# Retail Sales Data Analysis

# Tools

Python

Pandas

Jupyter NoteBook

Analysis

# Overview:

This project analyzes retail sales data from a Superstore dataset using  the Pandas library in Python, focusing on data cleaning, transformation, and exploratory data analysis (EDA). The goal is to extract actionable business insights without using any visualization libraries—demonstrating strong proficiency in data manipulation and analysis with Pandas.


Back to Agenda

```
import pandas as pd
data = pd.read_csv("Sample - Superstore.csv",encoding="cp1252")
```

```
data.head(5)
```

| | Row ID | Order ID | Order Date | Ship Date | Ship Mode | Customer ID | Customer Name | Segment | Country | City | ... | Postal Code | Region | Product ID | Category | Su Catego |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | CA-2016-152156 | 11/08/2016 | 11/11/2016 | Second Class | CG-12520 | Claire Gute | Consumer | United States | Henderson | ... | 42420 | South | FUR-BO-10001798 | Furniture | Bookca |
| **1** | 2 | CA-2016-152156 | 11/08/2016 | 11/11/2016 | Second Class | CG-12520 | Claire Gute | Consumer | United States | Henderson | ... | 42420 | South | FUR-CH-10000454 | Furniture | Cha |
| **2** | 3 | CA-2016- | 06/12/2016 | 6/16/2016 | Second Class | DV-13045 | Darrin Van Huff | Corporate | United States | Los Angeles | ... | 90036 | West | OFF-LA-10000240 | Office Supplies | Lab |

[41]:
```
data.tail(10)
```

[41]:

| | Row ID | Order ID | Order Date | Ship Date | Ship Mode | Customer ID | Customer Name | Segment | Country | City | ... | Postal Code | Region | Product ID | Category | S Categ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **9984** | 9985 | CA-2015-100251 | 5/17/2015 | 5/23/2015 | Standard Class | DV-13465 | Dianna Vittorini | Consumer | United States | Long Beach | ... | 11561 | East | OFF-LA-10003766 | Office Supplies | La |
| **9985** | 9986 | CA-2015-100251 | 5/17/2015 | 5/23/2015 | Standard Class | DV-13465 | Dianna Vittorini | Consumer | United States | Long Beach | ... | 11561 | East | OFF-SU-10000898 | Office Supplies | Supp |
| **9986** | 9987 | CA-2016- | 9/29/2016 | 10/03/2016 | Standard | ML-17410 | Maris LaWare | Consumer | United | Los Angeles | ... | 90008 | West | TEC-AC- | Technology | Accesso |

[55]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 4042 entries, 0 to 9993
Data columns (total 21 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   Row ID         4042 non-null   int64  
 1   Order ID       4042 non-null   object 
 2   Order Date     4042 non-null   object 
 3   Ship Date      4042 non-null   object 
 4   Ship Mode      4042 non-null   object 
 5   Customer ID    4042 non-null   object 
 6   Customer Name  4042 non-null   object 
 7   Segment        4042 non-null   object 
 8   Country        4042 non-null   object 
 9   City           4042 non-null   object 
 10  State          4042 non-null   object 
 11  Postal Code    4042 non-null   int64  
 12  Region         4042 non-null   object 
 13  Product ID     4042 non-null   object 
 14  Category       4042 non-null   object 
 15  Sub-Category   4042 non-null   object 
 16  Product Name   4042 non-null   object 
 17  Sales          4042 non-null   float64
 18  Quantity       4042 non-null   int64  
 19  Discount       4042 non-null   float64
 20  Profit         4042 non-null   float64
dtypes: float64(3), int64(3), object(15)
memory usage: 694.7+ KB
```

**drop unnecessary column`**

[42]:
```
data.drop(columns=["Postal Code"])
```

[42]:

| Customer ID | Customer Name | Segment | Country | City | State | Region | Product ID | Category | Sub-Category | Product Name | Sales | Quantity | Discount | Profit |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CG-12520 | Claire Gute | Consumer | United States | Henderson | Kentucky | South | FUR-BO-10001798 | Furniture | Bookcases | Bush Somerset Collection Bookcase | 261.9600 | 2 | 0.00 | 41.9136 |
| CG-12520 | Claire Gute | Consumer | United States | Henderson | Kentucky | South | FUR-CH-10000454 | Furniture | Chairs | Hon Deluxe Fabric Upholstered Stacking Chairs,... | 731.9400 | 3 | 0.00 | 219.5820 |
| DV-13045 | Darrin Van Huff | Corporate | United States | Los Angeles | California | West | OFF-LA-10000240 | Office Supplies | Labels | Self-Adhesive Address Labels for Typewriters b... | 14.6200 | 2 | 0.00 | 6.8714 |
| SO-20335 | Sean O'Donnell | Consumer | United States | Fort Lauderdale | Florida | South | FUR-TA-10000577 | Furniture | Tables | Bretford CR4500 Series Slim Rectangular Table | 957.5775 | 5 | 0.45 | -383.0310 |
| SO-20335 | Sean O'Donnell | Consumer | United States | Fort Lauderdale | Florida | South | OFF-ST-10000760 | Office Supplies | Storage | Eldon Fold 'N Roll Cart | 22.3680 | 2 | 0.20 | 2.5164 |

## Convert Order Date to datetime

[43]:
```
# Convert while allowing for format variations
data['Order Date'] = pd.to_datetime(data['Order Date'], errors='coerce', dayfirst=True)
data['Order Date'] = data['Order Date'].dt.strftime('%d/%m/%Y')
```

[45]:
```
data["Order Date"]
```

[45]:
```
0       11/08/2016
1       11/08/2016
2       06/12/2016
3       10/11/2015
4       10/11/2015
           ...
9989           NaN
9990           NaN
9991           NaN
9992           NaN
9993    05/04/2017
Name: Order Date, Length: 9994, dtype: object
```

## Check and remove null values if needed

[50]:
```
data.isnull().sum()
```

[50]:
```
Row ID            0
Order ID          0
Order Date     5952
Ship Date         0
Ship Mode         0
Customer ID       0
Customer Name     0
Segment           0
Country           0
City              0
State             0
Postal Code       0
Region            0
Product ID        0
Category          0
Sub-Category      0
Product Name      0
Sales             0
Quantity          0
Discount          0
Profit            0
dtype: int64
```

[54]:
```
data = data.dropna()
data.isnull().sum()
```

[54]:
```
Row ID            0
Order ID          0
Order Date        0
Ship Date         0
Ship Mode         0
Customer ID       0
Customer Name     0
Segment           0
Country           0
City              0
State             0
Postal Code       0
Region            0
Product ID        0
Category          0
Sub-Category      0
Product Name      0
Sales             0
Quantity          0
Discount          0
Profit            0
dtype: int64
```

## Add columns for Month, Day, Year

[70]:
```
#Convert Order Date to datetime
data["Order Date"] = pd.to_datetime(data["Order Date"])

#Extract Month, Day, Year
data["Month"] = data["Order Date"].dt.month
data["Day"] = data["Order Date"].dt.day
data["Year"] = data["Order Date"].dt.year

print(data[["Month","Day" ,"Year"]].head(5))
print()
print(data[["Month","Day" ,"Year"]].tail(5))
```

```
   Month  Day  Year
0     11    8  2016
1     11    8  2016
2      6   12  2016
3     10   11  2015
4     10   11  2015

      Month  Day  Year
9978     12    6  2016
9979     12    6  2016
9980      9    6  2015
9981      8    3  2017
9993      5    4  2017
```

## Create a Total_Sale column

[75]:
```
data["Total_sale"] = data["Sales"] - data["Sales"] * data["Discount"]
data[["Sales" , "Discount" , "Total_sale"]].head()
```

[75]:

| | Sales | Discount | Total_sale |
|---|---|---|---|
| **0** | 261.9600 | 0.00 | 261.960000 |
| **1** | 731.9400 | 0.00 | 731.940000 |
| **2** | 14.6200 | 0.00 | 14.620000 |
| **3** | 957.5775 | 0.45 | 526.667625 |
| **4** | 22.3680 | 0.20 | 17.894400 |

## Analysis

### 1 Top 5 products by sales

[79]:
```
data.groupby("Product Name")["Sales"].sum().sort_values(ascending=False).head(5)
```

[79]:
```
Product Name
Canon imageCLASS 2200 Advanced Copier                                17499.9500
Lexmark MX611dhe Monochrome Laser Printer                            11219.9340
HP Designjet T520 Inkjet Large Format Printer - 24" Color             8749.9500
GBC DocuBind TL300 Electric Binding System                            8521.4050
Riverside Palais Royal Lawyers Bookcase, Royale Cherry Finish         8298.8316
Name: Sales, dtype: float64
```

### 2. Most profitable cities

[80]:
```
data.groupby("City")["Profit"].sum().sort_values(ascending=False).head(5)
```

[80]:
```
City
New York City    26214.8443
Los Angeles      12456.9579
Lafayette         8915.0163
Seattle           7921.1530
San Francisco     7390.7013
Name: Profit, dtype: float64
```

### 3. Monthly Sales

[81]:
```
data.groupby("Month")["Sales"].sum()
```

[81]:
```
Month
1       29365.8146
2       32169.0160
3       57260.7674
4       58893.2035
5       53129.0038
6       61142.0400
7       48621.8990
8       50657.6457
9      125243.0076
10      78960.4915
11     150382.5500
12     142091.8725
Name: Sales, dtype: float64
```

### 4. Average discount given

[82]:
```
data["Discount"].mean()
```

[82]:
```
np.float64(0.15597723899059873)
```

# Thankyou.....