

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

```
#encoding='cp1252'
#n= pd.read_excel("./Data/netflix_titles.xlsx")
#n.head(1)
```

show_id	type	title	director	cast
country \				
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson NaN United States

date_added	release_year	rating	duration	listed_in \
0	2021-09-25	2020	PG-13	90 min Documentaries

	description
0	As her father nears the end of his life, filmm...

```
# Convert 'DateColumn' to datetime format
n['ExtractedDate'] = pd.to_datetime(n['date_added'], format='%B %d, %Y')
n = n.dropna(subset=['ExtractedDate'])

n['ExtractedDate']=n['ExtractedDate'].dt.year
n['ExtractedDate'].isnull().sum()
#n.head(1)
```

```
0
```

```
content_freq_year=[2008,2009,2010,2011,2012,2013,2014,2015,2016,2017,2018,2019,2020,2021]
```

```
n2=n[n.ExtractedDate.isin(content_freq_year)]
n2 = n2.groupby(["ExtractedDate", "type"])["type"].count()
n2 = n2.unstack()
n2
```

type	Movie	TV Show
ExtractedDate		
2008	1.0	1.0
2009	2.0	NaN
2010	1.0	NaN
2011	13.0	NaN
2012	3.0	NaN
2013	6.0	5.0
2014	19.0	5.0
2015	56.0	26.0
2016	253.0	176.0

2017	839.0	349.0
2018	1237.0	412.0
2019	1424.0	592.0
2020	1284.0	595.0
2021	993.0	505.0

```

n3=n[n.ExtractedDate.isin(content_freq_year)]
n3.isnull().sum()
n3=n3[n3["country"].isnull() ==False]
#n.isnull().sum()
n3.isnull().sum()

```

show_id	0
type	0
title	0
director	2216
cast	671
country	0
date_added	0
release_year	0
rating	3
duration	3
listed_in	0
description	0
ExtractedDate	0
dtype: int64	

```

n3= n3.groupby(["ExtractedDate", "type"])["type"].count()
n3 = n3.unstack()
n3

```

type	Movie	TV Show
ExtractedDate		
2008	1.0	1.0
2009	2.0	NaN
2010	1.0	NaN
2011	13.0	NaN
2012	3.0	NaN
2013	6.0	5.0
2014	19.0	5.0
2015	54.0	25.0
2016	244.0	166.0
2017	814.0	309.0
2018	1192.0	338.0
2019	1349.0	509.0
2020	1223.0	548.0
2021	770.0	370.0

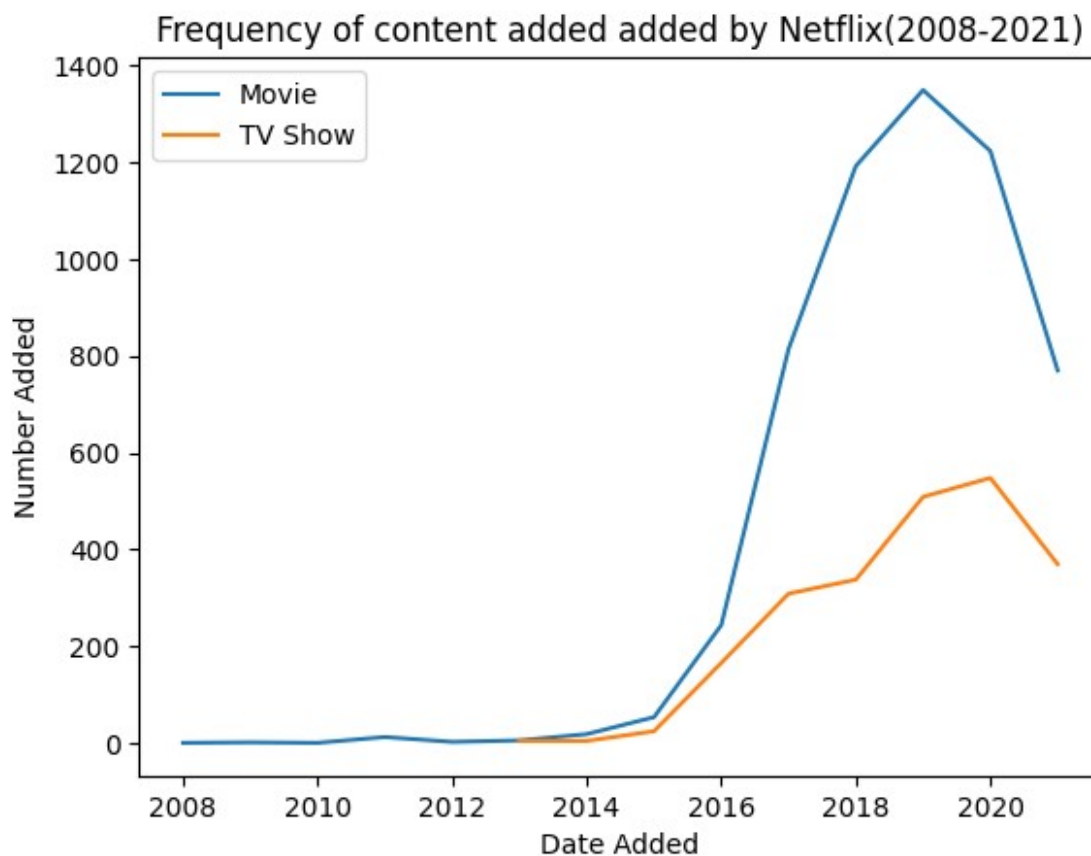
```

#n2.plot.line(marker='o')
n3.plot.line()
# Adding labels and title
plt.xlabel('Date Added')
plt.ylabel('Number Added')
plt.title('Frequency of content added added by Netflix(2008-2021)')

# Adding a legend
plt.legend()

# Display the plot
plt.show()

```



```

# Assuming "listed_in" contains lists
content_freq_year = [2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015,
2016, 2017, 2018, 2019, 2020, 2021]
n4 = n[n['ExtractedDate'].isin(content_freq_year)]

# Splitting "listed_in" values and exploding the DataFrame
n4['listed_in'] = n4['listed_in'].apply(lambda x: [item.strip() for
item in x.split(',')])
n4 = n4.explode('listed_in')

```

```

# Counting occurrences of each genre
genre_counts = n4['listed_in'].value_counts()

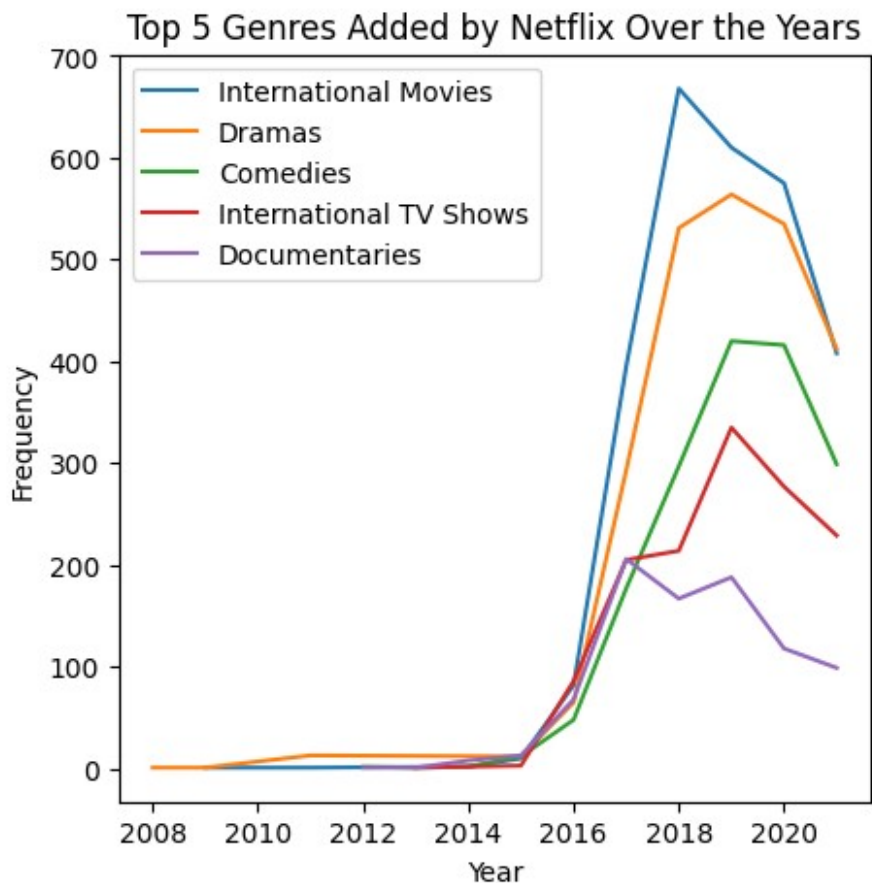
# Selecting the top 5 genres
top_5_genres = genre_counts.head(5)
plt.figure(figsize=(5,5))
# Plotting a line plot for each genre
for genre in top_5_genres.index:
    genre_data = n4[n4['listed_in'] ==
genre].groupby('ExtractedDate').size()
    plt.plot(genre_data.index, genre_data.values, label=genre)

# Adding labels and title
plt.xlabel('Year')
plt.ylabel('Frequency')
plt.title('Top 5 Genres Added by Netflix Over the Years')

# Adding a legend
plt.legend()

# Displaying the plot
plt.show()

```



```

# Assuming "listed_in" contains lists
plt.figure(figsize=(15, 8))

content_freq_year = [2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015,
2016, 2017, 2018, 2019, 2020, 2021]
n4 = n[n['ExtractedDate'].isin(content_freq_year)]

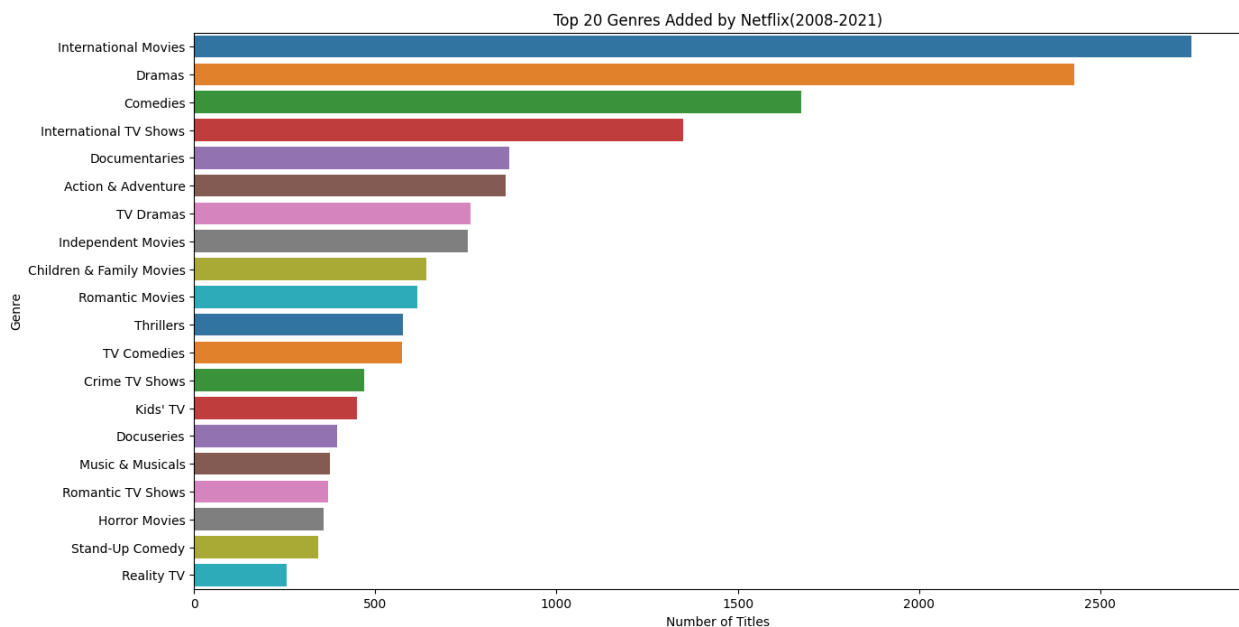
# Splitting "listed_in" values and exploding the DataFrame
n4['listed_in'] = n4['listed_in'].apply(lambda x: [item.strip() for
item in x.split(',')])
n4 = n4.explode('listed_in')

# Counting occurrences of each genre
genre_counts = n4['listed_in'].value_counts()

# Selecting the top 20 genres and sorting in descending order
top_20_genres = genre_counts.head(20).sort_values(ascending=False)

# Creating a horizontal bar plot with different colors for each bar
sns.barplot(x=top_20_genres.values, y=top_20_genres.index,
palette=sns.color_palette("tab10"))
)
# Adding labels and title
plt.xlabel('Number of Titles')
plt.ylabel('Genre')
plt.title('Top 20 Genres Added by Netflix(2008-2021)')
# Displaying the plot
plt.show()

```



```

#Top 20 Countries where Netflix Titles Added From(2008 -2021)

# Assuming "country" column contains information about the countries
plt.figure(figsize=(15, 8))

content_freq_year = [2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015,
2016, 2017, 2018, 2019, 2020, 2021]
n4 = n[n['ExtractedDate'].isin(content_freq_year)]

n4 = n4.dropna()
n4['SplitCountries'] = n4['country'].str.split(',').apply(lambda x:
[country.strip() for country in x])
n4 = n4.explode('SplitCountries')

# Counting occurrences of each country
country_counts = n4['SplitCountries'].value_counts()

# Selecting the top 20 countries and sorting in descending order
top_20_countries =
country_counts.head(20).sort_values(ascending=False)

# Custom color palette
custom_palette = sns.color_palette("husl", len(top_20_countries))

# Creating a horizontal bar plot with custom colors
sns.barplot(x=top_20_countries.values, y=top_20_countries.index,
palette=custom_palette)

# Adding labels and title
plt.xlabel('Number of Titles')
plt.ylabel('Country')
plt.title('Top 20 Countries Where Netflix Titles Were Added (2008-
2021)',loc='left')

# Displaying the plot
plt.show()

```

