

# Airline Data Management and Analysis Using Power Bi

## ⇒ Project Title: -

Airline Data Management and Analysis Using Power BI

## ⇒ 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.

## ⇒ Datasets Used: -

- **Flight Information:** FlightID, FlightNumber, Airline, Destination, Status
- **Passenger Information:** PassengerID, FlightID, SeatNumber
- **Ticket Information:** TicketID, FlightID, BookingStatus

## 1) Data Preparation and Cleaning

- **Extract and transform data in Power Query.**
- **Clean data: remove duplicates, handle missing values, & format columns.**
- **Deliverables: Screenshot of Power Query Editor showing cleaned data.**

The top screenshot displays a table with the following data:

FlightID	FlightNumber	Airline	Destination	Status
1	1001	FL1102	Airline D	Houston
2	1002	FL1405	Airline B	Chicago
3	1003	FL1980	Airline A	New York
4	1004	FL1270	Airline C	Chicago
5	1005	FL1308	Airline C	New York
6	1006	FL1071	Airline A	Phoenix
7	1007	FL1700	Airline C	Los Angeles
8	1008	FL1020	Airline C	Los Angeles
9	1009	FL1684	Airline A	Los Angeles
10	1010	FL1121	Airline D	Chicago
11	1011	FL1466	Airline A	Phoenix
12	1012	FL1214	Airline D	New York
13	1013	FL1330	Airline C	Houston
14	1014	FL1458	Airline C	New York
15	1015	FL1087	Airline C	Houston
16	1016	FL1372	Airline B	New York
17	1017	FL1059	Airline D	Phoenix
18	1018	FL1871	Airline B	Houston
19	1019	FL1668	Airline B	Chicago
20	1020	FL1130	Airline A	New York
21	1021	FL1661	Airline B	New York
22	1022	FL1308	Airline A	Houston
23	1023	FL1749	Airline A	Chicago
24	1024	FL1343	Airline B	Chicago
25	1025	FL1401	Airline D	Phoenix
26	1026	FL1413	Airline D	Chicago
27	1027	FL1805	Airline D	Chicago
28	1028	FL1385	Airline D	Chicago

The bottom screenshot displays a table with the following data:

TicketID	FlightID	BookingStatus
1	5001	1178 Pending
2	5002	1078 Confirmed
3	5003	1117 Cancelled
4	5004	1120 Cancelled
5	5005	1137 Cancelled
6	5006	1162 Pending
7	5007	1076 Pending
8	5008	1085 Cancelled
9	5009	1061 Cancelled
10	5010	1040 Cancelled
11	5011	1064 Pending
12	5012	1150 Cancelled
13	5013	1060 Cancelled
14	5015	1093 Confirmed
15	5016	1072 Pending
16	5017	1011 Cancelled
17	5018	1105 Cancelled
18	5019	1014 Confirmed
19	5021	1030 Confirmed
20	5023	1165 Confirmed
21	5024	1005 Confirmed
22	5025	1087 Cancelled
23	5026	1123 Cancelled
24	5028	1154 Pending
25	5029	1062 Pending
26	5030	1132 Pending
27	5031	1089 Pending
28	5032	1042 Confirmed

# Airline Data Management and Analysis Using Power Bi

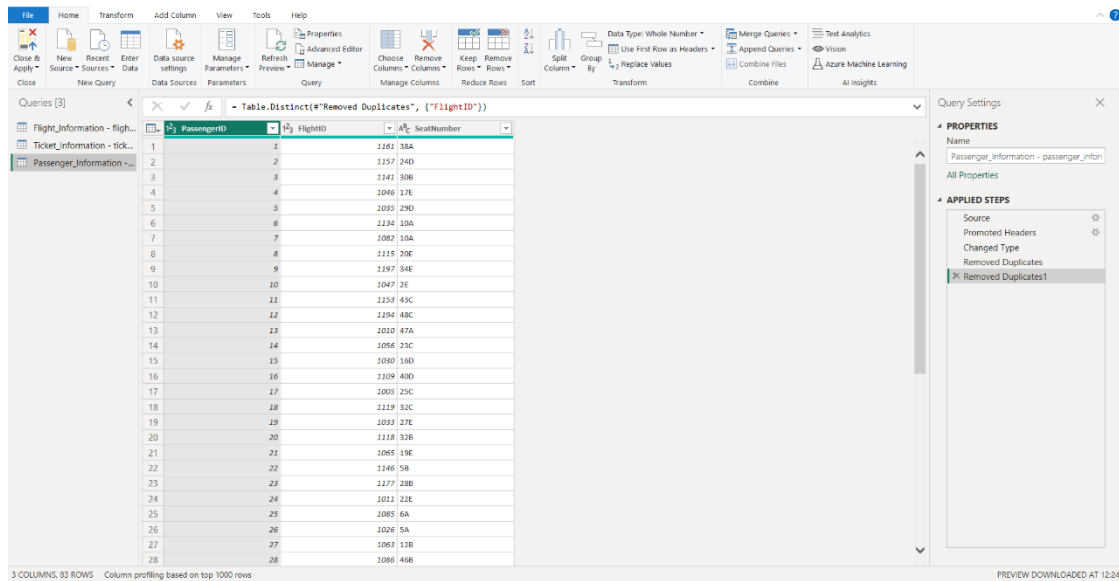


Table: Table.Distinct(#"Removed Duplicates", {"FlightID"})

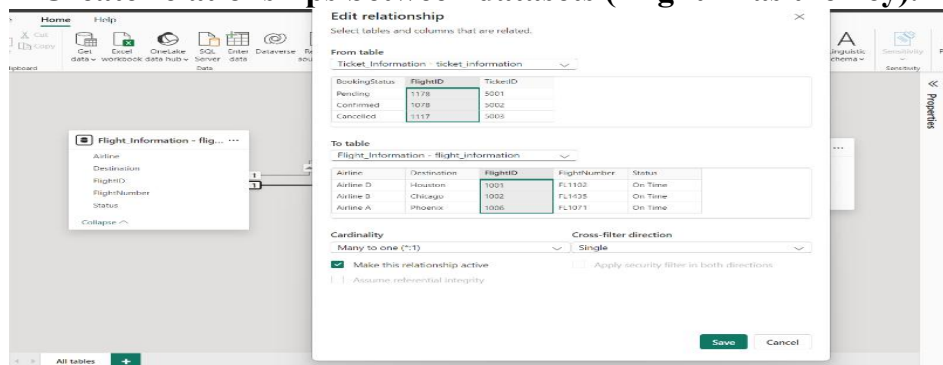
PassengerID	FlightID	SeatNumber
1	1141	38A
2	1157	24D
3	1141	30B
4	1046	17E
5	1033	29D
6	1134	10A
7	1082	10A
8	1115	20E
9	1197	34E
10	1047	2E
11	1153	43C
12	1194	48C
13	1010	47A
14	1056	23C
15	1030	19D
16	1109	40D
17	1009	25C
18	1119	32C
19	1037	27E
20	1118	31B
21	1065	19E
22	1146	5A
23	1177	28B
24	1011	22E
25	1083	6A
26	1026	5A
27	1063	12B
28	1086	46B

In the Data Preparation and Cleaning stage, the data was loaded and transformed using Power Query Editor:

- **Loading Data to Power Query:**
  - ✓ First, the data was loaded into Power BI using the Get Data option.
  - ✓ Once the data was loaded, the Power Query Editor was used to transform and clean the dataset.
- **Removing Duplicates:**
  - ✓ Duplicate entries were identified and removed to ensure the dataset had only unique records.
  - ✓ missing values were handled, and columns like FlightID, PassengerID, and TicketID
- **Handling Missing Values:**
  - ✓ Missing or null values were handled using various techniques
  - ✓ Remove the blank rows this help remove the null cells and filling the required data like n/a without keeping it empty

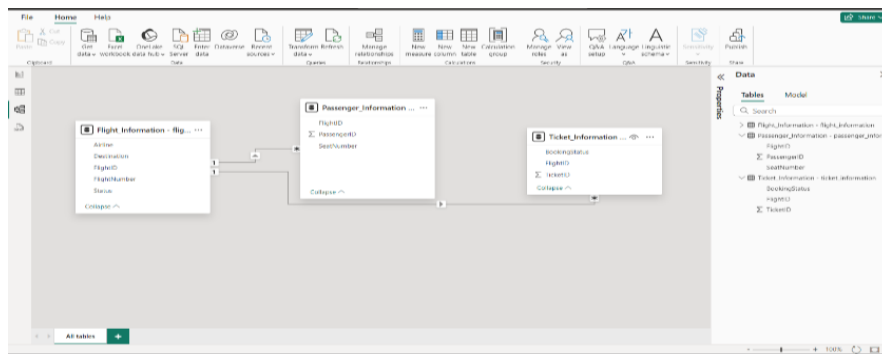
## 2) Data Modelling: -

- **Create relationships between datasets (FlightID as the key).**



# Airline Data Management and Analysis Using Power Bi

- **Understand cardinality and configure the model appropriately.**



- **Deliverables: Screenshot of the data model with relationships.**

In Data Modeling, relationships were created between the different datasets to enable seamless analysis and reporting.

- **Building Relationships:**  
The Flight\_Information, Passenger\_Information, and Ticket\_Information tables were connected using FlightID as the primary key.
- **One-to-Many Relationships:**  
One-to-Many relationships were set to ensure that each flight can have multiple passengers and tickets, but each passenger and ticket can be associated with only one flight.
- **Cardinality:**  
The Cardinality was set to control how data is filtered between tables. This step ensures that the filters work correctly and that no data is missed or misrepresented.

### 3) Enhanced Data Insights: -

- **Add a conditional column to classify flights as "Best" or "To Be Improved" based on status.**
- **Use "Column from Examples" to extract the flight number from FlightNumber.**
- **Deliverables: Screenshot of the transformed data**

Enhanced Data Insights were added by creating conditional columns and calculated fields to add business context to the data.

- **Creating Conditional Columns:**  
Flights were classified as either "Best" or "To Be Improved" based on their status, using the Conditional Column feature in Power Query.
- **Column With Examples:**  
The FlightNumber was extracted from the Flight\_Information table using the Column from Examples feature. This added a new field for analysis and visualization.

# Airline Data Management and Analysis Using Power Bi

**Add Conditional Column**

Add a conditional column that is computed from the other columns or values.

New column name:

Condition	Operator	Value	Then	Output
If Status	contains	On Time	Then	Best
Else if Status	contains	Delayed	Then	To Be Improved
Else if Status	contains	Cancelled	Then	To Be Improved

Table.TransformColumnTypes(\*Removed Duplicates\*,{{"FlightID", Int64.Type}})

FlightID	FlightNumber	Airline	Destination	Status
1	FL1343	Airline B	Chicago	Delayed
2	FL1491	Airline D	Phoenix	On Time
3	FL1413	Airline D	Chicago	Cancelled
4	FL1805	Chicago	Chicago	On Time
5	FL1385	Airline D	Chicago	On Time

**Add Column From Examples**

Enter sample values to create a new column (Ctrl+Enter to apply).

Transform:

Table.AddColumn(\*Changed Type\*, "classified\_flights\_status", each if Text.Contains([Status], "On Time") then "Best " else if

FlightID	FlightNumber	Airline	Destination	Status	classified_flights_status
1	FL1102	Airline D	Houston	On Time	Best
2	FL1435	Airline B	Chicago	On Time	Best
3	FL1860	Airline A	New York	Cancelled	To Be Improved
4	FL1270	Airline C	Chicago	Delayed	To Be Improved
5	FL1106	Airline C	New York	Delayed	To Be Improved
6	FL1071	Airline A	Phoenix	On Time	Best
7	FL1700	Airline C	Los Angeles	Cancelled	To Be Improved
8	FL1020	Airline C	Los Angeles	Delayed	To Be Improved
9	FL1614	Airline A	Los Angeles	Cancelled	To Be Improved
10	FL1121	Airline D	Chicago	Cancelled	To Be Improved
11	FL1466	Airline A	Phoenix	On Time	Best
12	FL1214	Airline D	New York	Delayed	To Be Improved
13	FL1330	Airline C	Houston	On Time	Best
14	FL1458	Airline C	New York	Delayed	To Be Improved
15	FL1087	Airline C	Houston	Delayed	To Be Improved
16	FL1372	Airline B	New York	Delayed	To Be Improved
17	FL1099	Airline D	Phoenix	Delayed	To Be Improved
18	FL1871	Airline B	Houston	Delayed	To Be Improved
19	FL1863	Airline B	Chicago	Cancelled	To Be Improved
20	FL1130	Airline A	New York	On Time	Best
21	FL1861	Airline B	New York	Cancelled	To Be Improved
22	FL1308	Airline A	Houston	Delayed	To Be Improved
23	FL1769	Airline A	Chicago	On Time	Best
24	FL1343	Airline B	Chicago	Delayed	To Be Improved
25	FL1491	Airline D	Phoenix	On Time	Best
26	FL1413	Airline D	Chicago	Cancelled	To Be Improved
27	FL1805	Airline D	Chicago	On Time	Best
28	FL1385	Airline D	Chicago	On Time	Best

**Add Column From Examples**

Enter sample values to create a new column (Ctrl+Enter to apply).

Transform:

Table.AddColumn(\*Changed Type\*, "classified\_flights\_status", each if Text.Contains([Status], "On Time") then "Best " else if

FlightID	FlightNumber	Airline	Destination	Status	Only_FlightNumber
1	FL1102	Airline D	Houston	On Time	1102
2	FL1435	Airline B	Chicago	On Time	1435
3	FL1860	Airline A	New York	Cancelled	1860
4	FL1270	Airline C	Chicago	Delayed	1270
5	FL1106	Airline C	New York	Delayed	1106
6	FL1071	Airline A	Phoenix	On Time	1071
7	FL1700	Airline C	Los Angeles	Cancelled	1700
8	FL1020	Airline C	Los Angeles	Delayed	1020
9	FL1614	Airline A	Los Angeles	Cancelled	1614

# Airline Data Management and Analysis Using Power Bi

File

Home

Transform

Add Column

View

Tools

Help

Conditional Column

Index Column

Duplicate Column

Format

Parse

From Text

From Number

From Date & Time

Merge Columns

Trigonometry

Rounding

Information

Date

Time

Duration

Test Analytics

Vision

Azure Machine Learning

AI Insights

Column From Custom

Invoke Custom

Examples

Column

Function

General

Queries [3]

Table.AddColumn(#"Added Conditional Column", "Only\_FlightNumber", each Text.AfterDelimiter([FlightNumber], "L"), type text)

Flight\_Information - flight\_...

Ticket\_Information - tick...

Passenger\_Information - ...

	FL#	FlightNumber	Airline	Destination	Status	classified_Flights_status	Only_FlightNumber
1	1001	FL1102	Airline D	Houston	On Time	Best	1102
2	1002	FL1435	Airline B	Chicago	On Time	Best	1435
3	1003	FL1860	Airline A	New York	Cancelled	To Be Improved	1860
4	1004	FL1270	Airline C	Chicago	Delayed	To Be Improved	1270
5	1005	FL1106	Airline C	New York	Delayed	To Be Improved	1106
6	1006	FL1071	Airline A	Phoenix	On Time	Best	1071
7	1007	FL1700	Airline C	Los Angeles	Cancelled	To Be Improved	1700
8	1008	FL1020	Airline C	Los Angeles	Delayed	To Be Improved	1020
9	1009	FL1614	Airline A	Los Angeles	Cancelled	To Be Improved	1614
10	1010	FL1121	Airline D	Chicago	Cancelled	To Be Improved	1121
11	1011	FL1466	Airline A	Phoenix	On Time	Best	1466
12	1012	FL1214	Airline D	New York	Delayed	To Be Improved	1214
13	1013	FL1330	Airline C	Houston	On Time	Best	1330
14	1014	FL1458	Airline C	New York	Delayed	To Be Improved	1458
15	1015	FL1087	Airline C	Houston	Delayed	To Be Improved	1087
16	1016	FL1372	Airline B	New York	Delayed	To Be Improved	1372
17	1017	FL1099	Airline D	Phoenix	Delayed	To Be Improved	1099
18	1018	FL1871	Airline B	Houston	Delayed	To Be Improved	1871
19	1019	FL1663	Airline B	Chicago	Cancelled	To Be Improved	1663
20	1020	FL1130	Airline A	New York	On Time	Best	1130
21	1021	FL1661	Airline B	New York	Cancelled	To Be Improved	1661
22	1022	FL1308	Airline A	Houston	Delayed	To Be Improved	1308
23	1023	FL1769	Airline A	Chicago	On Time	Best	1769
24	1024	FL1343	Airline B	Chicago	Delayed	To Be Improved	1343
25	1025	FL1491	Airline D	Phoenix	On Time	Best	1491
26	1026	FL1413	Airline D	Chicago	Cancelled	To Be Improved	1413
27	1027	FL1805	Airline D	Chicago	On Time	Best	1805
28							

Query Settings

PROPERTIES

Name

Flight\_Information - flight\_Information

All Properties

APPLIED STEPS

Source

Promoted Headers

Removed Duplicates

Changed Type

Added Conditional Column

Inserted Text After Delimiter

7 COLUMNS, 200 ROWS

Column profiling based on top 1000 rows

PREVIEW DOWNLOADED AT 19:12

## 4) Calculations Using DAX: -

### ■ Calculate:

#### ❖ Total passengers for a specific flight.

In DAX Calculations, various metrics were calculated to provide deeper insights into the data.

- Total Passengers:

A measure was created using the DAX function to count the number of passengers per flight.

- Dax Formula:

**TotalPassengers = COUNTROWS(Passenger\_Information)**

- Total Tickets:

Another measure was created to count the total number of tickets booked.

- Dax Formula:

**TotalTickets = COUNTROWS(Ticket\_Information)**

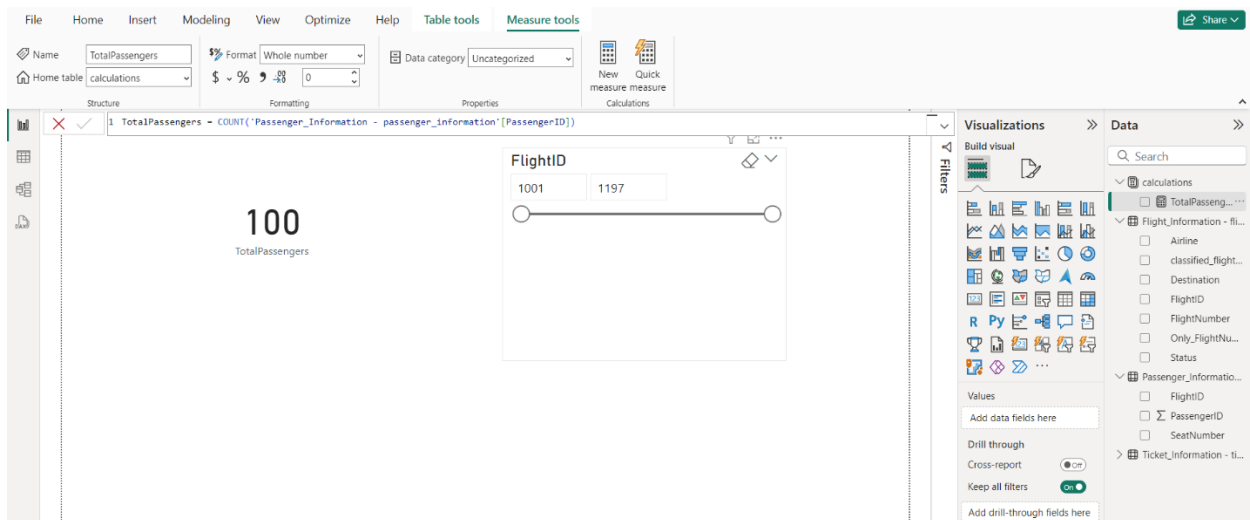
- Best Flights Table:

A new table was filtered to display only the flights classified as "Best".

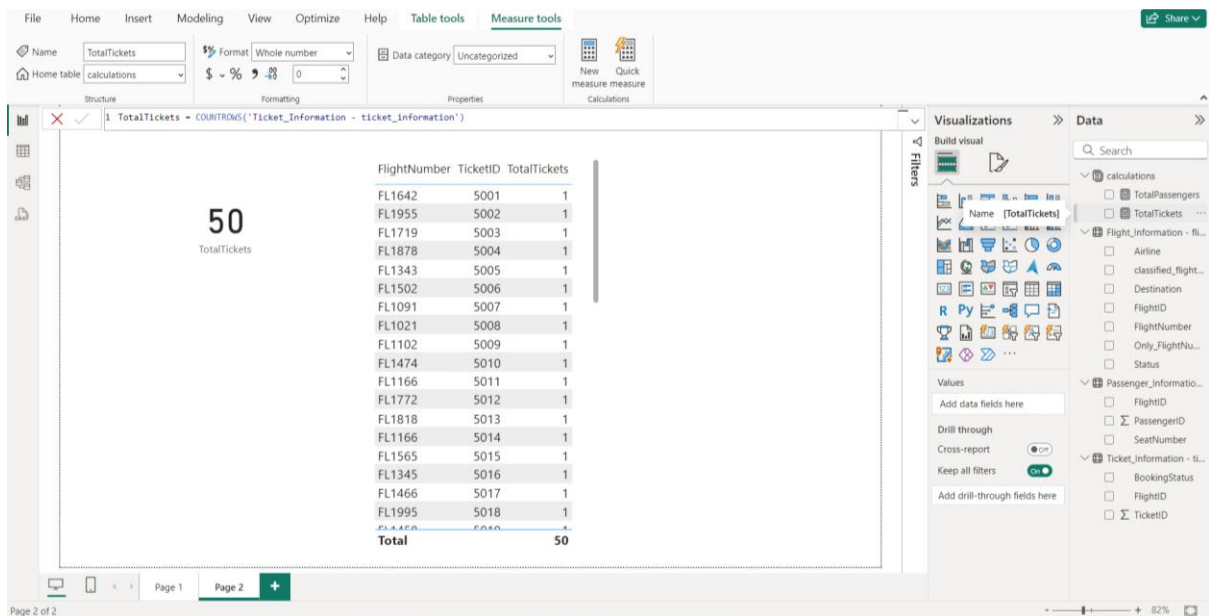
- Dax Formula:

**BestFlights = FILTER(Flight\_Information, Flight\_Information[Status] = "Best")**

# Airline Data Management and Analysis Using Power Bi



## ❖ Total tickets booked.



## ❖ Filtered table showing "Best" flights only.

- Deliverables: Screenshot of DAX calculations and results.

1 BestFlightsTable = FILTER(Flight\_Information, Flight\_Information[classified\_flights\_status] = 'Best')

FlightID	FlightNumber	Airline	Destination	Status	classified_flights_status	Only_FlightNumber
1001	FL1102	Airline D	Houston	On Time	Best	1102
1002	FL1435	Airline E	Chicago	On Time	Best	1435
1006	FL1071	Airline A	Phoenix	On Time	Best	1071
1011	FL1466	Airline A	Phoenix	On Time	Best	1466
1013	FL1330	Airline C	Houston	On Time	Best	1330
1020	FL1130	Airline A	New York	On Time	Best	1130
1023	FL1769	Airline A	Chicago	On Time	Best	1769
1025	FL1491	Airline D	Phoenix	On Time	Best	1491
1027	FL1805	Airline D	Chicago	On Time	Best	1805
1028	FL1385	Airline D	Chicago	On Time	Best	1385
1029	FL1191	Airline D	Los Angeles	On Time	Best	1191
1030	FL1955	Airline B	Phoenix	On Time	Best	1955
1031	FL1276	Airline B	New York	On Time	Best	1276
1033	FL1459	Airline D	New York	On Time	Best	1459
1034	FL1213	Airline B	Phoenix	On Time	Best	1213
1036	FL1252	Airline D	Phoenix	On Time	Best	1252
1039	FL1560	Airline B	Chicago	On Time	Best	1560
1043	FL1681	Airline C	Houston	On Time	Best	1681

Data

- calculations
  - TotalPassengers
  - TotalTickets
  - BestFlightsTable
- classified\_flights\_status
- Destination
- FlightID
- FlightNumber
- Only\_FlightNumber
- Status
- Flight\_Information
- Passenger\_information
- Ticket\_information

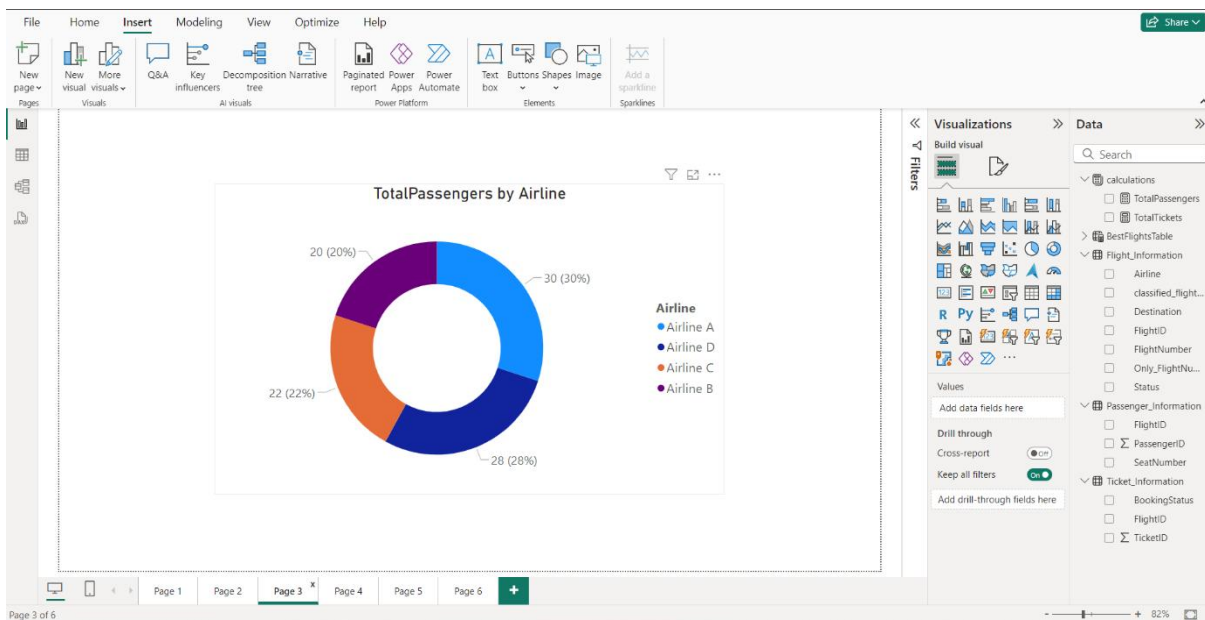
# Airline Data Management and Analysis Using Power Bi

## 5) Visualization and Interactive Features

### ▪ Create visuals for:

#### ❖ Passenger count by airline.

- Donut Chart: Passenger Counts Grouped by Airline
  - ✓ Data Field:Category: Airlines (e.g., Airline A, Airline B).
  - ✓ Values: Total Passenger Count (calculated using DAX measure: TotalPassengers).
- Visualization Details:
  - ✓ Chart Type: Donut Chart.
  - ✓ Data Colors: Distinct colors for each airline for better clarity.
  - ✓ Percentage Display: Enabled to show each airline's share of total passengers.

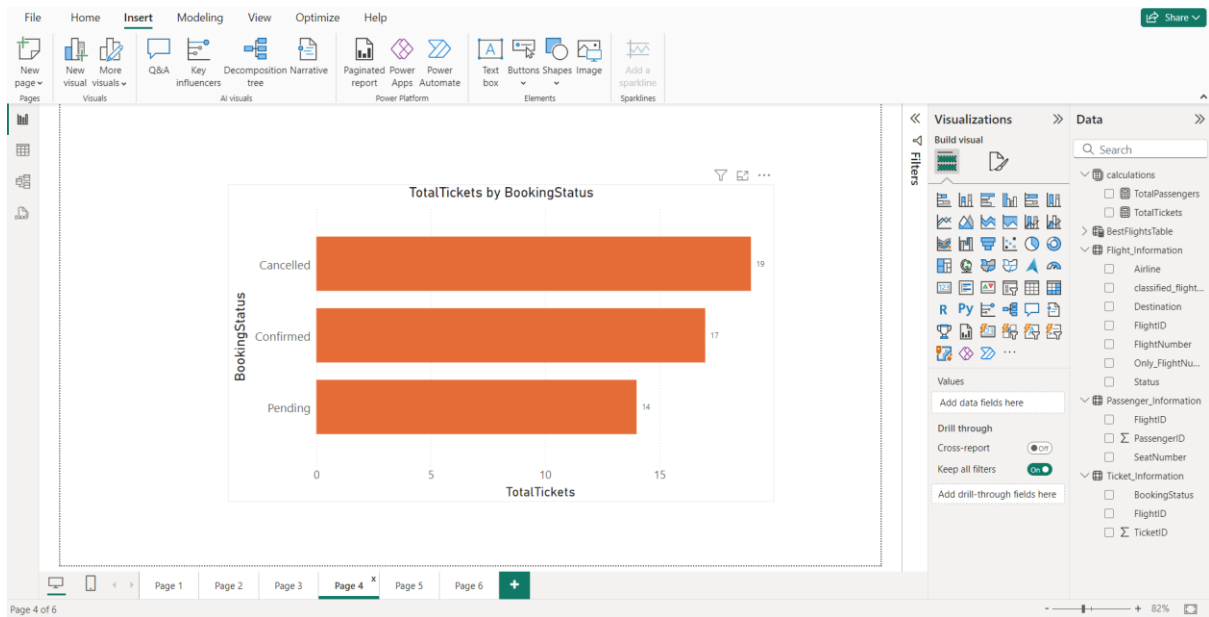


#### ❖ Ticket booking statuses.

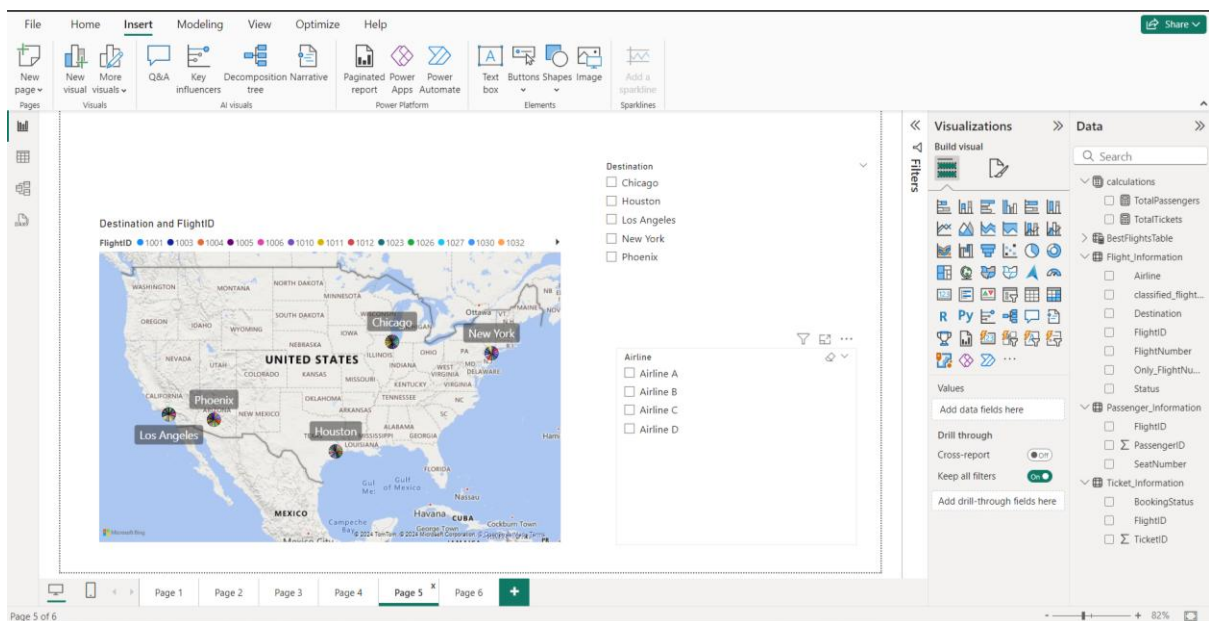
- Bar Chart: Ticket Booking Statuses
  - ✓ X-Axis: Ticket Status (Booked, Cancelled, Pending).
  - ✓ Y-Axis: Count of Tickets (using DAX measure: TotalTickets).
- Visualization Details:
  - ✓ Chart Type: Clustered Bar Chart.
  - ✓ Data Labels: Enabled for clear visibility of ticket counts.
  - ✓ Colors: Each bar represents a ticket status with a distinct color for better differentiation.



# Airline Data Management and Analysis Using Power Bi



## ❖ Flights by airline and destination.

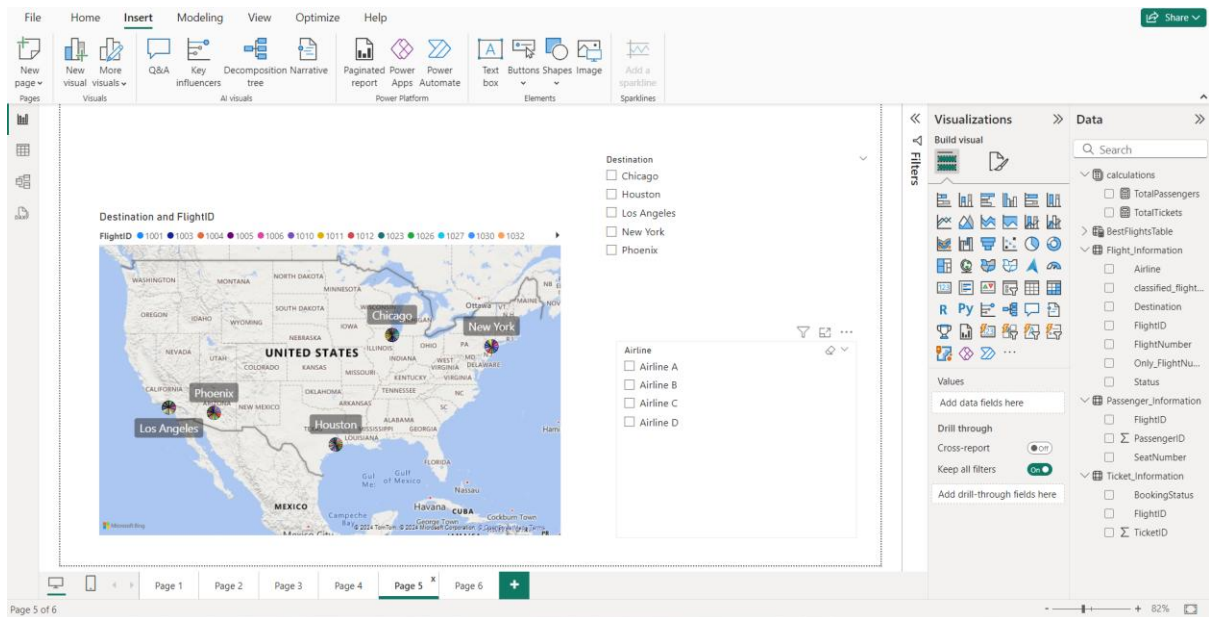


- Map Visual: Flights by Destination and Airline
  - ✓ Location Field: Destination City (e.g., New York, London, Tokyo).
  - ✓ Legend: Airlines (e.g., Airline A, Airline B).
- Visualization Details:
  - ✓ Chart Type: Map Visual.
  - ✓ Size: Proportional to the number of flights to each destination.
  - ✓ Tooltips: Display total flights and airline name on hover for detailed insights.

- Add interactive features for:
  - ❖ Destination and Airline.

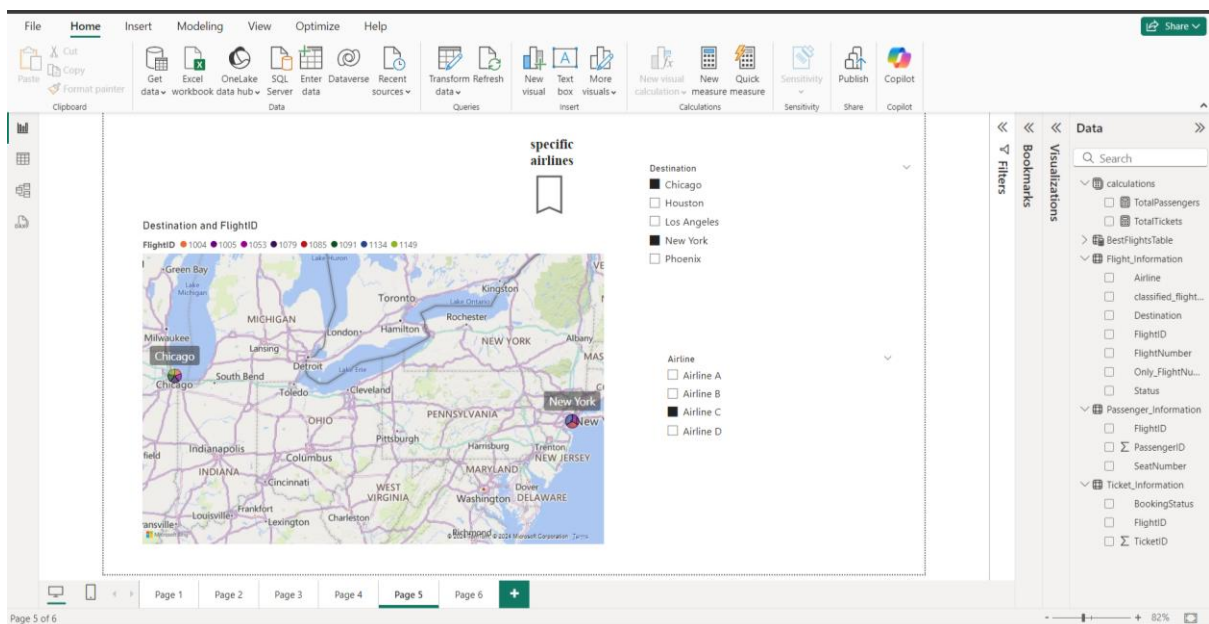


# Airline Data Management and Analysis Using Power Bi



- Slicers
  - Destination Slicers:
    - ✓ Field: Destination City.
    - ✓ Selection: Multi-select enabled for filtering multiple cities simultaneously.
  - Airline Slicers:
    - ✓ Field: Airline Name.
    - ✓ Selection: Dropdown style for compact display.

## ❖ Quick views.

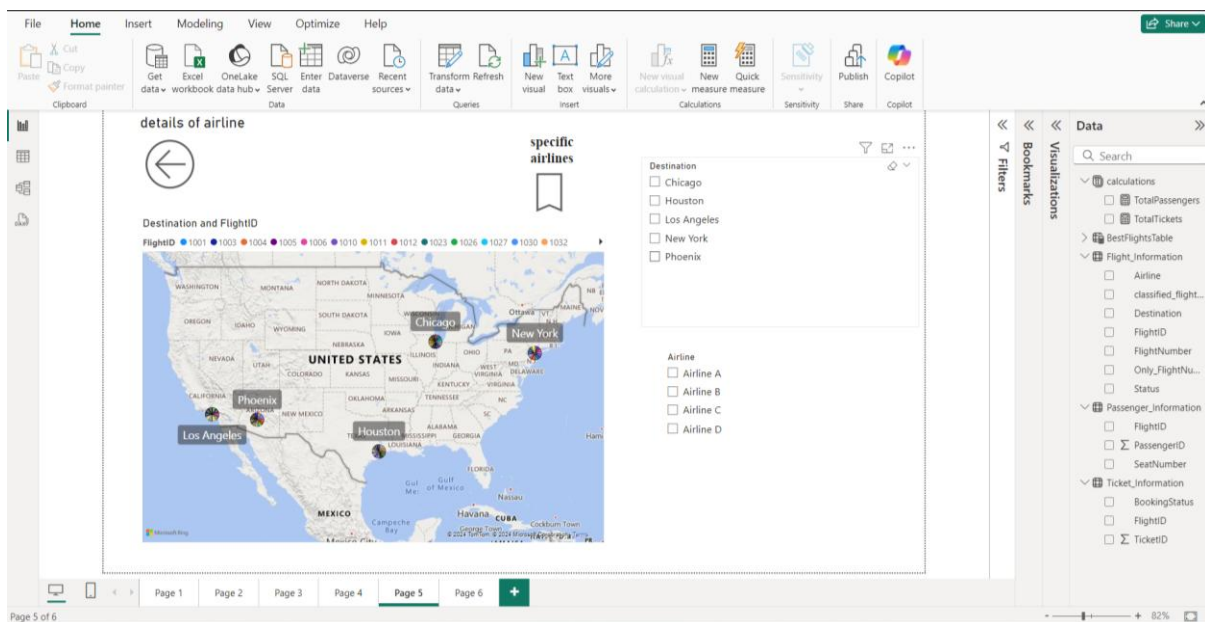


# Airline Data Management and Analysis Using Power Bi

- Predefined Bookmarks:
- Created bookmarks for common insights, such as:
  - ✓ Top Destinations: Displays cities with the highest passenger counts.
  - ✓ Airline Performance: Highlights metrics for all airlines.
- Buttons for Navigation:
  - ✓ Buttons linked to bookmarks were added for a user-friendly experience.
  - ✓ "View Top Destinations" button applies filters and navigates to the Top Destinations view.
  - ✓ "Reset View" button clears all filters and restores the default dashboard layout.

## ❖ Airline-specific pages.

### ▪ Deliverables: Screenshots of all visuals and interactive features.

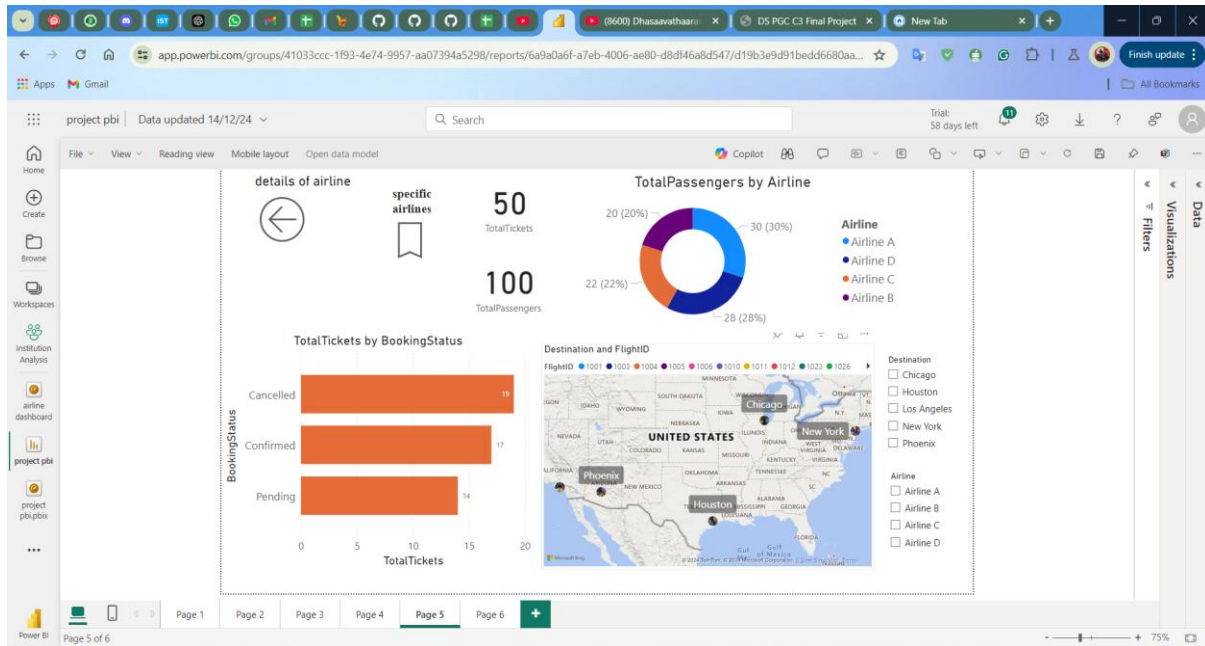


- Drill-Through Pages:
  - ✓ Separate pages were created for each airline to provide detailed, airline-specific insights.
  - ✓ Fields Used: Airline Name as the drill-through field.
  - ✓ Details Displayed on Drill-Through Pages:
    - Passenger counts for that airline,
    - Top destinations served by the airline,
    - Ticket statuses and flight performance metrics.

# Airline Data Management and Analysis Using Power Bi

## 6) Final Dashboard and Power BI Service

- Design a comprehensive dashboard with key visuals and insights.

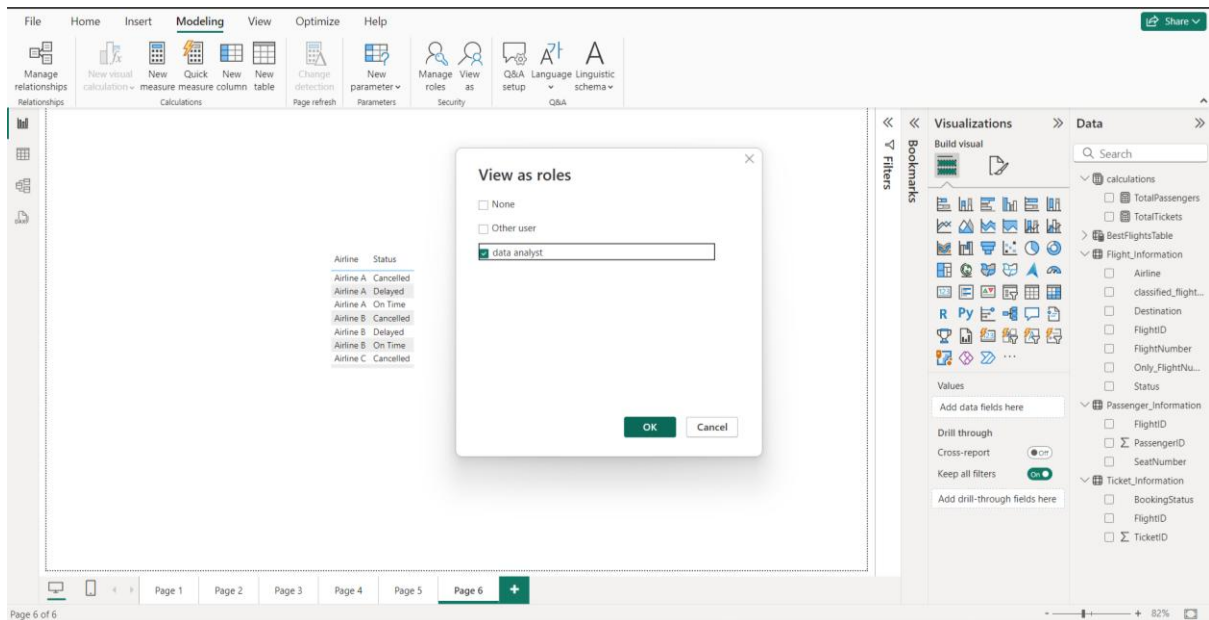


- Visuals Added:
  - ✓ **Passenger Counts by Airline** (Donut Chart).
  - ✓ **Ticket Statuses** (Clustered Bar Chart).
  - ✓ **Flights by Destination and Airline** (Map Visual).
- Layout Design: Arranged visuals in a grid pattern for clarity:
  - ✓ Top Section: Slicers for Destination and Airline.
  - ✓ Middle Section: Donut Chart and Bar Chart side by side for comparison.
  - ✓ Bottom Section: Map Visual for geographic insights.
- Formatting:
  - ✓ Used consistent fonts, colors, and borders for professional appearance.
  - ✓ Added a title: **"Airline Performance Dashboard"** for clear identification.
- By clicking in power bi service where we can create the final dashboard and publish as well.

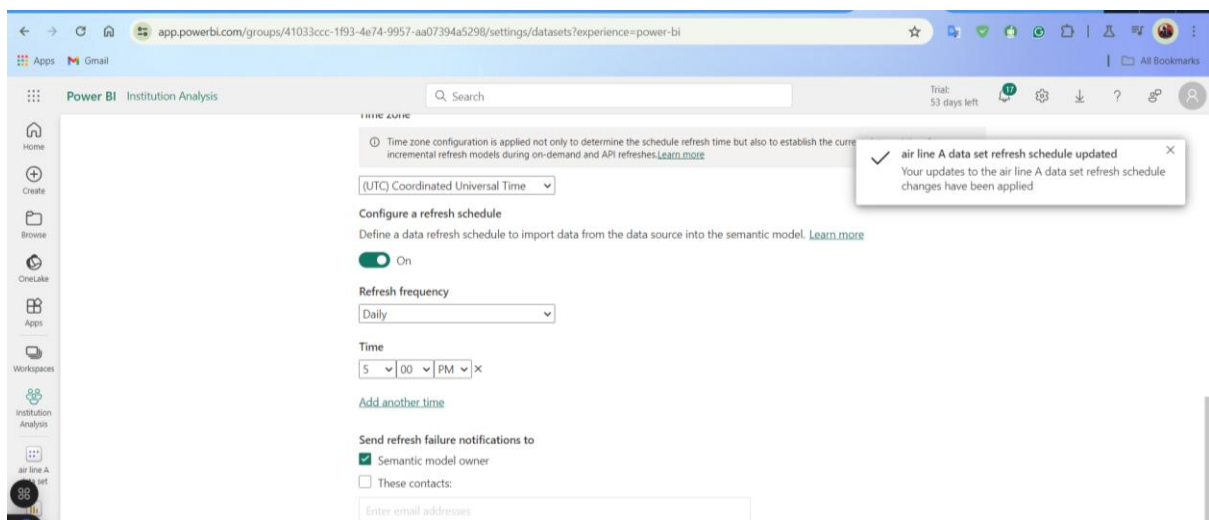
# Airline Data Management and Analysis Using Power Bi

## ■ Configure Row-Level Security (RLS) for Airline A data, assign it to a user.

- Define Roles:
  - ✓ Created a role called "data analyst".
  - ✓ Added a filter to the Airline Name column: [Airline] = "Airline A".
- Assign Users:
  - ✓ Assigned specific email IDs of internshala staff to the role in Power BI Service.
- Testing RLS:
  - ✓ Verified RLS functionality using the View as Role feature in Power BI Desktop.
- Users from Airline A can only see their airline's data, ensuring security and privacy.



## ■ Set up a schedule refresh at 5 PM daily.



## Airline Data Management and Analysis Using Power Bi

- Upload to Power BI Service: Published the dashboard to Power BI Service.
- Configure Scheduled Refresh:
  - ✓ Set up a refresh schedule from the settings in Power BI Service:
    - Frequency: Daily.
    - Time: 5 PM.
  - ✓ Ensured proper data source credentials were provided for seamless refresh.
- Test the Refresh:
  - ✓ Triggered a manual refresh to confirm the setup works as expected.
- Deliverables: Screenshot of the published dashboard and RLS configuration.

 **Note:- The video file size is large so I used this video link:**  
<https://drive.google.com/file/d/111o9N2Ok7XwnTPB4wHNCvCjUQhgz-qOh/view?usp=sharing>

**UMA MAHESWAR NETHI**  
**DATA SCIENCE LEARNER**