20 Problem Statement : (GroceryDataset)

Name : Mehdi Shaikh
Roll No : CS5-66
PRN: 202401100083

1. Clean the `Price` column and convert it to numeric.

```
df['Price'] = df['Price'].str.replace('$', '', regex=False).astype(float)
```

2. Extractnumeric ratingvalue from the`Rating` column.

```
import re
def extract_rating(text):if pd.isna(text):
return None                match=re.search(r'Rated ([0-9.]+)', text)          return
float(match.group(1))if matchelseNone


df['Rating Value'] = df['Rating'].apply(extract_rating)
```

3. Count how many products have 'No Discount'.

```
(df['Discount'] == 'No Discount').sum()
```

4. Find theaverage price ofallproducts.df['Price'].mean()

5. List the top 5 most expensive products.

```
df.nlargest(5, 'Price')[['Title', 'Price']]
```

6. Howmanyproducts aremissing arating?

```
df['Rating'].isna().sum()
```

7. Create a new column 'Has Discount' (True/False).

```
df['Has Discount'] = df['Discount'] != 'No Discount'
```

8. Whatarethedistinctsubcategories?

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

9. Find the subcategory with the most products.

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

10.Fill missing 'Currency' values with '$'.

```
df['Currency'] = df['Currency'].fillna('$')
```

11.Find products priced above $100.

```
df[df['Price'] > 100][['Title', 'Price']]
```

12.Calculatetheaverageratingbysubcategory.df.groupby('Sub Category')['Rating Value'].mean()

13.Create a binary column: High Rated (>4.5).

```
df['High Rated'] = df['Rating Value'] > 4.5
```

14.Drop rows where 'Price' or 'Sub Category' is missing.

```
df.dropna(subset=['Price', 'Sub Category'])
```

15.Create a NumPy array of all prices.

```
np.array(df['Price'].dropna())
```

16.Find median price using NumPy.

```
np.median(np.array(df['Price'].dropna()))
```

17.Find standard deviation of

ratings.np.std(df['Rating

Value'].dropna().to_numpy())

18.Replace missing 'Product Description' with 'No Description'.

df['Product Description'] = df['Product Description'].fillna('No Description')

19.Group products into price bins (cheap, moderate, expensive).

bins = [0, 50, 150, np.inf] labels =

['Cheap','Moderate', 'Expensive']df['Price

Category']=pd.cut(df['Price'],bins=bins,

labels=labels)

20.Find how many expensive products are highly rated.

df[(df['PriceCategory']=='Expensive') &(df['High Rated'])].shape[0]