

TASK-1. Data Preparation and Cleaning

(a). Extract and transform data in Power Query.

ANS.

The screenshot shows the Power Query Editor interface. The main area displays a table with 3 columns: PassengerID, FlightID, and SeatNumber. The data is extracted from a source named 'Passenger_Information'. The formula bar shows the query definition: `= Table.TransformColumns("#Extracted First Characters", {"SeatNumber", each Text.Start(_, 1), type text})`. The right-hand pane shows the 'Query Settings' for 'Passenger_Information - passenger_infor', including the 'APPLIED STEPS' list: Source, Promoted Headers, Extracted First Characters, and Extracted First Characters1.

PassengerID	FlightID	SeatNumber
1	1	1161
2	2	1157
3	3	1141
4	4	1046
5	5	1035
6	6	1134
7	7	1082
8	8	1115
9	9	1197
10	10	1047
11	11	1153
12	12	1194
13	13	1010
14	14	1056
15	15	1030
16	16	1109
17	17	1005
18	18	1119
19	19	1033
20	20	1118
21	21	1065
22	22	1146
23	23	1177
24	24	1011

(b). Clean data: remove duplicates, handle missing values, and format columns?

The screenshot shows the Power Query Editor interface with a table containing 5 columns: FlightID, FlightNumber, Airline, Destination, and Status. The data is extracted from a source named 'Flight_Information'. The formula bar shows the query definition: `= Table.Distinct("#Filled Up")`. The right-hand pane shows the 'Query Settings' for 'Flight_Information - flight_infor', including the 'APPLIED STEPS' list: Source, Promoted Headers, Changed Type, Filled Down, Filled Up, and Removed Duplicates.

FlightID	FlightNumber	Airline	Destination	Status
1	1001 FL1302	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	Phoenix	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 FL1390	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
21	1021 FL1661	Airline B	New York	Cancelled
22	1022 FL1308	Airline A	Houston	Delayed
23	1023 FL1769	Airline A	Chicago	On Time
24	1024 FL1343	Airline B	Chicago	Delayed

(C). Deliverables: Screenshot of Power Query Editor showing cleaned data.

ANS.

The screenshot displays the Microsoft Power Query Editor interface. The title bar indicates 'Untitled - Power Query Editor'. The ribbon menu includes 'File', 'Home', 'Transform', 'Add Column', 'View', 'Tools', and 'Help'. The 'Transform' tab is active, showing options like 'Merge Queries', 'Append Queries', 'Combine Files', 'Text Analytics', 'Vision', and 'Azure Machine Learning'. The main area shows a table with 5 columns: FlightID, FlightNumber, Airline, Destination, and Status. The data is filtered to show only rows where the status is 'On Time'. The 'Query Settings' pane on the right shows the query name 'Flight_Information - flight_information' and a list of applied steps: Source, Promoted Headers, Changed Type, Filled Down, Filled Up, Removed Duplicates, and Filtered Rows. The status bar at the bottom indicates '5 COLUMNS, 82 ROWS' and 'Column profiling based on top 1000 rows'. The preview was downloaded at 11:37 AM.

FlightID	FlightNumber	Airline	Destination	Status	
1	1001	FL1102	Airline D	Houston	On Time
2	1002	FL1435	Airline B	Chicago	On Time
3	1006	FL1071	Airline A	Phoenix	On Time
4	1011	FL1466	Airline A	Phoenix	On Time
5	1013	FL1330	Airline C	Houston	On Time
6	1020	FL1130	Airline A	New York	On Time
7	1023	FL1769	Airline A	Chicago	On Time
8	1025	FL1491	Airline D	Phoenix	On Time
9	1027	FL1805	Airline D	Chicago	On Time
10	1028	FL1385	Airline D	Chicago	On Time
11	1029	FL1191	Airline D	Los Angeles	On Time
12	1030	FL1955	Airline B	Phoenix	On Time
13	1031	FL1276	Airline B	New York	On Time
14	1033	FL1459	Airline D	New York	On Time
15	1034	FL1313	Airline B	Phoenix	On Time
16	1036	FL1252	Airline D	Phoenix	On Time
17	1039	FL1560	Airline B	Chicago	On Time
18	1043	FL1681	Airline C	Houston	On Time
19	1044	FL1475	Airline B	Phoenix	On Time
20	1046	FL1975	Airline D	Chicago	On Time
21	1048	FL1189	Airline A	New York	On Time
22	1050	FL1686	Airline C	Phoenix	On Time
23	1052	FL1562	Airline D	Phoenix	On Time
24	1053	FL1875	Airline C	Chicago	On Time

TASK-2. . Data Modeling .

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

ANS.

The screenshot shows the Power BI Desktop interface with two tables, **Ticket_Information** and **Flight_Information**, connected by a relationship line. The **Edit relationship** pane is open, showing the relationship configuration.

From table: Ticket_Information - ticket_information

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

To table: Flight_Information - 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

☐ Assume referential integrity

☐ Apply security filter in both directions

Save **Cancel**

The screenshot shows the Power BI Desktop interface with two tables, **Flight_Information** and **Passenger_Information**, connected by a relationship line. The **Edit relationship** pane is open, showing the relationship configuration.

From table: Passenger_Information - passenger_informati...

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

To table: Flight_Information - 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

☐ Assume referential integrity

☐ Apply security filter in both directions

Save **Cancel**

(b). Understand cardinality and configure the model appropriately.

ANS. Cardinality only use a many to one(*1). And not use a any cardinality .like a(one to one(1:1),one to many(1:*),many to many(*:*)).

Untitled - Power BI Desktop

File Home Help

Clipboard

Get data Excel OneLake SQL Server data

Ticket_Information - tic...

BookingStatus

FlightID

Σ TicketID

Collapse ^

All tables +

Edit relationship

Select tables and columns that are related.

From table

Ticket_Information - ticket_information

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

To table

Flight_Information - 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 many (*:*)

Cross-filter direction

Single ('Flight_Information - flight_informati

☒ Make this relationship active

☐ Apply security filter in both directions

☐ Assume referential integrity

This relationship has cardinality Many-Many. This should only be used if it is expected that neither column (FlightID and FlightID) contains unique values, and that the significantly different behavior of Many-Many

Save Cancel

vibhor gautam

Share

Q&A Language Linguistic schema Q&A Sensitivity Share

Properties Data

Passenger_Information ...

FlightID

PassengerID

SeatNumber

Collapse ^

100%

(C). Deliverables: Screenshot of the data model with relationships.

ANS. It is screenshot of the data model with relationship and cardinality relation is (many to many(*:*)).

The screenshot displays the Power BI Desktop interface with a data model view. Two tables are connected by a relationship line:

- Ticket_Information - tic...** (Columns: BookingStatus, FlightID, TicketID)
- Passenger_Information ...** (Columns: FlightID, PassengerID, SeatNumber)

The relationship is configured with the following settings:

- From table:** Ticket_Information - ticket_information
- To table:** Passenger_Information - passenger_informati
- Cardinality:** Many to many (*:*)
- Cross-filter direction:** Both
- ☒ Make this relationship active
- ☐ Apply security filter in both directions
- ☐ Assume referential integrity

A warning message is displayed: "This relationship has cardinality Many-Many. This should only be used if it is expected that neither column (FlightID and SeatNumber) contains unique values, and that the significantly different behavior of Many-".

The relationship is visualized as a many-to-many relationship with a cardinality of (*:*) and a cross-filter direction of Both.

TASK-3. Enhanced Data Insights.

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

ANS.

The screenshot displays the Microsoft Power BI Desktop interface. The main area shows a data table with 6 columns and 24 rows. The columns are: FlightNumber, Airline, Destination, Status, and STATUS. The STATUS column is a conditional column created based on the Status column. The formula bar at the top shows the DAX formula: `= Table.TransformColumnTypes("#Added Conditional Column",{{"STATUS", type text}})`. The right sidebar shows the Query Settings pane with the name "Flight_Information - flight_information" and a list of applied steps: Source, Promoted Headers, Changed Type, Filled Down, Filled Up, Removed Duplicates, Filtered Rows, Added Conditional Column, and Changed Type1. The bottom status bar indicates "6 COLUMNS, 82 ROWS" and "Column profiling based on top 1000 rows".

	FlightNumber	Airline	Destination	Status	STATUS	
1	1001	FL1102	Airline D	Houston	On Time	BEST
2	1002	FL1435	Airline B	Chicago	On Time	BEST
3	1006	FL1071	Airline A	Phoenix	On Time	BEST
4	1011	FL1466	Airline A	Phoenix	On Time	BEST
5	1013	FL1330	Airline C	Houston	On Time	BEST
6	1020	FL1130	Airline A	New York	On Time	BEST
7	1023	FL1769	Airline A	Chicago	On Time	BEST
8	1025	FL1491	Airline D	Phoenix	On Time	BEST
9	1027	FL1805	Airline D	Chicago	On Time	BEST
10	1028	FL1385	Airline D	Chicago	On Time	BEST
11	1029	FL1191	Airline D	Los Angeles	On Time	BEST
12	1030	FL1955	Airline B	Phoenix	On Time	BEST
13	1031	FL1276	Airline B	New York	On Time	BEST
14	1033	FL1459	Airline D	New York	On Time	BEST
15	1034	FL1313	Airline B	Phoenix	On Time	BEST
16	1036	FL1252	Airline D	Phoenix	On Time	BEST
17	1039	FL1560	Airline B	Chicago	On Time	BEST
18	1043	FL1681	Airline C	Houston	On Time	BEST
19	1044	FL1475	Airline B	Phoenix	On Time	BEST
20	1046	FL1975	Airline D	Chicago	On Time	BEST
21	1048	FL1189	Airline A	New York	On Time	BEST
22	1050	FL1686	Airline C	Phoenix	On Time	BEST
23	1052	FL1562	Airline D	Phoenix	On Time	BEST
24						

(b). Use "Column from Examples" to extract the flight number from FlightNumber.

ANS.

project

File Home Transform Add Column View Tools Help

Column From Custom Invoke Custom Examples Column Function Duplicate Column

Conditional Column Index Column Merge Columns Extract Format Parse

Statistics Standard Scientific Trigonometry Rounding Information

Date Time Duration Text Vision Azure Machine Learning

General From Text From Number From Date & Time AI Insights

Queries [3]

Passenger_Information - ...

Ticket_Information - tick...

Flight_Information - fligh...

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

	FlightNumber	Airline	Destination	Status	FLIGHT NUMBER
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
20	1020 FL1130	Airline A	New York	On Time	1130
21	1021 FL1661	Airline B	New York	Cancelled	1661
22	1022 FL1308	Airline A	Houston	Delayed	1308
23	1023 FL1769	Airline A	Chicago	On Time	1769

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

Query Settings

PROPERTIES

Name

Flight_Information - flight_information

APPLIED STEPS

Source

Promoted Headers

Changed Type

Filled Down

Filled Up

Removed Duplicates

Filtered Rows

Inserted Text After Delimiter

Changed Type1

PREVIEW DOWNLOADED AT 12:03 PM

(c). Deliverables: Screenshot of the transformed data

(A).

Add Conditional Column

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

New column name: STATUS

Column Name: STATUS, Operator: begins with, Value: STATUS, Output: BEST

Else If: STATUS, Operator: ends with, Value: STATUS, Output: TO BE IMPROVE

Else: NO IMPROVE

OK Cancel

FlightID	Airline	Destination	Status
1046	FL1975	Airline D	Chicago
1048	FL1189	Airline A	New York
1050	FL1686	Airline C	Phoenix
1052	FL1562	Airline D	Phoenix

6 COLUMNS, 62 ROWS Column profiling based on top 1000 rows PREVIEW DOWNLOADED AT 5:36 PM

(B).

Add Column From Examples

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

Transform: Text.AfterDelimiter(FlightNumber, ".")

OK Cancel

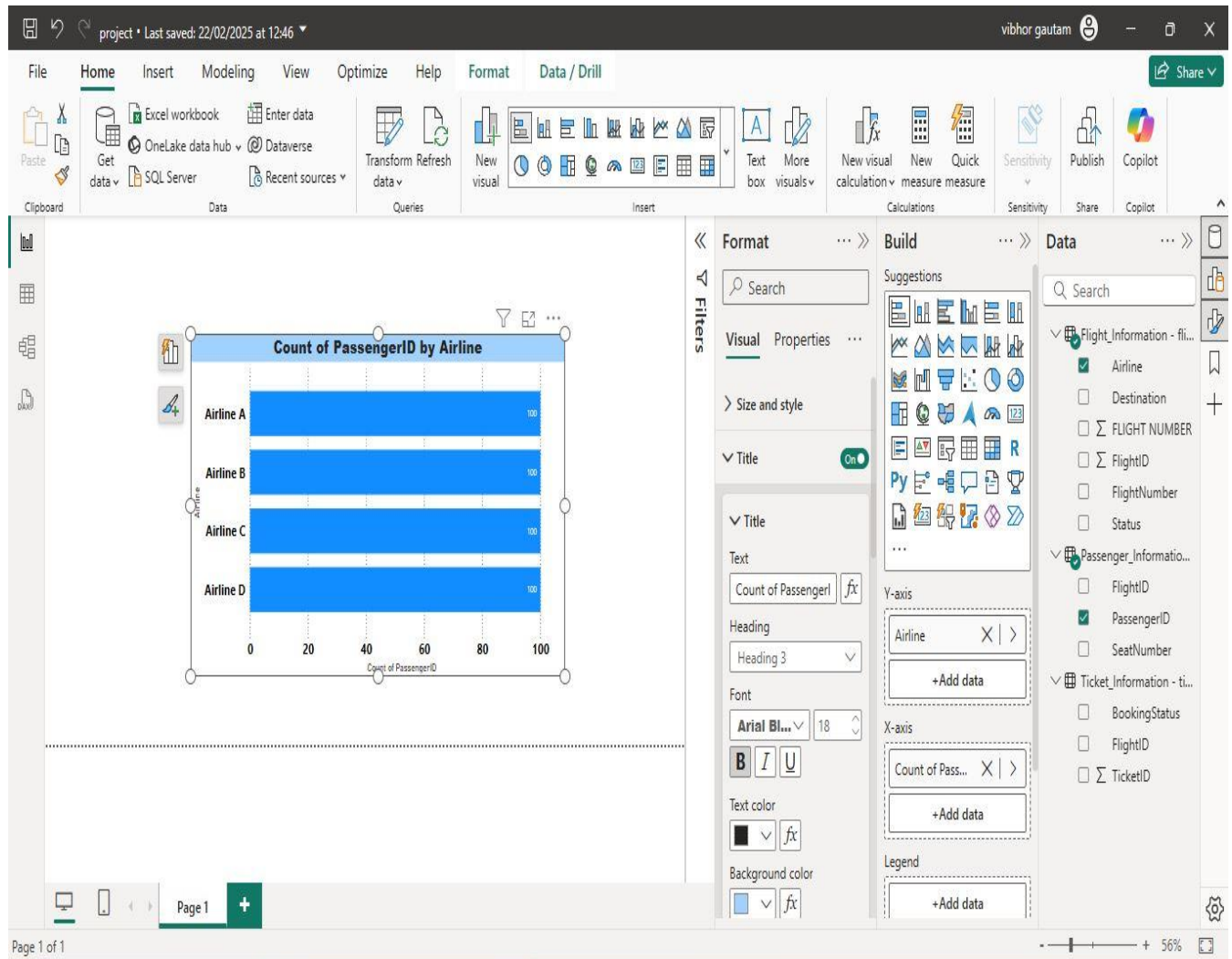
FlightID	FlightNumber	Airline	Destination	FLIGHT NUMBER
1	1001	FL1102	Airline D	Houston
2	1002	FL1435	Airline B	Chicago
3	1003	FL1860	Airline A	New York
4	1004	FL1270	Airline C	Chicago
5	1005	FL1106	Airline C	New York
6	1006	FL1071	Airline A	Phoenix
7	1007	FL1700	Airline C	Los Angeles
8	1008	FL1020	Airline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

5 COLUMNS, 200 ROWS Column profiling based on top 1000 rows PREVIEW DOWNLOADED AT 11:37 AM

TASK-5.(A) Create visuals for.

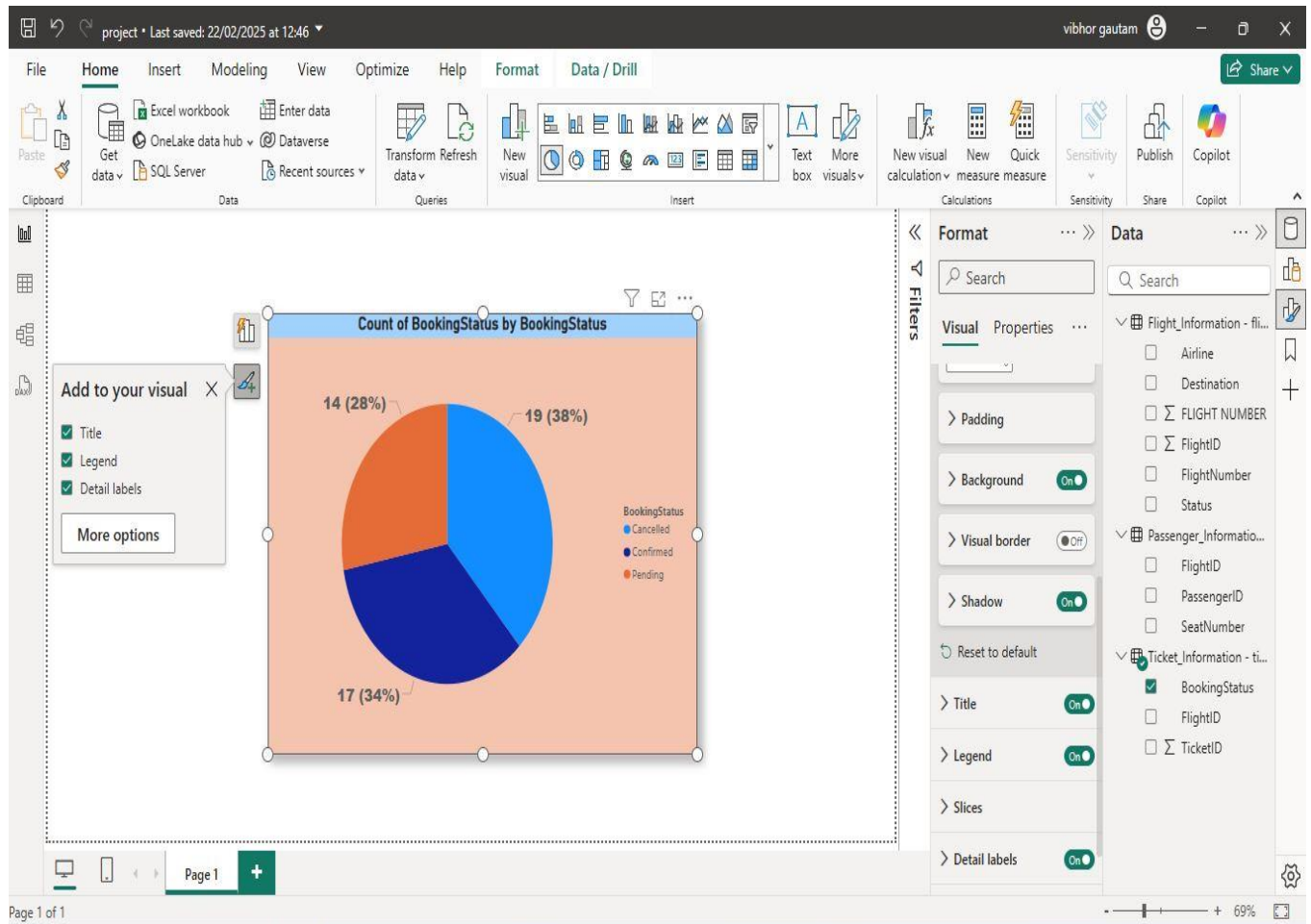
1. Passenger count by airline.

ANS.



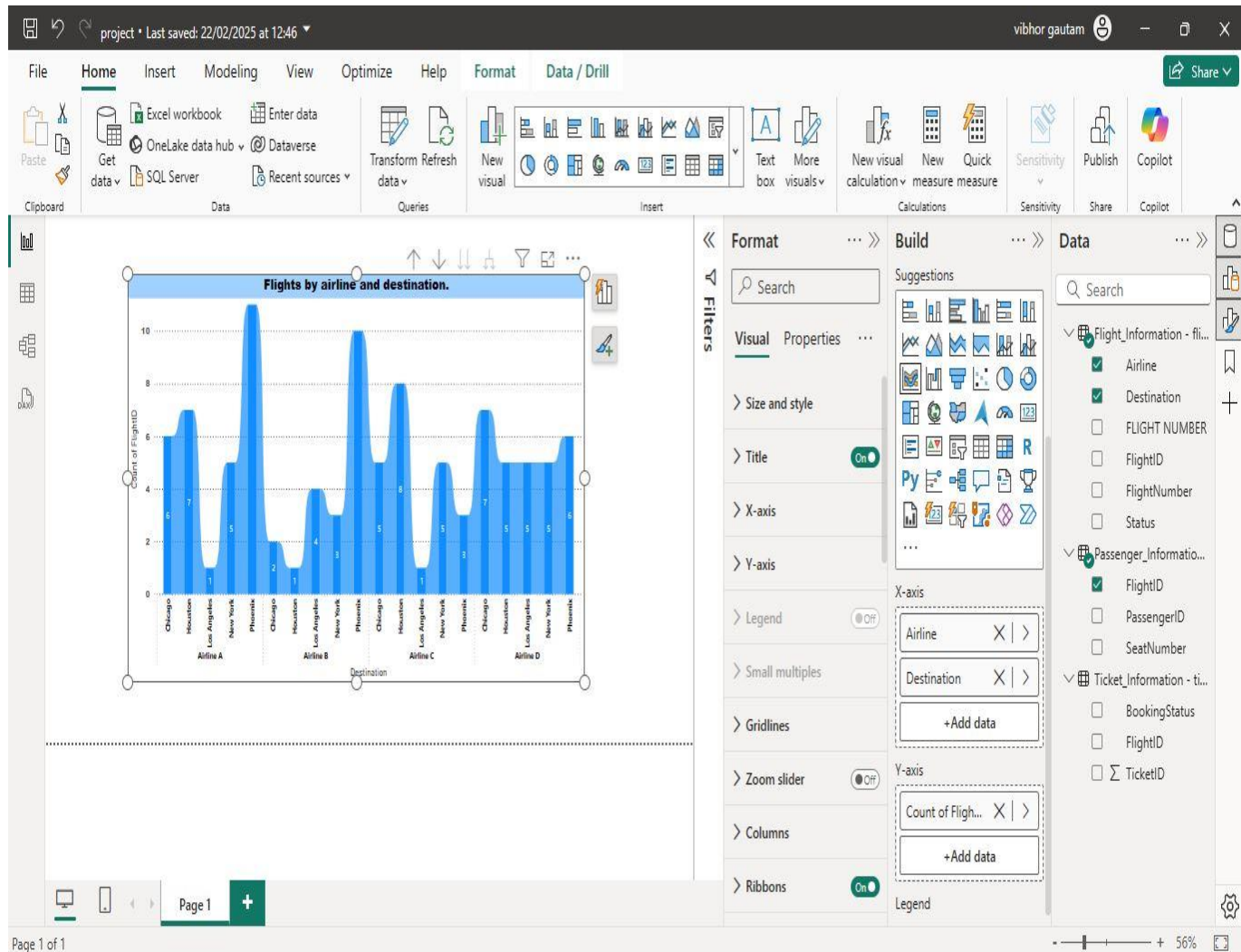
2. Ticket booking statuses.

ANS.



3. Flights by airline and destination.

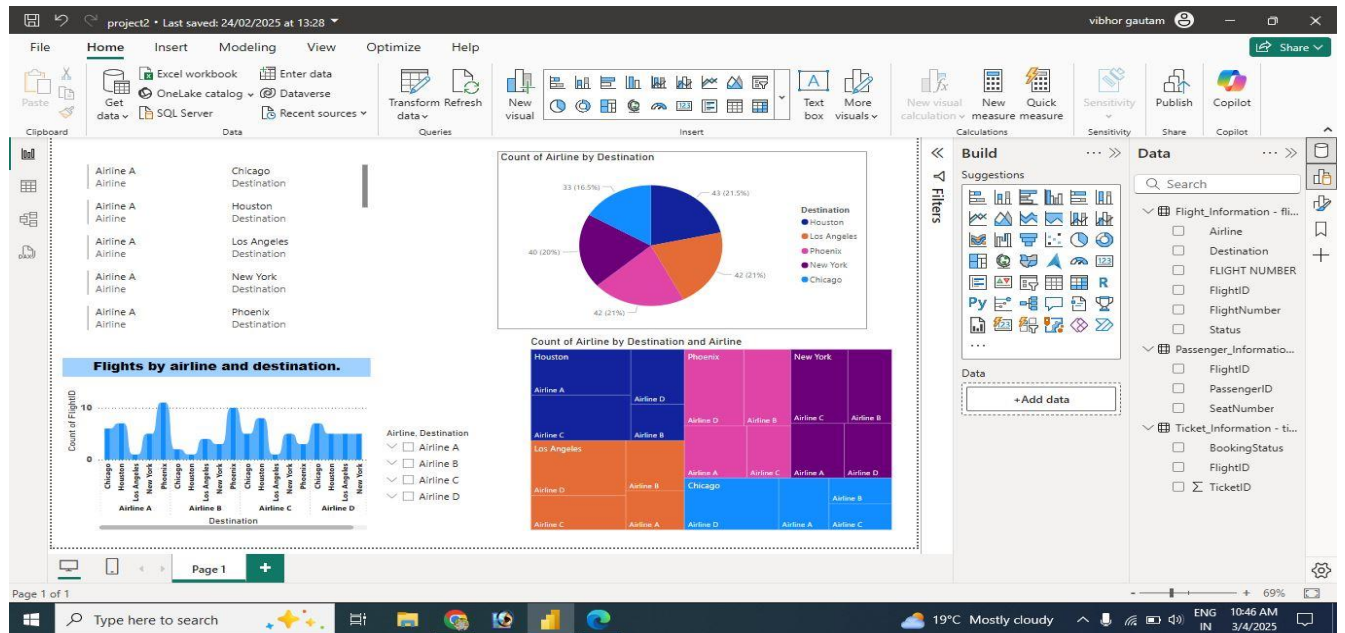
ANS.



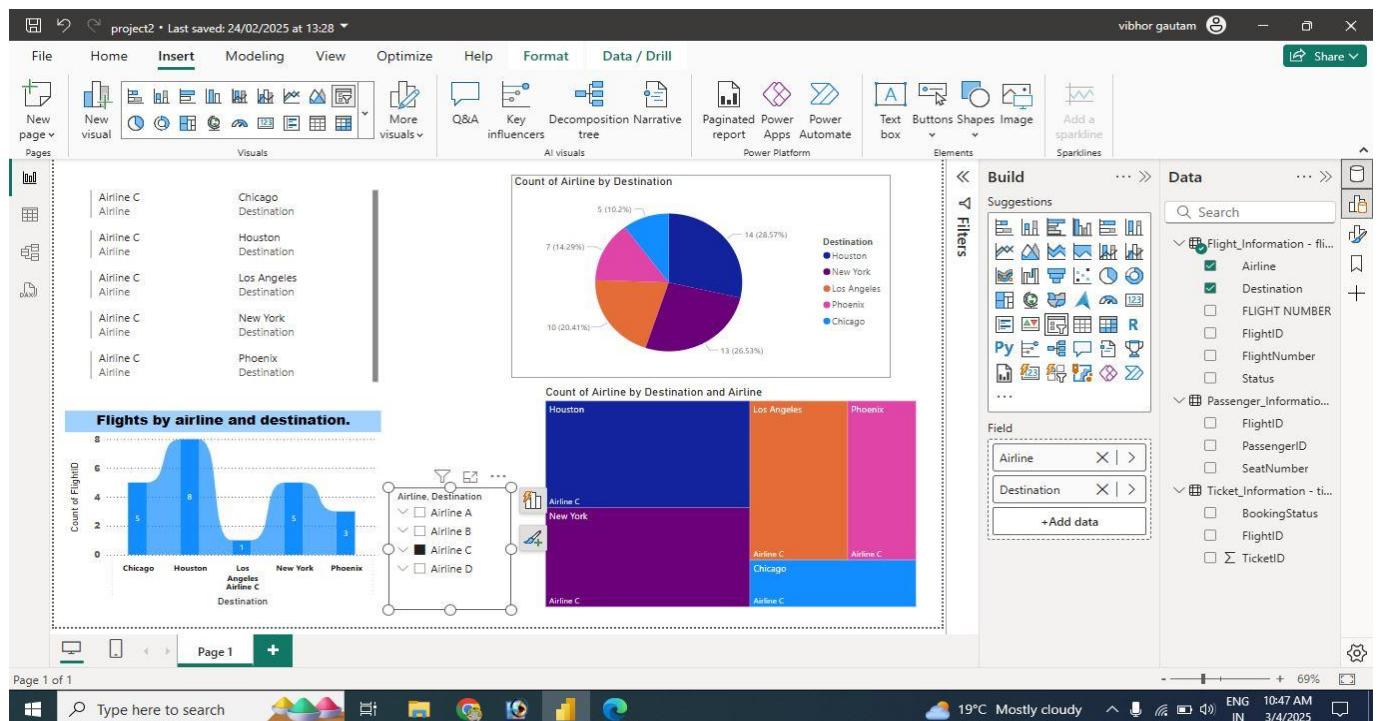
(B). Add interactive features for:

1. Destination and Airline.

ANS.

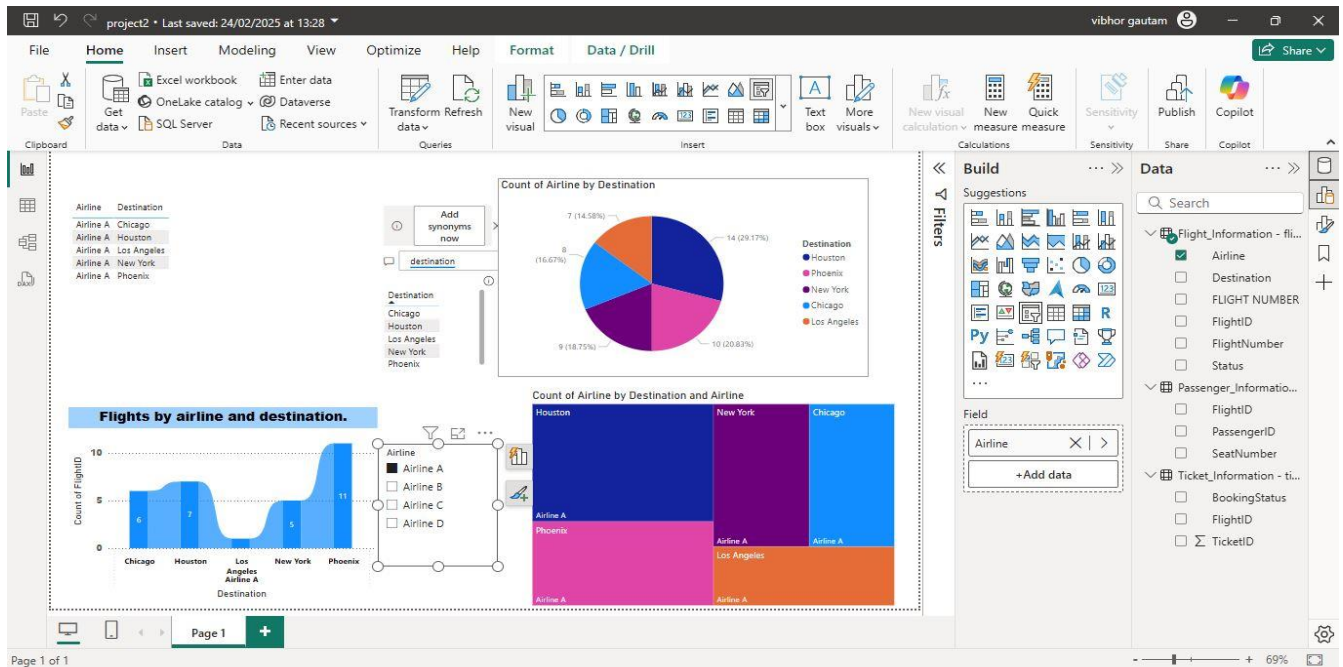


USE INTERACTIVE FEATURES.



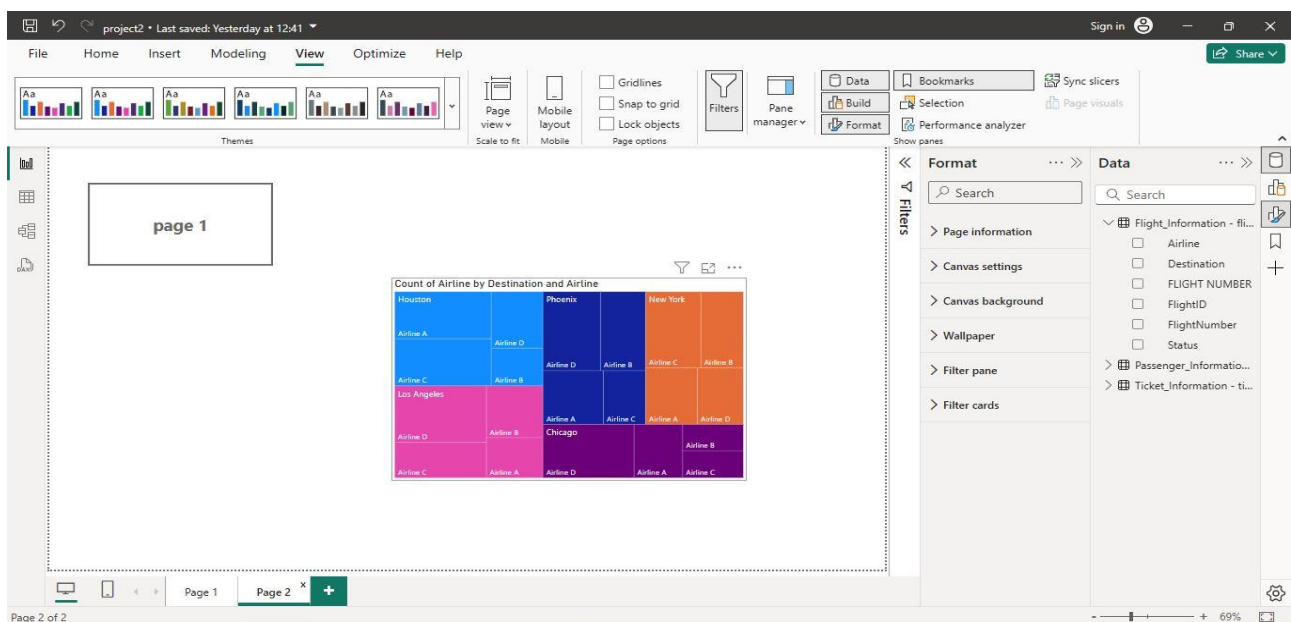
2. Quick views.

ANS. Quick views by airline a

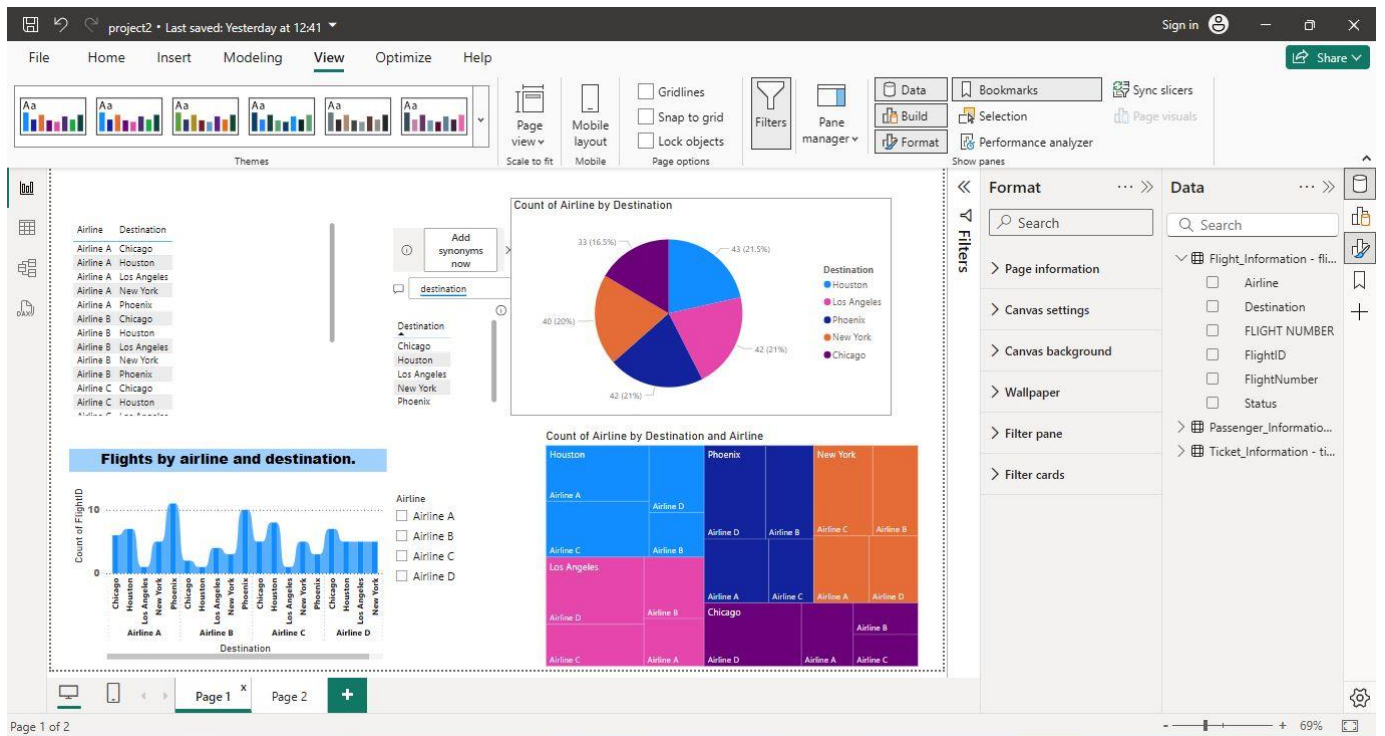


3. Airline-specific pages.

ANS. I am use a page 2 and click blank page button .and fill the text file name is PAGE 1 and go to the format button.



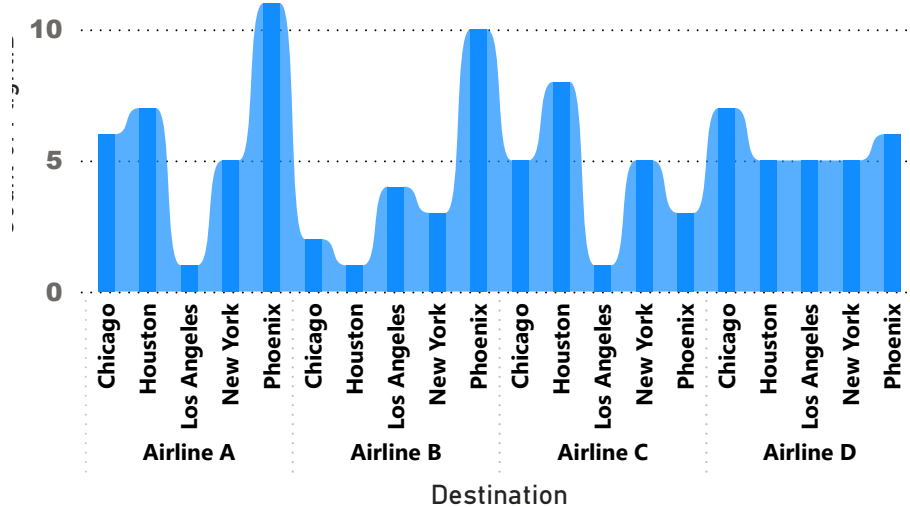
After.



Ask a question about your data

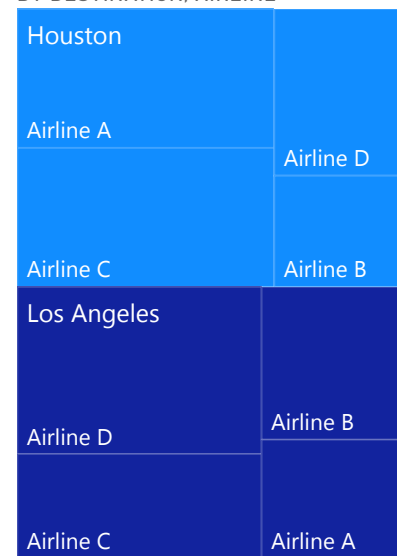
Airline C	Houston
Airline C	Los Angeles
Airline C	New York
Airline C	Phoenix
Airline D	Chicago
Airline D	Houston
Airline D	Los Angeles
Airline D	New York
Airline D	Phoenix

ights by airline and destination.



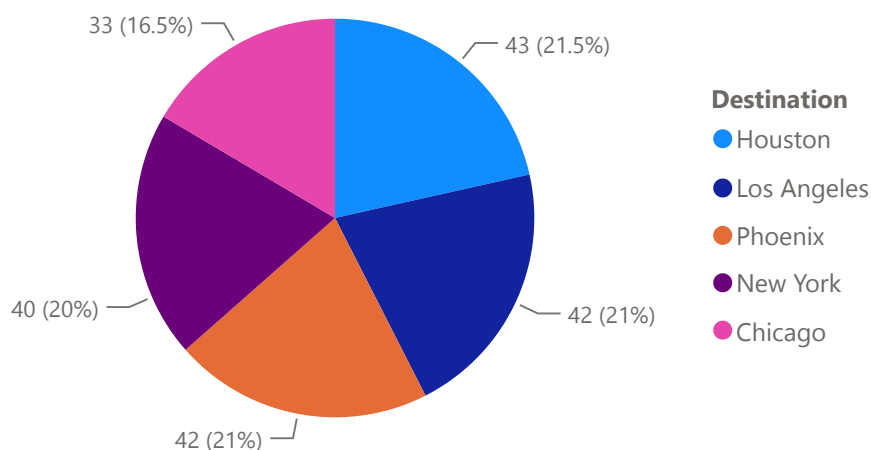
Count of Airline

BY DESTINATION, AIRLINE



ount of Airline

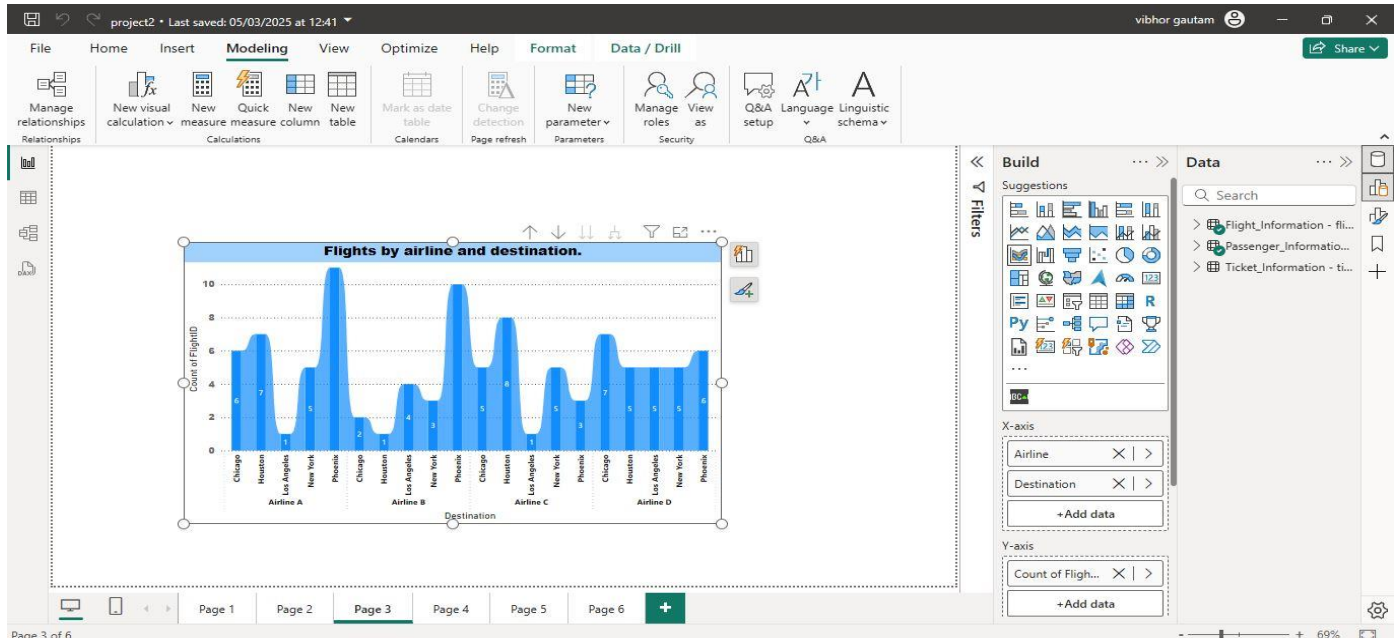
' DESTINATION



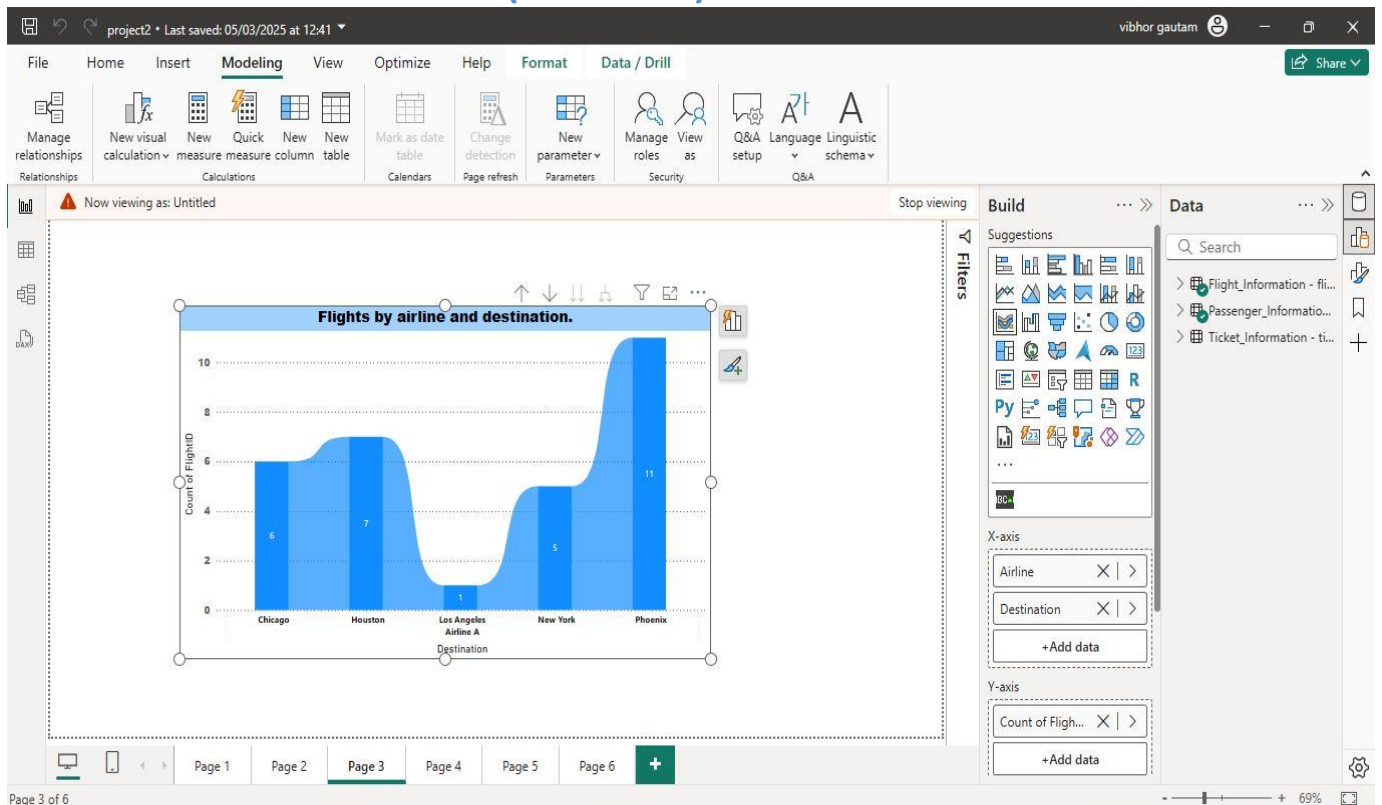
TASK.6 Final Dashboard and Power BI Service

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

ANS.

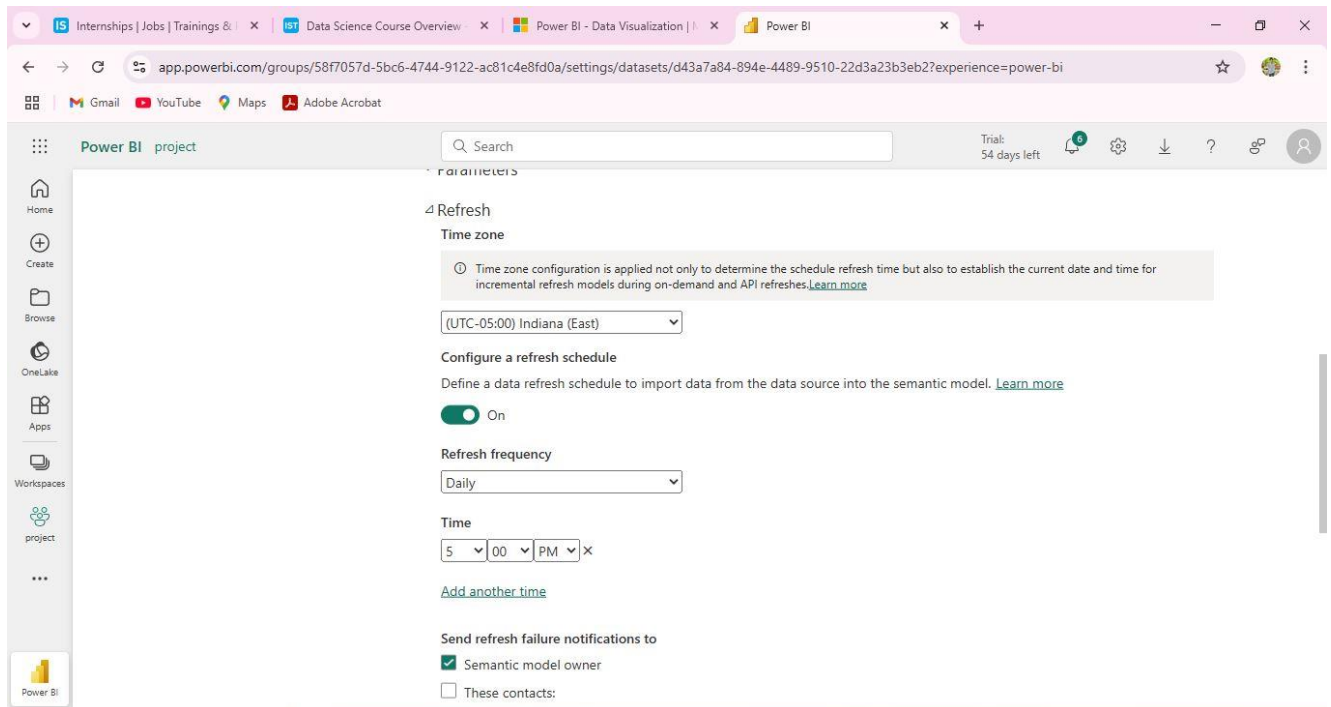


Use a ROW LEVEL SECURITY (AIRLINE A) .

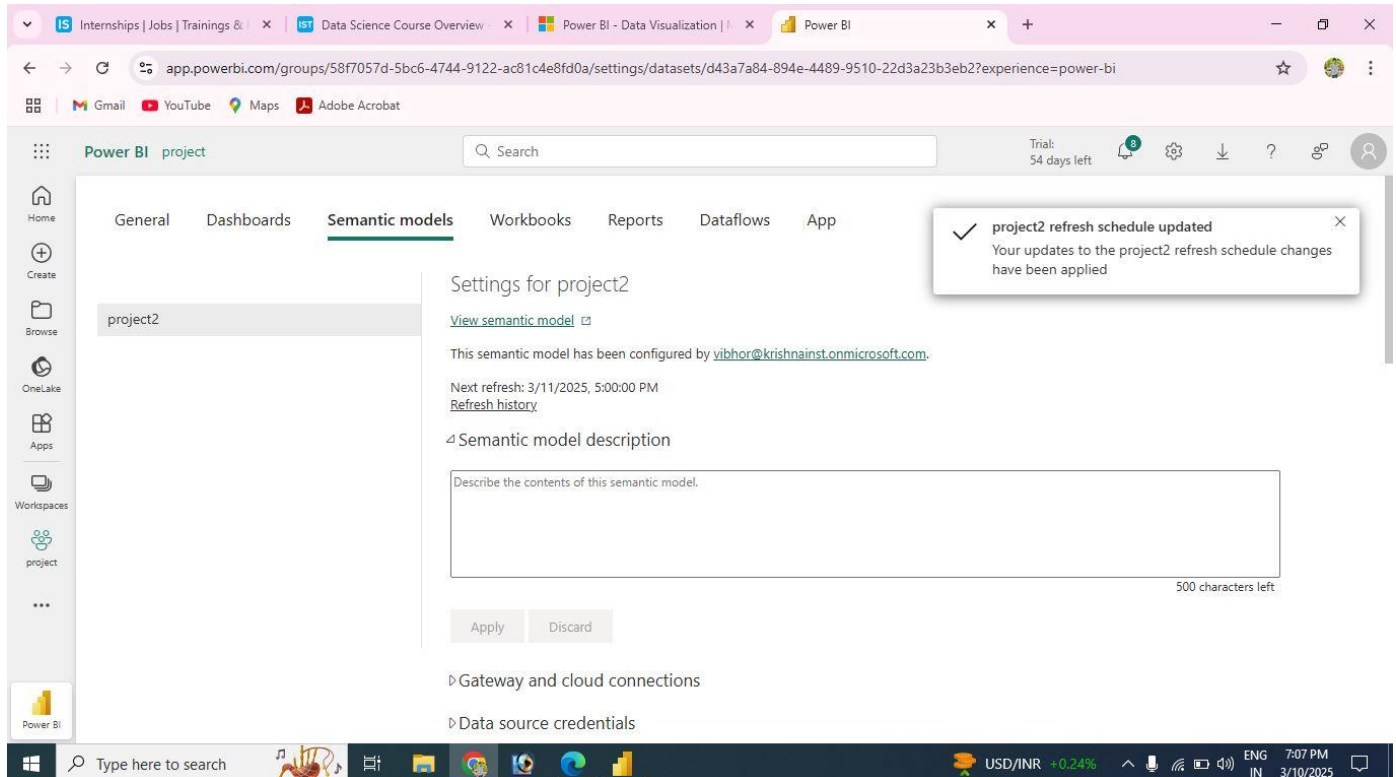


(3). Setup a schedule refresh at 5 PM daily.

ANS. PREVIOUS TIME

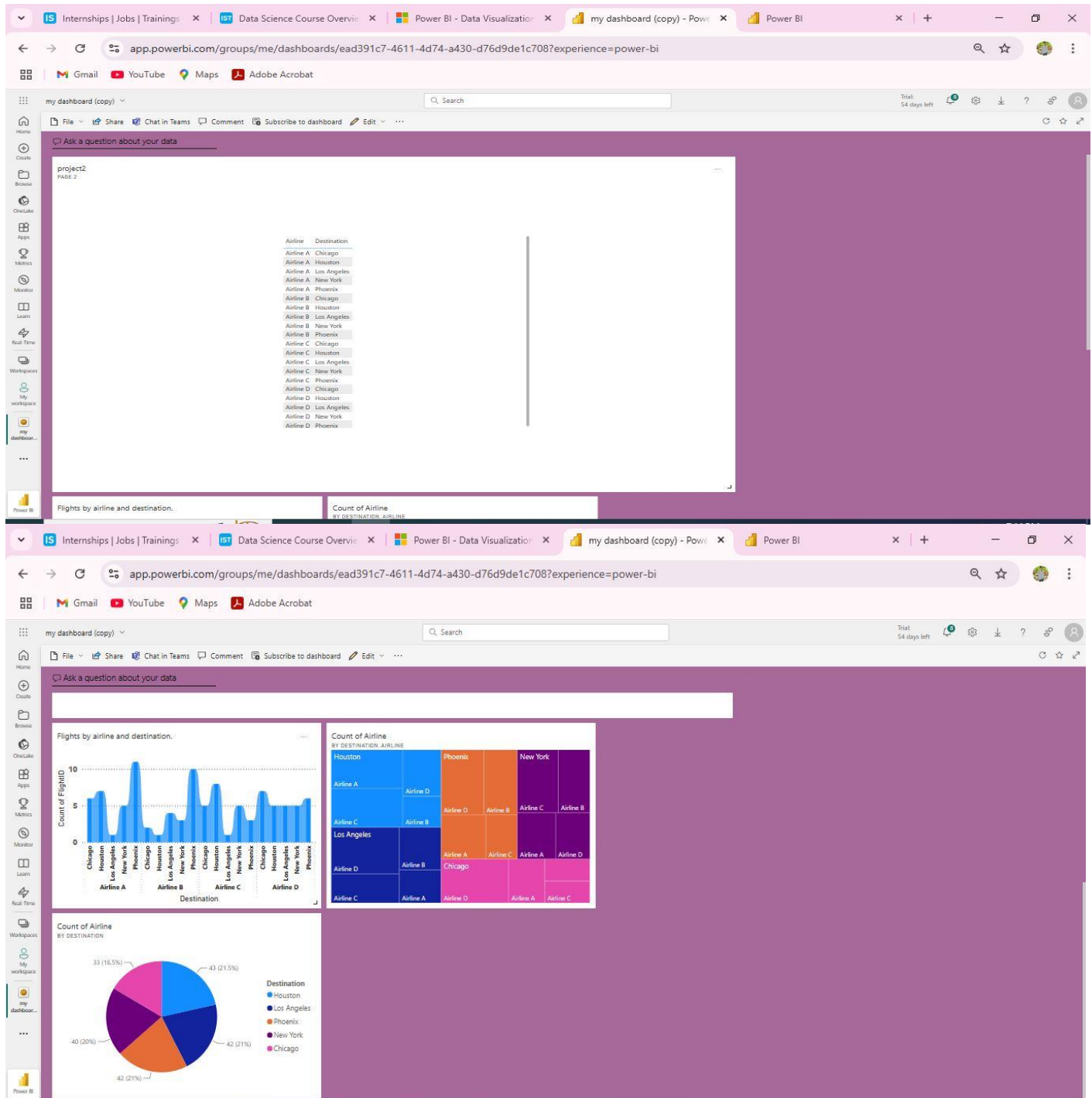


AFTER USE A SCHEDULE REFRESH TIME



(4). Deliverables: Screenshot of the published dashboard and RLS configuration

ANS 1



ANS. 2

project2 • Last saved: 05/03/2025 at 12:41

vibhor gautam

Share

Manage security roles

Create new security roles and use filters to define row-level data restrictions.

Roles

- + New
- Untitled

Select tables

- Flight_Informa...
- Passenger_Info...
- Ticket_Informa...

Filter data

+ New ☒ Select all ☐ Delete ☐ Group ☐ Ungroup

Show data if ☐ All of these rules are true

Column	Condition	Value
<input type="checkbox"/> Airline	In	AIRLINE A
Add +		

+ New

Switch to DAX editor

Save Close

Page 3 of 6

69%