

Sri Lankan Institute of Information Technology



Data Warehousing and Business Intelligence
(IT3021)

1.6 million UK traffic accidents

Assignment 2

Submitted to

Sri Lanka Institute of Information Technology

Bachelor of Science Special Honors Degree in Data Science

IT19176666

Aththanayake A.B.P.S

25.06.2021

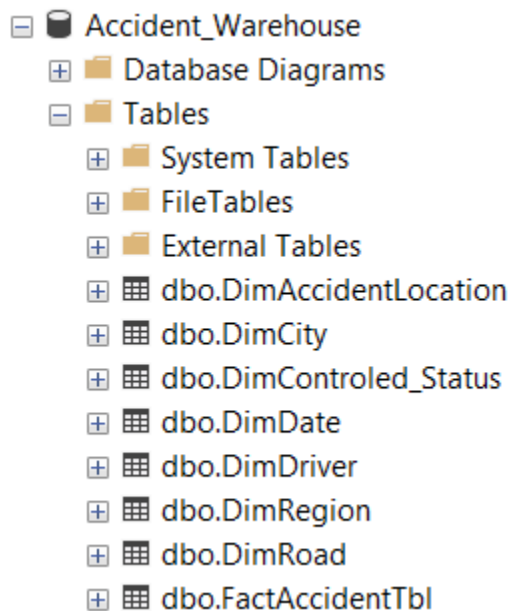
STEP 01

Data Source (Data Warehouse Introduction)

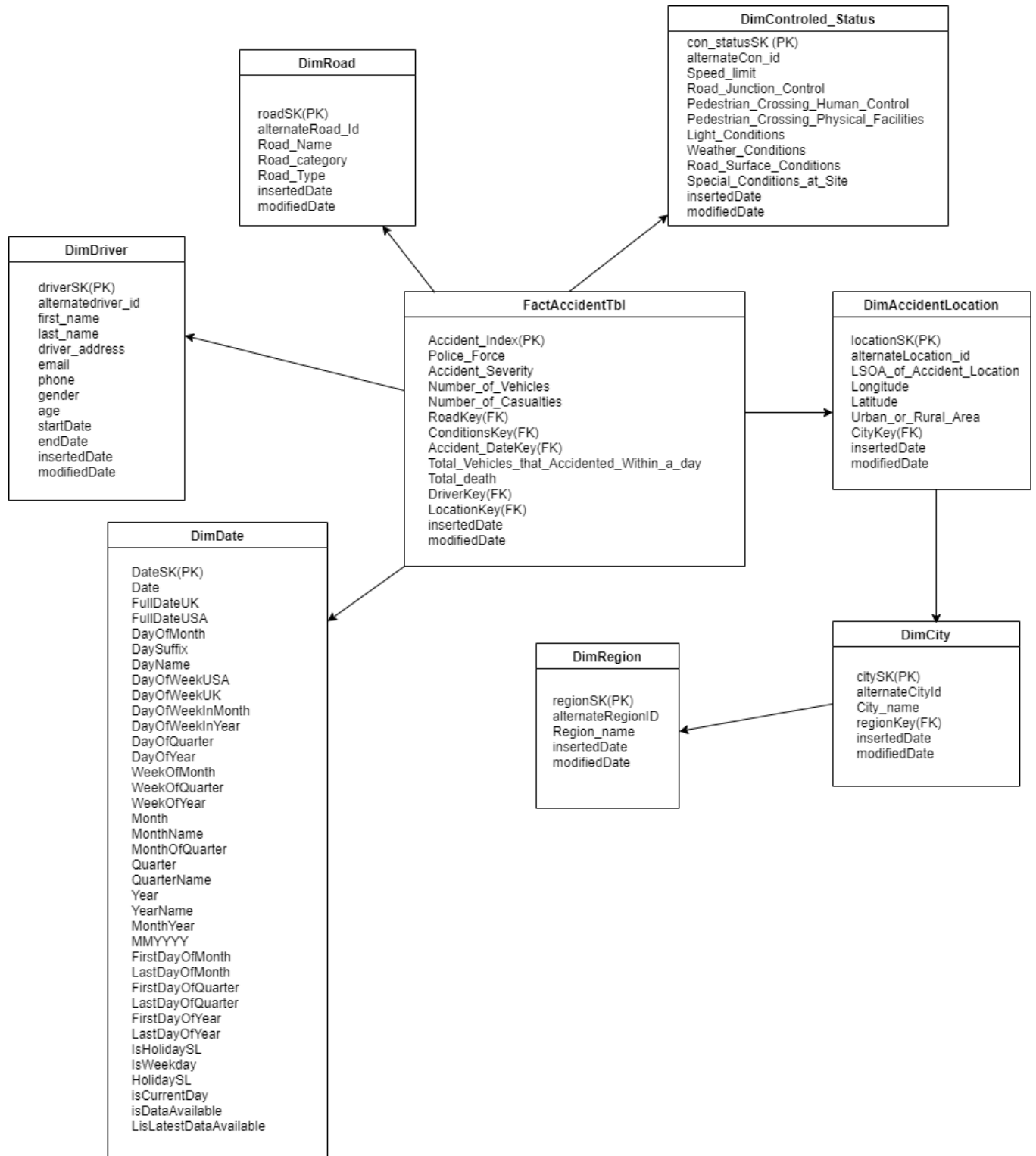
In here used data source is the '[Accident_Warehouse](#)' which is the database which I have created as assignment one. I have chosen **1.6 million UK traffic accidents** data set of UK government amassed traffic data from 2000-2016. But here I have inly used data from 2012 – 2014, because of complexity. As well as I have done some modifications accordingly to the data set derived from source for assignment one.

Therefore, I have divided my data set into seven tables and eight-dimension tables (Include fact table) using my scenario. As I have mentioned before my tables was, **FactAccidentTbl**, **DimRoad**, **DimControlled_Status**, **DimDriver**, **DimAccidentLocation**, **DimCity**, **DimRegion**.

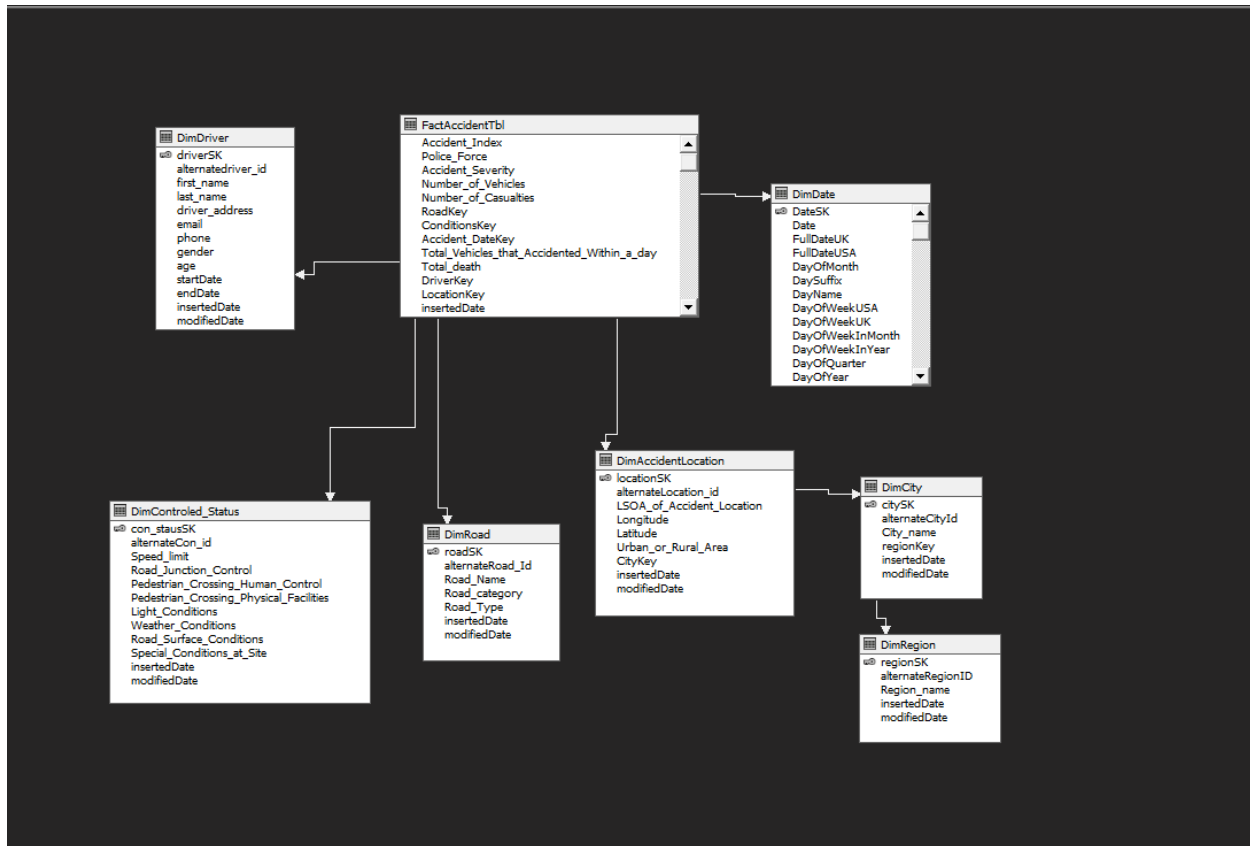
I. DATA SOURCE: Accident_Warehouse database



II. ER -Diagram for Data warehouse



III. Snowflake Schema used



STEP 02

Cube Implementation

OLAP cube is a method for storing data in a multi-dimensional form. This multi-dimensional cube is used for analytical purpose. In this process I have used SQL Server Data Tools. Dimensions and measures are main constituents of a cube. In addition to two main facts are,

- Measures: Provide aggregated numerical values of interest to the end user.
- Dimensions: Define the structure of the cube that you use to slice and dice over.

2.1

STEP 01: Data Source Creation

When creating data source, it was connected to the 'Accident_Warehouse' database via SQL Server Management Studio using windows username and password for connecting to the SQL Server Management Studio.

I. Data Source creation

Data Source Wizard

Impersonation Information
You can define what Windows credentials Analysis Services will use to connect to the data source.

☒ Use a specific Windows user name and password

User name:

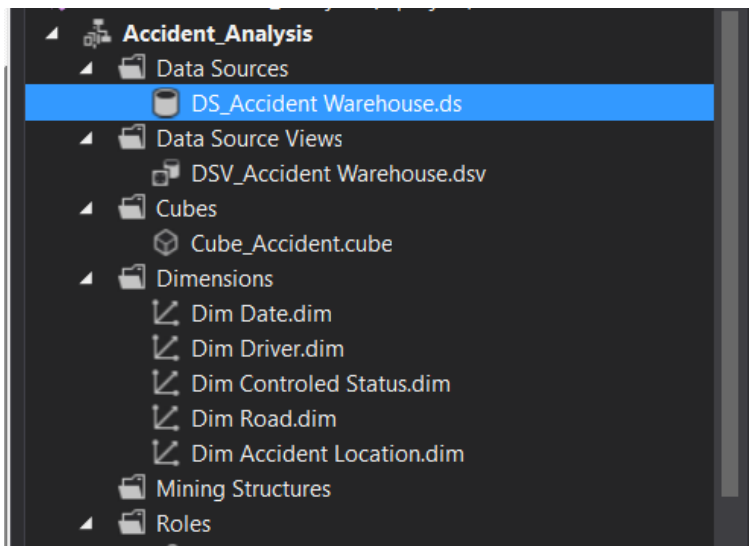
Password:

☐ Use the service account

☐ Use the credentials of the current user

☐ Inherit

< Back Next > Finish >> Cancel



Data Source Designer

General Impersonation Information

Data source name: DS_Accident Warehouse

Provider:

Connection string: Provider=SQLNCLI11.1;Data Source=LAPTOP-5RFD4QAA; Edit...

Data source references

☐ Maintain a reference to another object in the solution

Create a data source based on an existing data source

Isolation: ReadCommitted

Query timeout: 0 seconds

Maximum number of connections: 10

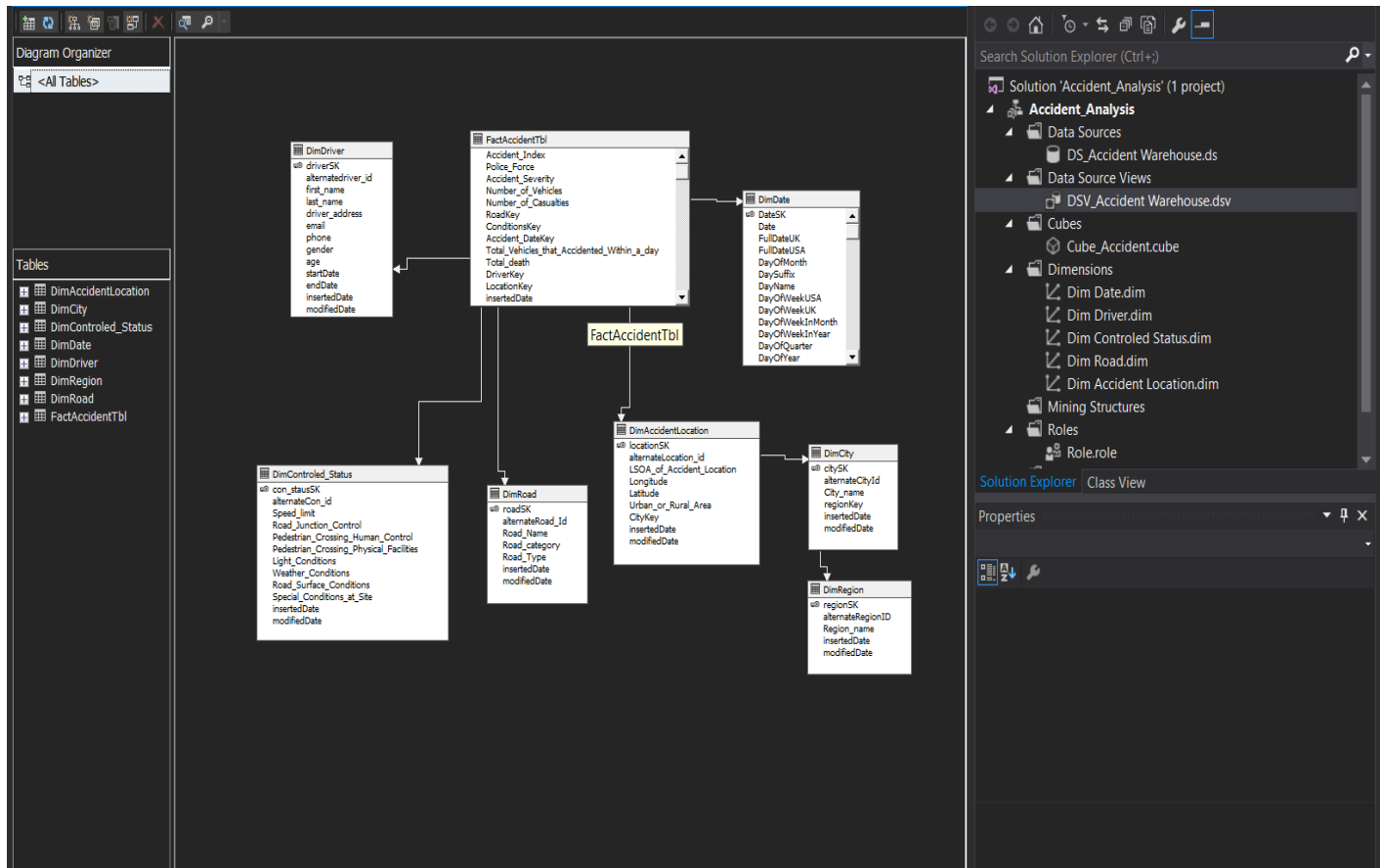
Data source description:

OK Cancel Help

2.2

STEP 02: Data Source Views Creation

After creating data sources my next step was creating data source view. The step describes the relations and views of dataset.



2.3

STEP 03: The Cube Creation

In here, the cube was created using data source view. We can use the existing data source to create the cube. As well as FactAccidentTbl is fact table and it contains all included measures required for the analysis process. In measure group there is FactAccidentTbl and in dimensions there is DimDate, DimCity, DimRegion, DimAccidentLocation, DimDriver, DimRoad, DimControlledStatus.

Completing the Wizard

Name the cube, review its structure, and then click Finish to save the cube.



Cube name:

Cube_Accident

Preview:

- Measure groups
 - Fact Accident Tbl
 - Police Force
 - Accident Severity
 - Number Of Vehicles
 - Number Of Casualties
 - Total Vehicles That Accided Within a Day
 - Total Death
 - Fact Accident Tbl Count
- Dimensions
 - Dim Date
 - Dim Driver
 - Dim Controlled Status
 - Dim Road
 - Dim Accident Location

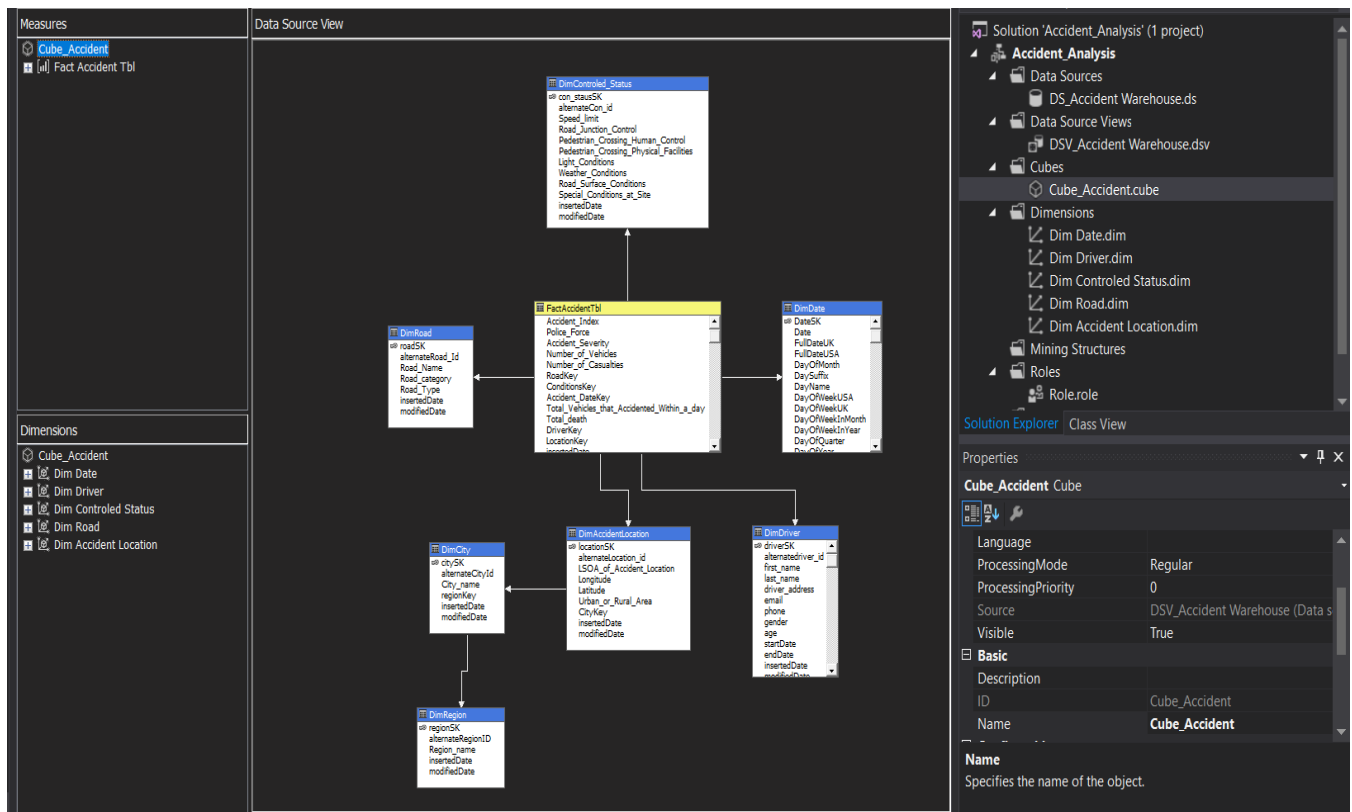
< Back

Next >

Finish

Cancel

Implemented cube is below.



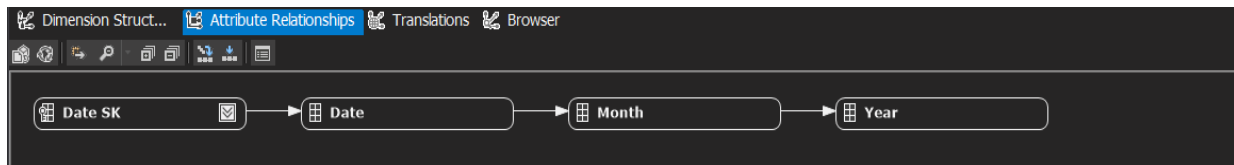
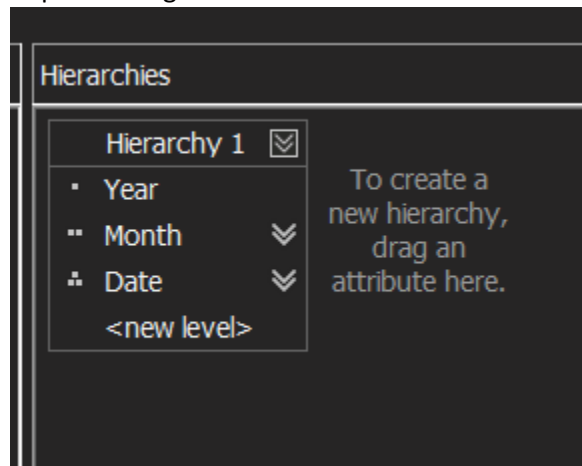
2.4

STEP 04: Hierarchies Creation

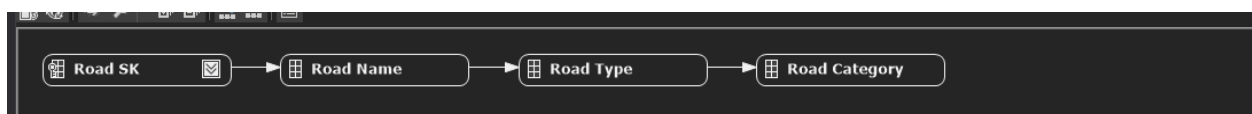
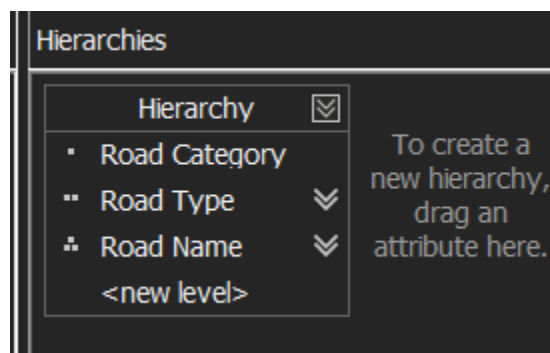
Hierarchies are used to reduce complexity and normalized tables. This will be used to drill down behavior.

I have created three hierarchies for Date, Location and Road.

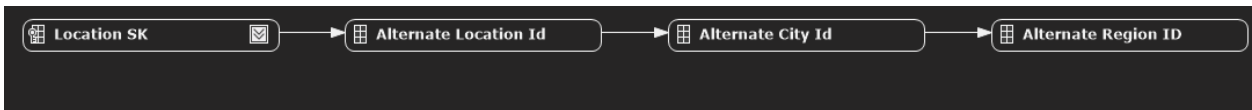
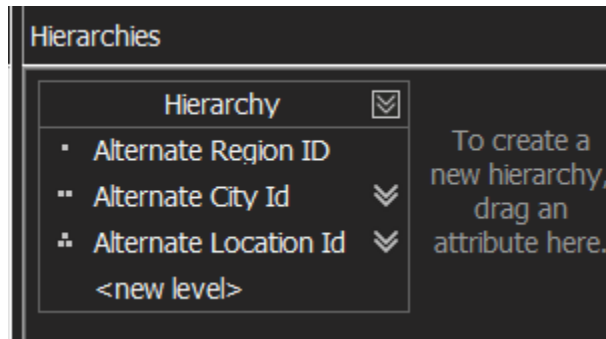
- Implementing hierarchies to the DimDate



- Implementing hierarchies to the DimAccidentLocation



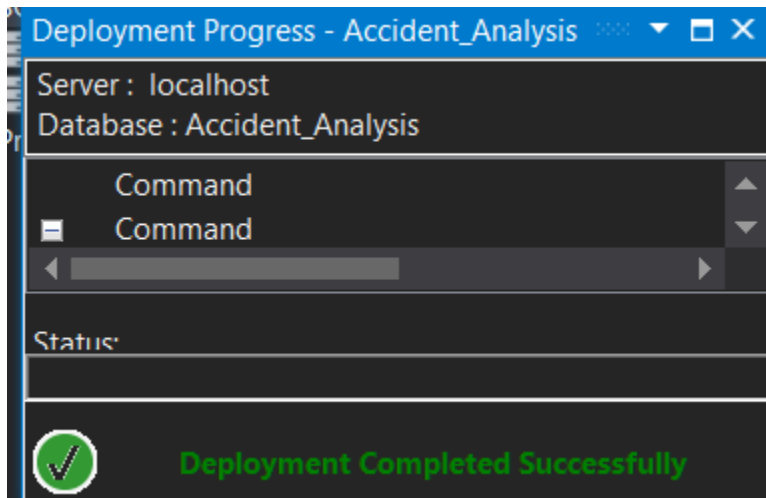
- Implementing hierarchies to the DimRoad



2.5

STEP 5: Deploy the Cube

The cube must be deployed in accident to be used for analysis. Having proper connection details provided will take effect in this stage when you try to deploy. Once deployed, SSAS Cube will be available for analysis under SSAS database accessible via SSMS.



2.6

STEP 6: KPIs Creation

KPI's are created based on the business requirements. KPIs depend on what the organization want to monitor and measure. According to my scenario I wanted to measure Accident Total deaths per day was less than 2000.

The screenshot displays the Qlik Sense KPI Designer interface. The top navigation bar includes tabs for Cube Struct..., Dimension Usage, Calculations, KPIs (active), Actions, Partitions, Aggregations, Perspectives, Translations, and Browser. Below the navigation bar is a toolbar with various icons for editing and saving.

The main workspace is divided into two panels. The left panel, titled 'KPI Organizer', shows a tree view with the following structure:

- Cube_Accident
 - Measures
 - KPI Accident total_death (selected)
 - Dim Accident Location
 - Dim Controlled Status
 - Dim Date
 - Dim Driver
 - Dim Road

The right panel, titled 'KPI Designer', contains the configuration for the selected KPI. It includes the following fields:

- Name:** KPI Accident total_death
- Associated measure group:** Fact Accident Tbl
- Value Expression:** [Measures].[Total Death]
- Goal Expression:** [Measures].[Total Death] < 2000
- Status:**
 - Status indicator:** Gauge
 - Status expression:** (empty field)

STEP 03

Demonstration of OLAP Operations

Online Analytical Processing (OLAP) is the technology behind many Business Intelligence (BI) Applications. As well as this will provide powerful capabilities for data mining and trend analysis.

OLAP technology has been defined as the ability to archive “Fast access to share multidimensional information”.

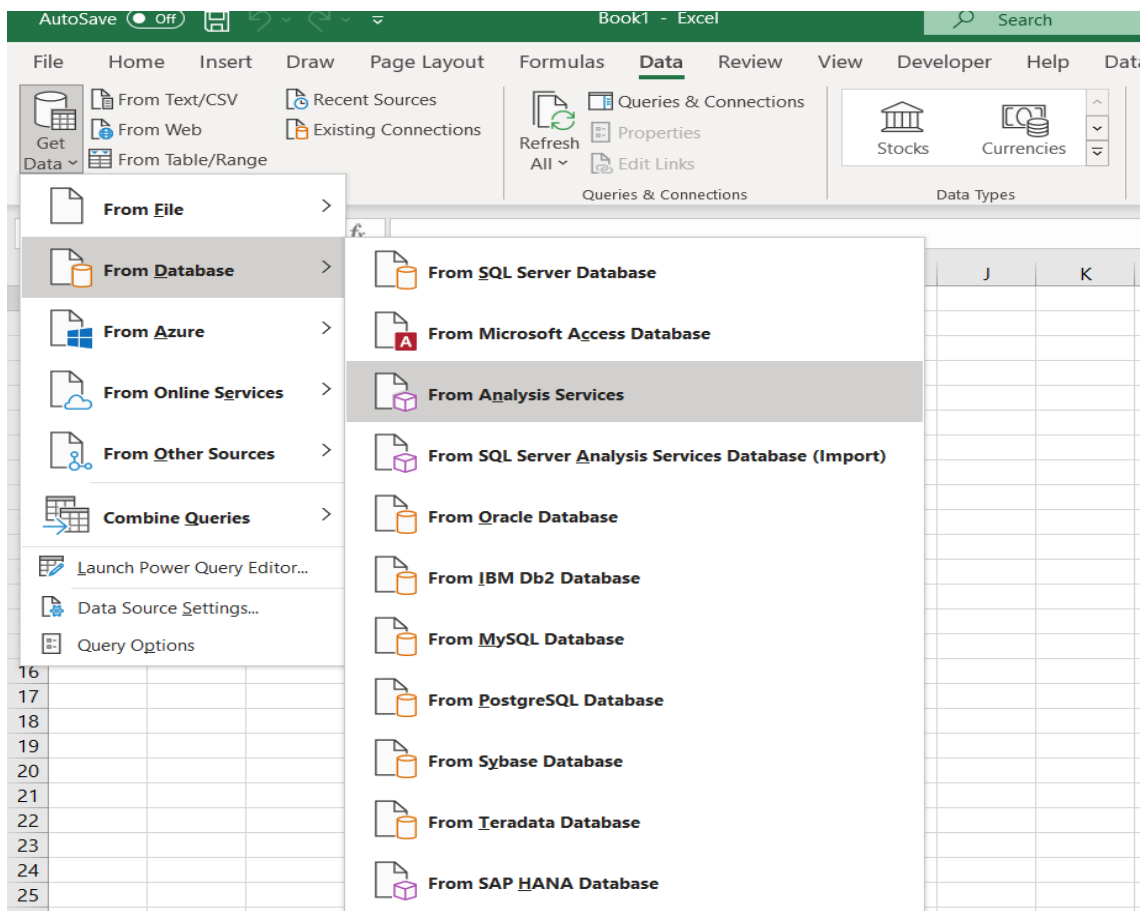
MDX: Multi-Dimensional Expressions is used for retrieving data from SSAS cubes.

According to my scenario I did not use MDX and I have used data tab in the ribbon.

3.1

STEP 01: SSAS Cube Connection

Data → Get Data → From Database → From Analysis Services



Connect to Database Server

Enter the information required to connect to the database server.

1. Server name:

2. Log on credentials

- ☒ Use Windows Authentication
☐ Use the following User Name and Password

User Name:

Password:

Cancel

< Back

Next >

Finish

Select Database and Table


Select the Database and Table/Cube which contains the data you want.

Select the database that contains the data you want:

Accident_Analysis



☒ Connect to a specific cube or table:

Name	Description	Modified	Created	Type
 Cube_Accident		6/25/2021 1:13:08 PM		CUBE

Cancel

< Back

Next >

Finish

3.2

STEP 02: Demonstration of Excel Report for OLAP Operations

There is five reports for showing OLAP operations. Those are,

- ❖ Report 1: Roll Up
- ❖ Report 2: Drill Down
- ❖ Report 3: Slice
- ❖ Report 4: Dice
- ❖ Report 5: Pivot

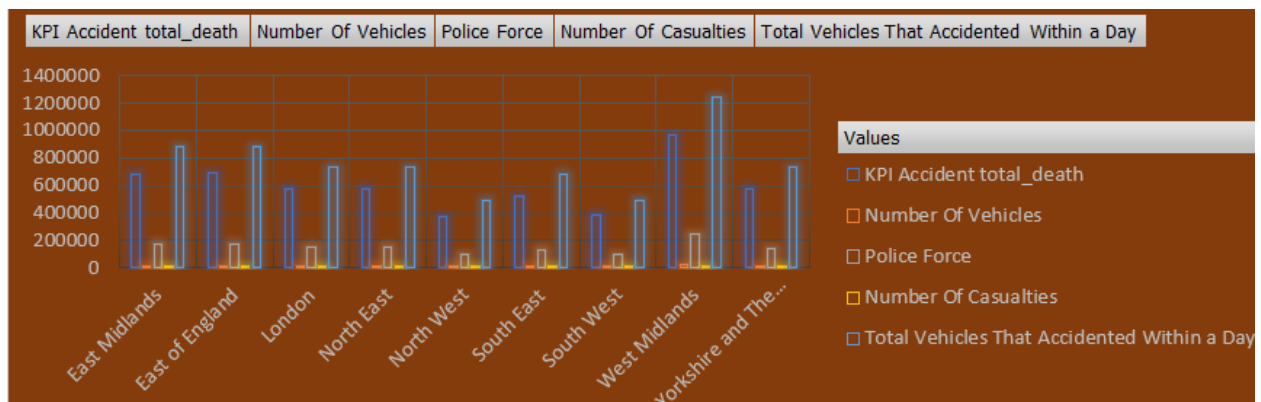
Report 01

Roll Up

Climbing up a hierarchy of a dimension or by dimension reduction to aggregate data. This is also known as consolidation or aggregation.

According to my scenario I have used total incidents of accidents in different regions and cities.

TOTAL INCIDENTS OF ACCIDENTS					
Row Labels	KPI Accident total_death	Number Of Vehicles	Police Force	Number Of Casualties	Total Vehicles That Accidented Within a Day
East Midlands	682641	17893	173128	13025	878501
East of England	687914	18029	174186	13125	884814
London	570706	15152	148185	10906	735386
North East	573450	15088	147647	10979	738850
North West	378053	9835	95596	7166	486013
South East	527185	13752	134135	10069	677765
South West	383962	10058	99797	7342	493742
West Midlands	968625	25571	249124	18564	1247745
Yorkshire and The Humber	575134	14912	145613	10914	738674
Grand Total	5347670	140290	1367411	102090	6881490

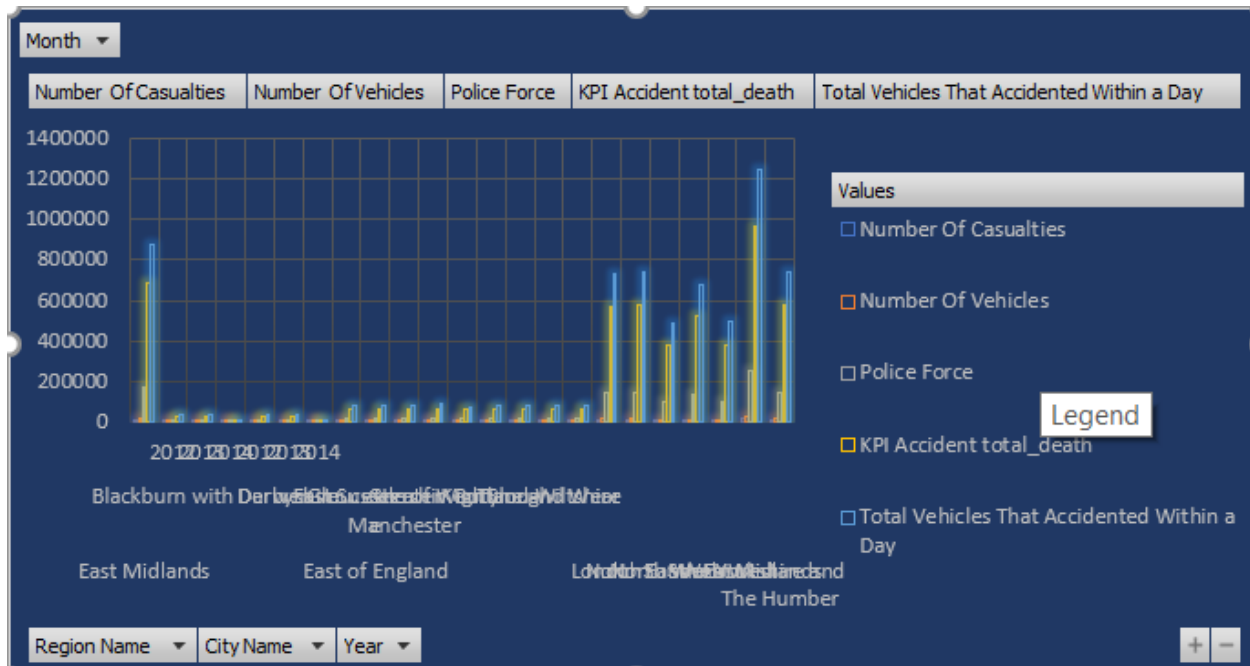


Report 02

Drill Down

Stepping down a hierarchy of a dimension allowing navigation through details. Data is divided into smaller parts. It is the opposite of the roll up process. According to my scenario I used drill down for check total Number of Casualties, Number of Vehicles, Total Deaths within a day and total vehicles that accident within a day by region, cities and year wise.

Row Labels	Number Of Casualties	Number Of Vehicles	Police Force	KPI Accident total_death	Total Vehicles That Accidented Within a Day
East Midlands	13025	17893	173128	682641	878501
East of England					
Blackburn with Darwen					
2012	531	743	7267	28261	36341
2013	610	839	8705	31487	40567
2014	89	118	1036	4148	5368
Derbyshire					
2012	547	711	6640	26694	34454
2013	553	769	7298	29165	37445
2014	94	127	1196	4708	6088
East Sussex	1161	1621	14999	60723	78363
Gloucestershire	1150	1582	15396	61667	79147
Greater Manchester	1168	1653	15522	61931	79791
Isle of Wight	1235	1717	16625	67627	86467
Kent	1142	1588	15462	59721	77281
Rutland	1241	1678	17197	63970	82290
Slough	1208	1615	14630	62000	79760
Tyne and Wear	1186	1613	15568	61600	79200
Wiltshire	1210	1655	16645	64212	82252
London	10906	15152	148185	570706	735386
North East	10979	15088	147647	573450	738850
North West	7166	9835	95596	378053	486013
South East	10069	13752	134135	527185	677765
South West	7342	10058	99797	383962	493742
West Midlands	18564	25571	249124	968625	1247745
Yorkshire and The Humber	10914	14912	145613	575134	738674
Grand Total	102090	140290	1367411	5347670	6881490



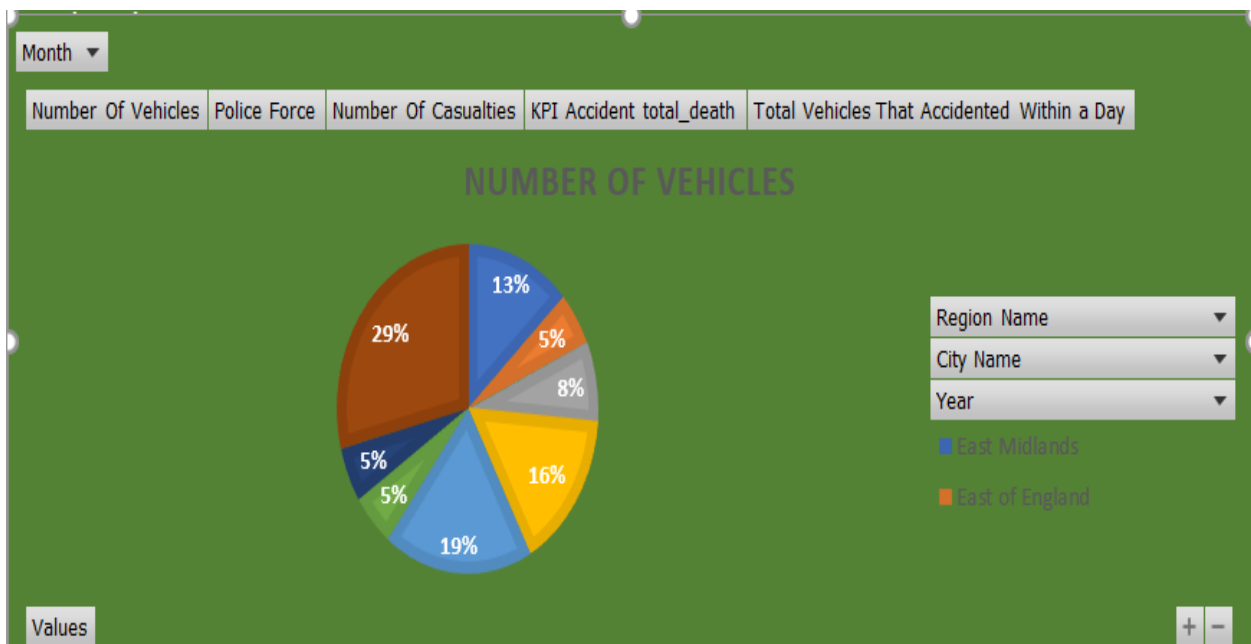
Report 03

Slice

A slice representation two or more dimensional views of an OLAP data cube. A slice function much like a report or a query. Then we can request data what we need to see.

According to my scenario I have used slice for check total Number of Casualties, Number of Vehicles, Total Deaths within a day and total vehicles that accident within a day in specific road name by region, city and year wise.

Row Labels	Number Of Vehicles	Police Force	Number Of Casualties	KPI Accident total_death	Total Vehicles That Accidented Within a Day	Road Name
East Midlands	5	41	3	228	288	A1031
East of England	2	42	2	144	184	A1033
North East	3	34	2	152	192	A1033
North West	6	50	6	197	257	A1033
South East	7	6	4	222	282	A1033
South West						A1033
Bedfordshire						A1033
2013	2	1	1	64	84	A1033
Lincolnshire						A1033
2012	2	20	1	46	66	A1033
West Midlands	11	115	5	218	298	A1033
					1651	

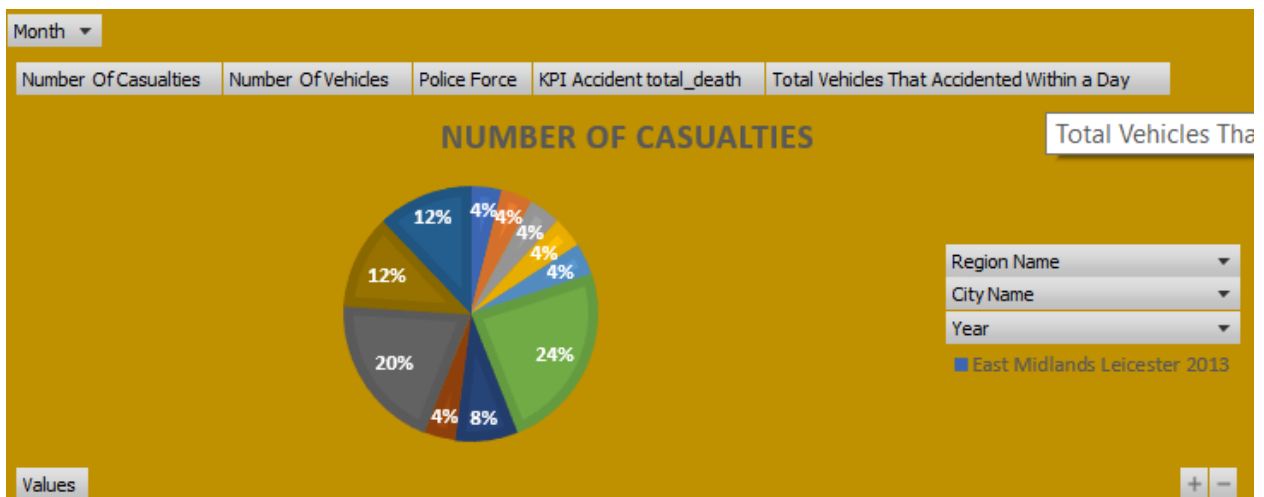


Report 04

Dice

Select two or more dimensions from a given cube and provides a new sub-cube by selecting specific values on those selected dimensions. Dice operation is similar to a slice.

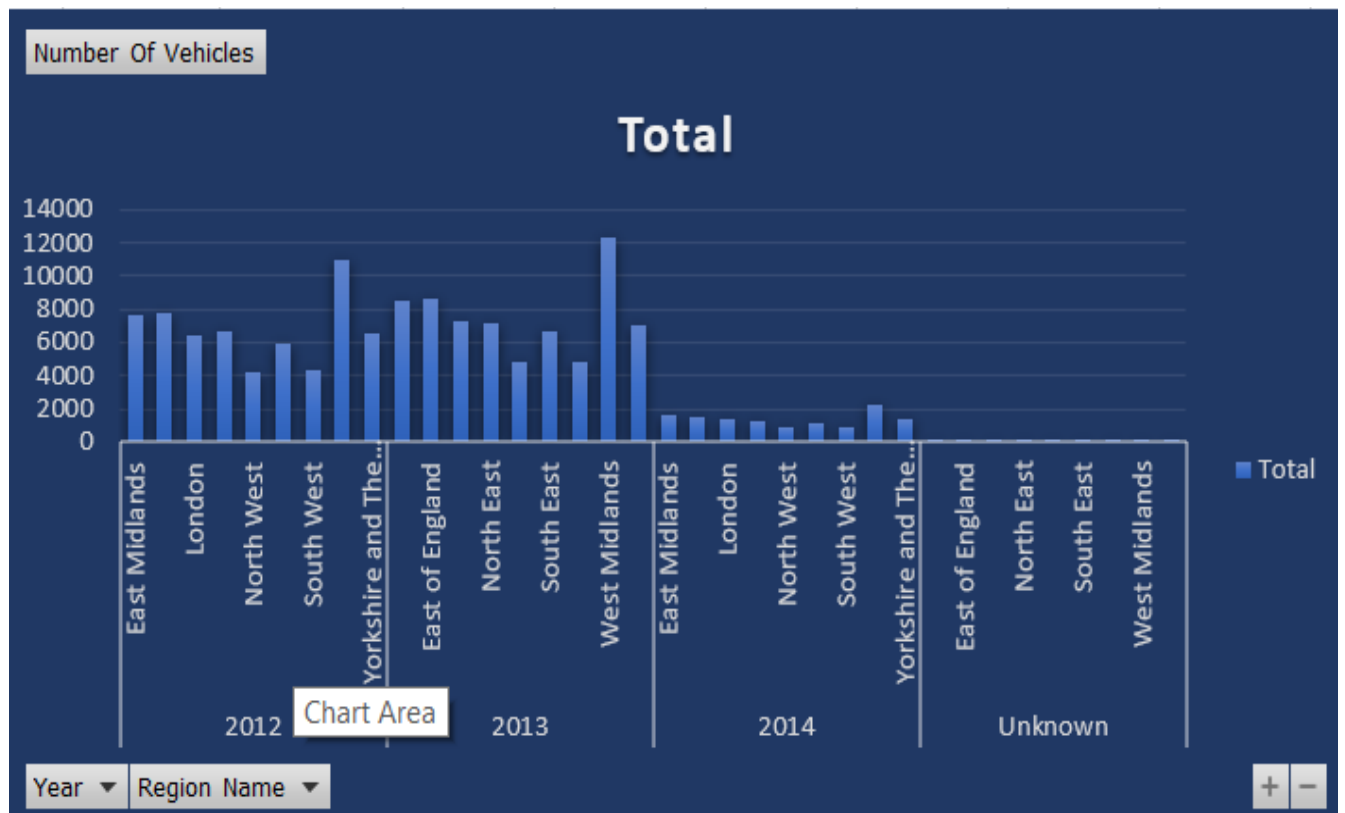
According to my scenario I have used slice for check total Number of Casualties, Number of Vehicles, Total Deaths within a day and total vehicles that accident within a day. in specific road name and speed limit by region, city and year wise.

[illegible]

Report 05

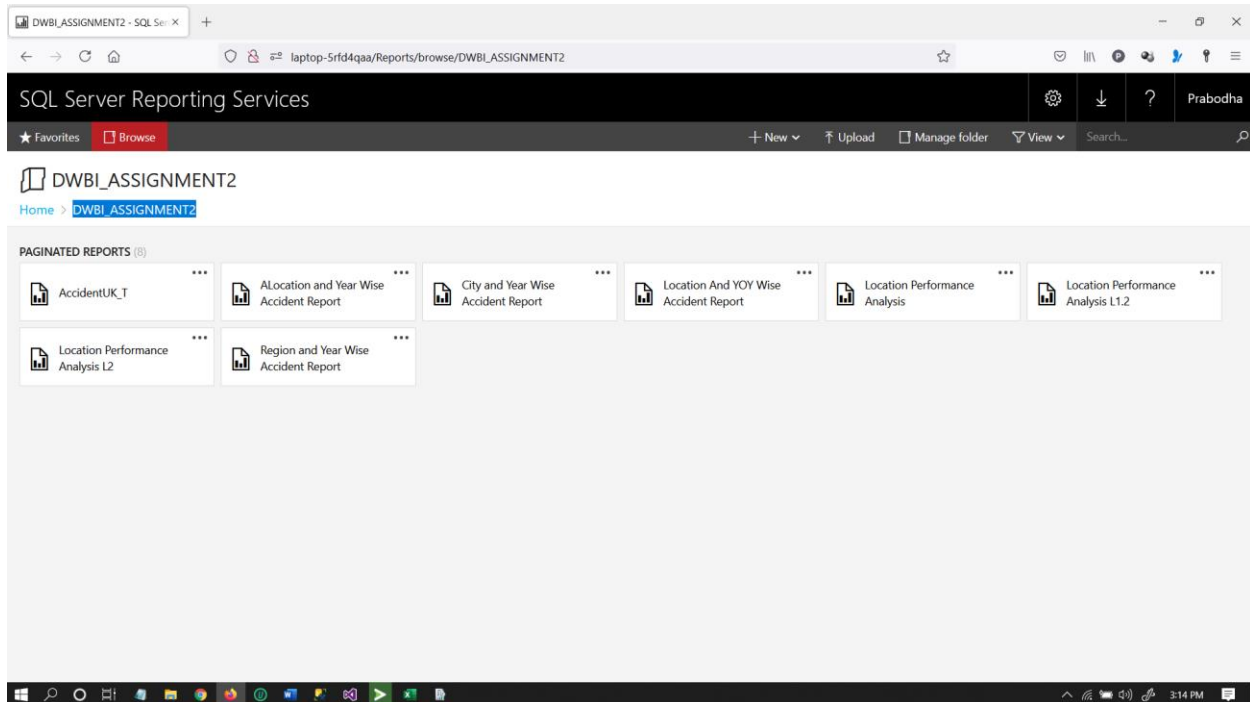
Pivot Chart

According to my scenario I have used Number of vehicles by year and regions.



STEP 04

SQL Service Reporting Service Report (SSRS Report)



The **SQL Server Reporting Service (SSRS)** is a reporting application that provides you to generate formatted tables reports in the form of data, pictures, diagrams and charts. The reports are hosted on a server which may be performed with user-defined settings any time. A reporting services server's web portal is a web-based experience. The report, mobile report, KPI and element in your report server instance may be viewed on the portal. You may also manage a single report server instance using the web portal.

I have used report builder application for this. The **Report Builder** is an independent software that you or an administrator installs on your computer. It may be installed through the Microsoft Download Center, the 2016 reporting services SQL Servers or later (SSRS) or from a reporting services SharePoint site.

4.1

Basic Steps of Report Builder

Step 01: Data Source Creation

Throughout this stage we will also be going to connect to our previously created data warehouse data source 'Accident_Warehouse'.

The image shows two overlapping dialog boxes from the SQL Server Reporting Services (SSRS) interface.

Data Source Properties Dialog:

- General Tab:** The 'Name' field is set to 'AccidentUK_DS'. The 'Use a connection embedded in my report' radio button is selected. The 'Select connection type' dropdown is set to 'Microsoft SQL Server'. The 'Connection string' field contains 'Data Source=.;Initial Catalog=Accident_Warehouse'. There are 'Build...' and 'Test Connection' buttons next to the connection string field.
- Buttons:** 'Help' and 'OK' buttons are at the bottom.

Connection Properties Dialog:

- Data source:** 'Microsoft SQL Server (SqlClient)' with a 'Change...' button.
- Server name:** '.' with a 'Refresh' button.
- Log on to the server:** 'Use Windows Authentication' is selected. There are fields for 'User name:' and 'Password:', and a 'Save my password' checkbox.
- Connect to a database:** 'Select or enter a database name:' is selected, with 'Accident_Warehouse' in the dropdown. There is also an 'Attach a database file:' option with a 'Browse...' button and a 'Logical name:' field.
- Buttons:** 'Test Connection', 'OK', and 'Cancel' buttons are at the bottom.

Step 02 : Dataset Creation

In this step a data set is imported into our report builder with the help of a query which is manual process assigned as actually needed.

Dataset Properties

Query

Fields

Options

Filters

Parameters

Choose a data source and create a query.

Name:

AccidentUK_DataSet

☐ Use a shared dataset.

☒ Use a dataset embedded in my report.

Data source:

AccidentUK_DS

New...

Query type:

☒ Text ☐ Table ☐ Stored Procedure

Query:

```
SELECT al.LSOA_of_Accident_Location, dc.City_name, dr.Region_name,
dd.alternatedriver_id, dd.first_name, dd.last_name, dd.email, dd.age, dd.gender,
dd.driver_address, ddd.Year, ddd.Month, dcs.Speed_limit, dcs.Weather_Conditions,
dcs.Light_Conditions, drr.Road_category, fa.Number_of_Casualties, fa.Number_of_Vehicles,
fa.Police_Force, fa.Total_death
FROM FactAccidentTbl fa Inner join DimAccidentLocation al on fa.LocationKey =
al.locationSK inner join DimCity dc on al.CityKey = dc.citySK inner join DimRegion dr on
dc.regionKey = dr.regionSK inner join DimDriver dd on fa.DriverKey = dd.driverSK inner
join DimDate ddd on fa.Accident_DateKey = ddd.DateSK inner join DimControlled_Status
dcs on fa.ConditionsKey = dcs.con_stausSK inner join DimRoad drr on fa.RoadKey =
drr.roadSK
```

Query Designer...

Import...

Refresh Fields

Help

OK

Cancel

4.2

Report Demonstrations

❖ **Report 1**

Report with a matrix

Matrix is similar to a table in SSRS, but it is configured to display data categorized by rows and columns, with statistical data at the crossing. In a spreadsheet, this is equivalent to using a pivot table.

Query for AccidentUK_Dataset:

```
SELECT al.LSOA_of_Accident_Location, dc.City_name, dr.Region_name,  
dd.alternatedriver_id, dd.first_name, dd.last_name, dd.email, dd.age, dd.gender,  
dd.driver_address, ddd.Year, ddd.Month, dcs.Speed_limit, dcs.Weather_Conditions,  
dcs.Light_Conditions, drr.Road_category, fa.Number_of_Casualties,  
fa.Number_of_Vehicles, fa.Police_Force, fa.Total_death
```

```
FROM FactAccidentTbl fa
```

```
    Inner join DimAccidentLocation al on fa.LocationKey = al.locationSK
```

```
    inner join DimCity dc on al.CityKey = dc.citySK
```

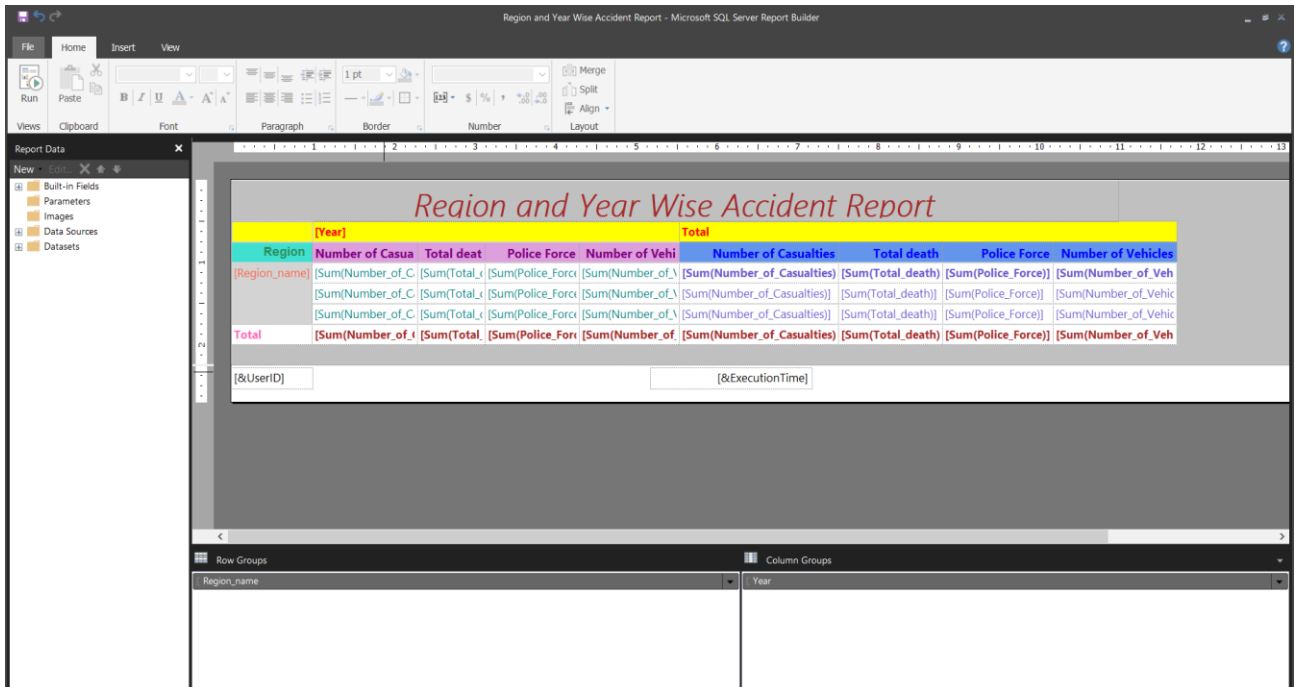
```
    inner join DimRegion dr on dc.regionKey = dr.regionSK
```

```
    inner join DimDriver dd on fa.DriverKey = dd.driverSK
```

```
    inner join DimDate ddd on fa.Accident_DateKey = ddd.DateSK
```

```
    inner join DimControlled_Status dcs on fa.ConditionsKey = dcs.con_stausSK
```

```
    inner join DimRoad drr on fa.RoadKey = drr.roadSK
```



Region and Year Wise Accident Report

SQL Server Reporting Services

Home > DWBI_ASSIGNMENT2 > Region and Year Wise Accident Report

1 of 1

100%

Find | Next

Region and Year Wise Accident Report												
2012				2013				2014				Total
Region name	Number of Casualties	Total death	Police Force	Number of Vehicles	Number of Casualties	Total death	Police Force	Number of Vehicles	Number of Casualties	Total death	Police Force	Number of Vehicles
East Midlands	5615	296996	74423	7708	6213	324278	82369	8529	1180	59965	16067	1631
	5615	296996	74423	7708	6213	324278	82369	8529	1180	59965	16067	1631
	5615	296996	74423	7708	6213	324278	82369	8529	1180	59965	16067	1631
East of England	5700	299324	75602	7805	6300	332096	83861	8643	1112	55620	14548	1562
	5700	299324	75602	7805	6300	332096	83861	8643	1112	55620	14548	1562
	5700	299324	75602	7805	6300	332096	83861	8643	1112	55620	14548	1562
London	4629	245250	63573	6457	5274	276957	71714	7331	987	47170	12634	1342
	4629	245250	63573	6457	5274	276957	71714	7331	987	47170	12634	1342
	4629	245250	63573	6457	5274	276957	71714	7331	987	47170	12634	1342
North East	4759	252895	64061	6612	5230	271443	70246	7130	966	47418	13024	1314
	4759	252895	64061	6612	5230	271443	70246	7130	966	47418	13024	1314
	4759	252895	64061	6612	5230	271443	70246	7130	966	47418	13024	1314
North West	2998	163992	41622	4174	3523	182290	46187	4805	637	31321	7723	846
	2998	163992	41622	4174	3523	182290	46187	4805	637	31321	7723	846
	2998	163992	41622	4174	3523	182290	46187	4805	637	31321	7723	846
South East	4367	229540	57765	5944	4879	256333	64781	6655	814	40673	11464	1140
	4367	229540	57765	5944	4879	256333	64781	6655	814	40673	11464	1140
	4367	229540	57765	5944	4879	256333	64781	6655	814	40673	11464	1140

❖ **Report 2**

Report with more than one parameter

Applying Multiparameter-values in SSRS enables us to transfer one or more than the input data to the report. This one has a "Select All" choice which always allows you to select all parameter values.

Query for AccidentUK_Dataset:

This data set contains the UK accident data which will be displayed in the report through the matrix.

```
SELECT al.LSOA_of_Accident_Location, dc.City_name, dr.Region_name,  
dd.alternatedriver_id, dd.first_name, dd.last_name, dd.email, dd.age, dd.gender,  
dd.driver_address, ddd.Year, ddd.Month, dcs.Speed_limit, dcs.Weather_Conditions,  
dcs.Light_Conditions, drr.Road_category, fa.Number_of_Casualties,  
fa.Number_of_Vehicles, fa.Police_Force, fa.Total_death
```

```
FROM FactAccidentTbl fa
```

```
    Inner join DimAccidentLocation al on fa.LocationKey = al.locationSK
```

```
    inner join DimCity dc on al.CityKey = dc.citySK
```

```
    inner join DimRegion dr on dc.regionKey = dr.regionSK
```

```
    inner join DimDriver dd on fa.DriverKey = dd.driverSK
```

```
    inner join DimDate ddd on fa.Accident_DateKey = ddd.DateSK
```

```
    inner join DimControlled_Status dcs on fa.ConditionsKey = dcs.con_stausSK
```

```
    inner join DimRoad drr on fa.RoadKey = drr.roadSK
```

```
    Where dc.City_name in (@City)
```

Query for Regions:

1st parameter. This dataset describes regions details.

```
select regionSK, alternateRegionID, Region_name
```


from DimRegion

Query for City:

2nd parameter. This dataset describes city details.

select citySK, alternateCityId, City_name, regionKey

from DimCity

where regionKey in (@Region)

parameters

Regions

Report Parameter Properties

General
Available Values
Default Values
Advanced

Choose the available values for this parameter.

Select from one of the following options:

☐ None
☐ Specify values
☒ Get values from a query

Dataset: (Warning: Possible performance impact)
Regions

Value field:
regionSK

Label field:
Region_name

Help OK Cancel

Report Parameter Properties

General
Available Values
Default Values
Advanced

Choose the available values for this parameter.

Select from one of the following options:

☐ None
☐ Specify values
☒ Get values from a query

Dataset: (Warning: Possible performance impact)
City

Value field:
City_name

Label field:
City_name

Help OK Cancel

City

1st parameter - Region

★ Favorites Browse

[Home](#) > [DWBI_ASSIGNMENT2](#) > ALocation and Year Wise Accident Report

Region City

- ☐ (Select All)
- ☒ East Midlands
- ☒ East of England
- ☒ London
- ☐ North East
- ☐ North West
- ☐ South East
- ☐ South West

2nd parameter -City

SQL Server Reporting Services

★ Favorites Browse

[Home](#) > [DWBI_ASSIGNMENT2](#) > ALocation and Year Wise Accident Report

Region City

- ☐ (Select All)
- ☐ Staffordshire
- ☒ North Yorkshire
- ☒ Dorset
- ☒ Leicester
- ☐ Luton
- ☐ South Gloucestershire
- ☐ Torbay

❖ Report 3

SSRS drill-down report

Using Drill Down in SSRS simply allows users to display or cover up column records by presenting plus and minus signs on such a text field (In short, providing interactivity to the user.).

Query for AccidentUK_Dataset:

```
SELECT al.LSOA_of_Accident_Location, dc.City_name, dr.Region_name,  
dd.alternatedriver_id, dd.first_name, dd.last_name, dd.email, dd.age, dd.gender,  
dd.driver_address, ddd.Year, ddd.Month, dcs.Speed_limit, dcs.Weather_Conditions,  
dcs.Light_Conditions, drr.Road_category, fa.Number_of_Casualties,  
fa.Number_of_Vehicles, fa.Police_Force, fa.Total_death
```

```
FROM FactAccidentTbl fa
```

```
    Inner join DimAccidentLocation al on fa.LocationKey = al.locationSK
```

```
    inner join DimCity dc on al.CityKey = dc.citySK
```

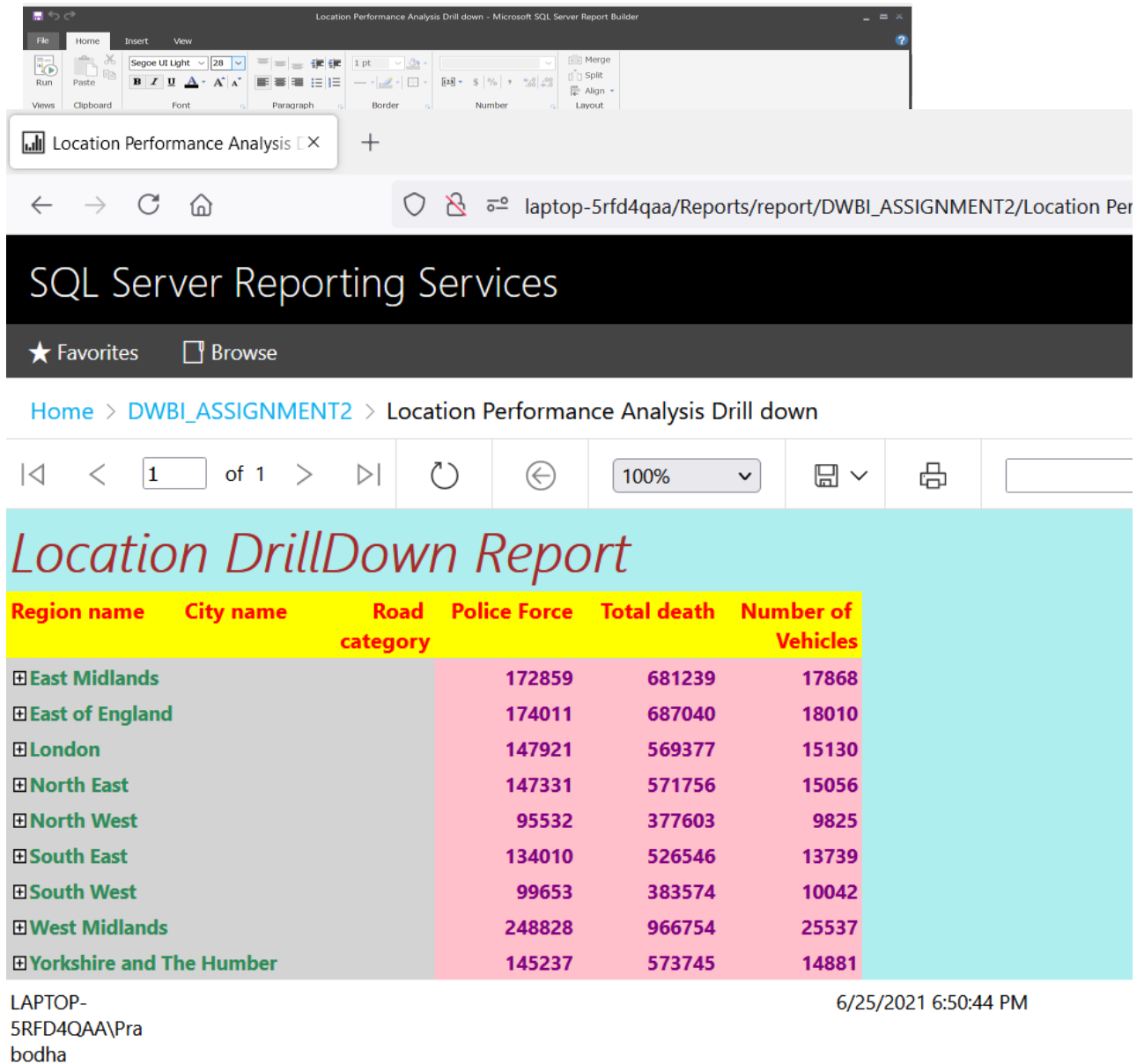
```
    inner join DimRegion dr on dc.regionKey = dr.regionSK
```

```
    inner join DimDriver dd on fa.DriverKey = dd.driverSK
```

```
    inner join DimDate ddd on fa.Accident_DateKey = ddd.DateSK
```

```
    inner join DimControlled_Status dcs on fa.ConditionsKey = dcs.con_stausSK
```

```
    inner join DimRoad drr on fa.RoadKey = drr.roadSK
```



Location Performance Analysis

+

←

→

↺

🏠

🛡️

🔗

🔍

laptop-5rfd4qaa/Reports/report/DWBI_ASSIGNMENT2/

SQL Server Reporting Services

★ Favorites

📁 Browse

Home > DWBI_ASSIGNMENT2 > Location Performance Analysis Drill down

⏮

⏪

1 of 1

⏩

⏭

↺

↻

⏮

100%

⏴

🖨

☐

Location DrillDown Report

Region name	City name	Road category	Police Force	Total death	Number of Vehicles
East Midlands	Dorset	PM	128	403	8
		PR	6972	27203	705
		PU	5711	23262	633
		TM	363	2160	51
		TR	1739	6854	176
		TU	583	2080	55
	Leicester		14156	58220	1534
	Luton		14819	59519	1578
	Milton Keynes		17785	69708	1830
	North Yorkshire		17592	64575	1691
	Plymouth		16289	64292	1697
	Reading		14659	57546	1490
	South Gloucestershire		14667	58860	1543
	Staffordshire		16981	64608	1696
Torbay		15539	63221	1677	
Windsor and Maidenhead		14876	58728	1504	
East of England		174011	687040	18010	
London		147921	569377	15130	
North East		147331	571756	15056	

❖ Report 4

SSRS drill-through report

Main Query :

```
SELECT al.LSOA_of_Accident_Location, dc.City_name, dr.Region_name,  
dd.alternatedriver_id, dd.first_name, dd.last_name, dd.email, dd.age, dd.gender,  
dd.driver_address, ddd.Year, ddd.Month, dcs.Speed_limit, dcs.Weather_Conditions,  
dcs.Light_Conditions, drr.Road_category, fa.Number_of_Casualties,  
fa.Number_of_Vehicles, fa.Police_Force, fa.Total_death
```

FROM FactAccidentTbl fa

Inner join DimAccidentLocation al on fa.LocationKey = al.locationSK

inner join DimCity dc on al.CityKey = dc.citySK

inner join DimRegion dr on dc.regionKey = dr.regionSK

inner join DimDriver dd on fa.DriverKey = dd.driverSK

inner join DimDate ddd on fa.Accident_DateKey = ddd.DateSK

inner join DimControlled_Status dcs on fa.ConditionsKey = dcs.con_stausSK

inner join DimRoad drr on fa.RoadKey = drr.roadSK

Sub Query :

```
SELECT al.LSOA_of_Accident_Location, dc.City_name, dr.Region_name,  
dd.alternatedriver_id, dd.first_name, dd.last_name, dd.email, dd.age, dd.gender,  
dd.driver_address, ddd.Year, ddd.Month, dcs.Speed_limit, dcs.Weather_Conditions,  
dcs.Light_Conditions, drr.Road_category, fa.Number_of_Casualties,  
fa.Number_of_Vehicles, fa.Police_Force, fa.Total_death
```

FROM FactAccidentTbl fa

Inner join DimAccidentLocation al on fa.LocationKey = al.locationSK

inner join DimCity dc on al.CityKey = dc.citySK

inner join DimRegion dr on dc.regionKey = dr.regionSK

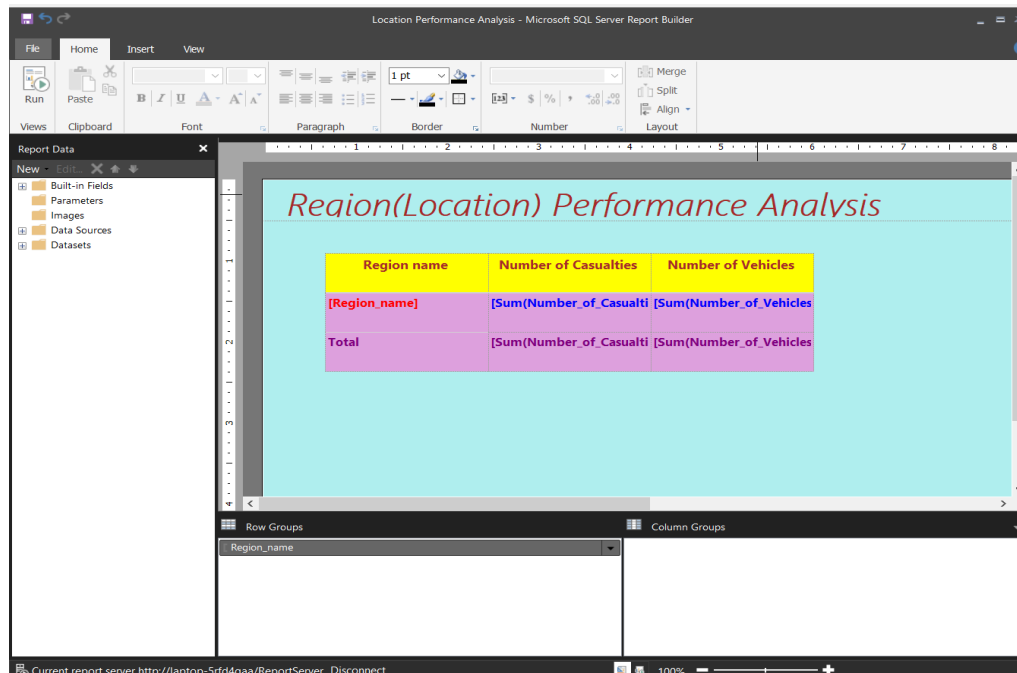
inner join DimDriver dd on fa.DriverKey = dd.driverSK

inner join DimDate ddd on fa.Accident_DateKey = ddd.DateSK

inner join DimControlled_Status dcs on fa.ConditionsKey = dcs.con_stausSK

inner join DimRoad drr on fa.RoadKey = drr.roadSK

Region (Location) Performance Analysis



Region(Location) Performance Analysis

Region name	Number of Casualties	Number of Vehicles
[Region_name]	[Sum(Number_of_Casualties)]	[Sum(Number_of_Vehicles)]
Total	[Sum(Number_of_Casualties)]	[Sum(Number_of_Vehicles)]

Text Box Properties

General
Number
Alignment
Font
Border
Fill
Visibility
Interactive Sorting
Action

Change action options.

Enable as an action:

☐ None
☒ Go to report
☐ Go to bookmark
☐ Go to URL

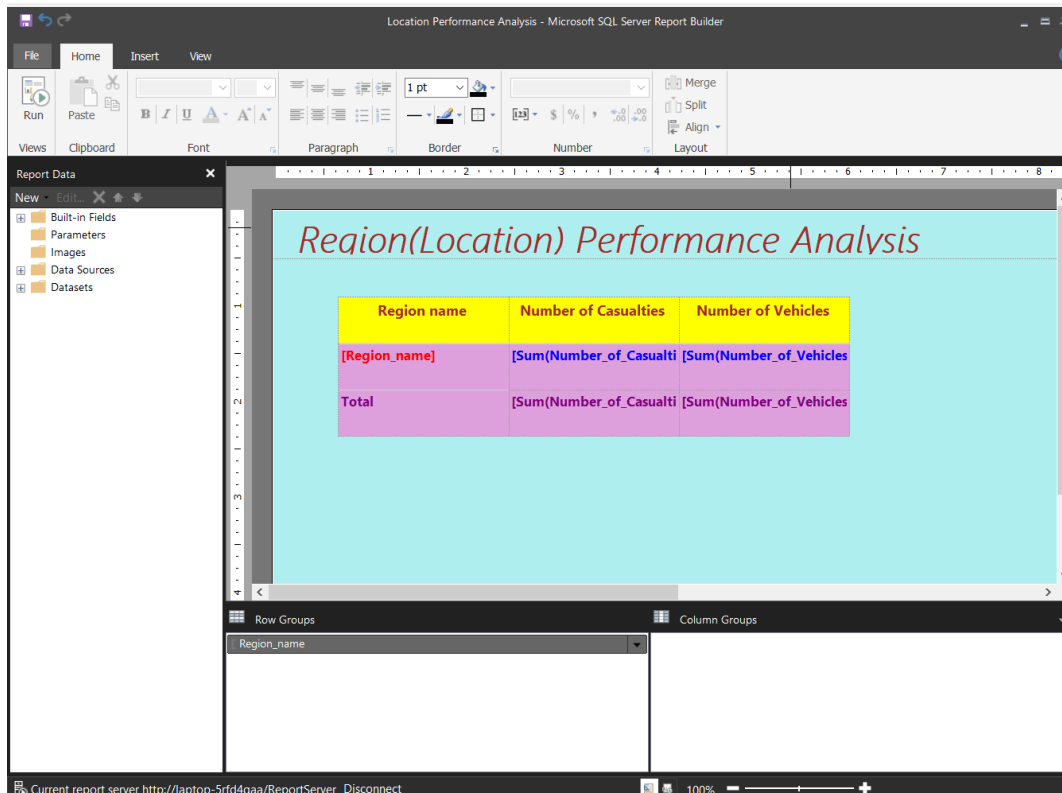
Specify a report:

/DWBI_ASSIGNMENT2/City(Location) Performance Analysis

Use these parameters to run the report:

Name	Value	Omit
Region	[Region_name] <input type="button" value="fx"/>	<input type="button" value="fx"/>

City (Location) Performance Analysis



SQL Server Reporting Services

★ Favorites □ Browse

Home > DWBL_ASSIGNMENT2 > Location Performance Analysis

1 of 1 100% Find | Next

Region(Location) Performance Analysis

Region name	Number of Casualties	Number of Vehicles
East Midlands	13008	17868
East of England	13112	18010
London	10890	15130
North East	10955	15056
North West	7158	9825
South East	10060	13739
South West	7334	10042
West Midlands	18536	25537
Yorkshire and The Humber	10895	14881

SQL Server Reporting Services

★ Favorites □ Browse

[Home](#) > [DWBL_ASSIGNMENT2](#) > Location Performance Analysis

|< < 1 of 1 > >| ↺ ↻ 100% ▾ 📄 🖨️ Find | Next

Region(Location) Performance Analysis

Region name	Number of Casualties	Number of Vehicles
East Midlands	13008	17868
East of England	13112	18010
London	10890	15130
North East	10955	15056
North West	7158	9825
South East	10060	13739
South West	7334	10042
West Midlands	18536	25537
Yorkshire and The Humber	10895	14881