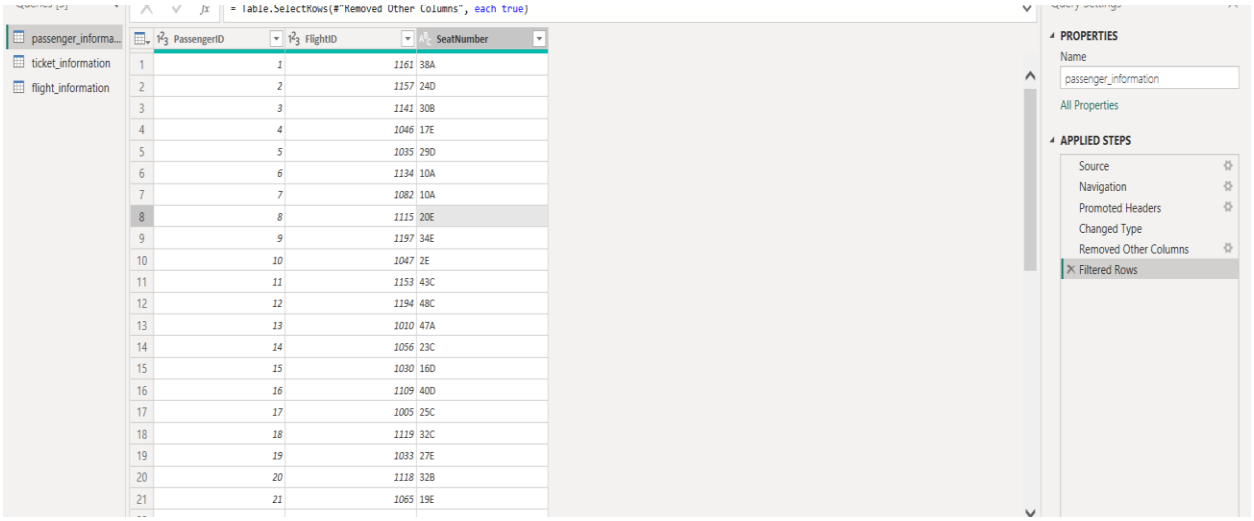
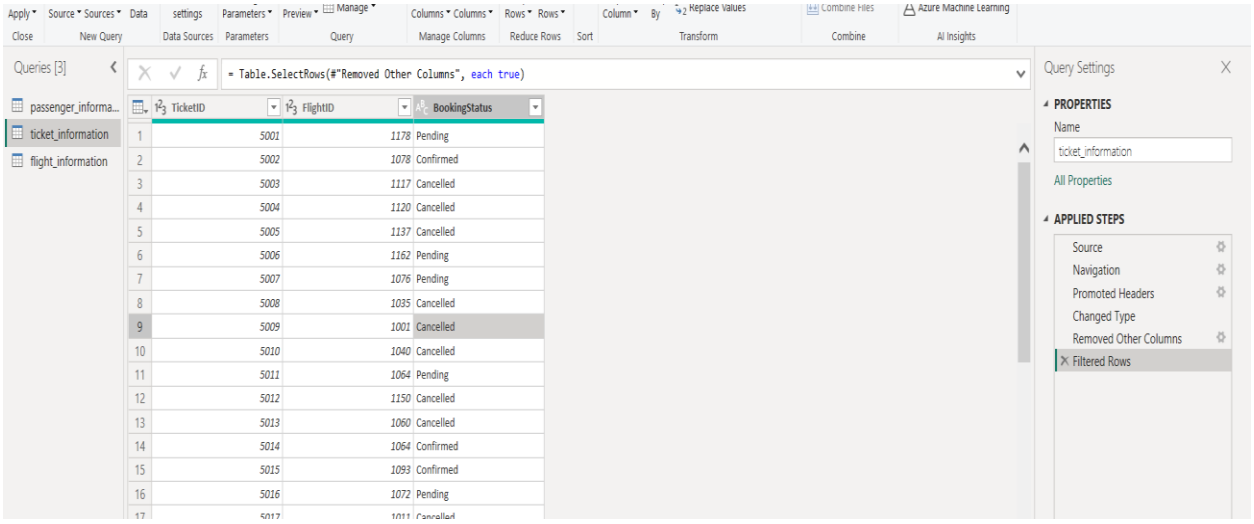
PowerBI Final Project

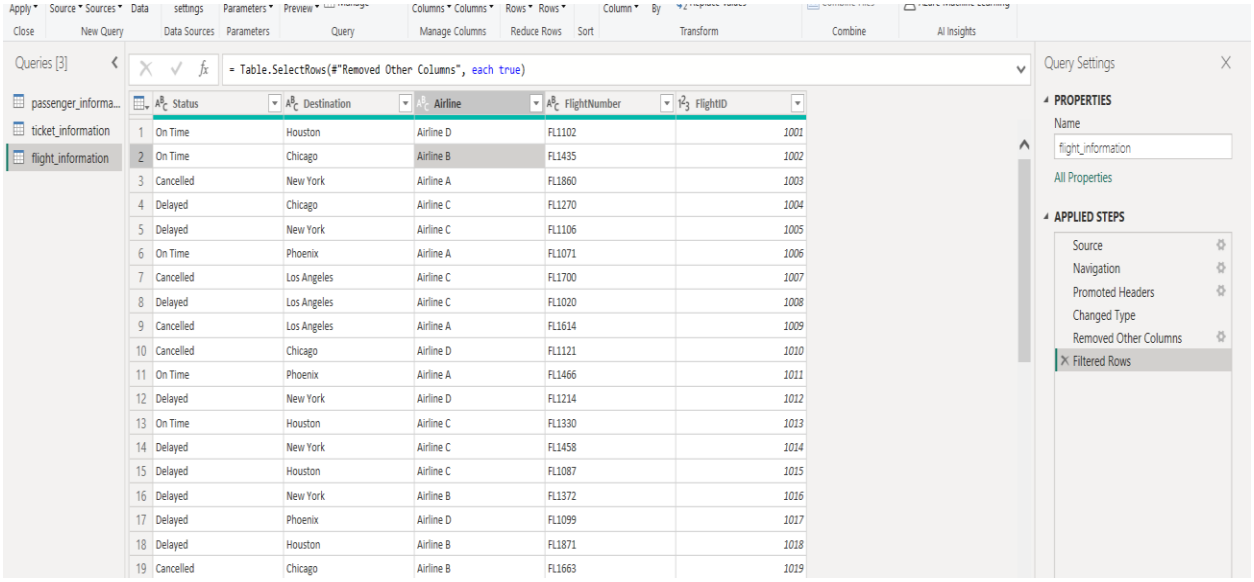
Task-1

1. Data Preparation and Cleaning

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







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

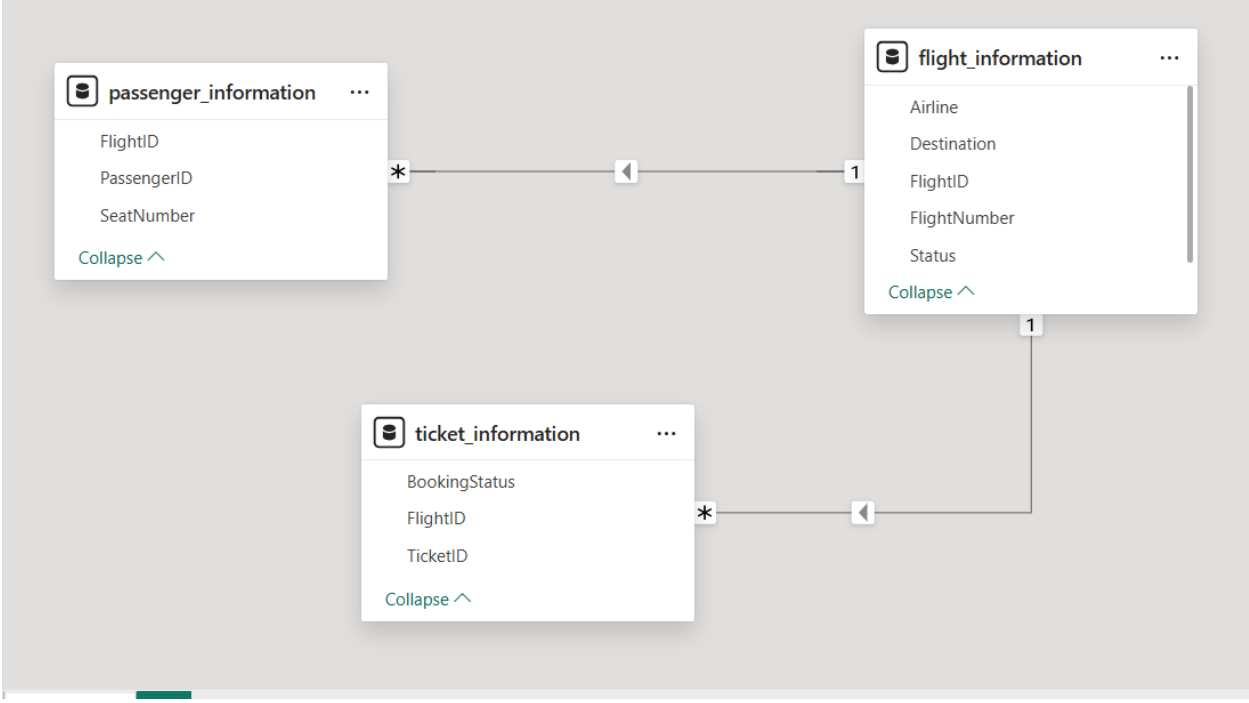
Steps:

- At the initial phase I transformed and loaded the data from “GetData” menu and then removed all the null columns from the datasets.
- Then I checked the data types and format for each column
- And then I clicked on the close&apply

Task-2

2. **Data Modeling**

- Create relationships between datasets (FlightID as the key).
- Understand cardinality and configure the model appropriately.
- Deliverables: Screenshot of the data model with relationships



Cardinality:

1. Flight Information and Ticket Information:

- **Cardinality**: **One-to-Many** (1:*).
    - A **Flight** can have many **Tickets** (one flight, multiple tickets).
    - Each **Ticket** is linked to exactly one **Flight**.

2. Flight Information and Passenger Information:

- **Cardinality**: **One-to-Many** (1:*).
    - A **Flight** can have many **Passengers** (one flight, multiple passengers).
    - Each **Passenger** is linked to exactly one **Flight**.

3. Ticket Information and Passenger Information:

- **Cardinality**: There isn't a direct relationship needed between **Ticket Information** and **Passenger Information**

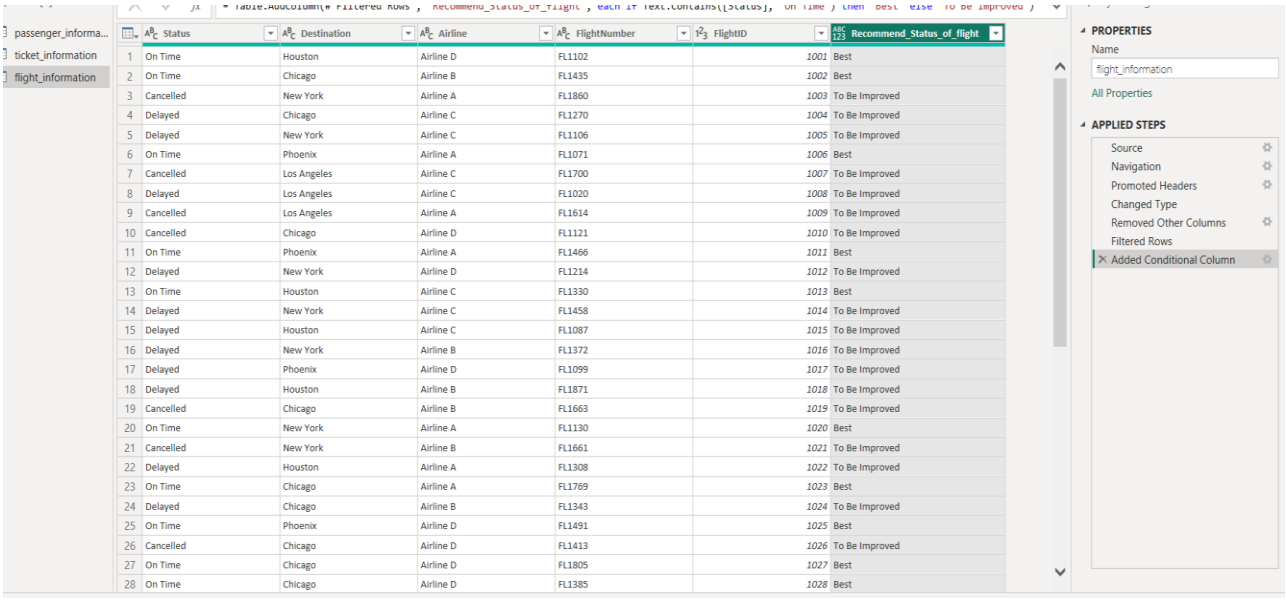
Task-3

3. **Enhanced Data Insights**

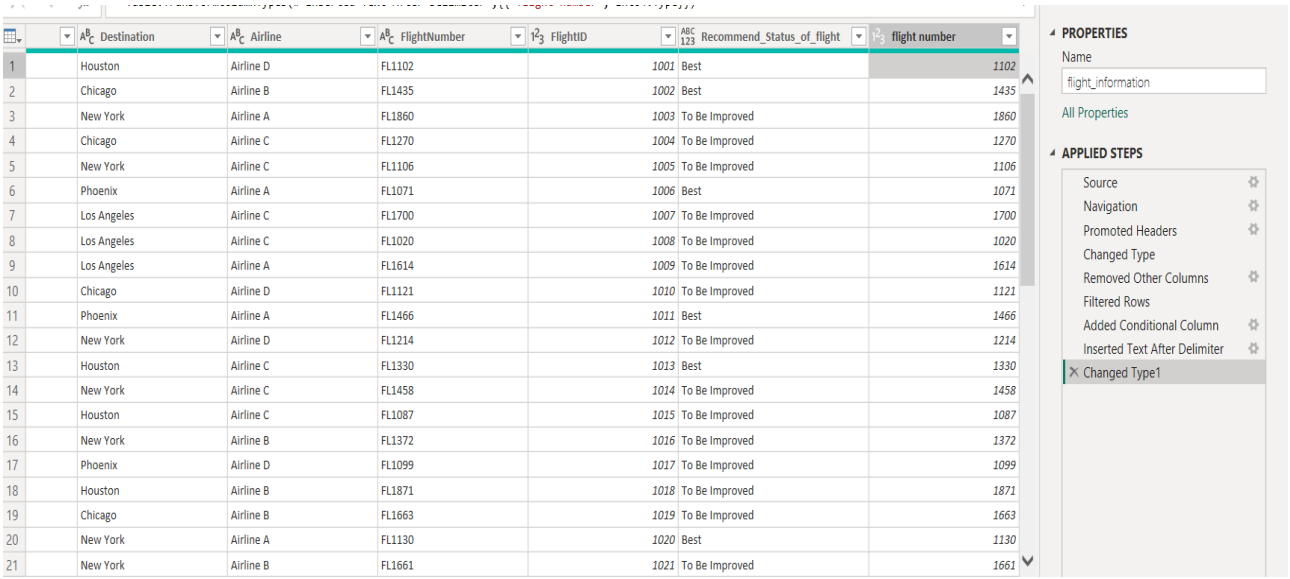
● Add a conditional column to classify flights as "Best" or "To Be Improved" based on status.

● Use"Column from Examples" to extract the flight number from FlightNumber.

● Deliverables: Screenshot of the transformed data.



	Status	Destination	Airline	FlightNumber	FlightID	Recommend_Status_of_flight
1	On Time	Houston	Airline D	FL1102	1001	Best
2	On Time	Chicago	Airline B	FL1435	1002	Best
3	Cancelled	New York	Airline A	FL1860	1003	To Be Improved
4	Delayed	Chicago	Airline C	FL1270	1004	To Be Improved
5	Delayed	New York	Airline C	FL1106	1005	To Be Improved
6	On Time	Phoenix	Airline A	FL1071	1006	Best
7	Cancelled	Los Angeles	Airline C	FL1700	1007	To Be Improved
8	Delayed	Los Angeles	Airline C	FL1020	1008	To Be Improved
9	Cancelled	Los Angeles	Airline A	FL1614	1009	To Be Improved
10	Cancelled	Chicago	Airline D	FL1121	1010	To Be Improved
11	On Time	Phoenix	Airline A	FL1466	1011	Best
12	Delayed	New York	Airline D	FL1214	1012	To Be Improved
13	On Time	Houston	Airline C	FL1330	1013	Best
14	Delayed	New York	Airline C	FL1458	1014	To Be Improved
15	Delayed	Houston	Airline C	FL1087	1015	To Be Improved
16	Delayed	New York	Airline B	FL1372	1016	To Be Improved
17	Delayed	Phoenix	Airline D	FL1099	1017	To Be Improved
18	Delayed	Houston	Airline B	FL1871	1018	To Be Improved
19	Cancelled	Chicago	Airline B	FL1663	1019	To Be Improved
20	On Time	New York	Airline A	FL1130	1020	Best
21	Cancelled	New York	Airline B	FL1661	1021	To Be Improved
22	Delayed	Houston	Airline A	FL1308	1022	To Be Improved
23	On Time	Chicago	Airline A	FL1769	1023	Best
24	Delayed	Chicago	Airline B	FL1343	1024	To Be Improved
25	On Time	Phoenix	Airline D	FL1491	1025	Best
26	Cancelled	Chicago	Airline D	FL1413	1026	To Be Improved
27	On Time	Chicago	Airline D	FL1805	1027	Best
28	On Time	Chicago	Airline D	FL1385	1028	Best



	Destination	Airline	FlightNumber	FlightID	Recommend_Status_of_flight	flight number
1	Houston	Airline D	FL1102	1001	Best	1102
2	Chicago	Airline B	FL1435	1002	Best	1435
3	New York	Airline A	FL1860	1003	To Be Improved	1860
4	Chicago	Airline C	FL1270	1004	To Be Improved	1270
5	New York	Airline C	FL1106	1005	To Be Improved	1106
6	Phoenix	Airline A	FL1071	1006	Best	1071
7	Los Angeles	Airline C	FL1700	1007	To Be Improved	1700
8	Los Angeles	Airline C	FL1020	1008	To Be Improved	1020
9	Los Angeles	Airline A	FL1614	1009	To Be Improved	1614
10	Chicago	Airline D	FL1121	1010	To Be Improved	1121
11	Phoenix	Airline A	FL1466	1011	Best	1466
12	New York	Airline D	FL1214	1012	To Be Improved	1214
13	Houston	Airline C	FL1330	1013	Best	1330
14	New York	Airline C	FL1458	1014	To Be Improved	1458
15	Houston	Airline C	FL1087	1015	To Be Improved	1087
16	New York	Airline B	FL1372	1016	To Be Improved	1372
17	Phoenix	Airline D	FL1099	1017	To Be Improved	1099
18	Houston	Airline B	FL1871	1018	To Be Improved	1871
19	Chicago	Airline B	FL1663	1019	To Be Improved	1663
20	New York	Airline A	FL1130	1020	Best	1130
21	New York	Airline B	FL1661	1021	To Be Improved	1661

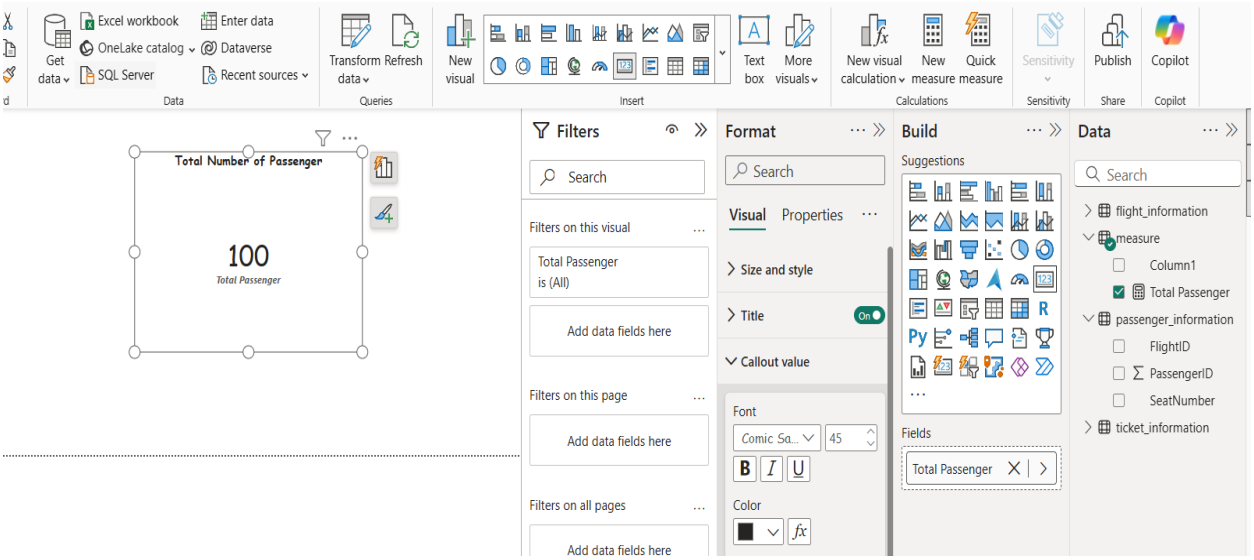
Steps:

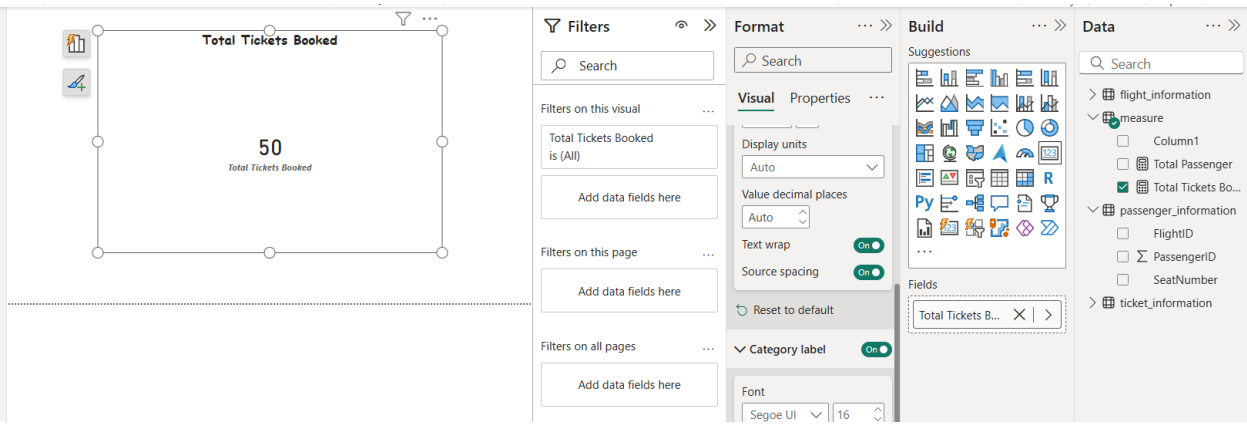
- So, the first I have opened the power query editor then clicked the **Add Column**
- Then used the first **Conditional Column** and added the condition as if the status of the flight is **On Time** then marked as "**Best**" otherwise marked as "**To Be Improved**"
- Then Created one more column as **flight number** and there I copied only the **Number** from then **FlightNumber** column
- The column was created from **Add Coulmn** menu as **Column from examples** and given the column name as **flight number**

Task-4

4. Calculations Using DAX

- Calculate:
  - Total passengers for a specific flight.
  - Total tickets booked.
- Deliverables: Screenshot of DAX calculations and results





Steps:

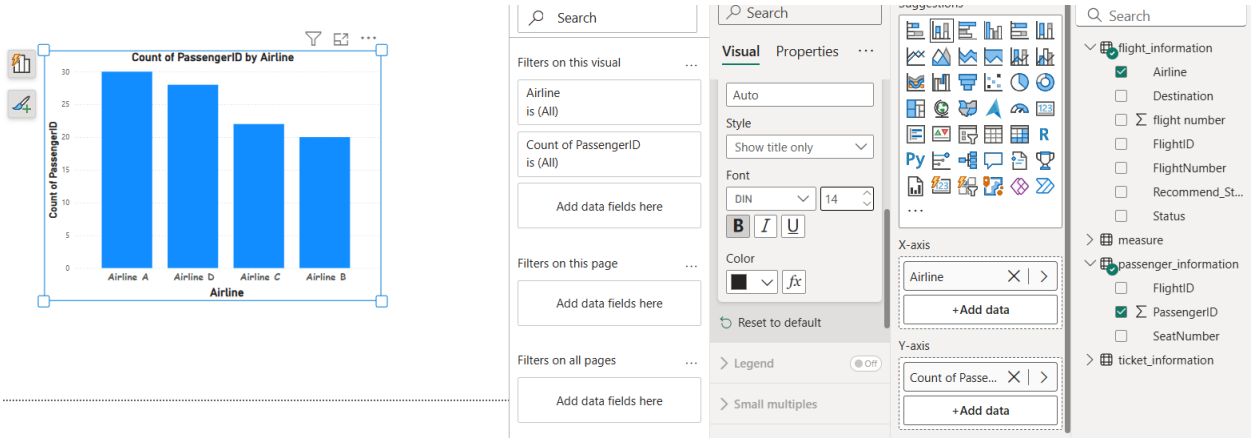
- First I created a measure from **Enter Data** ribbon under **Home** menubar
- Then for the first DAX I used as **Total Passenger = COUNTROWS(passenger_information)** for calculating the **Total Number of Passenger**
- Then I used the second DAX as **Total Tickets Booked = COUNTROWS(ticket_information)** for calculating the **Total Tickets Booked**
- Displayed the output

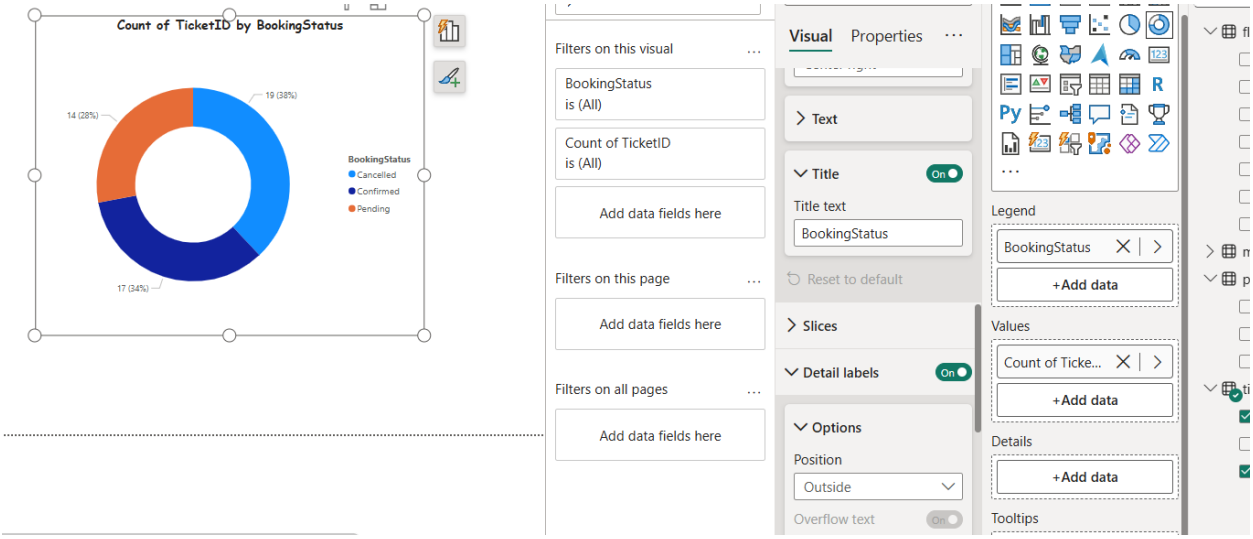
Task-5

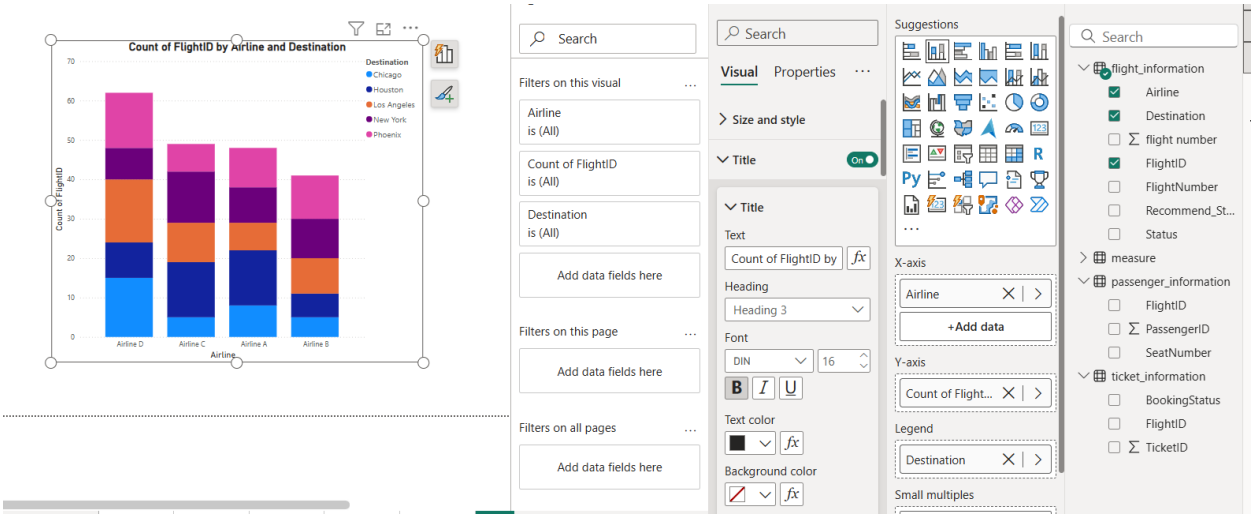
Visualization and Interactive Features

- 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.

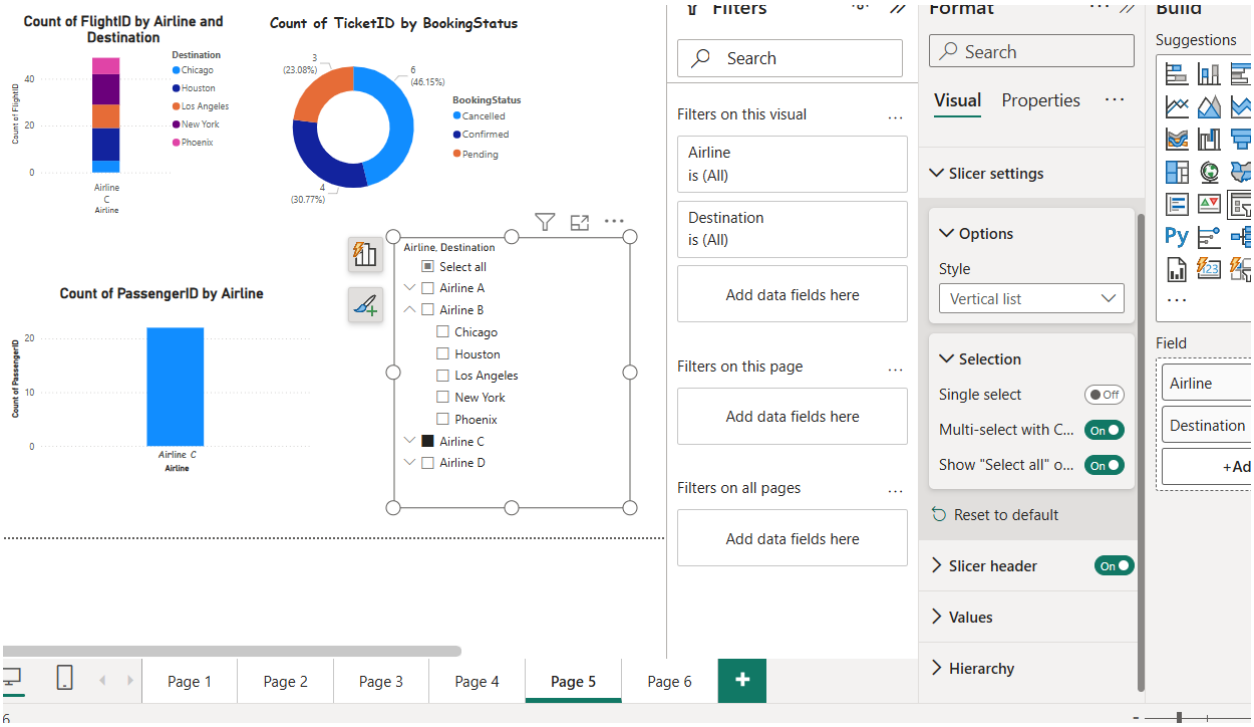
- Deliverables: Screenshots of all visuals and interactive features







Interactive Features :



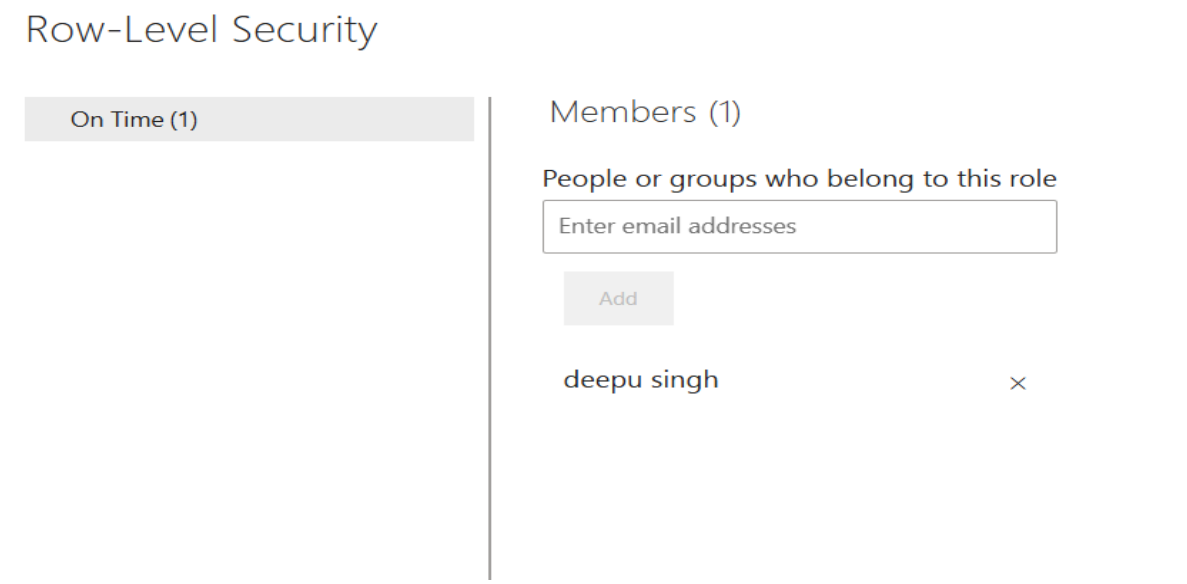
Steps:

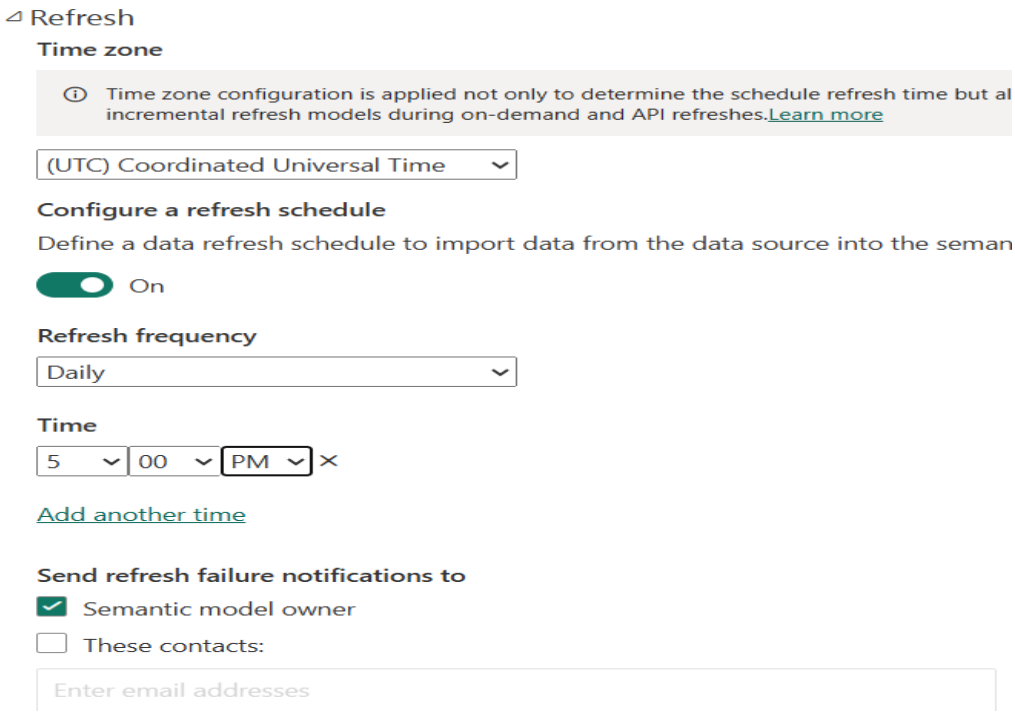
- Initially I created the visuals for all the 3 conditions as per given
- Then I copied all the reports and displayed in one page
- There I used **Slicer** and dragged the **Airline field** and **Destination** in fileds pane
- When I select any slicer then entire data is being filtered and displayed all the ui .

Task-6

Final Dashboard and Power BI Service

- Design a comprehensive dashboard with key visuals and insights.
- Configure Row-Level Security (RLS) for Airline A data and assign it to a user.
- Setup schedule refresh at 5 PM daily.
- Deliverables: Screenshot of the published dashboard and RLS configuration.





Steps:

- So, at first I created the visuals of all the and displayed in a single report
- Then I created a **RLS** from flight_information based on **status** and displayed only one status as **On Time**
- Then applied the **RLS** in the powerBI service as well to display me only
- Then at the last added the **Refresh** based in **Daily** at **5 pm**