



Sri Lanka Institute of Information Technology
IT3021 Data Warehousing and Business Intelligence

Assignment 2

Submitted to Sri Lanka Institute of Information Technology

Rangana PWM

IT19987880

Contents

1	Data source for the assignment 2	3
1.1	Description of the dataset	3
1.1.1	EER diagram for aircraft tracking dataset.	3
1.1.2	Diagram for Data Sources Patriating.....	4
1.1.3	Solution architecture	5
1.2	Description of the data source.....	5
2	SSAS Cube Implementation	6
2.1	Create an Analysis Services project.	6
2.2	Create new data source	7
2.3	Create new data source View.	8
2.4	Create a cube	10
2.5	Deploy the cube	14
2.6	Create a KPI.....	16
2.7	Browsing Cube Data.....	19
3	Demonstration of OLAP operations.....	20
3.1	PowerBI reports for OLAP operations.....	20
3.1.1	Roll-up	20
3.1.2	Drill Down.....	22
3.1.3	Slice operation	24
3.1.4	Dice operation.....	26
3.1.5	Pivot charts operation.....	27
4	SSRS Reports	28
4.1	Create a Data Source.	29
4.2	Create a Dataset.....	30
4.3	Report 1: Report with a matrix	32
4.4	Report 2: Parameterized Report.....	36
4.5	Report 3: SSRS drill-down report	46
4.6	Report 4: SSRS drill-through report	48
4.6.1	Level 1- Main report.....	48
4.6.2	Level 2- Sub report	50
4.6.3	Level 3- Main report Con.	54
4.6.4	Level 4- Sub report Con.....	56

1 Data source for the assignment 2

1.1 Description of the dataset

This dataset about flights of spy planes operated by the FBI and the Department of Homeland Security (DHS) which is the data comes from more than four months of plane tracking data provided by the website Flightradar24, plus the Federal Aviation Administration's aircraft registration database. The original source files can be found using the links provided below.

Data Set Link :- <https://github.com/BuzzFeedNews/2016-04-federal-surveillance-planes>

1.1.1 EER diagram for aircraft tracking dataset.

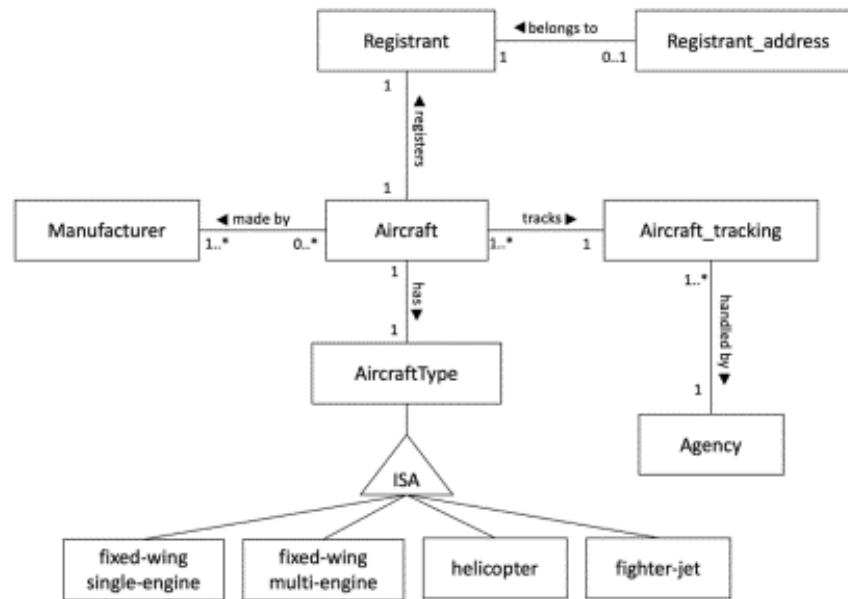


Figure 1 - EER diagram for aircraft tracking dataset

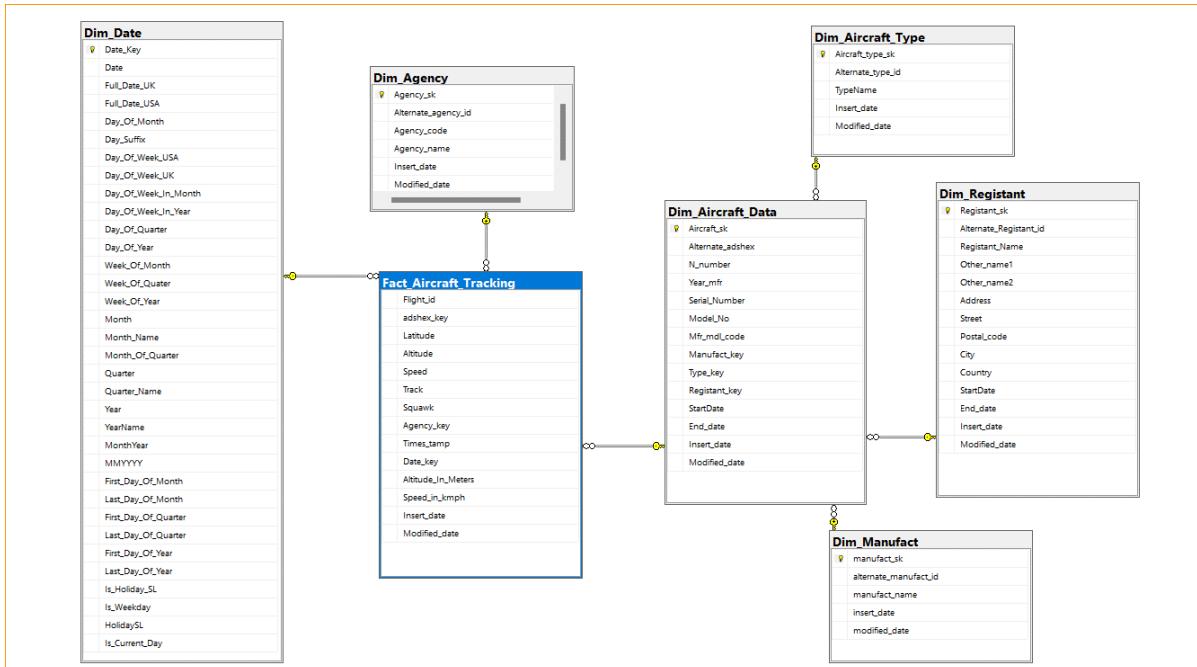


Figure 2 - Generated class diagram for the schema

1.1.2 Diagram for Data Sources Patriating

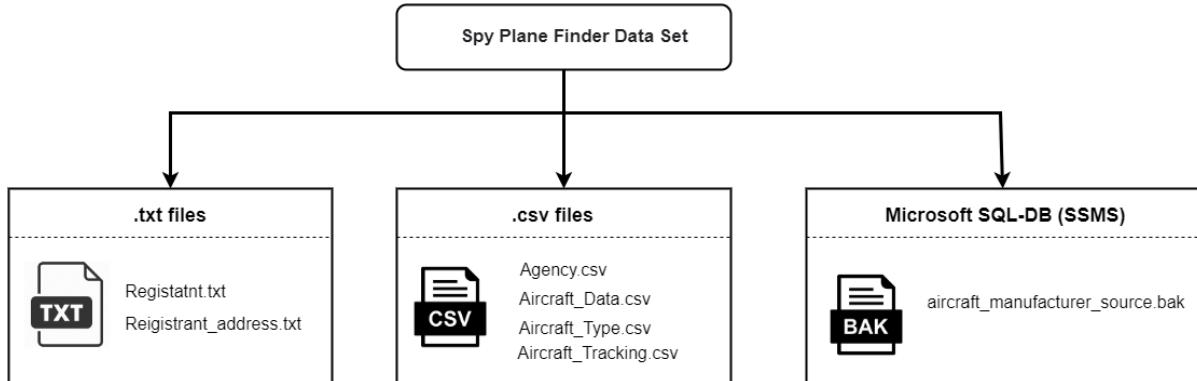


Figure 3 - Diagram for Data Source Patriating

1.1.3 Solution architecture

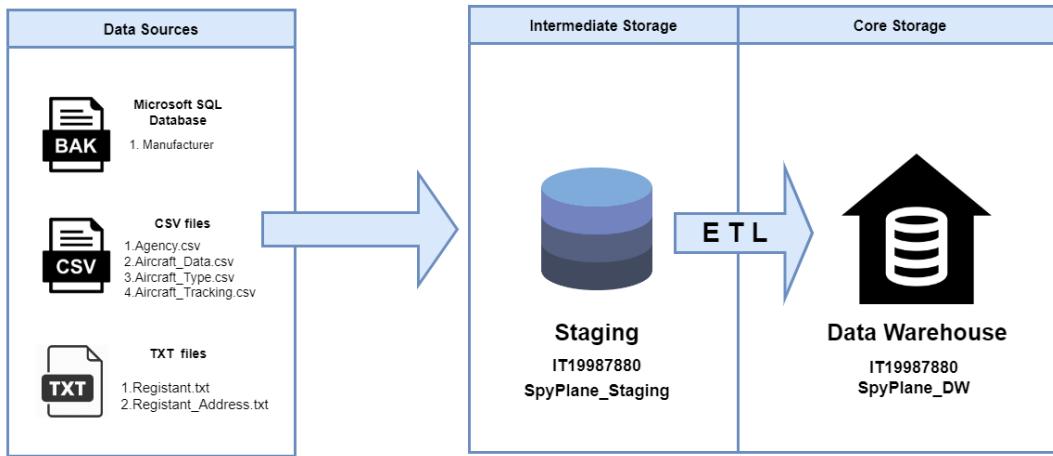


Figure 4 - Solution Architecture

1.2 Description of the data source

In this assignment, Firstly, we will create an OLAP cube using SQL Server Analysis Services; SSAS. For that, we have to use the data warehouse was created in 1st assignment. Before continuing with this assignment2, You can get the idea of previous SSIS assignment refer to brief description and diagrams given above. We can continue with the data which is available in the data warehouse layer now.

Tools Required:

- SQL Server Data Tools or Report Builder
- SQL Server Management Studio
- Microsoft Excel

2 SSAS Cube Implementation

Let's start using data warehouse as the data source and create an SSAS cube. In order to create required Cubes and related components, let's create an SSAS Project using Visual Studio Data Tools.

2.1 *Create an Analysis Services project.*

First, we have to open Visual Studio Data Tools in 'Administrator' mode. Then, we have to create an Analysis Services project.

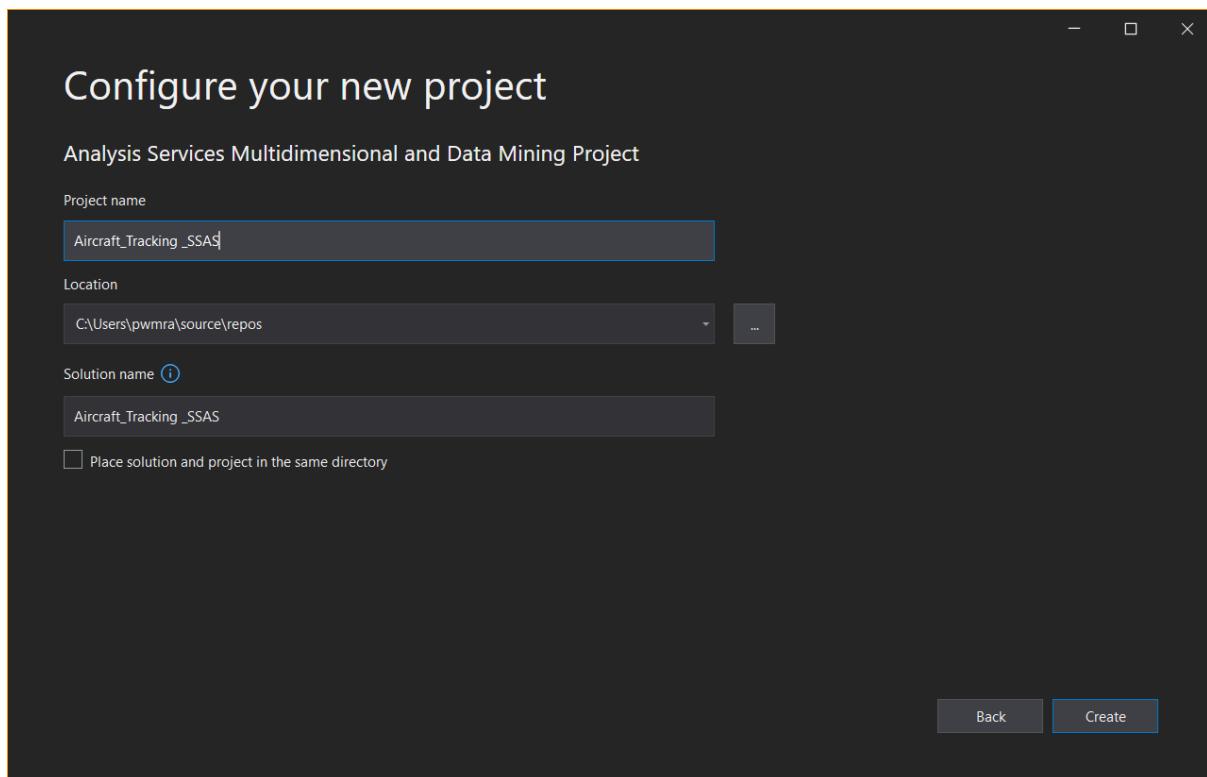


Figure 1.1 - configure your new project in SSAS

Once the project created you can see a similar folder structure to the image below, in Solution Explorer.

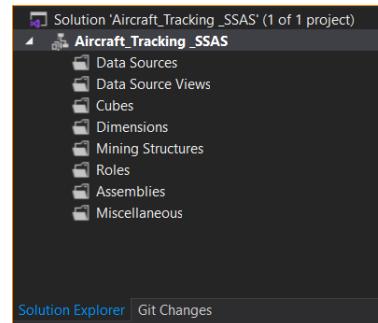


Figure 2.1.1 - Created Aircraft_Tracking_SSAS project

We need to configure components above, starting from Data Sources to Dimensions, in order to create a working SSAS Cube

2.2 Create new data source

Then, we have to configure a data source. Data source defines from where, the cube is extracting data. To configure, right click on Data Sources and create new data source. Then you can have the data source wizard like the image given below.

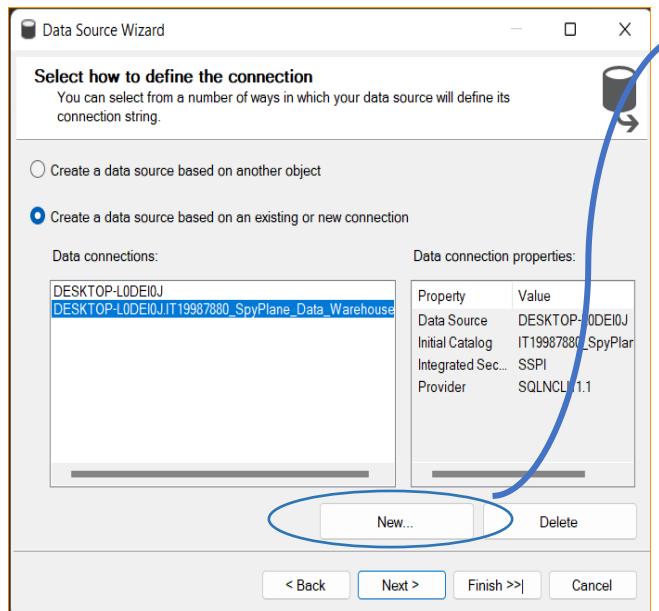


Figure 2.2.1 - select how to define the connection – step 1

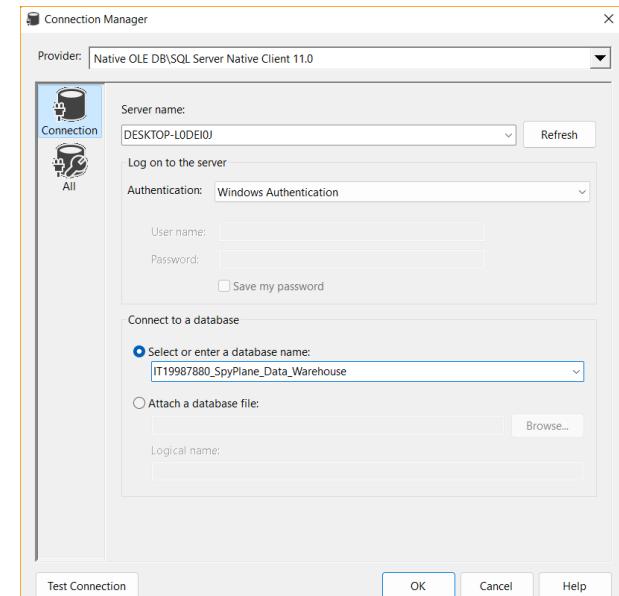


Figure 2.2.2 - how to define new connection - step 2

In here we are selecting a connection to the data source, Therefore, '**IT19987880_SpyPlane_Data_Warehouse**' as we have already created this connection previously in SSIS projects. However, after following the steps successfully we can see the data source we created under data sources.

2.3 Create new data source View.

Likewise, I have created the Data sources view so like this

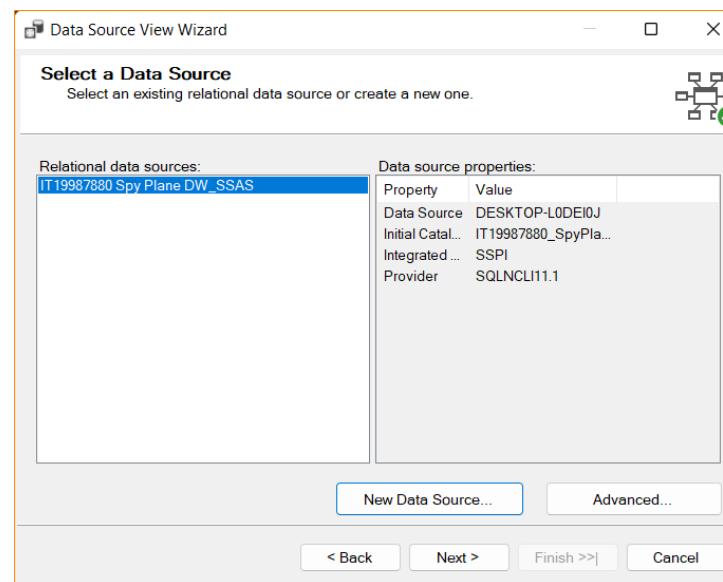


Figure 2.2.1 - Select data source & create a view

Then you can see the created data sources and data source view like the image given below

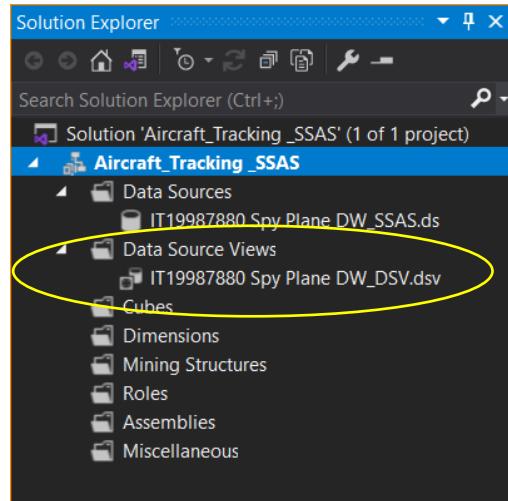


Figure 2.3.2 – created data source view file is shown in content

After I open the created data source view, you will be able to see that none of the tables are connected in the design view once completed

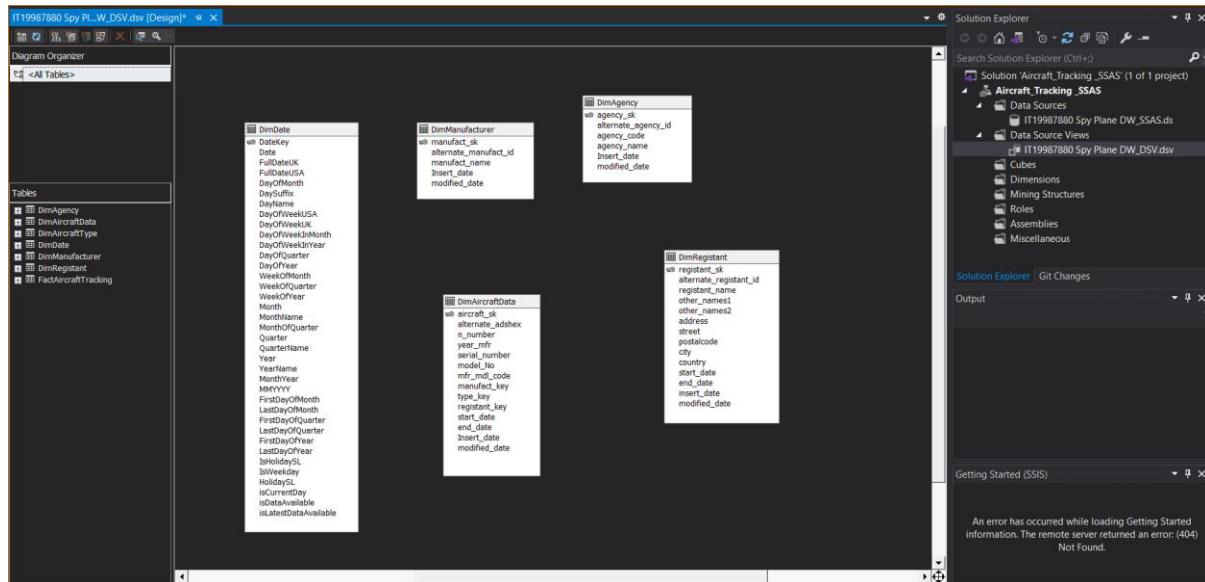


Figure 2.3.3 - Data source view without connection

In that case, we have to create table links. To link ‘DimAircraftData’ and ‘FactAircraftTracking’ tables, click on ‘adshex_key’ column of the ‘FactAircraftTracking’, drag and drop it on the ‘aircraft_sk’ column of the ‘DimAircraftData’ table. Similarly using corresponding SK/FKs link, link all the tables.

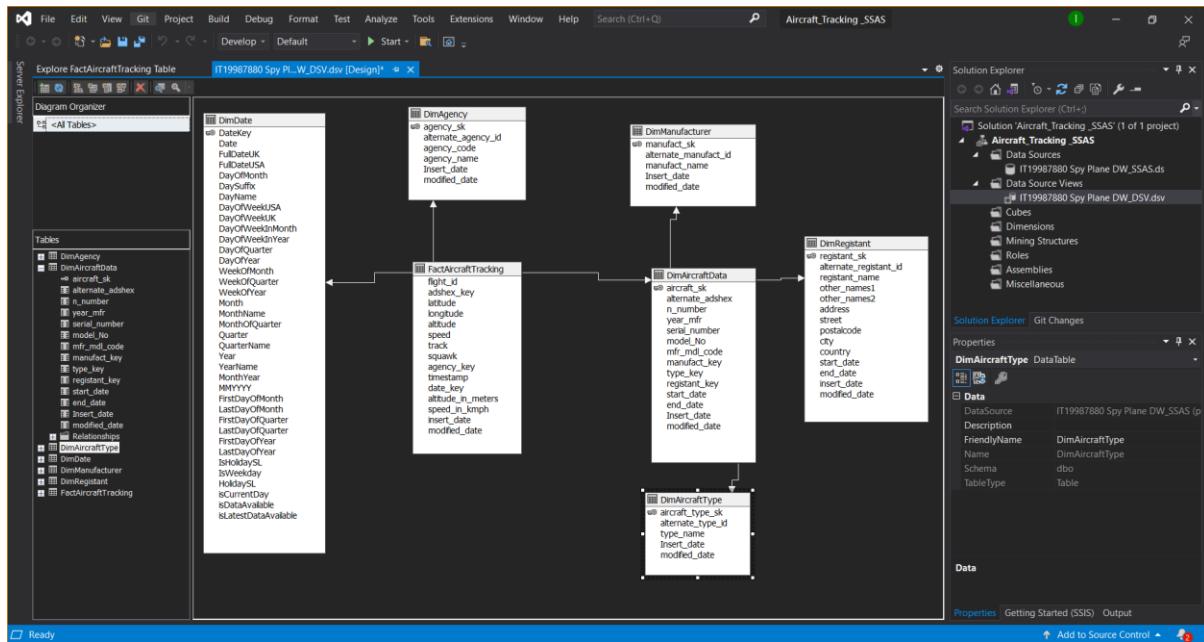


Figure 2.3.4 - Data source view with connections

2.4 Create a cube

Then we have to create a cube. For that, we can right click on Cubes and create a new cube. Then, cube wizard will be popped up. Then we can complete the wizard as follows.

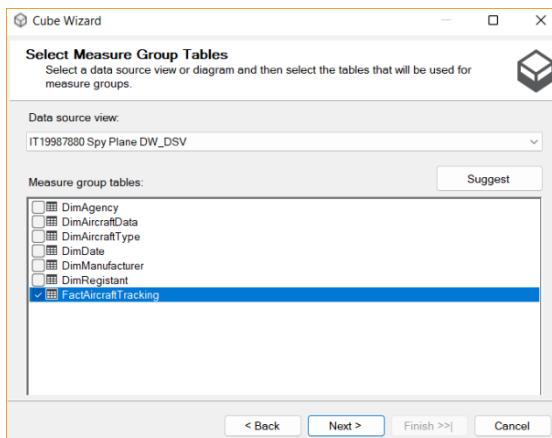


Figure 2.4.1 - Cube implementation - step 1

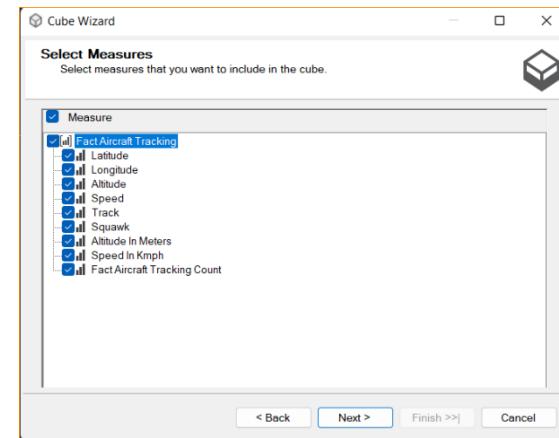


Figure 2.3.2. - Cube implementation - step 2

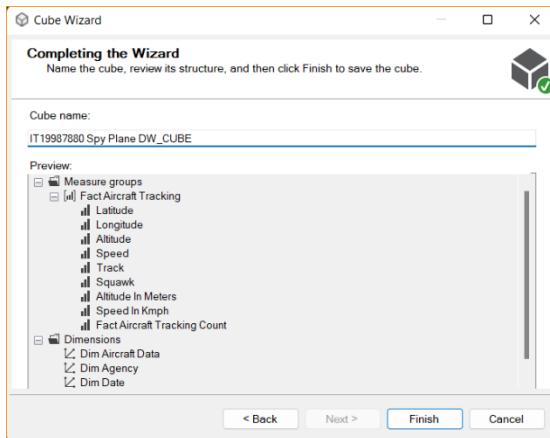


Figure 2.4.4 - Cube implementation - step 4

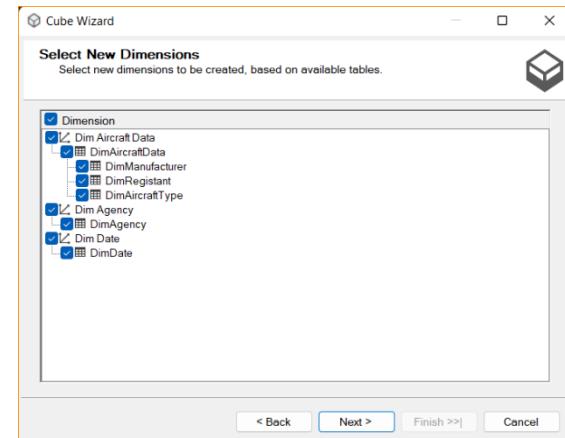
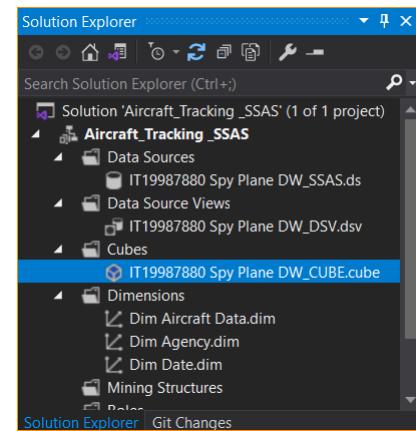


Figure 2.4.3 - Cube implementation - step 3

Then, you can see the solution explore with the created cube

Figure 2.4.5 - Created cube file is shown in content file list



Then you will see the same snowflake schema is built as a cube

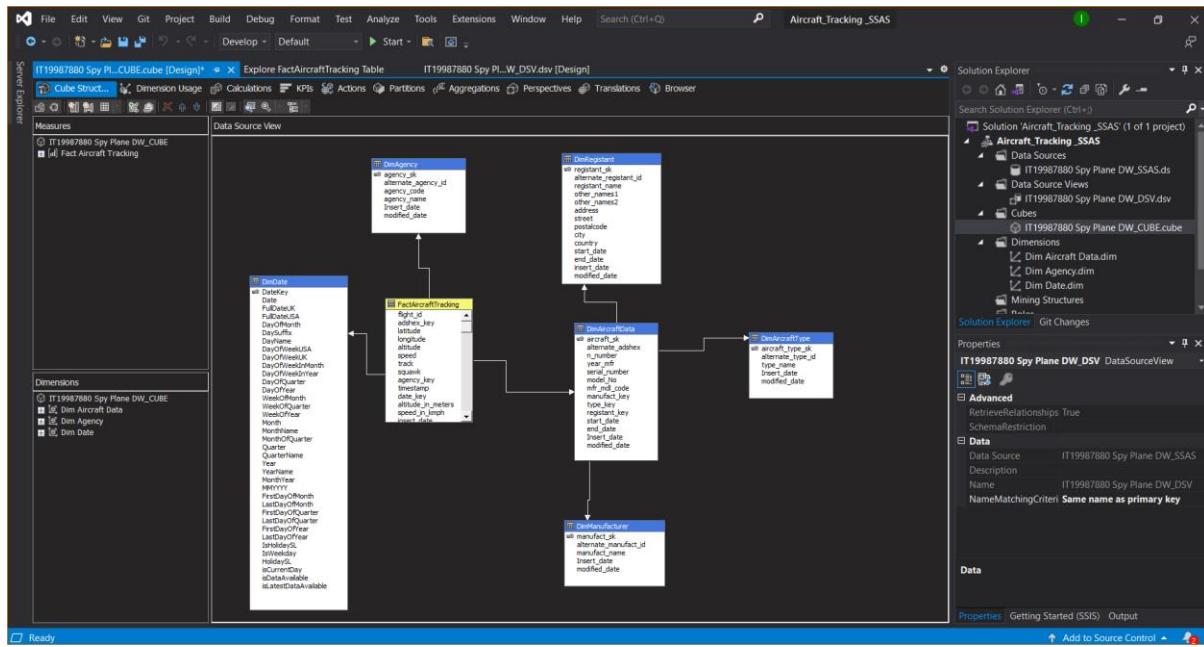


Figure 2.4.6 - Snowflake schema is built as a cube

In the Dimensions section, you can only see the dimension tables which are directly linked to the fact table. Then we have to expand those dimensions and check whether other attributes are missing or not. Most of the time attributes only consist with the surrogate keys. Therefore, we have to drag and drop the attributes, but derived attributes have to be avoided in this point. Reason is ‘Modified_Date’, ‘Insert_Date’ like reality fields makes no meaning for an analysis.

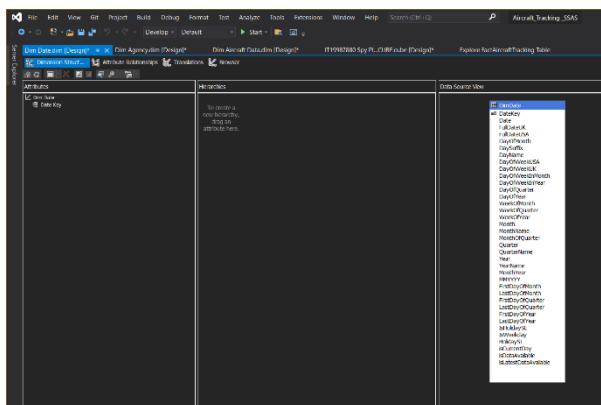


Figure 2.4.7 - Expand ‘DimDate’ dimensions and check whether attributes are missing or not

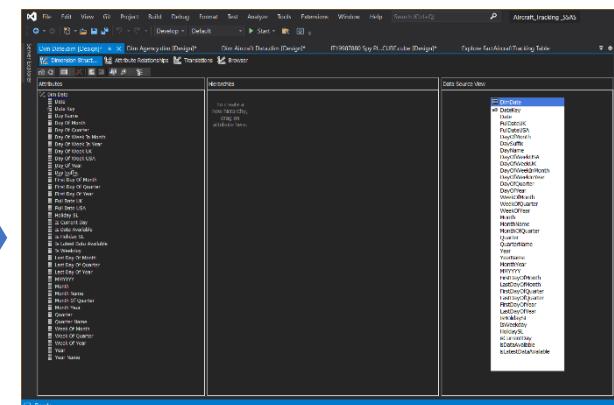


Figure 2.4.8 - Drag and drop missing attributes to 'DimDate' dimension

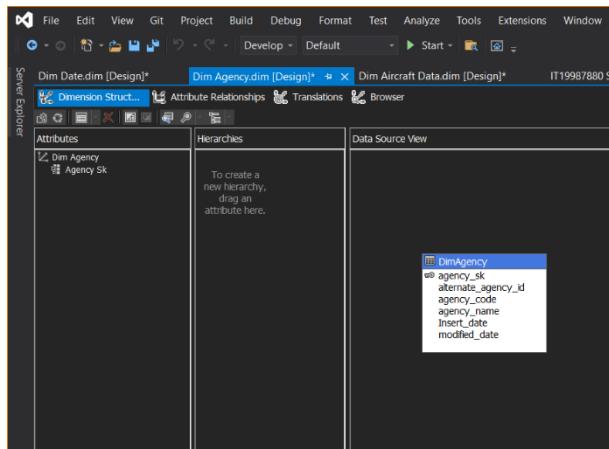


Figure 2.4.9 - Expand 'Agency' dimension and check whether attributes are missing or not

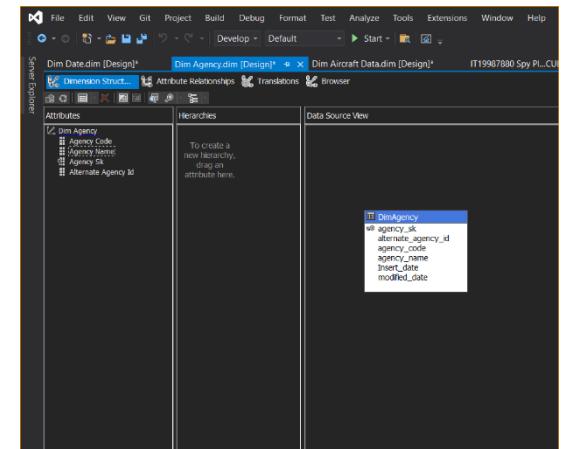


Figure 2.4.10 - Drag and drop missing attributes to 'Agency' dimension

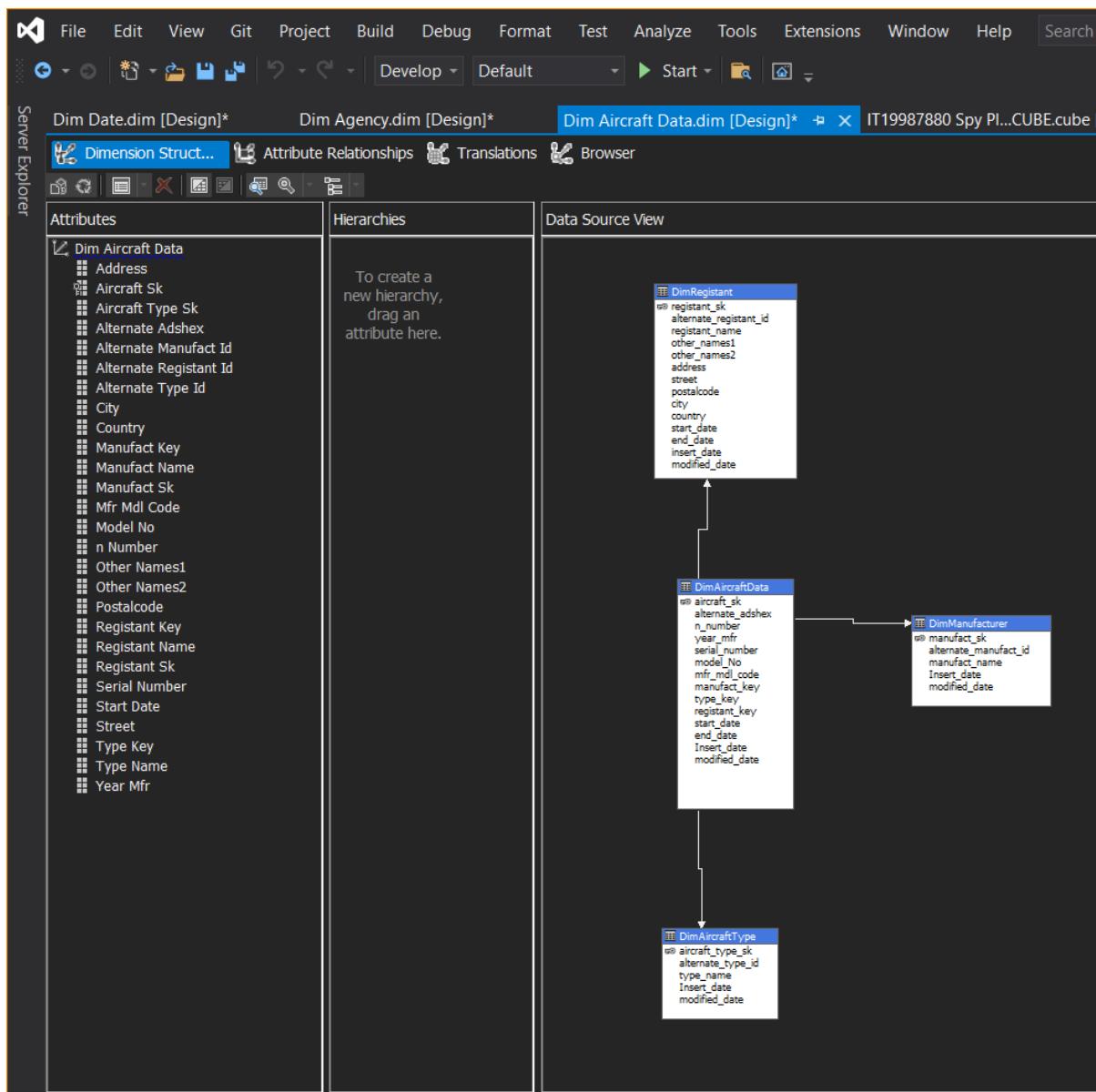


Figure 2.4.11 - Drag and drop missing attributes to 'AirCraft Data' dimension

2.5 Deploy the cube

Right click on the 'IT19987880_SSAS_SpyPlane_DW' and click on the Deploy

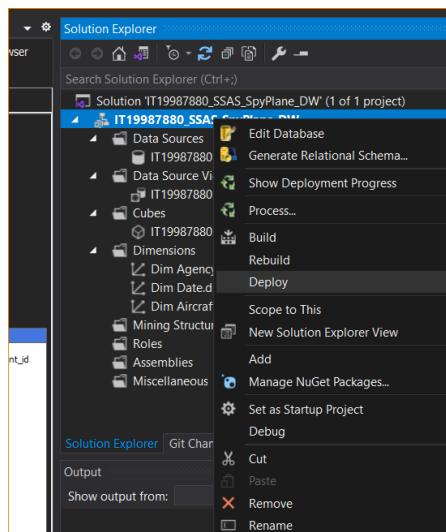


Figure 2.5.1 - Prepare for the deploy

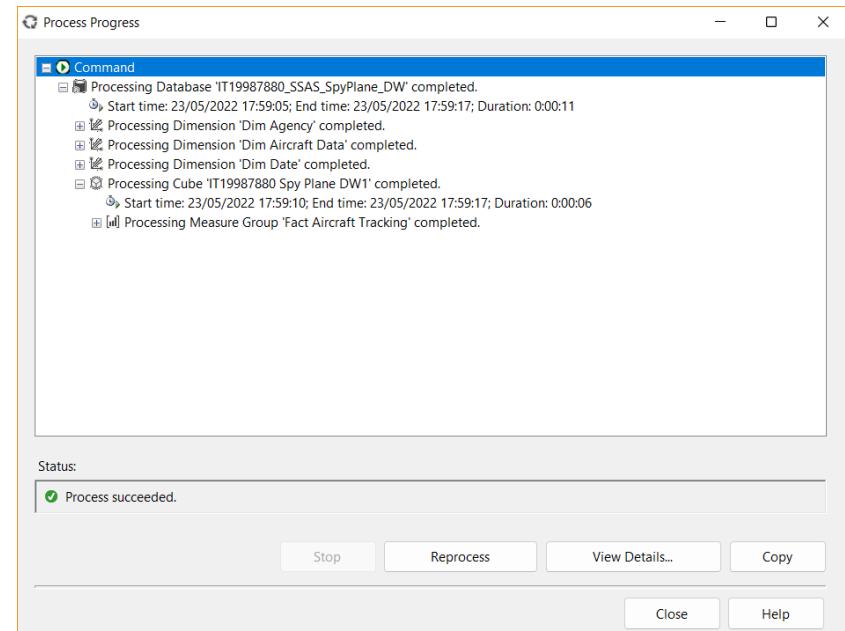


Figure 2.5.2 - Process the deploy progress

You will see a pop-up window displaying the progress of the deployment

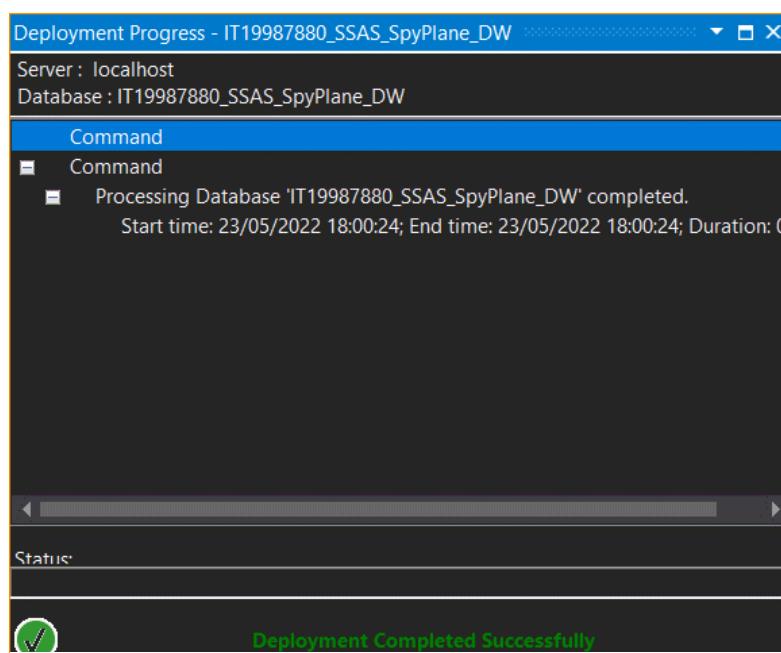


Figure 2.5.3 - Get the deployment completed successfully message

Once successfully deployed, we have to go to Browser tab under '→ **IT19987880_SSAS_SpyPlane_DW**' design window ('→ **IT19987880_SSAS_SpyPlane_DW [Design]**'), will have the attributes of the model on the left hand side, where you can drag and drop the into the design area on the right-hand side and do some test analysis. We have to open the SQL Server Management Studio for continue the other steps. Then like the image given below, we have to select Analysis Services...

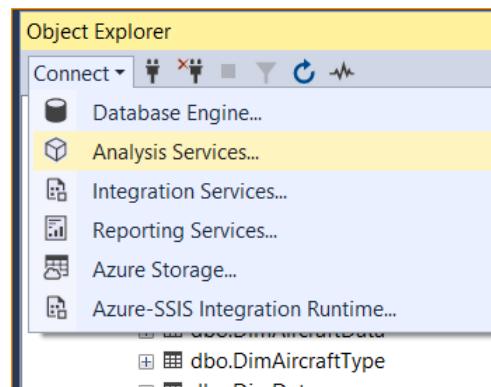


Figure 2.5.4 - In SSMS select the 'Analysis Service' to connect

There we can open the databases → Cubes → **IT19987880_SSAS_SpyPlane_DW** → right click → Browse. Then we can see the surface like the image given below.

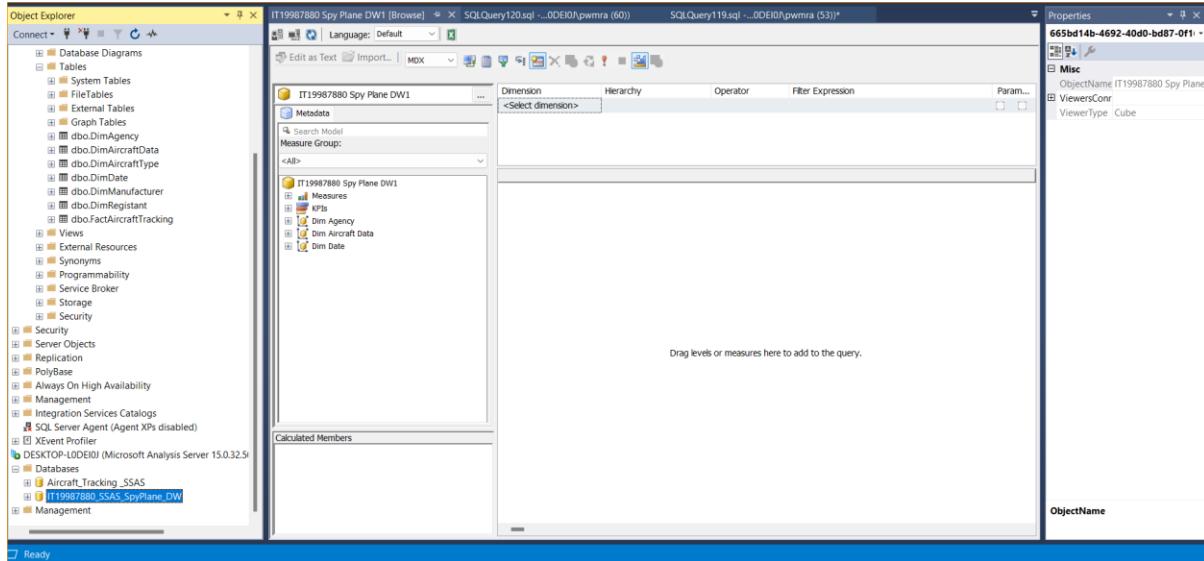


Figure 2.5.5 - In SSAS check whether deployed files are there

Its look like the same browser window you saw in Data Tools in '**IT19987880_SSAS_SpyPlane_DW**' design window ('**IT19987880_SSAS_SpyPlane_DW [Design]**') should appear here allowing you to analyse data. Optionally you can write MDX, DAX queries there

2.6 Create a KPI

Then we can create a KPI which are created based on the business requirements. KPIs depend on what the organization want to monitor and measure.

In order to create a sample KPI definition, go back to Data Tools, locate and go to KPIs tab in '**IT19987880_SSAS_SpyPlane_DW**' design window ('**IT19987880_SSAS_SpyPlane_DW [Design]**').

If the design window is not visible you can double click on '**IT19987880_SSAS_SpyPlane_DW**' to open the design window.

Then, In the KPIs tab, above KPI Organizer panel, locate and click on New KPI button. Alternatively, you right click on KPI Organizer panel area and select New KPI. After following steps for create the KPIs successfully. We can have a KPI like the image given below.

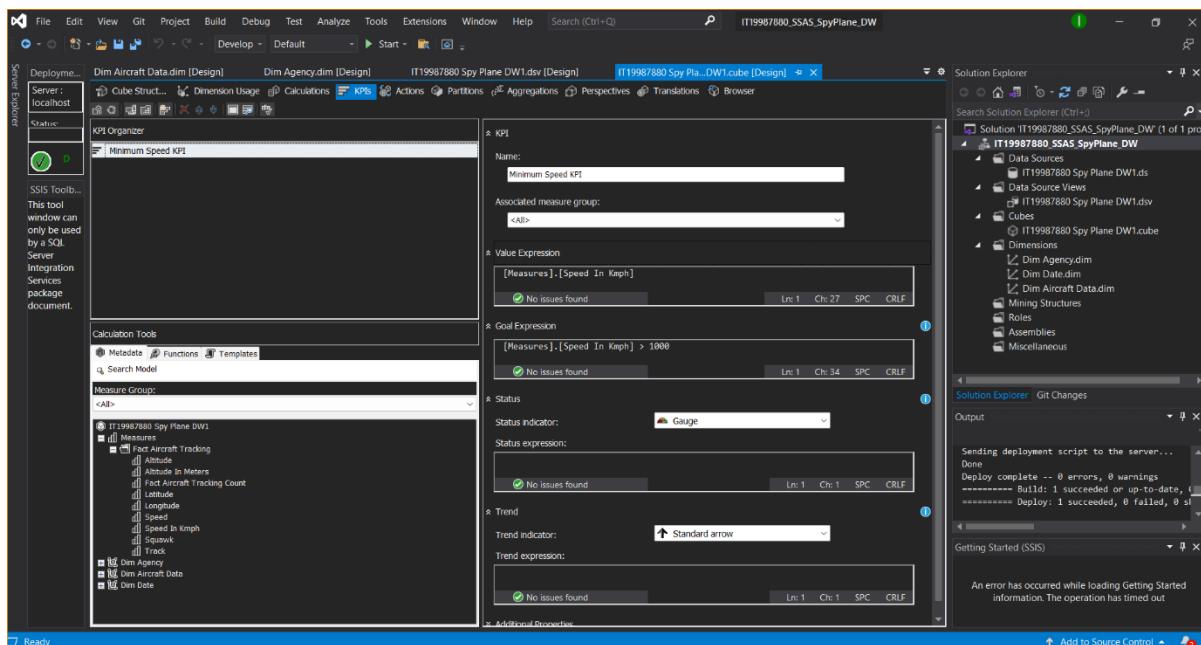


Figure 2.6.1 - Create KPI

Now you can include the KPI you created in your analysis to the test

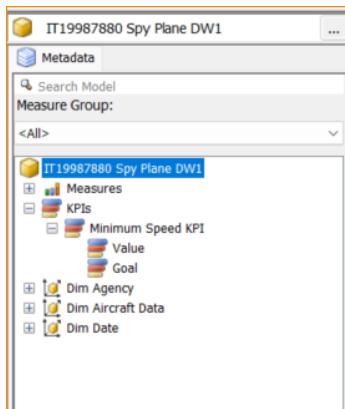


Figure 2.6.2 - In SSAS, Created KPI is shown in here

Then we can Do some research on creating hierarchies to dimensions, adding new measures to the cube, creating Business Intelligence, so on. Let's Implement few hierarchies in the cube.

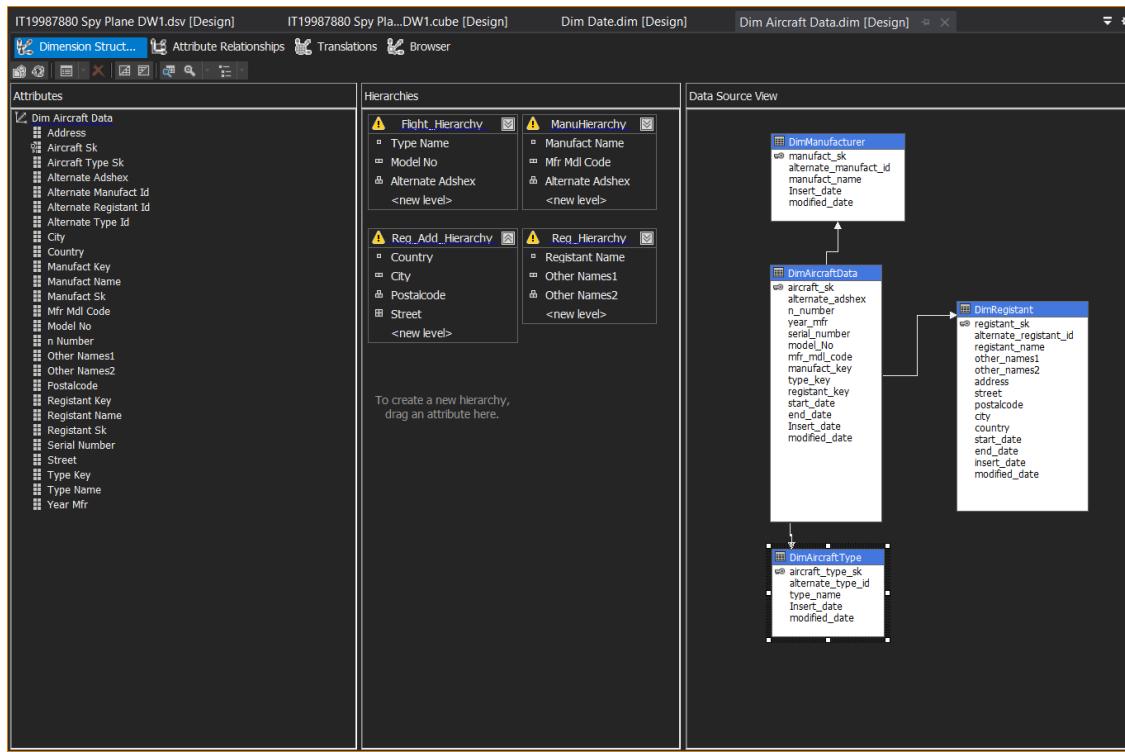


Figure 2.6.3 - Create Hierarchies for 'Dim Aircraft' Dimension

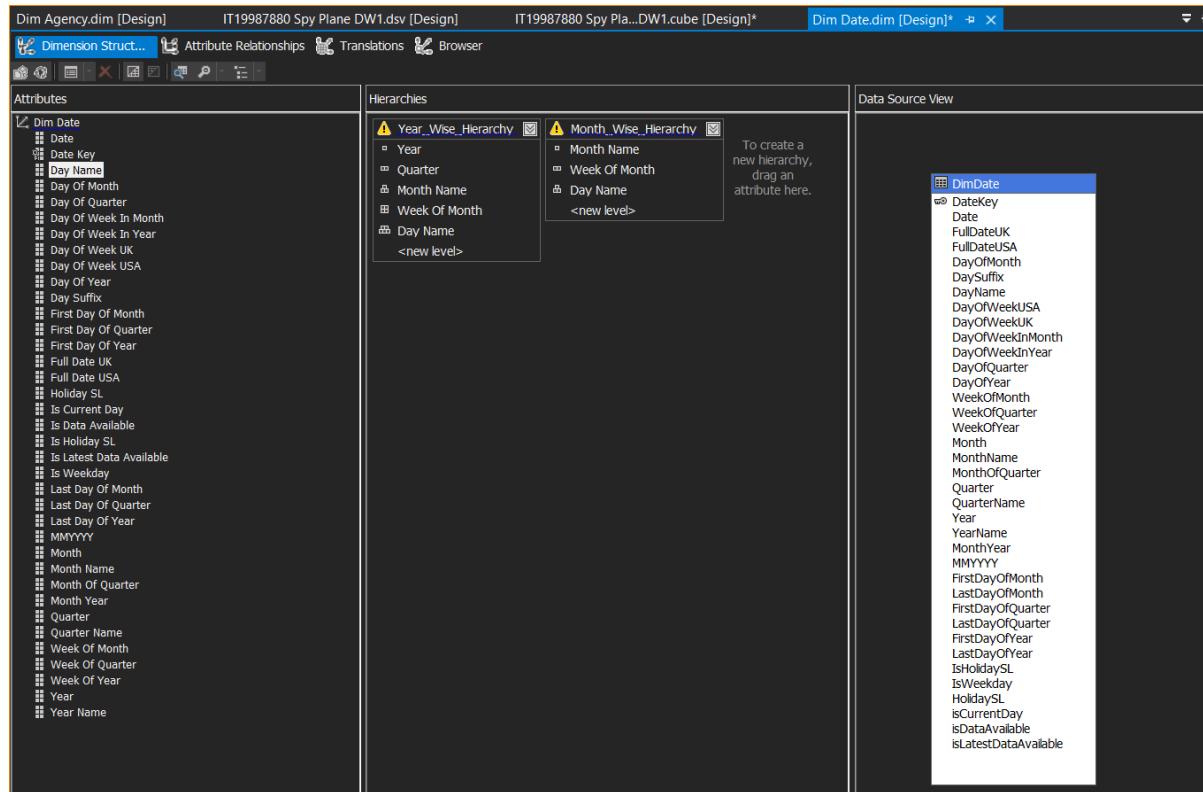
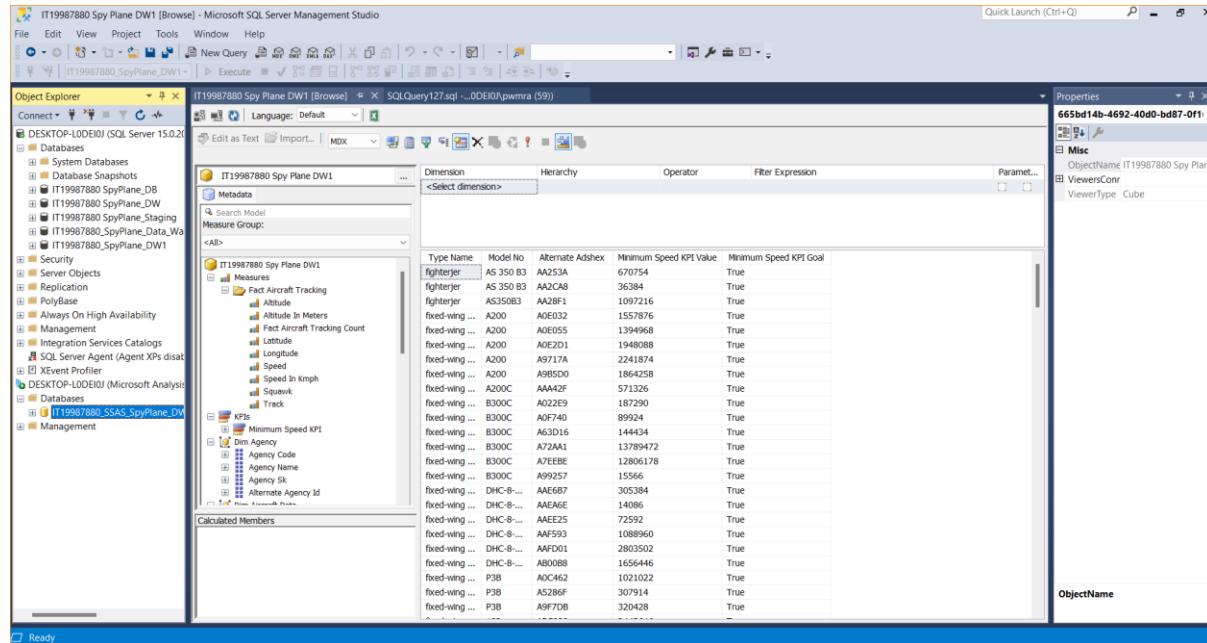


Figure 2.6.4 - Create Hierarchies for 'Dim Date' Dimension

2.7 Browsing Cube Data

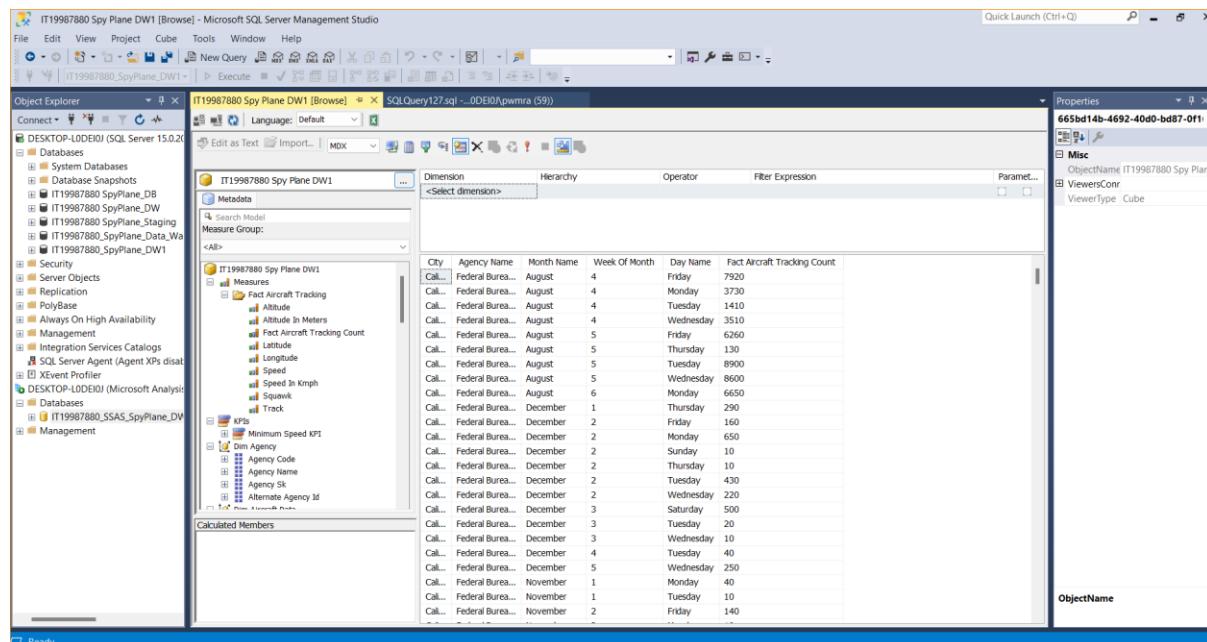
And after all, as we already know general browsing (analysis) can be done via the development tool; Data Tools or in SSMS. There snapshots included both development tools.

- Microsoft Visual Studio – Test Analyse:



The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The title bar reads "IT19987880 Spy Plane DW1 [Browse] - Microsoft SQL Server Management Studio". The left pane is the Object Explorer, showing the database structure including "IT19987880_SpyPlane_DW1". The center pane displays a table titled "IT19987880 Spy Plane DW1" with columns: Dimension, Hierarchy, Operator, Filter Expression, Parameter, and a large table body. The right pane is the Properties pane, showing details like "ObjectName IT19987880 Spy Plane" and "ViewType Cube".

Figure 2.7.1- Browsing Cube Data



The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The title bar reads "IT19987880 Spy Plane DW1 [Browse] - Microsoft SQL Server Management Studio". The left pane is the Object Explorer, showing the database structure including "IT19987880_SpyPlane_DW1". The center pane displays a table titled "IT19987880 Spy Plane DW1" with columns: Dimension, Hierarchy, Operator, Filter Expression, Parameter, and a large table body. The right pane is the Properties pane, showing details like "ObjectName IT19987880 Spy Plane" and "ViewType Cube".

Figure 2.7.2 - Browsing Cube Data

3 Demonstration of OLAP operations

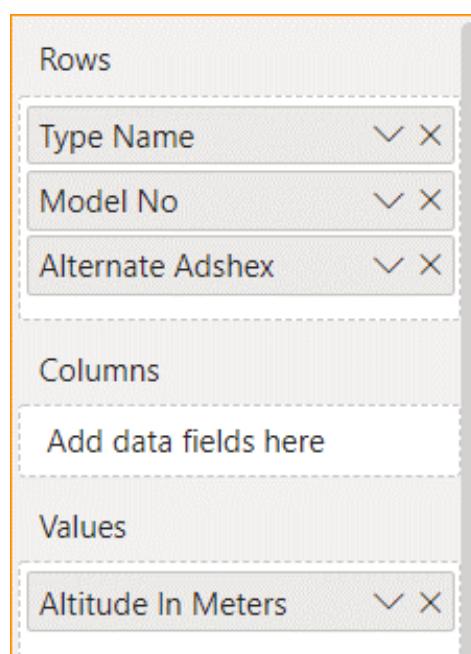
3.1 PowerBI reports for OLAP operations

PowerBI software which was developed by Microsoft Corporation is a popular visualization tool. The roll-up, slice, dice and drill-down operations have been visualized in this section using the PowerBI solution.

3.1.1 Roll-up

The roll up or roll up operation will climb up through the hierarchies of dimensions to aggregate data.

Here depicts the roll up operation where activity type rolled up according to Sector types.



Rows

- Type Name
- Model No
- Alternate Adshex

Columns

Add data fields here

Values

Altitude In Meters

Figure 3.1.1.1 - Drill-down Type > Model > Adshex No wise Altitude

Type > Model > <u>Adshex No wise Altitude (m)</u>	
Type Name	Altitude In Meters
fixed-wing single-engine	6,356,411,048.00
182T	2,284,474,206.00
A8A872	223,864,504.00
ADC1C3	181,202,006.00
A846DD	180,439,444.00
A8464D	172,133,980.00
A10FBB	170,322,304.00
AC19C5	140,028,566.00
AC9E93	120,494,680.00
A2F6FD	114,114,752.00
A4BC6E	113,702,280.00
A59DSB	82,598,880.00
A7E68F	79,646,044.00
AC97AB	68,169,046.00
A6718E	55,860,608.00
A385F0	51,587,736.00
AC872B	46,795,978.00
A5B556	38,168,934.00
Total	6,356,411,048.00

Figure 3.1.1.2 - Drill-down Type > Model > Adshex No wise Altitude

Aircraft Dashboard

Type > Model > Adshex No wise Altitude (m)

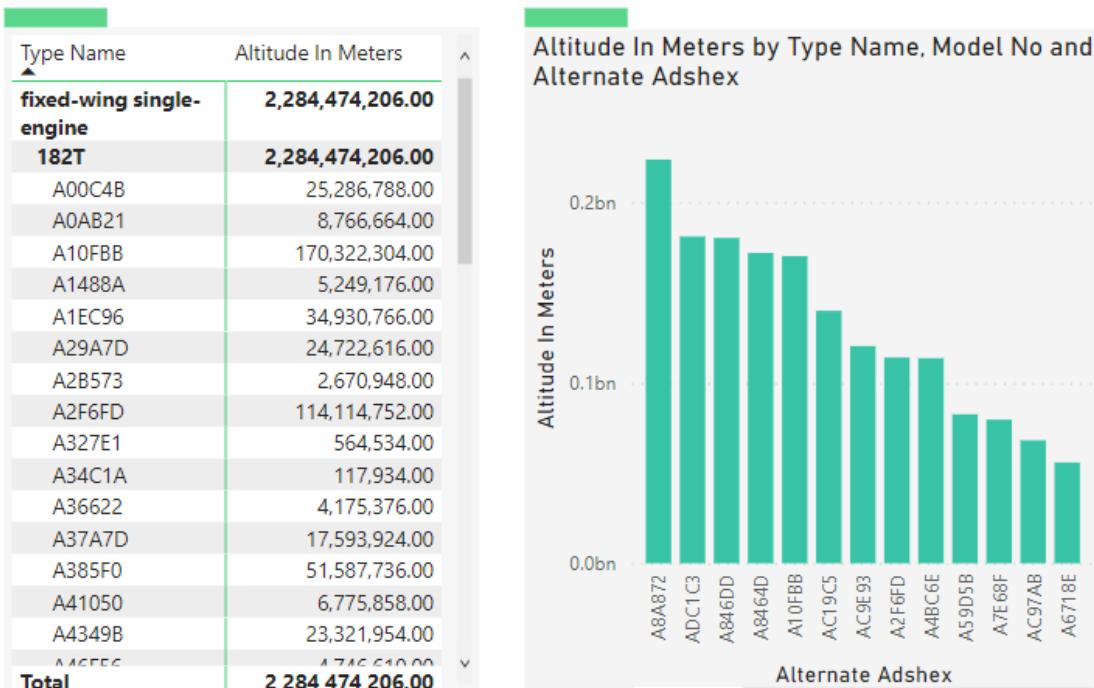
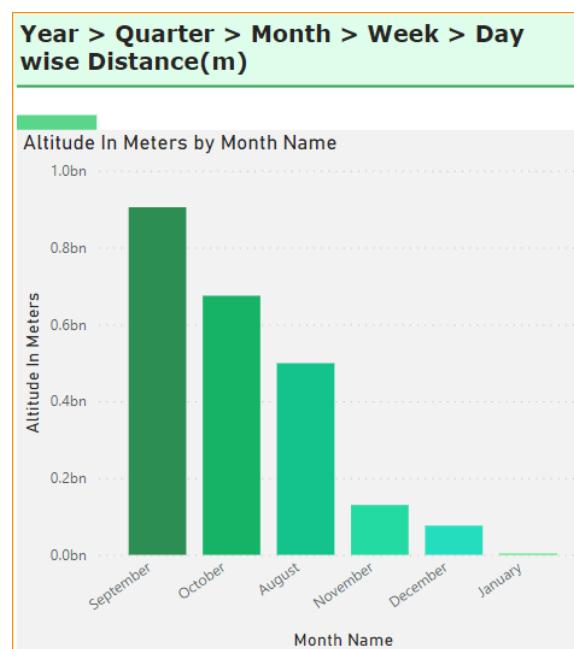


Figure 3.1.1.3 - Drill-down Type > Model > Adshex No wise Altitude

Here depicts the roll up operation where **Altitude(m)** rolled up according to date Hierarchy



3.1.2 Drill Down

Here I have used date hierarchy for show drill down

The below one demonstrates how the performance is drilled down through the Date Hierarchy.

(Year → Quarter → Month → Week → Day)

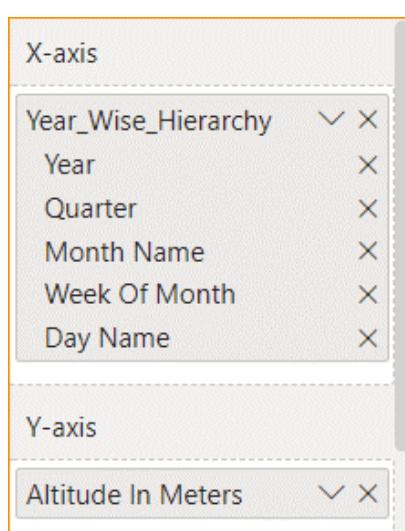


Figure 3.1.2.1 - Date Hierarchy have used in here

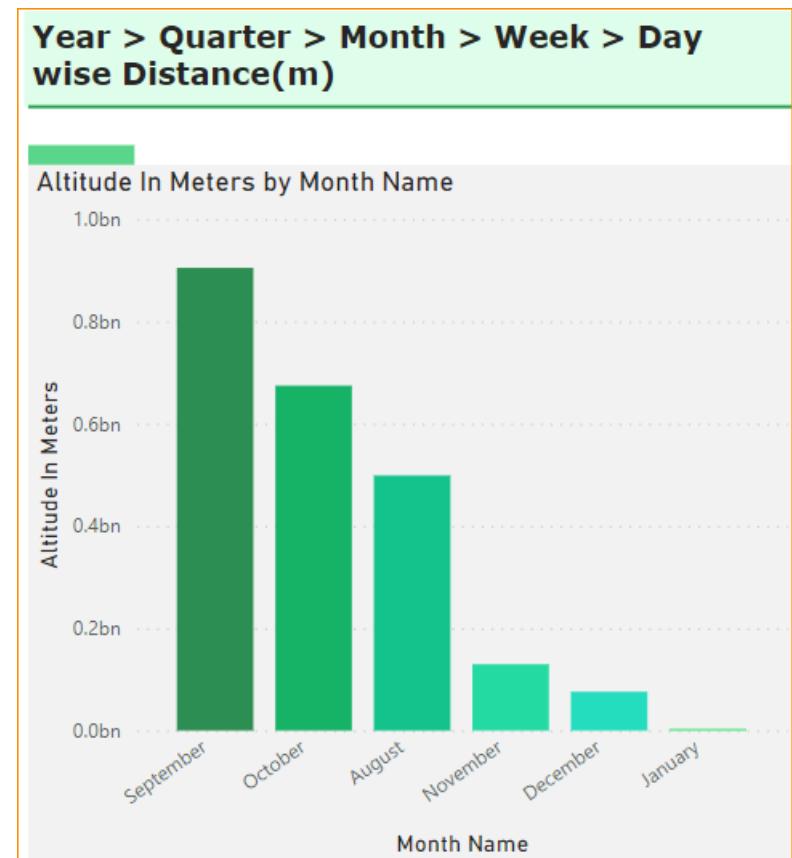


Figure 3.1.2.2 - Till (Year → Quarter → Month) have shown here

In here this figure shows full page of Roll up/Drill down.

Drill-Down - Power BI Desktop

File Home Insert Modeling View Help External Tools

New page ▾ New visual ▾ More visual ▾ Q&A Key influencers Decomposition tree Smart narrative Paginated report Power Apps Power Automate (preview) Text box Buttons Shapes Image Elements Add a sparkline Sparklines

Pages Visuals AI visuals Power Platform

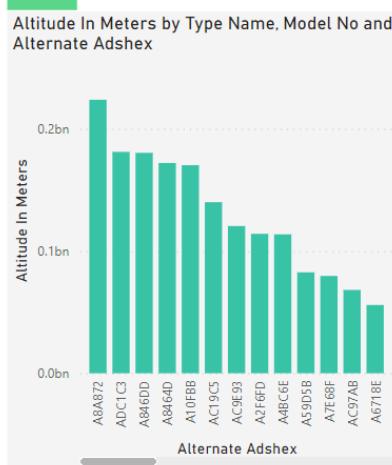
Auto recovery contains some recovered files that haven't been opened. View recovered file

Aircraft Dashboard [Roll up/ Drill Down]

Type > Model > Adshex No wise Altitude (m)

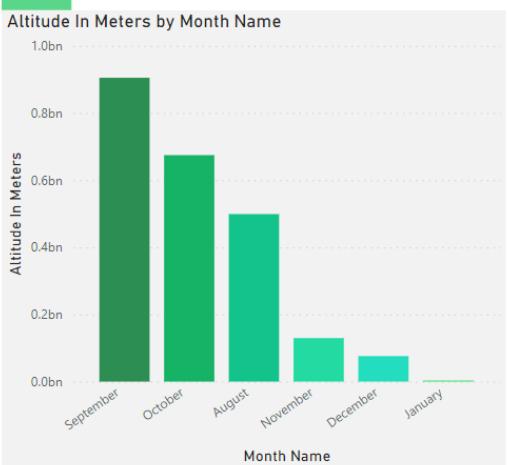
Type Name	Altitude In Meters
fixed-wing single-engine	2,284,474,206.00
182T	2,284,474,206.00
A00C4B	25,286,788.00
A0AB21	8,766,664.00
A10FBB	170,322,304.00
A1488A	5,249,176.00
A1EC96	34,930,766.00
A29A7D	24,722,616.00
A2B573	2,670,948.00
A2F6FD	114,114,752.00
A327E1	564,534.00
A34C1A	117,934.00
A36622	4,175,376.00
A37A7D	17,593,924.00
A385F0	51,587,736.00
A41050	6,775,858.00
A4349B	23,321,954.00
Total	2,284,474,206.00

Altitude In Meters by Type Name, Model No and Alternate Adshex



Year > Quarter > Month > Week > Day wise Distance(m)

Altitude In Meters by Month Name



Slice Year wise Rollup/Drill down Dice Pivot Table Rollup/Drill down +

Figure 3.1.2.3 - Roll up/ Drill Down Dashboard page

3.1.3 Slice operation

The slice operation allows a rectangular subset of an OLAP cube to be queried. A Slice function much like a report or a query that it returns data based on a request for what to see

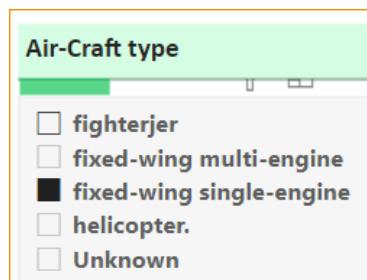


Figure 3.1.3.1 - slice operation

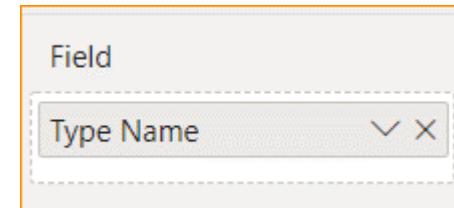


Figure 3.1.3.2 - Used fields for Slice operation

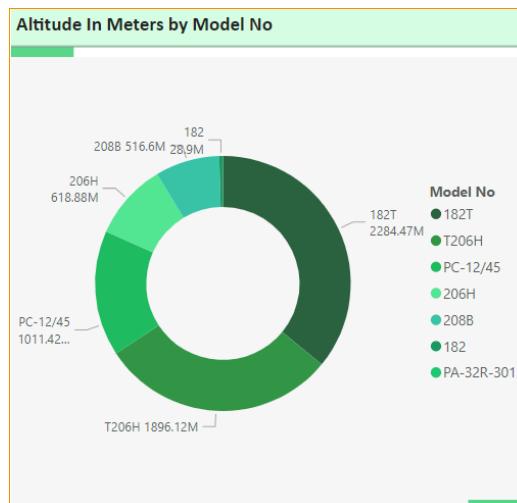


Figure 3.1.3.3 - Donut Chart - (Altitude by Model No)

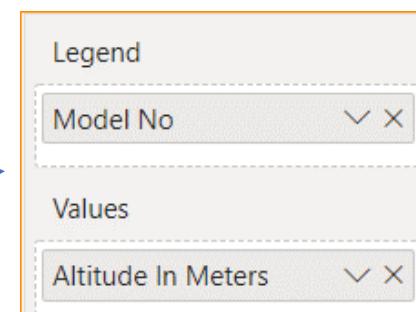


Figure 3.1.3.4 - Used fields for Donut chart

Air-Craft type wise Altitude in Meters	
Type Name	Altitude In Meters
fixed-wing single-engine	6,356,411,048.00
CESSNA	5,341,051,394.00
182	28,904,930.00
182T	2,284,474,206.00
206H	618,878,116.00
208B	516,601,022.00
T206H	1,892,193,120.00
PILATUS	1,011,421,220.00
PC-12/45	1,011,421,220.00
PIPER	8,836.00
PA-32R-301	8,836.00
TEXTRON AVIATION INC	3,929,598.00
T206H	3,929,598.00
Total	6,356,411,048.00

Figure 3.1.3.5 - In detailed table

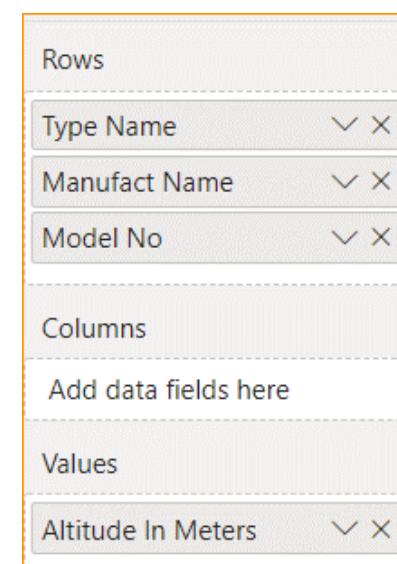


Figure 3.1.3.6 - Used fields for that table

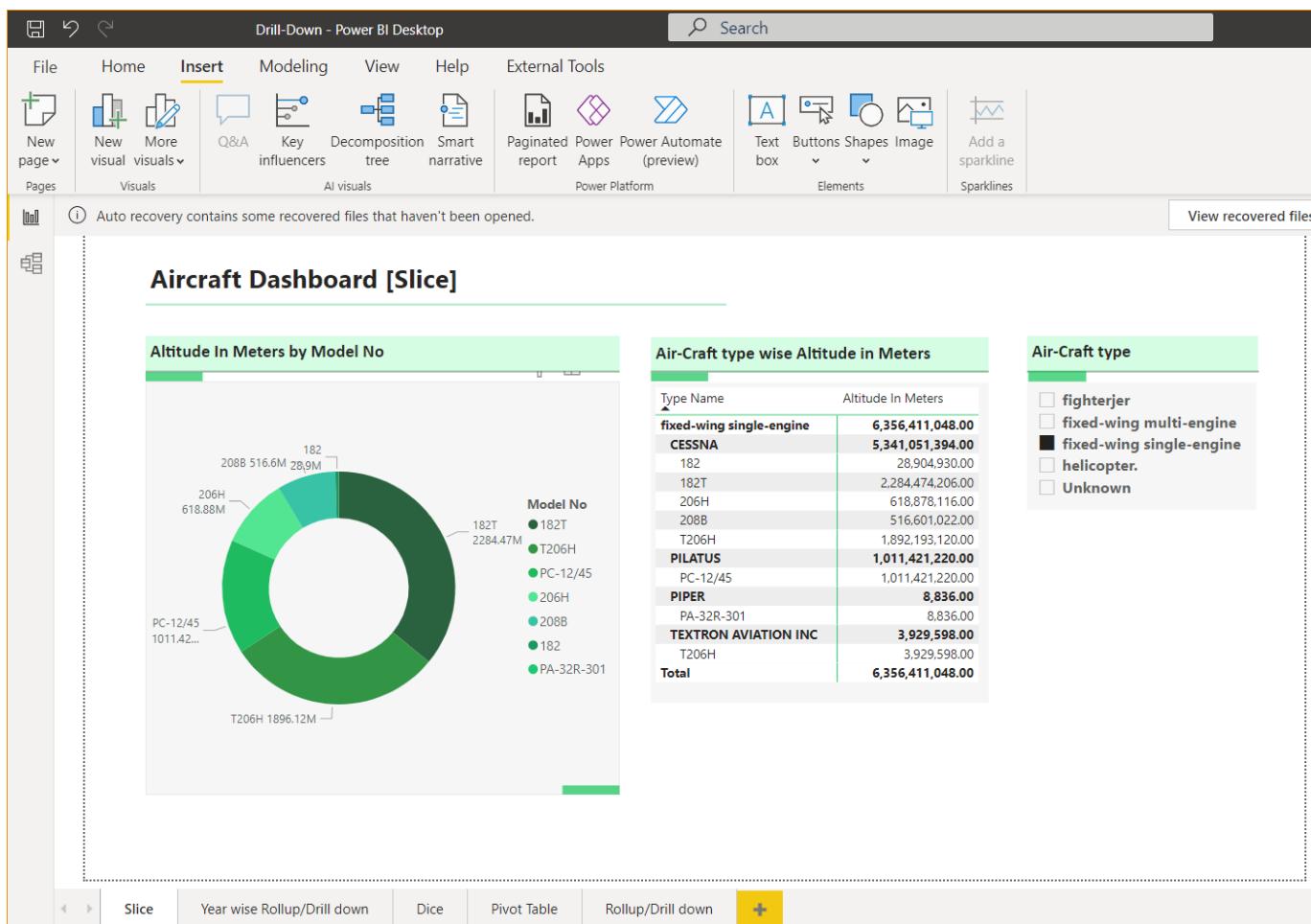


Figure 3.1.3.7 - Slice Dashboard page

3.1.4 Dice operation

The dice operation will select two or more dimensions from an OLAP cube and will provide a new subset of the cube by selecting relevant values on the dimensions that were selected.

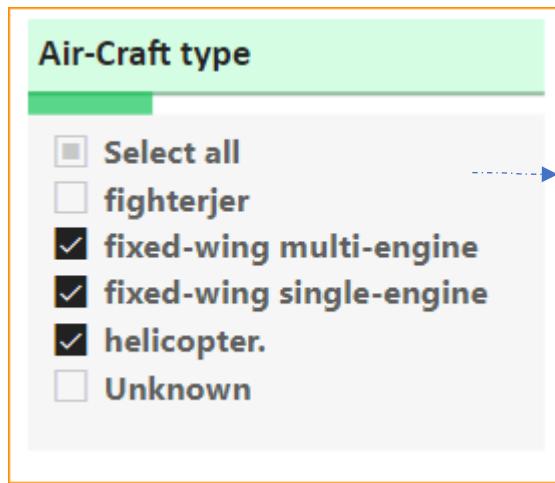


Figure 3.1.4.1 - Dice Operation

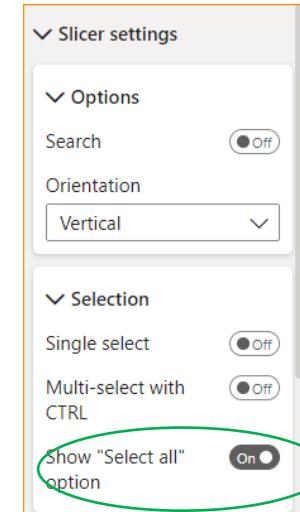
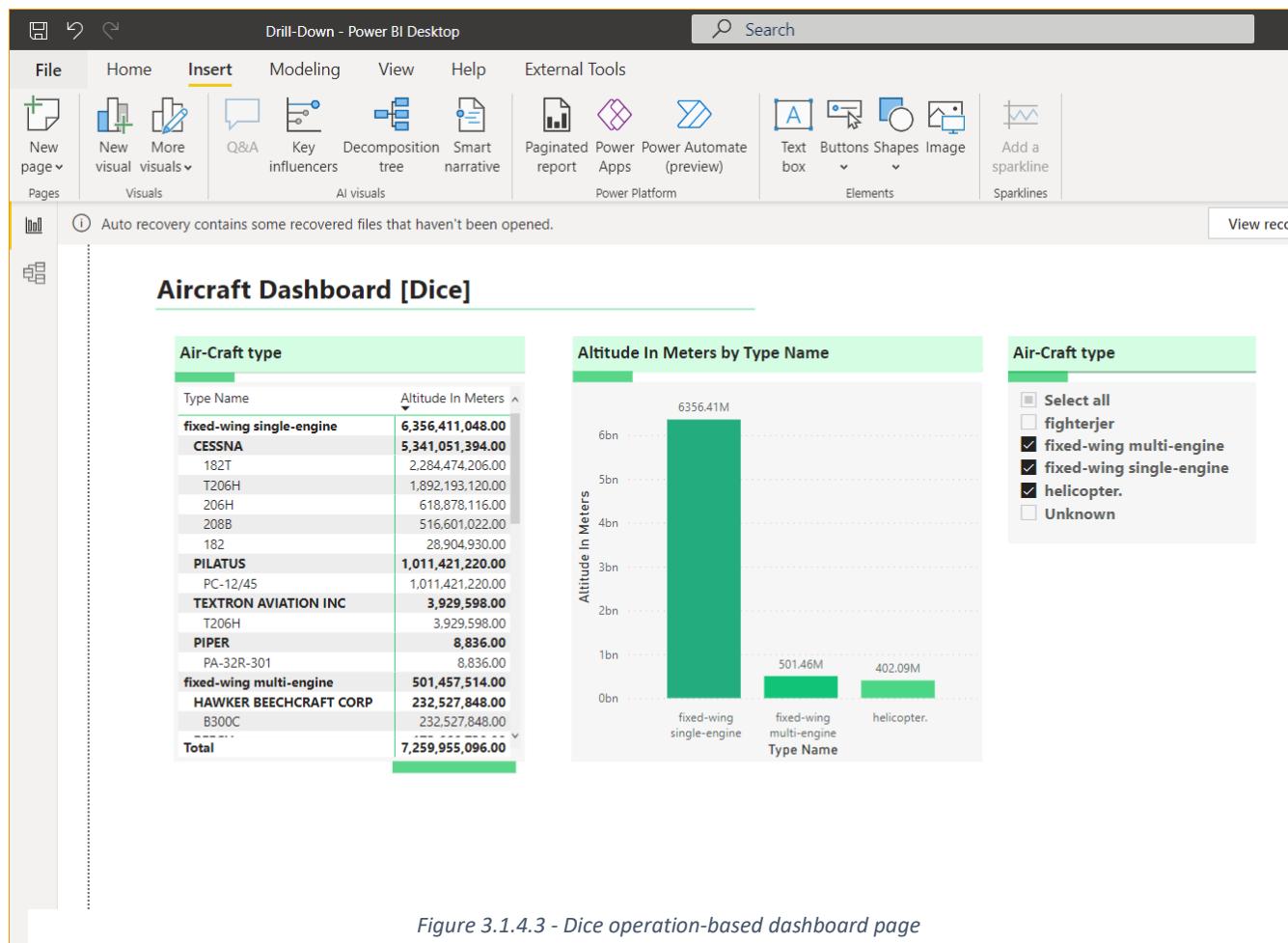


Figure 3.1.4.2 - Select on - "Select All" option

This feature provides select two or more dimensions from OLAP cube



A screenshot of a Power BI dashboard titled "Aircraft Dashboard [Dice]". The dashboard features three main components:

- Air-Craft type Table:** A table showing aircraft type names and their corresponding altitude in meters. The table includes rows for CESSNA, PILATUS, TEXTRON AVIATION INC, and HAWKER BEECHCRAFT CORP.
- Altitude In Meters by Type Name Bar Chart:** A bar chart showing the altitude in meters for different aircraft types. The y-axis ranges from 0bn to 6bn. The bars are labeled with their respective altitudes: 6356.41M (fixed-wing single-engine), 501.46M (fixed-wing multi-engine), and 402.09M (helicopter).
- Air-Craft type Slicer:** A slicer interface identical to the one shown in Figure 3.1.4.1, allowing users to select dimensions from the OLAP cube.

Figure 3.1.4.3 - Dice operation-based dashboard page

3.1.5 Pivot charts operation

Pivot charts allow users to rotate an OLAP cube so that the views of the cube can be seen. This rotation will change the orientation of dimensions giving various combinations.

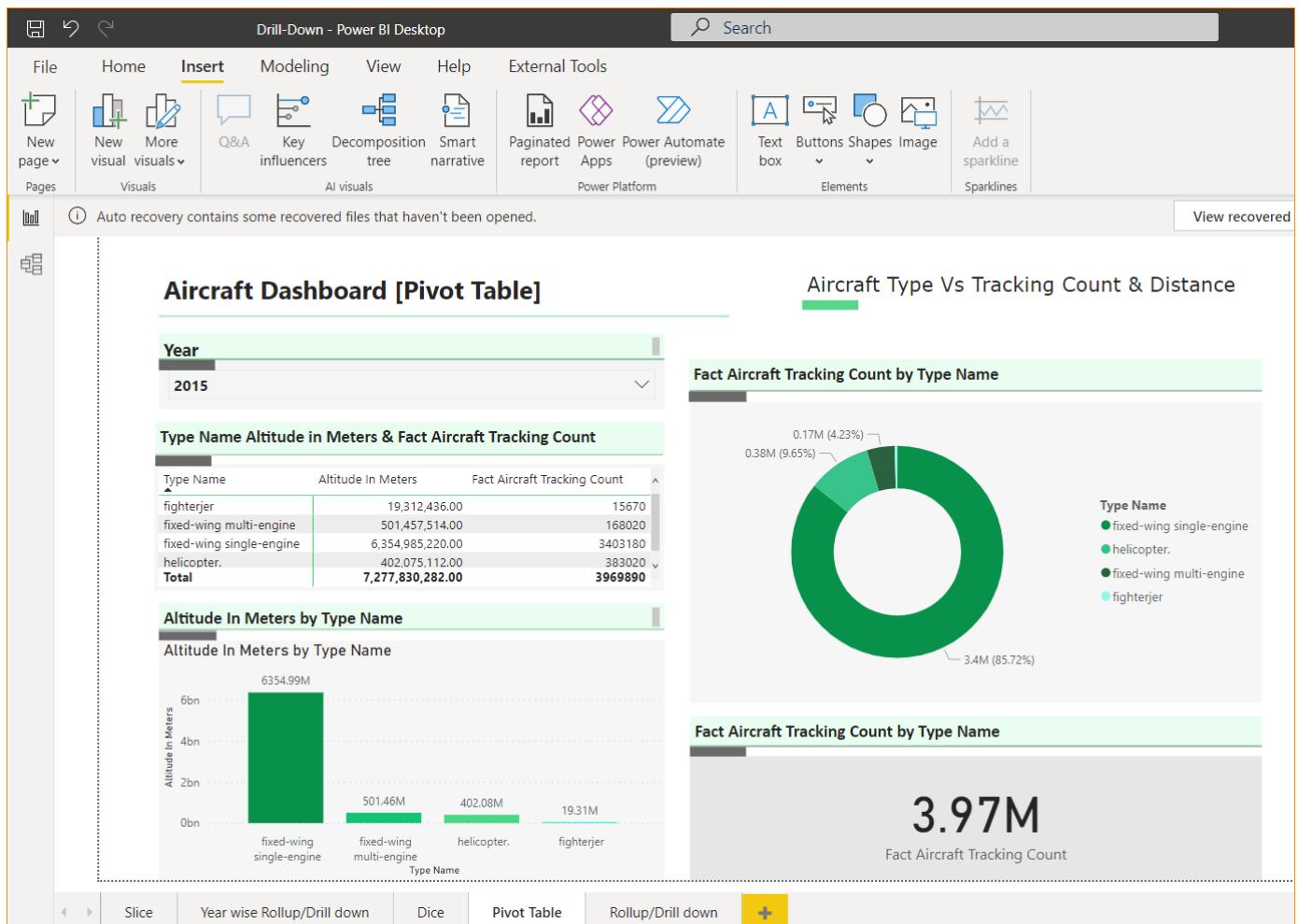
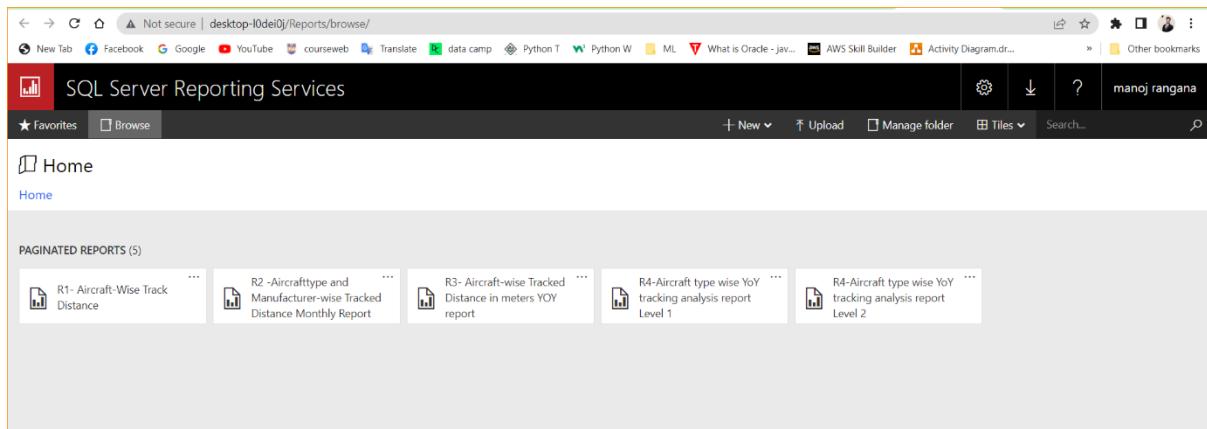


Figure 3.1.5.1 - Pivot Chart operation Dashboard

4 SSRS Reports

I have Developed and published the reports given below in SSRS Web Portal.
Followed steps given below.



1. First, I have opened the reporting service configuration manager.
2. Then, connected to the server using my credentials
3. After that follow the practical sheet 8 to Configure SQL server reporting services.
4. Paginated reports can be created using SQL Server Data Tools or Report Builder, but I have used light weight tool 'Report Builder'. After opening it we need to follow the report creating steps.
5. First, we need to create a Data Source

4.1 Create a Data Source.

Data source is a connection to the source of the data. In here, I have used data in the data warehouse, data source is the data warehouse I have built ('**IT19987880_SpyPlane_DataWarehouse**').

1. Right clicked on the Data Sources → click Added Data Sources...

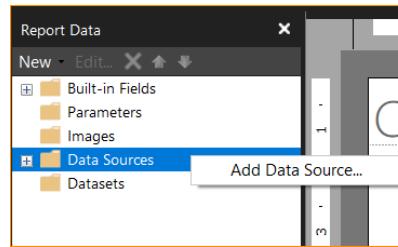


Figure 4.1.1 - Create a Data Source Step 01

2. Provided a data source name as '**IT19987880_SpyPlane_DW**'.
3. Then selected 'Use a connection embedded in my report' → Microsoft SQL Server as the connection type → Click on the 'Build...' button to create the connection → OK

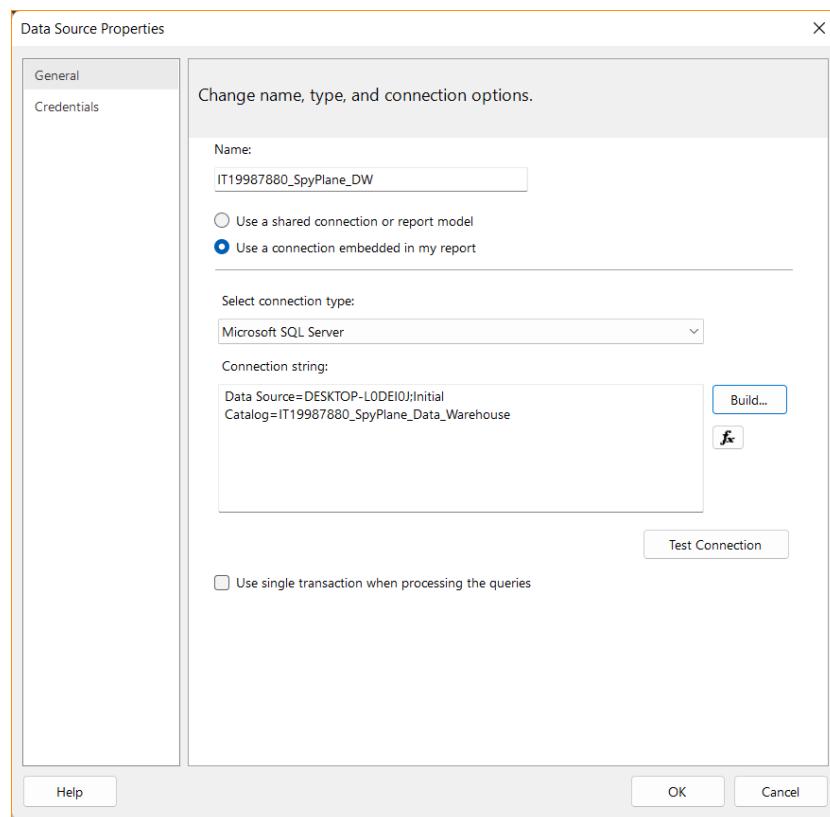


Figure 4.1.2 - Data source properties window

4. Then I could be able to see the created connection under 'Data Sources'.

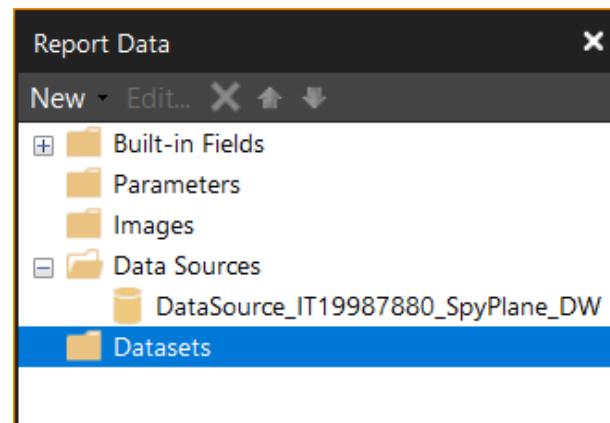


Figure 4.1.3 - Created Data Source file is shown under the data source

5. Next, we need to create a Dataset. Let's move to that step.

4.2 Create a Dataset.

Dataset is the actual data that will be loaded for used the report visualizations.

1. Right click on the Datasets → click Add Dataset...

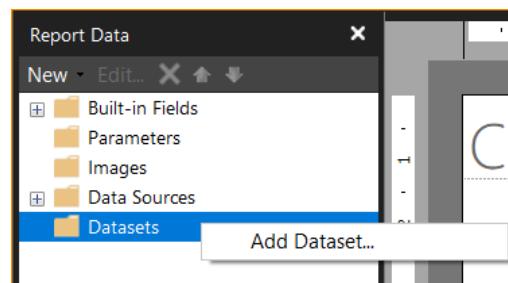


Figure 4.2.1- Right click on the 'Add Dataset'

- 1 In the Query section, provide a dataset name as '**DataSet_IT19987880_SpyPlane_DW**'.

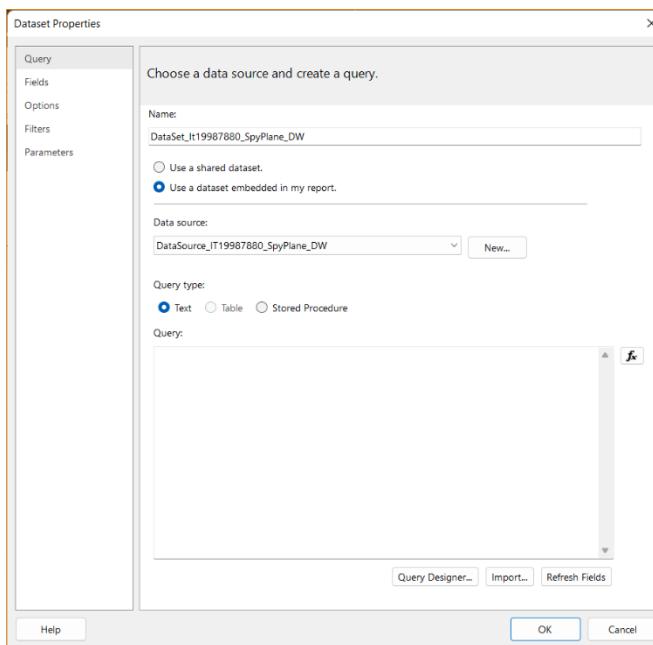


Figure 4.2.3 - Choose the data source

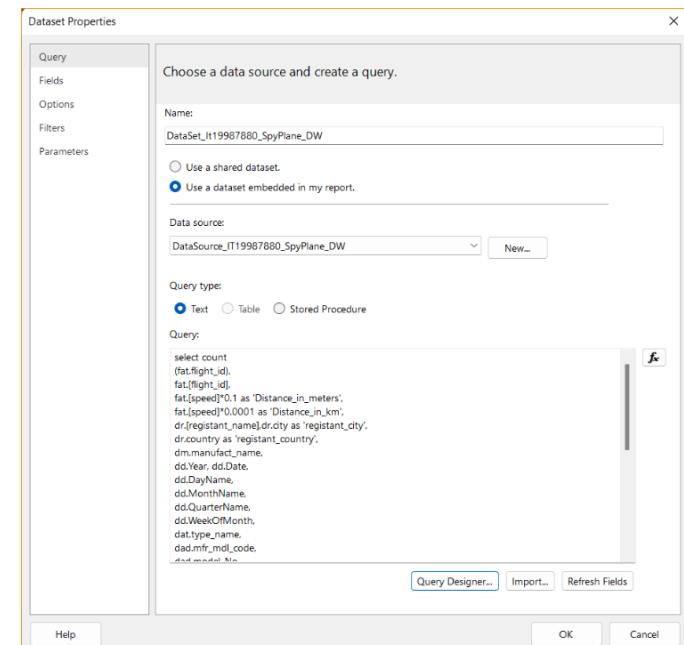


Figure 4.2.4 - Create the query (query has mentioned below)

- 2 Then I could be able to see the created dataset under 'Datasets'.

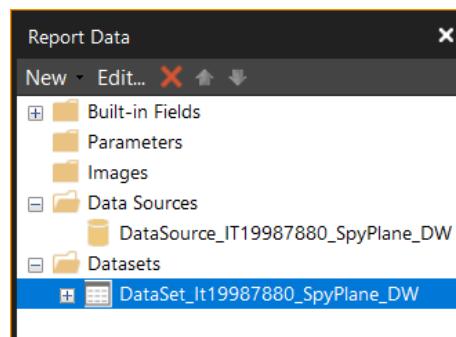


Figure 4.2.5 - Created Data set is shown in under the 'Data sets' folder

- 3 *Query I have executed in 'Query Designer':

```

select count
(fat.flight_id),
fat.[flight_id],
fat.[speed]*0.1 as 'Distance_in_meters',
fat.[speed]*0.0001 as 'Distance_in_km',

```

```

dr.[registant_name],dr.city as 'registant_city',
dr.country as 'registant_country',
dm.manufact_name,
dd.Year, dd.Date,
dd.DayName,
dd.MonthName,
dd.QuarterName,
dd.WeekOfMonth,
dat.type_name,
dad.mfr_mdl_code,
dad.model_No,
dad.serial_number,
dad.year_mfr,
dad.alternate_adshex,
da.agency_name

```

```

from [dbo].[FactAircraftTracking] fat
left outer join [dbo].[DimAircraftData] dad on dad.aircraft_sk=fat.adshex_key
left outer join [dbo].[DimAgency] da on da.agency_sk=fat.agency_key
left outer join [dbo].[DimAircraftType] dat on dat.aircraft_type_sk = dad.type_key
left outer join [dbo].[DimDate] dd on dd.DateKey = fat.date_key
left outer join [dbo].[DimManufacturer] dm on dm.manufact_sk = dad.manufact_key
left outer join [dbo].[DimRegistant] dr on dr.registant_sk = dad.registant_key
group by fat.flight_id,fat.speed*0.1,fat.speed*0.0001 , dr.registant_name,dr.city ,
dr.country,dm.manufact_name, dd.Year, dd.Date,dd.DayName,dd.MonthName,dd.QuarterName,
dd.WeekOfMonth,dat.type_name, dad.mfr_mdl_code,dad.model_No,dad.serial_number,
dad.year_mfr,dad.alternate_adshex,da.agency_name

```

4 Then, Create the visualization in my report.

4.3 Report 1: Report with a matrix

1. Go to 'Insert' tab on the ribbon of the Report Builder → select Table → Table Wizard... → to open the New Table or Matrix Wizard
2. Choose the '**DataSet_It19987880_SpyPlaneDW**' which was created as the dataset → next
3. At the point of selecting fields for Row groups and Column groups. I have drag and drop fields as given below.

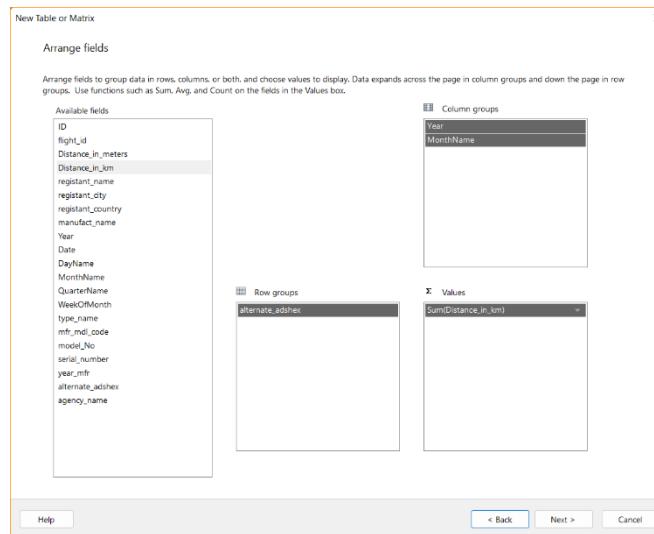


Figure 4.3.1- Arrange fields

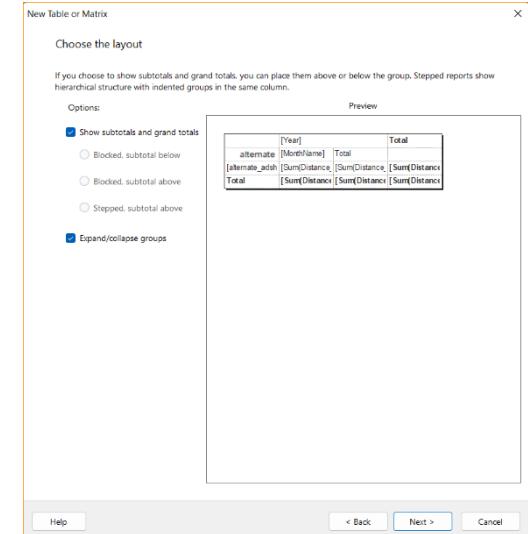


Figure 4.3.2 - Choose the layout

4. After that you can finished review the rest windows in the wizard.
5. Then, you could be able to see a matrix inserted in to the report body. I have provided a suitable report title and design the look of the report like the picture given below.

	[Year]	Total
alternate adshex	[MonthName]	Total
[alternate_adshex]	[Sum(Distance_in_km)]	[Sum(Distance_in_km)]
Total	[Sum(Distance_in_km)]	[Sum(Distance_in_km)]

Figure 4.3.3 - Matrix report body(Aircraft - Wise Tracked Distance Report)

Since I wanted to limit (filter) data based on the year to keep the report clean and less crowded. I have filtered the dataset. For that, right click on the data set '**DataSet_It19987880_SpyPlaneDW**' → select 'Dataset Properties' → Go to 'Filters section' → click on 'Add' button → fill fields like given below → OK

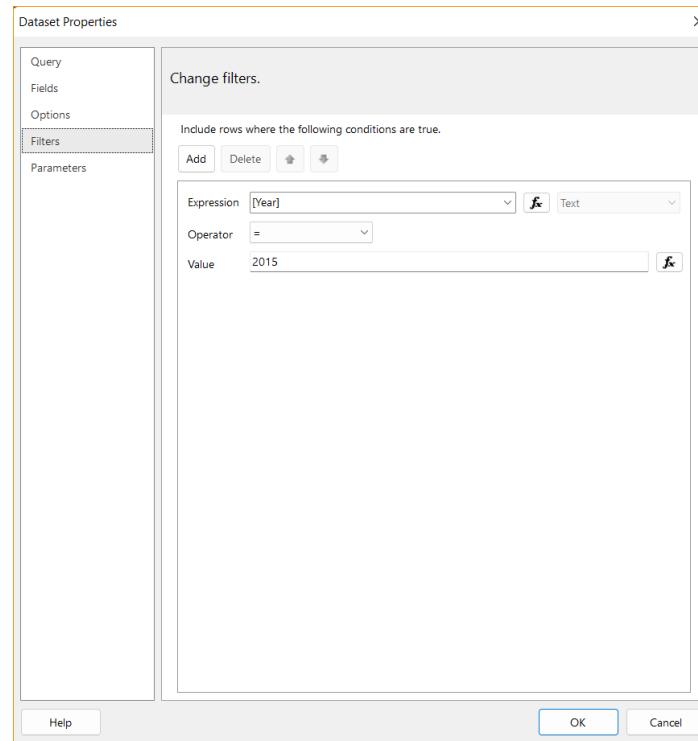
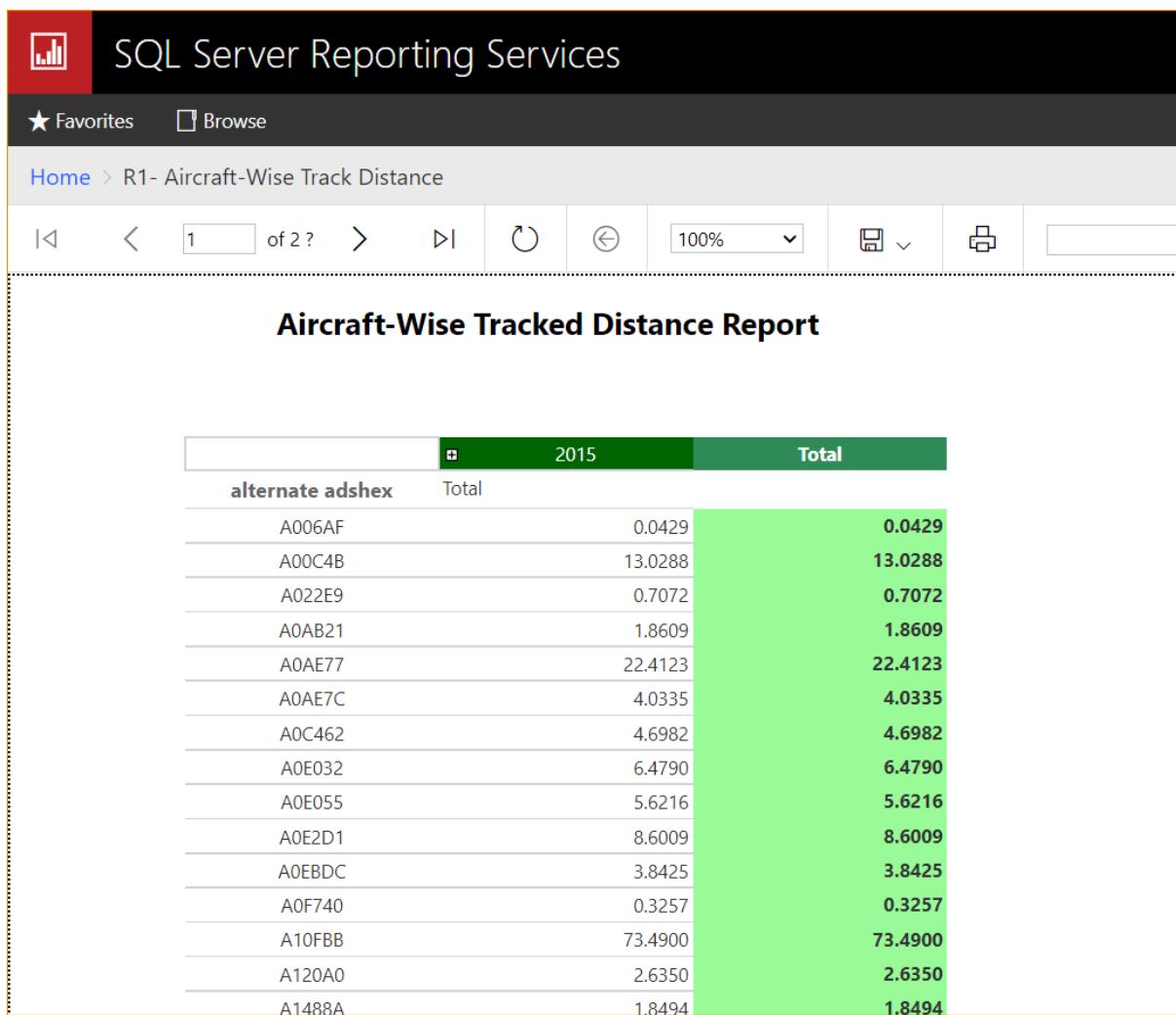


Figure 4.3.4 - Add a filter (Year = 2015)

6. After we run the report via the Portal of SSRS we can have a report like given above.



The screenshot shows the SQL Server Reporting Services portal interface. The title bar reads "SQL Server Reporting Services". Below it, there are navigation links for "Favorites" and "Browse". The breadcrumb navigation shows "Home > R1- Aircraft-Wise Track Distance". The main content area has a title "Aircraft-Wise Tracked Distance Report". Below the title is a table with the following data:

alternate adshex	Total	2015
A006AF	0.0429	0.0429
A00C4B	13.0288	13.0288
A022E9	0.7072	0.7072
A0AB21	1.8609	1.8609
A0AE77	22.4123	22.4123
A0AE7C	4.0335	4.0335
A0C462	4.6982	4.6982
A0E032	6.4790	6.4790
A0E055	5.6216	5.6216
A0E2D1	8.6009	8.6009
A0EBDC	3.8425	3.8425
A0F740	0.3257	0.3257
A10FBB	73.4900	73.4900
A120A0	2.6350	2.6350
A1488A	1.8494	1.8494

Figure 4.3.5 - Created report is shown in SSRS portal

4.4 Report 2: Parameterized Report

1. First, we have to followed 4.5.2 steps, then, we can continue with the steps below to create a Parameterized Report
2. Go to 'Insert' tab on the ribbon of the Report Builder → select Table → Table Wizard... → to open the New Table or Matrix Wizard
3. Chose the '**IT19987880_SpyPlane_DW**' which was created as the dataset → next
4. At the point of selecting fields for Row groups and Column groups. I have drag and drop fields as given below.

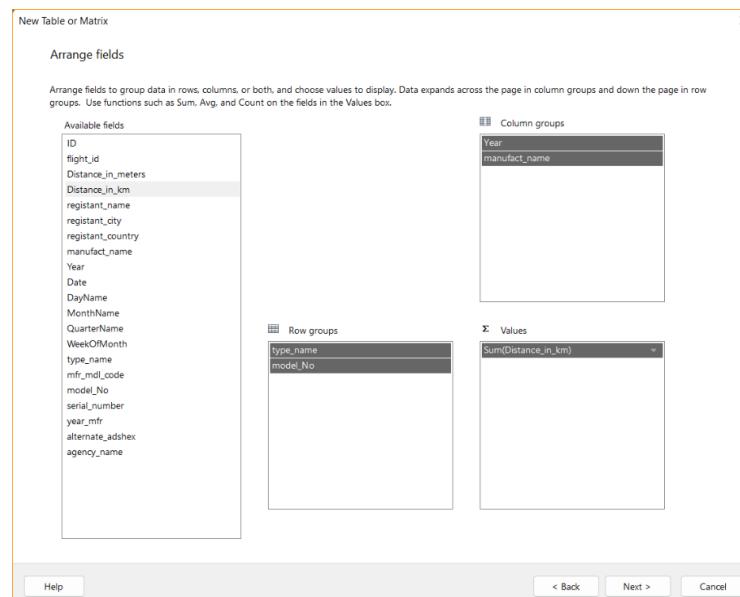


Figure 4.4.4.1 - Add fields

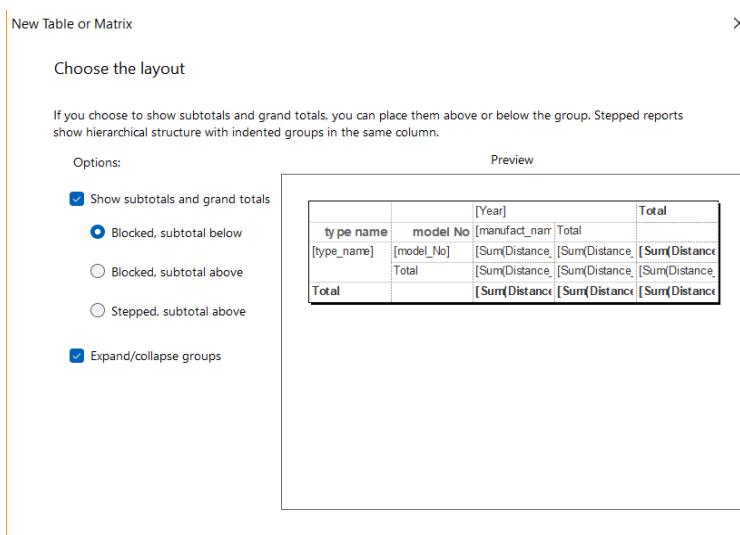
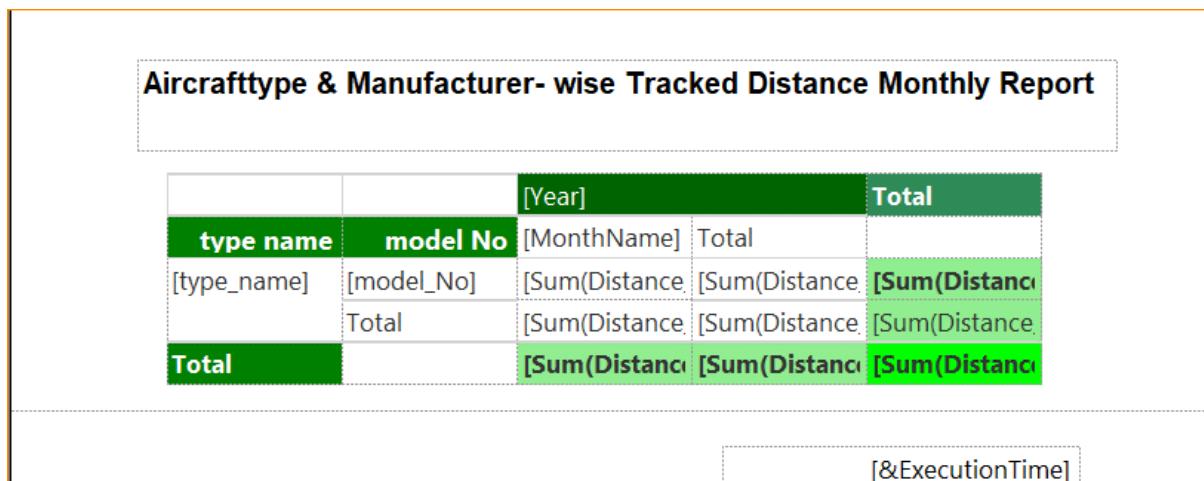


Figure 4.4.4.2 – Choose the layouts

6. After that you can finished review the rest windows in the wizard

7. Then, you could be able to see a matrix inserted in to the report body. I have provided a suitable report title and design the look of the report like the picture given below.



Aircrafttype & Manufacturer- wise Tracked Distance Monthly Report

		[Year]	Total	Total
type name	model No	[MonthName]	Total	Total
[type_name]	[model_No]	[Sum(Distance)]	[Sum(Distance)]	[Sum(Distance)]
	Total	[Sum(Distance)]	[Sum(Distance)]	[Sum(Distance)]
Total		[Sum(Distance)]	[Sum(Distance)]	[Sum(Distance)]

[&ExecutionTime]

Figure 4.4.7.1 - Report body

8. Let's add parameters
 - a. We need to add a where' clause to the dataset that we created as '**IT19987880_SpyPlane_DW**'.
 - b. Right click on the dataset '**IT19987880_SpyPlane_DW**' → select 'Dataset Properties'
 - c. Change the query adding the below where clause.
`where dat.aircraft_type_sk = @aircraft_type AND dad.model_No = @model_no`
 - d. Then you can have a query like given below.

```

select count
(fat.flight_id),
fat.[flight_id],
fat.[speed]*0.1 as 'Distance_in_meters',
fat.[speed]*0.0001 as 'Distance_in_km',
dr.[registant_name],dr.city as 'registant_city',
dr.country as 'registant_country',

```

```

dm.manufact_name,
dd.Year, dd.Date,
dd.DayName,
dd.MonthName,
dd.QuarterName,
dd.WeekOfMonth,
dat.type_name,
dad.mfr_mdl_code,
dad.model_No,
dad.serial_number,
dad.year_mfr,
dad.alternate_adshex,
da.agency_name

from [dbo].[FactAircraftTracking] fat
left outer join [dbo].[DimAircraftData] dad on dad.aircraft_sk=fat.adshex_key
left outer join [dbo].[DimAgency] da on da.agency_sk=fat.agency_key
left outer join [dbo].[DimAircraftType] dat on dat.aircraft_type_sk = dad.type_key
left outer join [dbo].[DimDate] dd on dd.DateKey = fat.date_key
left outer join [dbo].[DimManufacturer] dm on dm.manufact_sk = dad.manufact_key
left outer join [dbo].[DimRegistant] dr on dr.registant_sk = dad.registant_key

where dat.aircraft_type_sk = @aircraft_type_sk AND dad.model_No = @model_No

group by fat.flight_id,fat.speed]*0.1 ,fat.speed]*0.0001 , dr.registant_name,dr.city ,
dr.country,dm.manufact_name, dd.Year,
dd.Date,dd.DayName,dd.MonthName,dd.QuarterName, dd.WeekOfMonth,dat.type_name,
dad.mfr_mdl_code,dad.model_No,dad.serial_number,
dad.year_mfr,dad.alternate_adshex,da.agency_name

```

Where clause is newly added

9. The parameters will be created under Parameters as below

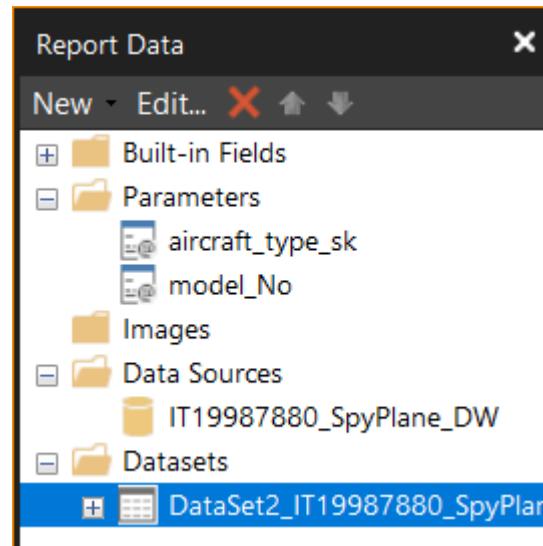


Figure 4.4.9.1- Create two parameters are shown in under the Parameters folder

10. First, Right click on the parameter '**aircraft_type_sk**' → select Parameter Properties

11. In General section, Name is '**aircraft_type_sk**' and Prompt is also '**aircraft_type_sk**'
 → Change the Prompt value to '**aircraft_type_sk**'.

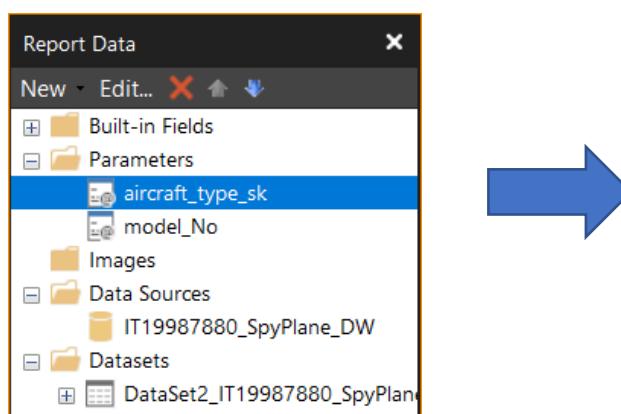


Figure 4.4.11.1 - Click on the 'aircraft_type_sk'

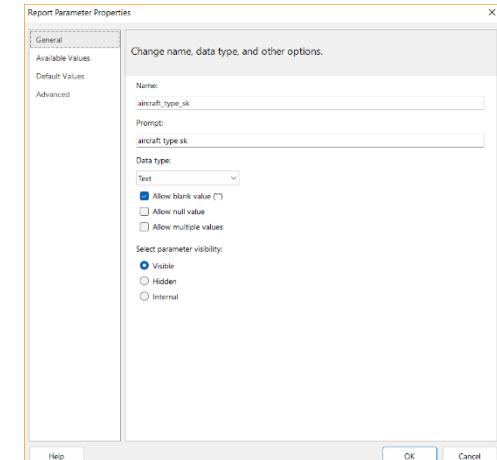


Figure 4.4.11.2 - Get the properties window

12. Then, Right click on the parameter '**model_No**' → select Parameter Properties

13. In General section, Name is 'Model' and Prompt is also '**model_No**' → Change the Prompt value to 'model'.

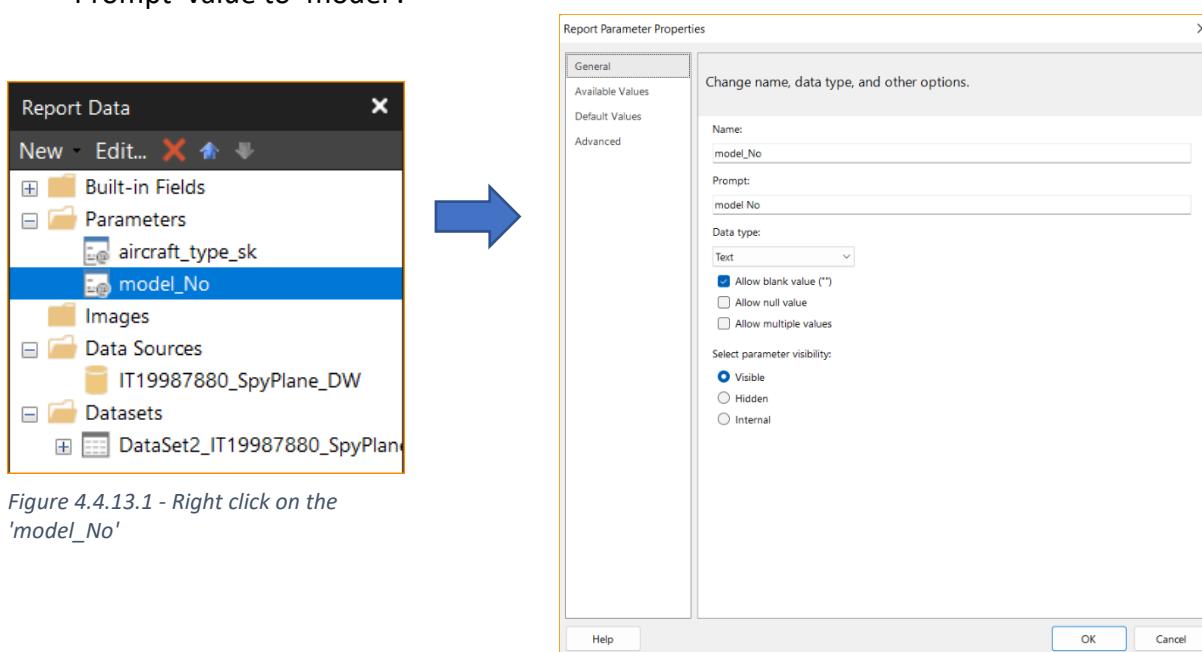


Figure 4.4.13.1 - Right click on the 'model_No'

Figure 4.4.13.2 - open the properties window

14. Then, we have to add the lists of values to the parameters

15. Right click on the Datasets → click Add Dataset... → dataset properties window will open

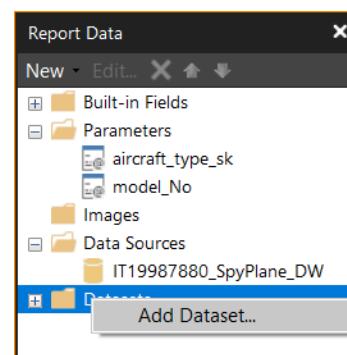


Figure 4.4.15.1 -Add another dataset

1. In the Query section, provide a dataset name as '**Dataset3_Aircrafttype**'
2. Then, select 'Use a dataset embedded in my report' → Select the data source I have created earlier, '**IT19987880_SpyPlane_DW**' → select 'Text' as the Query type
3. Click on the Query Designer... → Paste the query we executed for according to my need → we can execute the query using '!' in the query window. After it ran without any issue → Review what is there in other sections Fields, Options, Filters and Parameters → OK

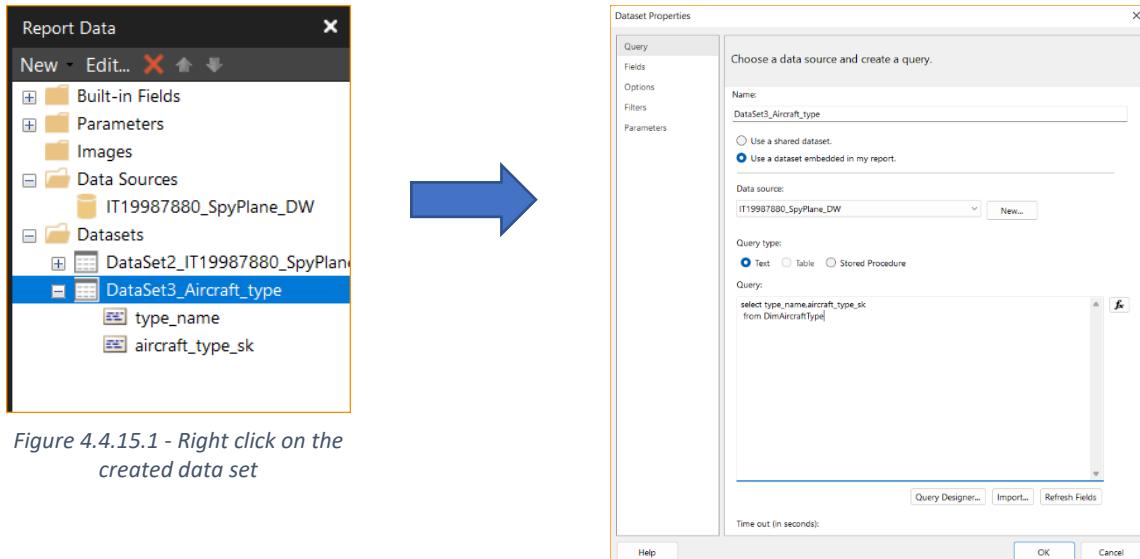


Figure 4.4.15.1 - Right click on the created data set

Figure 4.4.15.2 - open the properties window

16. Then again, Right click on the Datasets → click Add Dataset... → dataset properties window will open

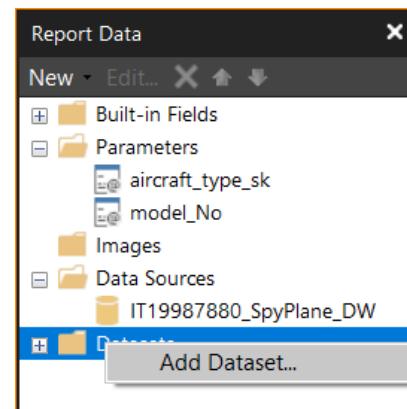


Figure 4.4.16.1 -Add another data set

1. In the Query section, provide a dataset name as 'model_list'.
2. Then, select 'Use a dataset embedded in my report' → Select the data source I have created earlier, 'IT19987880_SpyPlane_DW' → select 'Text' as the Query type
3. Click on the Query Designer... → Paste the query we executed for according to my need → we can execute the query using '!' in the query window. After it ran without any issue → Review what is there in other sections Fields, Options, Filters and Parameters → OK

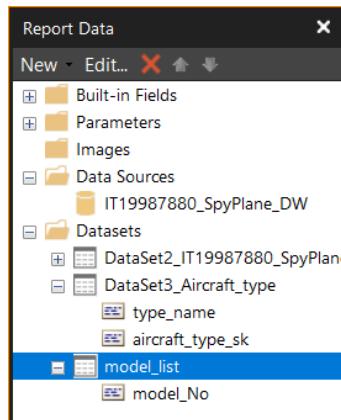


Figure 4.4.16.3.1 - Right click on the created dataset 'model_list'

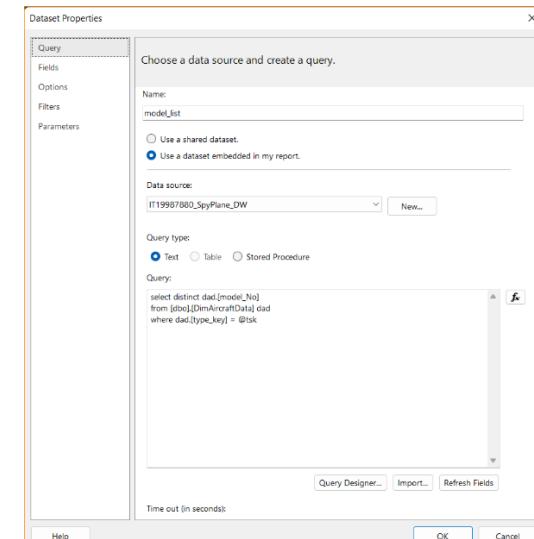


Figure 4.4.16.3.2 - open the properties window and paste the query

- Click on the Query Designer... → Paste the query we executed for according to my need → we can execute the query using ‘!’ in the query window. After it ran without any issue → Review what is there in other sections Fields, Options, Filters and Parameters → OK

17. Right click on the parameter ‘aircraft_type_sk’ → select Parameter Properties

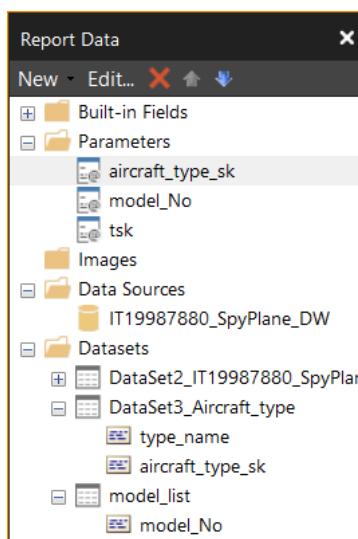


Figure 4.4.17.0.1 - Right click on the 'aircraft_type_sk'

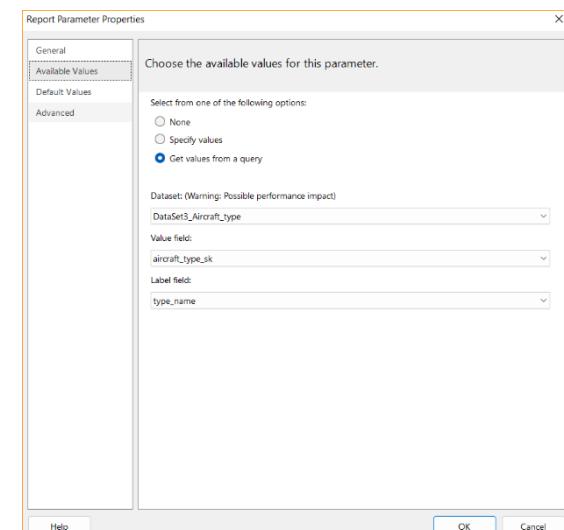


Figure 4.4.17.0.2- Open the properties window

- Go to Available Values section → select Get values from a query → select ‘Aircrafttype_list’ for the Dataset → ‘aircraft_type_sk’ for the Value field → ‘type_name’ for the Name field → OK

19. Right click on the parameter 'model_no' → select Parameter Properties

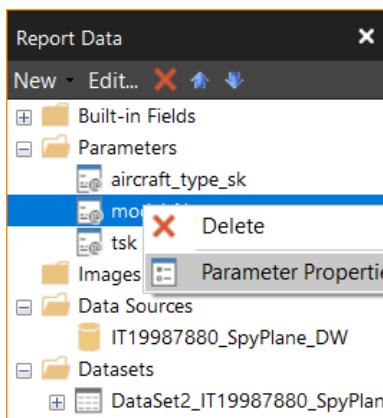


Figure 4.4.17.2.1 - Right click on the 'model_No'

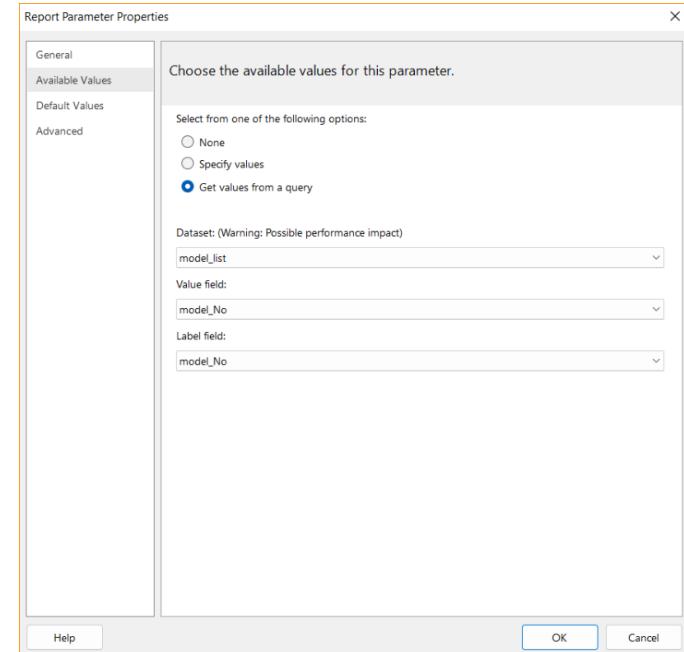


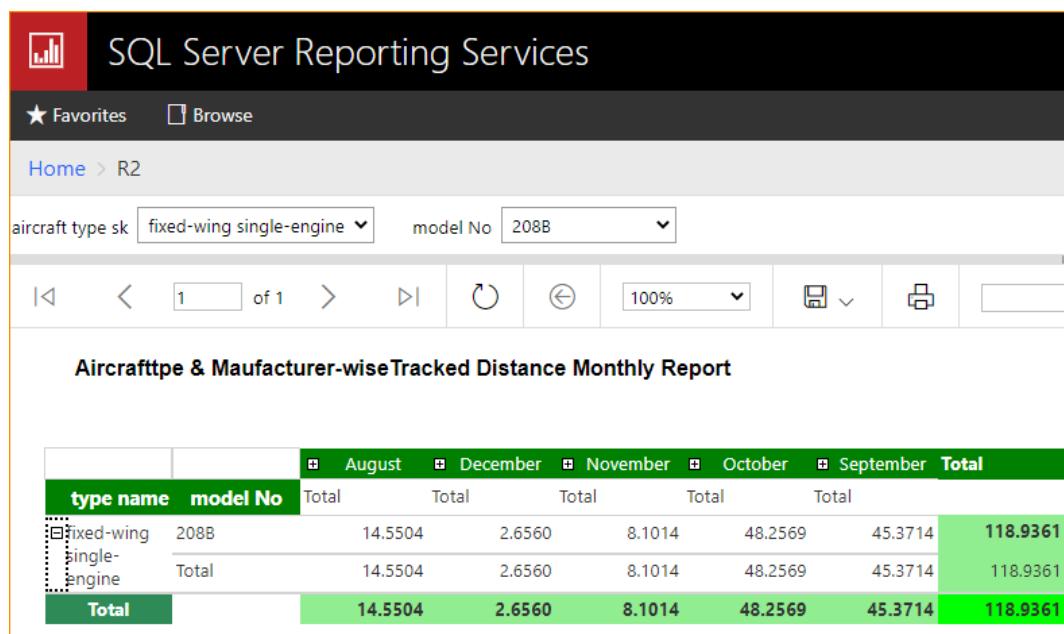
Figure 4.4.17.2.2 - open the properties window

20. Go to Available Values section → select

Get values from a query → select 'model_list' for the Dataset → 'model_No' for the Value field → 'model_No' for the Name field → OK

21. Save the report → execute

Then, we can have a report like images given below.



type	name	model No	August	December	November	October	September	Total
fixed-wing	208B		14.5504	2.6560	8.1014	48.2569	45.3714	118.9361
single-engine		Total	14.5504	2.6560	8.1014	48.2569	45.3714	118.9361
	Total		14.5504	2.6560	8.1014	48.2569	45.3714	118.9361

Figure 4.4.21.1 - Report 2 in SSRS portal

22. Let's create a chart.

1. Go to the Insert tab on the ribbon of the Report Builder → select Chart → Chart Wizard...
2. Select the 'Dataset_Aircraft_Data' you created as the dataset → Next
3. Select the suitable chart → Next → Drag and drop fields like given below → next

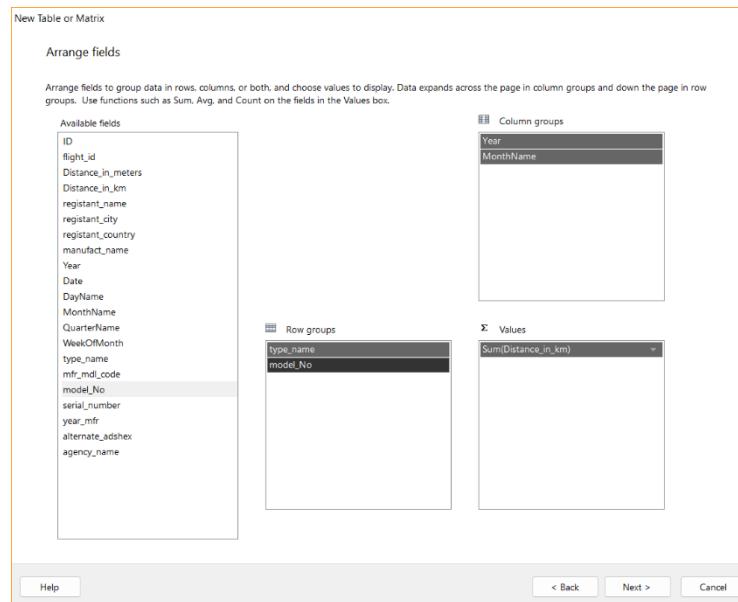


Figure 4.4.21.3.1 - insert fields

4. Review the chart → Finish
5. Then you could able to see the chart
6. I have done some modifications to the chart and saved
7. Then I could be able to have a chart like given in the image below

SQL Server Reporting Services

Favorites Browse

Home > R2

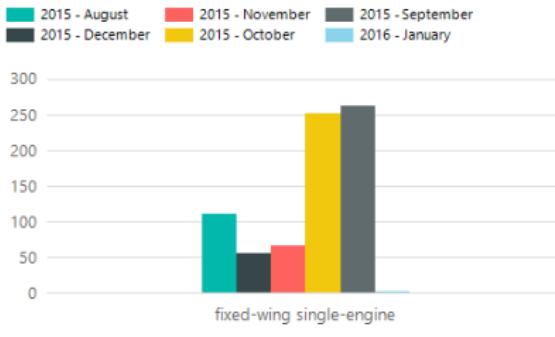
aircraft type sk fixed-wing single-engine model No 182T

|< 1 of 1 > | Refresh | Print | Find | Next

Aircrafttype & Maufacturer-wise Tracked Distance Monthly Report

		August	December	January	November	October	September	Total
type name	model No	Total	Total	Total	Total	Total	Total	
fixed-wing	Total	111.8702	56.1917	0.3322	66.7392	252.4841	263.4226	751.0400
	Total	111.8702	56.1917	0.3322	66.7392	252.4841	263.4226	751.0400

Monthly Tracking Distance



Month	Distance
August	111.8702
November	56.1917
September	0.3322
December	66.7392
October	252.4841
January	263.4226
February	751.0400

Figure 4.4.21.7.1 - Open parameterized report in SSRS portal

4.5 Report 3: SSRS drill-down report

First, we have to followed 4.5.2 steps, then, we can continue with the steps below to create a SSRS drill-down report

1. Go to 'Insert' tab on the ribbon of the Report Builder → select Table → Table Wizard... → to open the New Table or Matrix Wizard
2. Chose the 'DataSet_Aircraft_Data' which was created as the dataset → next
3. At the point of selecting fields for Row groups and Column groups. I have drag and drop fields as given below.

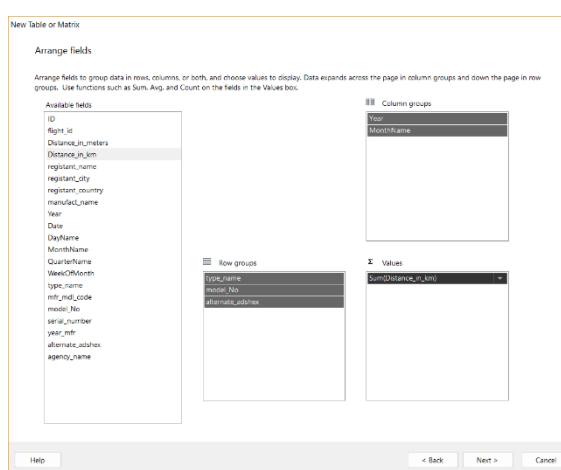


Figure 4.5.3.1 - Apply fields

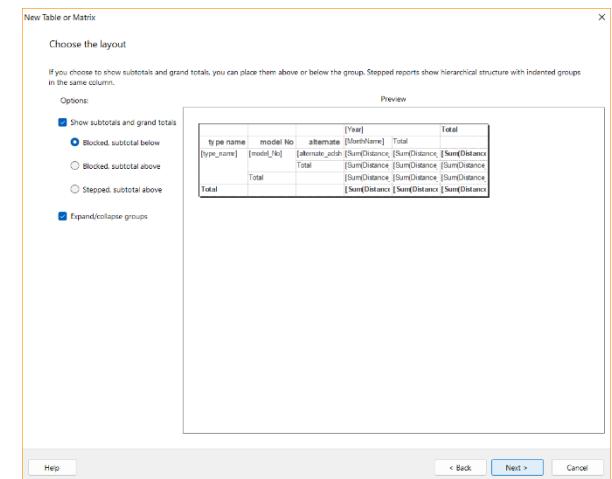


Figure 4.5.3.2 - Choose the layout

After that you can finished review the rest windows in the wizard.

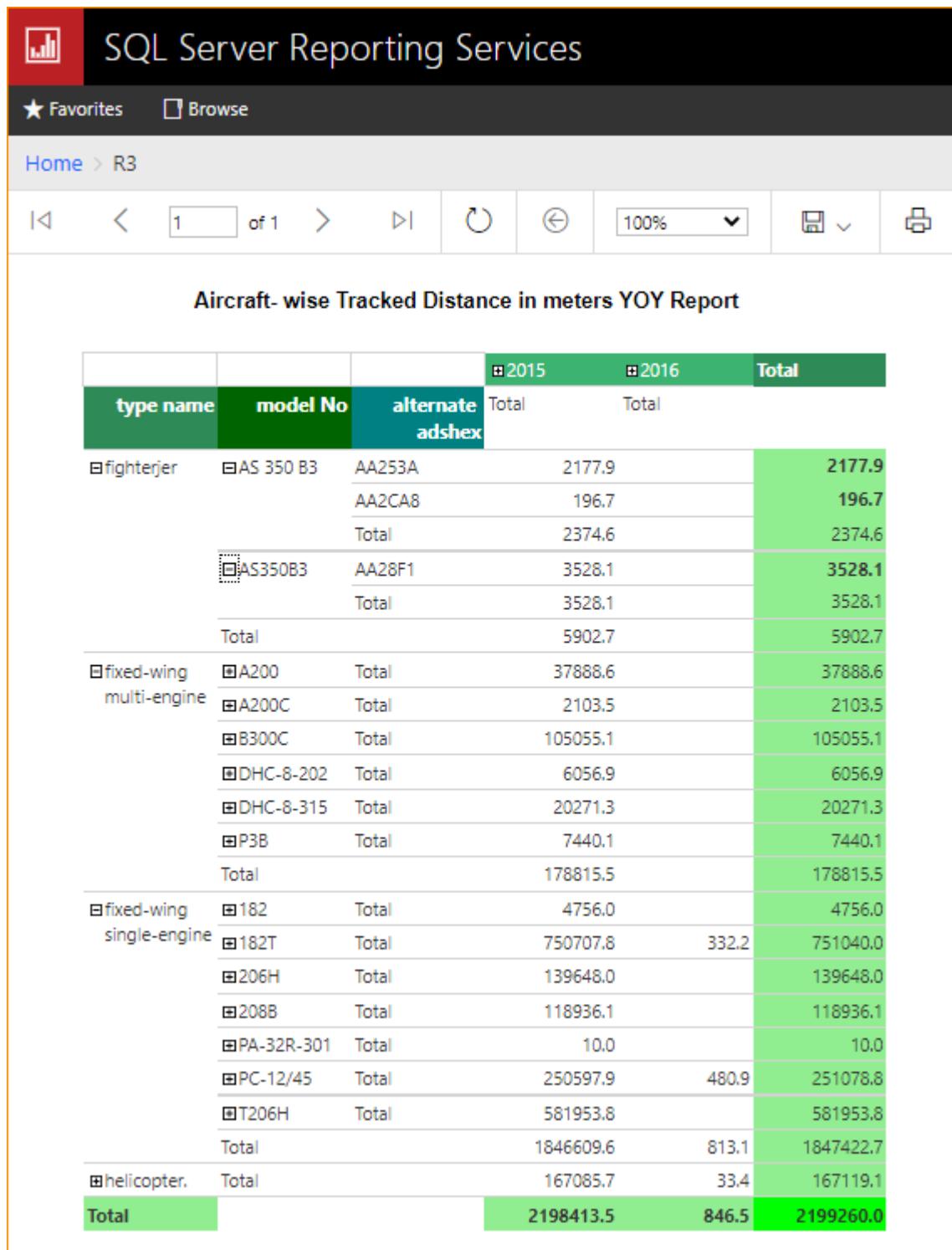
4. Then, you could be able to see a matrix inserted into the report body. I have provided a suitable report title and design the look of the report like the picture given below.

Aircraft- wise Tracked Distance in meters YOY Report					
			[Year]		Total
type name	model No	alternate ads	[MonthName]	Total	
[type_name]	[model_No]	[alternate_adshx]	[Sum(Distance_in_km)]	[Sum(Distance_in_km)]	[Sum(Distance_in_km)]
		Total	[Sum(Distance_in_km)]	[Sum(Distance_in_km)]	[Sum(Distance_in_km)]
	Total		[Sum(Distance_in_km)]	[Sum(Distance_in_km)]	[Sum(Distance_in_km)]
Total			[Sum(Distance_in_km)]	[Sum(Distance_in_km)]	[Sum(Distance_in_km)]

[&ExecutionTime]

Figure 4.5.4.1 - Drill down report 3 body

5. When we run the report using portal, we can have a report like given below. In here we can see aircraft-wise tracked distance in meters. Moreover, we can do drill-down or roll-up.



The screenshot shows a report titled "Aircraft- wise Tracked Distance in meters YOY Report" from the SQL Server Reporting Services portal. The report is displayed in a grid format with columns for type name, model No, alternate adshex, and total tracked distance for the years 2015 and 2016. The data is organized by aircraft type, model, and specific tracking codes. The report includes several rows for different aircraft types, such as fighter jets and fixed-wing multi-engine planes, with sub-rows for specific models and tracking codes. The total tracked distance for each row is also provided.

type name	model No	alternate adshex	2015	2016	Total
			Total	Total	
fighterjer	AS 350 B3	AA253A	2177.9		2177.9
		AA2CA8	196.7		196.7
		Total	2374.6		2374.6
	AS350B3	AA28F1	3528.1		3528.1
		Total	3528.1		3528.1
	Total		5902.7		5902.7
fixed-wing multi-engine	A200	Total	37888.6		37888.6
	A200C	Total	2103.5		2103.5
	B300C	Total	105055.1		105055.1
	DHC-8-202	Total	6056.9		6056.9
	DHC-8-315	Total	20271.3		20271.3
	P3B	Total	7440.1		7440.1
	Total		178815.5		178815.5
fixed-wing single-engine	I82	Total	4756.0		4756.0
	I82T	Total	750707.8	332.2	751040.0
	206H	Total	139648.0		139648.0
	208B	Total	118936.1		118936.1
	PA-32R-301	Total	10.0		10.0
	PC-12/45	Total	250597.9	480.9	251078.8
	T206H	Total	581953.8		581953.8
Total			1846609.6	813.1	1847422.7
helicopter.	Total		167085.7	33.4	167119.1
Total			2198413.5	846.5	2199260.0

Figure 4.5.5.1 -Drill down report is Show in SSRS portal

4.6 Report 4: SSRS drill-through report

4.6.1 Level 1- Main report

First, we have to followed 4.1 & 4.2 steps, then, we can continue with the steps below to create a SSRS drill-through report

1. Then, go to the 'Insert' tab on the ribbon of the Report Builder → select Chart → Chart Wizard...
2. Chose the '**IT19987880_SpyPlane_DW**' you created as the dataset → Next
3. Select Column chart → Next
4. Drag and drop fields given below

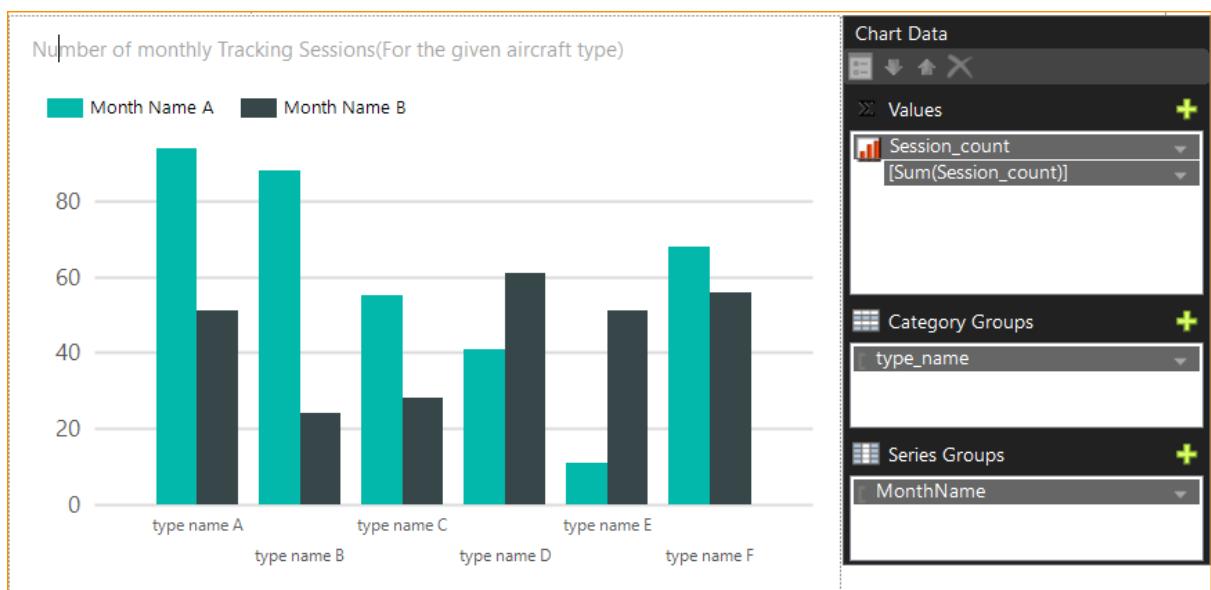


Figure 4.5.1.4.1 - Number of monthly tracking Sessions(for the given aircraft type)

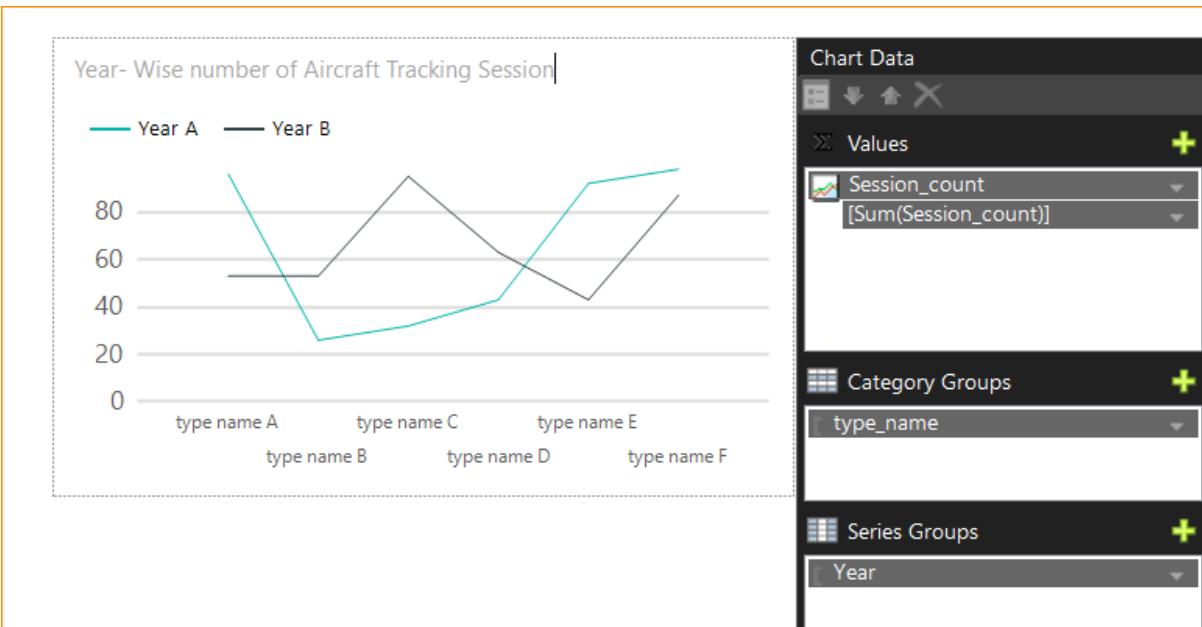


Figure 4.5.1.4.2 - Year wise number of Aircraft Tracking Session

5. After Review the chart → click Finish
6. Then you can see charts like given above
7. After that, Provide a suitable chart title and description

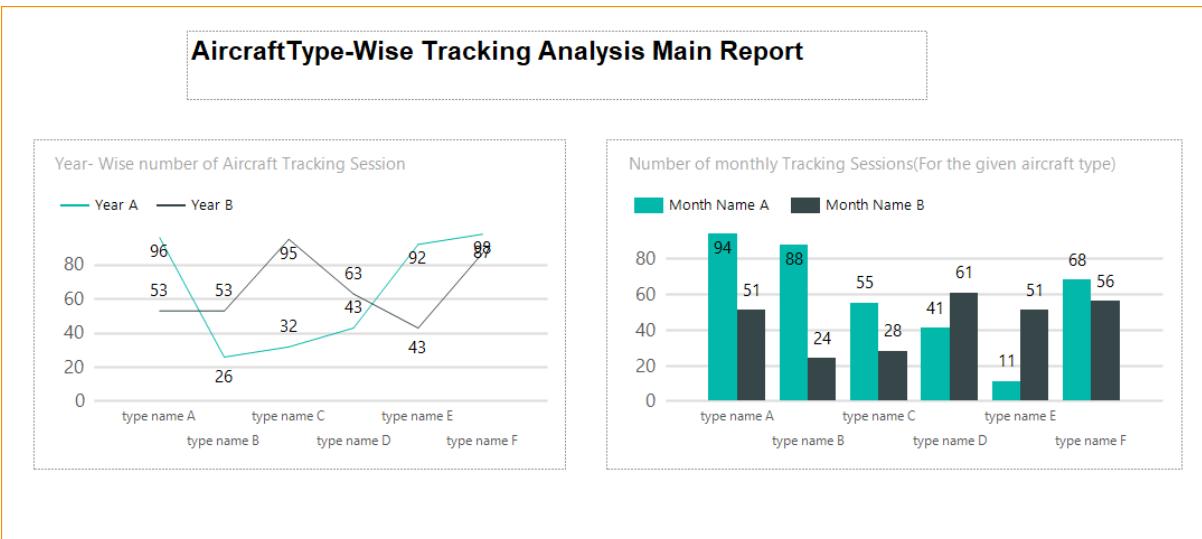


Figure 4.5.1.7.1 - Aircraft Type wise tracking analysing main report

4.6.2 Level 2- Sub report

Then again, we have to followed **4.1** and **4.2** steps, and we can continue with the steps given above to create the sub report which is going to connect.

11. again, go to the 'Insert' tab on the ribbon of the Report Builder → select Chart → Chart Wizard...
12. Chose the '**IT19987880_SpyPlane_DW**' you created as the dataset → Next
13. Select Column chart → Next
14. Drag and drop fields given below

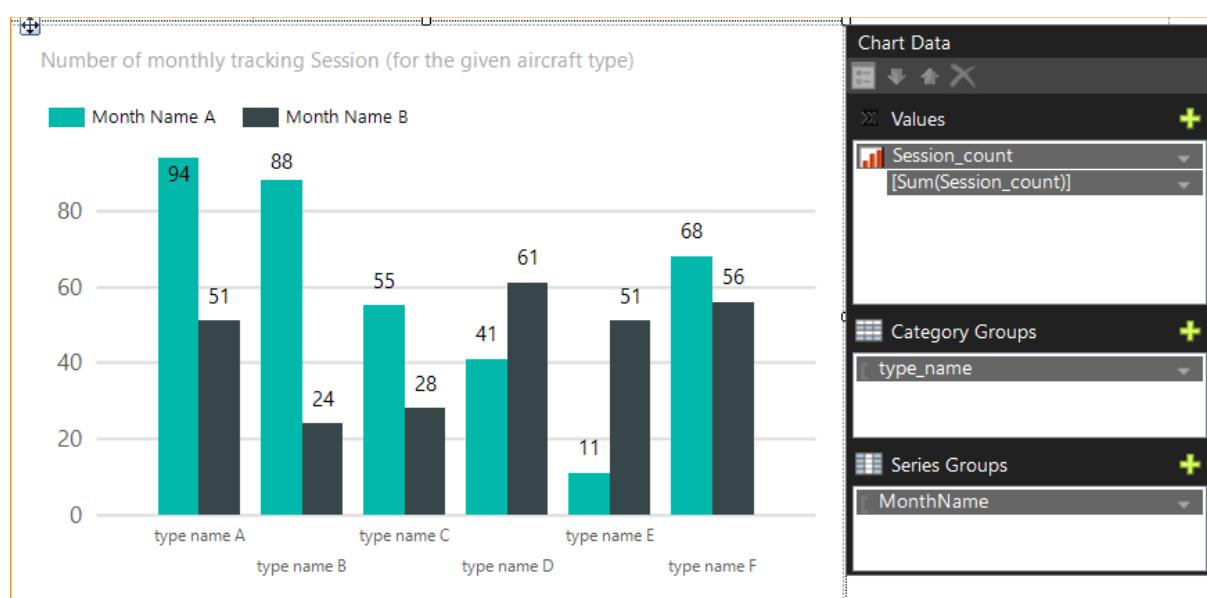


Figure 4.5.2.14.1 - Number of monthly tracking session (for the given aircraft type)

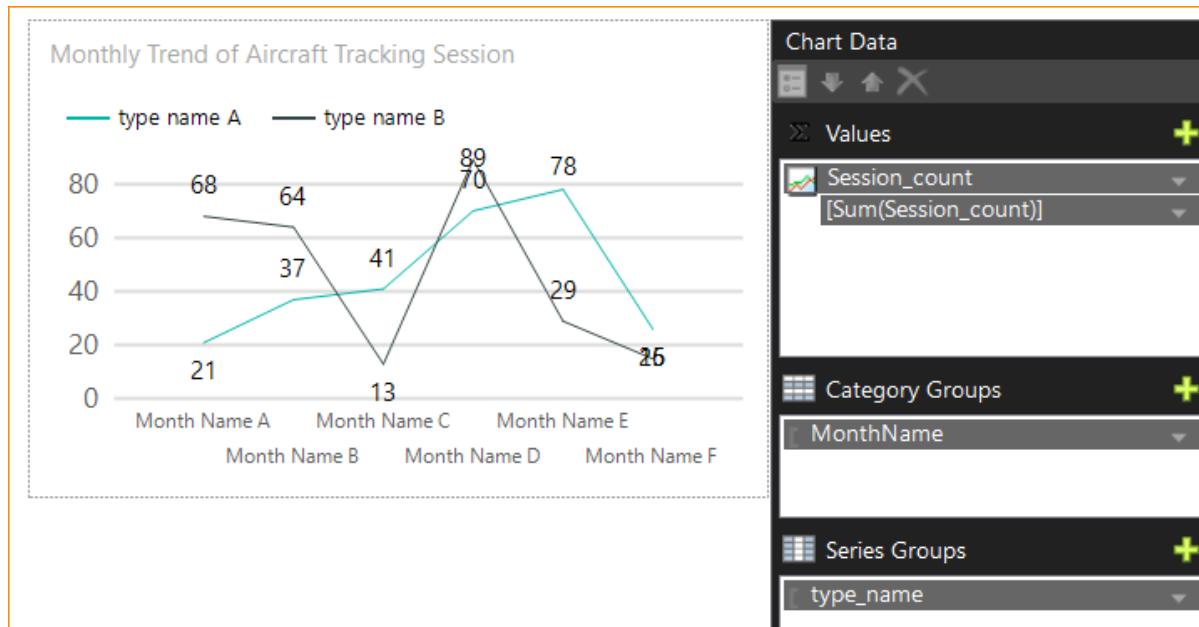


Figure 4.5.2.11.1 - Monthly trend of aircraft tracking session

12. After Review the chart → click Finish

13. Then you can see charts like given above

14. After that, Provide a suitable chart title and description.

15. Then, we have to add parameters to the second dataset.

15.1 We need to add a 'where' clause to the dataset that we created as 'DataSet_Aircraft_Data'.

15.2 Right click on the dataset 'DataSet_Aircraft_Data' → select 'Dataset Properties'.

15.3 Change the query adding the below where clause.

where dat.type_name = @atname AND dd.Year = @year

Then you can have a query like given below.

```
select count
(fat.flight_id) as 'Session count',
fat.[flight_id],
fat.[speed]*0.1 as 'Distance_in_meters',
fat.[speed]*0.0001 as 'Distance_in_km',
dr.[registant_name],dr.city as 'registant_city',
dr.country as 'registant_country',
dm.manufact_name,
dd.Year, dd.Date,
```

dd.DayName,
 dd.MonthName,
 dd.QuarterName,
 dd.WeekOfMonth,
 dat.type_name,
 dad.mfr_mdl_code,
 dad.model_No,
 dad.serial_number,
 dad.year_mfr,
 dad.alternate_adshex,
 da.agency_name

```

from [dbo].[FactAircraftTracking] fat
left outer join [dbo].[DimAircraftData] dad on dad.aircraft_sk=fat.adshex_key
left outer join [dbo].[DimAgency] da on da.agency_sk=fat.agency_key
left outer join [dbo].[DimAircraftType] dat on dat.aircraft_type_sk = dad.type_key
left outer join [dbo].[DimDate] dd on dd.DateKey = fat.date_key
left outer join [dbo].[DimManufacturer] dm on dm.manufact_sk = dad.manufact_key
left outer join [dbo].[DimRegistant] dr on dr.registant_sk = dad.registant_key
  
```

where dat.type_name = @atname AND dd.Year = @year

```

group by fat.flight_id,fat.speed]*0.1 ,fat.[speed]*0.0001 , dr.registant_name,dr.city ,
dr.country,dm.manufact_name, dd.Year,
dd.Date,dd.DayName,dd.MonthName,dd.QuarterName, dd.WeekOfMonth,dat.type_name,
dad.mfr_mdl_code,dad.model_No,dad.serial_number,
dad.year_mfr,dad.alternate_adshex,da.agency_name
  
```

The parameters will be created under Parameters as below

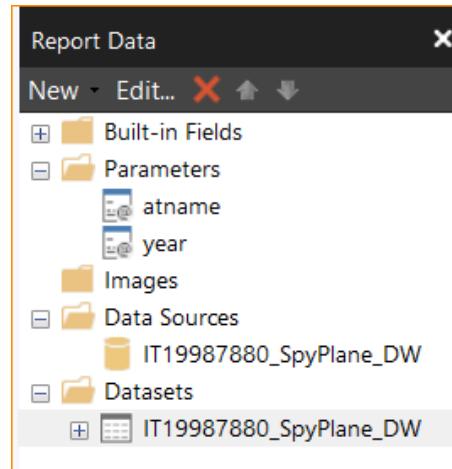
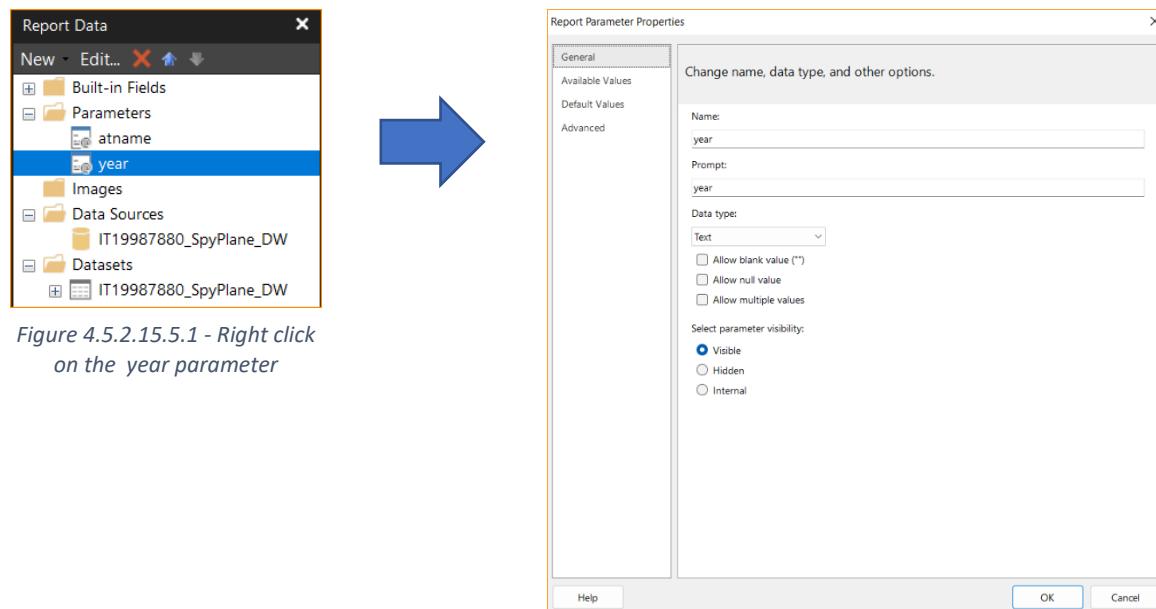


Figure 4.5.2.15.3.1 - created parameters are shown in under the parameters folder

15.4 Right click on the parameter 'year' → select Parameter Properties

15.5 In General section, Name is 'year' and Prompt is also 'year'



Report Data

- New - Edit... X ↑ ↓
- Built-in Fields
- Parameters
 - atname
 - year
- Images
- Data Sources
 - IT19987880_SpyPlane_DW
- Datasets
- IT19987880_SpyPlane_DW

→

Report Parameter Properties

Change name, data type, and other options.

General
Available Values
Default Values
Advanced

Name: year

Prompt: year

Data type: Text

Allow blank value (*)

Allow null value

Allow multiple values

Select parameter visibility:

Visible

Hidden

Internal

Help OK Cancel

Figure 4.5.2.15.5.1 - Right click on the year parameter

Figure 4.5.2.15.5.1 - Set the prompt name

15.6 Right click on the parameter 'atname' → select Parameter Properties

15.7 In General section, Name is 'atname' and Prompt is also 'atname'

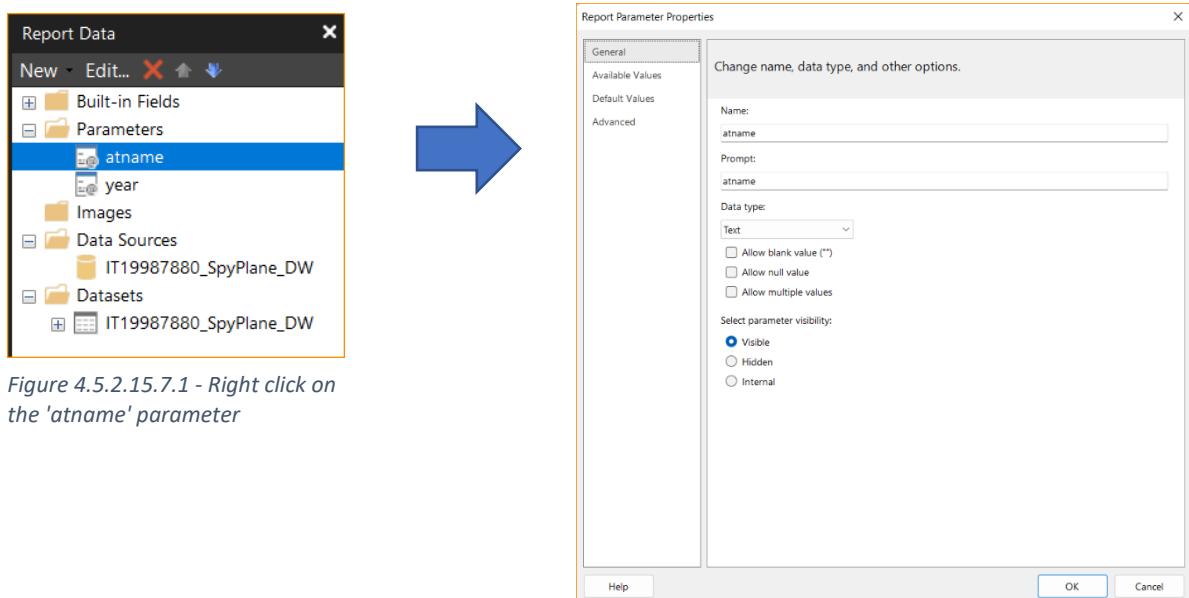


Figure 4.5.2.15.7.1 - Right click on the 'atname' parameter

Figure 4.5.2.15.7.2 - set the prompt name

15.8 Save the report.

Then, again we have to go to the first report

4.6.3 Level 3- Main report Con.

- 15.9 Right click on any column of the column chart in the main report → select Series Properties...→
- 15.10 In Series Properties window → go to Action section → select Go to report
- 15.11 Under, specify a report → select your second level report using the Browse... button → Click on Add button and select 'atname' and 'year' for Name → select '[type_name]' and '[Year]' for Value → OK.

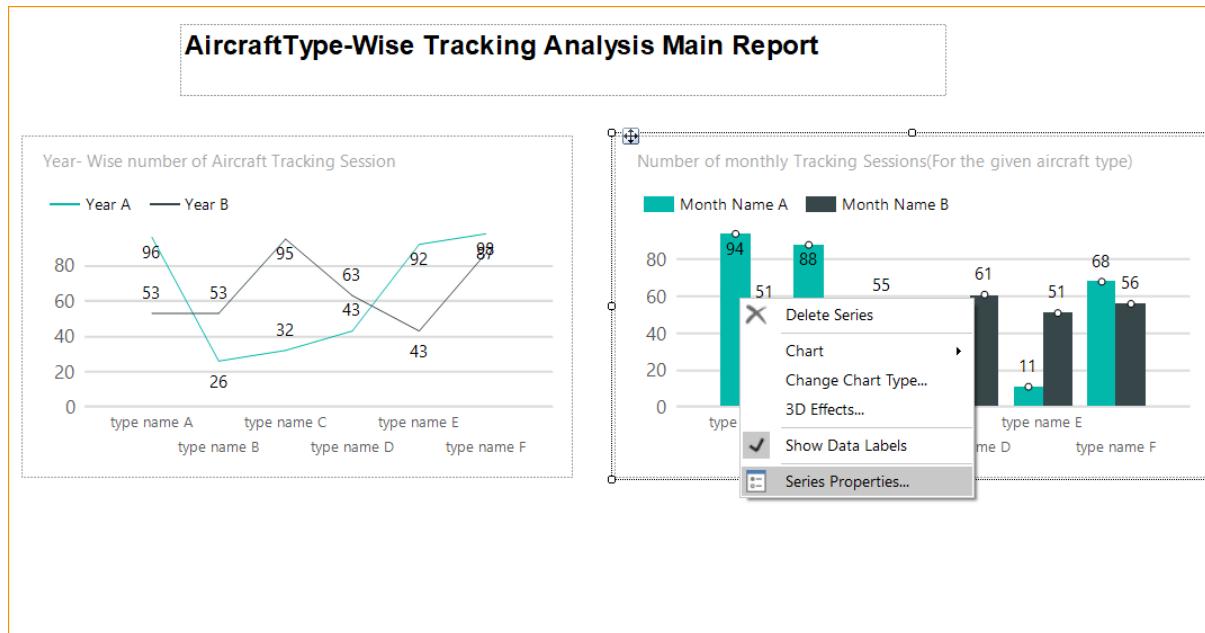


Figure 4.5.3.15.11.1 - Right click on the column and select series properties

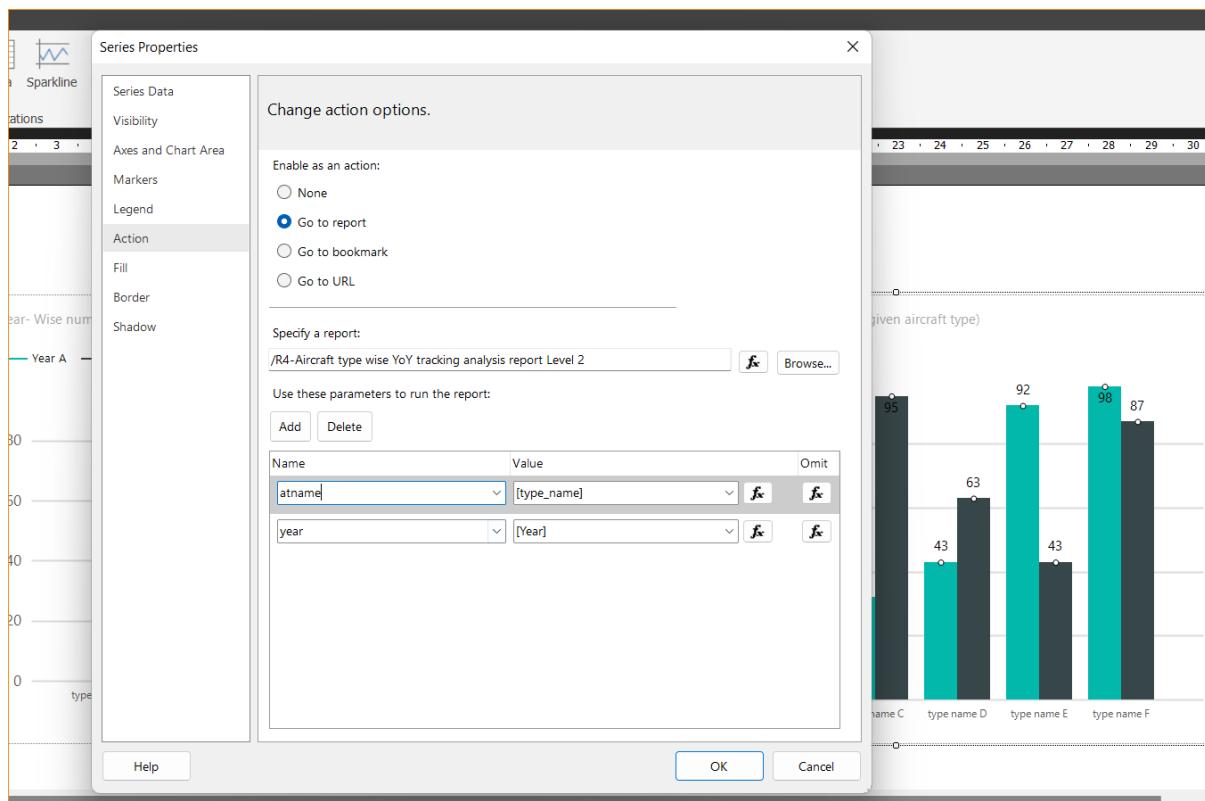


Figure 4.5.3.15.11.2 -Browse the sub report

Add a text box to show some guide line (It shows in red colour)

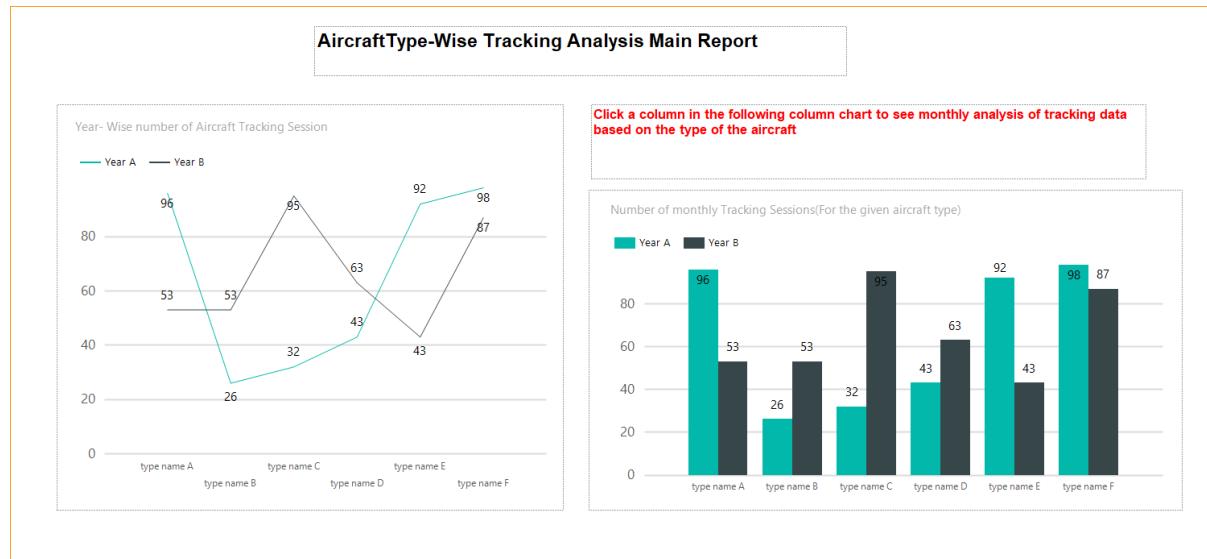


Figure 4.5.3.15.11.3 - Added a duide line in red colour

16. Save the report

4.6.4 Level 4- Sub report Con.

17. Add text box like the image given below to make the report level 2 clearer.

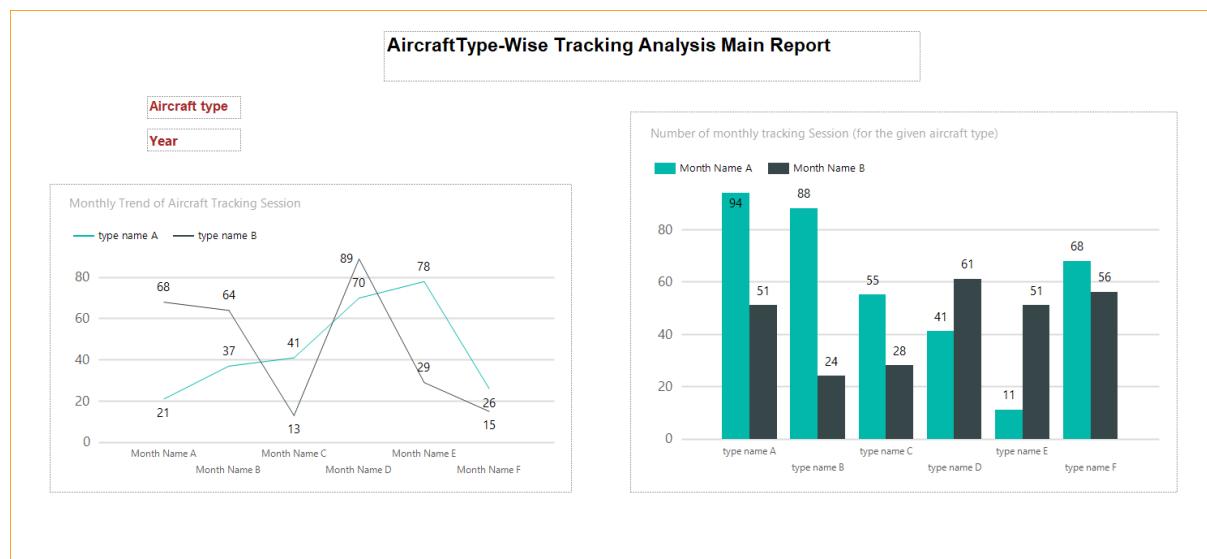


Figure 4.5.4.17.1- Add two test box to show Aircraft type and Year

18. Add another text box like the image given below to show the parameters in the report level 2 clearer.

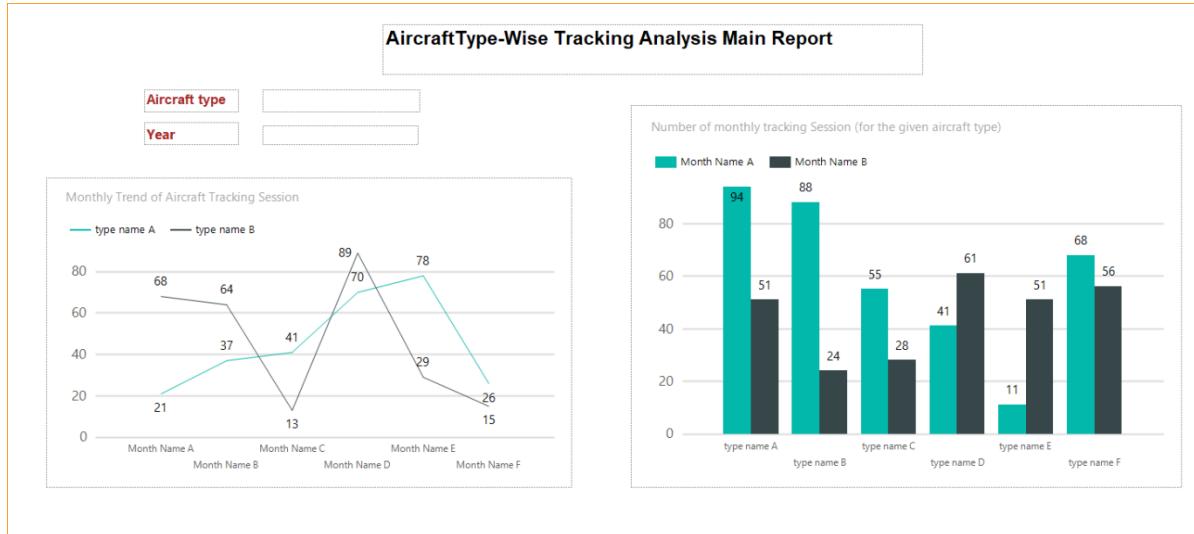


Figure 4.5.4.18.1- Add another two text boxes to get the parameter values

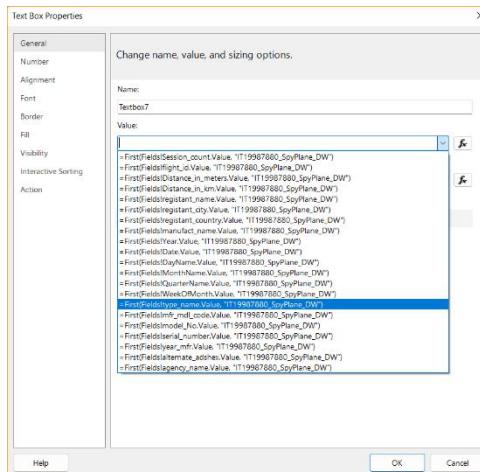


Figure 4.5.4.18.3 - Select the values

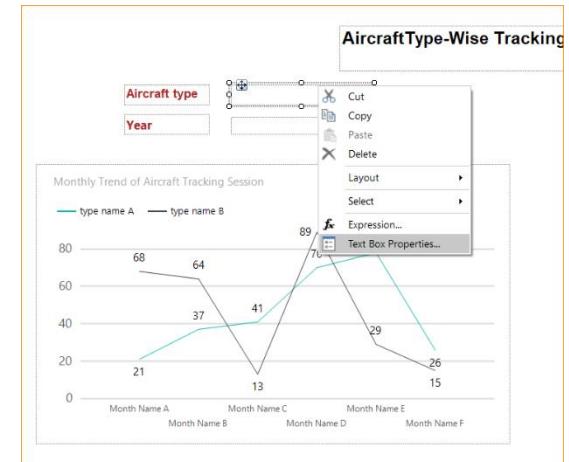


Figure 4.5.4.18.2- Right click on the text box select text box properties

19. Save the report

At the end, execute first level report from the SSRS portal

When you take the mouse cursor over any column in either of the charts in the first level report, the cursor should change to 'hand', which means it is enabled to be clicked. Once you clicked on a category

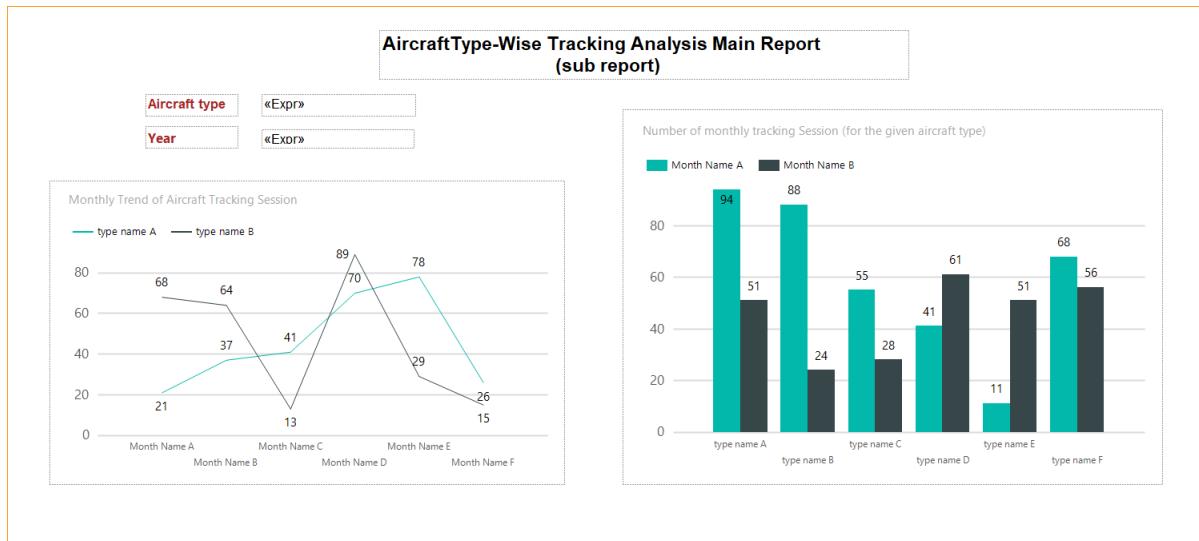


Figure 4.5.4.19.1 - Sub report

- I added title 'sub report 'in brackets in the main title of the sub report to identify easily . In previous sub report screen shots haven't that title

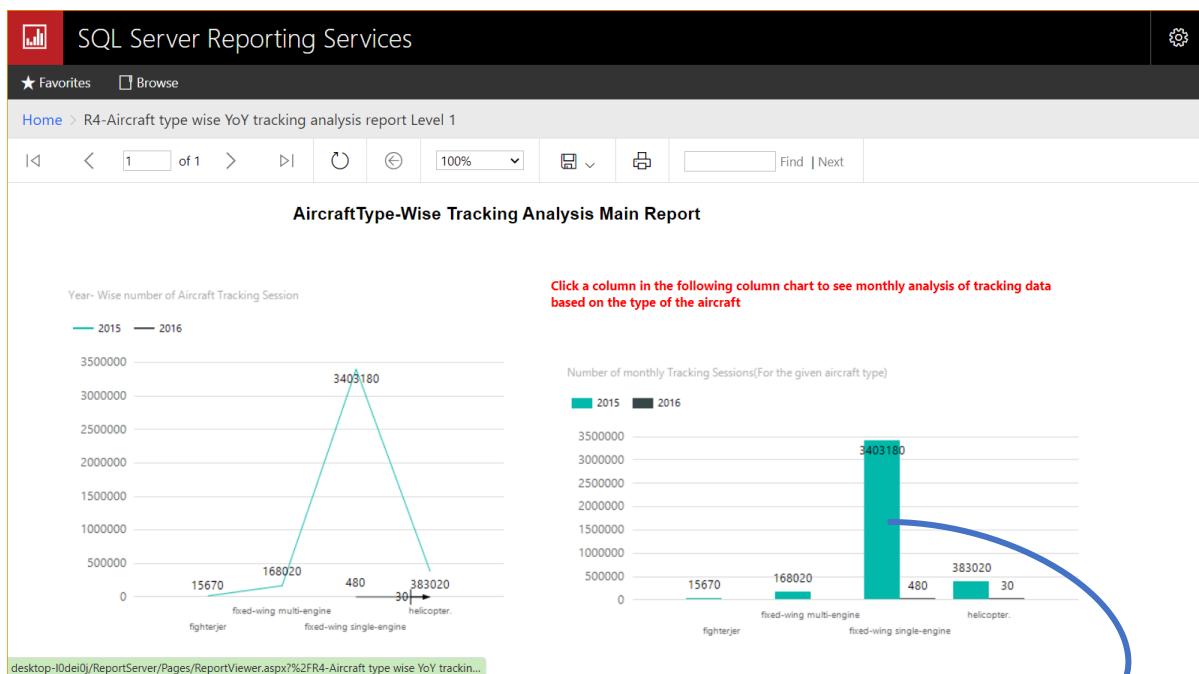


Figure 4.5.4.19.2 - Finalized main report and click on the column bar

When click on that column bar it automatically navigate to sub report

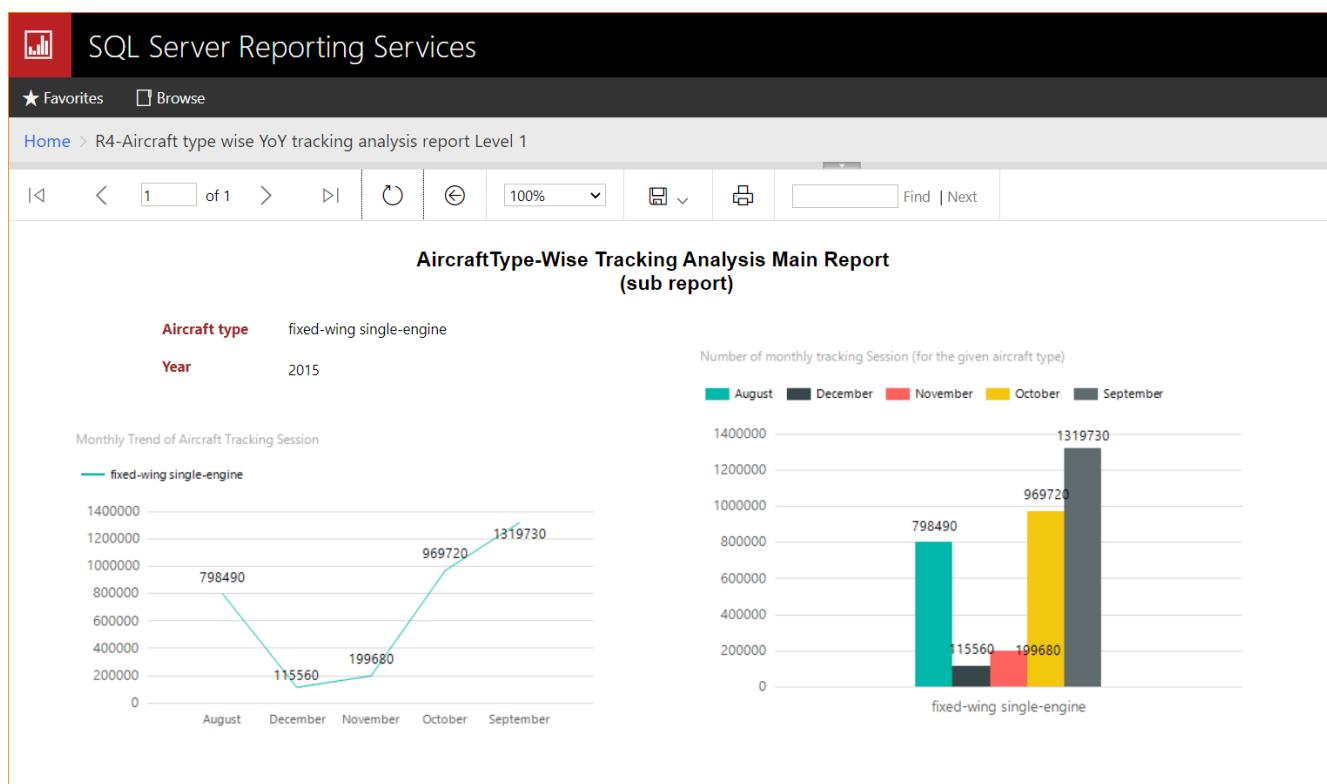


Figure 4.5.4.19.3- finalized sub report open when main report column is clicked