

# PRACTICAL: 1

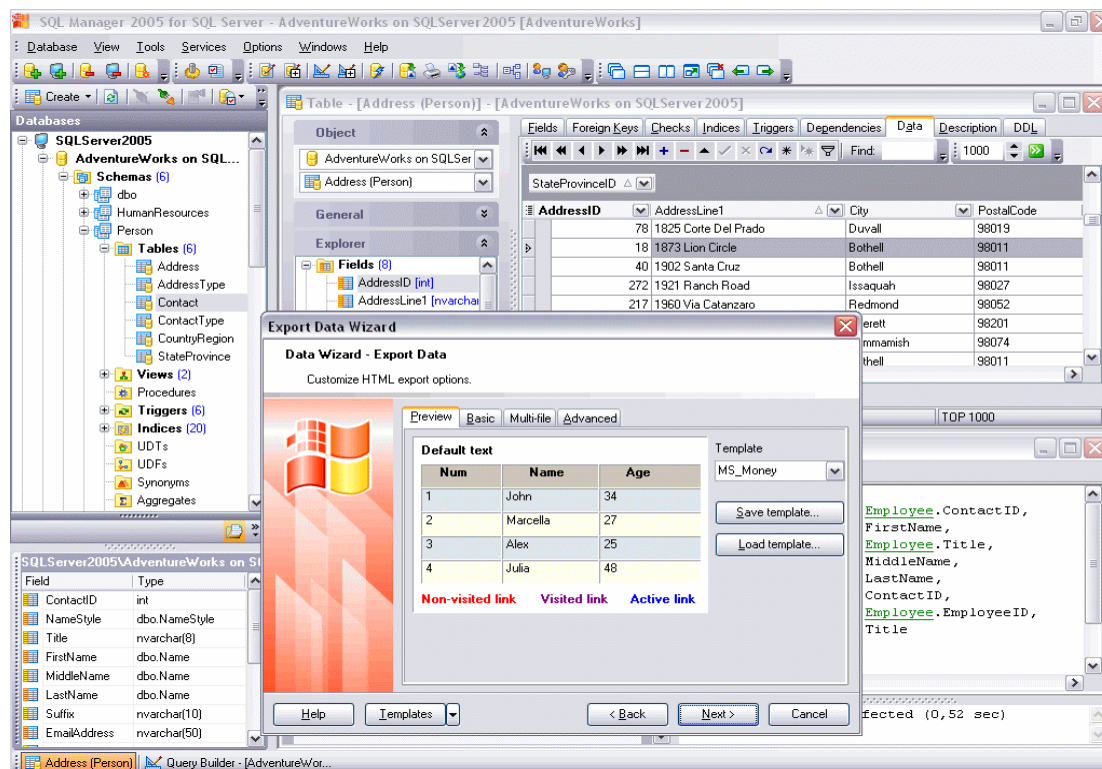
## INTRODUCTION TO SQL SERVER 2005, ITS ANALYSIS SERVICES AND BUSINESS INTELLIGENCE DEVELOPMENT STUDIO.

Microsoft SQL Server is a relational database server product by Microsoft. Its primary query languages are T-SQL and ANSI SQL. Microsoft SQL Server is developed in C, C++ and C # languages. In this server the data is managed in fashion like any Relational Database Management System.

### ➤ SERVICES OF SQL SERVER

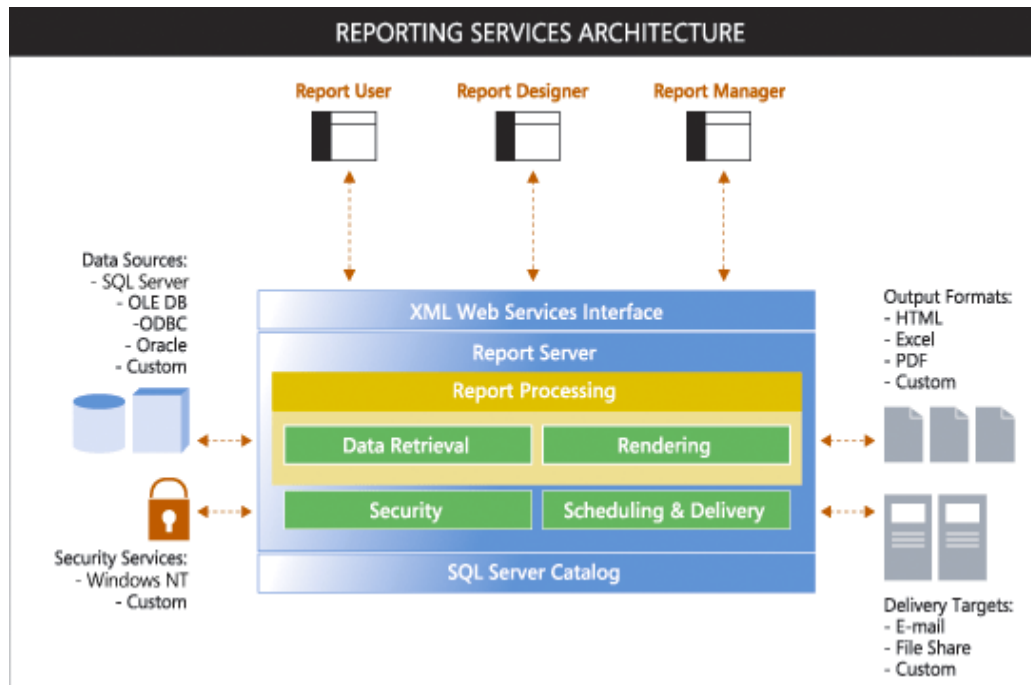
- 1) SQL Server Reporting Services (SSRS)
- 2) SQL Server Integration Service (SSIS)
- 3) SQL Server Analysis Services
- 4) SQL Server VSS Writer
- 5) SQL Server Full Text Search

### SCREEN SHOT OF SQL SERVER



## 1) SQL SERVER REPORTING SERVICES (SSRS):

It can be used to prepare and deliver a variety of interactive and printed reports.



## 2) SQL SERVER INTEGRATION SERVICE (SSIS)

Microsoft Integration Services is a platform for building enterprise-level data integration and data integration and data transformations solutions.

Integration Services includes a rich set of built-in tasks and transformations, tools for constructing packages and the Integration Services service for running and managing packages. You can use the graphical Integration Services to create solutions without writing a single line of code; or you can program the extensive Integration Services object model to create packages programmatically and code custom tasks and other package objects.

## 3) SQL SERVER ANALYSIS SERVICES

Analysis Services includes a group of OLAP and data mining capabilities. Online Analytical Processing, a category of software tools that provides analysis of data stored in a database. OLAP tools enable users to analyze different dimensions of multidimensional data. it provides time series and trend analysis views. OLAP often is used in data mining.

The chief component of OLAP is the OLAP server, which sits between a client and database management systems (DBMS). The OLAP server understands

how data is organized in the database and has special functions for analyzing the data. There are OLAP servers available for nearly all the major database systems.

The basic concepts of OLAP include:

- ✓ Cube
- ✓ Dimension table
- ✓ Dimension
- ✓ Level
- ✓ Fact table
- ✓ Measure
- ✓ Schema

## CUBE

The basic unit of storage and analysis in Analysis Services is the *cube*. A cube is a collection of data that's been aggregated to allow queries to return data quickly. For example, a cube of order data might be aggregated by time period and by title, making the cube fast when you ask questions concerning orders by week or orders by title. Cubes are ordered into *dimensions* and *measures*. Dimensions come from *dimension tables*, while measures come from *fact tables*.

## DIMENSION TABLE

A *dimension table* contains hierarchical data by which you'd like to summarize. Examples would be an Orders table that you might group by year, month, week, and day of receipt or a Books table that you might want to group by genre and title.

## DIMENSION

Each cube has one or more *dimensions*, each based on one or more dimension tables. A dimension represents a category for analyzing business data: time or category in the examples above. Typically, a dimension has a natural hierarchy so that lower results can be "rolled up" into higher results. For example, in a geographical level you might have city totals aggregated into state totals, or state totals into country totals.

## LEVEL

Each type of summary that can be retrieved from a single dimension is called a *level*. For example, you can speak of a week level or a month level in a time dimension.

## FACT TABLE

A *fact table* contains the basic information that you wish to summarize. This might be order detail information, payroll records, drug effectiveness information, or anything else that's amenable to summing and averaging. Any table that you've used with a Sum or Avg function in a totals query is a good bet to be a fact table.

## MEASURE

Every cube will contain one or more *measures*, each based on a column in a fact table that you'd like to analyze. In the cube of book order information, for example, the measures would be things such as unit sales and profit.

## SCHEMA

Fact tables and dimension tables are related, which is hardly surprising, given that you use the dimension tables to group information from the fact table. The relations within a cube form a *schema*. There are two basic OLAP schemas: star and snowflake. In a *star schema*, every dimension table is related directly to the fact table. In a *snowflake schema*, some dimension tables are related indirectly to the fact table.

For example, if your cube includes Order Details as a fact table, with Customers and Orders as dimension tables, and Customers is related to Orders, which in turn is related to Order Details, then you're dealing with a snowflake schema.

## 4) SQL SERVER VSS WRITER

The SQL Writer Service provides added functionality for backup and restore of SQL Server through the Volume Shadow Copy Service framework.

The SQL Writer Service is installed automatically. It must be running when the Volume Shadow Copy Service (VSS) application requests a backup or restore. To configure the service, use the Microsoft Windows Services applet. The SQL Writer Service installs on all operating systems.

## 5) SQL SERVER FULL TEXT SEARCH

Full-text search allows fast and flexible indexing for keyword-based query of text data stored in a SQL Server database. Unlike the LIKE predicate, which only works on character patterns, full-text queries perform a linguistic search

against this data, operating on words and phrases based on rules of a particular language.

## **BUSINESS INTELLIGENCE DEVELOPMENT STUDIO**

Business Intelligence Development Studio is the primary environment that you will use to develop business solutions that include Analysis Services, Integration Services, and Reporting Services projects.

Business Intelligence Development Studio (BIDS) is a new tool in SQL Server 2005 that you can use for analyzing SQL Server data in various ways. You can build three different types of solutions with BIDS:

- ✓ Analysis Services projects
- ✓ Integration Services projects
- ✓ Reporting Services projects

It is based on the Microsoft Visual Studio development environment but customizes with the SQL Server services-specific extensions and project types, including tools, controls and projects for reports, ETL data flows, OLAP cubes and data mining structure.

# PRACTICAL 2

## DESIGN & CREATE DATA CUBE BY IDENTIFYING DIMENSIONS & MEASURES FOR STAR SCHEMA.

### Definition of Data Cube:

A multidimensional database holds data more like a 3D spreadsheet rather than a relational database. The cube allows different views of the data to be quickly displayed.

A data cube is a multidimensional structure that contains an aggregate value at each point, i.e., the result of applying an aggregate function to an underlying relation. Data cubes are used to implement online analytical processing (OLAP).

A cube is defined by its measures and dimensions. The measures and dimensions in a cube are derived from tables and views in the data source view on which the cube is based.

A cube consists of measures based on one or more fact tables and dimensions based on one or more dimension tables. Dimensions are based on attributes, which are mapped to one or columns in the dimension tables or views in the data source view, and then hierarchies are defined from these attributes.

### Creation of Data Cube:

First of all, we need to create a data source containing table on which we would apply data mining techniques to produce data cube.

The tables used in the data source are as follows:

Supplier\_data:

- ✓ supplier\_id (primary key)
- ✓ supplier\_name
- ✓ address

item\_table:

- ✓ item\_id (primary key)
- ✓ item\_name
- ✓ brand\_name

location

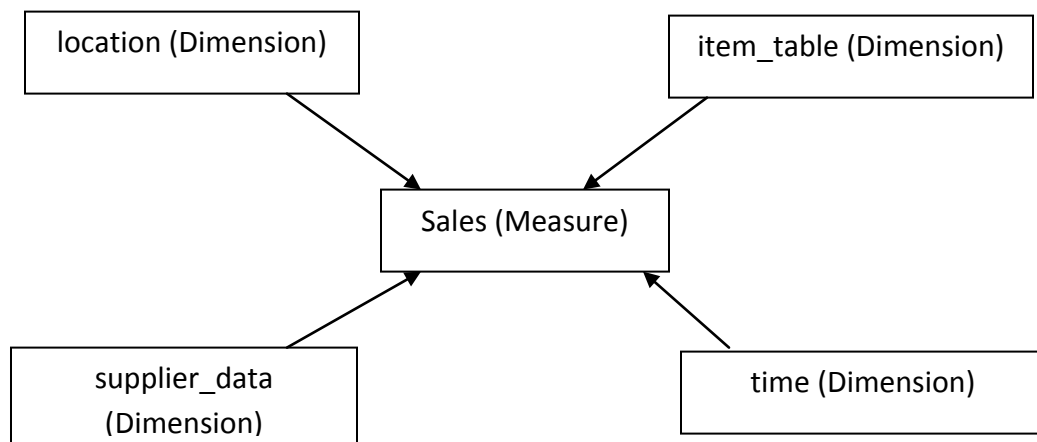
- ✓ location\_id (primary key)
- ✓ location\_name
- ✓ city

time

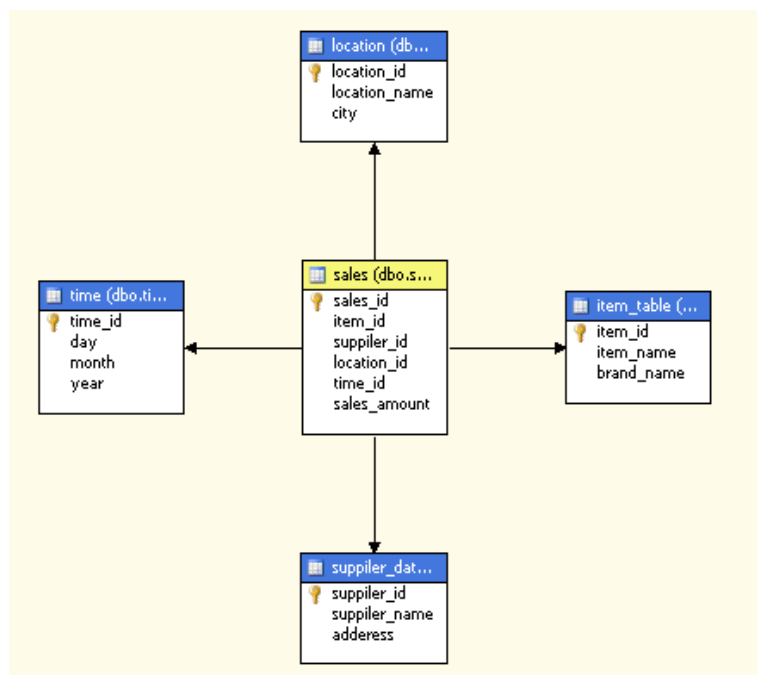
- ✓ time\_id (primary key)
- ✓ day
- ✓ month
- ✓ year

Attributes of sales table (which acts as measure in Data cube) are given below :

- ✓ sales\_id (primary key)
- ✓ sales\_amount
- ✓ item\_id (foreign key reference to item\_table)
- ✓ location\_id (foreign key reference to location)
- ✓ time\_id (foreign key reference to time)
- ✓ supplier\_id (foreign key reference to supplier\_data)

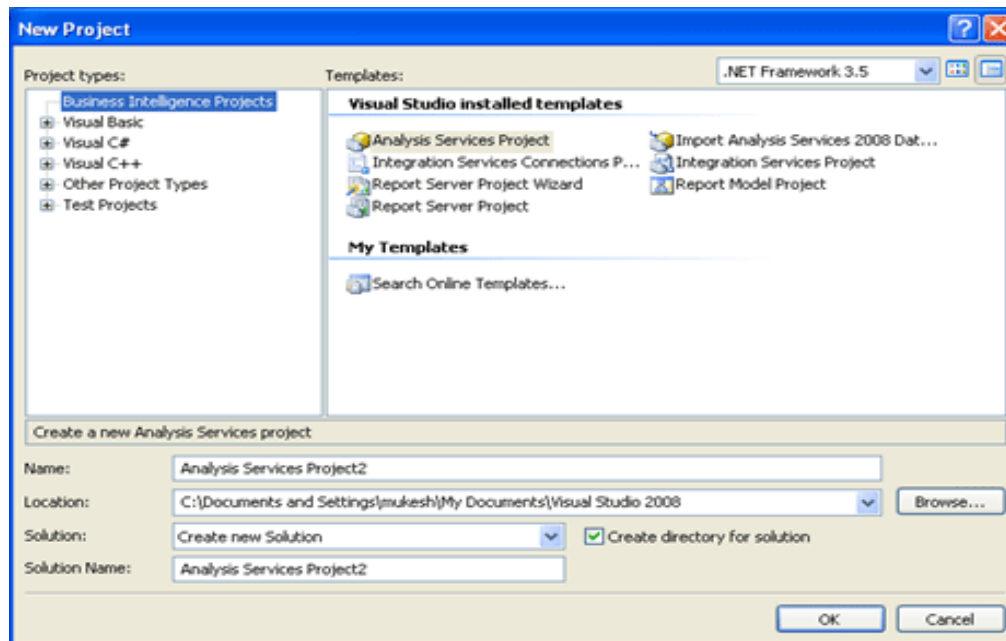


These tables are created in Microsoft SQL Server Management studio and records are filled in the tables. The relationship between the tables is given below:



After making the database we open Microsoft Business Intelligence Development Studio and follow the steps given below:

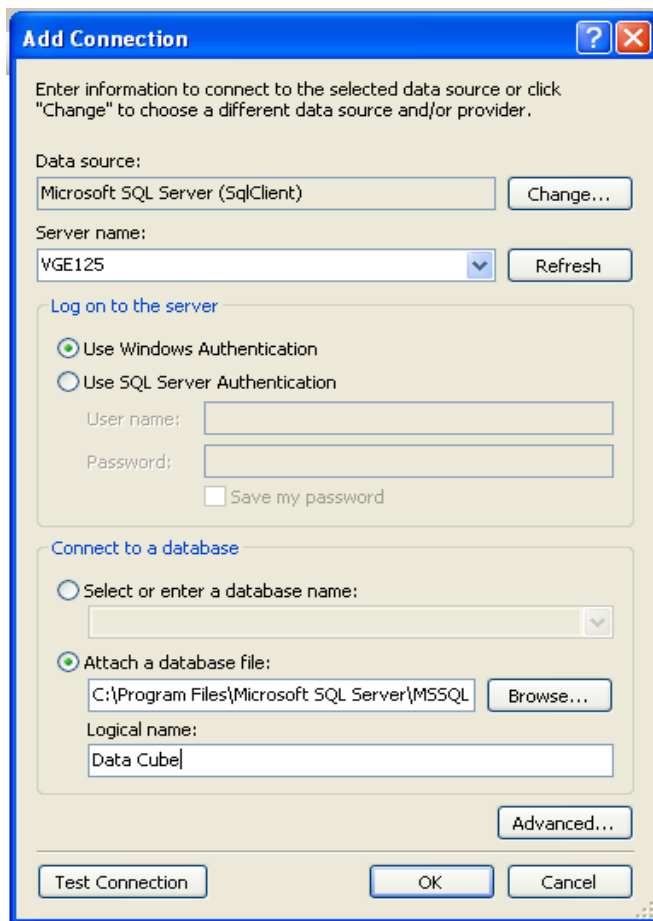
- ✓ Select a new project from file menu as show in the figure:



- ✓ Select Analysis Services Project and give proper name to the project and click on "OK" button

After that click on the "Server Explorer" present on the left side of the screen to choose data connection and add a new connection with database that we have already made using Microsoft SQL Server Management studio.





**Add Connection**

Enter information to connect to the selected data source or click "Change" to choose a different data source and/or provider.

Data source:  
Microsoft SQL Server (SqlClient) Change...

Server name:  
VGE125 Refresh

**Log on to the server**

☒ Use Windows Authentication  
☐ Use SQL Server Authentication

User name:   
Password:   
☐ Save my password

**Connect to a database**

☐ Select or enter a database name:

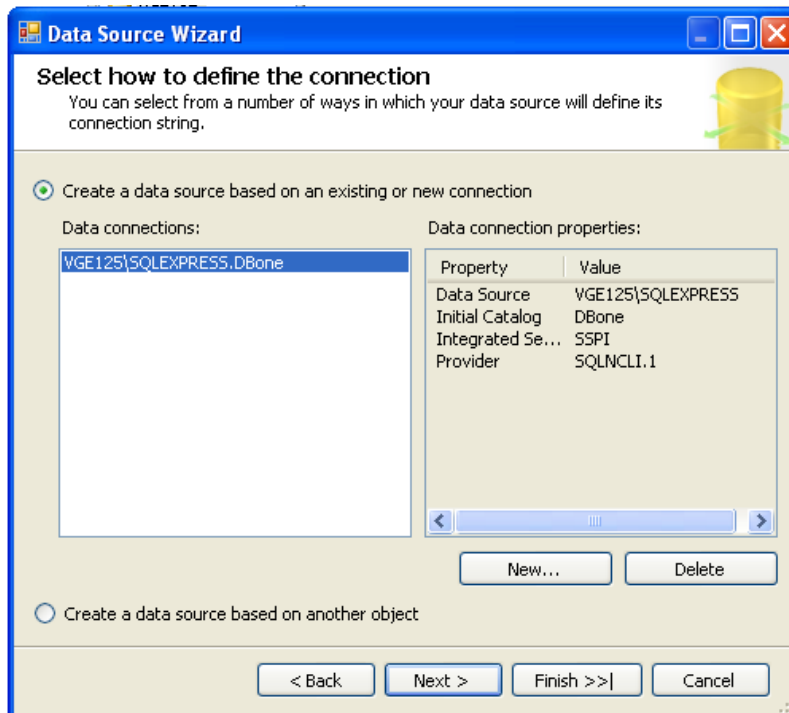
☒ Attach a database file:  
C:\Program Files\Microsoft SQL Server\MSSQL Browse...

Logical name:  
Data Cube

Advanced...

Test Connection OK Cancel

✓ First add connection then select new data source from Solution Explorer.



**Data Source Wizard**

**Select how to define the connection**  
You can select from a number of ways in which your data source will define its connection string.

☒ Create a data source based on an existing or new connection

Data connections:  
VGE125\SQLEXPRESS.DBone

Data connection properties:

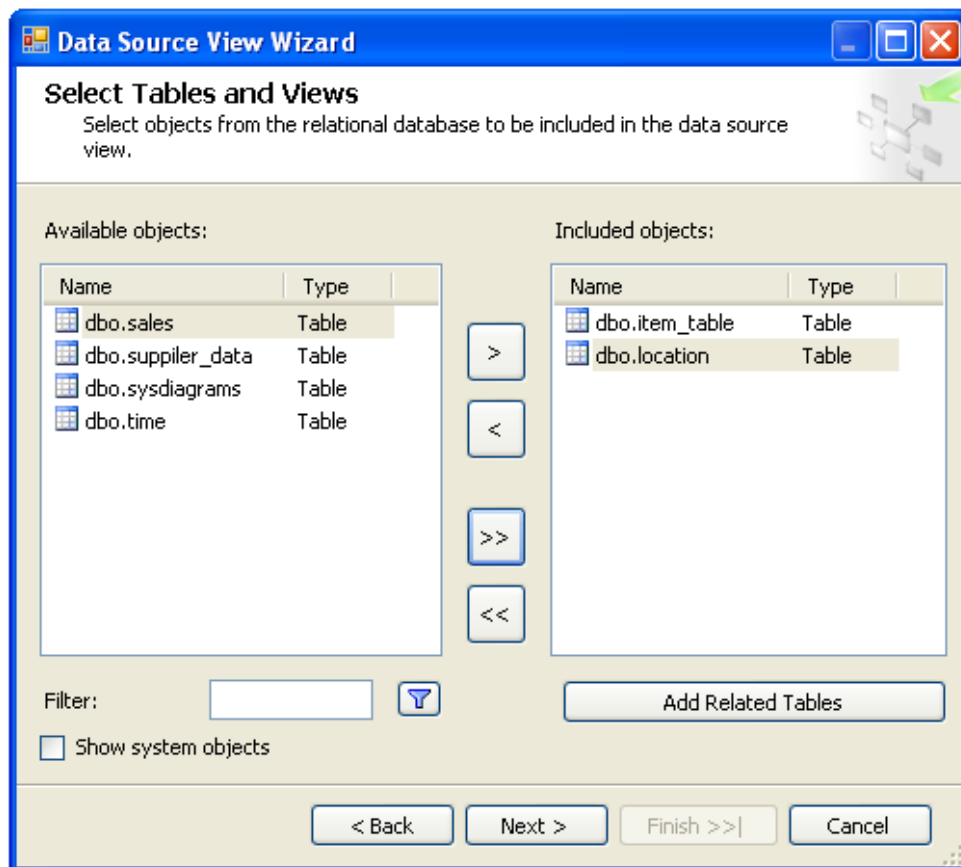
Property	Value
Data Source	VGE125\SQLEXPRESS
Initial Catalog	DBone
Integrated Se...	SSPI
Provider	SQLNCLI.1

New... Delete

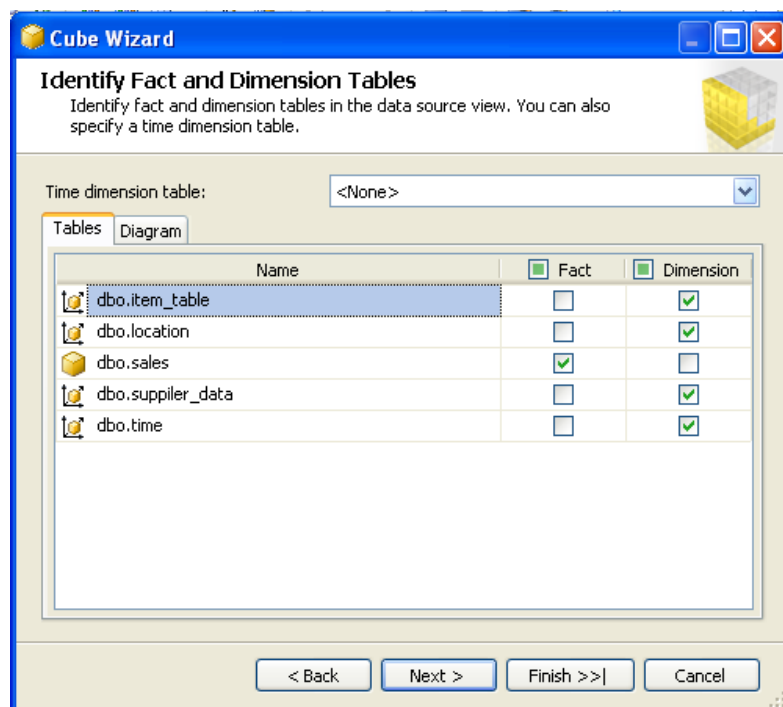
☐ Create a data source based on another object

< Back Next > Finish >>| Cancel

- ✓ Create view from the data source to get important Dimensions.



- ✓ After creating views create cube based on the views. Process the cube and the browse the cube.



- ✓ Creating a cube showing facts and dimensions

VG125

Measures

Sales

Sale

Sale

Tim

Item Table

Item Table

Location

Location 1

Supplier Da

Supplier Da

Time

Time 1

Dimension	Hierarchy	Operator	Filter Expression
<Select dimension>			

Drop Filter Fields Here

	Supiler Name									
	ami	ankita	chetan	chintan	devang	mitesh	mohit	rajan	vishal	Grand Total
City	Sales Amount	Sales Amount	Sales Amount	Sales Amount	Sales Amount	Sales Amount	Sales Amount	Sales Amount	Sales Amount	Sales Amount
ahmedabad	15000	4100	3250	56900	3600		4500	18500		105850
gandhinagar						5900			8300	14200
jaipur			36490							36490
Jamnagar	14000									14000
Rajkot					15255					15255
surat						4200			10500	14700
surendranagar				8000	8200					16200
Grand Total	29000	4100	39740	64900	27055	10100	4500	18500	18800	216695

# PRACTICAL 3

## DESIGN & CREATE DATA CUBE BY IDENTIFYING DIMENSIONS & MEASURES FOR SNOWFLAKE SCHEMA

A **snowflake schema** is a logical arrangement of tables in a multidimensional database such that the entity relationship diagram resembles a snowflake in shape. The snowflake schema is represented by centralized fact tables which are connected to multiple dimensions.

The snowflake schema is similar to the star schema. However, in the snowflake schema, dimensions are normalized into multiple related tables, whereas the star schema's dimensions are normalized with each dimension represented by a single table. A complex snowflake shape emerges when the dimensions of a snowflake schema are elaborate, having multiple levels of relationships, and the child tables have multiple parent tables ("forks in the road"). The "snowflaking" effect only affects the dimension tables and **NOT** the fact tables.

### ATTRIBUTES OF EACH TABLE (DIMENSION) ARE AS GIVEN BELOW :

#### Supplier:

- ✓ supplier\_id (primary key)
- ✓ supplier\_type

#### item:

- ✓ Item\_id (primary key)
- ✓ Item\_Brand
- ✓ Item\_name
- ✓ Item\_type

#### location :

- ✓ Lid (primary key)
- ✓ location\_street

#### City:

- ✓ City\_id
- ✓ City\_name
- ✓ City\_state
- ✓ City\_country

#### Branch:

- ✓ Branch\_id
- ✓ Branch\_name
- ✓ Branch\_type

#### time:

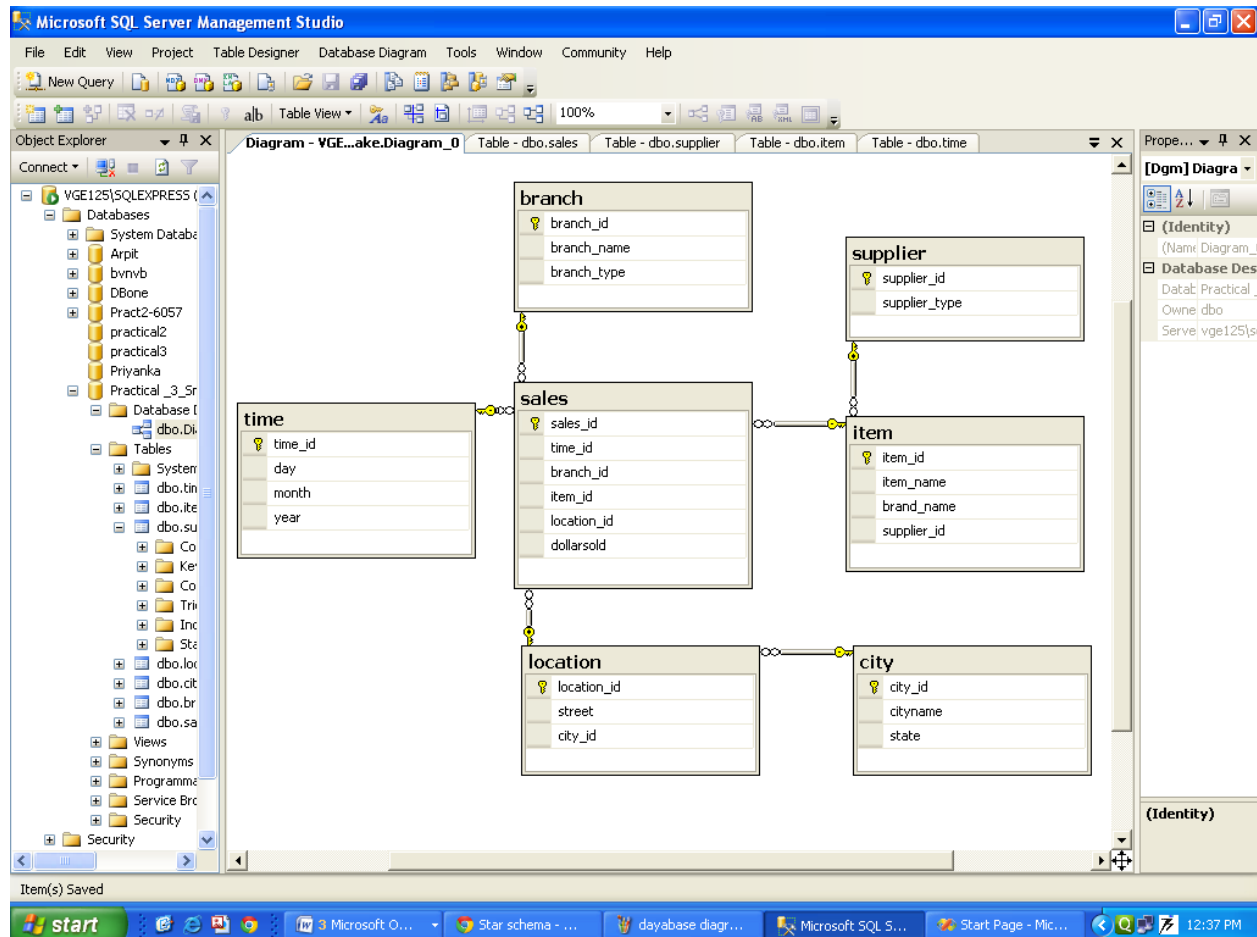
- ✓ Time\_id (primary key)
- ✓ day
- ✓ month
- ✓ year

### ➤ Attributes of sales table (Measure) are as given Below :

- ✓ sales\_id (primary key)
- ✓ Dollarsold

- ✓ Item\_id (foreign key)
- ✓ Location\_id (foreign key)
- ✓ Time\_id (foreign key)
- ✓ Branch\_id (foreign key)

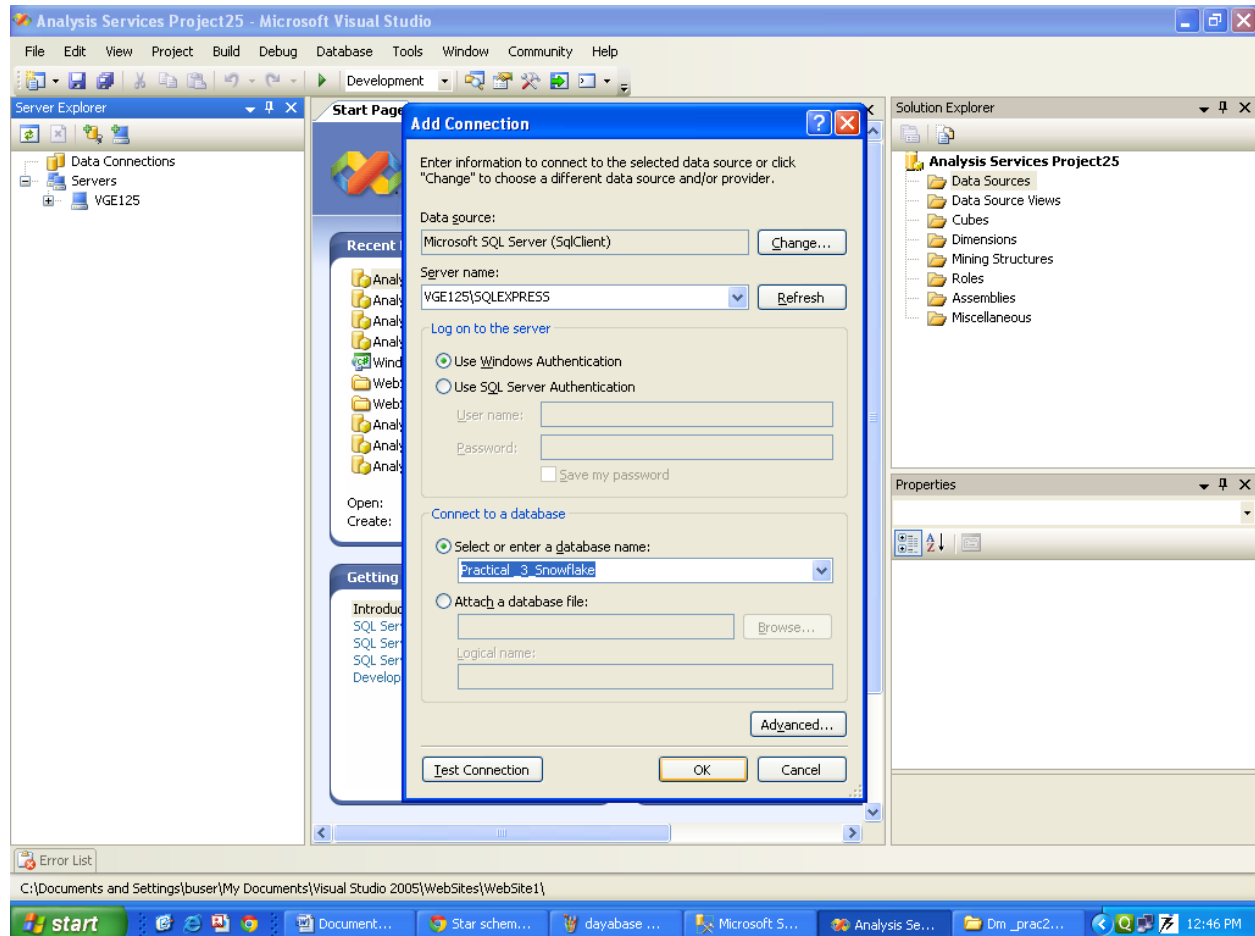
The relationship between the tables is given below:



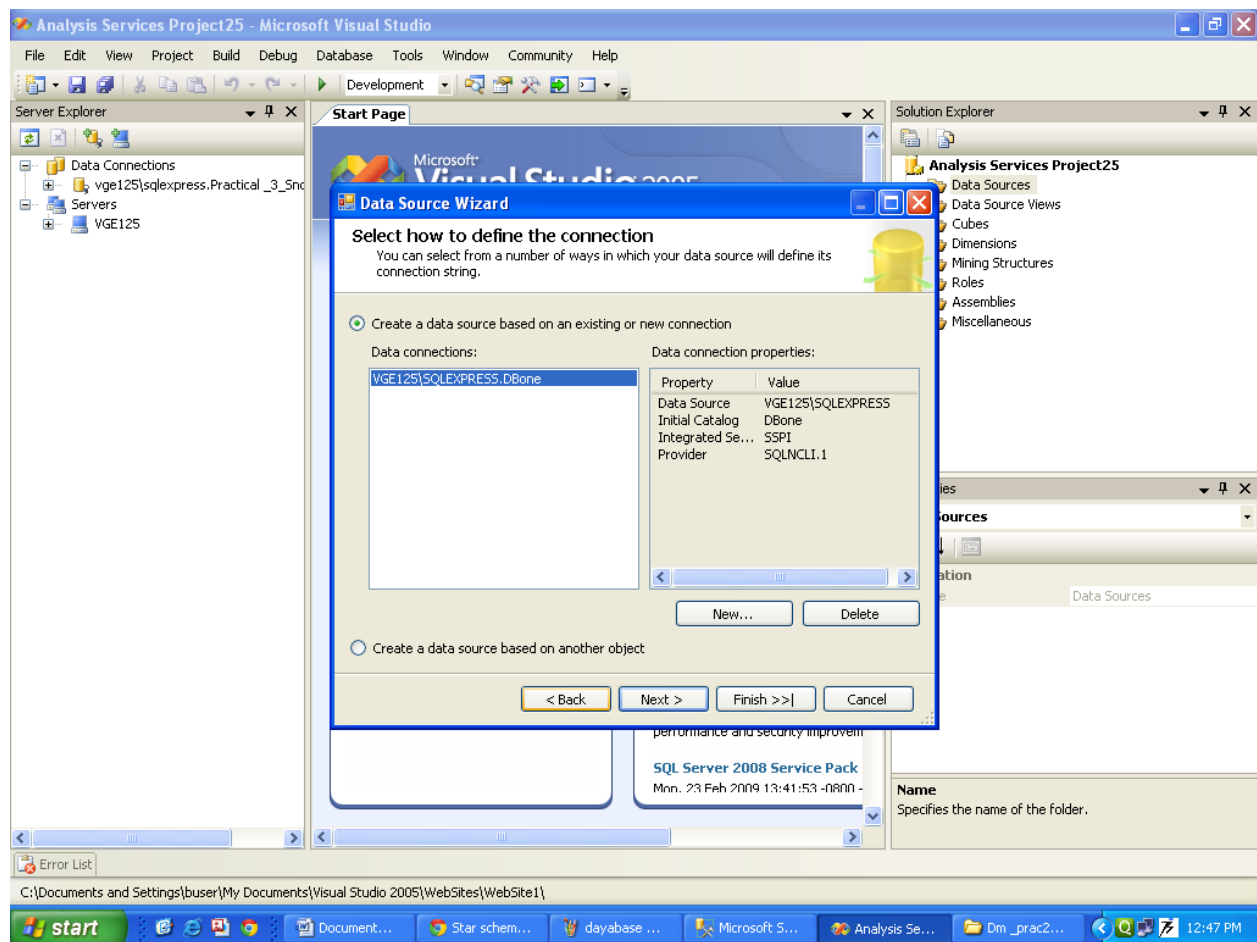
After making the database we open Microsoft Business Intelligence Development Studio and follow the steps given below:

- ✓ Select a new project from file menu.

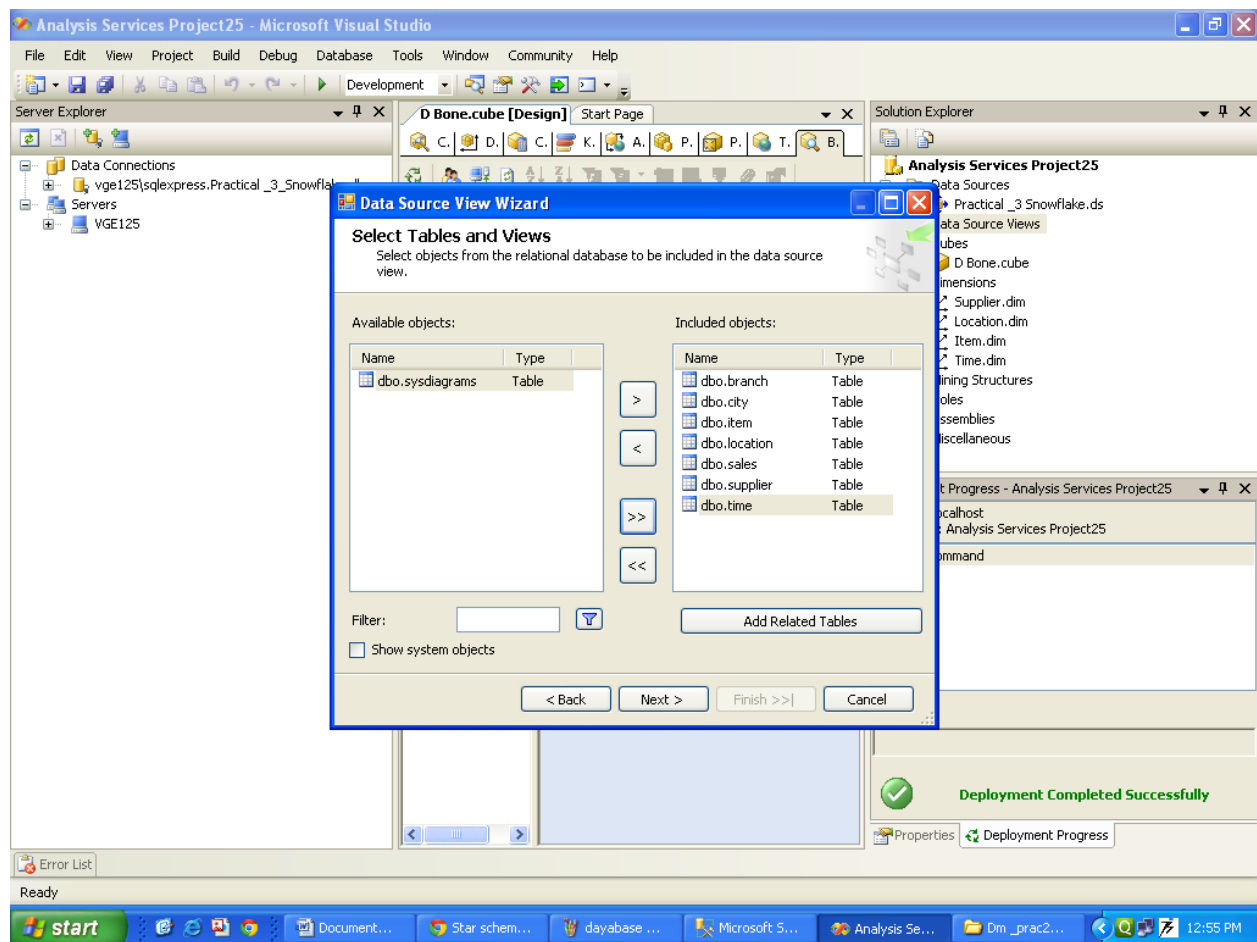
- ✓ Select Analysis Services Project and give proper name to the project and click on “OK” button
- ✓ After that click on the “Server Explorer” present on the left side of the screen to choose data connection and add a new connection with database that we have already made using Microsoft SQL Server Management studio.



- ✓ First add connection then select new data source from Solution Explorer.

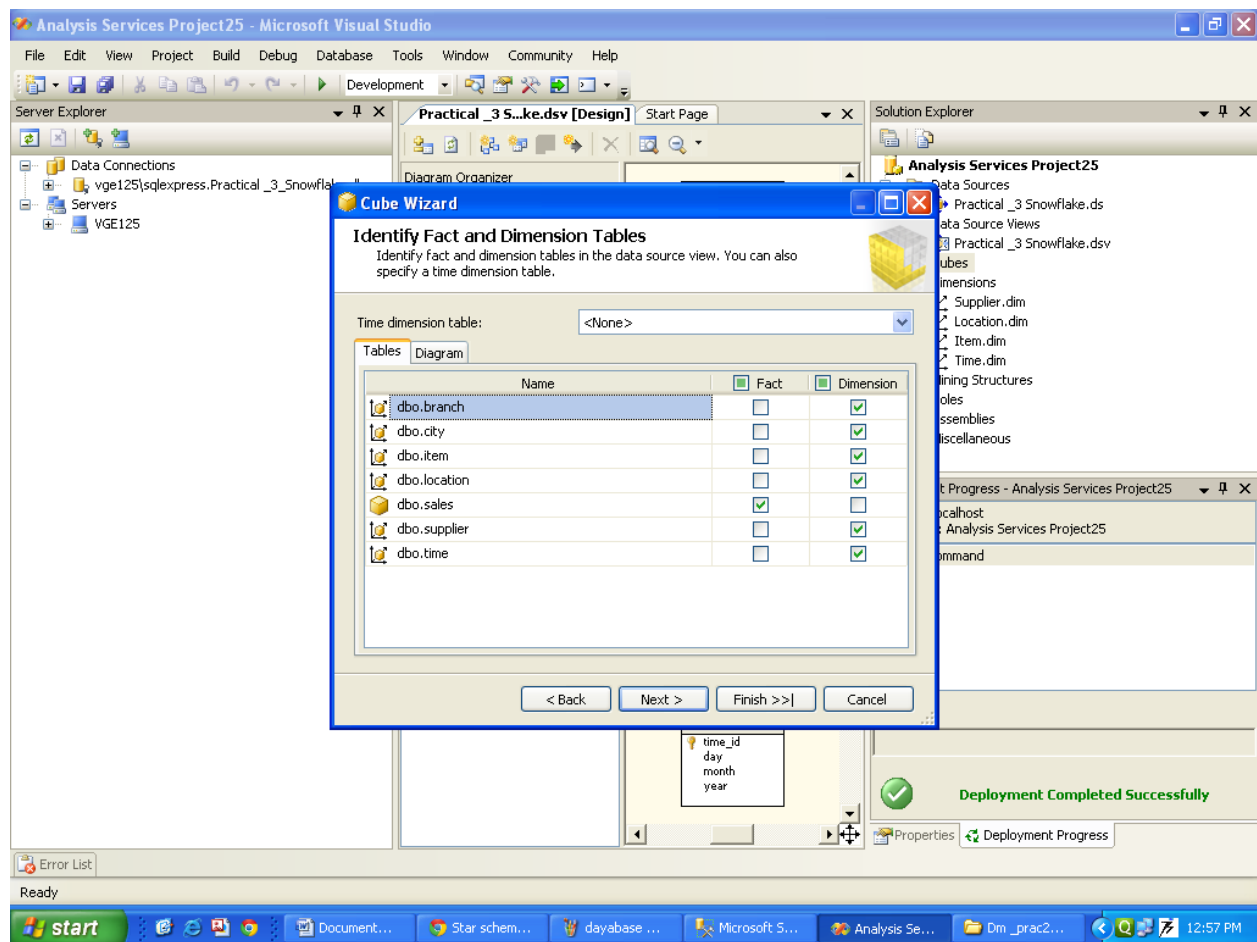


✓ Create view from the data source to get important Dimensions.

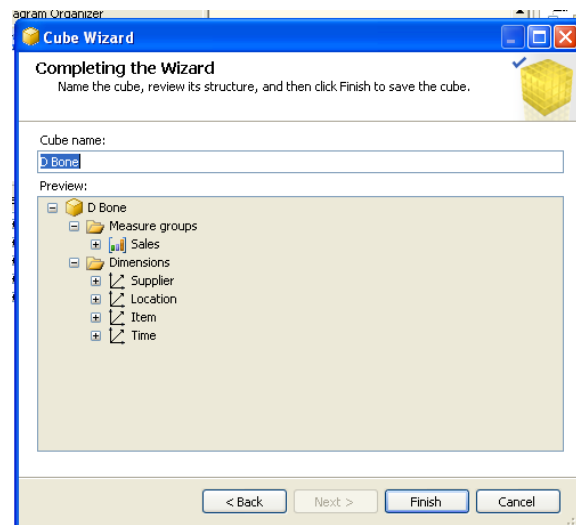


- ✓ After creating views create cube based on the views. Process the cube and then browse the cube. As shown in the figure the fact and dimension tables are automatically detected.





This figure shows the completion of building a cube



✓ Finally we can create a cube showing facts and dimensions and performing analysis on it.

**Analysis Services pract-3-6057 - Microsoft Visual Studio**

File Edit View Project Build Debug Database Cube Tools Window Community Help

Development

**Pract-3 6057.cube [Design]** Pract-3 6057.dsv [Design] Start Page

Cube Structure Dimension Usage Calculations KPIs Actions Partitions Perspectives Translations Browser

Perspective: Pract-3 6057 Language: Default

**Server Explorer**

- Pract-3 6057
  - Measures
    - Dollar Sales
    - Sales Count
  - Branch
  - Item
    - Brand Name
    - Item
    - Item Name
    - Supplier
    - Supplier Type
    - Supplier - Item
  - Location
  - Time

**Dimension** Hierarchy Operator Filter Expression

<Select dimension>

Drop Filter Fields Here

		City		Location		Branch				
		1	3	Total	2	Total	3	4	Grand Total	
Supplier	Item Name	Day	Dollar Sales	Dollar Sales	Dollar Sales	Dollar Sales	Dollar Sales	Dollar Sales	Dollar Sales	
1	dairymilk		1005		1005			500	1505	
	Total		1005		1005			500	1505	
2	fruit n nut				1400	1400			1400	
	munch	20	650		650			830	1480	
		25					580	780	1360	
	Total		650		650		580	1610	2840	
			650		650	1400	1400	580	4240	
3	dark chocolate			1400	1400	300	300	700	2400	
	Total			1400	1400	300	300	700	2400	
4			600	550	1150			1000	2150	
Grand Total			2255	1950	4205	1700	1700	1280	10295	

**Solution Explorer**

- Analysis Services
  - Data Sources
    - Pract-3 6057
  - Data Source View
    - Pract-3 6057
  - Cubes
    - Pract-3 6057
  - Dimensions
    - Location.dim
    - Item.dim
    - Branch.dim
    - Time.dim
  - Mining Structures
  - Roles
  - Assemblies

**Deployment Pro...**

Server: localhost  
Database: Analysis

Command

Status:

**Deployment Completed**

Properties Deploy...

Ready

start Microsoft SQL Server ... Analysis Services pra... Images - Microsoft Word 1:05 PM

# PRACTICAL 4

---

## CREATING VARIOUS STORAGE MODES FOR CUBE AGGREGATION.

---

### CUBE AGGREGATION:

- ❖ Efficient drilling or traversing of the cube data is a key factor in flexible and swift decision making and analysis. In order to maintain speed and consistency in reporting, data is usually pre-calculated or aggregated. An important factor in query performance is good aggregation design, which includes decisions about total storage space, available build time, storage location, and storage format.
- ❖ When planning your data storage and design, it is helpful to approximate the size of aggregations. A basis for estimating aggregation size is the number of distinct values in a dimension level, otherwise known as cardinality. The other factor that determines aggregations size is density. Density is a measure of how many members of each dimension in an aggregation occur in combination with the members of the other dimensions (For example, there might not be sales of a specific product on a specific date). The total cube size, as well as the resources that are available for the cube build process, determine the build time that is needed. It is also important to note that build time should not exceed the cube update interval.
- ❖ Microsoft SQL Server Analysis Services incorporates a sophisticated algorithm to select aggregations for pre-calculation so that other aggregations can be quickly computed from the pre-calculated values. For example, if the aggregations are pre-calculated for the Month level of a Time hierarchy, the calculation for a Quarter level requires only the summarization of three numbers, which can be quickly computed on demand.

## **STORAGE MODES:**

There are three standard storage modes (**MOLAP**, **ROLAP** and **HOLAP**) in OLAP applications which affect the performance of OLAP queries and cube processing, storage requirements and also determine storage locations.

### ❖ **MOLAP:**

This is the default and most frequently used storage mode. The MOLAP storage mode causes the aggregations of the partition and a copy of its source data to be stored in a multidimensional structure in Analysis Services when the partition is processed. This MOLAP structure is highly optimized to maximize query performance. The storage location can be on the computer where the partition is defined or on another computer running Analysis Services. Because a copy of the source data resides in the multidimensional structure, queries can be resolved without accessing the partition's source data. Query response times can be decreased substantially by using aggregations. The data in the partition's MOLAP structure is only as current as the most recent processing of the partition.

As the source data changes, objects in MOLAP storage must be processed periodically to incorporate those changes and make them available to users. Processing updates the data in the MOLAP structure, either fully or incrementally.

### ❖ **ROLAP:**

The ROLAP storage mode causes the aggregations of the partition to be stored in indexed views in the relational database that was specified in the partition's data source. Unlike the MOLAP storage mode, ROLAP does not cause a copy of the source data to be stored in the Analysis Services data folders. Instead, when results cannot be derived from the query cache, the indexed views in the data source is accessed to answer queries. Query response is generally slower with ROLAP storage than with the MOLAP or HOLAP storage modes. Processing time is also typically slower with ROLAP. However, ROLAP enables users to view data in real time and can save storage space when you are working with large datasets that are infrequently queried, such as purely historical data.

**❖ HOLAP:**

The HOLAP storage mode combines attributes of both MOLAP and ROLAP. Like MOLAP, HOLAP causes the aggregations of the partition to be stored in a multidimensional structure in an SQL Server Analysis Services instance. HOLAP does not cause a copy of the source data to be stored. For queries that access only summary data in the aggregations of a partition, HOLAP is the equivalent of MOLAP.

Queries that access source data—for example, if you want to drill down to an atomic cube cell for which there is no aggregation data—must retrieve data from the relational database and will not be as fast as they would be if the source data were stored in the MOLAP structure. With HOLAP storage mode, users will typically experience substantial differences in query times depending upon whether the query can be resolved from cache or aggregations versus from the source data itself.

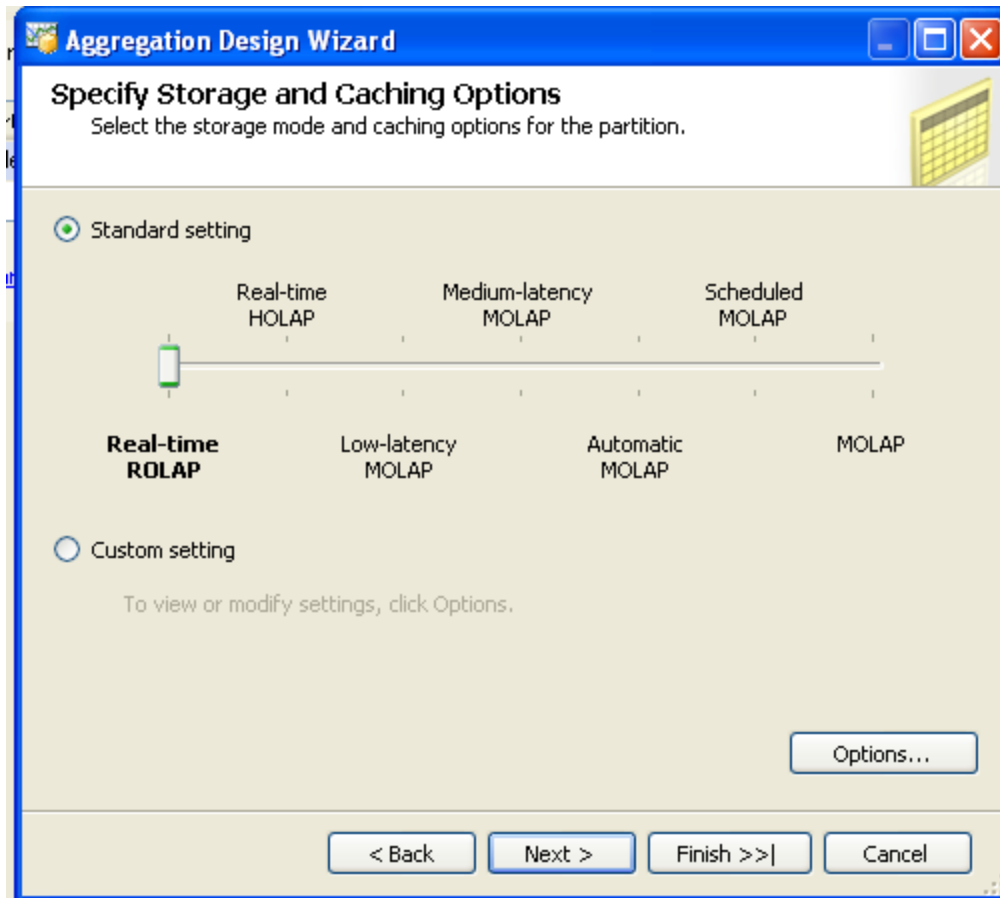
Partitions stored as HOLAP are smaller than the equivalent MOLAP partitions because they do not contain source data and respond faster than ROLAP partitions for queries involving summary data. HOLAP storage mode is generally suited for partitions in cubes that require rapid query response for summaries based on a large amount of source data.

**PROCEDURE FOR CUBE AGGREGATION:**

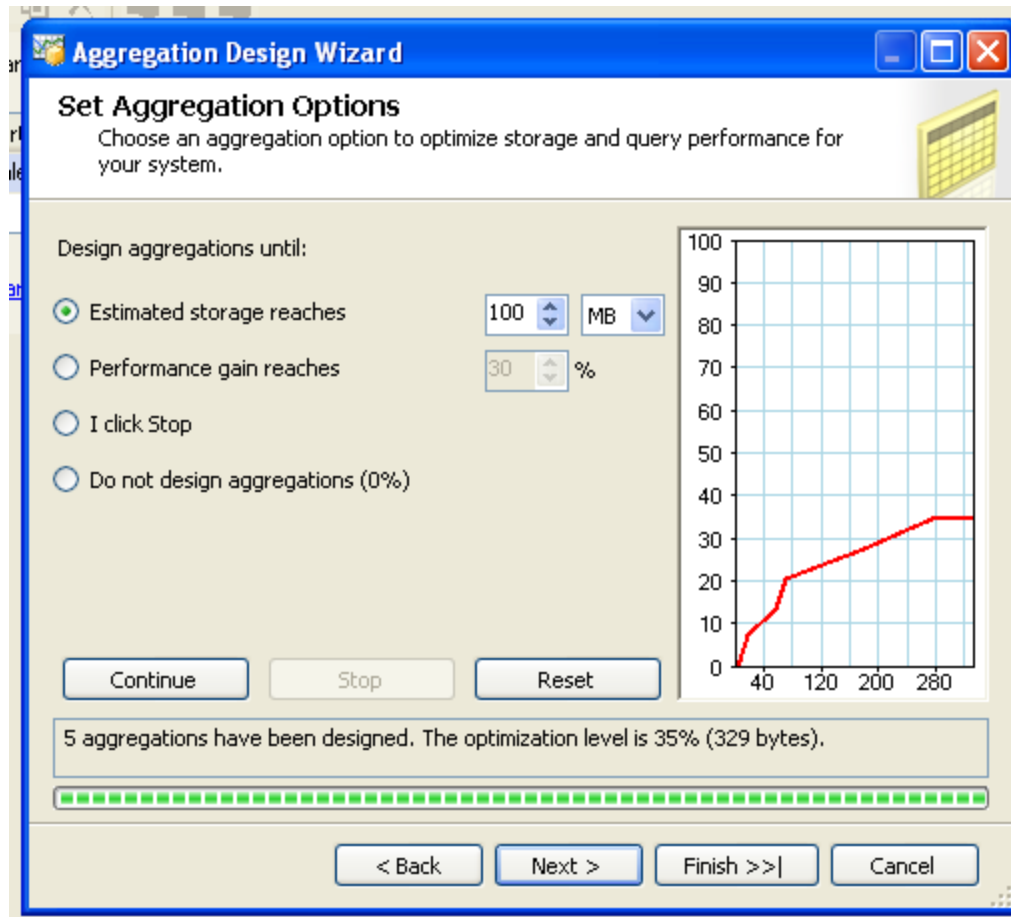
- ❖ Aggregations are designed by using the wizards but are not actually calculated until the partition for which the aggregations are designed is processed. After the aggregation has been created, if the structure of a cube ever changes, or if data is added to or changed in a cube's source tables, it is usually necessary to review the cube's aggregations and process the cube again. Primarily aggregation is 0% and default storage mode is MOLAP.



- ❖ The Aggregation Design Wizard provides options for you to specify storage mode to achieve a satisfactory tradeoff between query response time and storage requirements. We have choose **Real-time ROLAP**.

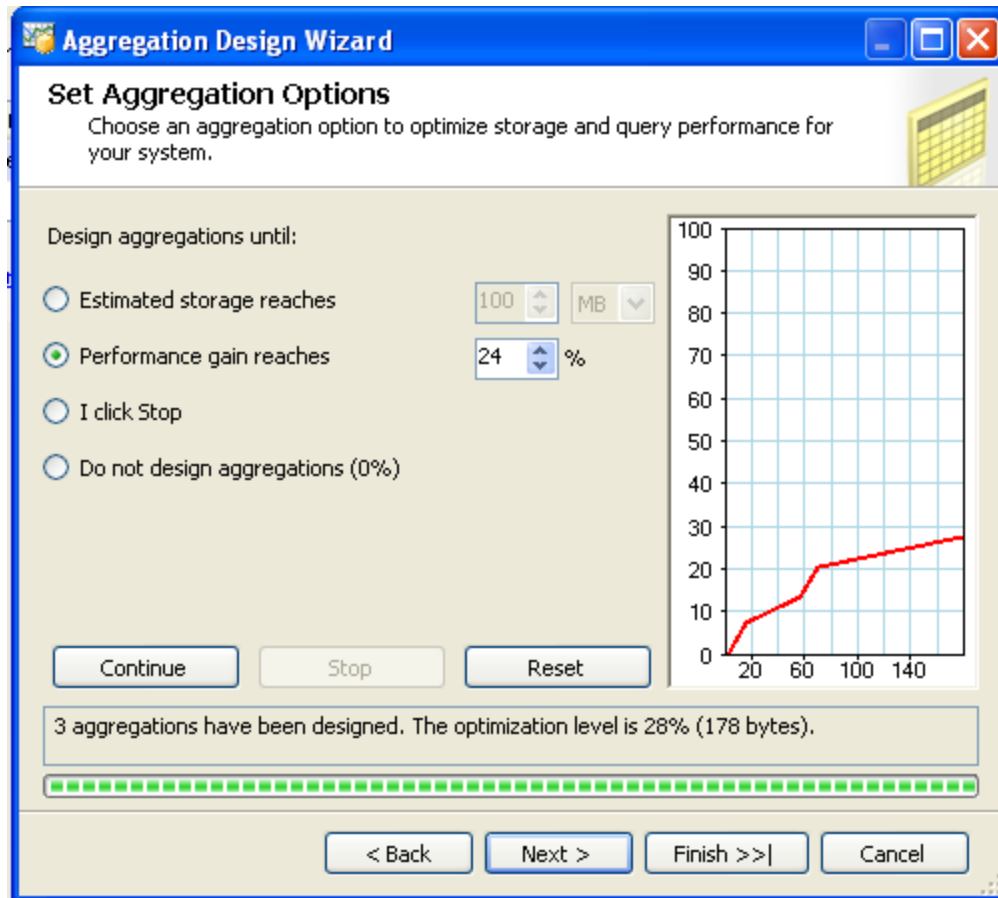


- ❖ After choosing storage mode, we can choose appropriate aggregation option to optimize storage and query performance of the system.  
For Example, we can choose criteria for estimated storage reaches 100MB. The performance of the system varies system by system. Here performance is nearly 35%.



- ❖ We can also choose second criteria that is performance gain reaches 24%. We can adjust the amount of percentage for performance gain.





- ❖ Similarly we can choose other storage modes such as Real-time HOLAP, Low-latency MOLAP, Medium-latency MOLAP, Automatic MOLAP, Scheduled MOLAP etc.

## CONCLUSION:

By creating cube aggregation using different storage modes, we can save processing time and reduce storage requirements, with minimal effect on query response time. We can also increase responsiveness to frequent queries and decrease responsiveness to infrequent queries without significantly affecting the storage needed for the cube.

# PRACTICAL 5

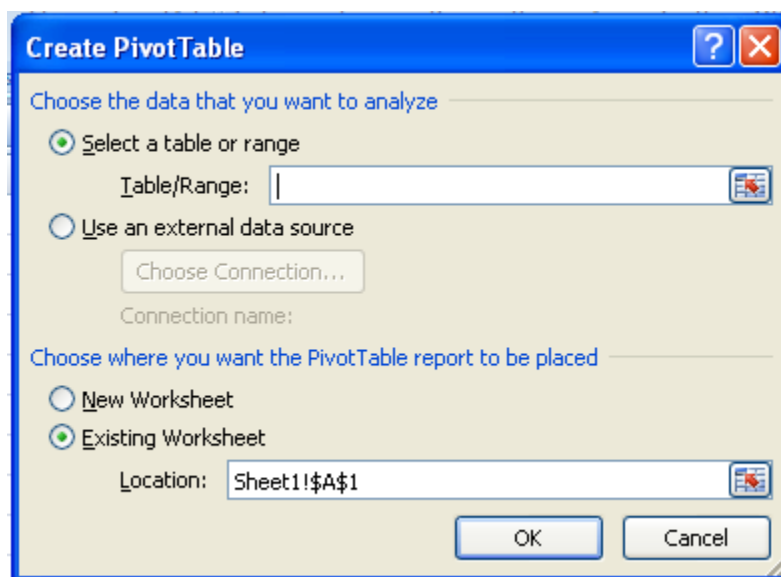
## CREATE & USE MICROSOFT OFFICE EXCEL PIVOT TABLE FOR A DATA CUBE.

### ❖ MICROSOFT OFFICE EXCEL PIVOT TABLE

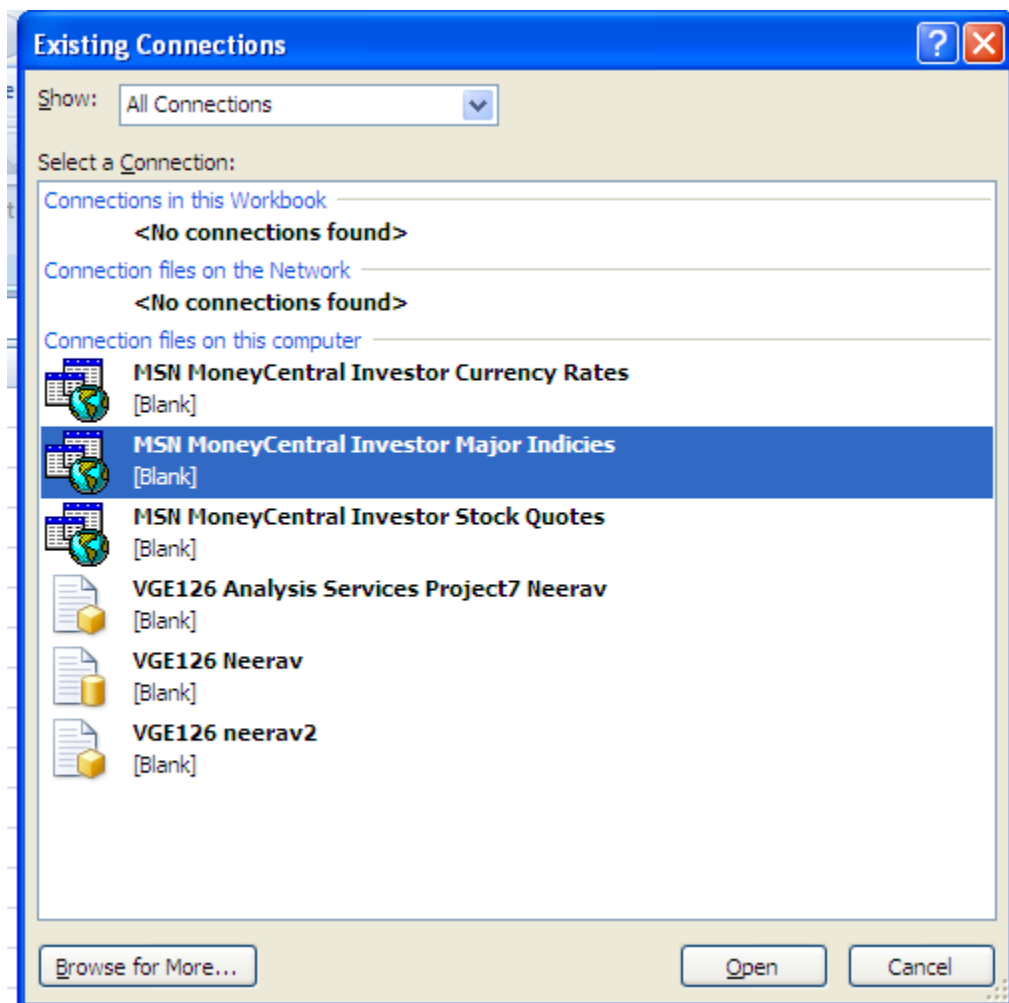
In data processing, a pivot table is a data summarization tool found in data visualization programs such as spreadsheets (for example, in Microsoft Excel, OpenOffice.org Calc, LibreOffice Calc, Google Docs and Lotus 1-2-3) or business intelligence software. Among other functions, pivot-table tools can automatically sort, count, total or give the average of the data stored in one table or spreadsheet. It displays the results in a second table (called a "pivot table") showing the summarized data. The user sets up and changes the summary's structure by dragging and dropping fields graphically.

### ❖ CREATING A PIVOT TABLE

- As we are creating pivot table using external data source we have to make connection with the external data source i.e. Microsoft SQL Server Analysis Services.

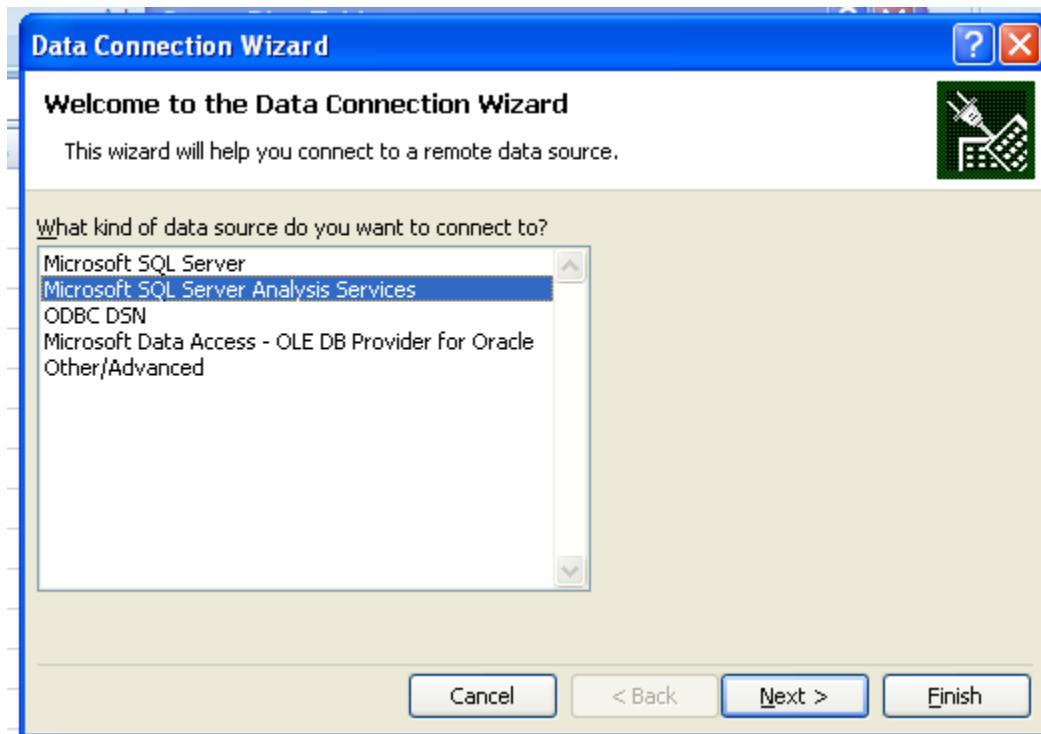


- As we don't have existing connection with the data source we have to browse it.

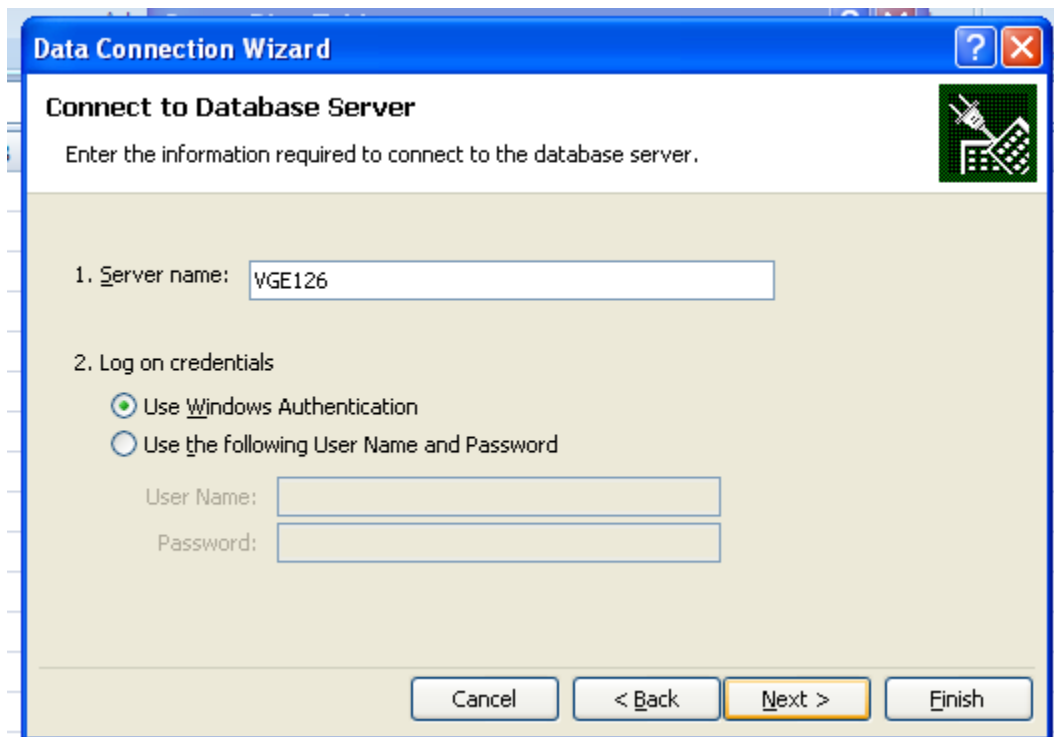


- Now the following three steps are use to connect with the cube that was created as a project in Microsoft SQL Server Business Intelligence Development Studio.

- This wizard is use to select remote data source connection.

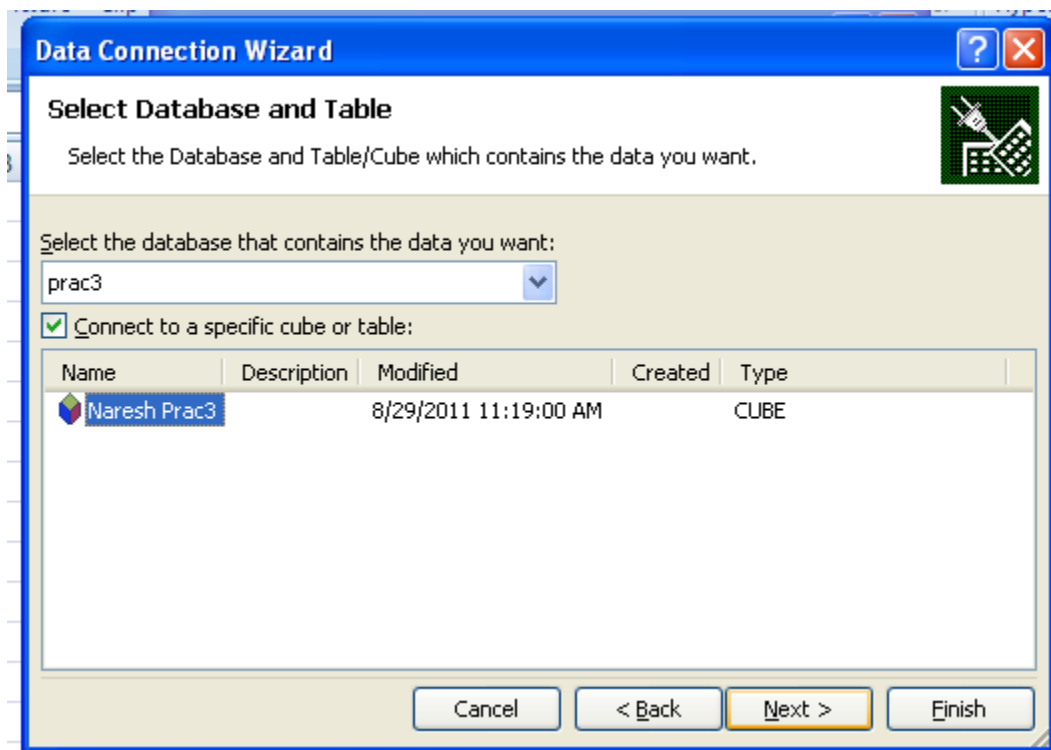


- We have to fill the database server name to which we have to connect.

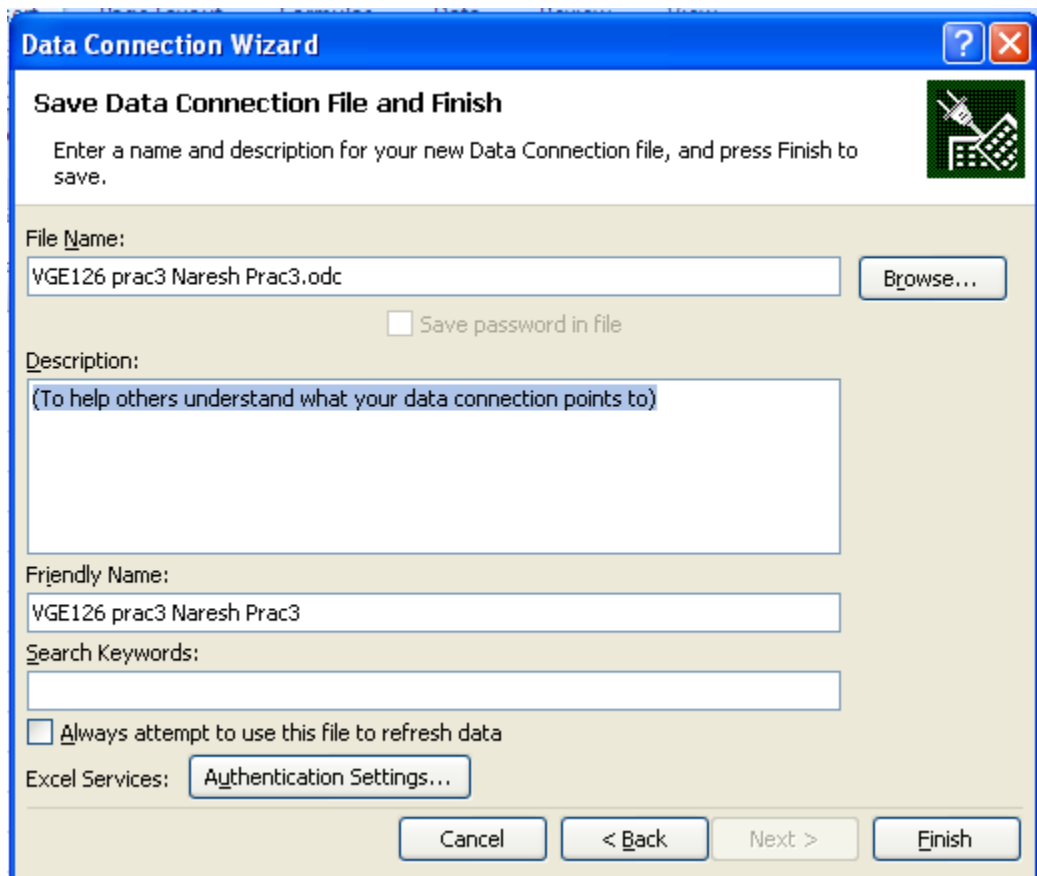


The screenshot shows a Windows-style dialog box titled "Data Connection Wizard". The main heading is "Connect to Database Server". Below the heading is the instruction "Enter the information required to connect to the database server." and a green icon of a plug connecting to a server rack. The first step is "1. Server name:" with a text box containing "VGE126". The second step is "2. Log on credentials" with two radio button options: "Use Windows Authentication" (which is selected) and "Use the following User Name and Password". Below these are empty text boxes for "User Name:" and "Password:". At the bottom are four buttons: "Cancel", "< Back", "Next >" (highlighted with a yellow border), and "Finish".

- In this step, we select the data cube on which the pivot table is to be generated.



- Once the connection process is completed we have to save the data connection file, as show in the below snapshot.



**Data Connection Wizard**

**Save Data Connection File and Finish**

Enter a name and description for your new Data Connection file, and press Finish to save.

File Name:  
VGE126 prac3 Naresh Prac3.odc Browse...

☐ Save password in file

Description:  
(To help others understand what your data connection points to)

Friendly Name:  
VGE126 prac3 Naresh Prac3

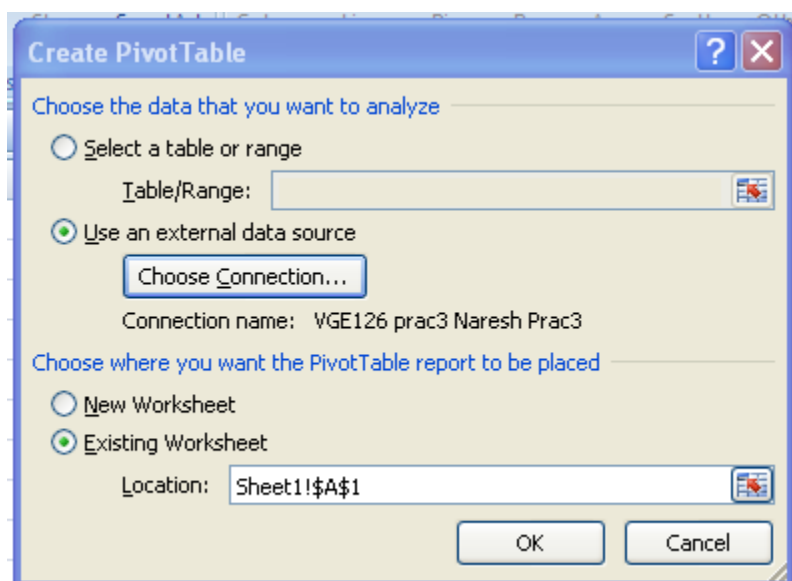
Search Keywords:

☐ Always attempt to use this file to refresh data

Excel Services: Authentication Settings...

Cancel < Back Next > Finish

- Now we are ready to create a pivot table using an external data source.



**Create PivotTable**

Choose the data that you want to analyze

☐ Select a table or range  
Table/Range:

☒ Use an external data source  
Choose Connection...  
Connection name: VGE126 prac3 Naresh Prac3

Choose where you want the PivotTable report to be placed

☐ New Worksheet

☒ Existing Worksheet  
Location: Sheet1!\$A\$1

OK Cancel

- Below are few snapshots obtained by selecting various fields of the dimension and fact table.

The screenshot shows Microsoft Excel with a PivotTable and the PivotTable Field List task pane. The PivotTable is located in the range A2:C25. The PivotTable Field List task pane is on the right, showing the following fields:

- Report Filter:** Branch (checked), Branch Name (unchecked), Branch Type (unchecked)
- Row Labels:** Branch (checked), Supplier - Item (checked), More fields (unchecked), Location (unchecked)
- Column Labels:** Sales Count (checked), Dollar Sold (checked)
- Values:** Sales Count (checked), Dollar Sold (checked)

The PivotTable data is as follows:

Row Labels	Sales Count	Dollar Sold
1	6	31135
2	3	9625
3	2	11510
4	1	10000
5	5	27984
6	3	15443
7	1	6541
8	1	6000
9	6	28622
10	1	5800
11	4	16948
12	1	5874
13	6	98708
14	4	84646
15	2	14062
16	6	132305
17	4	119480
18	2	12825
19	4	18497
20	1	100
21	2	12555
22	1	5842
<b>Grand Total</b>	<b>33</b>	<b>337251</b>



The screenshot displays the Microsoft Excel interface with a PivotTable and the PivotTable Field List task pane.

**PivotTable Data:**

Row Labels	Sales Count	Dollar Sold
<b>Gujarat</b>	27	301884
Chandkheda	6	31135
Gateway Of India	5	22748
Iscon	6	132305
Lal Darwaja	3	78800
Udhyogbhavan	3	18399
University	4	18497
<b>Maharashtra</b>	6	35367
Gateway Of India	1	5874
Lal Darwaja	3	19908
Udhyogbhavan	2	9585
<b>Grand Total</b>	33	337251

**PivotTable Field List:**

- Sales**
  - ☒ Dollar Sold
  - ☒ Sales Count
- Branch**
  - ☐ Branch
  - ☒ Branch Name
  - ☐ Branch Type
- Item**
  - ☐ Supplier - Item
  - ☐ More fields
- Location**
  - ☒ State - City
  - ☐ More fields
- Time**

**Report Filter:** [Empty]

**Column Labels:** [Σ] Values

**Row Labels:** [Σ] Values

**State - City:** [Dropdown]

**Sales Count:** [Dropdown]

**Defer Layout Update:** ☐ **Update:** [Button]

## Conclusion:-

Microsoft Excel Pivot Table provides the features to connect to various data sources like SQL, Oracle, Access etc provided their connectivity drivers are already installed. Various reports can be produced using Excel's Pivot Table.

## PRACTICAL 6

---

### PERFORM BOX-PLOT ANALYSIS FOR GIVEN DATA USING XLMiner.

---

**DATA (AGE):**

13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33, 33,  
35, 35, 35, 35, 36, 40, 45, 46, 52, 70

A **box plot** is a convenient way of graphically depicting groups of numerical data through their five-number summaries: the smallest observation (sample minimum), lower quartile (Q1), median (Q2), upper quartile (Q3), and largest observation (sample maximum). A box-plot may also indicate which observations, if any, might be considered outliers. Any data points which are laying outside the upper or lower extremes are considered as outliers.

For above given data,

Lower quartile = 35

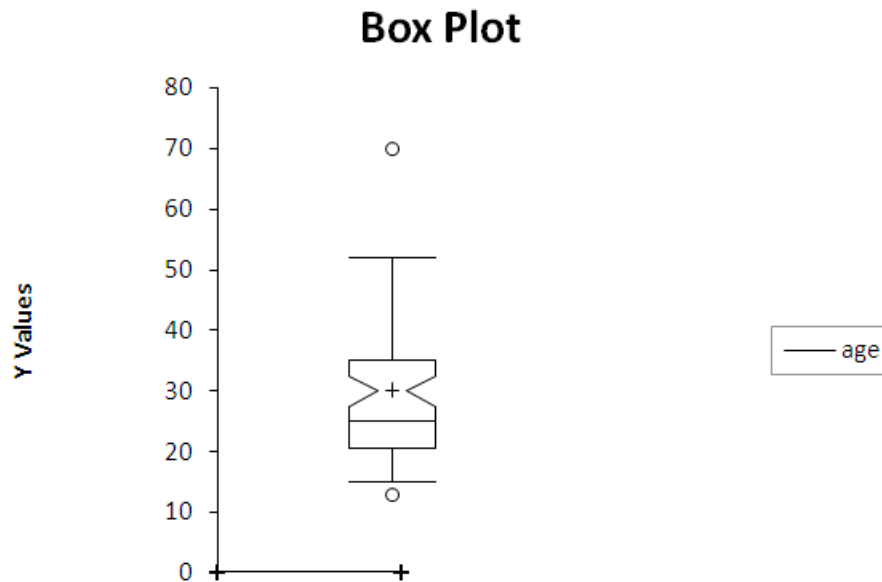
Median (Q2) = 25

Upper quartile = 20

Inter quartile Range (IQR) =  $Q3 - Q1$   
 $= 35 - 20 = 15$

XLMiner for Excel for Windows is the only comprehensive data mining add-in for Excel, with neural nets, classification and regression trees, logistic regression, linear regression, Bayes classifier, K-nearest neighbors, discriminate analysis, association rules, clustering, principal components, and more.

The box-plot analysis for given data is shown here :



### Conclusion:-

XLMiner for Excel for Windows is comprehensive data mining add-in for Excel, providing easy way to perform box-plot (and many other) analysis.

# PRACTICAL 7

## PERFORM CLASSIFICATION OPERATION USING DECISION-TREE IN XLMinOR.

Classification trees are used to predict membership of cases or objects in the classes of a categorical dependent variable from their measurements on one or more predictor variables.

The Iris dataset was introduced by R. A. Fisher as an example for discriminant analysis. The data report four characteristics (sepal width, sepal length, petal width and petal length) of three species of Iris flower.

Species\_No: Flower species as a code

Species\_Name: Species name

Petal\_Width: Petal Width

Petal\_Length: Petal Length

Sepal\_Width: Sepal Width

Sepal\_Length: Sepal Length

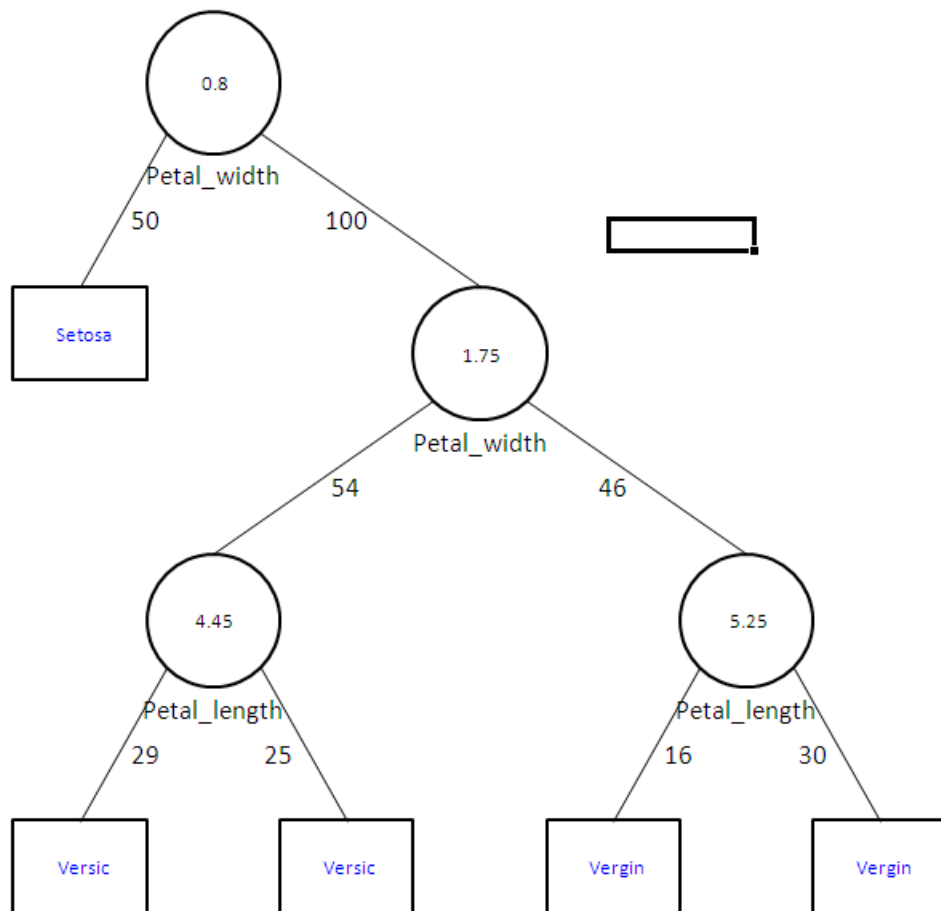
All measurements are lengths in cms.

The classification confusion matrix and decision-tree are shown here :

### Training Data scoring - Summary Report (Using Full Tree)

Classification Confusion Matrix			
	Predicted Class		
Actual Class	Setosa	Verginica	Versicolor
Setosa	50	0	0
Verginica	0	45	5
Versicolor	0	1	49

Error Report			
Class	# Cases	# Errors	% Error
Setosa	50	0	0.00
Verginica	50	5	10.00
Versicolor	50	1	2.00
Overall	150	6	4.00



### Conclusion:-

The decision process used by our *classification tree* provides an efficient method for sorting a pile of data, and more generally, can be applied to a wide variety of classification problems.

# PRACTICAL 8

## PERFORM K-MEANS CLUSTERING OPERATION USING XLMINER.

### ❖ GIVEN DATA DESCRIPTION:

Wine dataset contains properties of wine captured from three different wineries in the same region. There are 13 variables describing various properties of wine and 3 classes. This dataset can be used for classification with Type as a output variable OR can be used to perform clustering to without using Type variable to see the accuracy of prediction.

Type	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols
A	14.23	1.71	2.43	15.6	127	2.8
A	13.2	1.78	2.14	11.2	100	2.65
A	13.16	2.36	2.67	18.6	101	2.8
A	14.37	1.95	2.5	16.8	113	3.85
A	13.24	2.59	2.87	21	118	2.8
A	14.2	1.76	2.45	15.2	112	3.27
A	14.39	1.87	2.45	14.6	96	2.5
A	14.06	2.15	2.61	17.6	121	2.6
A	14.83	1.64	2.17	14	97	2.8
A	13.86	1.35	2.27	16	98	2.98
A	14.1	2.16	2.3	18	105	2.95
A	14.12	1.48	2.32	16.8	95	2.2
A	13.75	1.73	2.41	16	89	2.6

K-Means Training starts with a single cluster with its center as the mean of the data. This cluster is split into two and the means of the new clusters are iteratively trained. These two clusters are again split and the process continues until the specified number of clusters is obtained. If the specified number of clusters is not a power of two, then the nearest power of two above the number specified is chosen and then the least important clusters are removed and the remaining clusters are again iteratively trained to get the final clusters.

Parameters/Options	
# Clusters	8
# Starts	5
Seed	12345
# Iterations	10
Show data summary	Yes
Show distance from each cluster	Yes

Random Starts Summary					
Serial No.	Sum Of Square Distances	Starting Cluster Centres			A
		Alcohol	Malic_Acid	Ash	
1	84.463743	0.879519	2.959375	1.472111	
		2.520267	1.472275	0.803312	
		2.481881	0.554072	0.075788	
		3.290542	2.327782	0.79264	
		2.167833	1.155089	1.041406	
		1.049995	4.291897	0.453133	
		3.436317	2.190963	0.301898	
2	73.958235	0.245857	4.417598	1.231905	
		1.96848	2.108809	0.909176	
		0.462722	0.665257	0.726724	
		3.329972	1.945893	0.341162	
		2.451961	2.385537	0.704639	
		0.149718	2.58289	0.421573	
		2.366143	2.060475	1.799919	
3	66.019087	2.771923	2.972192	0.788702	
		1.01822	4.827438	1.760312	
		0.269863	1.560142	0.101184	
		2.208423	3.16167	1.024571	
		2.359185	2.97667	1.280985	
		2.431086	0.066402	0.628679	
		0.28262	0.051423	0.223256	
4	73.958235	0.467476	3.584019	1.811846	
		0.779321	4.961478	0.038807	
		0.15714	4.43968	0.305094	
		2.934745	3.206762	0.579656	
		3.100351	1.652179	1.129978	
		3.636946	0.306068	1.63864	
		3.255751	4.292823	1.228024	

Best Start-&gt;

XLMiner calculates the sum of square distances and decides the Best start. It then generates the further outputs taking the Best start as the starting point.

Cluster centers					
Cluster	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium
Cluster-1	13.24	2.59	2.87	21	118
Cluster-2	12.37	0.94	1.36	10.6	88
Cluster-3	13.2	1.78	2.14	11.2	100
Cluster-4	13.64	3.1	2.56	15.2	116
Cluster-5	14.37	1.95	2.5	16.8	113
Cluster-6	0	0	0	0	0
Cluster-7	14.23	1.71	2.43	15.6	127
Cluster-8	13.16	2.36	2.67	18.6	101

Distance between cluster centers	Cluster-1	Cluster-2	Cluster-3	Cluster-4	Cluster-5
Cluster-1	0	217.377339	315.6687997	110.1778358	745.0394427
Cluster-2	217.377339	0	530.1516285	326.279482	960.3758627
Cluster-3	315.6687997	530.1516285	0	205.6693181	430.251561
Cluster-4	110.1778358	326.279482	205.6693181	0	635.017745
Cluster-5	745.0394427	960.3758627	430.251561	635.017745	0
Cluster-6	744.8671775	527.659765	1054.919723	853.2125936	1484.511059
Cluster-7	330.1745001	546.4472463	31.26501239	220.2827469	415.2453999
Cluster-8	450.331053	665.2004256	135.224693	340.3514669	295.2627107

In the output for "cluster centers" above, the upper box shows the variable values at the cluster centers. The lower box shows the distance between those cluster centers.

Data summary		
Cluster	#Obs	Average distance in cluster
Cluster-1	37	56.477714
Cluster-2	78	83.517138
Cluster-3	14	32.01167
Cluster-4	16	36.829001
Cluster-5	7	67.178033
Cluster-6	0	0
Cluster-7	7	30.431411
Cluster-8	19	75.089048
Overall	178	66.019087

Data summary shows how many records (observations) there are in each cluster, and the average distance from cluster members to the center of the cluster.

XLMiner : k-Means Clustering - Predicted Clusters						
Row Id.	Cluster id	Dist clust-1	Dist clust-2	Dist clust-3	Dist clust-4	Dist clust-5
1	7	330.1745	546.447246	31.265012	220.282747	415.2454
2	3	315.6688	530.151628	0	205.669318	430.251561
3	8	450.331053	665.200426	135.224693	340.351467	295.262711
4	5	745.039443	960.375863	430.251561	635.017745	0
5	1	0	217.377339	315.6688	110.177836	745.039443
6	5	715.054595	930.343212	400.210294	605.017927	30.091969
7	8	555.475848	770.069236	240.063413	445.452513	190.79623
8	8	560.020404	775.748947	245.984974	450.035966	185.204752
9	3	310.797076	525.114555	6.786383	200.914294	435.313485
10	3	310.702995	525.164592	7.832918	200.829528	435.261604
11	5	775.117512	990.190778	460.082389	665.098649	31.159418
12	8	545.504306	760.070302	230.127051	435.514	200.840403
13	8	585.742194	800.034626	270.271069	475.77008	161.816385
14	8	415.996072	630.036891	100.441396	306.05577	330.788196
15	5	812.216512	1027.123528	497.020156	702.15434	68.076464

The final part of the output, above, shows the cluster to which each record belongs and its distance to each of the clusters. Note that, for record 5, the distance to cluster 1 is the minimum distance, so record 5 is assigned to cluster 1.



# EMPLOYEE & PROJECT MANAGEMENT SYSTEM

## DATA CUBE AND OLAP OPERATIONS USING SNOWFLAKE SCHEMA

### ❖ INTRODUCTION :

In any IT organization (like Tata Consultancy Services, Infosys etc.) there is a need to manage their employees according to the project requirements. There are many unrecognized patterns or some knowledge that if discovered then it can become a very fruitful to the organization.

The aim of this small project or module is to make a **data warehouse**, and by applying some **OLAP operations** an organization can find an interesting knowledge as well as patterns for Employee-Project assignment management.

For example, by these analyses an organization can find that how many employees will be needed in Banking Management Project in each branch, what will be the budget and duration for it. Because such kind of Banking Management Projects were done in the past of an organization's project history.

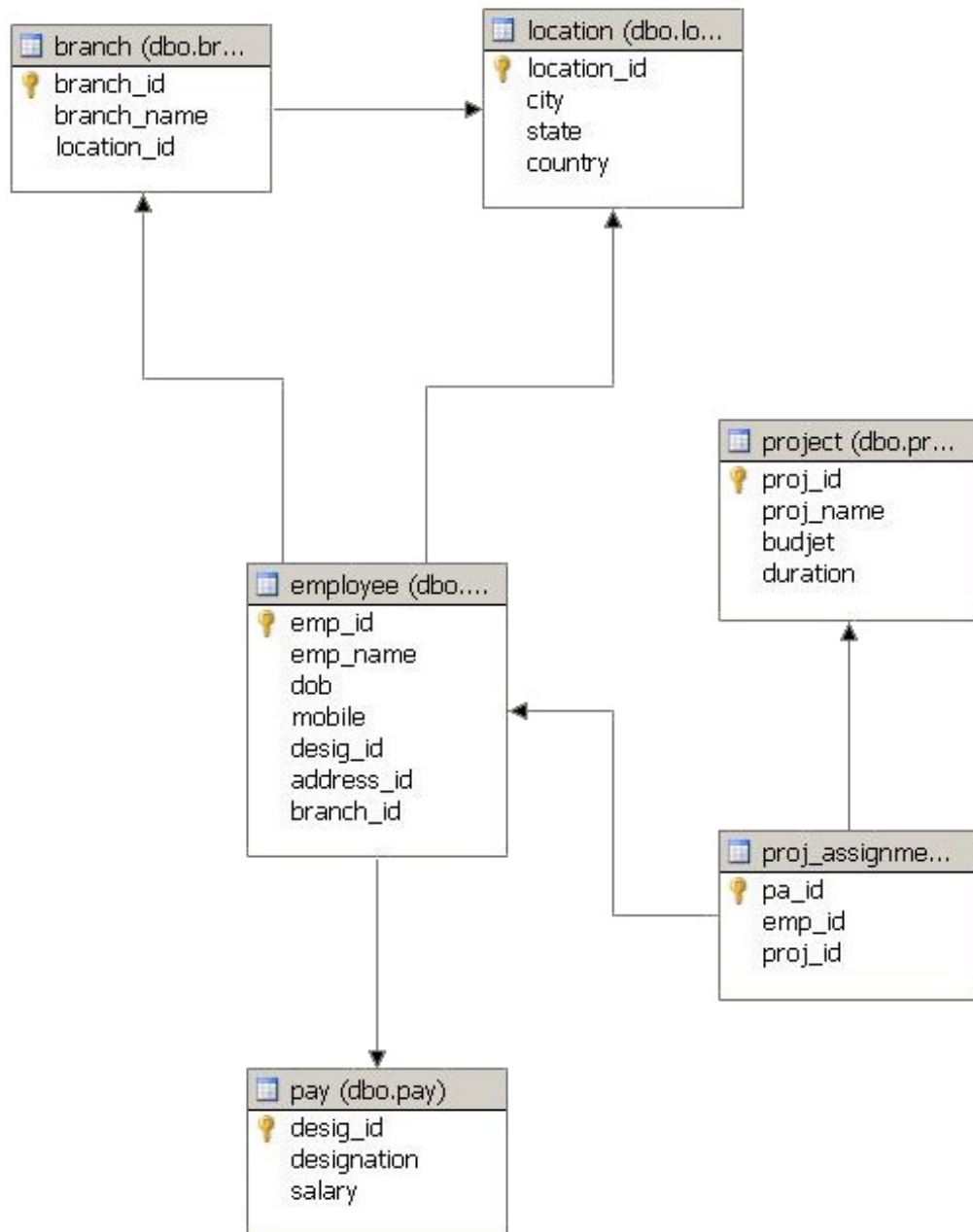
### ❖ METHODOLOGY & TOOLS USED:

For generating this data warehouse **Snowflake schema** is used here, and for finding some interesting knowledge **OLAP (On-line Analytical Processing) operations** are used. For these purpose two tools **SQL Server 2005** and **Business Intelligence Development Studio** are used.

A **snowflake schema** is a logical arrangement of tables in a multidimensional database such that the entity relationship diagram resembles a snowflake in shape. The snowflake schema is represented by centralized fact tables which are connected to multiple dimensions. In the snowflake schema, dimensions are normalized into multiple related tables.

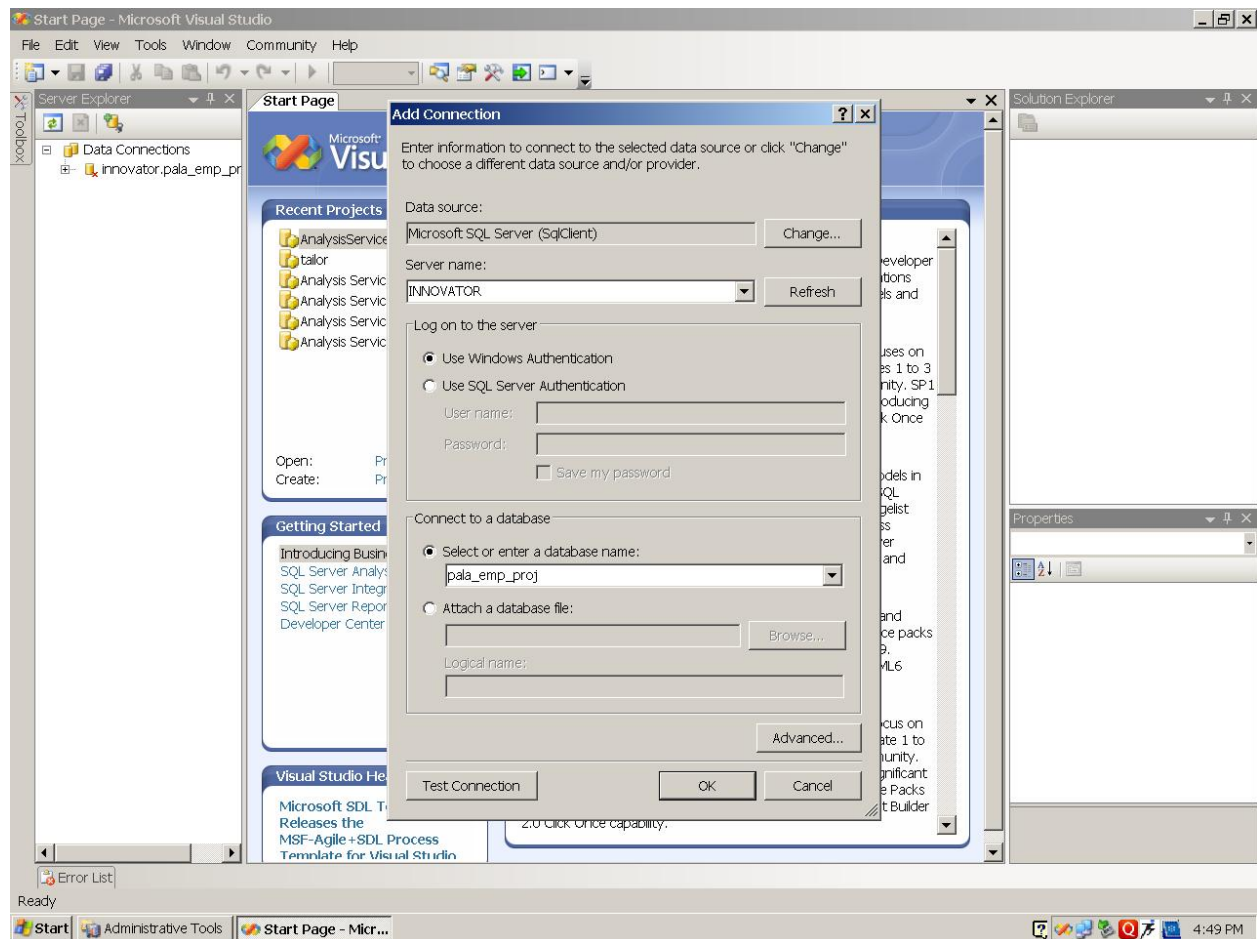
**OLAP** can be performed in data warehouse/marts using the **multidimensional data model**. Typical **OLAP operations** include *roll-up, drill-(down, across, through), slice & dice, pivot (rotate)*, as well as statistical operations such as ranking and computing moving averages and growth rates. OLAP operations can be implemented efficiently using the **Data Cube** structure.

**Business Intelligence Development Studio** is the primary environment that you will use to develop business solutions that include Analysis Services, Integration Services, and Reporting Services projects.

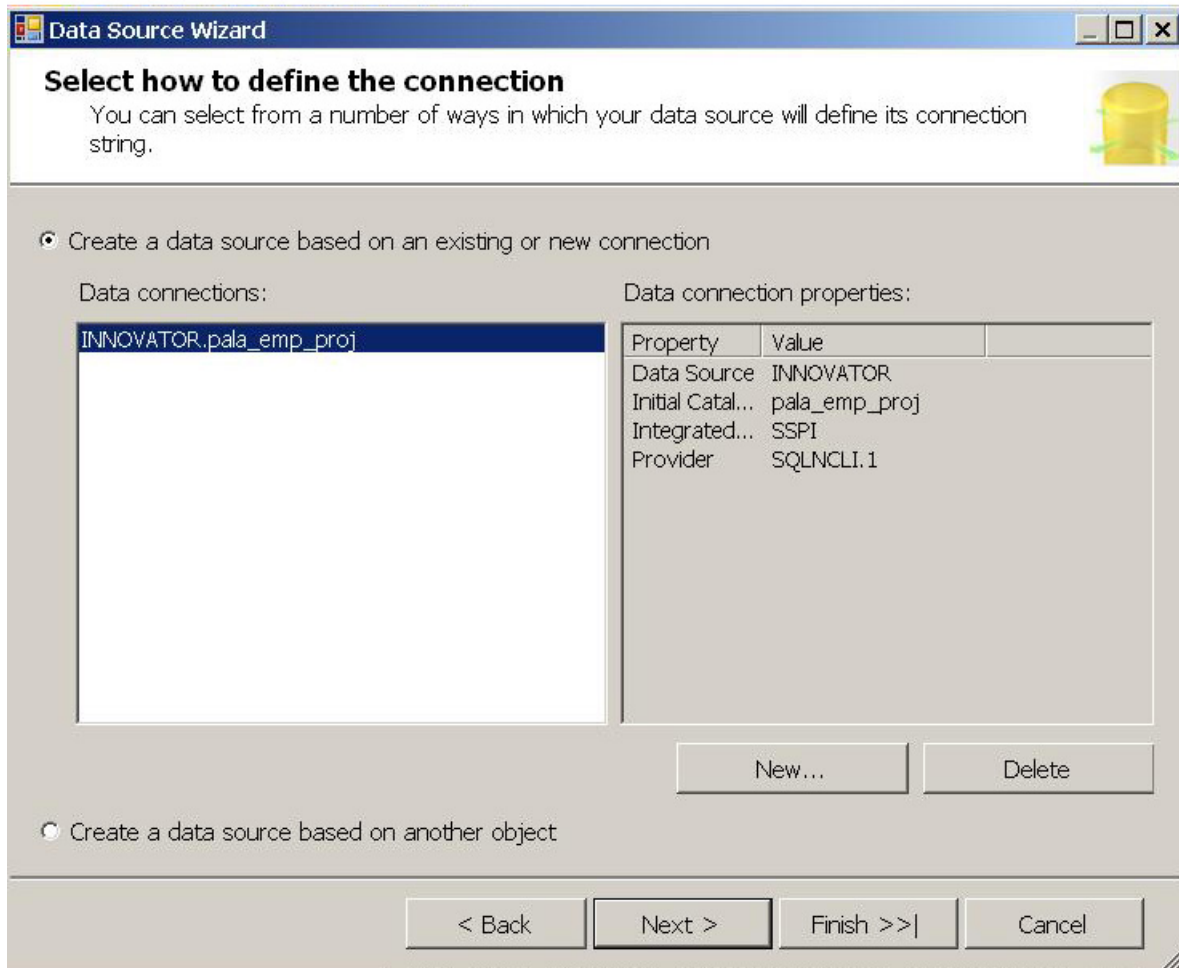
❖ **RELATIONSHIP BETWEEN THE TABLES USED :**

**❖ PROCEDURE:**

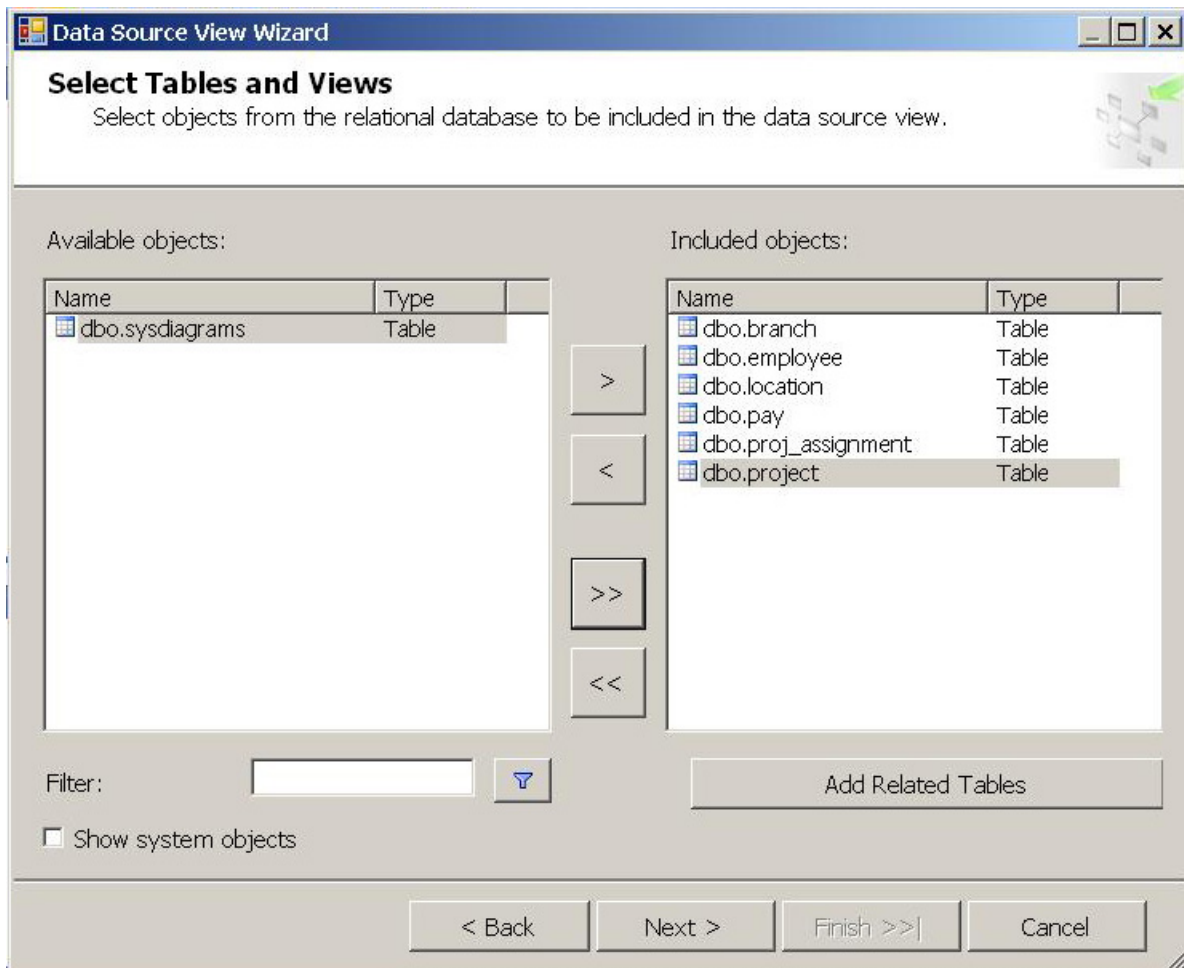
- ✓ Make the database according to the diagram using Microsoft SQL Server 2005.
- ✓ After making the database open Microsoft Business Intelligence Development Studio and follow the steps given below:
- ✓ Select a new project from file menu.
- ✓ Select Analysis Services Project and give proper name to the project and click on "OK" button
- ✓ After that click on the "Server Explorer" present on the left side of the screen to choose data connection and add a new connection with database that we have already made using Microsoft SQL Server Management studio.



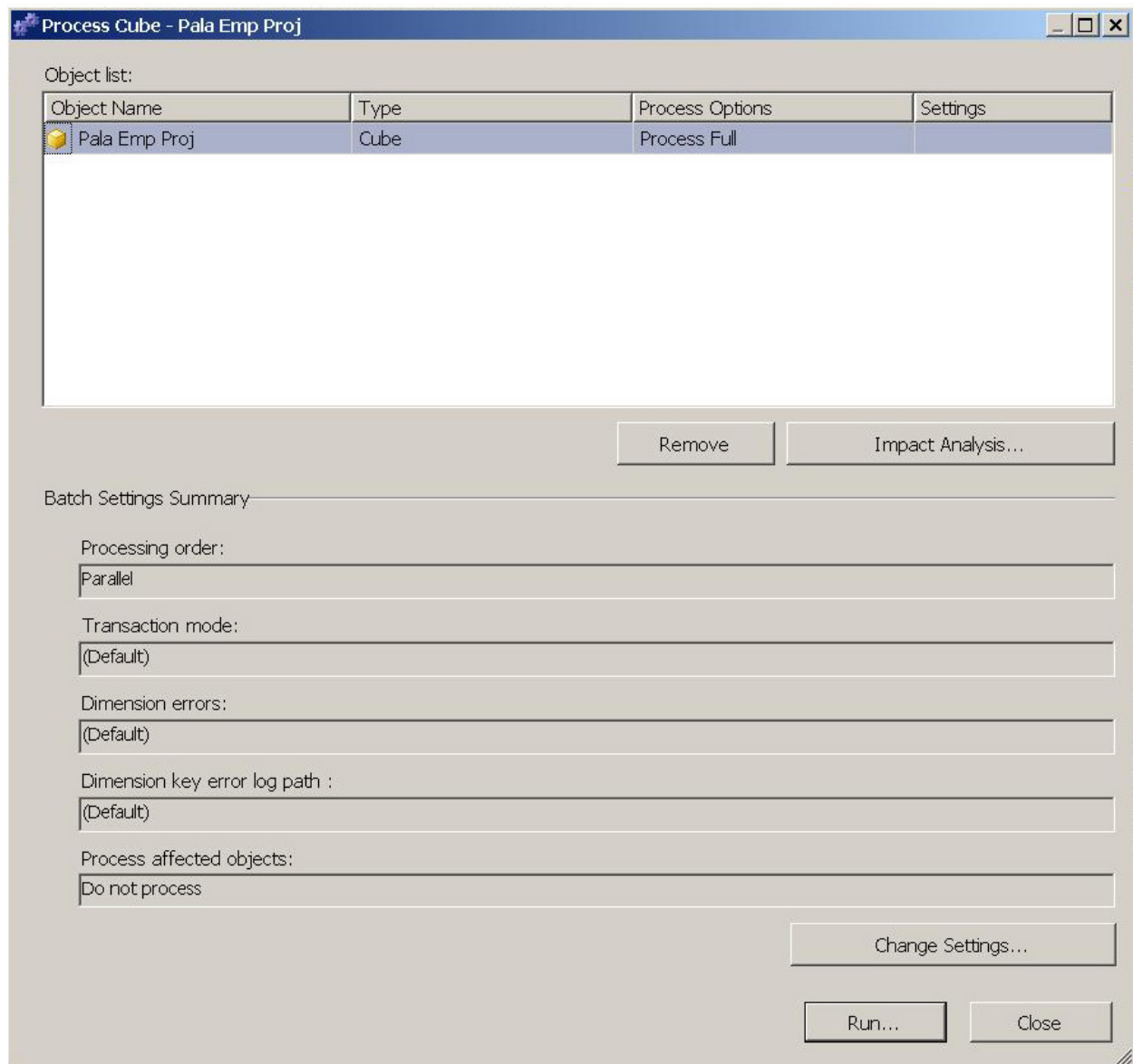
- ✓ First add connection then select new data source from Solution Explorer.



- ✓ Create view from the data source to get important Dimensions.



- ✓ After creating views create cube based on the views. Process the cube and the browse the cube. As shown in the figure the fact and dimension tables are automatically detected.



- ✓ Finally we can create a cube showing facts and dimensions and performing analysis on it.

**Pala Emp Proj.cube [Design]** Start Page

Tools: Cube Struct... Dimension Usage Calculations KPIs Actions Partitions Perspectives Translations Browser

Perspective: Pala Emp F Language: Default

Dimension	Hierarchy	Operator	Filter Expression
<Select dimension>			

Drop Filter Fields Here

	City						
	ahmedabad	banglore	baroda	berlin	captown	chennai	delhi
Proj Name	Proj Assignment Count	Proj Assignment Count	Proj Assignment Count	Proj Assignment Count	Proj Assignment Count	Proj Assignment Count	Proj Assignment Count
airline reservation	2	1		2	1		
banking mngt	3		2			2	
forensic	1	1		1	1		
hospital mngt	4	2	1			1	1
manufacturing_line	2	3			1	1	
medical store	2			1		2	
networking	2	1	1		1		1
online auction	3	2		1	1		
portal	2	2		1	1	1	
railway reservation	2	1					
transport	3	1	1	1		2	1
university	3		1		1	1	1
Grand Total	29	14	6	7	7	10	4

## CONCLUSION:

By creating data cube and performing OLAP operations for Employee-Project management, an organization can find some fruitful facts and using these facts an interesting knowledge and patterns can be generated.