Import required libraries

In [2]:

```
import pandas as pd    #Data loading and analysis
import numpy as np     #Numarical operations
import matplotlib.pyplot as plt     #Data visualizations
import seaborn as sns          #advanced and attractive plots
import warnings

sns.set(style="whitegrid")       #Set visualizations style
warnings.filterwarnings("ignore")    #Ignore warnings
```

In [3]:

```
df = pd.read_csv('netflix1.csv', lineterminator = '\n')  #Load Dataset
```

In [5]:

```
df.head()
```

Out[5]:

	show_id	type	title	director	country	date_added	release_year	rating	duration	lis
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	United States	9/25/2021	2020	PG-13	90 min	Docume
1	s3	TV Show	Ganglands	Julien Leclercq	France	9/24/2021	2021	TV-MA	1 Season	C Internati Shows,
2	s6	TV Show	Midnight Mass	Mike Flanagan	United States	9/24/2021	2021	TV-MA	1 Season	TV Drai Ho My
3	s14	Movie	Confessions of an Invisible Girl	Bruno Garotti	Brazil	9/22/2021	2021	TV-PG	91 min	Ch Family Cor
4	s8	Movie	Sankofa	Haile Gerima	United States	9/24/2021	1993	TV-MA	125 min	I Inde Inter

In [6]:

```
df.info()    #Check dataset info
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8790 entries, 0 to 8789
Data columns (total 10 columns):
 #   Column        Non-Null Count  Dtype 
---  ------        --------------  ----- 
 0   show_id       8790 non-null   object
 1   type          8790 non-null   object
 2   title         8790 non-null   object
 3   director      8790 non-null   object
 4   country       8790 non-null   object
 5   date_added    8790 non-null   object
 6   release_year  8790 non-null   int64
 7   rating        8790 non-null   object
 8   duration      8790 non-null   object
    8790 non-null   object
dtypes: int64(1), object(9)
memory usage: 686.8+ KB
```

Check NULL values in columns

df.isnull().sum()

check total NULL values in dataset

In [7]:

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

Out[7]:

```
np.int64(0)
```

Check duplicate values

In [8]:

```
df.duplicated().sum()
```

Out[8]:

```
np.int64(0)
```

Rename columns

In [9]:

```
df.rename(columns={'show_id': 'ShowId','type': 'Type','director': 'Directore','country':
```

In [10]:

```
df.rename(columns={'title':'Title'}, inplace= True)
```

In [12]:

```
df.head()
```

Out[12]:

	ShowId	Type	Title	Directore	Country	DateAdded	release_year	Rating	Duration	
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	United States	9/25/2021	2020	PG-13	90 min	Docume
1	s3	TV Show	Ganglands	Julien Leclercq	France	9/24/2021	2021	TV-MA	1 Season	C Internat Shows, '
2	s6	TV Show	Midnight Mass	Mike Flanagan	United States	9/24/2021	2021	TV-MA	1 Season	TV Dra Hc My
3	s14	Movie	Confessions of an	Bruno Garotti	Brazil	9/22/2021	2021	TV-PG	91 min	Ch Family

	ShowId	Type	Title	Directore	Country	DateAdded	release_year	Rating	Duration	Co
			Invisible Girl							Co
4	s8	Movie	Sankofa	Haile Gerima	United States	9/24/2021	1993	TV-MA	125 min	Inde Inte

In [13]:

```
df['Genre'].head()
```

Out[13]:

```
0                                       Documentaries\r
1     Crime TV Shows, International TV Shows, TV Act...
2                  TV Dramas, TV Horror, TV Mysteries\r
3                  Children & Family Movies, Comedies\r
4     Dramas, Independent Movies, International Movi...
Name: Genre, dtype: object
```

In [56]:

```
df.describe()    #check numarical columns
```

Out[56]:

	release_year
count	8790.000000
mean	2014.183163
std	8.825466
min	1925.000000
25%	2013.000000
50%	2017.000000
75%	2019.000000
max	2021.000000

convert column to Date Format

In [16]:

```
df['DateAdded'] = pd.to_datetime(df['DateAdded'])
```

In [17]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8790 entries, 0 to 8789
Data columns (total 10 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   ShowId        8790 non-null   object
 1   Type          8790 non-null   object
 2   Title         8790 non-null   object
 3   Directore     8790 non-null   object
 4   Country       8790 non-null   object
 5   DateAdded     8790 non-null   datetime64[ns]
 6   release_year  8790 non-null   int64
 7   Rating        8790 non-null   object
 8   Duration      8790 non-null   object
 9   Genre         8790 non-null   object
dtypes: datetime64[ns](1), int64(1), object(8)
memory usage: 686.8+ KB
```

In [18]:

```
df.shape
```

Out[18]:

```
(8790, 10)
```

Exploration summary

We have a dataframe consistin of 8790 rowa and 10 columns. Our dataset looks a No NULL and no duplicates values in all dataset. We did a converted column to Dateformat in our dataset. We have changed to all column names.

In [19]:

```
df['Title'].count() #total count of title
```

Out[19]:

```
np.int64(8790)
```

In [ ]:

```

```

How many movies vs TV Show are there

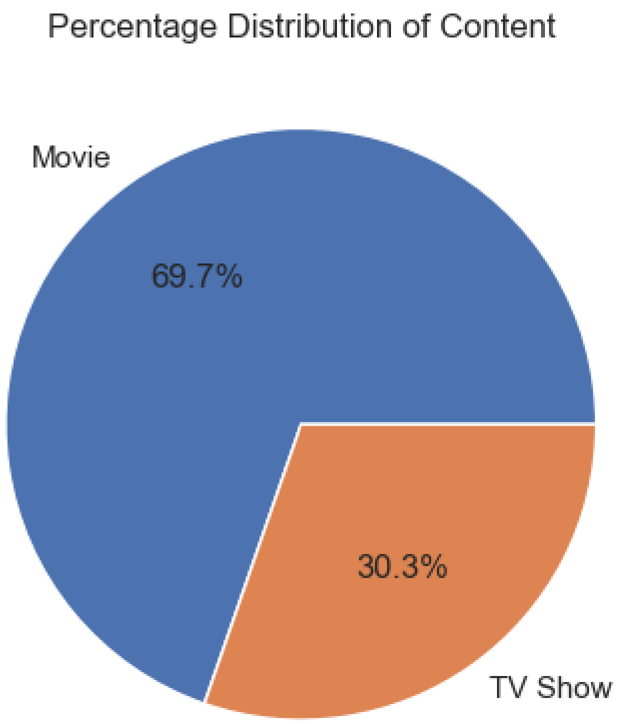
In [20]:

```
Movies_TV_count = df['Type'].value_counts()          #count movies vs TV show
print(Movies_TV_count)
```

```
Type
Movie      6126
TV Show    2664
Name: count, dtype: int64
```

In [21]:

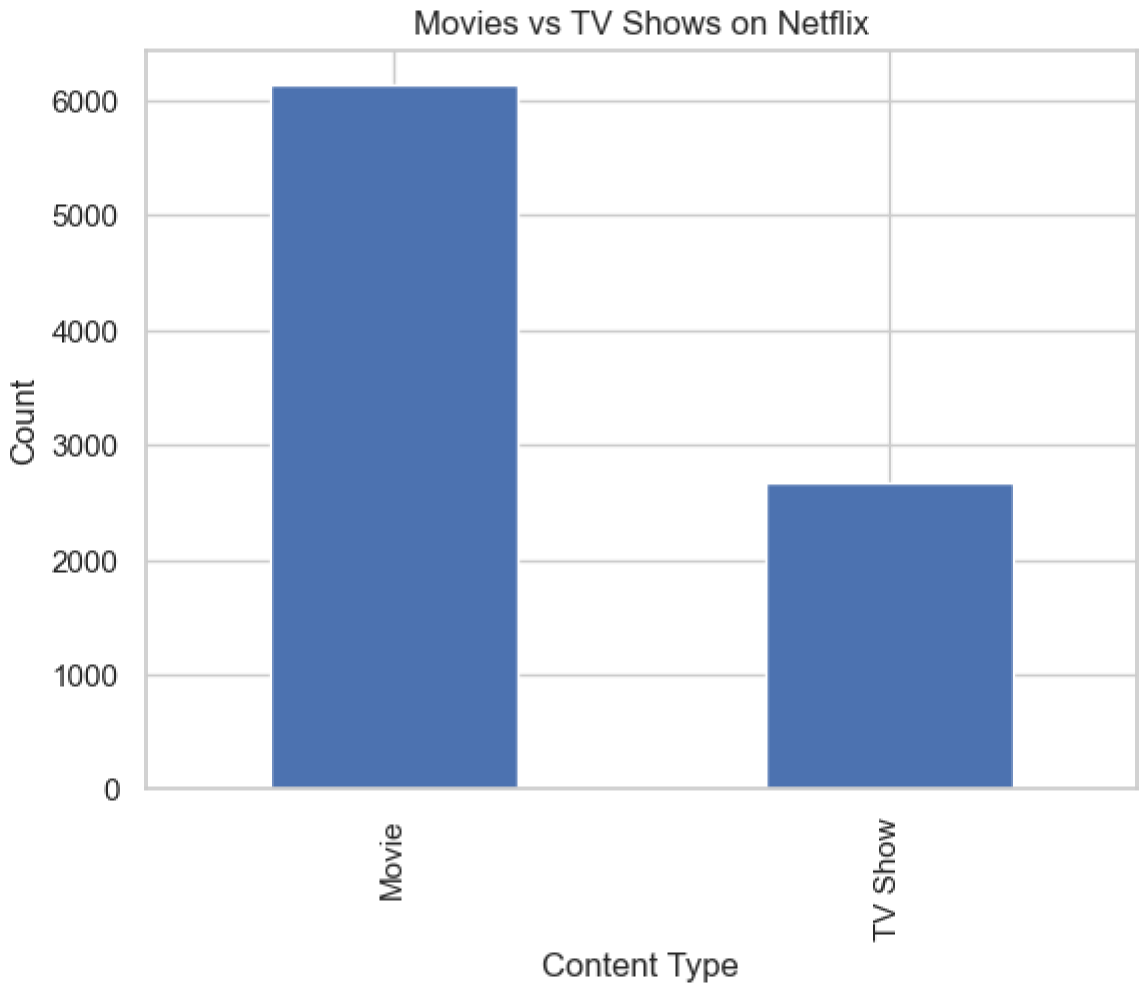
```
df['Type'].value_counts().plot(kind='pie', autopct='%1.1f%%')
plt.title("Percentage Distribution of Content")
plt.ylabel("")
plt.show()
```



How many Movies vs TV Shows are available on Netflix?

In [24]:

```
df['Type'].value_counts().plot(kind='bar')
plt.title("Movies vs TV Shows on Netflix")
plt.xlabel("Content Type")
plt.ylabel("Count")
plt.show()
```



Project Insight Statement

Movies make up approximately 70% of Netflix’s content, while TV Shows account for about 30%, indicating a stronger focus on movies.

```
In [ ]:
```

which year was the maximum content added

```
In [23]:
df['release_year'].value_counts().sort_index()
```

```
Out[23]:
release_year
1925       1
1942       2
1943       3
1944       3
1945       4
        ... 
2017    1030
2018    1146
2019    1030
2020     953
```

```
2021     592
Name: count, Length: 74, dtype: int64
```

Netflix added the maximum number of titles in 2018, indicating a peak in content expansion during that year.

In [ ]:

```

```

Top5 content producing countries in Netfilx

In [25]:

```python
# Remove rows where country is missing
df_country = df.dropna(subset=['Country'])

# Split multiple countries and count
top_countries = (
    df_country['Country']
    .str.split(', ')
    .explode()
    .value_counts()
    .head(5)
)

print(top_countries)
```

```
Country
United States     3240
India             1057
United Kingdom     638
Pakistan           421
Not Given          287
Name: count, dtype: int64
```

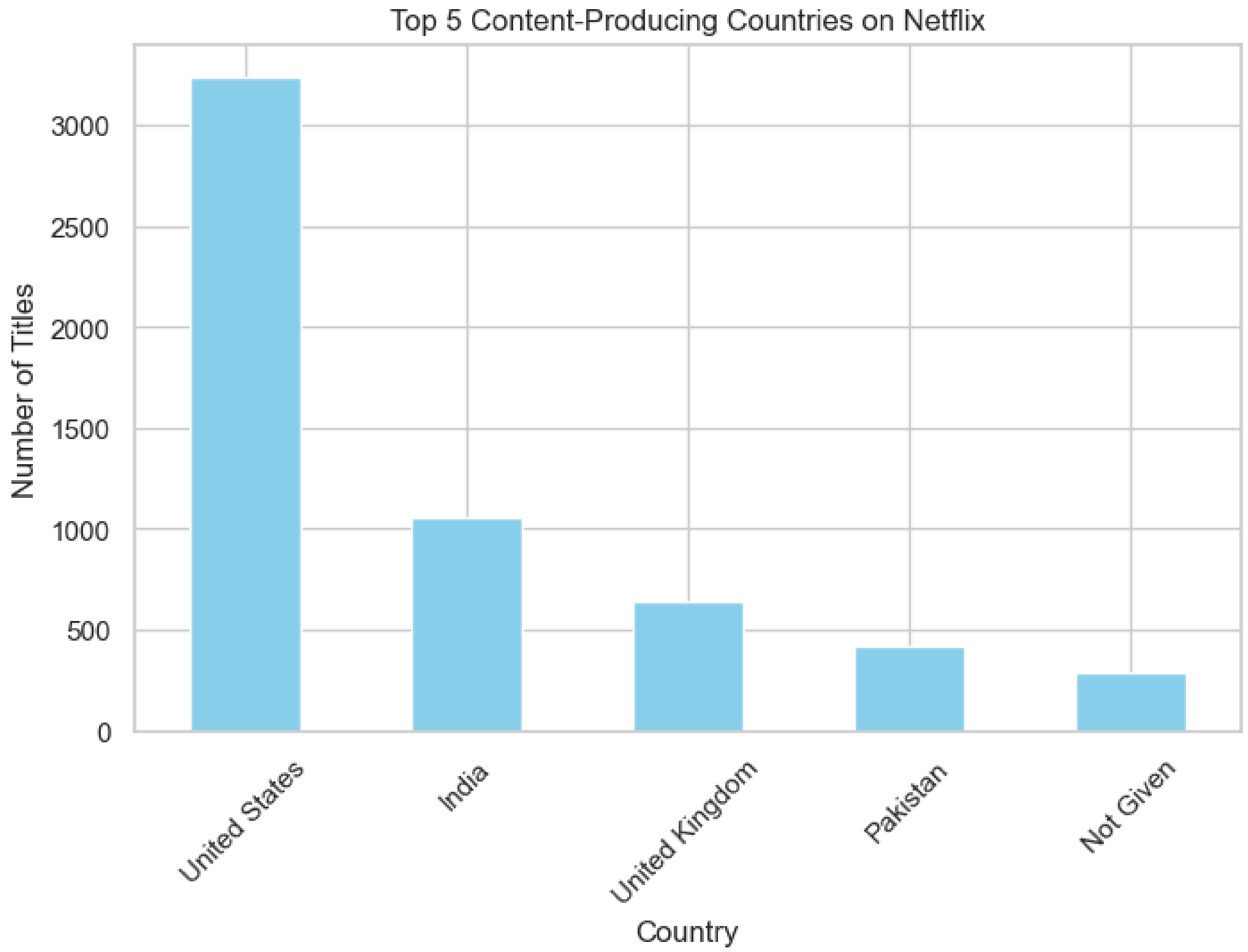
In [26]:

```python
# Fill missing country values
df['Country'] = df['Country'].fillna('Unknown')

# Count top 5 countries
Country_count = df['Country'].value_counts().head(5)
```

In [27]:

```python
plt.figure(figsize=(8,5))
Country_count.plot(kind='bar', color='skyblue')
plt.title('Top 5 Content-Producing Countries on Netflix')
plt.xlabel('Country')
plt.ylabel('Number of Titles')
plt.xticks(rotation=45)
plt.show()
```



Statment that United States is the most top content-producing in Netflix, followed by Inida and united kingdome reflecting netflix focus on north american and International content

```
In [ ]:
```

How has Netflix content grown year by year

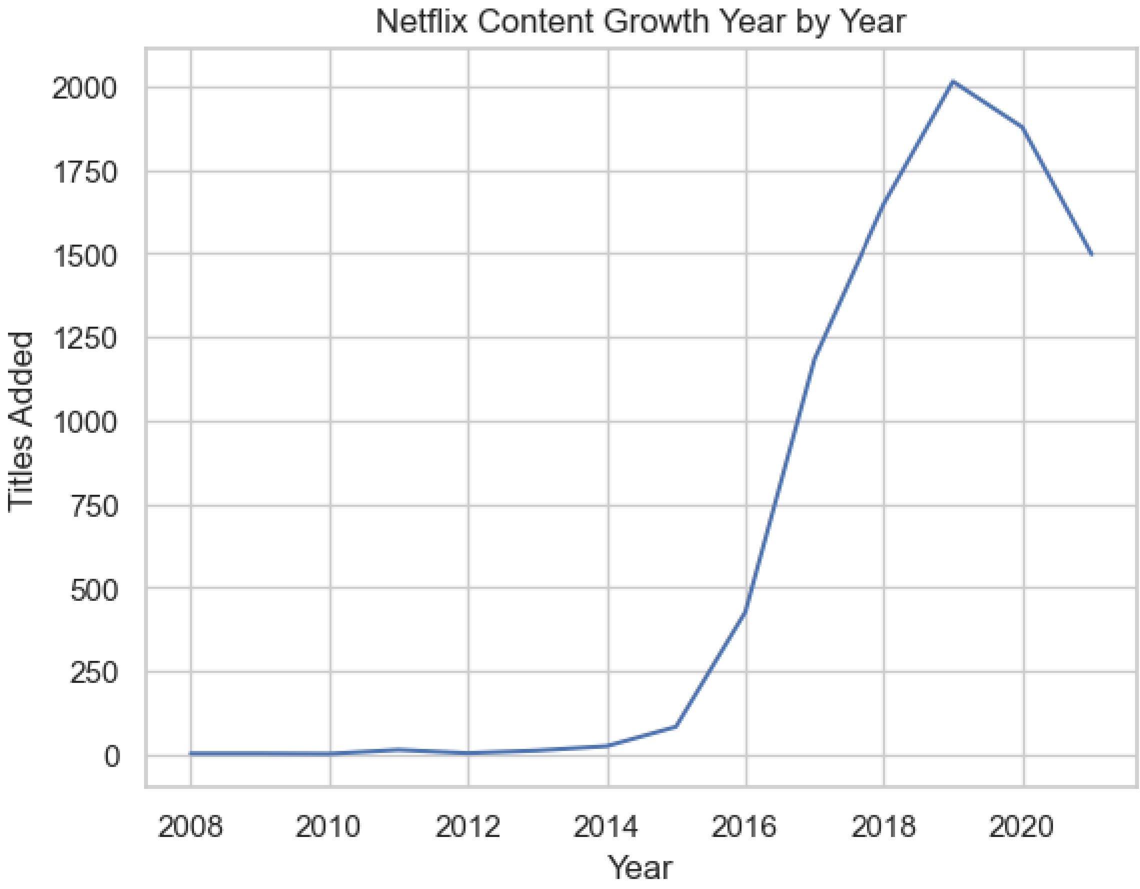
```
In [29]:
df['DateAdded'] = pd.to_datetime(df['DateAdded'], errors='coerce')
df['YearAdded'] = df['DateAdded'].dt.year
df['YearAdded'].value_counts().sort_index()
```

```
Out[29]:
YearAdded
2008       2
2009       2
2010       1
2011      13
2012       3
2013      11
2014      24
2015      82
2016     426
2017    1185
```

```
2018    1648
2019    2016
2020    1879
2021    1498
Name: count, dtype: int64
```

In [81]:

```
df['year_added'] = df['DateAdded'].dt.year
df['year_added'].value_counts().sort_index().plot(kind='line')
plt.title("Netflix Content Growth Year by Year")
plt.xlabel("Year")
plt.ylabel("Titles Added")
plt.show()
```



Rapid growth after 2016

In [ ]:

In [42]:

```
df['Genre'] = df['Genre'].astype(str)
```

In [ ]:

What are the most common genres are Netflix

In [30]:

```
df['Genre'].value_counts().head()
```

```
Out[30]:
Genre
Dramas, International Movies\r                        362
Documentaries\r                                       359
Stand-Up Comedy\r                                     334
Comedies, Dramas, International Movies\r              274
Dramas, Independent Movies, International Movies\r    252
Name: count, dtype: int64
```

Dramas,international movies and Documentaries type genres are top content in Netflix

What are the most common rating in Netflix

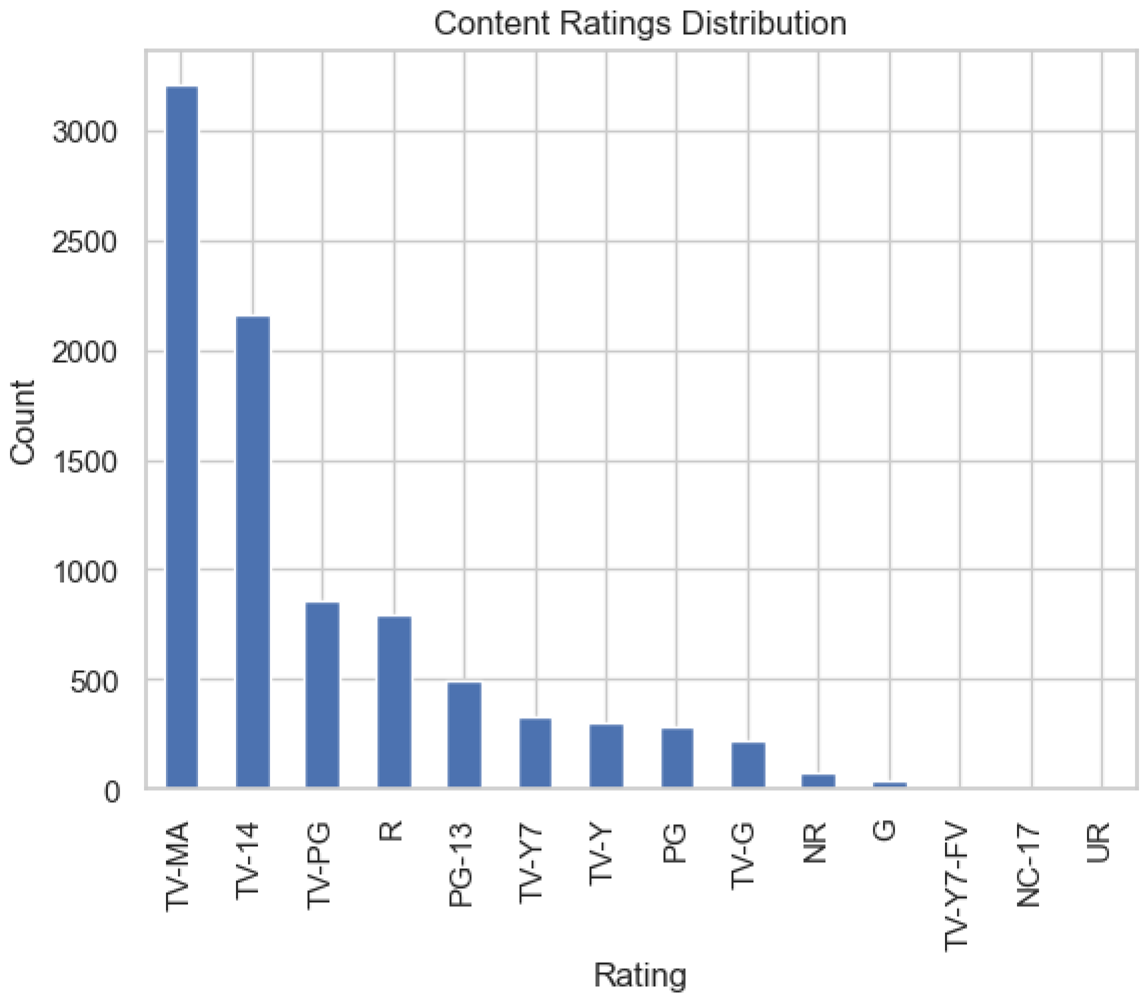
In [31]:

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

```
Out[31]:
Rating
TV-MA       3205
TV-14       2157
TV-PG        861
R            799
PG-13        490
TV-Y7        333
TV-Y         306
PG           287
TV-G         220
NR            79
G             41
TV-Y7-FV       6
NC-17          3
UR             3
Name: count, dtype: int64
```

In [33]:

```
df['Rating'].value_counts().plot(kind='bar')
plt.title("Content Ratings Distribution")
plt.xlabel("Rating")
plt.ylabel("Count")
plt.show()
```



highest rating on netflix is TV-MA and 2nd is the TV-14

```
In [ ]:
```

who are the top 5 directors with most content

```
In [34]:
df['Directore'].value_counts().head(5)
```

```
Out[34]:
Directore
Not Given                   2588
Rajiv Chilaka                 20
Alastair Fothergill           18
Raúl Campos, Jan Suter        18
Marcus Raboy                  16
Name: count, dtype: int64
```

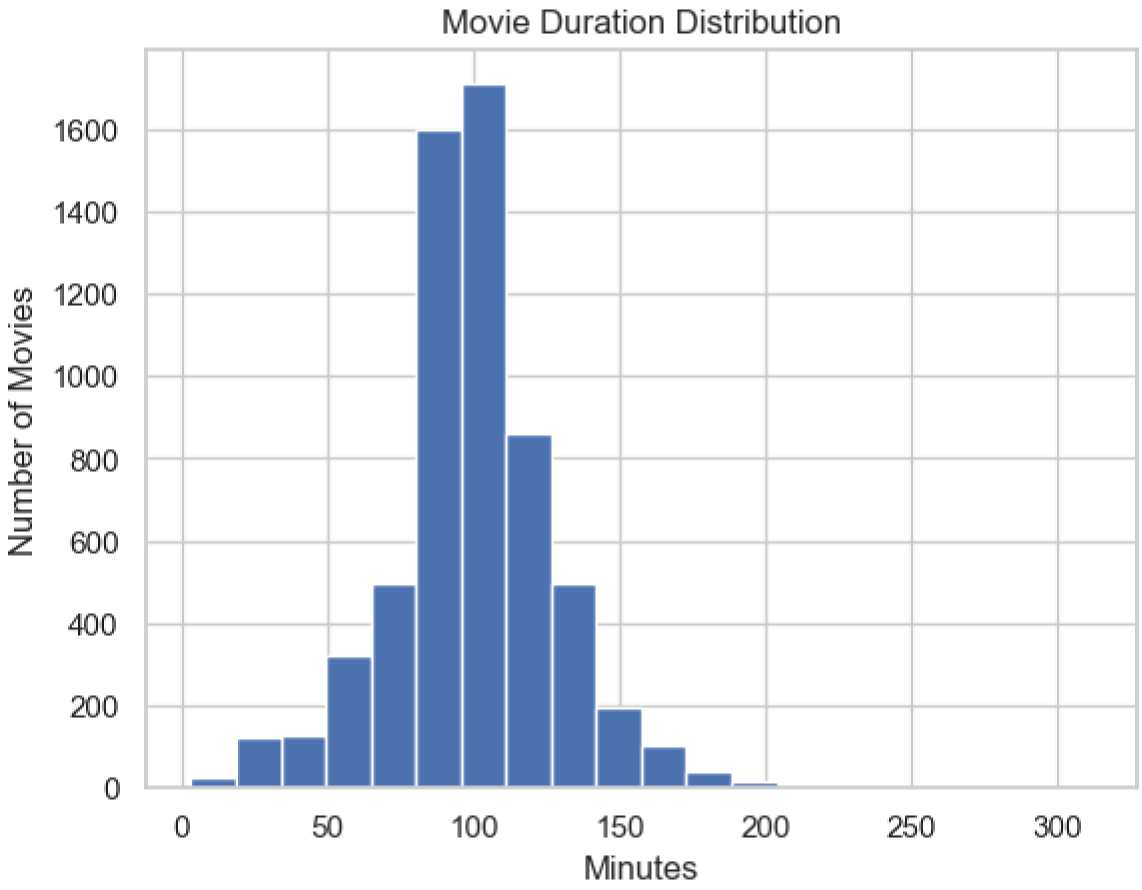
What is the distribution of movie durations?

In [35]:

```
movies = df[df['Type'] == 'Movie']
movies['Duration_min'] = movies['Duration'].str.replace(' min','')
movies['Duration_min'] = pd.to_numeric(movies['Duration_min'], errors='coerce')
```

In [36]:

```
plt.hist(movies['Duration_min'].dropna(), bins=20)
plt.title("Movie Duration Distribution")
plt.xlabel("Minutes")
plt.ylabel("Number of Movies")
plt.show()
```



Most are movies 80-110 minutes

In [ ]: