

Power BI



Airline Data Management and Analysis

Abhishek Ambasta | Data Science

Task 1: Data Preparation and Cleaning (10 Marks)

Question:

- 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.

Answer: Followed the following steps:

1. First, we will load the data in the power query editor → open Power BI desktop → Under Home → Transform Data → Opens Power Query editor.
2. Go to home → New Source → Excel Workbook → Load the 3 data set → Ticket Information, Passenger Information and Flight Information.
3. There are many columns which shows null values We will remove them first → Select the columns from each data set that are necessary like:

For Flight information data set:

- A. Flight information: Includes → FlightID, FlightNumber, Airline, Destination, and Status.
Select these by pressing Ctrl + click on each column with holding the ctrl button → Home → Remove columns → Select remove other columns. (This will keep the selected columns and remove other that shows the null values).
- B. Removed the Duplicates that are there in the FlightID Column → select the column → right click → Remove duplicates.

The screenshot shows the Microsoft Power BI Data Editor interface. The top navigation bar includes Home, Transform, Add Column, View, Tools, and Help. The main area displays a query titled "flight_information" which contains 20 rows of flight data. The columns are: FlightID, FlightNumber, Airline, Destination, and Status. The status values include On Time, Delayed, and Cancelled. The right side of the screen features a "Query Settings" pane with sections for Properties (Name: flight_information) and Applied Steps. The "Applied Steps" section lists several steps, with "Removed Duplicates" being the most recent one, indicated by a green highlight.

The above image shows the clean data of Flight Information table.

For Passenger Information table, taken the following steps to clean the data set:

1. Passenger Information: Includes PassengerID, FlightID, and SeatNumber.
2. Select these by pressing Ctrl + click on each column with holding the ctrl button → Home → Remove columns → Select remove other columns. (This will keep the selected columns and remove other that shows the null values).
3. Remove the Duplicates from PassengerID and FlightID → select both and then right click → remove duplicates (We can also do this by clicking on Home → Remove Rows → Remove Duplicates).

The screenshot shows the Microsoft Power BI Data Editor interface. The main area displays a table with three columns: PassengerID, FlightID, and SeatNumber. The formula bar at the top of the editor window contains the expression = Table.Distinct(#"Removed Other Columns", {"FlightID", "PassengerID"}). To the right, the 'Query Settings' pane is open, showing the properties for the 'passenger_information' query, including its name and applied steps. The 'APPLIED STEPS' section lists several actions taken on the source data, including 'Removed Duplicates'.

The above image shows the clean data of Passenger Information table.

For Ticket Information table, taken the following steps to clean the data set:

1. Ticket Information: Includes TicketID, FlightID, and BookingStatus.
2. Select these by pressing Ctrl + click on each column with holding the ctrl button → Home → Remove columns → Select remove other columns. (This will keep the selected columns and remove other that shows the null values).
3. Remove the Duplicates from TicketID and FlightID → select both and then right click → remove duplicates (We can also do this by clicking on Home → Remove Rows → Remove Duplicates).

The screenshot shows the Microsoft Power BI Data Editor interface. The top ribbon has tabs like Home, Transform, Add Column, View, Tools, and Help. Below the ribbon is a toolbar with icons for Close & Apply, New Source, Refresh, Properties, Manage Columns, and Sort. The main area shows a query named 'ticket_information' with three columns: TicketID, FlightID, and BookingStatus. The data table contains 21 rows of flight booking information. To the right, the 'Query Settings' pane shows the query name and properties. The 'APPLIED STEPS' pane lists the steps taken: Source, Navigation, Promoted Headers, Changed Type, Removed Other Columns, and Removed Duplicates.

The above image shows the clean data of Ticket Information table.

Then Load all the data by clicking on → close and apply.

TASK 2: DATA MODELING (10 MARKS)

- CREATE RELATIONSHIPS BETWEEN DATASETS (FLIGHT ID AS THE KEY).
- UNDERSTAND CARDINALITY AND CONFIGURE THE MODEL APPROPRIATELY.

Answer: Followed the following steps:

1. On the left-hand side go to the Mode View.

- Keep Flight_information at the center (fact/dimension hybrid), acting as the bridge for both:
- (a) Passenger_information ($1 \rightarrow \text{many}$): By dragging the FlightID column from Flight_Information table to the Passenger Information table (FlightID column), this will create one to many relationships.

New relationship

Select tables and columns that are related.

From 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

To table: passenger_information

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

Cardinality: One to many (1:*)

Cross-filter direction: Single

Make this relationship active

Apply security filter in both directions

Assume referential integrity

Save Cancel

- (b) Table_information ($1 \rightarrow \text{many}$): By dragging the FlightID column from Flight_Information table to the Table_Information table (FlightID column), this will create one to many relationships.

New relationship

Select tables and columns that are related.

From 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

To table: ticket_information

BookingStatus	FlightID	TicketID
Pending	1178	5001
Confirmed	1078	5002
Cancelled	1117	5003

Cardinality: One to many (1:*)

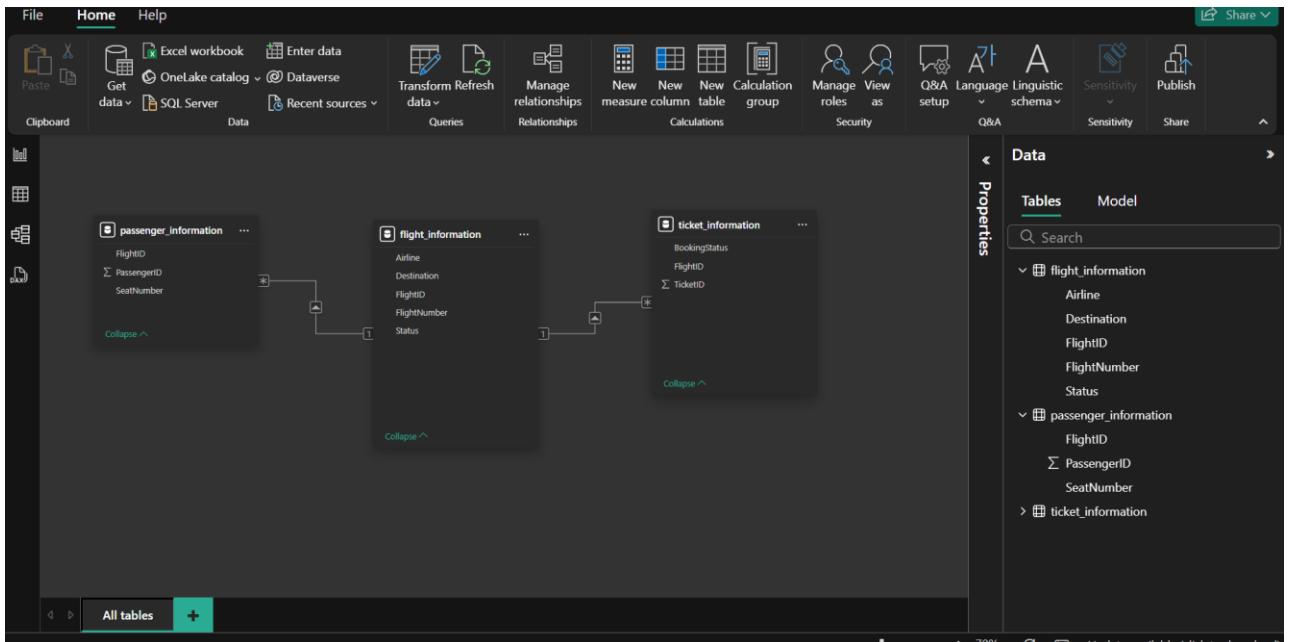
Cross-filter direction: Single

Make this relationship active

Apply security filter in both directions

Assume referential integrity

Save Cancel



The above image shows that:

1. **One-to-many** relationships are set with correct cardinality and direction.
2. Arrows confirm **single-directional filtering**, which is ideal for performance and logic.

TASK 3: ENHANCED DATA INSIGHTS (10 MARKS)

Question 1: Add a conditional column to classify flights as "Best" or "To Be Improved" based on status.

Answer: Followed the following steps:

1. Open power query editor by clicking on home → Transform data.

- From Flight information table → Click on the status column → Add column → Conditional Column → Name it as Flight Category → If column name → Status Equals to On Time then make Output as Best, Else To be Improved.

Add Conditional Column

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

New column name: FlightCategory

	Column Name	Operator	Value	Output
1	Status	equals	On Time	Then Best
Else			To Be Improved	

OK Cancel

After this change the column type as text.

Home Transform Add Column View Tools Help

Queries [3]

flight_information

passenger_informa...

ticket_information

Add Column

Conditional Column

Format

Parse

General

From Number

From Date & Time

Text Analytics

Query Settings

Name: flight_information

Properties

Applied Steps

Source

Navigation

Promoted Headers

Changed Type

Removed Other Columns

Removed Duplicates

Added Conditional Column

Changed Type1

5 COLUMNS, 200 ROWS Column profiling based on top 1000 rows

PREVIEW DOWNLOADED AT 7:55 PM

This shows On Time flights is the “Best” and rest are in “To be Improved” Category.

Question 2: Use "Column from Examples" to extract the flight number from FlightNumber.

Answer: Followed the following steps:

1. Select the column FlightNumber → click on Add Column → Column from Example → From Selection.
2. Type only the flight Number as 1102, and this will extract all the other rows flight numbers → Rename the column as FlightNumber_Only.

	FlightID	FlightNumber	FlightNumber_Only	Airline	Destination
1	1001	FL1102	1102	Airline D	Houston
2	1002	FL1435	1435	Airline B	Chicago
3	1003	FL1860	1860	Airline A	New York
4	1004	FL1270	1270	Airline C	Chicago
5	1005	FL1106	1106	Airline C	New York
6	1006	FL1071	1071	Airline A	Phoenix
7	1007	FL1700	1700	Airline C	Los Angeles
8	1008	FL1020	1020	Airline C	Los Angeles
9	1009	FL1614	1614	Airline A	Los Angeles
10	1010	FL1121	1121	Airline D	Chicago
11	1011	FL1466	1466	Airline A	Phoenix
12					
		1614			

In the above image, I have chosen Column from Example.

TASK 4: CALCULATIONS USING DAX (10 MARKS)

Question 1:

Calculate: Total passengers for a specific flight.

Answer: Followed the following steps:

1. Under the Passenger_information table on the Data Pane → right click → add new measure → Write the DAX formula as:
TotalPassengers = COUNTROWS(passenger_information)
2. Press Enter.

The screenshot shows the Power BI interface with the 'Measure tools' tab selected. In the top left, a new measure is being defined with the name 'TotalPassengers'. The formula bar contains the DAX formula: `1 TotalPassengers = COUNTROWS(passenger_information)`. The 'Format' dropdown is set to 'Whole number'. The 'Data category' is 'Uncategorized'. The 'Properties' pane shows the measure's type as 'Measure'. The 'Visualizations' pane on the right lists various chart types. The 'Data' pane on the far right shows the data model with tables like 'flight_information', 'passenger_information', and 'ticket_information', and their respective columns.

The above image shows how we have calculated total passengers.

Question 2: Using DAX Calculate Total tickets booked?

Answer: Followed the following steps:

- Under the Ticket_information table on the Data Pane → right click → add new measure → Write the DAX formula as:
Total Tickets Booked =
`CALCULATE(COUNTROWS(ticket_information), ticket_information[BookingStatus] = "Confirmed")`
- In this we are only counting the numbers of Tickets that are booked and status is Confirmed.

This screenshot is similar to the previous one but shows a different measure being created. The measure is named 'Total Tickets Booked' and its formula is `1 Total Tickets Booked = CALCULATE(COUNTROWS(ticket_information), ticket_information[BookingStatus] = "Confirmed")`. The 'Format' dropdown is set to 'Whole number'. The 'Data category' is 'Uncategorized'. The 'Properties' pane shows the measure's type as 'Measure'. The 'Visualizations' and 'Data' panes are also visible on the right side of the screen.

Question 3: Using DAX Calculate Filtered table showing "Best" flights only.

Answer: Followed the following steps:

1. Go to the table View → Under table tool click on New Table.
2. As we already have the column which shows the Flight_Category as “Best” and “To be Improved” (From Task 3.1 the conditional column).
3. Write a DAX Formula:
BestFlights = FILTER
(flight_information, flight_information[Flight_Category] = "Best").

The screenshot shows the Power BI Data Editor interface. At the top, there's a ribbon with tabs like File, Home, Help, and Table tools. The 'Table tools' tab is selected. Below the ribbon, there's a toolbar with icons for Name, Manage relationships, New measure, Quick measure, New table, Mark as date table, and Calendars. A search bar labeled 'Search' is also present. On the left, there's a table structure view showing columns: FlightID, FlightNumber, Airline, Destination, Status, FlightNumber_Only, and Flight Category. The table contains 48 rows of flight data. On the right, there's a 'Data' pane showing the structure of the 'BestFlights' table and its relationship with other tables: flight_information, passenger_information, and ticket_information. The 'flight_information' table is expanded, showing columns like Airline, Destination, FlightCategory, FlightID, FlightNumber, FlightNumber_Only, and Status. The 'passenger_information' and 'ticket_information' tables are also listed.

The above image shows that we have created a filtered table that only shows the flight those falls under “Best” Category.

TASK 5: VISUALIZATION AND INTERACTIVE FEATURES (20 MARKS)

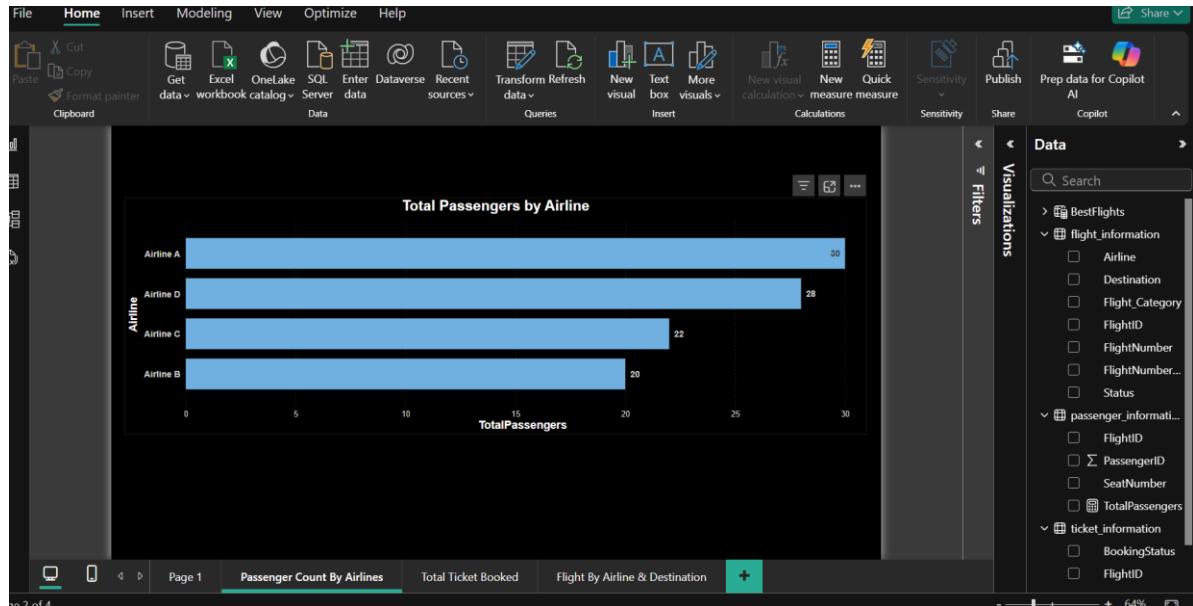
Question: Create visuals for:

1. Passenger count by airline

Answer: Followed the following steps:

1. Add a New Page and Rename it as Passenger Count By Airlines.

2. From the Visualization pane use **Clustered Bar Chart**.
3. In Y- Axis Drag → Airline from Flight_Information table.
4. In the X- Axis Drag → TotalPassengers (Measure that earlier created).
5. Click on Format your Visual under the Visualization Pane → Turn on the Data label and format it according to readability.

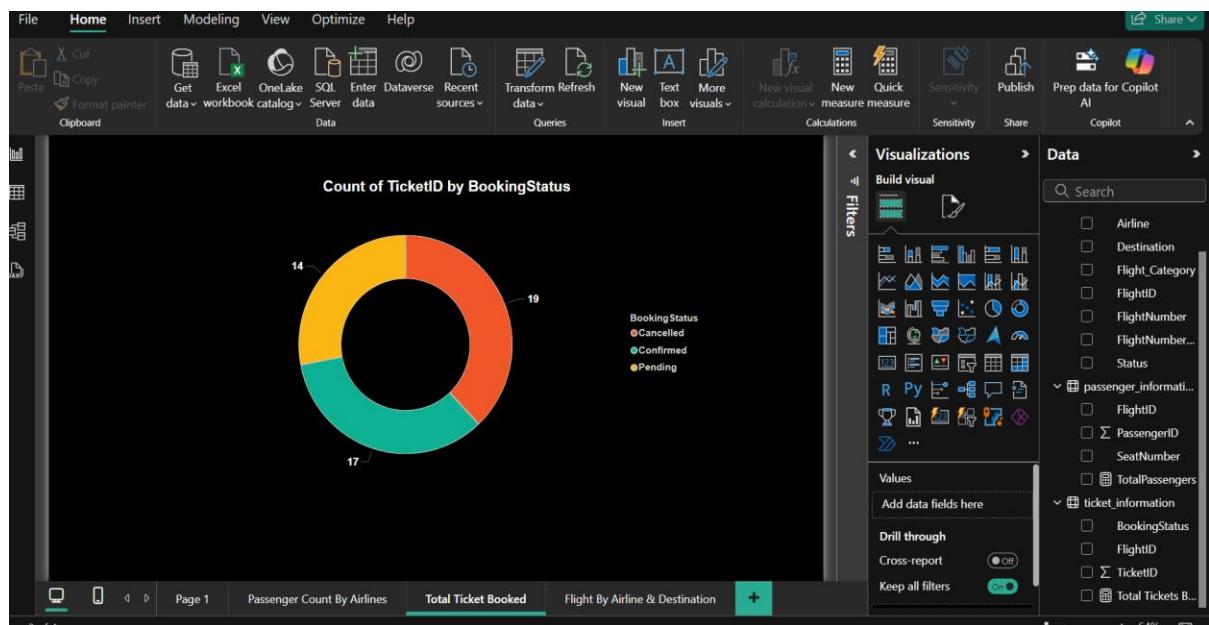


Airline A has the highest number of passengers (30), followed closely by Airline D (28), Airline B carried the fewest passengers (20).

Question 2: Ticket booking statuses.

Answer: Followed the following steps:

1. Add a New Page and Rename it as Total Ticket Booked.
2. Add a **Donut chart** from the Visualization Pane.
3. Under Legends Drag → BookingStatus from Ticket Information table.
4. Under Values Drag → TicketID (Set that as Count) from the same table.
5. From the Format your visual → Turn on the detailed Label and then Under Label Content → set as Data Value.
6. Format the chart according to the readability.

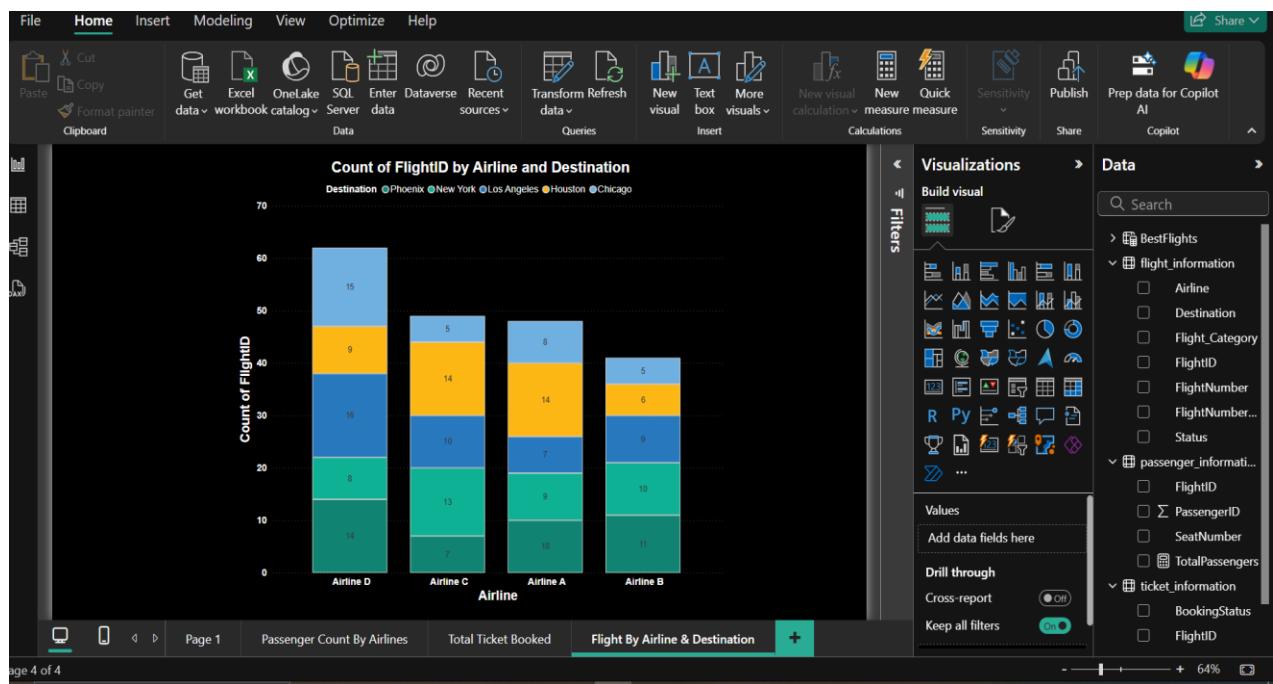


The above image shows the count of Ticket those are confirmed are 17, 19
tickets are cancelled and 14 is under Pending state.

Question 3: Flights by airline and destination.

Answer: Followed the following steps:

1. Add a New Page and Rename it as Flight by Airline & Destination.
2. From the Visualization pane select **Stacked Column Chart**.
3. Under X- Axis → Drag Airline from Flight Information Table.
4. Under Y-Axis → Count of FlightID from the same table.
5. Under Legends → Keep Destination.
6. Go to Format your Visual → Turn On Data Label.
7. Change the font size and the title of X and Y axis as per readability.



The above image shows that Airline D has the highest total flights, particularly to New York and Phoenix, while Airline B has the lowest number of total flights distributed fairly evenly across destinations.

Question 4: Add interactive features for:

→ Destination and Airline

Answer: Followed the following steps:

For doing this first copy all the visuals from the 3 pages:

1. Passengers count by Airlines.
2. Total Ticket Booked.
3. Flight by Airline & destination.

After this we will rename the page as Dashboard, and then arrange all these visuals as per readability.

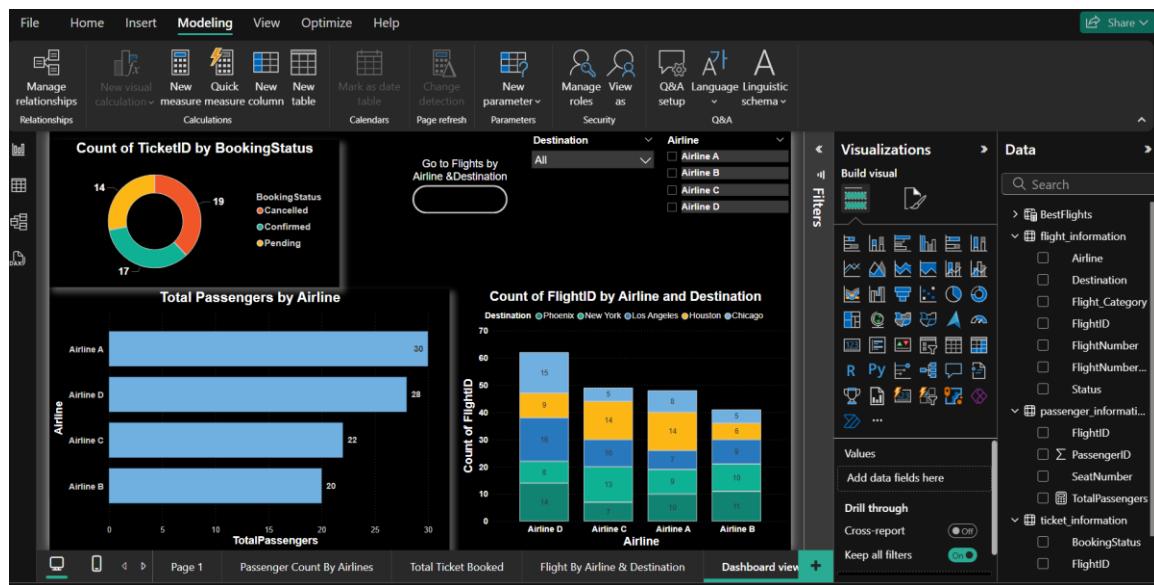
Now to Add Interactive feature we will use **Slicer** from the Visualization pane.

1. Add 1st slicer and under Field Drag → Airline from Flight Information table.
2. Add 2nd slicer and under Field Drag → Destination from Flight Information table → Go to format your visual and then slicer setting and set the style as dropdown.
3. Place the slicer as per convenience.

Question 2: → Quick views.

Answer: Followed the following steps:

1. For this we can Add a Quick View Button for Viewing the previous Page “Flight by Airline & Destination”.
2. Go to Insert → Buttons → Blank.
3. Click on the button and Add a title from Format your Visual → general → Title → “Go to Flight By Airline and Destination”.
4. Come back to button view in the same and then turn on Action → Type → set it as Page Navigation.
5. Under destination Choose Flight By Airline & Destination.



The above image shows that, it is a Dashboard where we can use the slicers to select the destination and the Airline and according to that it will show the data, also created a button that will navigate the user to go to the previous page of Flight By Airline & Destination.

Question: Airline-specific pages.

Answer: Followed the following steps:

Create a Dedicated Page for Airline A

1. Duplicate an Existing Page:
 - o Right-click on your main dashboard page ("Dashboard view") in the tabs at the bottom.
 - o Click Duplicate Page.
 - o Rename the new page to:
→ Airline A Dashboard
2. Apply Page-Level Filter:
 - o Go to the Filters pane (on the right side).
 - o Drag the Airline field from flight_information into the Page Filters area.
 - o Select only:
 - Airline A
3. Customize the Title:
 - o Add a text box (Insert → Text Box) and write:
→ "Airline A Operational Overview"
4. Confirm Filtering:
 - o All visuals on this page will now only show data related to Airline A.

Note: Removed the slicers and the Button as it requires no filtering after this.

The screenshot shows the Power BI desktop interface with a dark theme. The ribbon is visible at the top with tabs like File, Home, Insert, Modeling, View, Optimize, and Help. The Insert tab is selected. The main area displays three visualizations: a donut chart titled 'Count of TicketID by BookingStatus' with segments for Cancelled (red), Pending (orange), and Confirmed (teal); a horizontal bar chart titled 'Total Passengers by Airline' showing 30 passengers for Airline A; and a stacked bar chart titled 'Count of FlightID by Airline and Destination' showing counts for Phoenix, New York, Los Angeles, Houston, and Chicago. On the right side, there is a 'Filters' pane and a 'Data' pane showing the schema of the 'flight_information' table. The bottom navigation bar shows tabs for 'By Airline & Destination', 'Dashboard view', 'Airline A Dashboard' (which is the active tab), 'Airline B Dashboard', 'Airline C Dashboard', and 'Airline D Dashboard'. There is also a '+' button to add new visualizations.

Create a Dedicated Page for Airline B:

1. Duplicate an Existing Page:
 - o Right-click on your main dashboard page ("Dashboard view") in the tabs at the bottom.
 - o Click Duplicate Page.
 - o Rename the new page to:
→ Airline B Dashboard
2. Apply Page-Level Filter:
 - o Go to the Filters pane (on the right side).
 - o Drag the Airline field from flight_information into the Page Filters area.
 - o Select only:
 - Airline B
3. Customize the Title:
 - o Add a text box (Insert → Text Box) and write:
→ "Airline B Operational Overview"
4. Confirm Filtering:
 - o All visuals on this page will now only show data related to Airline B.

Note: Removed the slicers and the Button as it requires no filtering after this.

The screenshot shows the Power BI desktop interface with the 'Airline B Dashboard' page active. The dashboard includes:

- A donut chart titled "Count of TicketID by BookingStatus" with segments: Confirmed (3), Pending (6), and Cancelled (3).
- A horizontal bar chart titled "Total Passengers by Airline" showing 20 passengers for Airline B.
- A stacked bar chart titled "Count of FlightID by Airline and Destination" showing counts for various destinations: Phoenix (5), New York (8), Los Angeles (9), Houston (10), and Chicago (11).

The 'Visualizations' pane on the right lists filters applied to the 'flight_information' table, specifically for 'Airline' set to 'Airline B'. The top ribbon has the 'Insert' tab selected.

Create a Dedicated Page for Airline C

1. Duplicate an Existing Page:
 - o Right-click on your main dashboard page ("Dashboard view") in the tabs at the bottom.
 - o Click Duplicate Page.
 - o Rename the new page to:
→ Airline C Dashboard
2. Apply Page-Level Filter:
 - o Go to the Filters pane (on the right side).
 - o Drag the Airline field from flight_information into the Page Filters area.
 - o Select only:
 - Airline C
3. Customize the Title:
 - o Add a text box (Insert → Text Box) and write:
→ "Airline C Operational Overview"
4. Confirm Filtering:
 - o All visuals on this page will now only show data related to Airline C.

Note: Removed the slicers and the Button as it requires no filtering after this.

The screenshot shows the Power BI desktop interface with the 'Airline C Dashboard' selected. The ribbon is visible with the 'Insert' tab selected. The main area contains three visualizations:

- A donut chart titled "Count of TicketID by BookingStatus" showing values for Cancelled (3), Confirmed (6), and Pending (4).
- A bar chart titled "Total Passengers by Airline" showing passengers for Airline C (22) and Airline A (1).
- A stacked bar chart titled "Count of FlightID by Airline and Destination" showing flights for Airline C (14), Airline A (10), Airline B (13), and Airline D (5) across destinations Phoenix, New York, Los Angeles, Houston, and Chicago.

The 'Visualizations' pane on the right lists the data sources:

- flight_information
 - Airline
 - Destination
 - Flight_Category
 - FlightID
 - FlightNumber
 - FlightNumber...
 - Status
- passenger_informati...
 - FlightID
 - PassengerID
 - SeatNumber
 - TotalPassengers
- ticket_information
 - BookingStatus
 - FlightID

Create a Dedicated Page for Airline D

1. Duplicate an Existing Page:
 - o Right-click on your main dashboard page ("Dashboard view") in the tabs at the bottom.
 - o Click Duplicate Page.
 - o Rename the new page to:
→ Airline D Dashboard
2. Apply Page-Level Filter:
 - o Go to the Filters pane (on the right side).
 - o Drag the Airline field from flight_information into the Page Filters area.
 - o Select only:
 - Airline D
3. Customize the Title:
 - o Add a text box (Insert → Text Box) and write:
→ "Airline D Operational Overview"
4. Confirm Filtering:
 - o All visuals on this page will now only show data related to Airline D.

Note: Removed the slicers and the Button as it requires no filtering after this.

The screenshot shows the Power BI desktop interface with the following details:

- File, Home, Insert, Modeling, View, Optimize, Help** menu bar.
- Insert ribbon** with options: New page, New visual, Visuals, Q&A, Key influencers, Decomposition tree, AI visuals, Paginated report, Power report, Apps, Automate, Text, Buttons, Shapes, Image, Add a sparkline, Elements, Sparklines.
- Visuals** pane on the left showing three cards:
 - Count of TicketID by BookingStatus**: A donut chart with segments: Pending (4), Confirmed (6), and Cancelled (4).
 - Airline D Operational Overview**: A placeholder card.
 - Total Passengers by Airline**: A bar chart showing passengers for Airline D (28).
- Data** pane on the right showing the data model:
 - BestFlights
 - flight_information
 - Airline
 - Destination
 - Flight_Category
 - FlightID
 - FlightNumber
 - FlightNumber...
 - Status
 - passenger_information
 - FlightID
 - PassengerID
 - SeatNumber
 - TotalPassengers
 - ticket_information
 - BookingStatus
 - FlightID
- Visualizations** pane on the right showing the filters applied:
 - Selected: Airline D
- Bottom tabs**: By Airline & Destination, Dashboard view, Airline A Dasboard, Airline B Dashboard, Airline C Dashboard, **Airline D Dashboard** (highlighted in green), and a plus sign for a new page.

TASK 6: FINAL DASHBOARD AND POWER BI SERVICE (20 MARKS)

Question 1: Design a comprehensive dashboard with key visuals and insights.

Answer: Followed the following steps:

Note: Already Created the Dashboard in Task5 (Question 4), But writing these steps again to do that:

For doing this first copy all the visuals from the 3 pages:

1. Passengers count by Airlines.
2. Total Ticket Booked.
3. Flight by Airline & destination.

After this we will rename the page as Dashboard, and then arrange all these visuals as per readability.

Now to Add Interactive feature we will use **Slicer** from the Visualization pane.

4. Add 1st slicer and under Field Drag → Airline from Flight Information table.
5. Add 2nd slicer and under Field Drag → Destination from Flight Information table → Go to format your visual and then slicer setting and set the style as dropdown.
6. Place the slicer as per convenience.

For → Quick views.

7. For this we can Add a Quick View Button for Viewing the previous Page “Flight by Airline & Destination”.
8. Go to Insert → Buttons → Blank.
9. Click on the button and Add a title from Format your Visual → general → Title → “Go to Flight By Airline and Destination”.
10. Come back to button view in the same and then turn on Action → Type → set it as Page Navigation.
11. Under destination Choose Flight By Airline & Destination.



The final dashboard integrates all critical visualizations and interactivity features, providing users with dynamic insights into airline performance and operations.

Features:

- **Passenger Analysis:** Bar chart to show total passengers per airline.
- **Ticket Booking Overview:** Donut chart to visualize booking statuses.
- **Flight Distribution:** Stacked column chart by airline and destination.
- **Slicers:** Airline and Destination filters for focused insights.
- **Navigation:** Button added for quick access to detailed views.
- **Dedicated Page:** An Airline A-specific page created for role-based insights.

This dashboard serves as the central view for data-driven decision-making.

We can also create a Dashboard from Power BI service → In the power BI service create a New Workspace named as Airline Analysis and from Power BI Desktop → Go to Publish and select the Airline Analysis workspace.

In that once its published in Power BI service → Go to the report → Click on the Three dots for more options and then select Pin to Dashboard → you can either pin to existing Dashboard or can create a new → Rename it as

Airline_Analysis_New and then pin other report pages to the same.
 Click again on the Workspace named as Airline_Analysis.
 It will show the dashboard → click on that and you can format that accordingly.

The screenshot shows the Power BI workspace interface. The left sidebar lists categories like Home, Create, Browse, OneDrive, Apps, Metrics, Monitor, Learn, and Workspaces. Under Workspaces, 'Airline Analysis' is selected. The main area shows a list of items:

Name	Type	Task	Owner	Refreshed	Next refresh	Endorsement	Sensitivity	Included in app
Airline Data Management and Analysis	Report	—	Airline Analysis	6/15/2025, 11:26:27...	—	—	—	No
Airline Data Management and Analysis	Semantic model	—	Airline Analysis	6/15/2025, 11:26:27...	N/A	—	—	No
Airline_Analysis_New	Dashboard	—	Airline Analysis	—	—	—	—	No

Before clicking on the dashboard, it will show like this.

The screenshot shows the Power BI dashboard view titled 'Airline Data Management and Analysis'. The left sidebar shows the workspace navigation. The main area contains three visualizations:

- Count of TicketID by BookingStatus:** A donut chart showing the distribution of ticket counts by booking status: Pending (14), Confirmed (17), and Cancelled (19).
- Total Passengers by Airline:** A horizontal bar chart showing the total number of passengers for each airline: Airline A (30), Airline D (28), Airline C (22), and Airline B (20).
- Count of FlightID by Airline and Destination:** A stacked bar chart showing the count of flight IDs for different airlines and destinations. For example, Airline D has flights to Phoenix (14), Airline C to New York (11), Airline A to New York (9), and Airline B to New York (11).

It will show like this if we will pin the dashboard report page to this.

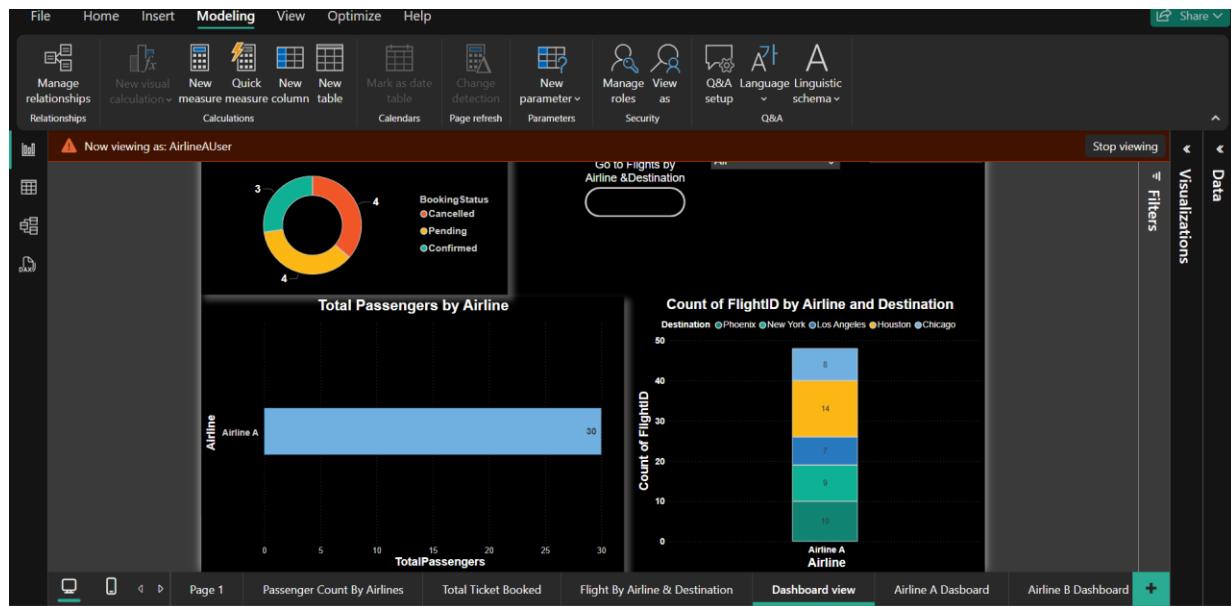
Question 2: Configure Row-Level Security (RLS) for Airline A data and assign it to a user.

Answer: Followed the following steps:

1. In Power BI Desktop:
Go to **Modeling** → **Manage Roles**
2. Click **Create** and name the role:
► **AirlineAUser**
3. Select the **flight_information** table
4. Add this DAX filter:
[Airline] = "Airline A"
5. Click on **Save**.

To test it:

1. Go to **Modeling** → **View as Roles**
2. Select **AirlineAUser** → visuals should now only show Airline A data.



Now save this and Publish it to the Airline Analysis Workspace.

To Assign it to the User:

1. Go to Power BI Service.
2. In your dataset → click Security.
3. Assigned my email address to the role AirlineAUser.

The screenshot shows the 'Row-Level Security' settings for a dataset named 'Airline Analysis'. On the left, there's a sidebar with various navigation options like Home, Create, Browse, etc. The main area is titled 'Members (1)'. It has a text input field 'Enter email addresses' with a placeholder 'People or groups who belong to this role'. Below it is a button labeled 'Add'. A single member, 'Abhishek Ambasta', is listed with a remove button 'x'. At the bottom are 'Save' and 'Cancel' buttons.

Question 3: Set up a schedule refresh at 5 PM daily.

Answer: Followed the following steps:

1. Go to setting → Power BI setting → Click on Semantic Models → expand Gateway and cloud connection → this will show like:

▫ Gateway and cloud connections

To use a data gateway, make sure the computer is online and the data source is added in [Manage Connections and Gateways](#). If you're using an On-premises data gateway (standard mode), please select the corresponding data sources and then click apply.

Gateway connections

Use an On-premises or VNet data gateway

On

Gateway	Department	Contact information	Status	Actions
Personal Gateway			Running on DESKTOP-R8U9DL6	

2. Under Data Source connect it will show like this:

▫ Data source credentials

⌚ Failed to test the connection to your data source. Please retry your credentials. [Learn more](#)

Flight_Information.xlsx ▲ [Edit credentials](#) [Show in lineage view](#) ☐
Passenger_Information.xlsx ▲ [Edit credentials](#) [Show in lineage view](#) ☐
Ticket_Information.xlsx ▲ [Edit credentials](#) [Show in lineage view](#) ☐

3. To resolve this, we will click on the edit credentials → click sign in do it for 3 times.

▫ Data source credentials

Flight_Information.xlsx [Edit credentials](#) [Show in lineage view](#) ☐
Passenger_Information.xlsx [Edit credentials](#) [Show in lineage view](#) ☐
Ticket_Information.xlsx [Edit credentials](#) [Show in lineage view](#) ☐

This will resolve the error.

1. By scrolling down, you will find Refresh option → expand that → Select time Zone as India UTC +05:30 → click on add time and set the time as 05:00 PM.
2. Click on Apply.

The screenshot shows the 'Power BI' interface for 'Airline Analysis'. On the left, there's a sidebar with various icons. The main area is titled 'Data source credentials' and lists three datasets: Flight_Information.xlsx, Passenger_Information.xlsx, and Ticket_Information.xlsx, each with 'Edit credentials' and 'Show in lineage view' links. Below this is a 'Parameters' section. The 'Refresh' section is expanded, showing a 'Time zone' dropdown set to '(UTC+05:30) Chennai, Kolkata, Mumbai'. A note says: '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.' Below this is a 'Configure a refresh schedule' section with a 'On' toggle switch turned on. Under 'Refresh frequency', 'Daily' is selected. Under 'Time', the time is set to '5 00 PM'. There's a 'Add another time' link. At the bottom of the refresh section, there are 'Send refresh failure notifications to' checkboxes for 'Semantic model owner' (which is checked) and 'These contacts' (unchecked), with a text input field for email addresses. At the very bottom are 'Apply' and 'Discard' buttons.

This will set the daily refresh time as 05:00 PM as per Indian time zone.