Airline Data Management and Analysis Using Power BI

Course: Data Visualization with Power BI

Name: Ravi Kant

Project Overview

This project is part of the **Data Visualization with Power BI** course and focuses on analysing and visualize airline data for operational insights, passenger management, and ticket booking trends using Power BI.

Datasets Used:

Flight_Information Ticket_Information Passenger_Information

1. **Flight Information:** Includes FlightID, FlightNumber, Airline, Destination, and Status.

2. **Passenger Information:** Includes PassengerID, FlightID, and SeatNumber.

3. **Ticket Information:** Includes TicketID, FlightID, and BookingStatus.

Methodology

Importing the Dataset

- Open Power BI Desktop.
- Click on Get Data → excel workbook.
- Load the dataset into Power BI all three data.

Tasks (1)

1. Data Preparation and Cleaning.

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

Explication: (1)

1.1 Extracting and Loading Data

Open Power BI Desktop.

Navigate to Home → Get Data → Excel/CSV (or your data source).

Select the relevant datasets:

Flight_Information

Passenger_Information

Ticket_Information

Click Transform Data to open Power Query Editor.

2.1 Data Cleaning Steps in Power Query

In Power Query, perform the following cleaning steps:

a): Remove Duplicates

select all columns of data (Flight_information, Passenger_information and Ticket_information).

Go to Home → Remove rows → Remove duplicates.

2.2: Handle Missing Values

Identify missing values using Transform → Replace Values or Remove Rows → Remove Blank Rows but there are no missing values in the data.

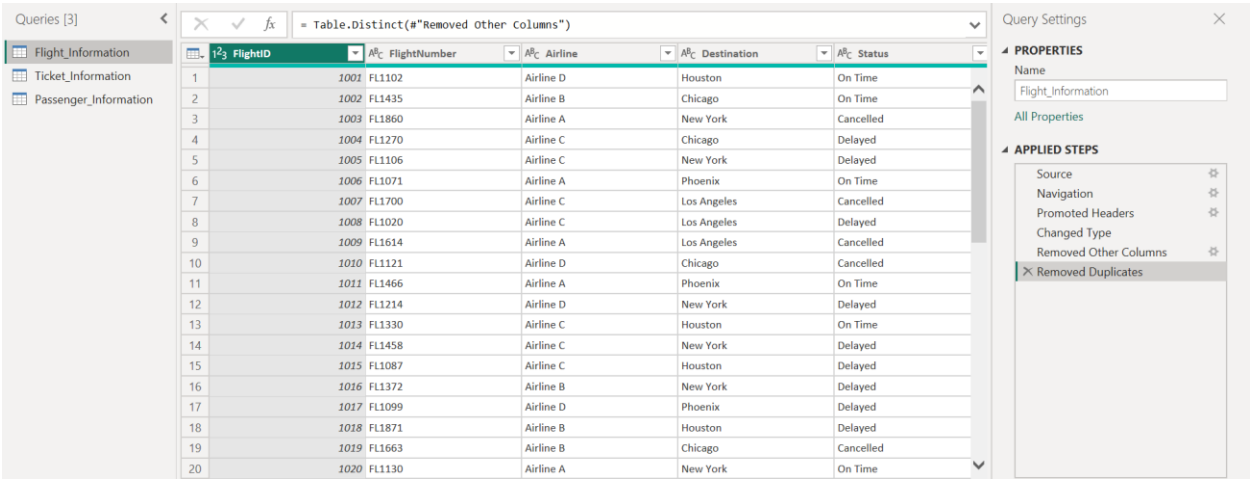
2.3 Format Columns

Select data in power query editor → In the left of every column there is format option.

Format FlightNumber, PassengerID, and TicketID as Text

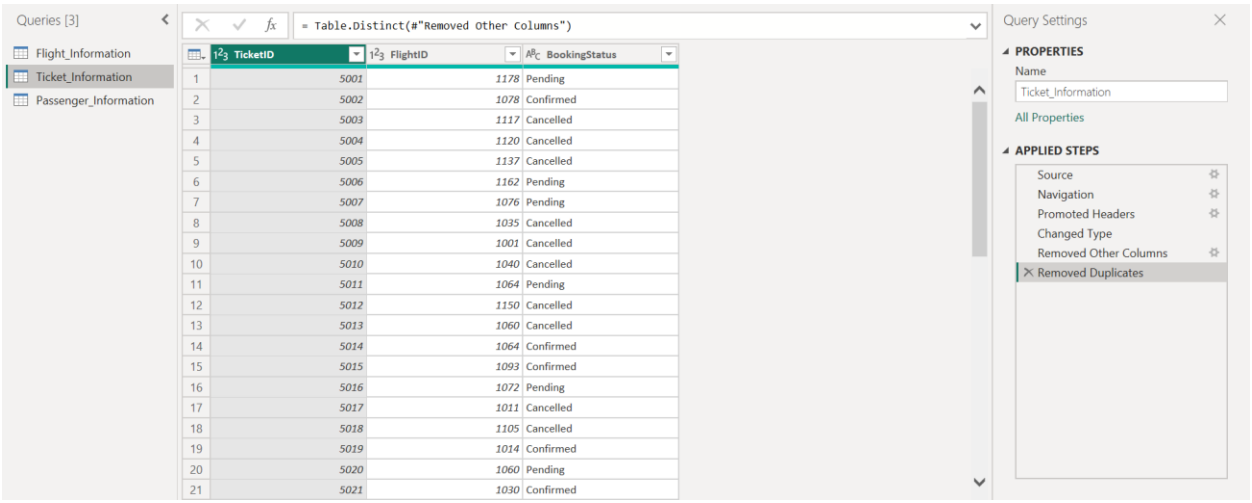
For numeric values, set appropriate data types like Whole Number or Decimal Number.

3.1 cleaned data of **Flight_Informatin:**



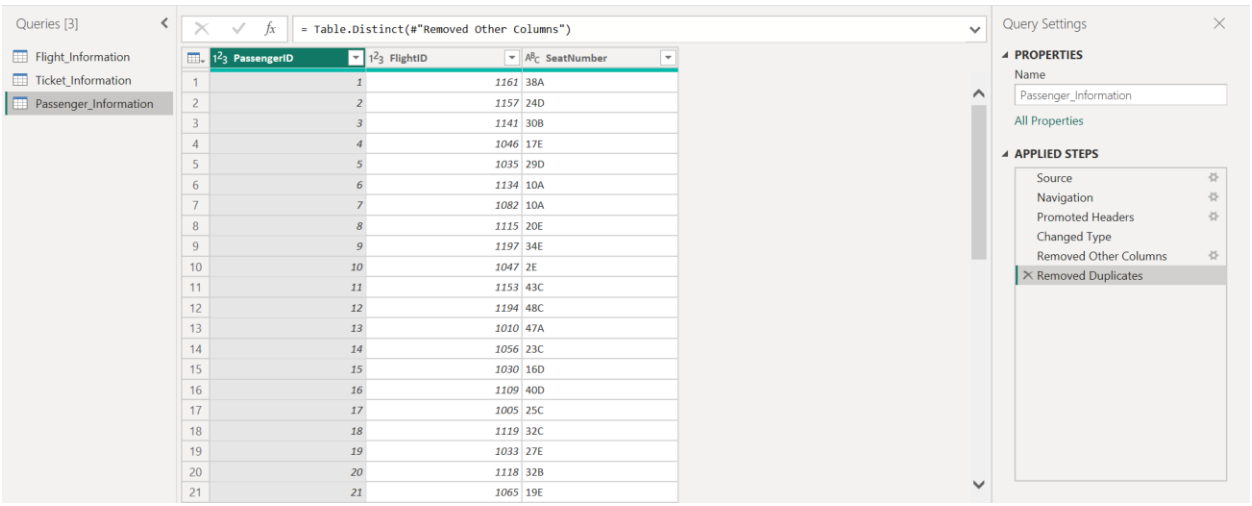
	FlightID	FlightNumber	Airline	Destination	Status
1	1001	FL1102	Airline D	Houston	On Time
2	1002	FL1435	Airline B	Chicago	On Time
3	1003	FL1860	Airline A	New York	Cancelled
4	1004	FL1270	Airline C	Chicago	Delayed
5	1005	FL1106	Airline C	New York	Delayed
6	1006	FL1071	Airline A	Phoenix	On Time
7	1007	FL1700	Airline C	Los Angeles	Cancelled
8	1008	FL1020	Airline C	Los Angeles	Delayed
9	1009	FL1614	Airline A	Los Angeles	Cancelled
10	1010	FL1121	Airline D	Chicago	Cancelled
11	1011	FL1466	Airline A	Phoenix	On Time
12	1012	FL1214	Airline D	New York	Delayed
13	1013	FL1330	Airline C	Houston	On Time
14	1014	FL1458	Airline C	New York	Delayed
15	1015	FL1087	Airline C	Houston	Delayed
16	1016	FL1372	Airline B	New York	Delayed
17	1017	FL1099	Airline D	Phoenix	Delayed
18	1018	FL1871	Airline B	Houston	Delayed
19	1019	FL1663	Airline B	Chicago	Cancelled
20	1020	FL1130	Airline A	New York	On Time

3.2 cleaned data of **Ticket_Information:**



	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	1035	Cancelled
9	5009	1001	Cancelled
10	5010	1040	Cancelled
11	5011	1064	Pending
12	5012	1150	Cancelled
13	5013	1060	Cancelled
14	5014	1064	Confirmed
15	5015	1093	Confirmed
16	5016	1072	Pending
17	5017	1011	Cancelled
18	5018	1105	Cancelled
19	5019	1014	Confirmed
20	5020	1060	Pending
21	5021	1030	Confirmed

3.3 cleaned data of **Passenger_Information:**



Tasks (2)

2. Data Modelling

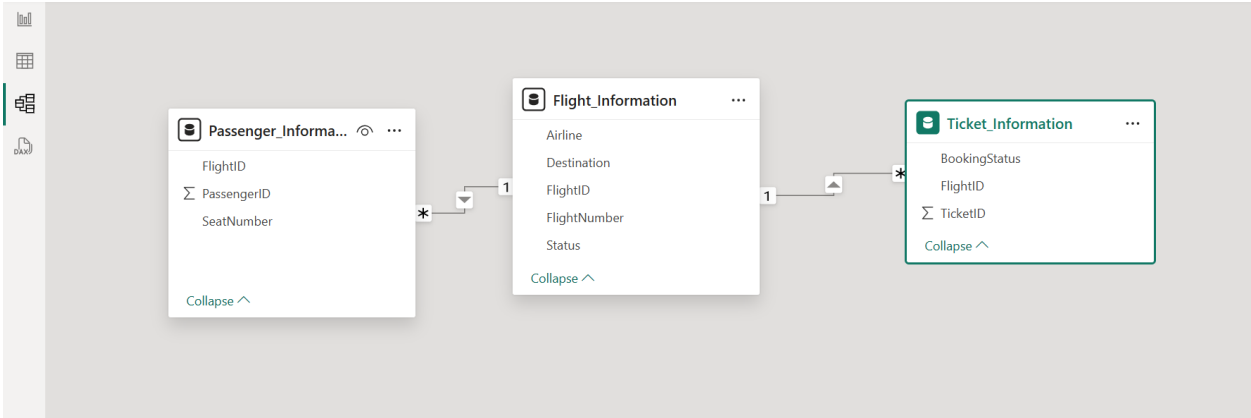
● Create relationships between datasets (FlightID as the key).

● Understand cardinality and configure the model appropriately.

● Deliverables: Screenshot of the data model with relationships.

Explication: (2)

- **a)** Creating Relationships in Power BI
- **b)** Go to the Model View in Power BI.
- **c)** Establish relationships between the datasets:
- **d)** Flight_Information[FlightID] → Passenger_Information[FlightID]
- **e)** Flight_Information[FlightID] → Ticket_Information[FlightID]
- **f)** The cardinality is One-to-Many for these relationships.



Tasks (3)

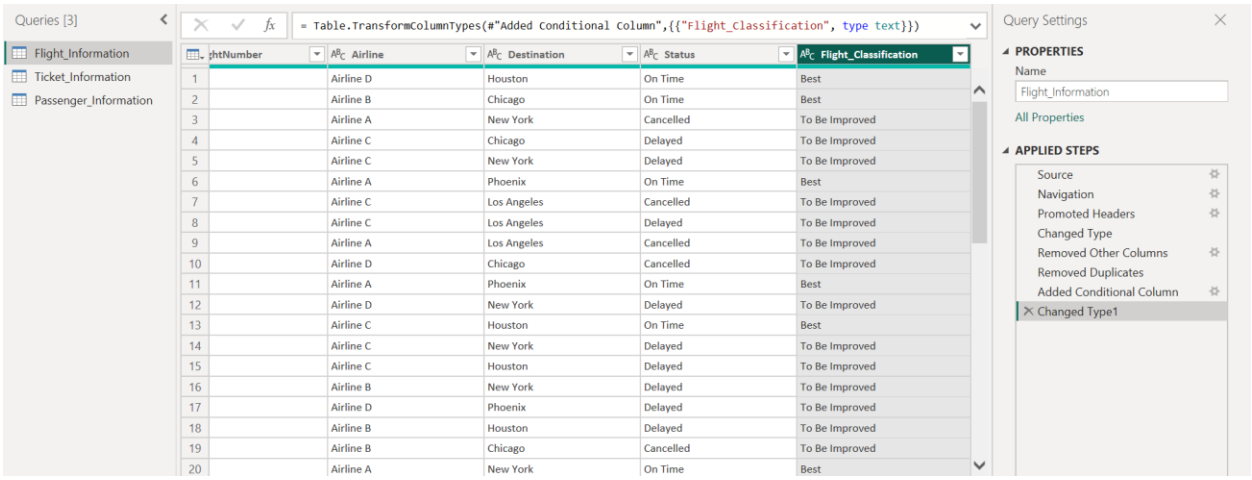
3. Enhanced Data Insights

- 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.

Explication: (3)

a) In Power BI Desktop, go to Transform Data to open Power Query Editor.

b) Add a Conditional Column
In Power Query, go to the Add Column tab.
Select Conditional Column.
In the dialog box:
Column Name: Flight_Classification
Condition:
If Status = "On Time" → Then "Best"
Else "To Be Improved"
Click OK to add the new column.



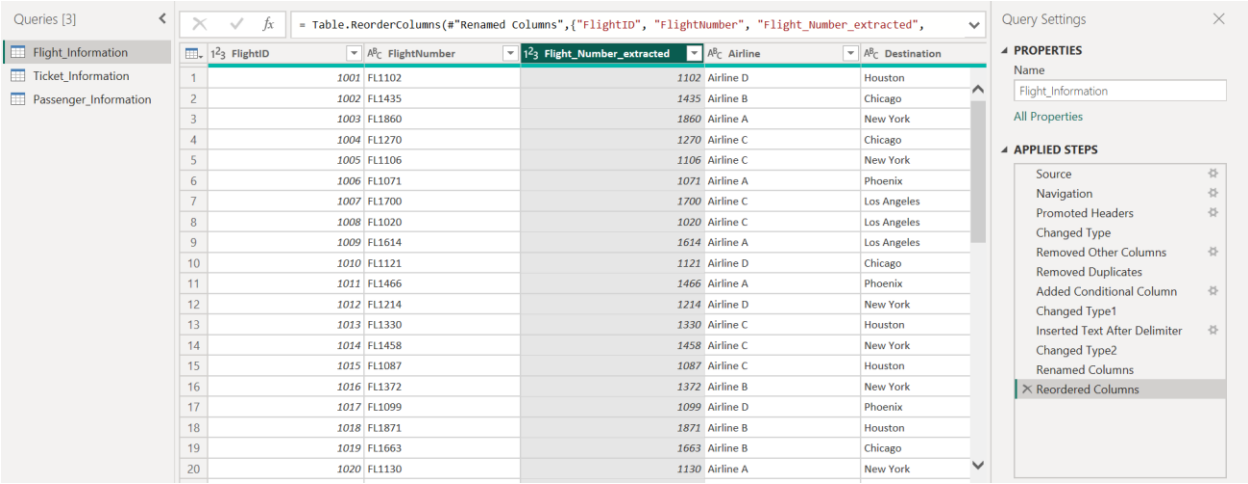
c) **Extracting Flight Numbers Using "Column from Examples"**
To extract the numeric portion from the FlightNumber column:

d) Select the FlightNumber Column.

e) Go to Add Column → Column from Examples → From Selection. Create the Example Column.

f) In the new column, type the flight number manually for the first row (e.g., the FlightNumber is "FL102", type 102). Power BI will automatically detect the pattern and extract flight numbers from the entire column.

g) Finalize the Column
Rename the new column as Flight_Number_Extracted. Set the column's data type to Whole Number for proper calculations.



Tasks (4)

Calculations Using DAX

● Calculate:

- Total passengers for a specific flight.
- Total tickets booked.
- Filtered table showing "Best" flights only.

Explication: (4)

a) Click on home → Enter Data → (Name) Measure.

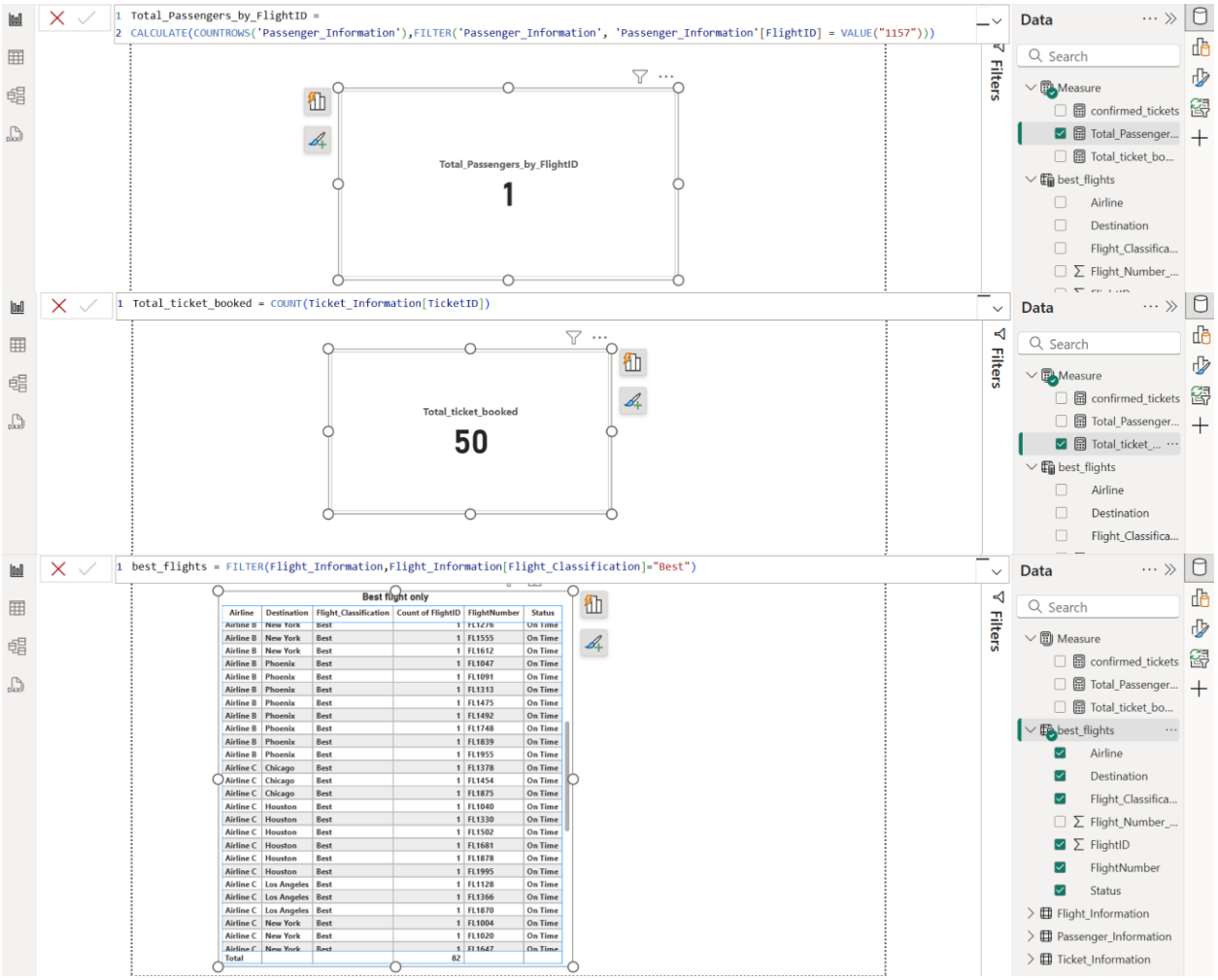
b) Right click on measure → new measure

c) DAX Formula: **Total_Passengers_by_FlightID = CALCULATE(COUNTROWS('Passenger_Information'),FILTER('Passenger_Information', 'Passenger_Information'[FlightID] = VALUE("1157")))**

d) Total tickets booked DAX formula : **Total_ticket_booked = COUNT(Ticket_Information[TicketID])**

e) For “Best” flight create a new table DAX : **best_flights = FILTER(Flight_Information,Flight_Information[Flight_Classification]="Best")**

f) For visual select table from visualization pane then drag the created measurements in data columns.



Tasks (5)

Visualization and Interactive Features

● Create visuals for:

○ Passenger count by airline.

○ Ticket booking statuses.

○ Flights by airline and destination.

● Add interactive features for:

○ Destination and Airline.

○ Quick views.

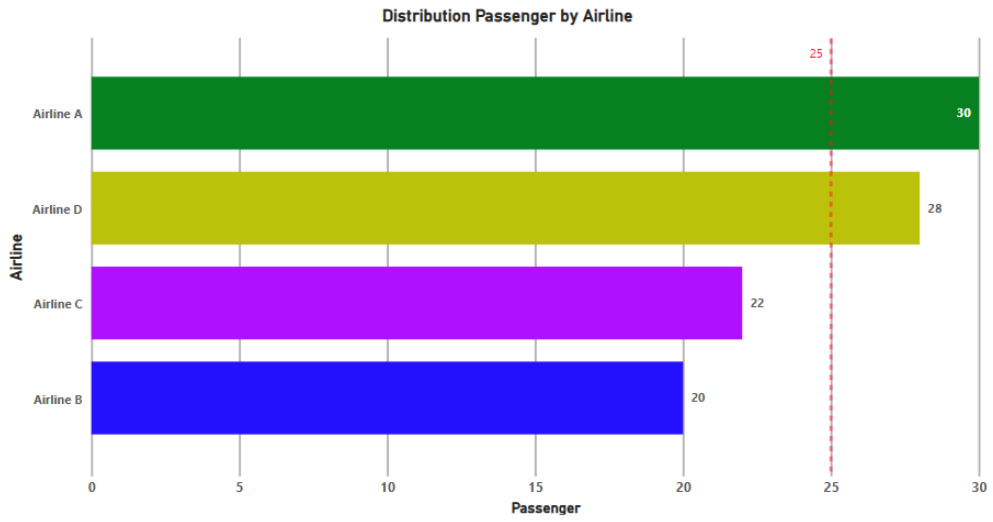
○ Airline-specific pages.

● Deliverables: Screenshots of all visuals and interactive features**.**

Explication: (5)

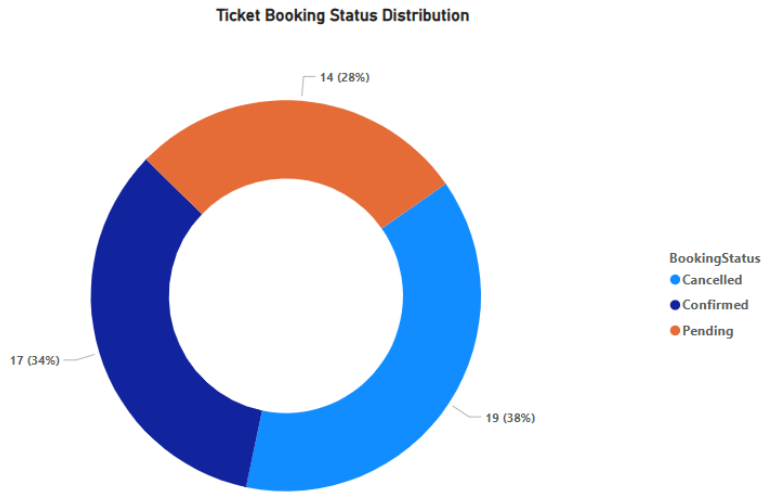
➔ **A) Passenger Count by Airline:**

- Selected Bar Chart from visualization pane.
- Drag airline column from flight_Information to y-axis.
- Drag PassengerID from Passenger_Information to x-axis.



➔ **B) Ticket Booking Statuses:**

- Select Donut chart from the visualization pane.
- Drag BookingStatus column from the Ticket_Information to legend.
- And ticked_id to values section and converted into count both columns.



➔ **C) Flights by Airline and Destination:**

- Select matrix table from visualization pane.
- Drag from Flight_Information table columns airline → rows, Destination→ Columns and Flight_ID columns to value change aggregation to count.

Back to report | FLIGHTS BY AIRLINE AND DESTINATION

Airline	Chicago	Houston	Los Angeles	New York	Phoenix	Total
Airline A	8	14	7	9	10	**48**
Airline B	5	6	9	10	11	**41**
Airline C	5	14	10	13	7	**49**
Airline D	15	9	16	8	14	**62**
Total	**33**	**43**	**42**	**40**	**42**	**200**

➔ **Adding Interactive Features**

a) Destination and Airline Filters (Slicers)

- Add slicers for:
- Flight_Information[Destination]
- Flight_Information[Airline]

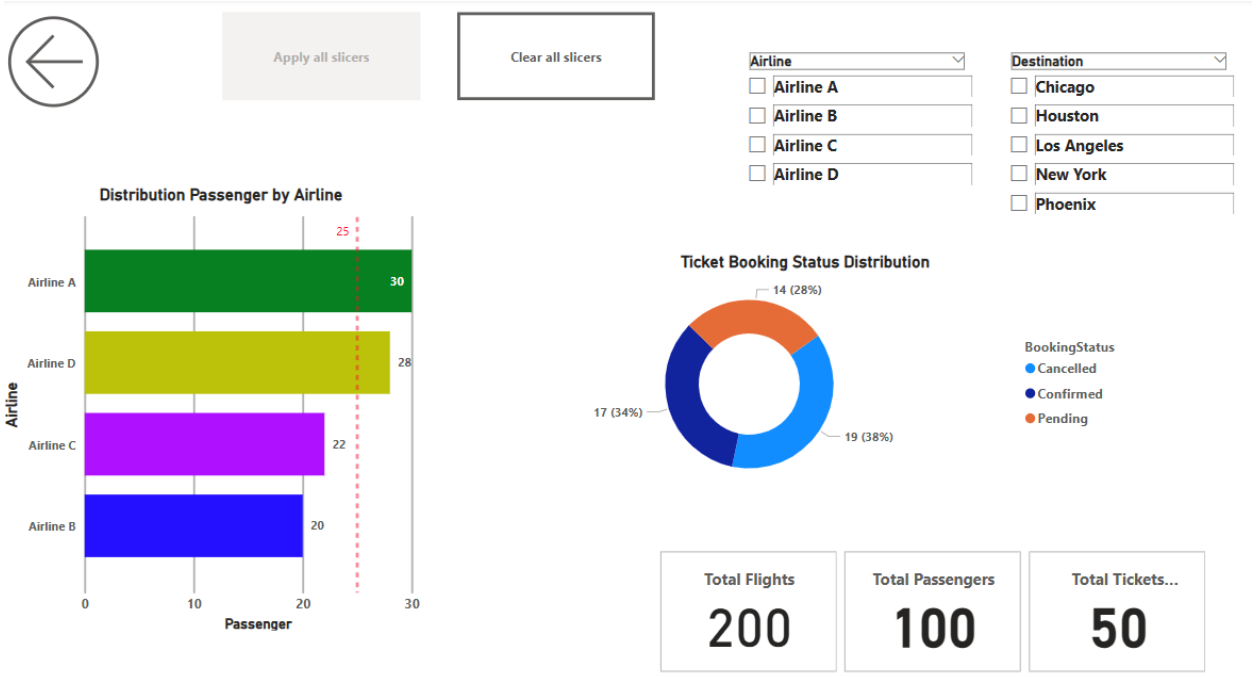
➔ **Quick Views:**

- Create a Card Visual for key metrics such as:
- Total Flights.

- Total Passengers.
- Booking Status.
- Drag flightID column from flight_Iformation to Data (Total Flights) aggregation count.
- Drag PassengerID from Passenger_Information to data (Total Passenger) aggregation count.
- Drag TicketID from Ticket_Information to Data (BookingStatus) aggregation count.

➔ **Airline-Specific Pages:**

- Set the page type to "Drillthrough" in the Format pane.
- Drag Flight_Information[Airline] to the "Drill through from" section.
- Added all Visuals on Airline Details Page (Passenger count, Ticket status, etc.).
- Test Drill-Through:
- Go to the main dashboard.
- Right-click on an airline in a chart → Select Drill through → Airline Details.



Airline	Count of FlightID	Count of PassengerID	Count of TicketID
Airline A	48	30	11
Airline B	41	20	12
Airline C	49	22	13
Airline D	62	28	14
Total	200	100	50

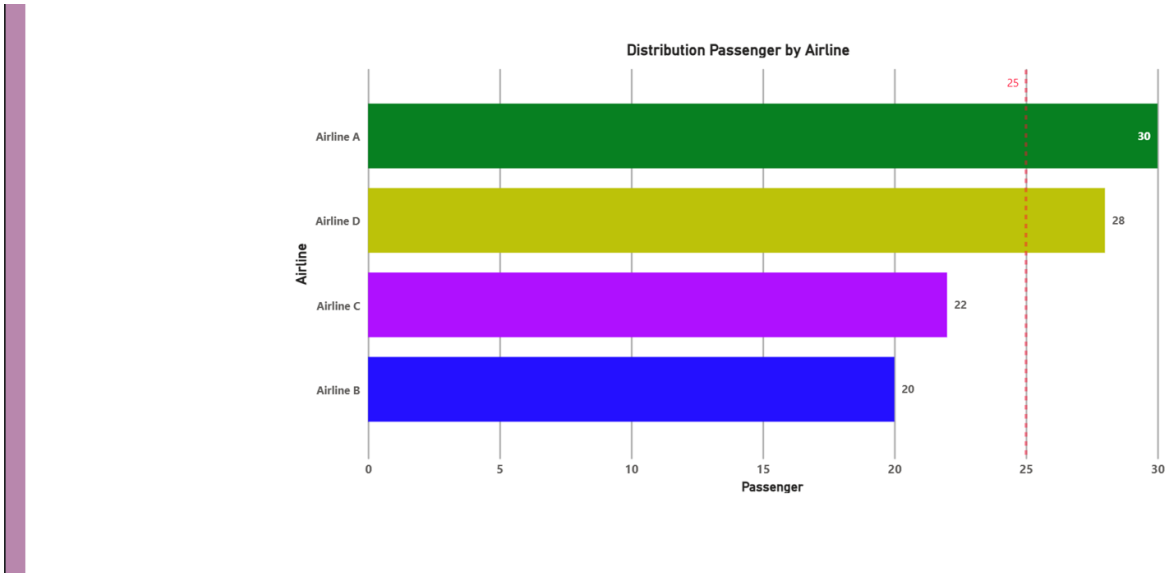
Tasks (6)

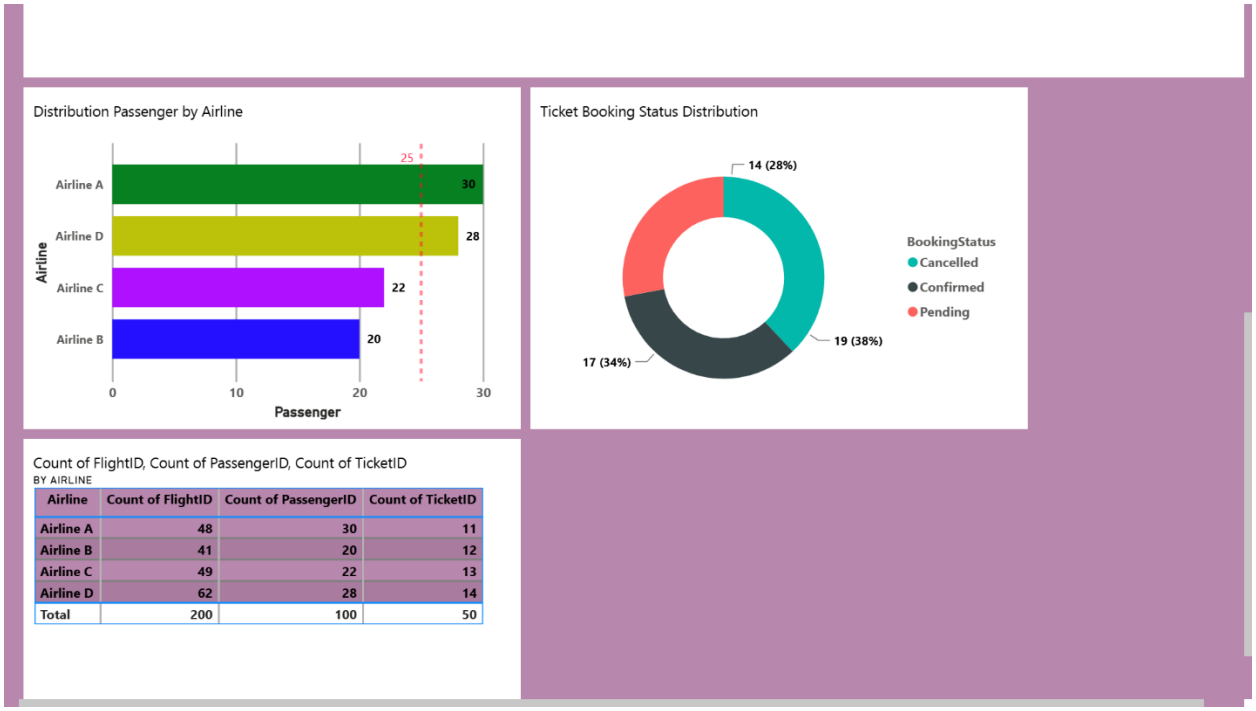
- **Final Dashboard and Power BI Service**
  - **a)** Design a comprehensive dashboard with key visuals and insights.
  - **b)** Configure Row-Level Security (RLS) for Airline A data and assign it to a user.
  - **c)** Set up a schedule refresh at 5 PM daily.

Explication: (6)

a) For Dashboard:

- Open power BI service → clicked on my workspace
- Open the saved file which name was airline.
- In the menu bar click on three dots → select to pin to a dashboard.
- Create a new dashboard, the new dashboard name given “Airline Dashboard”.
- Then pin three pages to the dashboard.
- From edit option we change the Dashboard theme.

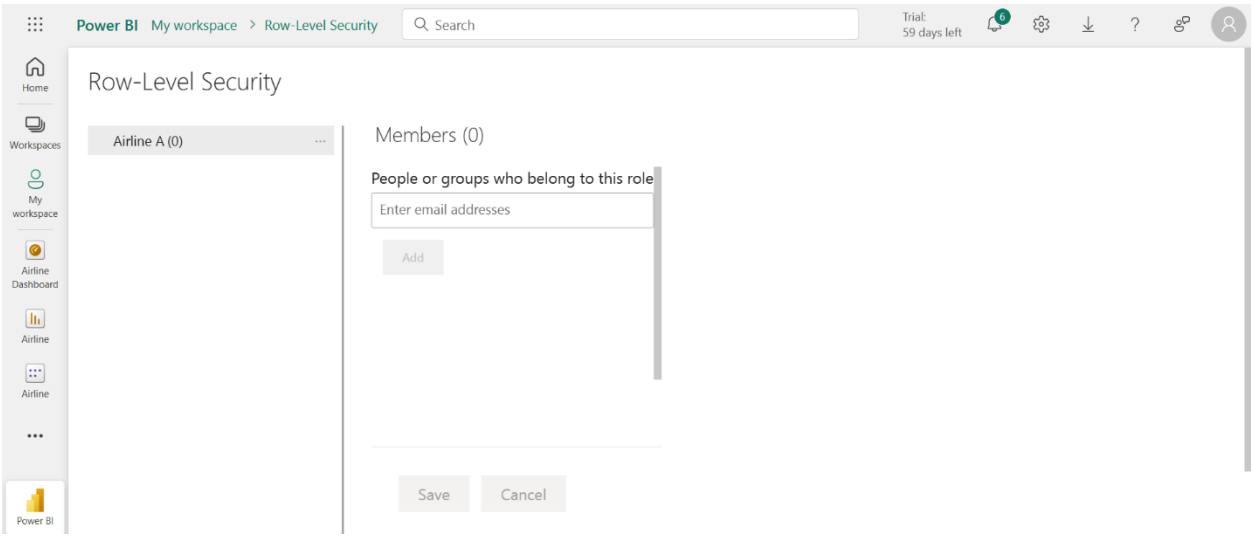




b) For RLS:

- Open Power BI Desktop and go to the Model View.
- Create a new role:
- Click Manage Roles → Create a Role → Name it "Airline_A".
- Apply filter to restrict data:
- Select the Flight_Information table.
- Enter this DAX filter: [Airline] = "Airline A"

- Click View As Roles in Power BI Desktop.
- Select "Airline_A" to confirm restricted access.
- Publish the report to Power BI Service.
- Go to Workspace → Security → Assign Users.
- Add the email of the user who should see only Airline A data.



c) schedule refresh:

- Publish the report to Power BI Service.
- Go to Datasets → Select the dataset.
- Go to the setting → Power BI setting.
- Select → Semantic Model → Refresh.
- Select the time zone → Refresh frequency select daily.
- Set the time zone 5:00 PM → Apply.