



ET

Eng. Tomas



Forwarded

Yesterday



IMDbMovies.csv

2.9 MB • CSV



4:44 PM

مساء الخير يا بشمهندس رفعت

4:44 PM

دا الداتا سيت بتاعت بروجيكت ال ML

Edited 4:45 PM

Eng. Tomas

مساء الخير يا بشمهندس رفعت

مساء النور يا بشمهندس

4:45 PM ✓✓

بص يا بشمهندس انا الداتا دي عايزك تعملها
preprocessing و EDA و تطلعلي شوية insights
كدا علي شكل أسئلة مثلا

4:51 PM

تمام يا بشمهندس اديني لحظات بس هبص علي الداتا
و اخذ فكره عنها و هرد عليك

4:54 PM ✓✓

تمام

4:56 PM

تمام يا بشمهندس انا بصيت علي الداتا و ان شاء الله
الداتا هتكون جاهز بكرة الصبح و التكلفة هتكون 400
جنيه

5:06 PM



Message





ET

Eng. Tomas



insights طلعي شوية Yesterday reprocessing

كدا علي شكل أسئلته مثلا

4:51 PM

تمام يا بشمهندس اديني لحظات بس هبص علي الداتا
و اخذ فكره عنها و هرد عليك

4:54 PM ✓✓

تمام

4:56 PM

تمام يا بشمهندس انا بصيت علي الداتا و ان شاء الله
الداتا هتكون جاهز بكرا الصبح و التكلفة هتكون 400
جنيه

5:06 PM ✓✓

تمام ماشي

5:08 PM



Today

صباح الخير يا بشمهندس

9:00 AM ✓✓



IMDB movies (Data
Preprocessing).ipynb

855 kB • BIN

9:00 AM ✓✓



دي الداتا بعد ال preprocessing و عملتك 15
ذال تقدر تطلع منهم insights مهمه من الداتا

9:02 AM ✓✓



Message





ET

Eng. Tomas



دي الداتا بعد ال preprocessing و عملتك 15
سؤال تقدر تطلع منهم insights مهمه من الداتا

9:02 AM ✓✓

تسلم ايدك يا بشمهندس بحد شغل عالي

9:09 AM



دقيقة و هحول لحضرتك

9:09 AM

Eng. Tomas

دقيقة و هحول لحضرتك

براحتك

9:10 AM ✓✓


400 EGP

Transfer Amount



Send Money



From

THOMAS AWAD MAHROUS AWAD
ALLAH-YOUT

thomas_tito@instapay



To: Instapay

REFAAT E*** E**** A***** ☆

01272360822

9:11 AM

تمام شكراً يا بشمهندس

9:11 AM ✓✓



Message



FileEditSelectionViewGoRunTerminalHelp

Search

IMDb movies (Data Preprocessing).ipynb

C:\Users> refaa> Downloads> IMDb movies (Data Preprocessing).ipynb> Load the Data

Generate+ Code+ MarkdownRun AllRestartClear All OutputsGo ToJupyter VariablesOutline

Python 3.13.7

Import Library

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

20.2s

Python

Load the Data

```
df = pd.read_csv(r"C:\Users\refaa\OneDrive\Desktop\IMDbMovies.csv")
df
```

0.3s

Python

| | Title | Summary | Director | Writer | Main Genres | Motion Picture Rating | Runtime | Release Year | Rating | Number of Ratings | Budget | Gross in US & Canada | Gross worldwide | Opening Weekend Gross in US & Canada |
|---|---|---|---|---|----------------------------|-----------------------|---------|--------------|--------|-------------------|---------------------------|----------------------|-----------------|--------------------------------------|
| 0 | Napoleon | An epic that details the checkered rise and fa... | Ridley Scott | David Scarpa | Action,Adventure,Biography | R | 2h 38m | 2023.0 | 6.7/10 | 38K | NaN | \$37,514,498 | \$84,968,381 | \$20,638,887Nov 26, 2023 |
| 1 | The Hunger Games: The Ballad of Songbirds & Sn... | Coriolanus Snow mentors and develops feelings ... | Francis Lawrence | Michael Lesslie,Michael Arndt,Suzanne Collins | Action,Adventure,Drama | PG-13 | 2h 37m | 2023.0 | 7.2/10 | 37K | \$100,000,000 (estimated) | \$105,043,414 | \$191,729,235 | \$44,607,143Nov 19, 2023 |
| 2 | The Killer | After a fateful near-miss, an assassin battles... | David Fincher | Andrew Kevin Walker,Luc Jacamon,Alexis Nolent | Action,Adventure,Crime | R | 1h 58m | 2023.0 | 6.8/10 | 117K | NaN | NaN | \$421,332 | NaN |
| 3 | Leo | A 74-year-old lizard named Leo and his turtle ... | David Wachtenheim,Robert Smigel,Robert Marianetti | Paul Sado,Robert Smigel,Adam Sandler | Animation,Comedy,Family | PG | 1h 42m | 2023.0 | 7.0/10 | 10K | NaN | NaN | NaN | NaN |
| 4 | Thanksgiving | After a Black Friday riot ends in tragedy, | Eli Roth | Eli Roth,Jeff Rendell | Horror,Mystery,Thriller | R | 1h 46m | 2023.0 | 7.0/10 | 9.1K | NaN | \$25,408,677 | \$29,666,585 | \$10,306,272Nov 19, 2023 |

Spaces 4 () Cell 3 of 79 Go Live Prettier

FileEditSelectionViewGoRunTerminalHelp

Search

Python 3.13.7

IMDB movies (Data Preprocessing).ipynb

C:\Users> reaa> Downloads> IMDB movies (Data Preprocessing).ipynb> Data Exploration

GenerateCode+MarkdownRun AllRestartClear All OutputsGo ToJupyter VariablesOutline

9082

Once Upon a Time in Anatolia

A group of men set out in search of a dead bod...

Nuri Bilge Ceylan

Ercan Kesal,Nuri Bilge Ceylan,Ebru Ceylan

Crime,Drama,Thriller

Not Rated

2h 37m

2011.0

7.8/10

49K

NaN

\$152,408

\$2,099,472

\$10,952Jan 8, 2012

9083 rows x 14 columns

▶✎📄⋮

Data Exploration

df.info()

0.0s

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9083 entries, 0 to 9082
Data columns (total 14 columns):
Column Non-Null Count Dtype

0 Title 9083 non-null object
1 Summary 9083 non-null object
2 Director 9052 non-null object
3 Writer 8759 non-null object
4 Main Genres 9076 non-null object
5 Motion Picture Rating 8285 non-null object
6 Runtime 8918 non-null object
7 Release Year 9076 non-null float64
8 Rating 8813 non-null object
9 Number of Ratings 8813 non-null object
10 Budget 5879 non-null object
11 Gross in US & Canada 6064 non-null object
12 Gross worldwide 7128 non-null object
13 Opening Weekend Gross in US & Canada 5695 non-null object
dtypes: float64(1), object(13)
memory usage: 993.6+ KB

Data Cleaning

Cell 5 of 79Go Live

FileEditSelectionViewGoRunTerminalHelp<-->Search

IMDB movies (Data Preprocessing).ipynb

C:\Users> reaa> Downloads> IMDB movies (Data Preprocessing).ipynb> Data Cleaning> fix Columns' Headers> import re

Generate+Code+MarkdownRun AllRestartClear All OutputsGo ToJupyter VariablesOutlinePython 3.13.7

artists, a
down...

lyndon

| | | | | | | | | | | | | | |
|------|------------------------------|---|-------------------|---|----------------------|-----------|--------|--------|--------|-----|-----|-----------|-------------|
| 9081 | Deep Water | A group of international passengers en route f... | Renny Harlin | Pete Bridges | Thriller | Passed | 2h 32m | 1952.0 | NaN | NaN | NaN | NaN | NaN |
| 9082 | Once Upon a Time in Anatolia | A group of men set out in search of a dead bod... | Nuri Bilge Ceylan | Ercan Kesal,Nuri Bilge Ceylan,Ebru Ceylan | Crime,Drama,Thriller | Not Rated | 2h 37m | 2011.0 | 7.8/10 | 49K | NaN | \$152,408 | \$2,099,472 |

9076 rows x 14 columns

import re

def runtime_to_minutes(runtime):
 if pd.isna(runtime):
 return None
 match = re.match(r'^((\d+))h((\d+))m\$', str(runtime).strip())
 if match:
 hours = int(match.group(1)) if match.group(1) else 0
 minutes = int(match.group(2)) if match.group(2) else 0
 return hours * 60 + minutes
 return None
df = df.copy()
df['Runtime_Minutes'] = df['Runtime'].apply(runtime_to_minutes).astype('Int64')

✓ 0.0sPython

df

✓ 0.0sPython

| | Title | Summary | Director | Writer | Main_Genres | Motion_Picture_Rating | Runtime | Release_Year | Rating | Number_of_Ratings | Budget | Gross_in_US_&_Canada | Gross_worldwide | Opening_We |
|---|----------|-------------------------------------|--------------|--------------|----------------------------|-----------------------|---------|--------------|--------|-------------------|--------|----------------------|-----------------|------------|
| 0 | Napoleon | An epic that details the checked... | Ridley Scott | David Scarpa | Action,Adventure,Biography | R | 2h 38m | 2023.0 | 6.7/10 | 38K | NaN | \$37,514,498 | \$84,968,381 | |

Call 18 of 79Go LivePrettier

File Edit Selection View Go Run Terminal Help Search

IMDB movies (Data Preprocessing).ipynb

C:\Users> reaa> Downloads> IMDB movies (Data Preprocessing).ipynb> Data Cleaning> fix Columns' Headers> import re

Generate + Code + Markdown Run All Restart Clear All Outputs Go To Jupyter Variables Outline Python 3.13.7

9076 rows x 20 columns

```
def fix_price(value):
    if pd.isna(value):
        return None
    value = str(value).strip()
    value = re.sub(r'\\$', '', value)
    value = re.sub(r'\\(estimated\\)', '', value, flags=re.IGNORECASE)
    value = re.sub(r'\\([0-9]+\\.?)\\', r'-\\1', value) # Handle (500) ~ -500
    value = re.sub(r'[^0-9.-]', '', value) # Keep only digits, '.', and '-'
    try:
        return float(value) if value else None
    except ValueError:
        return None

# Check column names first
print("Columns in DataFrame:", df.columns.tolist())

# Update this list to match your actual column names
money_columns = ['Budget', 'Gross_in_US_&Canada', 'Gross_worldwide', 'Opening_Weekend_Gross'] # Adjusted names

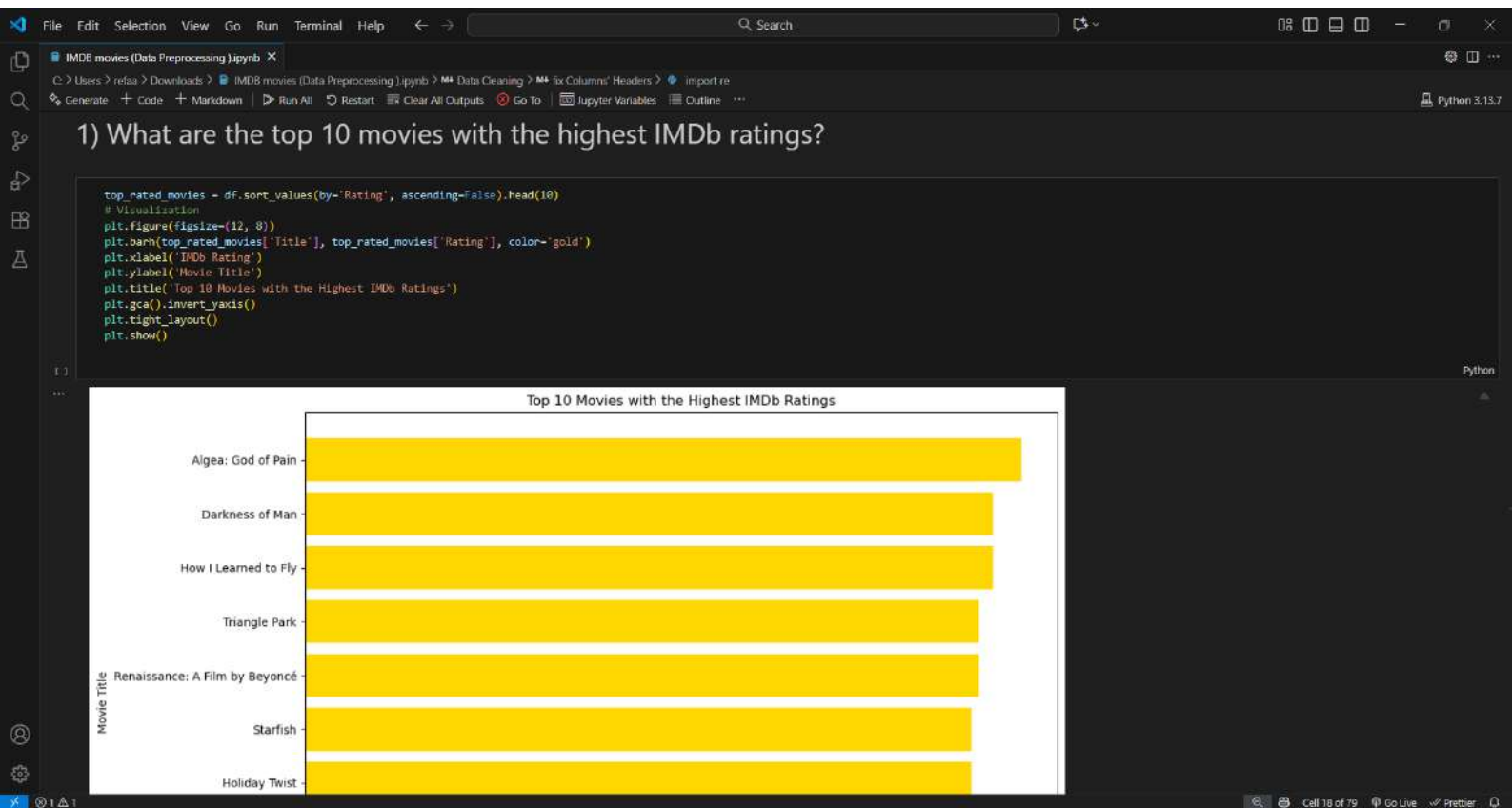
# Apply cleaning
for col in money_columns:
    if col in df.columns: # Check if column exists
        df[col + '_Num'] = df[col].apply(clean_money_column)
    else:
        print(f"Warning: Column '{col}' not found in DataFrame.")
```

Columns in DataFrame: ['Title', 'Summary', 'Director', 'Writer', 'Main_Genres', 'Motion_Picture_Rating', 'Runtime', 'Release_Year', 'Rating', 'Number_of_Ratings', 'Budget', 'Gross_in_US_&Canada', 'Gross_worldwide', 'Runtime_Minutes', 'Opening_Weekend_Gross']

df

| Title | Summary | Director | Writer | Main_Genres | Motion_Picture_Rating | Runtime | Release_Year | Rating | Number_of_Ratings | Budget | Gross_in_US_&Canada | Gross_worldwide | Runtime_Minutes | Opening_Weekend_Gross |
|-------|---------|----------|--------|-------------|-----------------------|---------|--------------|--------|-------------------|--------|---------------------|-----------------|-----------------|-----------------------|
|-------|---------|----------|--------|-------------|-----------------------|---------|--------------|--------|-------------------|--------|---------------------|-----------------|-----------------|-----------------------|

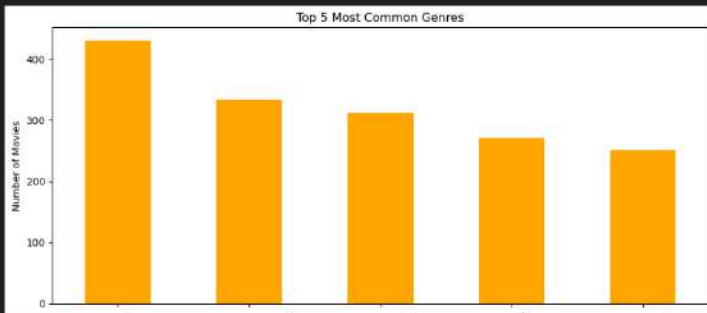
Call 18 of 79 Go Live Prettier



3) Which genres are most common among the movies listed?

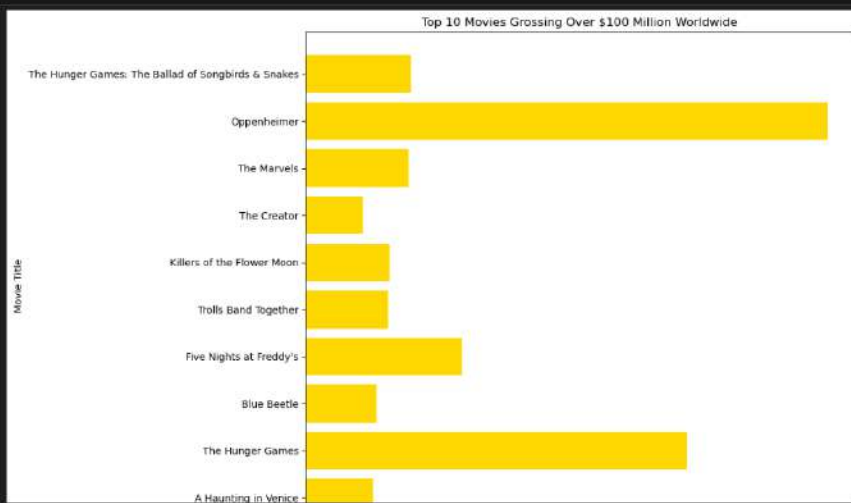
```
common_genres = df['Main_Genres'].value_counts().head(5)
print(common_genres)
plt.figure(figsize=(10, 6))
common_genres.plot(kind='bar', color='orange')
plt.title('Top 5 Most Common Genres')
plt.xlabel('Genre')
plt.ylabel('Number of Movies')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

```
Main_Genres
Drama      430
Comedy, Drama, Romance  333
Drama, Romance  312
Comedy      271
Comedy, Drama  251
Name: count, dtype: int64
```



6) Which movies have grossed over \$100 million worldwide?

```
high_grossing_movies = df[df['Gross_worldwide_Num'] > 100_000_000].head(10)
plt.figure(figsize=(12, 8))
plt.barh(high_grossing_movies['title'], high_grossing_movies['Gross_worldwide_Num'], color='gold')
plt.xlabel('Worldwide Gross Revenue ($)')
plt.ylabel('Movie Title')
plt.title('Top 10 Movies Grossing Over $100 Million Worldwide')
plt.gca().invert_yaxis()
plt.tight_layout()
plt.show()
```



Transaction Details



400 EGP

Transfer Amount



Received Money



From

THOMAS AWAD MAHROUS AWAD
ALLAH-YOUT



thomas_tito@instapay



To

REFAAT ELIA ESHAK ABDELMALEK

refaat_elia@instapay

Reference

120486677239

Date:

31 Oct 2025 09:10 AM

Note

Living Expenses

POWERED BY

