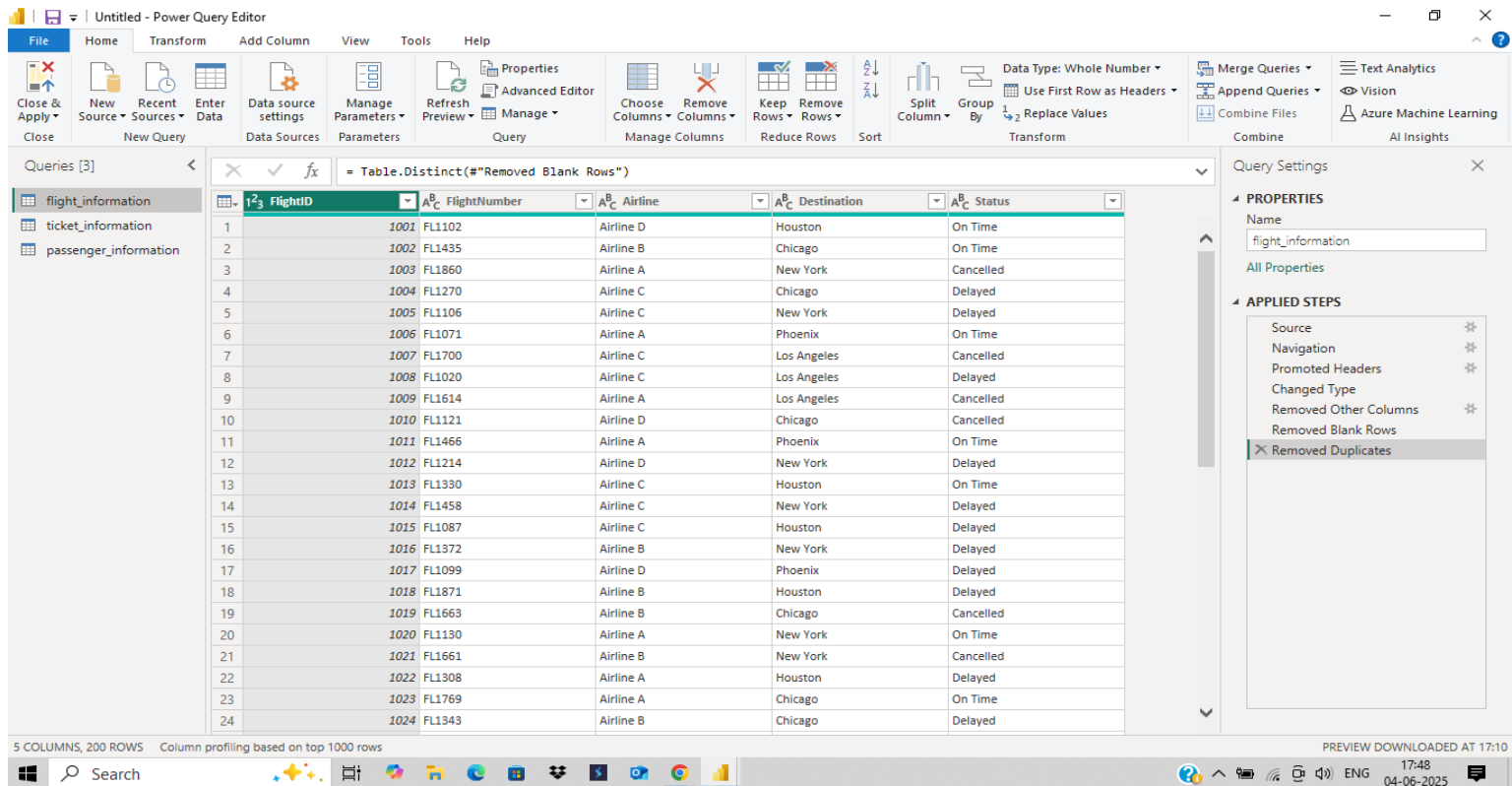# 1. Data Preparation and Cleaning:

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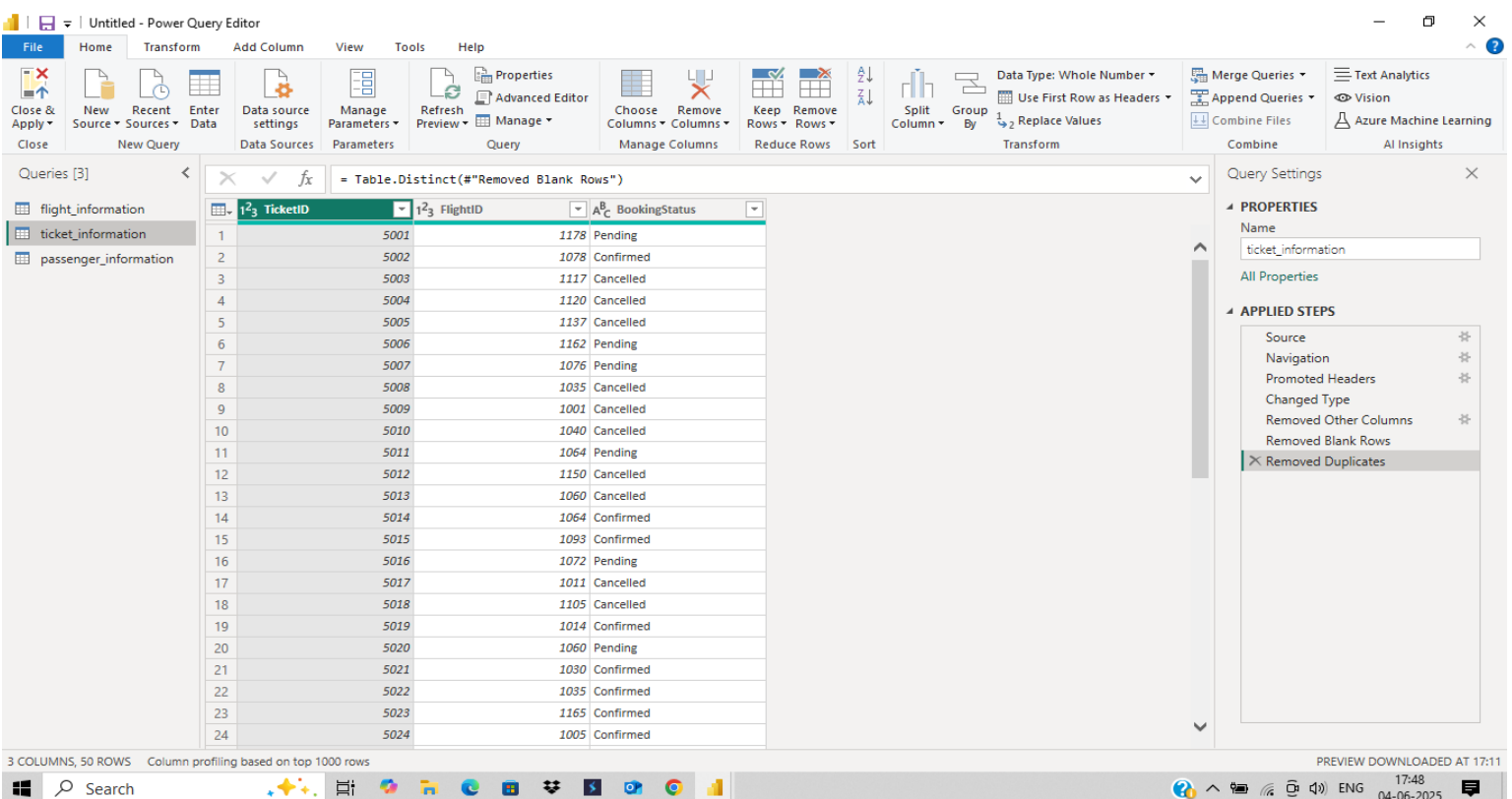- **Deliverables**: Screenshot of Power Query Editor showing cleaned data.

❖ **Flight_information Table:**

❖ **Ticket_information Table:**

❖ **Passeanger_information Table:**

To Perform This task:

- First Import all the datasets using Power BI Query

- Select all the data > Remove Rows > Remove Duplicates

- Any null value replaces it with specific value or Remove Rows > Remove errors

- Format the data types of column correctly.

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.

❖ **Create Relationships:**

To perform this task:
- Go to the Model View
- Drag FlightID column form flight_information table to FlightID column in passenger_information.
- Drag FlightID column from flight_information table to FlightID column in ticket_information.
- New Relationship box will open > set cardinality to one to many.
- And Cross filter direction to single.
- Click save
- Then you will see the relationship between table.

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.

❖ Conditional Column:

❖ Column with example:

To perform this task:

- **Conditional Column:**
    - Select Column Status > Add Column > Conditional Column
    - Name the Column > **Operatinal_Rating.**
    - If **Status** is equal to "**On Time**" set output as "**Best**"
    - Else set output as "**To Be Improved**"
- **Column with example:**
    - Select Column FlightNumber > Add Column > Column from example
    - Select from selection
    - Then provide with number as output
    - For example -> **FL1102 – "1102" > Ctrl+enter > ok**.

4. Calculations Using DAX:

- **Calculate:**
    - Total passengers for a specific flight.
    - Total tickets booked.
    - Filtered table showing "Best" flights only.
- **Deliverables**: Screenshot of DAX calculations and results.

❖ **Total Passengers For specific flight:**

❖ **Total Tickets Booked:**

Formula bar:
`1  Total Tickets Booked = CALCULATE(COUNT(ticket_information[TicketID]),ticket_information[BookingStatus]="Confirmed")`

Card visual showing:
17
Total Tickets Booked

❖ **Only Best Flights:**

Formula bar:
`1  Best Flights = FILTER(flight_information,flight_information[Operational_Rating]="Best")`

| FlightID	FlightNumber	Airline	Destination	Status	flight number	Operational_Rating
1001	FL1102	Airline D	Houston	On Time	1102	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	FL1330	Airline C	Houston	On Time	1330	Best
1020	FL1130	Airline A	New York	On Time	1130	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	1805	Best
1028	FL1385	Airline D	Chicago	On Time	1385	Best
1029	FL1191	Airline D	Los Angeles	On Time	1191	Best
1030	FL1955	Airline B	Phoenix	On Time	1955	Best
1031	FL1276	Airline B	New York	On Time	1276	Best
1033	FL1459	Airline D	New York	On Time	1459	Best
1034	FL1313	Airline B	Phoenix	On Time	1313	Best
1036	FL1252	Airline D	Phoenix	On Time	1252	Best
1039	FL1560	Airline B	Chicago	On Time	1560	Best
1043	FL1681	Airline C	Houston	On Time	1681	Best
1044	FL1475	Airline B	Phoenix	On Time	1475	Best
1046	FL1975	Airline D	Chicago	On Time	1975	Best
1048	FL1189	Airline A	New York	On Time	1189	Best
1050	FL1686	Airline C	Phoenix	On Time	1686	Best
1052	FL1562	Airline D	Phoenix	On Time	1562	Best

Table: Best Flights (82 rows)

To Perform this task:

- First Go to Home > Enter data > Create new table name "Measure">
- **Total Passengers:**
  - Right Click Measure Table > New Measue.
  - Write the Dax Function using COUNT function.
  - **Total Passengers = COUNT(passenger_information(PassengerID))**
  - Then select a table from visualization pane
  - Add Airline from flight_information and Total Passengers from Measure table
  - It will show total passenger for different airline
- **Total tickets booked:**
  - Right Click Meaure table > New Measure
  - Write Dax Function Using CALCULATE
  - **CALCULATE(COUNT(ticket_information[TicketID]),ticket_information[BookingStatus]= "Confirmed")**
  - Then add single row card > add Total tickets booked measure.
- **Only Best Flights:**
  - Go to table view > New table
  - Write DAX function to create new table with filtered data
  - **FILTER(flight_information, flight_information[Operational_Rating]="Best")**
 - New table with filtered data will be created

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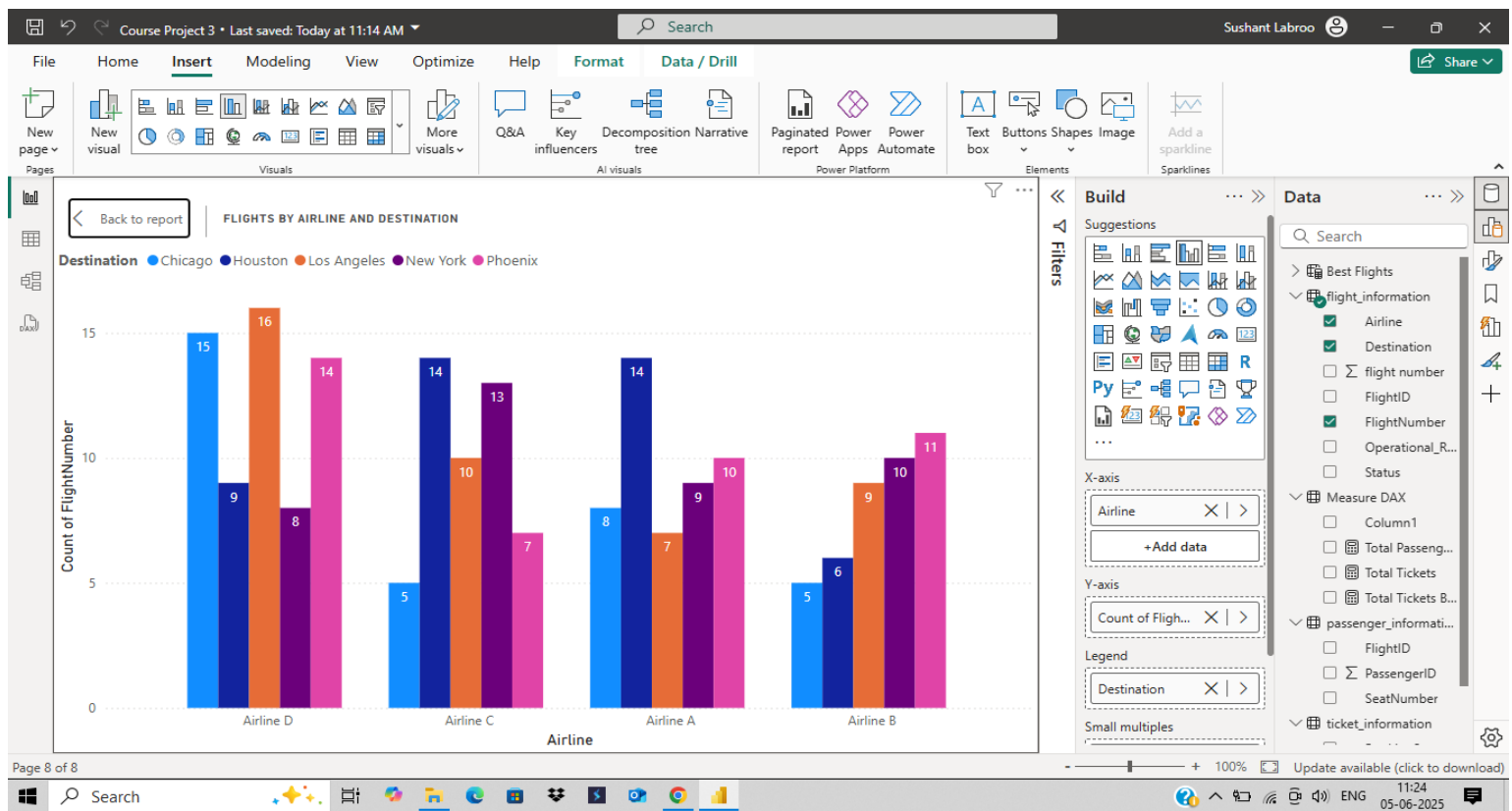- **Deliverables**: Screenshots of all visuals and interactive features.

❖ **Passenger Count by Airline:**
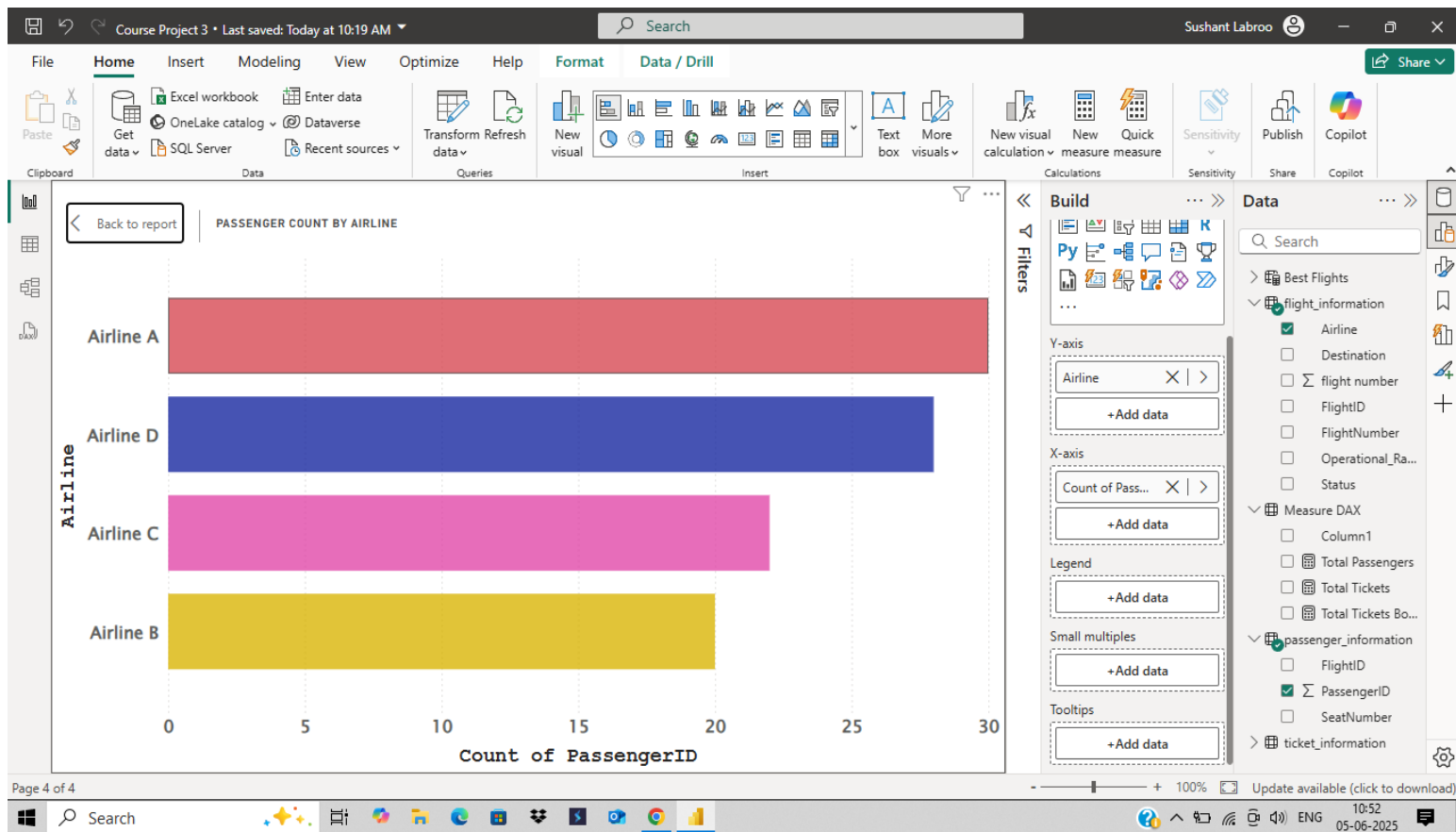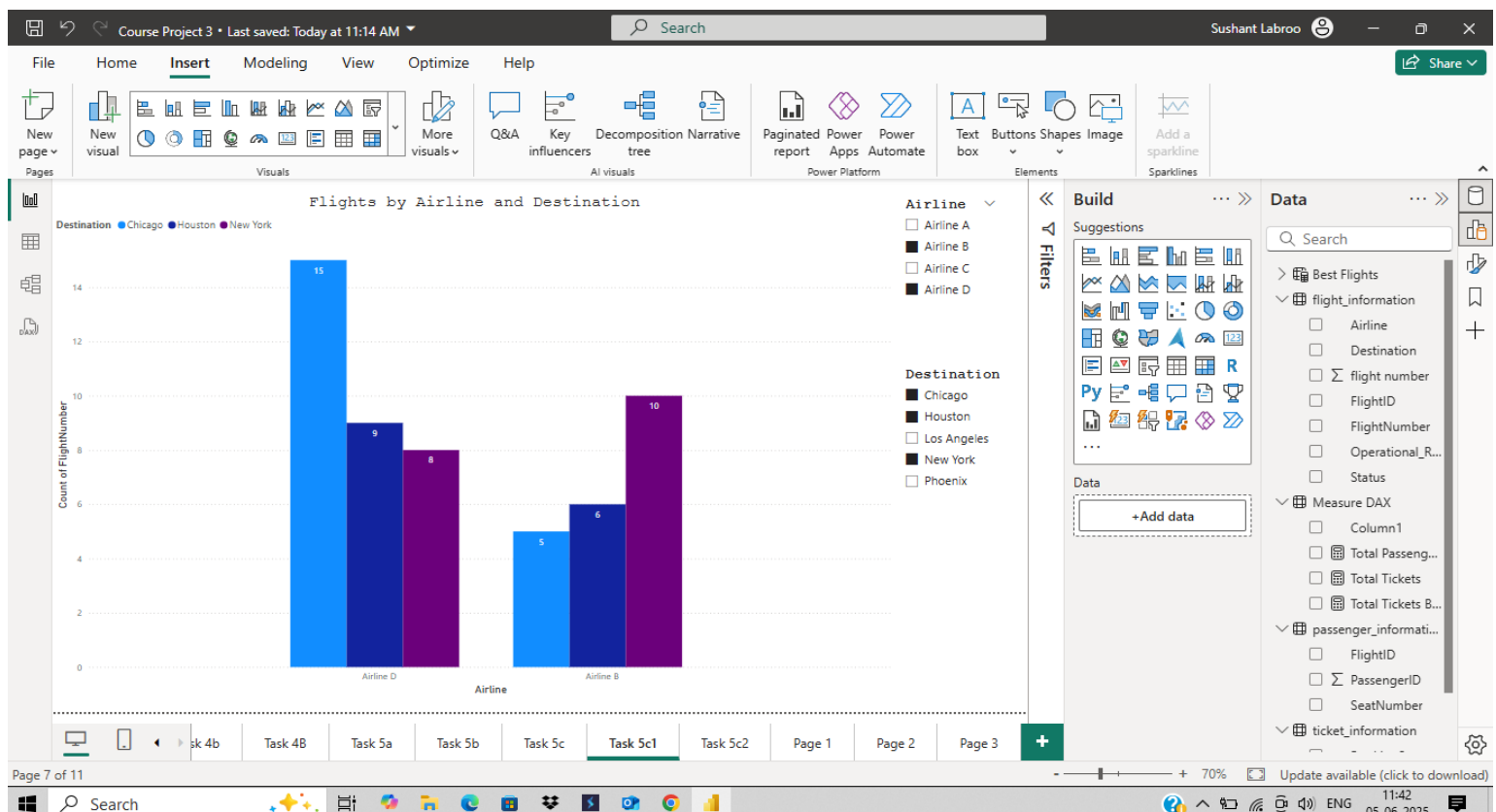

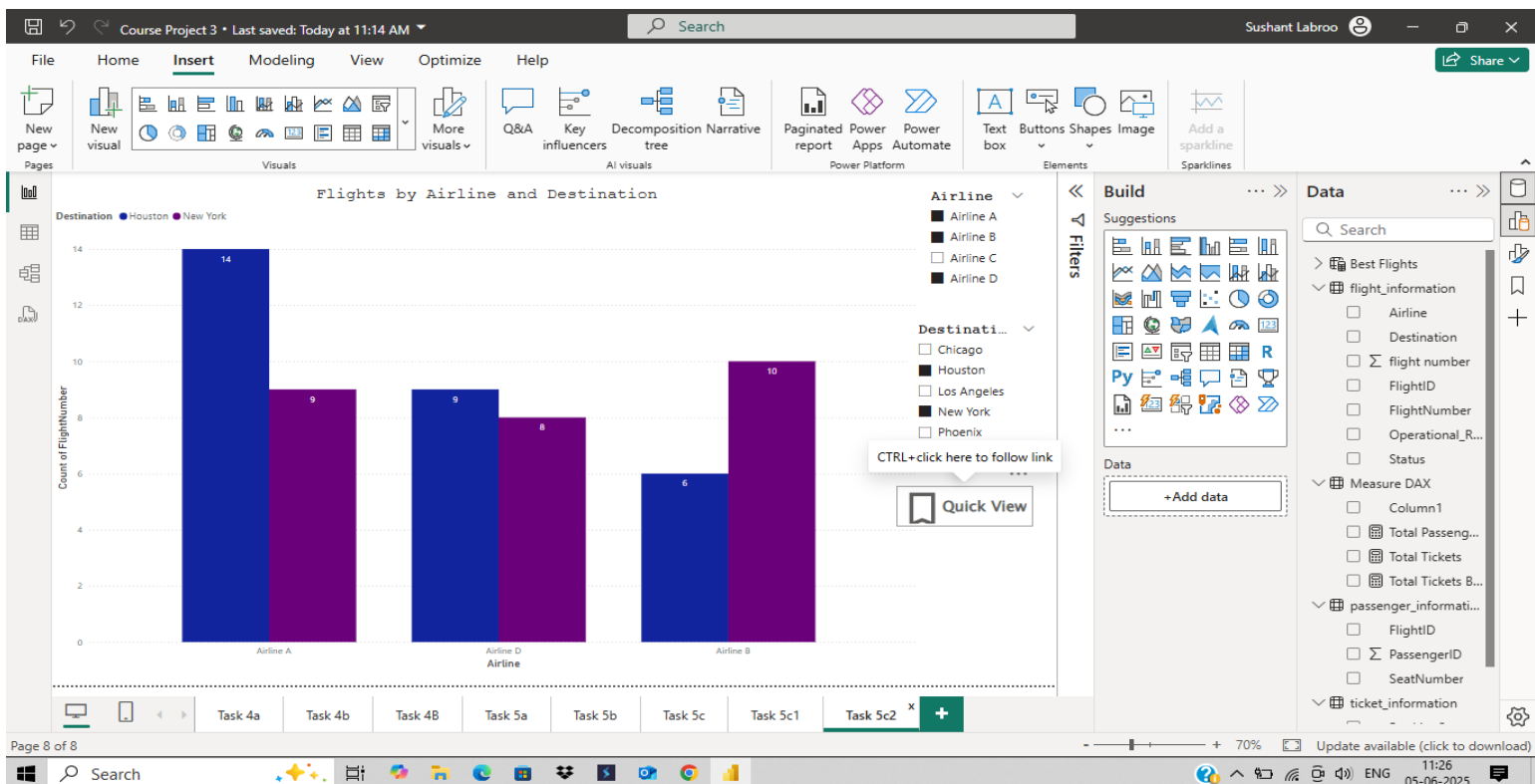❖ **Ticket Booking Status:**
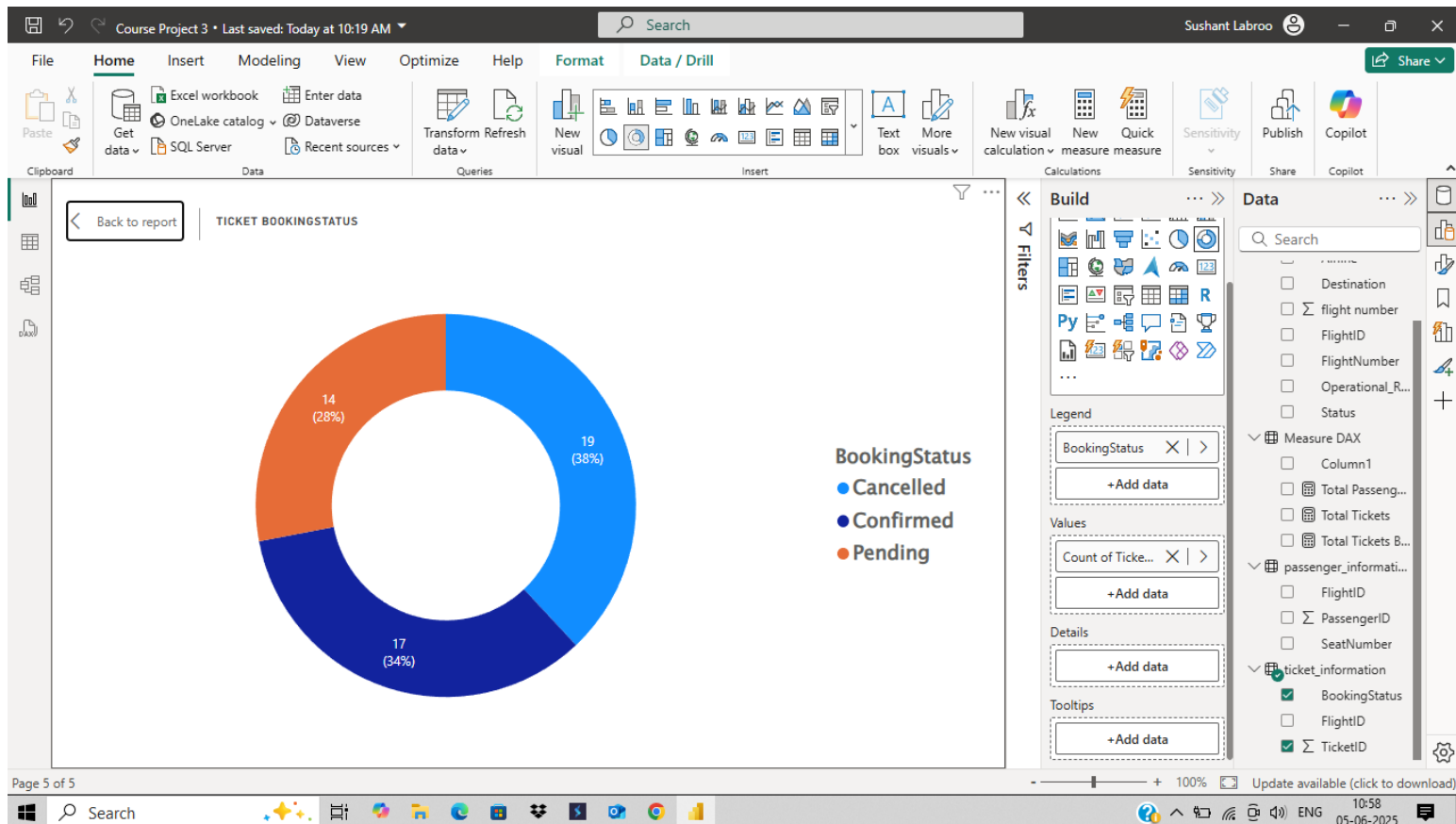
❖ **Flights By airline and destination:**

❖ **Interactive Functions:**
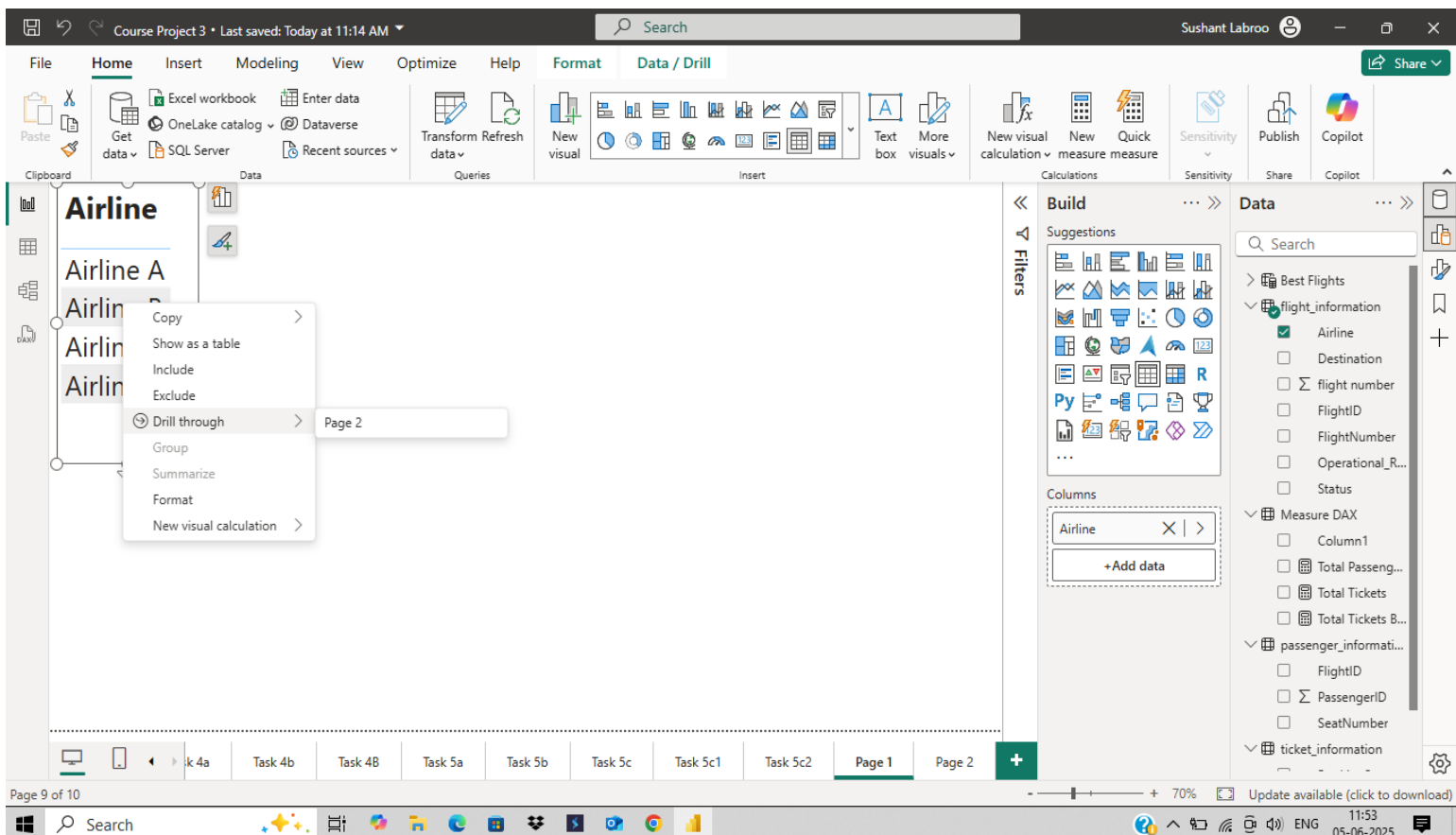   o **Using Slicers**
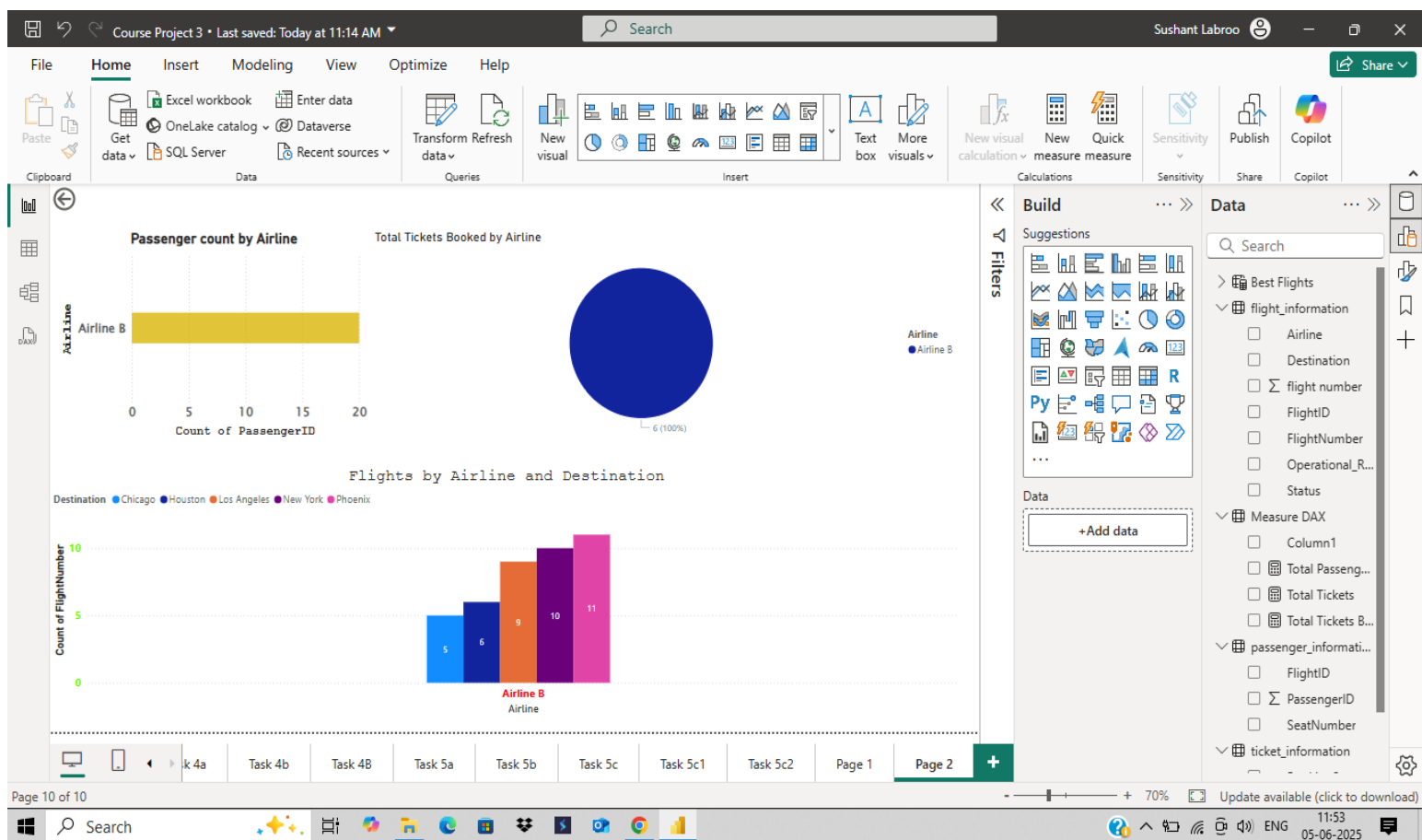
- **Bookmarks:**

- **Drill Through:**
  - Airline page to airline specific data

- All Airline data for specific airline:
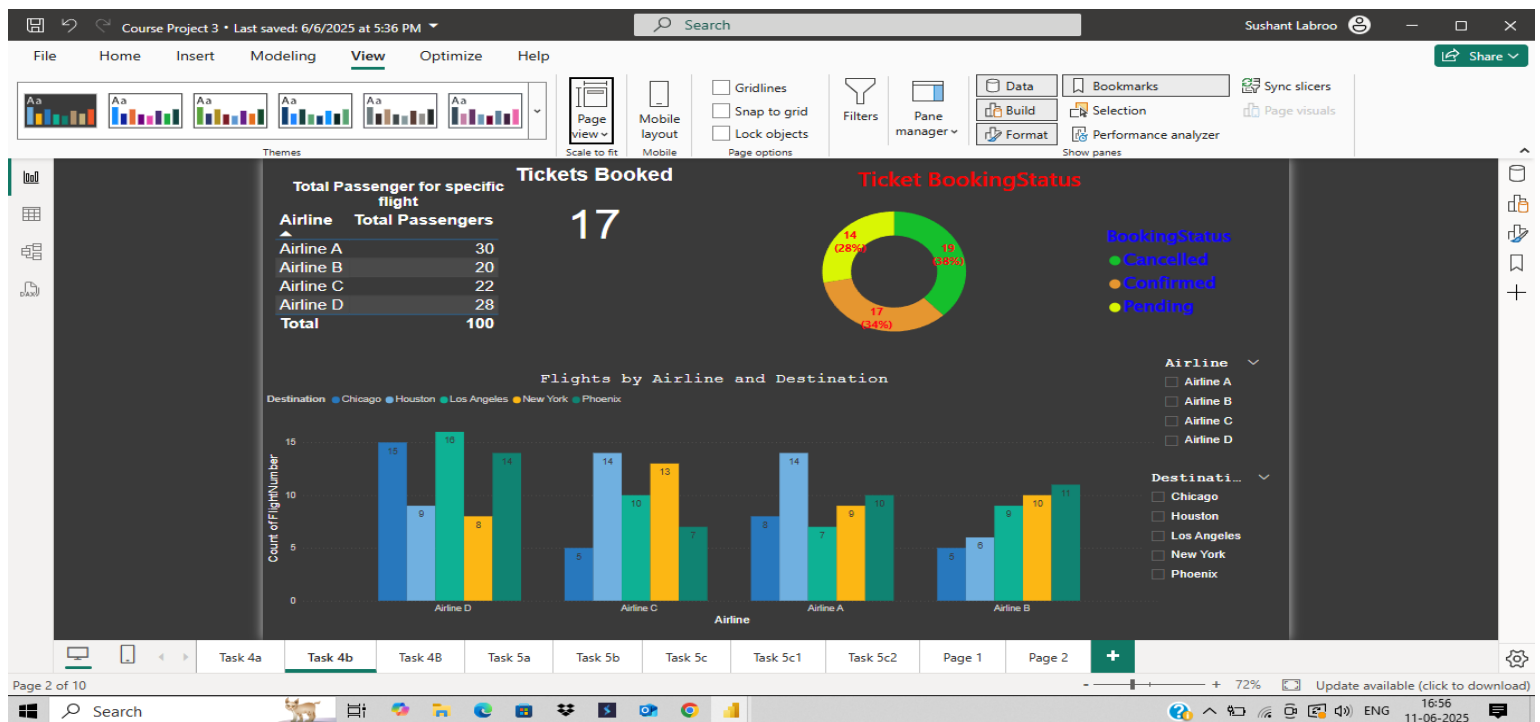

To Perform this task:
- **Visuals are:**
  - Bar Chart for Passenger count by airline
  - Pie Chart for Ticket Booking Status
  - Clustered Column Chart for Flights by Airline and destination.
- **Interactive Features:**
  - Slicers for airline and destination
  - Bookmarks for selected visual
  - Drill Through for all data related to specific airline.

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.
  - Set up a **schedule refresh** at 5 PM daily.
  - **Deliverables**: Screenshot of the published dashboard and RLS configuration.

❖ **Dashboard:**

❖ **Row Level Security:**

➢ **Next step:**

➢ **View As:**

> **Assign to a user:**

❖ **Schedule A Refresh:**

To Perform this task:
- **Dashboard:**
    - First publish the file to Power BI Service in a workspace.
    - Then open the report file in the workspace
    - Then select the visual you want to be there in dashboard.
    - Click the three dots next to edit option > Pin to dashboard.
    - Create new > Pin live
    - Then repeat the steps for other visuals.
    - Then you will get to see a clean dashboard in dashboard view.
- **Row-Level Security (RLS):**
    - Go to Modelling > Manage Roles > New > Type role name > Select table from you want to filter > New > Give Filter e.g. -> Airline A
    - Switch to DAX editor: [Airline]= "Airline A".
    - Click Save
    - Click Views As under Modelling > Select the role name
    - Now you will view only the given value data.
    - Publish the file to Power BI Service in a workspace.
    - Go to the workspace > Click 3 dots next to data model > Security > Select the role name > Give mail id to assign the RLS.
- **Schedule a Refresh:**
    - Go to Workspaces > Settings > Power BI Settings > Semantic Models > Refresh > Set the time accordingly.
    - Then set the refresh failure notification to.

Video submission –

https://drive.google.com/file/d/1HD-ghlh8o4zYTgA1EvnwzeJsl5hwWbcA/view?usp=drive_link

https://www.loom.com/share/c000b4a513b846abb04bacebd541cd86?sid=7b81fcac-fdf4-4e92-8248-a0c0ef1c2f2b