

Blinkit Sales Analysis – Python Data Analytics Project

Project Overview

This project performs an end-to-end **Exploratory Data Analysis (EDA)** on Blinkit's sales dataset using Python. The primary goal is to uncover actionable insights about product performance, outlet characteristics, sales trends, and customer preferences.

The project successfully demonstrates:

- Data cleaning and preprocessing
- Descriptive statistics
- Visual EDA using Matplotlib & Seaborn
- KPI calculations
- Actionable business insights

Tech Stack

The following technologies and libraries were utilized for the analysis:

- **Python**
 - **Pandas**
 - **NumPy**
 - **Matplotlib**
 - **Seaborn**
 - **Jupyter Notebook / VS Code**
-

Key Performance Indicators (KPIs)

The core sales metrics calculated from the dataset are:

Metric	Value
Total Sales	1,201,681
Average Sales per Item	141
Items Sold	8,523
Average Rating	4.0

Key Insights

1. Fat Content

- **Key Insight:** Low Fat items generate **64.6%** of total sales, indicating that health-oriented products show significantly higher demand among customers.

2. Item Type

- **Top-selling categories:**
 - Fruits & Vegetables
 - Snack Foods
 - Household Items
- **Low-selling categories:**
 - Seafood
 - Breakfast
- **Key Insight:** Top-selling categories like **Fruits & Vegetables**, **Snack Foods**, and **Household Items** indicate strong demand for daily essentials. Low performance in **Seafood** and **Breakfast** suggests an opportunity to improve promotion, pricing, or product availability for these categories.

3. Fat Content by Outlet for Total Sales

- **Key Insight:** Low Fat items consistently generate higher sales than **Regular** items across all outlet tiers. **Tier 3** outlets show the highest demand for both categories, highlighting strong consumer preference in smaller markets.

4. Outlet Establishment Year

- **Finding:** Older outlets (**1998–2000**) show stronger and more stable sales.
- **Key Insight:** Older outlets, especially those established before 2000, show significantly higher sales compared to newer ones. The sharp dip in **2011** highlights performance issues in outlets opened that year, indicating a need for review or improvement.

5. Outlet Size

- **Finding:** Medium outlets perform best, while **High**-size outlets underperform.
- **Key Insight:** Medium outlets contribute the highest share of sales, indicating optimal customer traffic and efficiency. **High**-size outlets show the lowest share, suggesting underutilized space or weaker sales performance.

6. Outlet Location Performance

- **Finding:** Tier 3 outlets generate the highest revenue, followed by Tier 2 and Tier 1.
 - **Key Insight:** Tier 3 outlets lead in revenue, indicating strong sales momentum in smaller markets. Tier 2 and Tier 1 outlets follow, suggesting potential to enhance presence and performance in higher-tier locations.
-

Visualizations Included

The following key visualizations were generated to support the findings:

- Sales by Fat Content (Pie Chart)
 - Sales by Item Type (Bar Graph)
 - Outlet Tier vs Fat Content (Bar Chart)
 - Sales by Establishment Year (Line Chart)
 - Sales by Outlet Size (Pie Chart)
 - Sales by Outlet Location Type (Bar Chart)
-

Data Cleaning Steps

The following critical steps were taken to ensure data quality and consistency:

- Standardized “**Item Fat Content**” to handle inconsistent labels (e.g., converting 'LF', 'low fat' to 'Low Fat').
 - Checked data types and handled missing values.
 - Formatted categorical columns for proper analysis.
-

Conclusion

The analysis provides crucial insights into customer behavior, product demand, and outlet performance. It helps Blinkit to:

- **Optimize Inventory:** Prioritize stocking and marketing for **Low Fat** items and top categories like **Fruits & Vegetables** and **Snack Foods**.
- **Improve Strategy:** Review sales practices in underperforming categories (**Seafood**, **Breakfast**) and underperforming outlets (e.g., those from **2011**, or **High-Size** locations).
- **Strengthen Outlet Planning:** Leverage the success of **Tier 3** and **Medium-Size** outlets to inform future expansion and operational strategies.

◆ DATA ANALYSISI PYTHON PROJECT- BLINKIT ANALYSIS ◆

✧ Import Libaries

```
In [6]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

✧ Import Raw Data

```
In [7]: df = pd.read_csv("blinkit_data.csv")
```

✧ Sample Data

```
In [8]: df.head(5)
```

	Item Fat Content	Item Identifier	Item Type	Outlet Establishment Year	Outlet Identifier	Outlet Location Type	Outlet Size	Outlet Type	Item Visibility	Item Weight
0	Regular	FDX32	Fruits and Vegetables	2012	OUT049	Tier 1	Medium	Supermarket Type1	0.100014	15.10
1	Low Fat	NCB42	Health and Hygiene	2022	OUT018	Tier 3	Medium	Supermarket Type2	0.008596	11.80
2	Regular	FDR28	Frozen Foods	2010	OUT046	Tier 1	Small	Supermarket Type1	0.025896	13.85
3	Regular	FDL50	Canned	2000	OUT013	Tier 3	High	Supermarket Type1	0.042278	12.15
4	Low Fat	DRI25	Soft Drinks	2015	OUT045	Tier 2	Small	Supermarket Type1	0.033970	19.60

```
In [9]: df.tail(5)
```

	Item Fat Content	Item Identifier	Item Type	Outlet Establishment Year	Outlet Identifier	Outlet Location Type	Outlet Size	Outlet Type	Item Visibility	Item Weight
8518	low fat	NCT53	Health and Hygiene	1998	OUT027	Tier 3	Medium	Supermarket Type3	0.000000	NaN
8519	low fat	FDN09	Snack Foods	1998	OUT027	Tier 3	Medium	Supermarket Type3	0.034706	NaN
8520	low fat	DRE13	Soft Drinks	1998	OUT027	Tier 3	Medium	Supermarket Type3	0.027571	NaN
8521	reg	FDT50	Dairy	1998	OUT027	Tier 3	Medium	Supermarket Type3	0.107715	NaN
8522	reg	FDM58	Snack Foods	1998	OUT027	Tier 3	Medium	Supermarket Type3	0.000000	NaN

✧ Size of Data

```
In [10]: print("Size of data:", df.shape)
```

```
Size of data: (8523, 12)
```

```
In [11]: df.columns
```

```
Out[11]: Index(['Item Fat Content', 'Item Identifier', 'Item Type',
   'Outlet Establishment Year', 'Outlet Identifier',
   'Outlet Location Type', 'Outlet Size', 'Outlet Type', 'Item Visibility',
   'Item Weight', 'Sales', 'Rating'],
  dtype='object')
```

✧ Data Type

```
In [12]: df.dtypes
```

```
Out[12]: Item Fat Content          object
Item Identifier           object
Item Type                  object
Outlet Establishment Year  int64
Outlet Identifier         object
Outlet Location Type      object
Outlet Size                object
Outlet Type                object
Item Visibility            float64
Item Weight                float64
Sales                      float64
Rating                     float64
dtype: object
```

✧ Data Cleaning

```
In [13]: print(df['Item Fat Content'].unique())
```

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

```
In [14]: df['Item Fat Content'] = df['Item Fat Content'].replace({
    'LF' : 'Low Fat',
    'low fat' : 'Low Fat',
    'reg' : 'Regular'
})
```

```
In [15]: print(df['Item Fat Content'].unique())
```

```
['Regular' 'Low Fat']
```

✧ BUSINESS REQUIREMENTS

✧ KPI's REQUIREMENTS

```
In [16]: total_sales = df['Sales'].sum()
```

```
avg_sales = df['Sales'].mean()
```

```
no_of_items_sold = df['Sales'].count()
```

```
avg_ratings = df['Rating'].mean()

print(f"Total Sales: {total_sales:.0f}")
print(f"Average Sales: {avg_sales:.1f}")
print(f"No of Items Sold: {no_of_items_sold:.0f}")
print(f"Average Rating: {avg_ratings:.1f}")
```

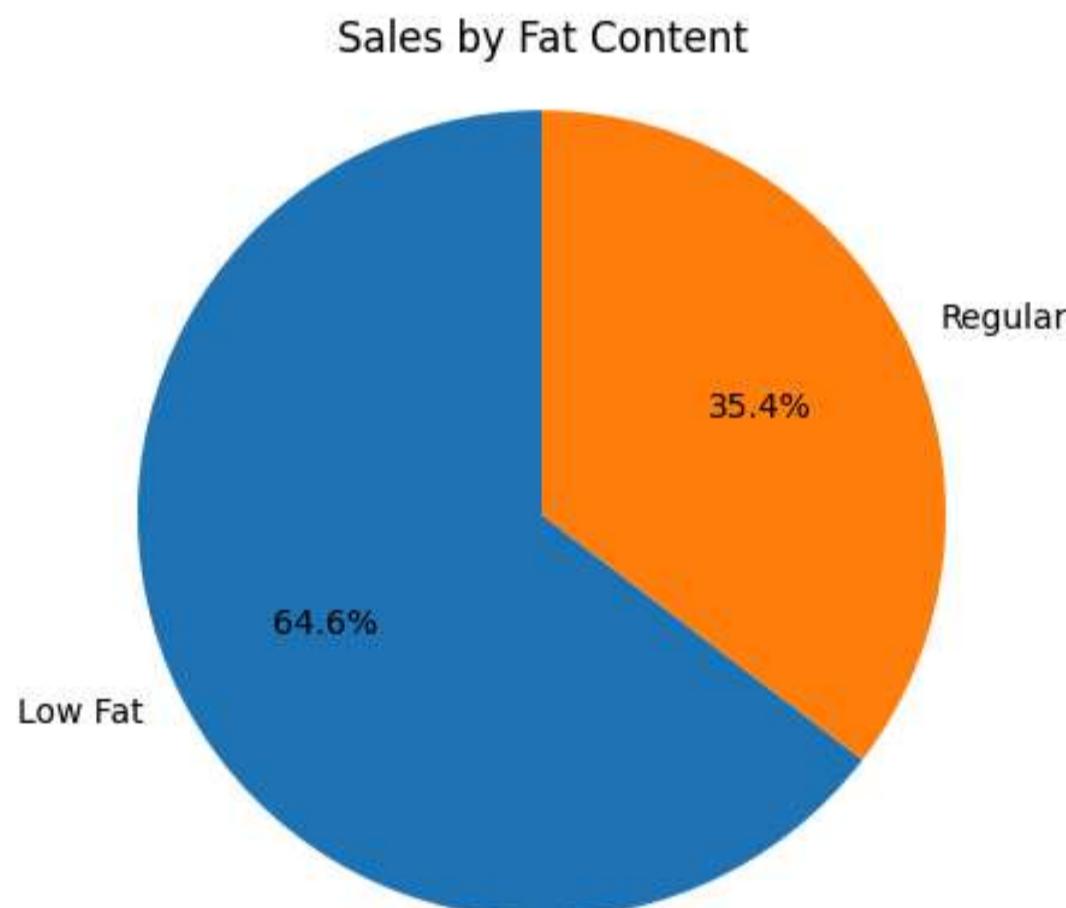
Total Sales: 1201681
 Average Sales: 141.0
 No of Items Sold: 8523
 Average Rating: 4.0

❖ CHART REQUIREMENTS

Q1. Total Sales By Fat Content

```
In [17]: sales_by_fat = df.groupby('Item Fat Content')['Sales'].sum()

plt.pie(
    sales_by_fat,
    labels=sales_by_fat.index,
    autopct='%.1f%%',
    startangle=90
)
plt.title("Sales by Fat Content")
plt.axis('equal')
plt.show()
```



Key Insight:

- Low Fat items generate 64.6% of total sales.
 - Health-oriented products show higher demand.
-

Q2. Total Sales By Item Type

```
In [18]: sales_by_type = df.groupby('Item Type')['Sales'].sum().sort_values(ascending=False)
```

```

plt.figure(figsize=(10, 6))

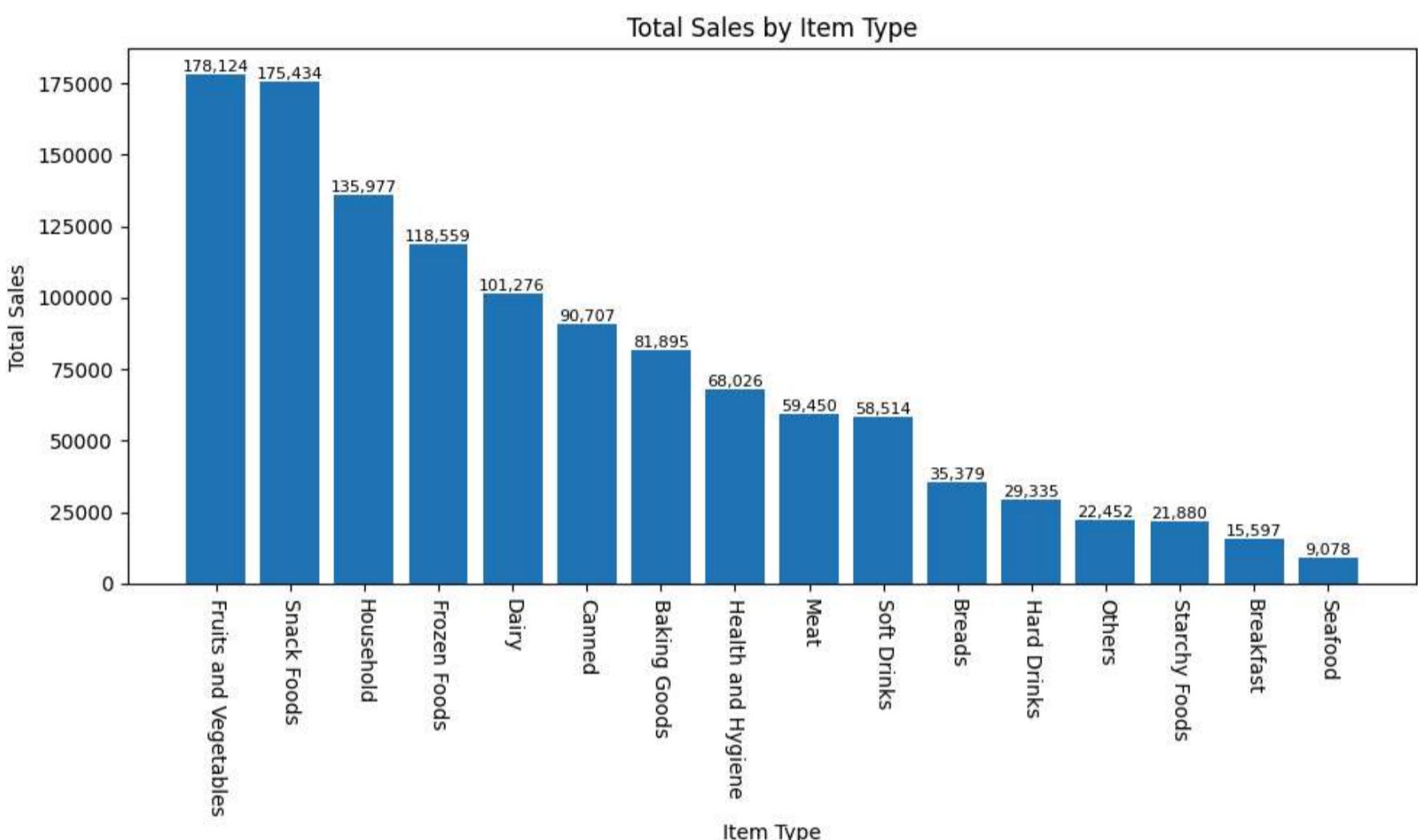
bars = plt.bar(sales_by_type.index, sales_by_type.values)

plt.xticks(rotation=-90)

plt.xlabel('Item Type')
plt.ylabel('Total Sales')
plt.title('Total Sales by Item Type')

for bar in bars:
    plt.text(
        bar.get_x() + bar.get_width() / 2,
        bar.get_height(),
        f'{bar.get_height():,.0f}',
        ha='center', va='bottom', fontsize=8
    )
plt.tight_layout()
plt.show()

```



Key Insight:

- Top-selling categories like `Fruits & Vegetables`, `Snack Foods`, and `Household Items` indicate strong demand for daily essentials.
- Low performance in `Seafood` and `Breakfast` suggests an opportunity to improve promotion, pricing, or product availability.

Q3. Fat Content by Outlet for Total Sales

```

In [19]: grouped = df.groupby(['Outlet Location Type', 'Item Fat Content'])['Sales'].sum().unstack()

grouped = grouped[['Regular', 'Low Fat']]

ax = grouped.plot(kind='bar', figsize=(8,5), title='Outlet Tier by Item Fat Content')

plt.xlabel('Outlet Location Tire')
plt.ylabel('Total Sales')
plt.legend(title='Item Fat Content')

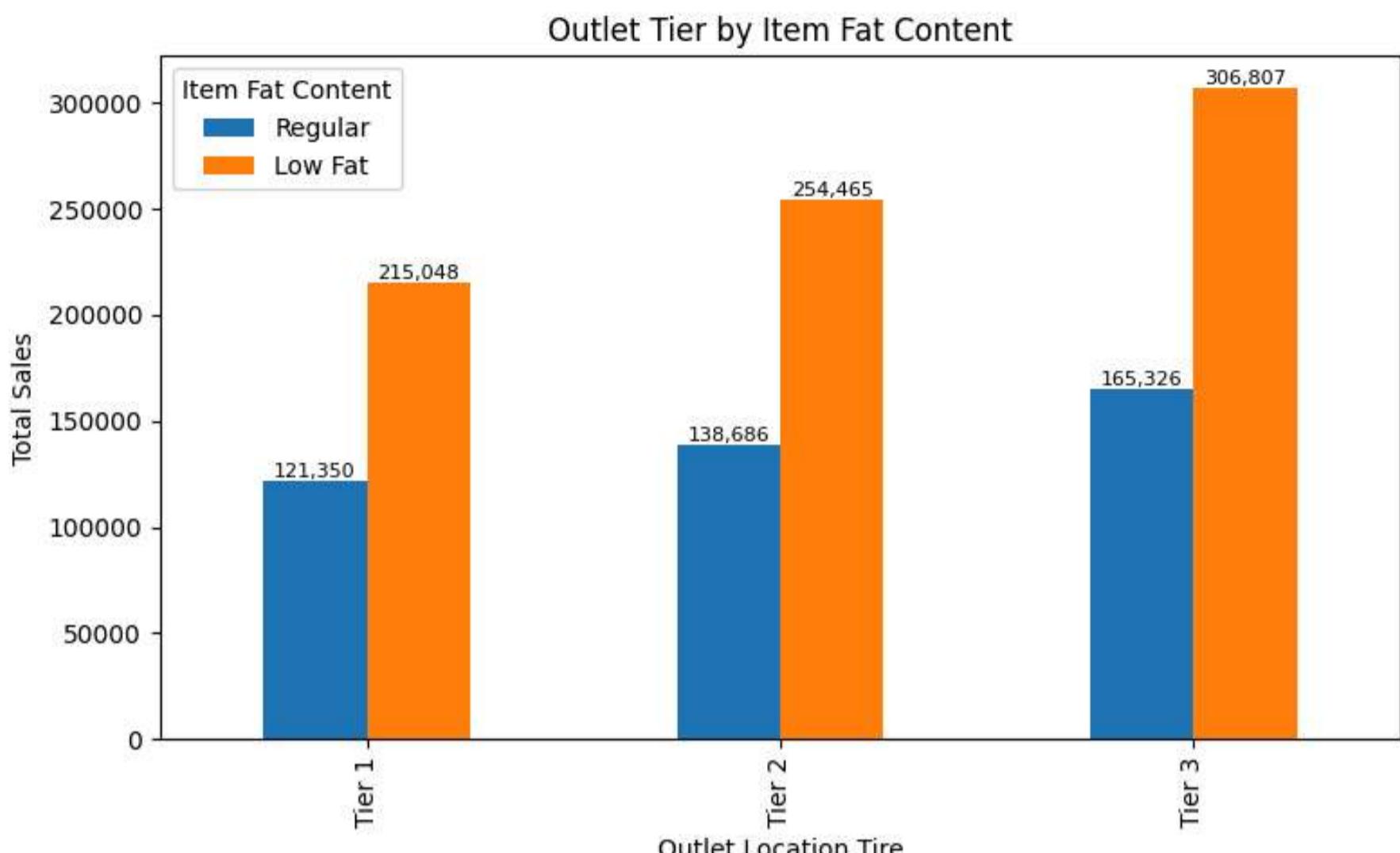
for bars in ax.containers:

```

```

for bar in bars:
    ax.text(
        bar.get_x() + bar.get_width()/2,
        bar.get_height(),
        f'{bar.get_height():,.0f}',
        ha='center', va='bottom', fontsize=8
    )
plt.tight_layout()
plt.show()

```



Key Insight:

- Low Fat items consistently generate higher sales than Regular items across all outlet tiers.
- Tier 3 outlets show the highest demand for both categories, highlighting strong consumer preference in smaller markets.

Q4. Total Sales by Outlet Establishment

```

In [24]: sales_by_year = df.groupby('Outlet Establishment Year')['Sales'].sum().sort_index()

plt.figure(figsize=(9,5))

plt.plot(sales_by_year.index, sales_by_year.values, marker='o', linestyle='--')

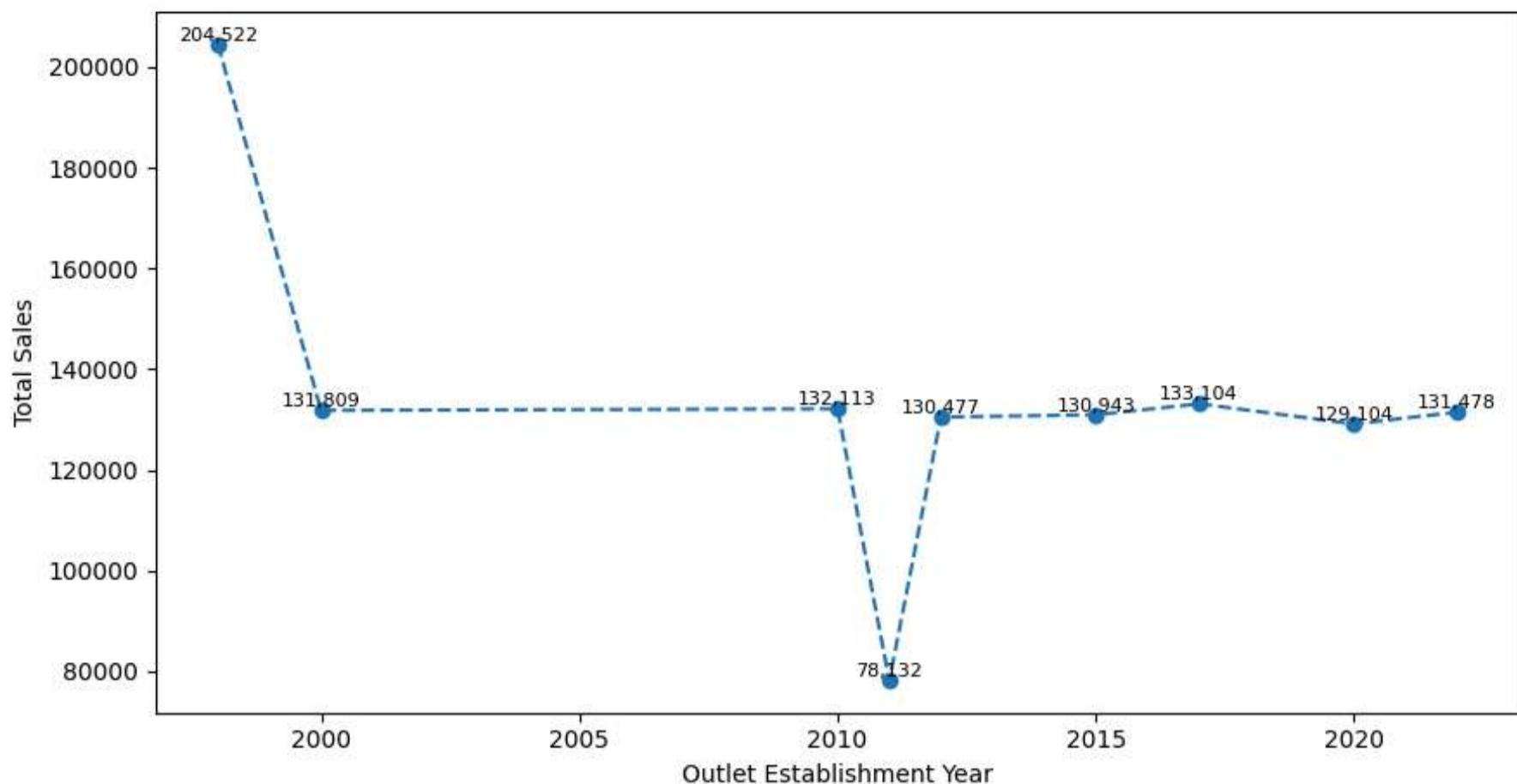
plt.xlabel('Outlet Establishment Year')
plt.ylabel('Total Sales')
plt.title('Outlet Establishment')

for x, y in zip(sales_by_year.index, sales_by_year.values):
    plt.text(x, y, f'{y:.0f}', ha='center', va='bottom', fontsize=8)

plt.tight_layout()
plt.show()

```

Outlet Establishment



Key Insight:

- Older outlets, especially those established before 2000, show significantly higher sales compared to newer ones.
- The sharp dip in 2011 highlights performance issues in outlets opened that year, indicating a need for review or improvement.

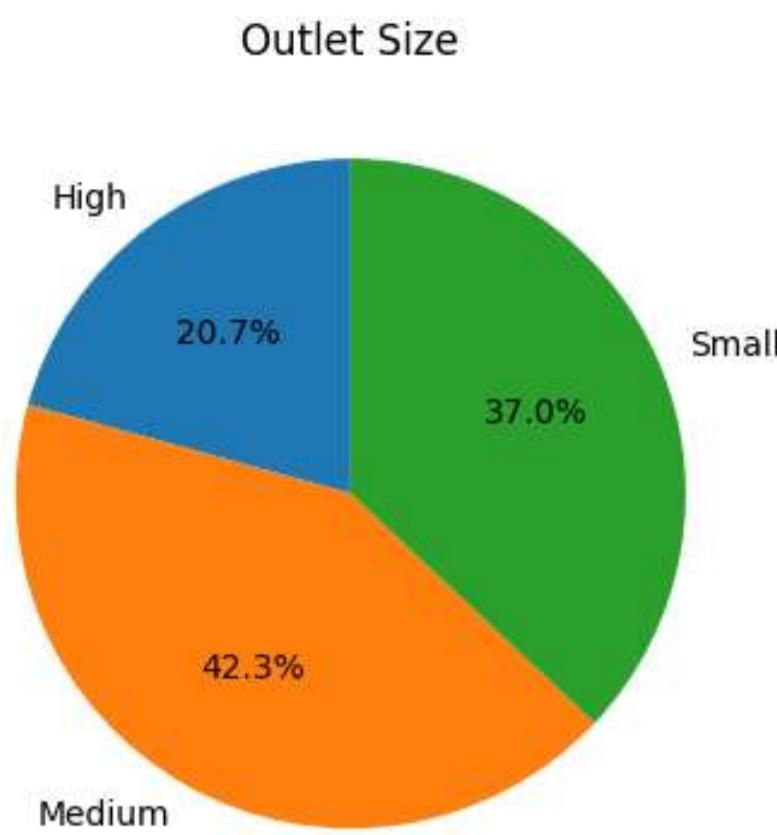
Q5. Sales by Outlet Size

```
In [27]: sales_by_size = df.groupby('Outlet Size')['Sales'].sum()

plt.figure(figsize=(4, 4))

plt.pie(
    sales_by_size,
    labels=sales_by_size.index,
    autopct='%1.1f%%',
    startangle=90
)

plt.title('Outlet Size')
plt.tight_layout()
plt.show()
```



Key Insight:

- Medium outlets contribute the highest share of sales, indicating optimal customer traffic and efficiency.
- High-size outlets show the lowest share, suggesting underutilized space or weaker sales performance.

Q6. Sales by Outlet Location

```
In [22]: sales_by_location = df.groupby('Outlet Location Type')['Sales'].sum().reset_index()

sales_by_location = sales_by_location.sort_values('Sales', ascending=False)

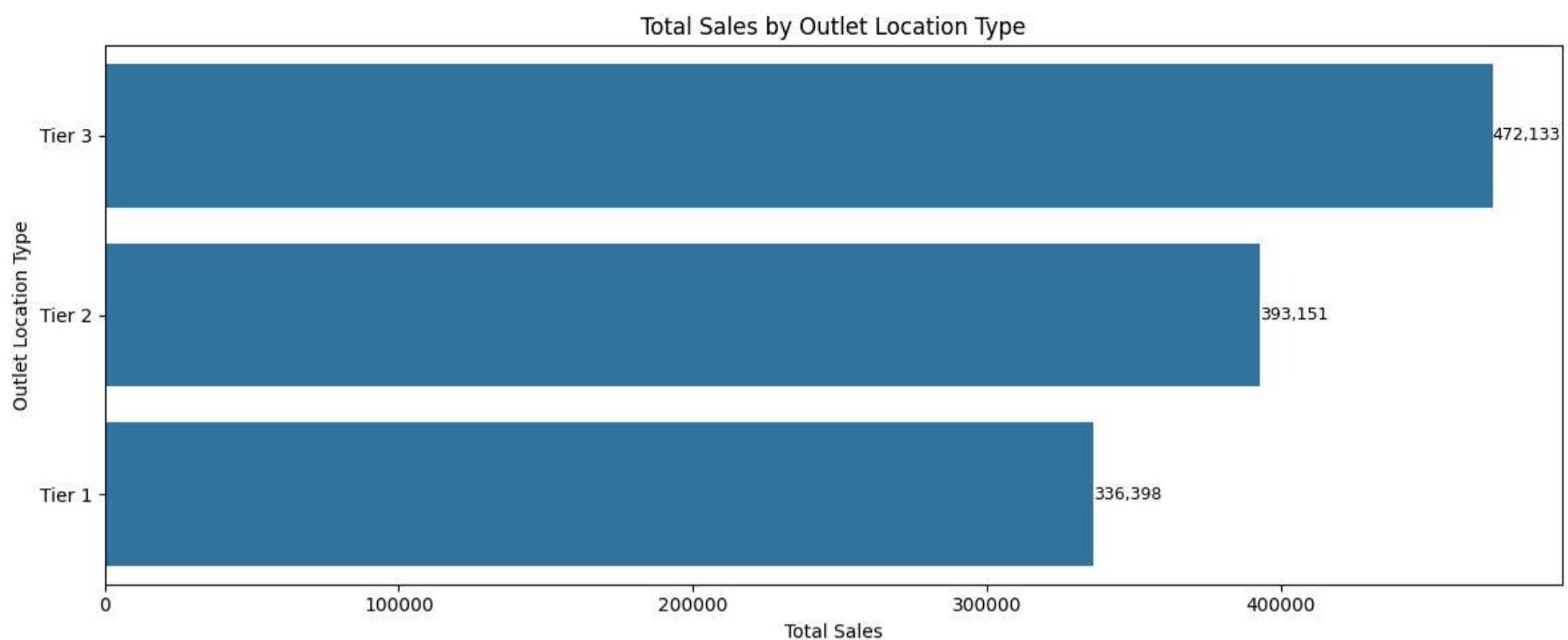
plt.figure(figsize=(12, 5))

ax = sns.barplot(
    x='Sales',
    y='Outlet Location Type',
    data=sales_by_location,
)

for i, value in enumerate(sales_by_location['Sales']):
    ax.text(
        value, i,
        f'{value:.0f}',
        va='center', ha='left', fontsize=9
    )

plt.title('Total Sales by Outlet Location Type')
plt.xlabel('Total Sales')
plt.ylabel('Outlet Location Type')

plt.tight_layout()
plt.show()
```

**Key Insight:**

- Tier 3 outlets lead in revenue, indicating strong sales momentum in smaller markets.
 - Tier 2 and Tier 1 outlets follow, suggesting potential to enhance presence and performance in higher-tier locations.
-
-