

TASK 6

The screenshot shows a Jupyter Notebook with the following code and output:

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

data = pd.read_csv('c:\\Users\\User\\Downloads\\disney_plus_titles.csv')
data.head()
```

Output [1]:

show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description	
0	s1	Movie	A Spark Story	Jason Sterman, Leanne Dare	Aphthon Corbin, Louis Gonzales	NaN	September 24, 2021	2021	TV-PG	88 min	Documentary	Two Pixar filmmakers strive to bring their uni...
1	s2	Movie	Spooky Buddies	Robert Vince	Tucker Albrizzi, Diedrich Bader, Ameke Eks Mas...	United States, Canada	September 24, 2021	2011	G	93 min	Comedy, Fantasy, Kids	The puppies go on a spooky adventure through a...
2	s3	Movie	The Fault in Our Stars	Josh Boone	Shailene Woodley, Ansel Elgort, Laura Dern, Sa...	United States	September 24, 2021	2014	PG-13	127 min	Coming of Age, Drama, Romance	Hazel and Gus share a love that sweeps them on...
3	s4	TV Show	Dog: Impossible	NaN	Matt Beisner	United States	September 22, 2021	2019	TV-PG	2 Seasons	Animals & Nature, Docuseries, Family	Matt Beisner uses unique approaches to modify...
4	s5	TV Show	Spidey And His Amazing Friends	NaN	Benjamin Valic, Lily Sanfelippo, Jakari Fraser...	United States	September 22, 2021	2021	TV-Y	1 Season	Action-Adventure, Animation, Kids	Spidey teams up with pals to become The Spidey...

Output [2]:

```
data.tail()
```

Output [3]:

show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description	
1363	s1364	Movie	The Sword in the Stone	Wolfgang Reitherman	Sebastian Cabot, Karl Swenson, Rickie Sorensen...	United States	October 1, 2019	1963	G	80 min	Action-Adventure, Animation, Comedy	Merlin trains a young orphan who's destined to...

The screenshot shows a Jupyter Notebook with the following code and output:

```
data.tail()
```

Output [3]:

show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description	
1363	s1364	Movie	The Sword in the Stone	Wolfgang Reitherman	Sebastian Cabot, Karl Swenson, Rickie Sorensen...	United States	October 1, 2019	1963	G	80 min	Action-Adventure, Animation, Comedy	Merlin trains a young orphan who's destined to...
1364	s1365	Movie	Those Calloways	Norman Tokar	Brian Keith, Vera Miles, Brandon de Wilde, Wal...	United States	October 1, 2019	1965	PG	132 min	Animals & Nature, Drama, Family	A strong-willed family struggles to establish ...
1365	s1366	TV Show	Disney Kirby Buckets	NaN	Jacob Bertrand, Mekai Curtis, Cade Sutton, Oli...	United States	NaN	2014	TV-Y7	3 Seasons	Action-Adventure, Comedy, Coming of Age	Welcome to Kirby's world! It's rude and sketchy.
1366	s1367	TV Show	Disney Mech-X4	NaN	Nathaniel Potvin, Raymond Cham, Kamran Lucas, ...	Canada	NaN	2016	TV-Y7	2 Seasons	Action-Adventure, Comedy, Science Fiction	Ryan discovers his ability to control a giant ...
1367	s1368	TV Show	Imagination Movers	NaN	Rich Collins, Dave Poche, Scott Durbin, Scott ...	United States	NaN	2008	TV-Y	3 Seasons	Kids, Music	Rock out with the Imagination Movers, Disney J...

```
data.columns.values
```

Output [3]:

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

```
data.isna().sum()
```

Output [4]:

show_id	type	title	director
0	0	0	440

```
File Edit Selection View Go Run Terminal Help Python
TASK6.ipynb x
TASK6.ipynb > import pandas as pd
+ Code + Markdown | Run All | Clear All Outputs | Outline ... Python 3.12.4

title
director 440
cast 174
country 175
date_added 3
release_year 0
rating 2
duration 0
listed_in 0
description 0
dtype: int64

data.isnull()

[5] Python
...

```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	False	False	False	False	True	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False	False	False
3	False	False	False	True	False	False	False	False	False	False	False	False
4	False	False	False	True	False	False	False	False	False	False	False	False
...
1363	False	False	False	False	False	False	False	False	False	False	False	False
1364	False	False	False	False	False	False	False	False	False	False	False	False
1365	False	False	False	True	False	False	True	False	False	False	False	False
1366	False	False	False	True	False	False	True	False	False	False	False	False
1367	False	False	False	True	False	False	True	False	False	False	False	False

1368 rows x 12 columns

Spaces: 4 CRLF Cell 1 of 46

```
File Edit Selection View Go Run Terminal Help Python
TASK6.ipynb x
TASK6.ipynb > import pandas as pd
+ Code + Markdown | Run All | Clear All Outputs | Outline ... Python 3.12.4

df=df.fillna(value=0)
df

[6] Python
...

```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	s1	Movie	A Spark Story	Jason Stermann, Leanne Dare	Aphthon Corbin, Louis Gonzales	0	September 24, 2021	2021	TV-PG	88 min	Documentary	Two Pixar filmmakers strive to bring their uni...
1	s2	Movie	Spooky Buddies	Robert Vince	Tucker Albrizzi, Diedrich Bader, Ameke Eks Mas...	United States, Canada	September 24, 2021	2011	G	93 min	Comedy, Fantasy, Kids	The puppies go on a spooky adventure through a...
2	s3	Movie	The Fault in Our Stars	Josh Boone	Shailene Woodley, Ansel Elgort, Laura Dern, Sa...	United States	September 24, 2021	2014	PG-13	127 min	Coming of Age, Drama, Romance	Hazel and Gus share a love that sweeps them on...
3	s4	TV Show	Dog: Impossible	0	Matt Beisner	United States	September 22, 2021	2019	TV-PG	2 Seasons	Animals & Nature, Docuseries, Family	Matt Beisner uses unique approaches to modify...
4	s5	TV Show	Spidey And His Amazing Friends	0	Benjamin Valic, Lily Sanfelippo, Jakari Fraser...	United States	September 22, 2021	2021	TV-Y	1 Season	Action-Adventure, Animation, Kids	Spidey teams up with pals to become The Spidey...
...
1363	s1364	Movie	The Sword in the Stone	Wolfgang Reitherman	Sebastian Cabot, Karl Swenson, Rickie Sorensen...	United States	October 1, 2019	1963	G	80 min	Action-Adventure, Animation, Comedy	Merlin trains a young orphan who's destined to...
1364	s1365	Movie	Those Calloways	Norman Tokar	Brian Keith, Vera Miles, Brandon de Wilde, Wal...	United States	October 1, 2019	1965	PG	132 min	Animals & Nature, Drama, Family	A strong-willed family struggles to establish ...
1365	s1366	TV Show	Disney Kirby Buckets	0	Jacob Bertrand, Mekai Curtis, Cade Sutton, Oli...	United States	0	2014	TV-Y7	3 Seasons	Action-Adventure, Comedy, Coming of Age	Welcome to Kirby's world! It's rude and sketchy.
1366	s1367	TV Show	Disney Mech-X4	0	Nathaniel Potvin, Raymond Cham, Kamran Lucas, ...	Canada	0	2016	TV-Y7	2 Seasons	Action-Adventure, Comedy, Science Fiction	Ryan discovers his ability to control a giant ...
1367	s1368	TV Show	Imagination Movers	0	Rich Collins, Dave Poche, Scott Durbin, Scott ...	United States	0	2008	TV-Y	3 Seasons	Kids, Music	Rock out with the Imagination Movers, Disney J...

1368 rows x 12 columns

Spaces: 4 CRLF Cell 1 of 46

```
File Edit Selection View Go Run Terminal Help Python
TASK6.ipynb x
TASK6.ipynb > import pandas as pd
+ Code + Markdown | Run All | Clear All Outputs | Outline ... Python 3.12.4

1367 s1368 TV Show Imagination Movers 0 Rich Collins, Dave Poche, Scott Durbin, Scott ... United States 0 2008 TV-Y 3 Seasons Kids, Music Rock out with the Imagination Movers, Disney J...

1368 rows x 12 columns

# Count how many duplicate rows exist
duplicate_count = df.duplicated().sum()

duplicate_count

[7] Python
...
np.int64(0)

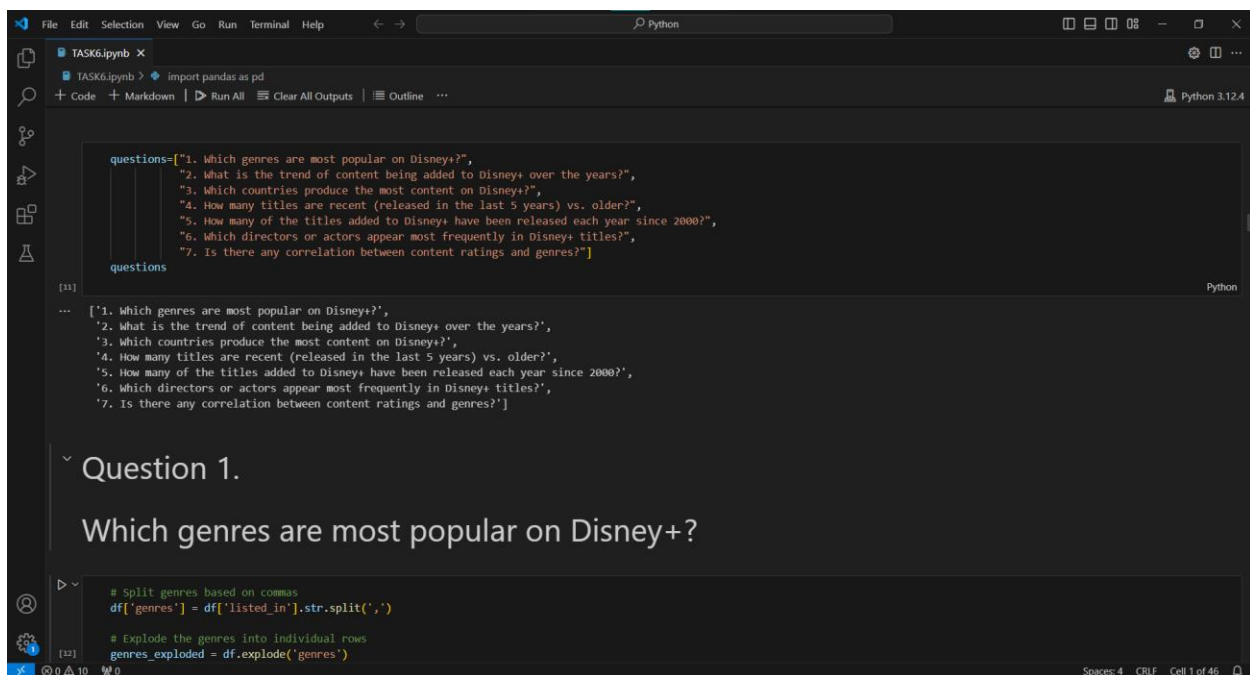
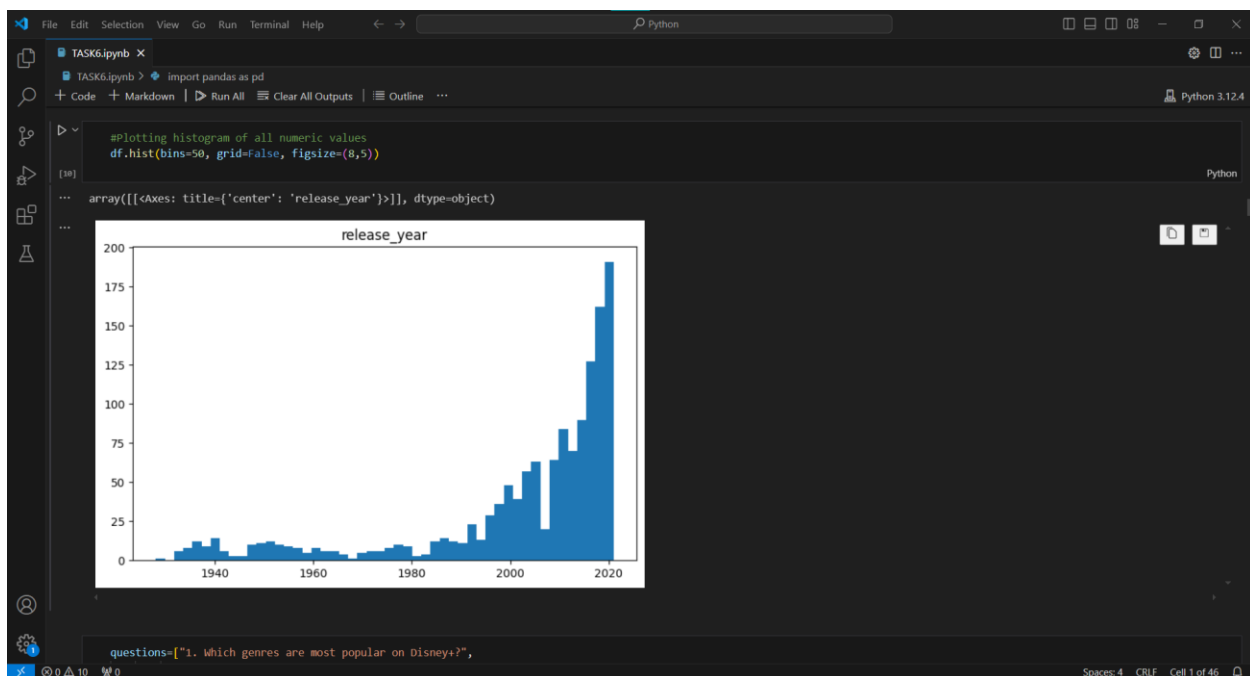
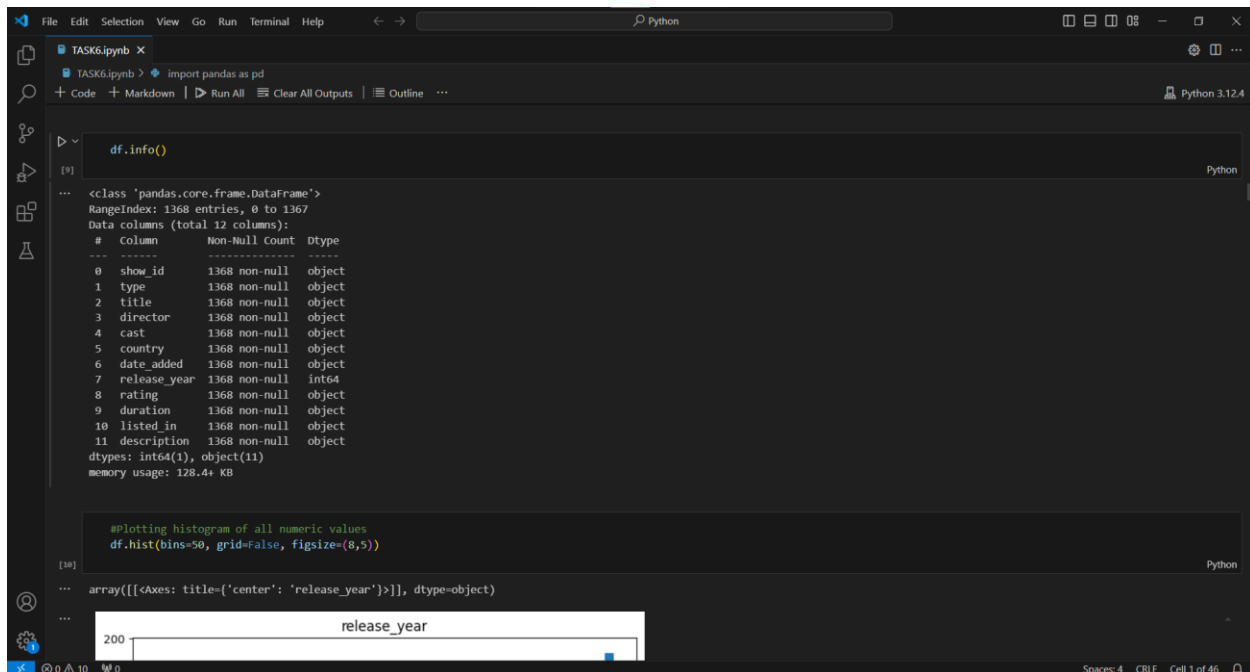
df.describe()

[8] Python
...

```

	release_year
count	1368.000000
mean	2002.348684
std	22.127559
min	1928.000000
25%	1998.000000
50%	2011.000000
75%	2018.000000
max	2021.000000

Spaces: 4 CRLF Cell 1 of 46



```
File Edit Selection View Go Run Terminal Help Python
TASK6.ipynb x
TASK6.ipynb > import pandas as pd
+ Code + Markdown | Run All | Clear All Outputs | Outline ... Python 3.12.4

# Split genres based on commas
df['genres'] = df['listed_in'].str.split(',')

# Explode the genres into individual rows
genres_exploded = df.explode('genres')

# Strip any extra whitespace in genre names
genres_exploded['genres'] = genres_exploded['genres'].str.strip()

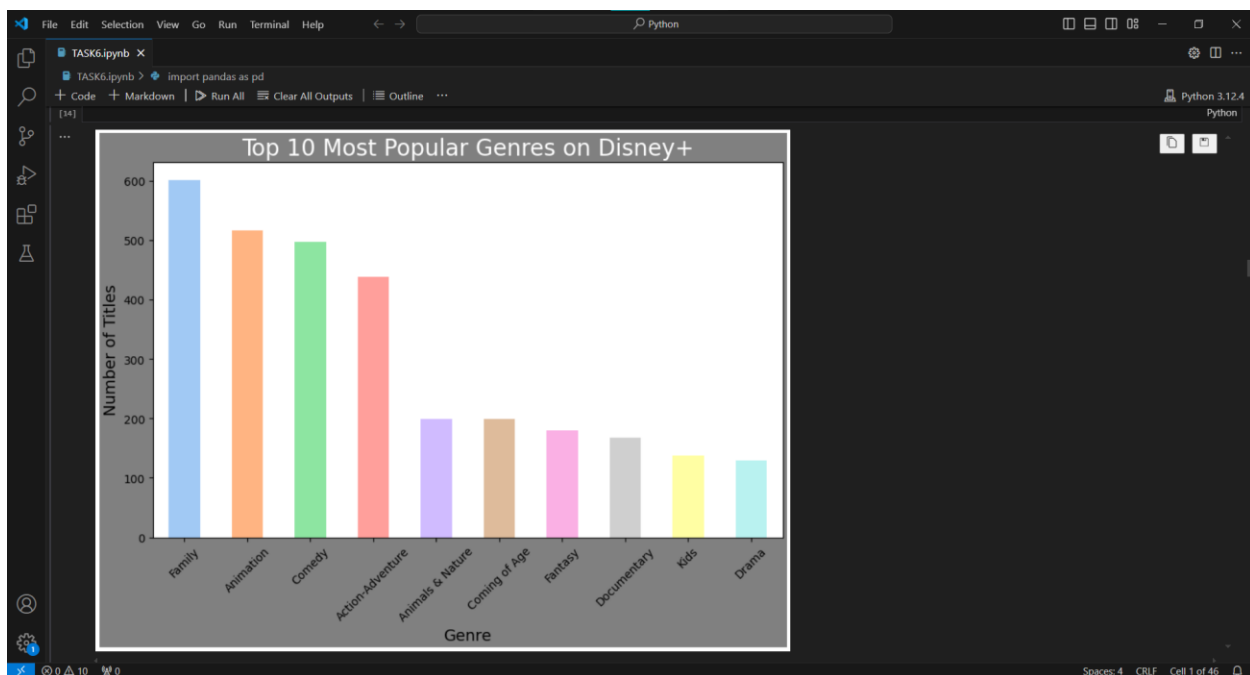
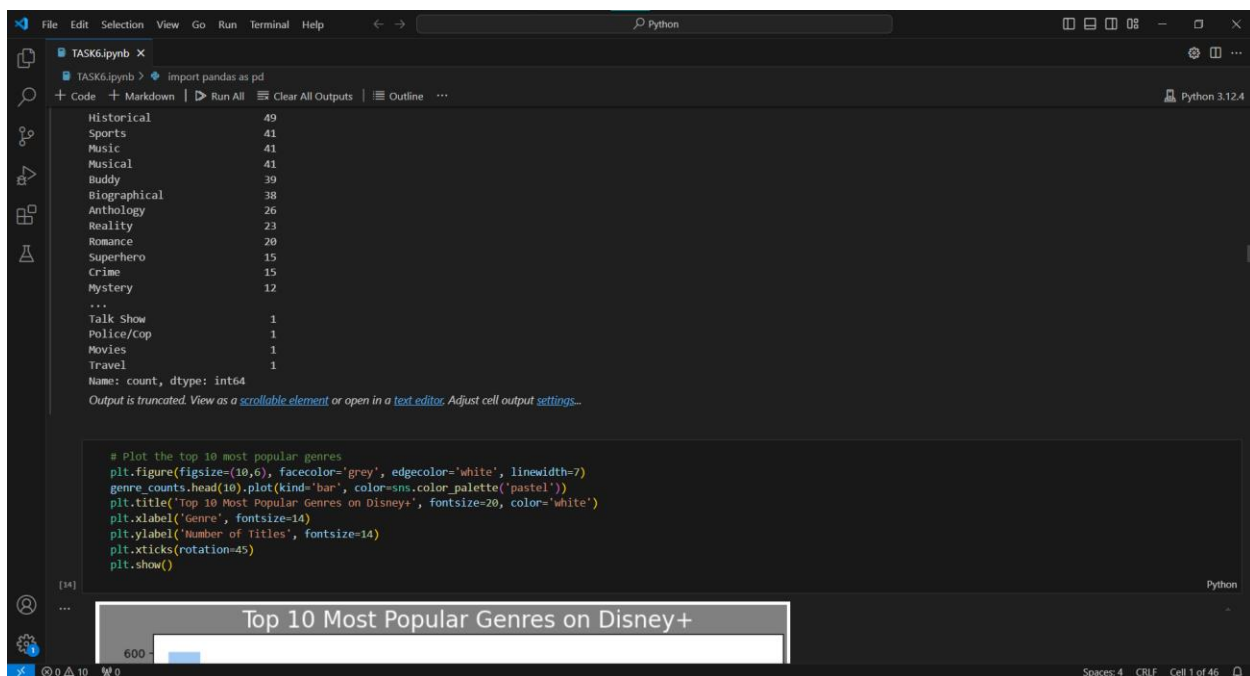
[12] Python

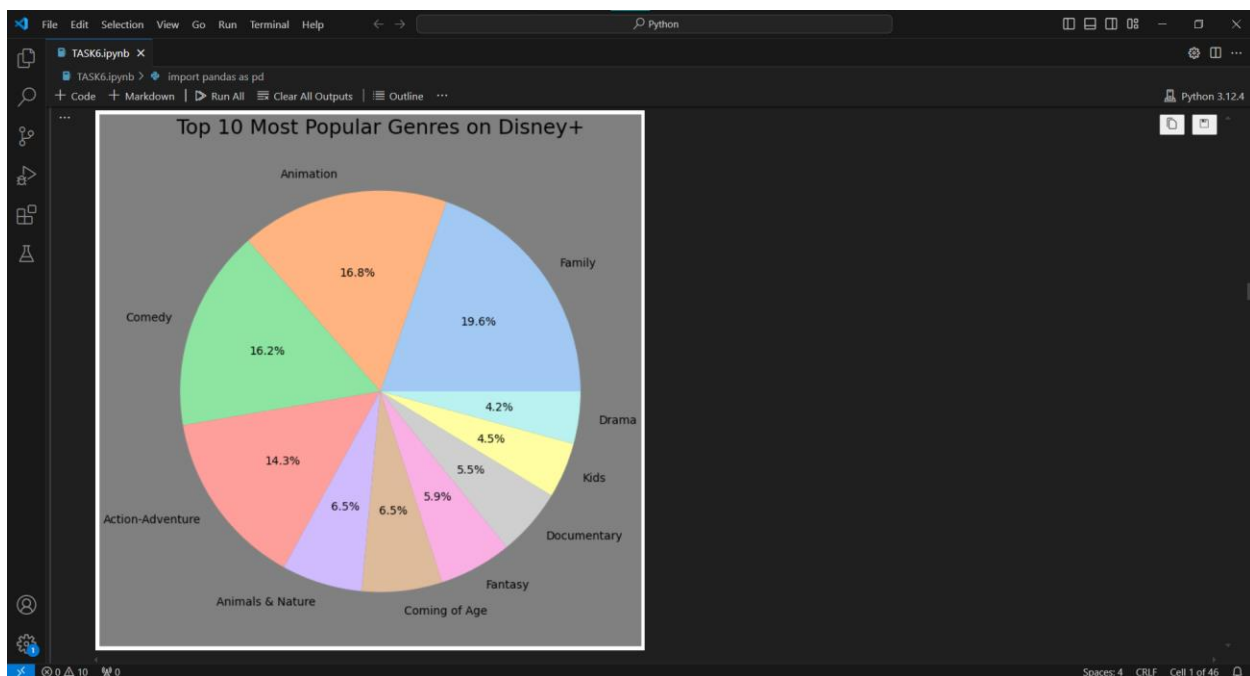
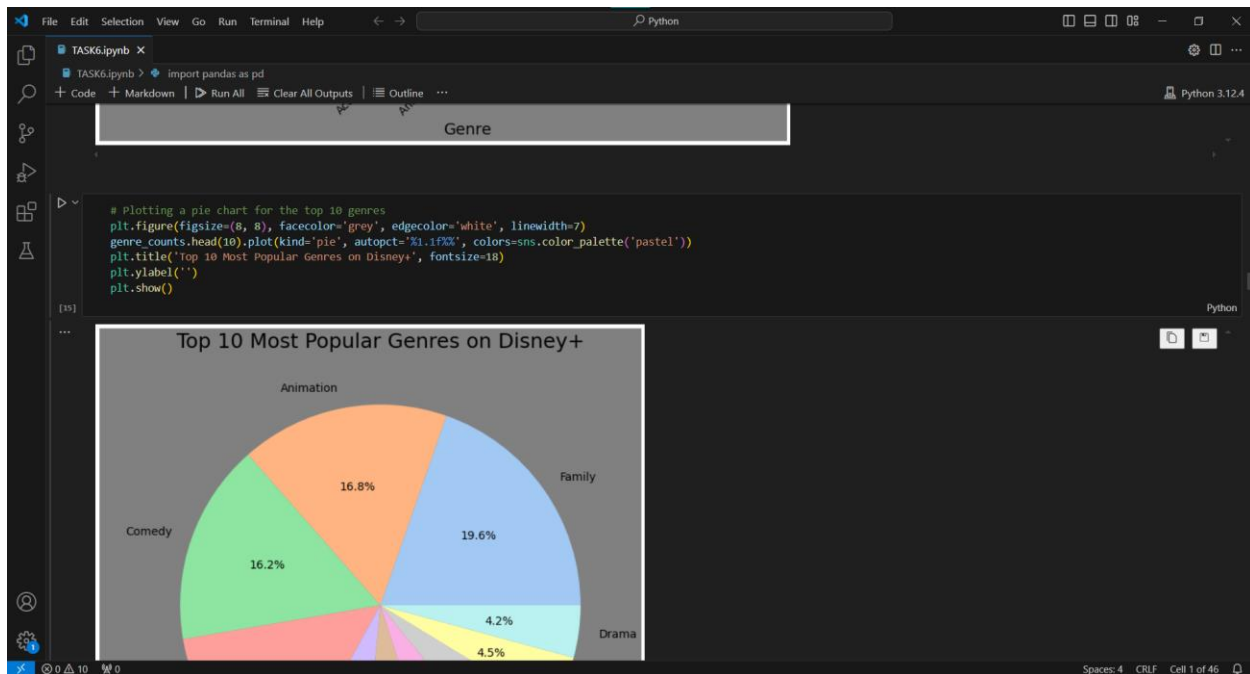
# Count the occurrences of each genre
genre_counts = genres_exploded['genres'].value_counts()

# Display the most popular genres
genre_counts

[13] Python

...
genres
Family                602
Animation              516
Comedy                 497
Action-Adventure       438
Animals & Nature        199
Coming of Age           199
Fantasy                180
Documentary            167
Kids                   137
Drama                  129
Docuseries             111
Science Fiction         84
Historical              49
Sports                  41
Name: count, dtype: int64
```





Python

TASK6.ipynb

TASK6.ipynb > import pandas as pd

+ Code + Markdown | Run All | Clear All Outputs | Outline

Python 3.12.4

- Family, Animation and comedy are the most prominent genres on Disney+, collectively comprising over 52% of their content offerings, highlighting the platform's focus on family-friendly, animated and comedy films and series.

Question 2.

What is the trend of content being added to Disney+ over the years?

```
# Convert date_added to datetime format
df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')

# Extract the year from 'date_added'
df['year_added'] = df['date_added'].dt.year

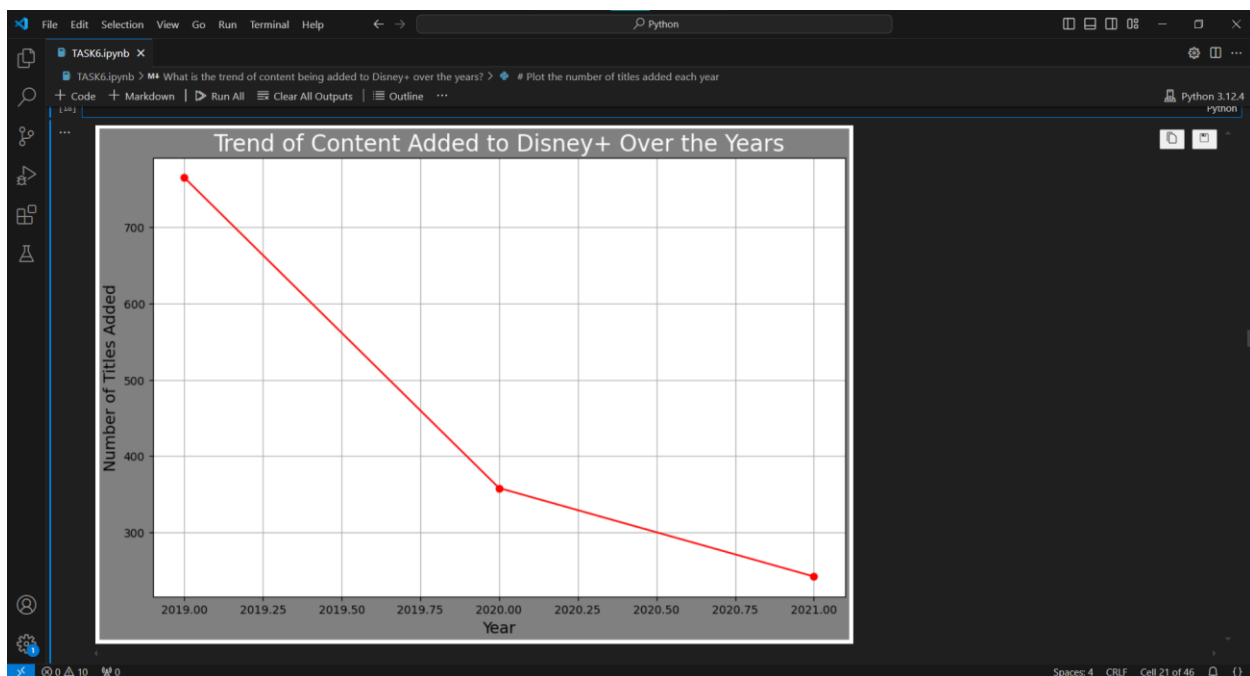
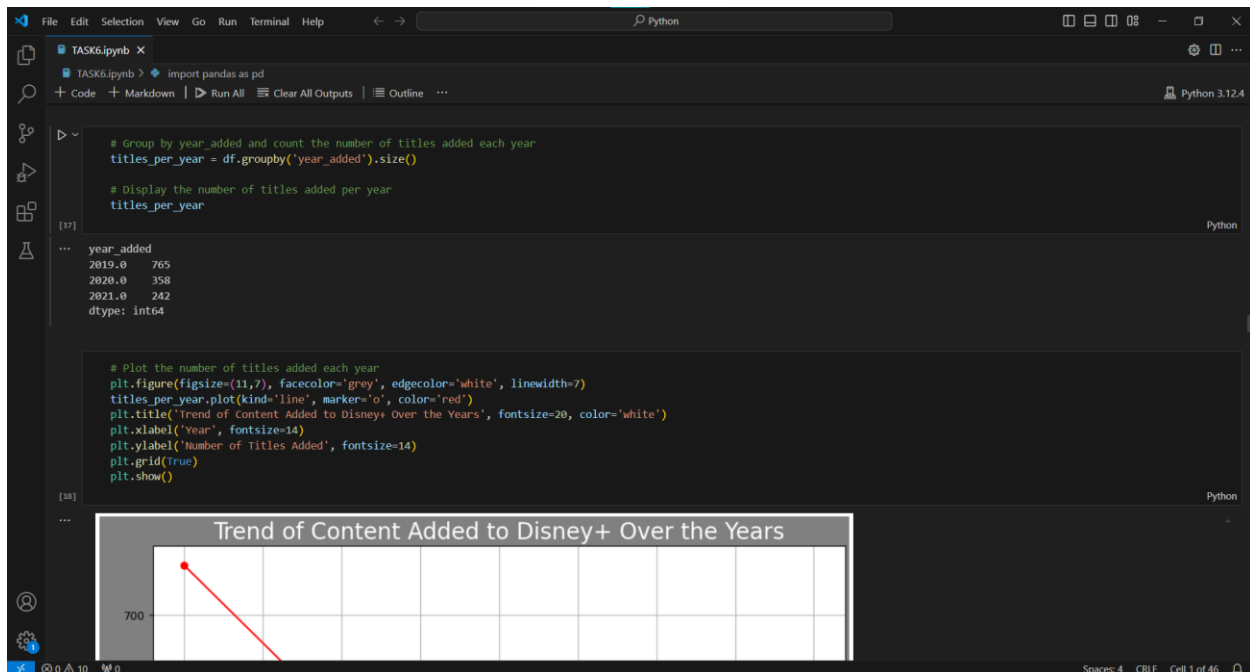
# Display the first few rows to check
df[['title', 'date_added', 'year_added']].head()
```

[16]

Python

	title	date_added	year_added
0	A Spark Story	2021-09-24	2021.0
1	Spooky Buddies	2021-09-24	2021.0
2	The Fault in Our Stars	2021-09-24	2021.0
3	Dog: Impossible	2021-09-22	2021.0
4	Spidey And His Amazing Friends	2021-09-22	2021.0

Spaces: 4 CRLF Cell 1 of 46



Python

```
# Split the country column where multiple countries are listed, separated by commas
df['countries'] = df['country'].str.split(',')

# Strip any extra whitespace
df['countries'] = df['countries'].apply(lambda x: [i.strip() for i in x] if isinstance(x, list) else x)

# Display the first few rows to check
df[['title', 'country', 'countries']].head()
```

[19]

	title	country	countries
0	A Spark Story	0	NaN
1	Spooky Buddies	United States, Canada	[United States, Canada]
2	The Fault in Our Stars	United States	[United States]
3	Dog: Impossible	United States	[United States]
4	Spidey And His Amazing Friends	United States	[United States]

Spaces: 4 CRLF Cell 21 of 46

File Edit Selection View Go Run Terminal Help Python

TASK6.ipynb X

TASK6.ipynb > What is the trend of content being added to Disney+ over the years? > Plot the number of titles added each year

+ Code + Markdown Run All Clear All Outputs Outline Python 3.12.4

- United States is the dominant country in terms of content production on Disney+, significantly outpacing other countries like the United Kingdom, Canada, and Australia, which also contribute to the platform but at much lower volumes.

Question 4.

How many titles are recent (released in the last 5 years) vs. older?

```
import datetime

# Get the current year
current_year = datetime.datetime.now().year

# Define a new column category to classify titles as recent or older
df['category'] = df['release_year'].apply(lambda x: 'Recent' if (current_year - x) <= 5 else 'older')

# Count the number of recent vs older titles
title_counts = df['category'].value_counts()

# Display the counts
title_counts
```

```
category
older    1080
Recent    288
Name: count, dtype: int64
```

Python

Spaces: 4 CRLF Cell 21 of 46

File Edit Selection View Go Run Terminal Help Python

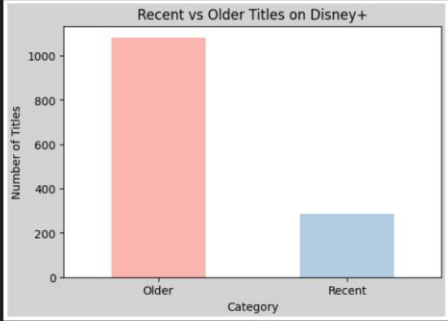
TASK6.ipynb X

TASK6.ipynb > What is the trend of content being added to Disney+ over the years? > Plot the number of titles added each year

+ Code + Markdown Run All Clear All Outputs Outline Python 3.12.4

```
# Plot a bar chart to show recent vs older titles
plt.figure(figsize=(6, 4), facecolor='lightgrey', edgecolor='white', linewidth=7)
title_counts.plot(kind='bar', color=sns.color_palette('Pastel1'))
plt.title('Recent vs Older Titles on Disney+')
plt.xlabel('Category')
plt.ylabel('Number of Titles')
plt.xticks(rotation=0)
plt.show()
```

Python



Spaces: 4 CRLF Cell 21 of 46

File Edit Selection View Go Run Terminal Help Python

TASK6.ipynb X

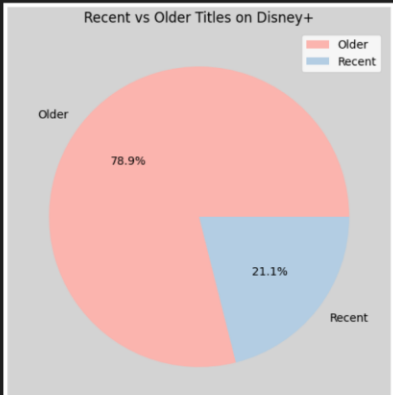
TASK6.ipynb > What is the trend of content being added to Disney+ over the years? > Plot the number of titles added each year

+ Code + Markdown Run All Clear All Outputs Outline Python 3.12.4

```
plt.figure(figsize=(6,6), facecolor='lightgrey', edgecolor='white', linewidth=7)
title_counts.plot(kind='pie', autopct='%1.1f%%', colors=sns.color_palette('Pastel1'))
plt.legend(['Older', 'Recent'])
plt.ylabel('')
plt.title('Recent vs Older Titles on Disney+')
```

Python

```
Text(0.5, 1.0, 'Recent vs Older Titles on Disney+')
```



Spaces: 4 CRLF Cell 21 of 46

File Edit Selection View Go Run Terminal Help Python

TASK6.ipynb X

TASK6.ipynb > What is the trend of content being added to Disney+ over the years? > Plot the number of titles added each year

+ Code + Markdown Run All Clear All Outputs Outline Python 3.12.4

- There are significantly more older titles, approximately 1000 (78.9% of all titles), compared to recent titles, around 250 (21.1%), available on Disney+.

Question 5.

How many of the titles added to Disney+ have been released each year since 2000?

```
# Filter the dataset to include only titles released after 2000
titles_after_2000 = df[df['release_year'] >= 2000]

# Group by release year and count the number of titles
titles_per_year = titles_after_2000.groupby('release_year').size()

# Reset the index for easier plotting
titles_per_year = titles_per_year.reset_index(name='count')

# Display the data
titles_per_year
```

[26]

	release_year	count
0	2000	26
1	2001	18
2	2002	21
3	2003	29
4	2004	28

Python

Spaces: 4 CRLF Cell 21 of 46

File Edit Selection View Go Run Terminal Help Python

TASK6.ipynb X

TASK6.ipynb > What is the trend of content being added to Disney+ over the years? > Plot the number of titles added each year

+ Code + Markdown Run All Clear All Outputs Outline Python 3.12.4

	release_year	count
0	2000	26
1	2001	18
2	2002	21
3	2003	29
4	2004	28
5	2005	32
6	2006	31
7	2007	20
8	2008	31
9	2009	33
10	2010	36
11	2011	48
12	2012	39
13	2013	31
14	2014	47
15	2015	43
16	2016	60
17	2017	67
18	2018	65
19	2019	97
20	2020	109
21	2021	82

Python

Spaces: 4 CRLF Cell 21 of 46

File Edit Selection View Go Run Terminal Help Python

TASK6.ipynb X

TASK6.ipynb > What is the trend of content being added to Disney+ over the years? > Plot the number of titles added each year

+ Code + Markdown Run All Clear All Outputs Outline Python 3.12.4

```
# Create a bar plot using Seaborn
plt.figure(figsize=(10, 6), facecolor='lightgrey', edgecolor='white', linewidth=7)
sns.barplot(x='release_year', y='count', data=titles_per_year, palette='Blues_d')


# Set plot title and labels
plt.title('Number of Titles Released Per Year Since 2000', fontsize=20)
plt.xlabel('Release Year', fontsize=14)
plt.ylabel('Number of Titles', fontsize=14)
plt.xticks(rotation=45)
plt.show()
```

[27]

FutureWarning:

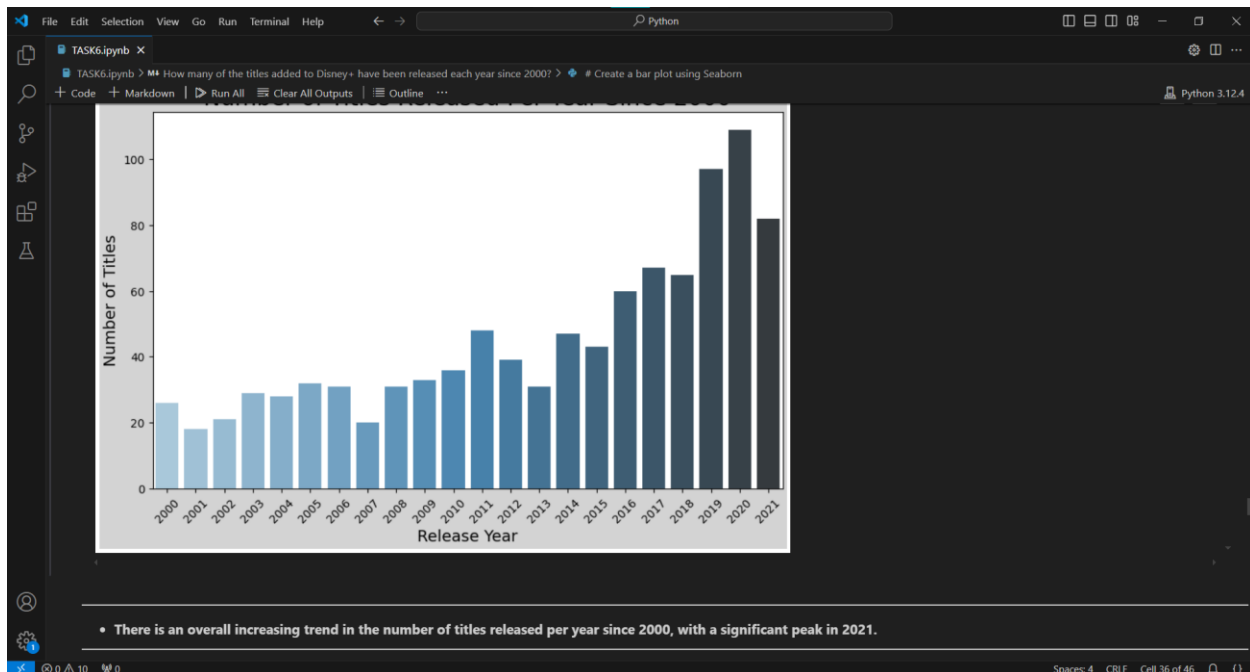
Passing 'palette' without assigning 'hue' is deprecated and will be removed in v0.14.0. Assign the 'x' variable to 'hue' and set 'legend=False' for the same effect.

```
sns.barplot(x='release_year', y='count', data=titles_per_year, palette='Blues_d')
```



Python

Spaces: 4 CRLF Cell 21 of 46



File Edit Selection View Go Run Terminal Help Python

TASK6.ipynb x

TASK6.ipynb > How many of the titles added to Disney+ have been released each year since 2000? > # Create a bar plot using Seaborn

+ Code + Markdown | Run All | Clear All Outputs | Outline ...

Python 3.12.4

Question 6.

Which directors or actors appear most frequently in Disney+ titles?

```
# Split director names and actor names into lists
df['director_list'] = df['director'].str.split(',')
df['cast_list'] = df['cast'].str.split(',')

# Cleaning up extra spaces
df['director_list'] = df['director_list'].apply(lambda x: [i.strip() for i in x] if isinstance(x, list) else x)
df['cast_list'] = df['cast_list'].apply(lambda x: [i.strip() for i in x] if isinstance(x, list) else x)

# Display the result to verify
print(df['director_list'].head())
print('\n')
print(df['cast_list'].head())
```

[28]

```
0    [Jason Stermann, Leanne Dare]
1    [Robert Vince]
2    [Josh Boone]
3    NaN
4    NaN
Name: director_list, dtype: object
```

```
0    [Aphron Corbin, Louis Gonzales]
1    [Tucker Albrizzi, Diedrich Bader, Aneko Eke Ma...
2    [Shallene Woodley, Ansel Elgort, Laura Dern, S...
3    [Matt Beisner]
4    [Benjamin Valic, Lily Sanfelippo, Jakari Frase...
```

Python

Spaces: 4 CRLF Cell 36 of 46

File Edit Selection View Go Run Terminal Help Python

TASK6.ipynb x

TASK6.ipynb > How many of the titles added to Disney+ have been released each year since 2000? > # Create a bar plot using Seaborn

+ Code + Markdown | Run All | Clear All Outputs | Outline ...

Python 3.12.4

```
3    [Matt Beisner]
4    [Benjamin Valic, Lily Sanfelippo, Jakari Frase...
Name: cast_list, dtype: object
```

```
# Exploding lists into rows
directors = df.explode('director_list')
actors = df.explode('cast_list')

# Counting occurrences
director_counts = directors['director_list'].value_counts()
actor_counts = actors['cast_list'].value_counts()

print(director_counts, '\n')
print(actor_counts)
```

[29]

```
director_list
Jack Hannah      17
John Lasseter    16
Paul Hoen        16
Wilfred Jackson  16
Clyde Geronimi   13
..
Leanne Dare      1
Jason Stermann   1
Dave Michener    1
Victor Gonzalez  1
Lewis Foster     1
Name: count, Length: 603, dtype: int64
```

```
cast_list
Jim Cummings      32
Walt Disney        18
Larry the Cable Guy 17
Keith Ferguson     16
```

Python

Spaces: 4 CRLF Cell 36 of 46

```
File Edit Selection View Go Run Terminal Help Python
TASK6.ipynb x
TASK6.ipynb > How many of the titles added to Disney+ have been released each year since 2000? > # Create a bar plot using Seaborn
+ Code + Markdown | Run All | Clear All Outputs | Outline ... Python 3.12.4

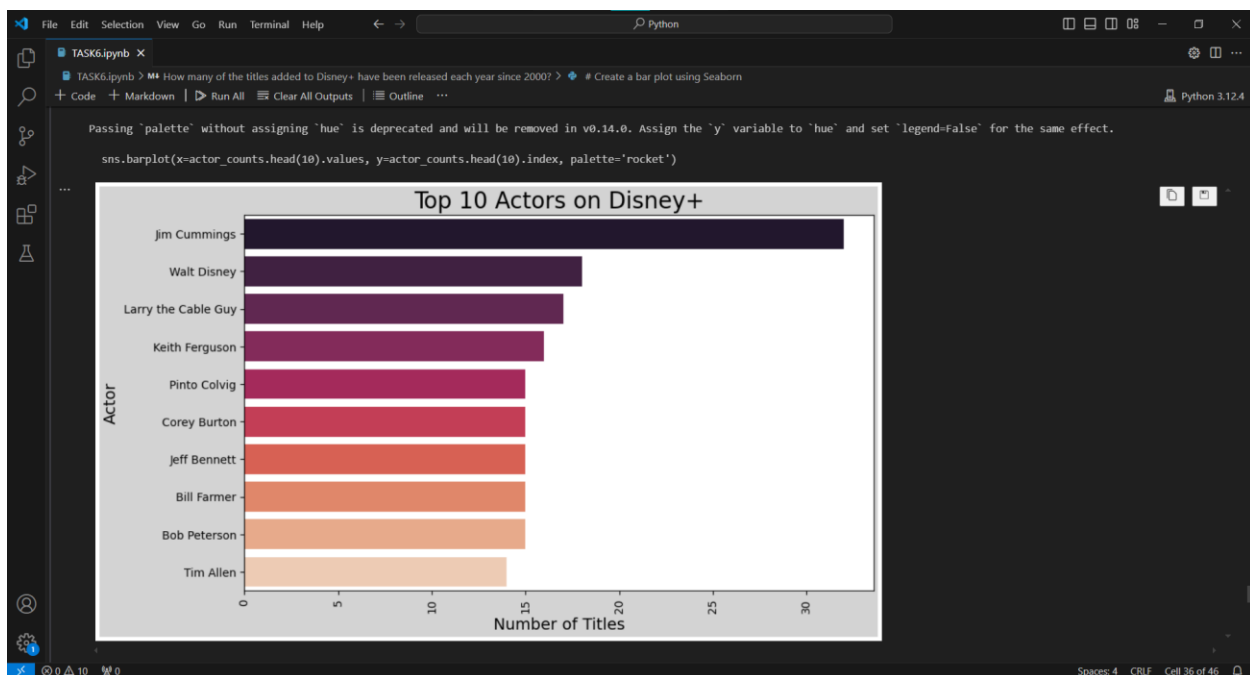
Larry the Cable Guy 17
Keith Ferguson 16
Pinto Colvig 15
..
Shailene Woodley 1
Ansel Elgort 1
Laura Dern 1
Sam Trammell 1
Nat Wolff 1
Name: count, length: 3601, dtype: int64

# Plotting the top 10 directors
plt.figure(figsize=(10, 6), facecolor='lightgrey', edgecolor='white', linewidth=7)
sns.barplot(x=director_counts.head(10).values, y=director_counts.head(10).index, palette='Blues_d')
plt.title('Top 10 Directors on Disney+', fontsize=20)
plt.xlabel('Number of Titles', fontsize=14)
plt.ylabel('Director', fontsize=14)
plt.xticks(rotation=90)
plt.show()

# Plotting the top 10 actors
plt.figure(figsize=(10, 6), facecolor='lightgrey', edgecolor='white', linewidth=7)
sns.barplot(x=actor_counts.head(10).values, y=actor_counts.head(10).index, palette='rocket')
plt.title('Top 10 Actors on Disney+', fontsize=20)
plt.xlabel('Number of Titles', fontsize=14)
plt.ylabel('Actor', fontsize=14)
plt.xticks(rotation=90)
plt.show()

C:\Users\User\AppData\Local\Temp\ipykernel_7436\1479956810.py:3: FutureWarning:
Passing 'palette' without assigning 'hue' is deprecated and will be removed in v0.14.0. Assign the 'y' variable to 'hue' and set 'legend=False' for the same effect.

sns.barplot(x=director_counts.head(10).values, y=director_counts.head(10).index, palette='Blues_d')
```



File Edit Selection View Go Run Terminal Help Python

TASK6.ipynb x

TASK6.ipynb > How many of the titles added to Disney+ have been released each year since 2000? > # Create a bar plot using Seaborn

+ Code + Markdown | Run All | Clear All Outputs | Outline ... Python 3.12.4

- Jack Hannah, John Lasseter, Paul Hoen, and Wilfred Jackson are among the top directors with most number of titles on Disney+.
- Jim Cummings leads as the actor with the highest number of titles on Disney+, significantly outpacing other actors like Walt Disney and Tim Allen, highlighting his prominent presence on the platform.

Question 7.

Is there any correlation between content ratings and genres?

```
from scipy.stats import chi2_contingency

# Separated and exploded genres in the first question

# Cross tabulation of genres and ratings
contingency_table = pd.crosstab(genres_exploded['genres'], genres_exploded['rating'])

# Chi-square test of independence
chi2, p, dof, expected = chi2_contingency(contingency_table)

print(f"Chi-square test results:\nchi2 statistic: {chi2}, p-value: {p}")
```

[33]

... Chi-square test results:
chi2 statistic: 3021.324469148122, p-value: 0.0

```
contingency_table.plot(kind='bar', stacked=True, figsize=(12, 6))
```

Python

Spaces: 4 CRLF Cell 36 of 46

