Importing of the dataset

```
[2] import pandas as pd
    df = pd.read_csv("/content/market_data.csv")
```

	Item_Identifier	Item_Weight	Item_Fat_Content	Item_Visibility	Item_Type	Item_MRP	Outlet_Identifier	Outlet_Establishment_Year	Outlet_Size	Outlet_Location_Type	Outlet_Type	Item_Out
0	FDA15	9.300	Low Fat	0.016047	Dairy	249.8092	OUT049	1999	Medium	Tier 1	Supermarket Type1	
1	DRC01	5.920	Regular	0.019278	Soft Drinks	48.2692	OUT018	2009	Medium	Tier 3	Supermarket Type2	
2	FDN15	17.500	Low Fat	0.016760	Meat	141.6180	OUT049	1999	Medium	Tier 1	Supermarket Type1	
3	FDX07	19.200	Regular	0.000000	Fruits and Vegetables	182.0950	OUT010	1998	NaN	Tier 3	Grocery Store	
4	NCD19	8.930	Low Fat	0.000000	Household	53.8614	OUT013	1987	High	Tier 3	Supermarket Type1	

Description of Dataset

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8523 entries, 0 to 8522
Data columns (total 12 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   Item_Identifier            8523 non-null   object
 1   Item_Weight                7060 non-null   float64
 2   Item_Fat_Content           8523 non-null   object
 3   Item_Visibility            8523 non-null   float64
 4   Item_Type                  8523 non-null   object
 5   Item_MRP                   8523 non-null   float64
 6   Outlet_Identifier          8523 non-null   object
 7   Outlet_Establishment_Year  8523 non-null   int64
 8   Outlet_Size                6113 non-null   object
 9   Outlet_Location_Type       8523 non-null   object
 10  Outlet_Type                8523 non-null   object
 11  Item_Outlet_Sales          8523 non-null   float64
dtypes: float64(4), int64(1), object(7)
memory usage: 799.2+ KB
```

Dropping unuseful columns

```
[ ] cols = ['Outlet_Establishment_Year']
    df = df.drop(cols, axis=1)
```

```
[ ] df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8523 entries, 0 to 8522
Data columns (total 11 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   Item_Identifier      8523 non-null   object
 1   Item_Weight          7060 non-null   float64
 2   Item_Fat_Content     8523 non-null   object
 3   Item_Visibility      8523 non-null   float64
 4   Item_Type            8523 non-null   object
 5   Item_MRP             8523 non-null   float64
 6   Outlet_Identifier    8523 non-null   object
 7   Outlet_Size          6113 non-null   object
 8   Outlet_Location_Type 8523 non-null   object
 9   Outlet_Type          8523 non-null   object
 10  Item_Outlet_Sales    8523 non-null   float64
dtypes: float64(4), object(7)
memory usage: 732.6+ KB
```

Taking Care of Missing Data

```
[3] df['Item_Weight'] = df['Item_Weight'].fillna(df['Item_Weight'].mean())
    df
```

	Item_Identifier	Item_Weight	Item_Fat_Content	Item_Visibility	Item_Type	Item_MRP	Outlet_Identifier	Outlet_Establishment_Year	Outlet_Size	Outlet_Location_Type	Outlet_Type	Item_Outlet_Sales
0	FDA15	9.300	Low Fat	0.016047	Dairy	249.8092	OUT049	1999	Medium	Tier 1	Supermarket Type1	3735.1380
1	DRC01	5.920	Regular	0.019278	Soft Drinks	48.2692	OUT018	2009	Medium	Tier 3	Supermarket Type2	443.4228
2	FDN15	17.500	Low Fat	0.016760	Meat	141.6180	OUT049	1999	Medium	Tier 1	Supermarket Type1	2097.2700
3	FDX07	19.200	Regular	0.000000	Fruits and Vegetables	182.0950	OUT010	1998	NaN	Tier 3	Grocery Store	732.3800
4	NCD19	8.930	Low Fat	0.000000	Household	53.8614	OUT013	1987	High	Tier 3	Supermarket Type1	994.7052

Creating the dummy data

```
dummy_data = {
    "Item_Identifier": "FDX01",
    "Item_Weight": 12.5,
    "Item_Fat_Content": "Low Fat",
    "Item_Visibility": 0.05,
    "Item_Type": "Snack Foods",
    "Item_MRP": 250.75,
    "Outlet_Identifier": "OUT013",
    "Outlet_Size": "Medium",
    "Outlet_Location_Type": "Tier 1",
    "Outlet_Type": "Supermarket Type1",
    "Item_Outlet_Sales": 3400.25
}
new_row = pd.DataFrame([dummy_data])
result = pd.concat([df, new_row], ignore_index=True)
```

```
result.tail()
```

	Item_Identifier	Item_Weight	Item_Fat_Content	Item_Visibility	Item_Type	Item_MRP	Outlet_Identifier	Outlet_Establishment_Year	Outlet_Size	Outlet_Location_Type	Outlet_Type
6109	FDF22	6.865	Low Fat	0.056783	Snack Foods	214.5218	OUT013	1987.0	High	Tier 3	Supermarket Type1
6110	NCJ29	10.600	Low Fat	0.035186	Health and Hygiene	85.1224	OUT035	2004.0	Small	Tier 2	Supermarket Type1
6111	FDN46	7.210	Regular	0.145221	Snack Foods	103.1332	OUT018	2009.0	Medium	Tier 3	Supermarket Type2
6112	DRG01	14.800	Low Fat	0.044878	Soft Drinks	75.4670	OUT046	1997.0	Small	Tier 1	Supermarket Type1
6113	FDX01	12.500	Low Fat	0.050000	Snack Foods	250.7500	OUT013	NaN	Medium	Tier 1	Supermarket Type1

Standardization and Normalization of the data

```
from sklearn.preprocessing import StandardScaler, MinMaxScaler
import pandas as pd


numerical_columns = ['Item_Weight', 'Item_Visibility', 'Item_MRP', 'Item_Outlet_Sales']

# Initialize scalers
standard_scaler = StandardScaler()
minmax_scaler = MinMaxScaler()

# Standardization of the data
standardized_data = standard_scaler.fit_transform(df[numerical_columns])
df_standardized = pd.DataFrame(standardized_data, columns=numerical_columns)

# Normalization of the data
normalized_data = minmax_scaler.fit_transform(df[numerical_columns])
df_normalized = pd.DataFrame(normalized_data, columns=numerical_columns)


print("Standardized Data:")
print(df_standardized.head())

print("\nNormalized Data:")
print(df_normalized.head())
```

```
Standardized Data:
   Item_Weight  Item_Visibility  Item_MRP  Item_Outlet_Sales
0    -0.881033        -0.967450  1.744524           0.811077
1    -1.710793        -0.902945 -1.494387          -1.079139
2     1.131996        -0.953220  0.005804          -0.129443
3    -0.971864        -1.287832 -1.404516          -0.762573
4    -0.612220        -1.287832 -1.444060          -1.014143

Normalized Data:
   Item_Weight  Item_Visibility  Item_MRP  Item_Outlet_Sales
0     0.282525         0.048866  0.927507           0.283550
1     0.081274         0.058705  0.072068           0.031370
2     0.770765         0.051037  0.468288           0.158072
3     0.260494         0.000000  0.095805           0.073604
4     0.347723         0.000000  0.085361           0.040041
```

Finding Outliners (manually)

```
import matplotlib.pyplot as plt

x = df['Item_Visibility']
y = df['Item_Outlet_Sales']

plt.figure(figsize=(3, 3))
plt.scatter(x, y, c="green", s=80, alpha=0.6, edgecolor="black", marker="^")
plt.title("Scatter Plot: Item Visibility vs. Item Outlet Sales", fontsize=14)
plt.xlabel("Item Visibility", fontsize=12)
plt.ylabel("Item Outlet Sales", fontsize=12)
plt.grid(True, linestyle="--", alpha=0.5)
plt.show()
```