

Power BI Project

Airline Data Management and Analysis

Objective: To analyze and visualize airline data for operational insights, passenger management, and ticket booking trends using Power BI.

Dataset used: Flight information, Ticket information, Passenger information.

Task 1. Data Preparation and Cleaning

The main objective of this task is to clean and prepare raw data from the three datasets for analysis by transforming and correcting any data quality issues.

This task involves Remove Blank Row, Remove Duplicate and Remove Errors features of power query which clean the dataset and then prepare the dataset by checking and ensuring appropriate datatype for every column. Now the dataset is ready to apply.

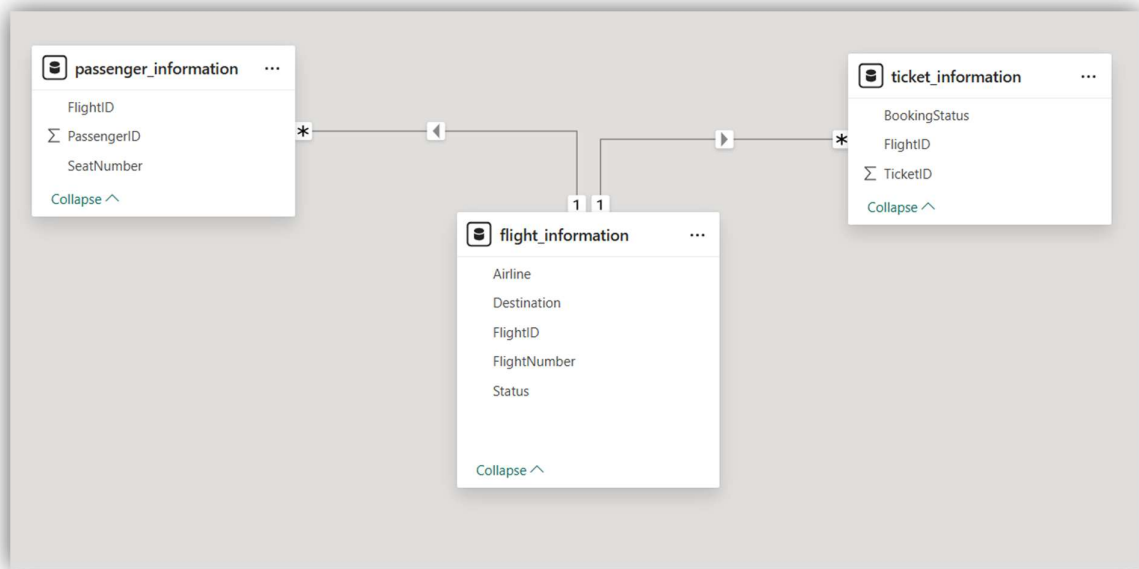
<

Table.RemoveRowsWithErrors("#Removed Blank Rows", {"TicketID"})			
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	1080	Pending
21	5021	1090	Confirmed
22	5022	1095	Confirmed
23	5023	1165	Confirmed
24	5024	1005	Confirmed
25	5025	1083	Cancelled
26	5026	1123	Cancelled

Table.RemoveRowsWithErrors("#Removed Blank Rows", {"PassengerID"})			
PassengerID	FlightID	SeatNumber	
1	1	1181	98A
2	2	1157	24D
3	3	1141	30B
4	4	1046	17E
5	5	1035	29D
6	6	1134	10A
7	7	1082	10A
8	8	1115	20E
9	9	1197	34E
10	10	1047	2E
11	11	1153	43C
12	12	1194	48C
13	13	1010	47A
14	14	1056	23C
15	15	1030	16D
16	16	1109	40D
17	17	1005	25C
18	18	1119	32C
19	19	1033	27E
20	20	1118	32B
21	21	1065	19E
22	22	1146	5B
23	23	1177	28B
24	24	1011	12E
25	25	1085	6A

Task 2. Data Modelling

To create the relationships, I selected the "Model" view in Power BI, where we can see the datasets listed. Then drag the *FlightID* field from **Flight_Information** and drop it onto the corresponding *FlightID* field in **Passenger_Information** and **Ticket_Information**.



To understand the cardinality of dataset I noticed that many passengers are associated with one flight. Therefore, the relationship is many- to- one and also ensures the correct cardinality is applied to the relationship between other tables. Also ensure that the relationships are configured with the proper direction of filtering.

Edit relationship



Select tables and columns that are related.

From table

passenger_information

FlightID	PassengerID	SeatNumber
1161	1	38A
1157	2	24D
1141	3	30B

To table

flight_information

Airline	Destination	FlightID	FlightNumber	Status
Airline D	Houston	1001	FL1102	On Time
Airline B	Chicago	1002	FL1435	On Time
Airline A	Phoenix	1006	FL1071	On Time

Cardinality

Many to one (*:1)

Cross-filter direction

Single

☒ Make this relationship active

☐ Apply security filter in both directions

☐ Assume referential integrity

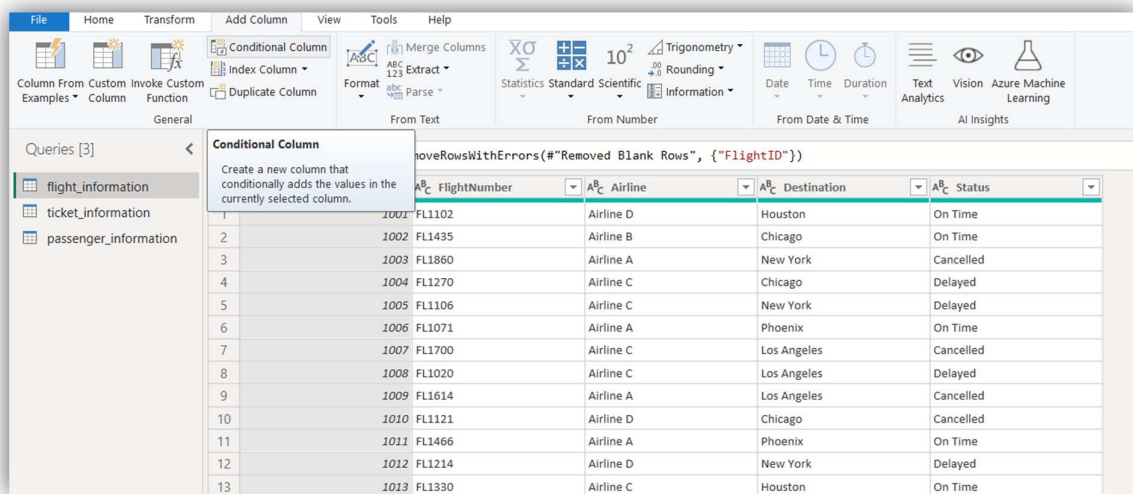
Save

Cancel

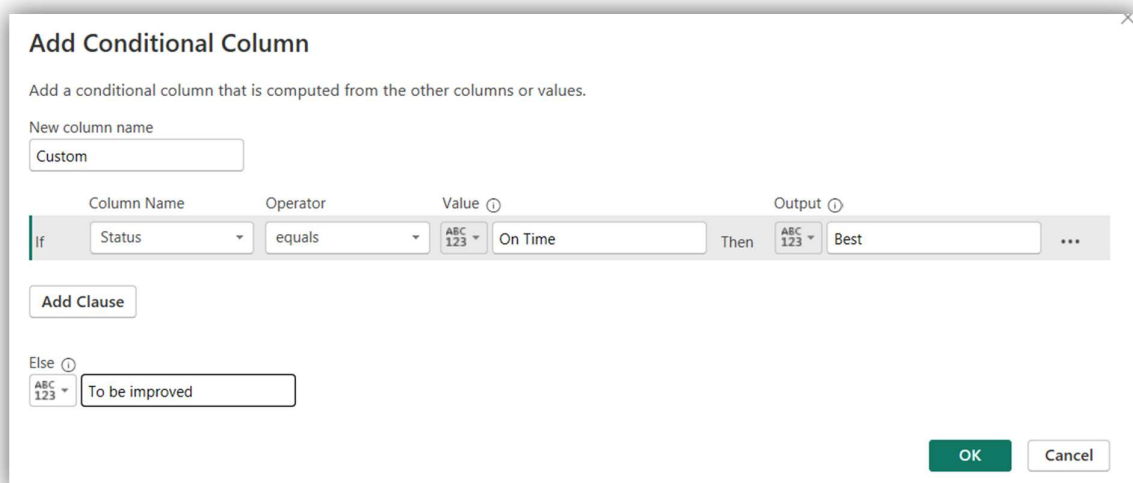
Task 3. Enhanced Data Insights

In this task, the main objective to enhance data insights by adding calculated columns to the dataset which provide deeper insights into the flight data, such as classifying flights based on status and extracting related information from flight number.

To add column in the dataset, I selected “Add Column” tab of Power query editor and selected “conditional column” then set the condition based on flight status. For example, Flights with status “On Time” are classified as “Best” and Flights with status “Cancelled” or “Delayed” are classified as “To be improved” and named column “Classification”.



	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



Add Conditional Column

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

New column name:

	Column Name	Operator	Value	Output
If	Status	equals	On Time	Best

Else

	To be improved
--	----------------

Query Settings

Query Name: flight_information

APPLIED STEPS

- Source
- Navigation
- Promoted Headers
- Changed Type
- Removed Other Columns
- Removed Duplicates
- Removed Blank Rows
- Removed Errors
- Added Conditional Column
- Changed Type1

AB Destination	AB Status	AB Classification
Houston	On Time	Best
Chicago	On Time	Best
New York	Cancelled	To be improved
Chicago	Delayed	To be improved
New York	Delayed	To be improved
Phoenix	On Time	Best
Los Angeles	Cancelled	To be improved
Los Angeles	Delayed	To be improved
Los Angeles	Cancelled	To be improved
Chicago	Cancelled	To be improved
Phoenix	On Time	Best
New York	Delayed	To be improved
Houston	On Time	Best
New York	Delayed	To be improved
Houston	Delayed	To be improved
New York	Delayed	To be improved
Phoenix	Delayed	To be improved
Houston	Delayed	To be improved
Chicago	Cancelled	To be improved
New York	On Time	Best
New York	Cancelled	To be improved
Houston	Delayed	To be improved
Chicago	On Time	Best

Now, to extract Flight Number in Power Query, I use the "Column from Examples" feature to create a new column that extracts the flight number from the *Flight Number* field. This is done by providing an example value in the new column and letting Power BI automatically detect the pattern.

Table.TransformColumnTypes(#"Added Conditional Column",{"Custom", type text})

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 FL1071	Airline D	Houston	Delayed

1

2

Add Column From Examples

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

Transform: `Text.AfterDelimiter([FlightNumber], 'L')`

OK

Cancel

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

1

2

Table.TransformColumnTypes(#"Inserted Text After Delimiter",{{"Text After Delimiter", Int64.Type}})

	FlightNumber	Airline	Destination	Status	Classification	Text After Delimiter	
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	FL1412	Airline D	Chicago	Cancelled	To be improved	1412

Query Settings

PROPERTIES

Name

flight_information

All Properties

APPLIED STEPS

Source

Navigation

Promoted Headers

Changed Type

Removed Other Columns

Removed Duplicates

Removed Blank Rows

Removed Errors

Added Conditional Column

Changed Type1

Inserted Text After Delimiter

Changed Type2

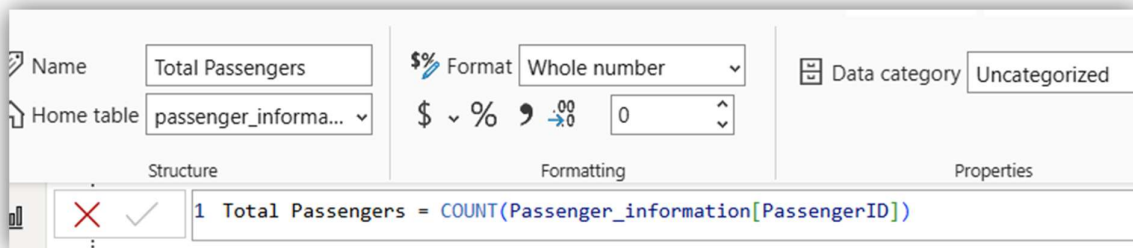
Task 4. Calculations using DAX

Here the objective is to create DAX calculations to generate insights such as total number of passengers on a flight, total number of ticket booked and the filtered table showing only “Best” flights.

To calculate the total passengers for specific flight, I created a measure in *passenger information* table.

Dax Formula :

Total Passengers = COUNT(Passenger_information[PassengerID])

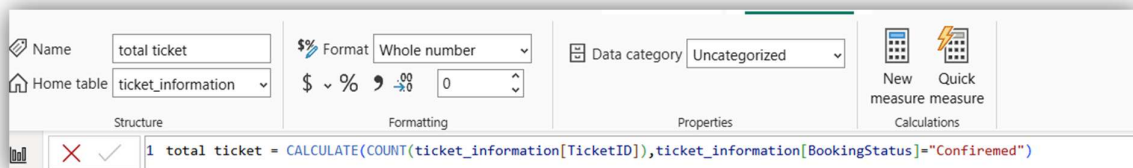


To calculate total ticket booked, I created another measure on *ticket information* table.

DAX Formula :

total ticket =

CALCULATE(COUNT(ticket_information[TicketID]),ticket_information[BookingStatus]="Confiredmed")

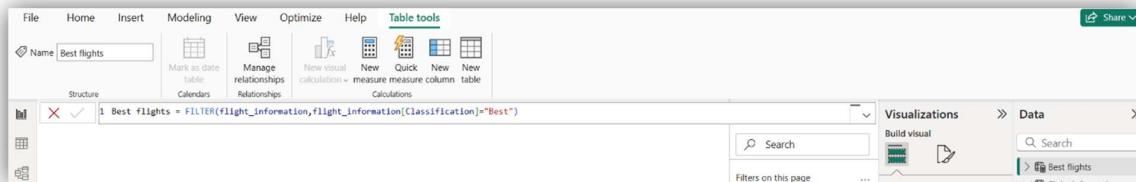
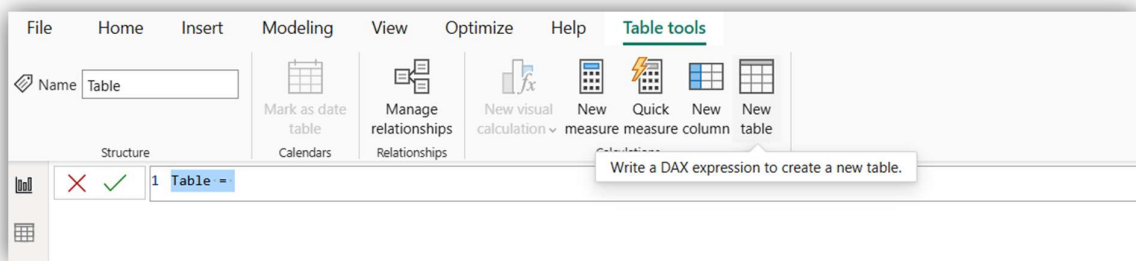


To filter the table showing only “Best” Flights, I created a Filtered table to display only the flights classified as “Best” based on previously created Classification column.

DAX Formula –

Best flights =

`FILTER(flight_information,flight_information[Classification]="Best")`

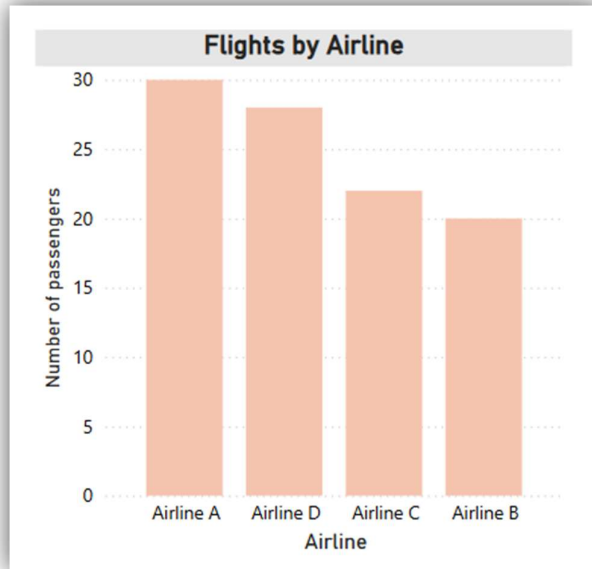


FlightID	FlightNumber	Airline	Destination	Status	Text After Delimiter	Classification
1001	FL1102	Airline D	Houston	On Time	7102	Best
1002	FL1435	Airline B	Chicago	On Time	1435	Best
1006	FL1071	Airline A	Phoenix	On Time	1071	Best
1011	FL1466	Airline A	Phoenix	On Time	1466	Best
1013	FL1230	Airline C	Houston	On Time	7330	Best
1020	FL1130	Airline A	New York	On Time	7130	Best
1023	FL1769	Airline A	Chicago	On Time	1769	Best
1025	FL1491	Airline D	Phoenix	On Time	1491	Best
1027	FL1805	Airline D	Chicago	On Time	7805	Best
1028	FL1305	Airline D	Chicago	On Time	7305	Best
1029	FL1191	Airline D	Los Angeles	On Time	7191	Best
1030	FL1995	Airline B	Phoenix	On Time	7995	Best
1031	FL1276	Airline B	New York	On Time	1276	Best
1033	FL1459	Airline D	New York	On Time	1459	Best
1044	FL1313	Airline B	Phoenix	On Time	7313	Best
1046	FL1252	Airline D	Phoenix	On Time	1252	Best
1049	FL1590	Airline B	Chicago	On Time	1590	Best
1043	FL1601	Airline C	Houston	On Time	1601	Best
1044	FL1475	Airline B	Phoenix	On Time	7475	Best
1046	FL1975	Airline D	Chicago	On Time	7975	Best
1048	FL1189	Airline A	New York	On Time	7189	Best
1050	FL1686	Airline C	Phoenix	On Time	7686	Best
1052	FL1562	Airline D	Phoenix	On Time	7562	Best
1053	FL1875	Airline C	Chicago	On Time	7875	Best
1055	FL1243	Airline B	New York	On Time	7243	Best

Task 5. Visualization and Interactive Features

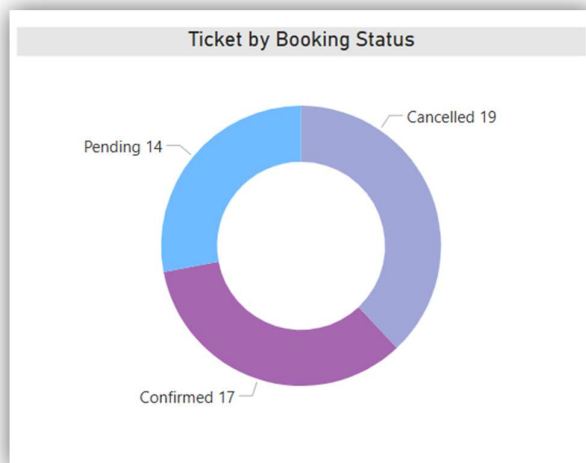
- Create visuals for:
 - Passenger count by airline.

To create a visual for Passengers by Airline, I selected column chart from Visualization Pane.



- Ticket booking statuses.

To showcase the ticket booking status I selected Donut Chart visual.



- o Flights by airline and destination

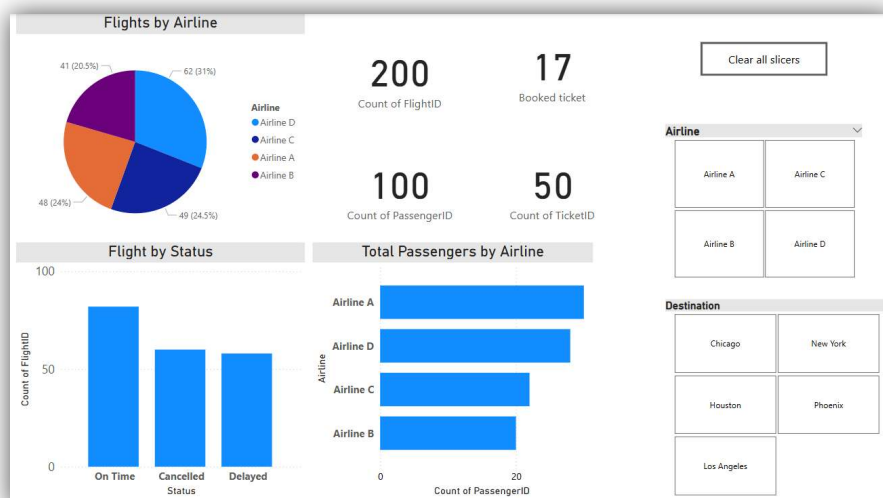
To display flights by airline and destination I selected Clustured bar chart.



- Add interactive features for:

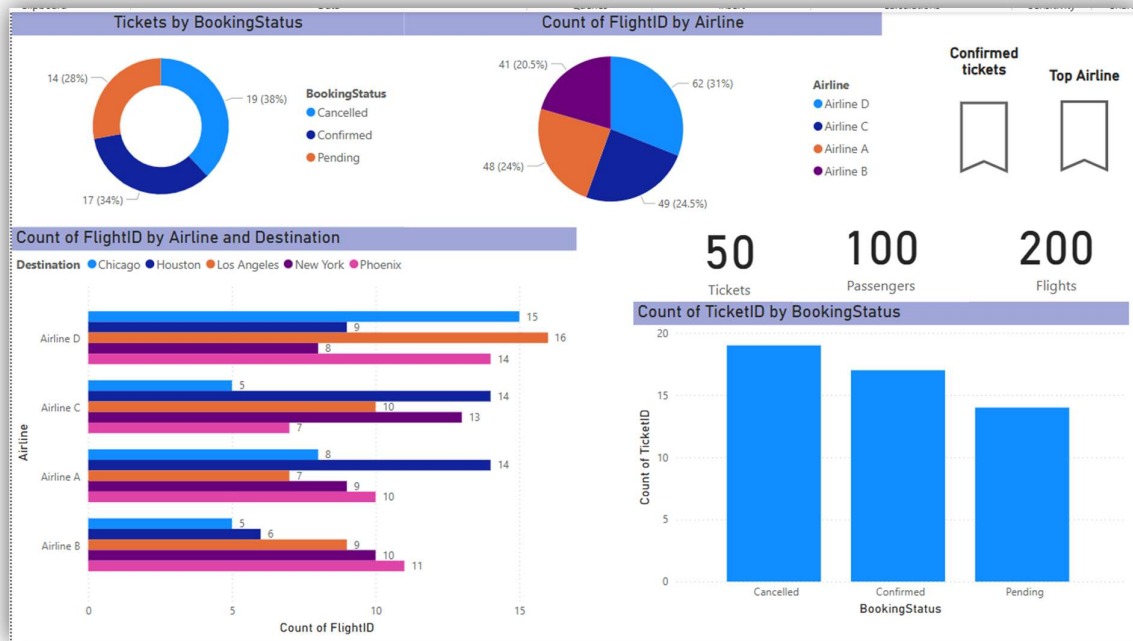
- o Destination and Airline

To add interactive feature for Destination and Airline, I selected Slicers to show the visuals in interactive way.

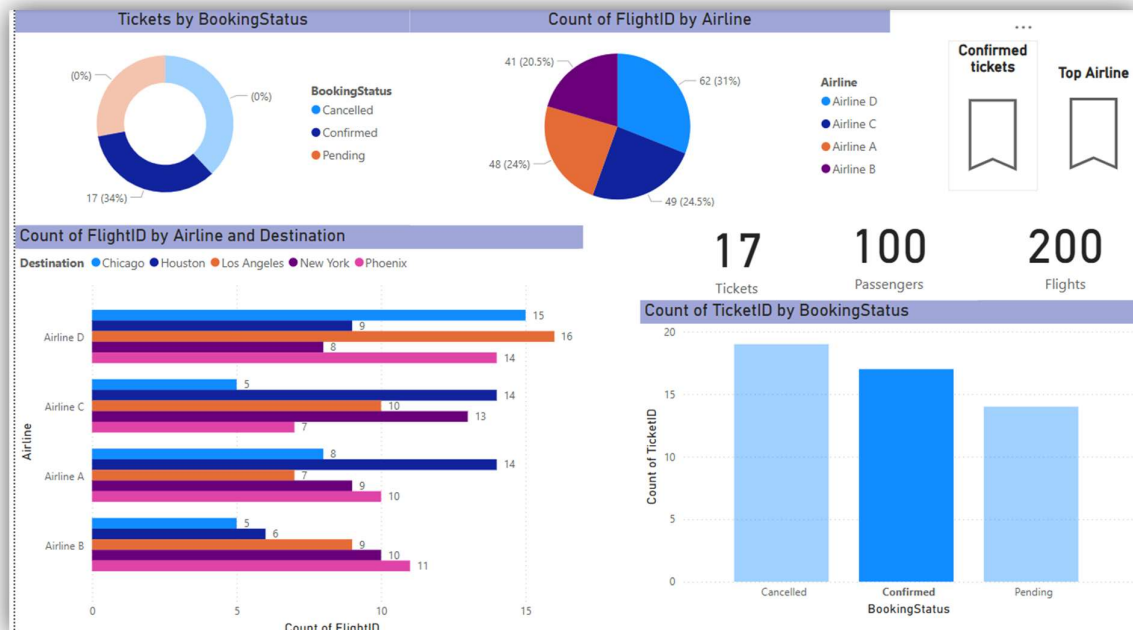


o Quick views

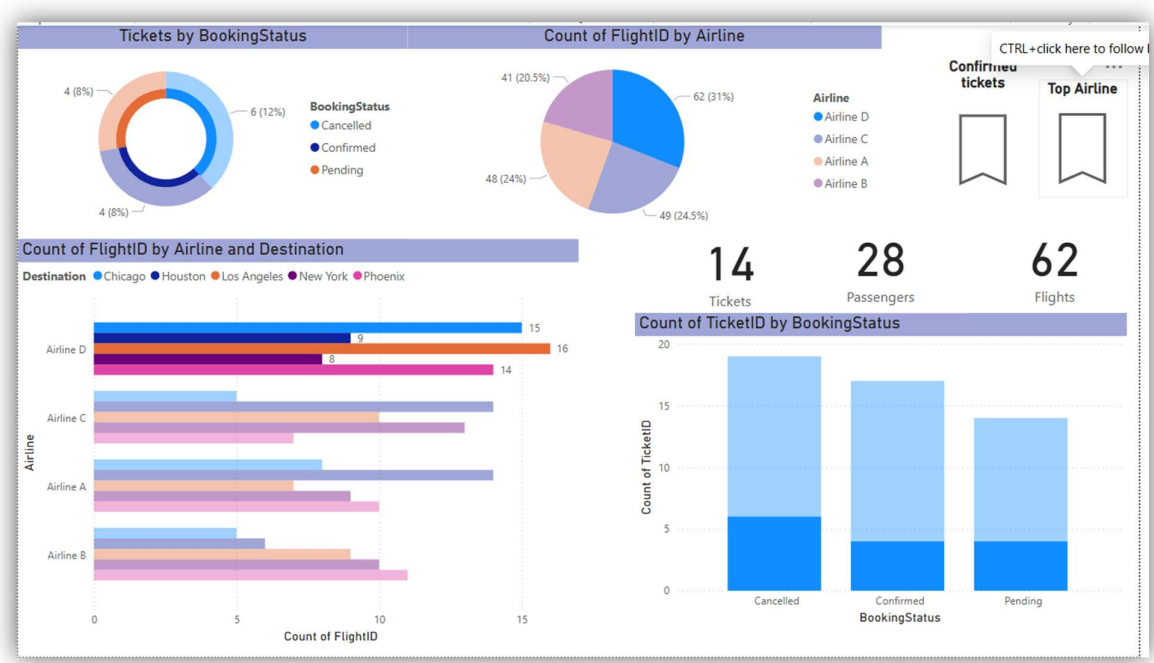
Here I created a bookmark to show quick views.



By Bookmark “Confirmed Ticket”



By bookmark “Top Airline”

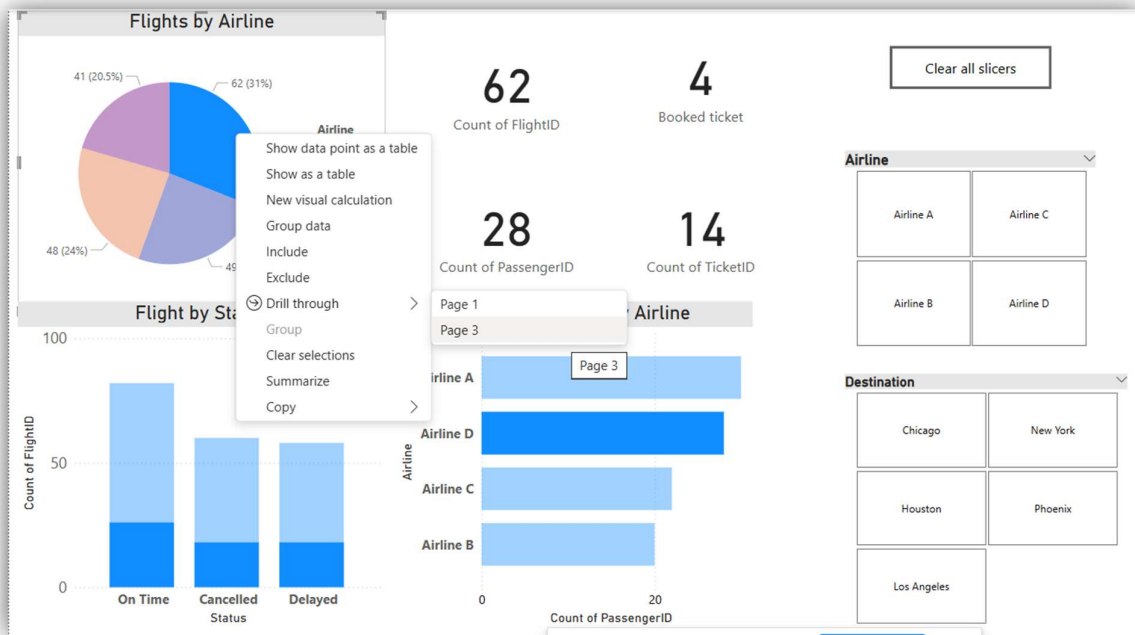


o Airline-specific pages.

To show Airline specific pages, I used Drill Through feature of Power BI.

The table displays flight details for Airline D, including FlightID, Destination, Count of PassengerID, and Count of TicketID. A back arrow icon is visible in the top left corner.

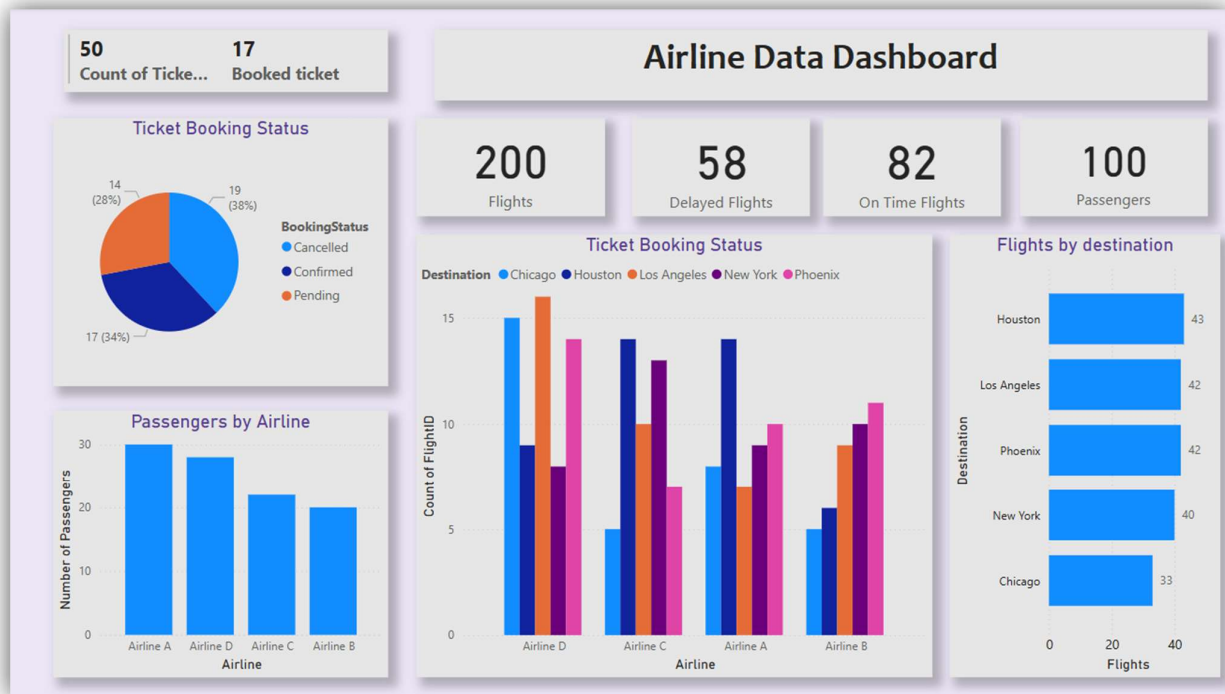
Airline	FlightID	Destination	Count of PassengerID	Count of TicketID
Airline D	1001	Houston	1	4
Airline D	1010	Chicago	1	3
Airline D	1012	New York	1	3
Airline D	1026	Chicago	1	3
Airline D	1027	Chicago	1	3
Airline D	1033	New York	1	3
Airline D	1046	Chicago	1	3
Airline D	1052	Phoenix	1	1
Airline D	1065	Chicago	1	3
Airline D	1066	Los Angeles	1	3
Airline D	1075	Chicago	1	3
Airline D	1083	Houston	1	4
Airline D	1102	Houston	1	4
Airline D	1109	Houston	1	4
Airline D	1123	New York	1	3
Airline D	1129	Los Angeles	1	3
Airline D	1133	Phoenix	2	1
Airline D	1141	Phoenix	1	1
Airline D	1142	New York	1	3
Airline D	1151	New York	1	3
Airline D	1154	Phoenix	1	1
Airline D	1165	Los Angeles	1	3
Airline D	1179	Los Angeles	2	3
Airline D	1190	Phoenix	1	1
Airline D	1192	Chicago	1	3
Airline D	1194	Houston	1	4
Airline C	1004	Chicago	1	4
Airline C	1005	New York	1	4
Total			100	50



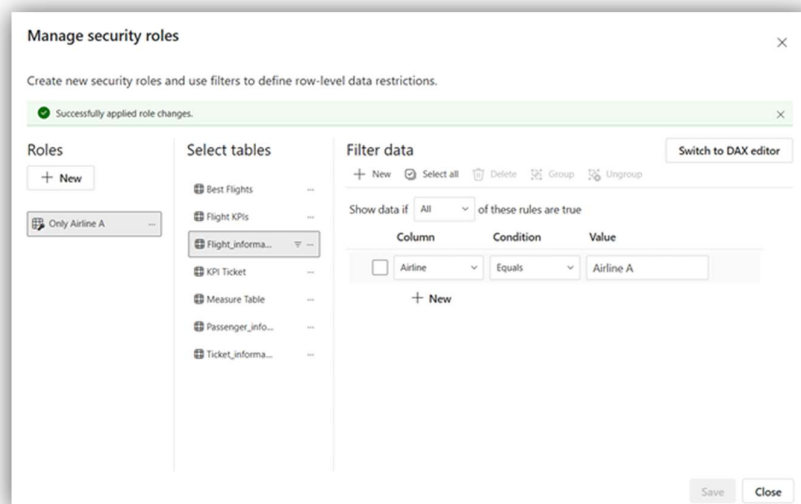
Airline	FlightID	Destination	Count of PassengerID	Count of TicketID
Airline D	1001	Houston	1	4
Airline D	1010	Chicago	1	3
Airline D	1012	New York	1	3
Airline D	1026	Chicago	1	3
Airline D	1027	Chicago	1	3
Airline D	1033	New York	1	3
Airline D	1046	Chicago	1	3
Airline D	1052	Phoenix	1	1
Airline D	1065	Chicago	1	3
Airline D	1066	Los Angeles	1	3
Airline D	1075	Chicago	1	3
Airline D	1083	Houston	1	4
Airline D	1102	Houston	1	4
Airline D	1109	Houston	1	4
Airline D	1123	New York	1	3
Airline D	1129	Los Angeles	1	3
Airline D	1133	Phoenix	2	1
Airline D	1141	Phoenix	1	1
Airline D	1142	New York	1	3
Airline D	1151	New York	1	3
Airline D	1154	Phoenix	1	1
Airline D	1165	Los Angeles	1	3
Airline D	1179	Los Angeles	2	3
Airline D	1190	Phoenix	1	1
Airline D	1192	Chicago	1	3
Airline D	1194	Houston	1	4
Total			28	14

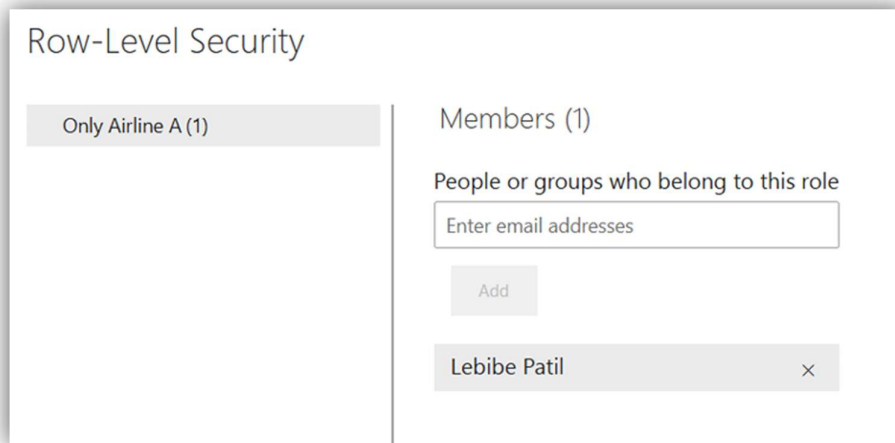
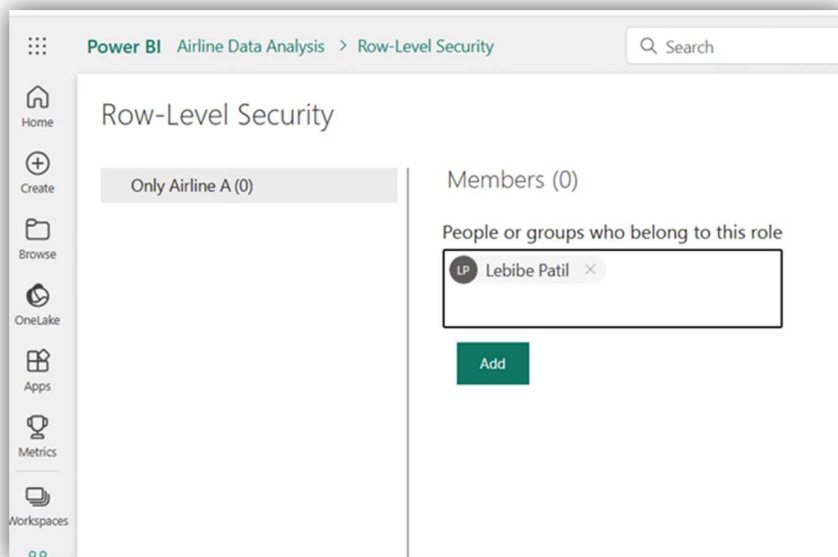
Task 6. Final Dashboard and Power BI Service

I have designed comprehensive dashboard and summarized key insights and interactive visuals.



Then I configure Row-Level Security (RLS) for Airline A , ensuring restricting data access to authorized users only. And from Power BI Service account managed the RLS by assigning to other user email.





Then I have set up a scheduled data refresh at 5 PM daily to ensure up to-date reporting and live updating of datasets.

Refresh

Time zone

ⓘ

Time zone configuration is applied not only to determine the schedule refresh time but also to establish the current date and time for incremental refresh models during on-demand and API refreshes [Learn more](#)

(UTC+05:30) Chennai, Kolkata, Mumbai

Configure a refresh schedule

Define a data refresh schedule to import data from the data source into the semantic model. [Learn more](#)

On

Refresh frequency

Daily

Time

500PM

[Add another time](#)

Send refresh failure notifications to

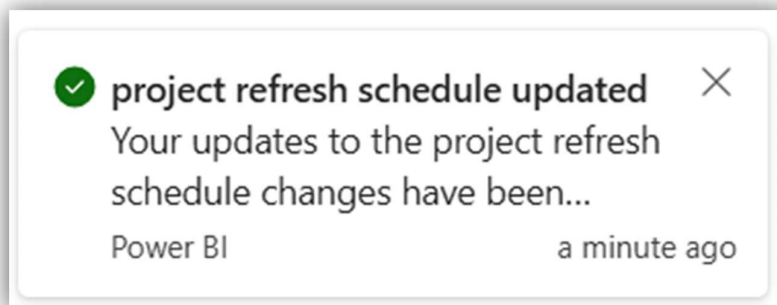
☒ Semantic model owner

☐ These contacts:

Enter email addresses

Apply

Discard



Video explanation

<https://drive.google.com/file/d/1Utqtk8lVhHc9PNPPViE8WObI3gtZ9rP7/view?usp=sharing>