498
2017.0    1188
2016.0     429
2015.0      82
2014.0      24
2011.0      13
2013.0      11
2012.0       3
2009.0       2
2008.0       2
2010.0       1
```

```
Name: year_added, dtype: int64
4.0    2498
3.0    1396
2.0    1288
1.0    1197
0.0     851
5.0     816
6.0     751
Name: weekday_added, dtype: int64
1     8193
2      542
3       37
4       15
5        7
10       3
9        2
7        2
11       2
12       2
13       1
8        1
Name: directorcnt, dtype: int64
1     1706
10    1442
8     1179
9      688
6      620
7      617
11     474
5      370
12     307
4      272
2      201
3      193
13     186
14     125
15     120
16      72
17      47
18      37
19      28
20      26
21      16
22      11
25       9
23       9
24       7
28       6
26       6
27       4
33       3
47       3
34       3
31       2
40       2
39       2
50       2
38       2
41       2
30       2
44       1
32       1
36       1
42       1
```

```
35       1
29       1
Name: castcnt, dtype: int64
3    3729
2    3058
1    2020
Name: listed_incnt, dtype: int64
1     7491
2      869
3      273
4      115
5       36
6       14
7        5
8        2
12       1
10       1
Name: countrycnt, dtype: int64
1      1793
2       425
3       200
90      152
94      146
       ... 
189       1
273       1
212       1
224       1
191       1
Name: duration_0, Length: 210, dtype: int64
```

In [ ]: