

EDS ASSIGNMENT

Name: Sudarshan

Tarange

Roll No: ME-60

Batch: ME3

PRN: 202401090075

Subject: EDS

Ques: Formulate 20 problem statements for a given dataset using Numpy and Pandas and Apply Numpy and pandas methods to find the solution for the formulated problem statements.

Dataset: Amazon product dataset

Problem Statement 1:

Load the dataset and show first 5 rows & columns.

Code and output Screenshot:

```
1.py>_
1 import pandas as pd
2
3 # Load CSV file
4 df = pd.read_csv('amazon.csv')
5
6 # Display column names only once
7 print("Column Names:")
8 print(df.columns.tolist())
9
10 print("\nFirst 5 Rows:")
11 print(df.head())
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

```
> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/Python/1.py

Column Names:
['product_id', 'product_name', 'category', 'discounted_price', 'actual_price', 'discount_percentage', 'rating', 'rating_count', 'about_product', 'user_id', 'user_name', 'review_id', 'review_title', 'review_content', 'img_link', 'product_link']

First 5 Rows:
  product_id product_name ... img_link product link
0  B07J89H4J1 Wayona Nylon Braided USB to Lightning Fast Cha... ... https://m.media-amazon.com/images/W/WEBP_40237... https://www.amazon.in/Wayona-Braided-WN161G1-Sy...
1  B098NS6PM6 Ambrane Unbreakable 60W / 3A Fast Charging 1.5... ... https://m.media-amazon.com/images/W/WEBP_40237... https://www.amazon.in/Ambrane-Unbreakable-Char...
2  B09GMSM6CT Sounce Fast Phone Charging Cable & Data Sync U... ... https://m.media-amazon.com/images/W/WEBP_40237... https://www.amazon.in/Sounce-iPhone-Charging-C...
3  B08HD186NZ boAt Deuce USB 300 2 in 1 Type-C & Micro USB S... ... https://m.media-amazon.com/images/I/41V5FEWPK... https://www.amazon.in/Deuce-300-Resistant-Tang...
4  B08CF3B7N1 Portronics Konnect L 1.2M Fast Charging 3A 8 P... ... https://m.media-amazon.com/images/W/WEBP_40237... https://www.amazon.in/Portronics-Konnect-POR-1...

[5 rows x 16 columns]
PS C:\Users\LENOVO\OneDrive\Desktop\Python>
```

Problem Statement 2: Average , highest and lowest ratings

```
1 # Mean, max, min of ratings
2 import pandas as pd
3
4 df = pd.read_csv("amazon.csv")
5 df['rating'] = pd.to_numeric(df['rating'], errors='coerce')
6
7 print("Average Rating:", df['rating'].mean())
8 print("Highest Rating:", df['rating'].max())
9 print("Lowest Rating:", df['rating'].min())
10
11 # Summary statistics for numerical columns
12 print(df.describe())
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

```
PS C:\Users\LENOVO\OneDrive\Desktop\Python> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/Python/2.py
Average Rating: 4.096584699453552
Highest Rating: 5.0
Lowest Rating: 2.0
   rating
count  1464.000000
count  1464.000000
mean    4.096585
std     0.291674
min     2.000000
25%     4.000000
50%     4.100000
75%     4.300000
max     5.000000
PS C:\Users\LENOVO\OneDrive\Desktop\Python> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/Python/2.py
Average Rating: 4.096584699453552
Highest Rating: 5.0
Lowest Rating: 2.0
   rating
count  1464.000000
mean    4.096585
std     0.291674
min     2.000000
25%     4.000000
```

Problem Statement 3:

Products with rating greater than 4.0

Code and output Screenshot:

```
3py >-
1 # Filter products with rating >= 4.0
2 import pandas as pd
3 df = pd.read_csv("amazon.csv")
4 df['rating'] = pd.to_numeric(df['rating'], errors='coerce')
5 high_rated = df[df['rating'] >= 4.0]
6 print(high_rated[['product_name', 'rating']])
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

```
> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/python/3.py
product_name rating
0 Wayona Nylon Braided USB to Lightning Fast Cha... 4.2
1 Ambrane Unbreakable 60W / 3A Fast Charging 1.5... 4.0
3 boAt Duice USB 300 2 in 1 Type-C & Micro USB S... 4.2
4 Portronics Connect L 1.2M Fast Charging 3A 8 P... 4.2
6 boAt Micro USB 55 Tangle-free, Sturdy Micro US... 4.1
... ..
1457 Philips HD9306/06 1.5-litre Electric Kettle (M... 4.4
1460 Noir Aqua - 5pcs PP Spun Filter + 1 Spanner | ... 4.0
1461 Prestige Delight PRMD Electric Rice Cooker (1 ... 4.1
1463 Havells Ventil Air DSP 230mm Exhaust Fan (Pist... 4.0
1464 Borosil Jumbo 1000-Watt Grill Sandwich Maker (... 4.3

[1118 rows x 2 columns]
PS C:\Users\LENOVO\OneDrive\Desktop\Python> |
```

Problem Statement 4:

Checking for missing values

Code and output Screenshot:

```
4.py > ...
1 # Check for missing values
2 import pandas as pd
3 df = pd.read_csv("amazon.csv")
4 print(df.isnull().sum())
5 # Drop rows with missing ratings
6 df_clean = df.dropna(subset=['rating'])
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
product_id      0
product_name     0
category         0
discounted_price 0
actual_price     0
discount_percentage 0
product_id      0
product_name     0
category         0
discounted_price 0
actual_price     0
product_id      0
product_name     0
category         0
discounted_price 0
product_id      0
product_name     0
category         0
product_id      0
product_name     0
product_id      0
product_name     0
product_id      0
product_name     0
category         0
product_name     0
product_name     0
product_name     0
```

```
product_name     0
category         0
category         0
discounted_price 0
actual_price     0
discount_percentage 0
rating           0
rating_count     2
about_product    0
user_id          0
user_name        0
review_id        0
review_title     0
review_content   0
review_title     0
review_content   0
img_link         0
product_link     0
dtype: int64
PS C:\Users\LENOVO\OneDrive\Desktop\Python> & C:/Users/LENOVO/AppData/Local/Programs/Python/python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/Python/4.py
product_id      0
product_name     0
category         0
discounted_price 0
actual_price     0
discount_percentage 0
rating           0
rating_count     2
about_product    0
user_id          0
```

Code and output Screenshot:

```

380 Fire-Boltt Ninja Call Pro Plus 1.83 Smart Wat... 91%
368 Fire-Boltt Ninja Call Pro Plus 1.83 Smart Wat... 91%
334 Fire-Boltt Ninja Call Pro Plus 1.83 Smart Wat... 91%
PS C:\Users\LENOVO\OneDrive\Desktop\Python> & C:\Users\LENOVO\AppData\Local\Programs\Python\Python313\python.exe c:\Users\LENOVO\OneDrive\Desktop\Python\6.py
product_name discount_percentage
395 rts [2 Pack] Mini USB C Type C Adapter Plug, T... 94%
364 Fire-Boltt Ninja Call Pro Plus 1.83 Smart Wat... 91%
380 Fire-Boltt Ninja Call Pro Plus 1.83 Smart Wat... 91%
368 Fire-Boltt Ninja Call Pro Plus 1.83 Smart Wat... 91%
334 Fire-Boltt Ninja Call Pro Plus 1.83 Smart Wat... 91%
PS C:\Users\LENOVO\OneDrive\Desktop\Python>

```

Problem Statement 6:

Products with Fast charging in their name

Code and output Screenshot:

```
7.py > ...
1 # Find all products with "Fast Charging" in their name
2 import pandas as pd
3 df = pd.read_csv("amazon.csv")
4
5 fast_charging = df[df['product_name'].str.contains('Fast Charging', case=False)]
6 print(fast_charging[['product_name']])
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS + v

```
PS C:\Users\LENOVO\OneDrive\Desktop\Python> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/Python/7.py
product_name
0 Wayona Nylon Braided USB to Lightning Fast Cha...
1 Ambrane Unbreakable 60W / 3A Fast Charging 1.5...
3 boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...
4 Portronics Konnect L 1.2M Fast Charging 3A 8 P...
5 pTron Solero TB301 3A Type-C Data and Fast Cha...
...
990 Ambrane 2 in 1 Type-C & Micro USB Cable with 6...
992 Ambrane 60W / 3A Fast Charging Output Cable wi...
1000 SWAPKART Fast Charging Cable and Data Sync USB...
1010 Wayona Usb Type C Fast Charger Cable Fast Char...
1018 Lapster 65W compatible for OnePlus Dash Warp C...

[171 rows x 1 columns]
PS C:\Users\LENOVO\OneDrive\Desktop\Python> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/Python/7.py
product_name
0 Wayona Nylon Braided USB to Lightning Fast Cha...
1 Ambrane Unbreakable 60W / 3A Fast Charging 1.5...
3 boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...
4 Portronics Konnect L 1.2M Fast Charging 3A 8 P...
5 pTron Solero TB301 3A Type-C Data and Fast Cha...
...
990 Ambrane 2 in 1 Type-C & Micro USB Cable with 6...
992 Ambrane 60W / 3A Fast Charging Output Cable wi...
1000 SWAPKART Fast Charging Cable and Data Sync USB...
1010 Wayona Usb Type C Fast Charger Cable Fast Char...
1018 Lapster 65W compatible for OnePlus Dash Warp C...

[171 rows x 1 columns]
```

Problem Statement 7:

Top 5 Products

Code and output Screenshot:

```
1 import pandas as pd
2 df = pd.read_csv("amazon.csv")
3 category_counts = df['category'].value_counts()
4 print(category_counts.head()) # Top 5 categories
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

```
> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/Python/8.py

category
Computers&Accessories|Accessories&Peripherals|Cables&Accessories|Cables|USBCables    233
Electronics|WearableTechnology|Smartwatches                                         76
Electronics|Mobility&Accessories|Smartphones&BasicMobility|Smartphones              68
Electronics|HomeTheater,TV&Video|Televisions|SmartTelevisions                      63
Electronics|Headphones,Earbuds&Accessories|Headphones|In-Ear                       52
Name: count, dtype: int64
PS C:\Users\LENOVO\OneDrive\Desktop\Python >
```

Problem Statement 8:

Number of products per category

Code and output Screenshot:

```
1 #Number of Products per Category
2 import pandas as pd
3 df = pd.read_csv("amazon.csv")
4 count_by_category = df.groupby('category')['product_id'].nunique()
5 print("Number of Products per Category:")
6 print(count_by_category)
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

```
> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/Python/9.py

Number of Products per Category:
category
Car&Motorbike|CarAccessories|InteriorAccessories|AirPurifiers&Ionizers              1
Computers&Accessories|Accessories&Peripherals|Adapters|USBtoUSBAdapters            2
Computers&Accessories|Accessories&Peripherals|Audio&VideoAccessories|PCHearsets     1
Computers&Accessories|Accessories&Peripherals|Audio&VideoAccessories|PCHearsets     2
Computers&Accessories|Accessories&Peripherals|Audio&VideoAccessories|PCHearsets     2
OfficeProducts|OfficePaperProducts|Paper|Stationery|Pens,Pencils&WritingSupplies|Pens&Refills|GelInkRollerballPens  2
OfficeProducts|OfficePaperProducts|Paper|Stationery|Pens,Pencils&WritingSupplies|Pens&Refills|LiquidInkRollerballPens  2
OfficeProducts|OfficePaperProducts|Paper|Stationery|Pens,Pencils&WritingSupplies|Pens&Refills|RetractableBallpointPens  2
OfficeProducts|OfficePaperProducts|Paper|Stationery|Pens,Pencils&WritingSupplies|Pens&Refills|StickBallpointPens    3
Toys&Games|Arts&Crafts|Drawing&PaintingSupplies|ColouringPens&Markers                1
Name: product_id, length: 211, dtype: int64
PS C:\Users\LENOVO\OneDrive\Desktop\Python >
```


Problem Statement 9:

maximum discounted product

Code and output Screenshot:

```
11.py 2 ...
1 #maximum discounted product
2 import pandas as pd
3 import numpy as np
4 df = pd.read_csv("amazon.csv")
5 print(df.loc[df['discount_percentage'].idxmax()][['product_name', 'discount_percentage']])
6
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

Name: 695, dtype: object
PS C:\Users\LENOVO\OneDrive\Desktop\Python> & c:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/Python/11.py
product_name rts [2 Pack] Mini USB C Type C Adapter Plug, T...
discount_percentage 94%
Name: 695, dtype: object
PS C:\Users\LENOVO\OneDrive\Desktop\Python>

Problem Statement 10:

Product with the minimum rating

Code and output Screenshot:

```
1 #product with the minimum rating
2 import pandas as pd
3 import numpy as np
4 df = pd.read_csv("amazon.csv")
5 print(df.loc[df['rating'].idxmin()][['product_name', 'rating']])
6
```

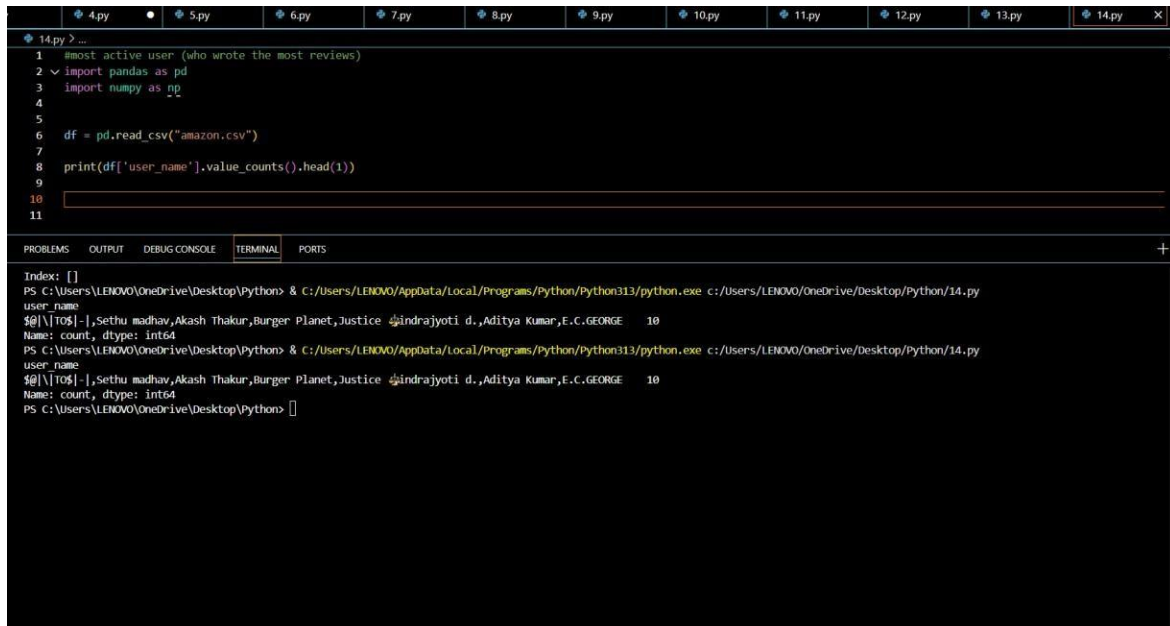
PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

PS C:\Users\LENOVO\OneDrive\Desktop\Python> & c:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/Python/13.py
product_name khalitan ORFin Fan heater for Home and kitchen-...
rating 2
Name: 1309, dtype: object
PS C:\Users\LENOVO\OneDrive\Desktop\Python> & c:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/Python/13.py
product_name khalitan ORFin Fan heater for Home and kitchen-...
rating 2
Name: 1309, dtype: object
PS C:\Users\LENOVO\OneDrive\Desktop\Python>

Problem Statement 11:

Most active users(who wrote most reviews)

Code and output Screenshot:



```
14.py > ...
1 #most active user (who wrote the most reviews)
2 import pandas as pd
3 import numpy as np
4
5 df = pd.read_csv("amazon.csv")
6
7 print(df['user_name'].value_counts().head(1))
8
9
10
11
```

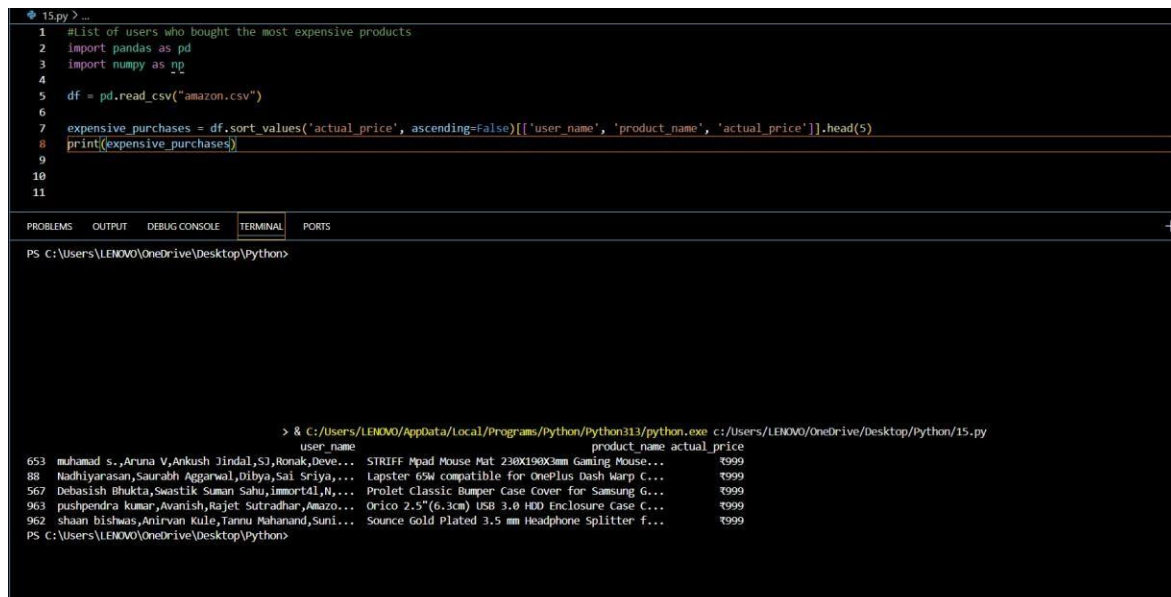
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Index: []
PS C:\Users\LENOVO\OneDrive\Desktop\Python> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/Python/14.py
user_name
%@\|T05|-|,Sethu madhav,Akash Thakur,Burger Planet,Justice @indrajyoti d.,Aditya Kumar,E.C.GEORGE 10
Name: count, dtype: int64
PS C:\Users\LENOVO\OneDrive\Desktop\Python> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/Python/14.py
user_name
%@\|T05|-|,Sethu madhav,Akash Thakur,Burger Planet,Justice @indrajyoti d.,Aditya Kumar,E.C.GEORGE 10
Name: count, dtype: int64
PS C:\Users\LENOVO\OneDrive\Desktop\Python>

Problem Statement 12:

List of users who bought most expensive products

Code and output Screenshot:



```
15.py > ...
1 #List of users who bought the most expensive products
2 import pandas as pd
3 import numpy as np
4
5 df = pd.read_csv("amazon.csv")
6
7 expensive_purchases = df.sort_values('actual_price', ascending=False)[['user_name', 'product_name', 'actual_price']].head(5)
8 print(expensive_purchases)
9
10
11
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\LENOVO\OneDrive\Desktop\Python>

> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/Python/15.py

	user_name	product_name	actual_price
652	muhamad s.,Aruna V,Ankush Jindal,SJ,Ronak,Deve...	STRIFF Mpad Mouse Mat 230X190X3mm Gaming Mouse...	₹999
88	Nadhiyaran,Saurabh Aggarwal,Dibya,Sai Sriya,...	Lapster 694 compatible for OnePlus Dash Warp C...	₹999
567	Debasish Bhukta,Seastik Suman Sahu,immortal,N...	Prolet Classic Bumper Case Cover for Samsung G...	₹999
963	pushpendra kumar,Avanish,Rajet Sutradhar,Amazo...	Orico 2.5"(6.3cm) USB 3.0 HDD Enclosure Case C...	₹999
962	shaan bishwas,Anirvan Kule,Tannu Mahanand,Suni...	Source Gold Plated 3.5 mm Headphone Splitter f...	₹999

PS C:\Users\LENOVO\OneDrive\Desktop\Python>

Problem Statement 13:

Categories with the most products listed

Code and output Screenshot:

```
1 #categories with the most products listed
2 import pandas as pd
3 import numpy as np
4
5 df = pd.read_csv("amazon.csv")
6 top_categories = df['category'].value_counts().head(5)
7 print(top_categories)
8
9
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Name: count, dtype: int64
PS C:\Users\LENOVO\OneDrive\Desktop\Python> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/Python/16.py
category
computers&Accessories|Accessories&Peripherals|Cables&Accessories|cables|USBCables 233
Electronics|WearableTechnology|Smartwatches 76
Electronics|Mobiles&Accessories|Smartphones&BasicMobiles|Smartphones 68
Electronics|HomeTheater,TV&Video|Televisions|SmartTelevisions 63
Electronics|Headphones,Earbuds&Accessories|Headphones|In-Ear 52
Name: count, dtype: int64
PS C:\Users\LENOVO\OneDrive\Desktop\Python>

Problem Statement 14:

top users who reviewed the most unique products

Code and output Screenshot:

```
1 #top users who reviewed the most unique products
2 import pandas as pd
3 import numpy as np
4
5 df = pd.read_csv("amazon.csv")
6 unique_reviews = df.groupby('user_name')['product_id'].nunique().sort_values(ascending=False).head(5)
7 print(unique_reviews)
8
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\LENOVO\OneDrive\Desktop\Python> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/Python/17.py
user_name
\$?| \TOS|-,Sethu madhav,Akash Thakur,Burger Planet,Justice @indrajyoti d.,Aditya Kumar,E.C.GEORGE 8
Satheesh Kadiam,Pritom Chakraborty,Vishwa,Simranpreet Singh,Saptarshi,Amazon Customer,D,RAGHUL,Dharmendra kumar 8
sidharth patnaik,Dr Sunilkumar H,Krishna,K. S. Rao,vinayp,Indhu,Jogi,DRISHITI VASHISTH GUPTA 6
Manav,Adarsh gupta,Sundeep,S.Sayed Ahmed,jaspreet singh,khaja moin,Anand,S.ARIJUGAM 6
venkatesh kg,Raghuram,YOGESH,Nanan Parmar,Nikunj .R. kukadiya,Shivansh Sharma,KAPIL NIDKER,Suriya 5
Name: product_id, dtype: int64
PS C:\Users\LENOVO\OneDrive\Desktop\Python>

Problem Statement 15:

Products above the average rating

Code and output Screenshot:

```
18.py > ...
1 #Products Above the Average Rating
2 import pandas as pd
3 import numpy as np
4
5 df = pd.read_csv("amazon.csv")
6
7 df['rating'] = pd.to_numeric(df['rating'], errors='coerce')
8
9
10 avg_rating = np.nanmean(df['rating'])
11 print("Average Rating:", avg_rating)
12
13 above_avg_products = df[df['rating'] > avg_rating]
14
15
```

PRODUCTS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Products with a Rating Above Average (first 5 rows):

	product name	rating
0	Wayona Nylon Braided USB to Lightning Fast Cha...	4.2
3	boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...	4.2
4	Portronics Connect L 1.2M Fast Charging 3A 8 P...	4.2
6	boAt Micro USB 55 Tangle-free, Sturdy Micro US...	4.1
7	MI Usb Type-C Cable Smartphone (Black)	4.3

PS C:\Users\LENOVO\OneDrive\Desktop\Python>

0	Wayona Nylon Braided USB to Lightning Fast Cha...	4.2
3	boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...	4.2
4	Portronics Connect L 1.2M Fast Charging 3A 8 P...	4.2
6	boAt Micro USB 55 Tangle-free, Sturdy Micro US...	4.1
7	MI Usb Type-C Cable Smartphone (Black)	4.3
0	Wayona Nylon Braided USB to Lightning Fast Cha...	4.2
3	boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...	4.2
4	Portronics Connect L 1.2M Fast Charging 3A 8 P...	4.2
6	boAt Micro USB 55 Tangle-free, Sturdy Micro US...	4.1
0	Wayona Nylon Braided USB to Lightning Fast Cha...	4.2
3	boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...	4.2
4	Portronics Connect L 1.2M Fast Charging 3A 8 P...	4.2

3	boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...	4.2
0	Wayona Nylon Braided USB to Lightning Fast Cha...	4.2
3	boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...	4.2
4	Portronics Connect L 1.2M Fast Charging 3A 8 P...	4.2
0	Wayona Nylon Braided USB to Lightning Fast Cha...	4.2
3	boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...	4.2
0	Wayona Nylon Braided USB to Lightning Fast Cha...	4.2
3	boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...	4.2
0	Wayona Nylon Braided USB to Lightning Fast Cha...	4.2
0	Wayona Nylon Braided USB to Lightning Fast Cha...	4.2
0	Wayona Nylon Braided USB to Lightning Fast Cha...	4.2
0	Wayona Nylon Braided USB to Lightning Fast Cha...	4.2
3	boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...	4.2
4	Portronics Connect L 1.2M Fast Charging 3A 8 P...	4.2
6	boAt Micro USB 55 Tangle-free, Sturdy Micro US...	4.1
7	MI Usb Type-C Cable Smartphone (Black)	4.3
0	Wayona Nylon Braided USB to Lightning Fast Cha...	4.2
3	boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...	4.2
4	Portronics Connect L 1.2M Fast Charging 3A 8 P...	4.2
6	boAt Micro USB 55 Tangle-free, Sturdy Micro US...	4.1
0	Wayona Nylon Braided USB to Lightning Fast Cha...	4.2
3	boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...	4.2

0	Wayona Nylon Braided USB to Lightning Fast Cha...	4.2
3	boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...	4.2
4	Portronics Connect L 1.2M Fast Charging 3A 8 P...	4.2
0	Wayona Nylon Braided USB to Lightning Fast Cha...	4.2
3	boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...	4.2
0	Wayona Nylon Braided USB to Lightning Fast Cha...	4.2
0	Wayona Nylon Braided USB to Lightning Fast Cha...	4.2
0	Wayona Nylon Braided USB to Lightning Fast Cha...	4.2
3	boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...	4.2
4	Portronics Connect L 1.2M Fast Charging 3A 8 P...	4.2
6	boAt Micro USB 55 Tangle-free, Sturdy Micro US...	4.1
7	MI Usb Type-C Cable Smartphone (Black)	4.3

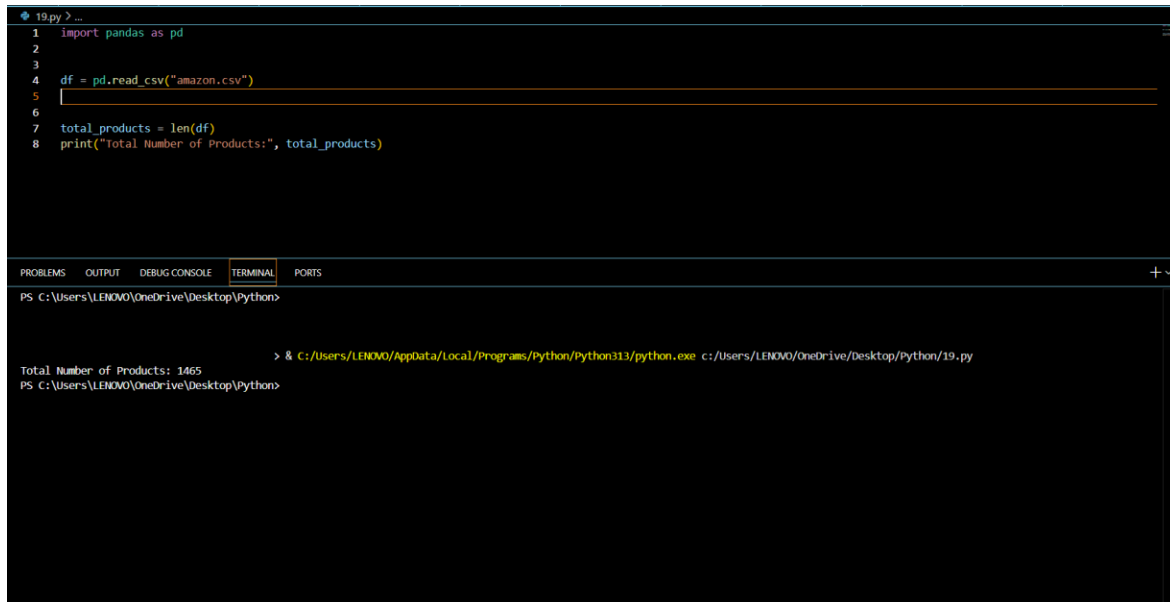
PS C:\Users\LENOVO\OneDrive\Desktop\Python> & C:\Users\LENOVO\AppData\Local\Programs\Python\Python313\python.exe c:\Users\LENOVO\OneDrive\Desktop\python\18.py

Average Rating: 4.096584699453552

Problem Statement 16:

Total number of products

Code and output Screenshot:



```
19.py > ...
1 import pandas as pd
2
3
4 df = pd.read_csv("amazon.csv")
5
6
7 total_products = len(df)
8 print("Total Number of Products:", total_products)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\LENOVO\OneDrive\Desktop\Python>

> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/Python/19.py

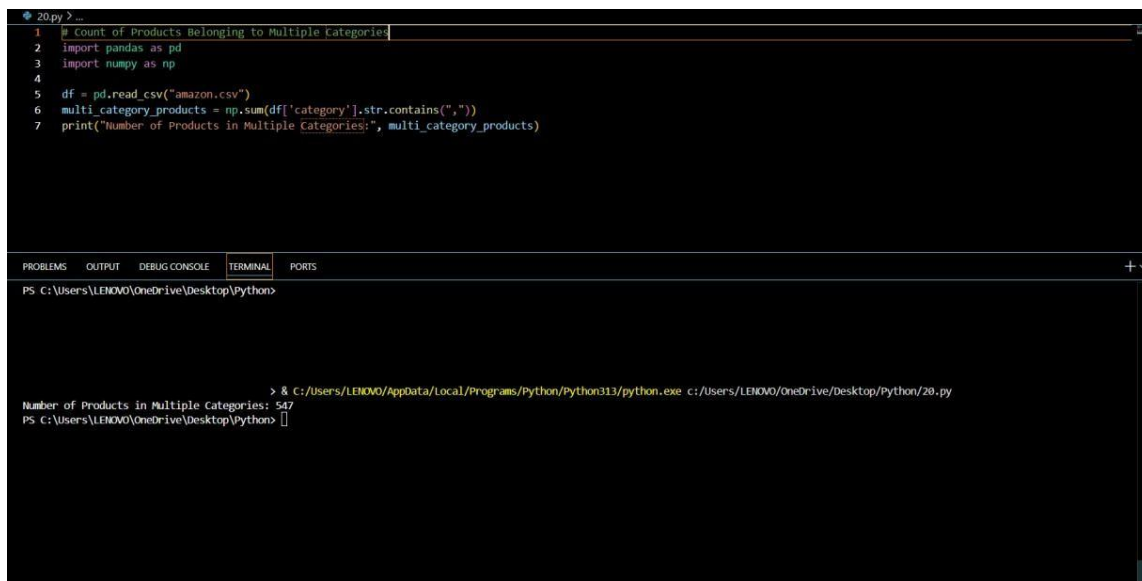
Total Number of Products: 1465

PS C:\Users\LENOVO\OneDrive\Desktop\Python>

Problem Statement 17:

Count of products that belong to multiple categories

Code and output Screenshot:



```
20.py > ...
1 # Count of Products Belonging to Multiple Categories
2 import pandas as pd
3 import numpy as np
4
5 df = pd.read_csv("amazon.csv")
6 multi_category_products = np.sum(df['category'].str.contains(","))
7 print("Number of Products in Multiple Categories:", multi_category_products)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\LENOVO\OneDrive\Desktop\Python>

> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/Python/20.py

Number of Products in Multiple Categories: 547

PS C:\Users\LENOVO\OneDrive\Desktop\Python> []

Problem Statement 17:

Average rating per category

Code and output Screenshot:

```
1 # Average rating by category
2 import pandas as pd
3 import numpy as np
4
5 df = pd.read_csv("amazon.csv")
6
7
8 df['rating'] = pd.to_numeric(df['rating'], errors='coerce')
9
10
11 category_avg = df.groupby('category', as_index=False)['rating'].mean()
12
13
14 sorted_category_avg = category_avg.sort_values('rating', ascending=False)
15
16 print("Average Rating by Category:")
17 print(sorted_category_avg)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/python/5.py

Average Rating by Category:

	category	rating
57	Computers&Accessories Tablets	4.6
48	Computers&Accessories NetworkingDevices Network...	4.5
62	Electronics Cameras&Photography Accessories Film	4.5
38	Computers&Accessories Components Memory	4.5
81	Electronics HomeAudio MediaStreamingDevices St...	4.5
..
3	Computers&Accessories Accessories&Peripherals ...	3.6
88	Electronics HomeTheater,TV&Video Accessories 3...	3.5
2	Computers&Accessories Accessories&Peripherals ...	3.5
14	Computers&Accessories Accessories&Peripherals ...	3.4
146	Home&Kitchen Kitchen&HomeAppliances Coffee,Tea...	3.3

[211 rows x 2 columns]

PS C:\Users\LENOVO\OneDrive\Desktop\Python>

Problem Statement 18:

Number of Products with 0 discount

Code and output Screenshot:

```
10.py > no_discount_count
1
2 import pandas as pd
3 df = pd.read_csv("amazon.csv")
4 no_discount_count = (df['discount_percentage'] == 0).sum()
5 print("Number of Products with No Discount:", no_discount_count)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\LENOVO\OneDrive\Desktop\Python>

> & C:/Users/LENOVO/AppData/Local/Programs/Python/Python313/python.exe c:/Users/LENOVO/OneDrive/Desktop/Python/10.py

Number of Products with No Discount: 0

PS C:\Users\LENOVO\OneDrive\Desktop\Python>

Problem Statement 19:

Checking missing ratings

Code and output Screenshot:

```
1 import pandas as pd
2 df = pd.read_csv("amazon.csv")
3 missing_ratings = df['rating'].isnull().sum()
4 print(f"Missing Ratings: {missing_ratings}")
```

Missing Ratings: 0

PS C:\Users\LENOVO\OneDrive\Desktop\Python>

Problem Statement 20:

Removed duplicate products

Code and output Screenshot:

```
1 import pandas as pd
2 df = pd.read_csv("amazon.csv")
3 df_unique = df.drop_duplicates(['product_name'])
4 print(df_unique)
5
```

product_id ... product_link

0 B073M9H4J1 ... https://www.amazon.in/Wayona-Braided-W1B1G1-Sy...

1 B098NS6PVG ... https://www.amazon.in/Ambrane-Unbreakable-Char...

2 B096MSWCT ... https://www.amazon.in/Source-iPhone-Charging-C...

3 B08UDJ86NZ ... https://www.amazon.in/Deuce-300-Resistant-Tang...

4 B08CF3B7N1 ... https://www.amazon.in/Portronics-Konnect-POR-1...

... ...

1460 B08L7J3T31 ... https://www.amazon.in/Noir-Aqua-Spanner-Purifi...

1461 B01M6453MB ... https://www.amazon.in/Prestige-Delight-PRMO-1-...

1462 B089P2L1L4 ... https://www.amazon.in/Bajaj-RX-10-2000-Watt-Co...

1463 B08J5DYCCA ... https://www.amazon.in/Havells-Ventilair-230mm-...

1464 B01486F4G6 ... https://www.amazon.in/Borosil-Jumbo-1000-Watt-...

[1337 rows x 16 columns]

PS C:\Users\LENOVO\OneDrive\Desktop\Python>