

Make Up Session Assignment

Name: Prajwal Bhiku Waghmare

Div: CS2

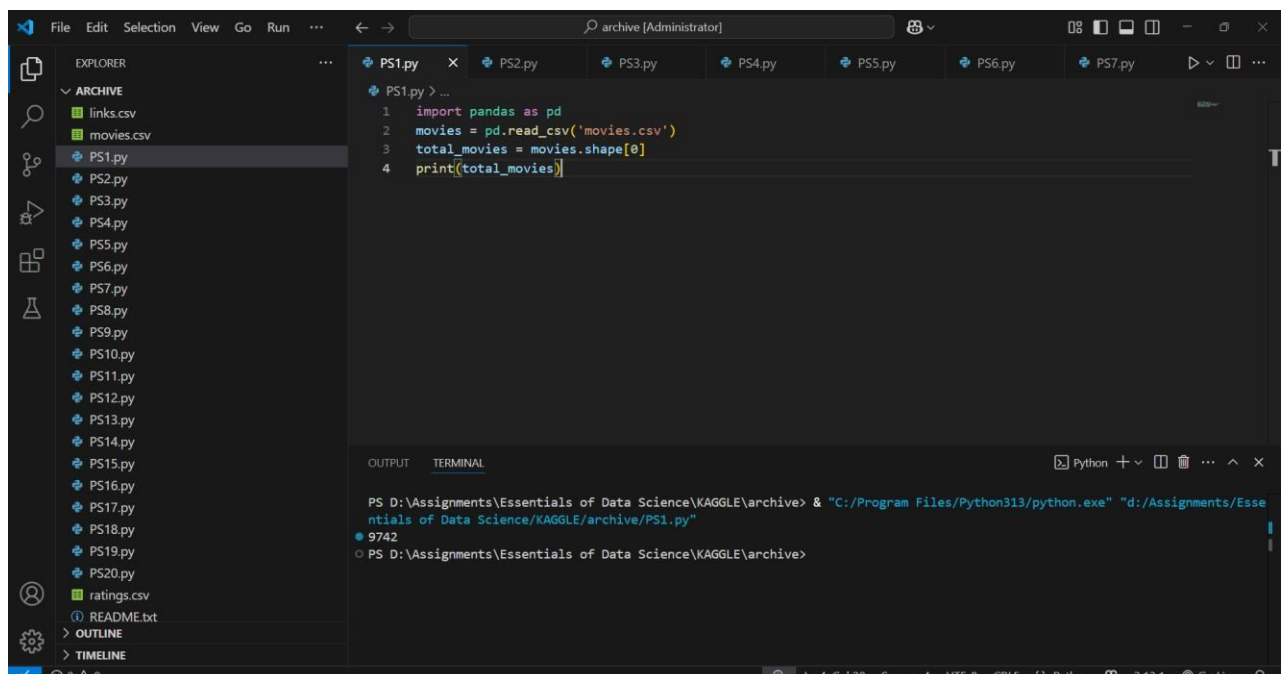
Roll no: 69

PRN: 202401040095

URL:

<https://www.kaggle.com/datasets/shubhammehta21/movie-lens-small-latest-dataset>

- 1) Count the total number of movies in the dataset.



The screenshot shows a Jupyter Notebook interface with a file explorer on the left and a code editor on the right. The file explorer shows a folder named 'ARCHIVE' containing several files, including 'links.csv', 'movies.csv', and 'ratings.csv'. The code editor displays a Python script in a file named 'PS1.py'. The script imports pandas, reads the 'movies.csv' file, and prints the total number of movies. The output of the script is shown in the terminal at the bottom, indicating that there are 9742 movies in the dataset.

```
1 import pandas as pd
2 movies = pd.read_csv('movies.csv')
3 total_movies = movies.shape[0]
4 print(total_movies)
```

```
PS D:\Assignments\Essentials of Data Science\KAGGLE\archive> "C:/Program Files/Python313/python.exe" "d:/Assignments/Essentials of Data Science/KAGGLE/archive/PS1.py"
9742
PS D:\Assignments\Essentials of Data Science\KAGGLE\archive>
```

- 2) Determine the number of unique genres spanning all movies.

The screenshot shows a VS Code editor window with a file explorer on the left and a code editor on the right. The file explorer shows a folder named 'archive' containing files: links.csv, movies.csv, PS1.py, PS2.py, ratings.csv, README.txt, and tags.csv. The code editor shows the contents of PS2.py, which is a Python script that imports pandas, reads a CSV file named 'movies.csv', splits the 'genres' column by the pipe character, explodes the resulting series, and prints the length of the unique genres. The terminal at the bottom shows the command to run the script and the output, which is the number 28.

```
PS D:\Assignments\Essentials of Data Science\KAGGLE\archive> "C:/Program Files/Python313/python.exe" "d:/Assignments/Essentials of Data Science/KAGGLE/archive/PS2.py"
28
PS D:\Assignments\Essentials of Data Science\KAGGLE\archive>
```

3) Compute how many times each genre appears by splitting and exploding the genre strings.

The screenshot shows a VS Code editor window with a file explorer on the left and a code editor on the right. The file explorer shows the same 'archive' folder as the previous screenshot, but now it includes PS3.py. The code editor shows the contents of PS3.py, which is a Python script that imports pandas, reads a CSV file named 'movies.csv', splits the 'genres' column by the pipe character, explodes the resulting series, and prints the value counts of the genres. The terminal at the bottom shows the command to run the script and the output, which is a list of genres and their counts.

```
PS D:\Assignments\Essentials of Data Science\KAGGLE\archive> "C:/Program Files/Python313/python.exe" "d:/Assignments/Essentials of Data Science/KAGGLE/archive/PS3.py"
genres
Drama          4361
Comedy          3756
Thriller        1894
Action          1828
Romance         1596
Adventure       1263
Crime           1199
Sci-Fi          988
Horror           978
Fantasy          779
Children         664
Animation        611
Mystery          573
Documentary      448
War              382
Musical          334
Western          167
IMAX             158
Film-Noir        87
(no genres listed) 34
Name: count, dtype: int64
```

4) Identify the five most frequent genres in the dataset.

The screenshot shows a Visual Studio Code editor window with a file explorer on the left containing files like links.csv, movies.csv, PS1.py, PS2.py, PS3.py, PS4.py, ratings.csv, README.txt, and tags.csv. The main editor displays PS4.py with the following code:

```
1 import pandas as pd
2 movies = pd.read_csv('movies.csv')
3 top5_genres = movies['genres'].str.split('|').explode().value_counts().nlargest(5)
4 print(top5_genres)
```

The output terminal shows the result of running the code:

```
PS D:\Assignments\Essentials of Data Science\KAGGLE\archive> "C:/Program Files/Python313/python.exe" "d:/Assignments/Essentials of Data Science/KAGGLE/archive/PS4.py"
genres
Drama      4361
Comedy     3756
Thriller   1894
Action     1828
Romance    1596
Name: count, dtype: int64
```

5) Extract the release year from the movie titles.

The screenshot shows a Visual Studio Code editor window with a file explorer on the left containing files like links.csv, movies.csv, PS1.py, PS2.py, PS3.py, PS4.py, PS5.py, ratings.csv, README.txt, and tags.csv. The main editor displays PS5.py with the following code:

```
1 import pandas as pd
2 movies = pd.read_csv('movies.csv')
3 movies['ReleaseYear'] = movies['title'].str.extract(r'\((\d{4})\)')
4 print(movies[['title', 'ReleaseYear']].head())
```

The output terminal shows the result of running the code:

```
PS D:\Assignments\Essentials of Data Science\KAGGLE\archive> "C:/Program Files/Python313/python.exe" "d:/Assignments/Essentials of Data Science/KAGGLE/archive/PS5.py"
  title ReleaseYear
0  Toy Story (1995)    1995
1   Jumanji (1995)    1995
2  Grumpier Old Men (1995)  1995
3  Waiting to Exhale (1995)  1995
4  Father of the Bride Part II (1995)  1995
```

6) Filter movies that have titles starting with the letter “A”.

```

1 import pandas as pd
2 movies = pd.read_csv('movies.csv')
3 movies_starting_A = movies[movies['title'].str.lower().str.startswith('a')]
4 print(movies_starting_A)

```

OUTPUT

```

PS D:\Assignments\Essentials of Data Science\KAGGLE\archive> "C:/Program Files/Python313/python.exe" "d:/Assignments/Essentials of Data Science/KAGGLE/archive/PS11.py"

```

movieId	title	genres
10	American President, The (1995)	Comedy Drama Romance
18	Ace Ventura: When Nature Calls (1995)	Comedy
22	Assassins (1995)	Action Crime Thriller
74	Antonia's Line (Antonia) (1995)	Comedy Drama
76	Angels and Insects (1995)	Drama Romance
...
9699	A Quiet Place (2018)	Drama Horror Thriller
9700	Alpha (2018)	Adventure Thriller
9713	Ant-Man and the Wasp (2018)	Action Adventure Comedy Fantasy Sci-Fi
9733	anohana: The Flower We Saw That Day - The Movie (2016)	Animation Drama
9741	Andrew Dice Clay: Dice Rules (1991)	Comedy

[551 rows x 3 columns]

7) Derive each movie's decade (based on the release year) and show the distribution.

```

1 import pandas as pd
2 movies = pd.read_csv('movies.csv')
3 movies['ReleaseYear'] = movies['title'].str.extract(r'\(((\d{4}))\)').astype(float)
4 movies['Decade'] = (movies['ReleaseYear'] // 10) * 10
5 decade_distribution = movies['Decade'].value_counts().sort_index()
6 print(decade_distribution)

```

OUTPUT

```

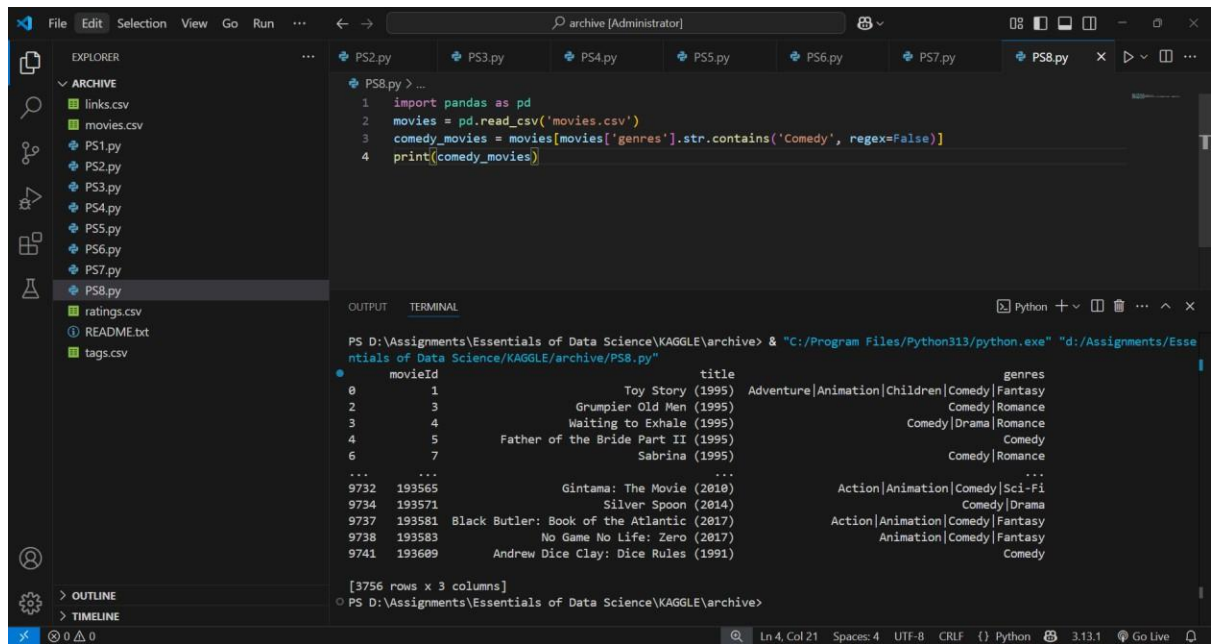
PS D:\Assignments\Essentials of Data Science\KAGGLE\archive> "C:/Program Files/Python313/python.exe" "d:/Assignments/Essentials of Data Science/KAGGLE/archive/PS7.py"

```

Decade	count
1900.0	3
1910.0	7
1920.0	37
1930.0	136
1940.0	197
1950.0	279
1960.0	401
1970.0	500
1980.0	1177
1990.0	2212
2000.0	2849
2010.0	1931

Name: count, dtype: int64

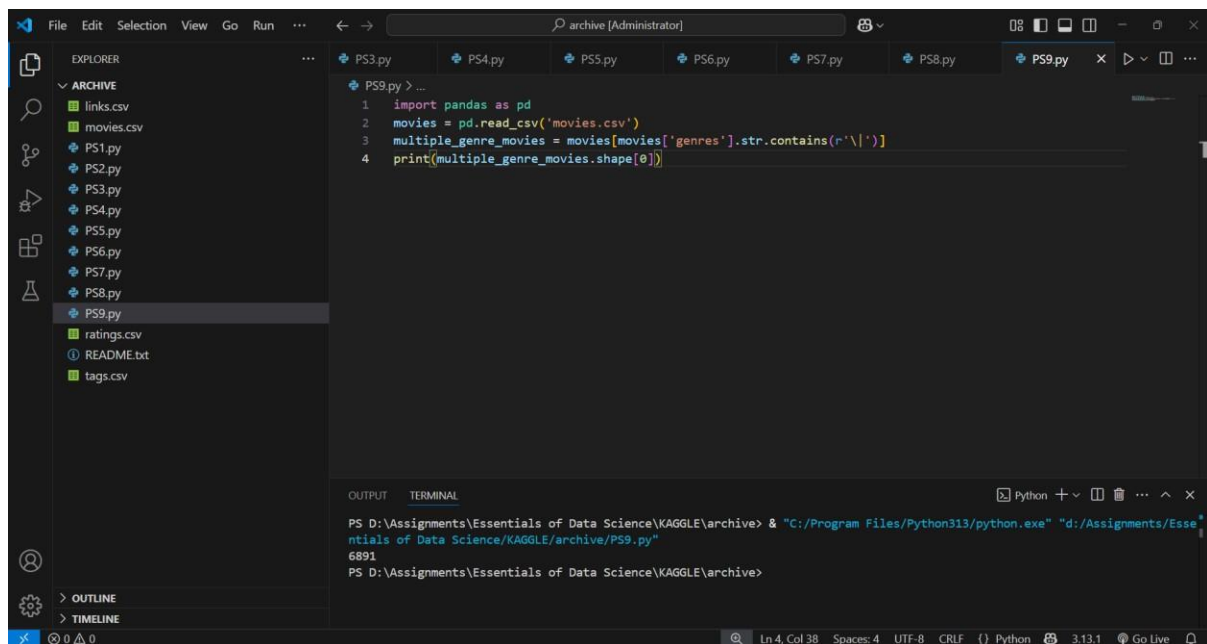
8) Filter out the movies that belong to the "Comedy" genre.



```
1 import pandas as pd
2 movies = pd.read_csv('movies.csv')
3 comedy_movies = movies[movies['genres'].str.contains('Comedy', regex=False)]
4 print(comedy_movies)
```

movieId	title	genres
0	Toy Story (1995)	Adventure Animation Children Comedy Fantasy
2	Grumpier Old Men (1995)	Comedy Romance
3	Waiting to Exhale (1995)	Comedy Drama Romance
4	Father of the Bride Part II (1995)	Comedy
6	Sabrina (1995)	Comedy Romance
...
9732	Gintama: The Movie (2010)	Action Animation Comedy Sci-Fi
9734	Silver Spoon (2014)	Comedy Drama
9737	Black Butler: Book of the Atlantic (2017)	Action Animation Comedy Fantasy
9738	No Game No Life: Zero (2017)	Animation Comedy Fantasy
9741	Andrew Dice Clay: Dice Rules (1991)	Comedy

9) Count the number of movies that have more than one genre listed.



```
1 import pandas as pd
2 movies = pd.read_csv('movies.csv')
3 multiple_genre_movies = movies[movies['genres'].str.contains(r'\|')]
4 print(multiple_genre_movies.shape[0])
```

6891

10) Identify the movie with the longest title.

The image shows a Visual Studio Code editor window with the following components:

- Explorer (Left Panel):** Displays a file tree for a project named "ARCHIVE". The files listed are: `links.csv`, `movies.csv`, `PS1.py`, `PS2.py`, `PS3.py`, `PS4.py`, `PS5.py`, `PS6.py`, `PS7.py`, `PS8.py`, `PS9.py`, `PS10.py` (selected), `ratings.csv`, `README.txt`, and `tags.csv`.
- Editor (Main Area):** Shows the content of `PS10.py`. The code is as follows:

```
1 import pandas as pd
2 movies = pd.read_csv('movies.csv')
3 movies['title_length'] = movies['title'].str.len()
4 longest_title_movie = movies.loc[movies['title_length'].idxmax()]
5 print(longest_title_movie)
```
- OUTPUT/TERMINAL (Bottom Panel):** Displays the output of the script. The output is a JSON-like representation of a movie record:

```
ntials of Data Science/KAGGLE/archive/PS10.py"
movieId      95165
title      Dragon Ball Z the Movie: The World's Strongest...
genres      Action|Adventure|Animation|Sci-Fi|Thriller
title_length      158
Name: 7905, dtype: object
PS D:\Assignments\Essentials of Data Science\KAGGLE\archive>
```
- Status Bar (Bottom):** Shows the current cursor position as "Ln 5, Col 27", the file encoding as "UTF-8", and the Python version as "3.13.1".