Airline Data Management and Analysis Using Power BI

BY- BULBUL SINGH

Table of Contents

Video explanation link- https://drive.google.com/file/d/1OljztVJh01-NEqXzcJ4_-Y74DYhyZk4G/view?usp=sharing

Introduction

This project focuses on analyzing airline operational data using Power BI. The goal is to transform raw data from three different sources — Flight_Information, Ticket_Information, and Passenger_Information — into meaningful insights through data cleaning, modeling, DAX calculations, and interactive dashboards.

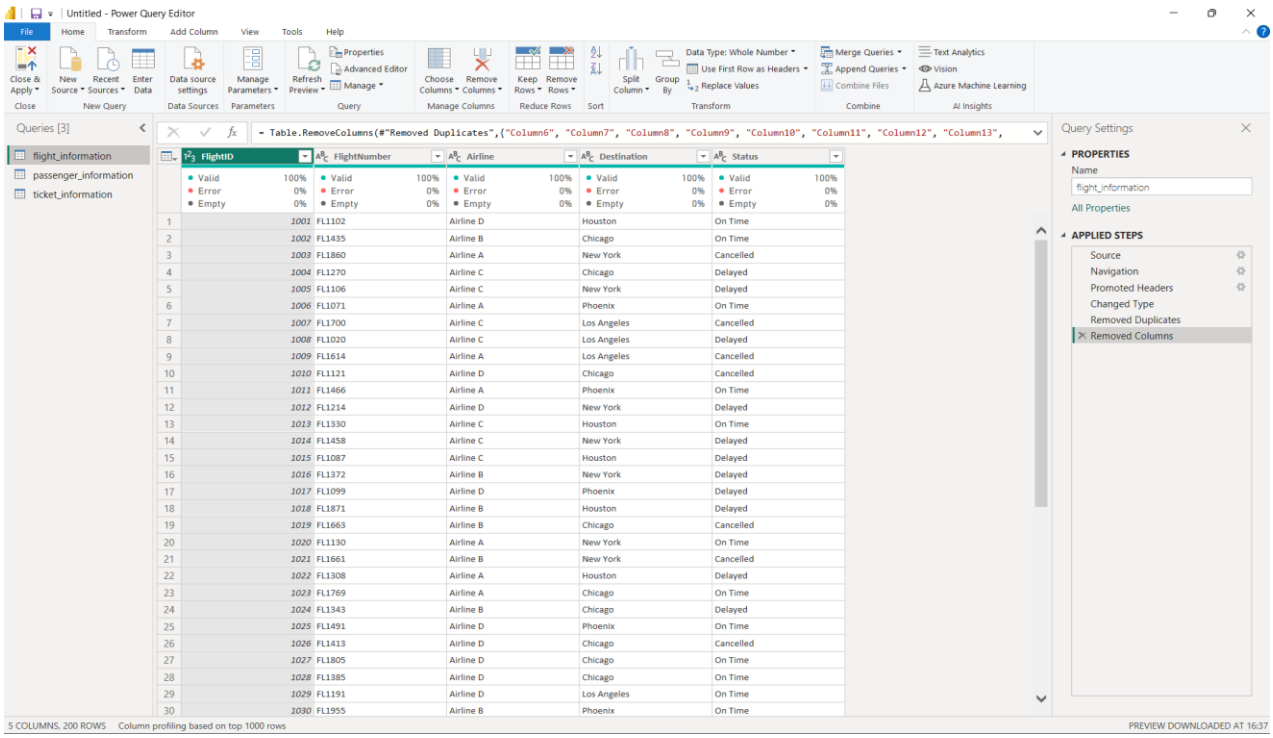
By applying various Power BI features such as relationships, conditional columns, measures, slicers, and drillthrough pages, the report delivers a complete view of flight performance, passenger trends, and ticket booking status. The final dashboard provides a powerful tool for decision-making, with enhanced visuals and automated data refresh for real-time insights.

Problem Statement: The airline industry operates with numerous complexities, requiring effective data management and insights into flight schedules, passenger details, and ticketing systems. This project aims to analyze airline operations for improving efficiency and customer satisfaction.

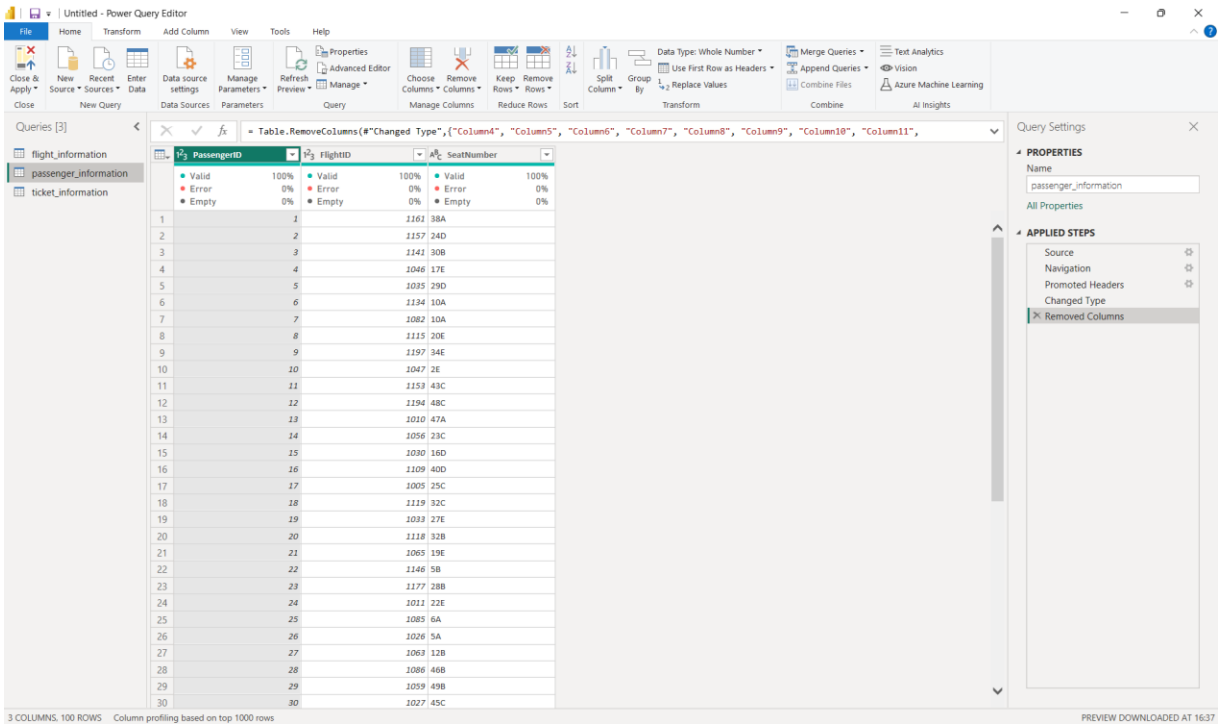
Task 1: Data Cleaning and Preparation

In this task, I cleaned and prepared all three datasets — Flight_Information, Ticket_Information, and Passenger_Information — using **Power BI's Power Query Editor**.

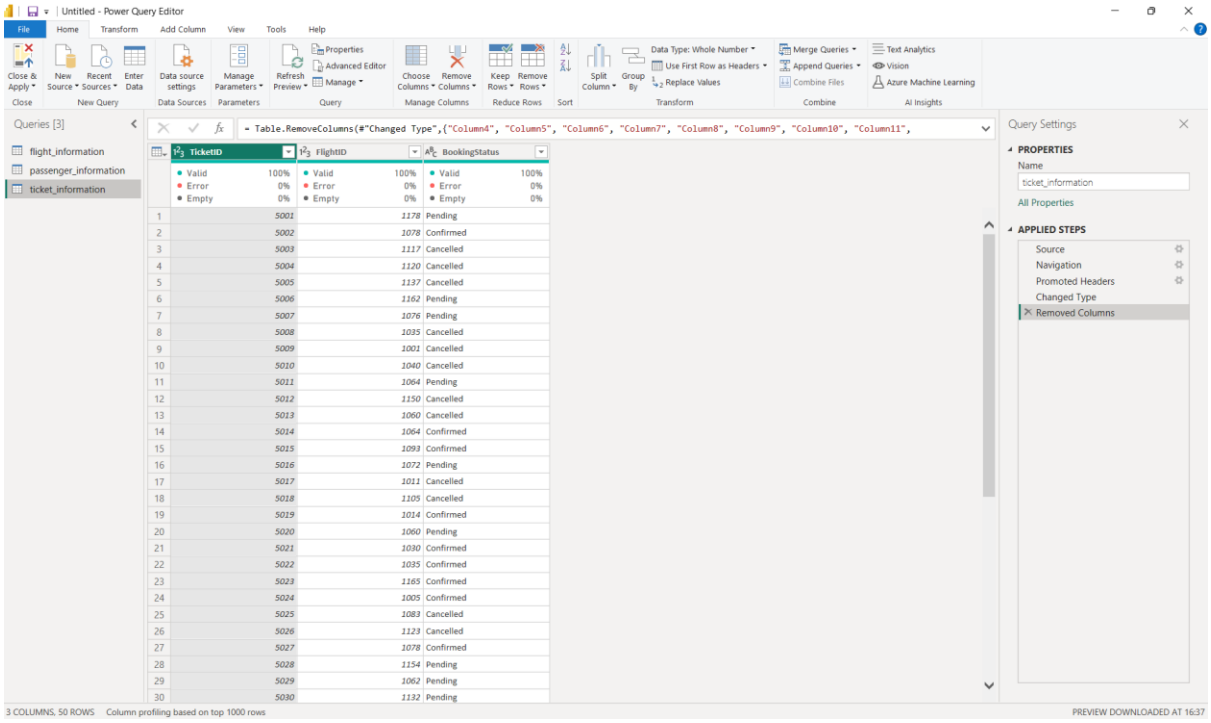
I checked each sheet individually for appropriate data types (text, integers, and dates) and looked for any irregular formatting, such as leading or trailing spaces. This stage aided in getting the dataset ready for precise modelling and computations in the future.



In the Flight Information sheet, Many column contained only null values. Since it had no usable data, I deleted the entire column to maintain a clean and relevant dataset. I also ensured that the columns such as FlightID, Airline, and Status were in the correct data types.



The Passenger_Information sheet also had a column filled with nulls, which I removed to streamline the dataset. I also confirmed that each column had the correct data type (e.g., PassengerID as whole number and Name as text) and no extra whitespace or format issues.



This screenshot shows the Ticket_Information table where I identified and deleted a column containing only null values. Additionally, I ensured proper formatting of columns like TicketID, FlightID, and Status, and replaced nulls in numeric fields with suitable placeholders where needed.

Task 2: Data Modeling

For the data model, I created relationships between the three main tables: Flight_Information, Passenger_Information, and Ticket_Information. The common key used was FlightID, which allowed me to establish proper one-to-many relationships.

In Power BI, **cardinality** refers to the relationship between tables based on how many matching rows exist in each table. There are three main types of cardinality:

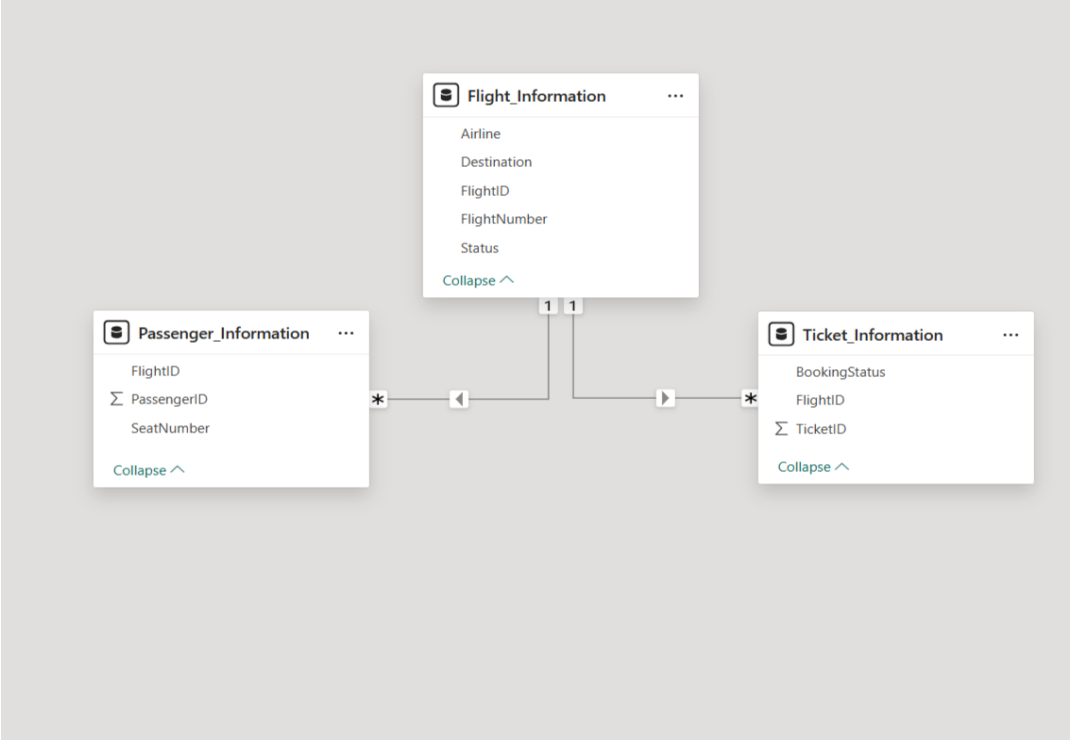
- **One-to-many (1:*)**
- **Many-to-one (*:1)**
- **Many-to-many (:)**

In my project, I understood how the datasets relate to one another and used **One-to-Many (1:*) relationships** for proper data modeling. Here's how I applied cardinality:

- One flight in the Flight_Information table can have **many passengers** (in Passenger_Information) and **many tickets** (in Ticket_Information).
- Therefore, I made Flight_Information[FlightID] the **"One"** side, and both Ticket_Information[FlightID] and Passenger_Information[FlightID] the **"Many"** side.

I ensured that the **relationship direction** was configured properly (usually single-direction filtering from the "One" to "Many" table), which allowed the filters to work correctly in visuals and calculations.

This step was crucial for creating accurate insights in my dashboard, such as calculating passenger counts by airline or filtering ticket data based on flight status.

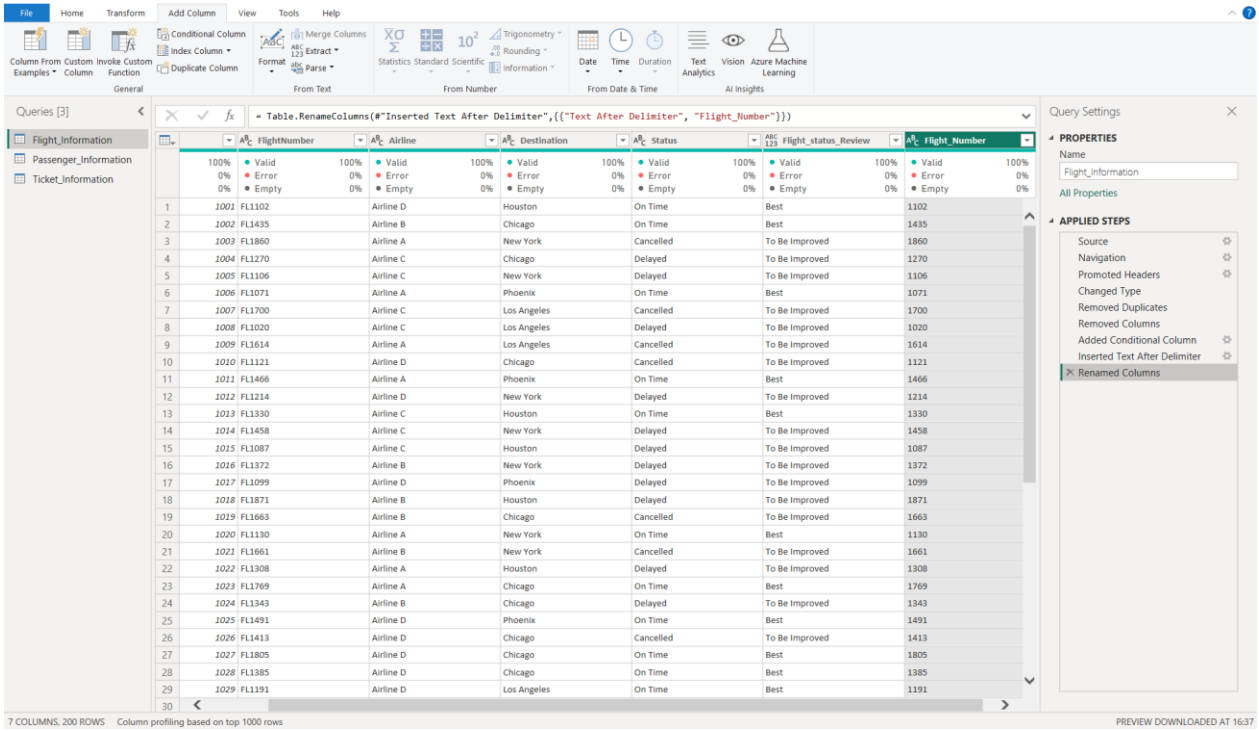


This diagram shows the relationships created between the tables using the FlightID key. It also displays the cardinality (one-to-many) and direction of relationships.

Task 3: Enhanced Data Insights

I created a conditional column that classified flights as either “Best” or “To Be Improved”, based on the Status field in the Flight_Information table. If the flight status was “On Time”, it was marked as "Best", otherwise "To Be Improved".

Additionally, I used the Column From Example feature in Power Query to extract the flight number from a compound FlightNumber field. Instead of splitting by delimiter, I typed examples like "A123" and Power BI automatically detected the pattern, helping extract the flight number part efficiently.



Here, I created a conditional column that classifies each flight as either ‘Best’ or ‘To Be Improved’ based on the flight’s status. This helps quickly filter high-performing flights.

This screenshot also shows the Column From Example feature in Power Query used to extract just the flight number portion from the full FlightNumber string.

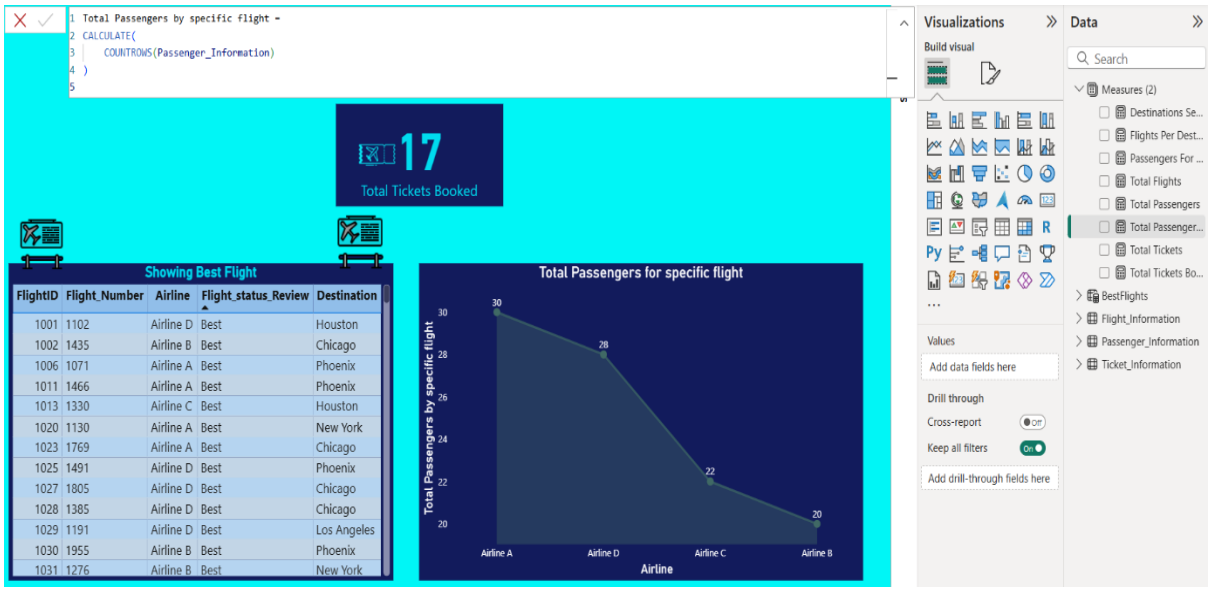
Task 4: Calculations Using DAX

In this task, I used DAX to create calculated measures:

- Total passengers for a specific flight: Counted the number of unique passengers per flight
- Total tickets booked: Counted ticket records from the Ticket_Information table
- Filtered table: Created a visual/table showing only the "Best" flights based on the conditional column

These measures helped in generating KPIs and analytical visuals to understand flight and passenger data better.

- **Total passengers for a specific flight Measure**



The DAX formula shown here calculates total passengers per flight using the COUNT or COUNTROWS function, enabling airline-specific passenger analysis.

- **Total tickets booked Measure**

This screenshot displays a DAX measure counting the number of tickets booked, derived from the Ticket_Information table.

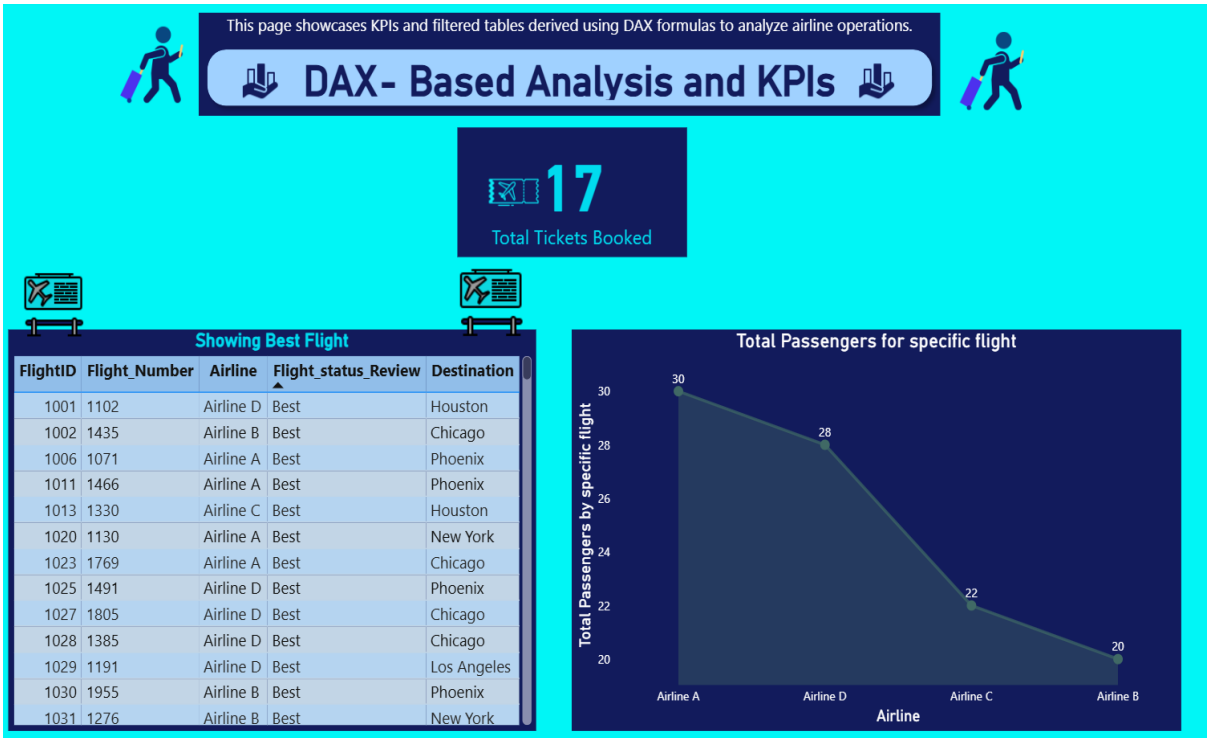
- **Table Visual of Best Flights**

Showing Best Flight

FlightID	Flight_Number	Airline	Flight_status_Review	Destination
1001	1102	Airline D	Best	Houston
1002	1435	Airline B	Best	Chicago
1006	1071	Airline A	Best	Phoenix
1011	1466	Airline A	Best	Phoenix
1013	1330	Airline C	Best	Houston
1020	1130	Airline A	Best	New York
1023	1769	Airline A	Best	Chicago
1025	1491	Airline D	Best	Phoenix
1027	1805	Airline D	Best	Chicago
1028	1385	Airline D	Best	Chicago
1029	1191	Airline D	Best	Los Angeles
1030	1955	Airline B	Best	Phoenix
1031	1276	Airline B	Best	New York

This visual shows a filtered table displaying only 'Best' flights using a DAX measure or filter based on the conditional column.

I created multiple DAX measures to perform key calculations using the data model. These calculations were then visualized on this page to provide focused insights using custom logic.



Task 5: Visualization and Interactive Features

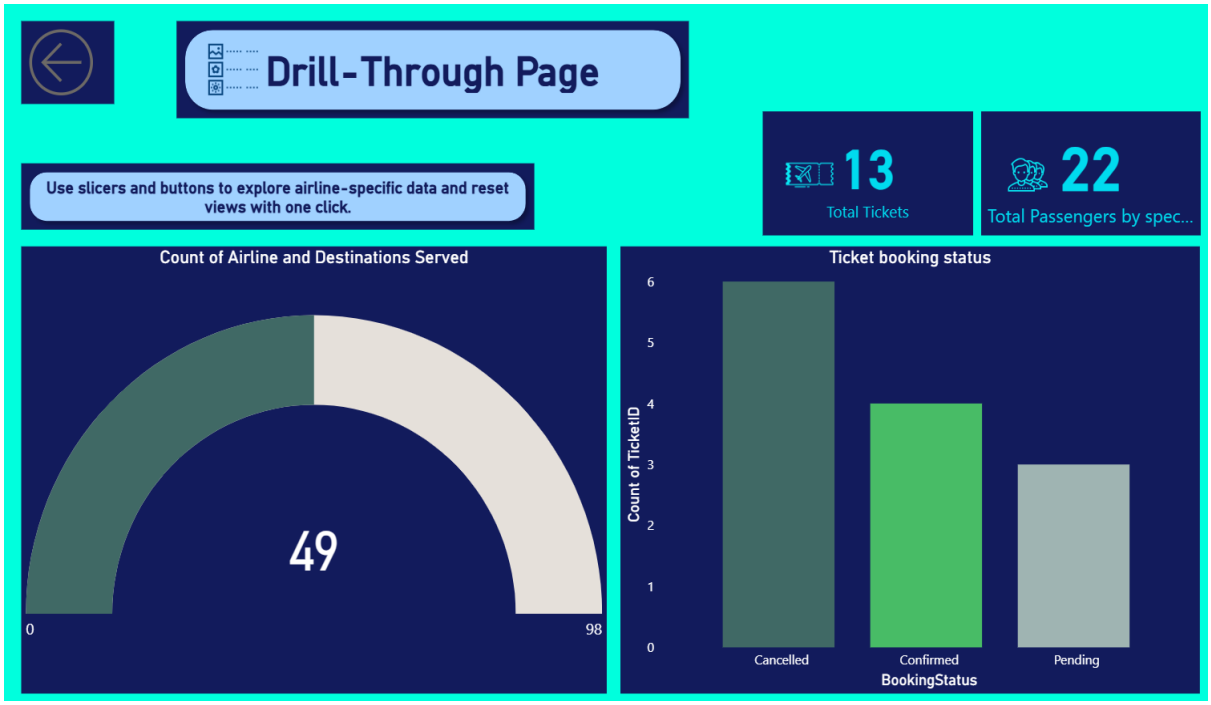
I created several visuals to explore the data interactively:

- **Passenger count by airline** using bar charts
- **Ticket booking statuses** using donut or pie charts
- **Flights by airline and destination** using stacked bar charts

To enhance user experience, I added **interactive slicers** for Airline and Destination. These slicers help users quickly filter and explore relevant data.

I also created **drillthrough pages**, allowing the user to right-click on an airline and navigate to a dedicated page showing its detailed data. Additionally, I used **buttons with bookmarks** for quick views of filtered dashboards.

This screenshot shows all key visuals and interactive features created. It includes a bar chart displaying passenger count by airline, a donut chart for ticket booking statuses, and a matrix showing flights by airline and destination.



*This screenshot displays a dedicated drillthrough page for **Airline C**. I used Power BI's **Drillthrough feature** to allow users to right-click on any airline (from a chart or slicer) and navigate to a separate page that shows detailed visuals and metrics specific to that airline.*

On this page, I've included key metrics such as passenger count, total tickets, flight distribution, and destination insights — all filtered automatically for Airline C. This helps decision-makers analyze performance and trends for a single airline in a focused way.

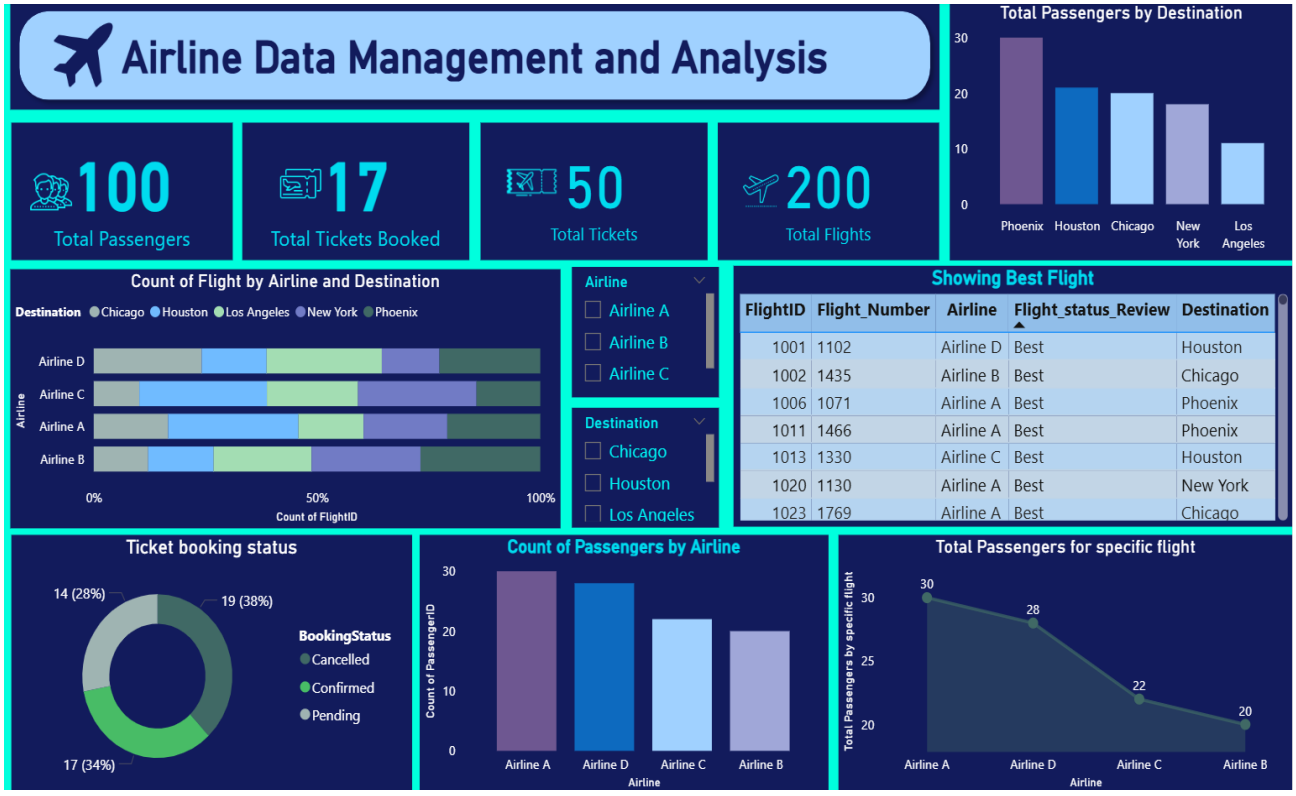
Task 6: Final Dashboard and Power BI Service

I designed the final dashboard with a clean layout using Power BI Desktop and then published it to Power BI Service. The dashboard includes multiple pages showing visual insights, KPIs, interactive filters, and detailed airline-specific data.

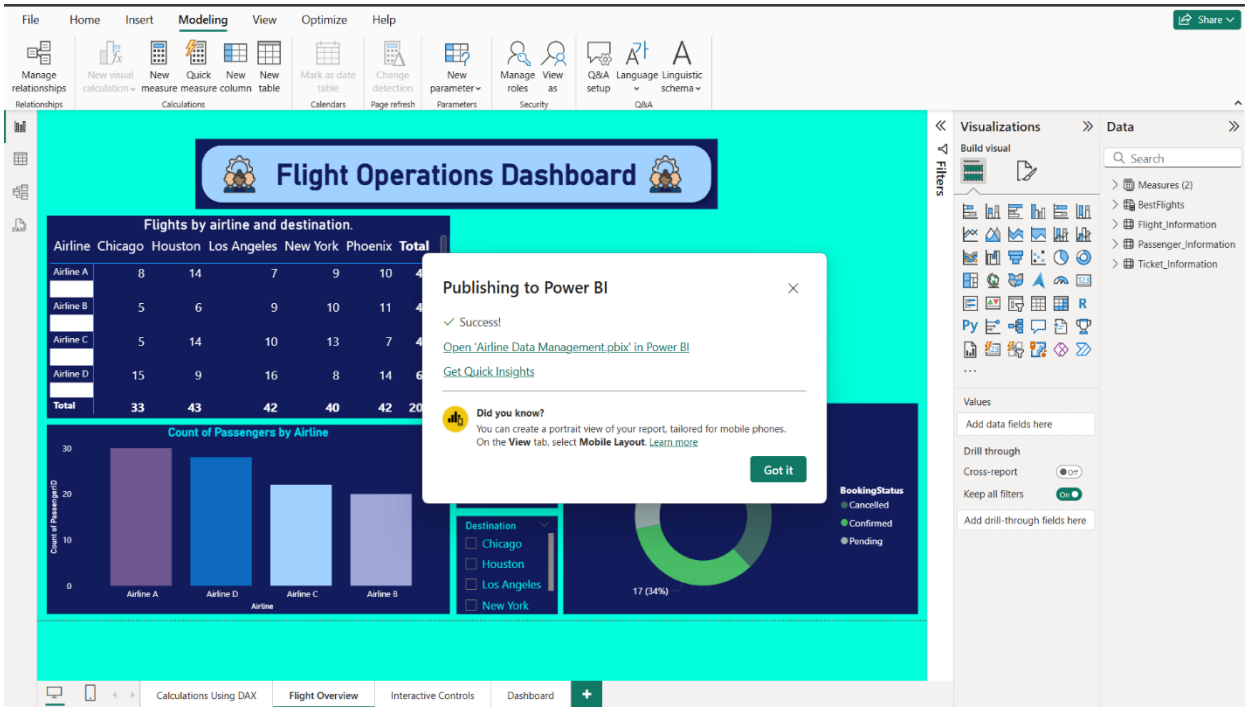
- I configured **Row-Level Security (RLS)** so that users assigned to the "Airline A" role only see data related to Airline A.
- I also set up a **scheduled refresh at 5 PM daily** by connecting the dataset to Excel files stored on OneDrive and enabling refresh settings in the Power BI Service.

These steps ensure the dashboard is both secure and automatically up-to-date.

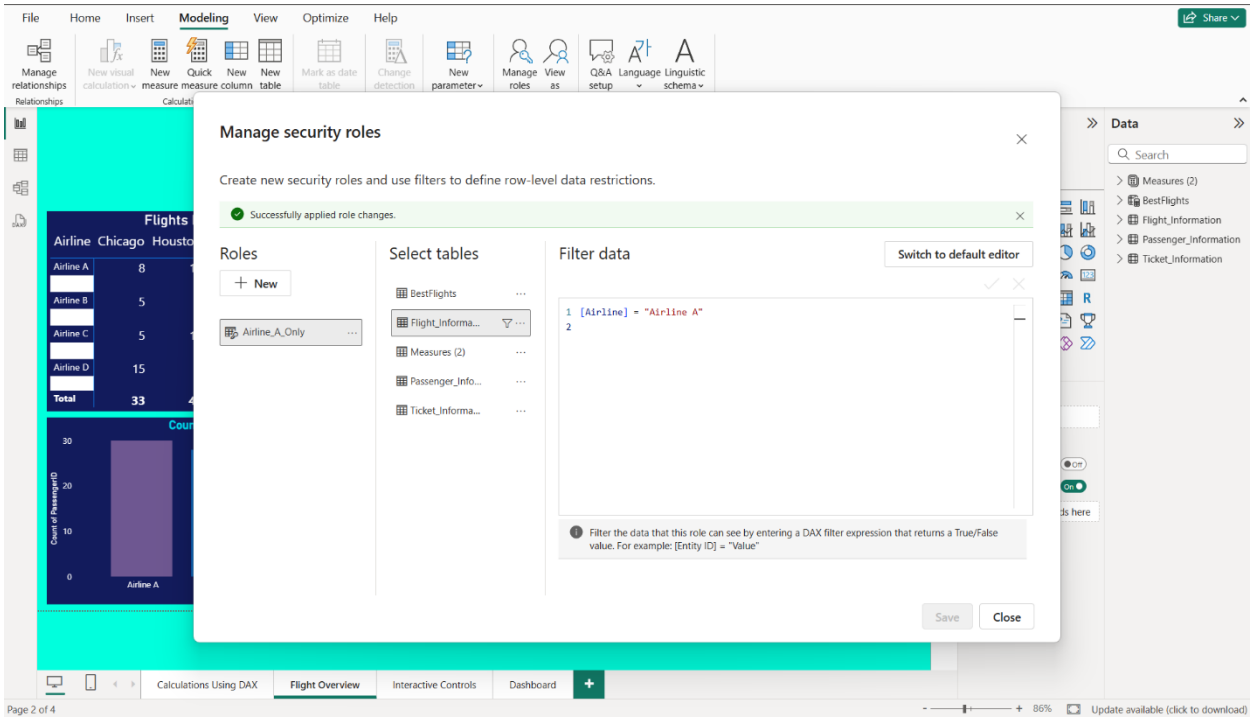
Final Dashboard View (Published)



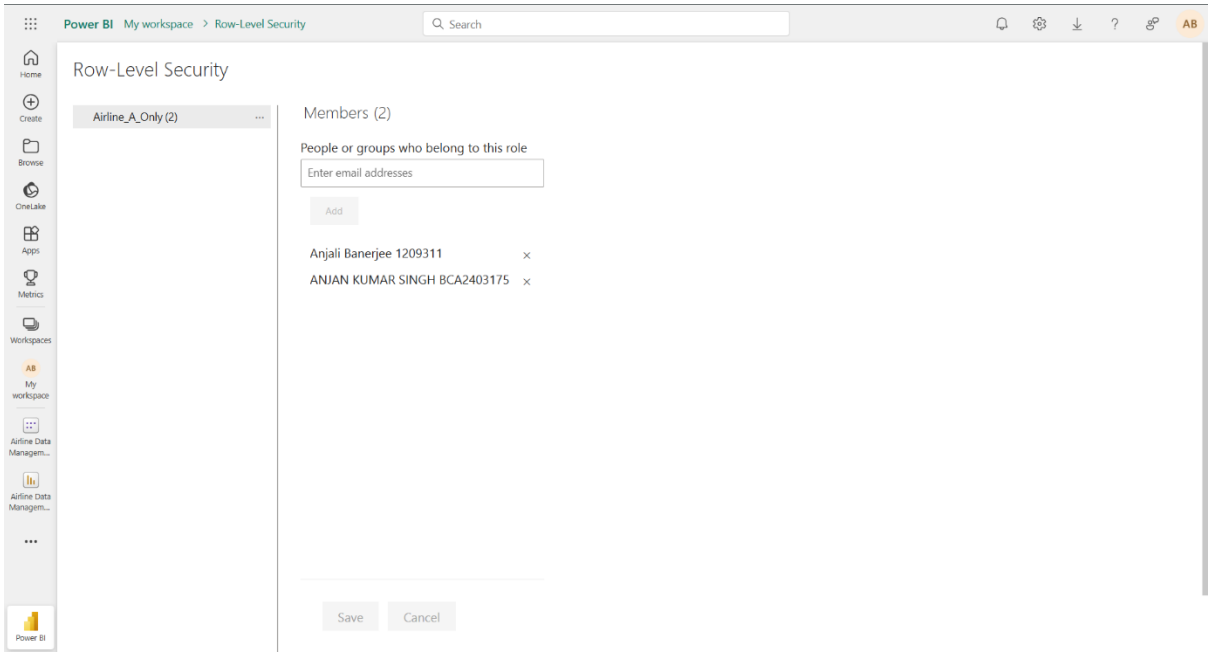
This is the complete dashboard view as published to Power BI Service, showing how visuals are arranged and grouped meaningfully.



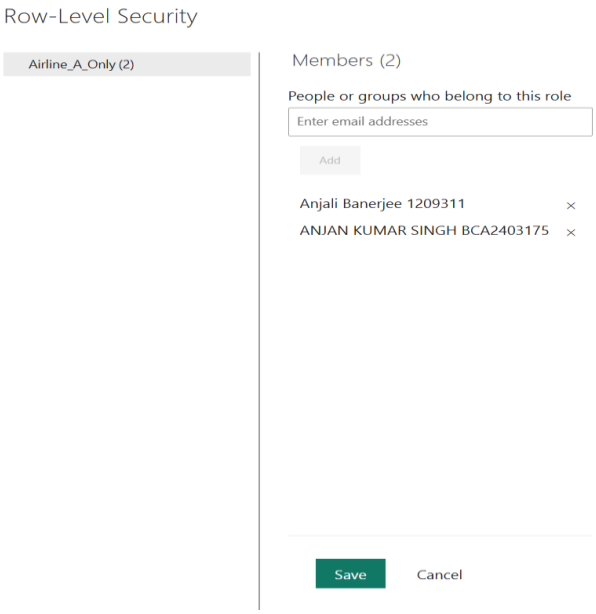
This screenshot displays published dashboard on power BI sercice.



This screenshot shows the role setup that restricts Airline A users to only view data relevant to them using [Airline] = 'Airline A' logic.



This shows how the Airline A role was assigned to a specific user in Power BI Service under the Security tab.



This shows a close view of how the Airline A role was assigned to a specific user in Power BI Service under the Security tab.

Conclusion

This Power BI project has provided me with valuable hands-on experience in the complete data analysis life cycle — from data cleaning and modeling to creating calculated measures and building interactive dashboards.

I began by cleaning and preparing the datasets (Flight_Information, Ticket_Information, and Passenger_Information) in Power Query Editor, ensuring the data was reliable and consistent. By

establishing proper relationships using FlightID and understanding cardinality, I laid a strong foundation for accurate data modeling.

Using DAX, I created custom calculations such as total passengers, ticket counts, and filtered flight views, which added deep analytical power to the report. The visuals — including bar charts, donut charts, and stacked visuals — offered key insights into airline performance, ticket statuses, and destinations served.

Interactive features like slicers, drillthrough pages, quick views, and bookmarks made the dashboard dynamic and user-friendly. I also implemented Row-Level Security (RLS) to restrict data access for specific airlines and scheduled a daily refresh at 5 PM to keep the report updated.

Overall, this project has helped me understand how to convert raw datasets into an insightful and interactive dashboard using Power BI. It has strengthened my analytical thinking, data modeling, and visualization skills — and prepared me for real-world reporting tasks in data-driven environments.