In [1]:
```python
import pandas as pd
import numpy as np
```

In [2]:
```python
df=pd.read_csv(r"C:\Users\lenovo\Downloads\train_v9rqX0R.csv")
df.head()
```

Out[2]:

	Item_Identifier	Item_Weight	Item_Fat_Content	Item_Visibility	Item_Type	Item_MRP	Outlet_Identifier	Outlet_Establishment_Year	Outlet_Size	Out
0	FDA15	9.30	Low Fat	0.016047	Dairy	249.8092	OUT049	1999	Medium	
1	DRC01	5.92	Regular	0.019278	Soft Drinks	48.2692	OUT018	2009	Medium	
2	FDN15	17.50	Low Fat	0.016760	Meat	141.6180	OUT049	1999	Medium	
3	FDX07	19.20	Regular	0.000000	Fruits and Vegetables	182.0950	OUT010	1998	NaN	
4	NCD19	8.93	Low Fat	0.000000	Household	53.8614	OUT013	1987	High	

In [3]:
```python
df.isnull().sum().nlargest()
```

Out[3]:
```
Outlet_Size         2410
Item_Weight         1463
Item_Identifier        0
Item_Fat_Content       0
Item_Visibility        0
dtype: int64
```

In [4]:
```
df.dtypes
```

Out[4]:
```
Item_Identifier               object
Item_Weight                  float64
Item_Fat_Content              object
Item_Visibility              float64
Item_Type                     object
Item_MRP                     float64
Outlet_Identifier             object
Outlet_Establishment_Year      int64
Outlet_Size                   object
Outlet_Location_Type          object
Outlet_Type                   object
Item_Outlet_Sales            float64
dtype: object
```

In [5]:
```
mode=df["Outlet_Size"].mode()
mode[0]
df["Outlet_Size"].replace(np.nan,mode[0],inplace=True)
```

In [6]:
```
mean1=df["Item_Weight"].mean()
df["Item_Weight"].replace(np.nan,mean1,inplace=True)
```

In [7]:
```
df.shape
```

Out[7]:
```
(8523, 12)
```

In [8]:
```
df.isnull().sum()
```

Out[8]:
```
Item_Identifier              0
Item_Weight                  0
Item_Fat_Content             0
Item_Visibility              0
Item_Type                    0
Item_MRP                     0
Outlet_Identifier            0
Outlet_Establishment_Year    0
Outlet_Size                  0
Outlet_Location_Type         0
Outlet_Type                  0
Item_Outlet_Sales            0
dtype: int64
```

In [9]:
```
print(df["Item_Fat_Content"].unique())
```

```
['Low Fat' 'Regular' 'low fat' 'LF' 'reg']
```

In [10]:
```
df["Item_Fat_Content"].replace(("low Fat","LF"),"Low Fat",inplace=True)
```

In [11]:
```
df["Item_Fat_Content"].replace("reg","Regular",inplace=True)
```

In [12]:
```
df.groupby("Outlet_Identifier").count()
```

Out[12]:

Outlet_Identifier	Item_Identifier	Item_Weight	Item_Fat_Content	Item_Visibility	Item_Type	Item_MRP	Outlet_Establishment_Year	Outlet_Size	Outlet_
OUT010	555	555	555	555	555	555	555	555	
OUT013	932	932	932	932	932	932	932	932	
OUT017	926	926	926	926	926	926	926	926	
OUT018	928	928	928	928	928	928	928	928	
OUT019	528	528	528	528	528	528	528	528	
OUT027	935	935	935	935	935	935	935	935	
OUT035	930	930	930	930	930	930	930	930	
OUT045	929	929	929	929	929	929	929	929	
OUT046	930	930	930	930	930	930	930	930	
OUT049	930	930	930	930	930	930	930	930	

In [13]:
```
df.groupby("Outlet_Location_Type").count()
```

Out[13]:

Outlet_Location_Type	Item_Identifier	Item_Weight	Item_Fat_Content	Item_Visibility	Item_Type	Item_MRP	Outlet_Identifier	Outlet_Establishment_Yea
Tier 1	2388	2388	2388	2388	2388	2388	2388	238
Tier 2	2785	2785	2785	2785	2785	2785	2785	278
Tier 3	3350	3350	3350	3350	3350	3350	3350	335

In [14]:
```
df.groupby("Item_Weight").count()
```

Out[14]:

Item_Weight	Item_Identifier	Item_Fat_Content	Item_Visibility	Item_Type	Item_MRP	Outlet_Identifier	Outlet_Establishment_Year	Outlet_Size	Outlet_
4.555	4	4	4	4	4	4	4	4	
4.590	5	5	5	5	5	5	5	5	
4.610	7	7	7	7	7	7	7	7	
4.615	4	4	4	4	4	4	4	4	
4.635	5	5	5	5	5	5	5	5	
...	...	...	...	...	...	...	...	...	
21.000	6	6	6	6	6	6	6	6	
21.100	17	17	17	17	17	17	17	17	
21.200	5	5	5	5	5	5	5	5	
21.250	24	24	24	24	24	24	24	24	
21.350	7	7	7	7	7	7	7	7	

416 rows × 11 columns

In [15]:
```
a=0
b=0
for i in range (len(df["Item_Fat_Content"])):
    if (df["Item_Fat_Content"][i]=="Low Fat"):
        a=a+(df["Item_Weight"][i])
    else:
        b=b+(df["Item_Weight"][i])
print("low fat",a)
print("regular fat",b)
```

```
low fat 69884.34150495697
regular fat 39701.368399433115
```

In [16]:
```
df.groupby("Item_Fat_Content")["Item_Weight"].sum()
```

Out[16]:
```
Item_Fat_Content
Low Fat    69884.341505
Regular    38285.979334
low fat     1415.389065
Name: Item_Weight, dtype: float64
```

In [17]:
```
df.describe()
```

Out[17]:

	Item_Weight	Item_Visibility	Item_MRP	Outlet_Establishment_Year	Item_Outlet_Sales
count	8523.000000	8523.000000	8523.000000	8523.000000	8523.000000
mean	12.857645	0.066132	140.992782	1997.831867	2181.288914
std	4.226124	0.051598	62.275067	8.371760	1706.499616
min	4.555000	0.000000	31.290000	1985.000000	33.290000
25%	9.310000	0.026989	93.826500	1987.000000	834.247400
50%	12.857645	0.053931	143.012800	1999.000000	1794.331000
75%	16.000000	0.094585	185.643700	2004.000000	3101.296400
max	21.350000	0.328391	266.888400	2009.000000	13086.964800

In [18]:
```
obj=df.groupby("Outlet_Location_Type")
obj
```

Out[18]:
```
<pandas.core.groupby.generic.DataFrameGroupBy object at 0x000001B9CCC83FD0>
```

In [19]:
```
obj.groups
```

Out[19]:
```
{'Tier 1': [0, 2, 10, 11, 12, 13, 15, 17, 23, 24, 29, 34, 35, 40, 42, 48, 49, 50, 57, 58, 59, 63, 69, 70, 74, 75, 76, 7
7, 80, 81, 83, 88, 89, 91, 95, 96, 99, 102, 108, 110, 112, 115, 126, 131, 135, 143, 145, 154, 163, 164, 178, 182, 186,
187, 189, 190, 191, 195, 196, 197, 204, 206, 208, 220, 222, 225, 227, 234, 236, 248, 250, 252, 255, 270, 274, 284, 289,
295, 297, 299, 301, 308, 311, 312, 321, 324, 334, 336, 344, 345, 346, 347, 348, 353, 354, 355, 356, 358, 361, 363,
...], 'Tier 2': [8, 9, 19, 22, 25, 26, 33, 46, 47, 53, 54, 56, 61, 66, 67, 68, 72, 73, 78, 79, 85, 86, 92, 93, 94, 97,
100, 107, 111, 114, 116, 117, 118, 120, 121, 123, 124, 125, 127, 129, 137, 138, 140, 141, 142, 144, 146, 147, 148, 149,
150, 157, 158, 165, 166, 170, 171, 176, 179, 181, 188, 192, 200, 201, 202, 207, 210, 211, 212, 213, 219, 221, 223, 228,
232, 233, 240, 241, 242, 243, 244, 245, 247, 249, 254, 256, 258, 259, 261, 262, 263, 264, 268, 273, 277, 281, 283, 285,
288, 290, ...], 'Tier 3': [1, 3, 4, 5, 6, 7, 14, 16, 18, 20, 21, 27, 28, 30, 31, 32, 36, 37, 38, 39, 41, 43, 44, 45, 5
1, 52, 55, 60, 62, 64, 65, 71, 82, 84, 87, 90, 98, 101, 103, 104, 105, 106, 109, 113, 119, 122, 128, 130, 132, 133, 13
4, 136, 139, 151, 152, 153, 155, 156, 159, 160, 161, 162, 167, 168, 169, 172, 173, 174, 175, 177, 180, 183, 184, 185, 1
93, 194, 198, 199, 203, 205, 209, 214, 215, 216, 217, 218, 224, 226, 229, 230, 231, 235, 237, 238, 239, 246, 251, 253,
257, 260, ...]}
```

In [20]:
```
for name ,group in obj:
    print(name,"contains",group.shape[0],"rows")
```

```
Tier 1 contains 2388 rows
Tier 2 contains 2785 rows
Tier 3 contains 3350 rows
```

In [21]:
```
obj.get_group("Tier 1")["Item_Weight"].sum()
```

Out[21]:
```
30768.186657223905
```

In [22]:
```
obj.agg([np.mean,np.median,np.sum])
```

```
C:\Users\lenovo\AppData\Local\Temp\ipykernel_103352\2897686275.py:1: FutureWarning: ['Item_Identifier', 'Item_Fat_Conte
nt', 'Item_Type', 'Outlet_Identifier', 'Outlet_Size', 'Outlet_Type'] did not aggregate successfully. If any error is ra
ised this will raise in a future version of pandas. Drop these columns/ops to avoid this warning.
  obj.agg([np.mean,np.median,np.sum])
```

Out[22]:

	Item_Weight			Item_Visibility			Item_MRP			Outlet_Establishment_	
	mean	median	sum	mean	median	sum	mean	median	sum	mean	median
Outlet_Location_Type											
Tier 1	12.884500	12.857645	30768.186657	0.071205	0.056450	170.038072	140.870106	143.2641	336397.8120	1995.125628	1997.0
Tier 2	12.768628	12.500000	35560.630000	0.061038	0.051766	169.990299	141.167196	143.2812	393150.6416	2004.330341	2004.0
Tier 3	12.912505	12.857645	43256.893247	0.066751	0.053906	223.614910	140.935232	142.2483	472133.0272	1994.358507	1987.0

In [23]:
```
obj.agg(max)
```

Out[23]:

	Item_Identifier	Item_Weight	Item_Fat_Content	Item_Visibility	Item_Type	Item_MRP	Outlet_Identifier	Outlet_Establishment_Yea
Outlet_Location_Type								
Tier 1	NCZ54	21.35	low fat	0.328391	Starchy Foods	266.8884	OUT049	199
Tier 2	NCZ54	21.35	low fat	0.188620	Starchy Foods	266.8884	OUT045	200
Tier 3	NCZ54	21.35	low fat	0.311090	Starchy Foods	266.6884	OUT027	200

In [24]:
```
obj.agg(min)
```

Out[24]:

Outlet_Location_Type	Item_Identifier	Item_Weight	Item_Fat_Content	Item_Visibility	Item_Type	Item_MRP	Outlet_Identifier	Outlet_Establishment_Yea
Tier 1	DRA24	4.555	Low Fat	0.0	Baking Goods	32.4558	OUT019	198
Tier 2	DRA12	4.555	Low Fat	0.0	Baking Goods	32.0558	OUT017	200
Tier 3	DRA12	4.555	Low Fat	0.0	Baking Goods	31.2900	OUT010	198

In [25]:
```
ff=df.groupby(["Outlet_Location_Type","Outlet_Establishment_Year"])
```

In [26]:
```
ff.agg({"Outlet_Size":pd.Series.mode,"Item_Outlet_Sales":np.mean})
```

Out[26]:

Outlet_Location_Type	Outlet_Establishment_Year	Outlet_Size	Item_Outlet_Sales
Tier 1	**1985**	Small	340.329723
	1997	Small	2277.844267
	1999	Medium	2348.354635
Tier 2	**2002**	Medium	2192.384798
	2004	Small	2438.841866
	2007	Medium	2340.675263
Tier 3	**1985**	Medium	3694.038558
	1987	High	2298.995256
	1998	Medium	339.351662
	2009	Medium	1995.498739

In [27]:
```
ff.agg({"Item_MRP":np.mean,"Item_Outlet_Sales":np.mean})
```

Out[27]:

		Item_MRP	Item_Outlet_Sales
Outlet_Location_Type	**Outlet_Establishment_Year**		
Tier 1	**1985**	139.787088	340.329723
	1997	142.057387	2277.844267
	1999	140.297699	2348.354635
Tier 2	**2002**	140.950246	2192.384798
	2004	143.122481	2438.841866
	2007	139.421119	2340.675263
Tier 3	**1985**	139.801791	3694.038558
	1987	141.425982	2298.995256
	1998	140.777594	339.351662
	2009	141.678634	1995.498739

In [28]:
```
ff.get_group(("Tier 2",2002))["Outlet_Size"]
```

Out[28]:
```
8       Medium
33      Medium
46      Medium
47      Medium
56      Medium
         ...
8483    Medium
8502    Medium
8508    Medium
8514    Medium
8519    Medium
Name: Outlet_Size, Length: 929, dtype: object
```

In [29]:
```
df.groupby(["Outlet_Type","Item_Type"]).agg(mean_MRP=("Item_MRP",np.mean),mean_sales=("Item_Outlet_Sales",np.mean))
```

Out[29]:

Outlet_Type	Item_Type	mean_MRP	mean_sales
Grocery Store	Baking Goods	126.438068	292.082544
	Breads	146.452873	381.967442
	Breakfast	147.026989	412.831042
	Canned	138.080808	352.864879
	Dairy	147.166715	341.866589
...	...	...	...
Supermarket Type3	Others	106.779053	2700.928667
	Seafood	124.028286	2687.073686
	Snack Foods	144.574508	3745.168739
	Soft Drinks	123.313587	3284.938836
	Starchy Foods	143.078386	3512.190114

64 rows × 2 columns

In [30]:
```
df["Item_Weight"]=df.groupby(["Item_Fat_Content","Item_Type"])["Item_Weight"].transform(lambda x:x.fillna(x.mean()))
```

In [31]:
```
def filter_fun(x):
    return x["Item_Weight"].std()<3
df_filter=df.groupby(["Item_Weight"]).filter(filter_fun)
df_filter.shape
```

Out[31]: (8519, 12)