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

df=pd.read_csv("netflix.csv")

df.head()
```

```
  show_id     type                   title         director  \
0      s1    Movie   Dick Johnson Is Dead  Kirsten Johnson
1      s2  TV Show          Blood & Water              NaN
2      s3  TV Show              Ganglands  Julien Leclercq
3      s4  TV Show  Jailbirds New Orleans              NaN
4      s5  TV Show           Kota Factory              NaN

                                                cast        country  \
0                                                NaN  United States
1  Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...   South Africa
2  Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...            NaN
3                                                NaN            NaN
4  Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...          India

           date_added  release_year rating   duration  \
0  September 25, 2021          2020  PG-13     90 min
1  September 24, 2021          2021  TV-MA  2 Seasons
2  September 24, 2021          2021  TV-MA   1 Season
3  September 24, 2021          2021  TV-MA   1 Season
4  September 24, 2021          2021  TV-MA  2 Seasons

                                          listed_in  \
0                                    Documentaries
1    International TV Shows, TV Dramas, TV Mysteries
2  Crime TV Shows, International TV Shows, TV Act...
3                            Docuseries, Reality TV
4  International TV Shows, Romantic TV Shows, TV ...

                                        description
0  As her father nears the end of his life, filmm...
1  After crossing paths at a party, a Cape Town t...
2  To protect his family from a powerful drug lor...
3  Feuds, flirtations and toilet talk go down amo...
4  In a city of coaching centers known to train I...
```

```
df.shape
```

```
(8807, 12)
```

```
#Number of rows
df.shape[0]
```

```
8807
```

```
#number of columns
df.shape[1]
```

```
12
```

```
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   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
```

```
#data type of each column
df.dtypes
```

```
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
```

```
#Descriptive statistics for the numerical columns
df.describe()
```

```
       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
```

```
#Finding number of unique values in each column to understand the
cardinality(number of distinct values in each column.
df.nunique()
```

```
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
```

```
df.isnull()
```

```
      show_id   type  title  director   cast  country  date_added  \
0       False  False  False     False   True    False       False
1       False  False  False      True  False    False       False
2       False  False  False     False  False     True       False
3       False  False  False      True   True     True       False
4       False  False  False      True  False    False       False
...       ...    ...    ...       ...    ...      ...         ...
8802    False  False  False     False  False    False       False
8803    False  False  False      True   True     True       False
8804    False  False  False     False  False    False       False
8805    False  False  False     False  False    False       False
8806    False  False  False     False  False    False       False

      release_year  rating  duration  listed_in  description
0            False   False     False      False        False
1            False   False     False      False        False
2            False   False     False      False        False
3            False   False     False      False        False
4            False   False     False      False        False
...            ...     ...       ...        ...          ...
8802         False   False     False      False        False
8803         False   False     False      False        False
```

```
8804         False   False     False      False        False
8805         False   False     False      False        False
8806         False   False     False      False        False

[8807 rows x 12 columns]
```

```
df.isnull().sum()
```

```
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
```

```
#finding the precentage of null values in each columns
round(df.isnull().sum()/len(df)*100,2).sort_values(ascending=False)
```

```
director        29.91
country          9.44
cast             9.37
date_added       0.11
rating           0.05
duration         0.03
show_id          0.00
type             0.00
title            0.00
release_year     0.00
listed_in        0.00
description      0.00
dtype: float64
```

```
# Splitting the 'director' column into multiple rows
dir_constraint = df['director'].apply(lambda x: str(x).split(',
')).tolist()
df1 = pd.DataFrame(dir_constraint, index=df['title'])
df1 = df1.stack().reset_index(level=1, drop=True)  # Drop the level_1
index
df1 = pd.DataFrame(df1, columns=['Directors'])  # Rename the column
df1.reset_index(inplace=True)  # Reset index to make 'title' a column
again
```

```
df1.head(20)
```

```
                                                 title
Directors
0                                Dick Johnson Is Dead         Kirsten
Johnson
1                                       Blood & Water
nan
2                                           Ganglands         Julien
Leclercq
3                                Jailbirds New Orleans
nan
4                                        Kota Factory
nan
5                                       Midnight Mass           Mike
Flanagan
6                  My Little Pony: A New Generation           Robert
Cullen
7                  My Little Pony: A New Generation          José
Luis Ucha
8                                           Sankofa            Haile
Gerima
9                     The Great British Baking Show         Andy
Devonshire
10                                       The Starling
Theodore Melfi
11               Vendetta: Truth, Lies and The Mafia
nan
12                                   Bangkok Breaking       Kongkiat
Komesiri
13                                       Je Suis Karl     Christian
Schwochow
14                 Confessions of an Invisible Girl           Bruno
Garotti
15                  Crime Stories: India Detectives
nan
16                                  Dear White People
nan
17  Europe's Most Dangerous Man: Otto Skorzeny in ...  Pedro de Echave
García
18  Europe's Most Dangerous Man: Otto Skorzeny in ...   Pablo Azorín
Williams
19                                    Falsa identidad
nan

# Splitting the 'cast' column into multiple rows
cast_constraint = df['cast'].apply(lambda x: str(x).split(',
')).tolist()
df2 = pd.DataFrame(cast_constraint, index=df['title'])
df2 = df2.stack().reset_index(level=1, drop=True)  # Drop the level_1
index
df2 = pd.DataFrame(df2, columns=['Actors'])  # Rename the column
```

```
df2.reset_index(inplace=True)  # Reset index to make 'title' a column
again

df2.head(20)
```

	title	Actors
0	Dick Johnson Is Dead	nan
1	Blood & Water	Ama Qamata
2	Blood & Water	Khosi Ngema
3	Blood & Water	Gail Mabalane
4	Blood & Water	Thabang Molaba
5	Blood & Water	Dillon Windvogel
6	Blood & Water	Natasha Thahane
7	Blood & Water	Arno Greeff
8	Blood & Water	Xolile Tshabalala
9	Blood & Water	Getmore Sithole
10	Blood & Water	Cindy Mahlangu
11	Blood & Water	Ryle De Morny
12	Blood & Water	Greteli Fincham
13	Blood & Water	Sello Maake Ka-Ncube
14	Blood & Water	Odwa Gwanya
15	Blood & Water	Mekaila Mathys
16	Blood & Water	Sandi Schultz
17	Blood & Water	Duane Williams
18	Blood & Water	Shamilla Miller
19	Blood & Water	Patrick Mofokeng

```
# Splitting the 'listed_in' column into multiple rows
listed_constraint = df['listed_in'].apply(lambda x: str(x).split(',
')).tolist()
df3 = pd.DataFrame(listed_constraint, index=df['title'])
df3 = df3.stack().reset_index(level=1, drop=True)  # Drop the level_1
index
df3 = pd.DataFrame(df3, columns=['Genre'])  # Rename the column
df3.reset_index(inplace=True)  # Reset index to make 'title' a column
again

df3.head(20)
```

	title	Genre
0	Dick Johnson Is Dead	Documentaries
1	Blood & Water	International TV Shows
2	Blood & Water	TV Dramas
3	Blood & Water	TV Mysteries
4	Ganglands	Crime TV Shows
5	Ganglands	International TV Shows
6	Ganglands	TV Action & Adventure
7	Jailbirds New Orleans	Docuseries
8	Jailbirds New Orleans	Reality TV
9	Kota Factory	International TV Shows

```
10                        Kota Factory         Romantic TV Shows
11                        Kota Factory               TV Comedies
12                       Midnight Mass                 TV Dramas
13                       Midnight Mass                 TV Horror
14                       Midnight Mass              TV Mysteries
15  My Little Pony: A New Generation  Children & Family Movies
16                             Sankofa                   Dramas
17                             Sankofa        Independent Movies
18                             Sankofa      International Movies
19     The Great British Baking Show           British TV Shows
```

```
# Splitting the 'country' column into multiple rows
country_constraint = df['country'].apply(lambda x: str(x).split(',
')).tolist()
df4 = pd.DataFrame(country_constraint, index=df['title'])
df4 = df4.stack().reset_index(level=1, drop=True)  # Drop the level_1
index
df4 = pd.DataFrame(df4, columns=['Country'])  # Rename the column
df4.reset_index(inplace=True)  # Reset index to make 'title' a column
again

df4.head(20)
```

```
                                    title         Country
0                   Dick Johnson Is Dead   United States
1                          Blood & Water    South Africa
2                              Ganglands             nan
3                  Jailbirds New Orleans             nan
4                           Kota Factory           India
5                          Midnight Mass             nan
6      My Little Pony: A New Generation             nan
7                                Sankofa   United States
8                                Sankofa           Ghana
9                                Sankofa    Burkina Faso
10                               Sankofa  United Kingdom
11                               Sankofa         Germany
12                               Sankofa        Ethiopia
13        The Great British Baking Show  United Kingdom
14                           The Starling   United States
15  Vendetta: Truth, Lies and The Mafia             nan
16                       Bangkok Breaking             nan
17                           Je Suis Karl         Germany
18                           Je Suis Karl  Czech Republic
19     Confessions of an Invisible Girl             nan
```

```
# Merging df2 and df1 on 'title' column
df5 = df2.merge(df1, on=['title'], how='inner')

# Merging df5 and df3 on 'title' column
df6 = df5.merge(df3, on=['title'], how='inner')
```

```
# Merging df6 and df4 on 'title' column
df7 = df6.merge(df4, on=['title'], how='inner')

# Display the first few rows of the merged dataframe
df7.head()
```

```
                  title       Actors        Directors
Genre  \
0  Dick Johnson Is Dead          nan  Kirsten Johnson
Documentaries
1         Blood & Water   Ama Qamata              nan  International
TV Shows
2         Blood & Water   Ama Qamata              nan                TV
Dramas
3         Blood & Water   Ama Qamata              nan             TV
Mysteries
4         Blood & Water  Khosi Ngema              nan  International
TV Shows

         Country
0  United States
1   South Africa
2   South Africa
3   South Africa
4   South Africa
```

```
df7.shape
```

```
(201991, 5)
```

```
# Merging df7 with the original dataframe df on the 'title' column
df = df7.merge(df[['show_id', 'type', 'title', 'date_added',
'release_year', 'rating', 'duration']],
               on=['title'],
               how='left')

# Display the first few rows of the merged dataframe
df.head()
```

```
                  title       Actors        Directors
Genre  \
0  Dick Johnson Is Dead          nan  Kirsten Johnson
Documentaries
1         Blood & Water   Ama Qamata              nan  International
TV Shows
2         Blood & Water   Ama Qamata              nan                TV
Dramas
3         Blood & Water   Ama Qamata              nan             TV
Mysteries
4         Blood & Water  Khosi Ngema              nan  International
```

```
TV Shows

         Country show_id     type          date_added  release_year
rating  \
0  United States      s1    Movie  September 25, 2021          2020
PG-13
1   South Africa      s2  TV Show  September 24, 2021          2021
TV-MA
2   South Africa      s2  TV Show  September 24, 2021          2021
TV-MA
3   South Africa      s2  TV Show  September 24, 2021          2021
TV-MA
4   South Africa      s2  TV Show  September 24, 2021          2021
TV-MA

    duration
0     90 min
1  2 Seasons
2  2 Seasons
3  2 Seasons
4  2 Seasons

df.shape

(201991, 11)

df.isnull().sum()

title              0
Actors             0
Directors          0
Genre              0
Country            0
show_id            0
type               0
date_added       158
release_year       0
rating            67
duration           3
dtype: int64

# Calculate the total number of missing values for each column
total_null = df.isnull().sum().sort_values(ascending=False)

# Calculate the percentage of missing values for each column
percent = ((df.isnull().sum() / df.isnull().count()) *
100).sort_values(ascending=False)

# Print the total number of records in the DataFrame
print("Total records = ", df.shape[0])
```

```
# Concatenate the total null counts and the percentage of missing
values into a single DataFrame
missing_data = pd.concat([total_null, percent.round(2)], axis=1,
keys=['Total Missing', 'In Percent'])

# Display the first 10 rows of the missing_data DataFrame
missing_data.head(10)
```

```
Total records =  201991
```

	Total Missing	In Percent
date_added	158	0.08
rating	67	0.03
duration	3	0.00
title	0	0.00
Actors	0	0.00
Directors	0	0.00
Genre	0	0.00
Country	0	0.00
show_id	0	0.00
type	0	0.00

```
#Above table gives missing values summary in absolute value and in
Percentage, date added has the maximum missing values

# Replace missing values in the 'Actors' column with 'Unknown Actor'
df['Actors'].replace('nan', 'Unknown Actor', inplace=True)

# Replace missing values in the 'Directors' column with 'Unknown
Director'
df['Directors'].replace('nan', 'Unknown Director', inplace=True)

# Replace missing values in the 'Country' column with NaN
df['Country'].replace('nan', np.nan, inplace=True)

# Display the first few rows of the DataFrame
df.head()
```

	title	Actors	Directors	Genre	Country	show_id	type	date_added
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States	s1	Movie	September 25, 2021
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021
2	Blood & Water	Ama Qamata	Unknown Director					
3	Blood & Water	Ama Qamata	Unknown Director					
4	Blood & Water	Khosi Ngema	Unknown Director					

```
2               TV Dramas   South Africa      s2  TV Show  September
24, 2021
3            TV Mysteries   South Africa      s2  TV Show  September
24, 2021
4  International TV Shows   South Africa      s2  TV Show  September
24, 2021

   release_year rating   duration
0          2020  PG-13     90 min
1          2021  TV-MA  2 Seasons
2          2021  TV-MA  2 Seasons
3          2021  TV-MA  2 Seasons
4          2021  TV-MA  2 Seasons
```

```
# Calculate the total number of missing values for each column
total_null = df.isnull().sum().sort_values(ascending=False)

# Calculate the percentage of missing values for each column
percent = ((df.isnull().sum() / df.isnull().count()) *
100).sort_values(ascending=False)

# Print the total number of records in the DataFrame
print("Total records = ", df.shape[0])

# Concatenate the total null counts and the percentage of missing
values into a single DataFrame
missing_data = pd.concat([total_null, percent.round(2)], axis=1,
keys=['Total Missing', 'In Percent'])

# Display the first 10 rows of the missing_data DataFrame
missing_data.head(10)
```

```
Total records =  201991

             Total Missing  In Percent
Country              11897        5.89
date_added             158        0.08
rating                  67        0.03
duration                 3        0.00
title                    0        0.00
Actors                   0        0.00
Directors                0        0.00
Genre                    0        0.00
show_id                  0        0.00
type                     0        0.00
```

```
df[df.duration.isnull()]
```

```
                                       title      Actors   Directors
Genre  \
126537                      Louis C.K. 2017  Louis C.K.  Louis C.K.
```

```
Movies
131603                 Louis C.K.: Hilarious  Louis C.K.  Louis C.K.
Movies
131737  Louis C.K.: Live at the Comedy Store  Louis C.K.  Louis C.K.
Movies

               Country show_id   type          date_added  release_year
\
126537  United States   s5542  Movie       April 4, 2017          2017

131603  United States   s5795  Movie  September 16, 2016          2010

131737  United States   s5814  Movie     August 15, 2016          2015


        rating duration
126537  74 min      NaN
131603  84 min      NaN
131737  66 min      NaN
```

duration and rating columns got messed up and values got exchanged will add rating column values into duration column missing values

```
# Replace 'duration' column missing values with 'rating' column values
where 'duration' is missing
df.loc[df['duration'].isnull(), 'duration'] =
df.loc[df['duration'].isnull(), 'rating']

# Update 'rating' column values where the original values were in the
format of minutes to 'NR'
df.loc[df['rating'].str.contains('min', na=False), 'rating'] = 'NR'

# Fill any remaining missing values in the 'rating' column with 'NR'
df['rating'].fillna('NR', inplace=True)

# Check for missing values in the DataFrame after the adjustments
df.isnull().sum()
```

```
title              0
Actors             0
Directors          0
Genre              0
Country        11897
show_id            0
type               0
date_added       158
release_year       0
rating             0
```

```
duration             0
dtype: int64
```

Filling missing values of date added column with mode value with respective release years

```
for i in df[df['date_added'].isnull()]['release_year'].unique():
    # Calculate the mode value of 'date_added' for the current
'release_year'
    date = df[df['release_year'] == i]['date_added'].mode().values[0]

    # Fill missing values in 'date_added' for the current
'release_year' with the mode value
    df.loc[df['release_year'] == i, 'date_added'] =
df.loc[df['release_year'] == i, 'date_added'].fillna(date)

df[df.Country.isna()]
```

```
               title         Actors         Directors
Genre  \
58        Ganglands  Sami Bouajila   Julien Leclercq          Crime TV
Shows
59        Ganglands  Sami Bouajila   Julien Leclercq  International TV
Shows
60        Ganglands  Sami Bouajila   Julien Leclercq   TV Action &
Adventure
61        Ganglands   Tracy Gotoas   Julien Leclercq          Crime TV
Shows
62        Ganglands   Tracy Gotoas   Julien Leclercq  International TV
Shows
...             ...            ...               ...
...
201424          YOM     Mayur Vyas  Unknown Director
Kids' TV
201425          YOM     Ketan Kava  Unknown Director
Kids' TV
201932  Zombie Dumb  Unknown Actor  Unknown Director
Kids' TV
201933  Zombie Dumb  Unknown Actor  Unknown Director         Korean TV
Shows
201934  Zombie Dumb  Unknown Actor  Unknown Director             TV
Comedies

       Country show_id     type          date_added  release_year
rating  \
58         NaN      s3  TV Show  September 24, 2021          2021  TV-
MA
59         NaN      s3  TV Show  September 24, 2021          2021  TV-
MA
```

```
60          NaN      s3  TV Show  September 24, 2021          2021  TV-
MA
61          NaN      s3  TV Show  September 24, 2021          2021  TV-
MA
62          NaN      s3  TV Show  September 24, 2021          2021  TV-
MA
...         ...     ...      ...                 ...           ...    .
..
201424      NaN   s8786  TV Show        June 7, 2018          2016  TV-
Y7
201425      NaN   s8786  TV Show        June 7, 2018          2016  TV-
Y7
201932      NaN   s8804  TV Show        July 1, 2019          2018  TV-
Y7
201933      NaN   s8804  TV Show        July 1, 2019          2018  TV-
Y7
201934      NaN   s8804  TV Show        July 1, 2019          2018  TV-
Y7

          duration
58        1 Season
59        1 Season
60        1 Season
61        1 Season
62        1 Season
...            ...
201424    1 Season
201425    1 Season
201932   2 Seasons
201933   2 Seasons
201934   2 Seasons

[11897 rows x 11 columns]
```

Filling missing values of country column with mode value with respective directors

```
for i in df[df['Country'].isnull()]['Directors'].unique():
    # Check if the director has non-null country values
    if i in df[~df['Country'].isnull()]['Directors'].unique():
        # Calculate the mode value of 'Country' for the current
'Directors'
        country = df[df['Directors'] == i]['Country'].mode().values[0]

        # Fill missing values in 'Country' for the current 'Directors'
with the mode value
        df.loc[df['Directors'] == i, 'Country'] =
df.loc[df['Directors'] == i, 'Country'].fillna(country)

df.isnull().sum()
```

```
title              0
Actors             0
Directors          0
Genre              0
Country         4276
show_id            0
type               0
date_added         0
release_year       0
rating             0
duration           0
dtype: int64
```

```
#remaing missing values will be replaced using actors column

for i in df[df['Country'].isnull()]['Actors'].unique():
    # Check if the actor has non-null country values
    if i in df[~df['Country'].isnull()]['Actors'].unique():
        # Calculate the mode value of 'Country' for the current
'Actors'
        imp = df[df['Actors'] == i]['Country'].mode().values[0]

        # Fill missing values in 'Country' for the current 'Actors'
with the mode value
        df.loc[df['Actors'] == i, 'Country'] = df.loc[df['Actors'] ==
i, 'Country'].fillna(imp)

df['Country'].fillna('Unknown Country',inplace=True)
df.isnull().sum()
```

```
title           0
Actors          0
Directors       0
Genre           0
Country         0
show_id         0
type            0
date_added      0
release_year    0
rating          0
duration        0
duration2       0
dtype: int64
```

Now missing values handling is over, will deep dive into data analysis

```
#converting date added data type into datetime format to extract
years, month

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

```
df ['duration'] = df['duration'].str.replace(" min","")
df.head(6)
```

```
                  title         Actors         Directors  \
0  Dick Johnson Is Dead  Unknown Actor   Kirsten Johnson
1         Blood & Water     Ama Qamata  Unknown Director
2         Blood & Water     Ama Qamata  Unknown Director
3         Blood & Water     Ama Qamata  Unknown Director
4         Blood & Water    Khosi Ngema  Unknown Director
5         Blood & Water    Khosi Ngema  Unknown Director

                     Genre        Country show_id     type
date_added  \
0           Documentaries  United States      s1    Movie 2021-09-25

1  International TV Shows   South Africa      s2  TV Show 2021-09-24

2               TV Dramas   South Africa      s2  TV Show 2021-09-24

3             TV Mysteries   South Africa      s2  TV Show 2021-09-24

4  International TV Shows   South Africa      s2  TV Show 2021-09-24

5               TV Dramas   South Africa      s2  TV Show 2021-09-24


   release_year rating   duration  duration2
0          2020  PG-13         90         90
1          2021  TV-MA  2 Seasons  2 Seasons
2          2021  TV-MA  2 Seasons  2 Seasons
3          2021  TV-MA  2 Seasons  2 Seasons
4          2021  TV-MA  2 Seasons  2 Seasons
5          2021  TV-MA  2 Seasons  2 Seasons
```

```
df['duration2'] = df.duration.copy()
df_ = df.copy()

df_.loc[df_['duration2'].str.contains('Season'),'duration2'] = 0
df_['duration2'] = df_.duration2.astype('int')
df_.head()
```

```
                  title         Actors         Directors  \
0  Dick Johnson Is Dead  Unknown Actor   Kirsten Johnson
1         Blood & Water     Ama Qamata  Unknown Director
2         Blood & Water     Ama Qamata  Unknown Director
3         Blood & Water     Ama Qamata  Unknown Director
4         Blood & Water    Khosi Ngema  Unknown Director

                     Genre        Country show_id     type
date_added  \
0           Documentaries  United States      s1    Movie 2021-09-25
```

```
1  International TV Shows   South Africa      s2  TV Show 2021-09-24

2               TV Dramas   South Africa      s2  TV Show 2021-09-24

3            TV Mysteries   South Africa      s2  TV Show 2021-09-24

4  International TV Shows   South Africa      s2  TV Show 2021-09-24
```

```
   release_year rating   duration  duration2
0          2020  PG-13         90         90
1          2021  TV-MA  2 Seasons          0
2          2021  TV-MA  2 Seasons          0
3          2021  TV-MA  2 Seasons          0
4          2021  TV-MA  2 Seasons          0
```

```
df_.duration2.describe()
```

```
count    201991.000000
mean         77.152789
std          52.269154
min           0.000000
25%           0.000000
50%          95.000000
75%         112.000000
max         312.000000
Name: duration2, dtype: float64
```

```
df_.T.apply(lambda x: x.nunique(), axis=1)
```

```
title            8807
Actors          36440
Directors        4994
Genre              42
Country           128
show_id          8807
type                2
date_added       1714
release_year       74
rating             14
duration          220
duration2         206
dtype: int64
```

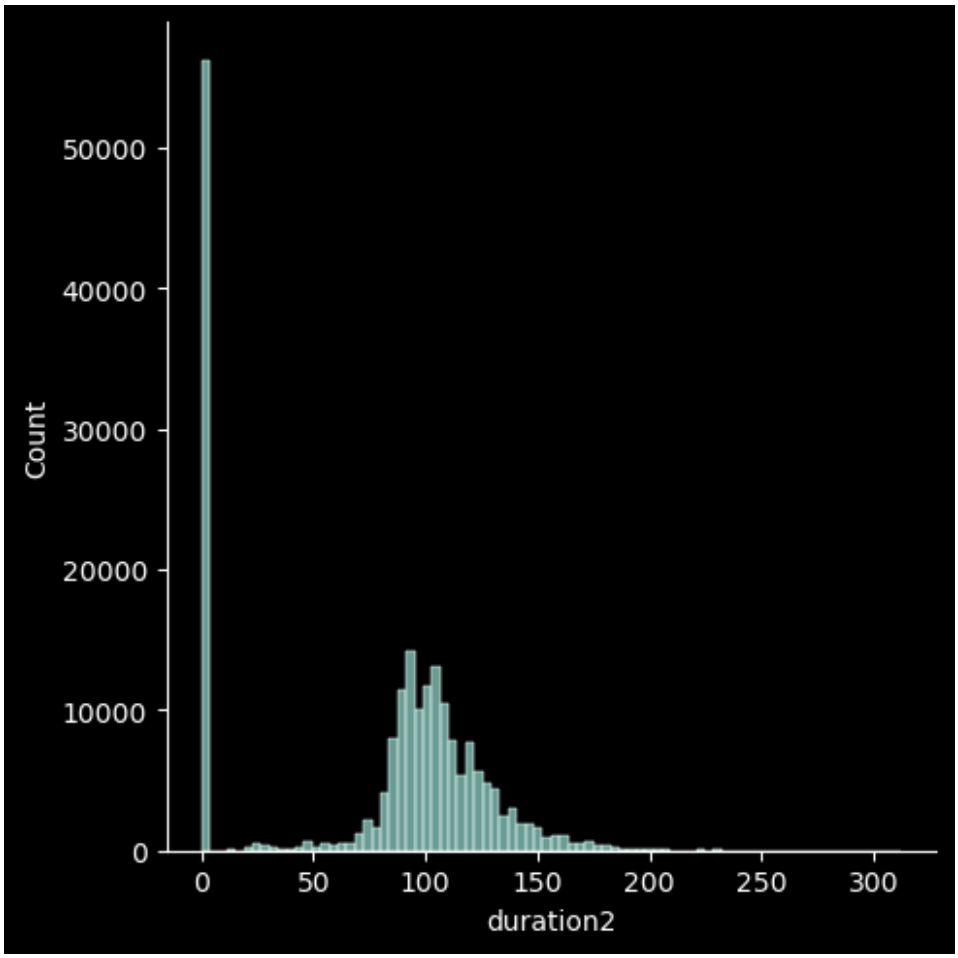
Actors has the most unique values follwed by title and directors

```
#Univariate analysis of duration column

## Histogram to see the distribution of duration
```

```
plt.style.use('dark_background')
plt.figure(figsize=(10,2))
sns.displot(df_['duration2'])

plt.show()
```

```
C:\Users\vinut\anaconda3\Lib\site-packages\seaborn\axisgrid.py:118:
UserWarning: The figure layout has changed to tight
  self._figure.tight_layout(*args, **kwargs)
```

```
<Figure size 1000x200 with 0 Axes>
```



Most of the values is around 100 and basically 0 is the TV shows

```
#Will convert them into bins, for easy visulaization

bins = [-1,1,50,80,100,120,150,200,315]
labels = ['<1','1-50','50-80','80-100','100-120','120-150','150-
200','200-315']
df_['duration2'] = pd.cut(df_['duration2'],bins = bins, labels =
```

```
labels )
df_.head()
```

```
                  title         Actors         Directors  \
0  Dick Johnson Is Dead  Unknown Actor   Kirsten Johnson
1         Blood & Water     Ama Qamata  Unknown Director
2         Blood & Water     Ama Qamata  Unknown Director
3         Blood & Water     Ama Qamata  Unknown Director
4         Blood & Water    Khosi Ngema  Unknown Director

                     Genre        Country show_id     type
date_added  \
0           Documentaries  United States      s1    Movie 2021-09-25

1  International TV Shows   South Africa      s2  TV Show 2021-09-24

2               TV Dramas   South Africa      s2  TV Show 2021-09-24

3            TV Mysteries   South Africa      s2  TV Show 2021-09-24

4  International TV Shows   South Africa      s2  TV Show 2021-09-24


   release_year rating   duration duration2
0          2020  PG-13         90    80-100
1          2021  TV-MA  2 Seasons        <1
2          2021  TV-MA  2 Seasons        <1
3          2021  TV-MA  2 Seasons        <1
4          2021  TV-MA  2 Seasons        <1
```

```
df_.loc[~df_['duration'].str.contains('Season'),'duration'] =
df_.loc[~df_['duration'].str.contains('Season'),'duration2']
df_.drop(['duration2'],axis=1,inplace=True)
df_.head()
```

```
                  title         Actors         Directors  \
0  Dick Johnson Is Dead  Unknown Actor   Kirsten Johnson
1         Blood & Water     Ama Qamata  Unknown Director
2         Blood & Water     Ama Qamata  Unknown Director
3         Blood & Water     Ama Qamata  Unknown Director
4         Blood & Water    Khosi Ngema  Unknown Director

                     Genre        Country show_id     type
date_added  \
0           Documentaries  United States      s1    Movie 2021-09-25

1  International TV Shows   South Africa      s2  TV Show 2021-09-24

2               TV Dramas   South Africa      s2  TV Show 2021-09-24

3            TV Mysteries   South Africa      s2  TV Show 2021-09-24
```

```
4  International TV Shows   South Africa      s2  TV Show 2021-09-24


   release_year rating   duration
0          2020  PG-13     80-100
1          2021  TV-MA  2 Seasons
2          2021  TV-MA  2 Seasons
3          2021  TV-MA  2 Seasons
4          2021  TV-MA  2 Seasons
```

extracting day, week, year, month from date added column helps in checking which month got more TV shows like that

```
from datetime import datetime
from dateutil.parser import parse
df_["year_added"] = df_['date_added'].dt.year
df_["year_added"] = df_["year_added"].astype("Int64")
df_["month_added"] = df_['date_added'].dt.month
df_['month_name'] = df['date_added'].dt.month_name()
df_["month_added"] = df_["month_added"].astype("Int64")
df_["day_added"] = df_['date_added'].dt.day
df_["day_added"] = df_["day_added"].astype("Int64")
df_['Weekday_added'] = df_['date_added'].apply(lambda x:
parse(str(x)).strftime("%A"))
df_.head()
```

```
                  title         Actors         Directors  \
0  Dick Johnson Is Dead  Unknown Actor   Kirsten Johnson
1         Blood & Water     Ama Qamata  Unknown Director
2         Blood & Water     Ama Qamata  Unknown Director
3         Blood & Water     Ama Qamata  Unknown Director
4         Blood & Water    Khosi Ngema  Unknown Director

                     Genre        Country show_id     type
date_added  \
0           Documentaries  United States      s1    Movie 2021-09-25

1  International TV Shows   South Africa      s2  TV Show 2021-09-24

2               TV Dramas   South Africa      s2  TV Show 2021-09-24

3            TV Mysteries   South Africa      s2  TV Show 2021-09-24

4  International TV Shows   South Africa      s2  TV Show 2021-09-24


   release_year rating   duration  year_added  month_added month_name
\
0          2020  PG-13     80-100        2021            9  September
```

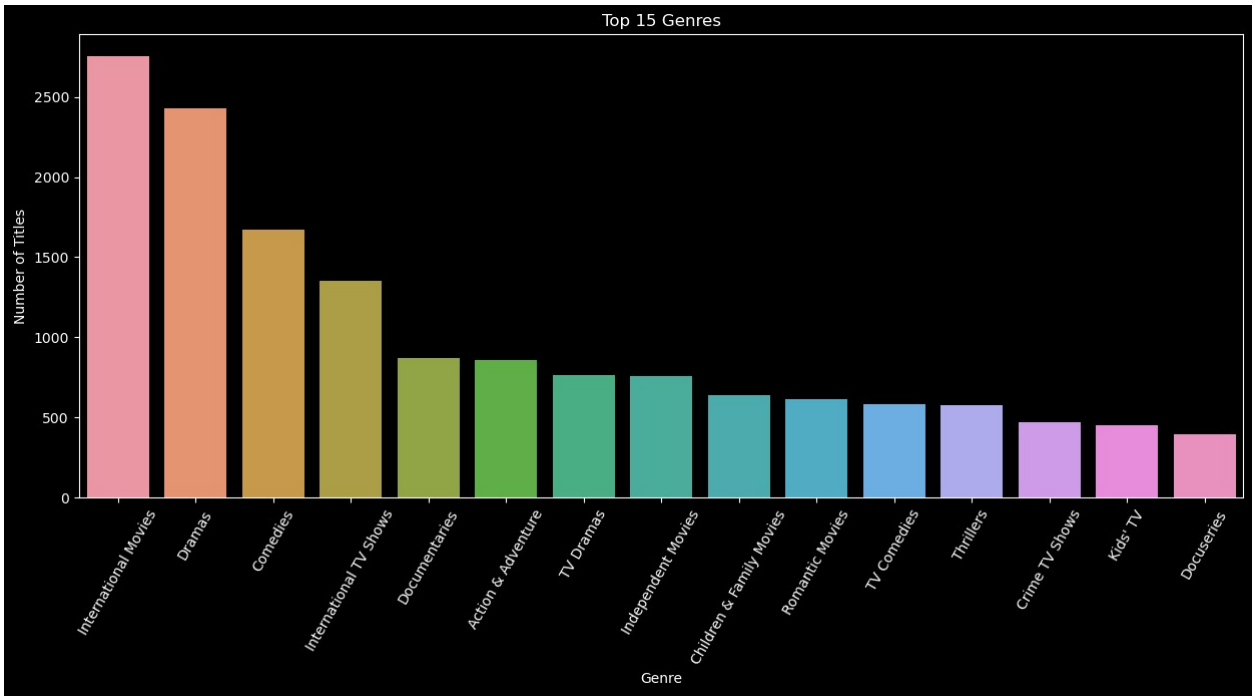
```
1          2021  TV-MA  2 Seasons        2021             9  September

2          2021  TV-MA  2 Seasons        2021             9  September

3          2021  TV-MA  2 Seasons        2021             9  September

4          2021  TV-MA  2 Seasons        2021             9  September
```

```
   day_added Weekday_added
0         25      Saturday
1         24        Friday
2         24        Friday
3         24        Friday
4         24        Friday
```

```
# Remove text within parentheses from the 'title' column
df_['title'] = df_['title'].str.replace(r"\(.*\)", "", regex=True)

# Display the first few rows of the DataFrame
df_.head()
```

Univariate Analysis

```
# Group by 'Genre' and count the number of unique titles in each genre
df_genre = df_.groupby(['Genre']).agg({"title":
"nunique"}).reset_index().sort_values(by=['title'], ascending=False)
[:15]

# Create the plot
plt.figure(figsize=(15, 6))
sns.barplot(x="Genre", y='title', data=df_genre)

# Rotate x-axis labels for better readability
plt.xticks(rotation=60)

# Set plot title and labels
plt.title('Top 15 Genres')
plt.xlabel('Genre')
plt.ylabel('Number of Titles')

# Show the plot
plt.show()
```

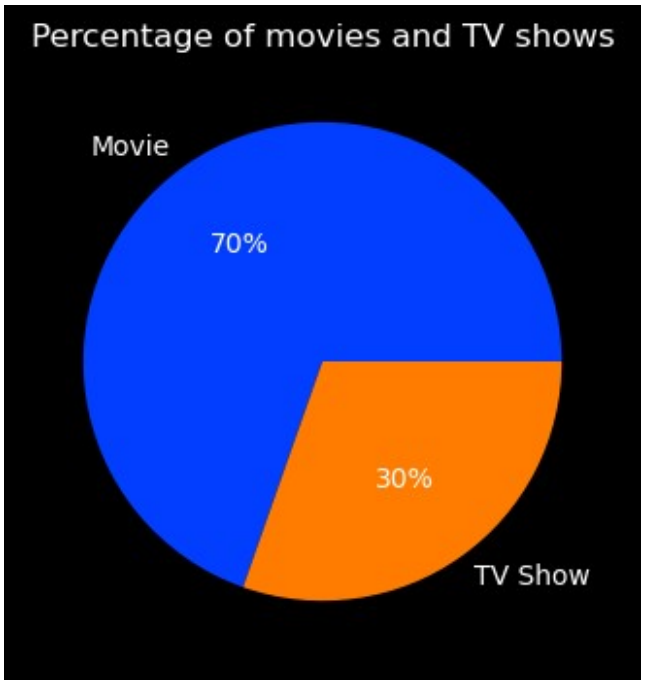


International Movies, Dramas and Comedies are the most popular

```
df_pie = df_.groupby(['type']).agg({'title':'nunique'}).reset_index()

df_pie
```

```
      type  title
0    Movie   6131
1  TV Show   2676
```

```
# Define colors for the pie chart
colors = sns.color_palette('bright')[0:5]

# Create the pie chart
plt.figure(figsize=(10, 4))
plt.pie(df_pie['title'], labels=df_pie['type'], colors=colors,
autopct='%.0f%%')

# Add title to the pie chart
plt.title('Percentage of movies and TV shows')

# Show the pie chart
plt.show()
```



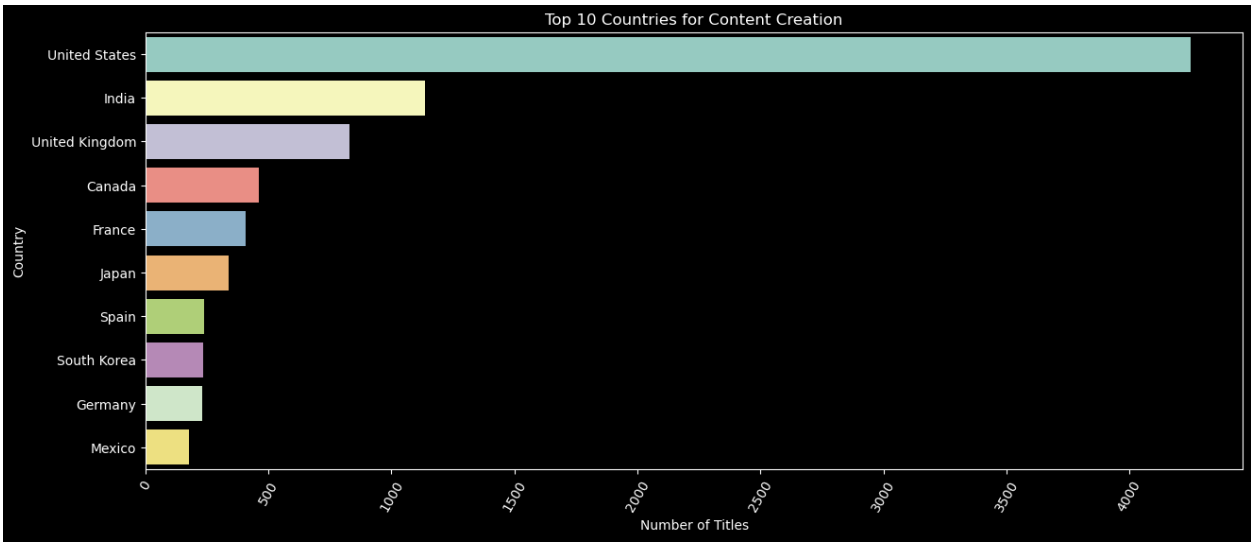
We have 70:30 ratio of Movies and TV Shows in our data

```
df_['Country'] = df_['Country'].str.replace(',', '')
df_.head()
```

```
                  title         Actors         Directors  \
0  Dick Johnson Is Dead  Unknown Actor   Kirsten Johnson
1         Blood & Water     Ama Qamata  Unknown Director
2         Blood & Water     Ama Qamata  Unknown Director
3         Blood & Water     Ama Qamata  Unknown Director
4         Blood & Water    Khosi Ngema  Unknown Director

                     Genre        Country show_id     type
date_added  \
0           Documentaries  United States      s1    Movie  September
25, 2021
1  International TV Shows   South Africa      s2  TV Show  September
24, 2021
2               TV Dramas   South Africa      s2  TV Show  September
24, 2021
3            TV Mysteries   South Africa      s2  TV Show  September
24, 2021
4  International TV Shows   South Africa      s2  TV Show  September
24, 2021

   release_year rating   duration
0          2020  PG-13     80-100
1          2021  TV-MA  2 Seasons
2          2021  TV-MA  2 Seasons
```

```
3          2021  TV-MA  2 Seasons
4          2021  TV-MA  2 Seasons
```

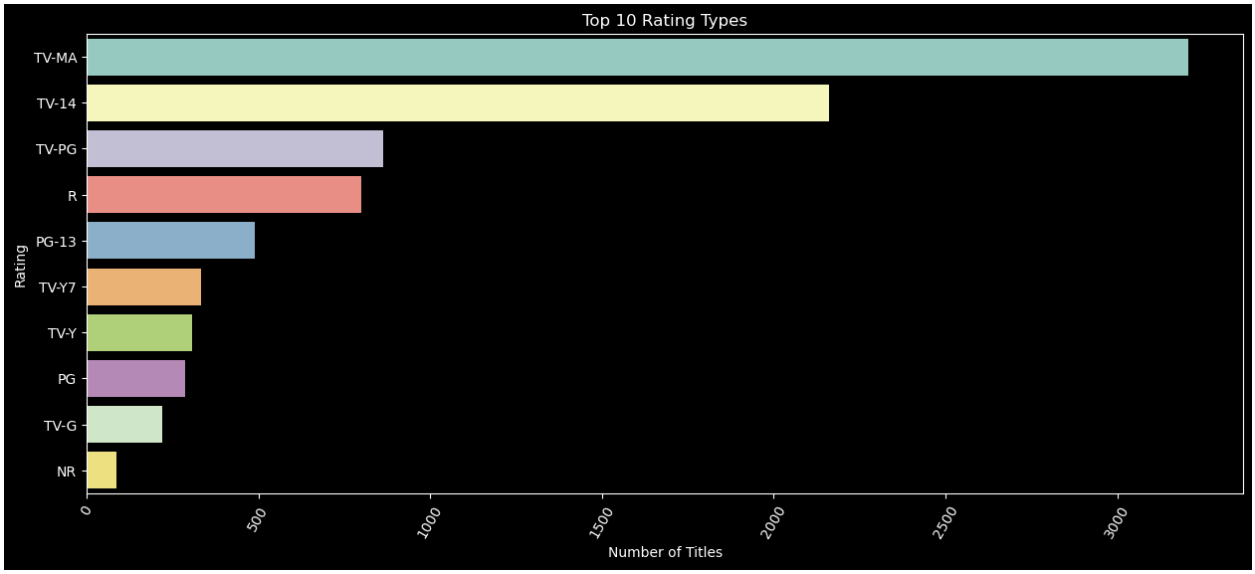
```
# Group by 'Country' and count the number of unique titles in each
country
df_country = df_.groupby(['Country']).agg({'title':
'nunique'}).reset_index().sort_values(by=['title'], ascending=False)
[:10]

# Create the bar plot
plt.figure(figsize=(15, 6))
sns.barplot(y="Country", x='title', data=df_country)

# Rotate x-axis labels for better readability
plt.xticks(rotation=60)

# Set plot title and labels
plt.title('Top 10 Countries for Content Creation')
plt.xlabel('Number of Titles')
plt.ylabel('Country')

# Show the plot
plt.show()
```



US,India,UK,Canada and France are leading countries in Content Creation on Netflix

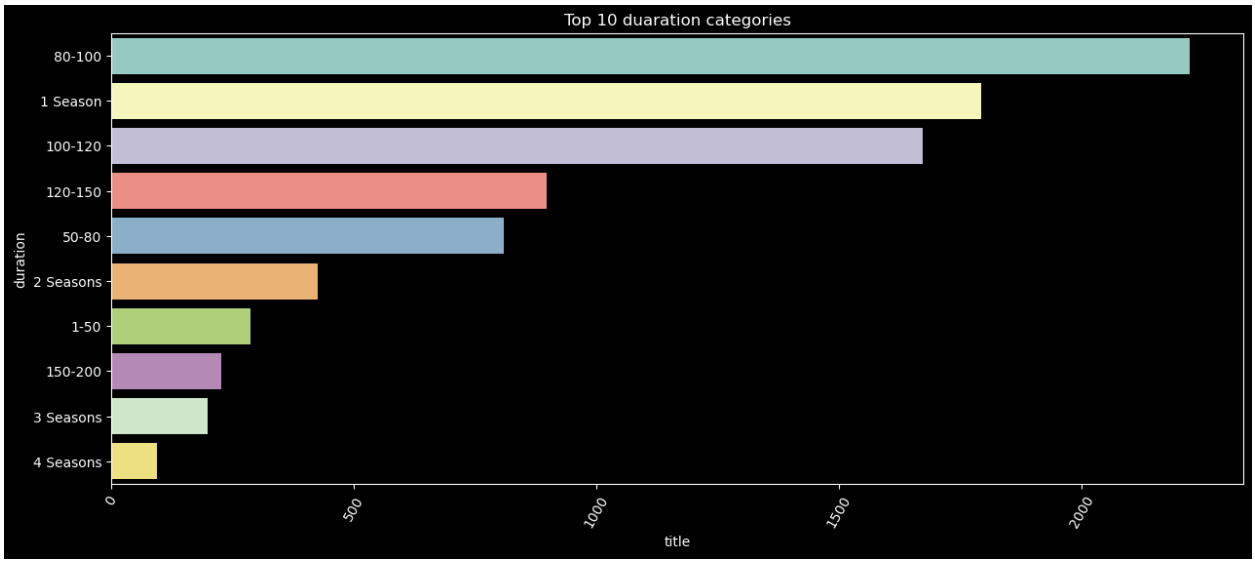
```
# Group by 'rating' and count the number of unique titles for each
rating
df_rating = df_.groupby(['rating']).agg({'title':
'nunique'}).reset_index().sort_values(by=['title'], ascending=False)
[:10]

# Create the bar plot
```

```
plt.figure(figsize=(15, 6))
sns.barplot(y="rating", x='title', data=df_rating)

# Rotate x-axis labels for better readability
plt.xticks(rotation=60)

# Set plot title and labels
plt.title('Top 10 Rating Types')
plt.xlabel('Number of Titles')
plt.ylabel('Rating')

# Show the plot
plt.show()
```



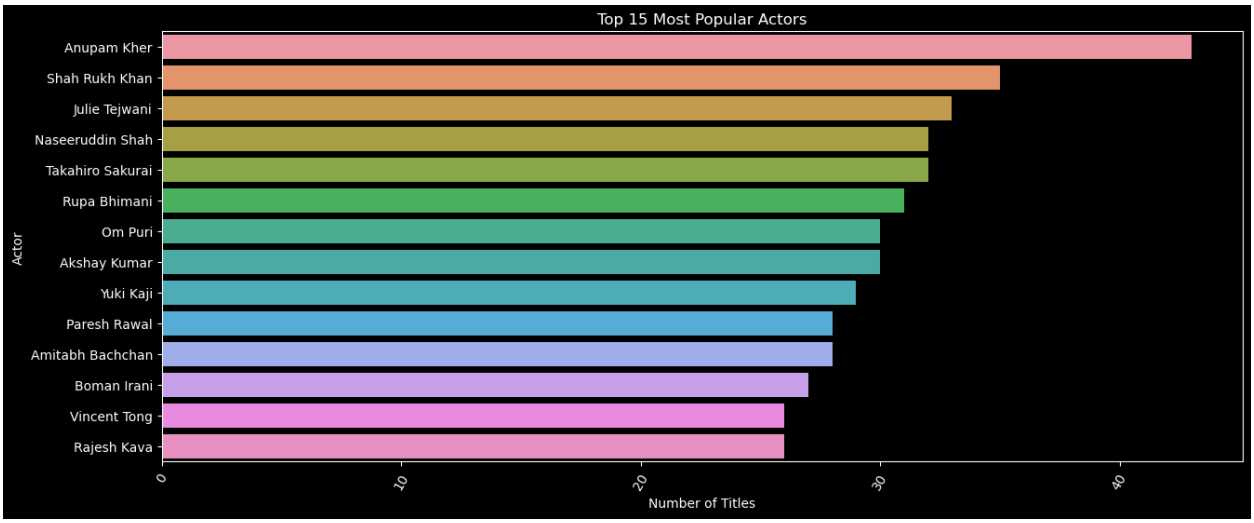
Most of the highly rated content on Netflix is intended for Mature Audiences

```
df_duration =
df_.groupby(['duration']).agg({'title':'nunique'}).reset_index().sort_
values(by=['title'],ascending=False)[:10]

plt.figure(figsize=(15,6))
sns.barplot(y = "duration",x = 'title', data = df_duration)
plt.xticks(rotation = 60)
plt.title('Top 10 duaration categories')
plt.show()
```



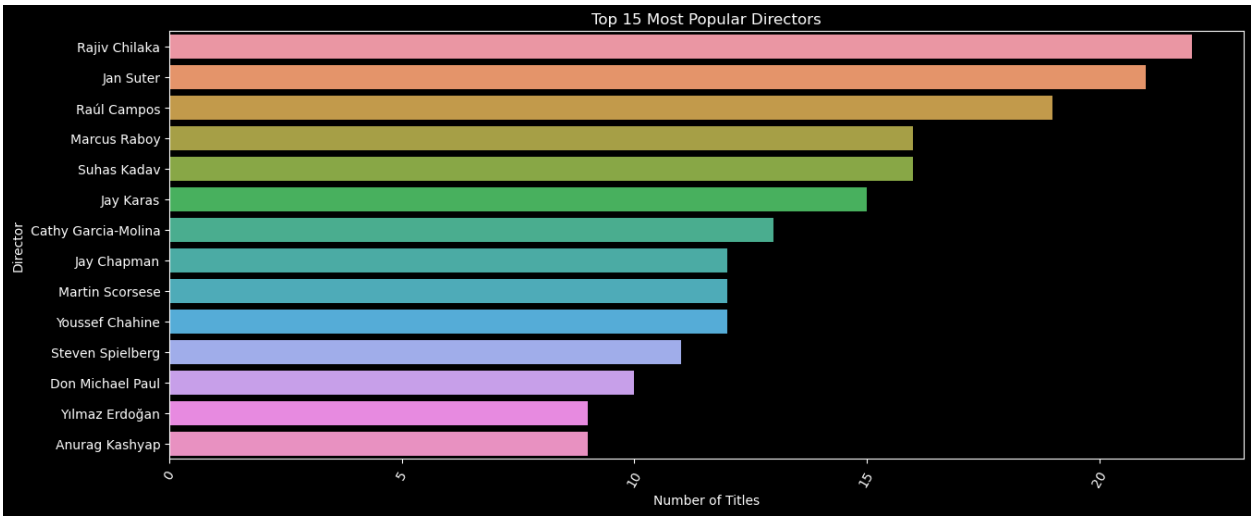
The duration of Most Watched content in our whole data is 80-100 mins. These must be movies and Shows having only 1 Season.

```
# Group by 'Actors' and count the number of unique titles for each
actor
df_actors = df_.groupby(['Actors']).agg({'title':
'nunique'}).reset_index().sort_values(by=['title'], ascending=False)
[:15]

# Exclude rows with 'Unknown Actor'
df_actors = df_actors[df_actors['Actors'] != 'Unknown Actor']

# Create the bar plot
plt.figure(figsize=(15, 6))
sns.barplot(y="Actors", x='title', data=df_actors)

# Rotate x-axis labels for better readability
plt.xticks(rotation=60)

# Set plot title and labels
plt.title('Top 15 Most Popular Actors')
plt.xlabel('Number of Titles')
plt.ylabel('Actor')

# Show the plot
plt.show()
```



Anupam Kher,SRK,Julie Tejwani, Naseeruddin Shah and Takahiro Sakurai occupy the top stop in Most Watched content.

```
# Group by 'Directors' and count the number of unique titles for each director
df_directors = df_.groupby(['Directors']).agg({'title': 'nunique'}).reset_index().sort_values(by=['title'], ascending=False)[:15]

# Exclude rows with 'Unknown Director'
df_directors = df_directors[df_directors['Directors'] != 'Unknown Director']

# Create the bar plot
plt.figure(figsize=(15, 6))
sns.barplot(y="Directors", x='title', data=df_directors)

# Rotate x-axis labels for better readability
plt.xticks(rotation=60)

# Set plot title and labels
plt.title('Top 15 Most Popular Directors')
plt.xlabel('Number of Titles')
plt.ylabel('Director')

# Show the plot
plt.show()
```



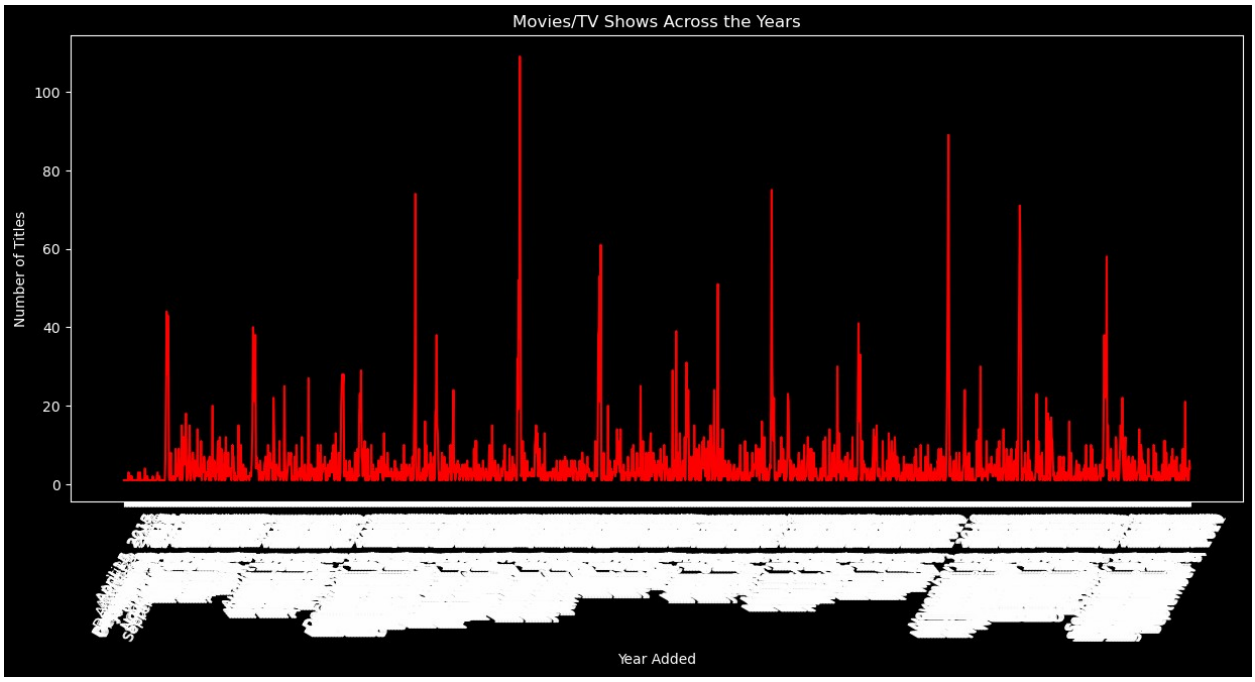
Rajiv Chilaka, Jan Suter and Raul Campos are the most popular directors across Netflix

```
df_.head()
```

```
                  title         Actors         Directors  \
0  Dick Johnson Is Dead  Unknown Actor   Kirsten Johnson
1         Blood & Water     Ama Qamata  Unknown Director
2         Blood & Water     Ama Qamata  Unknown Director
3         Blood & Water     Ama Qamata  Unknown Director
4         Blood & Water    Khosi Ngema  Unknown Director

                     Genre        Country show_id     type
date_added  \
0          Documentaries  United States      s1    Movie  September
25, 2021
1  International TV Shows   South Africa      s2  TV Show  September
24, 2021
2               TV Dramas   South Africa      s2  TV Show  September
24, 2021
3            TV Mysteries   South Africa      s2  TV Show  September
24, 2021
4  International TV Shows   South Africa      s2  TV Show  September
24, 2021

   release_year rating   duration
0          2020  PG-13     80-100
1          2021  TV-MA  2 Seasons
2          2021  TV-MA  2 Seasons
3          2021  TV-MA  2 Seasons
4          2021  TV-MA  2 Seasons
```

```
# Group by 'year_added' and count the number of unique titles for each
year
df_year = df_.groupby(['date_added']).agg({'title':
'nunique'}).reset_index()
```

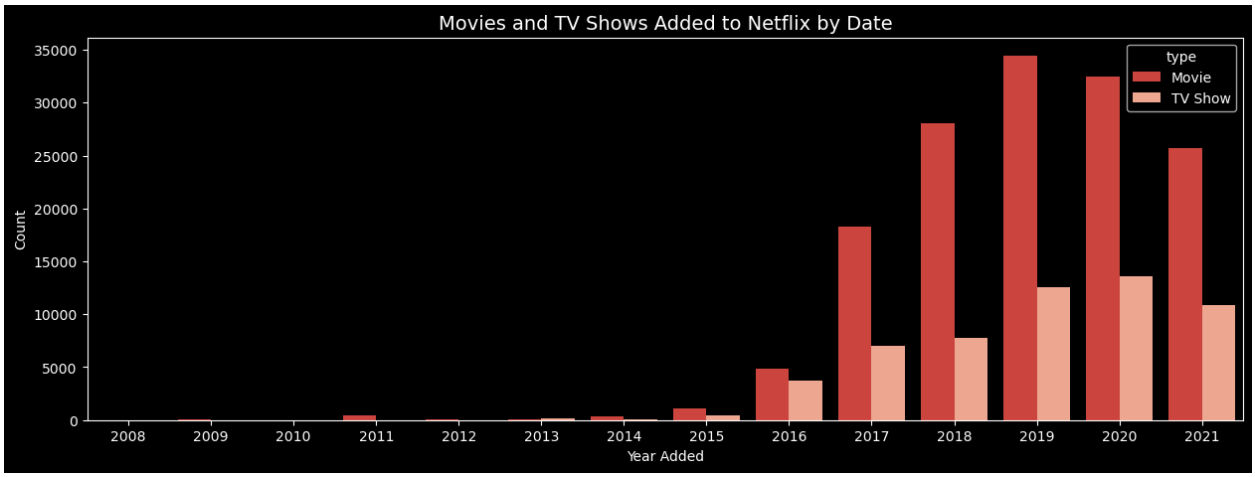
```
# Create the line plot
plt.figure(figsize=(15, 6))
sns.lineplot(x="date_added", y='title', data=df_year, color='red')

# Rotate x-axis labels for better readability
plt.xticks(rotation=60)

# Set plot title and labels
plt.title('Movies/TV Shows Across the Years')
plt.xlabel('Year Added')
plt.ylabel('Number of Titles')

# Show the plot
plt.show()
```



The Amount of Content across Netflix has increased from 2008 continuously till 2019. Then started decreasing from here(probably due to Covid)

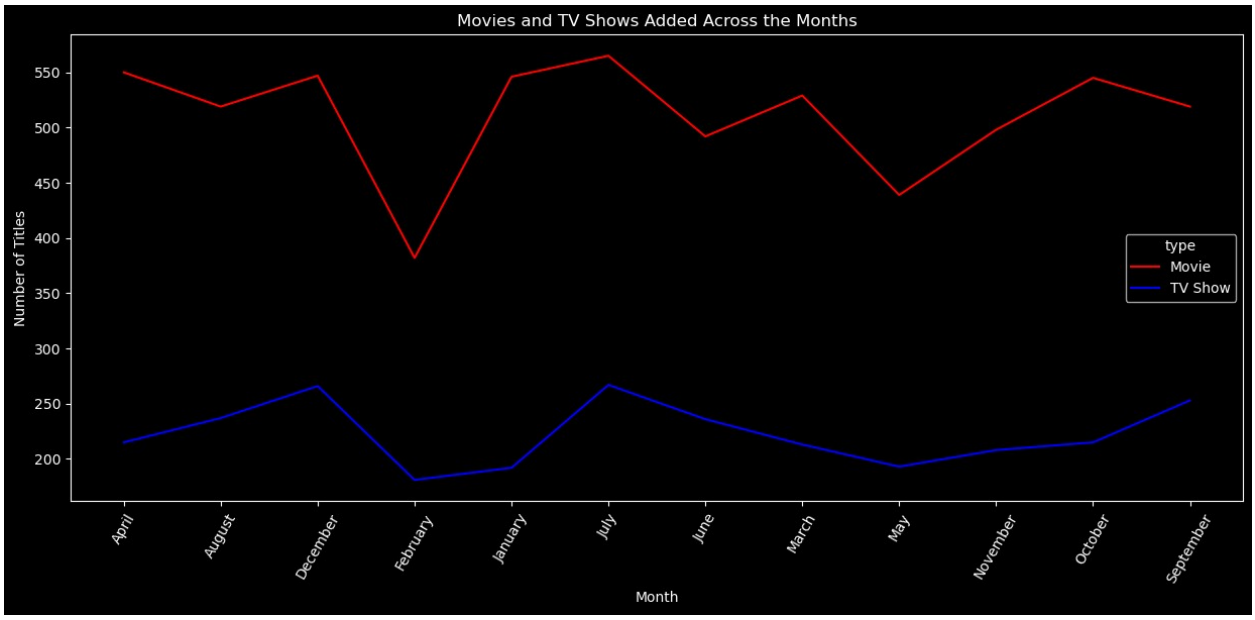
```
# Set the figure size and style
fig = plt.figure(figsize=(15, 5))
plt.style.use('dark_background')

# Create the count plot
sns.countplot(data=df_, x='year_added', hue='type', palette="Reds_r")

# Set plot title and labels
plt.title('Movies and TV Shows Added to Netflix by Date', fontsize=14)
plt.xlabel('Year Added')
```

```
plt.ylabel('Count')

# Show the plot
plt.show()
```



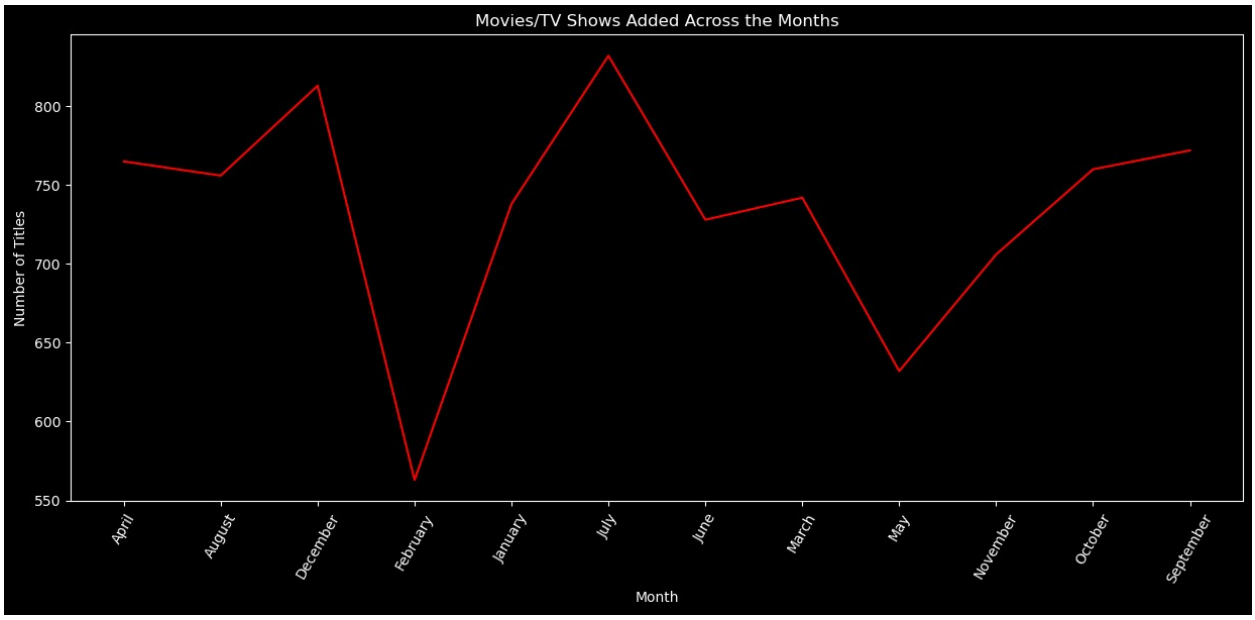
Over the years both TV shows and movie contents addtion has increased after 2020 its started declining may be due to Covid relief, Movies addtion is more compare to TV shows over the years

```
# Group by 'month_name' and 'type' and count the number of unique
titles for each combination
df_month = df_.groupby(['month_name',
'type']).agg({'title':'nunique'}).reset_index()

# Create the line plot
plt.figure(figsize=(15, 6))
sns.lineplot(x="month_name", y='title', data=df_month, hue='type',
palette={'Movie': 'red', 'TV Show': 'blue'})

# Rotate x-axis labels for better readability
plt.xticks(rotation=60)

# Set plot title and labels
plt.title('Movies and TV Shows Added Across the Months')
plt.xlabel('Month')
plt.ylabel('Number of Titles')

# Show the plot
plt.show()
```



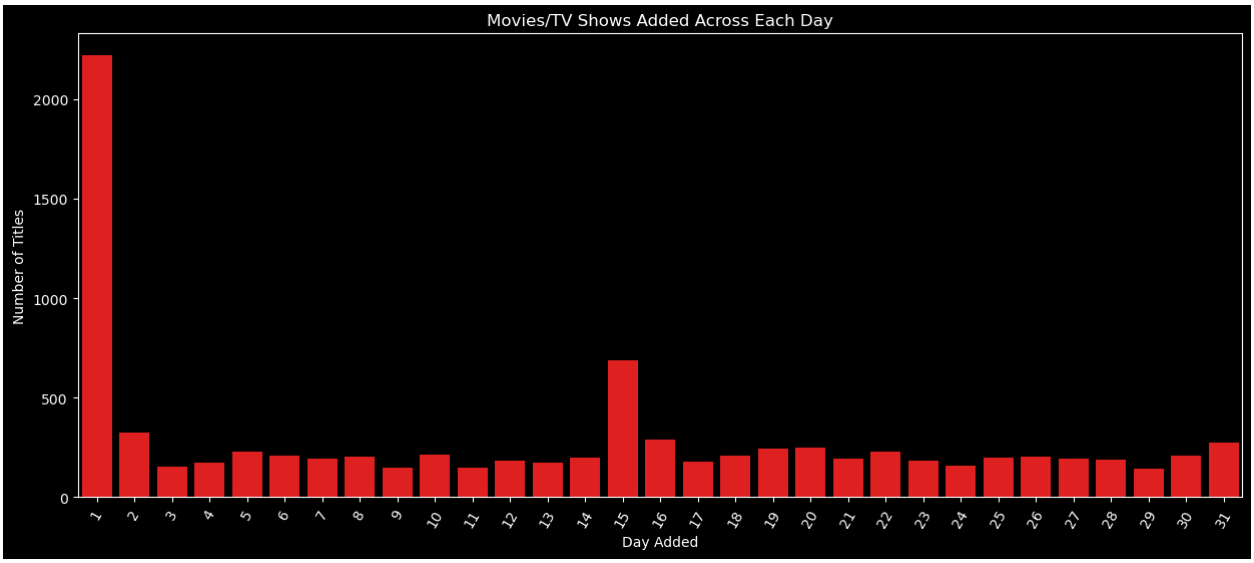
for both TV shows and Movies best launch month remain same which is July followed by December

```
# Group by 'month_name' and count the number of unique titles for each month
df_month = 
df_.groupby(['month_name']).agg({'title':'nunique'}).reset_index()

# Create the line plot
plt.figure(figsize=(15, 6))
sns.lineplot(x="month_name", y='title', data=df_month, color='red')

# Rotate x-axis labels for better readability
plt.xticks(rotation=60)

# Set plot title and labels
plt.title('Movies/TV Shows Added Across the Months')
plt.xlabel('Month')
plt.ylabel('Number of Titles')

# Show the plot
plt.show()
```



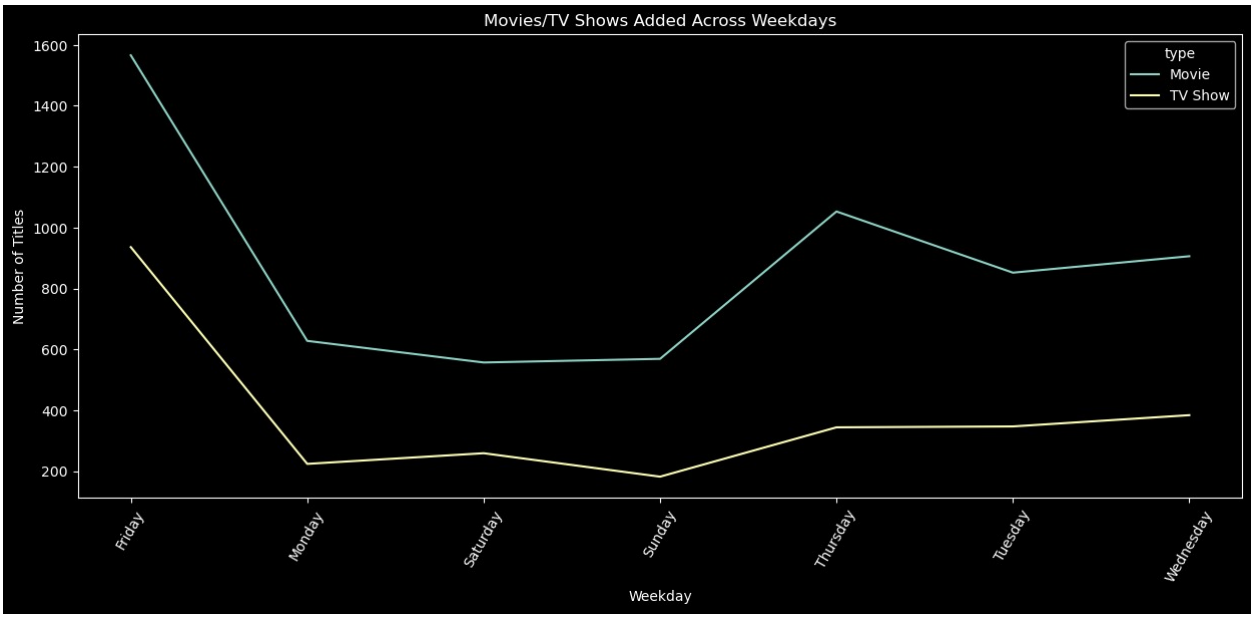
In general most of the content get added in december and july month

```
# Group by 'day_added' and count the number of unique titles for each
day
df_day =
df_.groupby(['day_added']).agg({'title':'nunique'}).reset_index()

# Create the bar plot
plt.figure(figsize=(15, 6))
sns.barplot(x="day_added", y='title', data=df_day, color='red')

# Rotate x-axis labels for better readability
plt.xticks(rotation=60)

# Set plot title and labels
plt.title('Movies/TV Shows Added Across Each Day')
plt.xlabel('Day Added')
plt.ylabel('Number of Titles')

# Show the plot
plt.show()
```

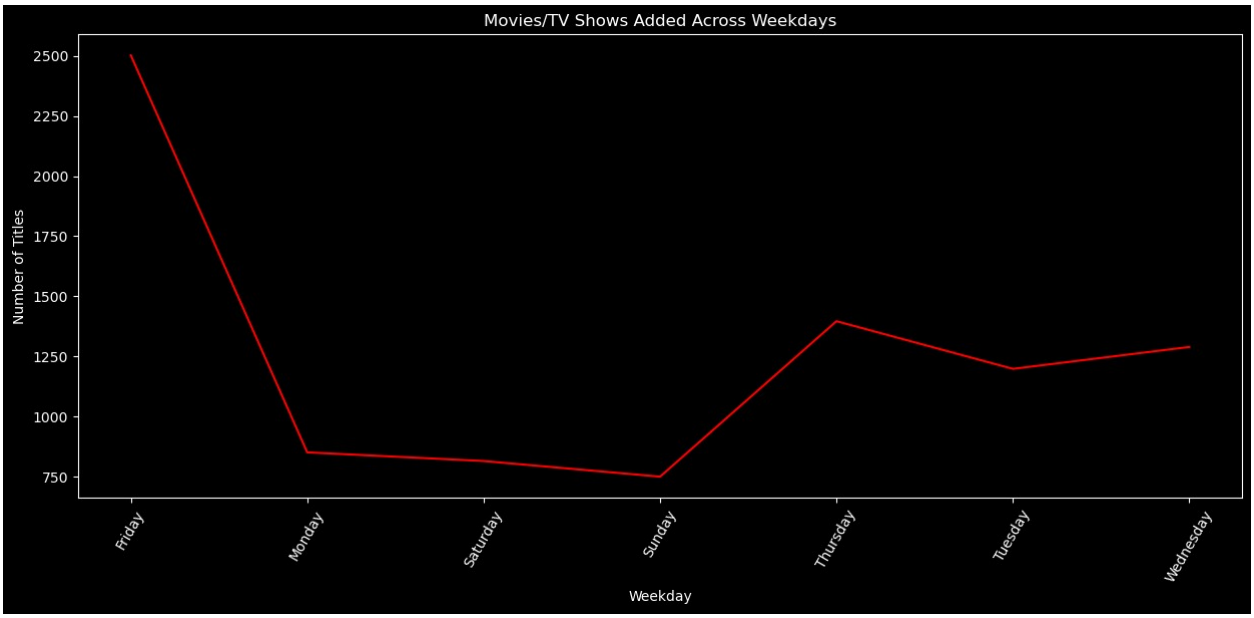


It was evident that 1st of every month was when the most content was added.

```
# Group by 'Weekday_added' and 'type', and count the number of unique
titles for each combination
df_weekday = df_.groupby(['Weekday_added',
'type']).agg({'title':'nunique'}).reset_index()

# Create the line plot
plt.figure(figsize=(15, 6))
sns.lineplot(x="Weekday_added", y='title', data=df_weekday,
color='red', hue='type')

# Rotate x-axis labels for better readability
plt.xticks(rotation=60)

# Set plot title and labels
plt.title('Movies/TV Shows Added Across Weekdays')
plt.xlabel('Weekday')
plt.ylabel('Number of Titles')

# Show the plot
plt.show()
```




```
# Group by 'Weekday_added' and count the number of unique titles for
each weekday
df_weekday = df_.groupby(['Weekday_added']).agg({'title':
'nunique'}).reset_index()

# Create the line plot
plt.figure(figsize=(15, 6))
sns.lineplot(x="Weekday_added", y='title', data=df_weekday,
color='red')

# Rotate x-axis labels for better readability
plt.xticks(rotation=60)

# Set plot title and labels
plt.title('Movies/TV Shows Added Across Weekdays')
plt.xlabel('Weekday')
plt.ylabel('Number of Titles')

# Show the plot
plt.show()
```

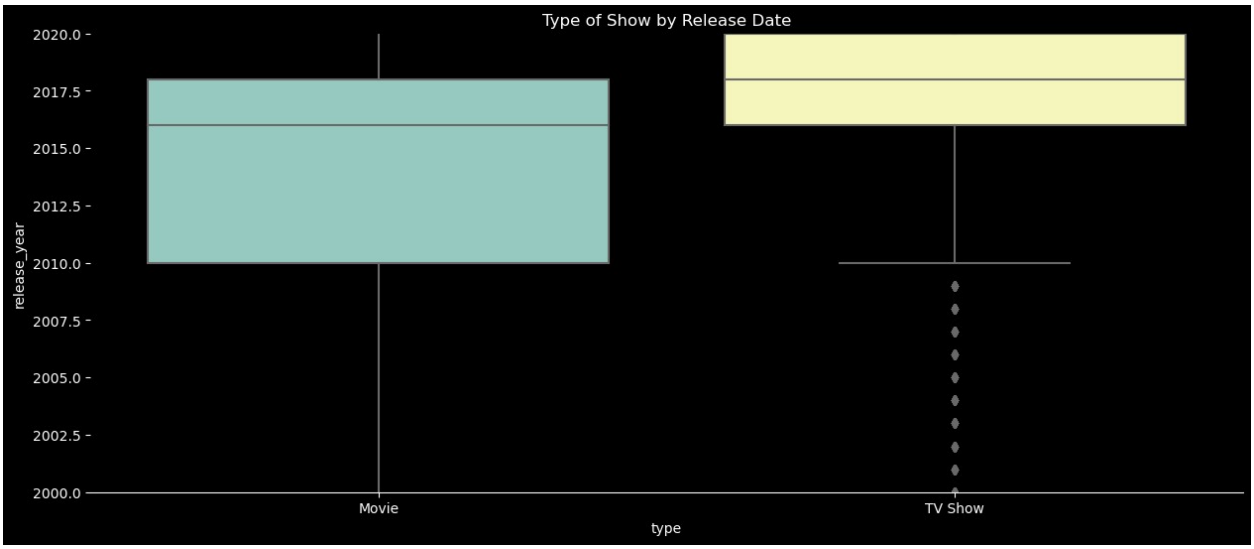


For content release on Netflix, Friday is the best day followed by Thursday

```
df_.columns
```

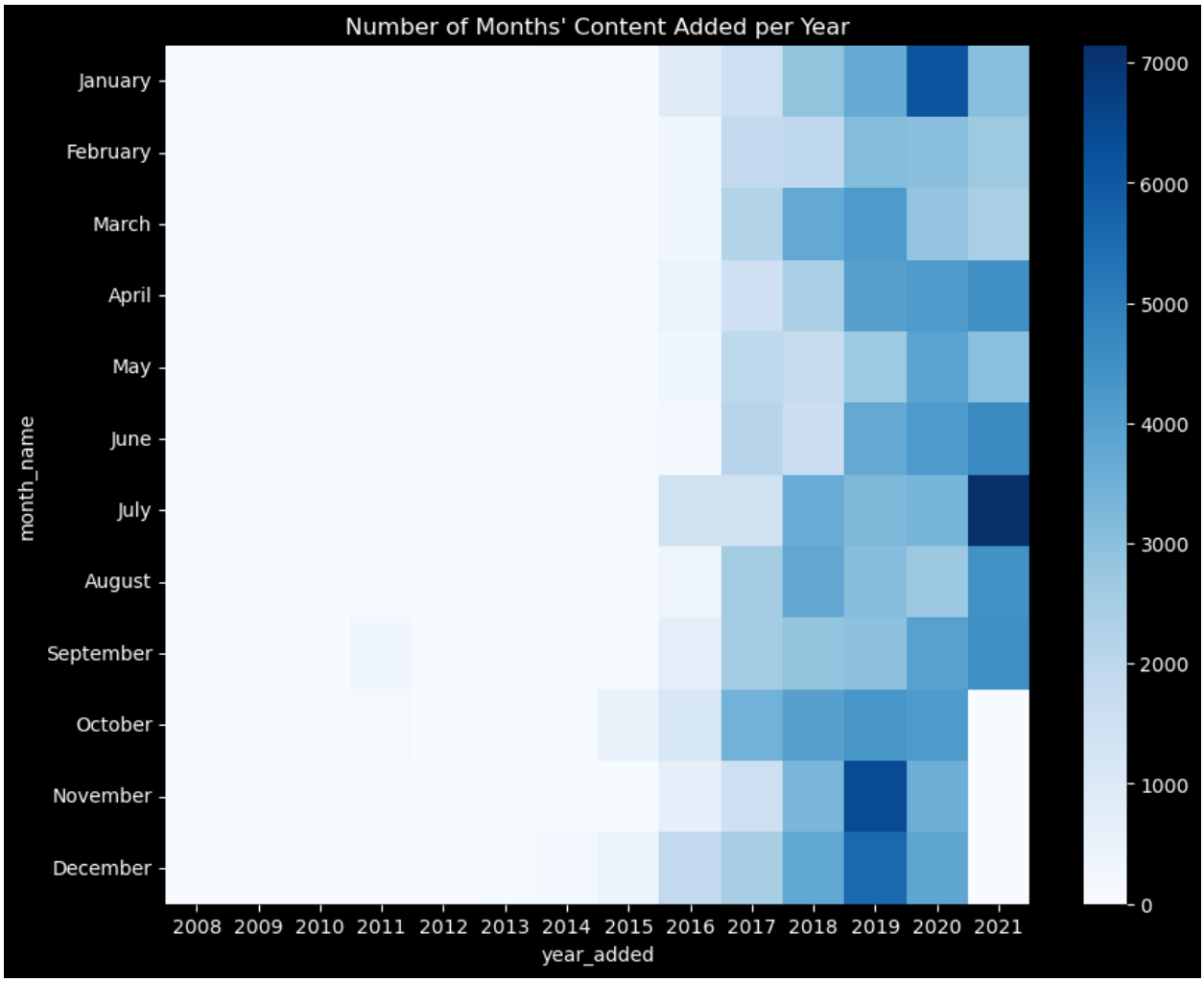
```
Index(['title', 'Actors', 'Directors', 'Genre', 'Country', 'show_id',
'type',
       'date_added', 'release_year', 'rating', 'duration',
'year_added',
       'month_added', 'month_name', 'day_added', 'Weekday_added'],
      dtype='object')
```

```
# Create the box plot
plt.figure(figsize=(15, 6))
sns.boxplot(x='type', y='release_year', data=df_)
sns.despine(left=True)

# Set plot title and y-axis limit
plt.title('Type of Show by Release Date')
plt.ylim(2000, 2020)  # Set y-axis limit to focus on the relevant
range of release years

# Show the plot
plt.show()
```



It sees tv shows have a more recent release_year. This means tv shows are releasing more in recent years

```
#Bivariate Analysis

# Define the order of months
month_order = ['January', 'February', 'March', 'April', 'May', 'June',
'July', 'August', 'September',
               'October', 'November', 'December']

# Group by 'year_added' and 'month_name', count the number of
occurrences, and reshape the data
content = df_.groupby('year_added')
['month_name'].value_counts().unstack().fillna(0)[month_order].T

# Create the heatmap
plt.figure(figsize=(10, 8))
plt.title("Number of Months' Content Added per Year")
sns.heatmap(content, cmap='Blues')

# Show the plot
plt.show()
```

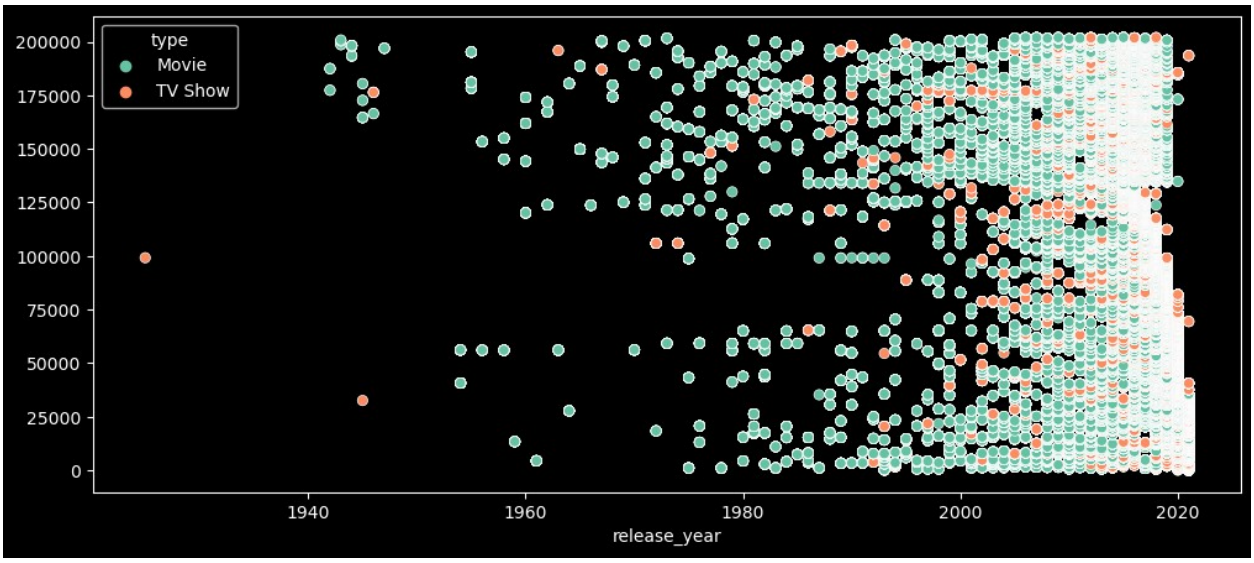


Most number of Movies and TV shows were added in November, 2019 and July, 2021

Fewer movies and TV shows were added from 2008 to 2015

```
plt.figure(figsize = (12,5))
sns.scatterplot(y = df_.index , x = df_.release_year , hue =
df_.type , palette='Set2')
```

```
<Axes: xlabel='release_year'>
```



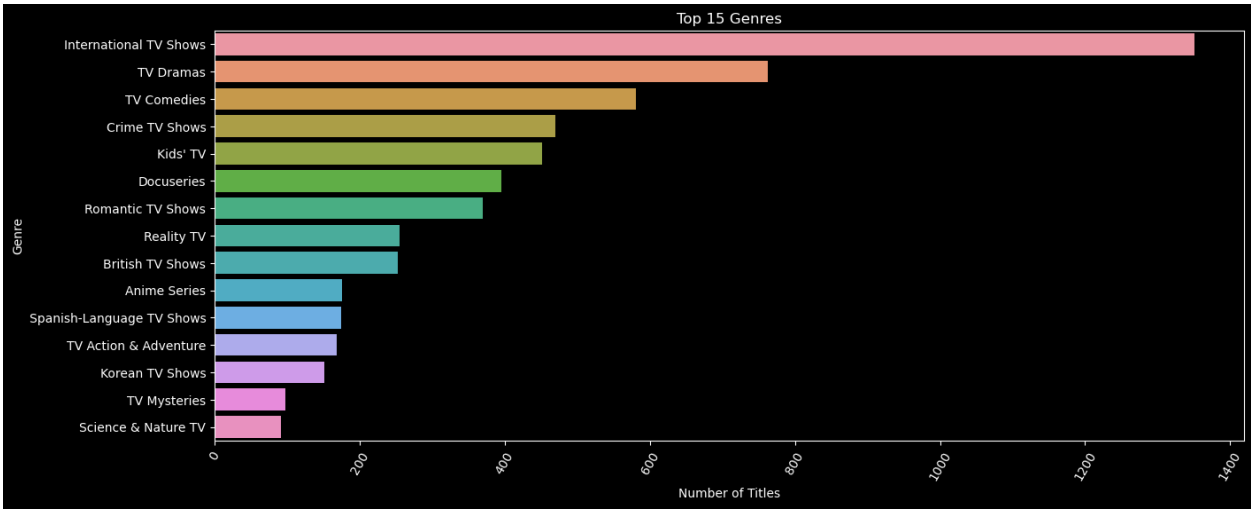

```
df_.groupby(['day_added']).agg({"title":"nunique"})
```

day_added	title
1	2219
2	325
3	151
4	175
5	231
6	210
7	194
8	201
9	148
10	214
11	149
12	181
13	175
14	198
15	688
16	289
17	180
18	207
19	243
20	249
21	193
22	230
23	184
24	159
25	197
26	206
27	195
28	190
29	141

```
30            211
31            274
```

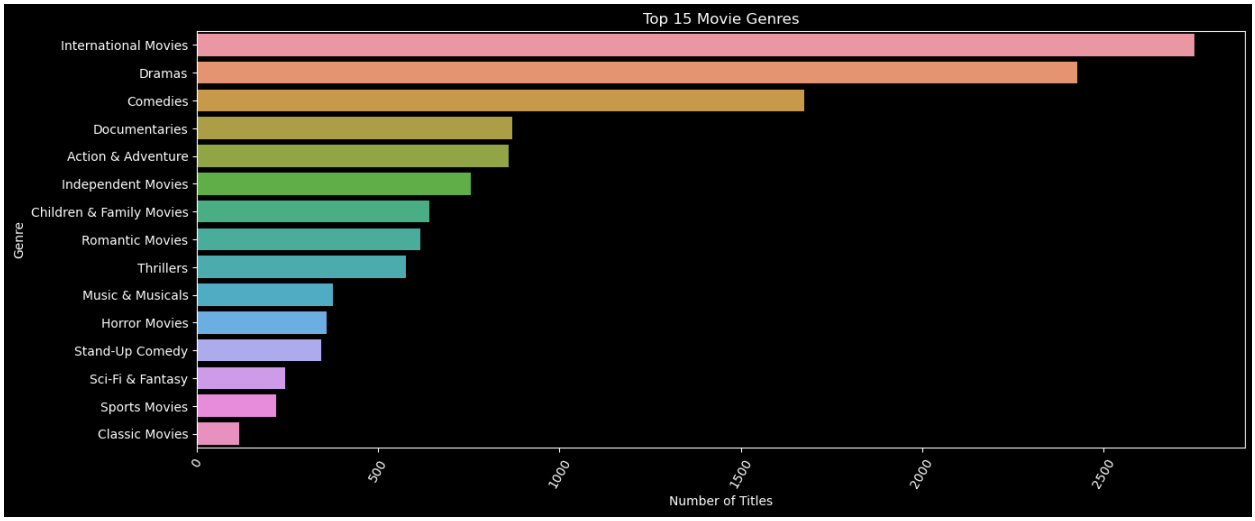
It was evident that 1st of every month was when the most content was added.

```
#Univariate Analysis separately for shows and movies

df_shows = df_[df_['type']=='TV Show']
df_movies = df_[df_['type']=='Movie']

# Group by 'Genre' and count the number of unique titles for each
genre
df_genre =
df_shows.groupby(['Genre']).agg({"title":"nunique"}).reset_index().sor
t_values(by=['title'], ascending=False)[:15]

# Create the bar plot
plt.figure(figsize=(15, 6))
sns.barplot(y="Genre", x='title', data=df_genre)

# Rotate x-axis labels for better readability
plt.xticks(rotation=60)

# Set plot title and labels
plt.title('Top 15 Genres')
plt.xlabel('Number of Titles')
plt.ylabel('Genre')

# Show the plot
plt.show()
```



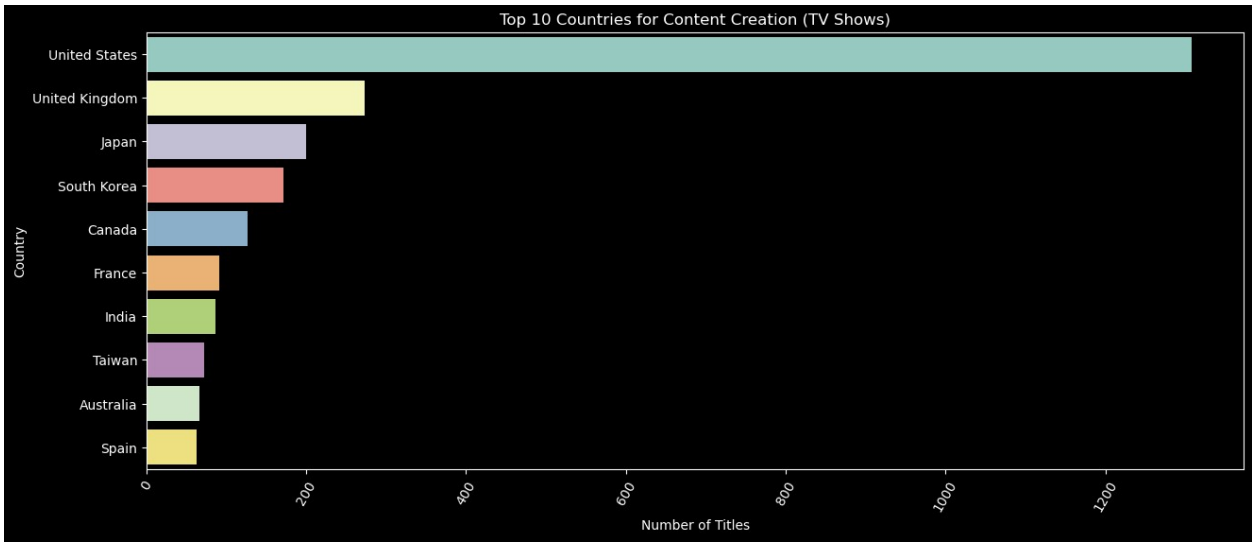

```
# Group by 'Genre' and count the number of unique titles for each
genre
df_genre = df_movies.groupby(['Genre']).agg({'title':
```

```
'nunique'}).reset_index().sort_values(by='title', ascending=False)
[:15]

# Create the bar plot
plt.figure(figsize=(15, 6))
sns.barplot(y='Genre', x='title', data=df_genre)

# Rotate x-axis labels for better readability
plt.xticks(rotation=60)

# Set plot title and labels
plt.title('Top 15 Movie Genres')
plt.xlabel('Number of Titles')
plt.ylabel('Genre')

# Show the plot
plt.show()
```

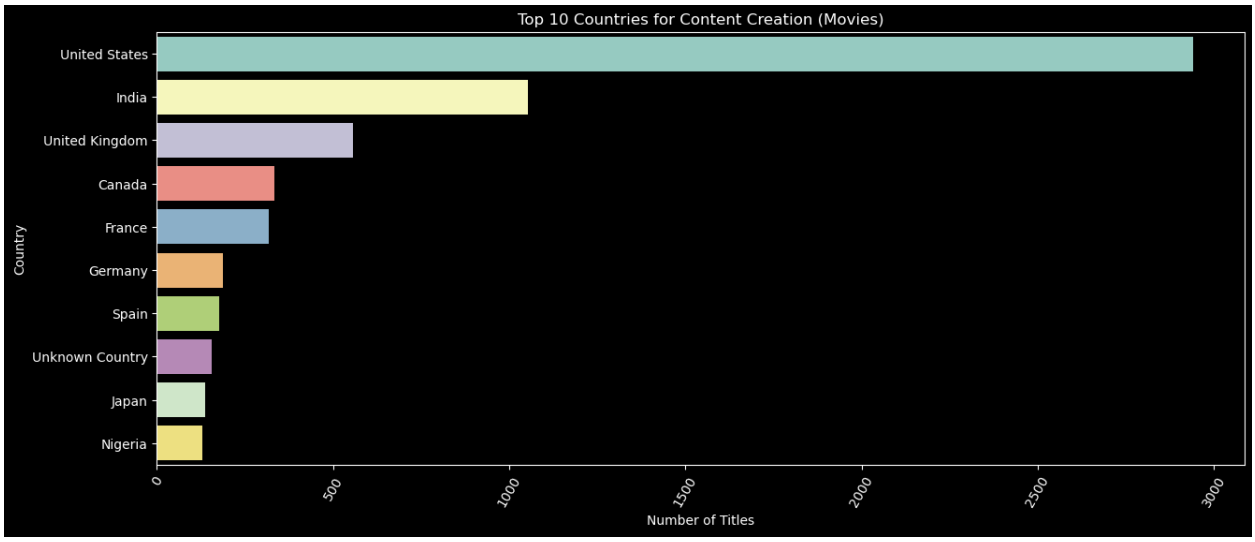



```
# Group by 'Country' and count the number of unique titles for each
country
df_country = df_shows.groupby(['Country']).agg({'title':
'nunique'}).reset_index().sort_values(by='title', ascending=False)
[:10]

# Create the bar plot
plt.figure(figsize=(15, 6))
sns.barplot(y='Country', x='title', data=df_country)

# Rotate x-axis labels for better readability
plt.xticks(rotation=60)

# Set plot title and labels
plt.title('Top 10 Countries for Content Creation (TV Shows)')
plt.xlabel('Number of Titles')
```

```
plt.ylabel('Country')

# Show the plot
plt.show()
```




```
# Group by 'Country' and count the number of unique titles for each
country
df_country = df_movies.groupby(['Country']).agg({'title':
'nunique'}).reset_index().sort_values(by='title', ascending=False)
[:10]

# Create the bar plot
plt.figure(figsize=(15, 6))
sns.barplot(y='Country', x='title', data=df_country)

# Rotate x-axis labels for better readability
plt.xticks(rotation=60)

# Set plot title and labels
plt.title('Top 10 Countries for Content Creation (Movies)')
plt.xlabel('Number of Titles')
plt.ylabel('Country')

# Show the plot
plt.show()
```



United States is leading across both TV Shows and Movies, UK also provides great content across TV Shows and Movies. Surprisingly India is much more prevalent in Movies as compared TV Shows.

Moreover the number of Movies created in India outweigh the sum of TV Shows and Movies across UK since India was rated as second in net sum of whole content across Netflix.

Business insights

Over the years both TV shows and movie contents addtion has increased till 2020, but after 2020 its started declining may be due to Covid relief, number of Movies added is more compare to TV shows over the years

Most of the content get added in december and july month, for day wise, Friday is the best day followed by Thursday

It was evident that 1st of every month was when the most content was added.

Anupam Kher,SRK,Julie Tejwani, Naseeruddin Shah and Takahiro Sakurai occupy the top stop in Most Watched content.

Rajiv Chilaka, Jan Suter and Raul Campos are the most popular directors across Netflix

Rajiv Chilaka director producing more movies

Netflix is more focussing on movies compare to TV shows

There is a 70:30 ratio of Movies and TV Shows content in Netflix platform

International Movies, Dramas and Comedies are the most popular are most popular Genre

US,India,UK,Canada and France are leading countries in Content Creation on Netflix

Most of the highly rated content on Netflix is intended for Mature Audiences

The duration of Most Watched content in our whole data is 80-120 mins. These must be movies and Shows having only 1 Season.

United States is leading across both TV Shows and Movies, UK also provides great content across TV Shows and Movies. Surprisingly India is much more prevalent in Movies as compared TV Shows.

Moreover the number of Movies created in India outweigh the sum of TV Shows and Movies across UK since India was rated as second in net sum of whole content across Netflix.

```
Recommendations
```

The most popular Genres across the countries and in both TV Shows and Movies are Drama, Comedy and International TV Shows/Movies, so recommended to generate more content on these genres.

Add TV Shows/ movies in the month of July 1st or August 1st.

Add movies for Indian Audience, it has been declining since 2018.

While creating content, take into consideration the popular actors/directors for that country. Also take into account the director-actor combination which is highly recommended.

For audience 80-120 mins is the recommended length for movies.