Netflix Dataset Project

"Analyze Netflix titles to gain insights into content trends, audience preferences, and market dynamics, and to inform data-driven decisions for content creation, acquisition, and marketing strategies."

In simpler terms, the project's main goal is to:

. Understand what types of content are popular on Netflix

. Identify trends and patterns in audience preferences

. Provide insights to inform content creation, acquisition, and marketing decisions By achieving this goal, the project aims to help Netflix improve its content strategy, enhance user experience, and drive business growth.

Analyzing Netflix titles can be an important project for several reasons:

1. Understanding Content Trends: Insights into popular genres, directors, and release years can help identify trends in content consumption.
2. Audience Preferences: Analyzing ratings and viewer engagement can reveal audience preferences and help content creators tailor their content.
3. Market Research: Understanding what types of content are popular can inform market research and strategic decisions for content creation and acquisition.
4. Personalization: Analyzing user behavior and preferences can help improve content recommendations and personalization.
5. Competitive Analysis: Comparing Netflix's content offerings with competitors can provide insights into market gaps and opportunities.

Some potential applications of this project include:

Content creation and acquisition strategies Targeted marketing and promotion User experience optimization Competitive analysis and market research

How to load the Netflix dataset?

```
import pandas as pd
import numpy as np
df = pd.read_csv("C:\\Users\\HP\\Downloads\\netflix_titles.csv")
df
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	D
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	T
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season	T
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	1 Season	
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons	

```
print(df['cast'].nunique())
```

```
7692
```

```
print(df['rating'].unique())
```

```
['PG-13' 'TV-MA' 'PG' 'TV-14' 'TV-PG' 'TV-Y' 'TV-Y7' 'R' 'TV-G' 'G'
 'NC-17' '74 min' '84 min' '66 min' 'NR' nan 'TV-Y7-FV' 'UR']
```

```
num_ratings = df['rating'].nunique()
```

```
Number of unique rating types: 17
```

```
# Optional: Show frequency of each rating type
print(df['rating'].value_counts())
```

```
rating
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: count, dtype: int64
```

How to check the number of rows and columns?

```
df.shape
```

```
(8807, 12)
```

Top view in Dataset

```
df.head()
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	lis
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	Docum
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban... Sami	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	Inter TV Sh Drai M

Top 15 view in Dataset

```
df.head(15)
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	1 Season
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons
5	s6	TV Show	Midnight Mass	Mike Flanagan	Kate Siegel, Zach Gilford, Hamish Linklater, H...	NaN	September 24, 2021	2021	TV-MA	1 Season
6	s7	Movie	My Little Pony: A New Generation	Robert Cullen, José Luis Ucha	Vanessa Hudgens, Kimiko Glenn, James Marsden, ...	NaN	September 24, 2021	2021	PG	91 min
7	s8	Movie	Sankofa	Haile Gerima	Kofi Ghanaba, Oyafunmike Ogunlano, Alexandra D...	United States, Ghana, Burkina Faso, United Kin...	September 24, 2021	1993	TV-MA	125 min
					Mel					

Similarly what if we want to see the last 20 rows ?

```
df.tail(20)
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	durati
8787	s8788	Movie	You Can’t Fight Christmas	Kenny Young	Brely Evans, Andra Fuller, Persia White, Porsc...	United States	December 19, 2017	2017	TV-PG	85 m
8788	s8789	Movie	You Carry Me	Ivona Juka	Lana Baric, Vojislav Brajovic, Natasa Janjic, ...	Croatia, Slovenia, Serbia, Montenegro	July 1, 2016	2015	TV-MA	157 m
8789	s8790	Movie	You Changed My Life	Cathy Garcia-Molina	John Lloyd Cruz, Sarah Geronimo, Rayver Cruz, ...	Philippines	February 27, 2019	2009	TV-PG	116 m
8790	s8791	Movie	You Don't Mess with the Zohan	Dennis Dugan	Adam Sandler, John Turturro, Emmanuelle Chriqu...	United States	September 1, 2019	2008	UR	113 m
8791	s8792	Movie	Young Adult	Jason Reitman	Charlize Theron, Patton Oswalt, Patrick Wilson...	United States	November 20, 2019	2011	R	94 m
8792	s8793	Movie	Young Tiger	Mu Chu	Qiu Yuen, Charlie Chin, Jackie Chan, Hu Chin, ...	Hong Kong	November 1, 2016	1973	NR	81 m
8793	s8794	Movie	Yours, Mine and Ours	Raja Gosnell	Dennis Quaid, Rene Russo, Sean Faris, Katija P...	United States	November 20, 2019	2005	PG	88 m
8794	s8795	Movie	اشتباك	Mohamed Diab	Nelly Karim, Hany Adel, Tarek Abdel Aziz, Ahme...	Egypt, France	October 11, 2018	2016	TV-14	98 m
8795	s8796	TV Show	Yu-Gi-Oh! Arc-V	NaN	Mike Liscio, Emily Bauer, Billy Bob Thompson, ...	Japan, Canada	May 1, 2018	2015	TV-Y7	Seaso
8796	s8797	TV Show	Yunus Emre	NaN	Gökhan Atalay, Payidar Tüfekçioglu, Baran Akbu...	Turkey	January 17, 2017	2016	TV-PG	Seaso
8797	s8798	TV Show	Zak Storm	NaN	Michael Johnston, Jessica Gee-George, Christin...	United States, France, South Korea, Indonesia	September 13, 2018	2016	TV-Y7	Seaso

How to get the column names and data types?

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8807 entries, 0 to 8806
Data columns (total 12 columns):
 #   Column        Non-Null Count  Dtype 
---  ------        --------------  ----- 
 0   show_id       8807 non-null   object
 1   type          8807 non-null   object
 2   title         8807 non-null   object
 3   director      6173 non-null   object
 4   cast          7982 non-null   object
 5   country       7976 non-null   object
 6   date_added    8797 non-null   object
 7   release_year  8807 non-null   int64 
 8   rating        8803 non-null   object
 9   duration      8804 non-null   object
 10  listed_in     8807 non-null   object
 11  description   8807 non-null   object
dtypes: int64(1), object(11)
memory usage: 825.8+ KB
```

the basic statistics of numerical columns?

```
df.describe() ## apply STATICS
```

	release_year
count	8807.000000
mean	2014.180198
std	8.819312
min	1925.000000
25%	2013.000000
50%	2017.000000
75%	2019.000000
max	2021.000000

find missing/null values in the dataset?

```
df.isnull().sum()  ## check null values
```

```
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
```

Basic operations on columns

Now what operations can we do using columns?

Maybe add a column

or delete a column

or we can rename the column too

How can we get the names of all these cols ?

```
df.columns # using attribute `columns` of dataframe
```

```
Index(['show_id', 'type', 'title', 'director', 'cast', 'country', 'date_added',
       'release_year', 'rating', 'duration', 'listed_in', 'description'],
      dtype='object')
```

```
df.keys # using method keys() of dataframe
```

```
<bound method NDFrame.keys of      show_id     type                  title         director  \
0         s1    Movie   Dick Johnson Is Dead  Kirsten Johnson
1         s2  TV Show           Blood & Water              NaN
2         s3  TV Show               Ganglands  Julien Leclercq
3         s4  TV Show  Jailbirds New Orleans              NaN
4         s5  TV Show            Kota Factory              NaN
...      ...      ...                     ...              ...
8802   s8803    Movie                  Zodiac    David Fincher
8803   s8804  TV Show             Zombie Dumb              NaN
8804   s8805    Movie              Zombieland  Ruben Fleischer
8805   s8806    Movie                   Zoom     Peter Hewitt
8806   s8807    Movie                 Zubaan      Mozez Singh

                                                  cast        country  \
0                                                  NaN  United States
1     Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...   South Africa
2     Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...            NaN
3                                                  NaN            NaN
4     Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...          India
...                                                ...            ...
8802  Mark Ruffalo, Jake Gyllenhaal, Robert Downey J...  United States
8803                                               NaN            NaN
8804  Jesse Eisenberg, Woody Harrelson, Emma Stone, ...  United States
8805  Tim Allen, Courteney Cox, Chevy Chase, Kate Ma...  United States
8806  Vicky Kaushal, Sarah-Jane Dias, Raaghav Chanan...          India

              date_added  release_year rating   duration  \
0     September 25, 2021          2020  PG-13     90 min
1     September 24, 2021          2021  TV-MA  2 Seasons
2     September 24, 2021          2021  TV-MA   1 Season
3     September 24, 2021          2021  TV-MA   1 Season
4     September 24, 2021          2021  TV-MA  2 Seasons
...                  ...           ...    ...        ...
8802   November 20, 2019          2007      R    158 min
8803        July 1, 2019          2018  TV-Y7  2 Seasons
8804    November 1, 2019          2009      R     88 min
8805    January 11, 2020          2006     PG     88 min
8806       March 2, 2019          2015  TV-14    111 min

                                        listed_in  \
0                                   Documentaries
1       International TV Shows, TV Dramas, TV Mysteries
2     Crime TV Shows, International TV Shows, TV Act...
3                             Docuseries, Reality TV
4     International TV Shows, Romantic TV Shows, TV ...
...                                             ...
8802                    Cult Movies, Dramas, Thrillers
8803            Kids' TV, Korean TV Shows, TV Comedies
8804                            Comedies, Horror Movies
```

```
8805                Children & Family Movies, Comedies
8806     Dramas, International Movies, Music & Musicals

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

what is so "special" about this dictionary?

It can take multiple keys

```
df[['title','release_year']].head()
```

	title	release_year
0	Dick Johnson Is Dead	2020
1	Blood & Water	2021
2	Ganglands	2021
3	Jailbirds New Orleans	2021
4	Kota Factory	2021

How can we find the countries that have Netflix ?

We can find the unique country in Netflix

How can we find unique values in a column?

```
df['country'].unique()
```

```
array(['United States', 'South Africa', nan, 'India',
       'United States, Ghana, Burkina Faso, United Kingdom, Germany, Ethiopia',
       'United Kingdom', 'Germany, Czech Republic', 'Mexico', 'Turkey',
       'Australia', 'United States, India, France', 'Finland',
       'China, Canada, United States',
       'South Africa, United States, Japan', 'Nigeria', 'Japan',
       'Spain, United States', 'France', 'Belgium',
       'United Kingdom, United States', 'United States, United Kingdom',
       'France, United States', 'South Korea', 'Spain',
       'United States, Singapore', 'United Kingdom, Australia, France',
       'United Kingdom, Australia, France, United States',
       'United States, Canada', 'Germany, United States',
       'South Africa, United States', 'United States, Mexico',
       'United States, Italy, France, Japan',
       'United States, Italy, Romania, United Kingdom',
       'Australia, United States', 'Argentina, Venezuela',
       'United States, United Kingdom, Canada', 'China, Hong Kong',
       'Russia', 'Canada', 'Hong Kong', 'United States, China, Hong Kong',
       'Italy, United States', 'United States, Germany',
       'United Kingdom, Canada, United States', ', South Korea',
       'Ireland', 'India, Nepal',
       'New Zealand, Australia, France, United States', 'Italy',
       'Italy, Brazil, Greece', 'Argentina', 'Jordan', 'Colombia',
       'United States, Japan', 'Belgium, United Kingdom',
       'Switzerland, United Kingdom, Australia', 'Israel, United States',
       'Canada, United States', 'Brazil', 'Argentina, Spain', 'Taiwan',
       'United States, Nigeria', 'Bulgaria, United States',
       'Spain, United Kingdom, United States', 'United States, China',
       'United States, France',
       'Spain, France, United Kingdom, United States',
       ', France, Algeria', 'Poland', 'Germany',
       'France, Israel, Germany, United States, United Kingdom',
       'New Zealand', 'Saudi Arabia', 'Thailand', 'Indonesia',
```

```
'Egypt, Denmark, Germany', 'United States, Switzerland',
'Hong Kong, Canada, United States', 'Kuwait, United States',
'France, Canada, United States, Spain',
'France, Netherlands, Singapore', 'France, Belgium',
'Ireland, United States, United Kingdom', 'Egypt', 'Malaysia',
'Israel', 'Australia, New Zealand', 'United Kingdom, Germany',
'Belgium, Netherlands', 'South Korea, Czech Republic',
'Australia, Germany', 'Vietnam', 'United Kingdom, Belgium',
'United Kingdom, Australia, United States',
'France, Japan, United States',
'United Kingdom, Germany, Spain, United States',
'United Kingdom, United States, France, Italy',
'United States, Germany, Canada',
'United States, France, Italy, United Kingdom',
'United States, United Kingdom, Germany, Hungary',
'United States, New Zealand', 'Sweden', 'China', 'Lebanon',
'Romania', 'Finland, Germany', 'Lebanon, Syria', 'Philippines',
'Iceland', 'Denmark', 'United States, India',
'Philippines, Singapore, Indonesia',
'China, United States, Canada', 'Lebanon, United Arab Emirates',
'Canada, United States, Denmark', 'United Arab Emirates',
'Mexico, France, Colombia', 'Netherlands',
'Germany, United States, France', 'United States, Bulgaria',
'United Kingdom, France, Germany, United States',
'Norway, Denmark', 'Syria, France, Lebanon, Qatar'
```

```
df['Country'].str.split(',')
```

```
0       [United States]
1        [South Africa]
2             [Unknown]
3             [Unknown]
4               [India]
              ...
8802    [United States]
8803          [Unknown]
8804    [United States]
8805    [United States]
8806            [India]
Name: Country, Length: 8807, dtype: object
```

Now what if you also want to check the count of each country in the dataframe?

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

```
country
United States                              2818
India                                       972
United Kingdom                              419
Japan                                       245
South Korea                                 199
                                           ...
Romania, Bulgaria, Hungary                    1
Uruguay, Guatemala                            1
France, Senegal, Belgium                      1
Mexico, United States, Spain, Colombia        1
United Arab Emirates, Jordan                  1
Name: count, Length: 748, dtype: int64
```

What if we want to change the name of a column ?

We can rename the column by:

passing the dictionary with old_name:new_name pair

specifying axis=1

```
df.rename({"country":'Country','release_year': 'Release_year'}, axis=1)
df
##  not parmenent use parmanent ke liye inplace use karna padta h
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration	
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	D
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	T
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season	T
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	1 Season	
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons	

```
df['country'] # before use in inplace then this time  not  change
```

```
0       United States
1        South Africa
2                 NaN
3                 NaN
4               India
             ...
8802    United States
8803              NaN
8804    United States
8805    United States
8806            India
Name: Country, Length: 8807, dtype: object
```

Note

rename has default value of axis=0

If two columns have the same name, then df['column'] will display both columns

```
df.rename({"country":'Country','release_year': 'Release_year'}, axis=1, inplace=True)
df
```

```
0       United States
1        South Africa
2                 NaN
3                 NaN
4               India
             ...
8802    United States
```

```
8803              NaN
8804    United States
8805    United States
8806            India
Name: Country, Length: 8807, dtype: object
```

Now lets try another way of accessing column vals

```
df.Country
```

```
0       United States
1        South Africa
2                 NaN
3                 NaN
4               India
            ...
8802    United States
8803              NaN
8804    United States
8805    United States
8806            India
Name: Country, Length: 8807, dtype: object
```

What do you think could be the problems with using attribute style for accessing the columns?

Problems such as * if the column names are not strings

```
      ~ Starting with number: E.g., 2nd

      ~  Contains a space: E.g., Roll Number

  * or if the column names conflict with methods of the DataFrame
      ~ E.g. shape
```

Are all the columns in our data necessary?

We already know the type in which each country lies

So we don't need this column

How can we delete cols in pandas dataframe ?

```
df.drop('description', axis=1)
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration	
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	D
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	T
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season	T
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	1 Season	
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons	

The drop function takes two parameters:

The column name

```
. The axis
```

By default the value of axis is 0

An alternative to the above approach is using the "columns" parameter as we did in rename

```
df.drop(columns=['description'])
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration	
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	D
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	T
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season	T
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	1 Season	
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons	

As you can see, column description is dropped

Has the column permanently been deleted?

```
df.head ()
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration	lis
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	Docum
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	Inter TV Sh Dra M
					Sami						

NO, the column description is still there

Do you see what's happening here?

We only got a view of dataframe with column description dropped

How can we permanently drop the column?

We can either re-assign it

df = df.drop('description', axis=1) OR

We can set parameter inplace=True

By default, inplace=False

```
df.drop('description', axis=1, inplace=True)
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
Cell In[52], line 1
----> 1 df.drop('description', axis=1, inplace=True)

File c:\ProgramData\anaconda3\Lib\site-packages\pandas\core\frame.py:5258, in DataFrame.drop(self, labels,
axis, index, columns, level, inplace, errors)
   5110 def drop(
   5111     self,
   5112     labels: IndexLabel = None,
   (...)
   5119     errors: IgnoreRaise = "raise",
   5120 ) -> DataFrame | None:
   5121     """
   5122     Drop specified labels from rows or columns.
   5123
   (...)
   5256             weight  1.0     0.8
   5257     """
-> 5258     return super().drop(
   5259         labels=labels,
   5260         axis=axis,
   5261         index=index,
   5262         columns=columns,
   5263         level=level,
   5264         inplace=inplace,
   5265         errors=errors,
   5266     )

File c:\ProgramData\anaconda3\Lib\site-packages\pandas\core\generic.py:4549, in NDFrame.drop(self, labels,
axis, index, columns, level, inplace, errors)
   4547 for axis, labels in axes.items():
   4548     if labels is not None:
-> 4549         obj = obj._drop_axis(labels, axis, level=level, errors=errors)
   4551 if inplace:
   4552     self._update_inplace(obj)

File c:\ProgramData\anaconda3\Lib\site-packages\pandas\core\generic.py:4591, in NDFrame._drop_axis(self,
labels, axis, level, errors, only_slice)
   4589         new_axis = axis.drop(labels, level=level, errors=errors)
   4590     else:
-> 4591         new_axis = axis.drop(labels, errors=errors)
   4592     indexer = axis.get_indexer(new_axis)
   4594 # Case for non-unique axis
   4595 else:

File c:\ProgramData\anaconda3\Lib\site-packages\pandas\core\indexes\base.py:6699, in Index.drop(self,
labels, errors)
   6697 if mask.any():
   6698     if errors != "ignore":
-> 6699         raise KeyError(f"{list(labels[mask])} not found in axis")
   6700     indexer = indexer[~mask]
   6701 return self.delete(indexer)
```

```
df.head() #we print the head to check
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration	lis
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	Docum
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	Inter TV Sh Drai M
					Sami Bouajila,						C

Now we can see the column description is permanently dropped

Now similarly, what if we want to create a new column?

We can either

```
. use values from existing columns
```

OR

```
.  create our own values
```

How to create a column using values from an existing column?

```
df["Release_year+7"] = df["Release_year"] + 7
df.head()
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration	lis
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	Docum
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	Inter TV Sh Drai M
					Sami Bouajila,						C

As we see, a new column year+7 is created from the column year

We can also use values from two columns to form a new column

Which two columns can we use to create a new column gdp ?

```
df['Release']=df['Release_year'] * df['rating']
df.head()
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration	lis
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	Docum
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane,	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	Inter TV Sh Drai M

As you can see

. An additional column has been created

. Values in this column are product of respective values in gdp_cap and population

What other operations we can use?

Subtraction, Addition, etc.

How can we create a new column from our own values?

. We can create a list

OR

. We can create a Pandas Series from a list/numpy array for our new column Country

```
df["Own"] = [i for i in range(8807)] # count of these values should be correct
df
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration	
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	D
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	T
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season	T
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	1 Season	
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons	
...	...	...	...	...	...	...	...	...	...	...	
8802	s8803	Movie	Zodiac	David Fincher	Mark Ruffalo, Jake Gyllenhaal, Robert Downey J...	United States	November 20, 2019	2007	R	158 min	
8803	s8804	TV Show	Zombie Dumb	NaN	NaN	NaN	July 1, 2019	2018	TV-Y7	2 Seasons	
8804	s8805	Movie	Zombieland	Ruben Fleischer	Jesse Eisenberg, Woody Harrelson, Emma Stone, ...	United States	November 1, 2019	2009	R	88 min	H
8805	s8806	Movie	Zoom	Peter Hewitt	Tim Allen, Courteney Cox, Chevy Chase, Kate Ma...	United States	January 11, 2020	2006	PG	88 min	Fa
8806	s8807	Movie	Zubaan	Mozez Singh	Vicky Kaushal, Sarah-Jane Dias, Raaghav Chanan...	India	March 2, 2019	2015	TV-14	111 min	M

8807 rows × 14 columns

Now that we know how to create new cols lets see some basic ops on rows

Before that lets drop the newly created cols

```
df.drop(columns=["Release_year+7"], inplace=True, errors='ignore')
df
```

```
df.drop(columns=["Release_year+7","Own","Release"], inplace=True, errors='ignore')
df
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration	
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	D
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	T
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season	T
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	1 Season	
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons	

Working with Rows

Just like columns, do rows also have labels?

YES

Notice the indexes in bold against each row ,
Lets see how can we access these indexes

```
df.index.values
```

```
array([   0,    1,    2, ..., 8804, 8805, 8806], dtype=int64)
```

Can we change row labels (like we did for columns)?

What if we want to start indexing from 1 (instead of 0)?

```
df.index = list(range(1, df.shape[0]+1)) # create a list of indexes of same length
df
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration	
1	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	D
2	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	T
3	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season	T
4	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	1 Season	
5	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons	

Explicit and Implicit Indices

What are these row labels/indices exactly ?

. They can be called identifiers of a particular row
.Specifically known as explicit indices

Additionally, can series/dataframes can also use python style indexing?

YES

The python style indices are known as implicit indices

How can we access explicit index of a particular row?

. Using df.index[]
. Takes impicit index of row to give its explicit index

```
df.index[1] #Implicit index 1 gave explicit index 2
```

```
2
```

But why not use just implicit indexing ?

Explicit indices can be changed to any value of any datatype

Eg: Explicit Index of 1st row can be changed to First

Or, something like a floating point value, say 1.0

```
df.index = np.arange(1, df.shape[0]+1, dtype='float')
df
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration
1.0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min
2.0	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons
3.0	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season
4.0	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	1 Season
5.0	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons

As we can see, the indices are floating point values now

```
sample = df.head()
sample
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration	l
1.0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	Docu
2.0	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	Int TV S Dr
					Sami Bouajila,						

Now what if we want to use string indices?

```
sample.index = ['a', 'b', 'c', 'd', 'e']
sample
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration	lis
a	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	Docum
b	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	Inter TV Sh Dra M
					Sami Bouajila,						C

loc and iloc

1. loc

Allows indexing and slicing that always references the explicit index

```
df.loc[1]
```

```
show_id                         s1
type                         Movie
title          Dick Johnson Is Dead
director            Kirsten Johnson
cast                            NaN
Country               United States
date_added        September 25, 2021
Release_year                    2020
rating                         PG-13
duration                      90 min
listed_in              Documentaries
Name: 1.0, dtype: object
```

```
df.loc[1:3]
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration	l
1.0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	Docu
2.0	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema,	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	Int TV S Dr

Now as previous Dataset index

```
df.index = np.arange(0, df.shape[0]+0, dtype='int')
df
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration	
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	D
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	T
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season	T
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	1 Season	
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons	

2. iloc

Allows indexing and slicing that always references the implicit Python-style index

```
df.iloc[1]
```

```
show_id                                                    s2
type                                                  TV Show
title                                           Blood & Water
director                                                  NaN
cast          Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...
Country                                           South Africa
date_added                                 September 24, 2021
Release_year                                             2021
rating                                                  TV-MA
duration                                            2 Seasons
listed_in       International TV Shows, TV Dramas, TV Mysteries
Name: 1, dtype: object
```

Now will iloc also consider the range inclusive?

```
df.iloc[0:2]
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration	liste
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	Documen
					Ama						

What if we want to access multiple non-consecutive rows at same time ?

For eg: rows 1, 10, 100

```
df.iloc[[1,10,100]]
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration	list
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	Intern TV S TV D My

What about negative index?

Which would work between iloc and loc ?

```
df.iloc[-1]
# Works and gives last row in dataframe
```

```
show_id                                                 s8807
type                                                    Movie
title                                                  Zubaan
director                                         Mozez Singh
cast          Vicky Kaushal, Sarah-Jane Dias, Raaghav Chanan...
Country                                                 India
date_added                                      March 2, 2019
Release_year                                             2015
rating                                                  TV-14
duration                                            111 min
listed_in       Dramas, International Movies, Music & Musicals
Name: 8806, dtype: object
```

```
df.loc[-1]
# Does NOT work
```

So, why did iloc[-1] worked, but loc[-1] didn't?

```
. Because iloc works with positional indices, while loc with assigned labels

. [-1] here points to the row at last position in iloc
```

How to find missing/null values in the dataset?

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

```
show_id          0
type             0
title            0
director         0
cast           825
Country          0
date_added      10
Release_year     0
rating           4
```

```
duration          3
listed_in         0
dtype: int64
```

How to remove duplicate rows from the dataset?

```
df.drop_duplicates(inplace=True)
df
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration	
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	D
1	s2	TV Show	Blood & Water	No Director	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	Unknown	September 24, 2021	2021	TV-MA	1 Season	
3	s4	TV Show	Jailbirds New Orleans	No Director	NaN	Unknown	September 24, 2021	2021	TV-MA	1 Season	
4	s5	TV Show	Kota Factory	No Director	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons	

```
df.drop_duplicates(subset=['Country'],keep='first')
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons
7	s8	Movie	Sankofa	Haile Gerima	Kofi Ghanaba, Oyafunmike Ogunlano,	United States, Ghana, Burkina Faso,	September 24, 2021	1993	TV-MA	125 min

How to handle missing values in the dataset?

```
df['director'].fillna('No Director', inplace=True)
df['Country'].fillna('Unknown', inplace=True)
df
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration	
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	D
1	s2	TV Show	Blood & Water	No Director	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	-
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	Unknown	September 24, 2021	2021	TV-MA	1 Season	-
3	s4	TV Show	Jailbirds New Orleans	No Director	NaN	Unknown	September 24, 2021	2021	TV-MA	1 Season	
4	s5	TV Show	Kota Factory	No Director	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons	

We can mention ranges using column labels as well in loc

```
df.loc[1:5, 'title':'duration']
```

	title	director	cast	Country	date_added	Release_year	rating	duration
1	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons
2	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season
3	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	1 Season

Sorting

If you notice, life_exp col is not sorted

How can we perform sorting in pandas ?

```
df.sort_values(['duration'])
```

	show_id	type	title	director	cast	Country	date_added	Release_year	rating	duration
8216	s8217	TV Show	The Bomb Squad	NaN	Christopher Eccleston	United Kingdom	February 1, 2019	2011	TV-MA	1 Season
5392	s5393	TV Show	Barbie Life in the Dreamhouse	NaN	Kate Higgins, Sean Hankinson, Haviland Stillwe...	United States, Canada	July 1, 2017	2012	TV-Y	1 Season
3794	s3795	TV Show	Historical Roasts	NaN	Jeff Ross, Bob Saget, James Adomian, Rachel Fe...	United States	May 27, 2019	2019	TV-MA	1 Season
1593	s1594	TV Show	Kings of Jo'Burg	NaN	Shona Ferguson, Zolisa Xaluva, Tsholofelo Mats...	South Africa	December 4, 2020	2020	TV-MA	1 Season
5393	s5394	TV Show	Breakout	NaN	Jeanette Aw, Elvin Ng, Zhou Ying, Christopher ...	NaN	July 1, 2017	2010	TV-14	1 Season

Which directors have the most content on Netflix?

```
df['director'].value_counts().head(10)
```

```
director
Rajiv Chilaka             19
Raúl Campos, Jan Suter    18
Marcus Raboy              16
Suhas Kadav               16
Jay Karas                 14
Cathy Garcia-Molina       13
Martin Scorsese           12
Youssef Chahine           12
Jay Chapman               12
Steven Spielberg          11
Name: count, dtype: int64
```

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

```
director
Rajiv Chilaka                     19
Raúl Campos, Jan Suter            18
Marcus Raboy                      16
Suhas Kadav                       16
Jay Karas                         14
                                  ..
Raymie Muzquiz, Stu Livingston     1
Joe Menendez                       1
Eric Bross                         1
Will Eisenberg                     1
Mozez Singh                        1
Name: count, Length: 4528, dtype: int64
```

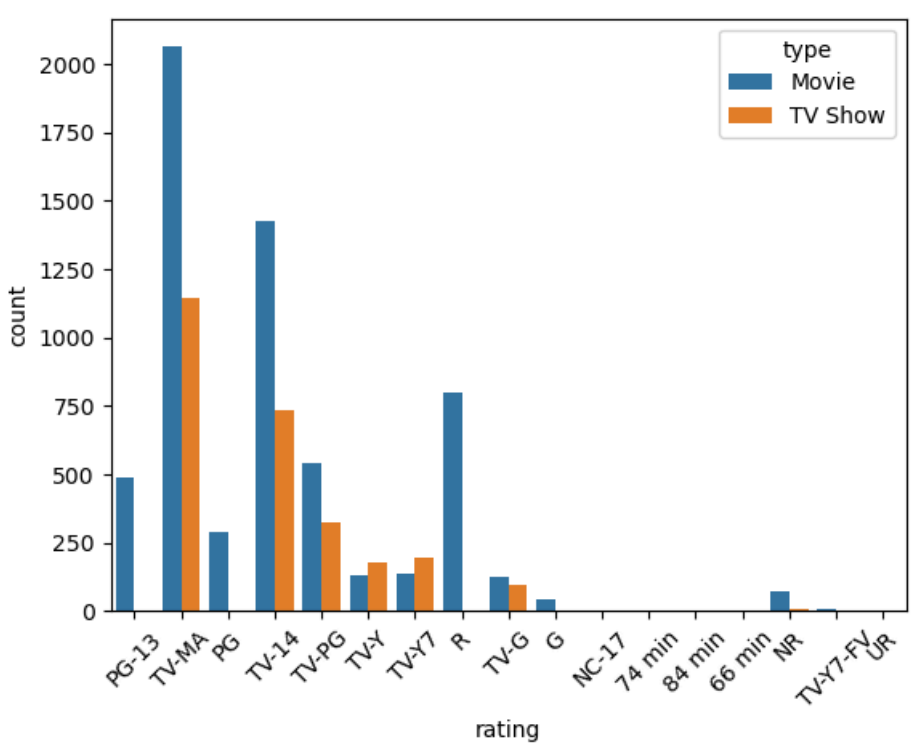
Which country has the most TV Shows on Netflix?

```
df[df['type'] == 'TV Show']['Country'].value_counts().head(10)
```

```
Country
United States     760
United Kingdom    213
Japan             169
South Korea       158
India              79
Taiwan             68
Canada             59
France             49
Australia          48
Spain              48
Name: count, dtype: int64
```

How are ratings distributed among content types?

```
sns.countplot(data=df, x='rating', hue='type')
plt.xticks(rotation=45)
plt.show()
```



Which countries have the most Netflix content?

```
df['Country'].value_counts().head(10)
```

```
Country
United States     2818
India              972
United Kingdom     419
Japan              245
South Korea        199
Canada             181
Spain              145
France             124
Mexico             110
Egypt              106
Name: count, dtype: int64
```

Which ratings are most common on Netflix?

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

```
rating
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: count, dtype: int64
```

Univariate Analysis

What are the counts of different content types (Movie or TV Show)?

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

```
type
Movie      6131
TV Show    2676
Name: count, dtype: int64
```

What is the distribution of content durations? (Movies vs. TV Shows)

```
df['duration'].value_counts().head(10)
```

```
duration
1 Season     1793
2 Seasons     425
3 Seasons     199
90 min        152
94 min        146
97 min        146
93 min        146
91 min        144
95 min        137
96 min        130
Name: count, dtype: int64
```

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

```
duration
1 Season     1793
2 Seasons     425
3 Seasons     199
90 min        152
94 min        146
             ...
16 min          1
```

```
186 min         1
193 min         1
189 min         1
191 min         1
Name: count, Length: 220, dtype: int64
```

Multivariate Analysis

Which countries produce more Movies vs. TV Shows?

```
df.groupby(['Country', 'type']).size().unstack().fillna(0).sort_values('Movie', ascending=False).head(10)
```

type	Movie	TV Show
Country		
United States	2058.0	760.0
India	893.0	79.0
United Kingdom	206.0	213.0
Canada	122.0	59.0
Spain	97.0	48.0
Egypt	92.0	14.0
Nigeria	86.0	9.0
Indonesia	77.0	2.0
Japan	76.0	169.0
Turkey	76.0	29.0

groupby

How many shows are available in each country?

```
df.groupby('Country').size().sort_values(ascending=False).head(10)
```

```
Country
United States     2818
India              972
Unknown            831
United Kingdom     419
Japan              245
South Korea        199
Canada             181
Spain              145
France             124
Mexico             110
dtype: int64
```

What is the distribution of Movies and TV Shows in each country?

```
df.groupby(['Country', 'type']).size().unstack().fillna(0)
```

Country	type Movie	TV Show
, France, Algeria	1.0	0.0
, South Korea	0.0	1.0
Argentina	38.0	18.0
Argentina, Brazil, France, Poland, Germany, Denmark	1.0	0.0
Argentina, Chile	2.0	0.0
...	...	...
Venezuela	1.0	0.0
Venezuela, Colombia	1.0	0.0
Vietnam	7.0	0.0
West Germany	1.0	0.0
Zimbabwe	1.0	0.0

749 rows × 2 columns

GroupBy with Multiple Columns and Conditions

. In which year did each country add the most content?

```
df.groupby(['Country', 'Release_year']).size().groupby('Country').idxmax()
```

```
Country
, France, Algeria                                                (, France, Algeria, 2014)
, South Korea                                                        (, South Korea, 2021)
Argentina                                                               (Argentina, 2020)
Argentina, Brazil, France, Poland, Germany, Denmark    (Argentina, Brazil, France, Poland, Germany, D...
Argentina, Chile                                                   (Argentina, Chile, 2006)
                                                                                ...
Venezuela                                                               (Venezuela, 2017)
Venezuela, Colombia                                          (Venezuela, Colombia, 2007)
Vietnam                                                                 (Vietnam, 2019)
West Germany                                                         (West Germany, 1977)
Zimbabwe                                                                (Zimbabwe, 2017)
Length: 749, dtype: object
```

How many Movies and TV Shows does each director have?

```
df.groupby(['director', 'type']).size().sort_values(ascending=False).head(10)
```

```
director               type
No Director            TV Show    2446
                       Movie       188
Rajiv Chilaka          Movie        19
Raúl Campos, Jan Suter Movie        18
Suhas Kadav            Movie        16
Marcus Raboy           Movie        15
Jay Karas              Movie        14
Cathy Garcia-Molina    Movie        13
Jay Chapman            Movie        12
Martin Scorsese        Movie        12
dtype: int64
```

How many contents were released each year per country? (Trend Analysis)

```
df.groupby(['Country', 'Release_year']).size().unstack(fill_value=0).head(1000)
```

Release_year	1925	1942	1943	1944	1945	1946	1947	1954	1955	1956	...	2012	2013	2014	2015	2016	2
Country																	
, France, Algeria	0	0	0	0	0	0	0	0	0	0	...	0	0	1	0	0	
, South Korea	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	
Argentina	0	0	0	0	0	0	0	0	0	0	...	0	0	0	4	7	
Argentina, Brazil, France, Poland, Germany, Denmark	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	
Argentina, Chile	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
Venezuela	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	
Venezuela, Colombia	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	

What is the distribution of content by each rating type (like TV-MA, PG)?

```
df.groupby('rating').size().sort_values(ascending=False)
```

```
rating
TV-MA       3207
TV-14       2160
TV-PG        863
R            799
PG-13        490
TV-Y7        334
TV-Y         307
PG           287
TV-G         220
NR            80
G             41
TV-Y7-FV       6
UR             3
NC-17          3
74 min         1
84 min         1
66 min         1
dtype: int64
```

What is the most popular genre for each director?

```
df.groupby(['director', 'listed_in']).size().sort_values(ascending=False).head(15)
```

```
director     listed_in
No Director  Kids' TV                                                    214
             International TV Shows, TV Dramas                           113
             Reality TV                                                   94
             Crime TV Shows, International TV Shows, TV Dramas            92
             Kids' TV, TV Comedies                                        91
```

```
International TV Shows, Romantic TV Shows, TV Dramas                 88
Anime Series, International TV Shows                                 83
International TV Shows, Romantic TV Shows, TV Comedies               82
Docuseries                                                           68
TV Comedies                                                          67
International TV Shows, Korean TV Shows, Romantic TV Shows           60
Crime TV Shows, International TV Shows, Spanish-Language TV Shows    57
Crime TV Shows, Docuseries                                           39
Docuseries, Science & Nature TV                                      37
International TV Shows, Reality TV                                   36
dtype: int64
```

Which country contributed the most content to each genre?

```
df.groupby(['Country', 'listed_in']).size().sort_values(ascending=False).head(20)
```

```
Country        listed_in
United States  Documentaries                                          249
               Stand-Up Comedy                                        209
India          Comedies, Dramas, International Movies                 120
               Dramas, International Movies                           118
               Dramas, Independent Movies, International Movies       108
United States  Children & Family Movies, Comedies                      90
               Dramas                                                  88
               Comedies                                                84
               Children & Family Movies                                80
               Kids' TV                                                77
Japan          Anime Series, International TV Shows                    75
United States  Dramas, Independent Movies                              74
               Reality TV                                              71
Unknown        Children & Family Movies                                70
United States  Action & Adventure                                      64
South Korea    International TV Shows, Korean TV Shows, Romantic TV Shows  63
United States  Comedies, Dramas, Independent Movies                    63
India          Dramas, International Movies, Romantic Movies           62
United States  Docuseries                                              61
               TV Comedies                                             61
dtype: int64
```

```
df['type'].value_counts()
# OR
df.groupby('type').size()
```

```
type
Movie      6131
TV Show    2676
dtype: int64
```

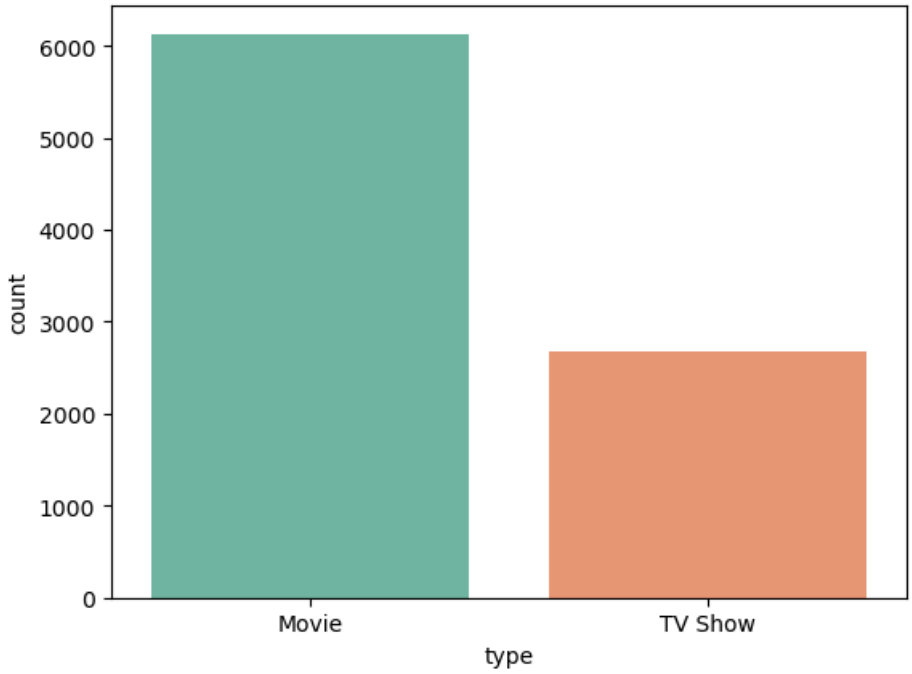
```
df.groupby('type').count()
```

type	show_id	title	director	cast	Country	date_added	Release_year	rating	duration	listed_in
Movie	6131	6131	6131	5656	6131	6131	6131	6129	6128	6131
TV Show	2676	2676	2676	2326	2676	2666	2676	2674	2676	2676

countplot()

```
sns.countplot(data=df, x='type', palette='Set2') # palette is color design
```

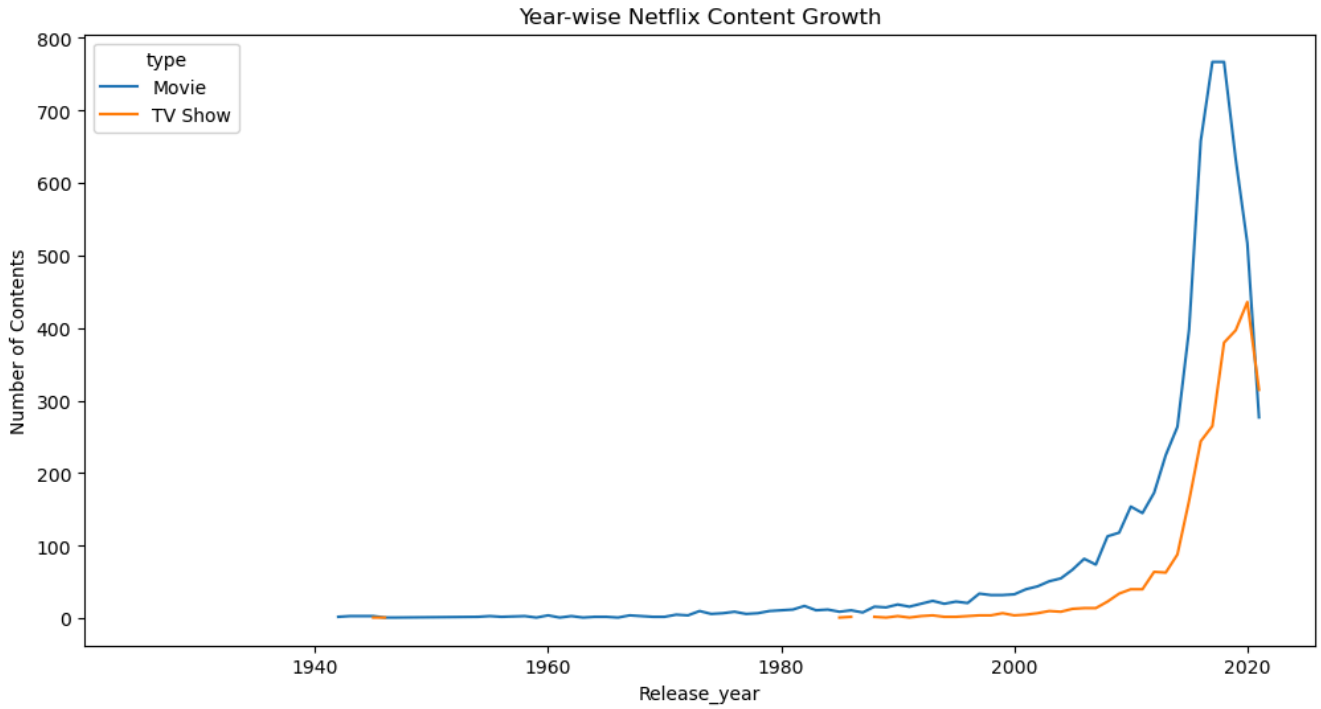
```
<Axes: xlabel='type', ylabel='count'>
```



use countplot()

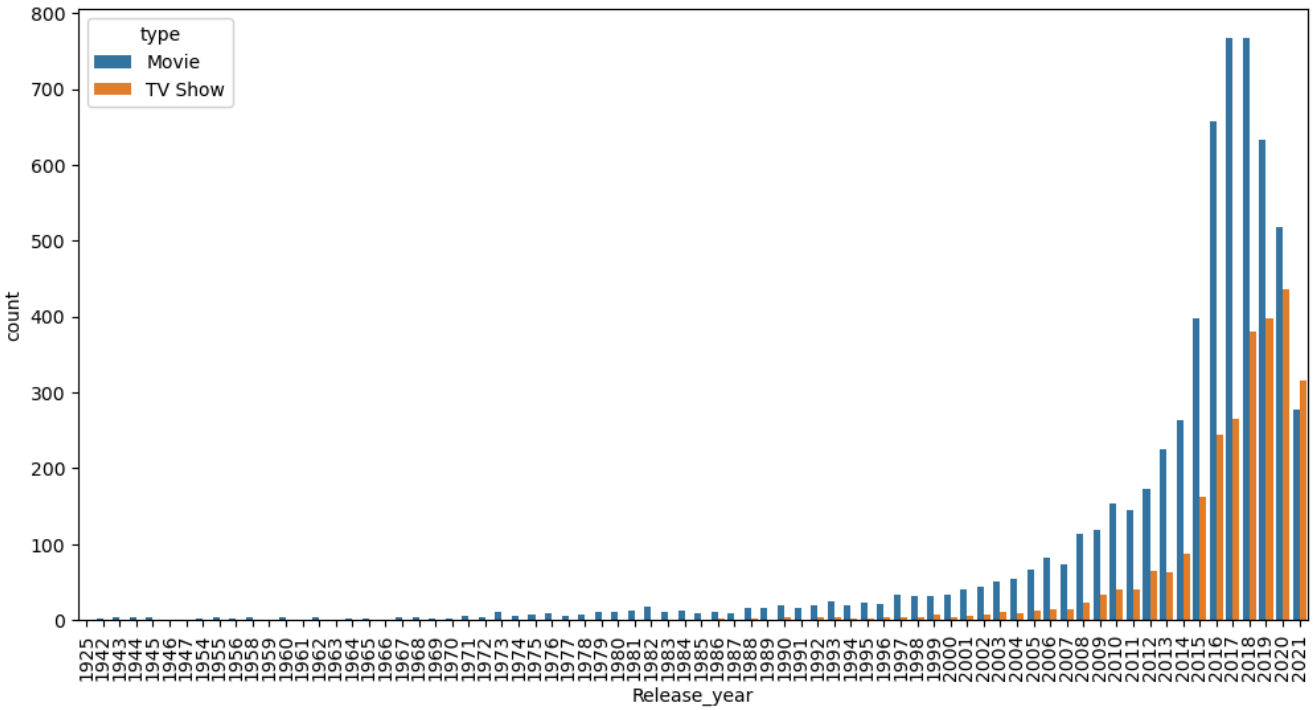
```
sns.countplot(data=df, x='type', hue='Country', palette='bright')

## large amount of Answer this code
# use hue = This gives different categories different colors, making it easier to identify patterns
```

```
import seaborn as sns
import matplotlib.pyplot as plt

df.groupby(['Release_year', 'type']).size().unstack().plot(kind='line', figsize=(12,6))
plt.title('Year-wise Netflix Content Growth')
plt.ylabel('Number of Contents')
plt.show()
```



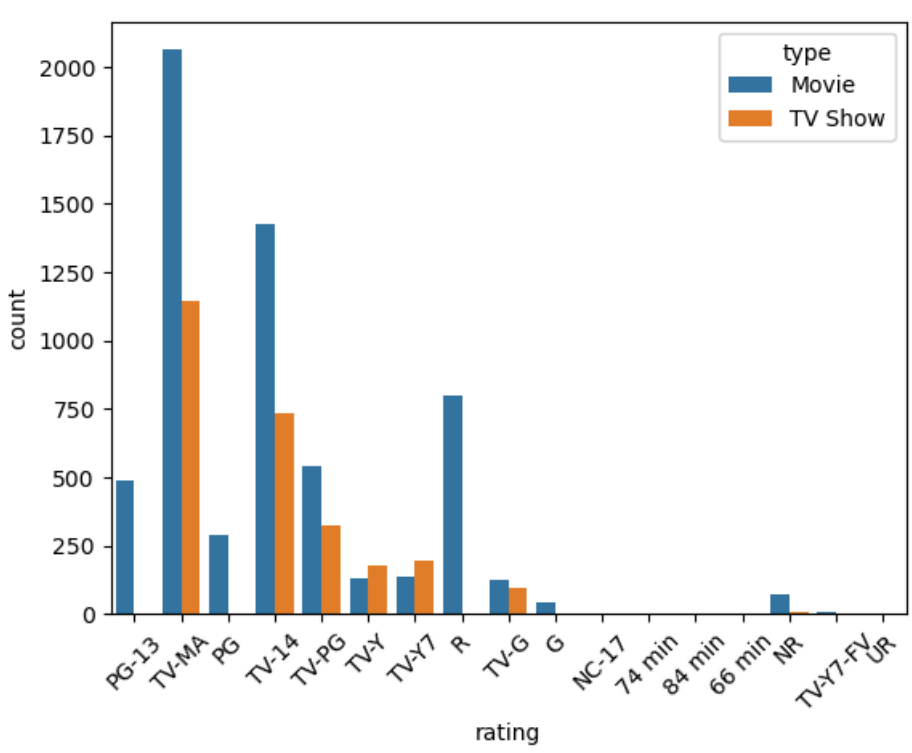
What is the yearly distribution of Movies and TV Shows on Netflix?

```
import seaborn as sns
import matplotlib.pyplot as plt

plt.figure(figsize=(12,6))
sns.countplot(data=df, x='Release_year', hue='type', order=sorted(df['Release_year'].unique()))
plt.xticks(rotation=90)
plt.show()
```



How are ratings distributed among content types?

```
sns.countplot(data=df, x='rating', hue='type')
plt.xticks(rotation=45)
plt.show()
```

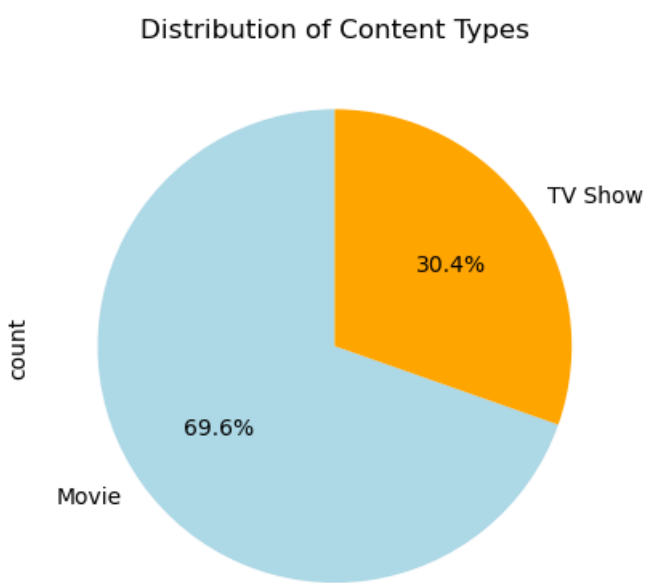


What is the year-wise growth trend of Movies and TV Shows on Netflix?

```
df.groupby(['Release_year', 'type']).size().unstack(fill_value=0).plot(kind='line', figsize=(12,6))
```

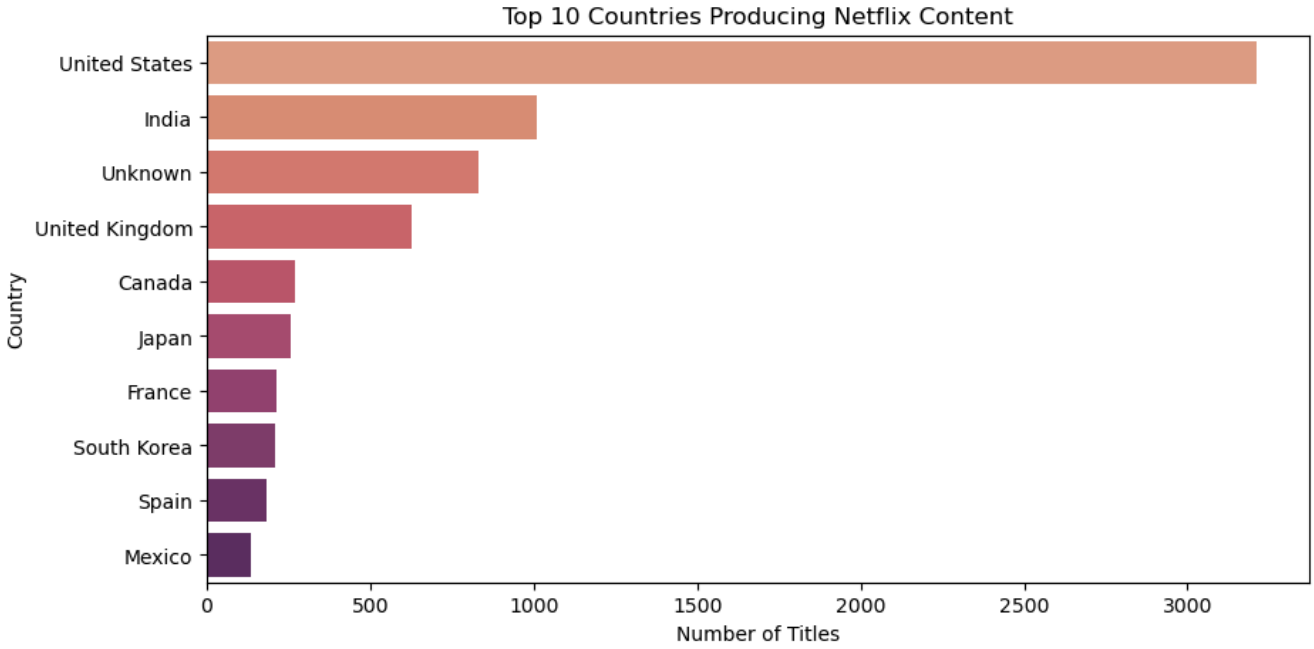
Example Final Visualization: Pie Chart

```
df['type'].value_counts().plot.pie(autopct='%1.1f%%', startangle=90, colors=['lightblue', 'orange'])
plt.title('Distribution of Content Types')
plt.show()
```



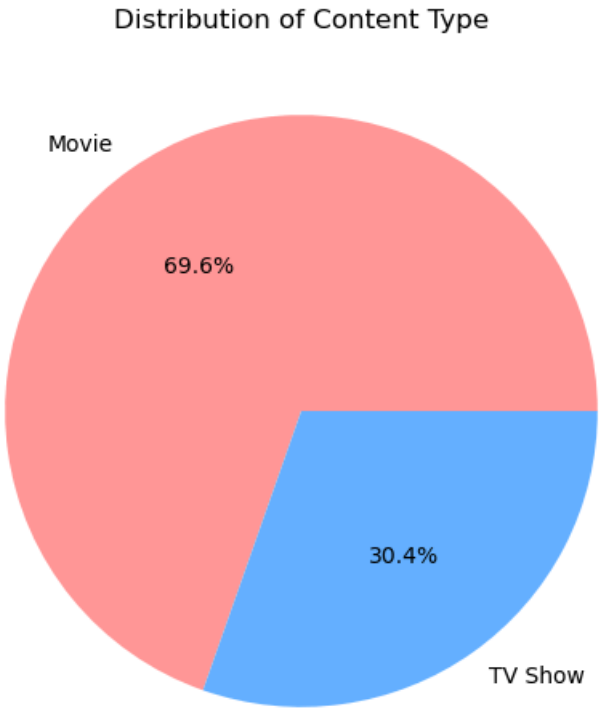
Content by Country: Bar Plot

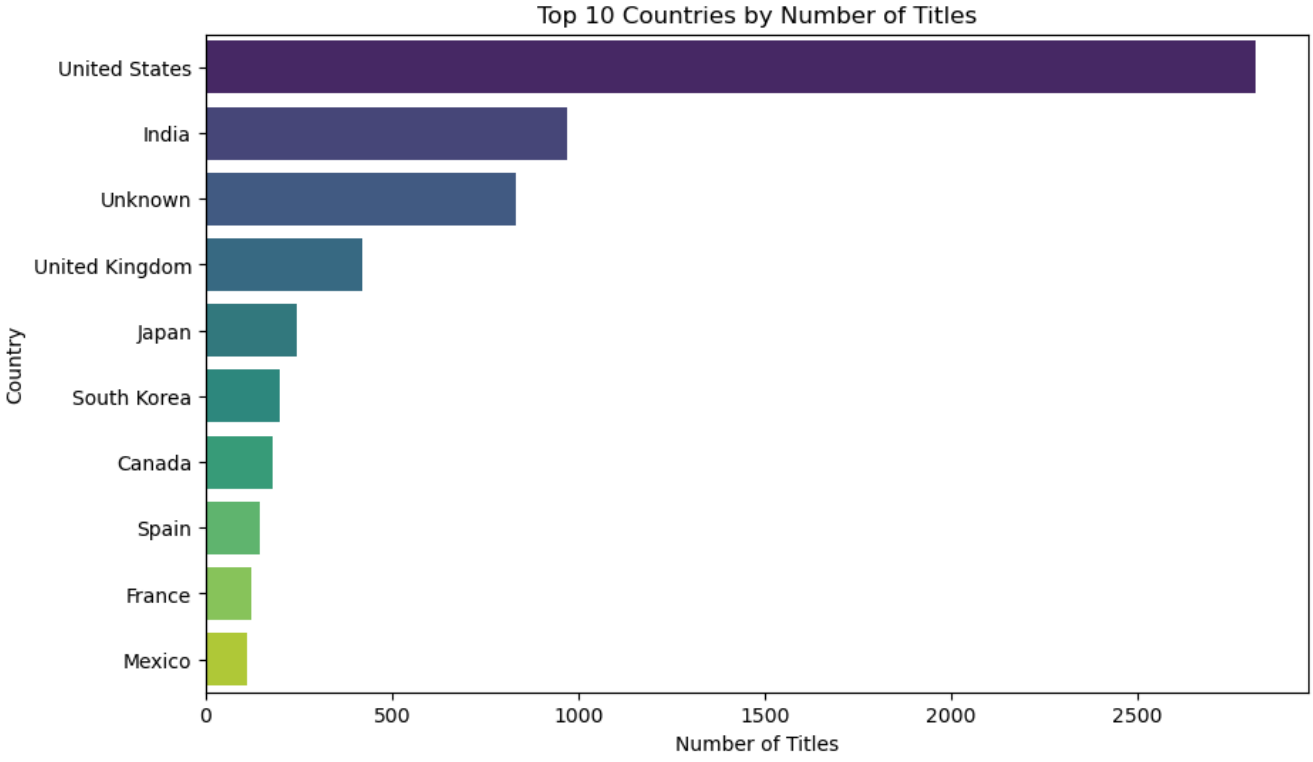
```
df['Country_simple'] = df['Country'].str.split(',').str[0]
top_Countries = df['Country_simple'].value_counts().head(10)
plt.figure(figsize=(10,5))
sns.barplot(x=top_Countries.values, y=top_Countries.index, palette='flare')
plt.title('Top 10 Countries Producing Netflix Content')
plt.xlabel('Number of Titles')
plt.ylabel('Country')
plt.show()
```

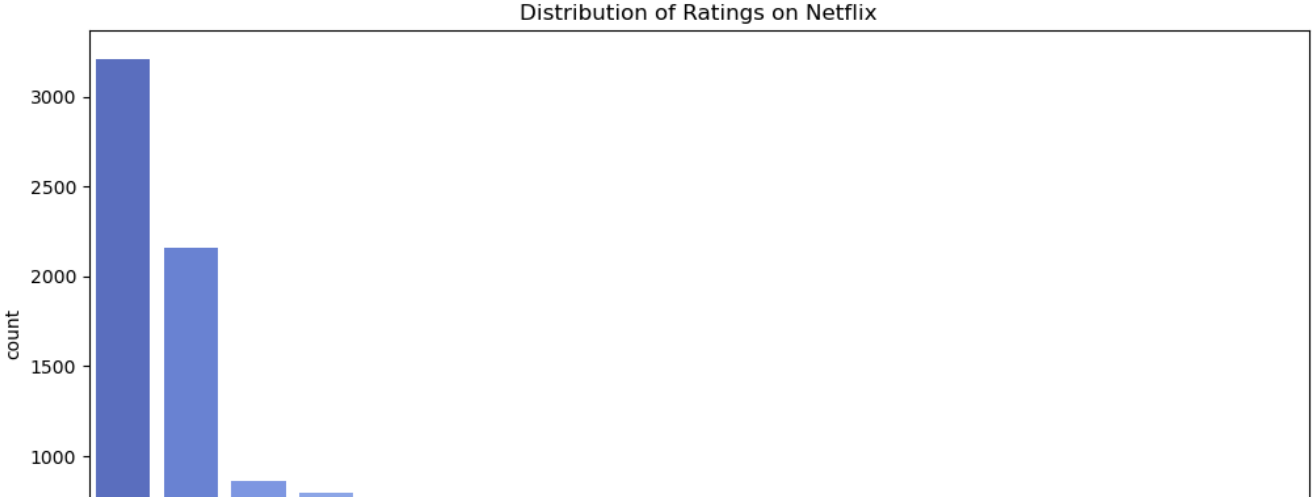


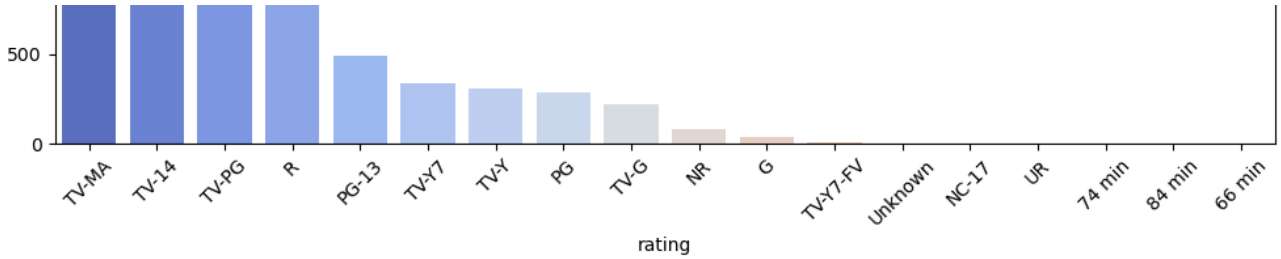

```
# Import required libraries
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px

# Load dataset
df = pd.read_csv('netflix_titles.csv')

# Data Cleaning
df['country'].fillna('Unknown', inplace=True)
df['rating'].fillna('Unknown', inplace=True)
df['release_year'].fillna(0, inplace=True)
df['listed_in'] = df['listed_in'].str.split(', ')
genres = df.explode('listed_in')

# Plot 1: Content Type Distribution (Matplotlib)
plt.figure(figsize=(6, 6))
type_counts = df['type'].value_counts()
plt.pie(type_counts, labels=type_counts.index, autopct='%1.1f%%', colors=['#ff9999','#66b3ff'])
plt.title('Distribution of Content Type')
plt.show()

# Plot 2: Top 10 Countries by Number of Titles (Seaborn)
plt.figure(figsize=(10, 6))
top_countries = df['country'].value_counts().head(10)
sns.barplot(x=top_countries.values, y=top_countries.index, palette='viridis')
plt.title('Top 10 Countries by Number of Titles')
plt.xlabel('Number of Titles')
plt.ylabel('Country')
plt.show()

# Plot 3: Rating Distribution (Seaborn)
plt.figure(figsize=(12, 6))
sns.countplot(data=df, x='rating', order=df['rating'].value_counts().index, palette='coolwarm')
plt.title('Distribution of Ratings on Netflix')
plt.xticks(rotation=45)
plt.show()

# Plot 4: Top 10 Genres (Seaborn)
plt.figure(figsize=(10, 6))
top_genres = genres['listed_in'].value_counts().reset_index().head(10)
```

```
top_genres.columns = ['Genre', 'Count']
sns.barplot(data=top_genres, x='Count', y='Genre', palette='magma')
plt.title('Top 10 Genres on Netflix')
plt.xlabel('Number of Titles')
plt.ylabel('Genre')
plt.show()

# Plot 5: Content Release Trend Over Years (Plotly)
release_trend = df['release_year'].value_counts().reset_index()
release_trend.columns = ['Year', 'Count']
release_trend = release_trend.sort_values('Year')
fig = px.line(release_trend, x='Year', y='Count', title='Content Release Trend Over Years')
fig.show()
```


Distribution of Content Type
Movie
69.6%
30.4%
TV Show


Top 10 Countries by Number of Titles
United States
India
Unknown
United Kingdom
Japan
South Korea
Canada
Spain
France
Mexico
Country
0
500
1000
1500
2000
2500
Number of Titles


Distribution of Ratings on Netflix
3000
2500
2000
1500
1000
count


500
0
TV-MA
TV-14
TV-PG
R
PG-13
TV-Y7
TV-Y
PG
TV-G
NR
G
TV-Y7-FV
Unknown
NC-17
UR
74 min
84 min
66 min
rating


Top 10 Genres on Netflix
International Movies
Dramas