In [235]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
df = pd.read_csv("netflix.csv")
```

In [3]:
```
#length of data
len(df)
```

Out[3]:
```
8807
```

In [4]:
```
#checking datatypes
df.dtypes
```

Out[4]:
```
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 [5]:
```
#Cheaking top 5 rows
df.head()
```

Out[5]:

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	Documentaries	As her father nears the end of his life, filmm...
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	After crossing paths at a party, a Cape Town t...
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season	Crime TV Shows, International TV Shows, TV Act...	To protect his family from a powerful drug lor...
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	1 Season	Docuseries, Reality TV	Feuds, flirtations and toilet talk go down amo...
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons	International TV Shows, Romantic TV Shows, TV ...	In a city of coaching centers known to train I...

In [6]:
```
#Cheaking bottom 5 rows
df.tail()
```

Out[6]:

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
8802	s8803	Movie	Zodiac	David Fincher	Mark Ruffalo, Jake Gyllenhaal, Robert Downey J...	United States	November 20, 2019	2007	R	158 min	Cult Movies, Dramas, Thrillers	A political cartoonist, a crime reporter and a...
8803	s8804	TV Show	Zombie Dumb	NaN	NaN	NaN	July 1, 2019	2018	TV-Y7	2 Seasons	Kids' TV, Korean TV Shows, TV Comedies	While living alone in a spooky town, a young g...
8804	s8805	Movie	Zombieland	Ruben Fleischer	Jesse Eisenberg, Woody Harrelson, Emma Stone, ...	United States	November 1, 2019	2009	R	88 min	Comedies, Horror Movies	Looking to survive in a world taken over by zo...
8805	s8806	Movie	Zoom	Peter Hewitt	Tim Allen, Courteney Cox, Chevy Chase, Kate Ma...	United States	January 11, 2020	2006	PG	88 min	Children & Family Movies, Comedies	Dragged from civilian life, a former superhero...
8806	s8807	Movie	Zubaan	Mozez Singh	Vicky Kaushal, Sarah-Jane Dias, Raaghav Chanan...	India	March 2, 2019	2015	TV-14	111 min	Dramas, International Movies, Music & Musicals	A scrappy but poor boy worms his way into a ty...

In [7]:
```
#checking null values in every column of our data
df.isnull().sum()
```

Out[7]:
```
show_id            0
type               0
title              0
director        2634
cast             825
country          831
date_added        10
release_year       0
rating             4
duration           3
listed_in          0
description        0
dtype: int64
```

In [8]:
```
df["rating"].value_counts()
```

Out[8]:
```
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
NC-17          3
UR             3
74 min         1
84 min         1
66 min         1
Name: rating, dtype: int64
```

In [9]:
```
#unnesting the directors column, i.e- creating separate lines for each director in a movie
constraint1 = df['director'].apply(lambda x: str(x).split(', ')).tolist()
df_new1=pd.DataFrame(constraint1,index=df['title'])
df_new1=df_new1.stack()
df_new1=pd.DataFrame(df_new1.reset_index())
df_new1.rename(columns={0:'Directors'},inplace=True)
df_new1.drop(['level_1'],axis=1,inplace=True)
df_new1.head()
```

Out[9]:

	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

In [10]:
```
#unnesting the cast column, i.e- creating separate lines for each cast member in a movie
constraint2=df['cast'].apply(lambda x: str(x).split(', ')).tolist()
df_new2=pd.DataFrame(constraint2,index=df['title'])
df_new2=df_new2.stack()
df_new2=pd.DataFrame(df_new2.reset_index())
df_new2.rename(columns={0:'Actors'},inplace=True)
df_new2.drop(['level_1'],axis=1,inplace=True)
df_new2.head()
```

Out[10]:

	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

In [11]:
```
#unnesting the listed_in column, i.e- creating separate lines for each genre in a movie
constraint3=df['listed_in'].apply(lambda x: str(x).split(', ')).tolist()
df_new3=pd.DataFrame(constraint3,index=df['title'])
df_new3=df_new3.stack()
df_new3=pd.DataFrame(df_new3.reset_index())
df_new3.rename(columns={0:'Genre'},inplace=True)
df_new3.drop(['level_1'],axis=1,inplace=True)
df_new3.head()
```

Out[11]:

	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

In [12]:
```
#unnesting the country column, i.e- creating separate lines for each country in a movie
constraint4=df['country'].apply(lambda x: str(x).split(', ')).tolist()
df_new4=pd.DataFrame(constraint4,index=df['title'])
df_new4=df_new4.stack()
df_new4=pd.DataFrame(df_new4.reset_index())
df_new4.rename(columns={0:'country'},inplace=True)
df_new4.drop(['level_1'],axis=1,inplace=True)
df_new4.head()
```

Out[12]:

	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

In [13]:
```
#merging the unnested director data with unnested actors data
df_new5=df_new2.merge(df_new1,on=['title'],how='inner')
#merging the above merged data with unnested genre data
df_new6=df_new5.merge(df_new3,on=['title'],how='inner')
#merging the above merged data with unnested country data
df_new=df_new6.merge(df_new4,on=['title'],how='inner')
```

In [14]:
```
df_new.head()
```

Out[14]:

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

In [15]:
```
#replacing nan values of director and actor by Unknown Actor and Director
df_new['Actors'].replace(['nan'],['Unknown Actor'],inplace=True)
df_new['Directors'].replace(['nan'],['Unknown Director'],inplace=True)
df_new['country'].replace(['nan'],[np.nan],inplace=True)
df_new.head()
```

Out[15]:

	title	Actors	Directors	Genre	country
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa
2	Blood & Water	Ama Qamata	Unknown Director	TV Dramas	South Africa
3	Blood & Water	Ama Qamata	Unknown Director	TV Mysteries	South Africa
4	Blood & Water	Khosi Ngema	Unknown Director	International TV Shows	South Africa

In [16]:
```
#merging our unnested data with the original data
df_final=df_new.merge(df[['show_id', 'type', 'title', 'date_added',
       'release_year', 'rating', 'duration']],on=['title'],how='left')
```

In [17]:
```
df_final.head()
```

Out[17]:

	title	Actors	Directors	Genre	country	show_id	type	date_added	release_year	rating	duration
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States	s1	Movie	September 25, 2021	2020	PG-13	90 min
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
2	Blood & Water	Ama Qamata	Unknown Director	TV Dramas	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
3	Blood & Water	Ama Qamata	Unknown Director	TV Mysteries	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
4	Blood & Water	Khosi Ngema	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons

In [18]:
```
#now checking nulls
df_final.isnull().sum()
```

Out[18]:
```
title              0
Actors             0
Directors          0
Genre              0
country        11897
show_id            0
type               0
date_added       158
release_year       0
rating            67
duration           3
dtype: int64
```

In duration column, it was observed that the nulls had values which were written in corresponding ratings column, i.e- you can't expect ratings to be in min. So the duration column nulls are replaced by corresponding values in ratings column

In [19]:
```
df_final.loc[df_final['duration'].isnull(),'duration']=df_final.loc[df_final['duration'].isnull(),'duration'].fillna(df_final['ra
df_final.loc[df_final['rating'].str.contains('min', na=False),'rating']='NR'
df_final.isnull().sum()
```

Out[19]:
```
title              0
Actors             0
Directors          0
Genre              0
country        11897
show_id            0
type               0
date_added       158
release_year       0
rating            67
duration           0
dtype: int64
```

In [20]:
```
#Ratings can't be in min, so it has been made NR(i.e- Non Rated)
df_final.loc[df_final['rating'].str.contains('min', na=False),'rating']='NR'
df_final['rating'].fillna('NR',inplace=True)
pd.set_option('display.max_rows',None)
```

In [21]:
```
df_final.isnull().sum()
```

Out[21]:
```
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
```

In [22]:
```
#Attempt to observe nulls in date_added column
df_final[df_final['date_added'].isnull()].head()
```

Out[22]:

	title	Actors	Directors	Genre	country	show_id	type	date_added	release_year	rating	duration
136893	A Young Doctor's Notebook and Other Stories	Daniel Radcliffe	Unknown Director	British TV Shows	United Kingdom	s6067	TV Show	NaN	2013	TV-MA	2 Seasons
136894	A Young Doctor's Notebook and Other Stories	Daniel Radcliffe	Unknown Director	TV Comedies	United Kingdom	s6067	TV Show	NaN	2013	TV-MA	2 Seasons
136895	A Young Doctor's Notebook and Other Stories	Daniel Radcliffe	Unknown Director	TV Dramas	United Kingdom	s6067	TV Show	NaN	2013	TV-MA	2 Seasons
136896	A Young Doctor's Notebook and Other Stories	Jon Hamm	Unknown Director	British TV Shows	United Kingdom	s6067	TV Show	NaN	2013	TV-MA	2 Seasons
136897	A Young Doctor's Notebook and Other Stories	Jon Hamm	Unknown Director	TV Comedies	United Kingdom	s6067	TV Show	NaN	2013	TV-MA	2 Seasons

In [23]:
```
#date added column is imputed on the basis of release year,i.e- suppose there's a null for date_added
#when release year was 2013.So below piece of code just checks the mode of date added for release year=2013
# and imputes in place of nulls the corresponding mode

for i in df_final[df_final['date_added'].isnull()]['release_year'].unique():
  imp=df_final[df_final['release_year']==i]['date_added'].mode().values[0]
  df_final.loc[df_final['release_year']==i,'date_added']=df_final.loc[df_final['release_year']==i,'date_added'].fillna(imp)
```

So we imputed the country column on the basis of directors whose other movie titles had countries given. But there might be directors who have only one occurence in our data. In that scenario, I have used Actors as a basis. i.e- for this Actor majorly acts in movies of which country? Imputation has been done on this basis. For remaining rows, country has been filled as Unknown Country

In [24]:
```
# Step 1: Create a mapping of actors to their most frequent country (mode)
actor_country_map = (
    df_final
    .dropna(subset=['country'])  # Only consider rows where country is not null
    .groupby('Actors')['country']
    .agg(lambda x: x.mode()[0] if not x.mode().empty else 'Unknown Country')
    .to_dict()
)

# Step 2: Use map to efficiently update the 'country' column based on the actor
df_final['country'] = df_final.apply(
    lambda row: actor_country_map.get(row['Actors'], 'Unknown Country')
    if pd.isnull(row['country']) else row['country'], axis=1
)

# Step 3: Replace any remaining null values with 'Unknown Country'
df_final['country'].fillna('Unknown Country', inplace=True)

# Step 4: Check the result
print(df_final.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
dtype: int64
```

In [25]:
```
df_final.head()
```

Out[25]:

	title	Actors	Directors	Genre	country	show_id	type	date_added	release_year	rating	duration
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States	s1	Movie	September 25, 2021	2020	PG-13	90 min
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
2	Blood & Water	Ama Qamata	Unknown Director	TV Dramas	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
3	Blood & Water	Ama Qamata	Unknown Director	TV Mysteries	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
4	Blood & Water	Khosi Ngema	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons

In [26]:
```
df_final['duration'].value_counts()
```

Out[26]:
```
1 Season      35035
2 Seasons      9559
3 Seasons      5084
94 min         4343
106 min        4040
97 min         3624
95 min         3560
96 min         3484
93 min         3480
90 min         3305
105 min        3209
107 min        3103
101 min        3048
102 min        3017
103 min        2985
98 min         2984
99 min         2956
91 min         2915
92 min         2863
```

In [27]:
```
#removing mins from data
df_final['duration']=df_final['duration'].str.replace(" min","")
df_final.head()
```

Out[27]:

	title	Actors	Directors	Genre	country	show_id	type	date_added	release_year	rating	duration
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States	s1	Movie	September 25, 2021	2020	PG-13	90
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
2	Blood & Water	Ama Qamata	Unknown Director	TV Dramas	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
3	Blood & Water	Ama Qamata	Unknown Director	TV Mysteries	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
4	Blood & Water	Khosi Ngema	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons

In [30]:
```
df_final['duration'].unique()
```

Out[30]:
```
array(['90', '2 Seasons', '1 Season', '91', '125', '9 Seasons', '104',
       '127', '4 Seasons', '67', '94', '5 Seasons', '161', '61', '166',
       '147', '103', '97', '106', '111', '3 Seasons', '110', '105', '96',
       '124', '116', '98', '23', '115', '122', '99', '88', '100',
       '6 Seasons', '102', '93', '95', '85', '83', '113', '13', '182',
       '48', '145', '87', '92', '80', '117', '128', '119', '143', '114',
       '118', '108', '63', '121', '142', '154', '120', '82', '109', '101',
       '86', '229', '76', '89', '156', '112', '107', '129', '135', '136',
       '165', '150', '133', '70', '84', '140', '78', '7 Seasons', '64',
       '59', '139', '69', '148', '189', '141', '130', '138', '81', '132',
       '10 Seasons', '123', '65', '68', '66', '62', '74', '131', '39',
       '46', '38', '8 Seasons', '17 Seasons', '126', '155', '159', '137',
       '12', '273', '36', '34', '77', '60', '49', '58', '72', '204',
       '212', '25', '73', '29', '47', '32', '35', '71', '149', '33', '15',
       '54', '224', '162', '37', '75', '79', '55', '158', '164', '173',
       '181', '185', '21', '24', '51', '151', '42', '22', '134', '177',
       '13 Seasons', '52', '14', '53', '8', '57', '28', '50', '9', '26',
       '45', '171', '27', '44', '146', '20', '157', '17', '203', '41',
       '30', '194', '15 Seasons', '233', '237', '230', '195', '253',
       '152', '190', '160', '208', '180', '144', '5', '174', '170', '192',
       '209', '187', '172', '16', '186', '11', '193', '176', '56', '169',
       '40', '10', '3', '168', '312', '153', '214', '31', '163', '19',
       '12 Seasons', '179', '11 Seasons', '43', '200', '196', '167',
       '178', '228', '18', '205', '201', '191'], dtype=object)
```

In [31]:
```
df_final['duration_copy']=df_final['duration'].copy()
df_final1=df_final.copy()
```

In [32]:
```
df_final1.loc[df_final1['duration_copy'].str.contains('Season'),'duration_copy']=0
df_final1['duration_copy']=df_final1['duration_copy'].astype('int')
df_final1.head()
```

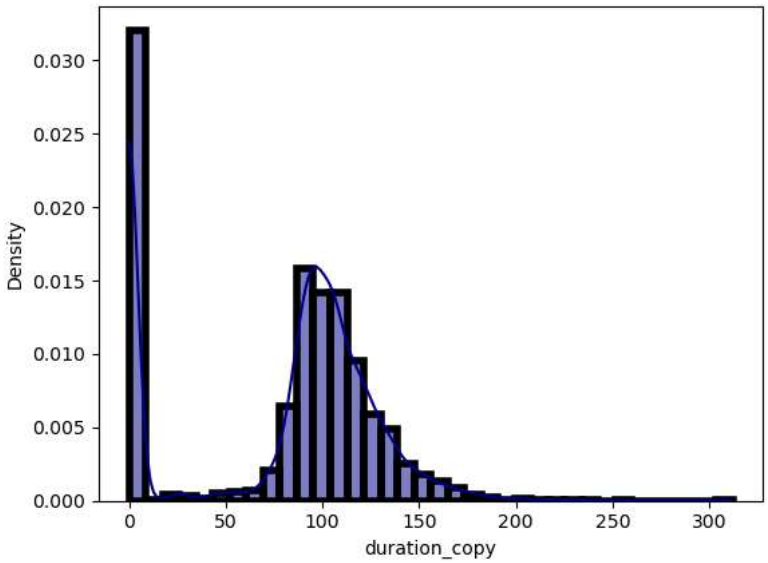
Out[32]:

	title	Actors	Directors	Genre	country	show_id	type	date_added	release_year	rating	duration	duration_copy
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States	s1	Movie	September 25, 2021	2020	PG-13	90	90
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	0
2	Blood & Water	Ama Qamata	Unknown Director	TV Dramas	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	0
3	Blood & Water	Ama Qamata	Unknown Director	TV Mysteries	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	0
4	Blood & Water	Khosi Ngema	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	0

In [33]:
```
df_final1['duration_copy'].describe()
```

Out[33]:
```
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: duration_copy, dtype: float64
```

In [40]:
```
sns.histplot(data=df_final1, x='duration_copy', bins=36, color='darkblue',
             kde=True, stat='density', edgecolor='black', linewidth=4)
plt.show()
```



In [41]:
```
bins1 = [-1,1,50,80,100,120,150,200,315]
labels1 = ['<1','1-50','50-80','80-100','100-120','120-150','150-200','200-315']
df_final1['duration_copy'] = pd.cut(df_final1['duration_copy'],bins=bins1,labels=labels1)
df_final1.head()
```

Out[41]:

	title	Actors	Directors	Genre	country	show_id	type	date_added	release_year	rating	duration	duration_copy
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States	s1	Movie	September 25, 2021	2020	PG-13	90	80-100
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	<1
2	Blood & Water	Ama Qamata	Unknown Director	TV Dramas	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	<1
3	Blood & Water	Ama Qamata	Unknown Director	TV Mysteries	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	<1
4	Blood & Water	Khosi Ngema	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	<1

In [42]:
```
df_final1.loc[~df_final1['duration'].str.contains('Season'),'duration']=df_final1.loc[~df_final1['duration'].str.contains('Seasor
df_final1.drop(['duration_copy'],axis=1,inplace=True)
df_final1.head()
```

Out[42]:

	title	Actors	Directors	Genre	country	show_id	type	date_added	release_year	rating	duration
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States	s1	Movie	September 25, 2021	2020	PG-13	80-100
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
2	Blood & Water	Ama Qamata	Unknown Director	TV Dramas	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
3	Blood & Water	Ama Qamata	Unknown Director	TV Mysteries	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
4	Blood & Water	Khosi Ngema	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons

In [43]:
```
df_final1['duration'].value_counts()
```

Out[43]:
```
80-100        52937
100-120       48724
1 Season      35035
120-150       26691
2 Seasons      9559
50-80          7700
150-200        6737
3 Seasons      5084
1-50           2530
4 Seasons      2134
5 Seasons      1698
7 Seasons       843
6 Seasons       633
200-315         524
8 Seasons       286
9 Seasons       257
10 Seasons      220
13 Seasons      132
12 Seasons      111
15 Seasons       96
17 Seasons       30
11 Seasons       30
Name: duration, dtype: int64
```

In [44]:
```
from datetime import datetime

# Directly convert 'date_added' column to datetime format
df_final1['Modified_Added_date'] = pd.to_datetime(df_final1['date_added'], errors='coerce')

# Extract month, week and year efficiently
df_final1['month_added'] = df_final1['Modified_Added_date'].dt.month
df_final1['week_Added'] = df_final1['Modified_Added_date'].dt.isocalendar().week
df_final1['year'] = df_final1['Modified_Added_date'].dt.year

# Display the first few rows
df_final1.head()
```

Out[44]:

	title	Actors	Directors	Genre	country	show_id	type	date_added	release_year	rating	duration	Modified_Added_date	month_added	week_
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States	s1	Movie	September 25, 2021	2020	PG-13	80-100	2021-09-25	9	
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	2021-09-24	9	
2	Blood & Water	Ama Qamata	Unknown Director	TV Dramas	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	2021-09-24	9	
3	Blood & Water	Ama Qamata	Unknown Director	TV Mysteries	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	2021-09-24	9	
4	Blood & Water	Khosi Ngema	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	2021-09-24	9	

In [47]:
```
#Titles such as Bahubali(Hindi Version),Bahubali(Tamil Version) were there. Since it's only one movie in different languages,
#presence of brackets and content between brackets is removed.
df_final1['title'] = df_final1['title'].str.replace(r"\(.*\)", "", regex=True)
df_final1.head()
```

Out[47]:

	title	Actors	Directors	Genre	country	show_id	type	date_added	release_year	rating	duration	Modified_Added_date	month_added	week
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States	s1	Movie	September 25, 2021	2020	PG-13	80-100	2021-09-25	9	
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	2021-09-24	9	
2	Blood & Water	Ama Qamata	Unknown Director	TV Dramas	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	2021-09-24	9	
3	Blood & Water	Ama Qamata	Unknown Director	TV Mysteries	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	2021-09-24	9	
4	Blood & Water	Khosi Ngema	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	2021-09-24	9	

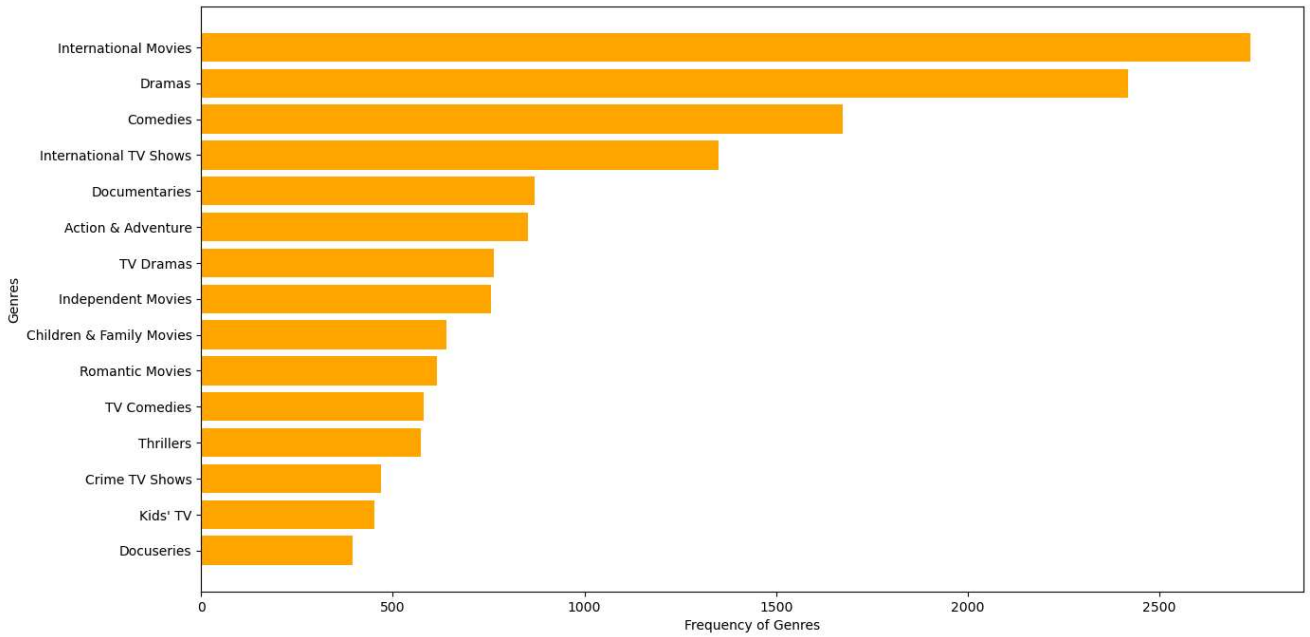
Univariate Analysis in terms of counts of each column

In [48]:
```
#number of distinct titles on the basis of genre
df_final1.groupby(['Genre']).agg({"title":"nunique"})
```

Out[48]:

Genre	title
Action & Adventure	854
Anime Features	71
Anime Series	176
British TV Shows	253
Children & Family Movies	639
Classic & Cult TV	28
Classic Movies	116
Comedies	1673
Crime TV Shows	470
Cult Movies	71
Documentaries	869
Docuseries	395
Dramas	2418
Faith & Spirituality	65
Horror Movies	353
Independent Movies	756
International Movies	2738
International TV Shows	1351
Kids' TV	451
Korean TV Shows	151
LGBTQ Movies	102
Movies	57
Music & Musicals	372
Reality TV	255
Romantic Movies	615
Romantic TV Shows	370
Sci-Fi & Fantasy	243
Science & Nature TV	92
Spanish-Language TV Shows	174
Sports Movies	219
Stand-Up Comedy	343
Stand-Up Comedy & Talk Shows	56
TV Action & Adventure	168
TV Comedies	581
TV Dramas	763
TV Horror	75
TV Mysteries	98
TV Sci-Fi & Fantasy	84
TV Shows	16
TV Thrillers	57
Teen TV Shows	69
Thrillers	573

In [49]:
```
df_genre=df_final1.groupby(['Genre']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
plt.figure(figsize=(15,8))
plt.barh(df_genre[::-1]['Genre'], df_genre[::-1]['title'],color=['orange'])
plt.xlabel('Frequency of Genres')
plt.ylabel('Genres')
plt.show()
```



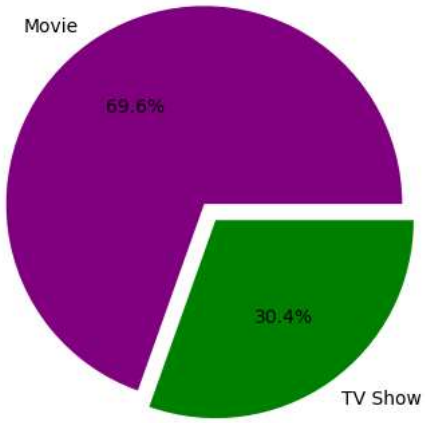
International Movies, Dramas and Comedies are the most popular

In [50]:
```
#number of distinct titles on the basis of type
df_final1.groupby(['type']).agg({"title":"nunique"})
```

Out[50]:

type	title
Movie	6115
TV Show	2676

In [51]:
```
df_type=df_final1.groupby(['type']).agg({"title":"nunique"}).reset_index()
plt.pie(df_type['title'],explode=(0.05,0.05), labels=df_type['type'],colors=['purple','green'],autopct='%.1f%%')
plt.show()
```



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

In [52]:
```
#number of distinct titles on the basis of country
df_final1.groupby(['country']).agg({"title":"nunique"})
```

Angola	2
Argentina	98
Armenia	1
Australia	165
Austria	13
Azerbaijan	1
Bahamas	1
Bangladesh	5
Belarus	1
Belgium	97
Bermuda	1
Botswana	1
Brazil	104

The above dataframe shows a flaw in which we are seeing countries, such as Cambodia and Cambodia, or United States and United States, are shown as different countries.They should have been same

In [53]:
```
df_final1['country'] = df_final1['country'].str.replace(',', '')
df_final1.head()
```

Out[53]:

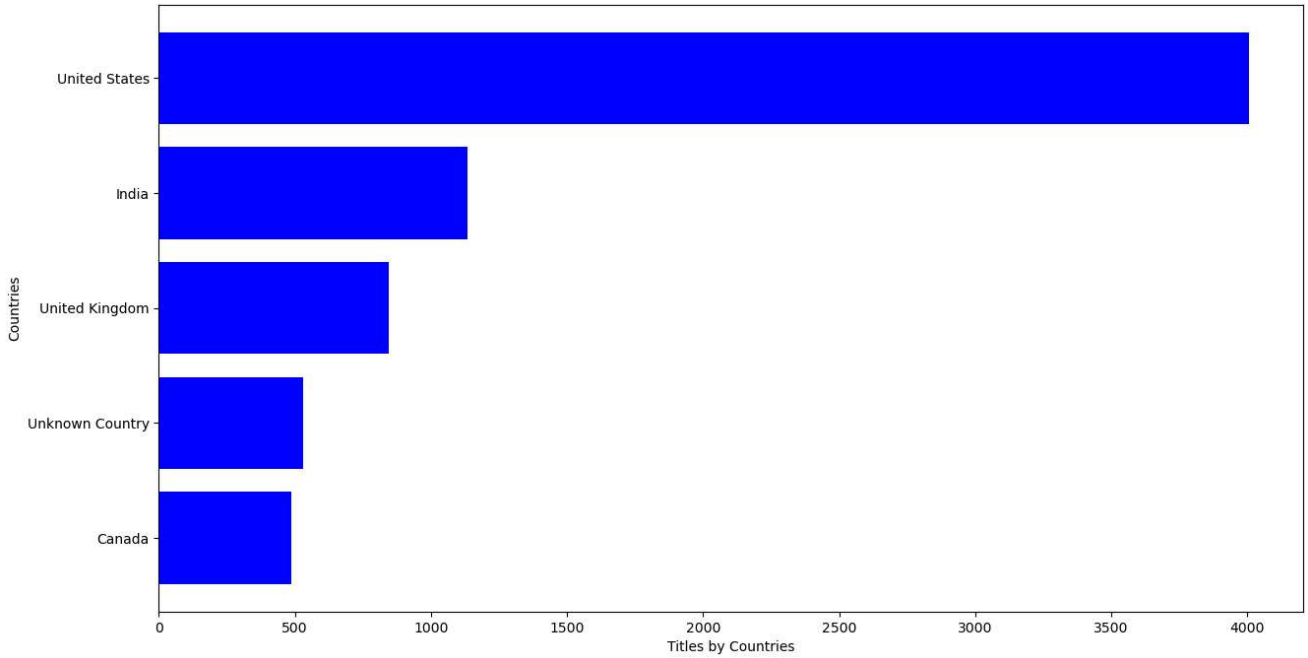
	title	Actors	Directors	Genre	country	show_id	type	date_added	release_year	rating	duration	Modified_Added_date	month_added	week
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States	s1	Movie	September 25, 2021	2020	PG-13	80-100	2021-09-25	9	
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	2021-09-24	9	
2	Blood & Water	Ama Qamata	Unknown Director	TV Dramas	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	2021-09-24	9	
3	Blood & Water	Ama Qamata	Unknown Director	TV Mysteries	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	2021-09-24	9	
4	Blood & Water	Khosi Ngema	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	2021-09-24	9	

In [54]:
```
#number of distinct titles on the basis of country
df_final1.groupby(['country']).agg({"title":"nunique"})
```

Albania	1
Algeria	3
Angola	2
Argentina	98
Armenia	1
Australia	165
Austria	13
Azerbaijan	1
Bahamas	1
Bangladesh	5
Belarus	1
Belgium	97
Bermuda	1

Now it looks great.

In [56]:
```
df_country=df_final1.groupby(['country']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:5]
plt.figure(figsize=(15,8))
plt.barh(df_country[::-1]['country'], df_country[::-1]['title'],color=['blue'])
plt.xlabel('Titles by Countries')
plt.ylabel('Countries')
plt.show()
```



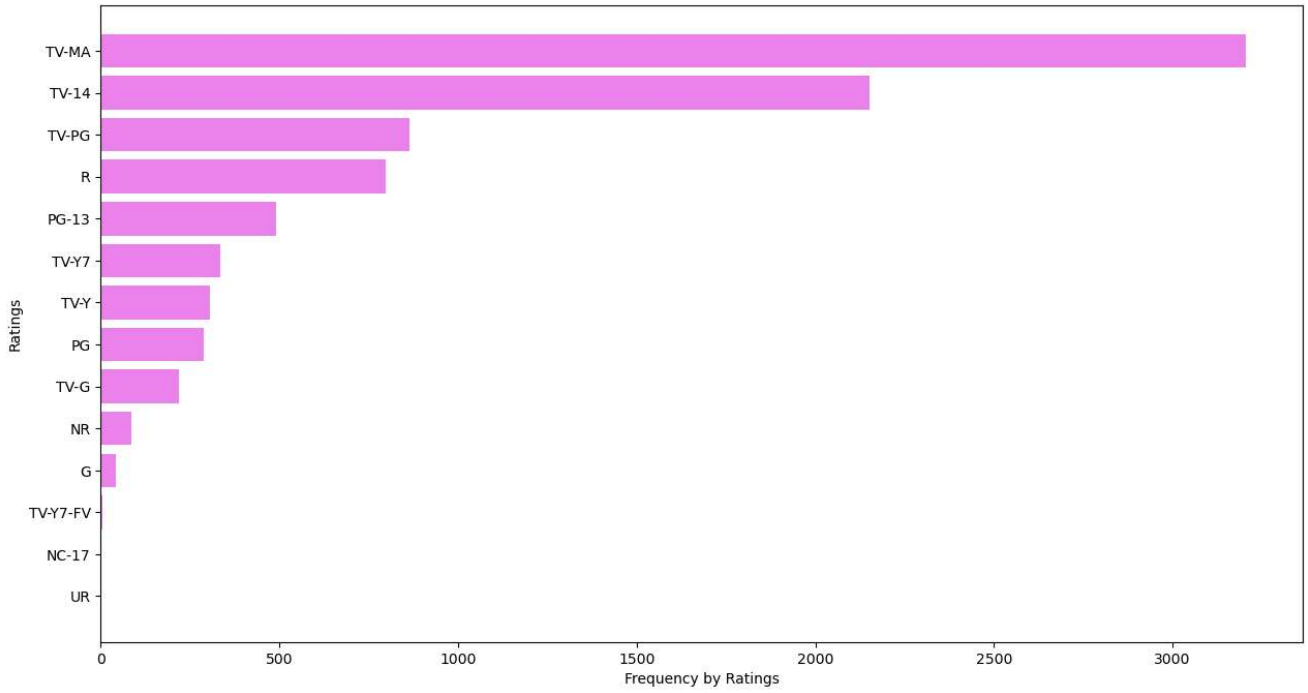
US,India,UK,Canada and France are leading countries in Content Creation on Netflix

In [58]:
```
#number of distinct titles on the basis of rating
df_final1.groupby(['rating']).agg({"title":"nunique"})
```

Out[58]:

rating	title
G	41
NC-17	3
NR	87
PG	287
PG-13	490
R	799
TV-14	2151
TV-G	220
TV-MA	3204
TV-PG	863
TV-Y	305
TV-Y7	334
TV-Y7-FV	6
UR	3

In [59]:
```
df_rating=df_final1.groupby(['rating']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
plt.figure(figsize=(15,8))
plt.barh(df_rating[::-1]['rating'], df_rating[::-1]['title'],color=['violet'])
plt.xlabel('Frequency by Ratings')
plt.ylabel('Ratings')
plt.show()
```



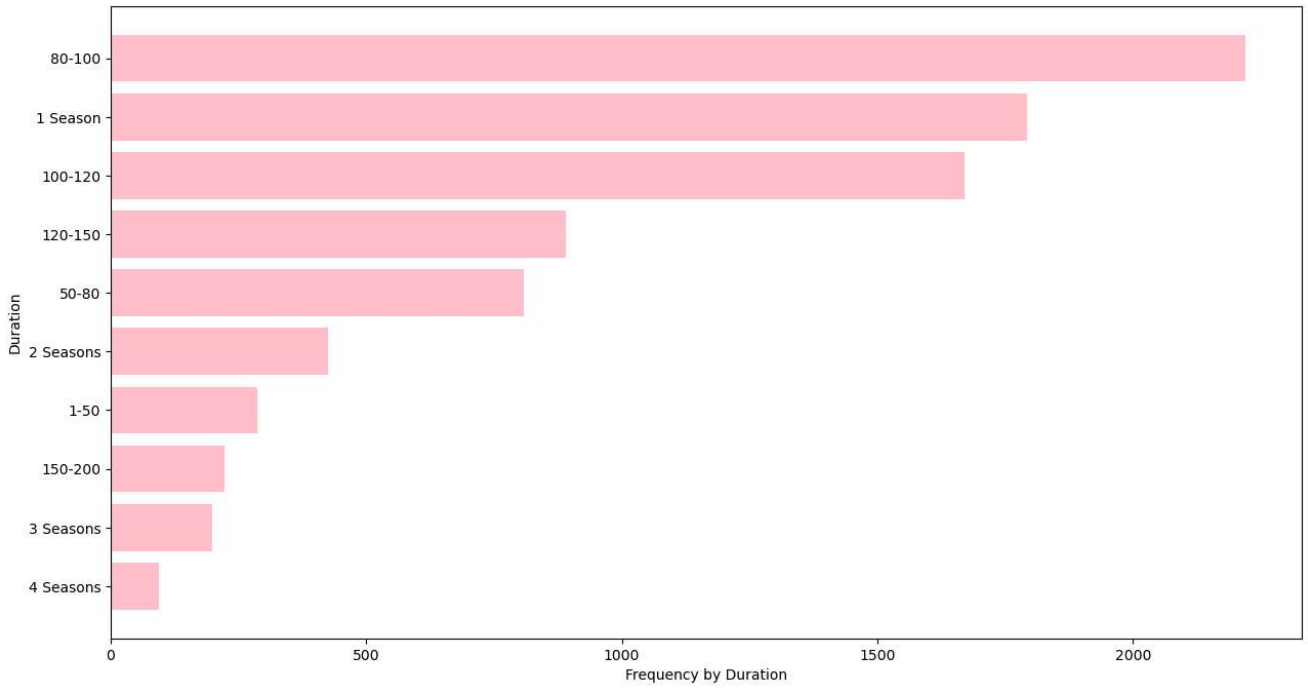
Most of the highly rated content on Netflix is intended for Mature Audiences, R Rated, content not intended for audience under 14 and those which require Parental Guidance

In [60]:
```
#number of distinct titles on the basis of duration
df_final1.groupby(['duration']).agg({"title":"nunique"})
```

Out[60]:

duration	title
1 Season	1793
1-50	287
10 Seasons	7
100-120	1671
11 Seasons	2
12 Seasons	2
120-150	891
13 Seasons	3
15 Seasons	2
150-200	222
17 Seasons	1
2 Seasons	425
200-315	19
3 Seasons	199
4 Seasons	95
5 Seasons	65
50-80	808
6 Seasons	33
7 Seasons	23
8 Seasons	17
80-100	2220
9 Seasons	9

In [61]:
```
df_duration=df_final1.groupby(['duration']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:10]
plt.figure(figsize=(15,8))
plt.barh(df_duration[::-1]['duration'], df_duration[::-1]['title'],color=['pink'])
plt.xlabel('Frequency by Duration')
plt.ylabel('Duration')
plt.show()
```

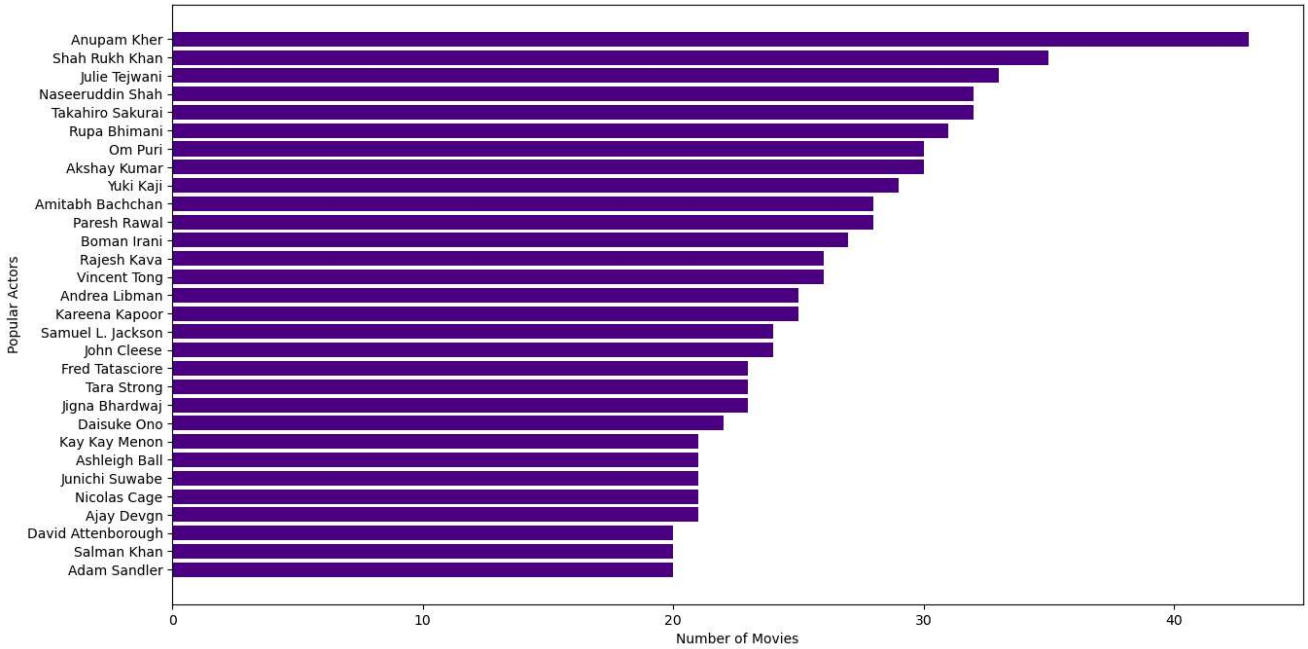


The duration of Most Watched content in our whole data is 80-100 mins.These must be movies and Shows having only 1 Season.

In [62]:
```
#number of distinct titles on the basis of Actors
df_final1.groupby(['Actors']).agg({"title":"nunique"})
```

A.D. Miles	3
A.J. Cook	2
A.J. Johnson	1
A.J. LoCascio	3
A.K. Hangal	4
A.R. Rahman	1
A.S. Sasi Kumar	1
AC Lim	1
AFRA	1
AJ Bowen	1
AJ Michalka	1
AJ Rivera	1
ARAH	2

In [63]:
```
df_actors=df_final1.groupby(['Actors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:31]
df_actors=df_actors[df_actors['Actors']!='Unknown Actor']
plt.figure(figsize=(15,8))
plt.barh(df_actors[::-1]['Actors'], df_actors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Movies')
plt.ylabel('Popular Actors')
plt.show()
```



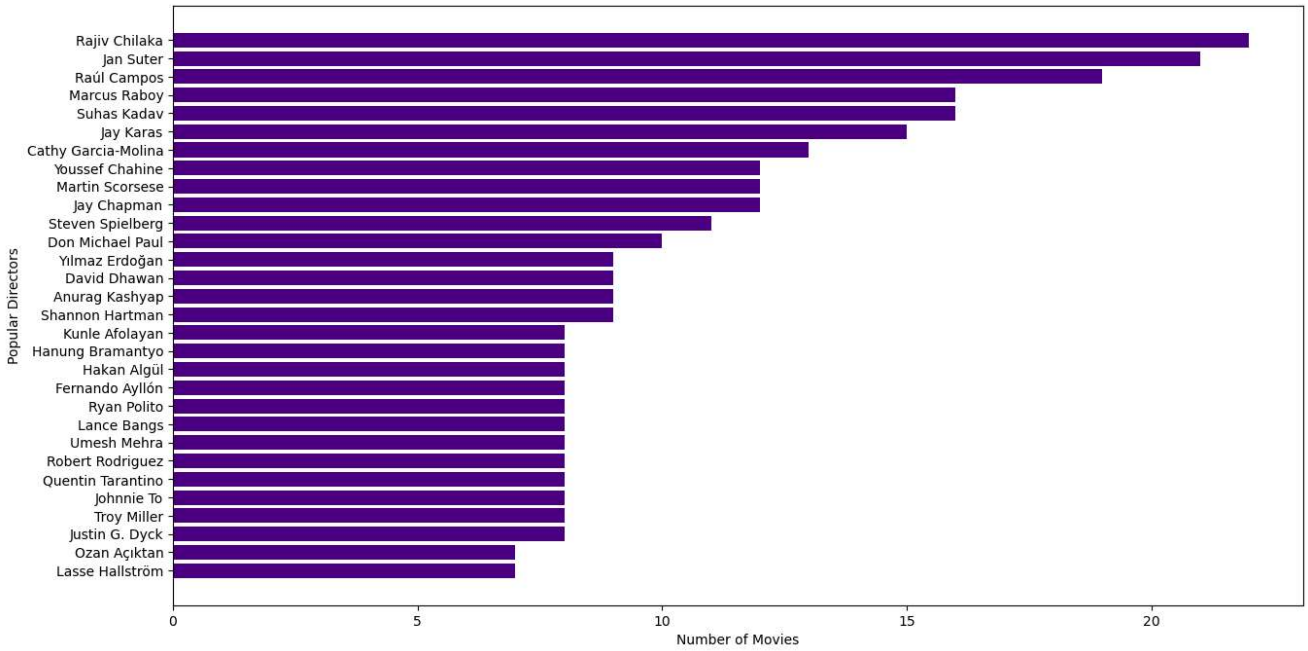
Anupam Kher,SRK,Julie Tejwani, Naseeruddin Shah and Takahiro Sakurai occupy the top stop in Most Watched content.

In [64]:
```
#number of distinct titles on the basis of Actors
df_final1.groupby(['Directors']).agg({"title":"nunique"})
```

Out[64]:

Directors	title
A. L. Vijay	2
A. Raajdheep	1
A. Salaam	1
A.R. Murugadoss	2
Aadish Keluskar	1
Aamir Bashir	1
Aamir Khan	1
Aanand Rai	1
Aaron Burns	1
Aaron Hancox	1

In [65]:
```
df_directors=df_final1.groupby(['Directors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:
df_directors=df_directors[df_directors['Directors']!='Unknown Director']
plt.figure(figsize=(15,8))
plt.barh(df_directors[::-1]['Directors'], df_directors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Movies')
plt.ylabel('Popular Directors')
plt.show()
```



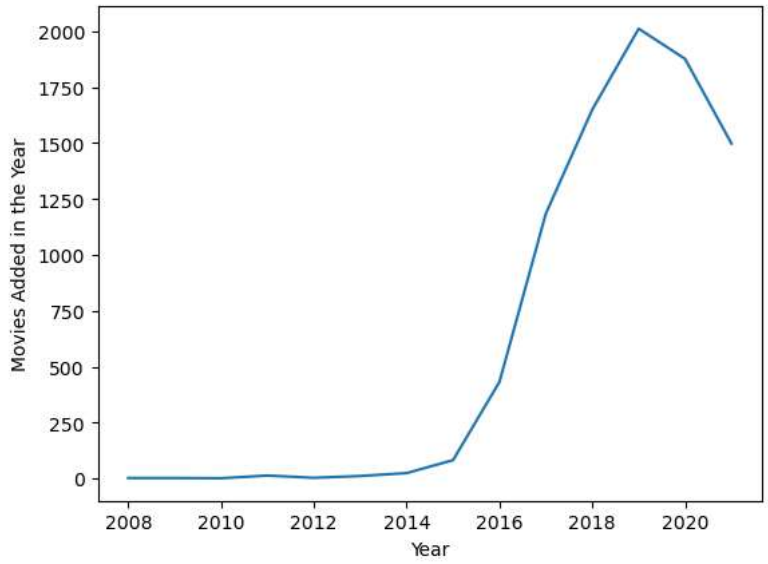
Rajiv Chilaka, Jan Suter and Raul Campos are the most popular directors across Netflix

In [66]:
```
#number of distinct titles on the basis of year
df_final1.groupby(['year']).agg({"title":"nunique"})
```

Out[66]:

year	title
2008	2
2009	2
2010	1
2011	13
2012	3
2013	11
2014	24
2015	82
2016	432
2017	1185
2018	1650
2019	2012
2020	1877
2021	1498

In [67]:
```
df_year=df_final1.groupby(['year']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_year, x='year', y='title')
plt.ylabel("Movies Added in the Year")
plt.xlabel("Year")
plt.show()
```



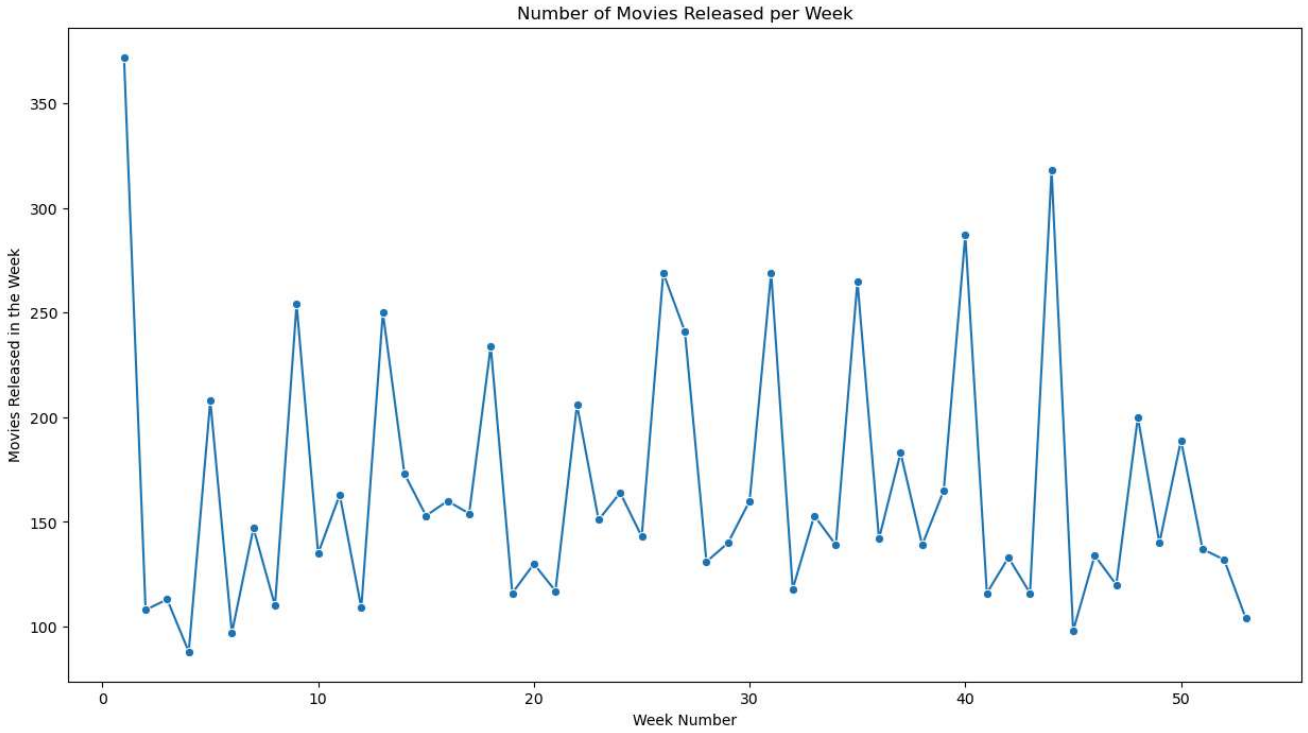
The Amount of Content across Netflix has increased from 2008 continuously till 2019. Then started decreasing from here(probably due to Covid)

In [68]:
```
#number of distinct titles on the basis of week
df_final1.groupby(['week_Added']).agg({"title":"nunique"})
```

Out[68]:

week_Added	title
1	372
2	108
3	113
4	88
5	208
6	97
7	147
8	110
9	254
10	135
11	163
12	109
13	250
14	173
15	153
16	160
17	154
18	234
19	116
20	130
21	117
22	206
23	151
24	164
25	143
26	269
27	241
28	131
29	140
30	160
31	269
32	118
33	153
34	139
35	265
36	142
37	183
38	139
39	165
40	287
41	116
42	133
43	116
44	318
45	98
46	134
47	120
48	200
49	140
50	189
51	137
52	132
53	104

In [74]:
```
# Convert 'week_Added' to numeric and drop NaNs directly
df_final1['week_Added'] = pd.to_numeric(df_final1['week_Added'], errors='coerce')
df_final1 = df_final1.dropna(subset=['week_Added'])

# Group by week and count unique titles
df_week = (
    df_final1.groupby('week_Added')['title'].nunique()
    .reset_index()
    .sort_values(by='week_Added')
)

# Ensure the week number is an integer for better visualization
df_week['week_Added'] = df_week['week_Added'].astype(int)

# Plotting the line plot
plt.figure(figsize=(15, 8))
sns.lineplot(data=df_week, x='week_Added', y='title', marker='o')
plt.ylabel("Movies Released in the Week")
plt.xlabel("Week Number")
plt.title("Number of Movies Released per Week")
plt.show()
```



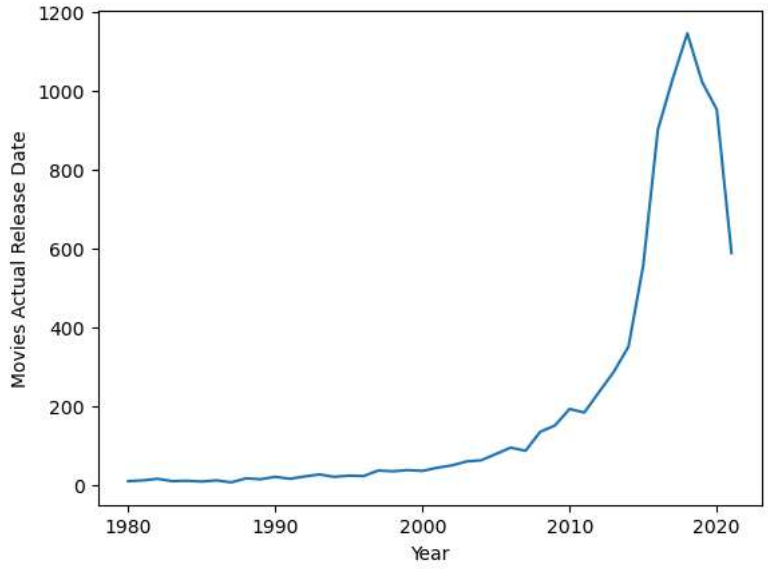
Most of the Content across Netflix is added in the first week of the year and it follows a bit of a cyclical pattern

In [75]:
```
#number of distinct titles on the basis of week
df_final1.groupby(['month_added']).agg({"title":"nunique"})
```

Out[75]:

month_added	title
1	738
2	563
3	742
4	765
5	631
6	726
7	832
8	748
9	771
10	758
11	706
12	813

In [76]:
```
df_release_year=df_final1[df_final1['release_year']>=1980].groupby(['release_year']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_release_year, x='release_year', y='title')
plt.ylabel("Movies Actual Release Date")
plt.xlabel("Year")
plt.show()
```

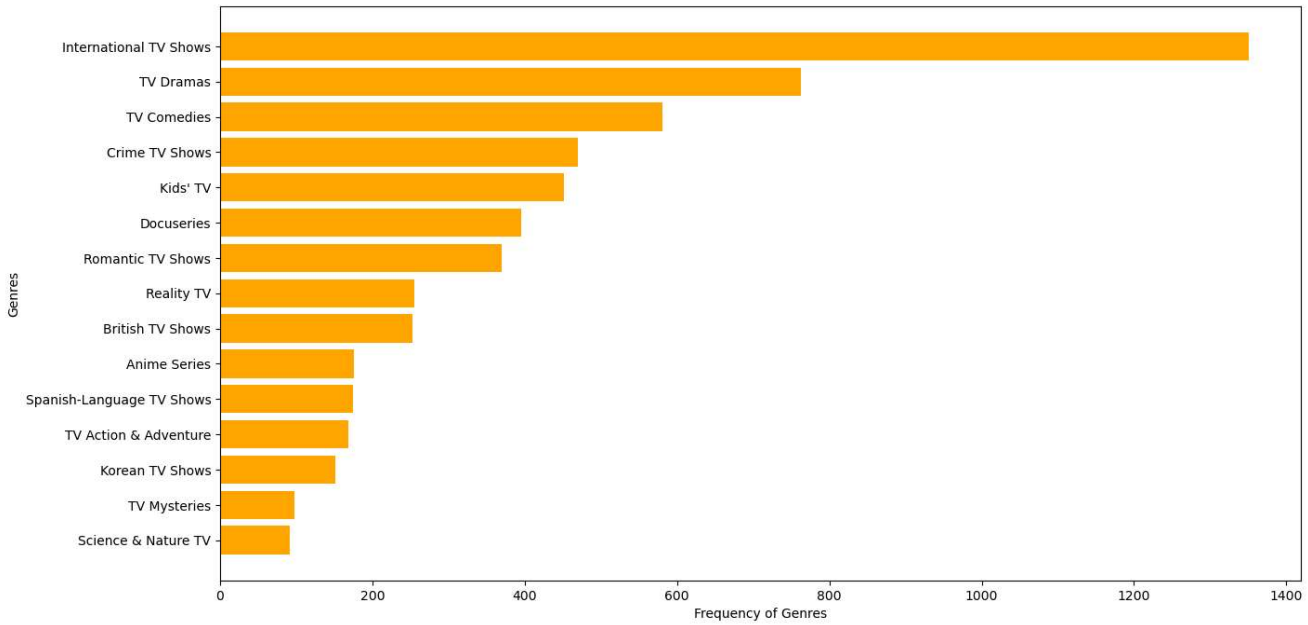


Net content release which are later uploaded to Netflix has increased since 1980 till 2020 though later reduced certainly due to COVID-19

Univariate Analysis separately for shows and movies

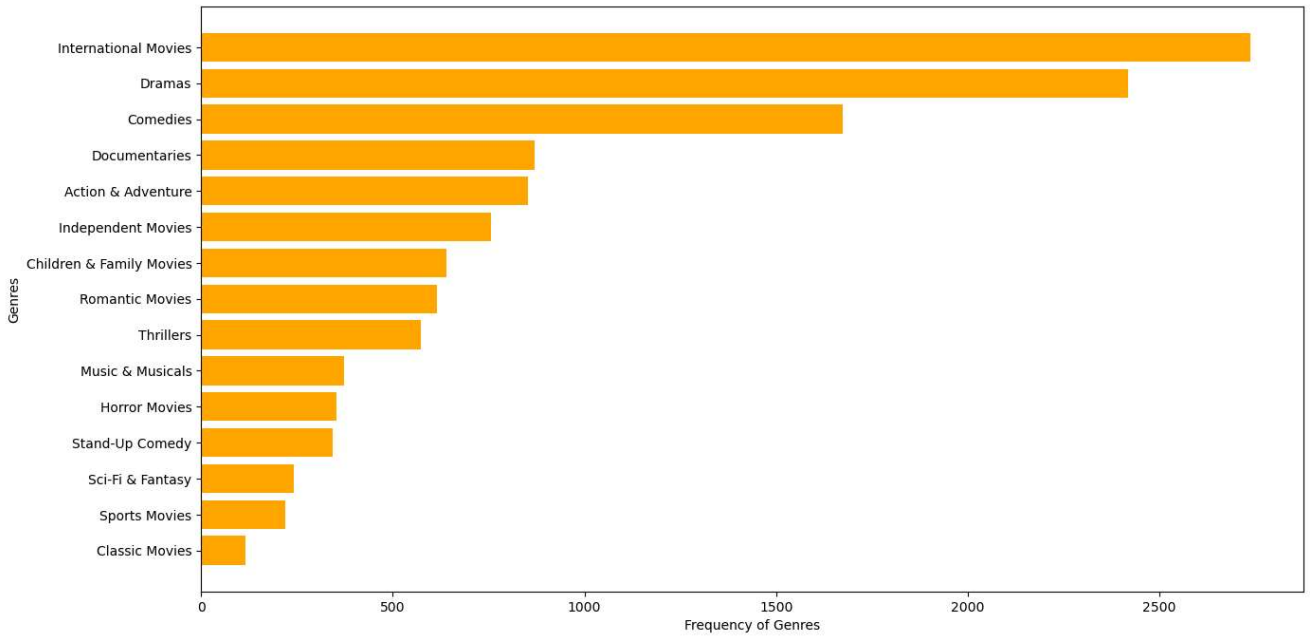
In [77]:
```
df_shows=df_final1[df_final1['type']=='TV Show']
df_movies=df_final1[df_final1['type']=='Movie']
```

In [78]:
```
df_genre=df_shows.groupby(['Genre']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
plt.figure(figsize=(15,8))
plt.barh(df_genre[::-1]['Genre'], df_genre[::-1]['title'],color=['orange'])
plt.xlabel('Frequency of Genres')
plt.ylabel('Genres')
plt.show()
```



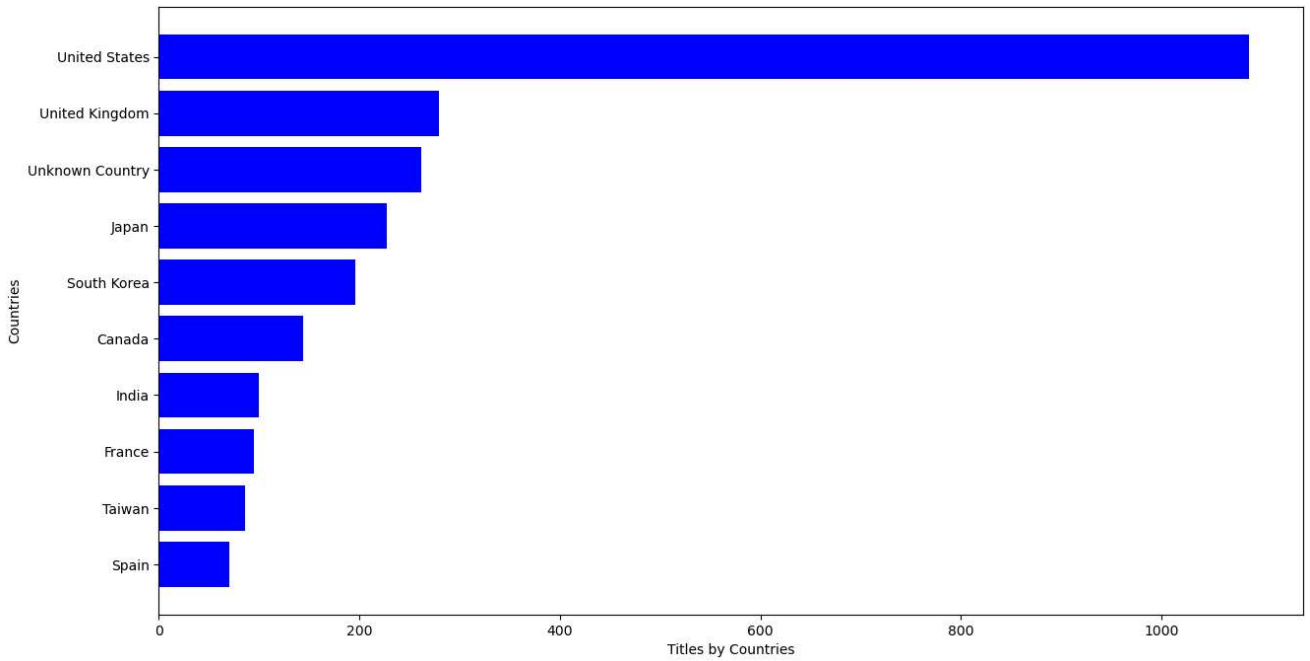
International TV Shows, Dramas and Comedy Genres are popular across TV Shows in Netflix

In [80]:
```
df_genre=df_movies.groupby(['Genre']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
plt.figure(figsize=(15,8))
plt.barh(df_genre[::-1]['Genre'], df_genre[::-1]['title'],color=['orange'])
plt.xlabel('Frequency of Genres')
plt.ylabel('Genres')
plt.show()
```

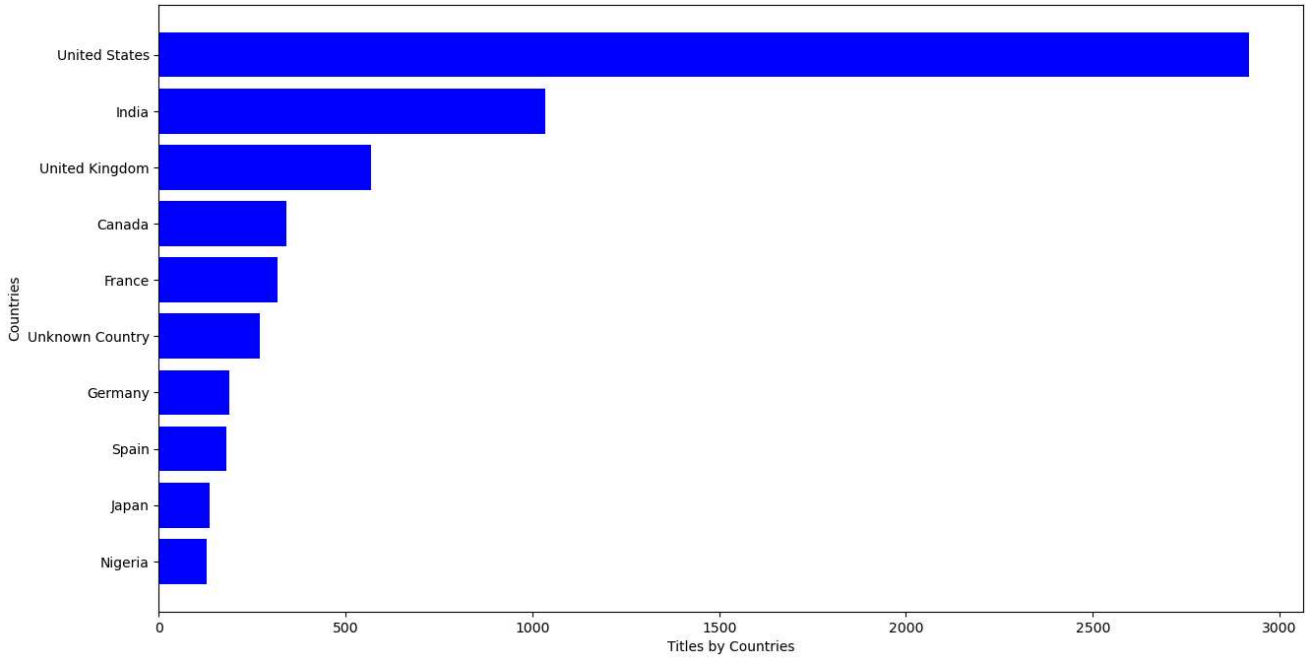


International Movies, Dramas and Comedy Genres are popular followed by Documentaries across Movies on Netflix

In [82]:
```
df_country=df_shows.groupby(['country']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:10]
plt.figure(figsize=(15,8))
plt.barh(df_country[::-1]['country'], df_country[::-1]['title'],color=['blue'])
plt.xlabel('Titles by Countries')
plt.ylabel('Countries')
plt.show()
```



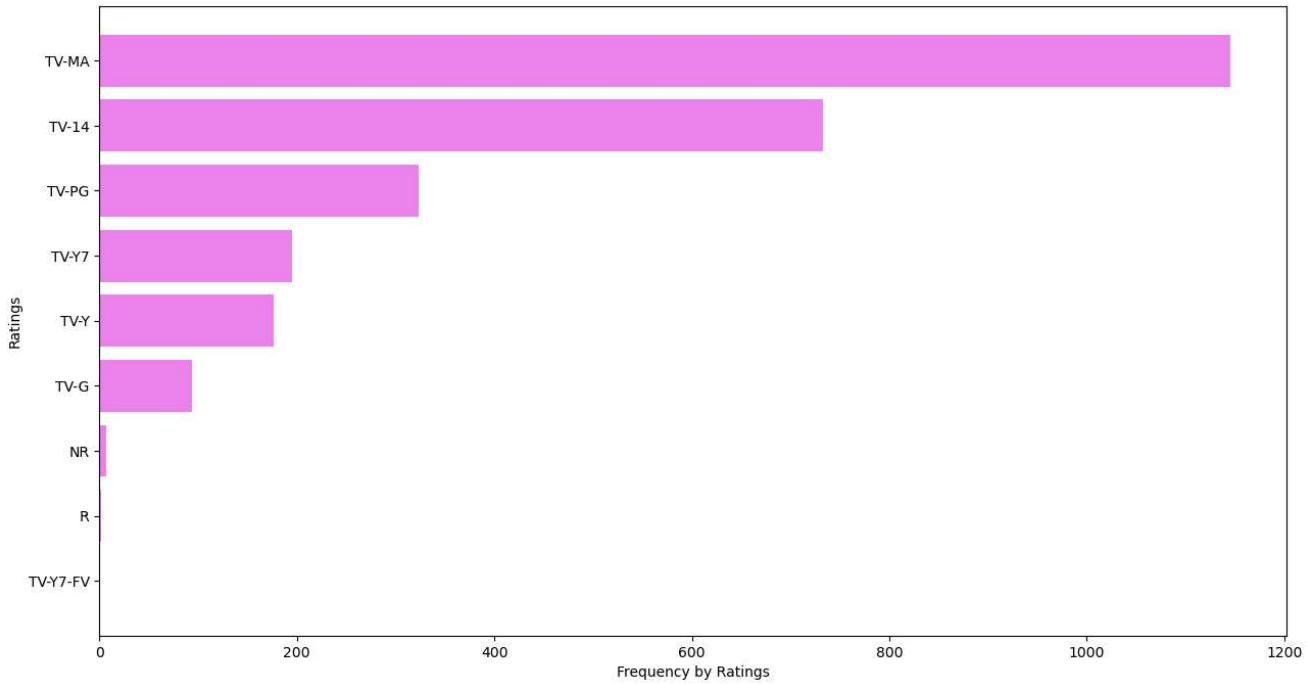
In [83]:
```
df_country=df_movies.groupby(['country']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:10]
plt.figure(figsize=(15,8))
plt.barh(df_country[::-1]['country'], df_country[::-1]['title'],color=['blue'])
plt.xlabel('Titles by Countries')
plt.ylabel('Countries')
plt.show()
```



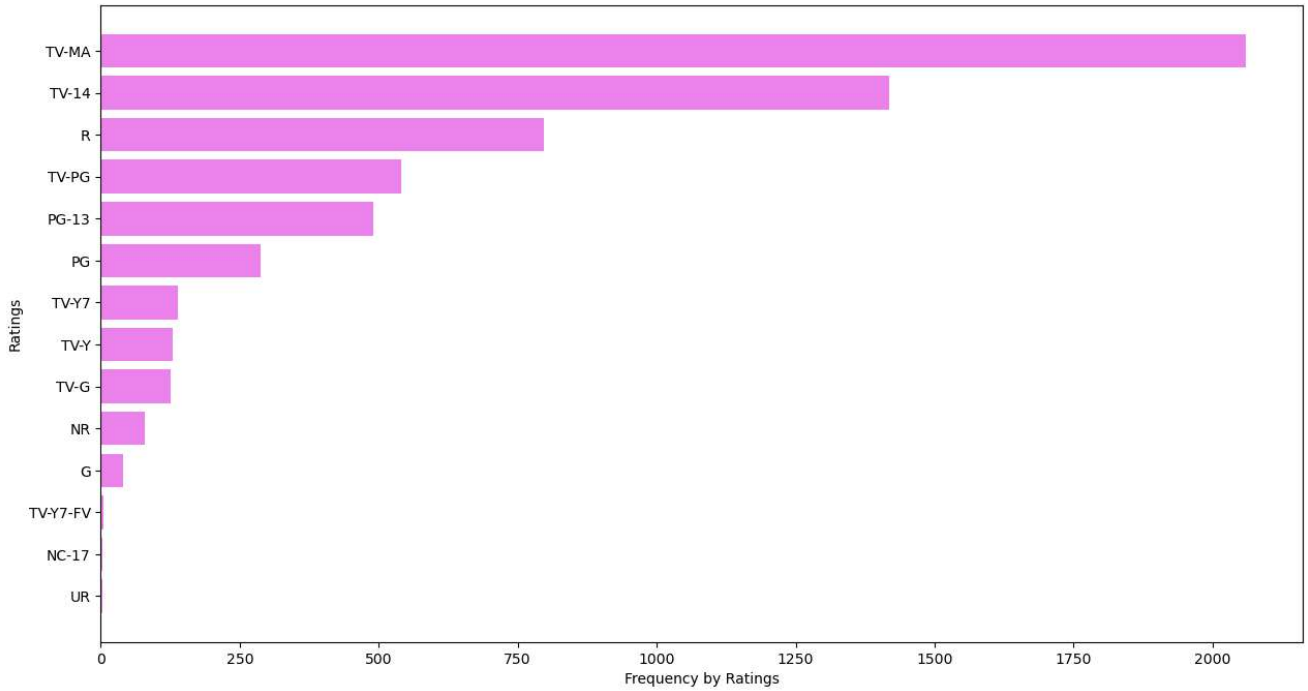
United States is leading across both TV Shows and Movies, UK also provides great content across TV Shows and Movies. Surprisingly India is much more prevalent in Movies as compared TV Shows.

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.

In [84]:
```
df_rating=df_shows.groupby(['rating']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
plt.figure(figsize=(15,8))
plt.barh(df_rating[::-1]['rating'], df_rating[::-1]['title'],color=['violet'])
plt.xlabel('Frequency by Ratings')
plt.ylabel('Ratings')
plt.show()
```



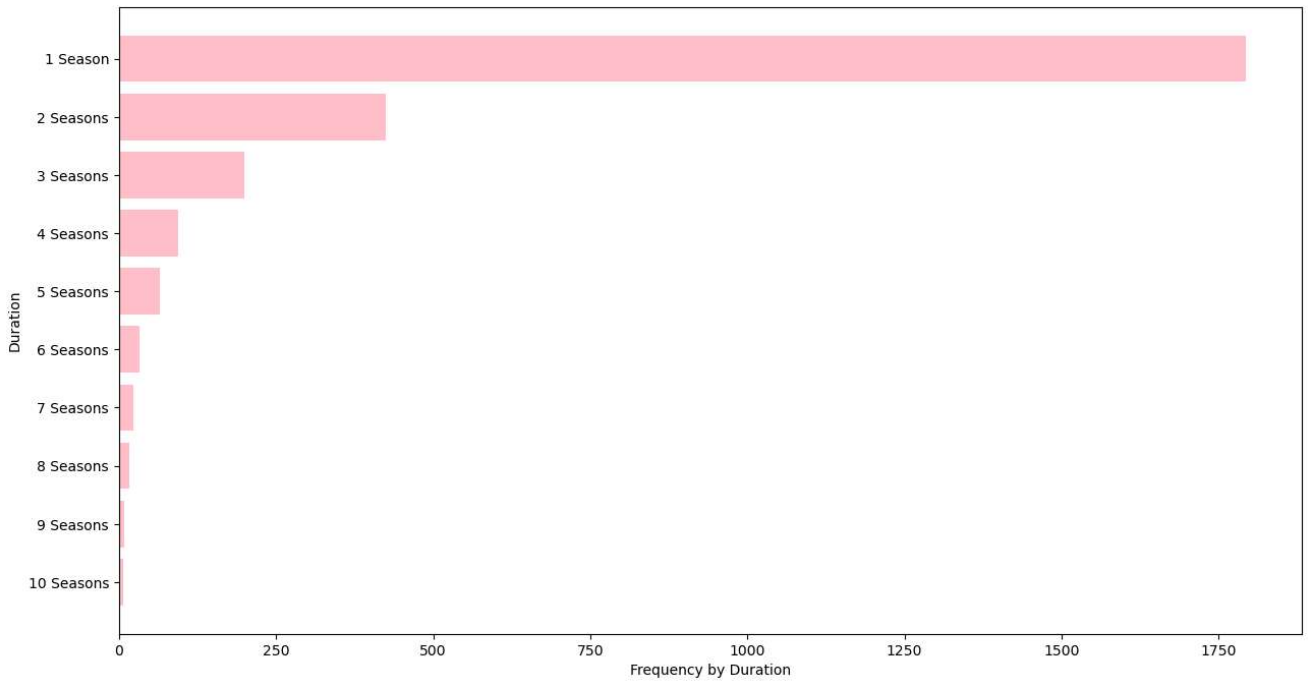
In [85]:
```
df_rating=df_movies.groupby(['rating']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
plt.figure(figsize=(15,8))
plt.barh(df_rating[::-1]['rating'], df_rating[::-1]['title'],color=['violet'])
plt.xlabel('Frequency by Ratings')
plt.ylabel('Ratings')
plt.show()
```



So it seems plaussible to conclude that the popular ratings across Netflix includes Mature Audiences and those appropriate for over 14/over 17 ages.

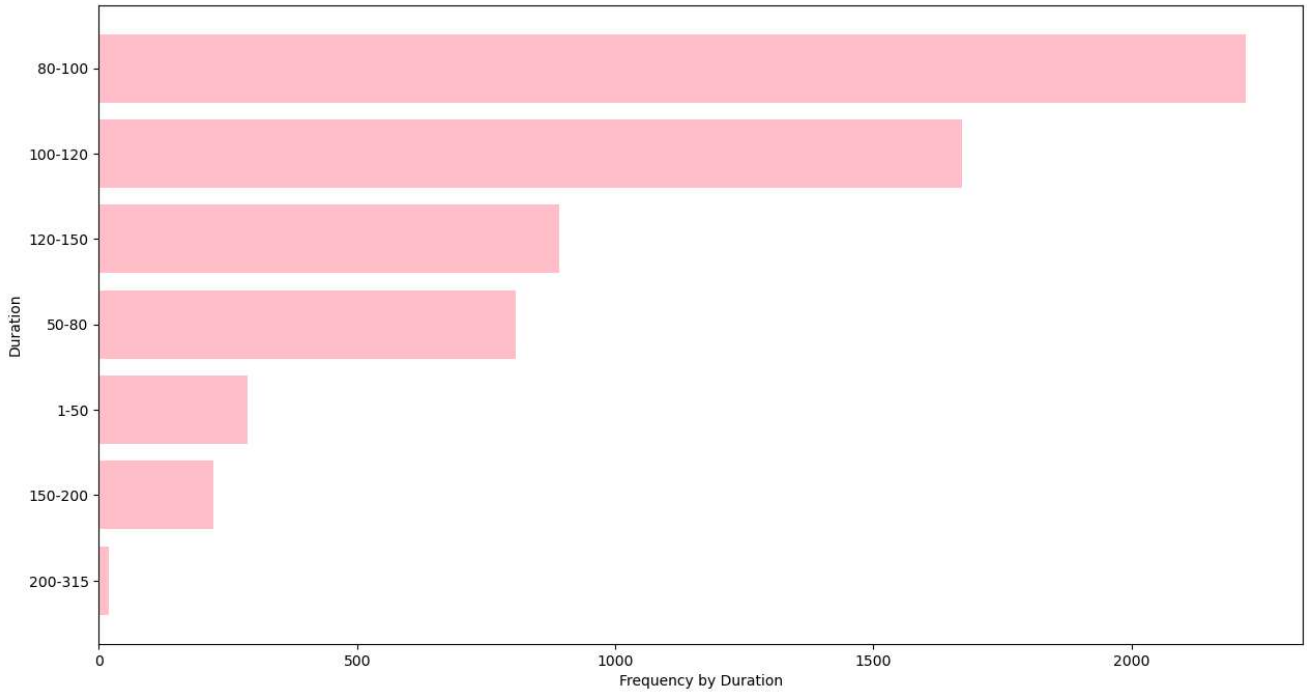
Moreover there are no TV Shows having a rating of R

In [86]:
```
df_duration=df_shows.groupby(['duration']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:10]
plt.figure(figsize=(15,8))
plt.barh(df_duration[::-1]['duration'], df_duration[::-1]['title'],color=['pink'])
plt.xlabel('Frequency by Duration')
plt.ylabel('Duration')
plt.show()
```



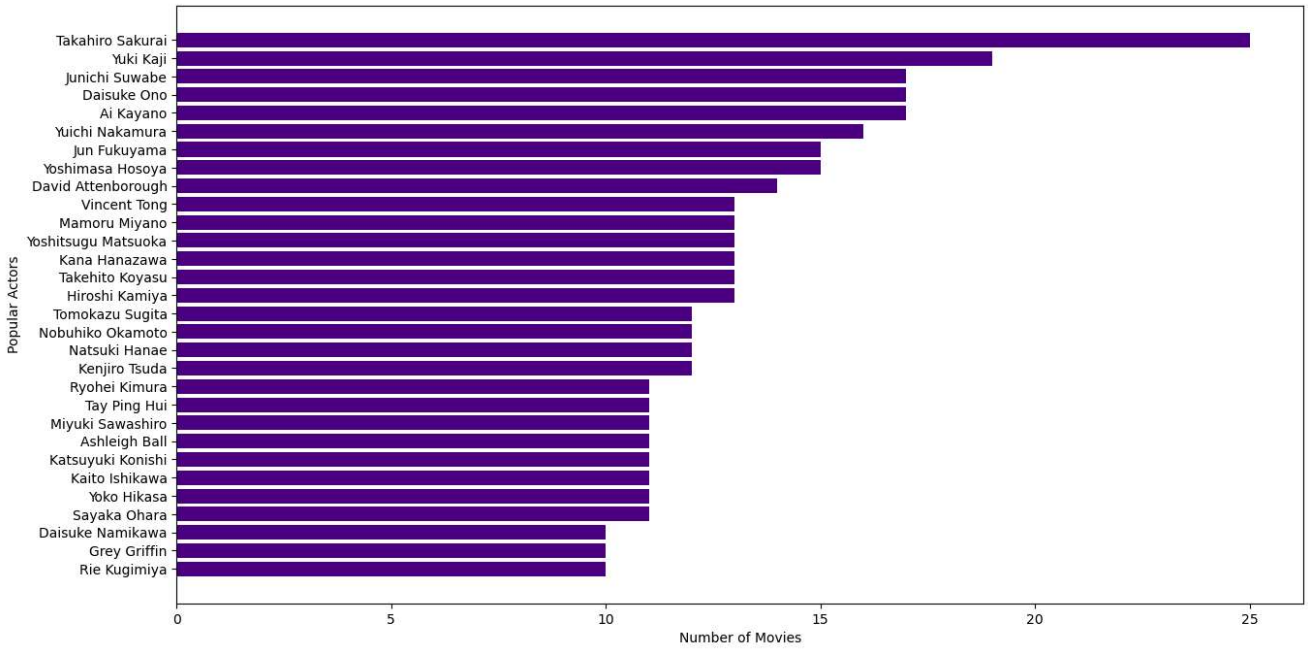
Across TV Shows, shows having only 1 Season are common as soon as the season length increases, the number of shows decrease and this definitely sounds as expected

In [88]:
```
df_duration=df_movies.groupby(['duration']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:10]
plt.figure(figsize=(15,8))
plt.barh(df_duration[::-1]['duration'], df_duration[::-1]['title'],color=['pink'])
plt.xlabel('Frequency by Duration')
plt.ylabel('Duration')
plt.show()
```



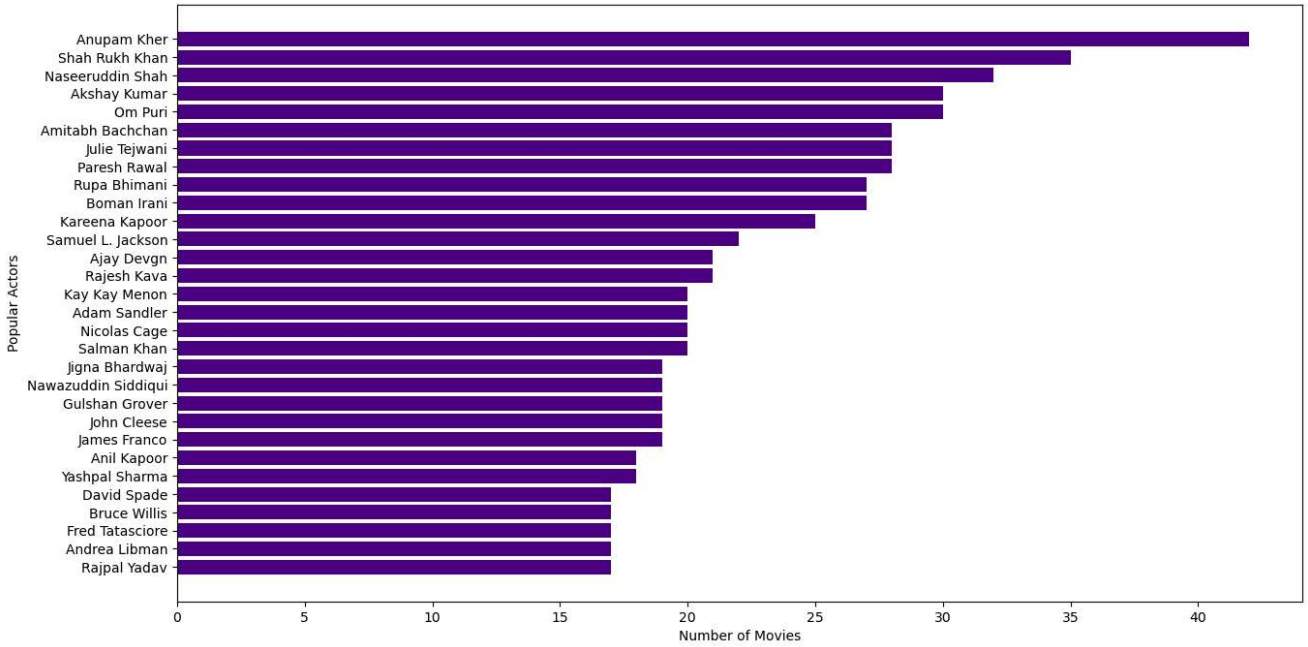
Across movies 80-100,100-120 and 120-150 is the ranges of minutes for which most movies lie. So quite possibly 80-150 mins is the sweet spot we would be wanting for movies.

In [89]:
```
df_actors=df_shows.groupby(['Actors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:31]
df_actors=df_actors[df_actors['Actors']!='Unknown Actor']
plt.figure(figsize=(15,8))
plt.barh(df_actors[::-1]['Actors'], df_actors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Movies')
plt.ylabel('Popular Actors')
plt.show()
```



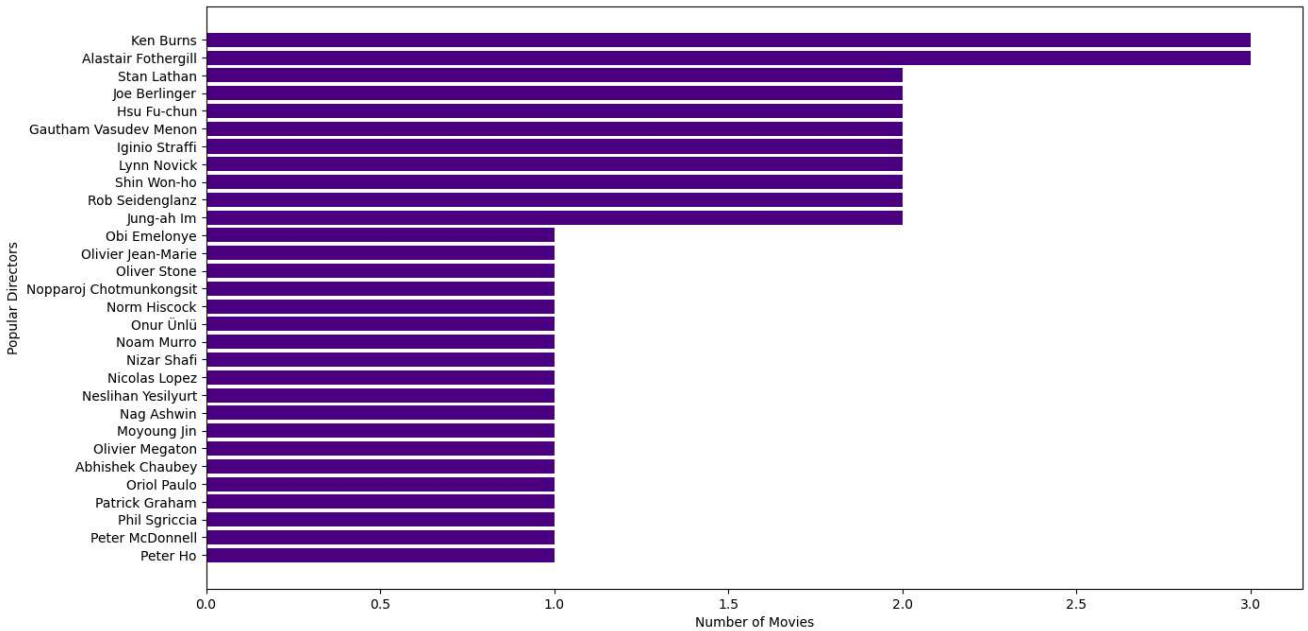
Takahiro Sakurai,Yuki Kaji and other South Korean/Japanese actors are the most popular actors across TV Shows

In [90]:
```
df_actors=df_movies.groupby(['Actors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:31]
df_actors=df_actors[df_actors['Actors']!='Unknown Actor']
plt.figure(figsize=(15,8))
plt.barh(df_actors[::-1]['Actors'], df_actors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Movies')
plt.ylabel('Popular Actors')
plt.show()
```



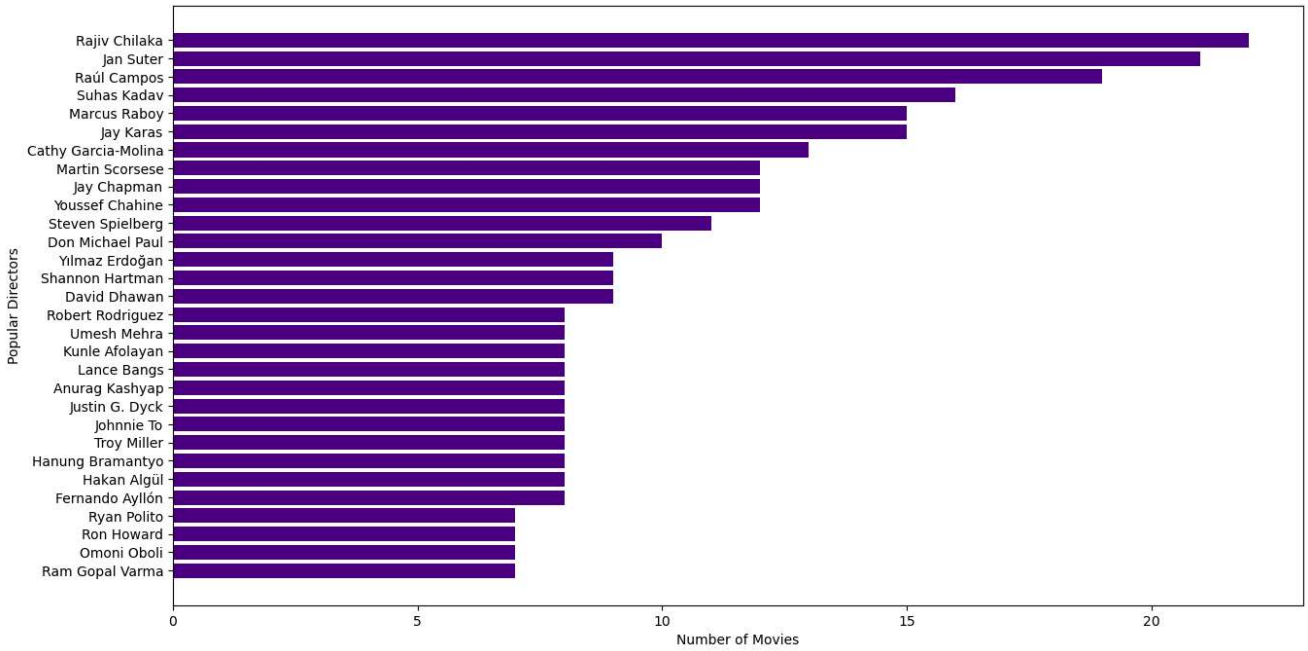
Our bollywood actors such as Anupam Kher, SRK, Naseeruddin Shah are very much popular acrossmovies on Netflix

In [91]:
```
df_directors=df_shows.groupby(['Directors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:31
df_directors=df_directors[df_directors['Directors']!='Unknown Director']
plt.figure(figsize=(15,8))
plt.barh(df_directors[::-1]['Directors'], df_directors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Movies')
plt.ylabel('Popular Directors')
plt.show()
```



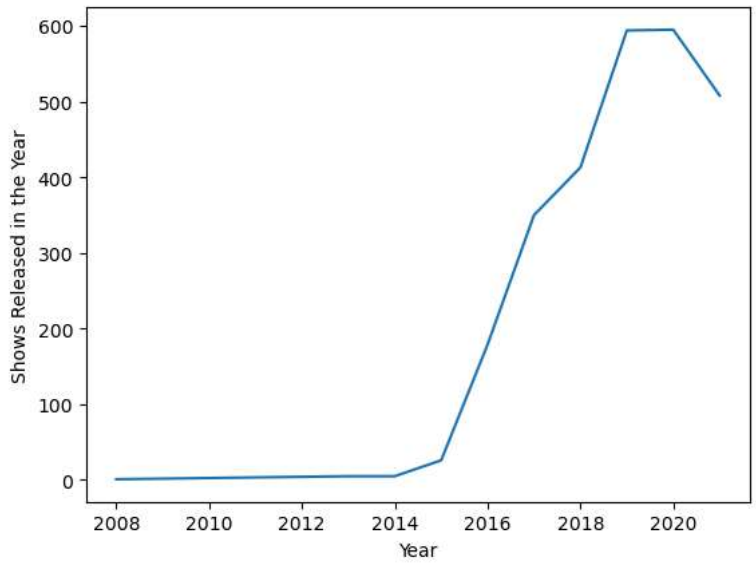
Ken Burns, Alastair Fothergill, Stan Lathan, Joe Barlinger are popular directors across TV Shows on Netflix

In [93]:
```
df_directors=df_movies.groupby(['Directors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:
df_directors=df_directors[df_directors['Directors']!='Unknown Director']
plt.figure(figsize=(15,8))
plt.barh(df_directors[::-1]['Directors'], df_directors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Movies')
plt.ylabel('Popular Directors')
plt.show()
```

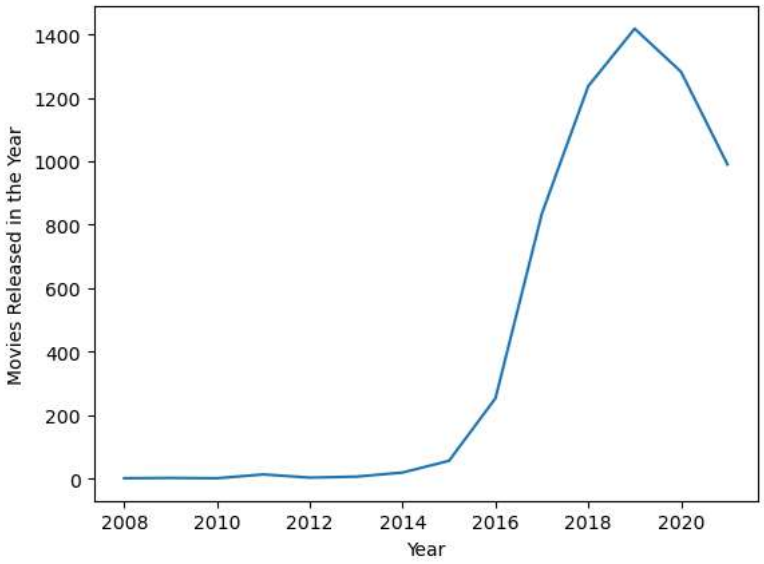


Rajiv Chilka, Jan Suter, Raul Campos, Suhas Kadav are popular directors across movies

In [95]:
```
df_year=df_shows.groupby(['year']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_year, x='year', y='title')
plt.ylabel("Shows Released in the Year")
plt.xlabel("Year")
plt.show()
```

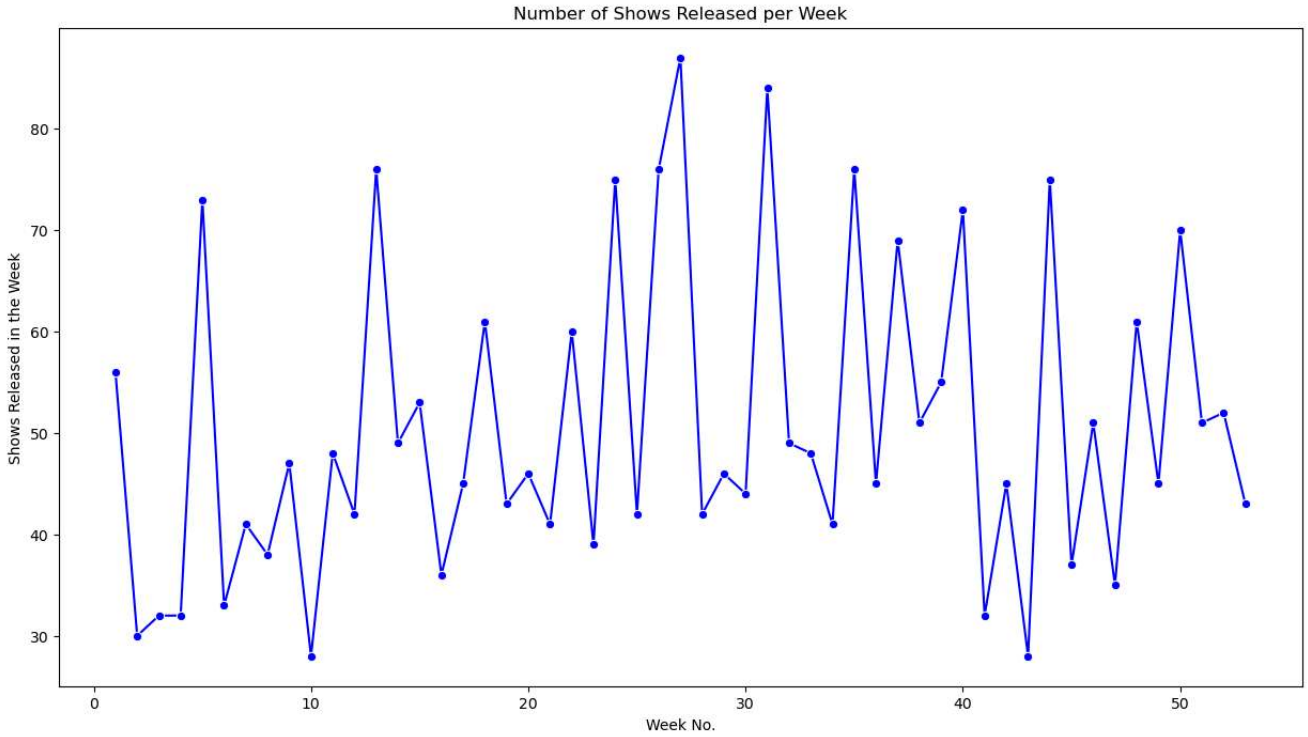


In [96]:
```
df_year=df_movies.groupby(['year']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_year, x='year', y='title')
plt.ylabel("Movies Released in the Year")
plt.xlabel("Year")
plt.show()
```

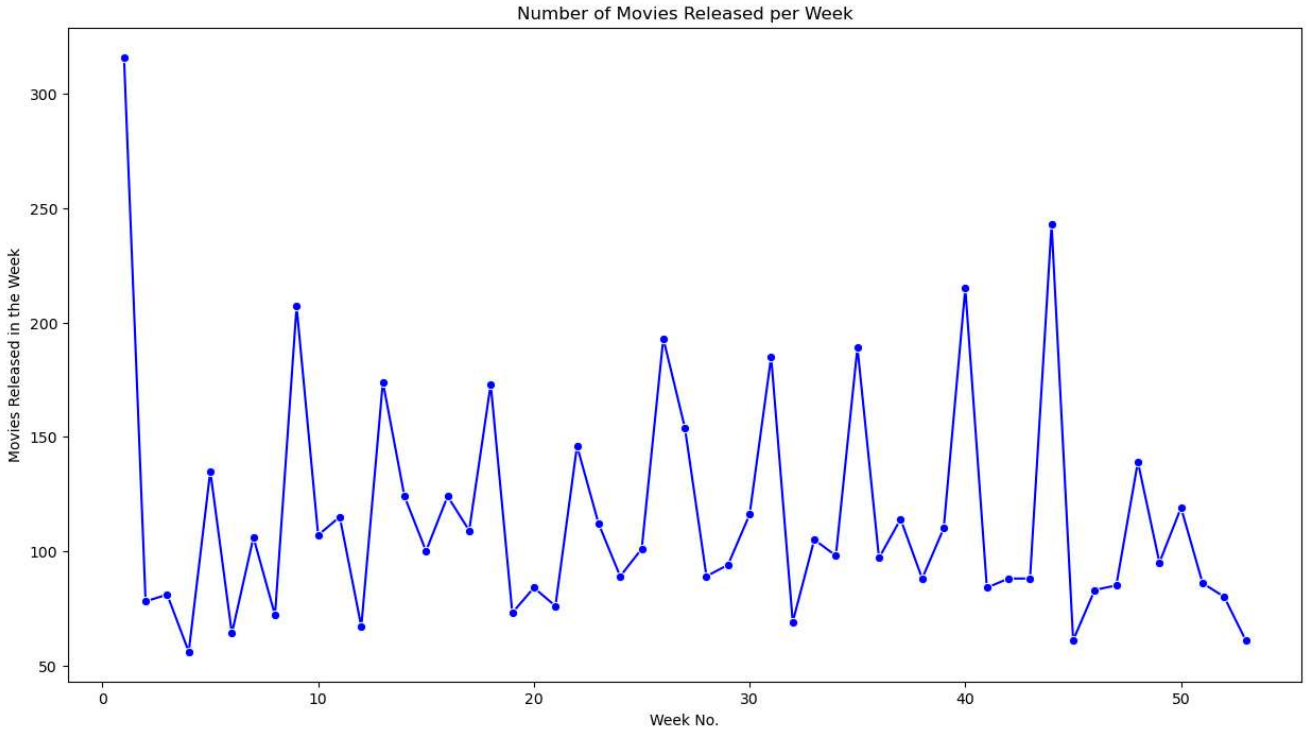


Till 2019, overall content across Netflix was increasing but due to Covid in 2020, though TV Shows didn't take a hit then Movies did take a hit. Well later in 2021, content across both was reduced significantly

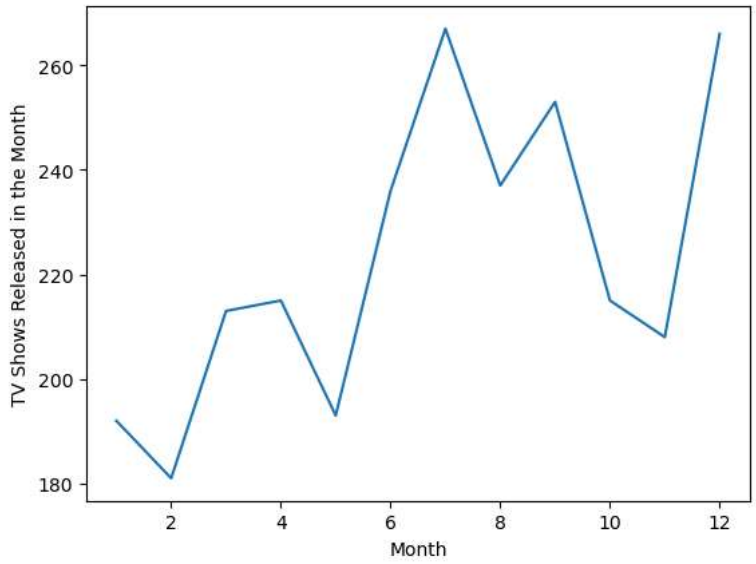
In [100]:
```
# Convert 'week_Added' to numeric, ignoring non-numeric values
df_shows['week_Added'] = pd.to_numeric(df_shows['week_Added'], errors='coerce')

# Drop rows with NaN values after conversion
df_shows = df_shows.dropna(subset=['week_Added'])

# Convert 'week_Added' back to integers for proper plotting
df_shows['week_Added'] = df_shows['week_Added'].astype(int)

# Group by 'week_Added' and count unique titles
df_week = (
    df_shows.groupby(['week_Added']).agg({"title": "nunique"})
    .reset_index()
    .sort_values(by=['week_Added'])
)

# Plotting the line plot
plt.figure(figsize=(15, 8))
sns.lineplot(data=df_week, x='week_Added', y='title', marker='o', color='blue')
plt.ylabel("Shows Released in the Week")
plt.xlabel("Week No.")
plt.title("Number of Shows Released per Week")
plt.show()
```



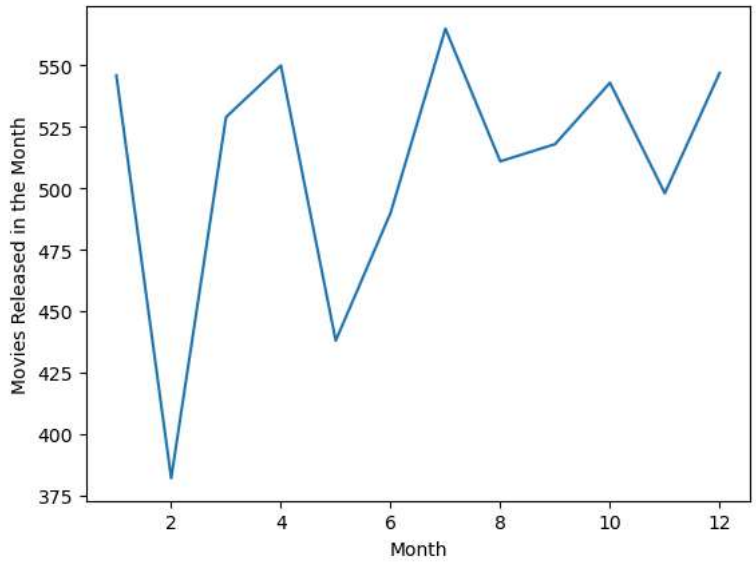
In [103]:
```
# Safely update 'week_Added' using .loc to avoid SettingWithCopyWarning
df_movies.loc[:, 'week_Added'] = pd.to_numeric(df_movies['week_Added'], errors='coerce')

# Drop rows with NaN values after conversion
df_movies = df_movies.dropna(subset=['week_Added'])

# Convert 'week_Added' back to integers for proper plotting
df_movies['week_Added'] = df_movies['week_Added'].astype(int)

# Group by 'week_Added' and count unique titles
df_week = (
    df_movies.groupby(['week_Added']).agg({"title": "nunique"})
    .reset_index()
    .sort_values(by=['week_Added'])
)

# Plotting the line plot
plt.figure(figsize=(15, 8))
sns.lineplot(data=df_week, x='week_Added', y='title', marker='o', color='blue')
plt.ylabel("Movies Released in the Week")
plt.xlabel("Week No.")
plt.title("Number of Movies Released per Week")
plt.show()
```



In [104]:
```
df_month=df_shows.groupby(['month_added']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_month, x='month_added', y='title')
plt.ylabel("TV Shows Released in the Month")
plt.xlabel("Month")
plt.show()
```



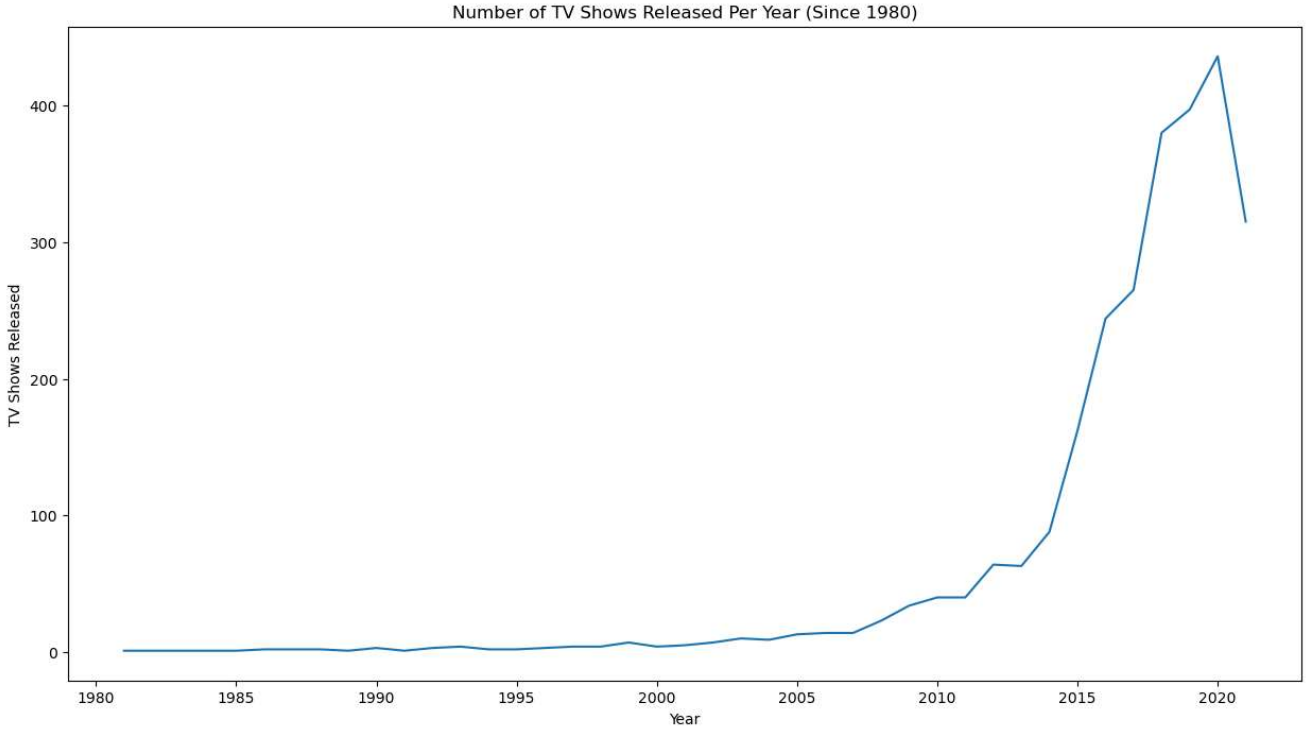
In [105]:
```
df_month=df_movies.groupby(['month_added']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_month, x='month_added', y='title')
plt.ylabel("Movies Released in the Month")
plt.xlabel("Month")
plt.show()
```



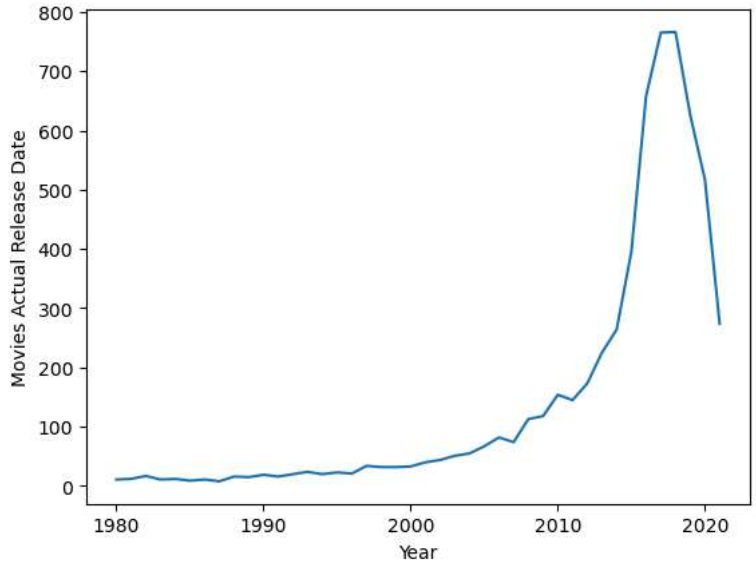
TV Shows are added in Netflix by a tremendous amount in mid weeks/months of the year, i.e- July

Movies are added in Netflix by a tremendous amount in first week/last month of current year and first month of next year

In [112]:
```
# Filter the DataFrame to include only TV shows released in or after 1980
df_release_year = df_shows[df_shows['release_year'] >= 1980].groupby(['release_year']).agg({"title": "nunique"}).reset_index()

# Plotting the line plot for TV shows
plt.figure(figsize=(15, 8))
sns.lineplot(data=df_release_year, x='release_year', y='title')
plt.ylabel("TV Shows Released")
plt.xlabel("Year")
plt.title("Number of TV Shows Released Per Year (Since 1980)")
plt.show()
```



In [113]:
```
df_release_year=df_movies[df_movies['release_year']>=1980].groupby(['release_year']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_release_year, x='release_year', y='title')
plt.ylabel("Movies Actual Release Date")
plt.xlabel("Year")
plt.show()
```



Actual Releases of both TV Shows and Movies have taken a hit after 2020

Questions to be Explored Now for Recommendations

1) So this time, the granularity level is country and analysis of TV Shows/Movies the country brings. I am going to consider only the top countries individually for TV Shows and Movies. There are definitely some common countries too which bring out quality content in both TV Shows and Movies.

2) Which Genres do these countries offer and what are the intended audiences(Ratings) which are popular in Netflix?

3)In case of Movies, what is the duration/length of movies which makes them special and depicts attention span?

4)Who are the popular actors/directors across TV Shows and Movies in these countries?

5)In what time of the year, people tend to watch movies and shows in these countries?

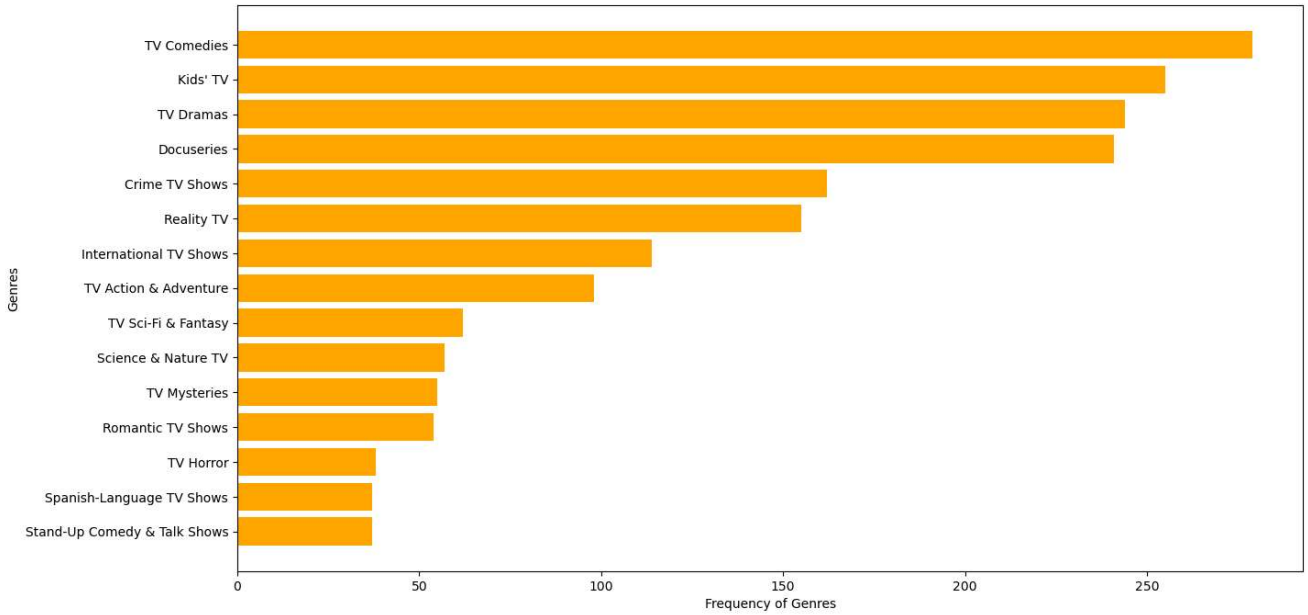
6)Popular Actor and Director Combinations in these countries

In [115]:
```
#below countries will be analyzed for both shows and movies
shows_and_movies=['United States','India','United Kingdom']
#below countries will be only analyzed on basis of shows
only_shows=['Japan','South Korea']
```

Univariate Analysis separately for shows and movies in USA

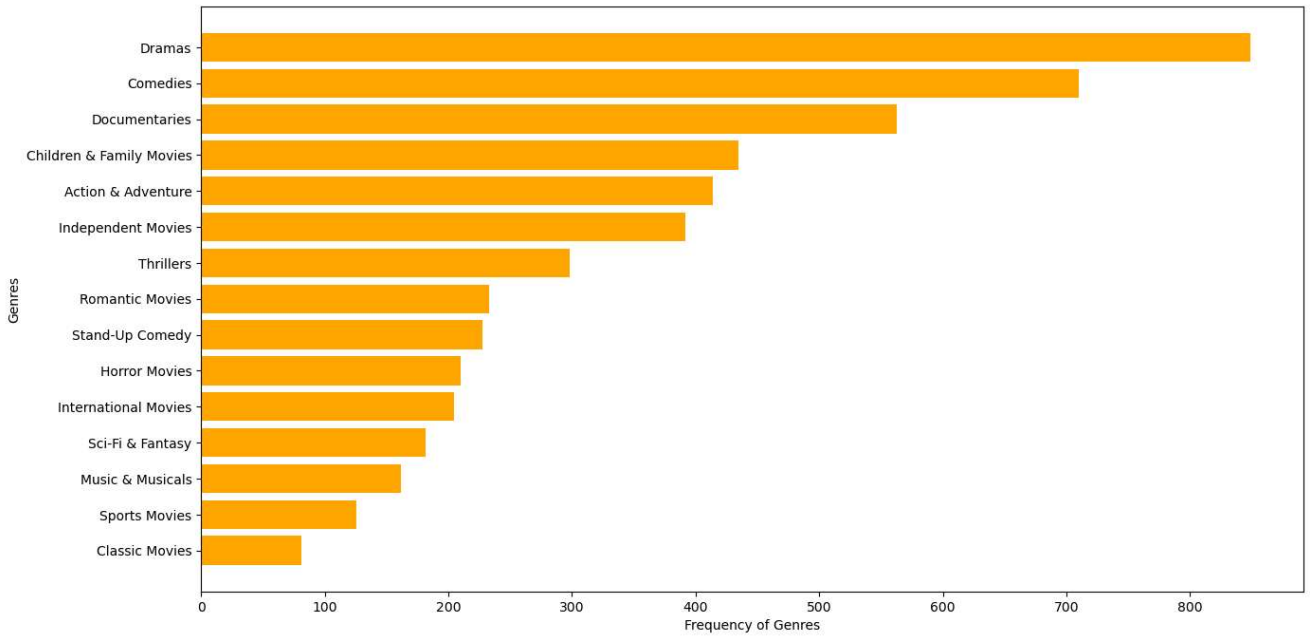
In [116]:
```
#Analyzing USA for both shows and movies
df_usa_shows=df_final1[df_final1['country']=='United States'][df_final1[df_final1['country']=='United States']['type']=='TV Show'
df_usa_movies=df_final1[df_final1['country']=='United States'][df_final1[df_final1['country']=='United States']['type']=='Movie']
```

In [117]:
```
df_genre=df_usa_shows.groupby(['Genre']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
plt.figure(figsize=(15,8))
plt.barh(df_genre[::-1]['Genre'], df_genre[::-1]['title'],color=['orange'])
plt.xlabel('Frequency of Genres')
plt.ylabel('Genres')
plt.show()
```



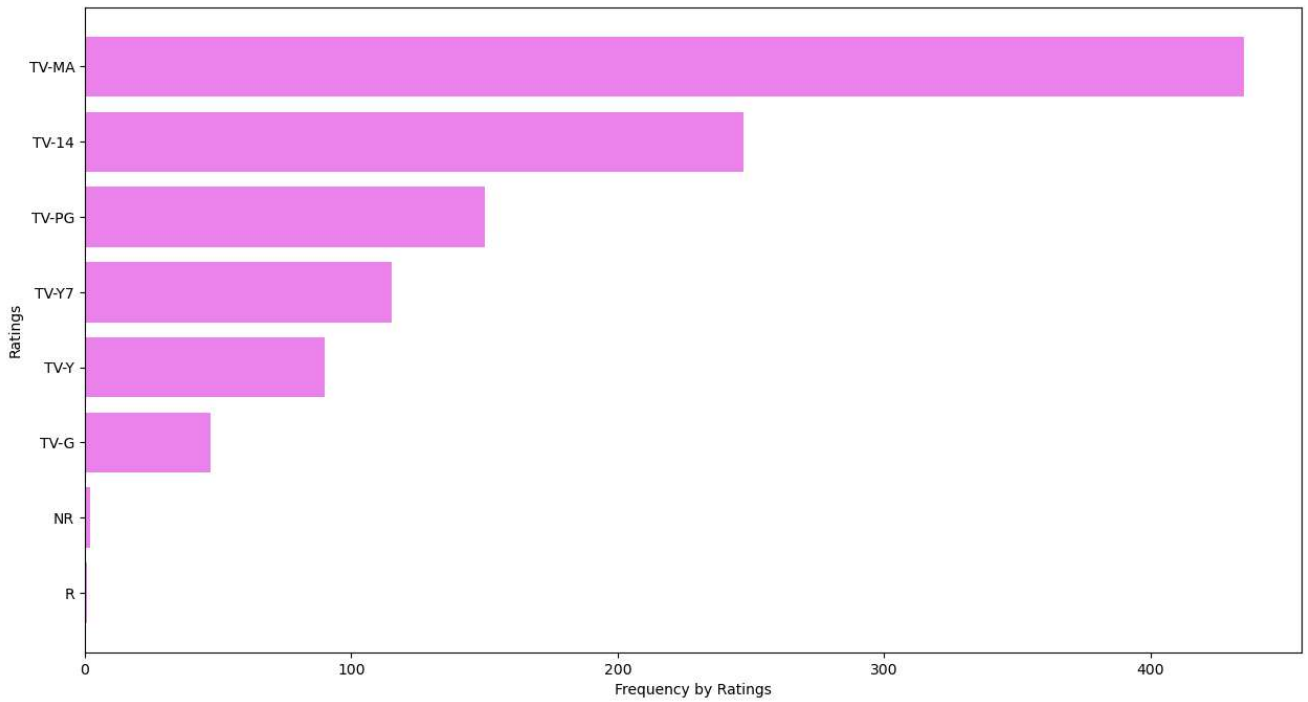
Dramas,Comedy, Kids 'TV Shows, International TV Shows and Docuseries, Genres are popular in TV Series in USA

In [119]:
```
df_genre=df_usa_movies.groupby(['Genre']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
plt.figure(figsize=(15,8))
plt.barh(df_genre[::-1]['Genre'], df_genre[::-1]['title'],color=['orange'])
plt.xlabel('Frequency of Genres')
plt.ylabel('Genres')
plt.show()
```

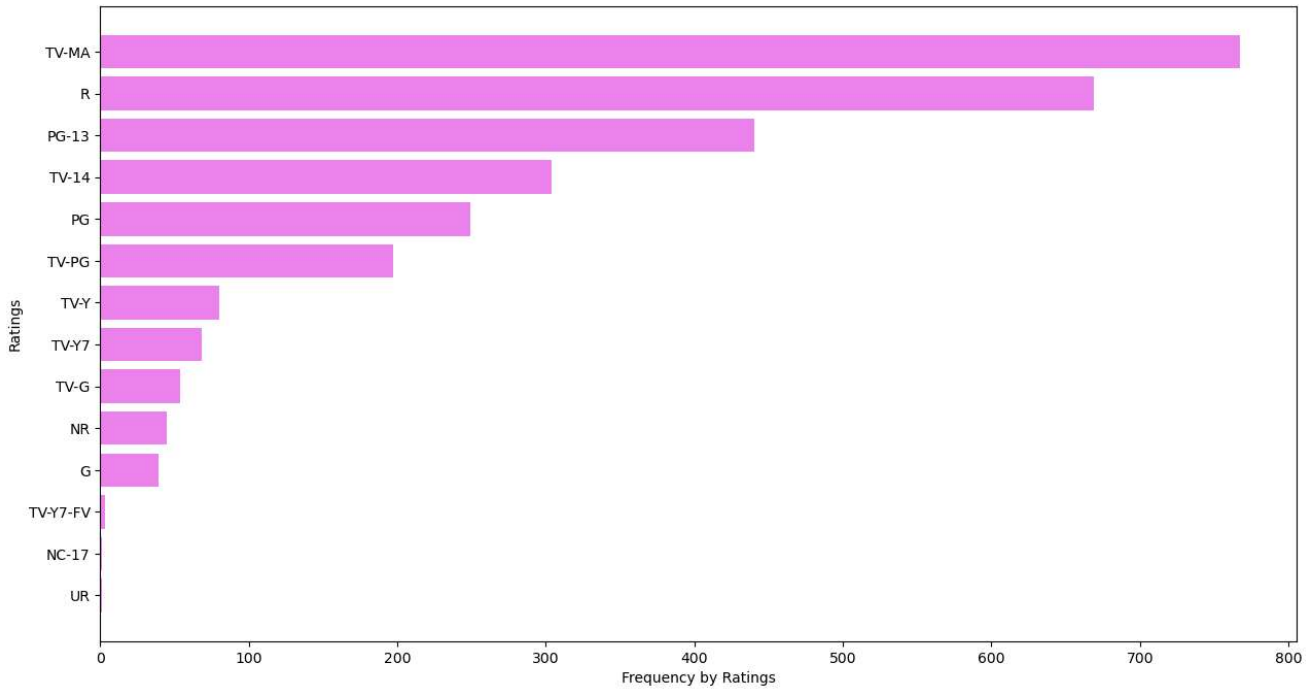


Dramas,Comedy, Documentaries, Family Movies and Action Genres in Movies are popular in USA

In [121]:
```
df_rating=df_usa_shows.groupby(['rating']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
plt.figure(figsize=(15,8))
plt.barh(df_rating[::-1]['rating'], df_rating[::-1]['title'],color=['violet'])
plt.xlabel('Frequency by Ratings')
plt.ylabel('Ratings')
plt.show()
```

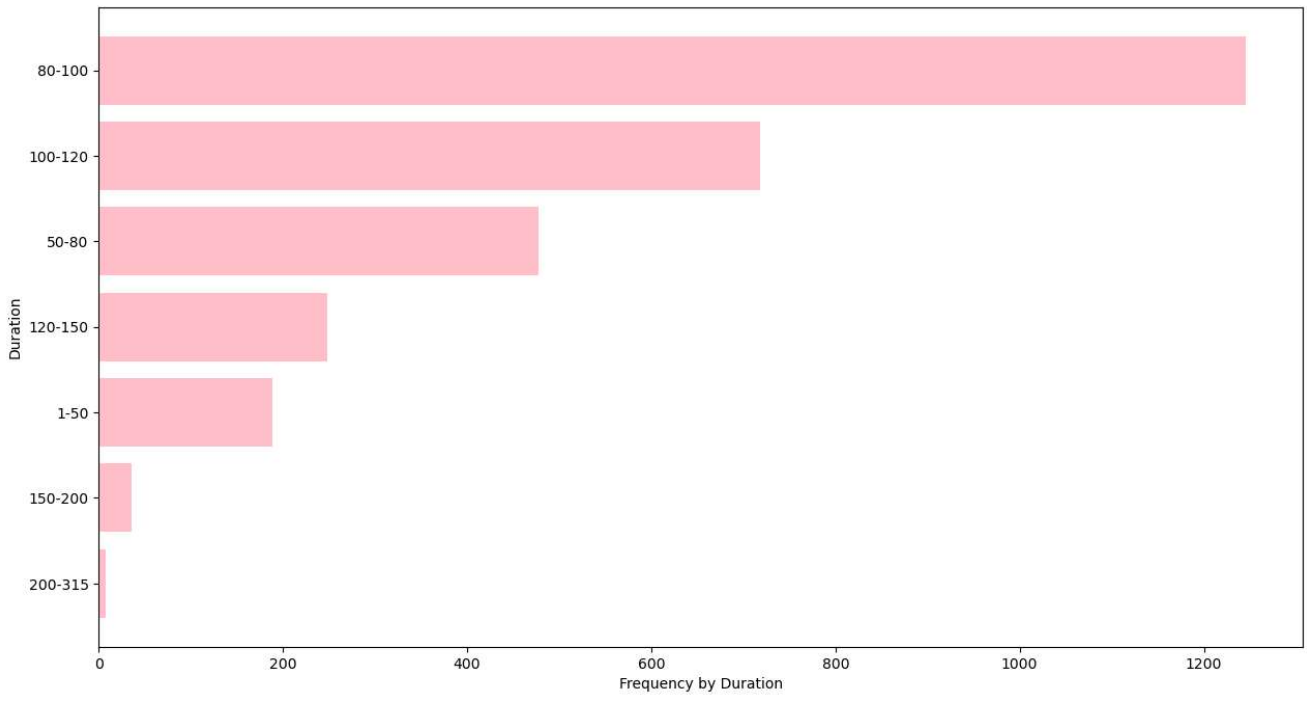


In [122]:
```
df_rating=df_usa_movies.groupby(['rating']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
plt.figure(figsize=(15,8))
plt.barh(df_rating[::-1]['rating'], df_rating[::-1]['title'],color=['violet'])
plt.xlabel('Frequency by Ratings')
plt.ylabel('Ratings')
plt.show()
```



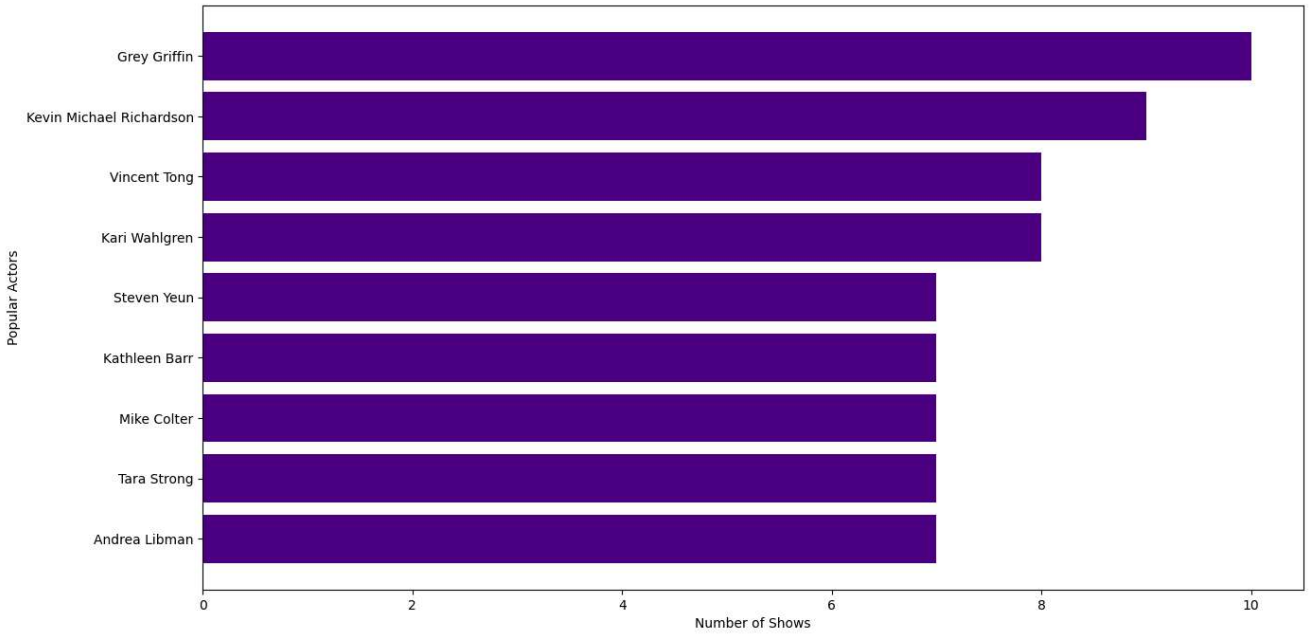
So it seems plaussible to conclude that the popular ratings across Netflix includes Mature Audiences and those appropriate for over 14/over 17 ages in both Movies and TV Shows in USA

In [123]:
```
df_duration=df_usa_movies.groupby(['duration']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[
plt.figure(figsize=(15,8))
plt.barh(df_duration[::-1]['duration'], df_duration[::-1]['title'],color=['pink'])
plt.xlabel('Frequency by Duration')
plt.ylabel('Duration')
plt.show()
```



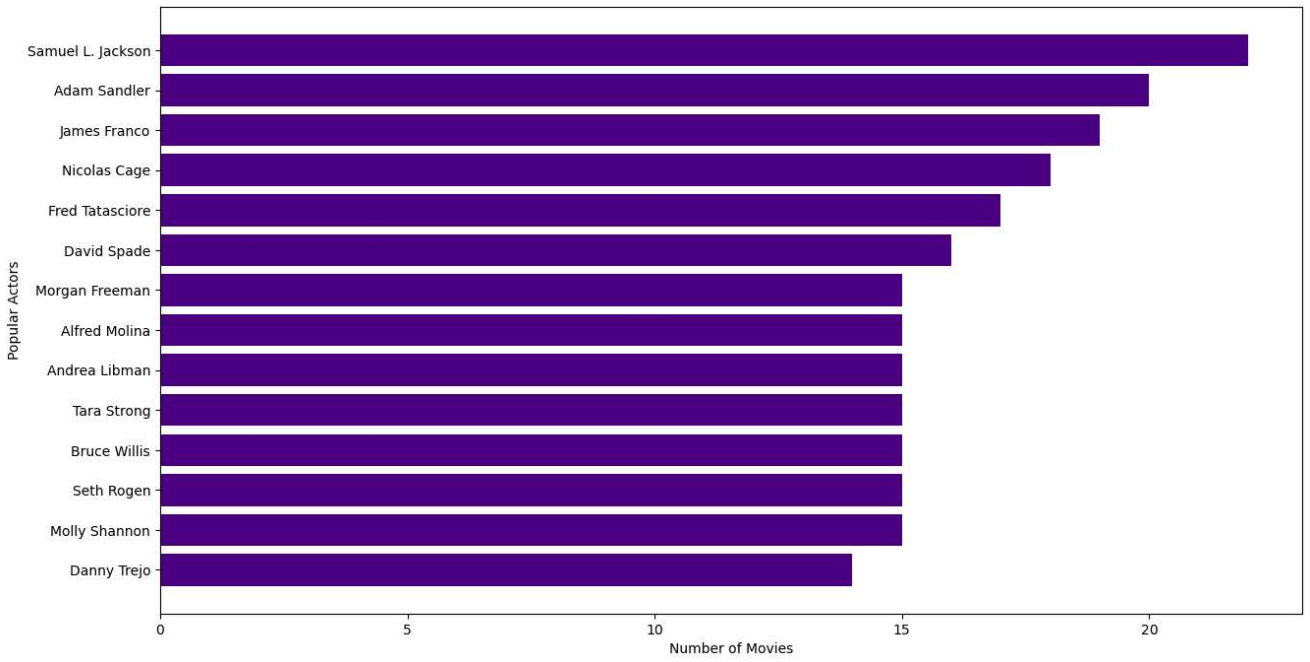
Across movies 80-100,100-120 is the ranges of minutes for which most movies lie. So quite possibly 80-120 mins is the sweet spot we would be wanting for movies in USA

In [124]:
```
df_actors=df_usa_shows.groupby(['Actors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:10]
df_actors=df_actors[df_actors['Actors']!='Unknown Actor']
plt.figure(figsize=(15,8))
plt.barh(df_actors[::-1]['Actors'], df_actors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Shows')
plt.ylabel('Popular Actors')
plt.show()
```



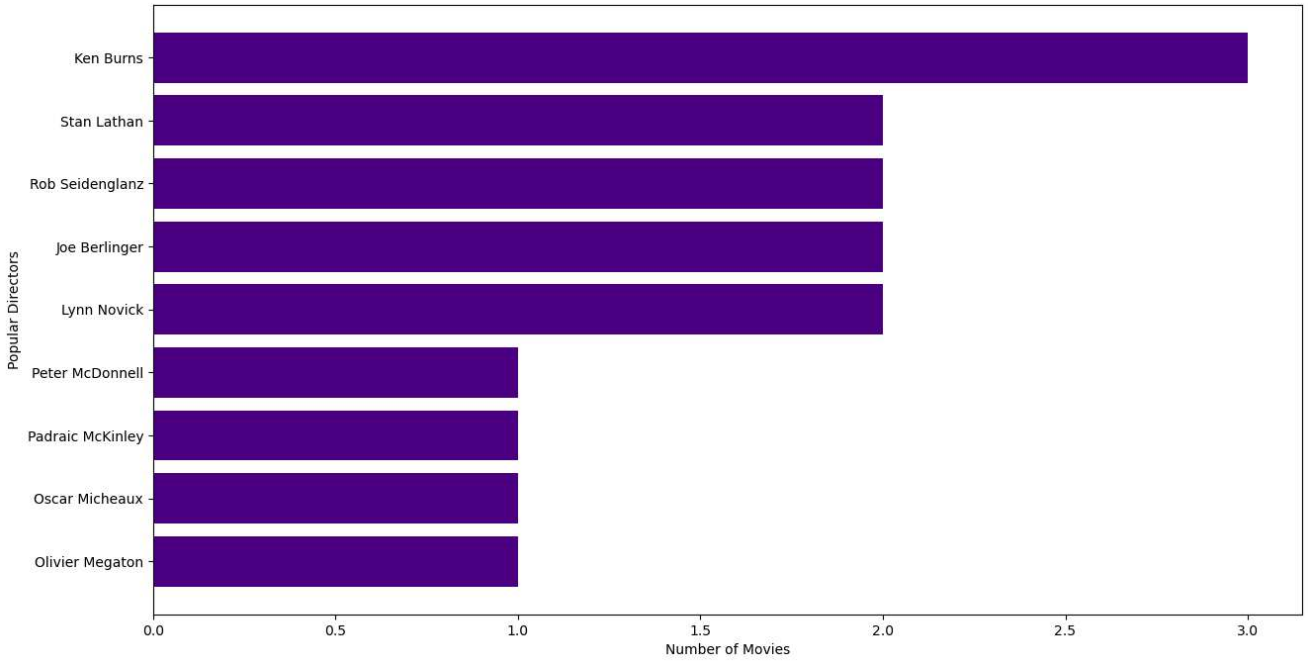
Grey Griffin and Kevin Richardson are the most popular actors across TV Shows in USA

In [125]:
```
df_actors=df_usa_movies.groupby(['Actors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
df_actors=df_actors[df_actors['Actors']!='Unknown Actor']
plt.figure(figsize=(15,8))
plt.barh(df_actors[::-1]['Actors'], df_actors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Movies')
plt.ylabel('Popular Actors')
plt.show()
```



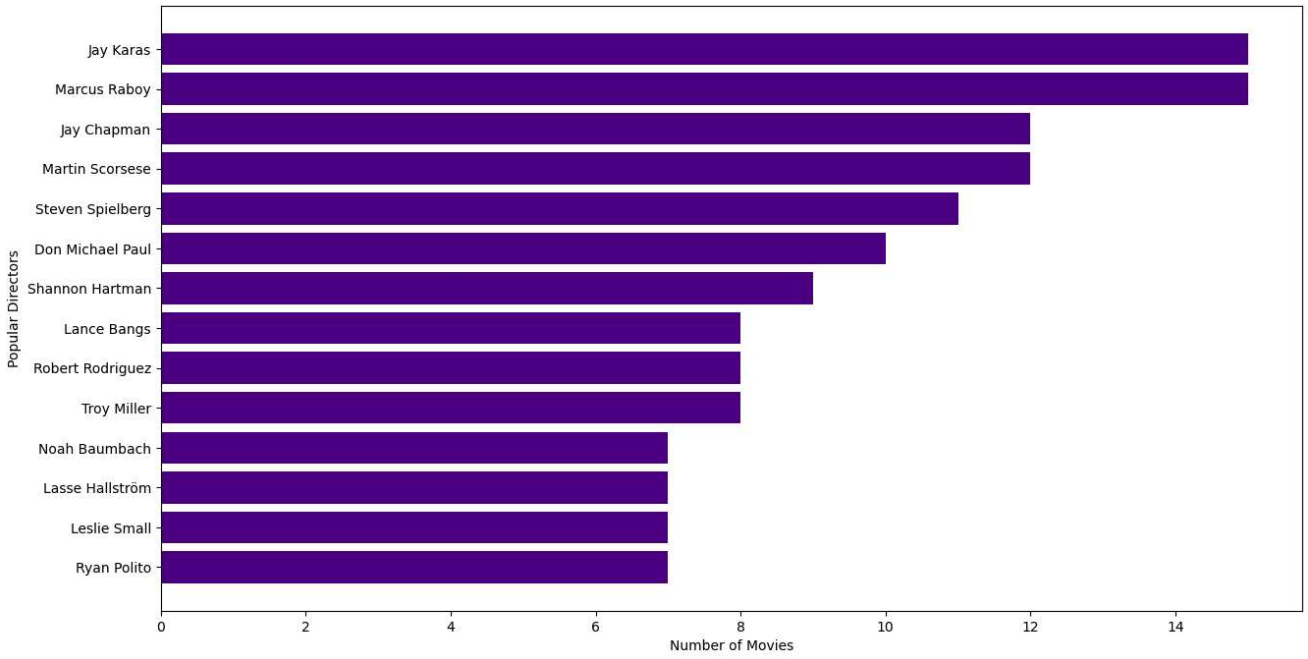
Samuel Jackson,Adam Sandler,James Franco and Nicolas Cage are very much popular across movies on Netflix in USA

In [127]:
```
df_directors=df_usa_shows.groupby(['Directors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)
df_directors=df_directors[df_directors['Directors']!='Unknown Director']
plt.figure(figsize=(15,8))
plt.barh(df_directors[::-1]['Directors'], df_directors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Movies')
plt.ylabel('Popular Directors')
plt.show()
```



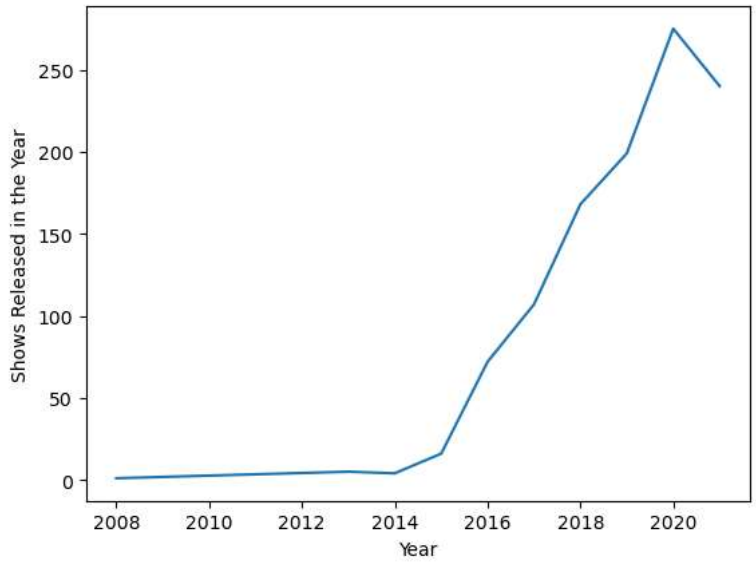
Ken Burns,Stan Lathan, Joe Barlinger are popular directors across TV Shows on Netflix in USA

In [128]:
```
df_directors=df_usa_movies.groupby(['Directors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)
df_directors=df_directors[df_directors['Directors']!='Unknown Director']
plt.figure(figsize=(15,8))
plt.barh(df_directors[::-1]['Directors'], df_directors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Movies')
plt.ylabel('Popular Directors')
plt.show()
```

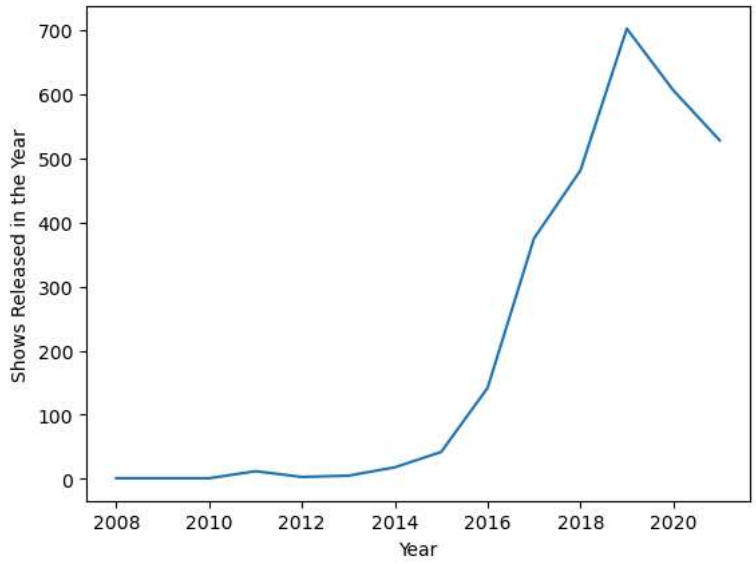


Jay Karas,Marcus Raboy,Martin Scorcese and Jay Chapman are popular directors across movies in USA

In [129]:
```
df_year=df_usa_shows.groupby(['year']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_year, x='year', y='title')
plt.ylabel("Shows Released in the Year")
plt.xlabel("Year")
plt.show()
```

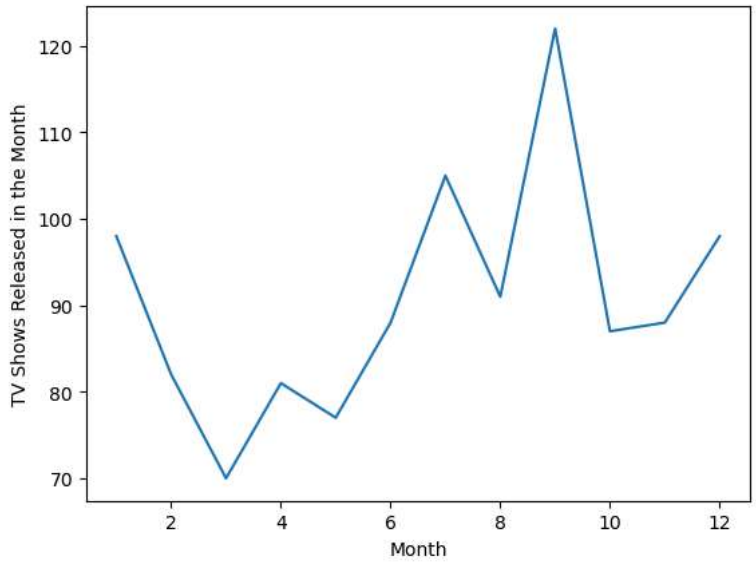


In [130]:
```
df_year=df_usa_movies.groupby(['year']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_year, x='year', y='title')
plt.ylabel("Shows Released in the Year")
plt.xlabel("Year")
plt.show()
```

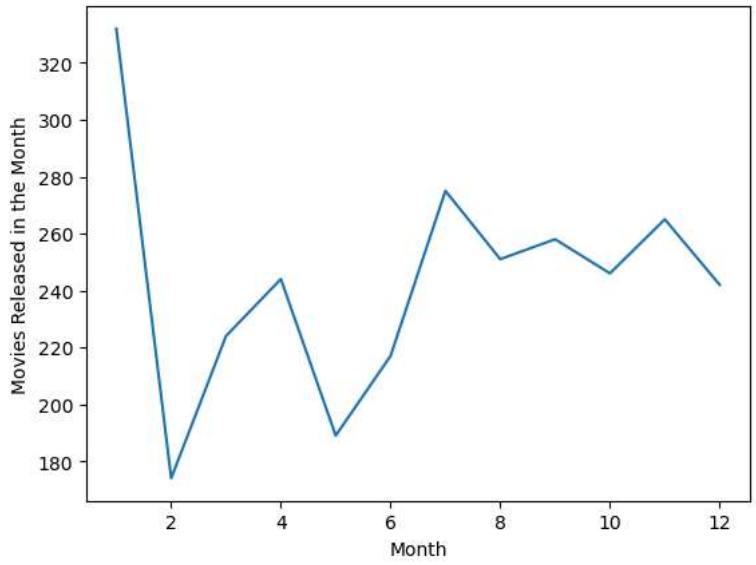


In USA, number of shows remained the same in 2021 as they were in 2020 while number of movies declined.

In [133]:
```
df_month=df_usa_shows.groupby(['month_added']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_month, x='month_added', y='title')
plt.ylabel("TV Shows Released in the Month")
plt.xlabel("Month")
plt.show()
```



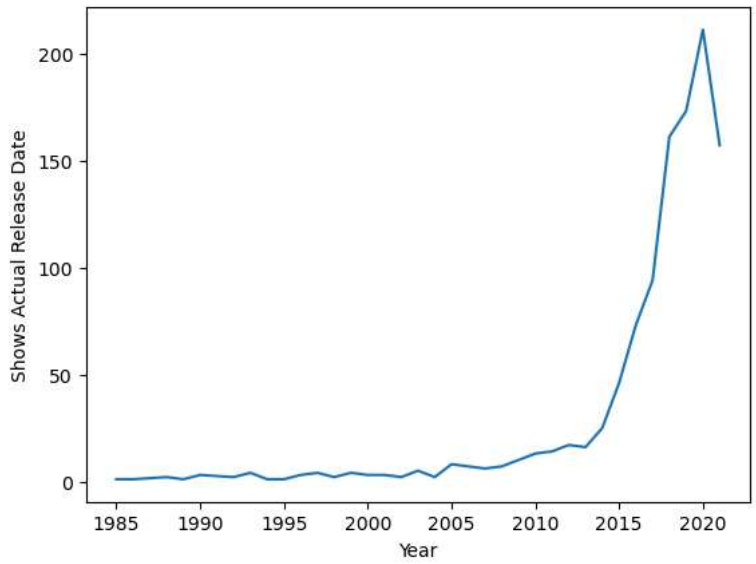
In [134]:
```
df_month=df_usa_movies.groupby(['month_added']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_month, x='month_added', y='title')
plt.ylabel("Movies Released in the Month")
plt.xlabel("Month")
plt.show()
```



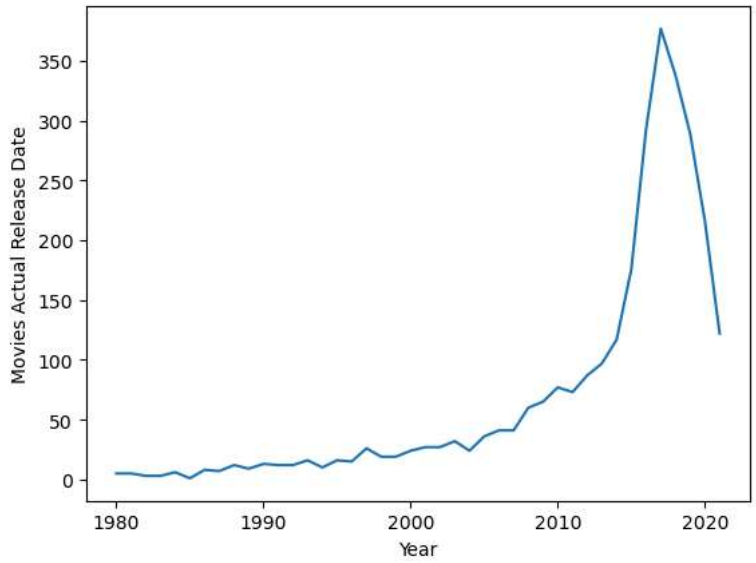
TV Shows are added in Netflix by a tremendous amount in July and September in USA

Movies are added in Netflix in USA by a tremendous amount in first week/last month of current year and first month of next year

In [136]:
```
df_release_year=df_usa_shows[df_usa_shows['release_year']>=1980].groupby(['release_year']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_release_year, x='release_year', y='title')
plt.ylabel("Shows Actual Release Date")
plt.xlabel("Year")
plt.show()
```



In [135]:
```
df_release_year=df_usa_movies[df_usa_movies['release_year']>=1980].groupby(['release_year']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_release_year, x='release_year', y='title')
plt.ylabel("Movies Actual Release Date")
plt.xlabel("Year")
plt.show()
```



In USA, though both Movies and Shows have reduced in 2021, the amount of decrease in number of TV Shows is small as compared to Movies

In [137]:
```
df_usa_movies.head()
```

Out[137]:

	title	Actors	Directors	Genre	country	show_id	type	date_added	release_year	rating	duration	Modified_Added_date	month_added	v
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States	s1	Movie	September 25, 2021	2020	PG-13	80-100	2021-09-25	9	
159	My Little Pony: A New Generation	Vanessa Hudgens	Robert Cullen	Children & Family Movies	United States	s7	Movie	September 24, 2021	2021	PG	80-100	2021-09-24	9	
160	My Little Pony: A New Generation	Vanessa Hudgens	José Luis Ucha	Children & Family Movies	United States	s7	Movie	September 24, 2021	2021	PG	80-100	2021-09-24	9	
161	My Little Pony: A New Generation	Kimiko Glenn	Robert Cullen	Children & Family Movies	United States	s7	Movie	September 24, 2021	2021	PG	80-100	2021-09-24	9	
162	My Little Pony: A New Generation	Kimiko Glenn	José Luis Ucha	Children & Family Movies	United States	s7	Movie	September 24, 2021	2021	PG	80-100	2021-09-24	9	

In [138]:
```
#Analysing a combination of actors and directors
df_usa_movies['Actor_Director_Combination'] = df_usa_movies.Actors.str.cat(df_usa_movies.Directors, sep=' and ')
df_usa_movies_subset=df_usa_movies[df_usa_movies['Actors']!='Unknown Actor']
df_usa_movies_subset=df_usa_movies_subset[df_usa_movies_subset['Directors']!='Unknown Director']
df_usa_movies_subset.head()
```

Out[138]:

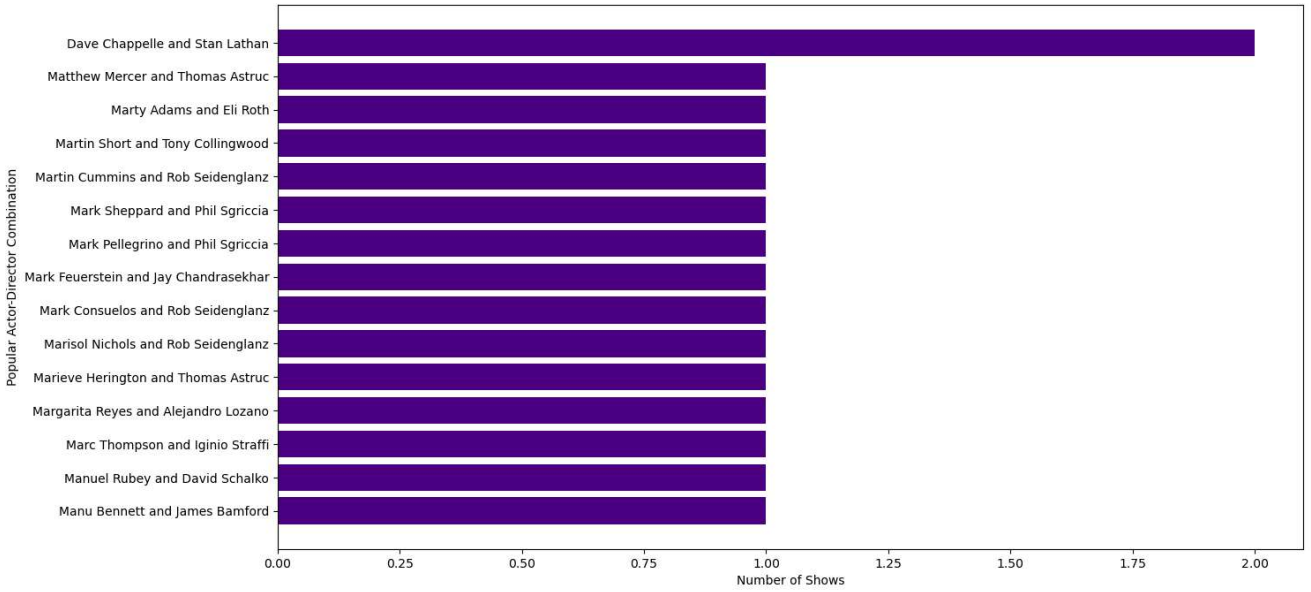
	title	Actors	Directors	Genre	country	show_id	type	date_added	release_year	rating	duration	Modified_Added_date	month_added	week_A
159	My Little Pony: A New Generation	Vanessa Hudgens	Robert Cullen	Children & Family Movies	United States	s7	Movie	September 24, 2021	2021	PG	80-100	2021-09-24	9	
160	My Little Pony: A New Generation	Vanessa Hudgens	José Luis Ucha	Children & Family Movies	United States	s7	Movie	September 24, 2021	2021	PG	80-100	2021-09-24	9	
161	My Little Pony: A New Generation	Kimiko Glenn	Robert Cullen	Children & Family Movies	United States	s7	Movie	September 24, 2021	2021	PG	80-100	2021-09-24	9	
162	My Little Pony: A New Generation	Kimiko Glenn	José Luis Ucha	Children & Family Movies	United States	s7	Movie	September 24, 2021	2021	PG	80-100	2021-09-24	9	
163	My Little Pony: A New Generation	James Marsden	Robert Cullen	Children & Family Movies	United States	s7	Movie	September 24, 2021	2021	PG	80-100	2021-09-24	9	

In [139]:
```
df_usa_shows['Actor_Director_Combination'] = df_usa_shows.Actors.str.cat(df_usa_shows.Directors, sep=' and ')
df_usa_shows_subset=df_usa_shows[df_usa_shows['Actors']!='Unknown Actor']
df_usa_shows_subset=df_usa_shows_subset[df_usa_shows_subset['Directors']!='Unknown Director']
df_usa_shows_subset.head()
```

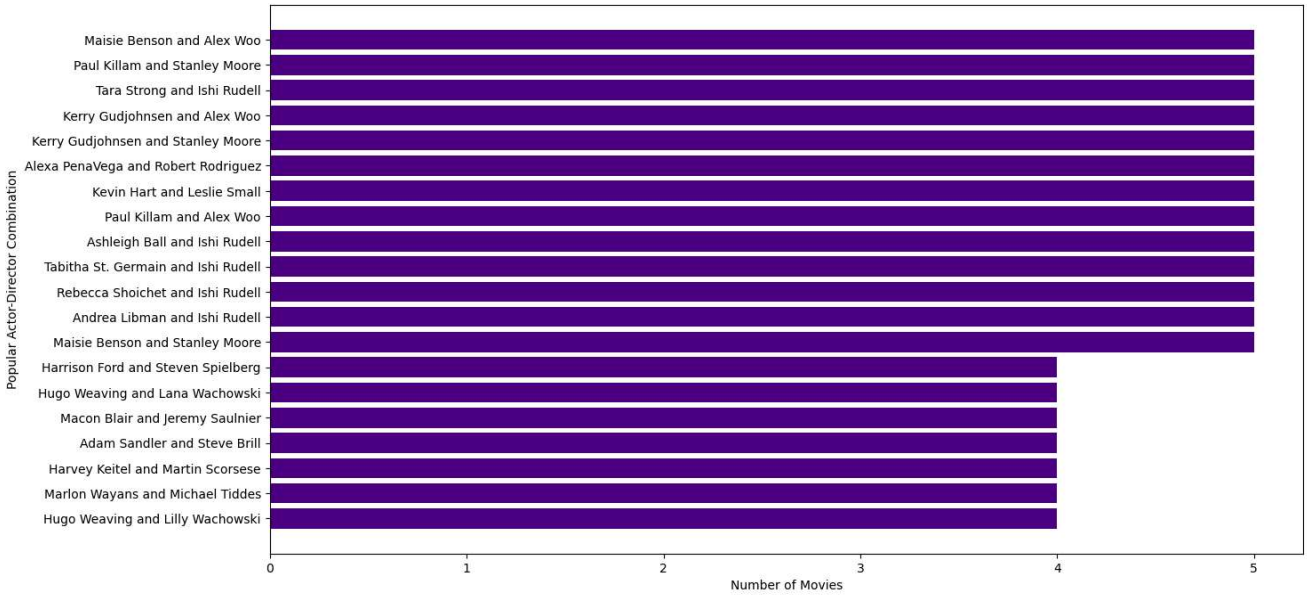
Out[139]:

	title	Actors	Directors	Genre	country	show_id	type	date_added	release_year	rating	duration	Modified_Added_date	month_added	week_Adde
111	Midnight Mass	Kate Siegel	Mike Flanagan	TV Dramas	United States	s6	TV Show	September 24, 2021	2021	TV-MA	1 Season	2021-09-24	9	
112	Midnight Mass	Kate Siegel	Mike Flanagan	TV Horror	United States	s6	TV Show	September 24, 2021	2021	TV-MA	1 Season	2021-09-24	9	
113	Midnight Mass	Kate Siegel	Mike Flanagan	TV Mysteries	United States	s6	TV Show	September 24, 2021	2021	TV-MA	1 Season	2021-09-24	9	
114	Midnight Mass	Zach Gilford	Mike Flanagan	TV Dramas	United States	s6	TV Show	September 24, 2021	2021	TV-MA	1 Season	2021-09-24	9	
115	Midnight Mass	Zach Gilford	Mike Flanagan	TV Horror	United States	s6	TV Show	September 24, 2021	2021	TV-MA	1 Season	2021-09-24	9	

In [140]:
```
df_actors_directors=df_usa_shows_subset.groupby(['Actor_Director_Combination']).agg({"title":"nunique"}).reset_index().sort_value
plt.figure(figsize=(15,8))
plt.barh(df_actors_directors[::-1]['Actor_Director_Combination'], df_actors_directors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Shows')
plt.ylabel('Popular Actor-Director Combination')
plt.show()
```



In [142]:
```
df_actors_directors=df_usa_movies_subset.groupby(['Actor_Director_Combination']).agg({"title":"nunique"}).reset_index().sort_valu
plt.figure(figsize=(15,8))
plt.barh(df_actors_directors[::-1]['Actor_Director_Combination'], df_actors_directors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Movies')
plt.ylabel('Popular Actor-Director Combination')
plt.show()
```



In [143]:
```
df_actors_directors[::-1]['Actor_Director_Combination'].values
```

Out[143]:
```
array(['Hugo Weaving and Lilly Wachowski',
       'Marlon Wayans and Michael Tiddes',
       'Harvey Keitel and Martin Scorsese',
       'Adam Sandler and Steve Brill', 'Macon Blair and Jeremy Saulnier',
       'Hugo Weaving and Lana Wachowski',
       'Harrison Ford and Steven Spielberg',
       'Maisie Benson and Stanley Moore', 'Andrea Libman and Ishi Rudell',
       'Rebecca Shoichet and Ishi Rudell',
       'Tabitha St. Germain and Ishi Rudell',
       'Ashleigh Ball and Ishi Rudell', 'Paul Killam and Alex Woo',
       'Kevin Hart and Leslie Small',
       'Alexa PenaVega and Robert Rodriguez',
       'Kerry Gudjohnsen and Stanley Moore',
       'Kerry Gudjohnsen and Alex Woo', 'Tara Strong and Ishi Rudell',
       'Paul Killam and Stanley Moore', 'Maisie Benson and Alex Woo'],
      dtype=object)
```

The Most Popular Actor Director Combination in Movies Across USA are:-

'Smith Foreman and Stanley Moore',

'Marlon Wayans and Michael Tiddes',

'Adam Sandler and Steve Brill',

'Maisie Benson and Stanley Moore',

'Ashleigh Ball and Ishi Rudell',

'Tara Strong and Ishi Rudell',

'Rebecca Shoichet and Ishi Rudell',

'Kerry Gudjohnsen and Alex Woo',

'Kerry Gudjohnsen and Stanley Moore',

'Paul Killam and Alex Woo',

'Paul Killam and Stanley Moore',

'Andrea Libman and Ishi Rudell',

'Kevin Hart and Leslie Small',

'Maisie Benson and Alex Woo',

'Alexa PenaVega and Robert Rodriguez',

'Tabitha St. Germain and Ishi Rudell'

The Second Most Popular Actor Director Combination in Movies Across USA are:-

'Rory Markham and Mike Gunther',

'Erin Mathews and Steve Ball',

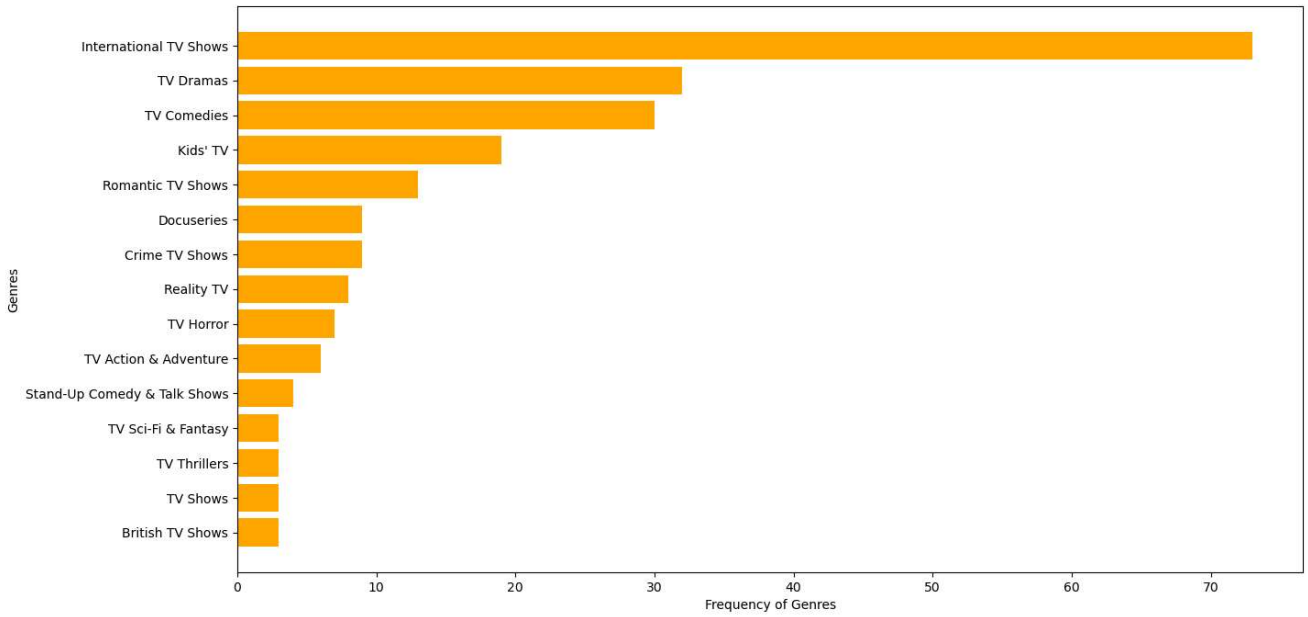
'Danny Trejo and Robert Rodriguez',

'Jeff Dunham and Michael Simon'

Univariate Analysis separately for shows and movies in India

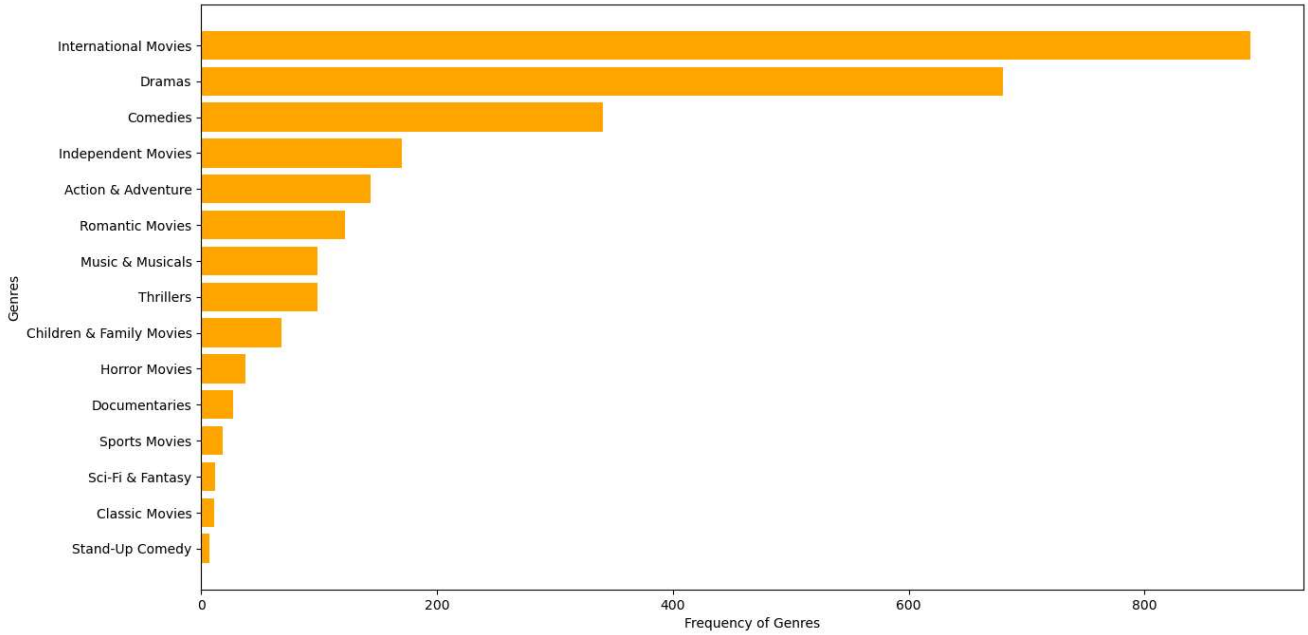
In [144]:
```
#Analyzing India for both shows and movies
df_india_shows=df_final1[df_final1['country']=='India'][df_final1[df_final1['country']=='India']['type']=='TV Show']
df_india_movies=df_final1[df_final1['country']=='India'][df_final1[df_final1['country']=='India']['type']=='Movie']
```

In [145]:
```
df_genre=df_india_shows.groupby(['Genre']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
plt.figure(figsize=(15,8))
plt.barh(df_genre[::-1]['Genre'], df_genre[::-1]['title'],color=['orange'])
plt.xlabel('Frequency of Genres')
plt.ylabel('Genres')
plt.show()
```



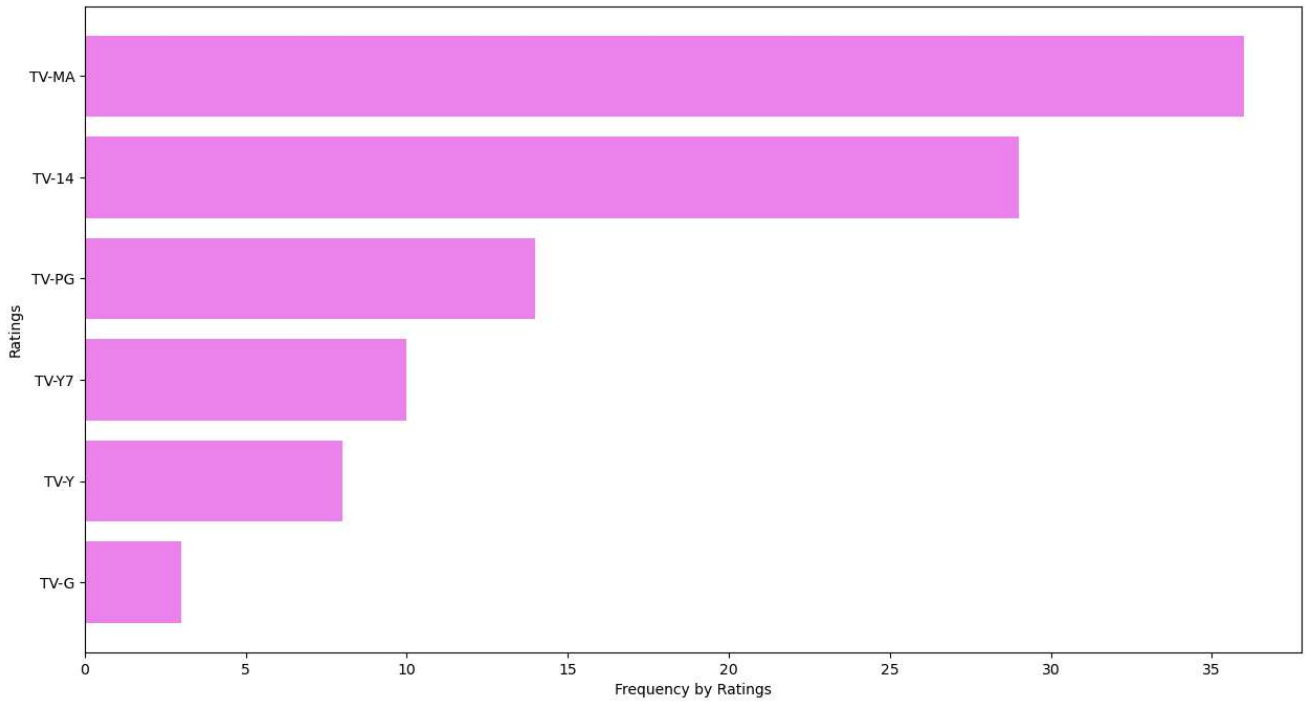
Dramas,Comedy, Kids 'TV Shows and International TV Shows Genres are popular in TV Series in India

In [147]:
```
df_genre=df_india_movies.groupby(['Genre']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
plt.figure(figsize=(15,8))
plt.barh(df_genre[::-1]['Genre'], df_genre[::-1]['title'],color=['orange'])
plt.xlabel('Frequency of Genres')
plt.ylabel('Genres')
plt.show()
```

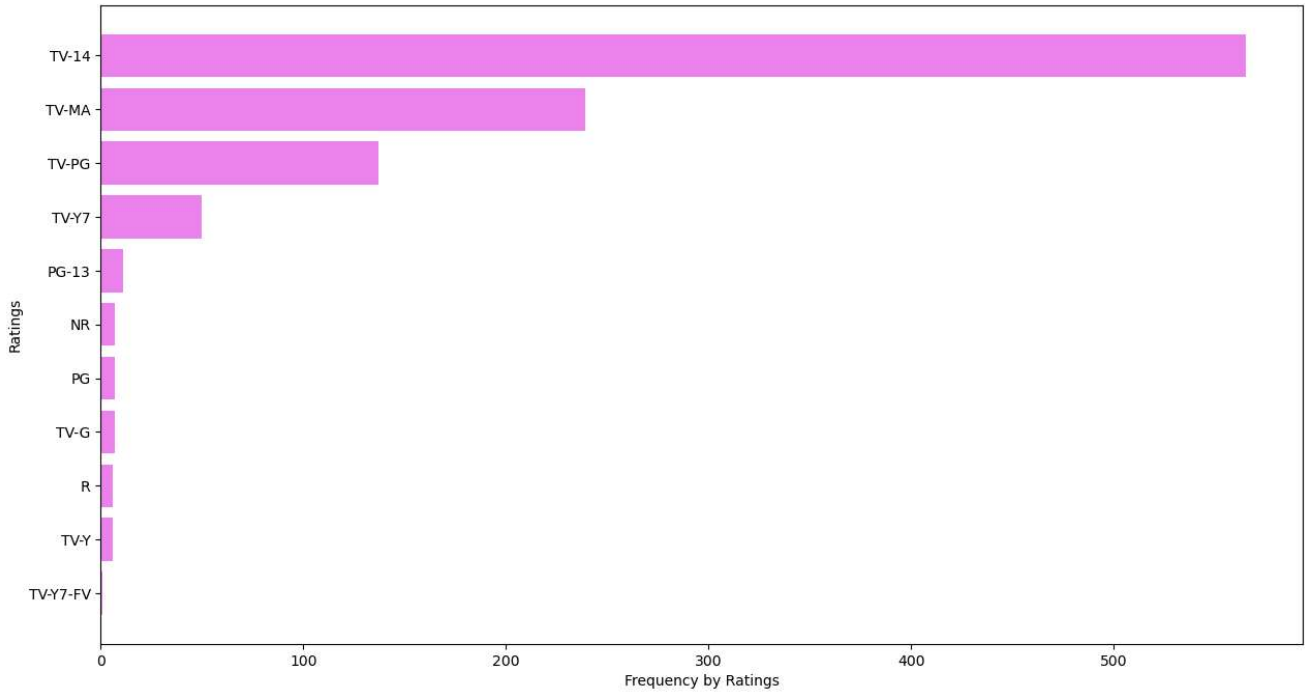


International Movies,Drama,Comedy,Indpeendent Movies and Action, Romance Genres are prevalent in India

In [149]:
```
df_rating=df_india_shows.groupby(['rating']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15
plt.figure(figsize=(15,8))
plt.barh(df_rating[::-1]['rating'], df_rating[::-1]['title'],color=['violet'])
plt.xlabel('Frequency by Ratings')
plt.ylabel('Ratings')
plt.show()
```



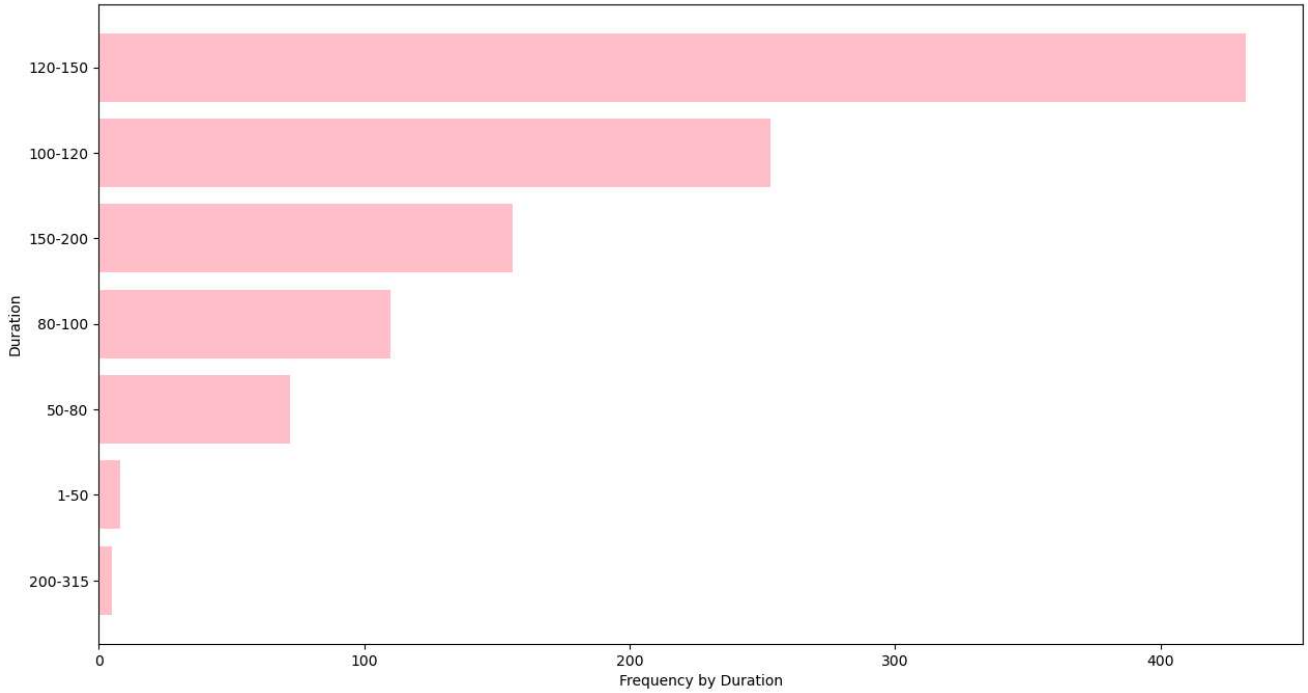
In [151]:
```
df_rating=df_india_movies.groupby(['rating']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:1
plt.figure(figsize=(15,8))
plt.barh(df_rating[::-1]['rating'], df_rating[::-1]['title'],color=['violet'])
plt.xlabel('Frequency by Ratings')
plt.ylabel('Ratings')
plt.show()
```



So it seems plaussible to conclude that the popular ratings across Netflix includes Mature Audiences in TV Shows and those appropriate for people over 14 in Movies in India.

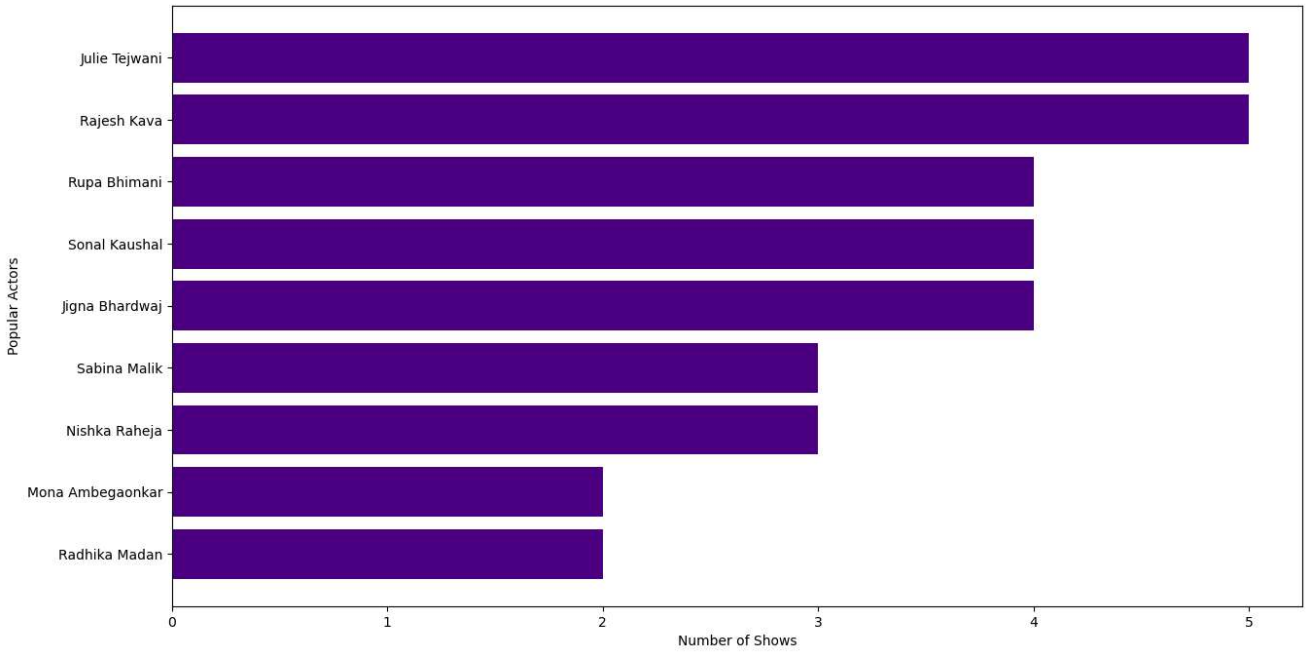
Now this indeed seems to be the case. Indian TV Shows in Netflix are without a shadow of doubt intended for Mature Audiences while Movies for over 14

In [152]:
```
df_duration=df_india_movies.groupby(['duration']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False
plt.figure(figsize=(15,8))
plt.barh(df_duration[::-1]['duration'], df_duration[::-1]['title'],color=['pink'])
plt.xlabel('Frequency by Duration')
plt.ylabel('Duration')
plt.show()
```



Across movies ranges of minutes in India are comparatively greater than USA with a sweet spot at 120-150 mins

In [153]:
```
df_actors=df_india_shows.groupby(['Actors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:10
df_actors=df_actors[df_actors['Actors']!='Unknown Actor']
plt.figure(figsize=(15,8))
plt.barh(df_actors[::-1]['Actors'], df_actors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Shows')
plt.ylabel('Popular Actors')
plt.show()
```



In [154]:
```
df_actors['Actors'].values
```

Out[154]:
```
array(['Julie Tejwani', 'Rajesh Kava', 'Rupa Bhimani', 'Sonal Kaushal',
       'Jigna Bhardwaj', 'Sabina Malik', 'Nishka Raheja',
       'Mona Ambegaonkar', 'Radhika Madan'], dtype=object)
```

Popular Actors in TV Shows in India are:-

'Julie Tejwani',

'Rajesh Kava',

'Rupa Bhimani',

'Sonal Kaushal',

'Jigna Bhardwaj',

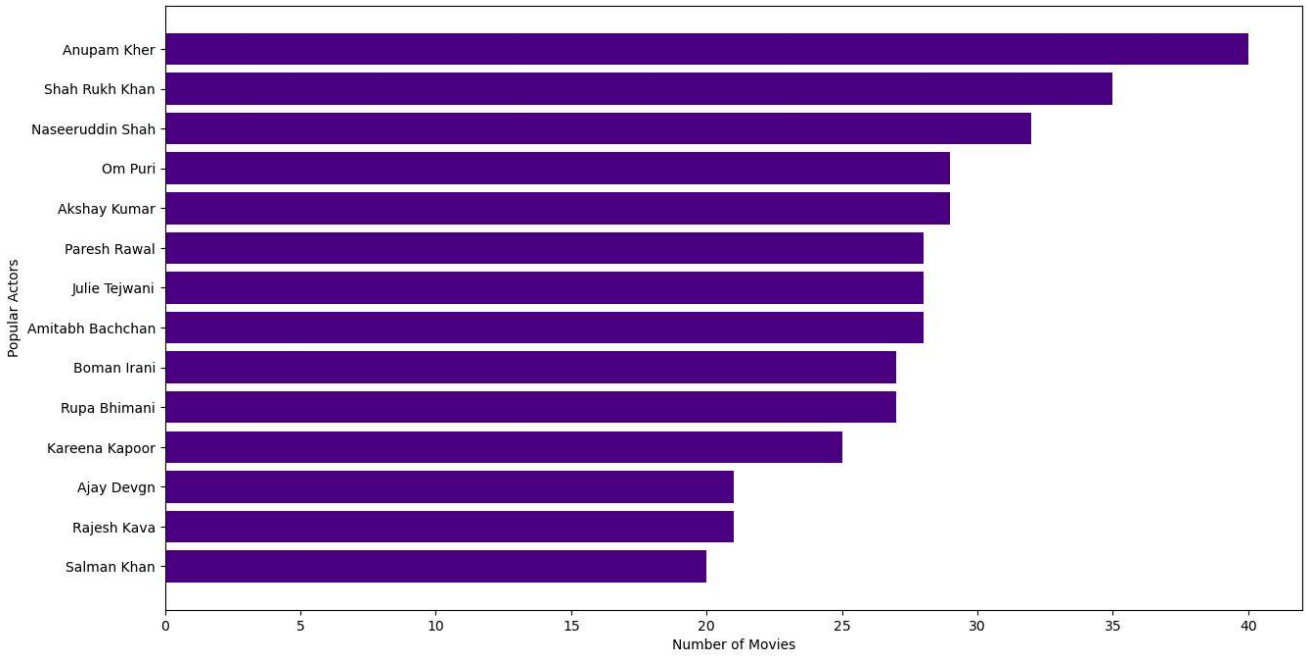
'Sabina Malik',

'Nishka Raheja',

'Mona Ambegaonkar',

'Radhika Madan'

In [155]:
```
df_actors=df_india_movies.groupby(['Actors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:1
df_actors=df_actors[df_actors['Actors']!='Unknown Actor']
plt.figure(figsize=(15,8))
plt.barh(df_actors[::-1]['Actors'], df_actors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Movies')
plt.ylabel('Popular Actors')
plt.show()
```



In [156]:
```
df_actors['Actors'].values
```

Out[156]:
```
array(['Anupam Kher', 'Shah Rukh Khan', 'Naseeruddin Shah', 'Om Puri',
       'Akshay Kumar', 'Paresh Rawal', 'Julie Tejwani',
       'Amitabh Bachchan', 'Boman Irani', 'Rupa Bhimani',
       'Kareena Kapoor', 'Ajay Devgn', 'Rajesh Kava', 'Salman Khan'],
      dtype=object)
```

Popular actors across Movies in India:-

'Anupam Kher',

'Shah Rukh Khan',

'Naseeruddin Shah',

'Om Puri',

'Akshay Kumar',

'Paresh Rawal',

'Julie Tejwani',

'Amitabh Bachchan',

'Boman Irani',

'Rupa Bhimani',

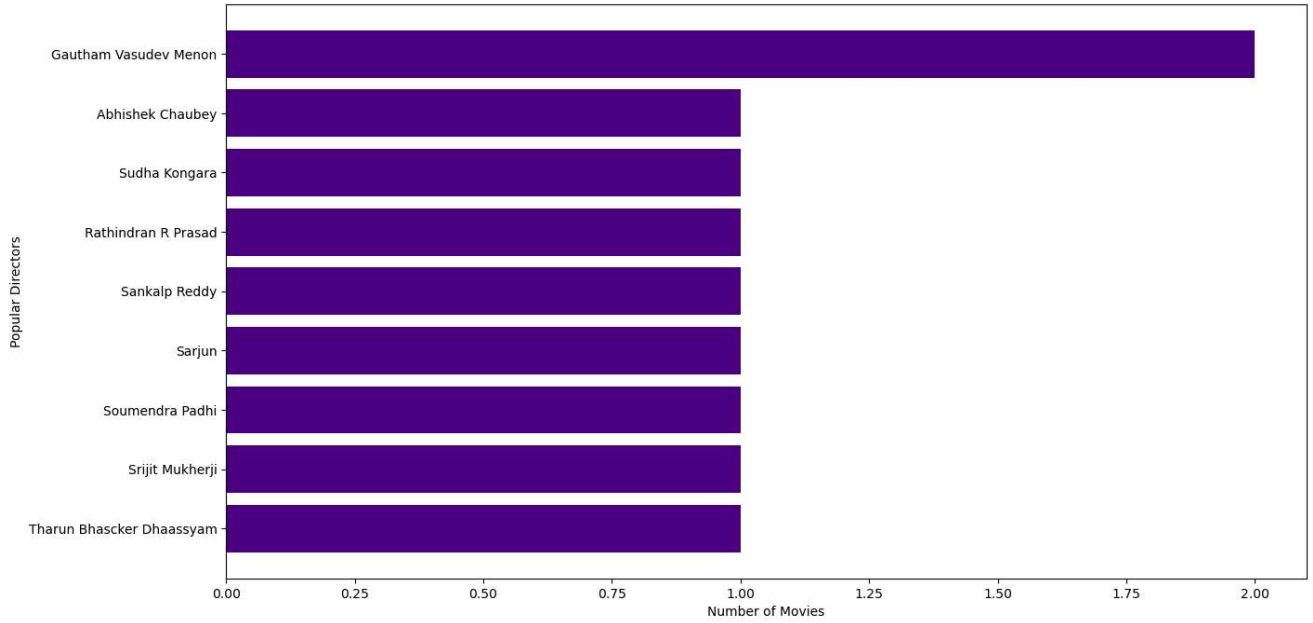
'Kareena Kapoor',

'Ajay Devgn',

'Rajesh Kava',

'Salman Khan'

In [157]:
```
df_directors=df_india_shows.groupby(['Directors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=Fals
df_directors=df_directors[df_directors['Directors']!='Unknown Director']
plt.figure(figsize=(15,8))
plt.barh(df_directors[::-1]['Directors'], df_directors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Movies')
plt.ylabel('Popular Directors')
plt.show()
```



In [158]:
```
df_directors['Directors'].values
```

Out[158]:
```
array(['Gautham Vasudev Menon', 'Abhishek Chaubey', 'Sudha Kongara',
       'Rathindran R Prasad', 'Sankalp Reddy', 'Sarjun',
       'Soumendra Padhi', 'Srijit Mukherji', 'Tharun Bhascker Dhaassyam'],
      dtype=object)
```

Popular Directors Across TV Shows in India:-

'Gautham Vasudev Menon',

'Abhishek Chaubey',

'Sudha Kongara',

'Rathindran R Prasad',

'Sankalp Reddy',

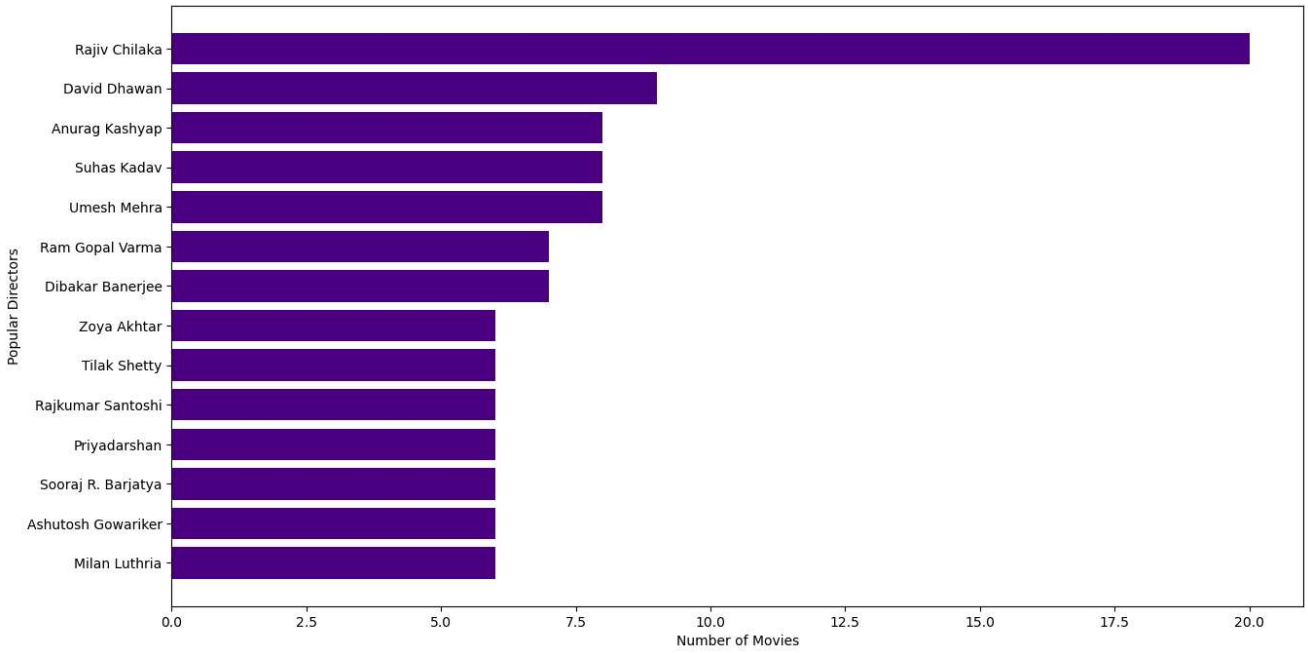
'Sarjun',

'Soumendra Padhi',

'Srijit Mukherji',

'Tharun Bhascker Dhaassyam'

In [159]:
```
df_directors=df_india_movies.groupby(['Directors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=Fal
df_directors=df_directors[df_directors['Directors']!='Unknown Director']
plt.figure(figsize=(15,8))
plt.barh(df_directors[::-1]['Directors'], df_directors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Movies')
plt.ylabel('Popular Directors')
plt.show()
```



In [160]:
```
df_directors['Directors'].values
```

Out[160]:
```
array(['Rajiv Chilaka', 'David Dhawan', 'Anurag Kashyap', 'Suhas Kadav',
       'Umesh Mehra', 'Ram Gopal Varma', 'Dibakar Banerjee',
       'Zoya Akhtar', 'Tilak Shetty', 'Rajkumar Santoshi', 'Priyadarshan',
       'Sooraj R. Barjatya', 'Ashutosh Gowariker', 'Milan Luthria'],
      dtype=object)
```

Popular directors across movies in India:-

'Rajiv Chilaka',

'David Dhawan',

'Anurag Kashyap',

'Suhas Kadav',

'Umesh Mehra',

'Ram Gopal Varma',

'Dibakar Banerjee',

'Zoya Akhtar',

'Tilak Shetty',

'Rajkumar Santoshi',

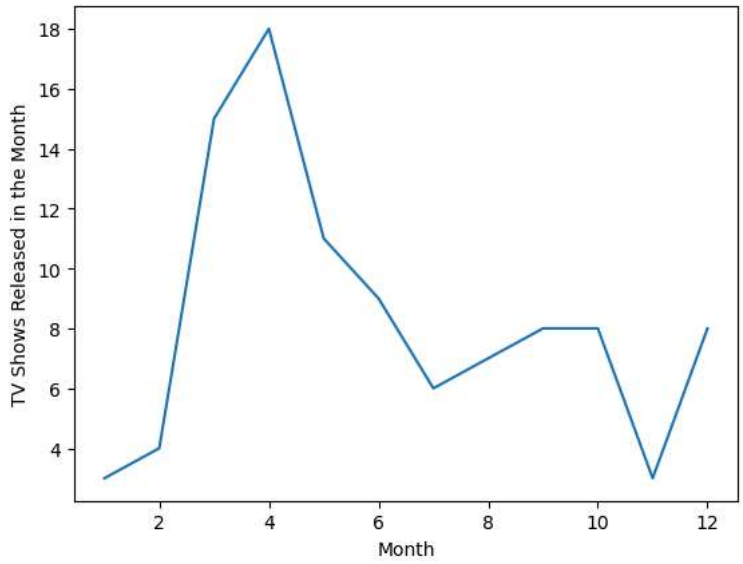
'Priyadarshan',

'Sooraj R. Barjatya',

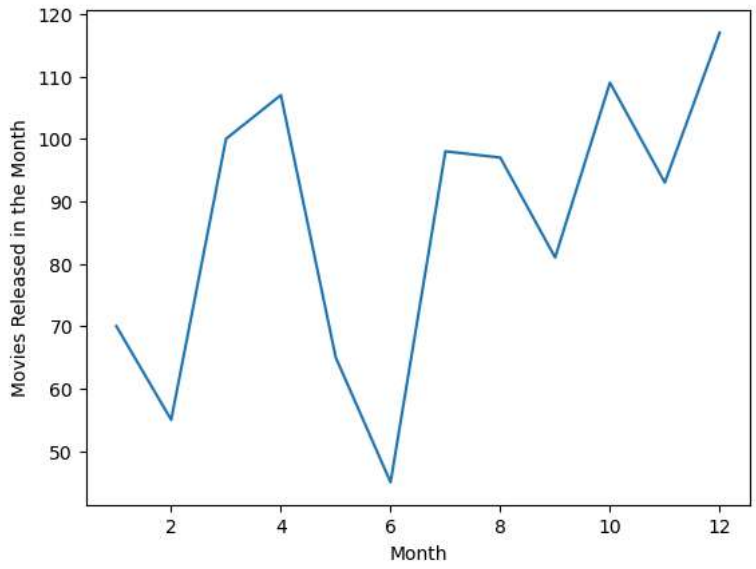
'Ashutosh Gowariker',

'Milan Luthria'

In [166]:
```
df_month=df_india_shows.groupby(['month_added']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_month, x='month_added', y='title')
plt.ylabel("TV Shows Released in the Month")
plt.xlabel("Month")
plt.show()
```



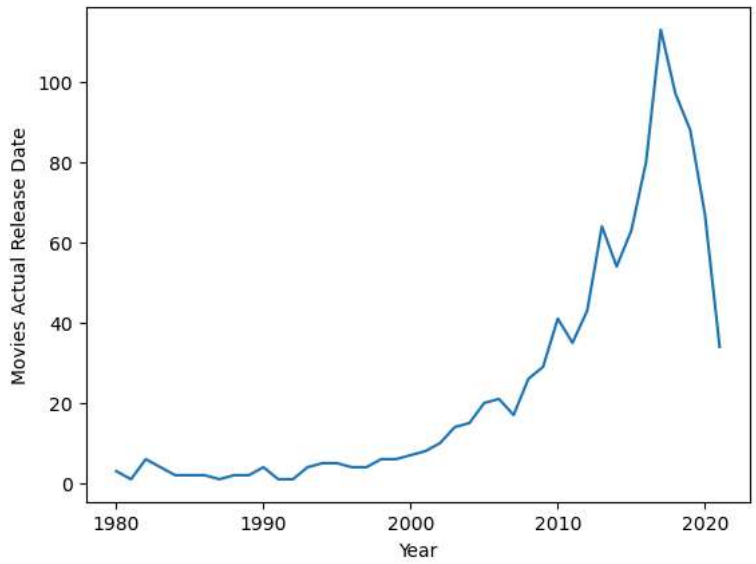
In [167]:
```
df_month=df_india_movies.groupby(['month_added']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_month, x='month_added', y='title')
plt.ylabel("Movies Released in the Month")
plt.xlabel("Month")
plt.show()
```



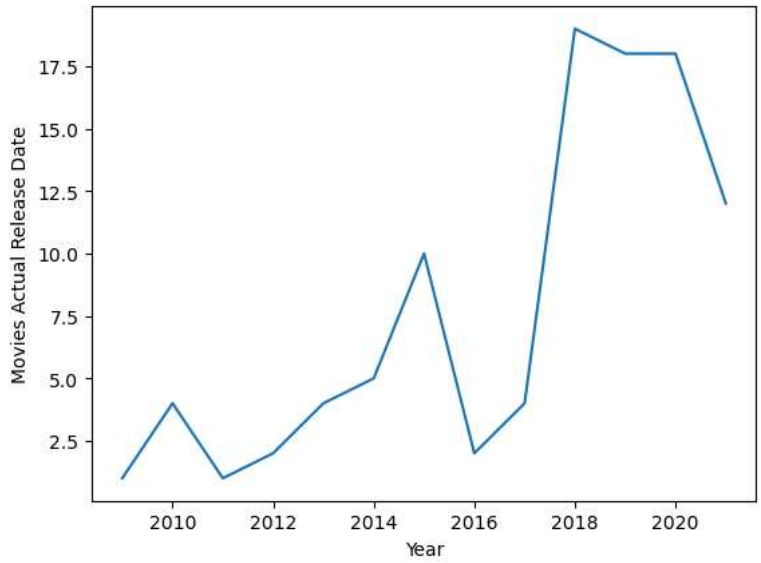
TV Shows are added in Netflix by a tremendous amount in April in India

Movies are added in Netflix in India by a tremendous amount in first week/last month of current year and first month of next year

In [168]:
```
df_release_year=df_india_movies[df_india_movies['release_year']>=1980].groupby(['release_year']).agg({"title":"nunique"}).reset_i
sns.lineplot(data=df_release_year, x='release_year', y='title')
plt.ylabel("Movies Actual Release Date")
plt.xlabel("Year")
plt.show()
```



In [169]:
```
df_release_year=df_india_shows[df_india_shows['release_year']>=1980].groupby(['release_year']).agg({"title":"nunique"}).reset_inc
sns.lineplot(data=df_release_year, x='release_year', y='title')
plt.ylabel("Movies Actual Release Date")
plt.xlabel("Year")
plt.show()
```



The understandable trend amongs movies and TV Shows across India in Netflix is the reduction of movies after 2020

In [171]:
```
#Analysing a combination of actors and directors
df_india_movies['Actor_Director_Combination'] = df_india_movies.Actors.str.cat(df_india_movies.Directors, sep=' and ')
df_india_movies_subset=df_india_movies[df_india_movies['Actors']!='Unknown Actor']
df_india_movies_subset=df_india_movies_subset[df_india_movies_subset['Directors']!='Unknown Director']
df_india_movies_subset.head()
```

Out[171]:

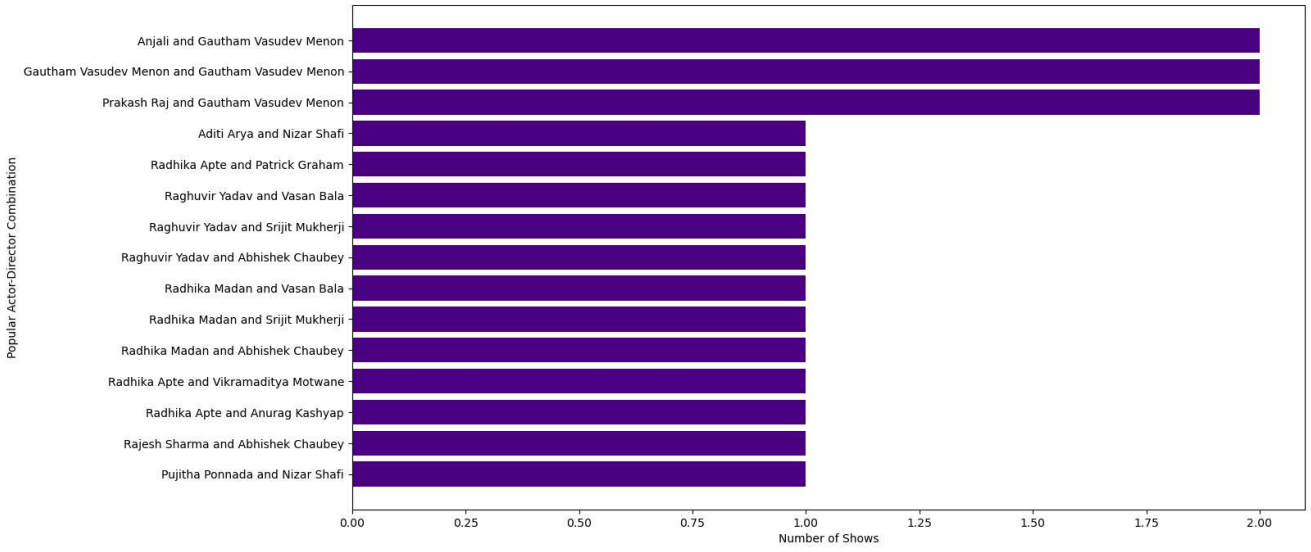
	title	Actors	Directors	Genre	country	show_id	type	date_added	release_year	rating	duration	Modified_Added_date	month_added	w
621	Avvai Shanmughi	Kamal Hassan	K.S. Ravikumar	Comedies	India	s23	Movie	September 21, 2021	1996	TV-PG	150-200	2021-09-21	9	
622	Avvai Shanmughi	Kamal Hassan	K.S. Ravikumar	International Movies	India	s23	Movie	September 21, 2021	1996	TV-PG	150-200	2021-09-21	9	
629	Avvai Shanmughi	Nassar	K.S. Ravikumar	Comedies	India	s23	Movie	September 21, 2021	1996	TV-PG	150-200	2021-09-21	9	
630	Avvai Shanmughi	Nassar	K.S. Ravikumar	International Movies	India	s23	Movie	September 21, 2021	1996	TV-PG	150-200	2021-09-21	9	
641	Jeans	Prashanth	S. Shankar	Comedies	India	s25	Movie	September 21, 2021	1998	TV-14	150-200	2021-09-21	9	

In [172]:
```
df_india_shows['Actor_Director_Combination'] = df_india_shows.Actors.str.cat(df_india_shows.Directors, sep=' and ')
df_india_shows_subset=df_india_shows[df_india_shows['Actors']!='Unknown Actor']
df_india_shows_subset=df_india_shows_subset[df_india_shows_subset['Directors']!='Unknown Director']
df_india_shows_subset.head()
```

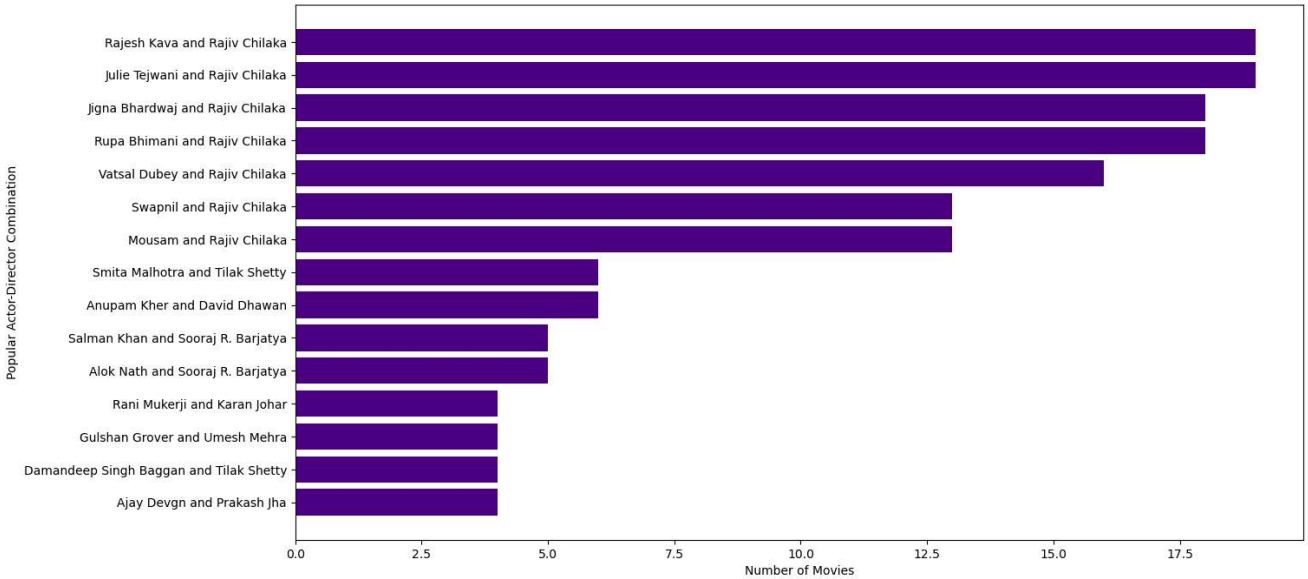
Out[172]:

	title	Actors	Directors	Genre	country	show_id	type	date_added	release_year	rating	duration	Modified_Added_date	month_added	week_A
7005	Navarasa	Suriya	Bejoy Nambiar	TV Shows	India	s298	TV Show	August 6, 2021	2021	TV-MA	1 Season	2021-08-06	8	
7006	Navarasa	Suriya	Priyadarshan	TV Shows	India	s298	TV Show	August 6, 2021	2021	TV-MA	1 Season	2021-08-06	8	
7007	Navarasa	Suriya	Karthik Narain	TV Shows	India	s298	TV Show	August 6, 2021	2021	TV-MA	1 Season	2021-08-06	8	
7008	Navarasa	Suriya	Vasanth Sai	TV Shows	India	s298	TV Show	August 6, 2021	2021	TV-MA	1 Season	2021-08-06	8	
7009	Navarasa	Suriya	Karthik Subbaraj	TV Shows	India	s298	TV Show	August 6, 2021	2021	TV-MA	1 Season	2021-08-06	8	

In [174]:
```
df_actors_directors=df_india_shows_subset.groupby(['Actor_Director_Combination']).agg({"title":"nunique"}).reset_index().sort_val
plt.figure(figsize=(15,8))
plt.barh(df_actors_directors[::-1]['Actor_Director_Combination'], df_actors_directors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Shows')
plt.ylabel('Popular Actor-Director Combination')
plt.show()
```



In [175]:
```
df_actors_directors=df_india_movies_subset.groupby(['Actor_Director_Combination']).agg({"title":"nunique"}).reset_index().sort_va
plt.figure(figsize=(15,8))
plt.barh(df_actors_directors[::-1]['Actor_Director_Combination'], df_actors_directors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Movies')
plt.ylabel('Popular Actor-Director Combination')
plt.show()
```



In [176]:
```
df_india_movies[df_india_movies['Directors']=='Rajiv Chilaka']
```

Out[176]:

	title	Actors	Directors	Genre	country	show_id	type	date_added	release_year	rating	duration	Modified_Added_date	month_addec
10058	Chhota Bheem - Neeli Pahaadi	Vatsal Dubey	Rajiv Chilaka	Children & Family Movies	India	s407	Movie	July 22, 2021	2013	TV-Y7	50-80	2021-07-22	7
10059	Chhota Bheem - Neeli Pahaadi	Julie Tejwani	Rajiv Chilaka	Children & Family Movies	India	s407	Movie	July 22, 2021	2013	TV-Y7	50-80	2021-07-22	7
10060	Chhota Bheem - Neeli Pahaadi	Rupa Bhimani	Rajiv Chilaka	Children & Family Movies	India	s407	Movie	July 22, 2021	2013	TV-Y7	50-80	2021-07-22	7
10061	Chhota Bheem - Neeli Pahaadi	Jigna Bhardwaj	Rajiv Chilaka	Children & Family Movies	India	s407	Movie	July 22, 2021	2013	TV-Y7	50-80	2021-07-22	7

It seems that Rajiv Chilaka has worked on Chota Bheem and has been able to create some good content in its movies. He can be relied on for more Chota Bheem stories

In [177]:
```
df_actors_directors['Actor_Director_Combination'].values
```

Out[177]:
```
array(['Rajesh Kava and Rajiv Chilaka', 'Julie Tejwani and Rajiv Chilaka',
       'Jigna Bhardwaj and Rajiv Chilaka',
       'Rupa Bhimani and Rajiv Chilaka', 'Vatsal Dubey and Rajiv Chilaka',
       'Swapnil and Rajiv Chilaka', 'Mousam and Rajiv Chilaka',
       'Smita Malhotra and Tilak Shetty', 'Anupam Kher and David Dhawan',
       'Salman Khan and Sooraj R. Barjatya',
       'Alok Nath and Sooraj R. Barjatya', 'Rani Mukerji and Karan Johar',
       'Gulshan Grover and Umesh Mehra',
       'Damandeep Singh Baggan and Tilak Shetty',
       'Ajay Devgn and Prakash Jha'], dtype=object)
```

The Most Popular Actor Director Combination in Movies Across India are:-

'Rajesh Kava and Rajiv Chilaka',

'Julie Tejwani and Rajiv Chilaka',

'Jigna Bhardwaj and Rajiv Chilaka',

'Rupa Bhimani and Rajiv Chilaka',

'Vatsal Dubey and Rajiv Chilaka',

'Swapnil and Rajiv Chilaka',

'Mousam and Rajiv Chilaka',

'Smita Malhotra and Tilak Shetty',

'Anupam Kher and David Dhawan',

'Salman Khan and Sooraj R. Barjatya',

'Alok Nath and Sooraj R. Barjatya',

'Rani Mukerji and Karan Johar',

'Gulshan Grover and Umesh Mehra',

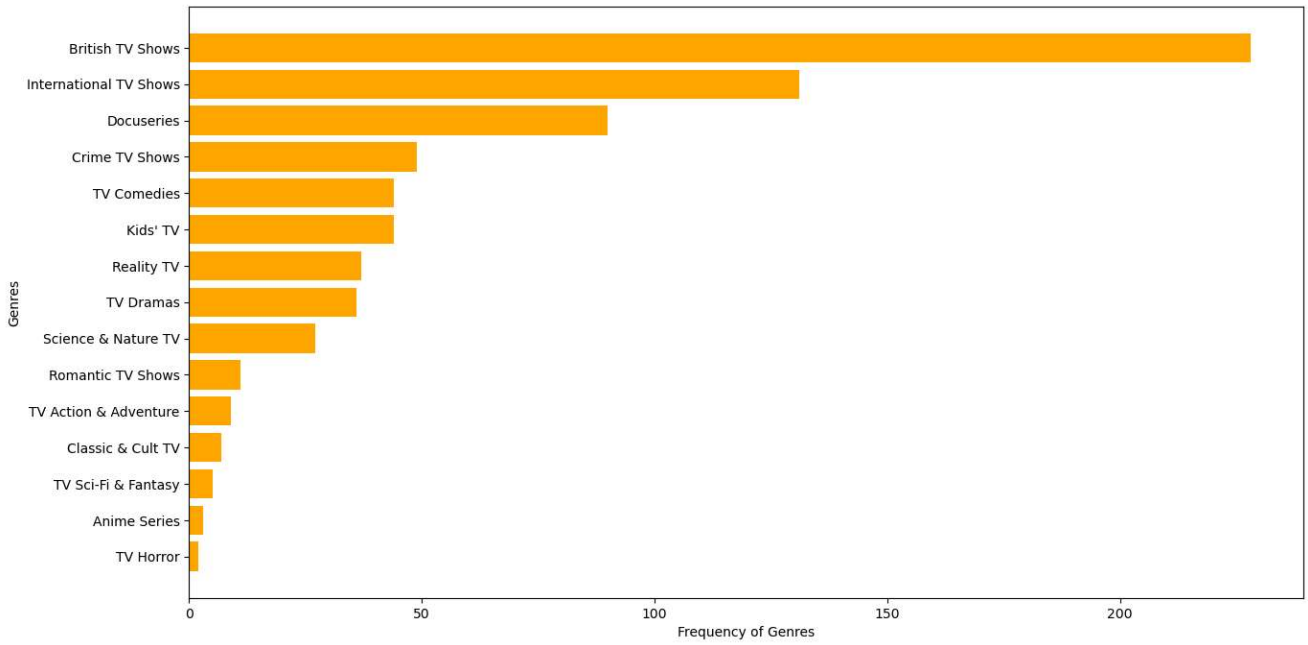
'Damandeep Singh Baggan and Tilak Shetty',

'Ajay Devgn and Prakash Jha'

Univariate Analysis separately for shows and movies in United Kingdom

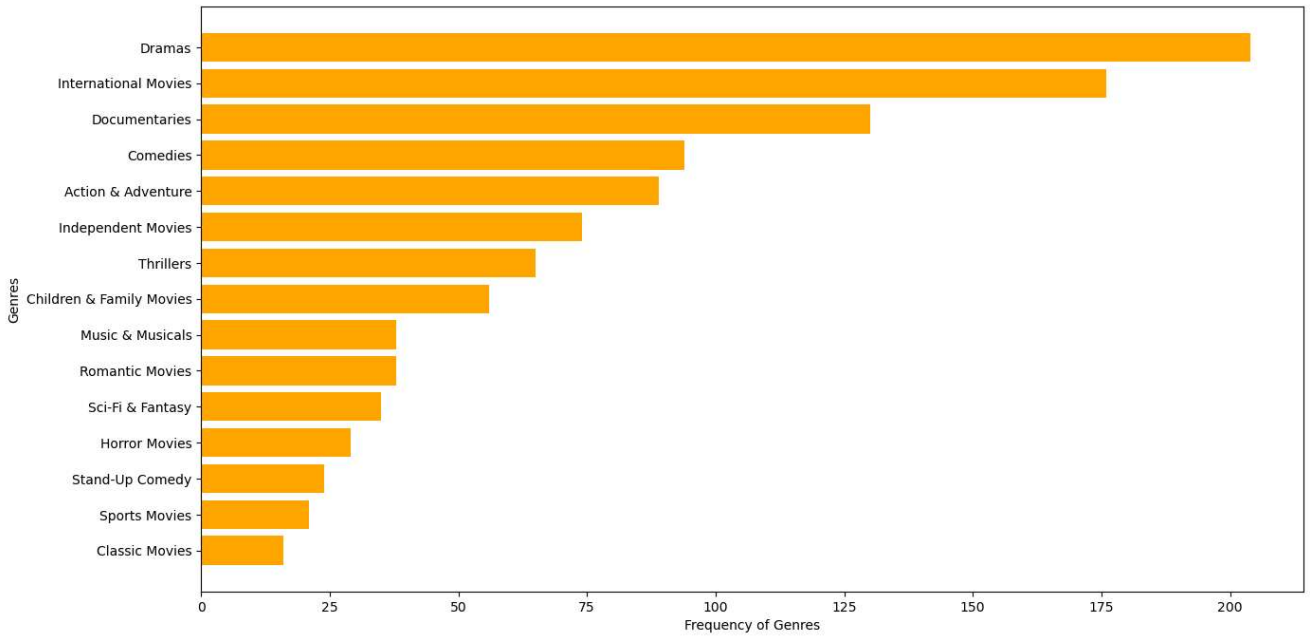
In [178]:
```
#Analyzing India for both shows and movies
df_uk_shows=df_final1[df_final1['country']=='United Kingdom'][df_final1[df_final1['country']=='United Kingdom']['type']=='TV Show
df_uk_movies=df_final1[df_final1['country']=='United Kingdom'][df_final1[df_final1['country']=='United Kingdom']['type']=='Movie'
```

In [179]:
```
df_genre=df_uk_shows.groupby(['Genre']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
plt.figure(figsize=(15,8))
plt.barh(df_genre[::-1]['Genre'], df_genre[::-1]['title'],color=['orange'])
plt.xlabel('Frequency of Genres')
plt.ylabel('Genres')
plt.show()
```



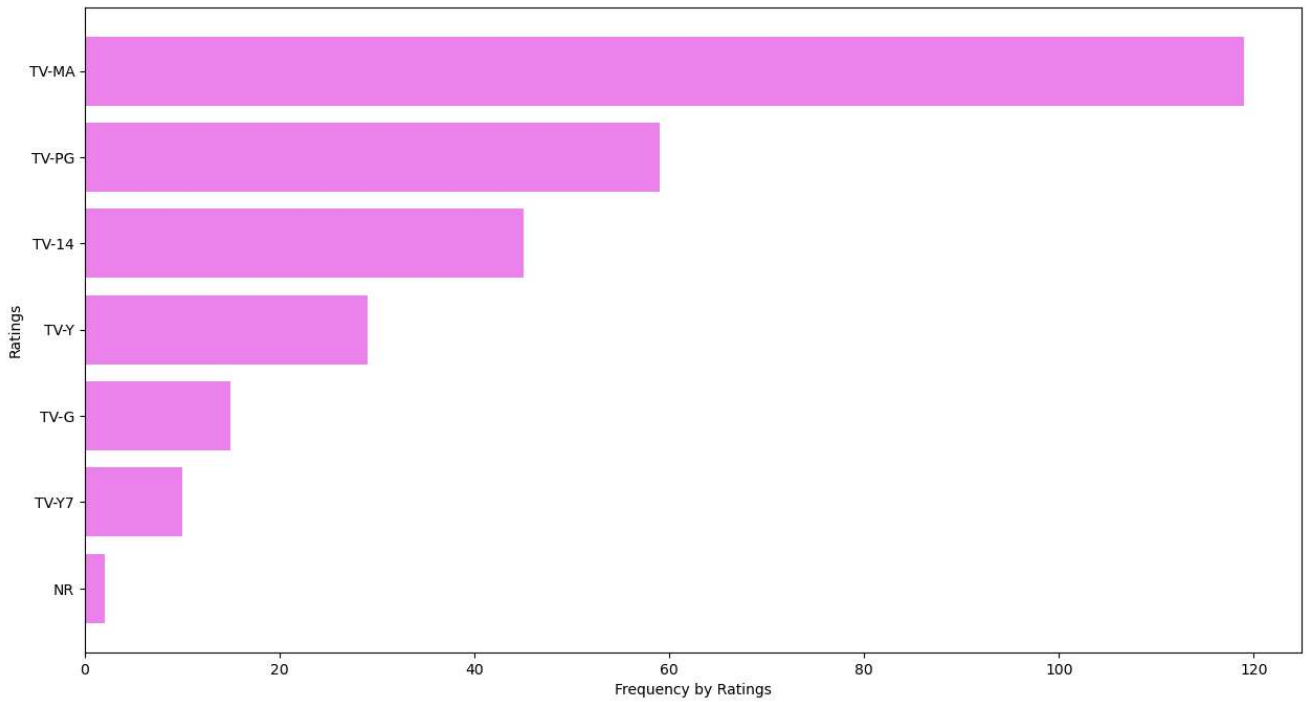
British TV Shows,International TV Shows,Docuseries, Crime, Comedy are widely watched Genres in TV Shows in UK

In [181]:
```
df_genre=df_uk_movies.groupby(['Genre']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
plt.figure(figsize=(15,8))
plt.barh(df_genre[::-1]['Genre'], df_genre[::-1]['title'],color=['orange'])
plt.xlabel('Frequency of Genres')
plt.ylabel('Genres')
plt.show()
```

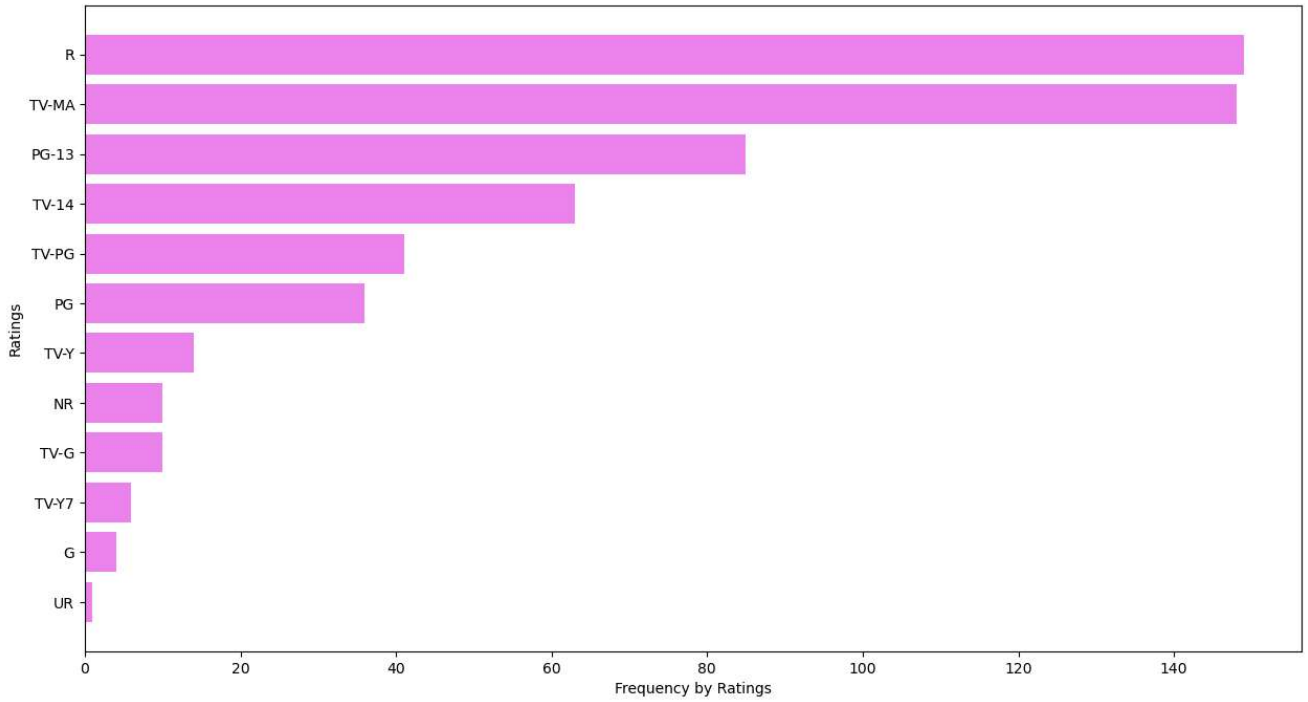


Drama, International Movies,, Documentaries, Comedy and Action Genres in Movies are prevalent in UK

In [183]:
```
df_rating=df_uk_shows.groupby(['rating']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
plt.figure(figsize=(15,8))
plt.barh(df_rating[::-1]['rating'], df_rating[::-1]['title'],color=['violet'])
plt.xlabel('Frequency by Ratings')
plt.ylabel('Ratings')
plt.show()
```

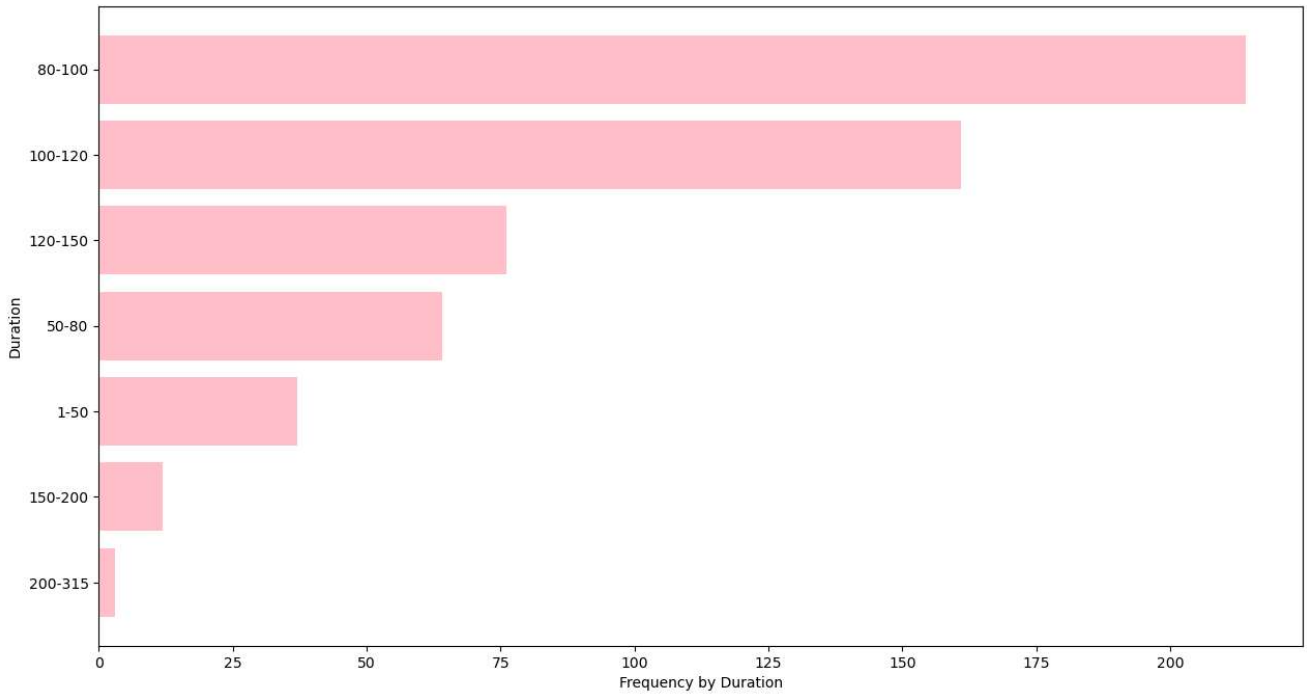


In [185]:
```
df_rating=df_uk_movies.groupby(['rating']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
plt.figure(figsize=(15,8))
plt.barh(df_rating[::-1]['rating'], df_rating[::-1]['title'],color=['violet'])
plt.xlabel('Frequency by Ratings')
plt.ylabel('Ratings')
plt.show()
```



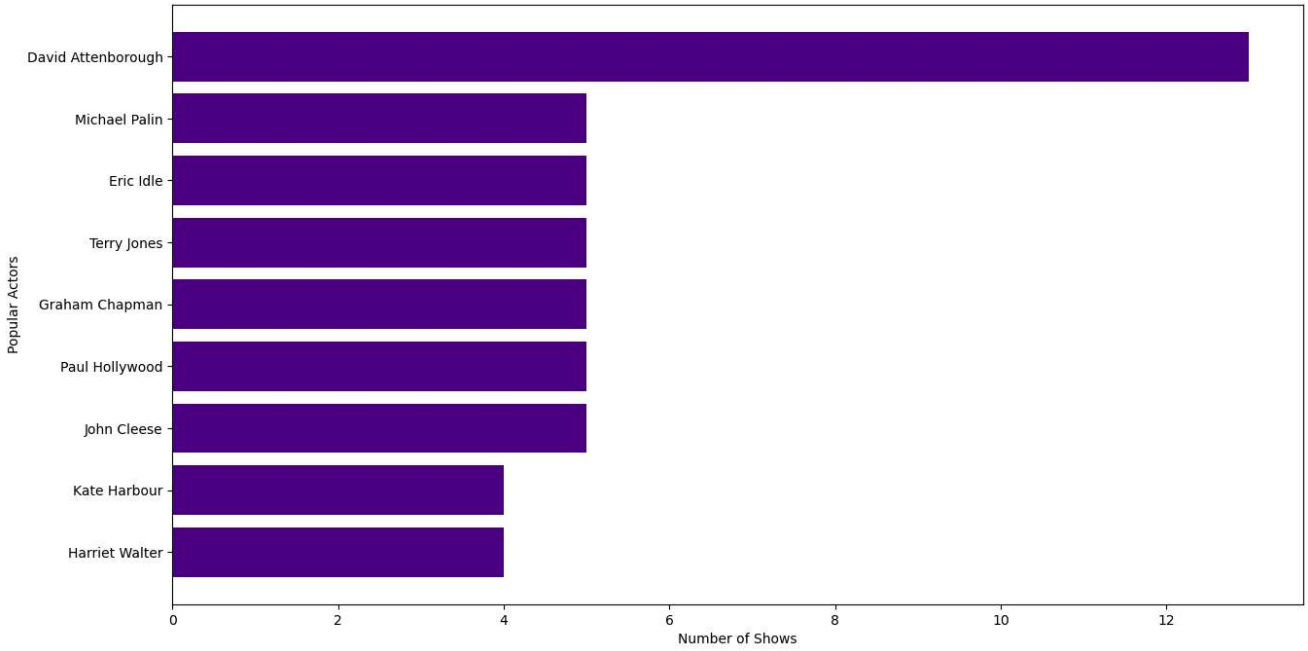
So it seems plaussible to conclude that the popular ratings across Netflix includes Parental Guidance and Mature Audiences in TV Shows and R Rated+MA Rated in Movies in UK

In [186]:
```
df_duration=df_uk_movies.groupby(['duration']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:
plt.figure(figsize=(15,8))
plt.barh(df_duration[::-1]['duration'], df_duration[::-1]['title'],color=['pink'])
plt.xlabel('Frequency by Duration')
plt.ylabel('Duration')
plt.show()
```



Across movies ranges of minutes in UK have a sweet spot at 80-120 mins.

In [187]:
```
df_actors=df_uk_shows.groupby(['Actors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:10]
df_actors=df_actors[df_actors['Actors']!='Unknown Actor']
plt.figure(figsize=(15,8))
plt.barh(df_actors[::-1]['Actors'], df_actors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Shows')
plt.ylabel('Popular Actors')
plt.show()
```



In [188]:
```
df_actors['Actors'].values
```

Out[188]:
```
array(['David Attenborough', 'Michael Palin', 'Eric Idle', 'Terry Jones',
       'Graham Chapman', 'Paul Hollywood', 'John Cleese', 'Kate Harbour',
       'Harriet Walter'], dtype=object)
```

Popular Actors in TV Shows in UK are:

'David Attenborough',

'Michael Palin',

'Eric Idle',

'Terry Jones',

'Graham Chapman',

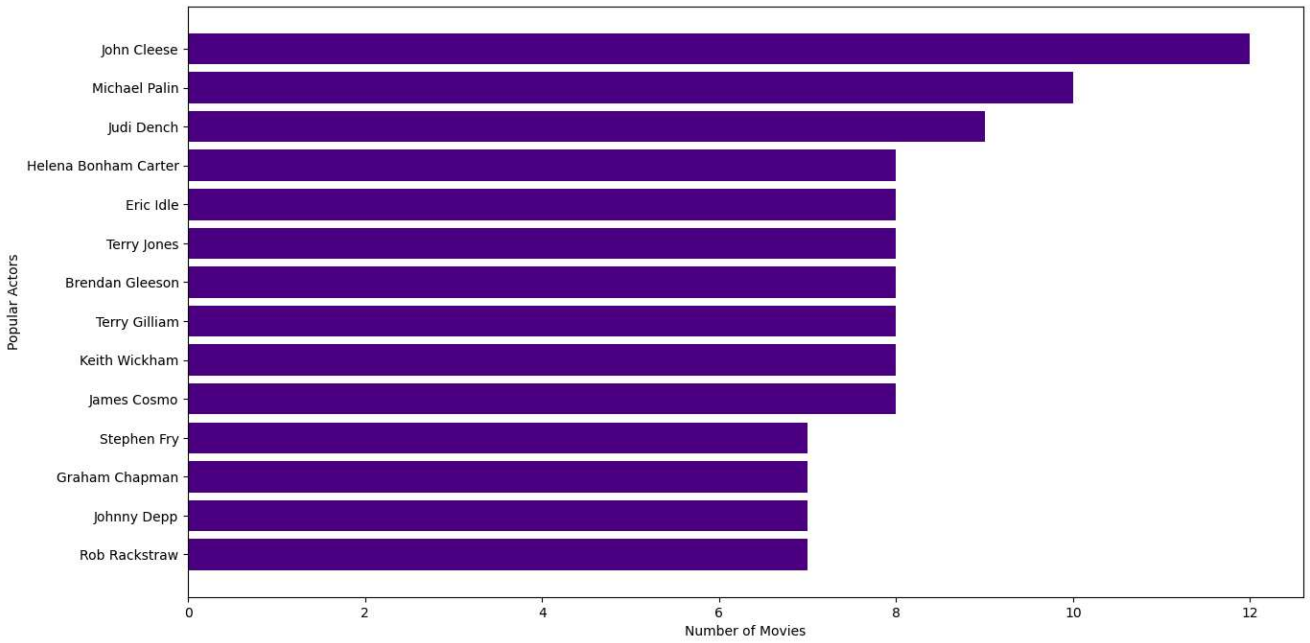
'Paul Hollywood',

'John Cleese',

'Kate Harbour',

'Harriet Walter'

In [190]:
```
df_actors=df_uk_movies.groupby(['Actors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
df_actors=df_actors[df_actors['Actors']!='Unknown Actor']
plt.figure(figsize=(15,8))
plt.barh(df_actors[::-1]['Actors'], df_actors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Movies')
plt.ylabel('Popular Actors')
plt.show()
```



In [191]:
```
df_actors['Actors'].values
```

Out[191]:
```
array(['John Cleese', 'Michael Palin', 'Judi Dench',
       'Helena Bonham Carter', 'Eric Idle', 'Terry Jones',
       'Brendan Gleeson', 'Terry Gilliam', 'Keith Wickham', 'James Cosmo',
       'Stephen Fry', 'Graham Chapman', 'Johnny Depp', 'Rob Rackstraw'],
      dtype=object)
```

Popular actors across Movies in UK:-

'John Cleese',

'Michael Palin',

'Judi Dench',

'Keith Wickham',

'Eric Idle',

'Brendan Gleeson',

'Terry Gilliam',

'Terry Jones',

'Helena Bonham Carter',

'Graham Chapman',

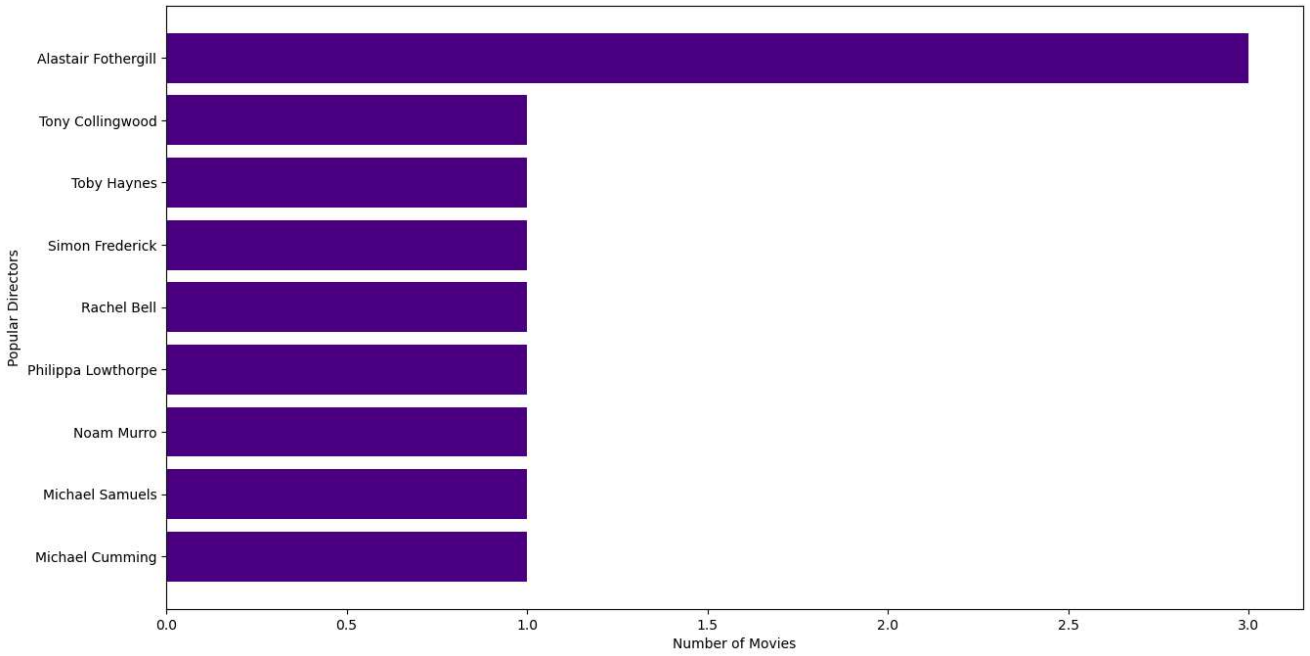
'Samuel West',

'Eddie Marsan',

'James Cosmo',

'Rob Rackstraw'

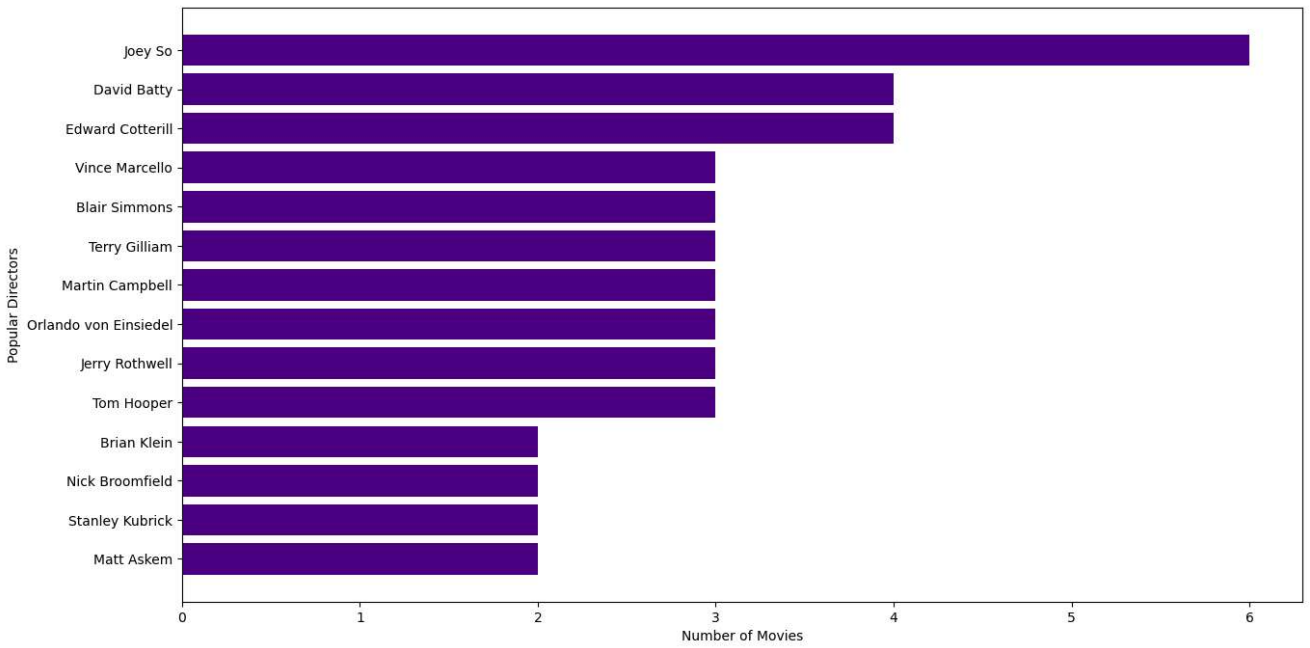
In [193]:
```
df_directors=df_uk_shows.groupby(['Directors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[
df_directors=df_directors[df_directors['Directors']!='Unknown Director']
plt.figure(figsize=(15,8))
plt.barh(df_directors[::-1]['Directors'], df_directors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Movies')
plt.ylabel('Popular Directors')
plt.show()
```



In [194]:
```
df_directors['Directors'].values
```

Out[194]:
```
array(['Alastair Fothergill', 'Tony Collingwood', 'Toby Haynes',
       'Simon Frederick', 'Rachel Bell', 'Philippa Lowthorpe',
       'Noam Murro', 'Michael Samuels', 'Michael Cumming'], dtype=object)
```

In [195]:
```
df_directors=df_uk_movies.groupby(['Directors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)
df_directors=df_directors[df_directors['Directors']!='Unknown Director']
plt.figure(figsize=(15,8))
plt.barh(df_directors[::-1]['Directors'], df_directors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Movies')
plt.ylabel('Popular Directors')
plt.show()
```



In [196]:
```
df_directors['Directors'].values
```

Out[196]:
```
array(['Joey So', 'David Batty', 'Edward Cotterill', 'Vince Marcello',
       'Blair Simmons', 'Terry Gilliam', 'Martin Campbell',
       'Orlando von Einsiedel', 'Jerry Rothwell', 'Tom Hooper',
       'Brian Klein', 'Nick Broomfield', 'Stanley Kubrick', 'Matt Askem'],
      dtype=object)
```

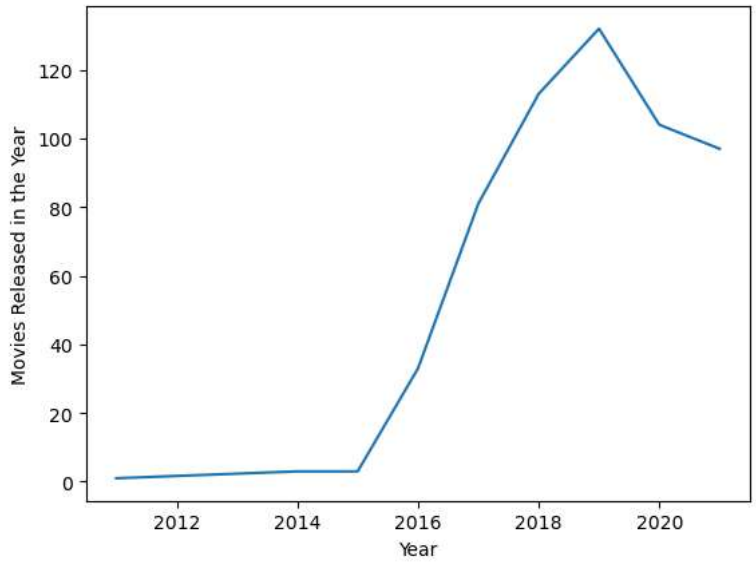
Popular directors across movies in UK:-

'Joey So',

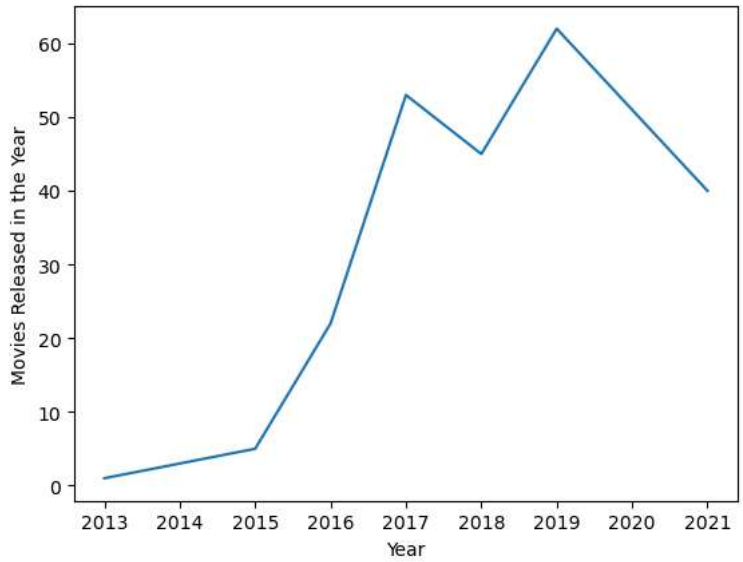
'David Batty',

'Edward Cotterill'

In [197]:
```
df_year=df_uk_movies.groupby(['year']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_year, x='year', y='title')
plt.ylabel("Movies Released in the Year")
plt.xlabel("Year")
plt.show()
```



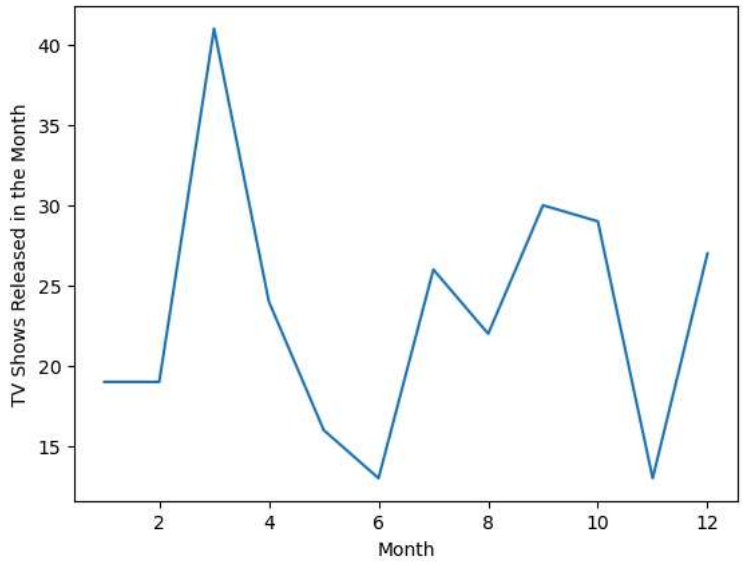
In [198]:
```
df_year=df_uk_shows.groupby(['year']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_year, x='year', y='title')
plt.ylabel("Movies Released in the Year")
plt.xlabel("Year")
plt.show()
```



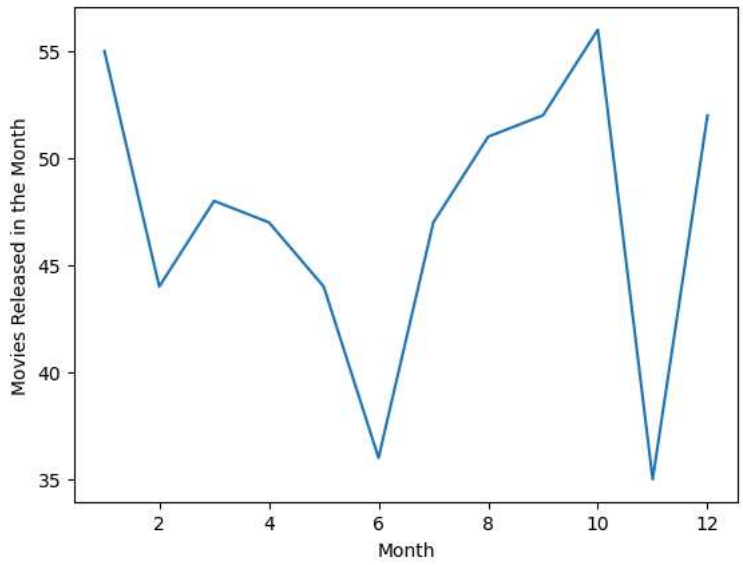
In terms of TV Shows, UK saw a downfall in 2018 from 2017, then a great increase in 2019 but has been reducing since then.

In terms of Movies the number of popular movies in UK increased till 2019, since then it's decreasing

In [200]:
```
df_month=df_uk_shows.groupby(['month_added']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_month, x='month_added', y='title')
plt.ylabel("TV Shows Released in the Month")
plt.xlabel("Month")
plt.show()
```



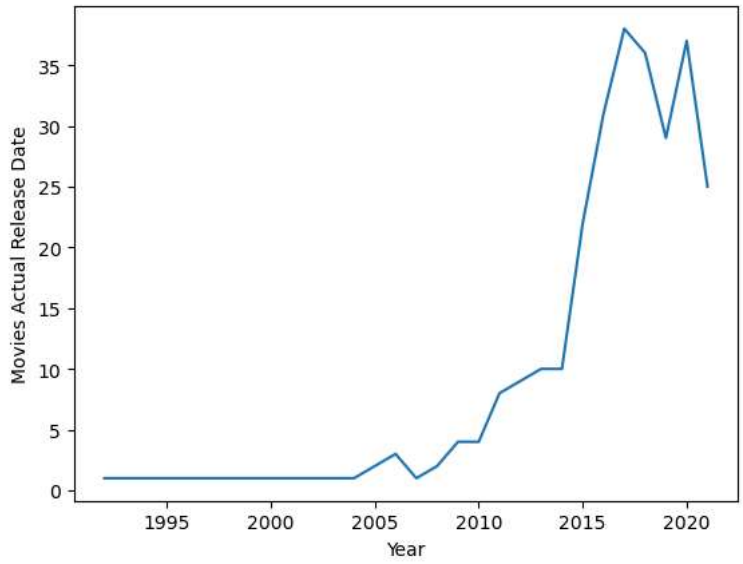
In [201]:
```
df_month=df_uk_movies.groupby(['month_added']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_month, x='month_added', y='title')
plt.ylabel("Movies Released in the Month")
plt.xlabel("Month")
plt.show()
```



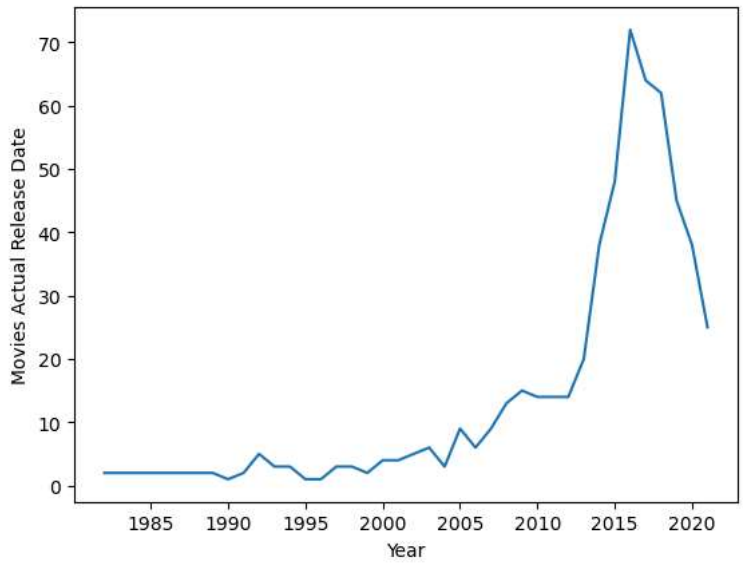
TV Shows are added in Netflix by a tremendous amount in March in UK

Movies are added in Netflix in India by a tremendous amount in first week/last month of current year and first month of next year

In [202]:
```
df_release_year=df_uk_shows[df_uk_shows['release_year']>=1980].groupby(['release_year']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_release_year, x='release_year', y='title')
plt.ylabel("Movies Actual Release Date")
plt.xlabel("Year")
plt.show()
```



In [203]:
```
df_release_year=df_uk_movies[df_uk_movies['release_year']>=1980].groupby(['release_year']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_release_year, x='release_year', y='title')
plt.ylabel("Movies Actual Release Date")
plt.xlabel("Year")
plt.show()
```



Same trend of reduction in movies and shows after 2020.

In [204]:
```
#Analysing a combination of actors and directors
df_uk_movies['Actor_Director_Combination'] = df_uk_movies.Actors.str.cat(df_uk_movies.Directors, sep=' and ')
df_uk_movies_subset=df_uk_movies[df_uk_movies['Actors']!='Unknown Actor']
df_uk_movies_subset=df_uk_movies_subset[df_uk_movies_subset['Directors']!='Unknown Director']
df_uk_movies_subset.head()
```

Out[204]:

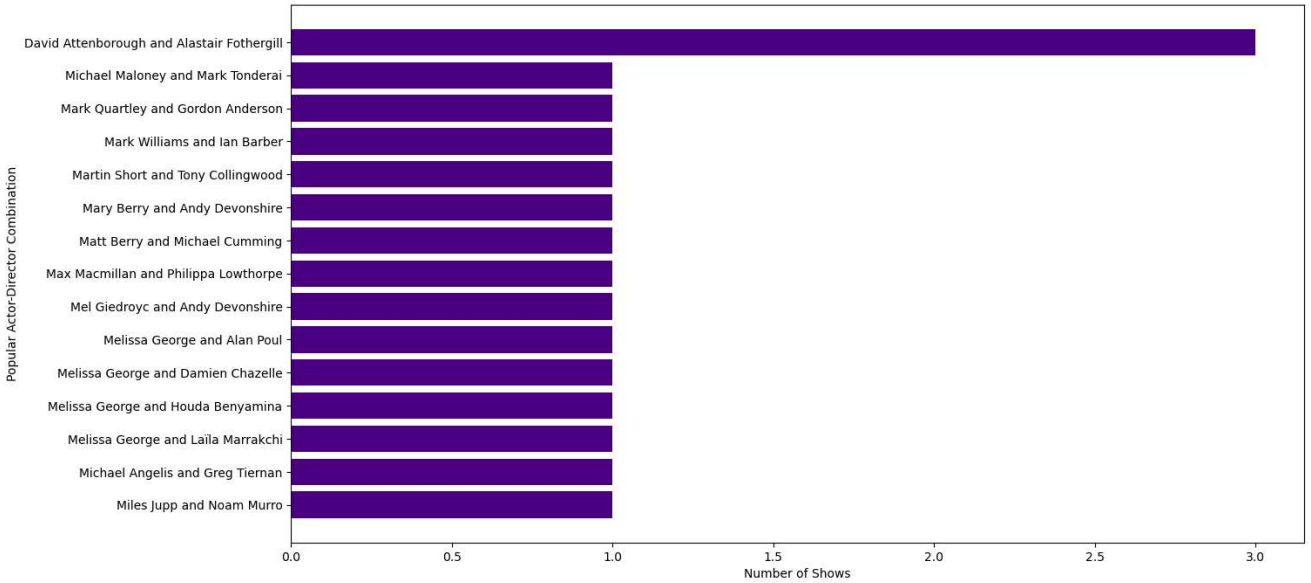
	title	Actors	Directors	Genre	country	show_id	type	date_added	release_year	rating	duration	Modified_Added_date	month_added	we
182	Sankofa	Kofi Ghanaba	Haile Gerima	Dramas	United Kingdom	s8	Movie	September 24, 2021	1993	TV-MA	120-150	2021-09-24	9	
188	Sankofa	Kofi Ghanaba	Haile Gerima	Independent Movies	United Kingdom	s8	Movie	September 24, 2021	1993	TV-MA	120-150	2021-09-24	9	
194	Sankofa	Kofi Ghanaba	Haile Gerima	International Movies	United Kingdom	s8	Movie	September 24, 2021	1993	TV-MA	120-150	2021-09-24	9	
200	Sankofa	Oyafunmike Ogunlano	Haile Gerima	Dramas	United Kingdom	s8	Movie	September 24, 2021	1993	TV-MA	120-150	2021-09-24	9	
206	Sankofa	Oyafunmike Ogunlano	Haile Gerima	Independent Movies	United Kingdom	s8	Movie	September 24, 2021	1993	TV-MA	120-150	2021-09-24	9	

In [205]:
```
df_uk_shows['Actor_Director_Combination'] = df_uk_shows.Actors.str.cat(df_uk_shows.Directors, sep=' and ')
df_uk_shows_subset=df_uk_shows[df_uk_shows['Actors']!='Unknown Actor']
df_uk_shows_subset=df_uk_shows_subset[df_uk_shows_subset['Directors']!='Unknown Director']
df_uk_shows_subset.head()
```

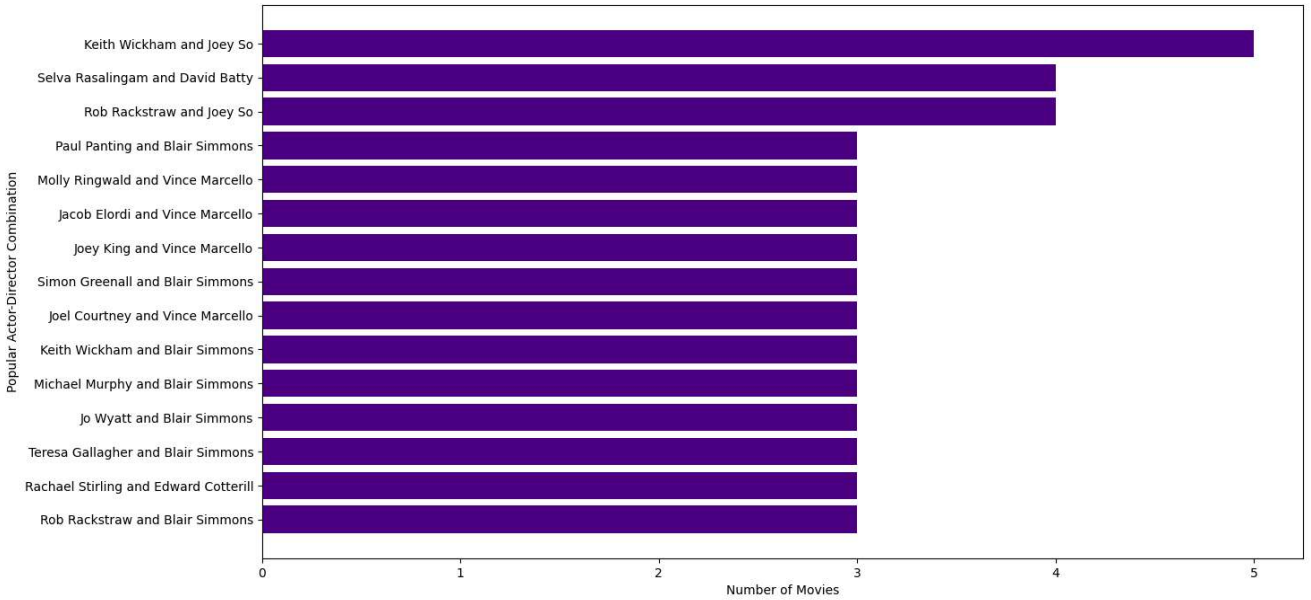
Out[205]:

	title	Actors	Directors	Genre	country	show_id	type	date_added	release_year	rating	duration	Modified_Added_date	month_added	week_Adde
323	The Great British Baking Show	Mel Giedroyc	Andy Devonshire	British TV Shows	United Kingdom	s9	TV Show	September 24, 2021	2021	TV-14	9 Seasons	2021-09-24	9	3
324	The Great British Baking Show	Mel Giedroyc	Andy Devonshire	Reality TV	United Kingdom	s9	TV Show	September 24, 2021	2021	TV-14	9 Seasons	2021-09-24	9	3
325	The Great British Baking Show	Sue Perkins	Andy Devonshire	British TV Shows	United Kingdom	s9	TV Show	September 24, 2021	2021	TV-14	9 Seasons	2021-09-24	9	3
326	The Great British Baking Show	Sue Perkins	Andy Devonshire	Reality TV	United Kingdom	s9	TV Show	September 24, 2021	2021	TV-14	9 Seasons	2021-09-24	9	3
327	The Great British Baking Show	Mary Berry	Andy Devonshire	British TV Shows	United Kingdom	s9	TV Show	September 24, 2021	2021	TV-14	9 Seasons	2021-09-24	9	3

In [206]:
```
df_actors_directors=df_uk_shows_subset.groupby(['Actor_Director_Combination']).agg({"title":"nunique"}).reset_index().sort_values
plt.figure(figsize=(15,8))
plt.barh(df_actors_directors[::-1]['Actor_Director_Combination'], df_actors_directors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Shows')
plt.ylabel('Popular Actor-Director Combination')
plt.show()
```



In [207]:
```
df_actors_directors=df_uk_movies_subset.groupby(['Actor_Director_Combination']).agg({"title":"nunique"}).reset_index().sort_value
plt.figure(figsize=(15,8))
plt.barh(df_actors_directors[::-1]['Actor_Director_Combination'], df_actors_directors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Movies')
plt.ylabel('Popular Actor-Director Combination')
plt.show()
```



In [208]:
```
df_actors_directors['Actor_Director_Combination'].values
```

Out[208]:
```
array(['Keith Wickham and Joey So', 'Selva Rasalingam and David Batty',
       'Rob Rackstraw and Joey So', 'Paul Panting and Blair Simmons',
       'Molly Ringwald and Vince Marcello',
       'Jacob Elordi and Vince Marcello', 'Joey King and Vince Marcello',
       'Simon Greenall and Blair Simmons',
       'Joel Courtney and Vince Marcello',
       'Keith Wickham and Blair Simmons',
       'Michael Murphy and Blair Simmons', 'Jo Wyatt and Blair Simmons',
       'Teresa Gallagher and Blair Simmons',
       'Rachael Stirling and Edward Cotterill',
       'Rob Rackstraw and Blair Simmons'], dtype=object)
```

The Most Popular Actor Director Combination in Movies Across UK are:-

'Keith Wickham and Joey So',

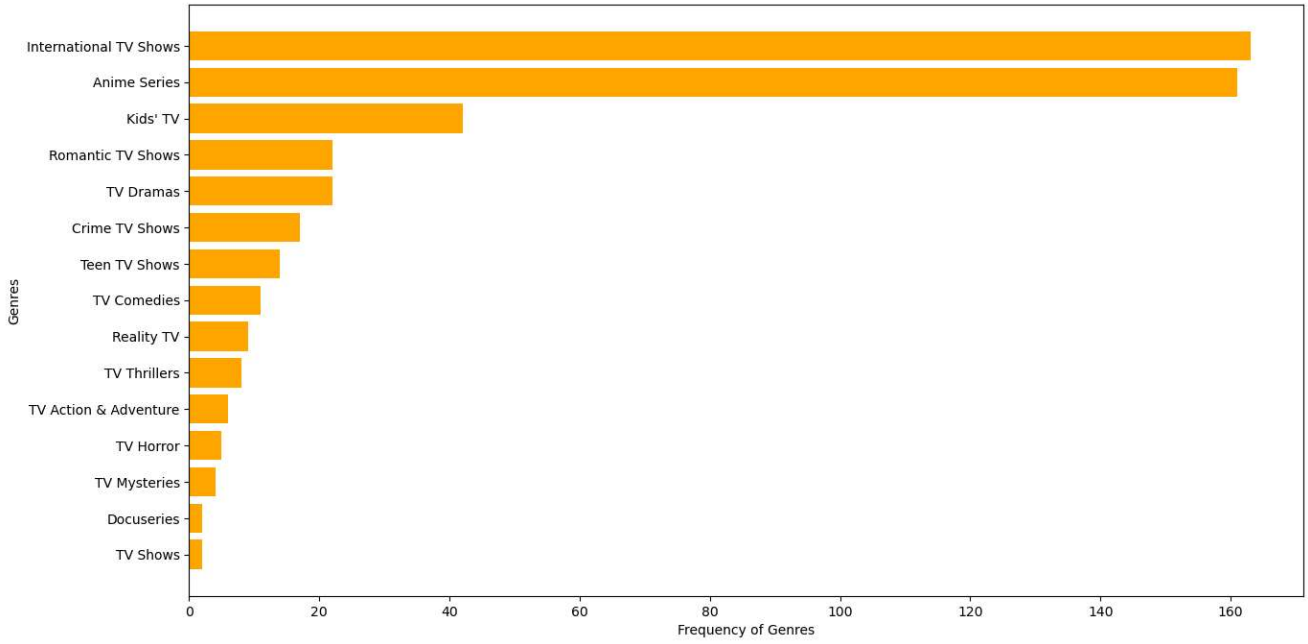
'Selva Rasalingam and David Batty',

'Rob Rackstraw and Joey So'

Univariate Analysis separately for shows in Japan

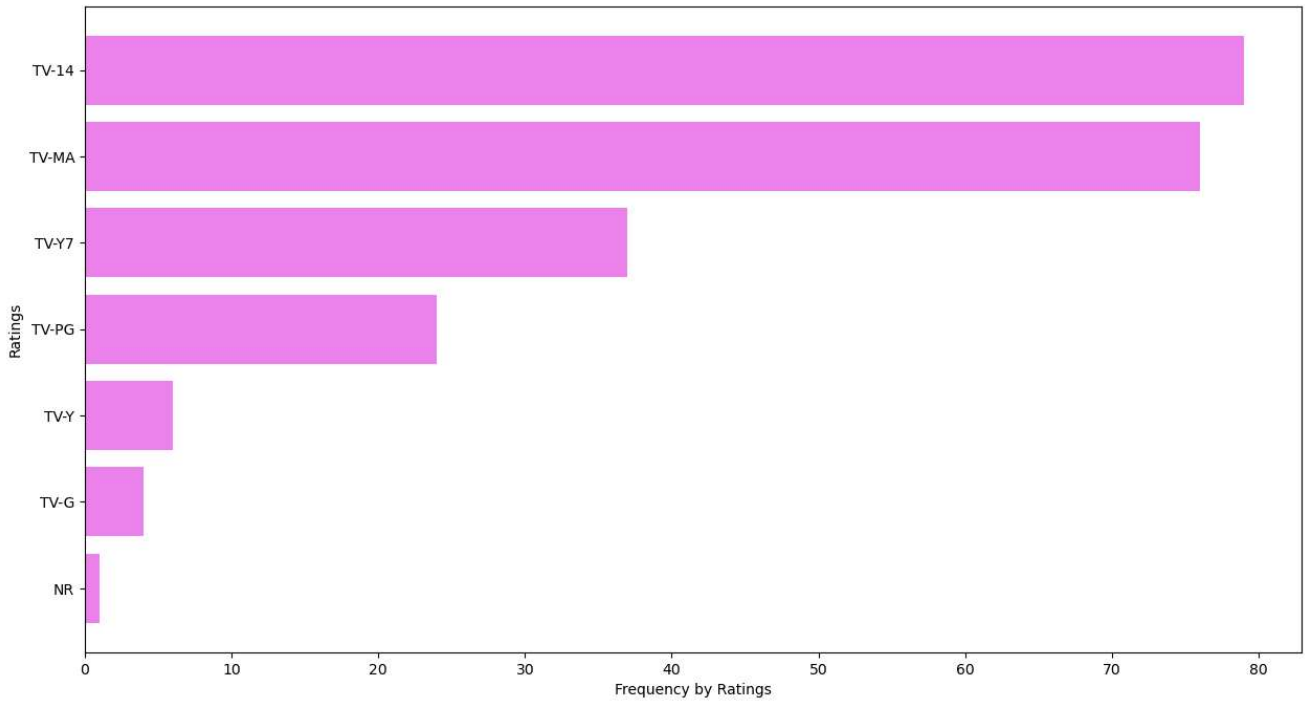
In [209]:
```
#Analyzing India for both shows and movies
df_japan_shows=df_final1[df_final1['country']=='Japan'][df_final1[df_final1['country']=='Japan']['type']=='TV Show']
```

In [210]:
```
df_genre=df_japan_shows.groupby(['Genre']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
plt.figure(figsize=(15,8))
plt.barh(df_genre[::-1]['Genre'], df_genre[::-1]['title'],color=['orange'])
plt.xlabel('Frequency of Genres')
plt.ylabel('Genres')
plt.show()
```



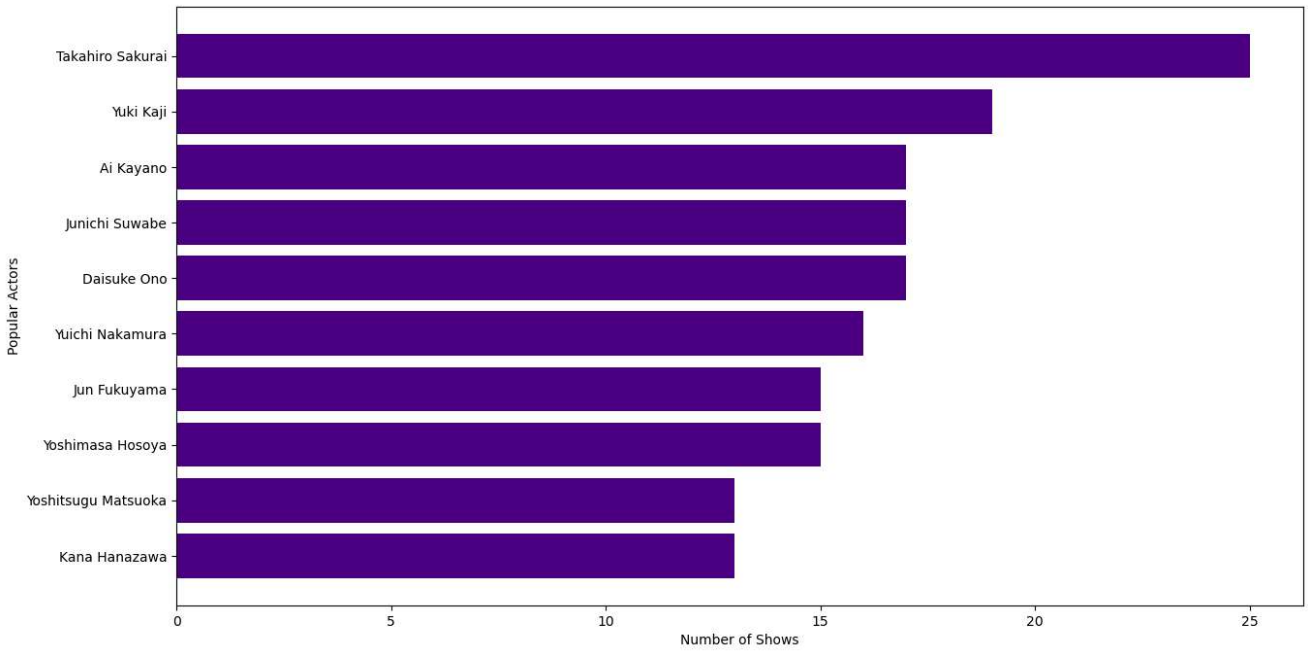
International TV Shows and Anime Genres are popular in TV Shows in Japan

In [211]:
```
df_rating=df_japan_shows.groupby(['rating']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15
plt.figure(figsize=(15,8))
plt.barh(df_rating[::-1]['rating'], df_rating[::-1]['title'],color=['violet'])
plt.xlabel('Frequency by Ratings')
plt.ylabel('Ratings')
plt.show()
```



So it seems plaussible to conclude that the popular ratings across Netflix includes TV-14 Mature Audiences in TV Shows

In [213]:
```
df_actors=df_japan_shows.groupby(['Actors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:10
df_actors=df_actors[df_actors['Actors']!='Unknown Actor']
plt.figure(figsize=(15,8))
plt.barh(df_actors[::-1]['Actors'], df_actors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Shows')
plt.ylabel('Popular Actors')
plt.show()
```



In [214]:
```
df_actors['Actors'].values
```

Out[214]:
```
array(['Takahiro Sakurai', 'Yuki Kaji', 'Ai Kayano', 'Junichi Suwabe',
       'Daisuke Ono', 'Yuichi Nakamura', 'Jun Fukuyama',
       'Yoshimasa Hosoya', 'Yoshitsugu Matsuoka', 'Kana Hanazawa'],
      dtype=object)
```

Popular Actors in TV Shows in Japan are:-

'Takahiro Sakurai',

'Yuki Kaji',

'Ai Kayano',

'Junichi Suwabe',

'Daisuke Ono',

'Yuichi Nakamura',

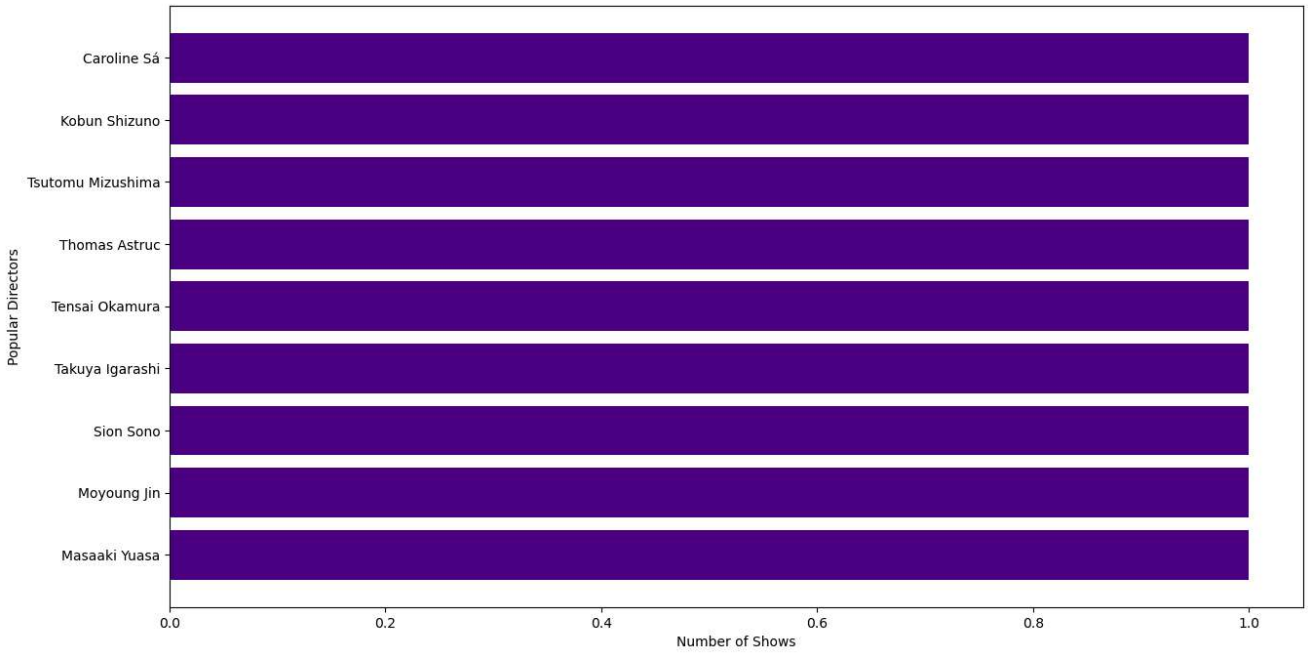
'Jun Fukuyama',

'Yoshimasa Hosoya',

'Yoshitsugu Matsuoka',

'Kana Hanazawa'

In [215]:
```
df_directors=df_japan_shows.groupby(['Directors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=Fals
df_directors=df_directors[df_directors['Directors']!='Unknown Director']
plt.figure(figsize=(15,8))
plt.barh(df_directors[::-1]['Directors'], df_directors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Shows')
plt.ylabel('Popular Directors')
plt.show()
```

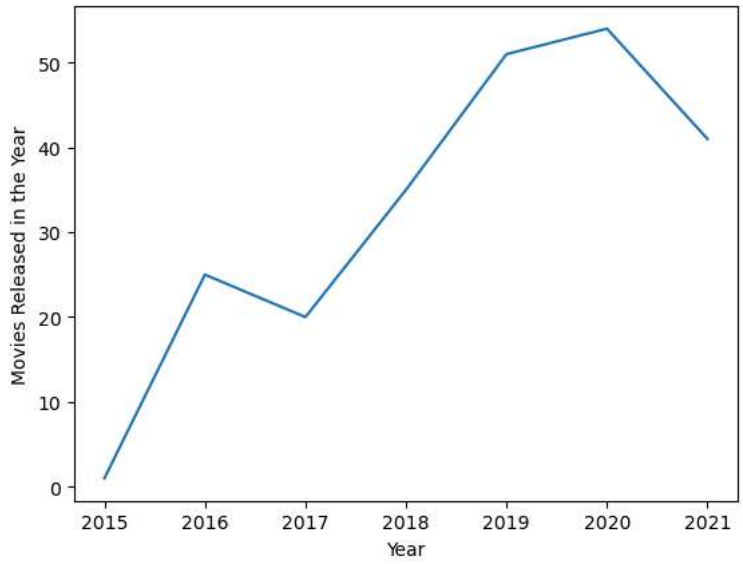


In [216]:
```
df_directors['Directors'].values
```

Out[216]:
```
array(['Caroline Sá', 'Kobun Shizuno', 'Tsutomu Mizushima',
       'Thomas Astruc', 'Tensai Okamura', 'Takuya Igarashi', 'Sion Sono',
       'Moyoung Jin', 'Masaaki Yuasa'], dtype=object)
```

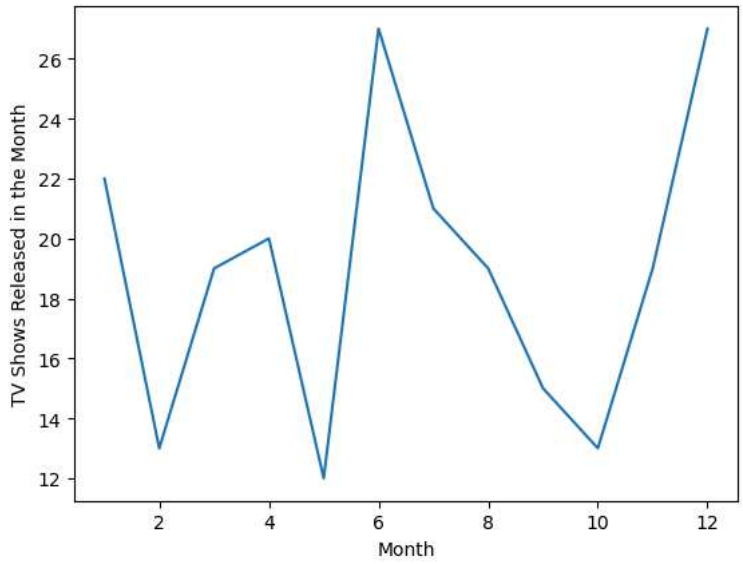
All Directors are one time directors only

In [217]:
```
df_year=df_japan_shows.groupby(['year']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_year, x='year', y='title')
plt.ylabel("Movies Released in the Year")
plt.xlabel("Year")
plt.show()
```



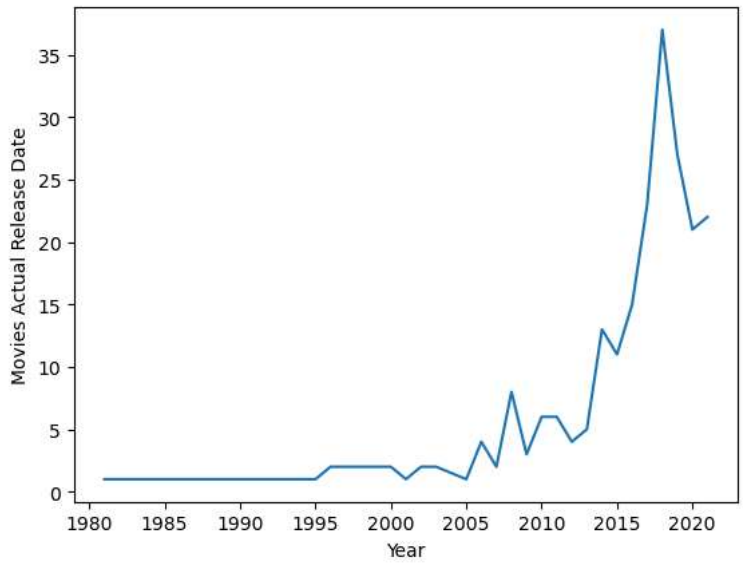
In Japan, TV Shows have diminished in 2017 from 2016 and then increased till 2020 after which it has reduced in 2021.

In [218]:
```
df_month=df_japan_shows.groupby(['month_added']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_month, x='month_added', y='title')
plt.ylabel("TV Shows Released in the Month")
plt.xlabel("Month")
plt.show()
```



TV Shows are added in Netflix by significant numbers in April and January in Japan

In [219]:
```
df_release_year=df_japan_shows[df_japan_shows['release_year']>=1980].groupby(['release_year']).agg({"title":"nunique"}).reset_inc
sns.lineplot(data=df_release_year, x='release_year', y='title')
plt.ylabel("Movies Actual Release Date")
plt.xlabel("Year")
plt.show()
```

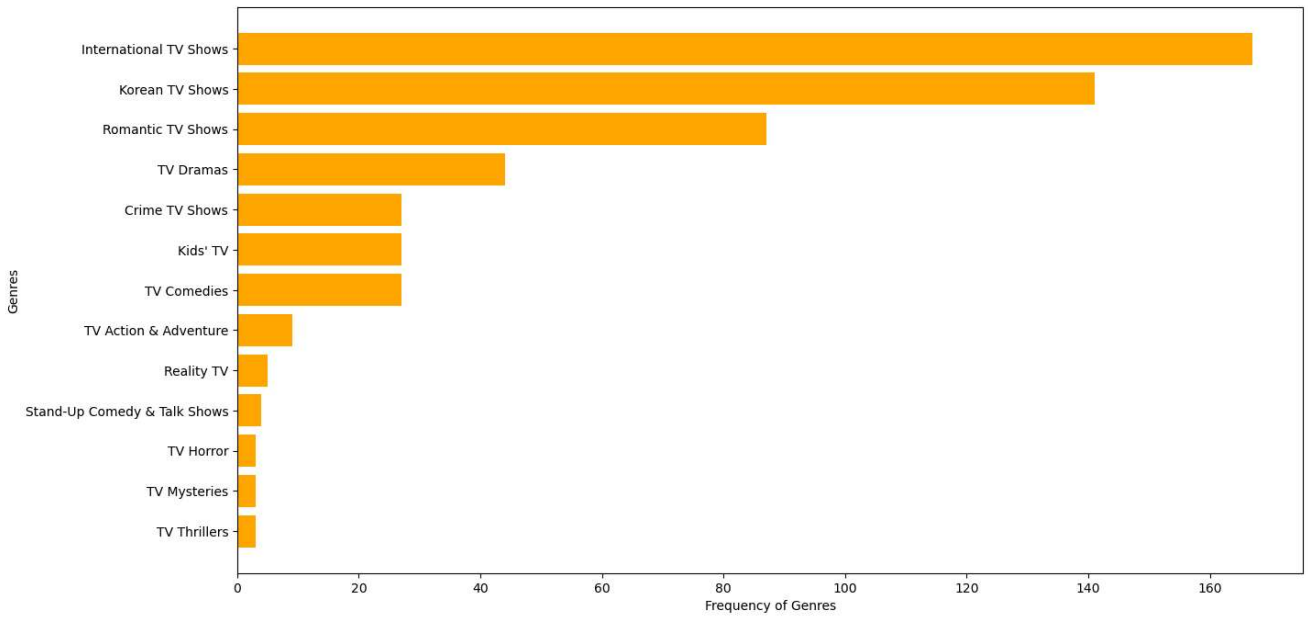


Reduction in TV Shows after 2019 in Japan

Univariate Analysis separately for shows in South Korea

In [222]:
```
#Analyzing India for both shows and movies
df_sk_shows=df_final1[df_final1['country']=='South Korea'][df_final1[df_final1['country']=='South Korea']['type']=='TV Show']
```

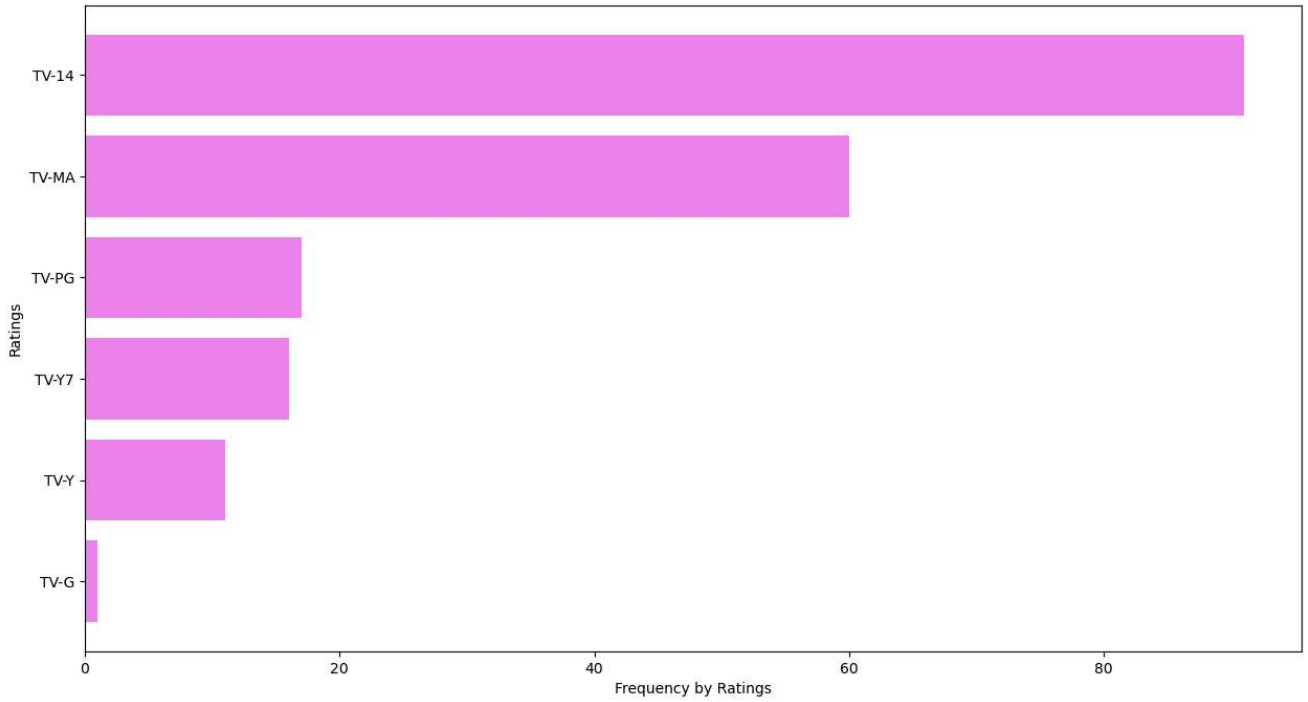
In [223]:
```
df_genre=df_sk_shows.groupby(['Genre']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
plt.figure(figsize=(15,8))
plt.barh(df_genre[::-1]['Genre'], df_genre[::-1]['title'],color=['orange'])
plt.xlabel('Frequency of Genres')
plt.ylabel('Genres')
plt.show()
```



International TV Shows, Korean TV shows, Romantic TV Shows,Drama,Crime and Comedy Genres are popular in TV Shows in S.Korea.

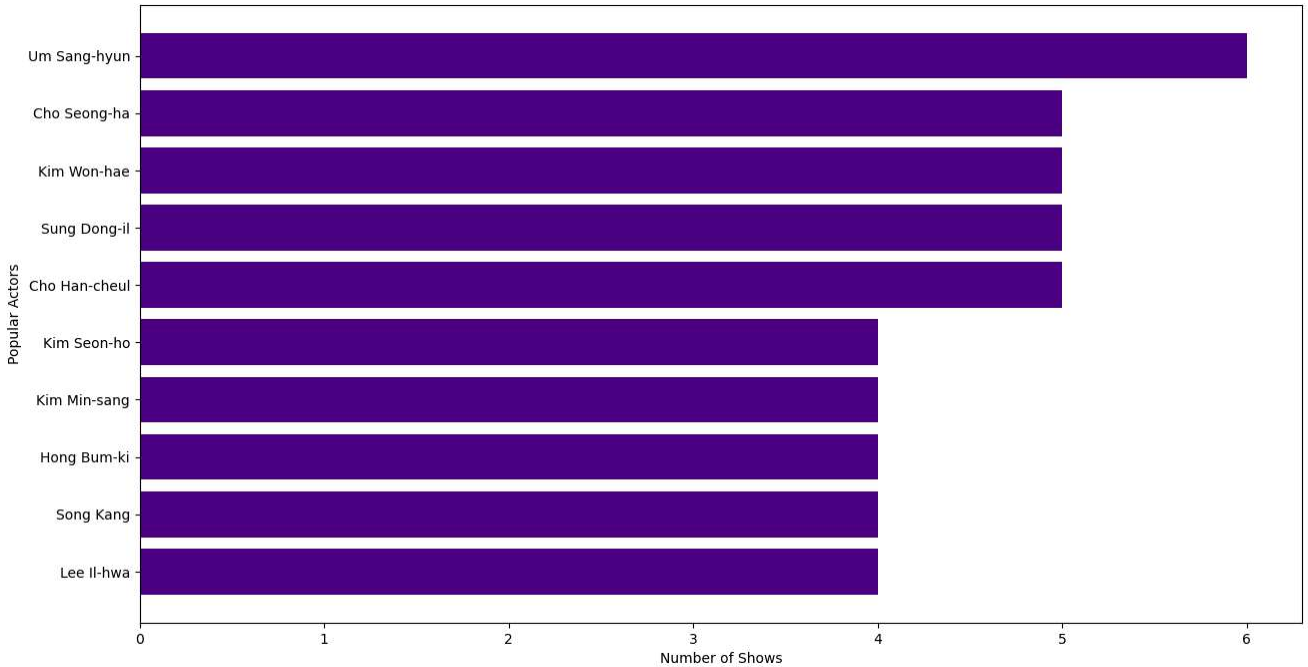
Only S.Korea has Romance as a top 3 favorable genre which depicts an inclination of their audience

In [225]:
```
df_rating=df_sk_shows.groupby(['rating']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:15]
plt.figure(figsize=(15,8))
plt.barh(df_rating[::-1]['rating'], df_rating[::-1]['title'],color=['violet'])
plt.xlabel('Frequency by Ratings')
plt.ylabel('Ratings')
plt.show()
```



So it seems plaussible to conclude that the popular ratings across Netflix includes TV-14 and Mature Audiences in TV Shows

In [227]:
```
df_actors=df_sk_shows.groupby(['Actors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:10]
df_actors=df_actors[df_actors['Actors']!='Unknown Actor']
plt.figure(figsize=(15,8))
plt.barh(df_actors[::-1]['Actors'], df_actors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Shows')
plt.ylabel('Popular Actors')
plt.show()
```



In [228]:
```
df_actors['Actors'].values
```

Out[228]:
```
array(['Um Sang-hyun', 'Cho Seong-ha', 'Kim Won-hae', 'Sung Dong-il',
       'Cho Han-cheul', 'Kim Seon-ho', 'Kim Min-sang', 'Hong Bum-ki',
       'Song Kang', 'Lee Il-hwa'], dtype=object)
```

Popular Actors in TV Shows in South Korea are:-

'Um Sang-hyun',

'Cho Seong-ha',

'Kim Won-hae',

'Sung Dong-il',

'Cho Han-cheul',

'Kim Seon-ho',

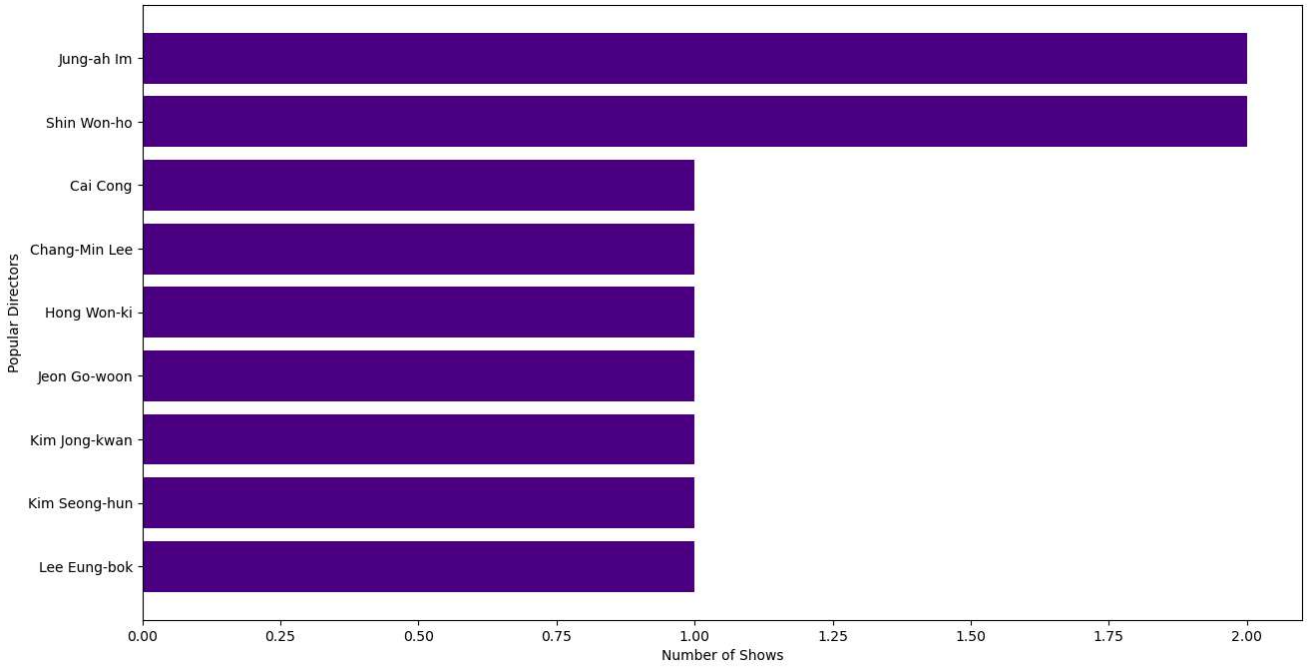
'Kim Min-sang',

'Hong Bum-ki',

'Song Kang',

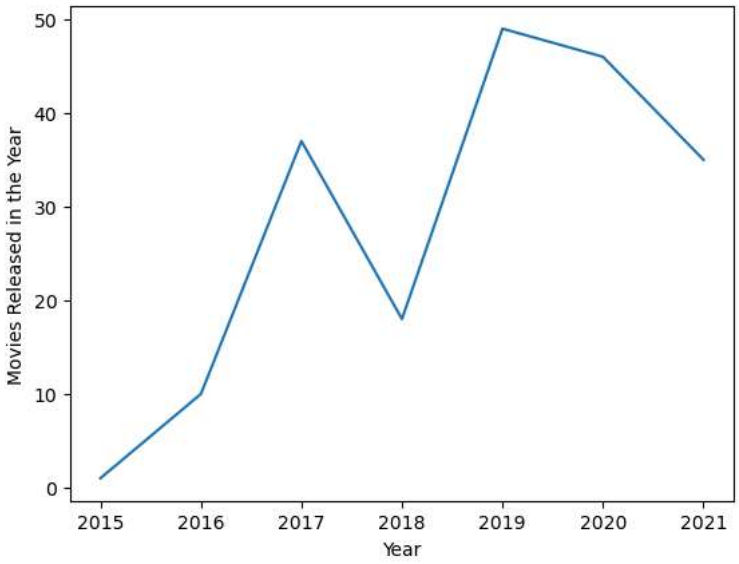
'Lee Il-hwa'

In [229]:
```
df_directors=df_sk_shows.groupby(['Directors']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[
df_directors=df_directors[df_directors['Directors']!='Unknown Director']
plt.figure(figsize=(15,8))
plt.barh(df_directors[::-1]['Directors'], df_directors[::-1]['title'],color=['indigo'])
plt.xlabel('Number of Shows')
plt.ylabel('Popular Directors')
plt.show()
```



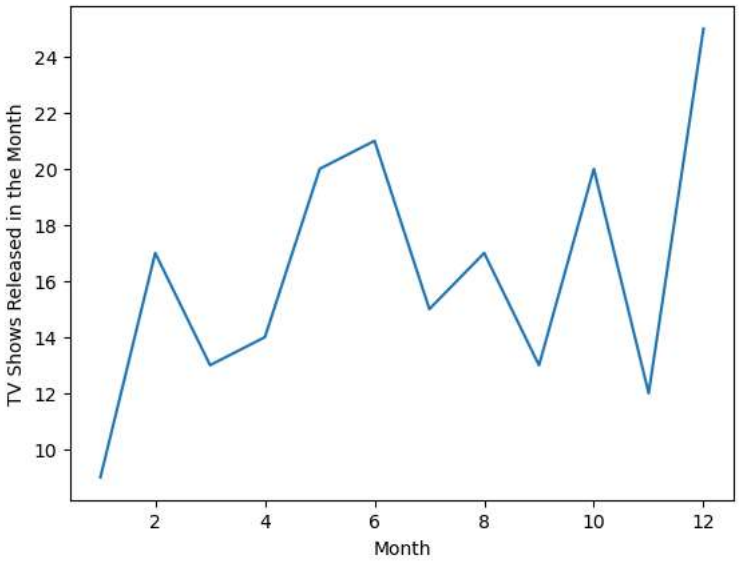
Two directors have directed 2 shows and rest all Directors are one time directors only

In [230]:
```
df_year=df_sk_shows.groupby(['year']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_year, x='year', y='title')
plt.ylabel("Movies Released in the Year")
plt.xlabel("Year")
plt.show()
```



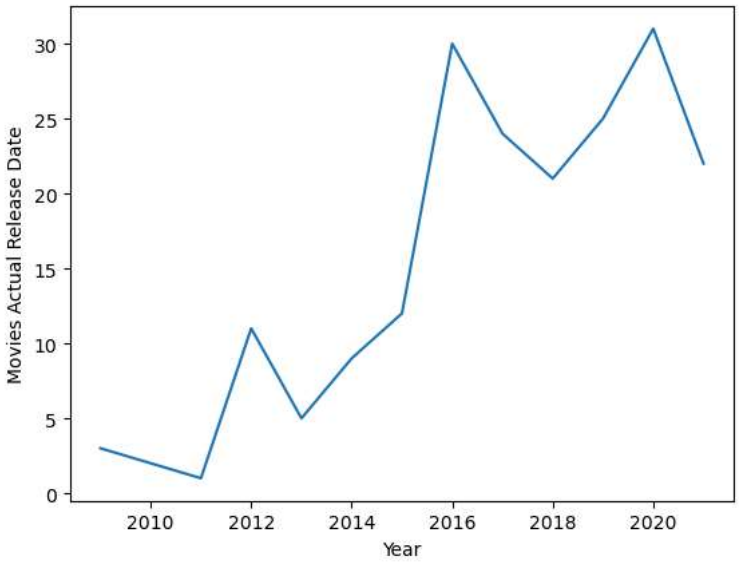
In South Korea, number of TV Shows reduced in 2018 from 2017, then increased till 2019 but have been on a heavy downfall since then

In [232]:
```
df_month=df_sk_shows.groupby(['month_added']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_month, x='month_added', y='title')
plt.ylabel("TV Shows Released in the Month")
plt.xlabel("Month")
plt.show()
```



TV Shows are added in Netflix by significant numbers in May and January in South Korea

In [233]:
```
df_release_year=df_sk_shows[df_sk_shows['release_year']>=1980].groupby(['release_year']).agg({"title":"nunique"}).reset_index()
sns.lineplot(data=df_release_year, x='release_year', y='title')
plt.ylabel("Movies Actual Release Date")
plt.xlabel("Year")
plt.show()
```



The number of TV Shows in S.Korea reached peak in 2016. It then reached a second peak in 2019. It has reduced in 2021 from 2020.

Recommendations

1) The most popular Genres across the countries and in both TV Shows and Movies are Drama, Comedy and International TV Shows/Movies, so content aligning to that is recommended.

2)Add TV Shows in July/August and Movies in last week of the year/first month of the next year.

3)For USA audience 80-120 mins is the recommended length for movies and Kids TV Shows are also popular along with the genres in first point, hence recommended.

4)For UK audience, recommended length for movies is same as that of USA (80-120 mins)

5)The target audience in USA and India is recommended to be 14+ and above ratings while for UK, its recommended to be completely Mature/R content .

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

7)Anime Genre for Japan and Romantic Genre in TV Shows for South Korean audiences is recommended.

8) 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.