Theory Activity No. 1

NAME- NEHA HIDDUGGI

DIVISION- CS7

ROLLNO- 24

PRN-202401080064

BATCH- CS72

1. #Find the total number of products in the inventory.

Python

```
import pandas as pd
df = pd.read_csv("/content/Grocery_Inventory_400.csv")
len(df)
```

```
400
```

2. #Find the total number of different Sub-Categories.

Python

```
df['Sub_Category'].nunique()
```

```
23
```

3. # what is the average Unit Price of all products?

Python

```
df['Unit_Price'].mean()
```

```
np.float64(250.54602500000001)
```

4. #Calculate the total Quantity of all Dairy products.

Python

```
df[df['Category'] == 'Dairy']['Quantity'].sum()
```

```
np.int64(19592)
```

5.# Find the oldest Manufacture Date.

Python

```
df['Manufacture_Date'] = pd.to_datetime(df['Manufacture_Date'])
df['Manufacture_Date'].min()
```

```
Timestamp('2020-05-21 00:00:00')
```

6.# Which supplier supplies the maximum number of products?

Python

```
df['Supplier'].value_counts().idxmax()
```

```
'TastyTreats'
```

7.# Calculate Profit (Unit_Price - Cost) for each product.

Python

```
df['Profit'] = df['Unit_Price'] - df['Cost']
df[['Product_Name', 'Profit']]
```

	Product_Name	Profit
0	Popcorn Game	77.98
1	Popcorn They	27.08
2	Bread Anyone	104.49
3	Nuts Career	22.31
4	Tea Up	95.49
...	...	...
395	Soda No	71.13
396	Cookies Race	86.42
397	Frozen Vegetables Standard	133.46
398	Chips Generation	39.42
399	Frozen Vegetables Strategy	40.24

400 rows × 2 columns

8.# Find the average Cost for each Category.

Python

```
df.groupby('Category')['Cost'].mean()
```

Category	Cost
Bakery	172.972464
Beverages	194.313971
Dairy	197.831111
Frozen	191.880606
Fruits	186.461017
Snacks	173.617727

dtype: float64

9.# Find the percentage of products that are not In Stock

Python

```
not_in_stock = df[df['In_Stock'] == 'No'].shape[0]
total = df.shape[0]
(not_in_stock/total)*100
```

```
0.0
```

10.# Find the Supplier with minimum average Cost.

Python

```
df.groupby('Supplier')['Cost'].mean().idxmin()
```

```
'OrganicWorld'
```

11.# Find how many products have a Profit greater than 100.

Python

```
df[df['Profit'] > 100].shape[0]
```

```
92
```

12.# Find the Category with the maximum number of products.

Python

```
df['Category'].value_counts().idxmax()
```

```
'Dairy'
```

13.# Find the top 3 Suppliers who supply the most products.

Python

```
df['Supplier'].value_counts().head(3)
```

Supplier	count
Tasty Treats	75
GoodFoods	72
YummyBrands	67

dtype: int64

14.# Which Sub-Category has the least number of products?

Python

```
df['Sub_Category'].value_counts().idxmin()
```

```
'Apple'
```

15.# Count how many Suppliers supply Dairy products.

Python

```
df[df['Category'] == 'Dairy']['Supplier'].nunique()
```

```
6
```

16.# Find the Category with the least total Quantity.

```
df.groupby('Category')['Quantity'].sum().idxmin()
```

```
'Beverages'
```

17.# Find the Supplier who provides the maximum quantity overall.

Python

```
df.groupby('Supplier')['Quantity'].sum().idxmax()
```

```
'GoodFoods'
```

18.# How many products have Manufacture Date in 2023?

Python

```
df[df['Manufacture_Date'].dt.year == 2023].shape[0]
```

```
77
```

19.# Find the most common Sub-Category.

python

```
df['Sub_Category'].value_counts().idxmax()
```

```
'Frozen Meals'
```

20.# List all unique Categories available

Python

```
df['Category'].unique()
```

```
array(['Snacks', 'Bakery', 'Beverages', 'Frozen', 'Fruits', 'Dairy'],
      dtype=object)
```