Retail Data Analytics – Sales & Store Performance Analysis

1.Project Overview

Retail companies generate large volumes of sales, promotion, and store-related data.

To make the right business decisions, it is important to analyze this data and understand store performance, customer behavior, and the factors influencing sales.

This project analyzes the Retail Data Analytics dataset using Excel, SQL, Python, and Power BI to identify key insights.

2. Business Problem

The retail company wants to understand why sales vary across different stores, departments, holidays, and seasons.

They are facing challenges like:

- Why sales vary across stores and departments
- How holidays and promotions affect revenue
- Which stores are high-performing or low-performing
- How economic factors (fuel price, CPI, unemployment) impact sales
- What seasonal trends exist in retail demand

The goal is to solve these problems using data analysis and provide clear insights to improve business decisions.

3.Dataset Details

The dataset contains three tables:

3.1 Sales Table

- Store
- Department
- Date
- Weekly_Sales
- IsHoliday

3.2 Features Table

- Store
- Date
- Temperature
- Fuel _Price
- CPI
- Unemployment
- MarkDown1 to MarkDown5

3.3 Stores Table

- Store
- Type
- Size

4. Tools Used

- Excel – Data cleaning & validation
- SQL – Data verification, joins, and complex queries
- Python – Exploratory Data Analysis (EDA)
- Power BI – Dashboard and insights visualization

5. Data Cleaning (Excel +Python)

Data Cleaning in Excel:

- Removed duplicate rows
- Handled missing values in MarkDown columns (replaced with 0)
- Cleaned date format and ensured consistency
- Corrected data types where needed
- Ensured store numbers and department IDs match correctly

Data Cleaning in Python (Pandas):

- Handle missing values
- Remove duplicate rows
- Corrected data types
- Merge all three tables

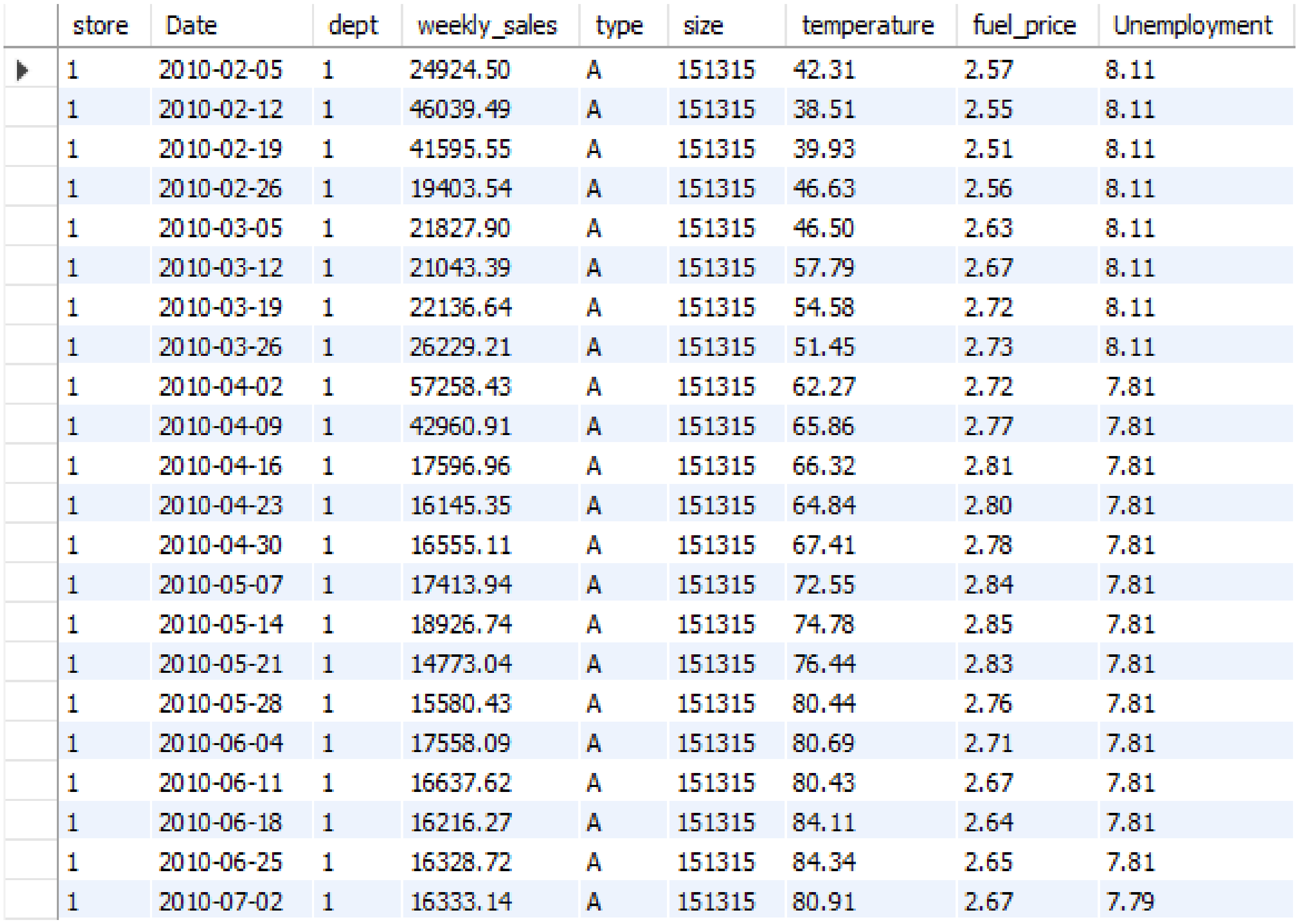
6. SQL Analysis

6.1 Data Verification

- Row count check
- Duplicate check
- Null value check
- Data type validation

6.2 SQL Joins

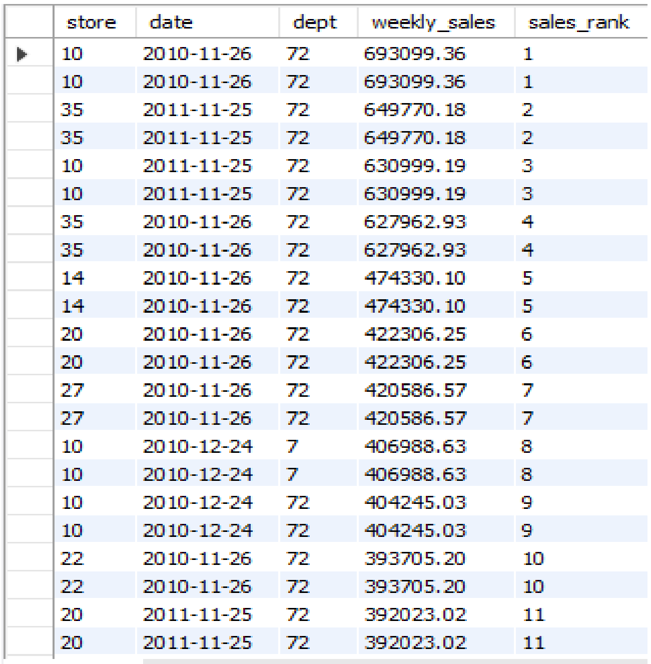
Performed joins between Sales, Features, and Stores to create a combined dataset.



store	Date	dept	weekly_sales	type	size	temperature	fuel_price	Unemployment
1	2010-02-05	1	24924.50	A	151315	42.31	2.57	8.11
1	2010-02-12	1	46039.49	A	151315	38.51	2.55	8.11
1	2010-02-19	1	41595.55	A	151315	39.93	2.51	8.11
1	2010-02-26	1	19403.54	A	151315	46.63	2.56	8.11
1	2010-03-05	1	21827.90	A	151315	46.50	2.63	8.11
1	2010-03-12	1	21043.39	A	151315	57.79	2.67	8.11
1	2010-03-19	1	22136.64	A	151315	54.58	2.72	8.11
1	2010-03-26	1	26229.21	A	151315	51.45	2.73	8.11
1	2010-04-02	1	57258.43	A	151315	62.27	2.72	7.81
1	2010-04-09	1	42960.91	A	151315	65.86	2.77	7.81
1	2010-04-16	1	17596.96	A	151315	66.32	2.81	7.81
1	2010-04-23	1	16145.35	A	151315	64.84	2.80	7.81
1	2010-04-30	1	16555.11	A	151315	67.41	2.78	7.81
1	2010-05-07	1	17413.94	A	151315	72.55	2.84	7.81
1	2010-05-14	1	18926.74	A	151315	74.78	2.85	7.81
1	2010-05-21	1	14773.04	A	151315	76.44	2.83	7.81
1	2010-05-28	1	15580.43	A	151315	80.44	2.76	7.81
1	2010-06-04	1	17558.09	A	151315	80.69	2.71	7.81
1	2010-06-11	1	16637.62	A	151315	80.43	2.67	7.81
1	2010-06-18	1	16216.27	A	151315	84.11	2.64	7.81
1	2010-06-25	1	16328.72	A	151315	84.34	2.65	7.81
1	2010-07-02	1	16333.14	A	151315	80.91	2.67	7.79

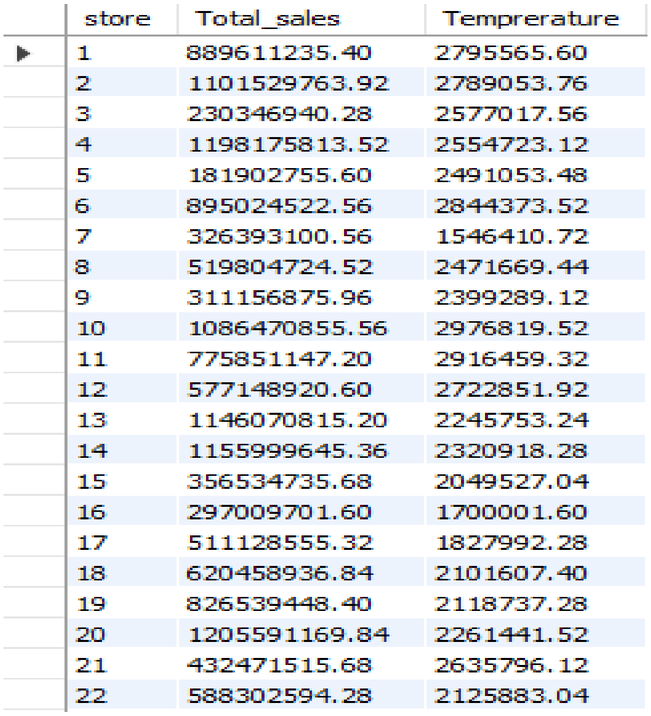
6.3 Important SQL Queries

- Total revenue by store
- Average weekly sales per department
- Holiday vs Non-Holiday sales comparison
- Top 10 stores by revenue
- Impact of CPI and unemployment on weekly sales
- Window functions for ranking store performance



store	date	dept	weekly_sales	sales_rank
10	2010-11-26	72	693099.36	1
10	2010-11-26	72	693099.36	1
35	2011-11-25	72	649770.18	2
35	2011-11-25	72	649770.18	2
10	2011-11-25	72	630999.19	3
10	2011-11-25	72	630999.19	3
35	2010-11-26	72	627962.93	4
35	2010-11-26	72	627962.93	4
14	2010-11-26	72	474330.10	5
14	2010-11-26	72	474330.10	5
20	2010-11-26	72	422306.25	6
20	2010-11-26	72	422306.25	6
27	2010-11-26	72	420586.57	7
27	2010-11-26	72	420586.57	7
10	2010-12-24	7	406988.63	8
10	2010-12-24	7	406988.63	8
10	2010-12-24	72	404245.03	9
10	2010-12-24	72	404245.03	9
22	2010-11-26	72	393705.20	10
22	2010-11-26	72	393705.20	10
20	2011-11-25	72	392023.02	11
20	2011-11-25	72	392023.02	11

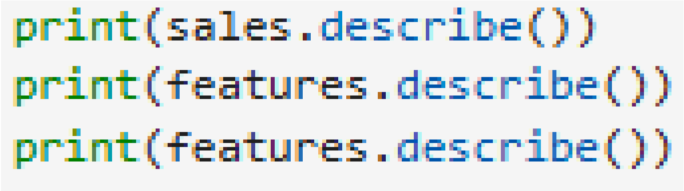
- CTE for impact of temperature on sales



store	Total_sales	Temprerature
1	889611235.40	2795565.60
2	1101529763.92	2789053.76
3	230346940.28	2577017.56
4	1198175813.52	2554723.12
5	181902755.60	2491053.48
6	895024522.56	2844373.52
7	326393100.56	1546410.72
8	519804724.52	2471669.44
9	311156875.96	2399289.12
10	1086470855.56	2976819.52
11	775851147.20	2916459.32
12	577148920.60	2722851.92
13	1146070815.20	2245753.24
14	1155999645.36	2320918.28
15	356534735.68	2049527.04
16	297009701.60	1700001.60
17	511128555.32	1827992.28
18	620458936.84	2101607.40
19	826539448.40	2118737.28
20	1205591169.84	2261441.52
21	432471515.68	2635796.12
22	588302594.28	2125883.04

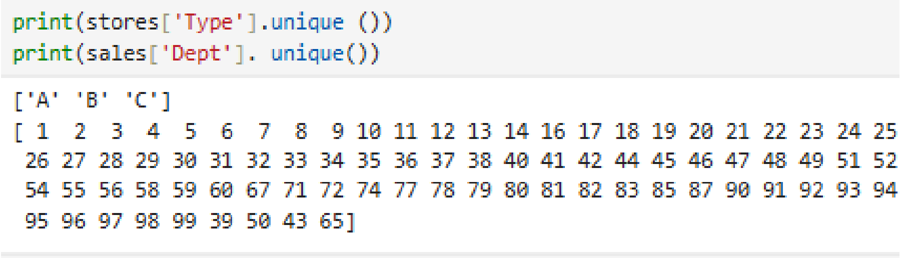
7. Exploratory Data Analysis (EDA)-- Python

- Describe () Overview



```
print(sales.describe())
print(features.describe())
print(features.describe())
```

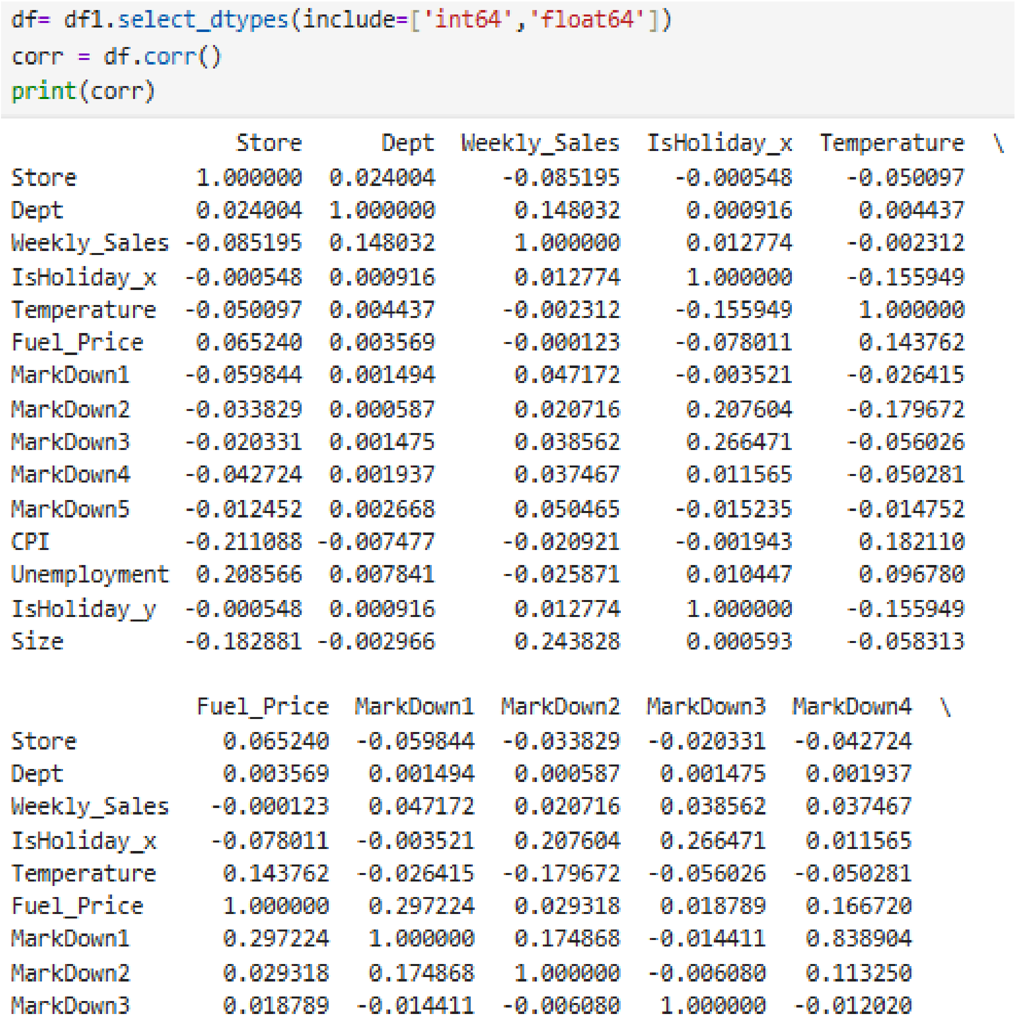
- Unique values



```
print(stores['Type'].unique ())
print(sales['Dept']. unique())
```

```
['A' 'B' 'C']
[ 1  2  3  4  5  6  7  8  9 10 11 12 13 14 16 17 18 19 20 21 22 23 24 25
 26 27 28 29 30 31 32 33 34 35 36 37 38 40 41 42 44 45 46 47 48 49 51 52
 54 55 56 58 59 60 67 71 72 74 77 78 79 80 81 82 83 85 87 90 91 92 93 94
 95 96 97 98 99 39 50 43 65]
```

- Check correlation

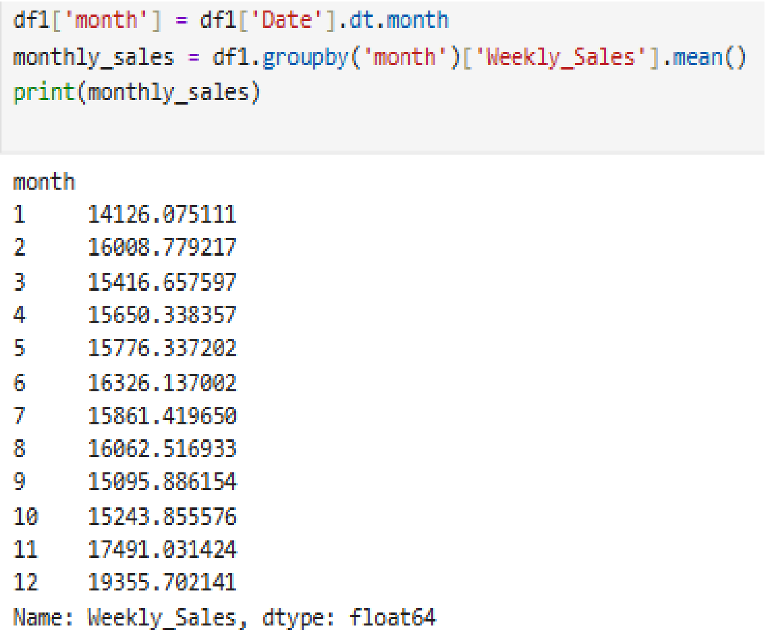


```
df= df1.select_dtypes(include=['int64','float64'])
corr = df.corr()
print(corr)
```

```
                 Store      Dept  Weekly_Sales  IsHoliday_x  Temperature  \
Store         1.000000  0.024004     -0.085195    -0.000548    -0.050097   
Dept          0.024004  1.000000      0.148032     0.000916     0.004437   
Weekly_Sales -0.085195  0.148032      1.000000     0.012774    -0.002312   
IsHoliday_x  -0.000548  0.000916      0.012774     1.000000    -0.155949   
Temperature  -0.050097  0.004437     -0.002312    -0.155949     1.000000   
Fuel_Price    0.065240  0.003569     -0.000123    -0.078011     0.143762   
MarkDown1    -0.059844  0.001494      0.047172    -0.003521    -0.026415   
MarkDown2    -0.033829  0.000587      0.020716     0.207604    -0.179672   
MarkDown3    -0.020331  0.001475      0.038562     0.266471    -0.056026   
MarkDown4    -0.042724  0.001937      0.037467     0.011565    -0.050281   
MarkDown5    -0.012452  0.002668      0.050465    -0.015235    -0.014752   
CPI          -0.211088 -0.007477     -0.020921    -0.001943     0.182110   
Unemployment  0.208566  0.007841     -0.025871     0.010447     0.096780   
IsHoliday_y  -0.000548  0.000916      0.012774     1.000000    -0.155949   
Size         -0.182881 -0.002966      0.243828     0.000593    -0.058313   

              Fuel_Price  MarkDown1  MarkDown2  MarkDown3  MarkDown4  \
Store           0.065240  -0.059844  -0.033829  -0.020331  -0.042724   
Dept            0.003569   0.001494   0.000587   0.001475   0.001937   
Weekly_Sales   -0.000123   0.047172   0.020716   0.038562   0.037467   
IsHoliday_x    -0.078011  -0.003521   0.207604   0.266471   0.011565   
Temperature     0.143762  -0.026415  -0.179672  -0.056026  -0.050281   
Fuel_Price      1.000000   0.297224   0.029318   0.018789   0.166720   
MarkDown1       0.297224   1.000000   0.174868  -0.014411   0.838904   
MarkDown2       0.029318   0.174868   1.000000  -0.006080   0.113250   
MarkDown3       0.018789  -0.014411  -0.006080   1.000000  -0.012020   
```

- Monthly sales trends

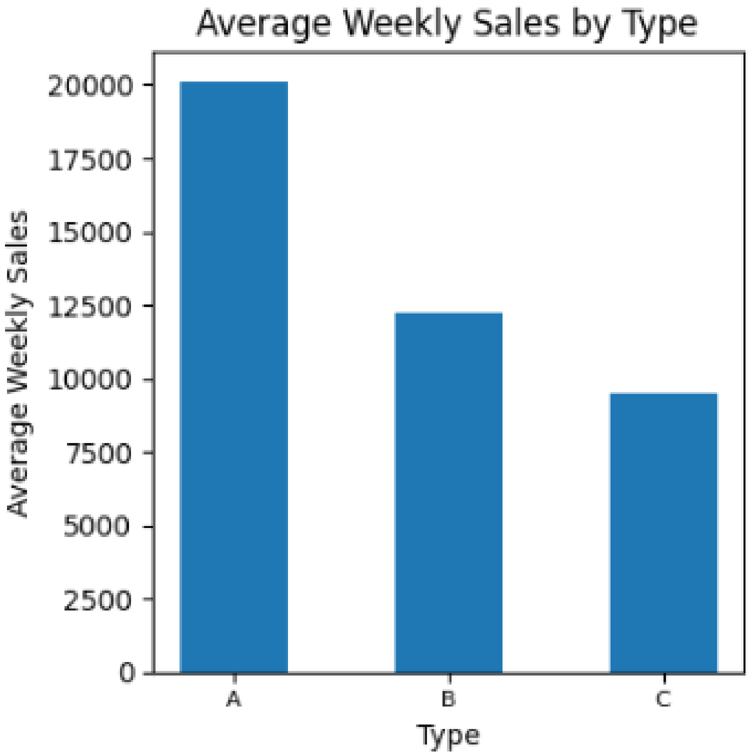


```
df1['month'] = df1['Date'].dt.month
monthly_sales = df1.groupby('month')['Weekly_Sales'].mean()
print(monthly_sales)
```

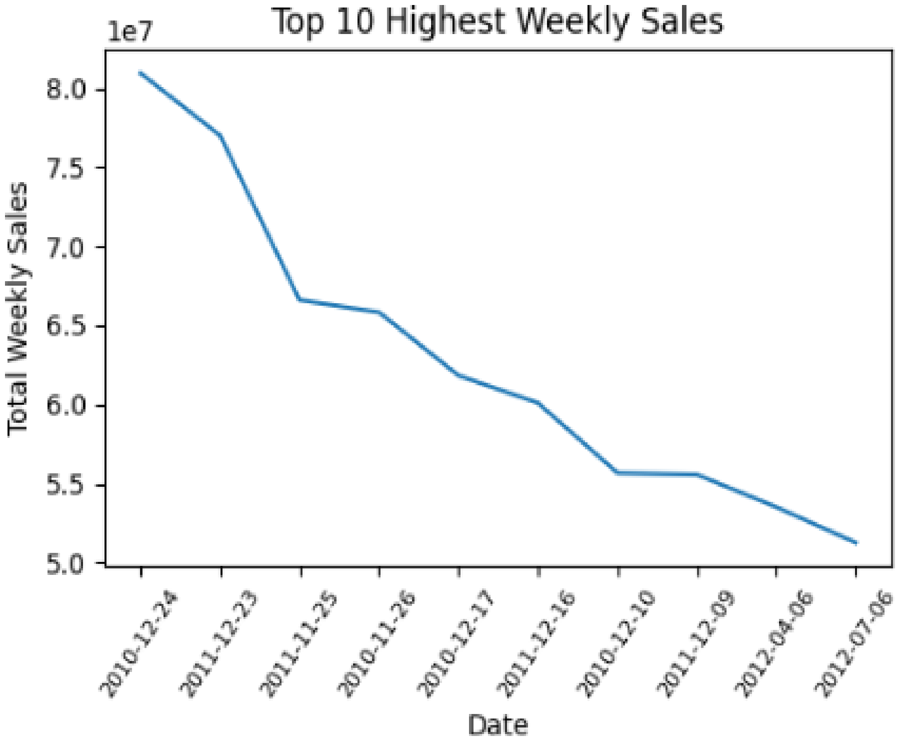
```
month
1     14126.075111
2     16008.779217
3     15416.657597
4     15650.338357
5     15776.337202
6     16326.137002
7     15861.419650
8     16062.516933
9     15095.886154
10    15243.855576
11    17491.031424
12    19355.702141
Name: Weekly_Sales, dtype: float64
```

Python Analysis (Matplotlib)

- **Average weekly sales by Type**



- **Top 10 weekly sales**



8.Data Modeling (Power BI)

Tables Loaded

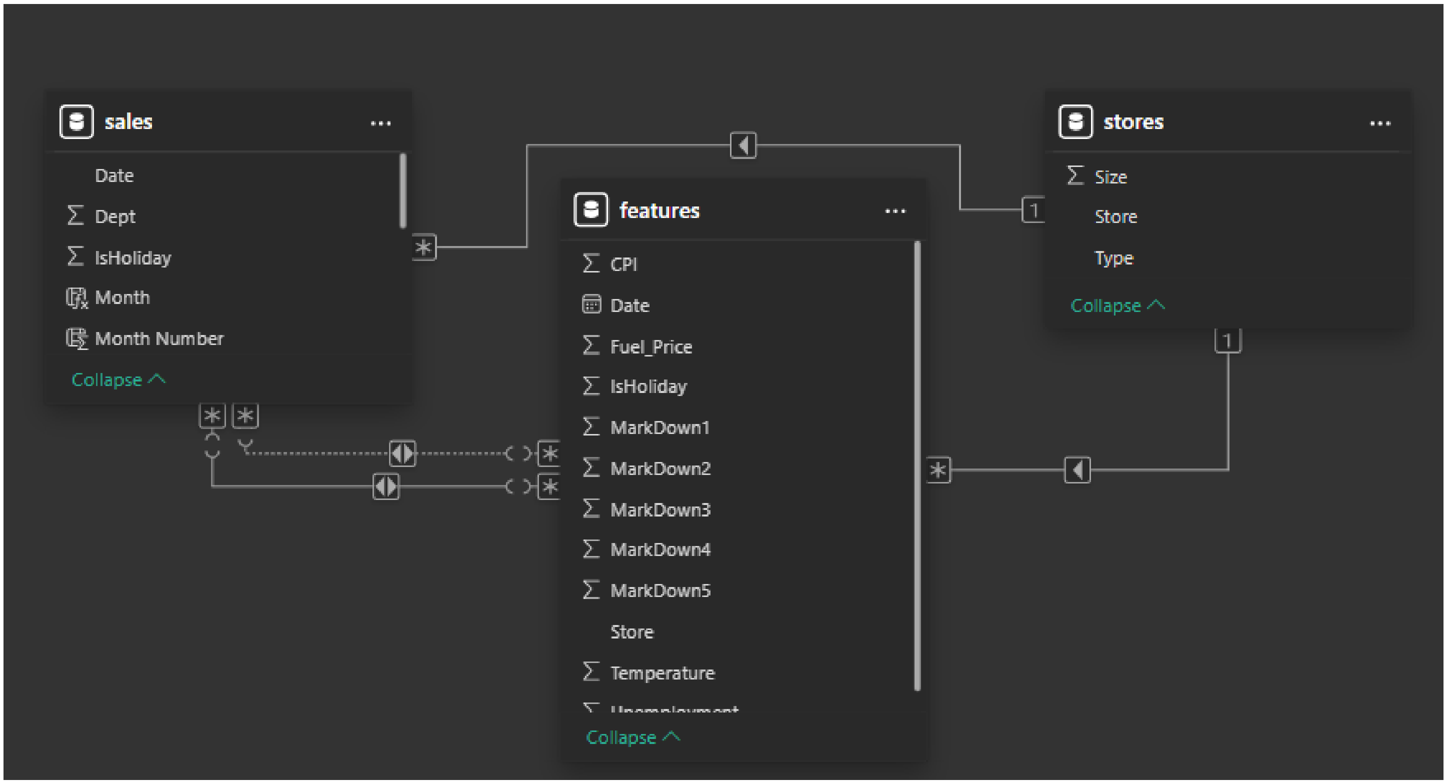
- Sales
- Features
- Stores

Relationships

- Sales [store]→Stores[store]
- Sales [store, date] → Features [store, date]

DAX measures

- Total records
- Total sales
- Average sales
- Holiday sales
- Non-Holiday sales



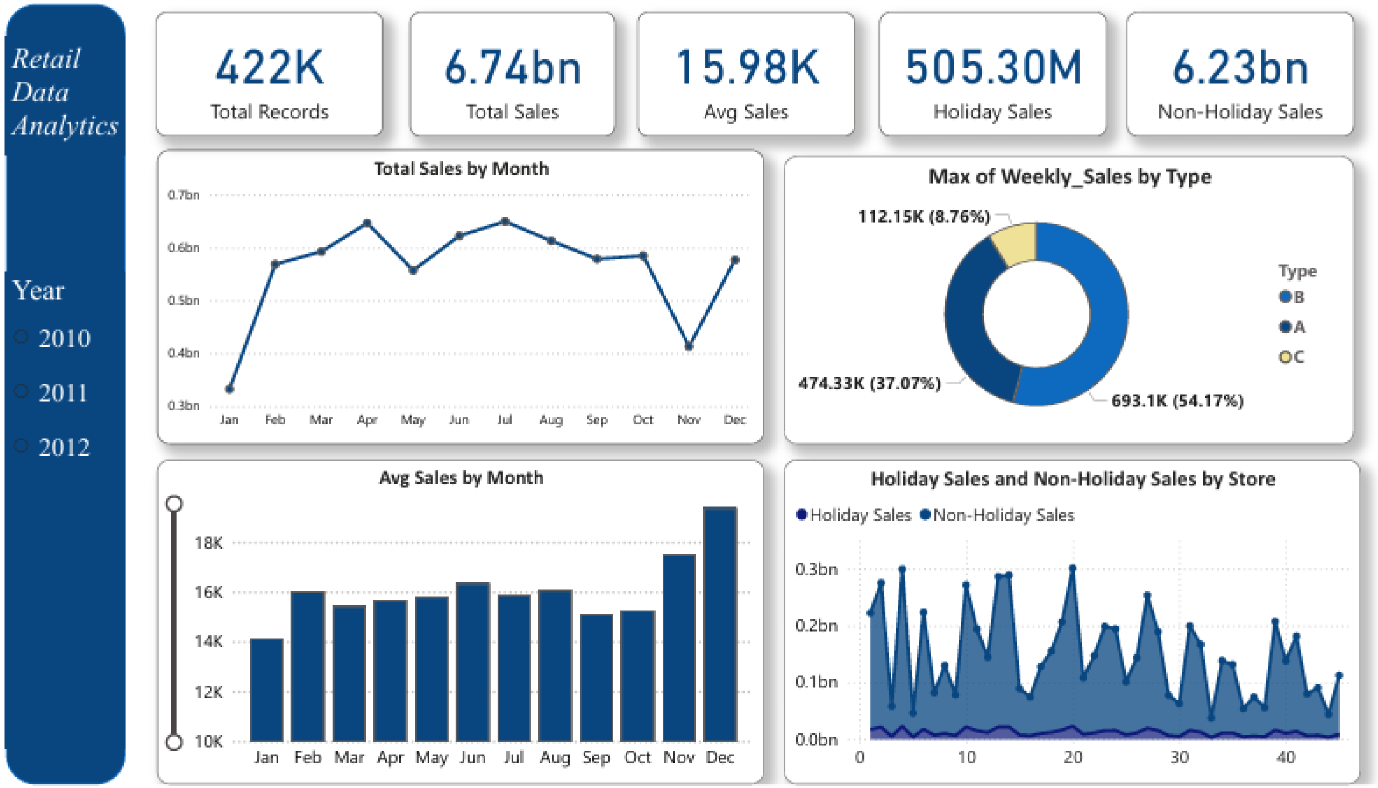
9. Power BI Dashboard

9.1 Visuals

- KPIs (Total records, Total Sales, Average Sales, Holiday sales, Non-Holiday sales)
- Line chart – Monthly sales trend
- Area chart – Holiday sales & Non-sales by store
- Column chart – Average sales by month
- Donut chart – max of weekly sales by type
- Slicers (Year)

9.2 Power BI Insights

- Peak sales observed in November -December
- Strong sales rise during non-holiday periods
- Store Type A generates the maximum sales
- Average sales dip in mid -year peak in November -December



10. Key Insight

10.1 Holiday vs Non-Holiday Sales

- Holiday Sales per week is higher than normal weeks
- Non-Holiday total sales are higher because non-holiday weeks occur more frequently
- Holiday weeks are highly profitable and should be targeted with strong promotions

10.2 Markdown Impact

- Markdown promotions help increase weekly sales, especially during holidays
- Stores with higher markdown usage show better uplift

10.3 Store Performance

- Type A stores (largest stores) generate the highest revenue
- Larger store size correlates with higher sales

10.4 Economic Factor Impact

- High CPI and high fuel price slightly reduce spending
- Temperature has mild influence on weekly sales

10.5 Seasonal Trends

- Strong seasonal pattern with highest sales in November and December
- Sales dip in some months (Feb–Apr), a good opportunity for targeted campaigns

11. Recommendations

11.1 Promotions & Marketing

- Increase markdown promotions during holiday weeks
- Introduce targeted offers during low-season months
- Provide store-level discount strategies for low-performing stores

11.2 Inventory Planning

- Increase inventory in November–December
- Allocate more stock to high-performing stores
- Improve staff planning and operations in low-performing stores

11.3 Store Strategy

- Expand characteristics of successful store type (Type A)

11.4 External Factor Monitoring

- Track fuel price and CPI for demand forecasting
- Adjust pricing strategies when macro-economic conditions worsen

12. Conclusion

This end-to-end Retail Data Analytics project provides a complete analysis of weekly sales, store performance, holiday impact, economic factors, and markdown influence.

Using Excel, SQL, Python, and Power BI, meaningful insights were generated that can support a retail business in decision-making related to planning, promotions, marketing, and store operations.

The project successfully demonstrates strong data analytics skills across data cleaning, EDA, SQL, and dashboarding—making it suitable for portfolio and interview presentations.