

Data Warehousing & Business Intelligence

(DS)

3rd Year,1st Semester

**Assignment 02**

Submitted to

Srilanka Institute of Information Technology

IT18125726

L.V.A Kumarage

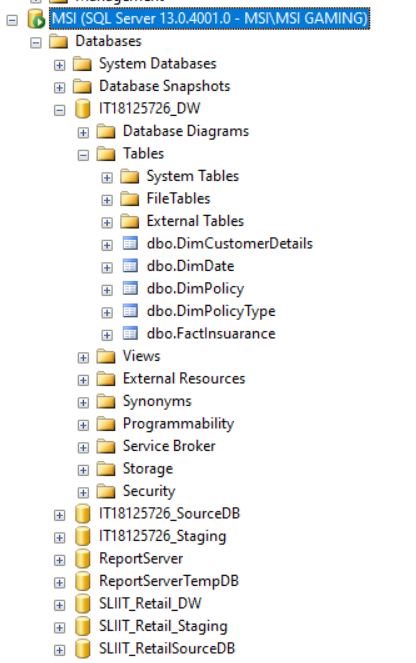
Weekday batch

## Step 1: Data source for the assignment 2

* **IT18125726\_DW** that I have implemented and loaded with data in Assignment 1 as the data source for the assignment 2.

Diagram, schematic

Description automatically generated



## Step 2: SSAS Cube implementation

## Created a new Analysis Service Multidimensional and Data Mining Project called IT18125726\_SSAS\_Insuarance.

## Then I have configured each option from top to bottom as shown below to create a data cube

## Text Description automatically generated

## I created a data source based on new connection.

## 

## Once I completed the data source configuration, it created a new data source in solution explorer under data source.

## Creating data source view

## Then I right clicked on data source views and select “New Data Source View”. Then in the welcome screen of the wizard, clicked “Next” to continue the configuration.

## After that, I clicked on the “Add Related Tables” to automatically select and add the related dimension tables. It should automatically add FactInsuarance, DimCustomerDetails, DimPolicy, DimDate dimensions to the list. However, it did not automatically add the DimpolicyType. So, I manually selected them and added to the list.

## Once completed creating data source view, the newly created data source view as shown in the below screenshot.

## Timeline Description automatically generated

## Creating a Data Cute

## Once cube wizard configurations completed, it created a data cube as shown in the screenshot

## A picture containing graphical user interface Description automatically generated

## Then I have configured each dimension are shown in the table view.

## First, I configured the Dim Policy dimension

## The policy, Policy Type all have keys as the displayed values in this cube.

## Changing the Name Column property will enable a different column value to be used. The end user does not want to see the primary key for policy and, they would want to see the policy Name.

## So, I have changed the name NameColumn to Type.

## As shown in the below I have created a policy in the cube.

## Type Policy

This indicator is only a warning and will not prevent the cube from being deployed. It is only a saying no relationship can cause performance issues when using large dimensions with this hierarchy type.

Text

Description automatically generated

(Policy Hierarchy)

Hierarchies are useful in visual reporting tools to show the parent/child relationship between attributes. So, I have Created another hierarchy called Location hierarchy.

Graphical user interface, text

Description automatically generated

Text

Description automatically generated

(Location Hierarchy)

**Creating a KPI’s**

Then I have created 4 KPI’s based on my business requirement.

A picture containing text, screenshot, monitor, screen

Description automatically generated

**KPI Total Paid Amount**

A picture containing text, screenshot, screen

Description automatically generated

**KPI Customer Lifetime Value**

A picture containing text, screenshot, monitor, screen

Description automatically generated

**KPI Rewards Points**

A picture containing text, screenshot, monitor, screen

Description automatically generated

**KPI Total Claim Amount**

Then I have created one **user Role** and provide permissions to access to the data cube

Graphical user interface, text, application, chat or text message

Description automatically generated

Finally, I have Deployed the project, I got the deployment is successful message as shown below.

A screenshot of a computer

Description automatically generated with medium confidence

**Browsing the data cube**

Graphical user interface, text, application, chat or text message

Description automatically generated

**Step 3: Demonstration of OLAP operations**

Create an Excel report using MDX query

Using the power pivot, Power Query, Power view in excel allow us to create a semantic layer inside excel.

To connect the excel workbook and to get the data to the semantic layer we use MDX query.

I dragged and dropped state from location hierarchy in customer details dimension, policy type name from policy hierarchy in policy type dimension, Total claim Amount from fact Insurance under measures, Customer Lifetime Value **Value** KPIs and Customer Lifetime Value **Goal** KPI**s.**

Additionally, I had added ‘State’ from ‘Dim Customer details as a filter into the section above the data grid as shown below

After that the data grid as displayed as below. Filter

Graphical user interface, application, table

Description automatically generated

Then I clicked on the Design Mode button to view the MDX query of the configuration

Graphical user interface

Description automatically generated

Then I have used this query in an Excel sheet to generate a report through Excel.

**Connecting Excel to SSAS Cube using a MDX Query**

After pasting the generated query, clicked on validate to make sure the query does not contain any errors.

Graphical user interface, text, application, email

Description automatically generated

Then I have clicked on finish to extract the data the cube.

Graphical user interface, text, application, email

Description automatically generated

**Excel report 01**

* **Pivot**

In the below pivot table, I have statically summarized the data of a more extensive insurance table. This summary includes sum of total claim Amount, sum of customer lifetime value and count of customer lifetime value, which the pivot table groups together in a descriptive manner in a state wise. And using this pivot table we can visualize our data by giving them a different perspective and view. We can rotate the axis of the dimension and see different pattern of the same data.

**Chart

Description automatically generated**

* **Slice**

Slice is a rectangular subset of a cube, by choosing a single value for one of its dimensions.

So here I have used a slicer to filter data in table and graph by State wise. So, this blue colored highlighted area displays the Total claim amount in California state. Likewise, we can view total claim amount of each state (Arizona, California, Nevada, Oregon, Washington)

Chart, bar chart

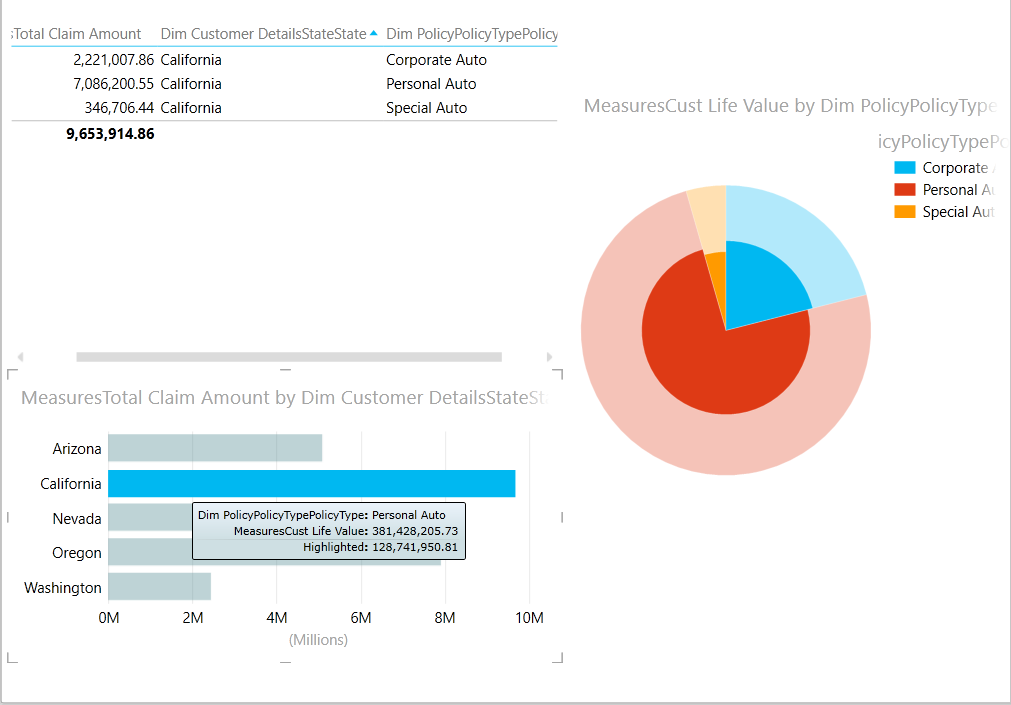
Description automatically generated

* **Dice**

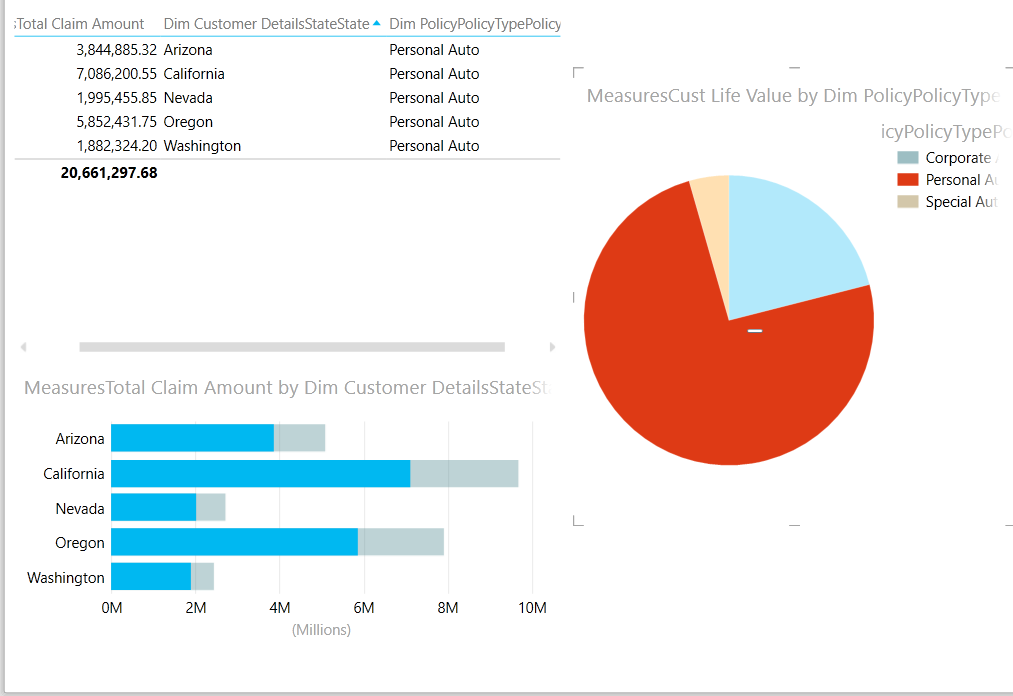
Selects two or more dimensions from a given cube and provides new sub-cube by selecting specific values on those selected dimensions.

Therefore, this report shows total claim amount in Arizona, California, Nevada, Oregon, Washington by Cooperate auto, Personal auto and Special auto policy type

As an instance in below figure, pie chart’s highlighted area emphasizes all values of 3 different types of policies belongs to state of California.



And, below figure pie chart’s highlighted area illustrate the personal auto policy type values which are belong to all states (Arizona, California, Nevada, Oregon, Washington) include in dataset.



* Personal Auto, Corporate Auto and Special Auto are policy type of policy.
* Arizona, California, Nevada, Oregon, and Washington are state of customer details

* **Roll-up and drill-down**

In this excel sheet it shows the sum of measures total amount. In the columns I have included a state attribute in customer details value and for the rows I have used policy hierarchy (PolicyType Policy). So, in this case report can be view by the state wise location wise

In this roll up and drill down report rows can drill down policy type to policy. So that report can view the policy claim amount .and the report can roll up from policy-to-policy type so we can view policy type total claim amount.

* **Roll up**

Climbing up a hierarchy of the dimension to aggregate data means the roll up OLAP operation in cubes

Graphical user interface, application

Description automatically generated

* **Drill down**

Stepping down a hierarchy of the dimension allowing navigation through details means the drill down OLAP operation in cube

**Graphical user interface

Description automatically generated with medium confidence**

**Excel Report 02**

* **Pivot**

In the below pivot table, I have statistically summarized the data to the more extensive Insurance table. This summary includes sum of Measures Reward points, sum of measures customer lifetime value and sum of measures total claim amount which the pivot table groups together in a meaningful way in Policy Type wise.

Graphical user interface

Description automatically generated

* **Slice**

So here I have used a slicer to filter data in table and graph by Policy Type wise. So, this blue colored highlighted area displays the Total claim amount in All three policy types. Likewise, we can view total claim amount of each Policy type (corporate Auto, Personal Auto, Special Auto)

**Chart, bar chart

Description automatically generated**

* **Dice**

So, this report shows Rewards Points in Arizona, California, Nevada, Oregon, Washington by Cooperate auto, Personal auto and Special auto policy type

As an instance in below **figure 01**, pie chart’s highlighted area emphasizes all values belongs to the reward points of 3 different types of policies belongs to state of Arizona.

And below **figure 02** pie chart’s highlighted area illustrate the corporate auto policy type values which are belong to all states (Arizona, California, Nevada, Oregon, Washington) include in dataset.

**Chart

Description automatically generated**

**Figure 01**

**Table

Description automatically generated**

**Figure 02**

* **Roll up**

Aggregate the values of the customer lifetime value and the reward points values from location wise to state wise. From this report User can go through more specific view of data to general view of data

**Graphical user interface

Description automatically generated**

* **Drill Down**

Split the values of the Customer lifetime value and reward point value from state wise to location code wise. Using this report user can go through general view of data to more specific view of data

**Graphical user interface, chart

Description automatically generatedStep 4: SSRS Reports**

SQL Server Reporting Services (SSRS) is a platform for creating, publishing, and managing mobile and paginated reports/dashboards, then delivering them to the right users in different ways, such as via a web browser, on their mobile device, or via email. Here I have mainly used **Report Builder** to create SSRS Reports because as a standalone application it will provide more freedom for users than using Report Design (SSDT) in visual studio.

Graphical user interface, application, Word

Description automatically generated

**Report 1: Report with a matrix**

In order retrieve dataset to the report builder, I have created data source and dataset in the report builder respectively and add below query to retrieve specific data (Before executing this query I have executed this in the sql server management studio to prevent the errors occur during the direct execution.

select dp.PolicyName,dpt.PolicyType,dcd.CustomerAlternateID,dcd.Gender,dcd.CustomerName,dcd.State,

dcd.LocationCode,dd.Month,dd.MonthName,dd.Year,fi.CustLifeValue,fi.MonthlyPremium,fi.TotalClaimAmount,

fi.TotalPaidAmount,fi.RewardPoints,fi.NoOfComplains,fi.MonthSinceLastClaim,fi.MonthSincePolicyIncept

from [dbo].[FactInsuarance] fi

inner join [dbo].[DimPolicy] dp on fi.[PolicyKey] = dp.[PolicySK]

inner join [dbo].[DimPolicyType] dpt

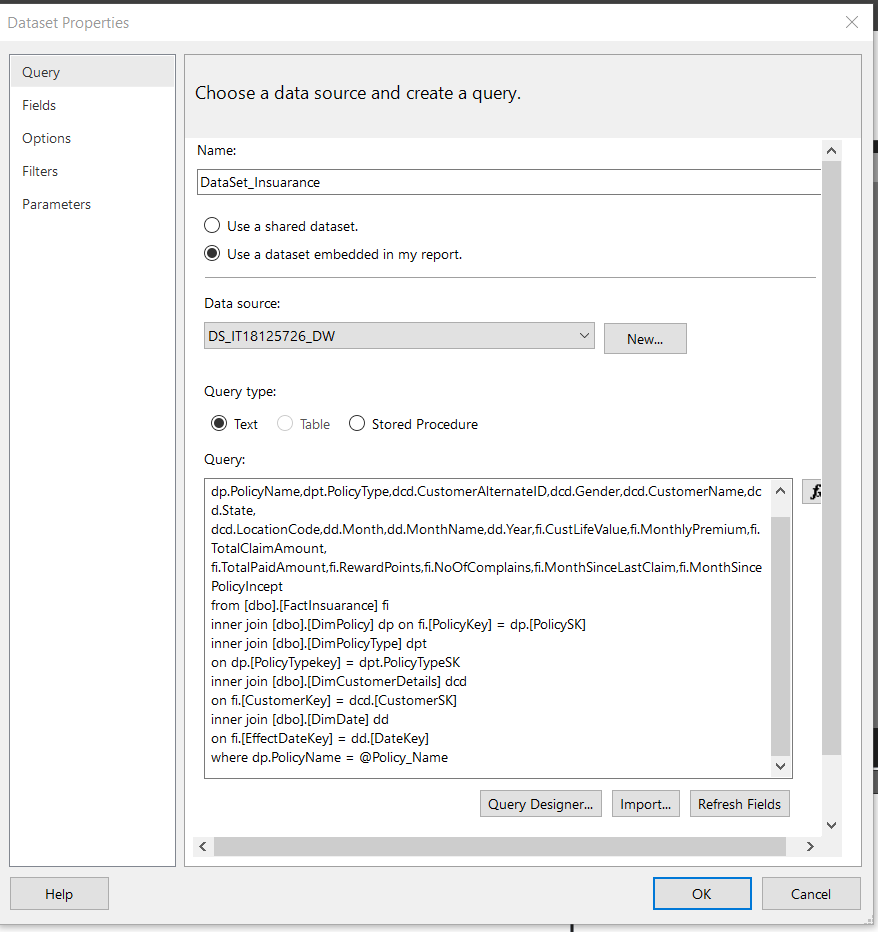
on dp.[PolicyTypekey] = dpt.PolicyTypeSK

inner join [dbo].[DimCustomerDetails] dcd

on fi.[CustomerKey] = dcd.[CustomerSK]

inner join [dbo].[DimDate] dd

on fi.[EffectDateKey] = dd.[DateKey]



Design view

Graphical user interface, application, table

Description automatically generated

Preview

Graphical user interface, application

Description automatically generated

Final Report View

**Graphical user interface, table

Description automatically generated**

Since I have data for Several years, we can limit (filter) data based on the year to keep the report clean and less crowded. Therefore, I have added filter (Year) to the dataset. I have added expression as ‘between’ and values between 2011 – 2012. This will sort the dataset values which are belongs only to the 2011 and 2012.

**Graphical user interface, text, application

Description automatically generated**

**Colum Chart for Matrix report**

As a visualization element I here used a column chart to provide an easier way to see and understand pattern in data. These visualization elements turn patterns which are invisible in raw data format into visible patterns, that people can understand intuitively.

Graphical user interface, application

Description automatically generated

**Report 2: Parameter Report**

**Multiple Parameterized Report – Policy Type and Policy Wise**

In this report I have used 2 parameters which have lists of values. Selection of 1st parameter value (Policy Type) changes the list of available values in the policy name parameter values

**Graphical user interface

Description automatically generated**

Selection of the Policy type (1st Parameter changed the listed value of 2nd parameter

Graphical user interface, application

Description automatically generated

Here I select Corporate Auto and Personal Auto as a first parameter. So according to the query I written,

Second parameter will filter the Policy names Related to the Policy types that I have choose in parameter 1 Then the report will display the values of selected areaGraphical user interface, website

Description automatically generated

**Colum Chart for Multiple Parameterized Report**

As a visualization element I here used a column chart to provide an easier way to see and understand pattern in data. These visualization elements turn patterns which are invisible in raw data format into visible patterns, that people can understand intuitively. This Colum chart will provide more understandability about the data and compare each policy type and policy wise values with each other.

Chart, bar chart

Description automatically generated

**Report 3: SSRS Drill Down Report:**

Drill down report will take users from general view of the data to a more specific view, enabling them to dig deeper into the layers in a hierarchy.

Annual Monthly Premium Report

Chart

Description automatically generated with low confidence

Design View

Graphical user interface, application

Description automatically generated

Preview View

the user has clicked the plus signs (+) in the report to show detail data

Graphical user interface, application

Description automatically generated

In this SSRS Report it shows the Monthly Premium details in the insurance Policy wise and Year wise

For the rows I have included a Policy type hierarchy (PolicyType Policy) so that report can be view by Policy Type wise and Policy Wise.

Timeline, bar chart

Description automatically generated with medium confidence

Policy Type Policy

**Report 4: Create an SSRS drill-through report.**

Drill-through report allow users to pass from one report to another while analyzing the same set of data

Policy Effectiveness Analysis

Here I had compared and analysis connection between most famous policy type and policy type among clients between a year and the no of complain got from the users for the particular policy type and policy

select dp.PolicyName,dpt.PolicyType,dcd.CustomerAlternateID,dcd.Gender,dcd.CustomerName,dcd.State,

dcd.LocationCode,dd.Month,dd.MonthName,dd.Year,fi.CustLifeValue,fi.MonthlyPremium,fi.TotalClaimAmount,

fi.TotalPaidAmount,fi.RewardPoints,fi.NoOfComplains,fi.MonthSinceLastClaim,fi.MonthSincePolicyIncept

from [dbo].[FactInsuarance] fi

inner join [dbo].[DimPolicy] dp

on fi.[PolicyKey] = dp.[PolicySK]

inner join [dbo].[DimPolicyType] dpt

on dp.[PolicyTypekey] = dpt.PolicyTypeSK

inner join [dbo].[DimCustomerDetails] dcd

on fi.[CustomerKey] = dcd.[CustomerSK]

inner join [dbo].[DimDate] dd **SQL query for the Detail**

**Report**

on fi.[EffectDateKey] = dd.[DateKey]

where dpt.PolicyType = @Type

Here I have created a Parameter called Type. So that when I clicked any kind of policy type it will show the detailed report about Customer Lifetime value and no. of complains from Policy Type Wise and Policy wise.

Chart, histogram, waterfall chart

Description automatically generated

**Design view of summary Graph**

Summary Report

Graphical user interface

Description automatically generated

In this report customer lifetime value is listed according to the policy type. When user click one of the policies type it will show the detailed report of the policies related to the clicked policy type. (Here I have clicked a personal Auto policy type in summary report and it will direct me to the detail report which have the details related to the **personal Auto** policy type’s policies

Detailed Report

Graphical user interface, application

Description automatically generated

As above query shows I have created a ‘**type’** parameter using where clause. And it is not necessary to fill this parameter by the user. instead, need a value for this parameter to be send by the first level report ‘Policy Effectiveness Analysis’.

Right clicked on the parameter ‘Type’ and select Parameter Properties to open the Parameter Properties window. Set the Select parameter visibility as Hidden.

Graphical user interface, text, application

Description automatically generated

To edit the first level report ‘Policy Effectiveness Analysis’ and enable a click event to allow the user to click on a Policy Type (a column in the column chart) to drill-through to the second level report I created. To do so, open the first level repot ‘Policy Effectiveness Analysis’ in Repot Builder. Right clicked on any column of the ‘Policy-wise Revenue’ chart and select Series Properties… to open Series Properties window.

In the **Series Properties** window, clicked to **Action** section and select **Go to report**.

Under, specify a report, select my second level report ‘Policy Effectiveness Analysis – L2’ using the Browse… button. Click on Add button and select ‘Type’ for Name (this is defined in the ‘Policy Effectiveness Analysis – L2’ report). Select ‘[PolicyType]’ for Value and then I clicked OK.

Graphical user interface, application

Description automatically generated

I have repeated the same step for the **no of** **complain submitted graph**.

**References: -**

* <https://courseweb.sliit.lk/course/view.php?id=4696>
* <https://docs.microsoft.com/en-us/sql/reporting-services/create-deploy-and-manage-mobile-and-paginated-reports?view=sql-server-ver15>
* <https://www.contextures.com/excel-pivot-table-report-filters.html>
* <https://support.microsoft.com/en-us/office/create-a-pivottable-to-analyze-worksheet-data-a9a84538-bfe9-40a9-a8e9-f99134456576>
* <https://www.youtube.com/watch?v=8gGcPDNXwo4&t=914s>
* <https://www.youtube.com/watch?v=_3wf2Cg8WHo&feature=youtu.be>

**END**