

There are 3 projects, 1 of them is a full website & mobile app to manage a clothing store, and can be accessed from [HERE](#) (have to download the PDF)

The other 2 projects can be found on this PDF, and they are made by Power BI.. Happy reading !



Airline Performance Analysis using Power BI

❖ 1. Project Overview

In this project, I developed a comprehensive Power BI dashboard to analyze airline performance using U.S. domestic flight data. The main goal was to uncover insights regarding delays, cancellations, and flight volumes across different airlines, airports, and cities. The dashboard supports both **desktop** and **mobile** views and is divided into two main sheets: **Airlines** and **Cities**.



2. Data Collection and Filtering (SQL)

The dataset included three large CSV files:

- `airline.csv`: airline codes and names
- `airport.csv`: airport metadata (city, state, coordinates, etc.)
- `flights.csv`: detailed flight-level information including delays, distance, and cancellations

To streamline the data and improve performance, I used **SQL** to import only the necessary attributes from the flights dataset:

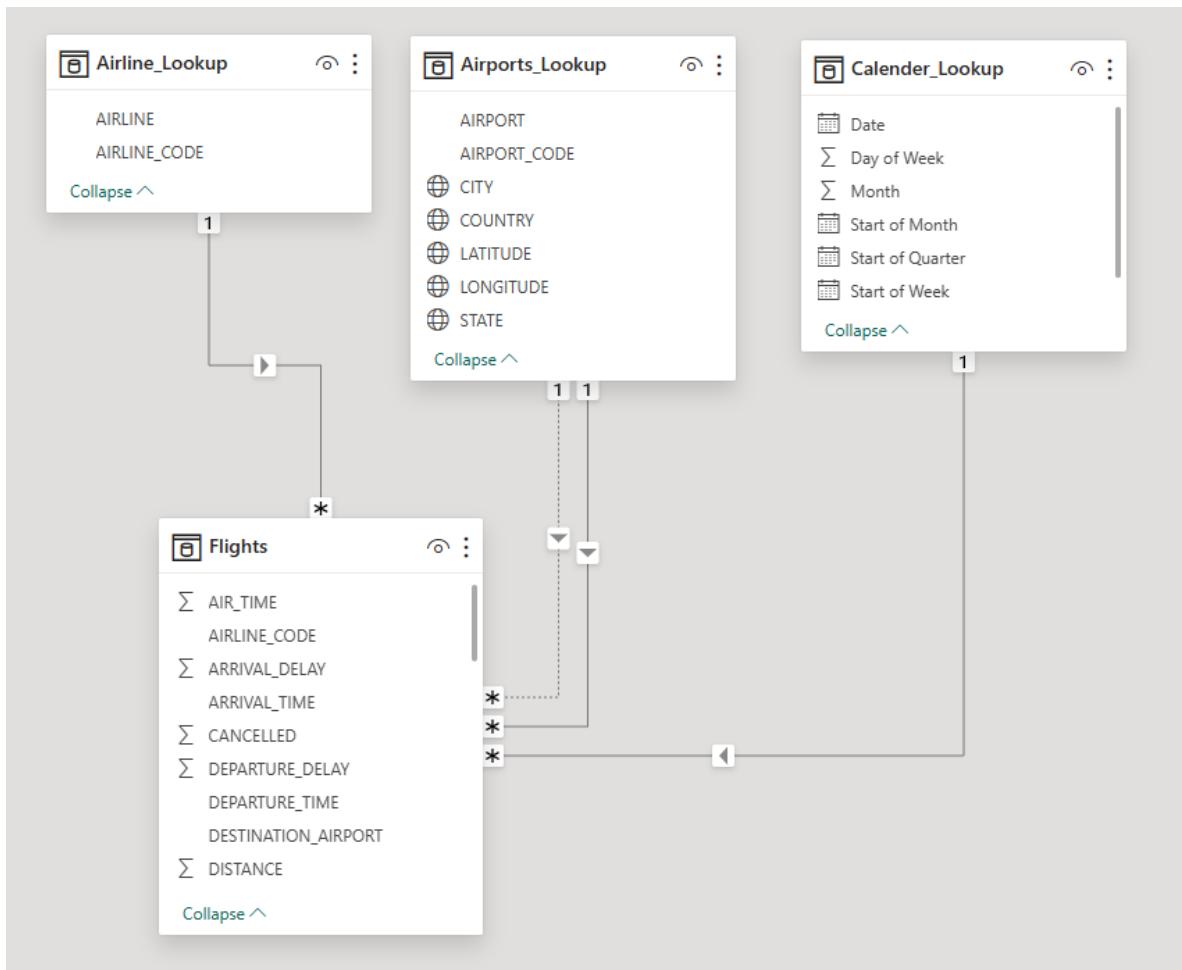


3. Calendar Table and Data Modeling

To enable time-based aggregation and comparisons, I created a **Calendar Lookup Table** that included fields such as Date, Month, Year, and Weekend flag. I used Power BI's DAX to generate this

Date	Start of Week	Start of Quarter	Start of Month	Day of Week	Weekend	Month
jeudi 1 janvier 2015	lundi 29 décembre 2014	jeudi 1 janvier 2015	jeudi 1 janvier 2015	3	Weekday	1
vendredi 2 janvier 2015	lundi 29 décembre 2014	jeudi 1 janvier 2015	jeudi 1 janvier 2015	4	Weekday	1
lundi 5 janvier 2015	lundi 5 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	0	Weekday	1
mardi 6 janvier 2015	lundi 5 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	1	Weekday	1
mercredi 7 janvier 2015	lundi 5 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	2	Weekday	1
jeudi 8 janvier 2015	lundi 5 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	3	Weekday	1
vendredi 9 janvier 2015	lundi 5 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	4	Weekday	1
lundi 12 janvier 2015	lundi 12 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	0	Weekday	1
mardi 13 janvier 2015	lundi 12 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	1	Weekday	1
mercredi 14 janvier 2015	lundi 12 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	2	Weekday	1
jeudi 15 janvier 2015	lundi 12 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	3	Weekday	1
vendredi 16 janvier 2015	lundi 12 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	4	Weekday	1
lundi 19 janvier 2015	lundi 19 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	0	Weekday	1
mardi 20 janvier 2015	lundi 19 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	1	Weekday	1
mercredi 21 janvier 2015	lundi 19 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	2	Weekday	1
jeudi 22 janvier 2015	lundi 19 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	3	Weekday	1
vendredi 23 janvier 2015	lundi 19 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	4	Weekday	1
lundi 26 janvier 2015	lundi 26 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	0	Weekday	1
mardi 27 janvier 2015	lundi 26 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	1	Weekday	1
mercredi 28 janvier 2015	lundi 26 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	2	Weekday	1

I then built data model relationships



✍ 4. Data Cleaning and Feature Engineering

I cleaned the data by:

- Handling null values in delay fields
- Filtering out invalid or empty airport codes
- Removing cancelled and diverted flights when needed for KPIs



5. DAX Measures and KPIs

I created multiple **custom DAX measures** to support advanced analytics. Here are some of the most important ones:

Flight-Level Metrics

```
Flights Count = COUNTROWS(Flights)

NonCancelled Flights =
CALCULATE([Flights Count],
    Flights[CANCELLED] = 0 && Flights[DIVERTED] = 0
)

Diverted Flights =
CALCULATE([Flights Count],
    Flights[DIVERTED] = 1
)

Weekend Flights =
CALCULATE([Flights Count],
    Calender_Lookup[Weekend] = "Weekend"
)
```

⌚ Time and Delay Metrics

```
Time in Air = SUM(Flights[AIR_TIME])

Time in Air per Flight = [Time in Air] / [NonCancelled Flights]

Total_Delays =
SUM(Flights[ARRIVAL_DELAY]) + SUM(Flights[DEPARTURE_DELAY])

Delays = SUM(Flights[Total_Delays])

Prev Month Delays =
CALCULATE([Delays],
    DATEADD(Calender_Lookup[Date], -1, MONTH)
)

Delays Target = [Prev Month Delays] * 1.2

MTD Delays =
CALCULATE([Delays], DATESMTD(Calender_Lookup[Date]))

10-day Rolling Delays =
CALCULATE([Delays],
    DATESINPERIOD(Calender_Lookup[Date], MAX(Calender_Lookup[Date]), -10, DAY)
)
```

📊 Percentage and Distance Metrics

```
% of All Flights =
DIVIDE([Flights Count], [ALL Flights])

% of Cancelled Flights =
DIVIDE([Cancelled Flights], [Flights Count])

Distance Covered = SUM(Flights[DISTANCE])

Distance per Flight =
DIVIDE([Distance Covered], [NonCancelled Flights])

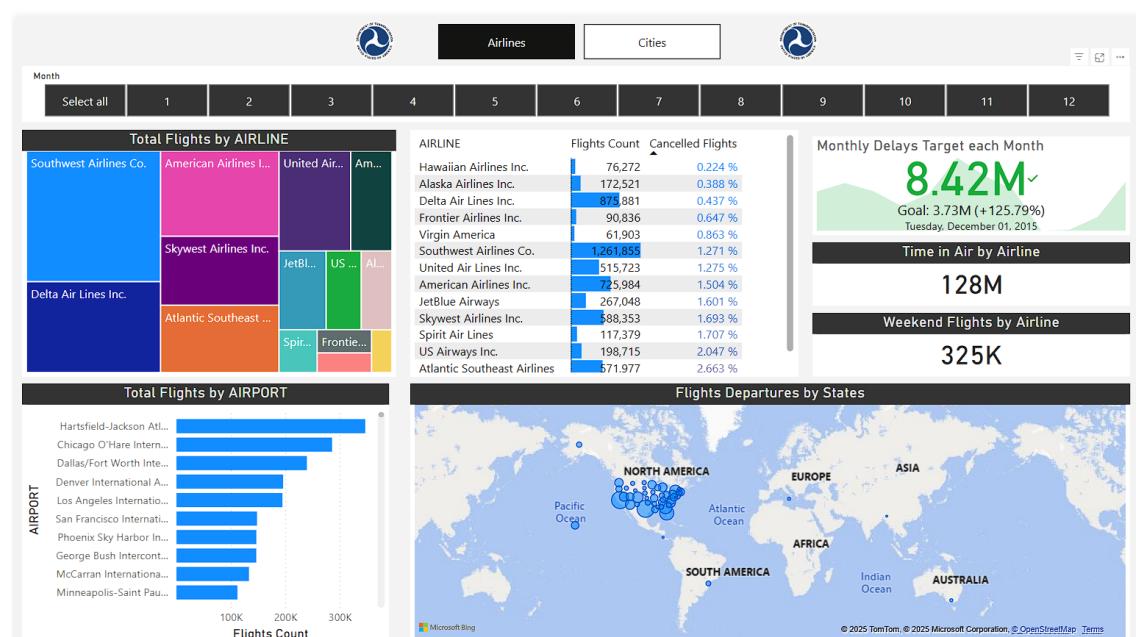
ALL Flights =
CALCULATE([Flights Count],
    ALL(Airports_Lookup), ALL(Airline_Lookup), ALL(Flights))

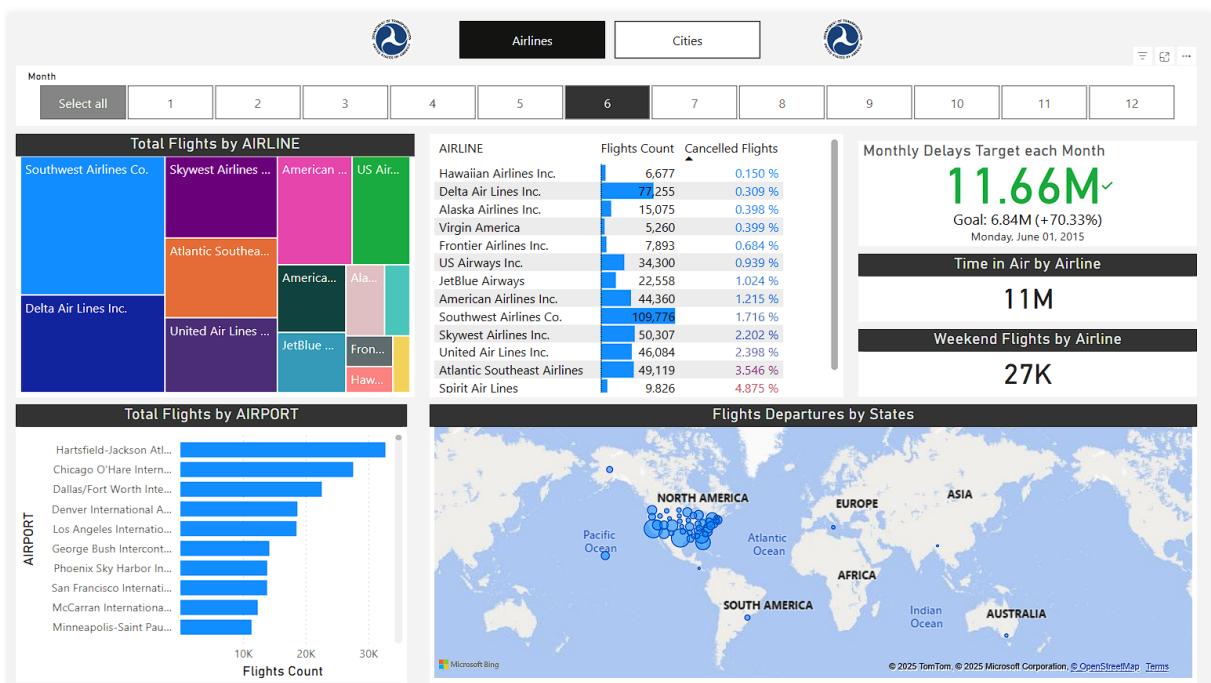
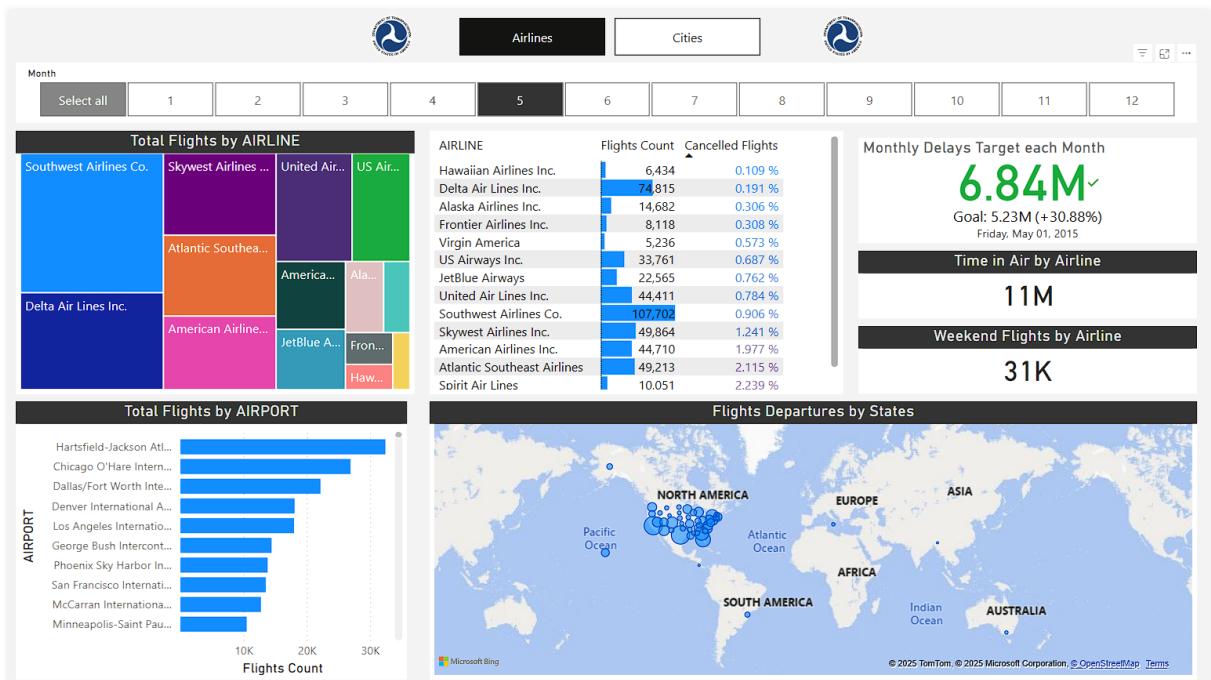
ALL NonCancelled Flights =
CALCULATE([NonCancelled Flights],
    ALL(Airports_Lookup), ALL(Airline_Lookup), ALL(Flights))
```

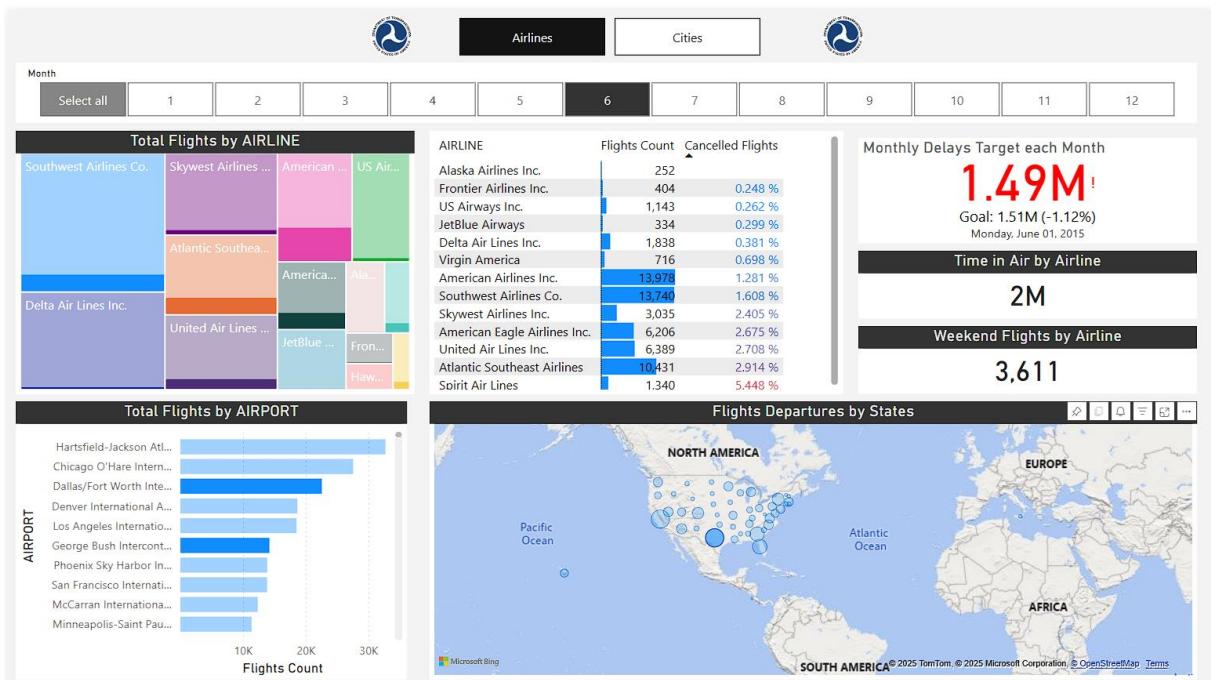
📊 6. Dashboard Design

The dashboard consists of two separate sheets — each targeting a specific analytical perspective:

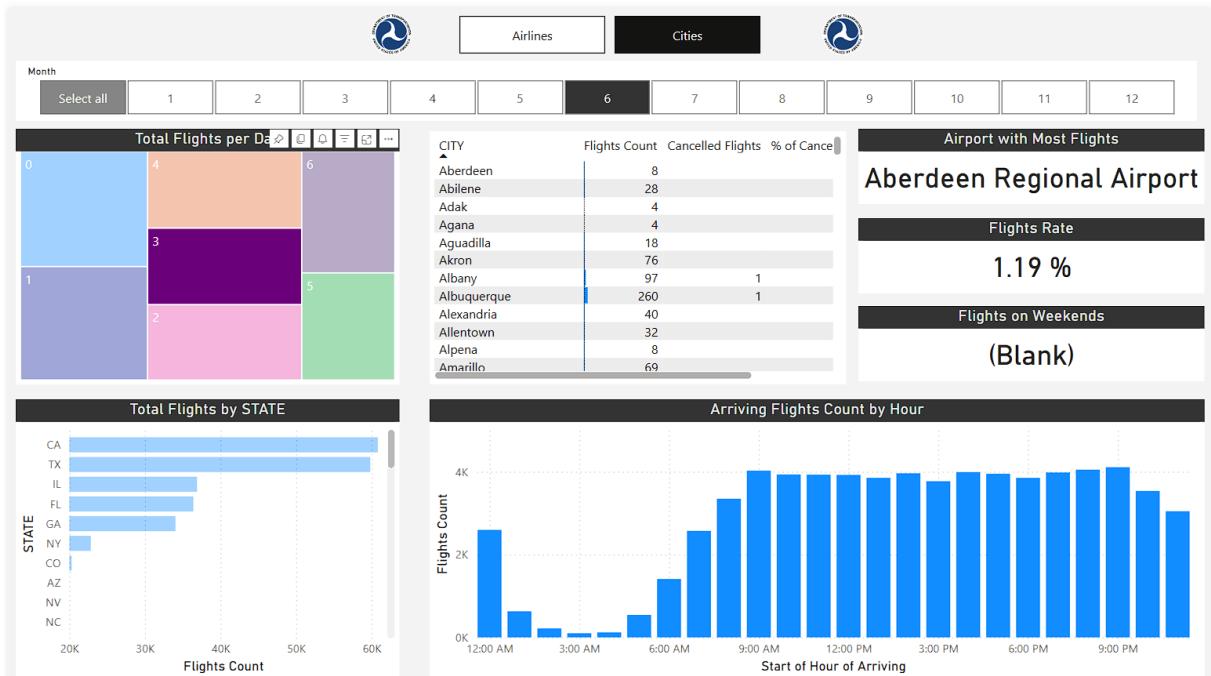
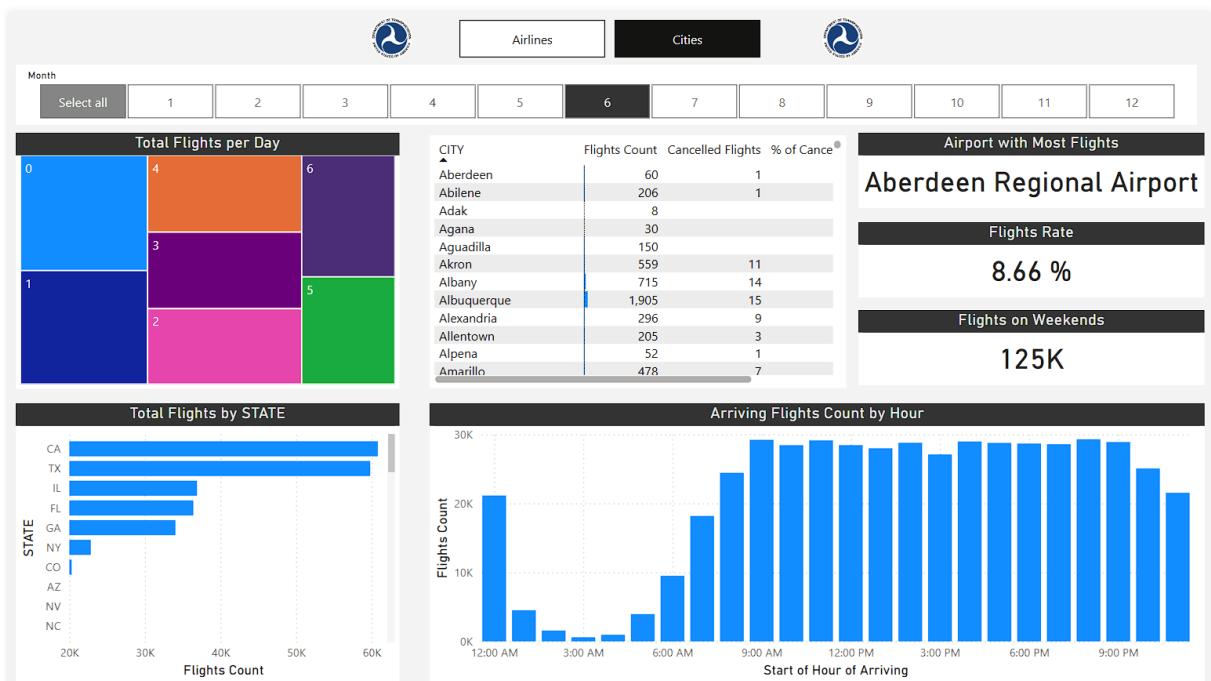
Sheet 1: Airlines Overview

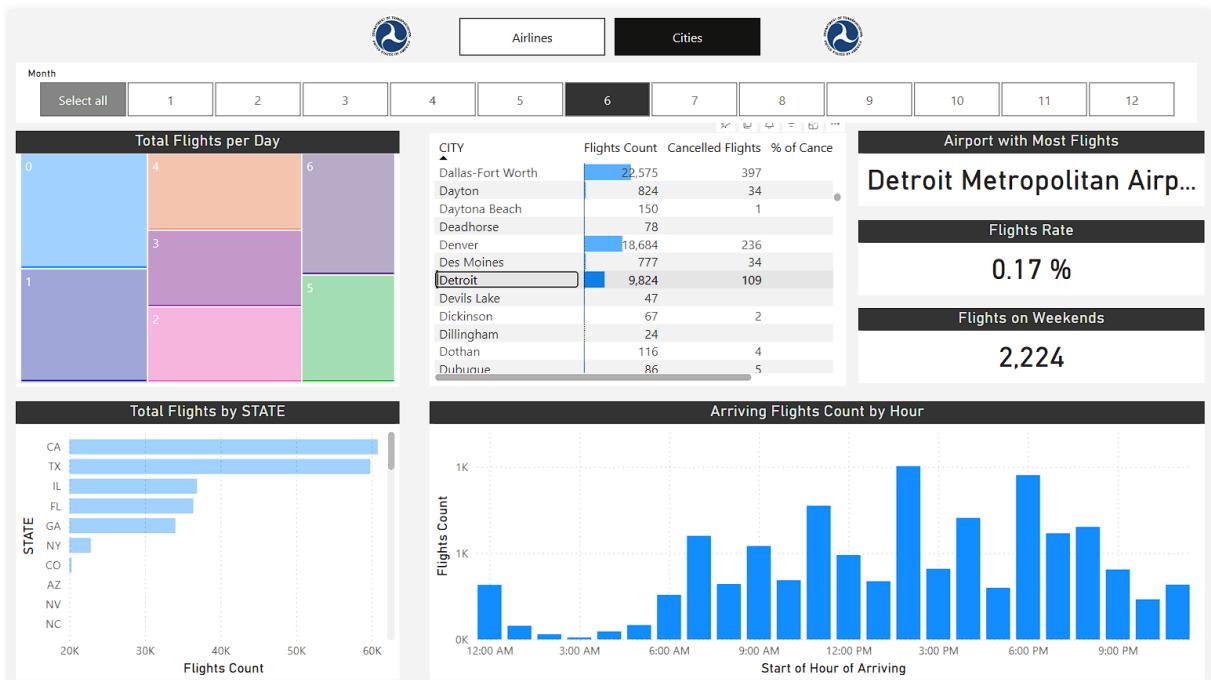




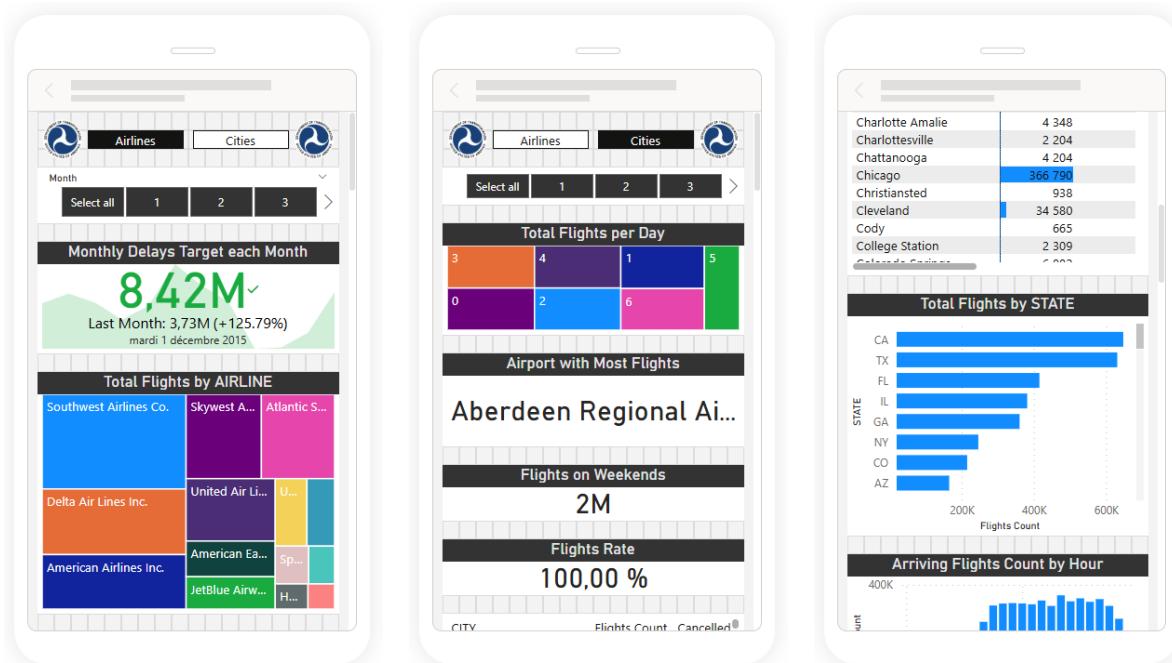


Sheet 2: Cities Overview





Mobile Version



7. Insights & Takeaways

From the dashboard, I was able to derive several important insights:

- Certain airlines consistently had higher cancellation rates

- Delays spiked during specific months, especially around holidays
- Flights during weekends showed different cancellation patterns
- The distance per flight and time in air helped analyze airline efficiency
- Most flights departed from major coastal states, especially California and New York



8. Future Improvements

To further enhance the dashboard:

- Integrate **real-time weather data** to correlate with delays
- Add **predictive analytics** for cancellation probability
- Automate data ingestion using **Power BI Dataflows** or **SQL Server Agent**



Walmart Store Sales Forecasting – Power BI Project



1. Project Overview

In this project, I developed a Power BI dashboard to visualize and forecast sales across Walmart stores. Using time series and categorical data, the aim was to understand sales trends, uncover seasonal patterns, and analyze external factors like fuel price, temperature, and holidays that influence performance. I focused on both **store-type performance** and **department-level trends**, building a dashboard with rich interactivity and insightful visualizations.



2. Data Sources and Preprocessing

The dataset consisted of three main files:

- features.csv: economic and marketing data including Fuel Price, CPI, Temperature, and five markdown campaigns
- stores.csv: store metadata including Type and Size
- sales.csv: weekly sales data by Store and Department

To prepare the data, I imported all three files into Power BI and performed several key preprocessing steps:



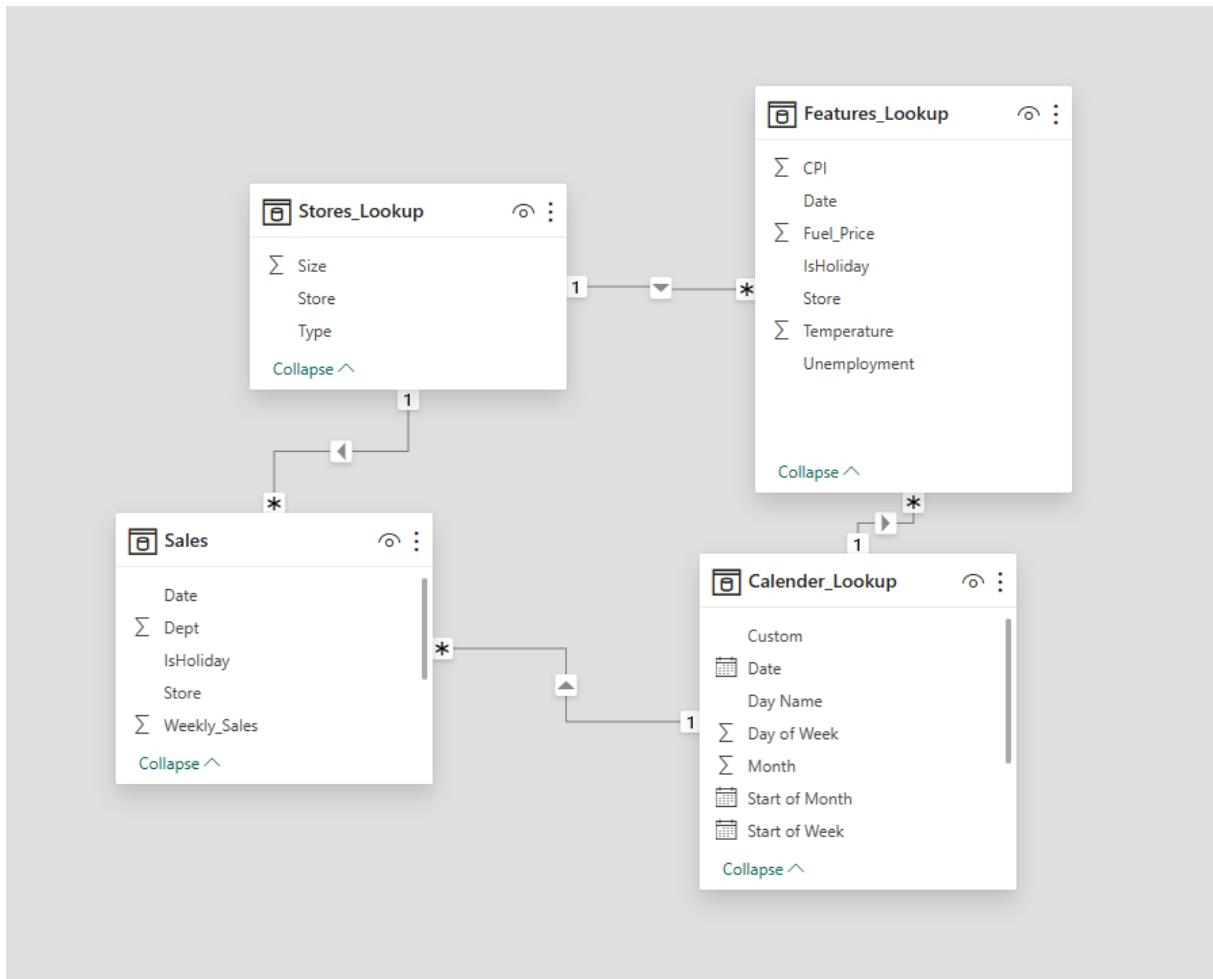
3. Calendar Table and Data Modeling

To support time-based analysis, I created a **calendar lookup table** using DAX:

Date	Start of Week	Start of Quarter	Start of Month	Day of Week	Weekend	Month
jeudi 29 janvier 2015	lundi 29 décembre 2014	jeudi 1 janvier 2015	jeudi 1 janvier 2015	3 Weekday	1	
vendredi 2 janvier 2015	lundi 29 décembre 2014	jeudi 1 janvier 2015	jeudi 1 janvier 2015	4 Weekday	1	
lundi 5 janvier 2015	lundi 5 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	0 Weekday	1	
mardi 6 janvier 2015	lundi 5 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	1 Weekday	1	
mercredi 7 janvier 2015	lundi 5 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	2 Weekday	1	
jeudi 8 janvier 2015	lundi 5 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	3 Weekday	1	
vendredi 9 janvier 2015	lundi 5 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	4 Weekday	1	
lundi 12 janvier 2015	lundi 12 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	0 Weekday	1	
mardi 13 janvier 2015	lundi 12 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	1 Weekday	1	
mercredi 14 janvier 2015	lundi 12 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	2 Weekday	1	
jeudi 15 janvier 2015	lundi 12 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	3 Weekday	1	
vendredi 16 janvier 2015	lundi 12 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	4 Weekday	1	
lundi 19 janvier 2015	lundi 19 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	0 Weekday	1	
mardi 20 janvier 2015	lundi 19 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	1 Weekday	1	
mercredi 21 janvier 2015	lundi 19 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	2 Weekday	1	
jeudi 22 janvier 2015	lundi 19 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	3 Weekday	1	
vendredi 23 janvier 2015	lundi 19 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	4 Weekday	1	
lundi 26 janvier 2015	lundi 26 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	0 Weekday	1	
mardi 27 janvier 2015	lundi 26 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	1 Weekday	1	
mercredi 28 janvier 2015	lundi 26 janvier 2015	jeudi 1 janvier 2015	jeudi 1 janvier 2015	2 Weekday	1	

🔗 4. Relationships and Data Model

I established the following relationships in the data model



✍ 5. Data Cleaning and Calculated Columns

I cleaned the data by:

- Removing records with missing or null Weekly_Sales
- Standardizing date formats across all tables
- Filling null values in Markdown columns with zeros where appropriate

I also added calculated columns..

🧠 6. DAX Measures

I created several DAX measures to enable advanced KPIs and time-based comparisons



7. Dashboard Design

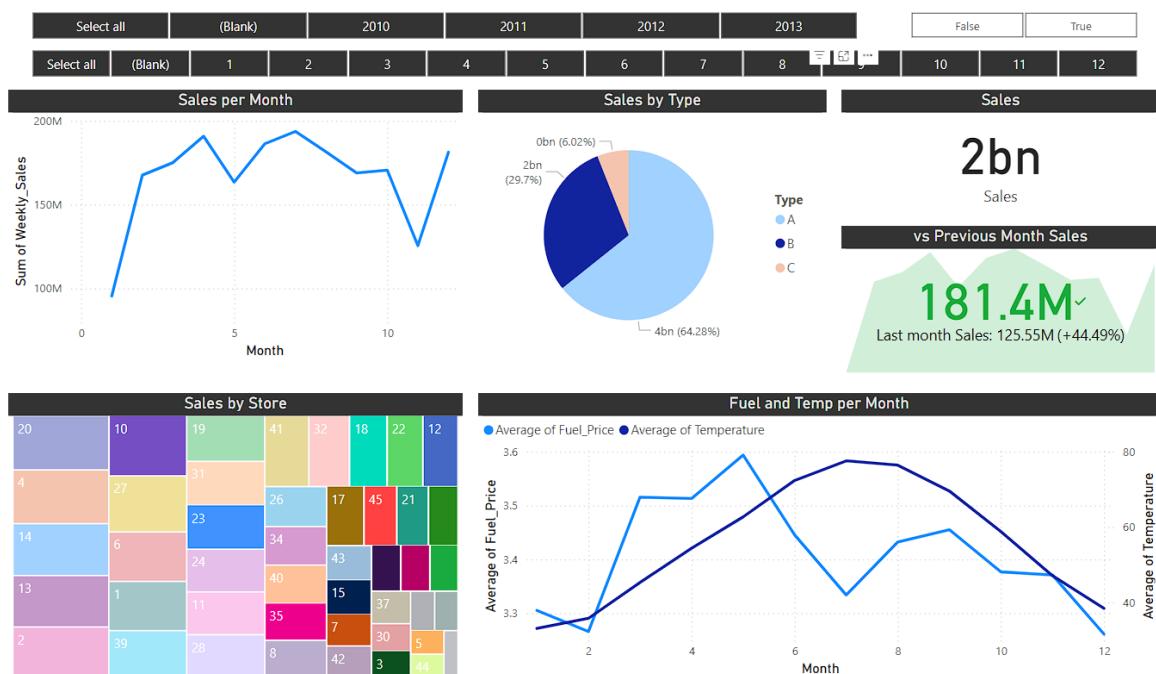
The dashboard includes several filters and visual elements to enhance exploration:

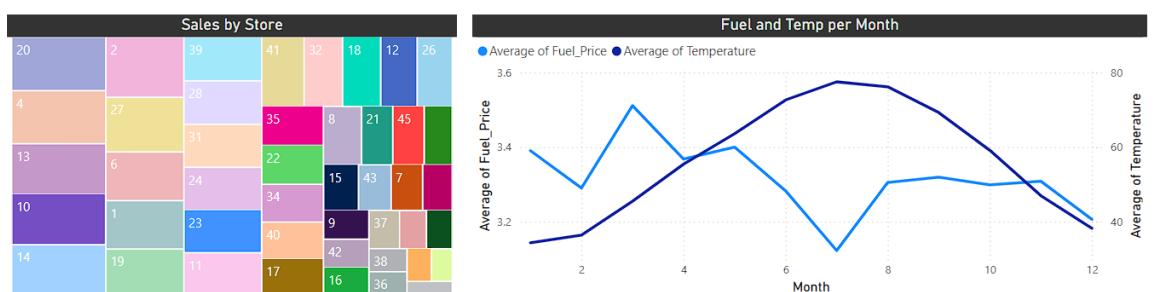
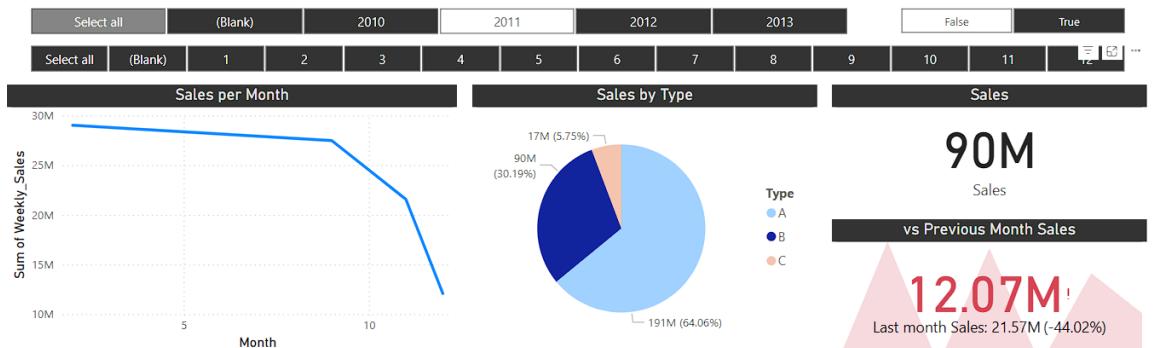
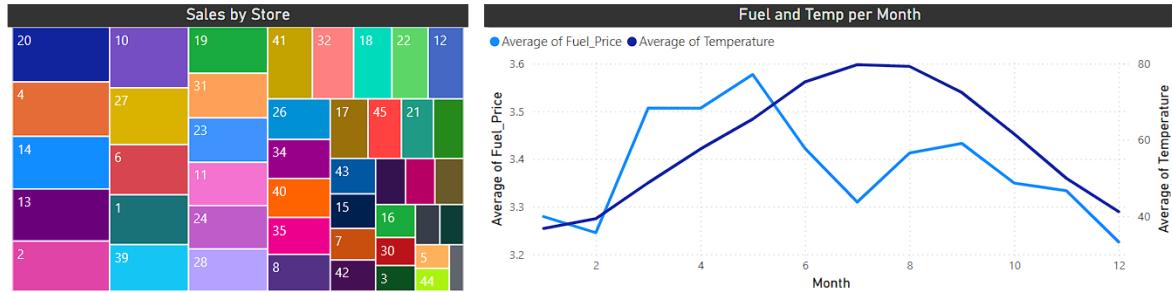
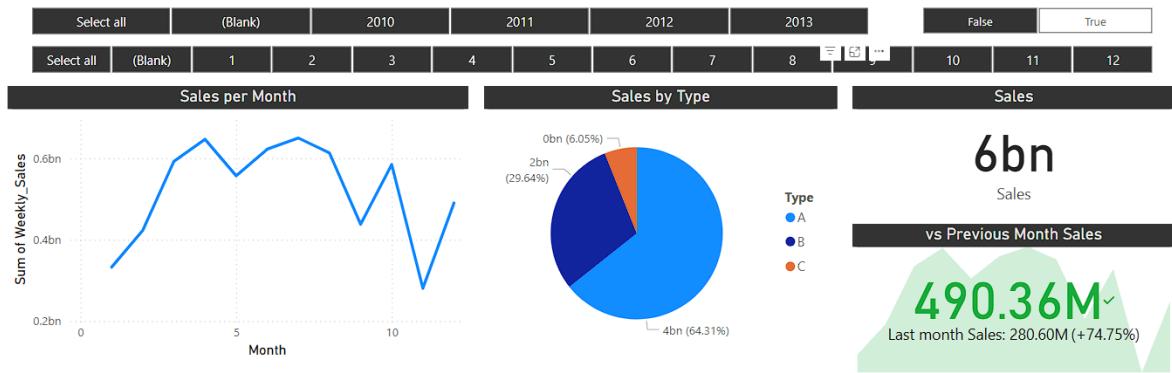
Top Filters

- Year and Month slicers (from the Calendar table)
- IsHoliday filter (to isolate holiday-related data)

Visuals

1. **Sales per Month**
→ Line chart showing total monthly sales trends over time
2. **Sales per Store Type**
→ Pie chart showing how much each store type contributes to total sales
3. **Sales per Store – Tree Map**
→ Tree map comparing individual stores by total sales
4. **Fuel Price and Temperature Trends**
→ Dual line chart to track monthly changes in fuel and temperature, enabling seasonal correlation with sales
5. **KPI: Sales vs Previous Month**
→ KPI card displaying this month's sales and a comparison to the previous month







8. Key Insights

From my dashboard and DAX analysis, I extracted the following insights:

- Holidays had a clear impact on sales, especially around November.
- Store Type A consistently outperformed other types in terms of revenue.
- There is a mild correlation between fuel price and sales dips, indicating possible consumer behavior shifts.
- Sales typically decrease in cold months, thus, they run markdown campaigns and holiday promotions to sell more. For example, January is the lowest selling month when it is cold and there is no holiday.



9. Next Steps and Enhancements

To further enhance the analysis:

- Add forecasting capabilities using Power BI's built-in analytics
- Use Python or R visuals for seasonal decomposition
- Build a predictive model using machine learning (e.g., regression or LSTM for time-series)