

EDS ASSIGNMENT - 1

NAME : RUGWEDA YENDE
DIVISION : CS2 BATCH : CS23
ROLL NO : CS2-65
PRN : 202401040262

```
#Determine the total quantity sold for each product across all stores.  
import pandas as pd  
import numpy as np  
  
df = pd.read_csv('grocery_dataset.csv')  
df['Date'] = pd.to_datetime(df['Date'])  
  
quantity_per_product = df.groupby('Product')['Quantity_Sold'].sum()  
print(quantity_per_product)
```

yruwg@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory

\$ C:/Users/yruwg/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yruwg/OneDrive/Desktop/Assignment 1 EDS theory/prob.1.py"

2349.0

```

# Calculate the mean, median, and standard deviation of product prices.
import pandas as pd
import numpy as np

df = pd.read_csv('grocery_dataset.csv')
df['Date'] = pd.to_datetime(df['Date'])

mean_price = df['Price'].mean()
median_price = df['Price'].median()
std_price = df['Price'].std()

print(f"Mean: {mean_price}, Median: {median_price}, Standard Deviation: {std_price}")

```

```

yrugw@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory
$ C:/Users/yrugw/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yrugw/OneDrive/Desktop/Assignment 1 EDS theory/prob.1 copy 2.py"
Category
Bakery      2.200000
Beverages   1.900000
Dairy       2.800000
Fruits      1.333333
Meat        6.333333
Vegetables   0.900000
Name: Price, dtype: float64

```

```
#Calculate the total revenue generated from all product sales.
import pandas as pd
import numpy as np

df = pd.read_csv('grocery_dataset.csv')
df['Date'] = pd.to_datetime(df['Date'])

df['Revenue'] = df['Price'] * df['Quantity_Sold']
total_revenue = df['Revenue'].sum()
print(total_revenue)
```

yrugw@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory

\$ C:/Users/yrugw/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yrugw/OneDrive/Desktop/Assignment 1 EDS theory/prob.1 copy 3.py"
Water

```

#Identify the product that has the highest quantity sold overall.
import pandas as pd
import numpy as np

df = pd.read_csv('grocery_dataset.csv')
df['Date'] = pd.to_datetime(df['Date'])

top_product = df.groupby('Product')['Quantity_Sold'].sum().idxmax()
print(top_product)

```

```

yrugw@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory
$ C:/Users/yrugw/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yrugw/OneDrive/Desktop/Assignment 1 EDS theory/prob.1 copy 4.py"
Product
Apples      100
Bananas     120
Beef        25
Bread       40
Butter      35
Carrots     75
Cereal      45
Cheese      20
Chicken     30
Eggs        60
Fish        20
Milk        50
Orange Juice 55
Potatoes    90
Soda        90
Spinach     60
Strawberries 80
Tomatoes    70
Water      150
Yogurt      65
Name: Quantity_Sold, dtype: int64

```



```
#Find the average price of products grouped by their category.
import pandas as pd
import numpy as np

df = pd.read_csv('grocery_dataset.csv')
df['Date'] = pd.to_datetime(df['Date'])

avg_price_by_category = df.groupby('Category')['Price'].mean()
print(avg_price_by_category)
```

```
yruqw@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory
```

```
$ C:/Users/yruqw/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yruqw/OneDrive/Desktop/Assignment 1 EDS theory/prob.1 copy 5.py"
```

```
City
```

```
New York      846.5
```

```
Los Angeles   757.0
```

```
Chicago       745.5
```

```
Name: Revenue, dtype: float64
```

```
#Count the number of unique products available in the dataset.  
import pandas as pd  
import numpy as np  
  
df = pd.read_csv('grocery_dataset.csv')  
df['Date'] = pd.to_datetime(df['Date'])  
  
unique_products = df['Product'].nunique()  
print(unique_products)
```

```
yrugw@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory
```

```
$ C:/Users/yrugw/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yrugw/OneDrive/Desktop/Assignment 1 EDS theory/prob.1 copy 6.py"
```

```
20
```

```
#Replace all missing values in the 'Category' column with 'Miscellaneous'.
import pandas as pd
import numpy as np

df = pd.read_csv('grocery_dataset.csv')
df['Date'] = pd.to_datetime(df['Date'])

df.fillna({'Category': 'Miscellaneous'}, inplace=True)
```

yruwg@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory

\$ C:/Users/yruwg/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yruwg/OneDrive/Desktop/Assignment 1 EDS theory/prob.1 copy 7.py"

Mean: 2.5349999999999997, Median: 1.9, Standard Deviation: 1.8968880332076312


```
# Compute the total number of transactions made by each unique customer.
import pandas as pd
import numpy as np

df = pd.read_csv('grocery_dataset.csv')
df['Date'] = pd.to_datetime(df['Date'])

transactions_per_customer = df['Customer_ID'].value_counts()
print(transactions_per_customer)
```

```
yrugw@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory
$ C:/Users/yrugw/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yrugw/OneDrive/Desktop/Assignment 1 EDS theory/prob.1 copy 8.py"
Store
Store_A    846.5
Store_B    757.0
Store_C    745.5
Name: Revenue, dtype: float64
```

```
# Calculate the percentage contribution of each product to the overall sales revenue.
import pandas as pd
import numpy as np

df = pd.read_csv('grocery_dataset.csv')
df['Date'] = pd.to_datetime(df['Date'])

df['Revenue'] = df['Price'] * df['Quantity_Sold']
product_revenue = df.groupby('Product')['Revenue'].sum()
percentage_contribution = (product_revenue / product_revenue.sum()) * 100
print(percentage_contribution)
```

yruwg@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory

\$ C:/Users/yruwg/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yruwg/OneDrive/Desktop/Assignment 1 EDS theory/prob.1 copy 9.py"

2025-04-04 00:00:00

```
#Find the top 5 cities that have generated the highest total revenue.  
import pandas as pd  
import numpy as np  
  
df = pd.read_csv('grocery_dataset.csv')  
df['Date'] = pd.to_datetime(df['Date'])  
  
df['Revenue'] = df['Price'] * df['Quantity_Sold']  
top_5_cities = df.groupby('City')['Revenue'].sum().sort_values(ascending=False).head(5)  
print(top_5_cities)
```

```
yrugw@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory
```

```
$ C:/Users/yrugw/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yrugw/OneDrive/Desktop/Assignment 1 EDS theory/prob.1 copy 10.py"
```

```
Empty DataFrame
```

```
Columns: [Product, Category, Price, Quantity_Sold, Date, Store, City, Customer_ID, Revenue]
```

```
Index: []
```

```
#Group the sales data by store and compute the total revenue per store.  
import pandas as pd  
import numpy as np  
  
df = pd.read_csv('grocery_dataset.csv')  
df['Date'] = pd.to_datetime(df['Date'])  
  
df['Revenue'] = df['Price'] * df['Quantity_Sold']  
revenue_per_store = df.groupby('Store')['Revenue'].sum()  
print(revenue_per_store)
```

```
yruwg@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory  
$ C:/Users/yruwg/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yruwg/OneDrive/Desktop/Assignment 1 EDS theory/prob.1 copy 11.py"
```



```
#Find the correlation between product price and quantity sold.
import pandas as pd
import numpy as np

df = pd.read_csv('grocery_dataset.csv')
df['Date'] = pd.to_datetime(df['Date'])

correlation = df['Price'].corr(df['Quantity_Sold'])
print(correlation)
```

```
yругw@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory
$ C:/Users/yругw/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yругw/OneDrive/Desktop/Assignment 1 EDS theory/prob.1 copy 12.py"
Customer_ID
C001      1
C002      1
C019      1
C018      1
C017      1
C016      1
C015      1
C014      1
C013      1
C012      1
C011      1
C010      1
C009      1
C008      1
C007      1
C006      1
C005      1
C004      1
C003      1
C020      1
Name: count, dtype: int64
```



```

# List all the transactions where the total bill amount exceeds $500.
import pandas as pd
import numpy as np

df = pd.read_csv('grocery_dataset.csv')
df['Date'] = pd.to_datetime(df['Date'])

df['Revenue'] = df['Price'] * df['Quantity_Sold']
high_value_transactions = df[df['Revenue'] > 500]
print(high_value_transactions)

```

yrugw@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory

\$ C:/Users/yrugw/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yrugw/OneDrive/Desktop/Assignment 1 EDS theory/prob.1 copy 13.py"

```

Product
Apples      5.108557
Bananas     4.086845
Beef        6.917837
Bread       2.554278
Butter      4.022989
Carrots     2.873563
Cereal      5.555556
Cheese      3.405705
Chicken     7.024266
Eggs        7.662835
Fish        5.959983
Milk        5.321413
Orange Juice 7.492550
Potatoes    2.298851
Soda        5.747126
Spinach     2.809706
Strawberries 6.811409
Tomatoes    2.979991
Water       6.385696
Yogurt      4.980843
Name: Revenue, dtype: float64

```

```
# Find the total number of transactions per city.
import pandas as pd
import numpy as np

df = pd.read_csv('grocery_dataset.csv')
df['Date'] = pd.to_datetime(df['Date'])

transactions_per_city = df['City'].value_counts()
print(transactions_per_city)
```

yrugw@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory

\$ C:/Users/yrugw/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yrugw/OneDrive/Desktop/Assignment 1 EDS theory/prob.1 copy 14.py"

City	Chicago	Los Angeles	New York
Product			
Apples	0.0	0.0	120.0
Bananas	0.0	96.0	0.0
Beef	162.5	0.0	0.0
Bread	0.0	0.0	60.0
Butter	0.0	0.0	94.5
Carrots	67.5	0.0	0.0
Cereal	130.5	0.0	0.0
Cheese	80.0	0.0	0.0
Chicken	165.0	0.0	0.0
Eggs	0.0	180.0	0.0
Fish	140.0	0.0	0.0
Milk	0.0	0.0	125.0
Orange Juice	0.0	0.0	176.0
Potatoes	0.0	54.0	0.0
Soda	0.0	0.0	135.0
Spinach	0.0	0.0	66.0
Strawberries	0.0	160.0	0.0
Tomatoes	0.0	0.0	70.0
Water	0.0	150.0	0.0
Yogurt	0.0	117.0	0.0

```
# Find the top 3 most expensive categories based on average product price.
import pandas as pd
import numpy as np

df = pd.read_csv('grocery_dataset.csv')
df['Date'] = pd.to_datetime(df['Date'])

top_3_expensive_categories = df.groupby('Category')['Price'].mean().sort_values(ascending=False).head(3)
print(top_3_expensive_categories)
```

yругw@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory

\$ C:/Users/yругw/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yругw/OneDrive/Desktop/Assignment 1 EDS theory/prob.1 copy 15.py"

	Product	Quantity_Sold	Quantity_Sold_Normalized
0	Milk	50	0.230769
1	Bread	40	0.153846
2	Eggs	60	0.307692
3	Apples	100	0.615385
4	Chicken	30	0.076923
5	Bananas	120	0.769231
6	Cheese	20	0.000000
7	Tomatoes	70	0.384615
8	Potatoes	90	0.538462
9	Beef	25	0.038462
10	Orange Juice	55	0.269231
11	Yogurt	65	0.346154
12	Cereal	45	0.192308
13	Butter	35	0.115385
14	Strawberries	80	0.461538
15	Carrots	75	0.423077
16	Spinach	60	0.307692
17	Water	150	1.000000
18	Fish	20	0.000000
19	Soda	90	0.538462


```
prob.1 copy 16.py - ...  
#Find the date on which the maximum sales revenue was recorded.  
import pandas as pd  
import numpy as np  
  
df = pd.read_csv('grocery_dataset.csv')  
df['Date'] = pd.to_datetime(df['Date'])  
  
df['Revenue'] = df['Price'] * df['Quantity_Sold']  
daily_revenue = df.groupby('Date')['Revenue'].sum()  
max_revenue_date = daily_revenue.idxmax()  
print(max_revenue_date)
```

yrugw@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory

\$ C:/Users/yrugw/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yrugw/OneDrive/Desktop/Assignment 1 EDS theory/prob.1 copy 16.py"
-0.7266459035184912

```
#Find the total quantity of fruits sold.  
import pandas as pd  
import numpy as np  
  
df = pd.read_csv('grocery_dataset.csv')  
df['Date'] = pd.to_datetime(df['Date'])  
  
total_fruits_quantity = df[df['Category'] == 'Fruits']['Quantity_Sold'].sum()  
print(total_fruits_quantity)
```

yruqw@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory

\$ C:/Users/yruqw/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yruqw/OneDrive/Desktop/Assignment 1 EDS theory/prob.1 copy 17.py"
300


```
# Find the average revenue generated per transaction.
import pandas as pd
import numpy as np

df = pd.read_csv('grocery_dataset.csv')
df['Date'] = pd.to_datetime(df['Date'])

df['Revenue'] = df['Price'] * df['Quantity_Sold']
avg_revenue_per_transaction = df['Revenue'].mean()
print(avg_revenue_per_transaction)
```

yrugw@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory

\$ C:/Users/yrugw/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yrugw/OneDrive/Desktop/Assignment 1 EDS theory/prob.1 copy 18.py"

117.45

prob.1 copy 19.py

```
#Create a pivot table showing total sales revenue per product for each city.
import pandas as pd
import numpy as np

df = pd.read_csv('grocery_dataset.csv')
df['Date'] = pd.to_datetime(df['Date'])

df['Revenue'] = df['Price'] * df['Quantity_Sold']
pivot_table = pd.pivot_table(df, values='Revenue', index='Product', columns='City', aggfunc='sum', fill_value=0)
print(pivot_table)
```

yruwg@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory

\$ C:/Users/yruwg/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yruwg/OneDrive/Desktop/Assignment 1 EDS theory/prob.1 copy 19.py"

City

New York 8

Los Angeles 6

Chicago 6

Name: count, dtype: int64

```
#Normalize the 'Quantity Sold' column using Min-Max normalization.
import pandas as pd
import numpy as np

df = pd.read_csv('grocery_dataset.csv')
df['Date'] = pd.to_datetime(df['Date'])

df['Quantity_Sold_Normalized'] = (df['Quantity_Sold'] - df['Quantity_Sold'].min()) / (df['Quantity_Sold'].max() - df['Quantity_Sold'].min())
print(df[['Product', 'Quantity_Sold', 'Quantity_Sold_Normalized']])
```

yrugw@Deadlux MINGW64 ~/OneDrive/Desktop/Assignment 1 EDS theory

\$ C:/Users/yrugw/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/yrugw/OneDrive/Desktop/Assignment 1 EDS theory/prob.1 copy 20.py"

Category

Meat 6.333333

Dairy 2.800000

Bakery 2.200000

Name: Price, dtype: float64