

2. Ticket Information

Queries [3]

flight_information

ticket_information

passenger_information

Table.RemoveColumns(#"Changed Type",{"Column26", "Column25", "Column24", "Column23", "Column22", "Column21", "Column20", "Column19",

	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	1060	Pending
21	5021	1030	Confirmed
22	5022	1035	Confirmed
23	5023	1165	Confirmed
24	5024	1005	Confirmed
25	5025	1083	Cancelled
26	5026	1123	Cancelled
27	5027	1078	Confirmed
28	5028	1154	Pending

3 COLUMNS, 50 ROWS Column profiling based on top 1000 rows

Query Settings

PROPERTIES

Name

ticket_information

All Properties

APPLIED STEPS

Source

Navigation

Promoted Headers

Changed Type

Removed Columns

PREVIEW DOWNLOADED AT 13:02

3. Passenger Information

Queries [3]

flight_information

ticket_information

passenger_information

Table.TransformColumnTypes(#"Removed Columns",{{"PassengerID", Int64.Type}, {"FlightID", Int64.Type}})

	PassengerID	FlightID	SeatNumber
1	1	1161	38A
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	22E
25	25	1085	6A
26	26	1026	5A
27	27	1063	12B
28	28	1086	46B

3 COLUMNS, 100 ROWS Column profiling based on top 1000 rows

Query Settings

PROPERTIES

Name

passenger_information

All Properties

APPLIED STEPS

Source

Navigation

Promoted Headers

Changed Type

Removed Columns

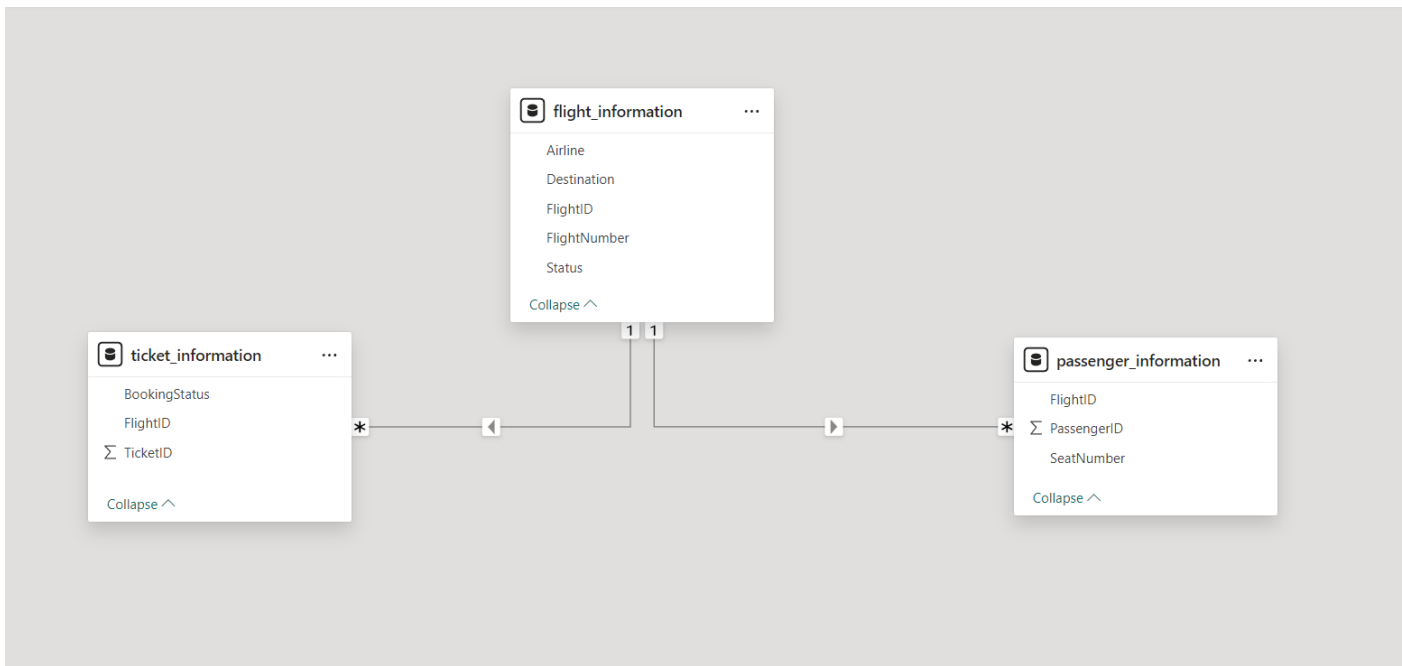
Changed Type1

PREVIEW DOWNLOADED AT 13:03

TASK 2: - DATA MODELING.

STEP 1: After cleaning the data, I click on 'close & apply' button and load the datasets in POWER BI DESKTOP.

STEP 2: Then I go to 'Model View' to create relationships between these tables and choose cardinality "Many to One".



TASK 3: - ENHANCED DATA INSIGHTS.

3.1 – ADD CONDITIONAL COLUMN.

STEP 1: To Add Conditional Column, I select ‘Status’ column from “flight information” table in power query editor.

STEP 2: After selecting the column, I go to transform tab and click on ‘conditional column’, then I put the values in it for On time flights I choose ‘Best’, for delayed flights I choose ‘To be improved’ and for cancelled flights I choose ‘Worse’.

Queries [3]

flight_information

ticket_information

passenger_information

fx

Table.RenameColumns(#"Added Conditional Column",{{"Custom", "Flight Remarks"}})

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

Query Settings

PROPERTIES

Name

flight_information

All Properties

APPLIED STEPS

Source

Navigation

Promoted Headers

Changed Type

Removed Columns

Added Conditional Column

Renamed Columns

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

PREVIEW DOWNLOADED AT 13:02

3.2 – COLUMN FROM EXAMPLE.

STEP 1: First, I go to “flight information” table and select ‘FlightNumber’ column.

STEP 2: Then I go to transform tab and go to column from examples and click on down arrow, then I click on ‘from selection’.

STEP 3: Then I extract the number from ‘FlightNumber’ column and name it as ‘Number_Flights’ column.

Queries [3] < 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
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
24	1024	FL1343	Airline B	Chicago	Delayed	1343

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

PREVIEW DOWNLOADED AT 13:02

Queries [3] < X ✓ f_x = Table.TransformColumnTypes(*Reordered Columns*,{{"Number_Flights", Int64.Type}})

flight_information ticket_information passenger_information

	FlightNumber	Number_Flights	Airline	Destination	Status	Flight Remarks
1	1001 FL1102	1102	Airline D	Houston	On Time	Best
2	1002 FL1435	1435	Airline B	Chicago	On Time	Best
3	1003 FL1860	1860	Airline A	New York	Cancelled	Worse
4	1004 FL1270	1270	Airline C	Chicago	Delayed	To Be Improved
5	1005 FL1106	1106	Airline C	New York	Delayed	To Be Improved
6	1006 FL1071	1071	Airline A	Phoenix	On Time	Best
7	1007 FL1700	1700	Airline C	Los Angeles	Cancelled	Worse
8	1008 FL1020	1020	Airline C	Los Angeles	Delayed	To Be Improved
9	1009 FL1614	1614	Airline A	Los Angeles	Cancelled	Worse
10	1010 FL1121	1121	Airline D	Chicago	Cancelled	Worse
11	1011 FL1466	1466	Airline A	Phoenix	On Time	Best
12	1012 FL1214	1214	Airline D	New York	Delayed	To Be Improved
13	1013 FL1330	1330	Airline C	Houston	On Time	Best
14	1014 FL1458	1458	Airline C	New York	Delayed	To Be Improved
15	1015 FL1087	1087	Airline C	Houston	Delayed	To Be Improved
16	1016 FL1372	1372	Airline B	New York	Delayed	To Be Improved
17	1017 FL1099	1099	Airline D	Phoenix	Delayed	To Be Improved
18	1018 FL1871	1871	Airline B	Houston	Delayed	To Be Improved
19	1019 FL1663	1663	Airline B	Chicago	Cancelled	Worse
20	1020 FL1130	1130	Airline A	New York	On Time	Best
21	1021 FL1661	1661	Airline B	New York	Cancelled	Worse
22	1022 FL1308	1308	Airline A	Houston	Delayed	To Be Improved
23	1023 FL1769	1769	Airline A	Chicago	On Time	Best
24	1024 FL1343	1343	Airline B	Chicago	Delayed	To Be Improved
25	1025 FL1491	1491	Airline D	Phoenix	On Time	Best
26	1026 FL1413	1413	Airline D	Chicago	Cancelled	Worse
27	1027 FL1805	1805	Airline D	Chicago	On Time	Best

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

PREVIEW DOWNLOADED AT 13:02

TASK 4: - CALCULATION USING DAX.

4.1: - CALCULATE TOTAL PASSENGERS FOR SPECIFIC FLIGHT.

STEP 1: First, I go to Home Tab and click on 'New Measure' and create a DAX formula.

STEP 2: The DAX formula which I create is specific flight for passengers =

`CALCULATE(COUNT(passenger_information[PassengerID]),flight_information[FlightNumber]="FL1686")`

STEP 3: After creating DAX formula, I used single-row card visual from visualization pane and put the column 'Specific flight for passengers' in field pane.

1 Specific flight for passengers = `CALCULATE(COUNT(passenger_information[PassengerID]),flight_information[FlightNumber]="FL1686")`



4.2: - CALCULATE TOTAL TICKETS BOOKED.

STEP 1: I add another page and go to Home tab and click on 'New Measure' and create a DAX formula.

STEP 2: The DAX formula is "Total_tickets_booked = `COUNT(ticket_information[TicketID])`"

STEP 3: After creating the DAX formula, I take single-row card visual from visualization pane and put 'Total_tickets_booked' column in field pane.

1 `Total_tickets_booked = COUNT(ticket_information[TicketID])`

50

Total_tickets_booked

4.3: - FILTER TABLE SHOWING 'BEST' FLIGHTS ONLY.

STEP 1: For creating filter table, I go to Table view and then I go to Home Tab > click on 'New Table'.

STEP 2: After creating new table I have to create the DAX formula which is Best_Flights = `FILTER(flight_information,flight_information[Flight Remarks]="Best")`

STEP 3: And after creating the DAX formula I click on ENTER and my new filter table created which has data only according to 'Best' flights.

```
1 Best_Flights = FILTER(flight_information,flight_information[Flight Remarks]="Best")
```

The screenshot displays the Microsoft Power BI interface. At the top, the formula bar shows the DAX formula: `1 Best_Flights = FILTER(flight_information,flight_information[Flight Remarks]="Best")`. Below the formula bar, a table view shows the resulting data. The table has columns: FlightID, FlightNumber, Airline, Destination, Status, Number_Flights, and Flight Remarks. The data is filtered to show only flights with the remark 'Best'. On the right side, the 'Data' pane shows the 'Best_Flights' table selected, with its columns listed: Airline, Destination, Flight Remarks, FlightID, FlightNumber, Number_Flights, Specific flight for passengers, Status, Total_tickets_booked, passenger_information, and ticket_information. The status bar at the bottom indicates 'Table: Best Flights (82 rows)'.

FlightID	FlightNumber	Airline	Destination	Status	Number_Flights	Flight Remarks
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
1053	FL1875	Airline C	Chicago	On Time	1875	Best
1055	FL1243	Airline B	New York	On Time	1243	Best
1057	FL1504	Airline A	Phoenix	On Time	1504	Best
1060	FL1818	Airline D	Chicago	On Time	1818	Best
1061	FL1646	Airline D	Los Angeles	On Time	1646	Best

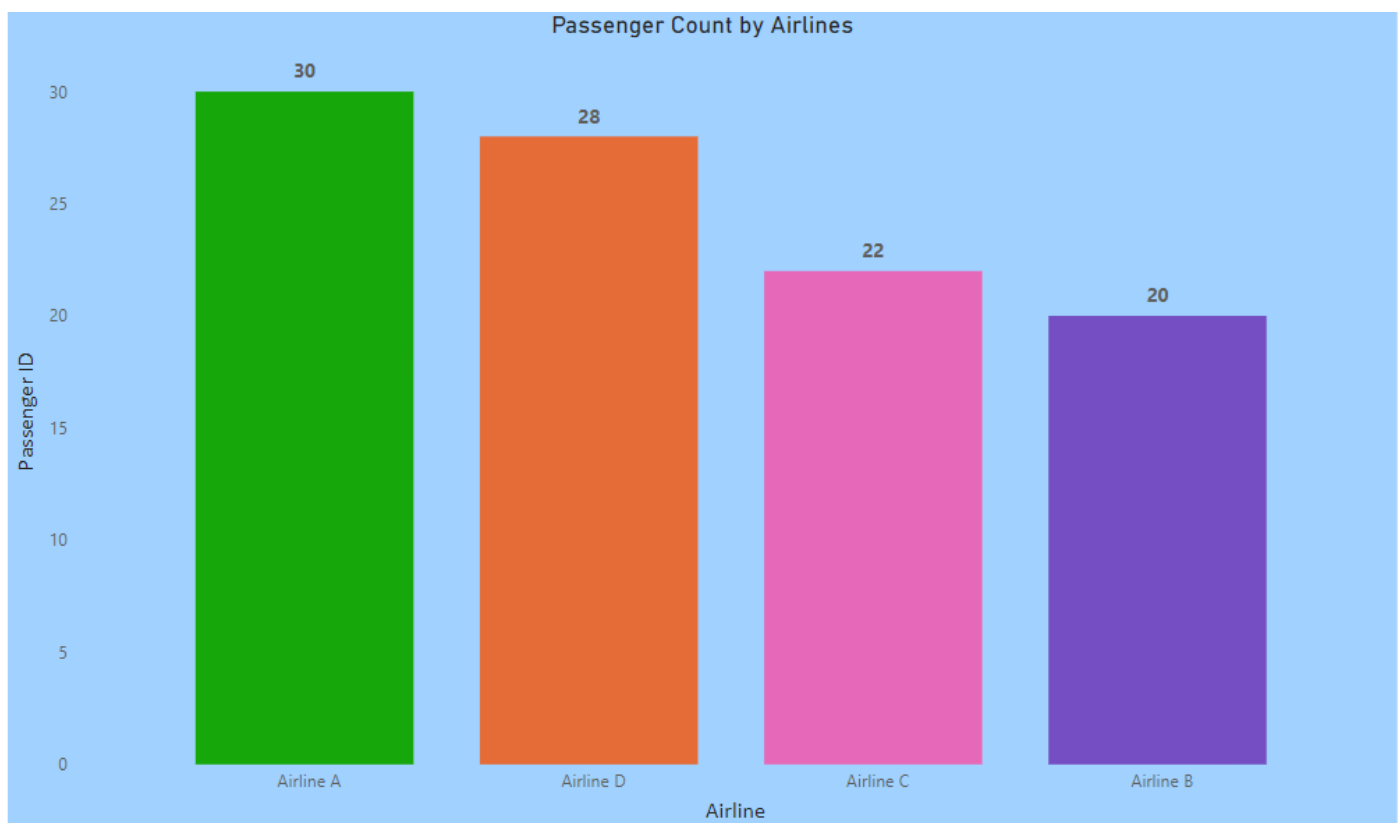
TASK 5: - VISUALIZATION AND INTERACTIVE FEATURES.

5.1.1: - CREATE VISUAL FOR PASSENGER COUNT BY AIRLINE.

STEP 1: First, I take a column chart visual from visualization pane.

STEP 2: Then I put 'Airline' column in X-axis section and 'PassengerID' column in Y-axis section, then I used count function in 'PassengerID' column.

STEP 3: After that I used format pane and change columns color differently, also I show data labels on each column. And I changed the background color too.

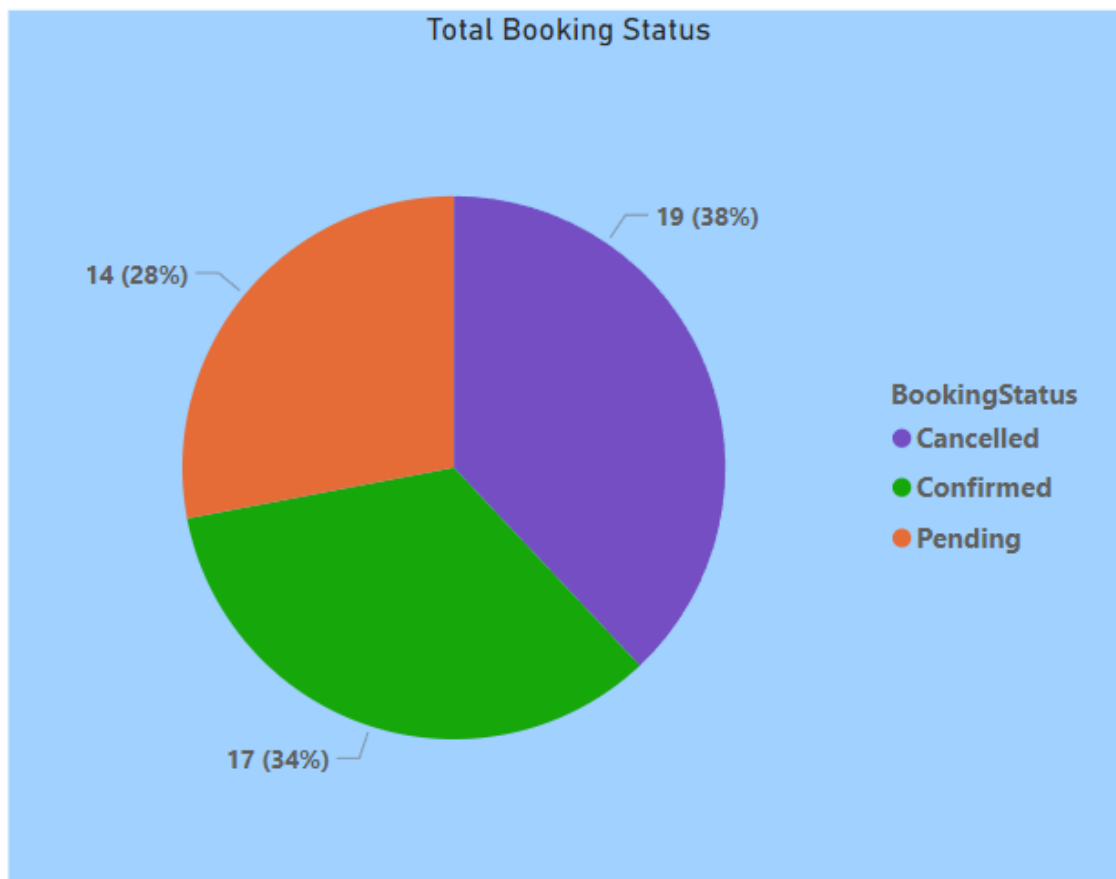


5.1.2: - CREATE VISUAL FOR TICKET BOOKING STATUS.

STEP 1: First, I take pie chart visual from visualization pane.

STEP 2: Then I put 'BookingStatus' column in Legend section and 'TicketID' in Values section, then I used the count function in 'TicketID' column.

STEP 3: Then I go to format pane and changes the background colour of the visual and changes the slices colour to make it attractive.



5.1.3: - CREATE VISUAL FOR FLIGHTS BY AIRLINE AND DESTINATION.

STEP 1: First, I take Matrix visual from visualization pane.

STEP 2: Then I put 'Destination' column in Rows section, 'Airline' column in Columns section and 'FlightID' column in Values section, then I used count function in 'FlightID' column.

STEP 3: After that I changed the background colour of the visual, changed the layout and style to minimal and make the column header to 'bold'.

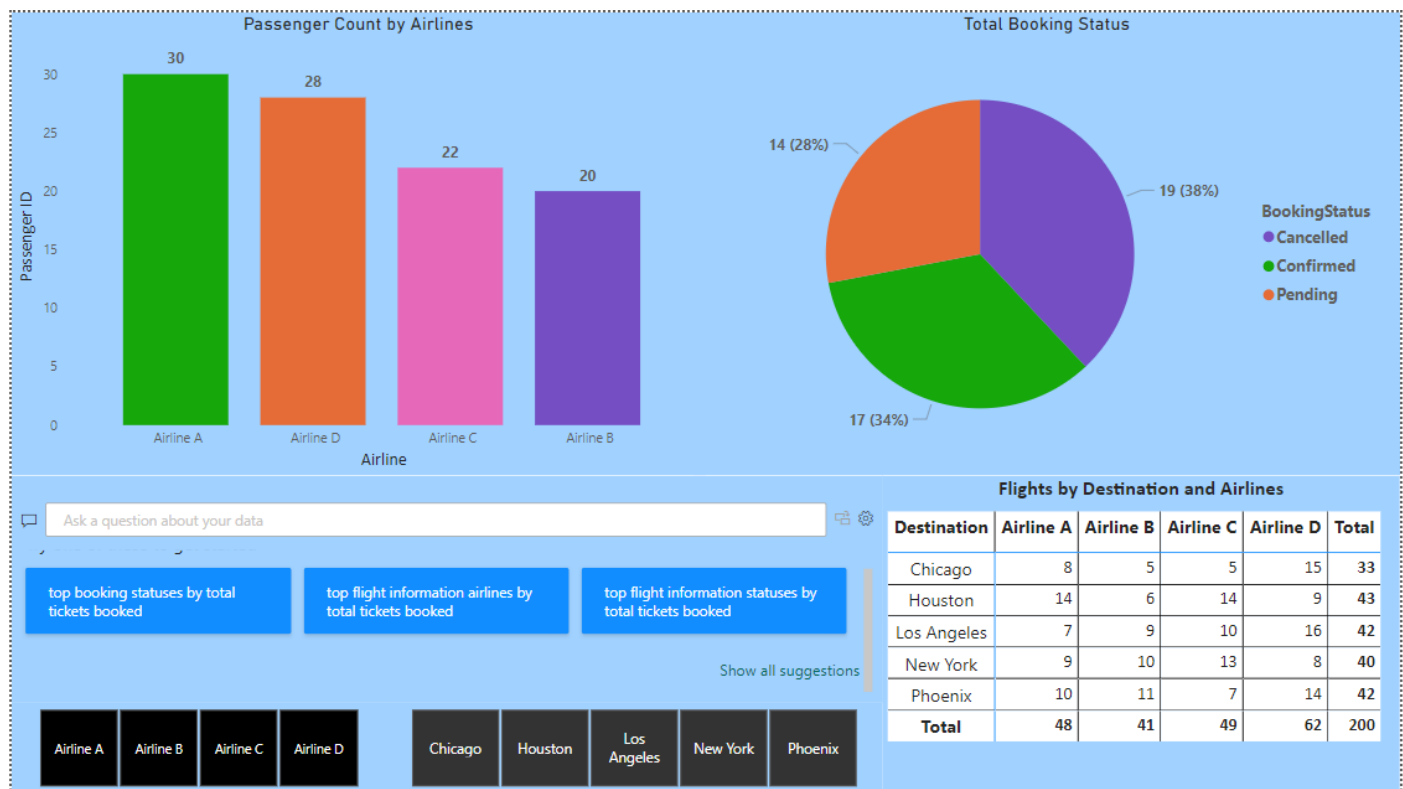
Flights by Destination and Airlines					
Destination	Airline A	Airline B	Airline C	Airline D	Total
Chicago	8	5	5	15	33
Houston	14	6	14	9	43
Los Angeles	7	9	10	16	42
New York	9	10	13	8	40
Phoenix	10	11	7	14	42
Total	48	41	49	62	200

5.2.1: - ADD INTERACTIVE FEATURES FOR DESTINATION AND AIRLINE & QUICK VIEWS.

STEP 1: First, I added all the visuals in a same page.

STEP 2: Then I added slicer from visualization pane for both Destination and Airline.

STEP 3: For quick view, I added q&a feature from visualization pane to search anything related to the visuals or given dataset.



5.2.2: - ADD INTERACTIVE FEATURES FOR AIRLINE-SPECIFIC PAGE.

STEP 1: For this I take a new page and add some visuals regarding to Airline data like a donut chart showing 'FlightID by Airline', a column chart showing 'PassengerID by Airline', a table and line chart visual showing 'Airline, Destination and count of FlightID' and a bar chart showing 'TicketID by Airline and Destination'.

STEP 2: After taking this, I go to visualization pane and below there is an option "Drill Through" in which I put the 'Airline' column and then a 'back' button is created automatically.

STEP 3: Then I go to previous page and right click on 'Airline B' in bar chart visual and go to Drill Through.

STEP 4: After clicking the Drill Through option, I get the data related to only 'Airline B'.



TASK 6: - FINAL DASHBOARD AND POWER BI SERVICE.

6.1: - DESIGN A COMPREHENSIVE DASHBOARD WITH KEY VISUALS AND INSIGHTS.

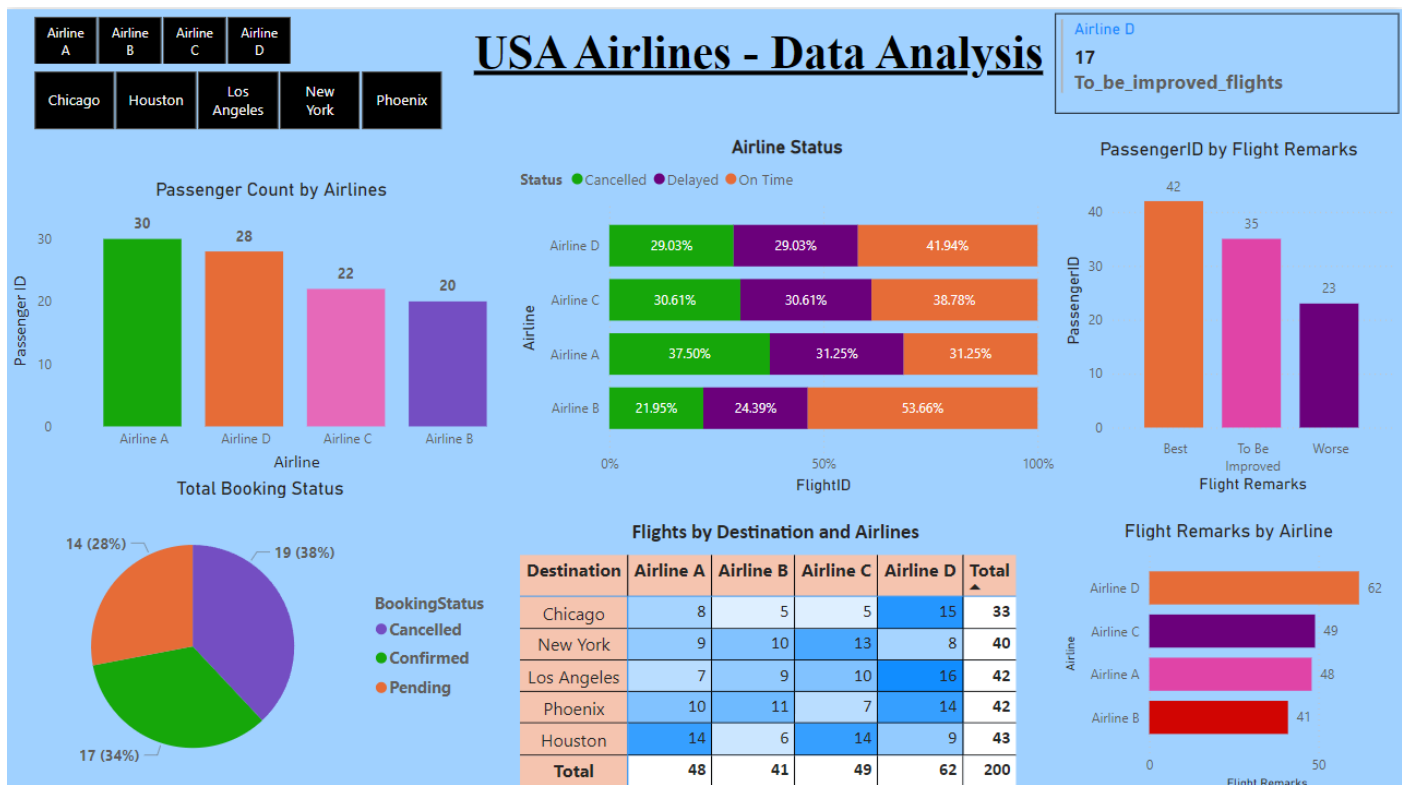
STEP 1: First, I take all the visuals from different pages into one page and added a few more like I take 100% stacked bar chart visual from visualization pane and put 'Airline' column in Y-axis, 'FlightID' column in X-axis and 'Status' column in Legend section.

STEP 2: Also, I take a column chart and put 'Flight Remarks' column in X-axis and 'PassengerID' in Y-axis section. Last, I added bar chart from visualization pane and put 'Airline' column in Y-axis and 'Flight Remarks' column in X-axis section.

STEP 3: Then I added a multi-row card but for that first I created a DAX formula which is

To_be_improved_flights =

`CALCULATE(DISTINCTCOUNT(flight_information[FlightNumber]),FILTER(flight_information,flight_information[Airline]="Airline D"&&flight_information[Flight Remarks]="To be improved"))`, then I take this created column named 'To_be_improved_flights' in fields section and 'Airline' column also in fields section.



6.2: - CONFIGURE ROW-LEVEL SECURITY(RLS) FOR AIRLINE A DATA AND ASSIGN IT TO A USER.

STEP 1: To configure RLS, first I go to Modelling tab > Go to manage roles.

STEP 2: After clicking on manage roles, select the role first and rename it to 'Airline A', then select table "Flight_information" in which the 'Airline' column is and then filter data by selecting the column, give the condition "equals" and value "Airline A" and save it.

STEP 3: Then to check the role, I go to 'modelling tab' again and go to 'view as' and choose the option 'Airline A', then my whole report changes to 'Airline A' data.

STEP 4: Now I have to assign a role in Power BI service, before that I created a workspace in Power BI service and name it as "Airline Data".

STEP 5: Then I published my report in my workspace and then I go to 'semantic model' and right click on it and select 'Security'. After that I assign a role in RLS and put email id of that user which I assigned it to.

Manage security roles



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

✓ Successfully applied role changes.



Roles

+ New

Airline A

Select tables

Best_Flights ...

flight_informat...

passenger_inf... ..

ticket_informat... ..

Filter data

Switch to DAX editor

+ New ☒ Select all Delete Group Ungroup

Show data if All of these rules are true

Column

Condition

Value



Airline

Equals

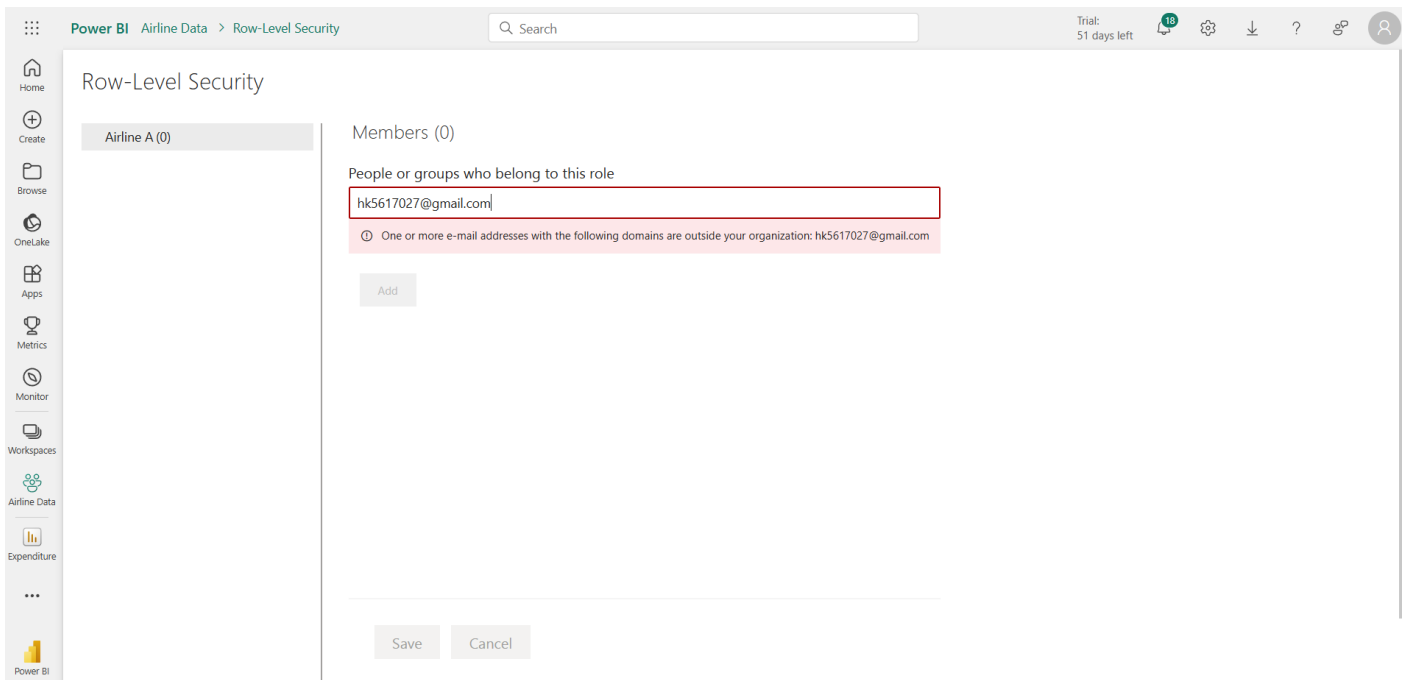
Airline A

+ New

Save

Close





6.3: - SET UP A SCHEDULE REFRESH AT 5PM DAILY.

STEP 1: First, I open workspace in Power Bi service and go to Power BI setting.

STEP 2: After opening the power Bi setting > go to semantic model and go to “Refresh”.

STEP 3: There I change the time zone to New Delhi, turn on the refresh schedule button, change the refresh frequency to “Daily” and change the time to 5P.M and click on apply.

Parameters

Refresh

Time zone

① 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

5 00 PM

[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/1szSEKDMB5rXL0HVZujRWcFkILVZDQ5K4/view?usp=sharing>

