

**ZES's
ZEAL COLLEGE OF ENGINEERING AND RESEARCH, NARHE, PUNE**

**Department of Computer Engineering
SEMESTER-II
[AY 2022 - 2023]**



**LABORATORY MANUAL
on
*Laboratory Practice VI Lab-Business Intelligence***

Savitribai Phule Pune University
Fourth Year of Computer Engineering (2019 Course)
PART II 410253 : Elective VI
410253(C) : Business Intelligence

Teaching Scheme: PR: 02 Hours/Week	Credit 01	Examination Scheme: TW: 50Marks
---	----------------------------	--

Course Objectives:

- To understand the fundamental concepts and techniques of natural language processing (NLP)
- To understand Digital Image Processing Concepts
- To learn the fundamentals of software defined networks
- Explore the knowledge of adaptive filtering and Multi-rate DSP
- To be familiar with the various application areas of soft computing.
- To introduce the concepts and components of Business Intelligence (BI)
- To study Quantum Algorithms and apply these to develop hybrid solutions

Course Outcomes:

On completion of this course, the students will be able to

CO1: Apply basic principles of elective subjects to problem solving and modeling.

CO2: Use tools and techniques in the area of software development to build mini projects.

CO3: Design and develop applications on subjects of their choice.

CO4: Generate and manage deployment, administration & security

List of Assignments

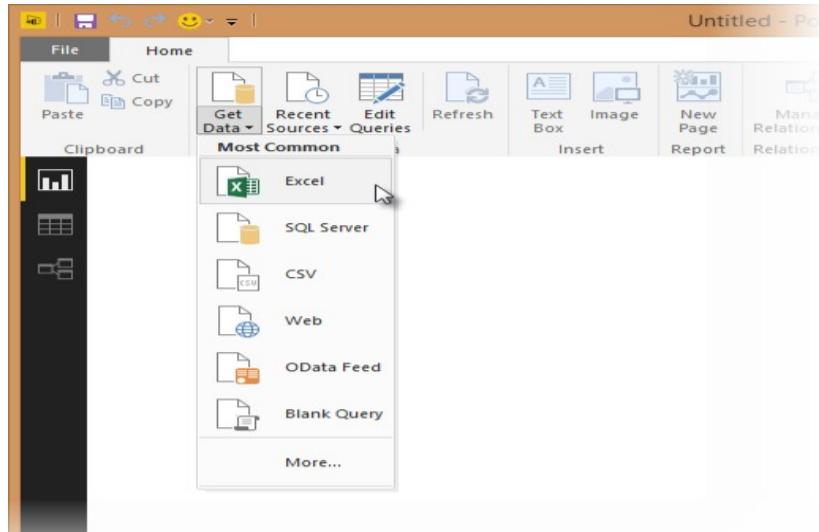
Sr. No.	TITLE	Page No.
Group 1		
1.	Import the legacy data from different sources such as (Excel, Sql Server, Oracle etc.) and load in the target system. (You can download sample database such as Adventure works, Northwind, foodmart etc.)	4-5
2.	Perform the Extraction Transformation and Loading (ETL) process to construct the database in the Sql server	6-16
3.	Create the cube with suitable dimension and fact tables based on ROLAP, MOLAP and HOLAP model.	17-38
4.	Import the data warehouse data in Microsoft Excel and create the Pivot table and Pivot Chart	39-43
5.	Perform the data classification using classification algorithm. Or Perform the data clustering using clustering algorithm.	44-46
Group 2		
6.	Mini Project: Each group of 4 Students (max) assigned one case study for this; A BI report must be prepared outlining the following steps: a) Problem definition, identifying which data mining task is needed. b) Identify and use a standard data mining dataset available for the problem.	

Assignment No. – 1

Aim: Import the legacy data from different sources such as (Excel, Sql Server, Oracle etc.) and load in the target system. (You can download sample database such as Adventure works, Northwind, foodmart etc.)

Importing Excel Data

- 1) Launch Power BI Desktop.
- 2) From the Home ribbon, select Get Data. Excel is one of the Most Common data connections,



so you can select it directly from the Get Data menu

- 3) If you select the Get Data button directly, you can also select File > Excel and select Connect.
- 4) In the Open File dialog box, select the Products.xlsx file.
- 5) In the Navigator pane, select the Products table and then select Edit.

ProductID	ProductName	SupplierID	CategoryID	Quantity
1	Chai	2	1	20
2	Chang	2	2	24
3	Aniseed Syrup	2	2	12
4	Chef Anton's Cajun Seasoning	2	2	48
5	Chef Anton's Gumbo Mix	2	2	36
6	Grandma's Boysenberry Spread	3	2	15
7	Uncle Bob's Organic Dried Pears	3	3	12
8	Northwoods Cranberry Sauce	3	2	12
9	Mishi Kobe Niku	4	5	18
10	Iwara	4	8	12
11	Queso Cabriles	5	4	1
12	Queso Manchego La Pastora	5	4	10
13	Konbu	6	8	2
14	Tofu	6	7	40
15	Genen Shouyu	6	2	24
16	Pavlova	7	3	32
17	Alice Mutton	7	5	20
18	Carnarvon Tigers	7	8	14
19	Teatime Chocolate Biscuits	8	9	16
20	Sir Rodney's Marmalade	8	3	30
21	Sir Rodney's Scones	8	3	24
22	Gustaf's Knäckebrodd	9	5	24

Importing Data from OData Feed

In this task, you will bring in order data .This steps represents connecting to a sales system. You import data into Power BI Desktop from the sample Northwind OData feed at the following URL,which you can copy (and then paste) in the step below:

<https://services.odata.org/v3/northwind/northwind.svc>

Connect to an OData feed:

- 1) From the Home ribbon tab in Query Editor, select Get Data.
- 2) Browse to the OData Feed data source.
- 3) In the OData Feed dialog box, paste the URL for the Northwind OData feed.
- 4) Select OK.
- 5) In the Navigator pane, select the Orders table, and then select Edit.

The screenshot shows the Power BI Desktop application. On the left, the Navigator pane is open, displaying a tree view of available data sources. One node under 'http://services.odata.org/v3/northwind/northwind.svc' is expanded, showing tables like 'Alphabetical_List_of_products', 'Categories', 'Customer_Sales_for_1997', etc. A checkbox next to 'Orders' is checked, indicating it's selected. To the right of the Navigator is a preview window titled 'Orders'. It shows a table with columns: OrderID, CustomerID, EmployeeID, OrderDate, and RequiredDate. The table contains 20 rows of order data from July 1996. At the bottom of the preview window are 'OK' and 'Cancel' buttons, with 'OK' being highlighted.

Note - You can click a table name, without selecting the checkbox, to see a preview.

Conclusion:

Hence we have import the legacy data from different sources such as (Excel, Sql Server, Oracle etc.) and load in the target system.

Assignment Question:

- 1.Define Legacy Data.
- 2.List the target system to import legacy data.
- 3.List the sample datasets you have used while implementation of assignment

Assignment No.-2

Aim: Perform the Extraction Transformation and Loading (ETL) process to construct the database in the Sql server

Step 1: Data Extraction

The data extraction is first step of ETL. There are 2 Types of Data Extraction

1. Full Extraction: All the data from source systems or operational systems gets extracted to staging area. (Initial Load)
2. Partial Extraction: Sometimes we get notification from the source system to update specific date. It is called as Delta load.

Source System Performance: The Extraction strategies should not affect source system performance.

Step 2: Data Transformation:

The data transformation is second step. After extracting the data there is big need to do the transformation as per the target system. I would like to give you some bullet points of Data Transformation.

- Data Extracted from source system is in to Raw format. We need to transform it before loading in to target server.
- Data has to be cleaned, mapped and transformed

• There are following important steps of Data Transformation:

1. **Selection :** Select data to load in target
2. **Matching:** Match the data with target system
3. **Data Transforming:** We need to change data as per target table structures

Real life examples of Data Transformation:

- Standardizing data: Data is fetched from multiple sources so it needs to be standardized as per the target system.
- Character set conversion: Need to transform the character sets as per the target systems. (Firstname and last name example)
- Calculated and derived values: In source system there is first val and second val and in target we need the calculation of first val and second val.
- Data Conversion in different formats: If in source system date in in DDMMYY format and in target the date is in DDMONYYYY format then this transformation needs to be done at transformation phase.

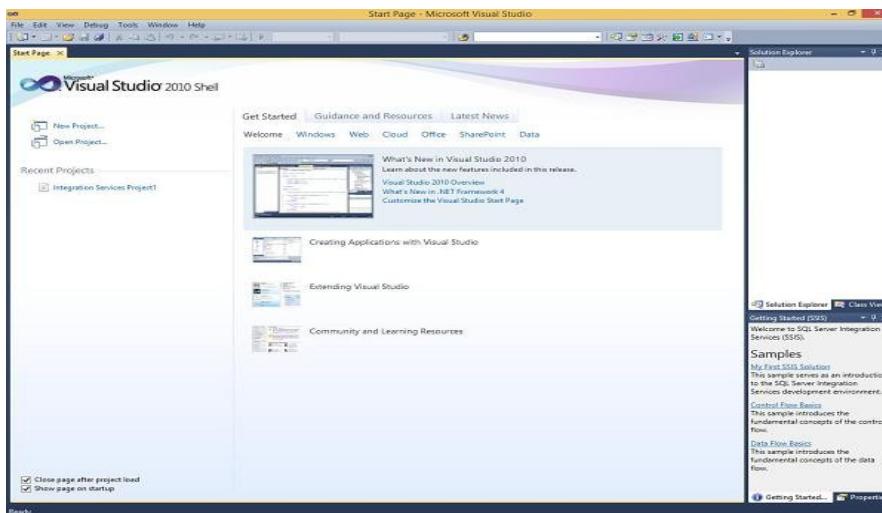
Step 3 : Data Loading

- Data loading phase loads the prepared data from staging tables to main tables.

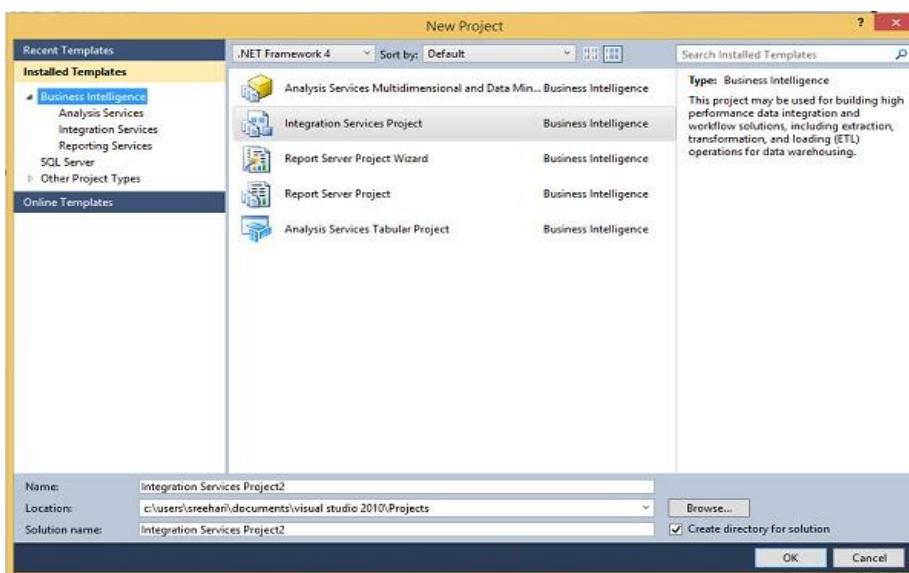
ETL process in SQL Server:

Following are the steps to open BIDS\SSDT.

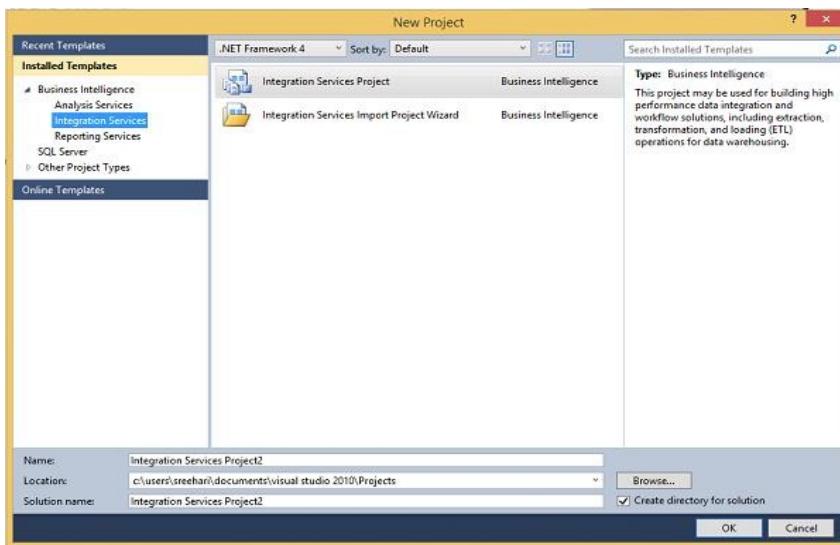
Step 1 – Open either BIDS\SSDT based on the version from the Microsoft SQL Server programs group. The following screen appears.



Step 2 – The above screen shows SSDT has opened. Go to file at the top left corner in the above image and click New. Select project and the following screen opens.



Step 3 – Select Integration Services under Business Intelligence on the top left corner in the above screen to get the following screen.



Step 4 – In the above screen, select either Integration Services Project or Integration Services Import Project Wizard based on your requirement to develop\create the package.

Modes

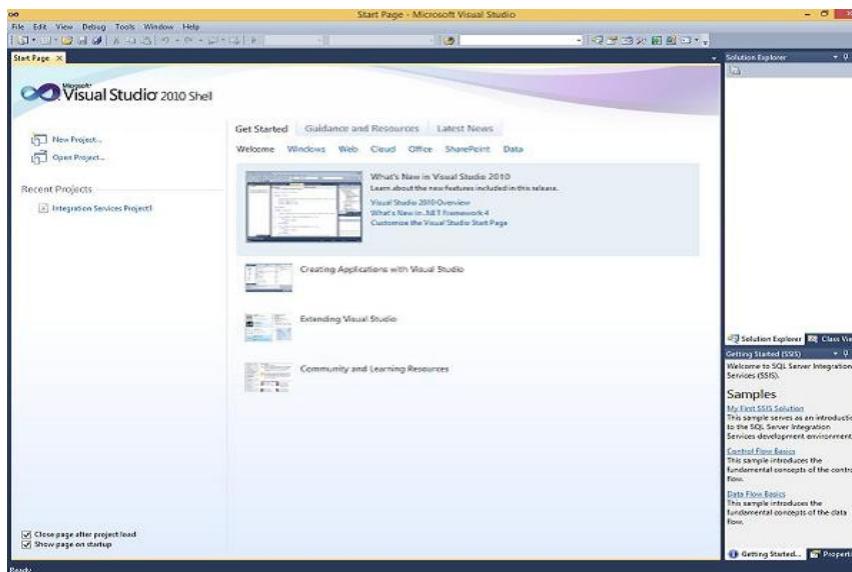
There are two modes – Native Mode (SQL Server Mode) and Share Point Mode.

Models

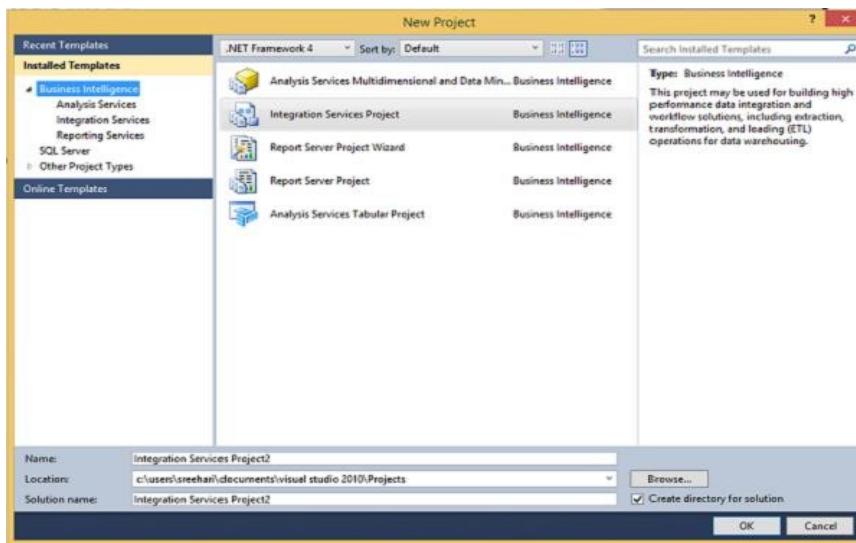
There are two models – Tabular Model (For Team and Personal Analysis) and Multi Dimensions Model (For Corporate Analysis).

The BIDS (Business Intelligence Studio till 2008 R2) and SSDT (SQL Server Data Tools from 2012) are environments to work with SSAS.

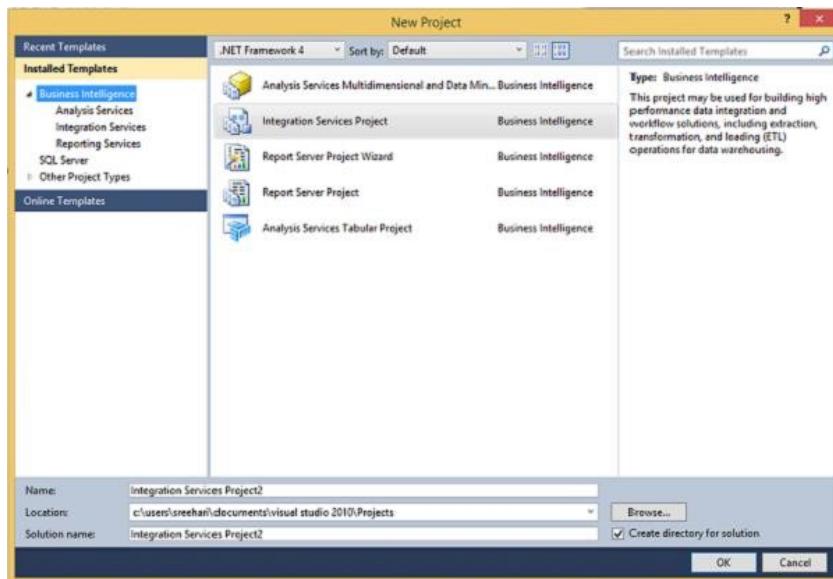
Step 1 – Open either BIDS\SSDT based on the version from the Microsoft SQL Server programs group. The following screen will appear.



Step 2 – The above screen shows SSDT has opened .Go to file on the top left corner in the above image and click New. Select project and the following screen opens.



Step 3 – Select Analysis Services in the above screen under Business Intelligence as seen on the top left corner. The following screen pops up.



Step 4 – In the above screen, select any one option from the listed five options based on your requirement to work with Analysis services.

ETL Process in Power BI

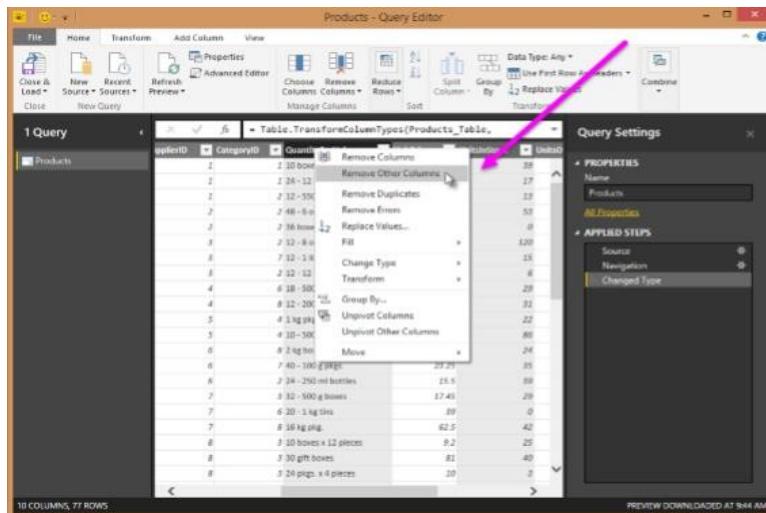
1) Remove other columns to only display columns of interest

In this step you remove all columns excepts Product ID, Product Name, UnitinStock and QualityPerUnit

Power BI Desktop includes Query Editor, which is where you shape and transform your data connections. Query Editor opens automatically when you select Edit from Navigator. You can also open the Query Editor by selecting Edit Queries from the Home ribbon in Power BI Desktop. The following steps are performed in Query Editor.

1. In Query Editor, select the ProductID, ProductName, QuantityPerUnit, and UnitsInStock columns (use Ctrl+Click to select more than one column, or Shift+Click to select columns that are beside each other).

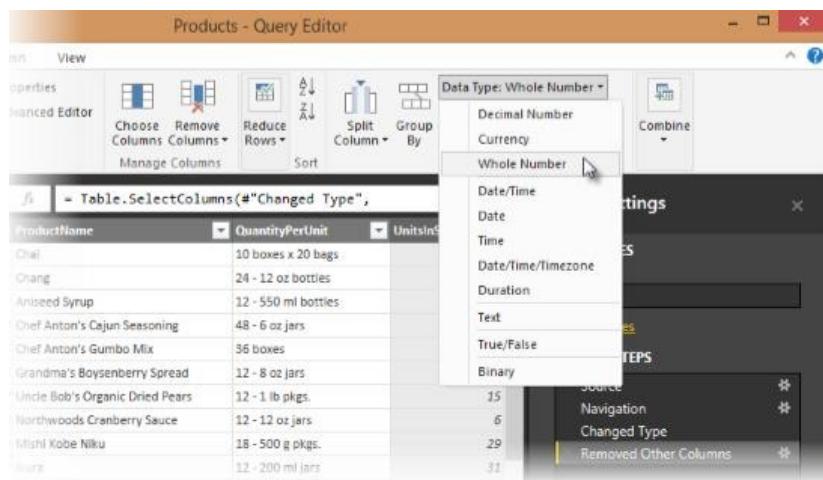
2. Select Remove Columns > Remove Other Columns from the ribbon, or right-click on a column header and click Remove Other Columns.



3. Change the data type of the UnitsInStock column

When Query Editor connects to data, it reviews each field and to determine the best data type. For the Excel workbook, products in stock will always be a whole number, so in this step you confirm the UnitsInStock column's datatype is Whole Number.

1. Select the UnitsInStock column.
2. Select the Data Type drop-down button in the Home ribbon.
3. If not already a Whole Number, select Whole Number for data type from the drop down (the Data Type: button also displays the data type for the current selection).



3. Expand the Order_Details table

The Orders table contains a reference to a Details table, which contains the individual products that were included in each Order. When you connect to data sources with multiples tables (such as a relational database) you can use these references to build up your query

In this step, you expand the Order_Details table that is related to the Orders table, to combine the ProductID, UnitPrice, and Quantity columns from Order_Details into the Orders table. This is a representation of the data in these tables:

The Expand operation combines columns from a related table into a subject table. When the query runs, rows from the related table (Order_Details) are combined into rows from the subject table (Orders).

After you expand the Order_Details table, three new columns and additional rows are added to the Orders table, one for each row in the nested or related table.

1.In the Query View, scroll to the Order_Details column.

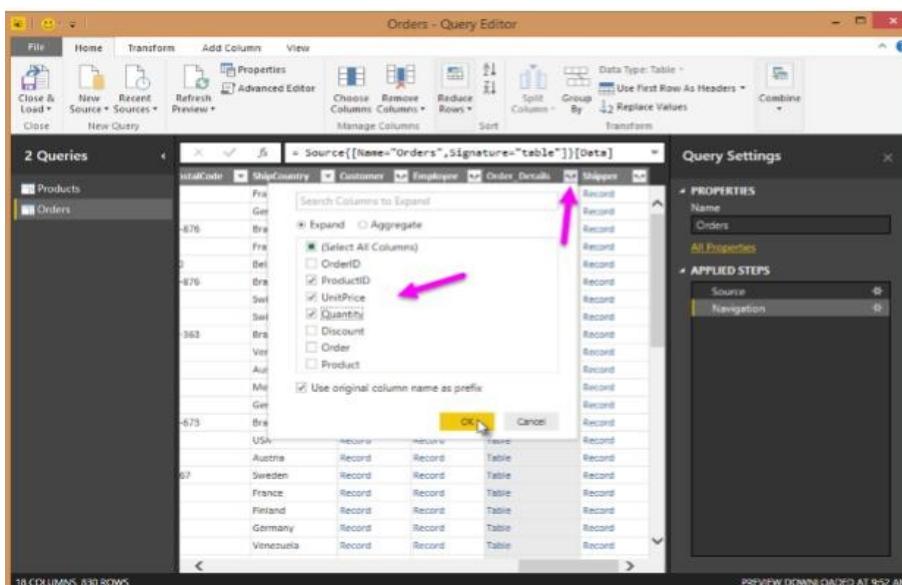
2.In the Order_Details column, select the expand icon ().

3.In the Expand drop-down:

a.Select (Select All Columns) to clear all columns.

b.Select ProductID, UnitPrice, and Quantity.

c.Click OK.



4. Calculate the line total for each Order_Details row

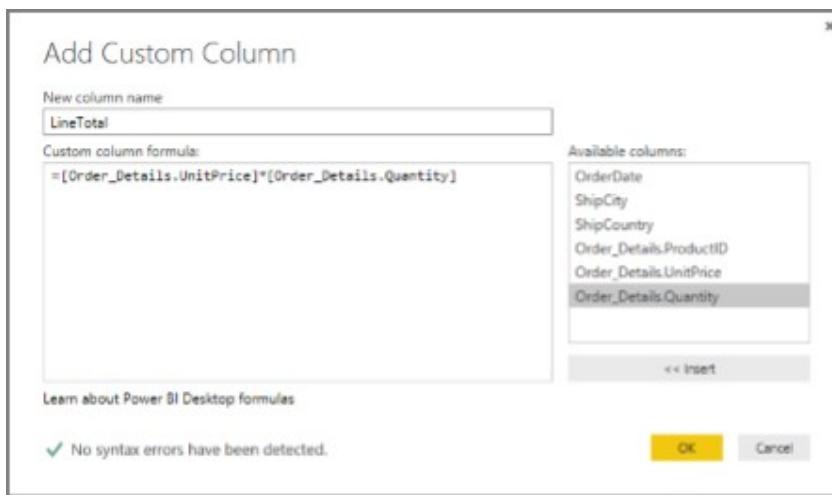
Power BI Desktop lets you to create calculations based on the columns you are importing, so you can enrich the data that you connect to. In this step, you create a Custom Column to calculate the line total for each Order_Details row.

Calculate the line total for each Order_Details row:

1. In the Add Column ribbon tab, click Add Custom Column.

	LineTotal	ShipCountry
	168	France
	98	France
	174	France
	167.4	Germany
	1696	Germany
	77	Brazil
	1484	Brazil
	252	Brazil
	100.8	France

2. In the Add Custom Column dialog box, in the Custom Column Formula textbox, enter `[Order_Details.UnitPrice] * [Order_Details.Quantity]`.
3. In the New column name textbox, enter `LineTotal`.
4. Click OK.



5. Rename and reorder columns in the query

In this step you finish making the model easy to work with when creating reports, by renaming the final columns and changing their order.

1. In Query Editor, drag the LineTotal column to the left, after ShipCountry.

2. Remove the Order_Details. prefix from the Order_Details.ProductID, Order_Details.UnitPrice and Order_Details.Quantity columns, by double-clicking on each column header, and then deleting that text from the column name.

6. Combine the Products and Total Sales queries

Power BI Desktop does not require you to combine queries to report on them. Instead, you can create Relationships between datasets. These relationships can be created on any column that is common to your datasets

We have Orders and Products data that share a common 'ProductID' field, so we need to ensure there's a relationship between them in the model we're using with Power BI Desktop. Simply specify in Power BI Desktop that the columns from each table are related (i.e. columns that have the same values). Power BI Desktop works out the direction and cardinality of the relationship for you. In some cases, it will even detect the relationships automatically.

In this task, you confirm that a relationship is established in Power BI Desktop between the Products and Total Sales queries

Step 1: Confirm the relationship between Products and Total Sales

1. First, we need to load the model that we created in Query Editor into Power BI Desktop. From the Home ribbon of Query Editor, select Close & Load.

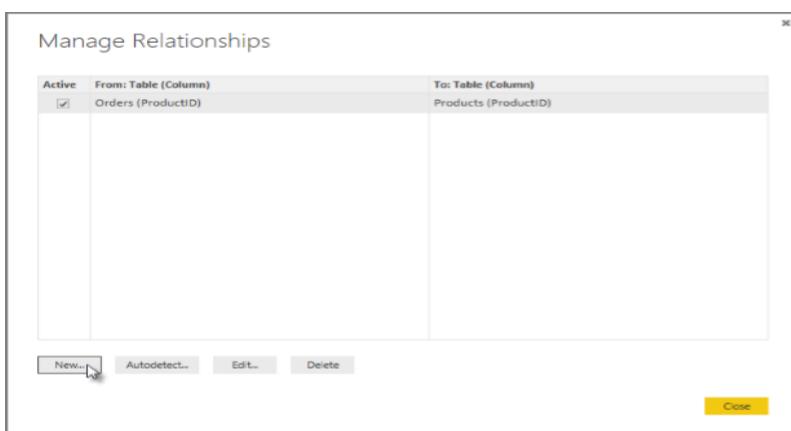
2. Power BI Desktop loads the data from the two queries.



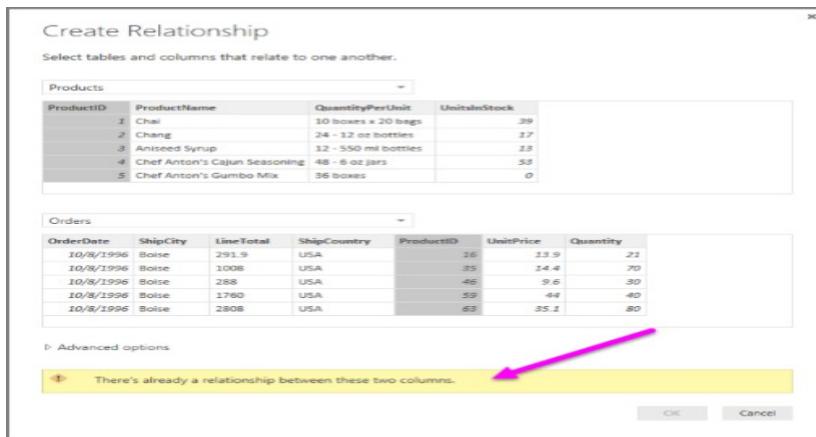
3. Once the data is loaded, select the Manage Relationships button Home ribbon.



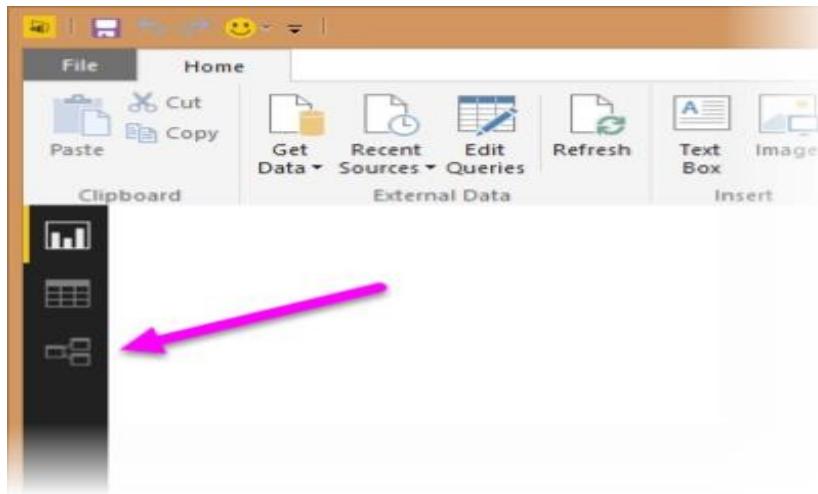
4. Select New button



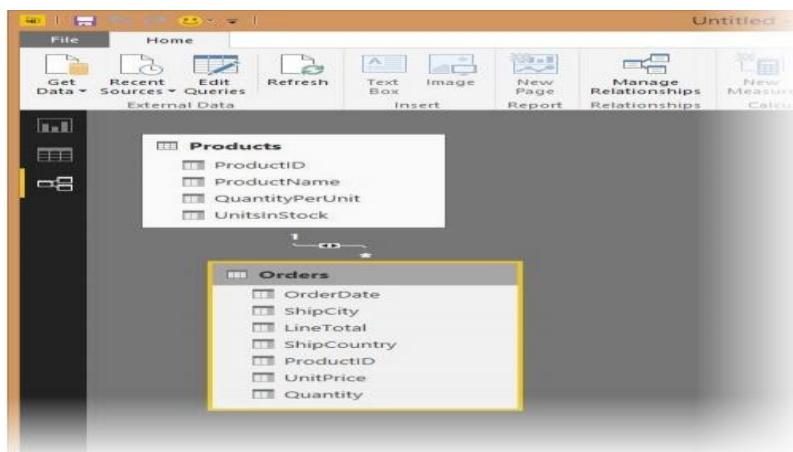
5. When we attempt to create the relationship, we see that one already exists! As shown in the Create Relationship dialog (by the shaded columns), the ProductsID fields in each query already have an established relationship.



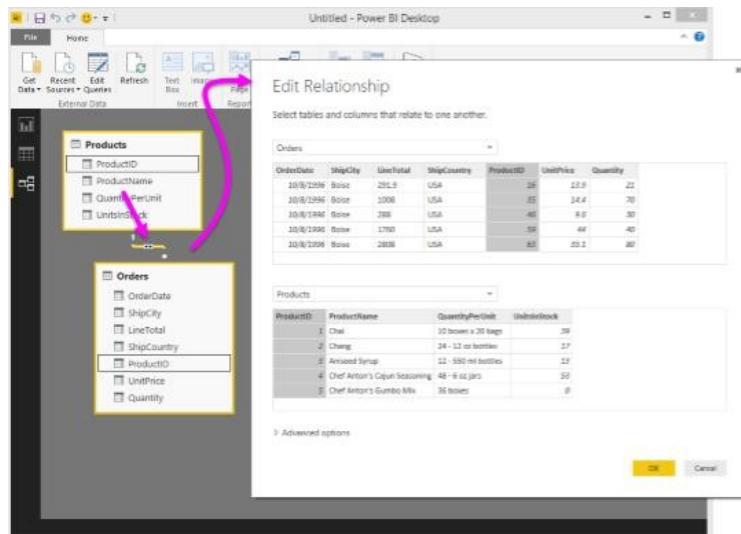
5. Select, Cancel and then Relationship view in Power BI Desktop



6. We see the following, which visualizes the relationship between the queries.



7. When you double-click the arrow on the line that connects the two queries, an Edit Relationship dialog appears.



8. No need to make any changes, so we'll just select Cancel to close the Edit Relationship dialog.

Conclusion:

Hence we performed the Extraction Transformation and Loading (ETL) process to construct the database in the Power BI.

Assignment Question:

1. How to import an Excel file into the Oracle table
2. Is Graphical visualization better than text data? Justify your answer and explain different data visualization techniques.
3. Explain in detail what is extract/transform/load (ETL)

Assignment No.-3

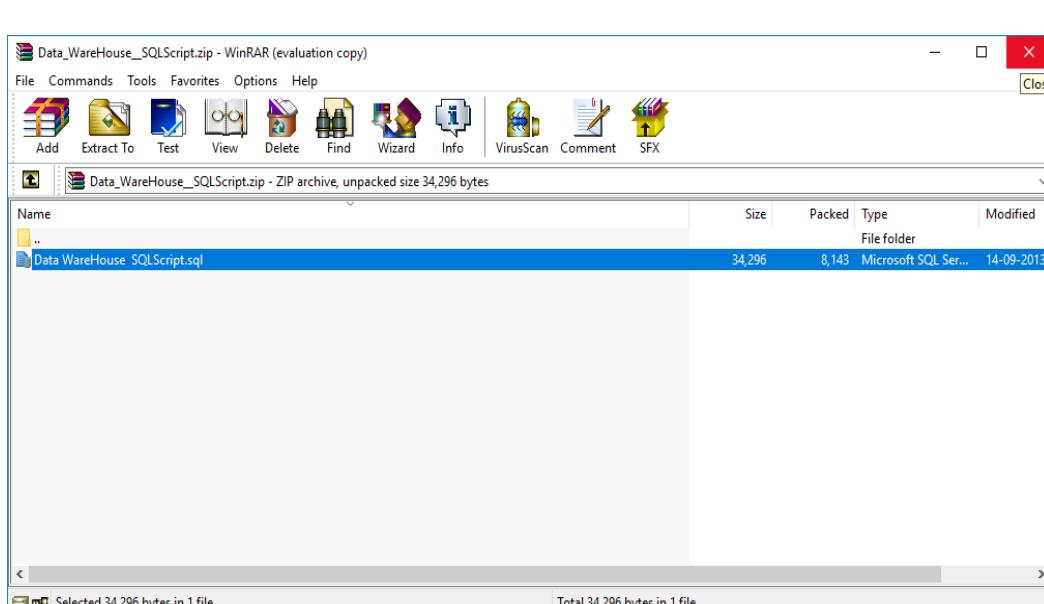
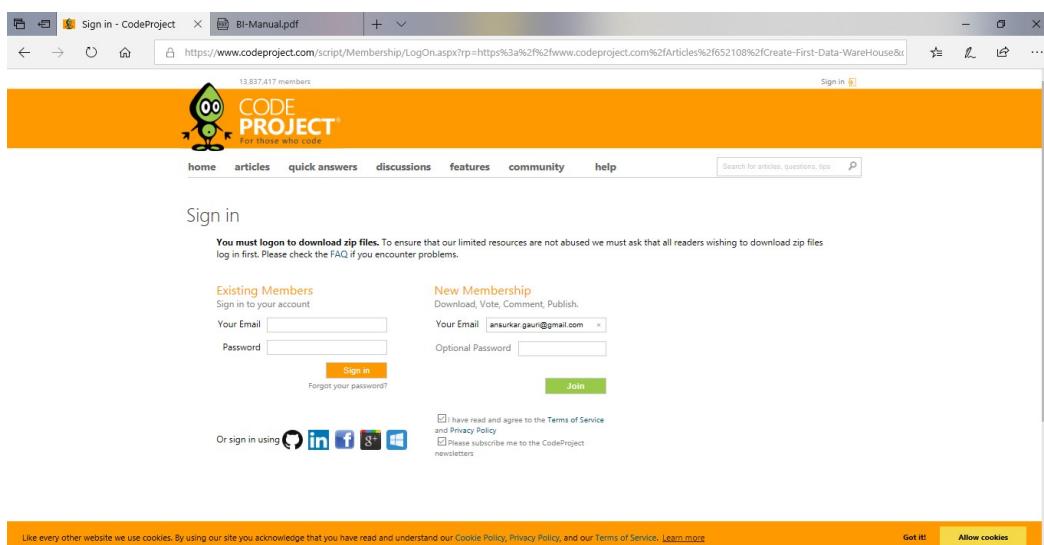
Aim: Create the cube with suitable dimension and fact tables based on ROLAP, MOLAP and HOLAP model.

Step 1: Creating Data Warehouse

Let us execute our T-SQL Script to create data warehouse with fact tables, dimensions and populate them with appropriate test values.

Download T-SQL script attached with this article for creation of Sales Data Warehouse or download from this article “Create First Data Warehouse” and run it in your SQL Server.

Downloading "Data_Warehouse_SQLScript.zip" from the article
<https://www.codeproject.com/Articles/652108/Create-First-Data-Warehouse>

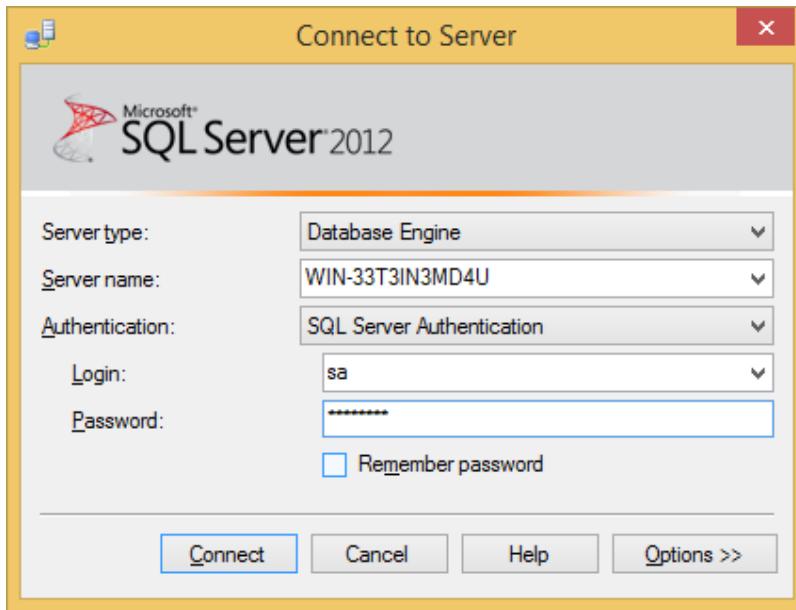


After downloading extract file in folder.

Follow the given steps to run the query in SSMS (SQL Server Management Studio).

1. Open SQL Server Management Studio 2012

2. Connect Database Engine



Password for sa: admin123 (as given during installation) Click Connect.

3. Open New Query editor

4. Copy paste Scripts given below in various steps in new query editor window one by one

5. To run the given SQL Script, press F5

6. It will create and populate “Sales_DW” database on your SQL Server
OR

1. Go to the extracted sql file and double click on it.

2. New Sql Query Editor will be opened containing Sales_DW Database.

```
--DROP DATABASE Sales_DW
GO
Create database Sales_DW
Go

--Create Customer dimension table in Data Warehouse which will hold customer personal details.

Create table DimCustomer
(
CustomerID int primary key identity,
CustomerAltID varchar(10) not null,
CustomerName varchar(50),
Gender varchar(20)
)
Go

--Fill the Customer dimension with sample Values

Insert Into DimCustomer(CustomerAltID,CustomerName,Gender)values
('IMI-001','Henry Ford','M'),
('IMI-002','Bill Gates','M'),
('IMI-003','Muskan Shaikh','F'),
('IMI-004','Richard Thrubin','M'),
('IMI-005','Emma Wattson','F');
Go

--Create basic level of Product Dimension table without considering any Category or Subcategory.

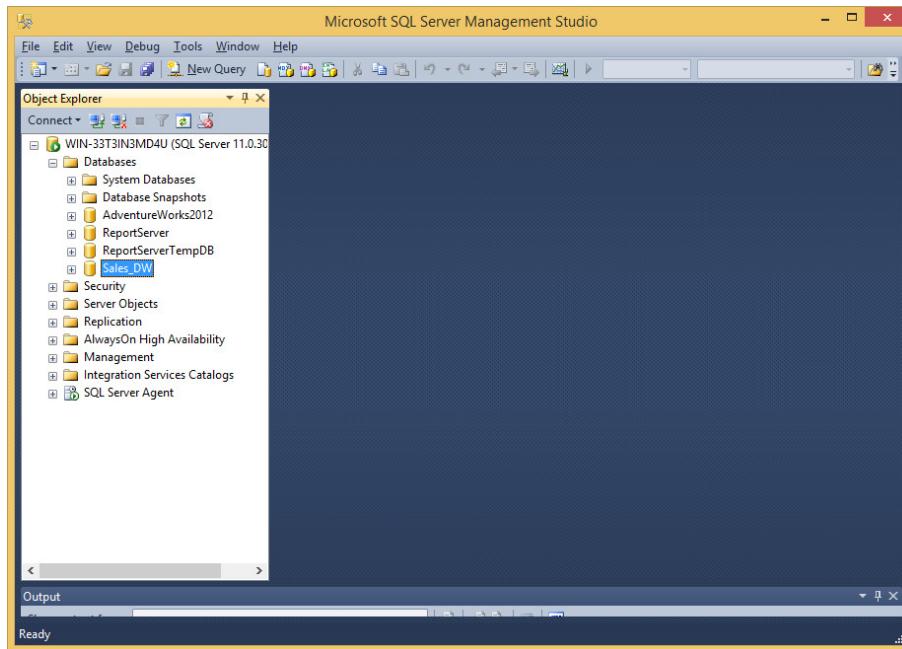
Create table DimProduct
(

```

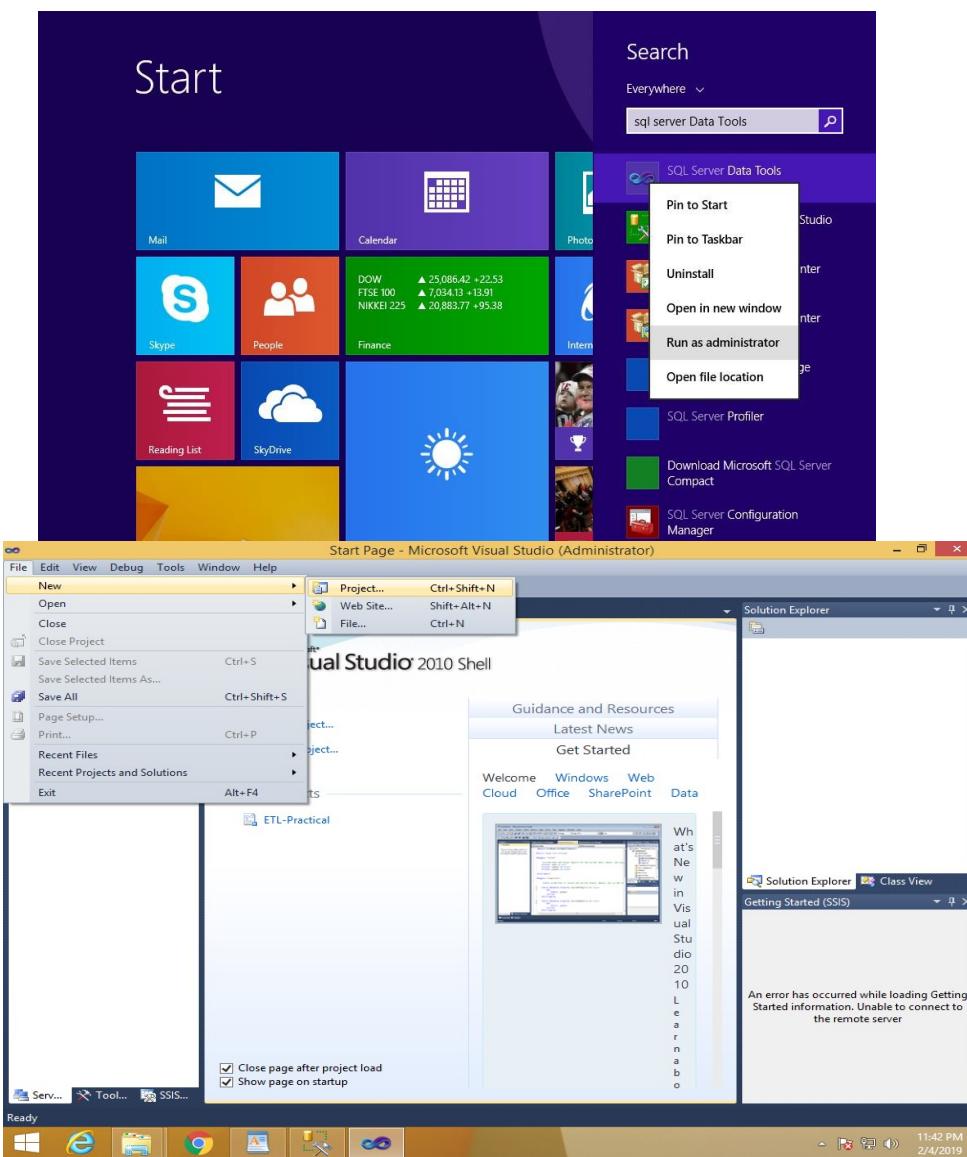
3. Click on executes or press F5 by selecting query one by one or directly click on Execute.

4. After completing execution save and close SQL Server Management studio &

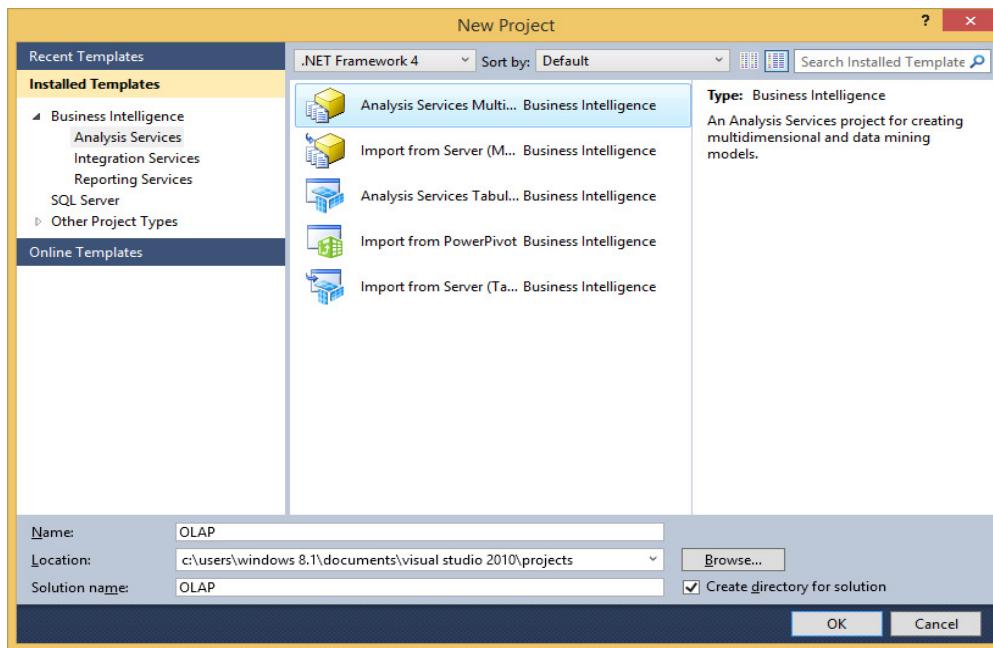
Reopen to see Sales_DW in Databases Tab.



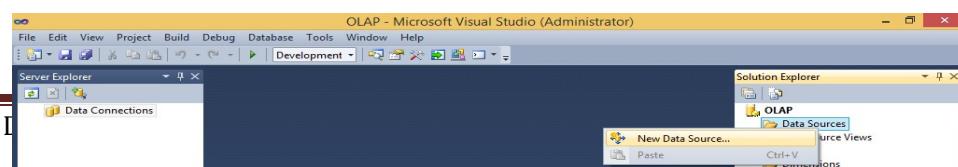
Step 2: Start SSDT environment and create New Data Source
Go to Sql Server Data Tools --> Right click and run as administrator

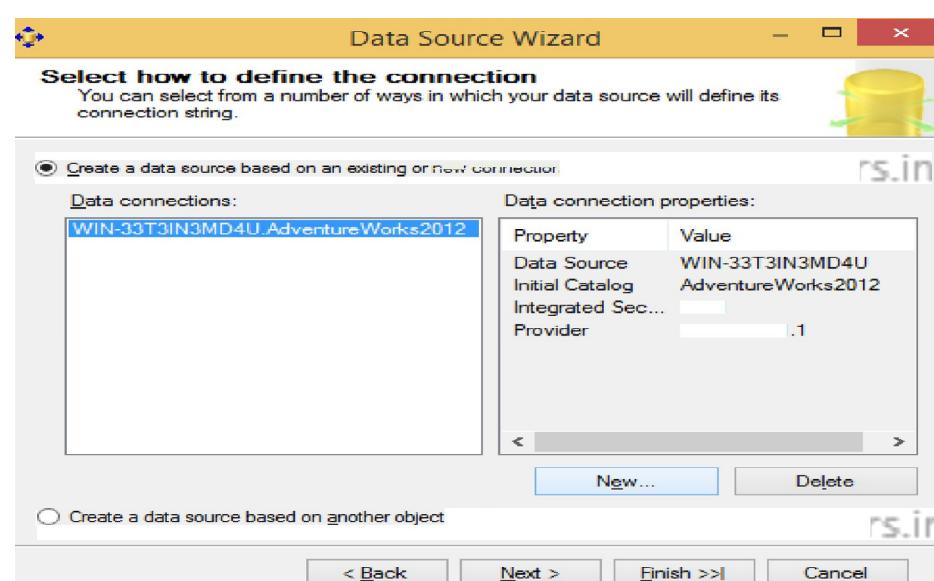


In Business Intelligence → Analysis Services Multidimensional and Data mining models → appropriate project name → click OK



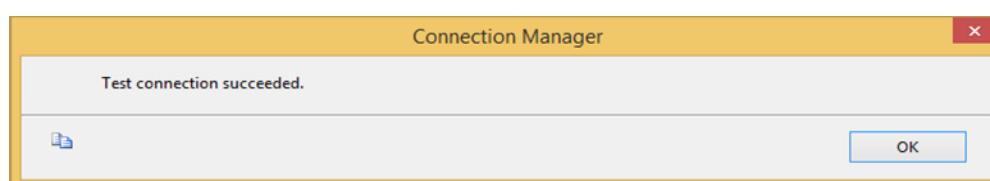
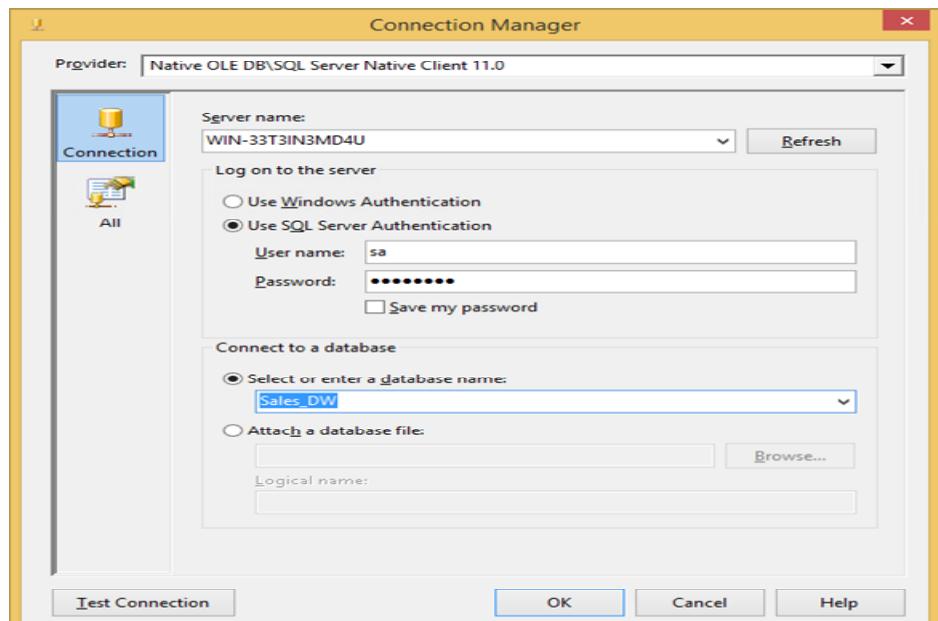
Right click on Data Sources in solution explorer → New Data Source



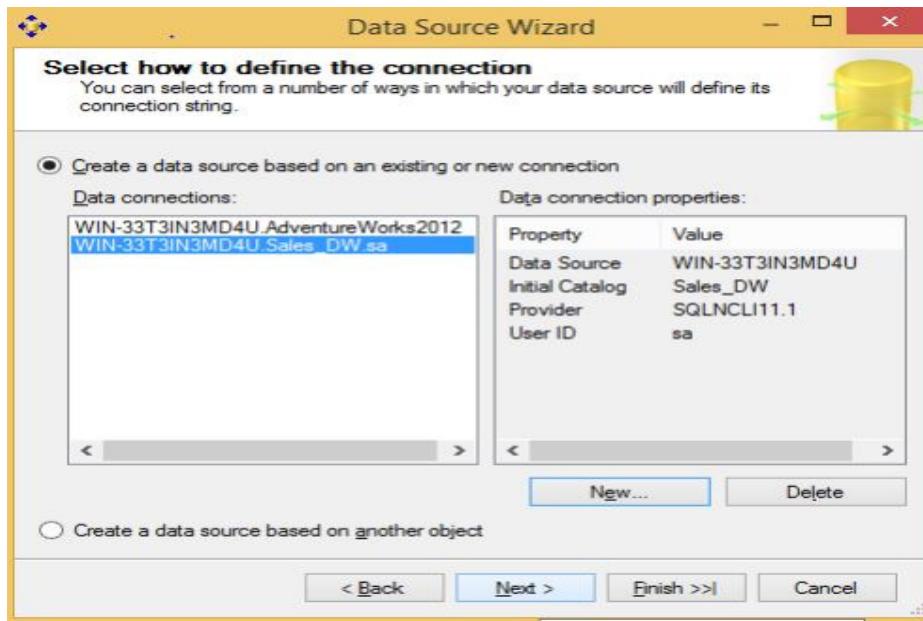
Data Source Wizard appears**Click on New**

Select Server Name → select Use SQL Server Authentication → Select or enter a database name (Sales_DW)

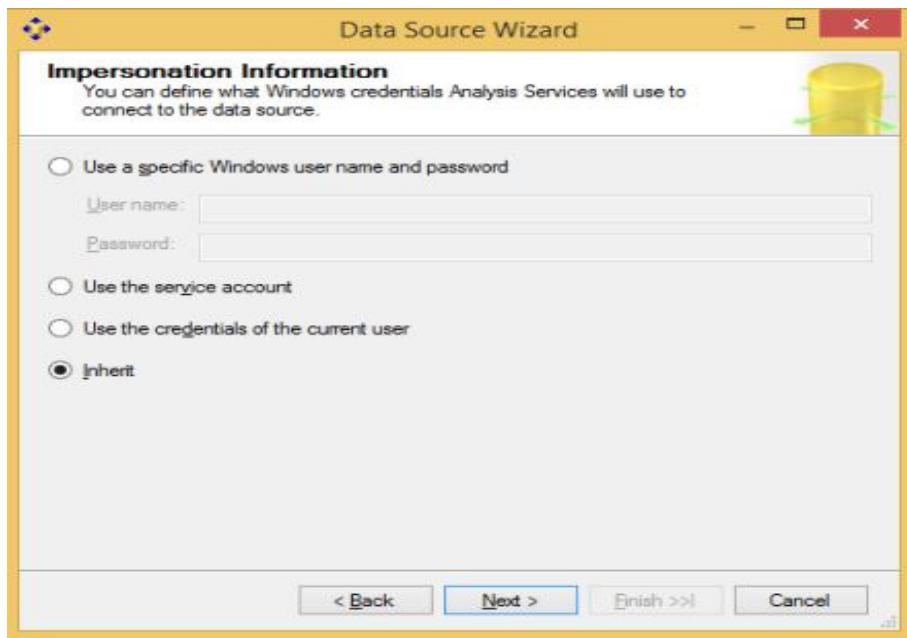
Note: Password for sa: admin123 (as given during installation of SQL 2012 full version)



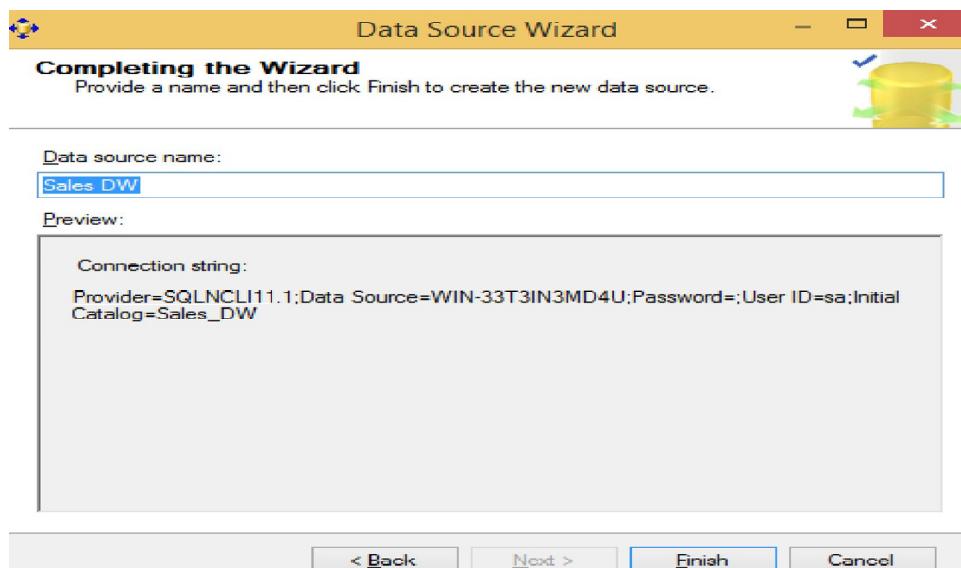
Click Next



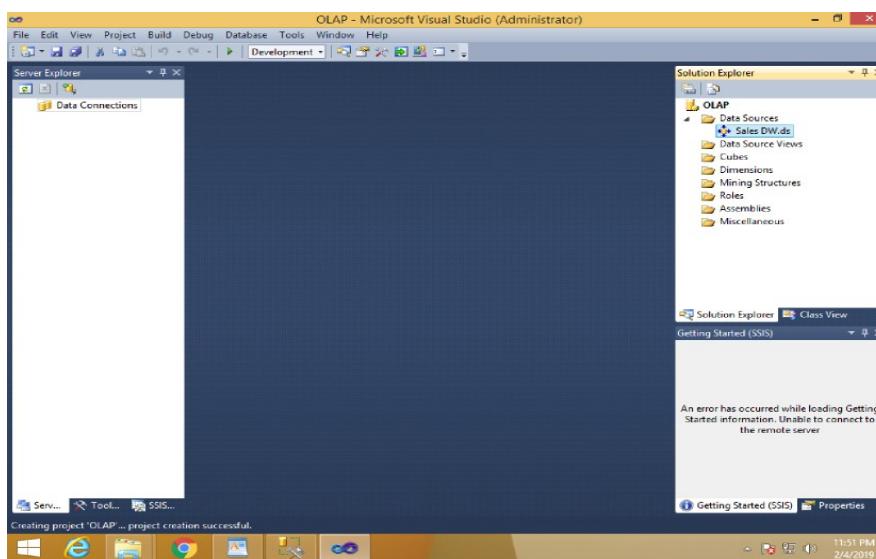
Select Inherit → Next



Click Finish

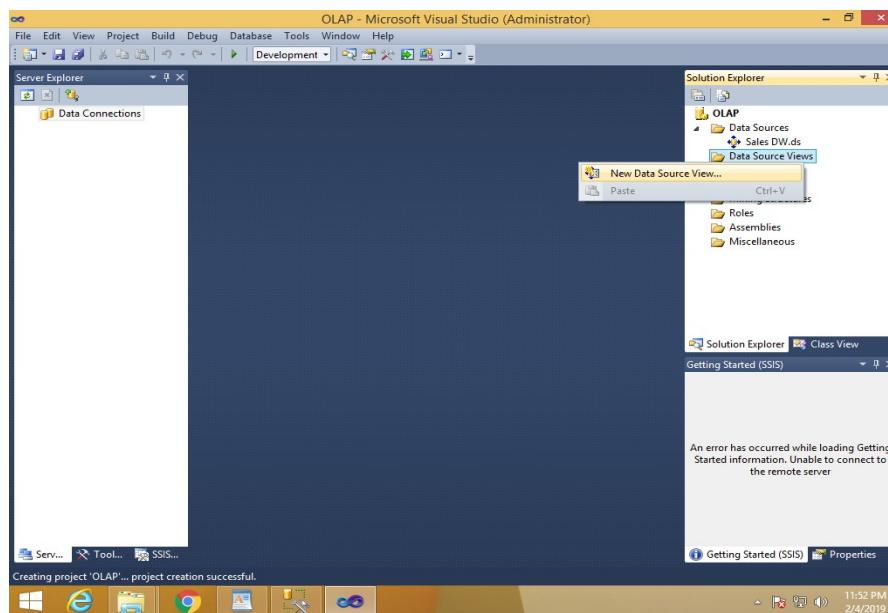


Sales_DW.ds gets created under Data Sources in Solution Explorer



Step 3: Creating New Data Source View

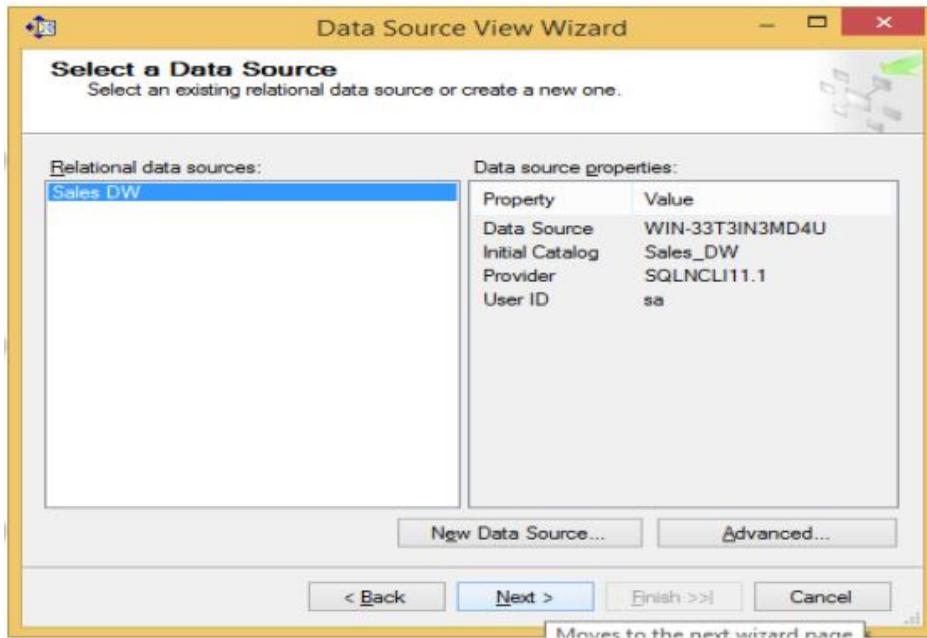
In Solution explorer right click on Data Source View → Select New Data Source View



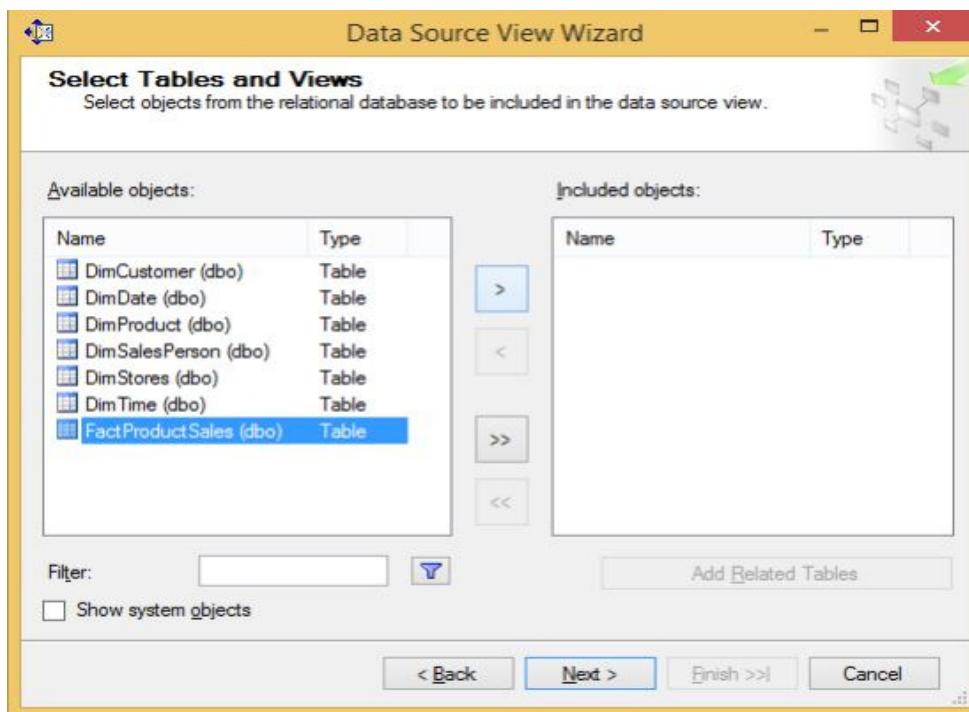
Click Next

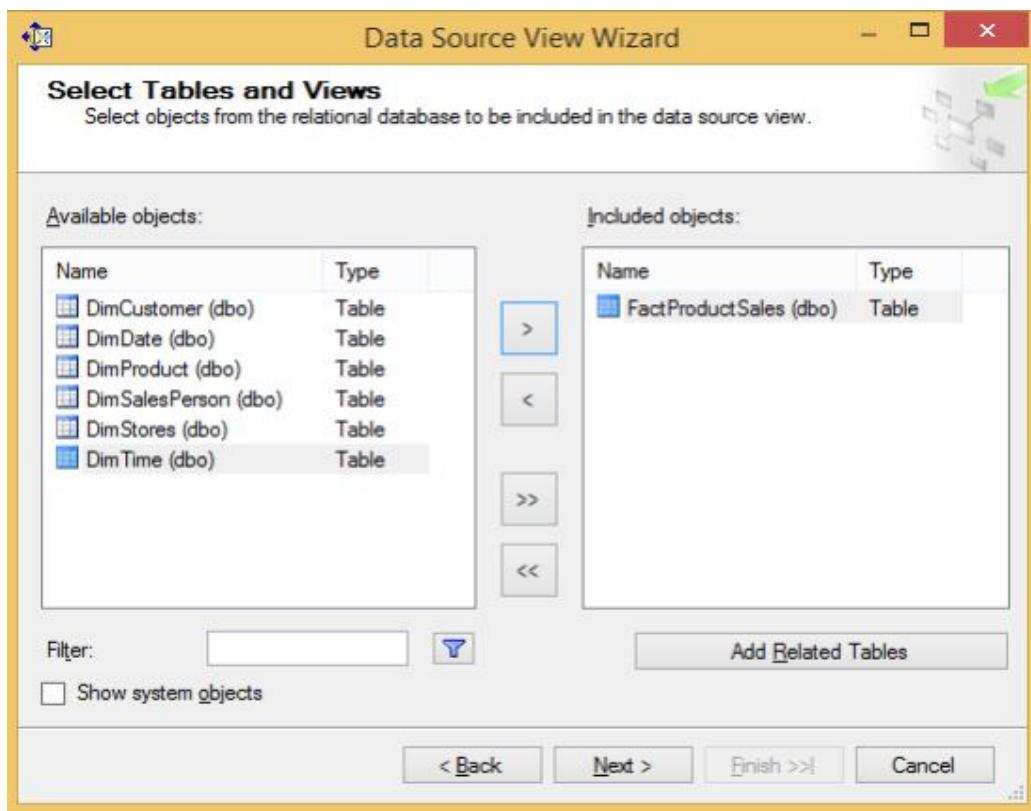


Click Next

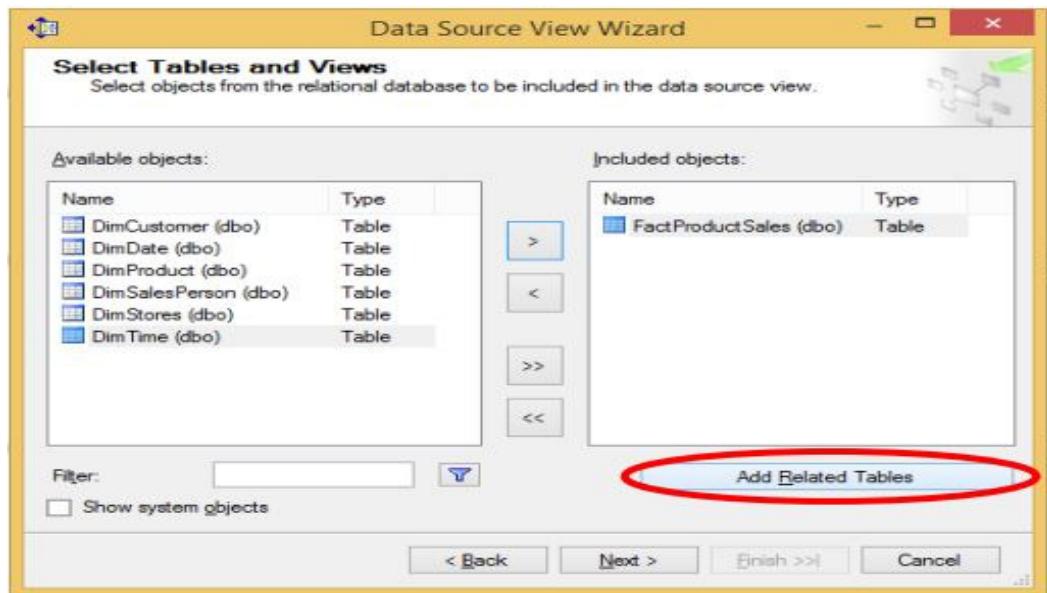


Select Fact Product Sales (dbo) from Available objects and put in Includes Objects by clicking on

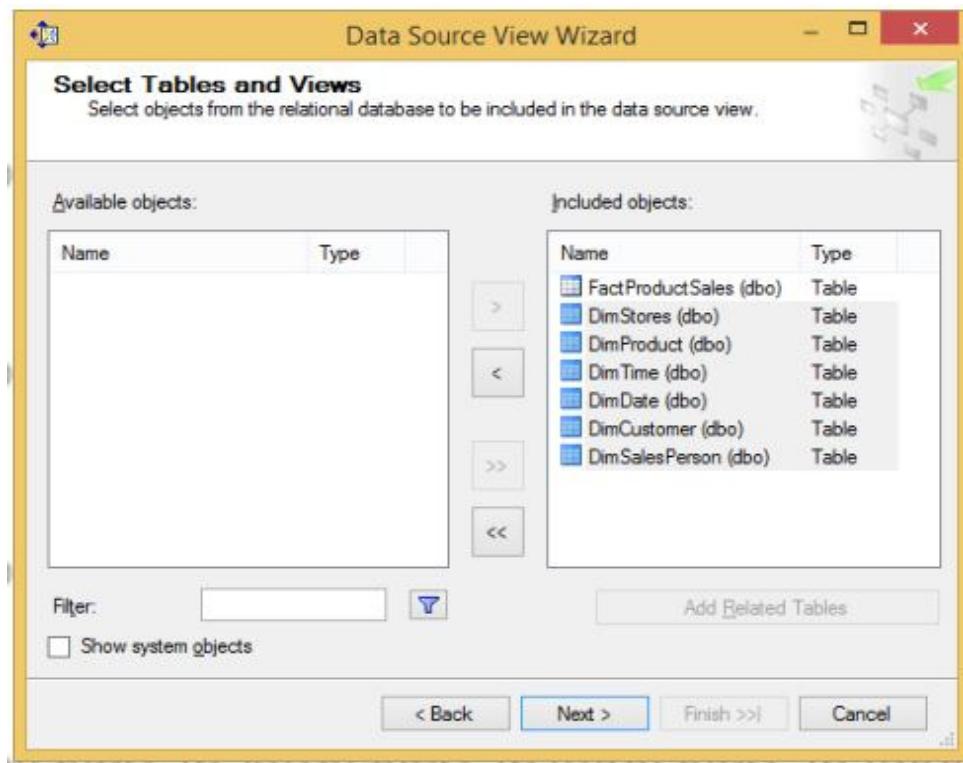




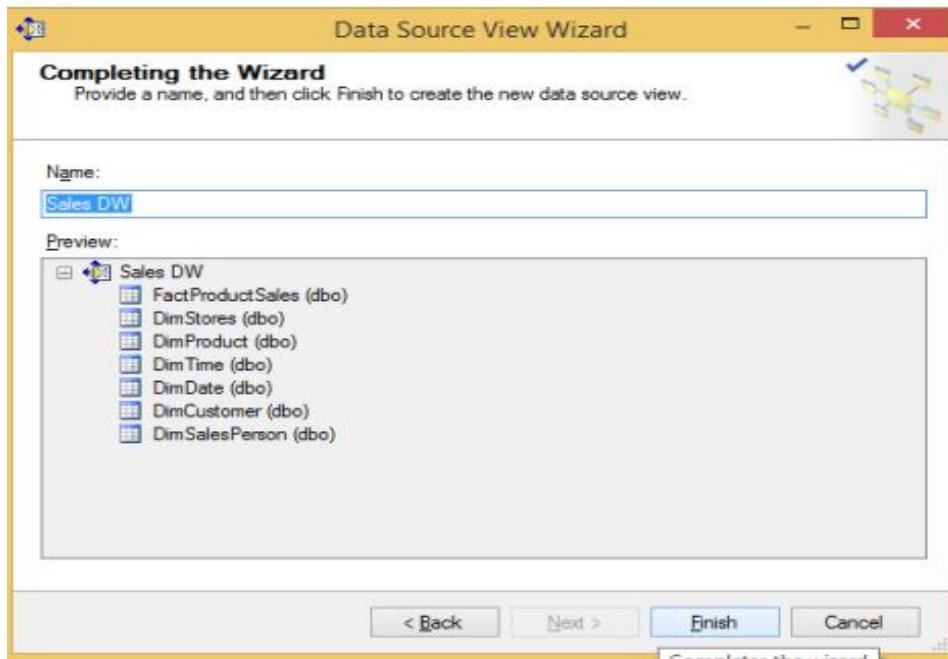
Click on Add Related Tables



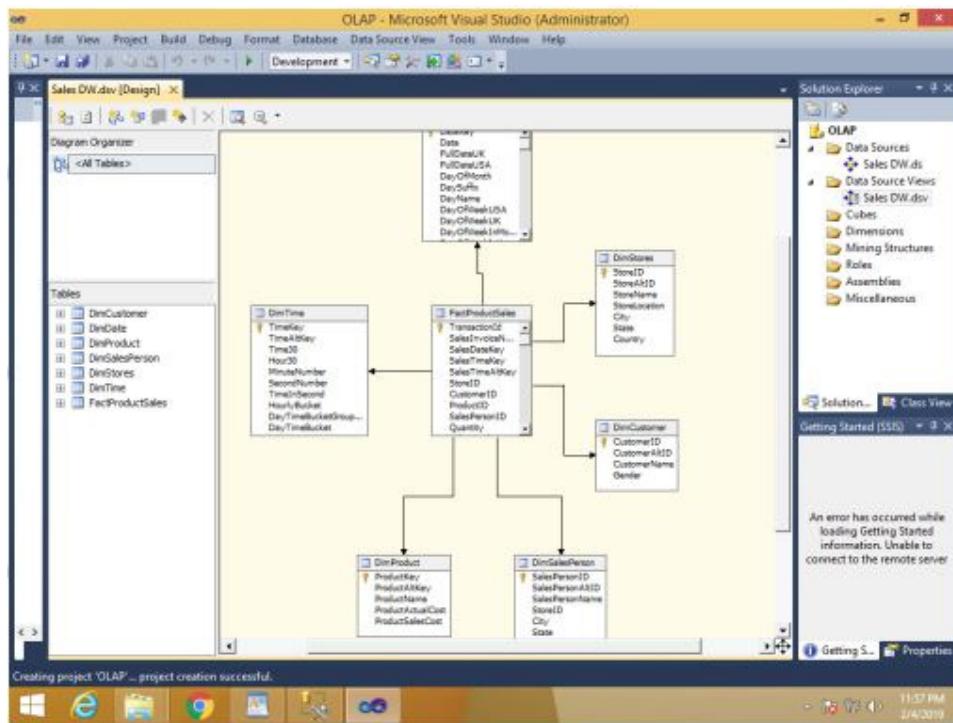
Click Next



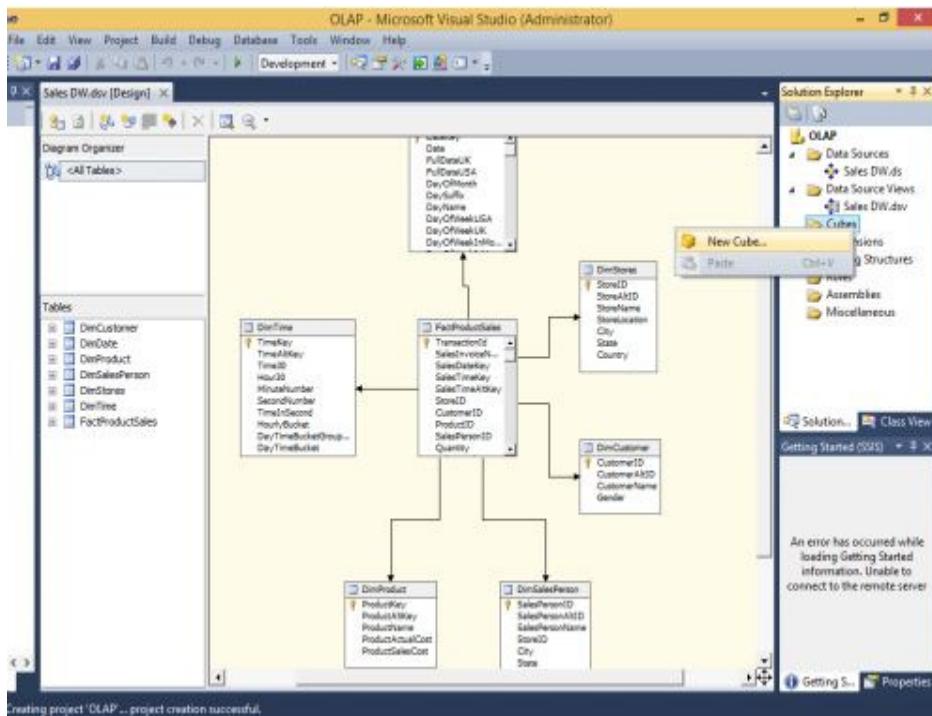
Click Finish



Sales DW.dsv appears in Data Source Views in Solution Explorer.

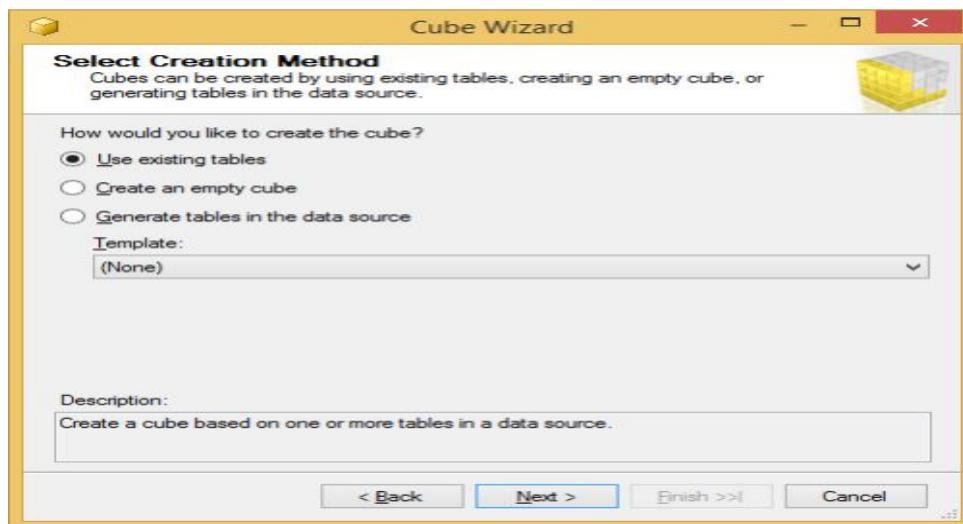


Step 4: Creating new cube Right click on Cubes → New Cube

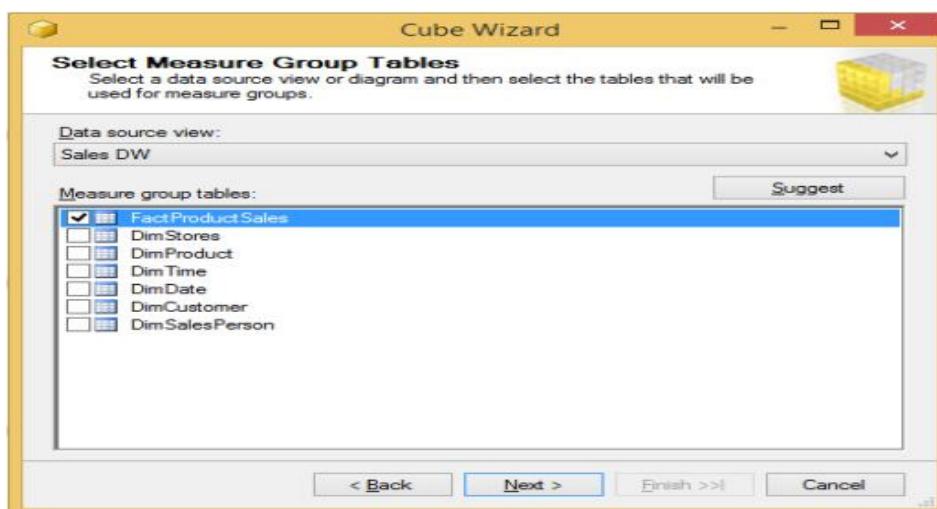




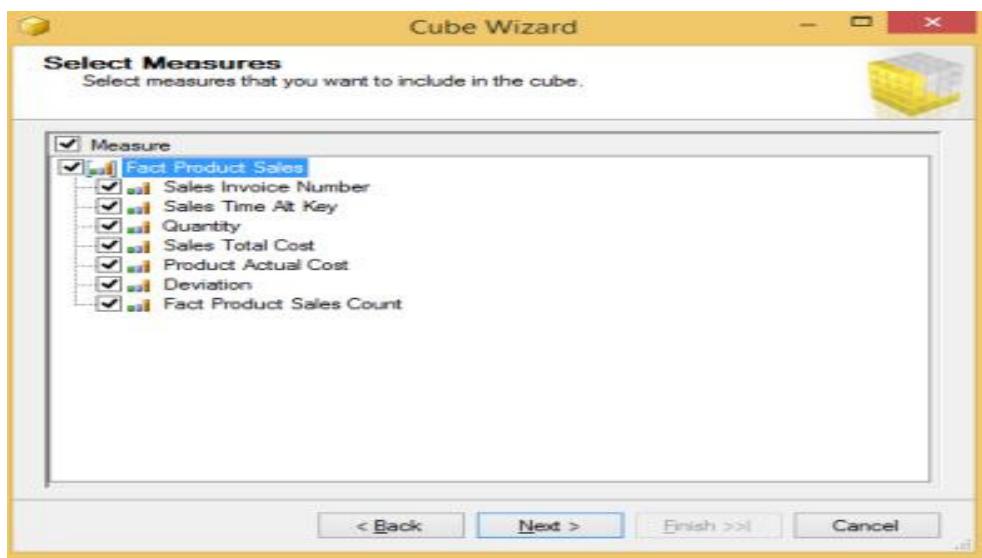
Select Use existing tables in Select Creation Method → Next



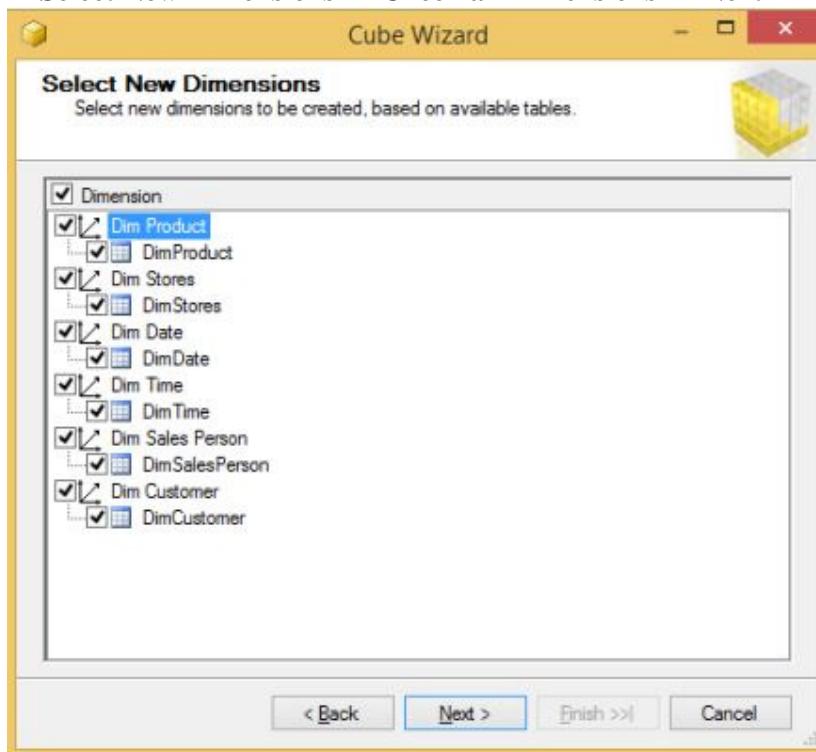
In Select Measure Group Tables → Select FactProductSales → Click Next



In Select Measures → check all measures → Next



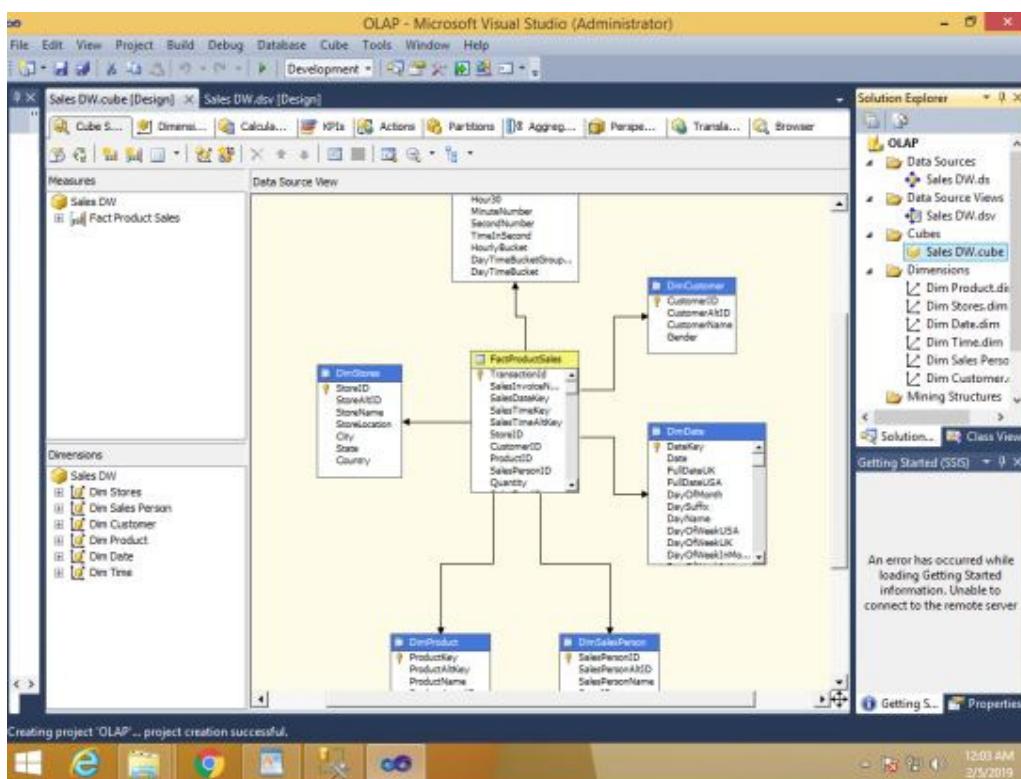
In Select New Dimensions → Check all Dimensions → Next



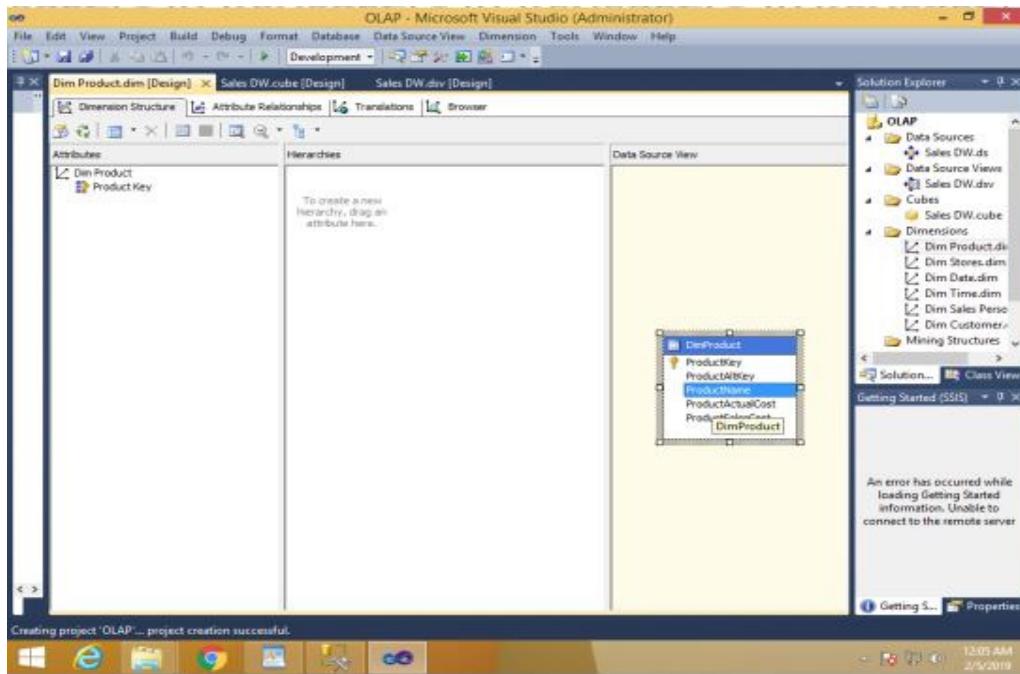
Click on Finish



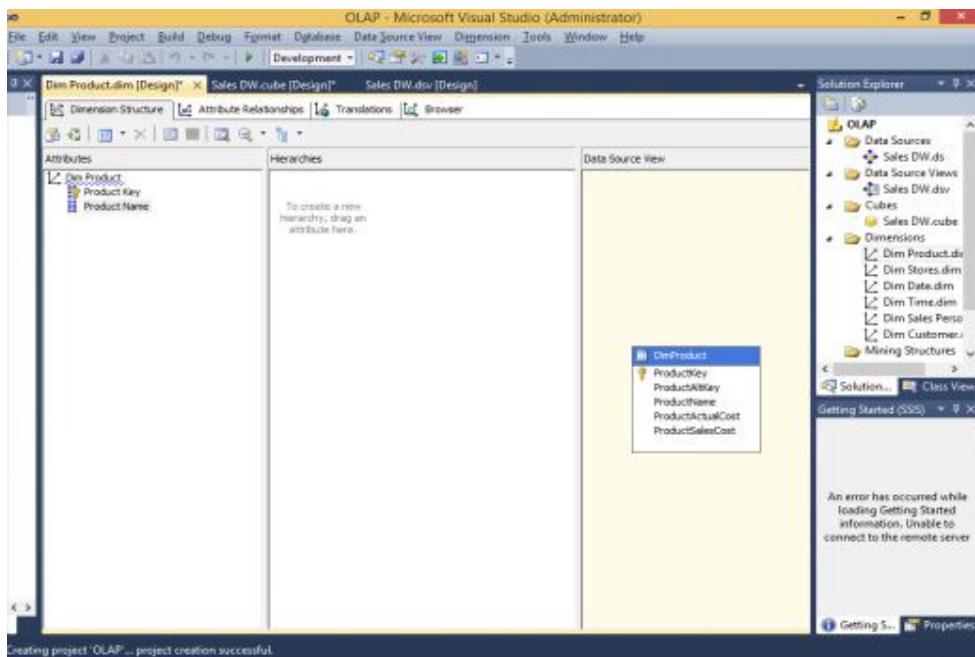
Sales_DW. Cube is created



Step 5: Dimension Modification In dimension tab → Double Click Dim Product.dim



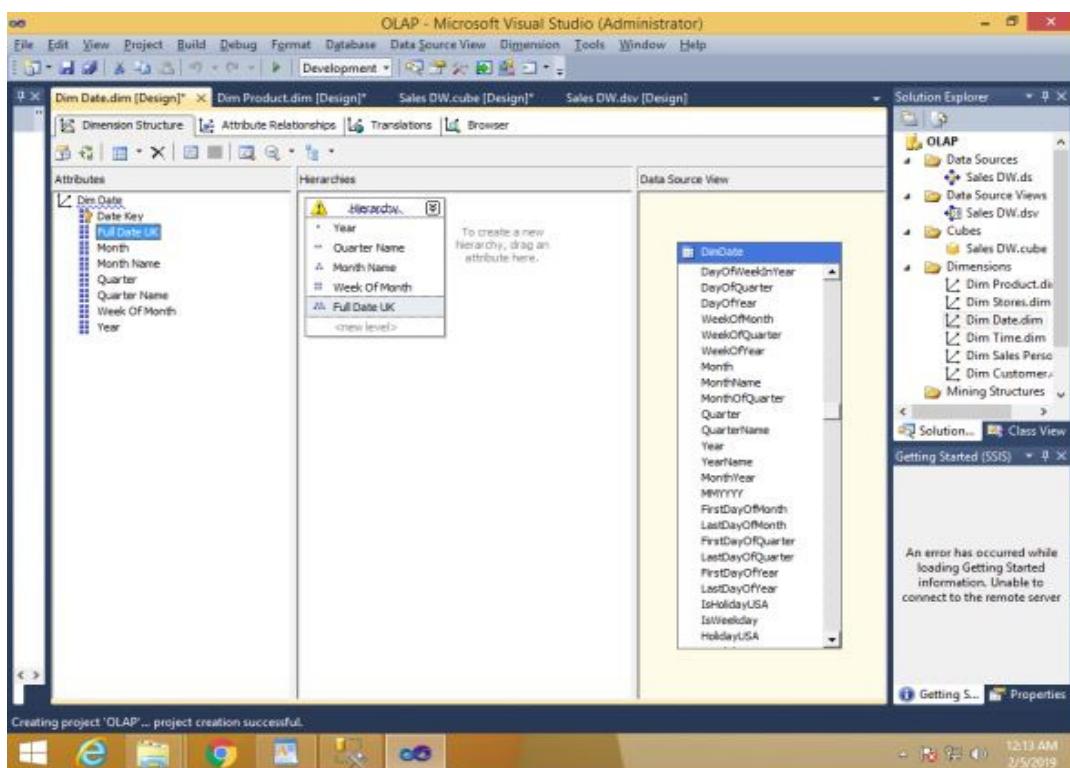
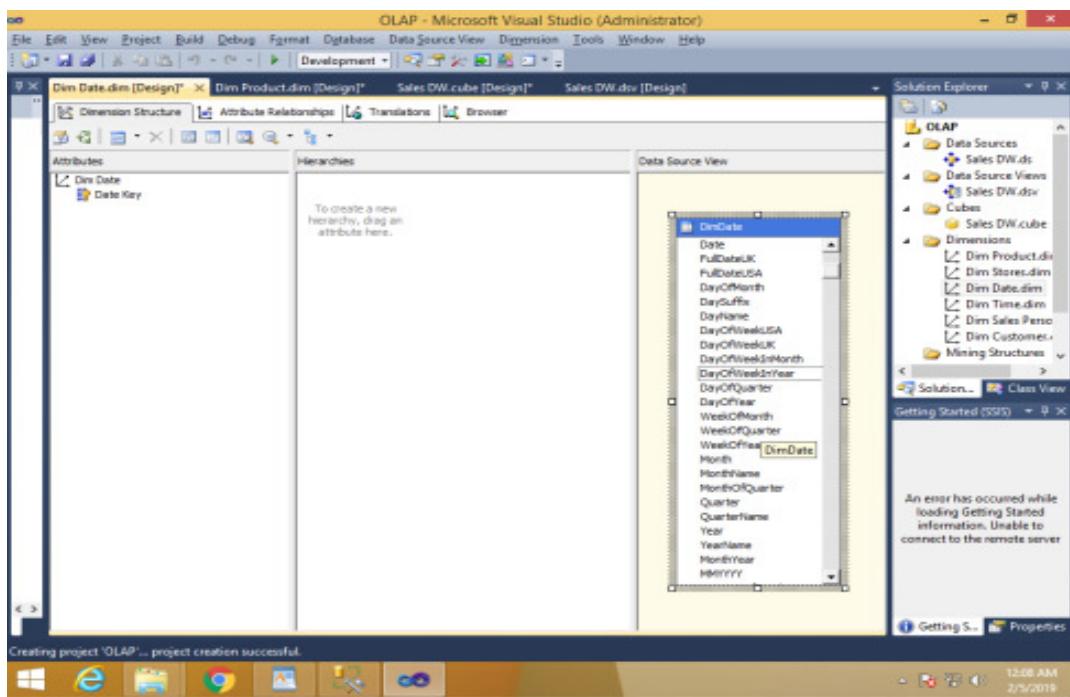
Drag and Drop Product Name from Table in Data Source View and Add in Attribute Pane at left side



Step 6: Creating Attribute Hierarchy in Date Dimension

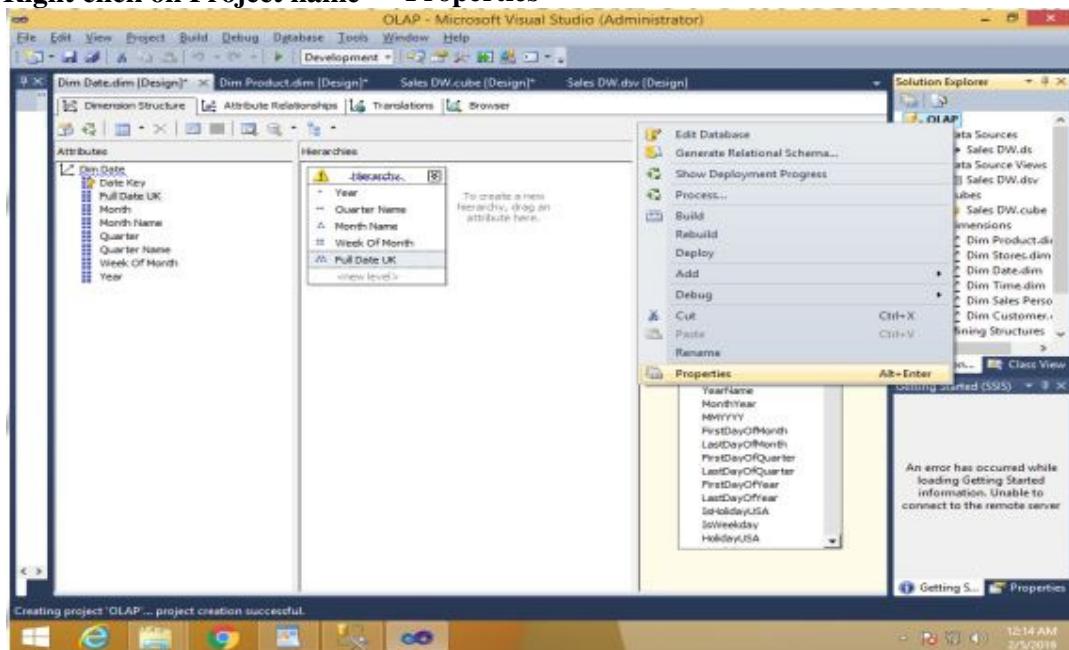
Double click On Dim Date dimension -> Drag and Drop Fields from Table shown in Data Source View to Attributes-> Drag and Drop attributes from leftmost pane of attributes to middle pane of Hierarchy.

Drag fields in sequence from Attributes to Hierarchy window (Year, Quarter Name, Month Name, Week of the Month, Full Date UK)

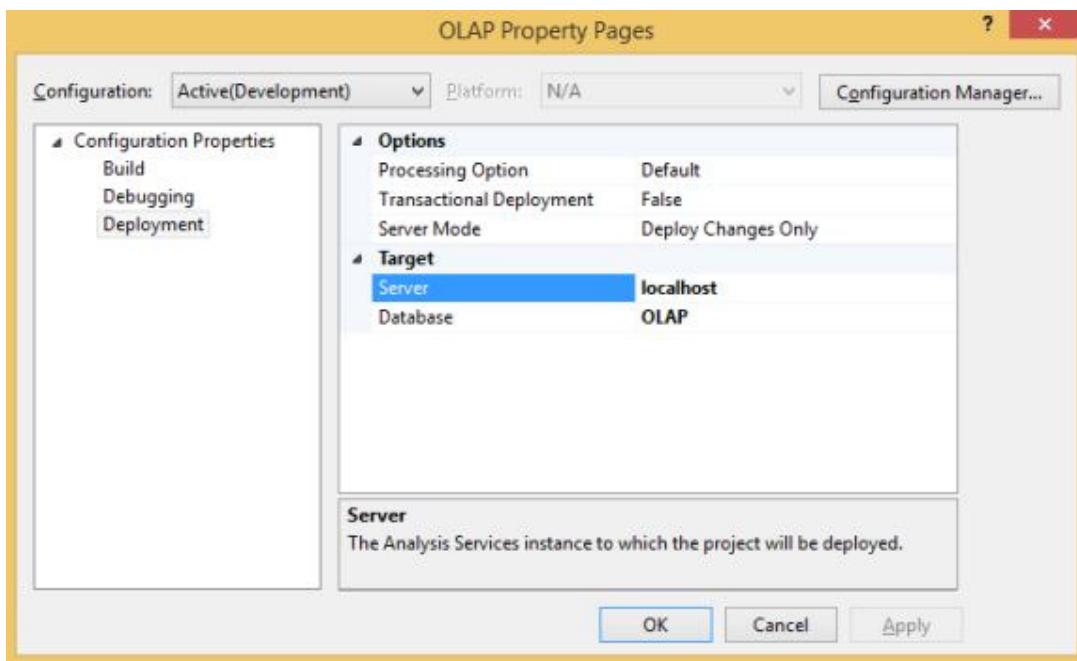


Step 7: Deploy Cube

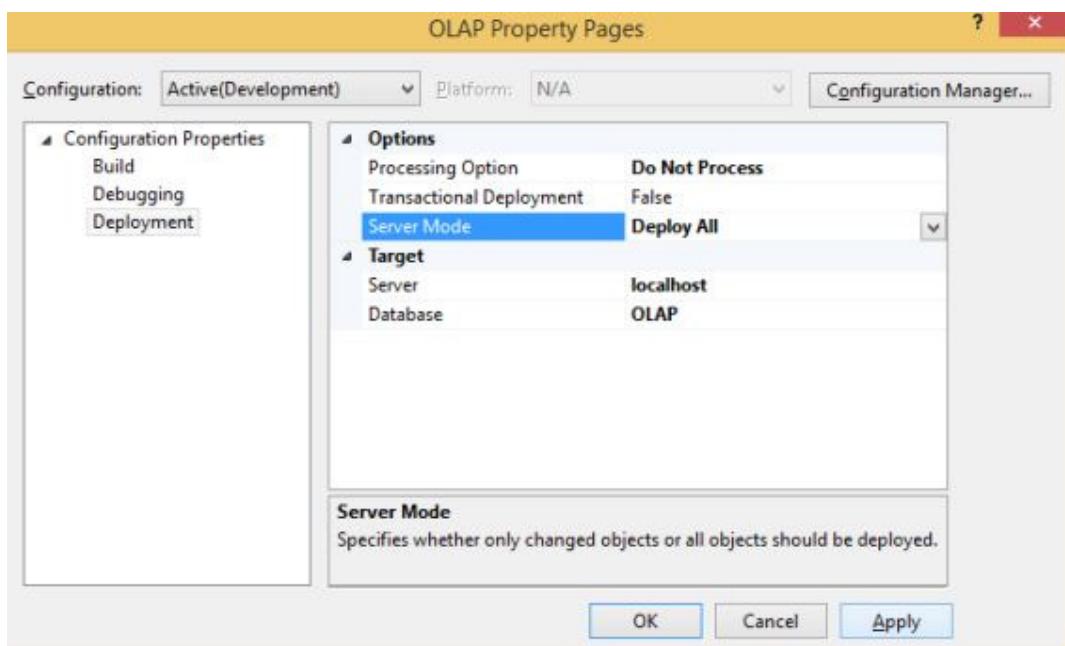
Right click on Project name → Properties



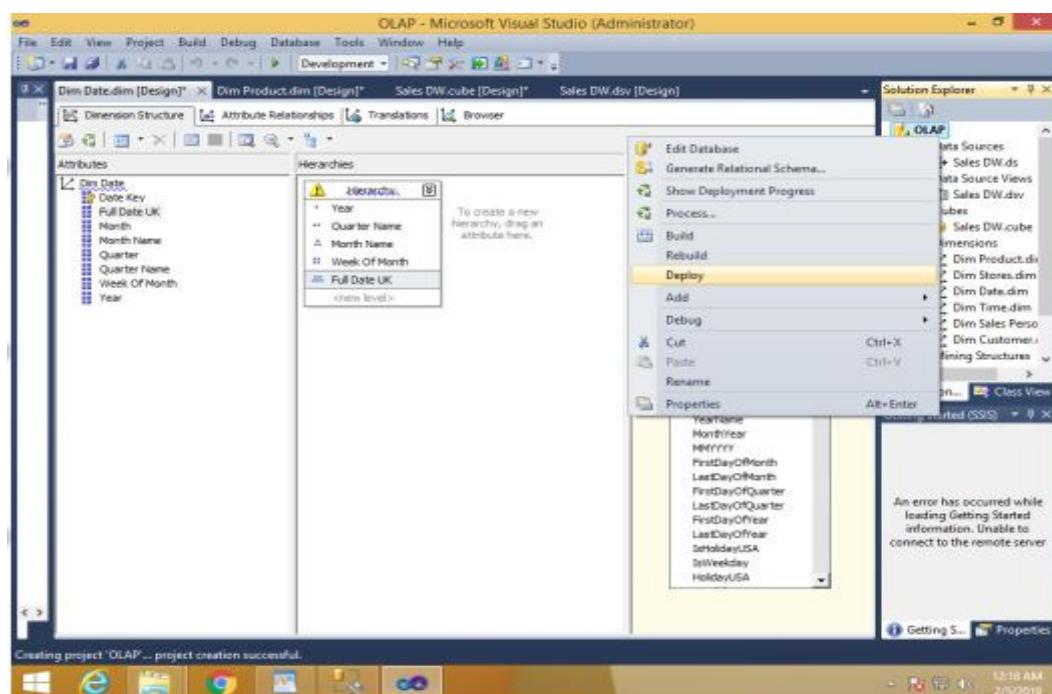
This window appears



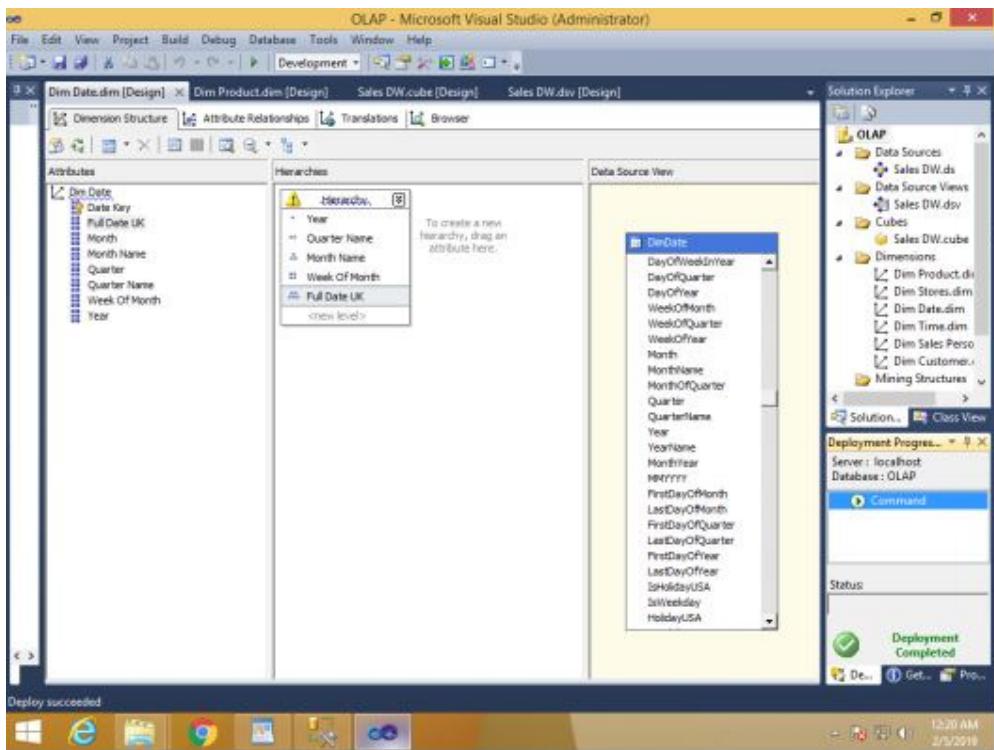
Do following changes and click on Apply & ok



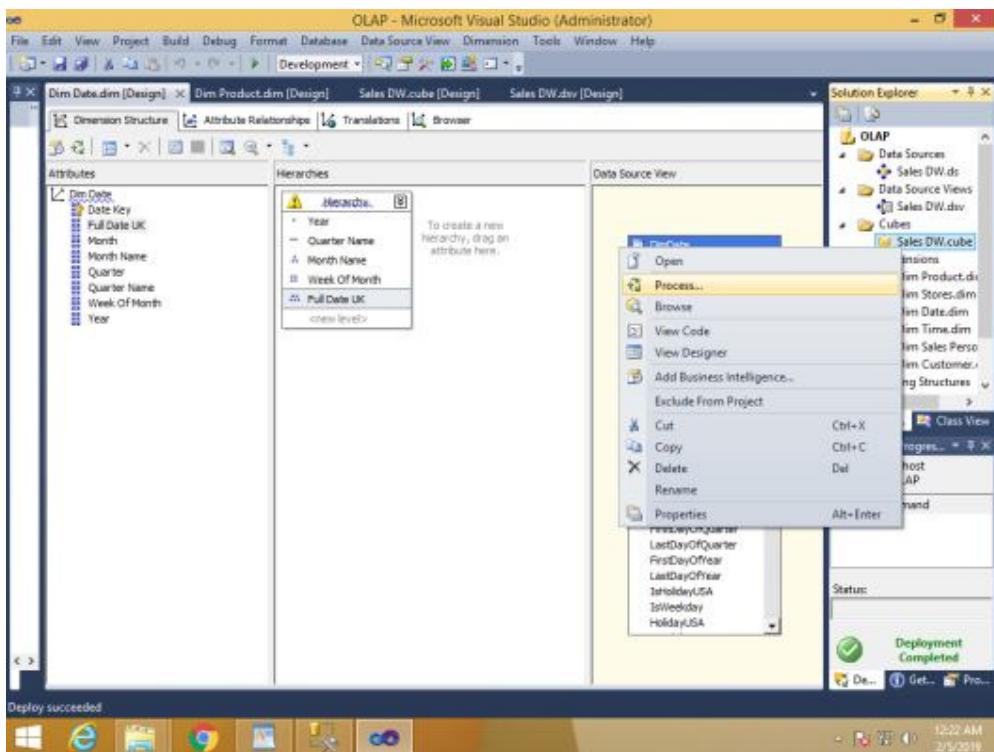
Right click on project name → Deploy



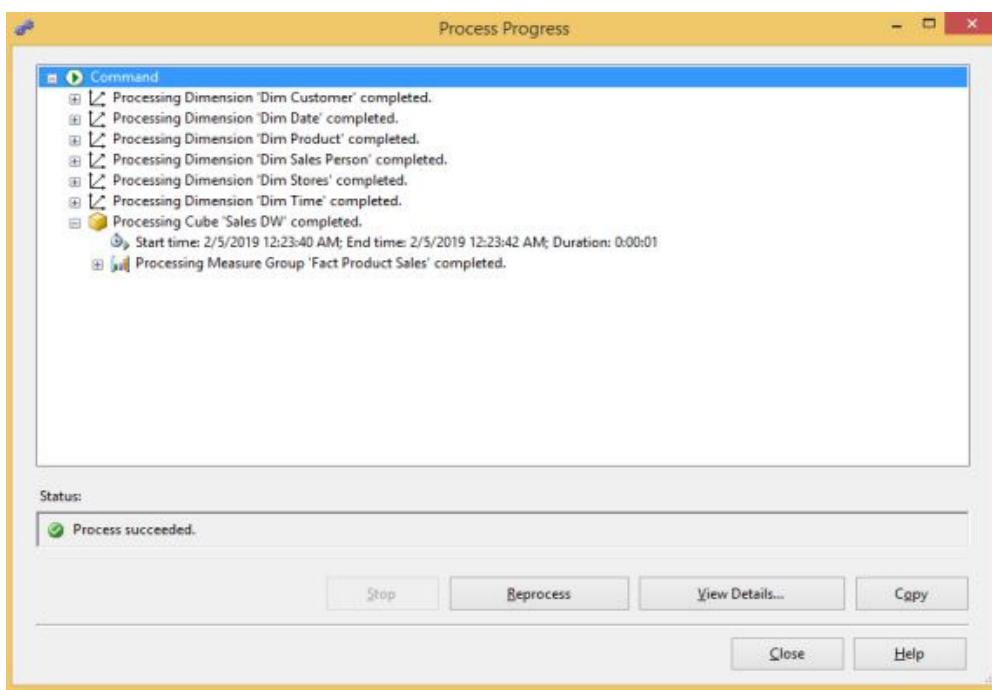
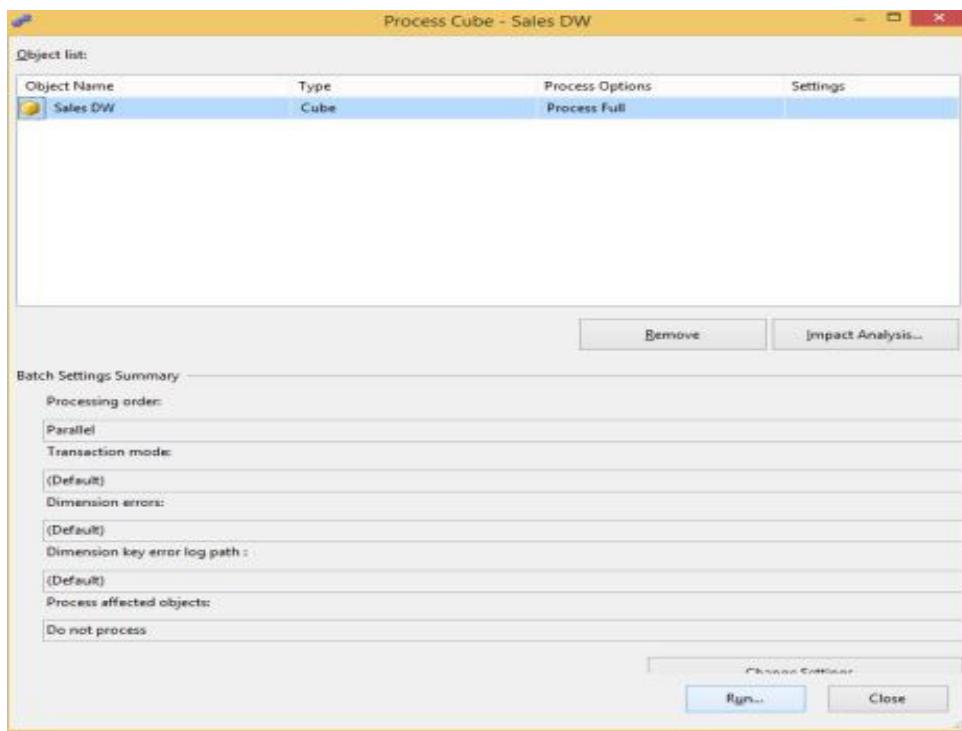
Deployment successful



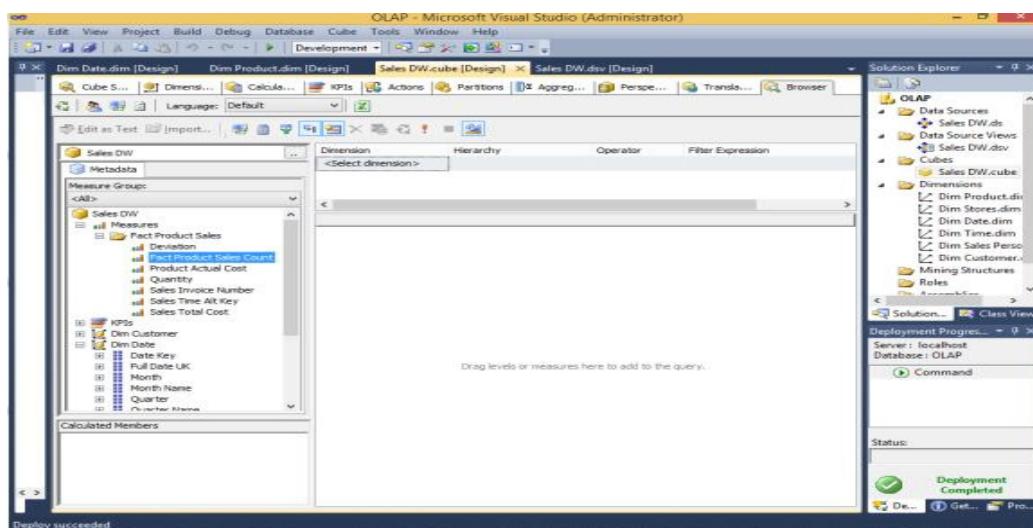
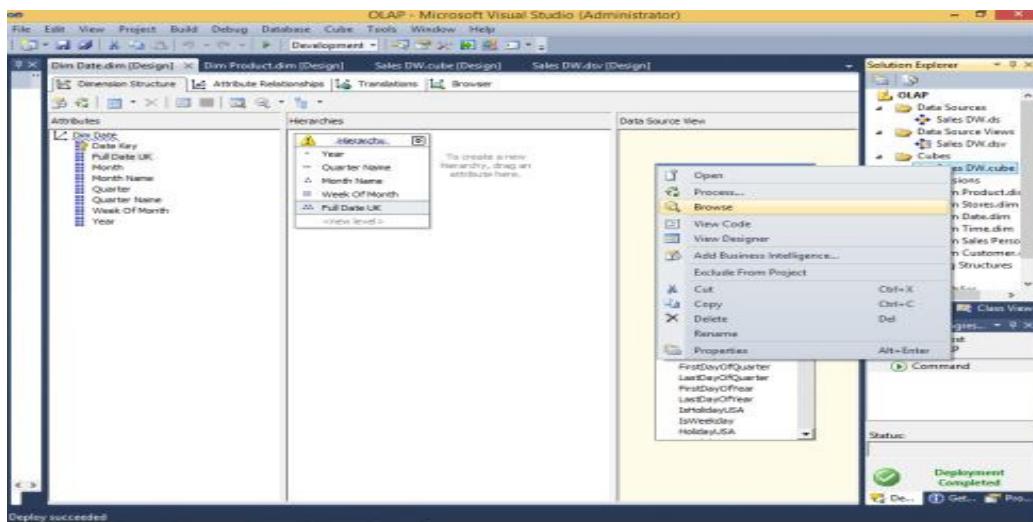
To process cube right click on Sales_DW(cube) → Process



Click run



Browse the cube for analysis in solution explorer



Conclusion:

Hence we have created cube with suitable dimension and fact tables based on OLAP

Assignment Questions

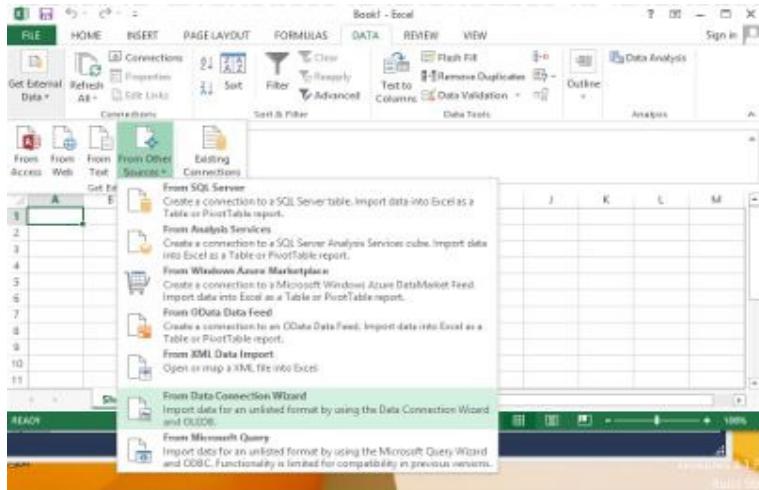
1. What do you understand by fact table?
2. Explain the dimensional model
3. What is dimension?
4. Discuss the components of OLAP.
5. Write down the concept of cube

Assignment No. -4

Aim: Import the data warehouse data in Microsoft Excel and create the Pivot table and Pivot Chart.

(Ms Office Professional is used to make sure Power View is enabled for visualization.)

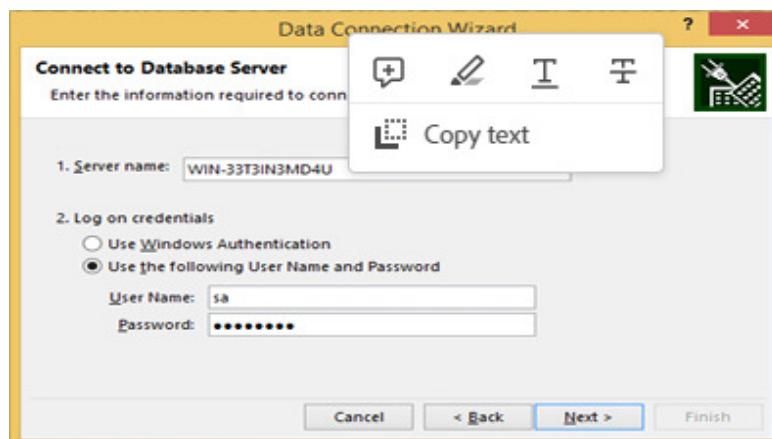
Step 1: Open Excel 2013 (Professional) Goto Data tab → GetExternalData → From Other Sources → Data Connection Wizard



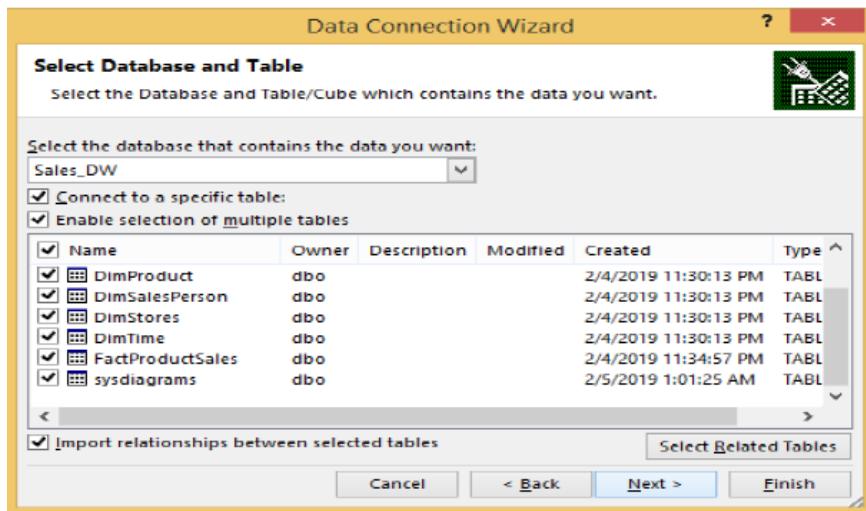
Step 2: In Data Connection Wizard → Select Microsoft SQL Server → Click on Next



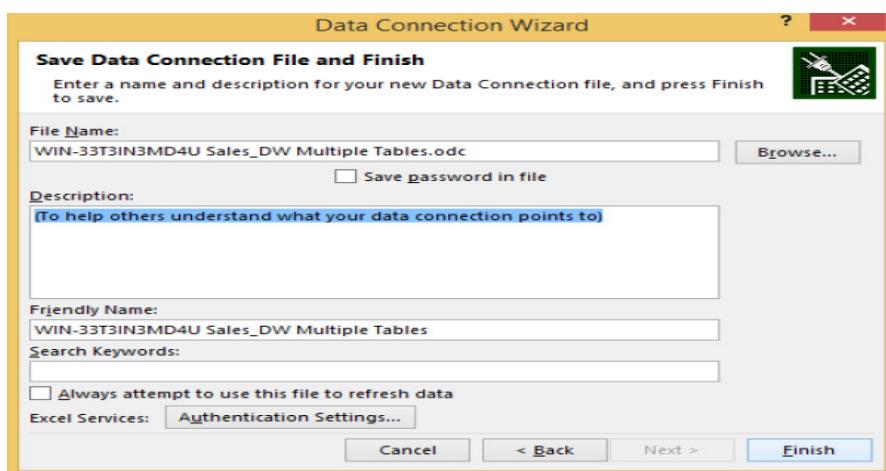
Step 3: In connect to Database Server provide Server name(Microsoft SQL Server Name)
Provide password for sa account as given during installation of SQL Server 2012 full version
Password: admin123 Click on Next



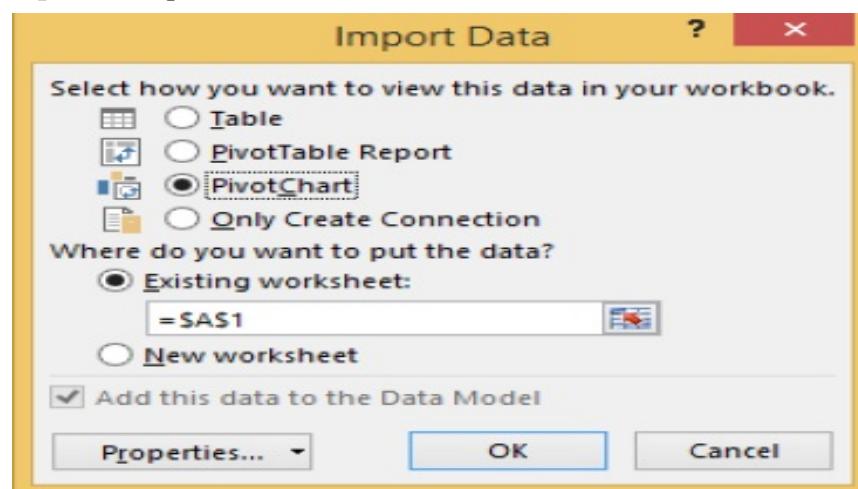
Step 4: In Select Database and Table → Select Sales_DW (already created in SQL) → check all dimensions and import relationships between selected tables



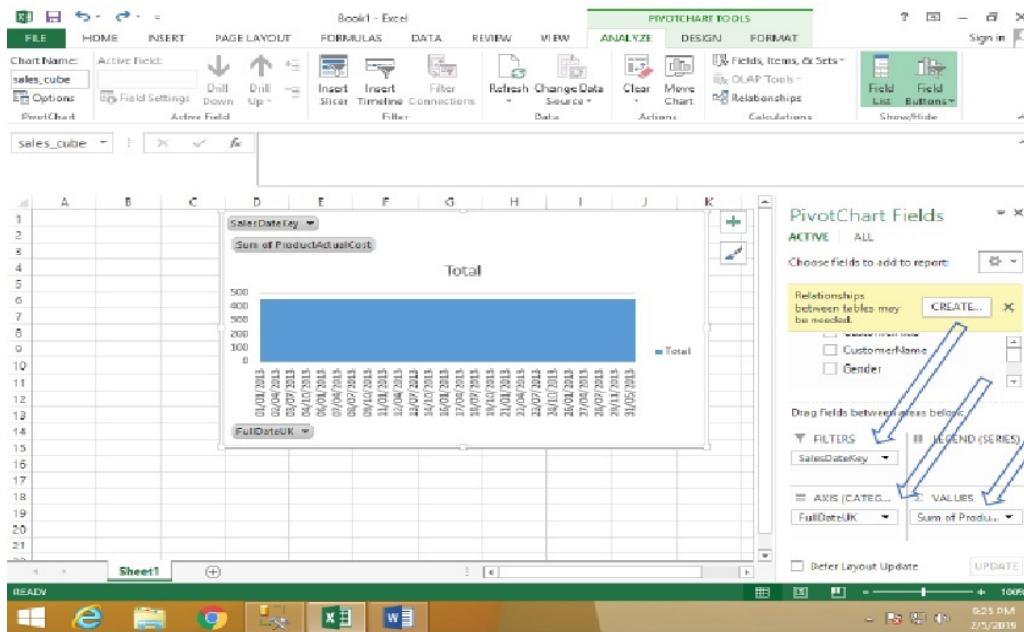
Step 5: In save data connection files browse path and click on Finish



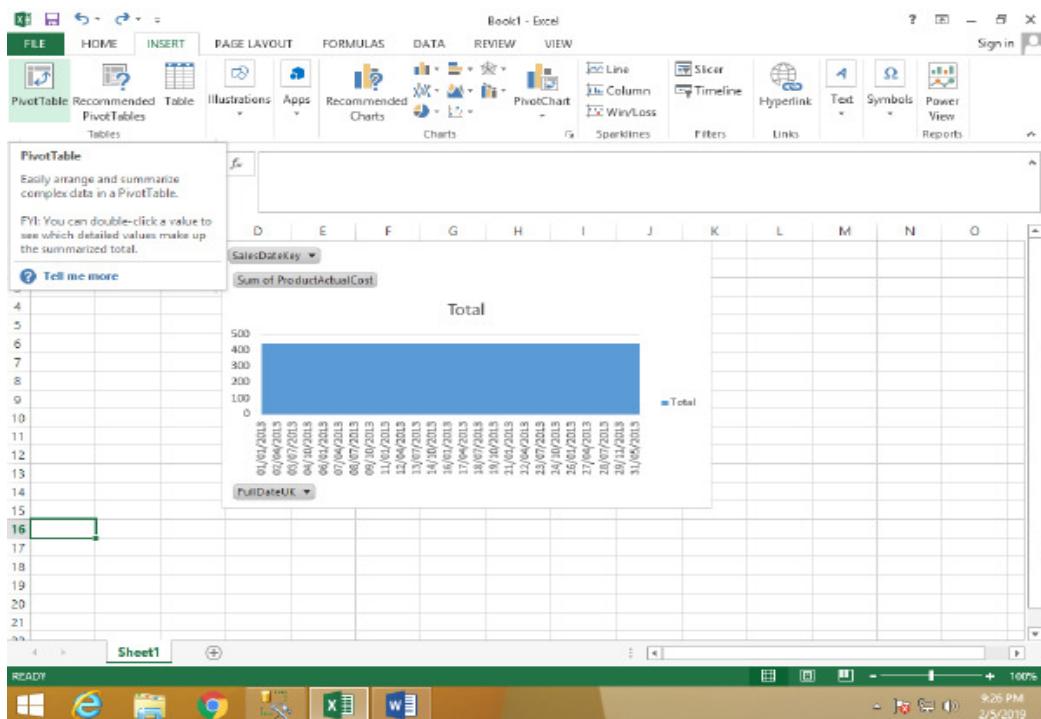
Step 6: In import data select Pivot Chart and click on OK



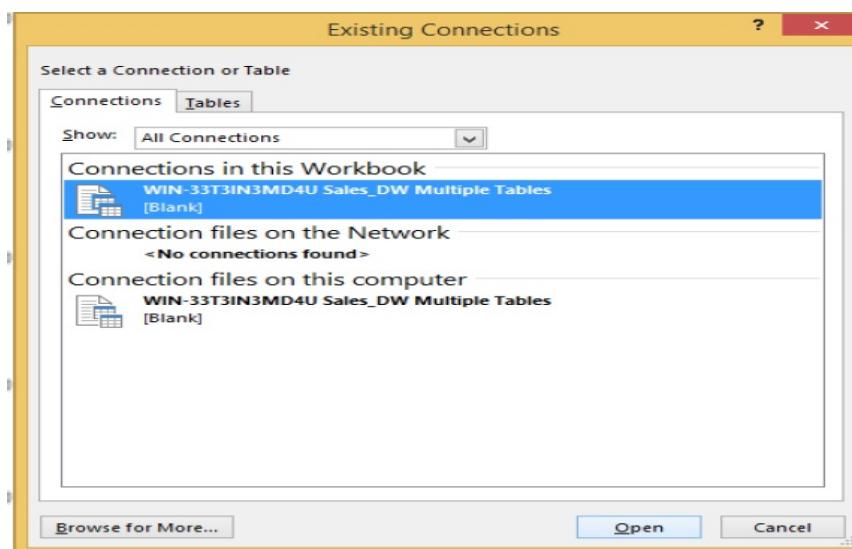
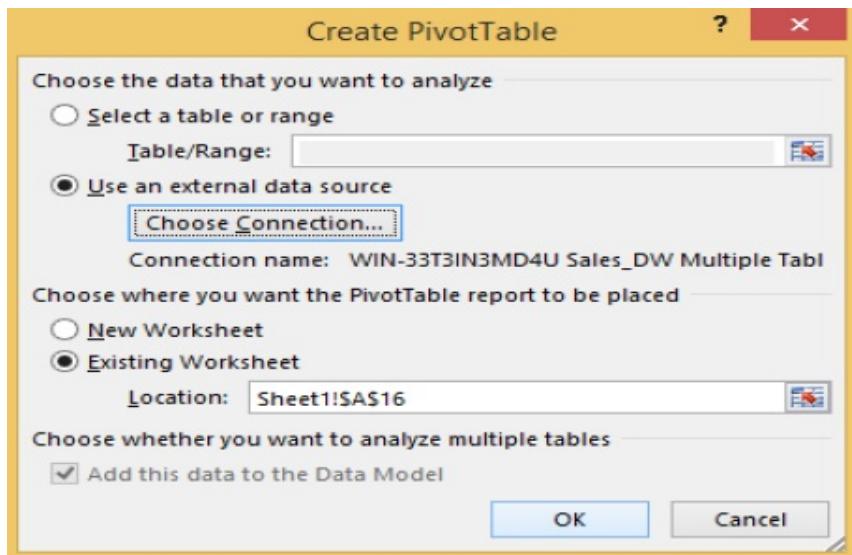
Step 7: In fields put SalesDateKey in filters, FullDateUK in axis and Sum of ProductActualCost in values.



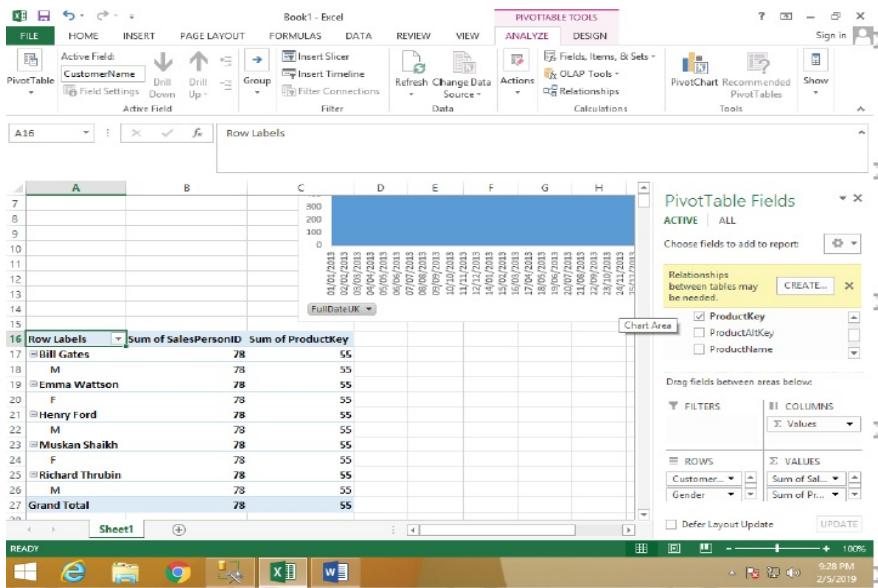
Step 8: In Insert Tab → go to Pivot Table



Step 9: Click on Choose Connection to select existing connection with Sales_DW and click on open



Pivot table and Pivot chart is created



Conclusion:

Hence we have created the Pivot table and Pivot Chart by importing the datawarehouse data in Microsoft Excel

Assignment Questions

1. What is Pivot table and Pivot chart?
2. Show Maximum Selling Product Name and its Sales month wise
3. How to create a basic Pivot Table write down steps .

Assignment No.-5

Aim: Perform the data classification using classification algorithm. Or Perform the data clustering using clustering algorithm.

Software required: R 3.5.1

Time series is a series of data points in which each data point is associated with a timestamp. A simple example is the price of a stock in the stock market at different points of time on a given day. Another example is the amount of rainfall in a region at different months of the year. R language uses many functions to create, manipulate and plot the time series data. The data for the time series is stored in an R object called time-series object. It is also a R data object like a vector or data frame.

The time series object is created by using the ts() function.

Syntax

The basic syntax for ts() function in time series analysis is –

```
timeseries.object.name <- ts(data, start, end, frequency)
```

Following is the description of the parameters used –

- data is a vector or matrix containing the values used in the time series.
- start specifies the start time for the first observation in time series.
- end specifies the end time for the last observation in time series.
- frequency specifies the number of observations per unit time.

Except the parameter "data" all other parameters are optional

Consider the annual rainfall details at a place starting from January 2012.

We create an R time series object for a period of 12 months and plot it.

Code to run in R

```
# Get the data points in form of a R vector.
rainfall <-
c(799,1174.8,865.1,1334.6,635.4,918.5,685.5,998.6,784.2,985,882.8,1071)

# Convert it to a time series object.
rainfall.timeseries <- ts(rainfall,start = c(2012,1),frequency = 12)

# Print the timeseries data.
print(rainfall.timeseries)

# Give the chart file a name.
png(file = "rainfall.png")

# Plot a graph of the time series.
plot(rainfall.timeseries)

# Save the file.
dev.off()
```

After this again plot to get chart

```
plot(rainfall.timeseries)
```

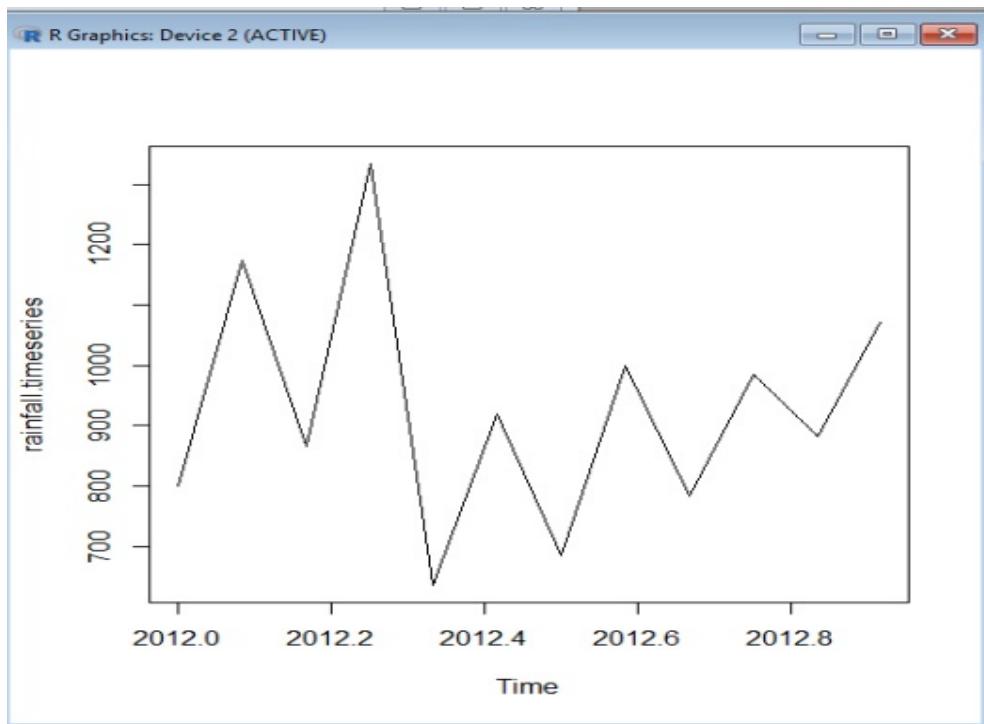
Output:

When we execute the above code, it produces the following result and chart

```
Jan Feb Mar Apr May Jun Jul Aug Sep 2012 799.0 1174.8
865.1 1334.6 635.4 918.5 685.5 998.6 784.2 Oct Nov Dec 2012
985.0 882.8 1071.0
```

```
R GUI (M-UI)
File Edit View Mag Packages Windows Help
Type 'q()' to quit R.

> # Get the data points in form of a R vector.
> rainfall <- c(799,1174.8,865.1,1334.6,635.4,918.5,685.5,998.6,784.2,985,882.8,1071)
> # Convert it to a time series object.
> rainfall.timeseries <- ts(rainfall,start = c(2012,1),frequency = 12)
> # Print the timeseries data.
> print(rainfall.timeseries)
  Jan   Feb   Mar   Apr   May   Jun   Jul   Aug   Sep   Oct
2012 799.0 1174.8 865.1 1334.6 635.4 918.5 685.5 998.6 784.2 985.0
  Nov   Dec
2012 882.8 1071.0
> # Give the chart file a name.
> png(file = "rainfall.png")
> # Plot a graph of the time series.
> plot(rainfall.timeseries)
> # Save the file.
> dev.off()
null device
1
> plot(rainfall.timeseries)
>
```



Conclusion:

Hence we have performed the data classification using classification algorithm

Assignment Question

1. What is data classification?
2. Explain Classification Algorithm?
3. What is data clustering?
4. Explain Clustering Algorithm?