In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
data=pd.read_csv('amazon_prime_titles.csv')
data
```

Out[2]:

	show_id	type	title	director	cast	country	date_added	release_year	ra
0	s1	Movie	The Grand Seduction	Don McKellar	Brendan Gleeson, Taylor Kitsch, Gordon Pinsent	Canada	March 30, 2021	2014	I
1	s2	Movie	Take Care Good Night	Girish Joshi	Mahesh Manjrekar, Abhay Mahajan, Sachin Khedekar	India	March 30, 2021	2018	
2	s3	Movie	Secrets of Deception	Josh Webber	Tom Sizemore, Lorenzo Lamas, Robert LaSardo, R...	United States	March 30, 2021	2017	I
3	s4	Movie	Pink: Staying True	Sonia Anderson	Interviews with: Pink, Adele, Beyoncé, Britney...	United States	March 30, 2021	2014	I
4	s5	Movie	Monster Maker	Giles Foster	Harry Dean Stanton, Kieran O'Brien, George Cos...	United Kingdom	March 30, 2021	1989	I
...	...	...	...	...	...	...	...	...	
9663	s9664	Movie	Pride Of The Bowery	Joseph H. Lewis	Leo Gorcey, Bobby Jordan	NaN	NaN	1940	
9664	s9665	TV Show	Planet Patrol	NaN	DICK VOSBURGH, RONNIE STEVENS, LIBBY MORRIS, M...	NaN	NaN	2018	
9665	s9666	Movie	Outpost	Steve Barker	Ray Stevenson, Julian Wadham, Richard Brake, M...	NaN	NaN	2008	
9666	s9667	TV Show	Maradona: Blessed Dream	NaN	Esteban Recagno, Ezequiel Stremiz, Luciano Vit...	NaN	NaN	2021	
9667	s9668	Movie	Harry Brown	Daniel Barber	Michael Caine, Emily Mortimer, Joseph Gilgun, ...	NaN	NaN	2010	

9668 rows × 12 columns

In [3]:
```
data.columns
```

Out[3]:
```
Index(['show_id', 'type', 'title', 'director', 'cast', 'country', 'date_adde
d',
       'release_year', 'rating', 'duration', 'listed_in', 'description'],
      dtype='object')
```

In [4]:
```
data.describe().round(2)
```

Out[4]:

	release_year
count	9668.00
mean	2008.34
std	18.92
min	1920.00
25%	2007.00
50%	2016.00
75%	2019.00
max	2021.00

In [5]:
```
data.duplicated().sum()
```

Out[5]:
```
0
```

In [6]:
```
numeric_columns = data.select_dtypes(include=['number']).columns

# Fill NaN values with mean for numeric columns
data[numeric_columns] = data[numeric_columns].fillna(data[numeric_columns].mea
```

In [7]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9668 entries, 0 to 9667
Data columns (total 12 columns):
 #   Column        Non-Null Count  Dtype 
---  ------        --------------  ----- 
 0   show_id       9668 non-null   object
 1   type          9668 non-null   object
 2   title         9668 non-null   object
 3   director      7585 non-null   object
 4   cast          8435 non-null   object
 5   country       672 non-null    object
 6   date_added    155 non-null    object
 7   release_year  9668 non-null   int64 
 8   rating        9331 non-null   object
 9   duration      9668 non-null   object
 10  listed_in     9668 non-null   object
 11  description   9668 non-null   object
dtypes: int64(1), object(11)
memory usage: 906.5+ KB
```

In [8]:
```
data.isna().sum()
```

Out[8]:
```
show_id            0
type               0
title              0
director        2083
cast            1233
country         8996
date_added      9513
release_year       0
rating           337
duration           0
listed_in          0
description        0
dtype: int64
```

In [9]:
```
data['cast'].fillna("Unknown", inplace=True)
```

In [10]:
```
data['director'].fillna("Unknown", inplace=True)
```

In [11]:
```
data['date_added'].fillna("Unknown", inplace=True)
```

In [12]:
```
most_common_country = data['country'].mode()[0]
data['country'].fillna(most_common_country, inplace=True)
```

In [13]:
```
most_common_rating = data['rating'].mode()[0]
data['rating'].fillna(most_common_rating, inplace=True)
```

In [14]:
```
data_remove=['description']
data=data.drop(columns=data_remove)
```

In [15]:
```
#data_remove=['show_id']
#data=data.drop(columns=data_remove)
```

In [16]:
```
#data_remove=['cast']
#data=data.drop(columns=data_remove)
```

In [17]:
```
data.isna().sum()
```

Out[17]:
```
show_id         0
type            0
title           0
director        0
cast            0
country         0
date_added      0
release_year    0
rating          0
duration        0
listed_in       0
dtype: int64
```

In [18]:
```
data.sample(10)
```

Out[18]:

	show_id	type	title	director	cast	country	date_added	release_year
9186	s9187	Movie	Hackers	Iain Softley	Angelina Jolie, Jonny Lee Miller, Matthew Lill...	United States	Unknown	1995
2746	s2747	TV Show	Cultureshock	Unknown	Judd Apatow, Steve Bannos	United States	Unknown	2018
5538	s5539	Movie	Infected	Dan Rickard	Samantha Bolter, Chris Wandell	United States	Unknown	2021
9389	s9390	Movie	Cracking Up	Chuck Staley, Rowby Goren	Michael Mislove, Fred Willard, Harry Shearer, ...	United States	Unknown	1977
5001	s5002	Movie	Forbidden Secrets	Richard Roy	Kristy Swanson, David Kelley, Christopher Bond...	United States	Unknown	2005
6097	s6098	Movie	Charade	Stanley Donen	Cary Grant, Audrey Hepburn, Walter Matthau, Ja...	United States	Unknown	1963
2639	s2640	Movie	DocoBanksy	Dominic Wade	Kate Brindley, Robbie Conal, Simon Hattenstone...	United Kingdom, United States	Unknown	2014
4480	s4481	Movie	Pataakha	Vishal Bhardwaj	Sanya Malhotra, Radhika Madan, Sunil Grover, V...	India	Unknown	2018
8684	s8685	Movie	Little White Lies	Philip Saville	Tara Fitzgerald, Cherie Lunghi, Gerard Butler	United States	Unknown	1998
3240	s3241	TV Show	Annedroids	Unknown	Addison Holley, Jadiel Dowlin, Adrianna Di Liello	United States	Unknown	2017

In [19]:
```
data['rating'].unique()
```

Out[19]:
```
array(['13+', 'ALL', '18+', 'R', 'TV-Y', 'TV-Y7', 'NR', '16+', 'TV-PG',
       '7+', 'TV-14', 'TV-NR', 'TV-G', 'PG-13', 'TV-MA', 'G', 'PG',
       'NC-17', 'UNRATED', '16', 'AGES_16_', 'AGES_18_', 'ALL_AGES',
       'NOT_RATE'], dtype=object)
```

In [20]:
```
rating_counts = data['rating'].value_counts()
rating_counts
```

Out[20]:
```
rating
13+         2454
16+         1547
ALL         1268
18+         1243
R           1010
PG-13        393
7+           385
PG           253
NR           223
TV-14        208
TV-PG        169
TV-NR        105
G             93
TV-G          81
TV-MA         77
TV-Y          74
TV-Y7         39
UNRATED       33
NC-17          3
AGES_18_       3
NOT_RATE       3
AGES_16_       2
16             1
ALL_AGES       1
Name: count, dtype: int64
```

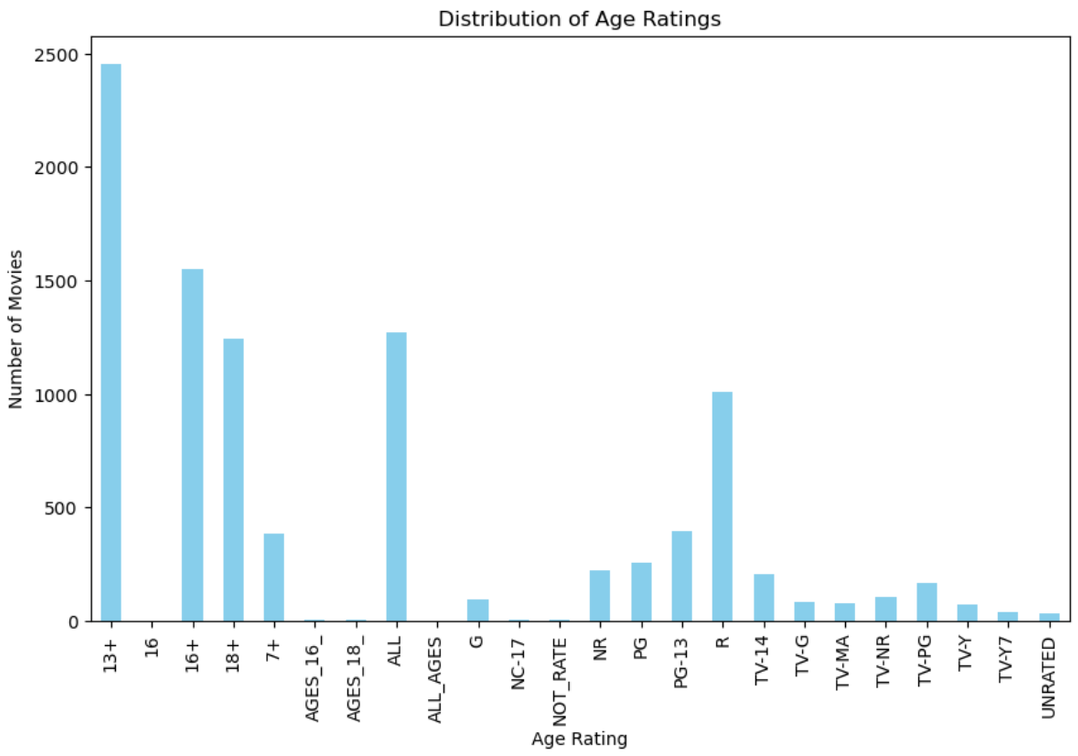
In [21]:
```
# Exclude rows where the director is "Unknown"
filtered_directors = data[data['director'] != 'Unknown']
top_directors = filtered_directors['director'].value_counts().head(10)
print("Top 10 Prolific Directors:")
print(top_directors)

# Exclude rows where the cast is "Unknown"
filtered_actors = data[data['cast'] != 'Unknown']
top_actors = filtered_actors['cast'].value_counts().head(10)
print("\nTop 10 Prolific Actors:")
print(top_actors)
```

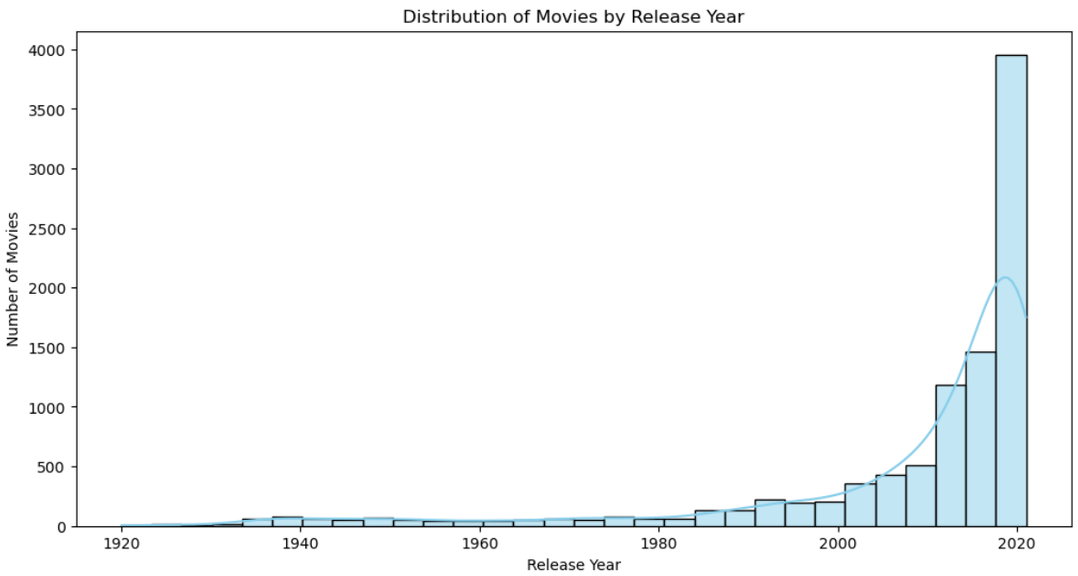
```
Top 10 Prolific Directors:
director
Mark Knight              113
Cannis Holder             61
Moonbug Entertainment     37
Jay Chapman               34
Arthur van Merwijk        30
Manny Rodriguez           22
John English              20
1                         16
Brian Volk-Weiss          15
Baeble Music              14
Name: count, dtype: int64

Top 10 Prolific Actors:
cast
Maggie Binkley                     56
1                                  34
Anne-Marie Newland                 24
Cassandra Peterson                 21
Grace Tamayo, Erin Webbs           17
Gene Autry, Champion, Gail Davis   12
Stevin John                        11
Gallagher                           9
LB, Aaron Michael                   9
Eddie Izzard                        9
Name: count, dtype: int64
```

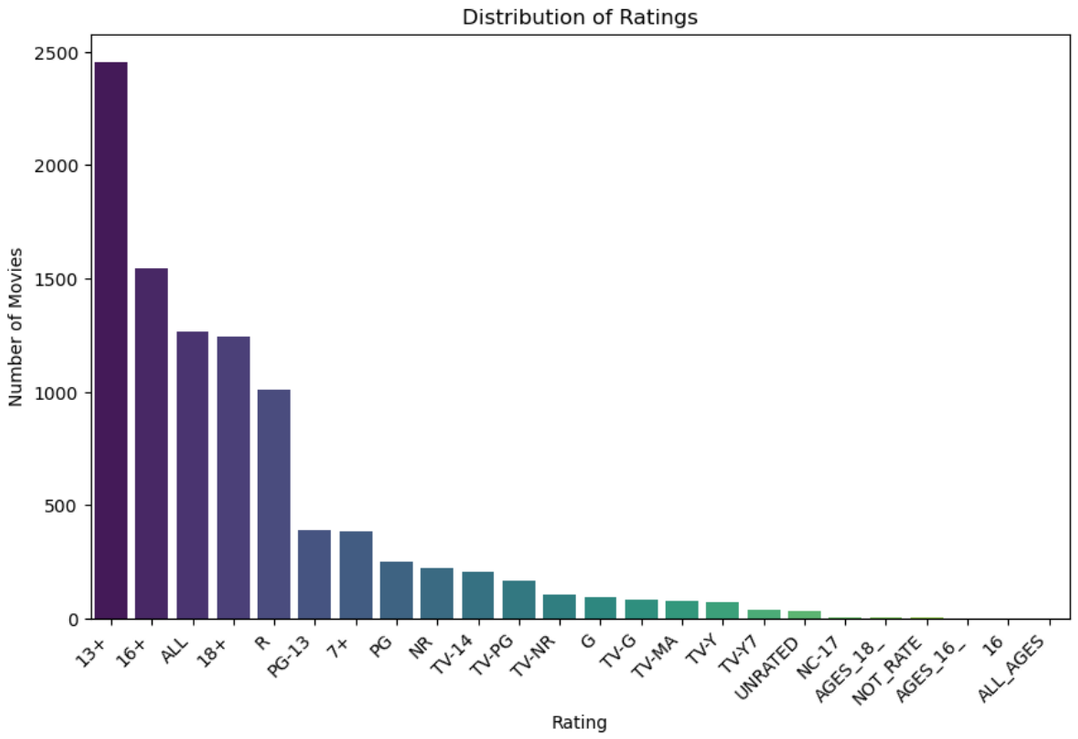
In [22]:
```
plt.figure(figsize=(10, 6))
rating_counts.sort_index().plot(kind='bar', color='skyblue')
plt.title('Distribution of Age Ratings')
plt.xlabel('Age Rating')
plt.ylabel('Number of Movies')
plt.show()
```



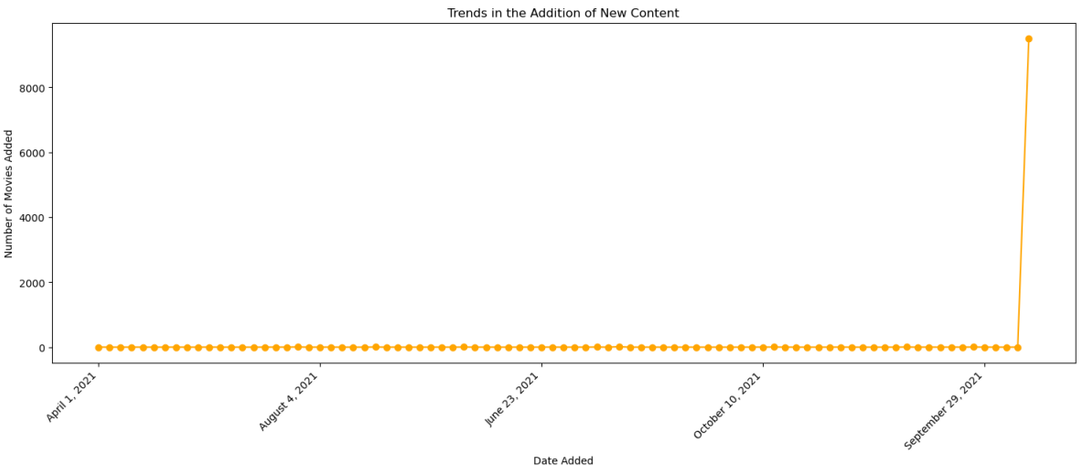
In [23]:
```
# Distribution of movies by release year
plt.figure(figsize=(12, 6))
sns.histplot(data['release_year'], bins=30, kde=True, color='skyblue')
plt.title('Distribution of Movies by Release Year')
plt.xlabel('Release Year')
plt.ylabel('Number of Movies')
plt.show()
```



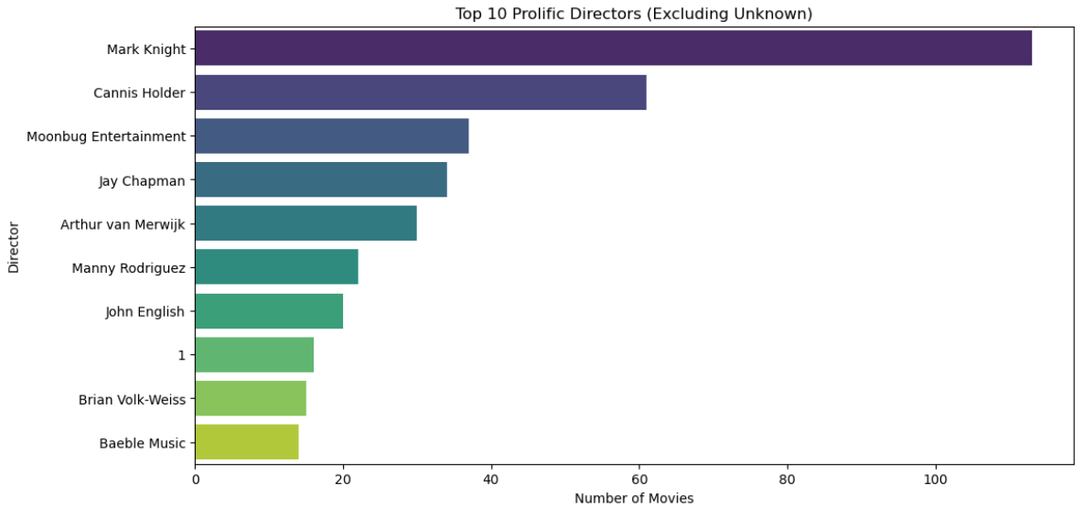
In [24]:
```
# Distribution of ratings
plt.figure(figsize=(10, 6))
sns.countplot(x='rating', data=data, order=data['rating'].value_counts().index
plt.title('Distribution of Ratings')
plt.xlabel('Rating')
plt.ylabel('Number of Movies')
plt.xticks(rotation=45, ha='right')
plt.show()
```



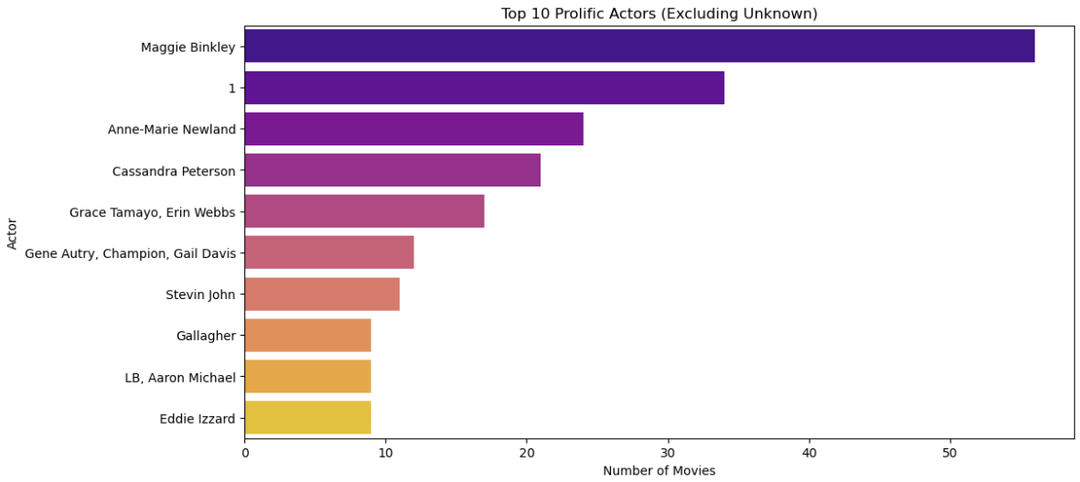
In [25]:
```
# Trends or patterns in the addition of new content
date_added_trends = data.groupby('date_added')['show_id'].count()
plt.figure(figsize=(18, 6))
date_added_trends.plot(marker='o', linestyle='-', color='orange')
plt.title('Trends in the Addition of New Content')
plt.xlabel('Date Added')
plt.ylabel('Number of Movies Added')
plt.xticks(rotation=45, ha='right')
plt.show()
```



In [26]:
```
plt.figure(figsize=(12, 6))
sns.barplot(x=top_directors.values, y=top_directors.index, palette='viridis')
plt.title('Top 10 Prolific Directors (Excluding Unknown)')
plt.xlabel('Number of Movies')
plt.ylabel('Director')
plt.show()
```

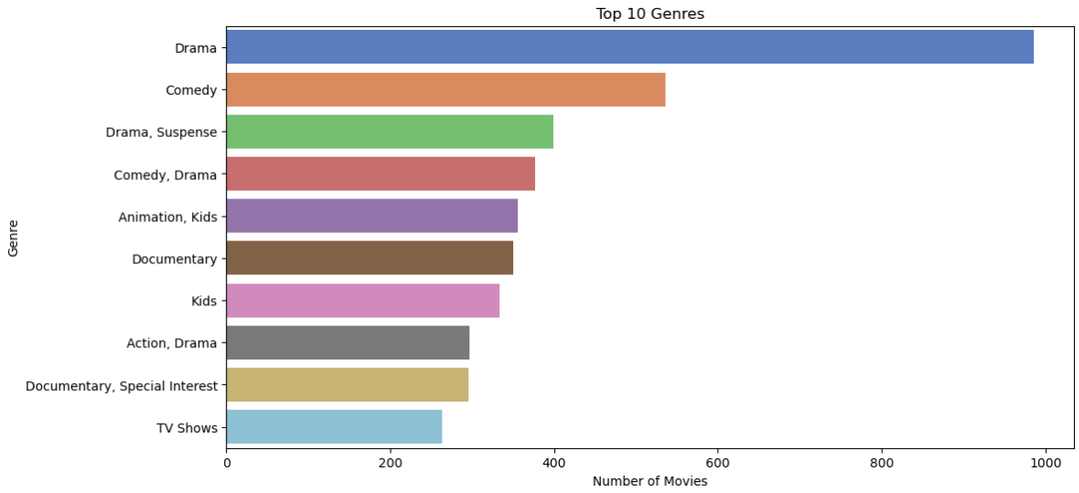


In [27]:
```
plt.figure(figsize=(12, 6))
sns.barplot(x=top_actors.values, y=top_actors.index, palette='plasma')
plt.title('Top 10 Prolific Actors (Excluding Unknown)')
plt.xlabel('Number of Movies')
plt.ylabel('Actor')
plt.show()
```

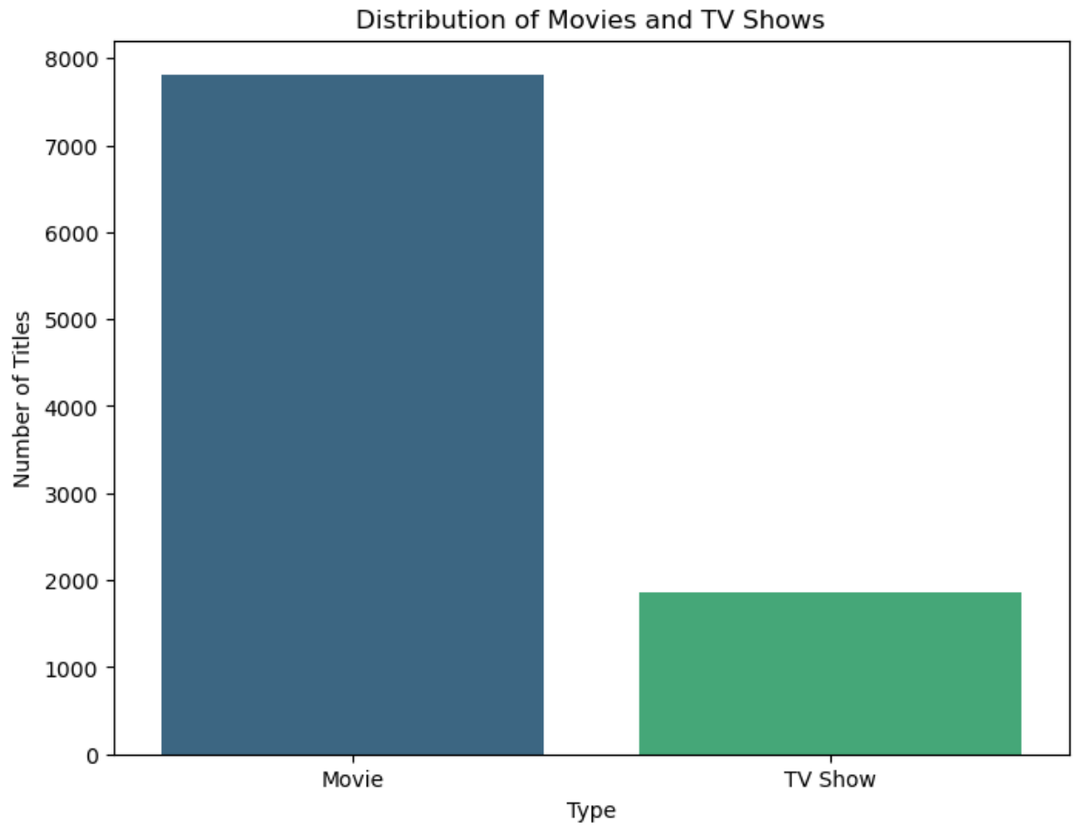


In [28]:
```
# Top genres
top_genres = data['listed_in'].value_counts().head(10)

# Plot Top Genres
plt.figure(figsize=(12, 6))
sns.barplot(x=top_genres.values, y=top_genres.index, palette='muted')
plt.title('Top 10 Genres')
plt.xlabel('Number of Movies')
plt.ylabel('Genre')
plt.show()
```



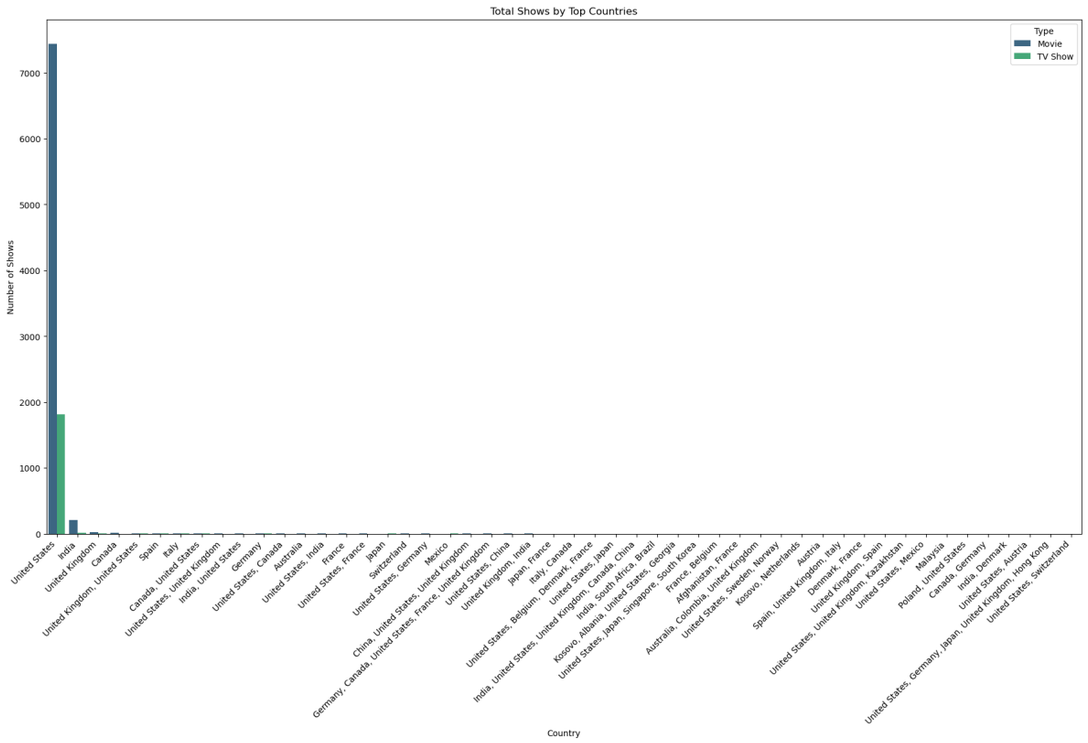
In [29]:
```
# Plotting the distribution of movies and TV shows
plt.figure(figsize=(8, 6))
sns.countplot(x='type', data=data, palette='viridis')
plt.title('Distribution of Movies and TV Shows')
plt.xlabel('Type')
plt.ylabel('Number of Titles')
plt.show()
```



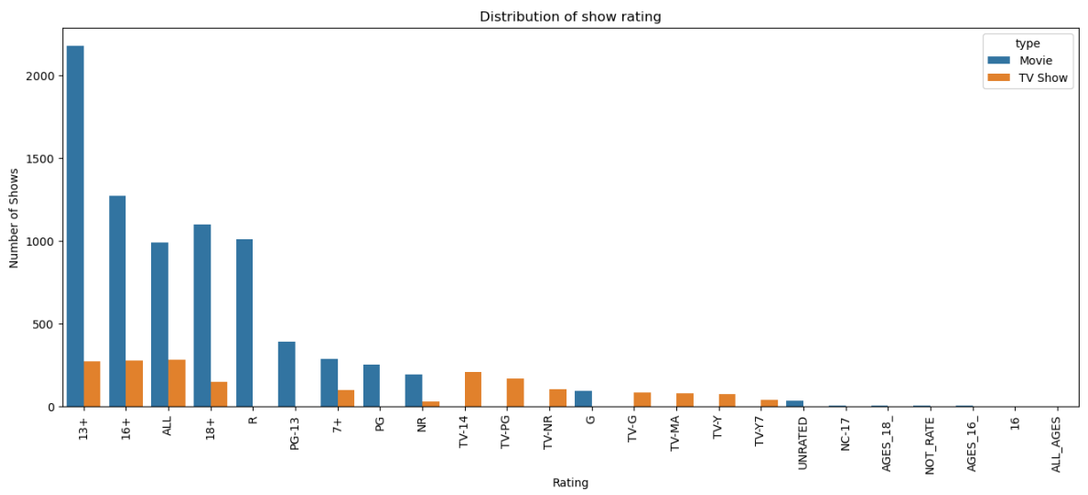
In [30]:
```
# Top N countries
top_countries = data['country'].value_counts().head(50).index

# Filter the data for the top countries
filtered_data = data[data['country'].isin(top_countries)]

# Visualize the total shows by country for the top countries
plt.figure(figsize=(22, 11))
sns.countplot(x='country', data=filtered_data, hue='type', order=top_countries
plt.title('Total Shows by Top Countries')
plt.xlabel('Country')
plt.ylabel('Number of Shows')
plt.legend(title='Type', loc='upper right')
plt.xticks(rotation=45, ha='right')
plt.show()
```



In [31]:
```
data_count1=data['rating'].value_counts().reset_index()
plt.figure(figsize=(16,6))
sns.countplot(x='rating',data=data,hue='type',order=data['rating'].value_count
plt.xticks(rotation=90)
plt.title('Distribution of show rating')
plt.xlabel('Rating')
plt.ylabel('Number of Shows')
plt.show()
```



In [ ]:
```
```