

Class:MSC-I

SEM:I

Subject:DataWarehousing&DataMining(DWDM)

Paper:IV

Academic Year:2022-2023

RollNo:509

INDEX

S.NO	Date	Topic	Page. No	Sign
1		Creation of Dimensions and Fact tables.	2	
2		Create Data Source using SSAS (SQLServer Analysis Services.)	4	
3		Create Data Source View using SSAS (SQLServer Analysis Services.)	9	
4		Create cube using SSAS (SQL Server Analysis Services) and process the cube.	13	
5		View cube data in multidimensional Format.(Ex:Pivot Chart.)	18	
6		Working with measures in the cube.	21	
7		Creating an Excel Pivot Table Pivot Chart by using OLAPcube data.	23	
8		Firing Queries on Tables.	28	
9		Calculation & KPI	-	
10		Data pre-processing.	32	
11		Data discretization.	37	
12		Classification problems.	39	
13		Clustering Analysis.	44	
14		Association Rule Mining.	49	
15		Data visualization.	52	

PracticalNo1

Aim : Creation of Dimensions and Fact

tables.Solution

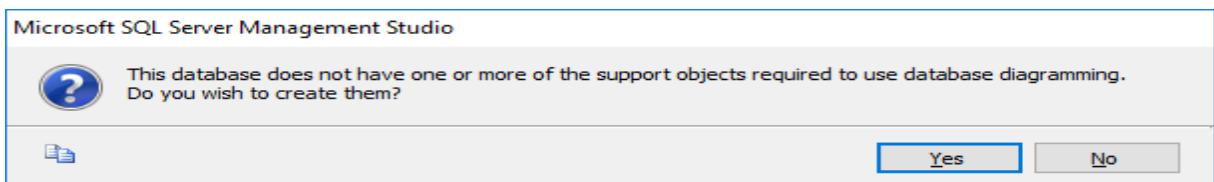
OpenApplication->MicrosoftSQLServer2008R2->SQLServerManagement Studio

1. SelectConnectTab->DatabaseEngine->SelectServerName(local)
2. Right ClicktheDatabase->NewDatabase
3. Types “SalesInformation” asthedatabasename,clickonOKtoclosethedialogboxandtocreate thedatabse.

CreateaDatabaseDiagrams

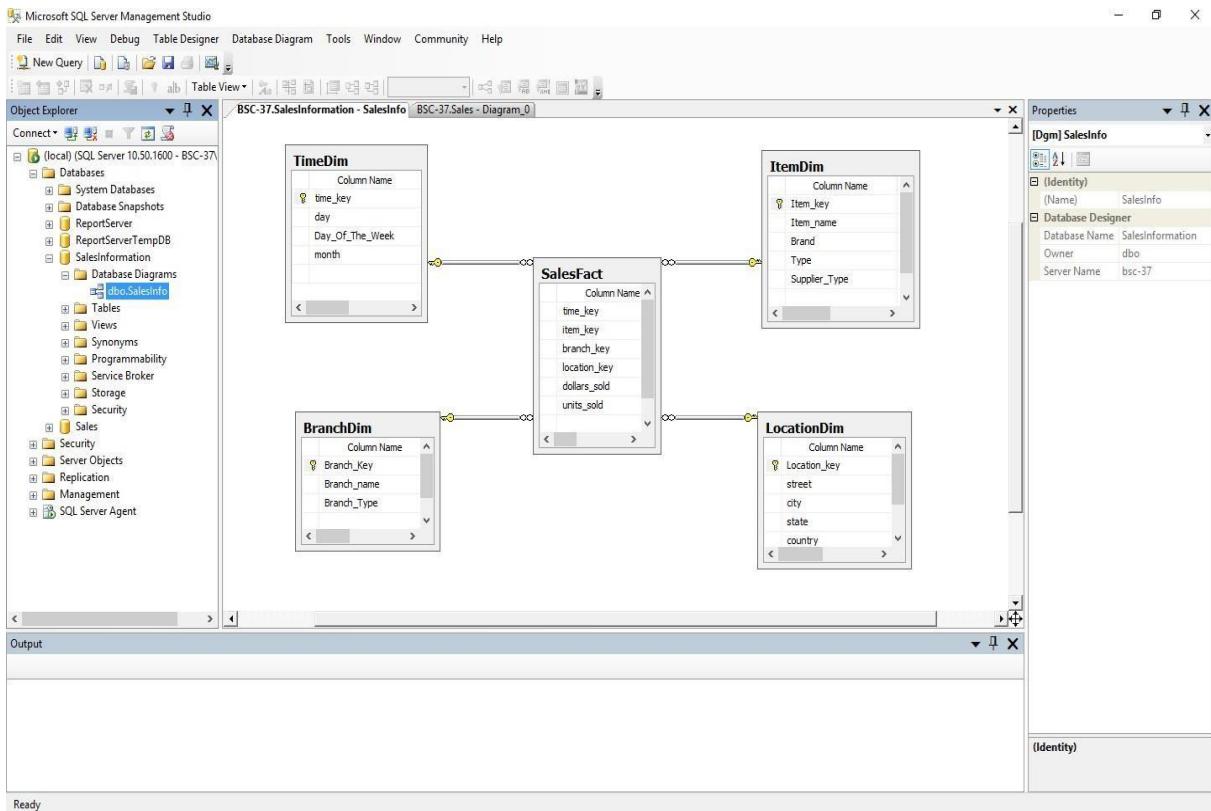
Expandthe “SalesInformation” databasefolder.

1. ClickonDatabaseDiagramstoexpandit



On clickofit,aboveDialogboxappears, clickonYestocloseit.

2. RightClickonDatabaseDiagrams- > NewDatabaseDiagrams
3. CreatefactandDimensionTables.(Rightclickonsurface,chooseNewTabletoadditonDatabase Diagrams.)



4. Establish relationship between fact and dimension tables.
5. Save Database Diagrams with name as "SalesInfo". (After saving Database Diagrams fact and dimension tables are automatically placed in Table tab.)

Practical No 2

Aim : Create Data Source using SSAS(SQL Server Analysis

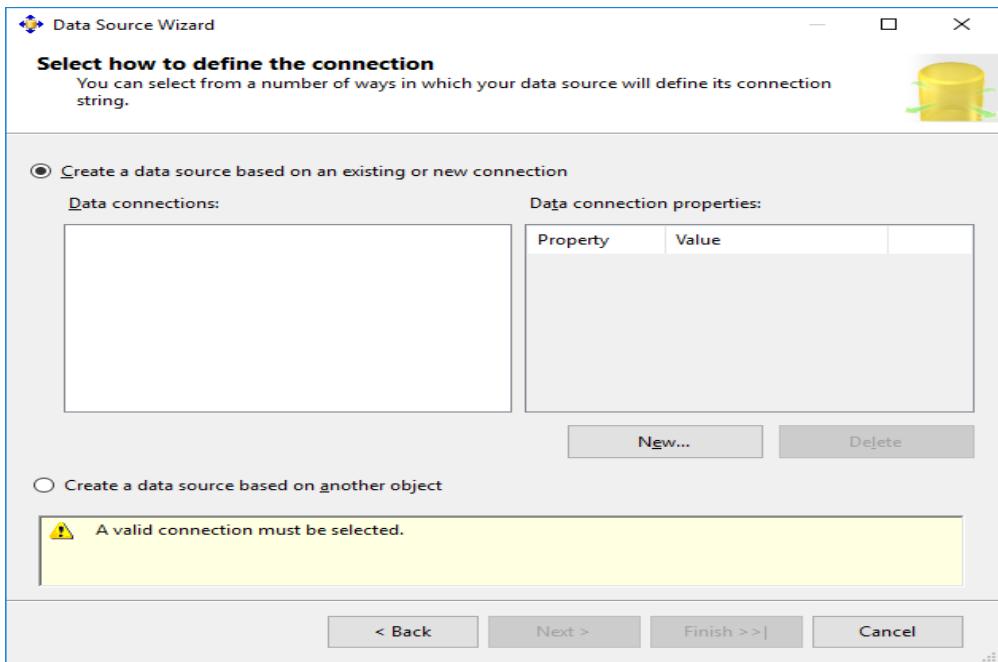
Services.)

Open Application -> Microsoft SQL Server 2008 R2 -> SQL Server Business
Intelligence Development Studio

1. Select File->New Project -> Choose Analysis Service Project -
> Name it as 'Star BI' and click on OK.
2. Right Click on Data Sources-> New Data Source

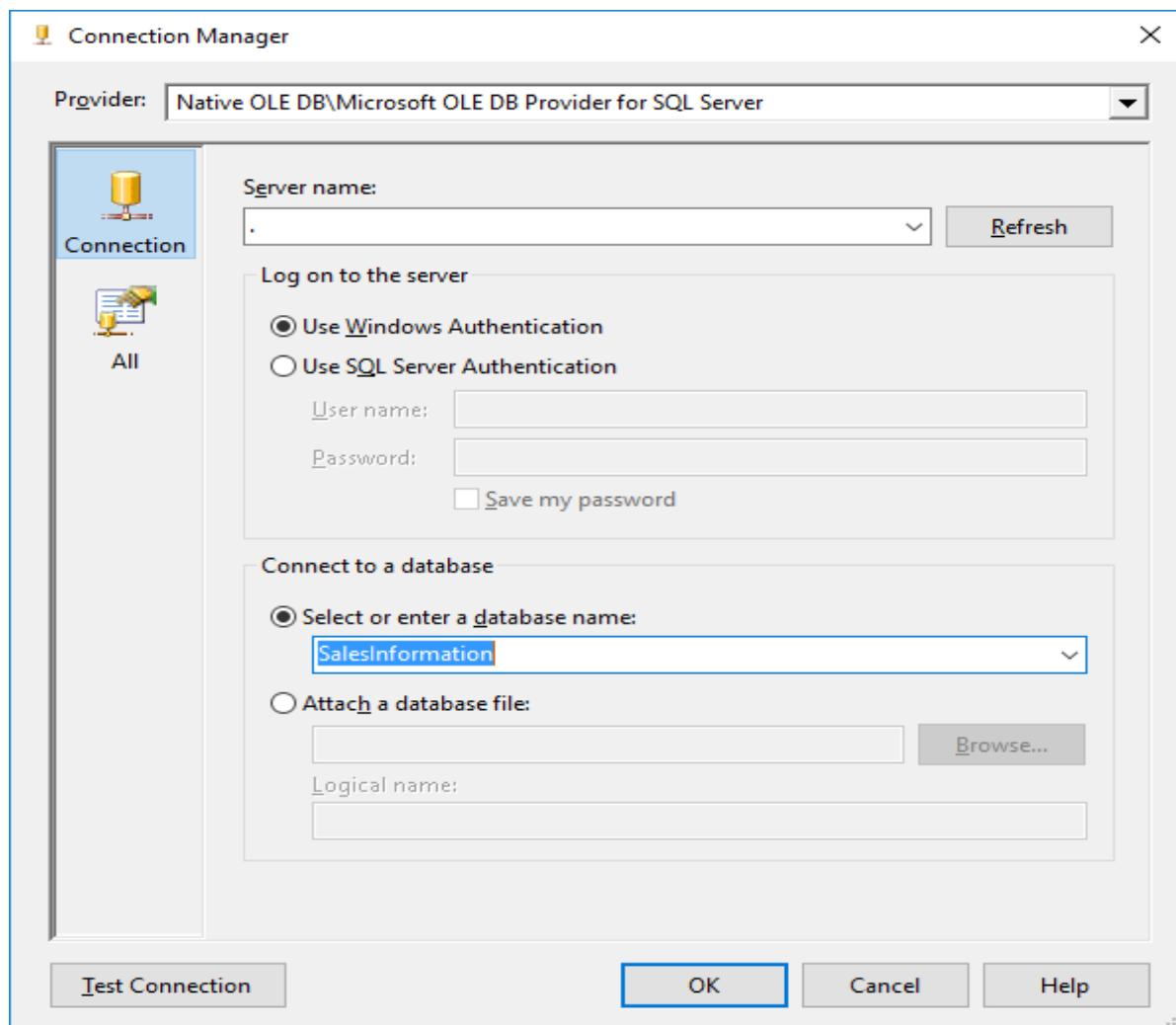


Click on Next.

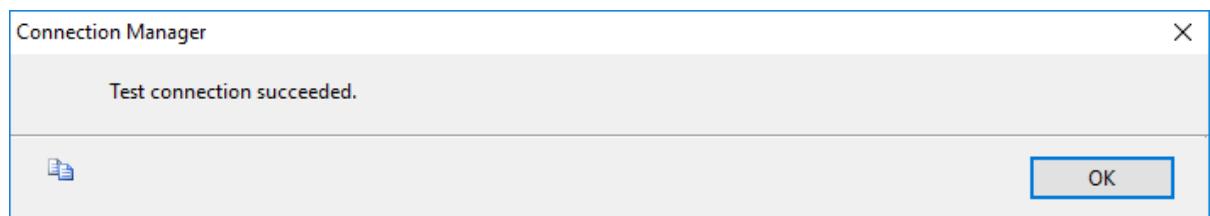


Click on New.

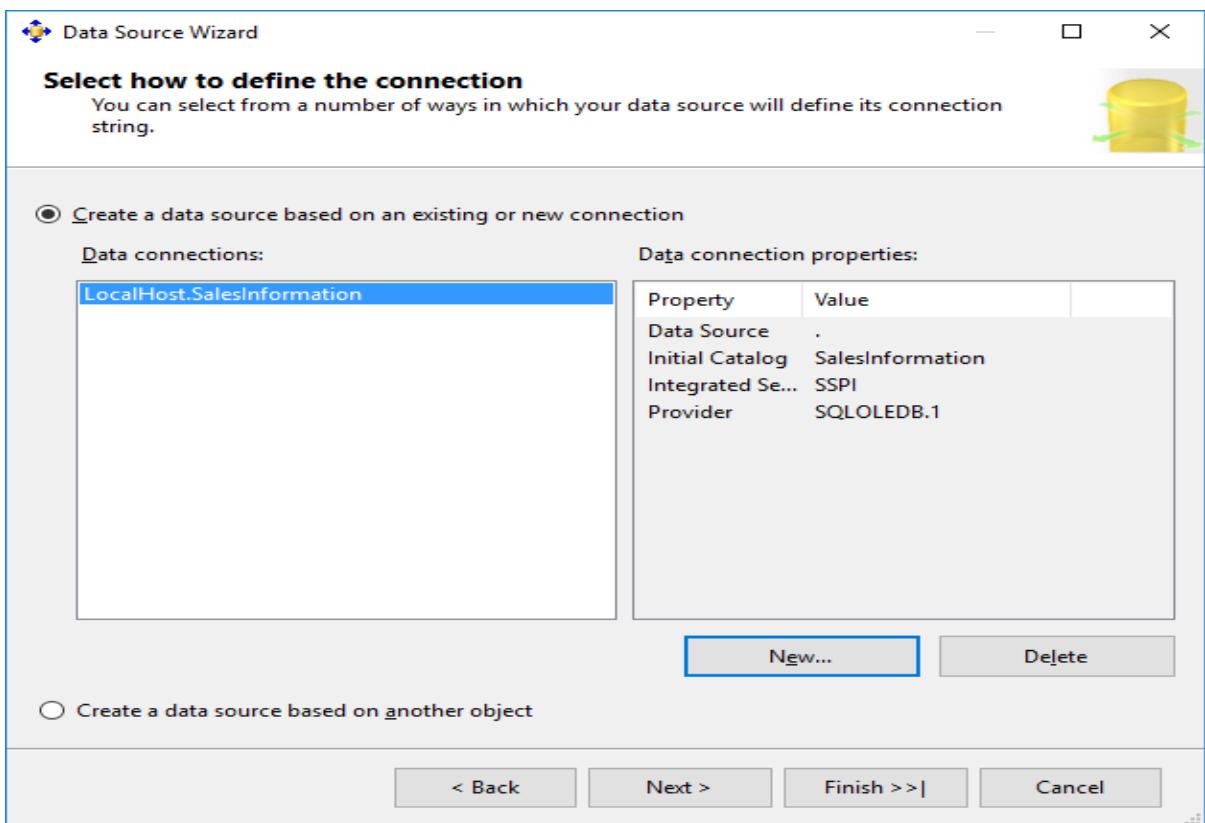
3. Choose Provider as “Microsoft OLEDB Provider for SQL Server”, Server Name as “~~SelectedServerName\SqlInforamtion\Default~~SQL Server Management studio).



4. Click on Test Connection.

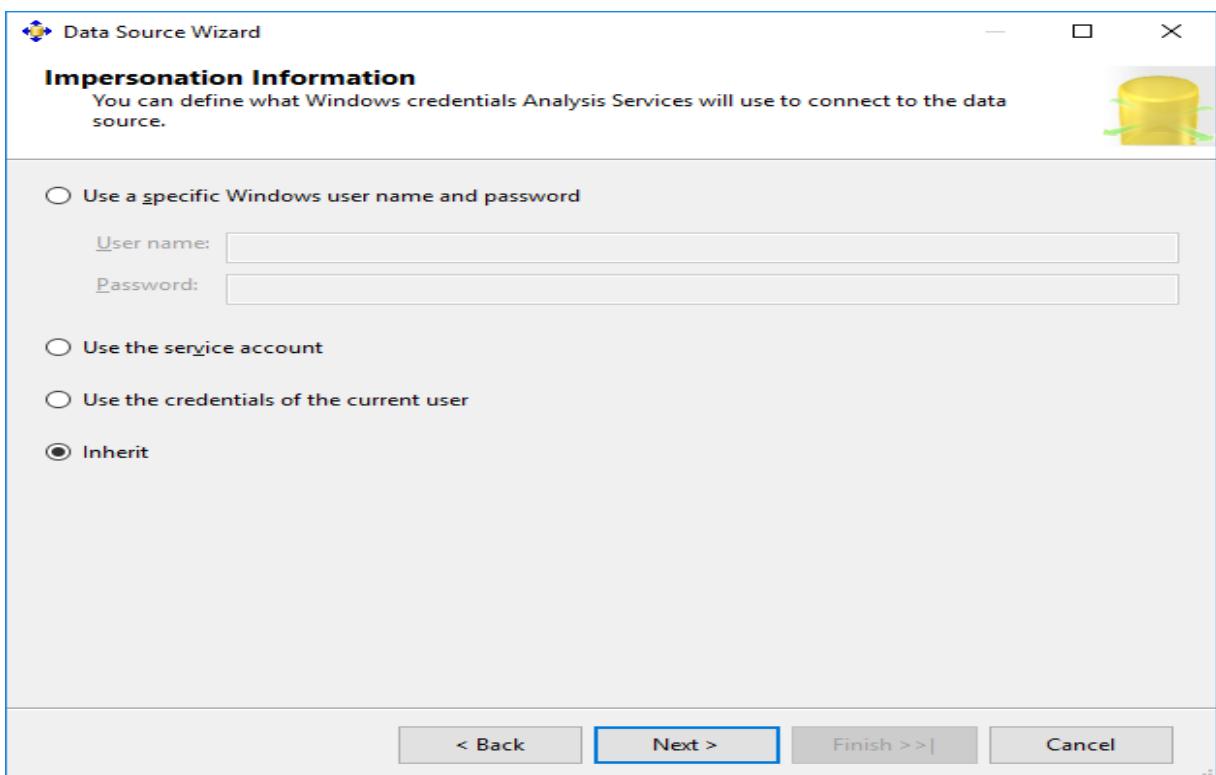


Click on OK.



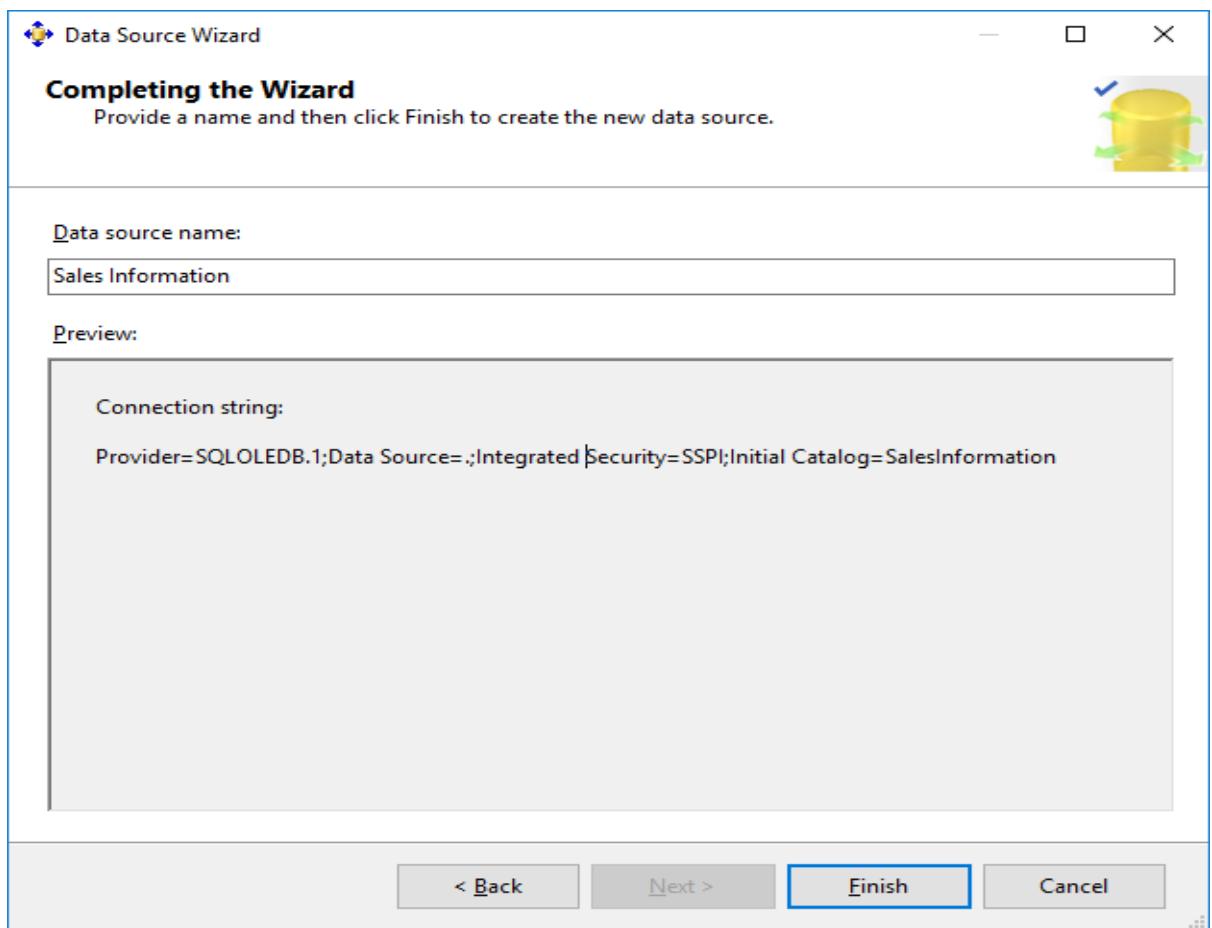
Click on Next

5. Choose "Inherit" option.



Click on Next.

6. Click on Finish.



NameDataSource as "SalesInformation".

PracticalNo3

Aim : Create Data Source View using SSAS(SQL Server Analysis Services.)

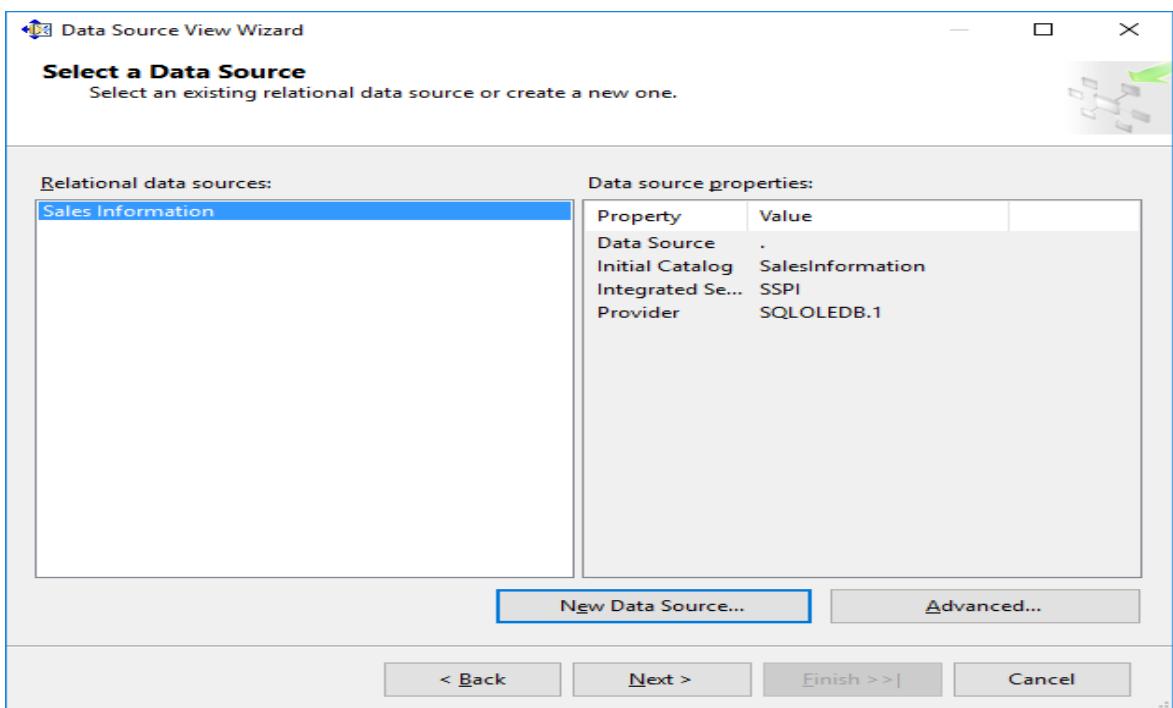
Services.)

1. RightclickonDataSourceView->NewDataSourceView

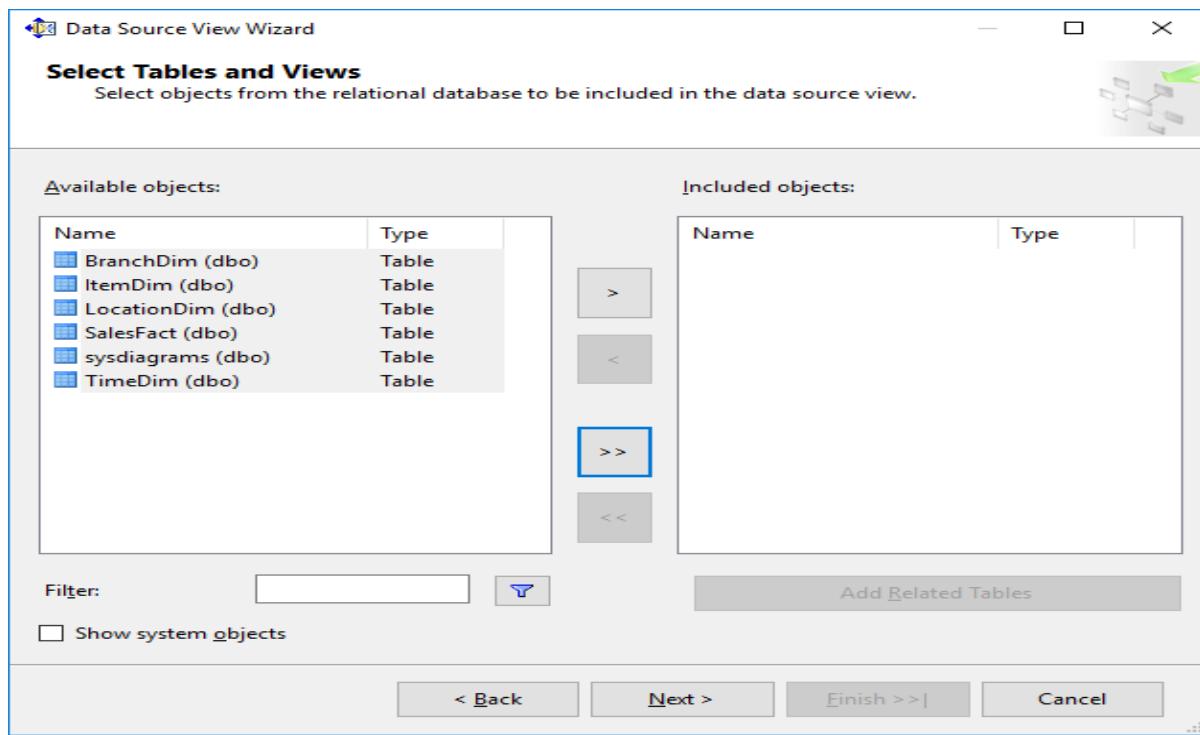


ClickonNext.

2. ClickonNext.



3. SelectTablesandViews.



 Data Source View Wizard

Select Tables and Views
Select objects from the relational database to be included in the data source view.

Available objects:

Name	Type
------	------

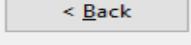
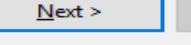
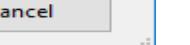
Included objects:

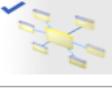
Name	Type
BranchDim (dbo)	Table
ItemDim (dbo)	Table
LocationDim (dbo)	Table
SalesFact (dbo)	Table
sysdiagrams (dbo)	Table
TimeDim (dbo)	Table

Filter: 

Show system objects

Add Related Tables

< Back  Next >  Finish >>  Cancel 

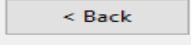
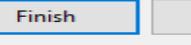
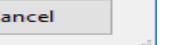
 Data Source View Wizard

Completing the Wizard
Provide a name, and then click Finish to create the new data source view.

Name:

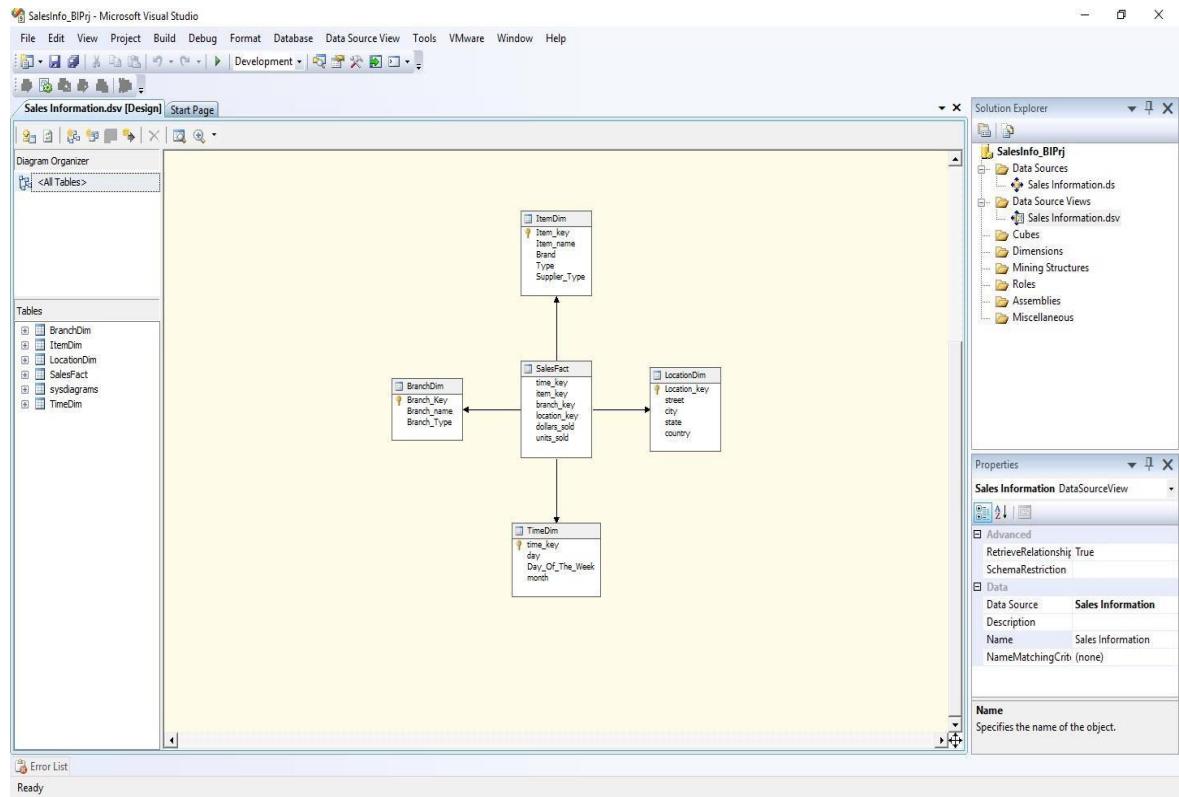
Preview:

- [-] Sales Information
 - [+] BranchDim (dbo)
 - [+] ItemDim (dbo)
 - [+] LocationDim (dbo)
 - [+] SalesFact (dbo)
 - [+] sysdiagrams (dbo)
 - [+] TimeDim (dbo)

< Back  Next >  Finish  Cancel 

Click on **Finish**.

4. Finally, we will get the Data Source View like:



Practical No 4

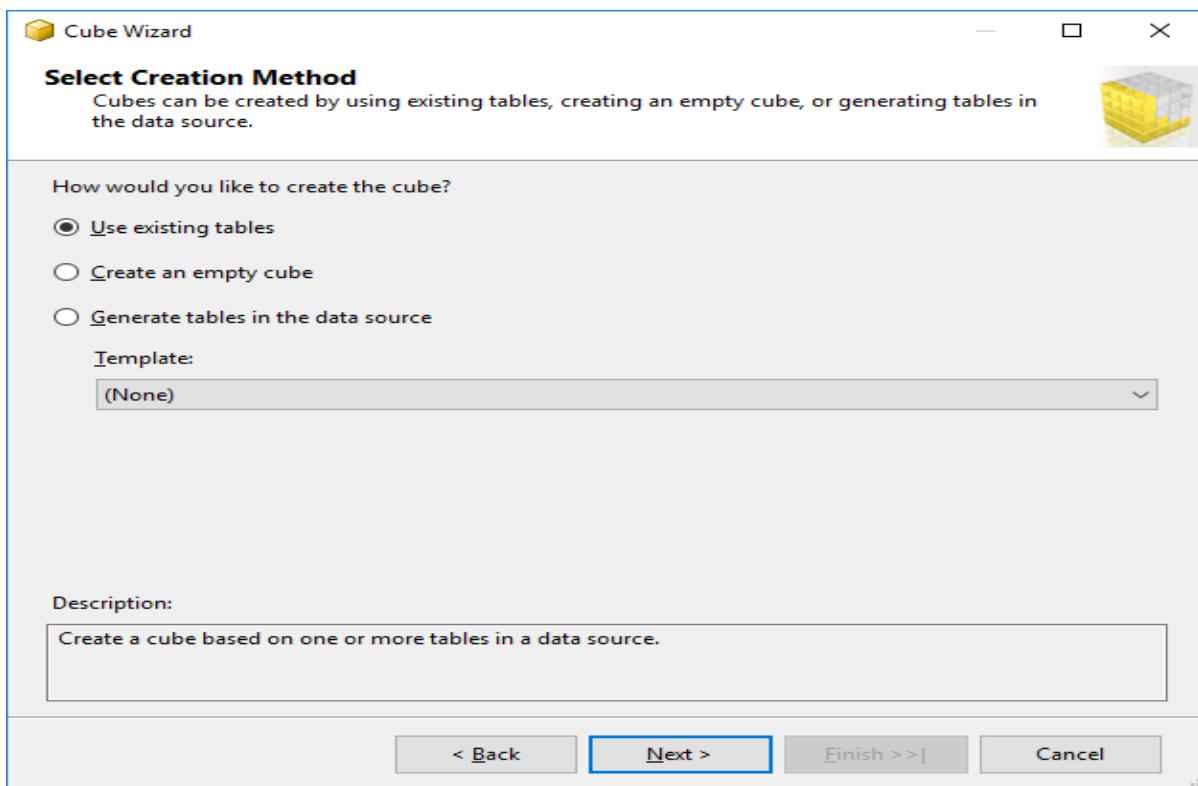
Aim: Create a cube using SSAS (SQL Server Analysis Services) and process the cube. Solution:

1. Right click on Cubes -> New Cube.

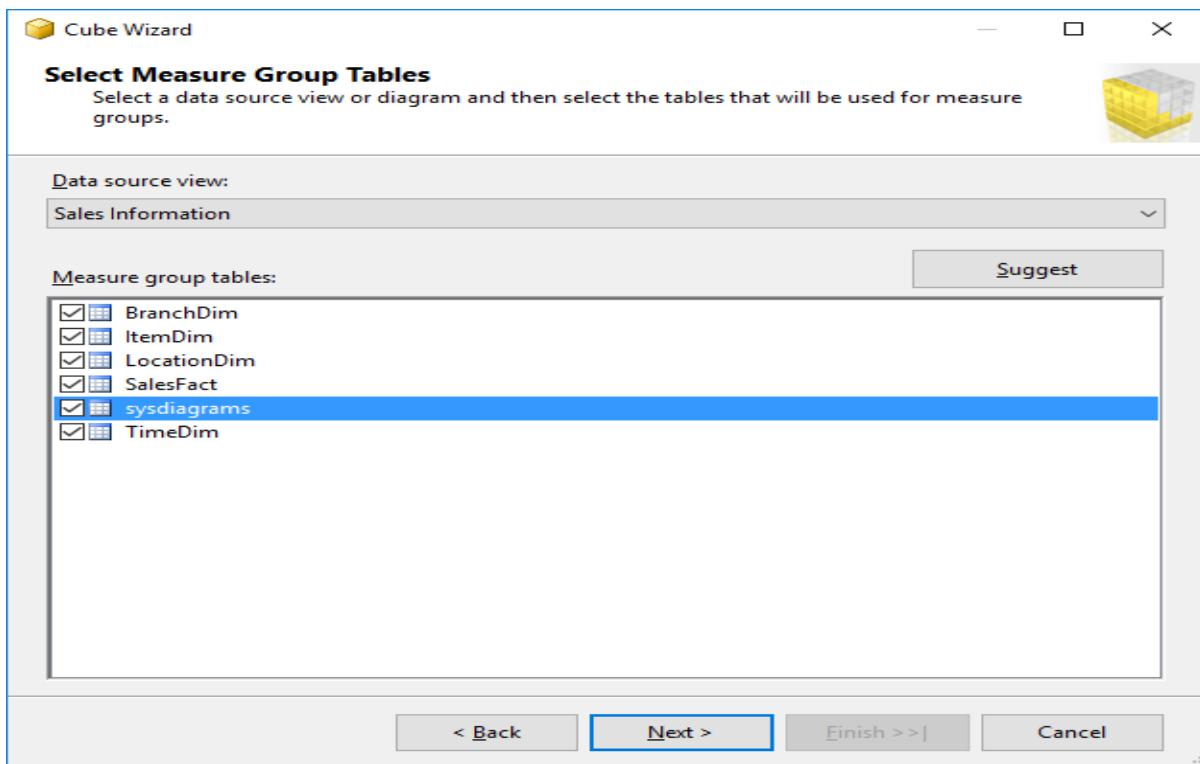


Click on Next.

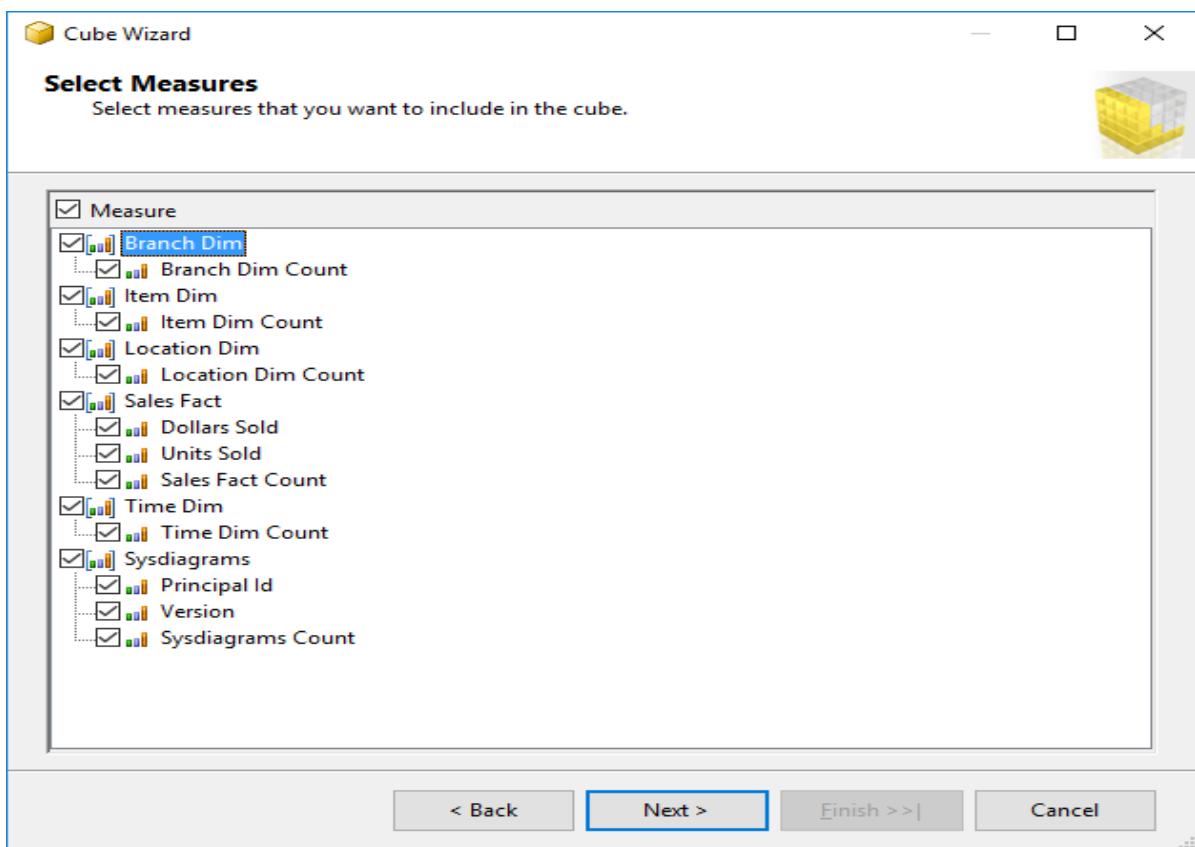
2. Select first option "Use existing tables". Click on Next.



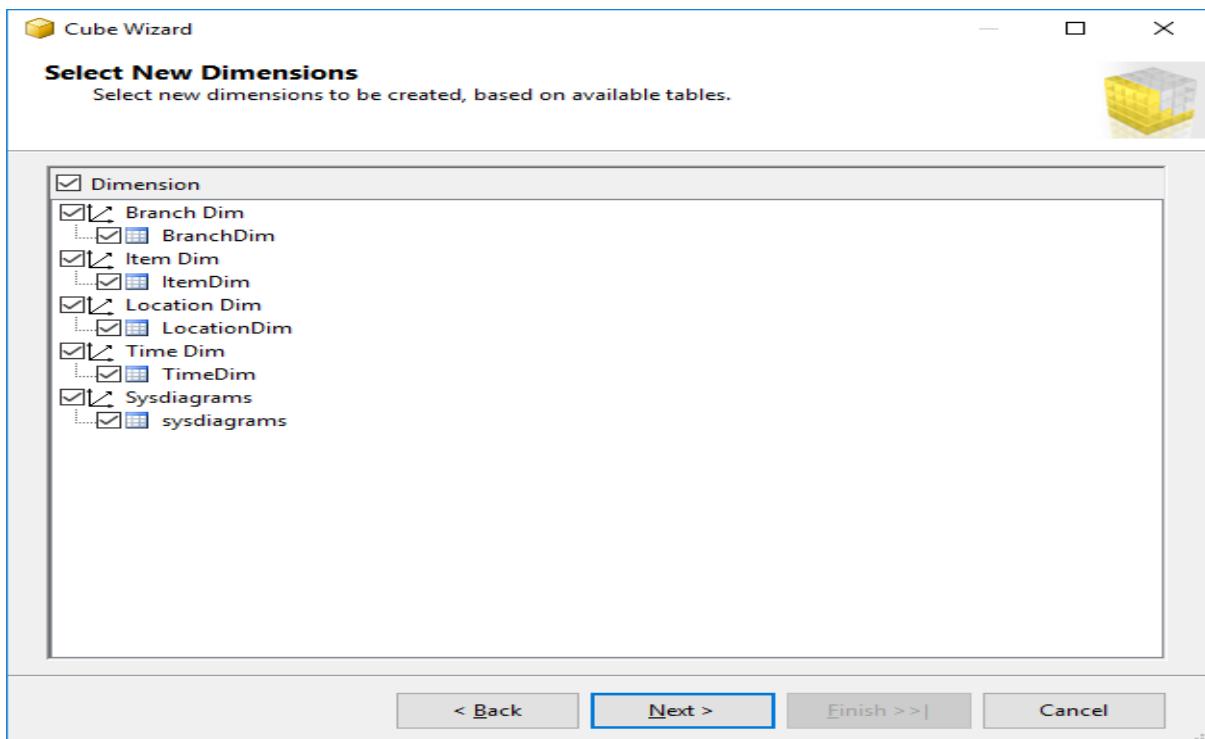
3. SelectDataSourceViewas“SalesInformation”andSelectallthetables.



Click on Next.

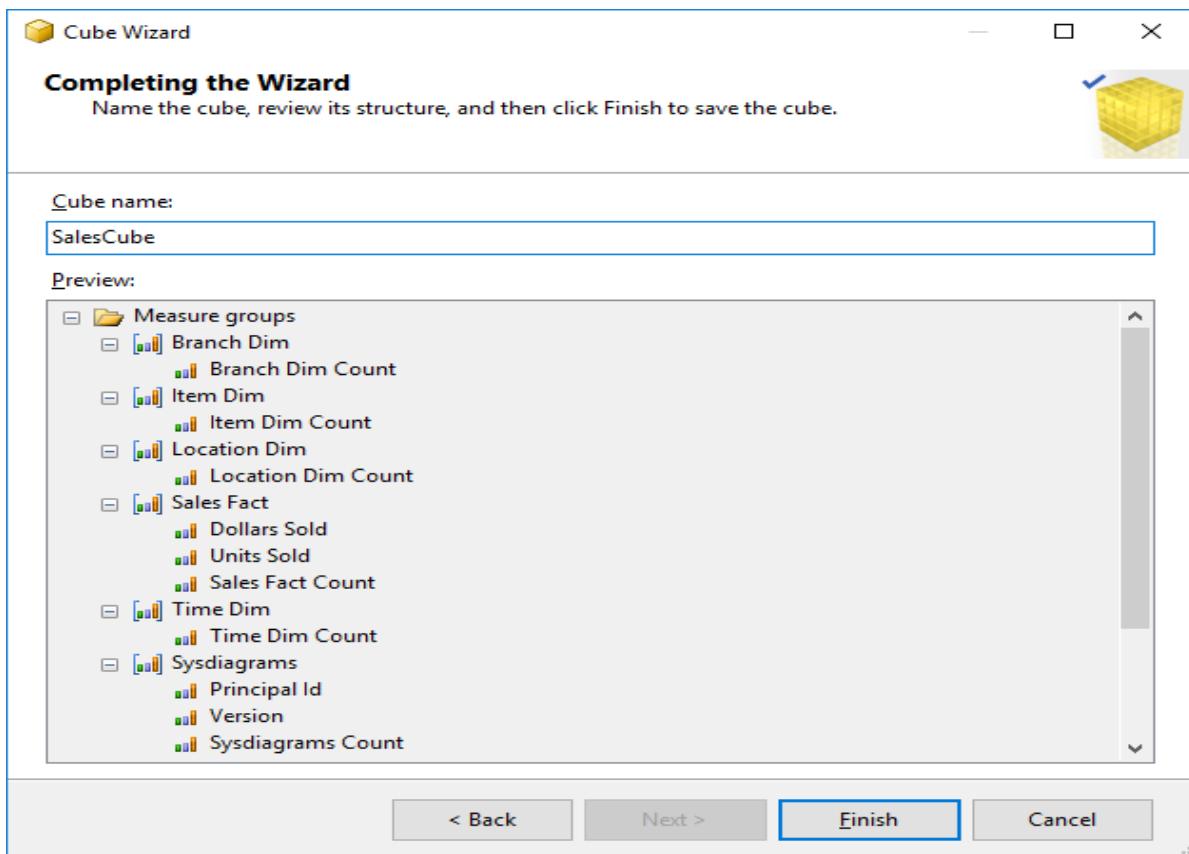


Click on Next.



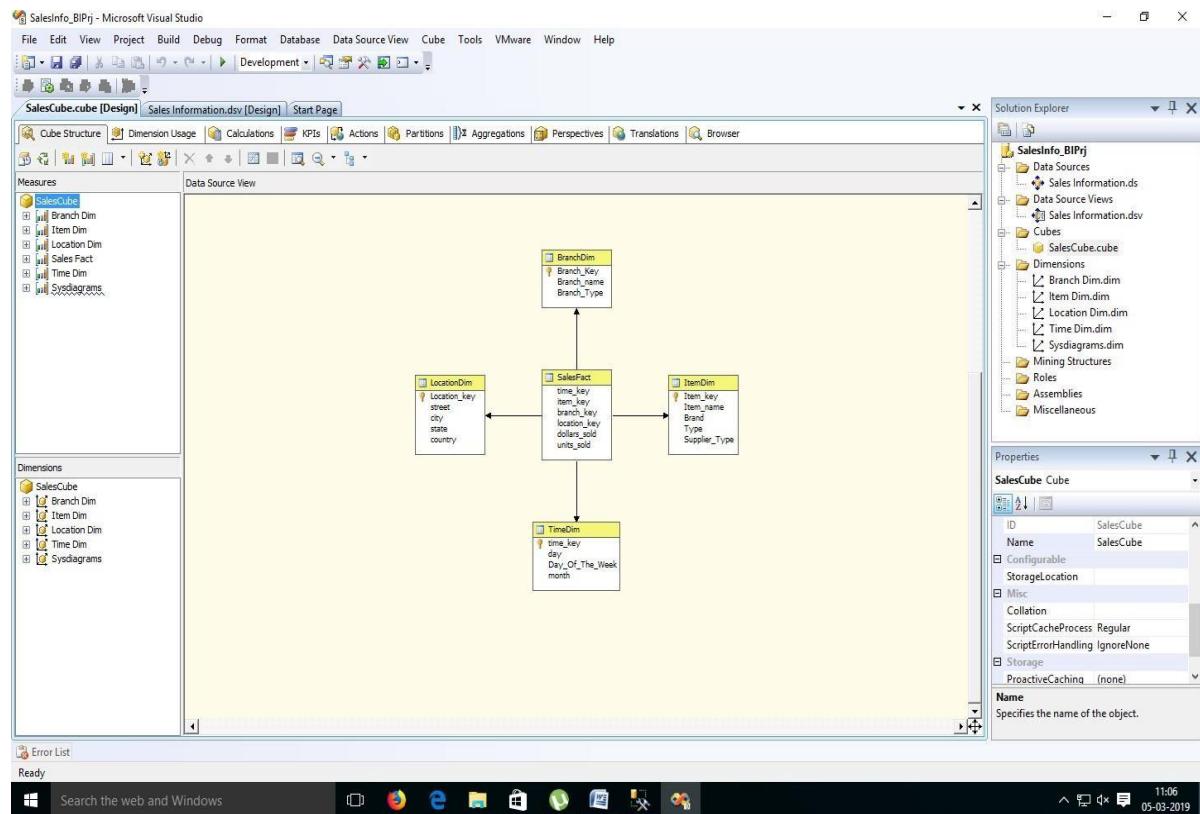
Click on Next.

4. Name Cube as "SalesCube".

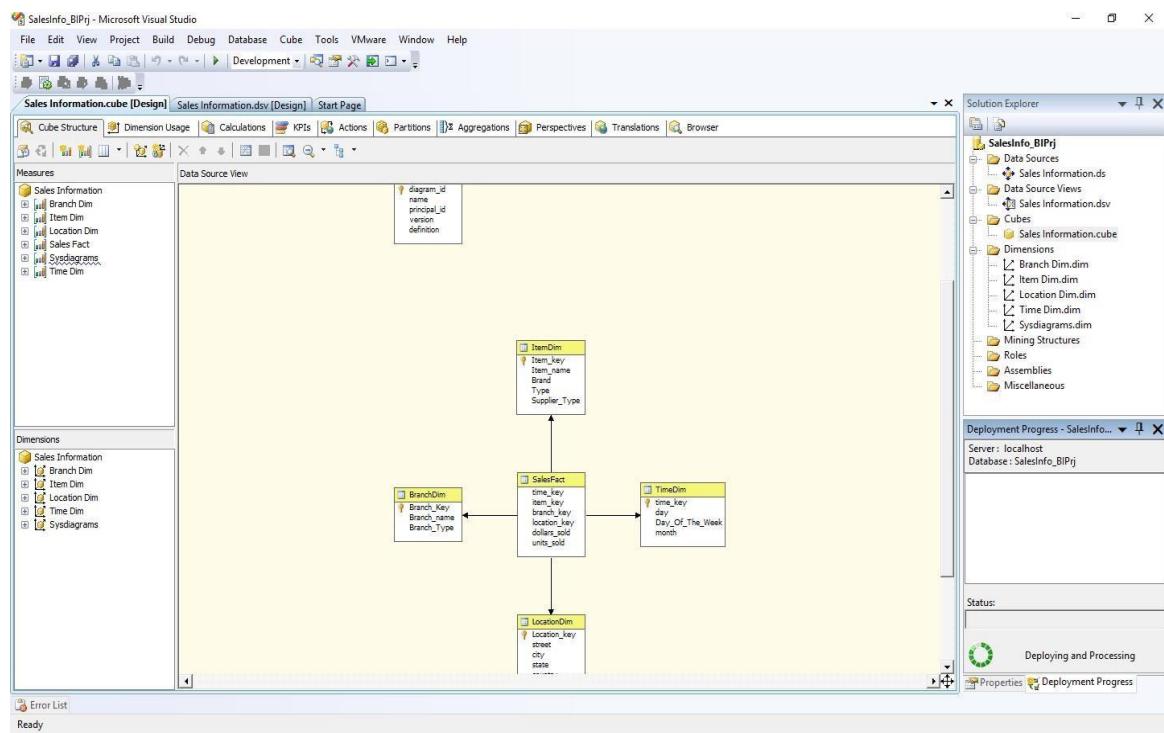


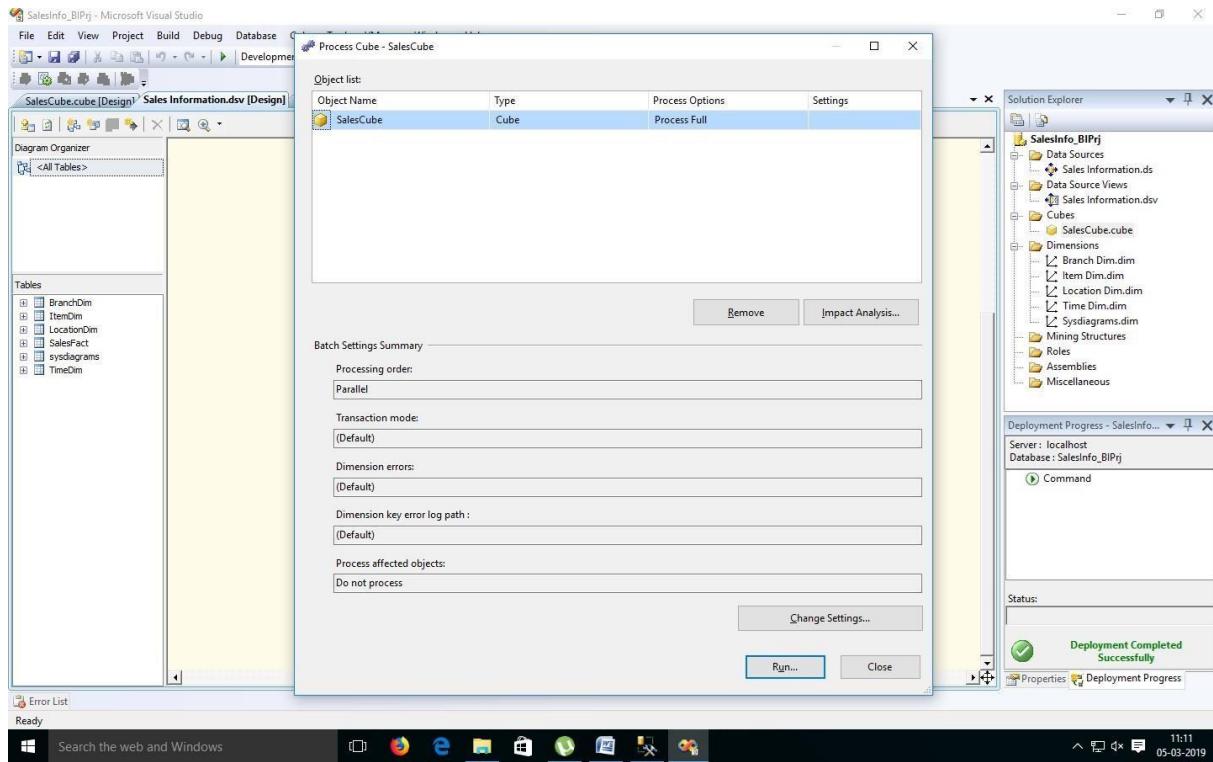
ClickonFinish.

5. Finally,wewillgettheCubeViewaswellDimensionsViewlike:

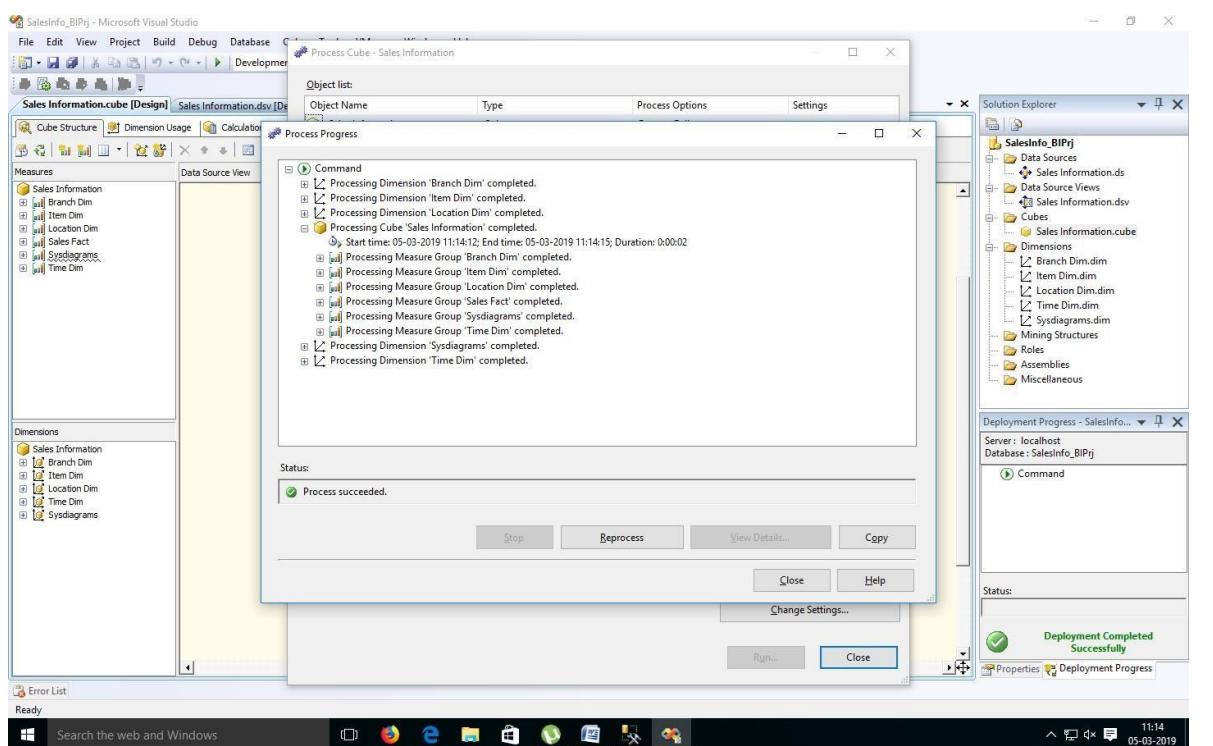


6. Finally,ProcesscubebyRight clickonSalesCube->Process.





7. Click on Run.

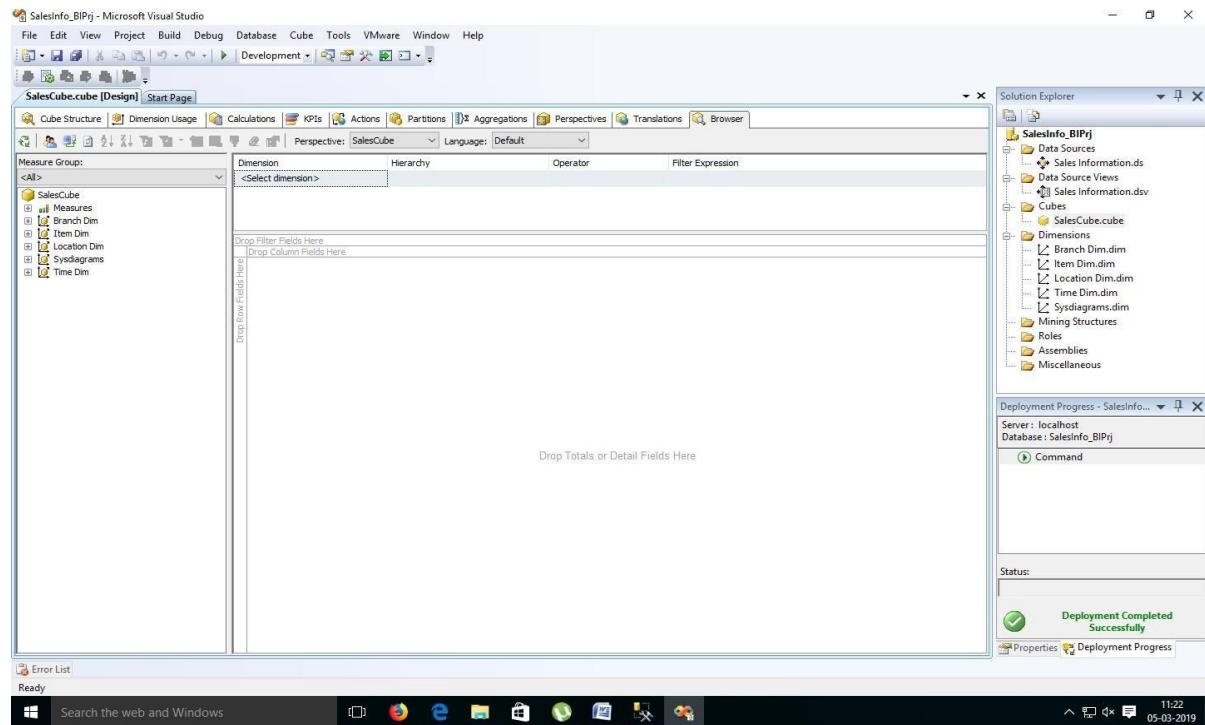


PracticalNo5

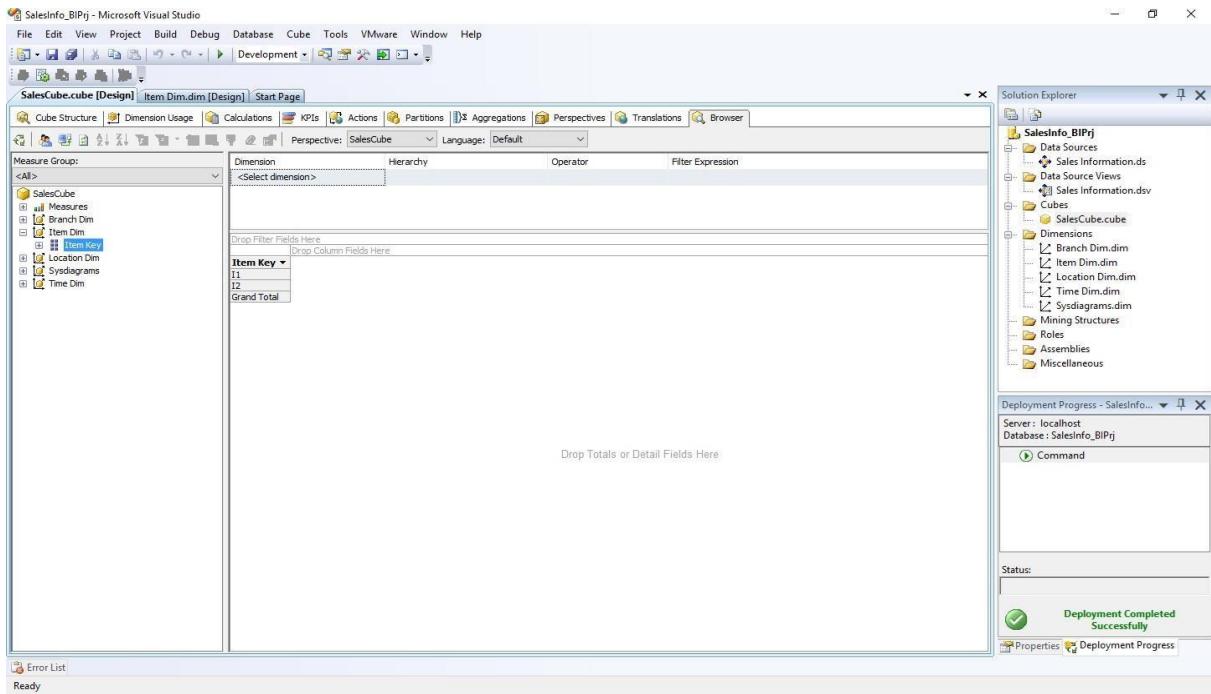
Aim : View cube data in multidimensional

Format.Solution

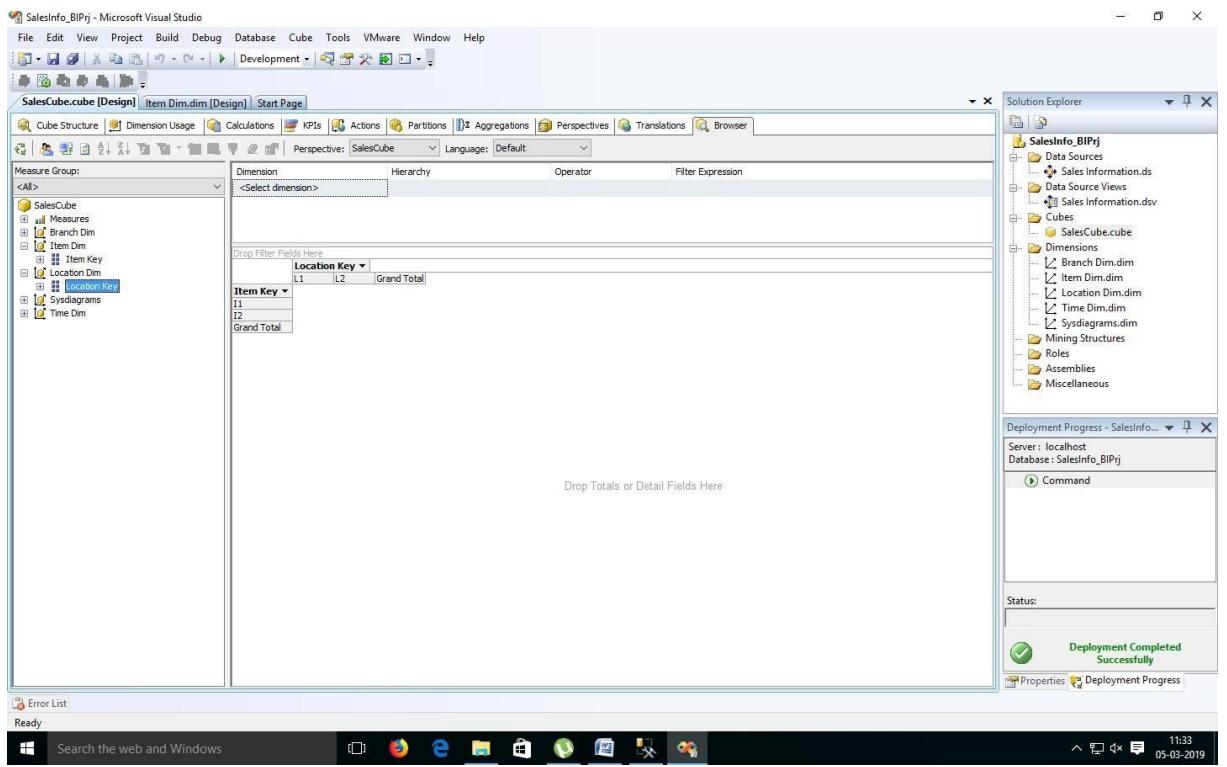
1. DoubleClickon “SalesCube”.Gotothe“Browser”Tab.



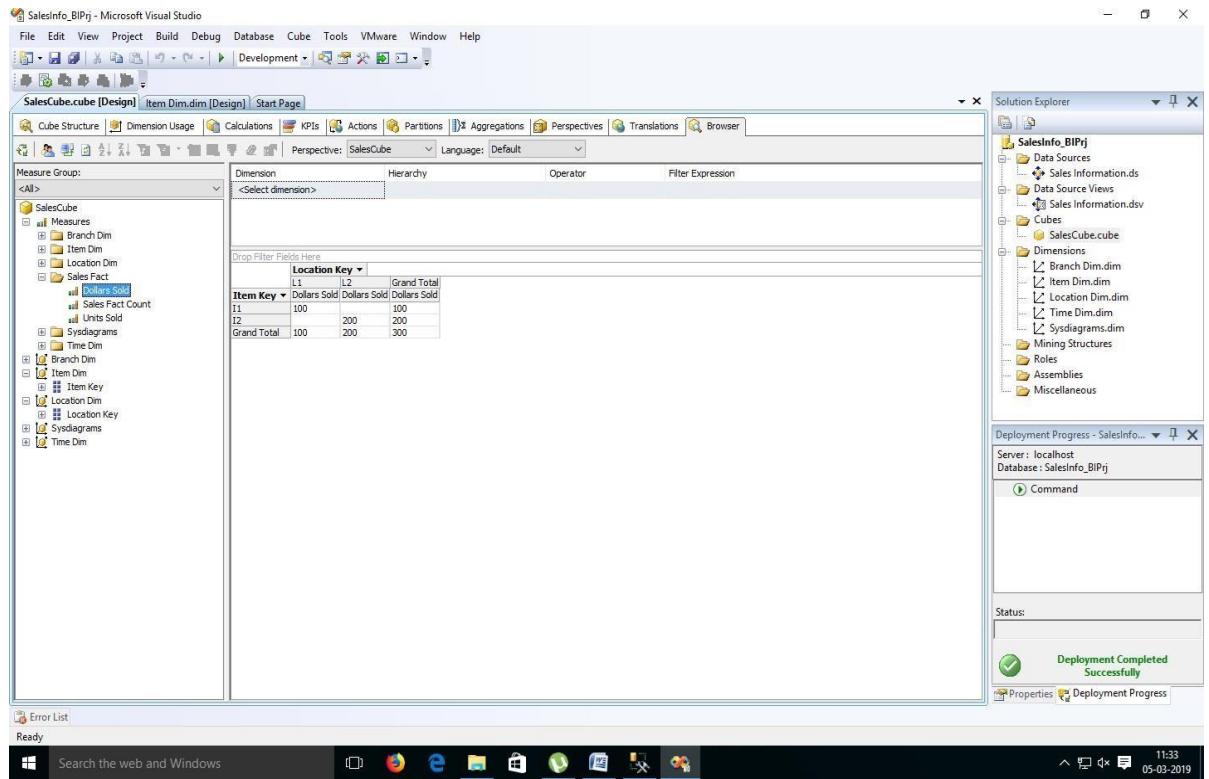
2. Gotothe“ItemDimension”.RightClickon‘ItemKey’->AddtoRowArea.



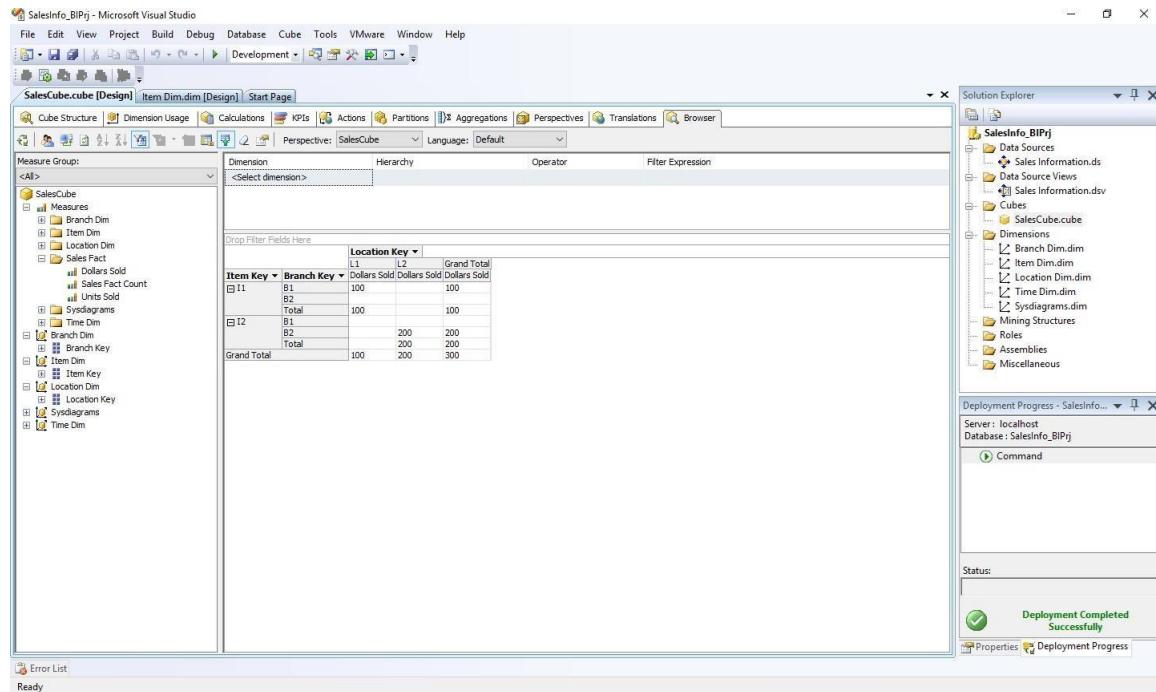
3. Go to the “Location Dimension”. Right Click on ‘Location Key’ -> Add to ColumnArea



4. Go to ‘Measures’. Select ‘SalesFact’ -> Right Click on “Dollars Sold” -> Add to Dataarea



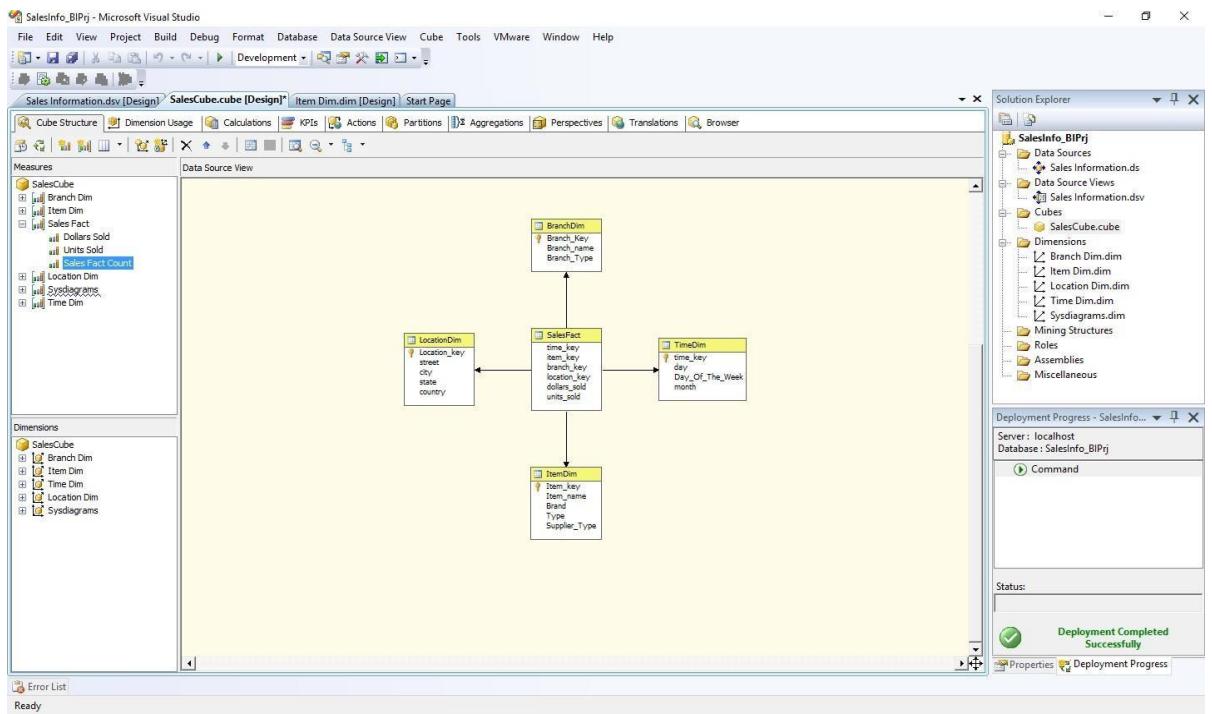
5. Goto the "BranchDimension". RightClickon 'BranchKey' -> AddtoRowArea.



Practical No 6

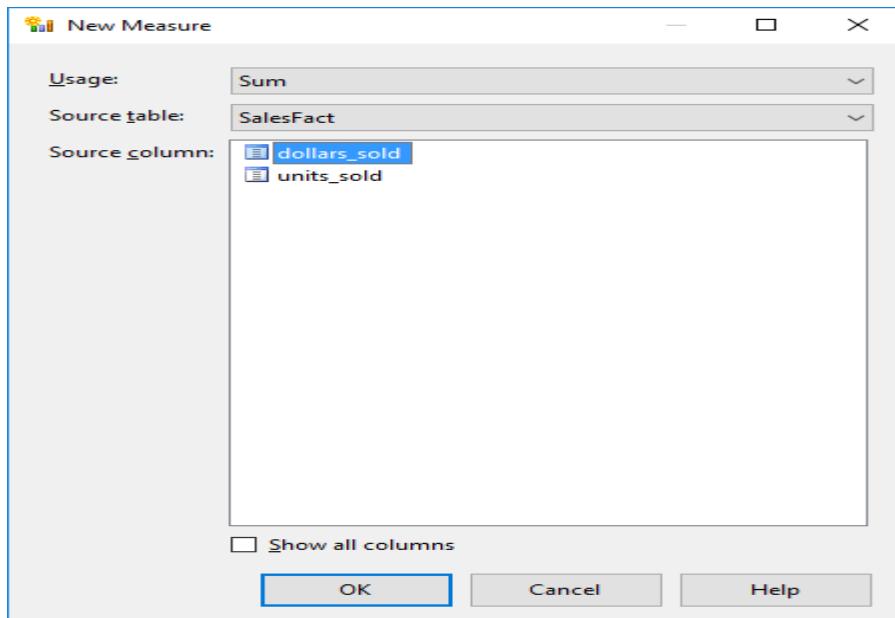
Aim : Working with measures in the cube.

1. Doubleclick on 'SalesCube'. Go to cube structure.



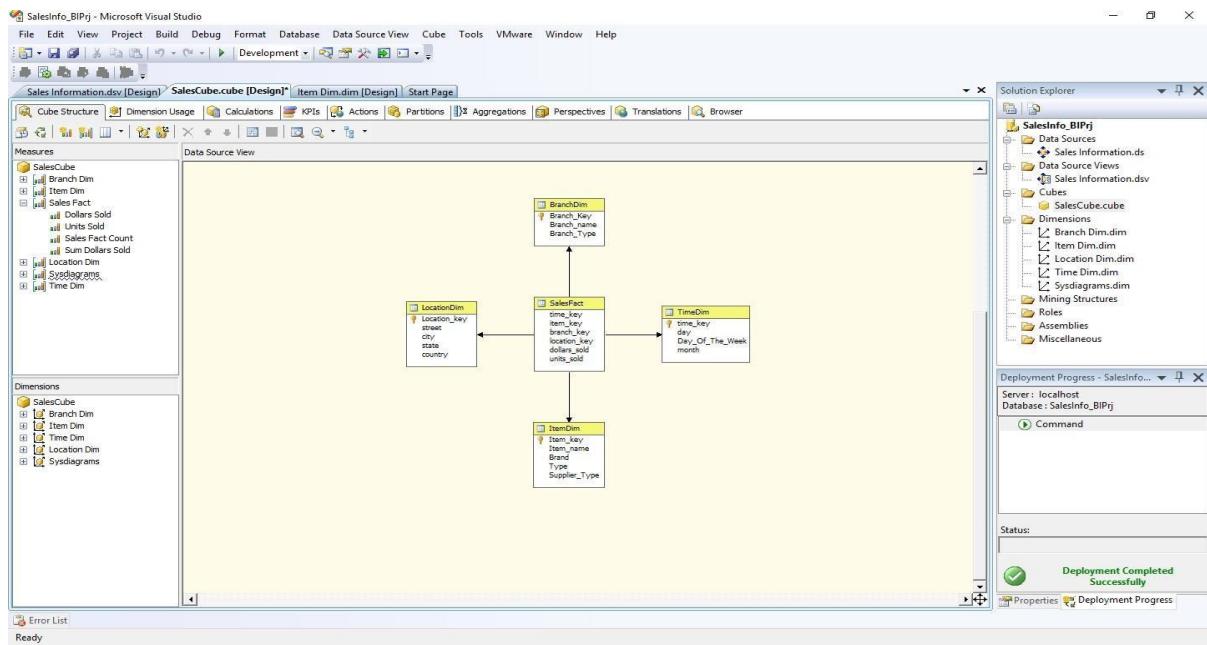
2. Rightclick on SalesCube->NewMeasure.

Select Usage = "Sum" , Source table = "SalesFact" and Source Column = "units_sold".



Click on OK.

3. Rename Measures as "SumDollarsSold".



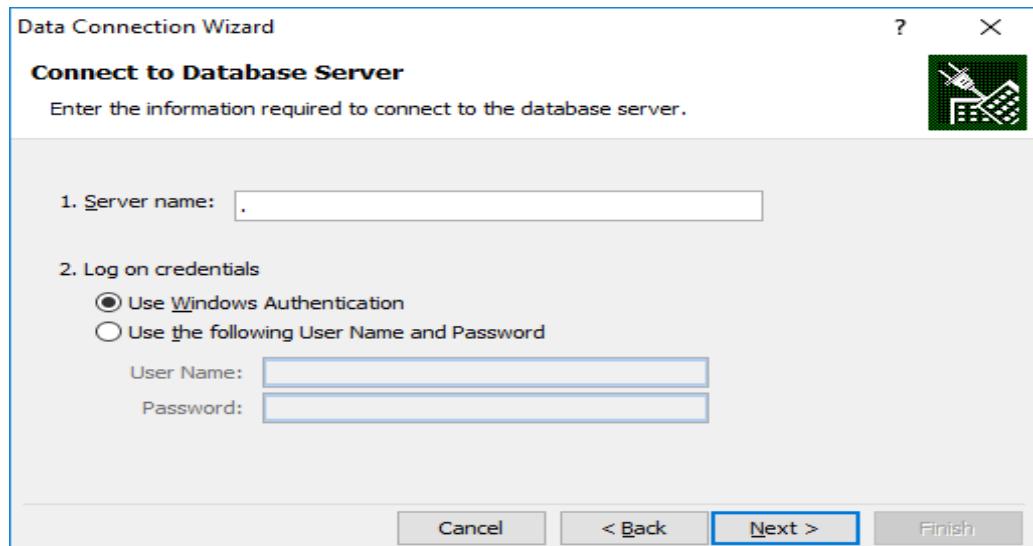
4. Process Cube and Go to Browser and Reconnect it. Right Click on "Sum Dollars Sold" -> Add to Data Area.

The screenshot shows the Microsoft Visual Studio interface for SalesCube.cube [Design]. The left pane shows a tree view of measures and dimensions. The right pane displays a measure group editor with a pivot table preview. The pivot table has columns for Item Key, Branch Key, and various sales metrics like Dollars Sold, Units Sold, and Grand Total. A detailed view of the "Sum Dollars Sold" measure is shown in the bottom right, with a value of 100. The Solution Explorer and Deployment Progress window are also visible.

Practical No 7

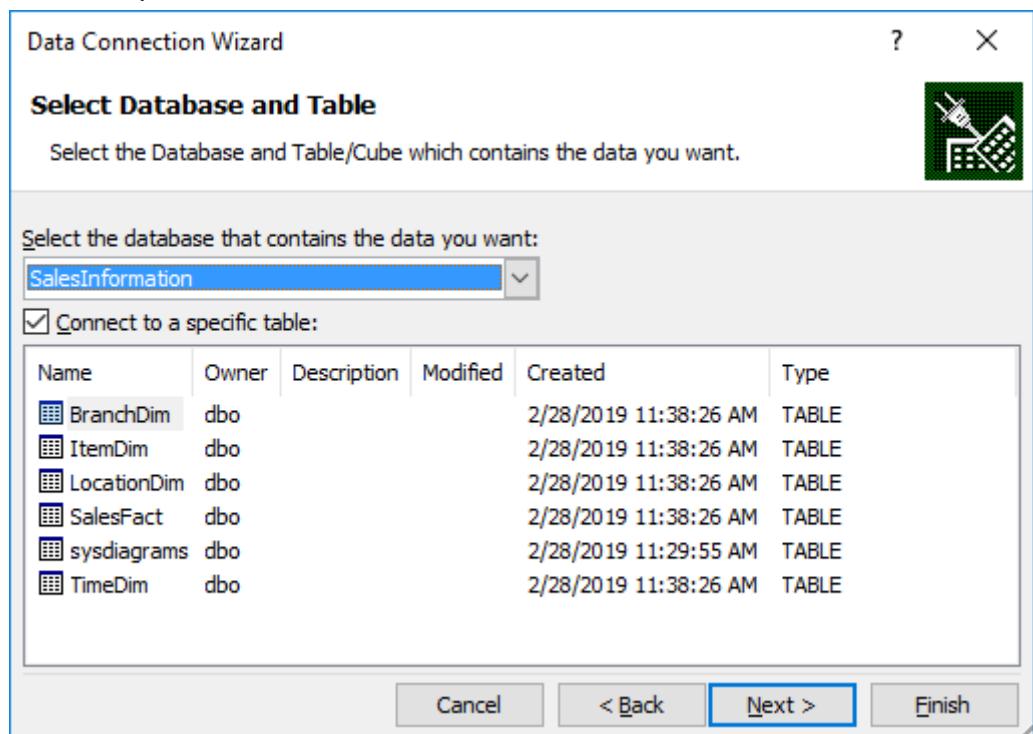
Aim: Creating an Excel PivotTable and PivotChart by using the OLAP cube data.

1. Open MS-Excel. Click on Data menu.
2. Go to From Other Sources.
- 2.1. From SQL Server -> Type Servername as “.”

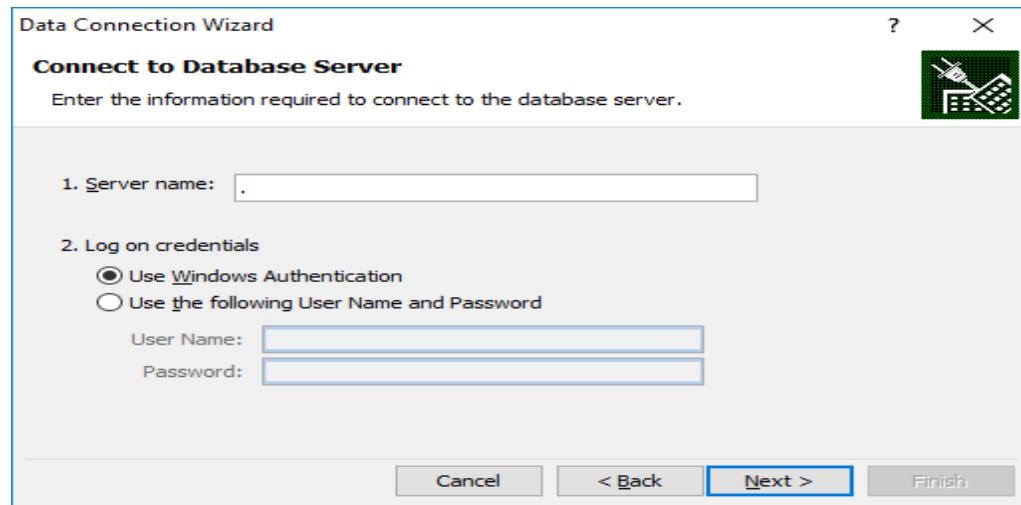


Click on Next.

Choose SQL Database -> "SalesInformation"

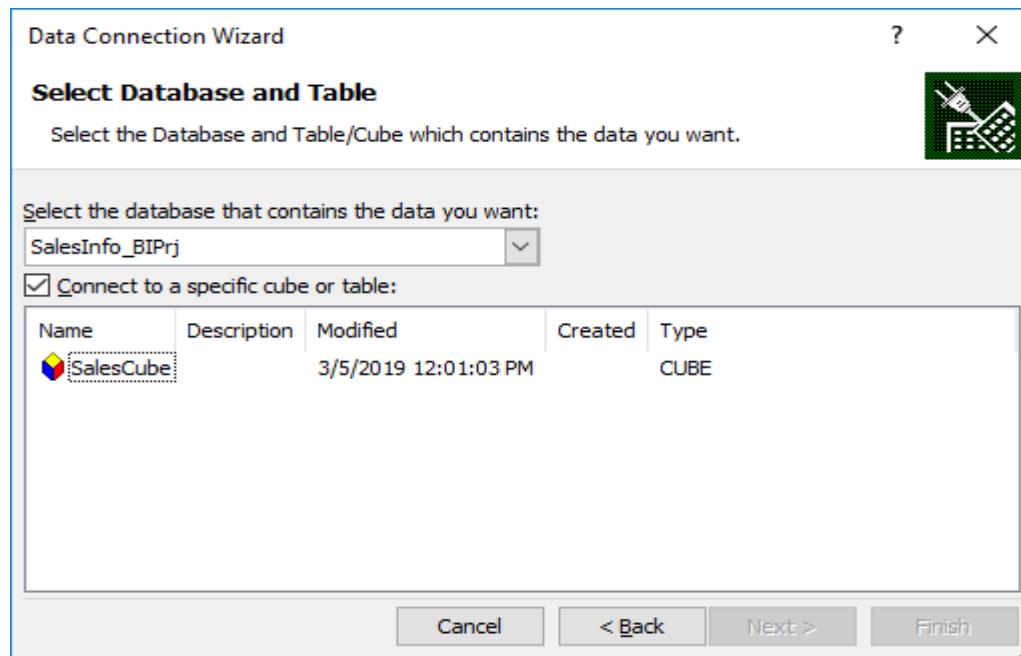


- 2.2. From Analysis Services -> Type Servername as “.”

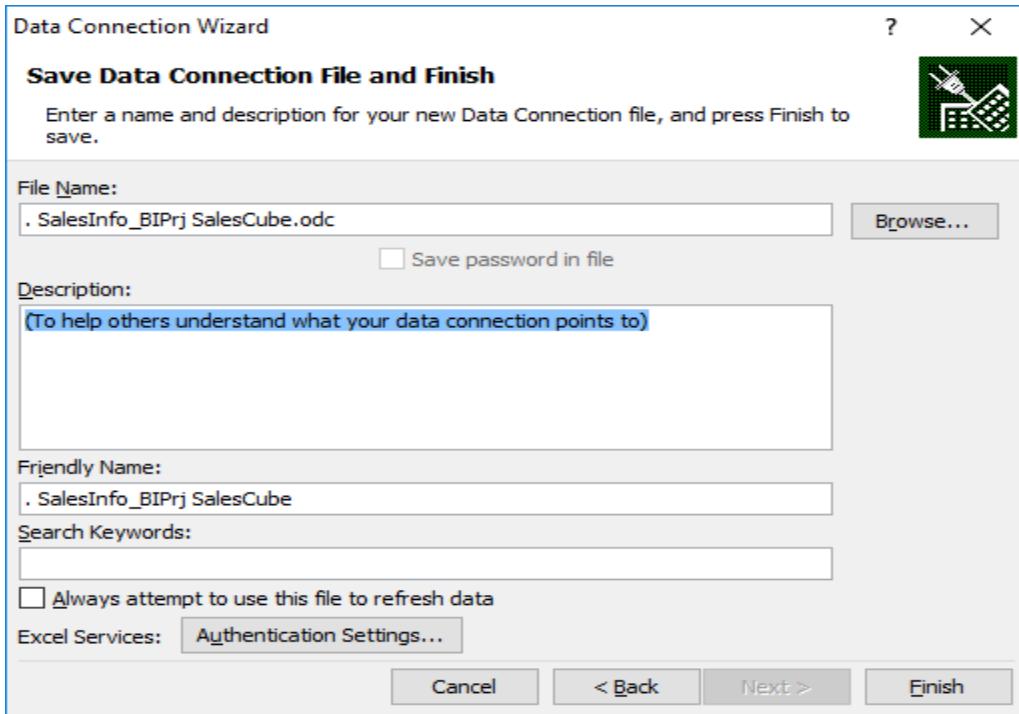


Click on Next.

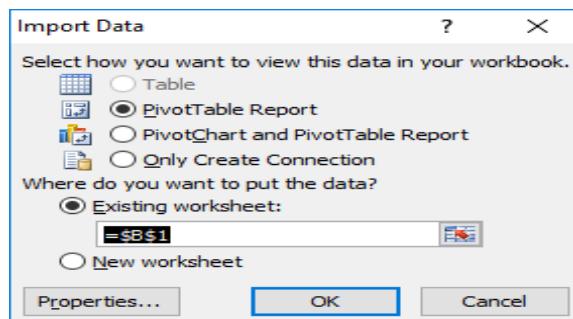
Choose Analysis Database as "SalesInfo_BIPrj". Click on Next.



Click on OK

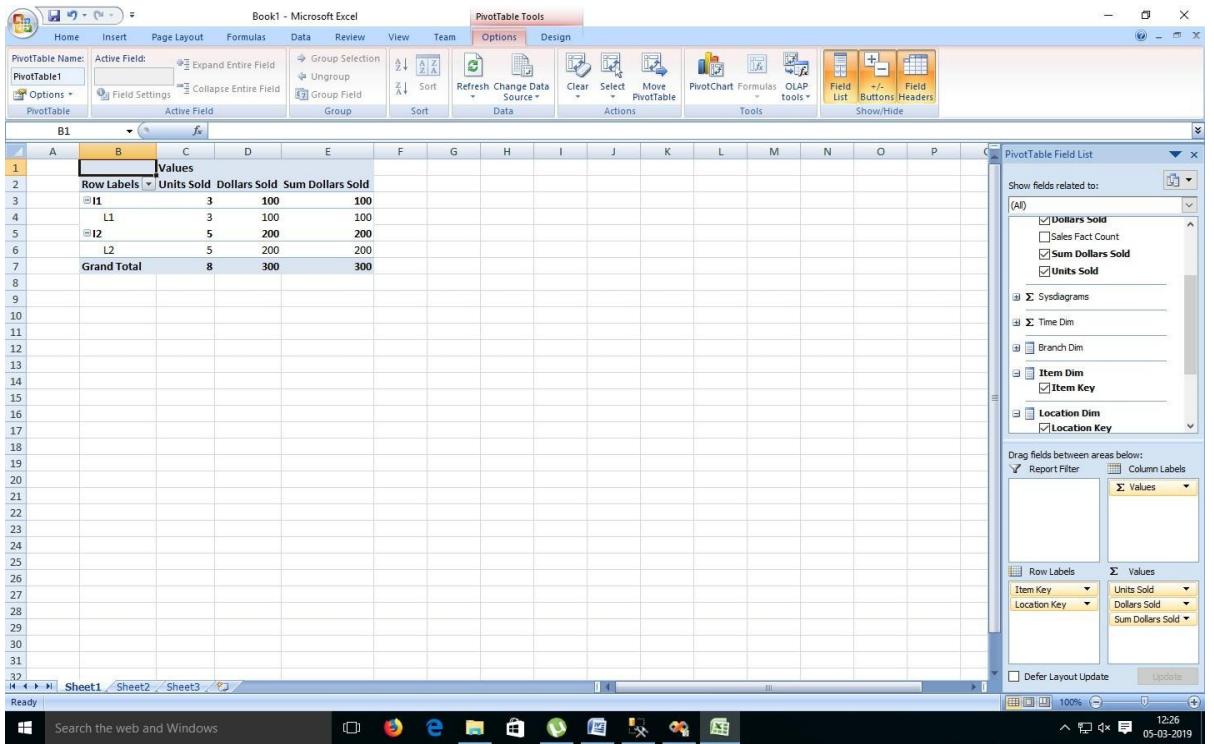


Click on **Finish**.

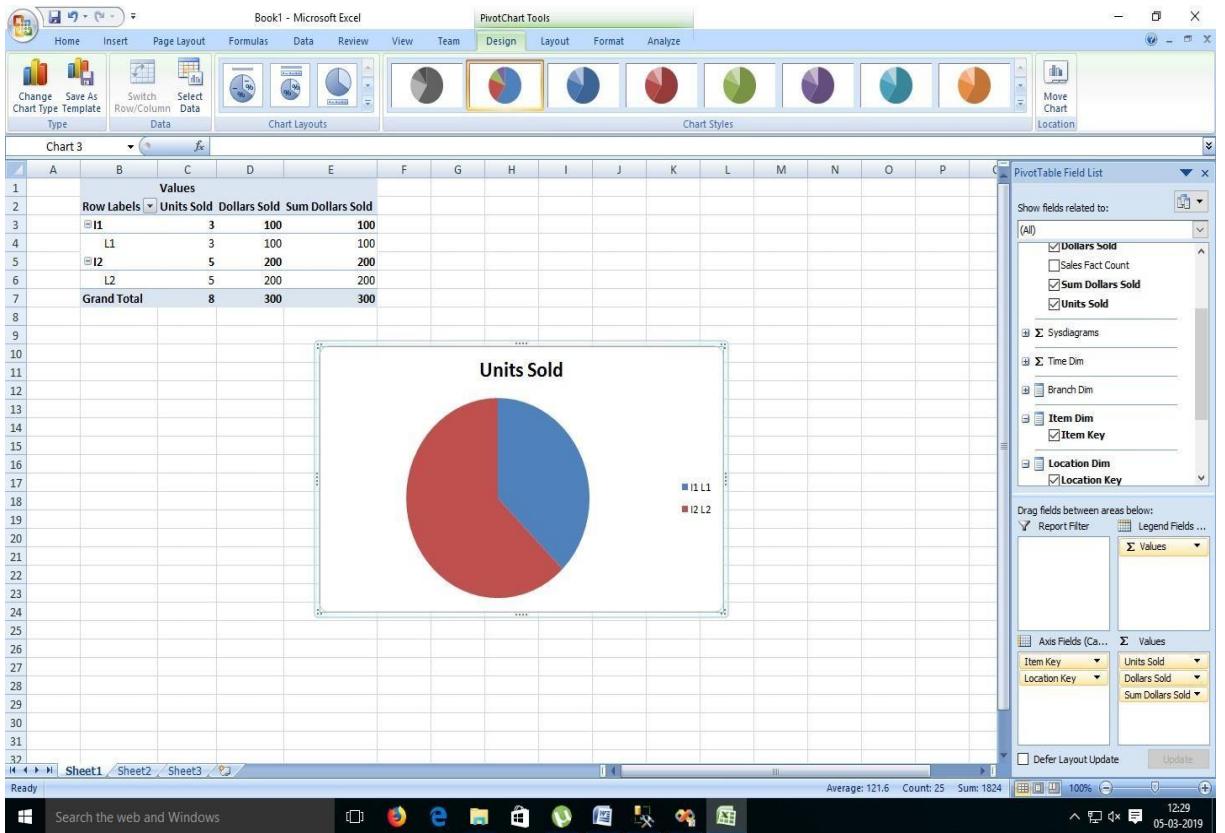


Click on **OK**.

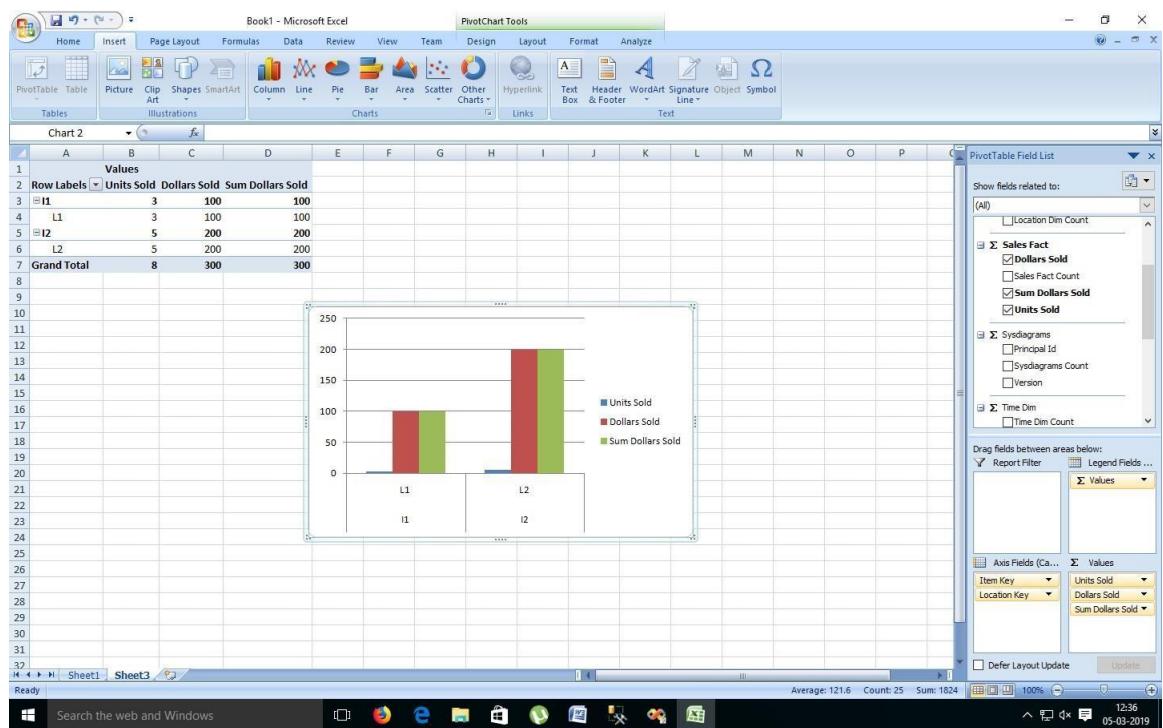
3. Select Item Key, Location Key and Measures as Dollars Sold, Units Sold and SumDollarsSold



4. SelectResultArea.GotoInsertMenu.SelectPieChartoption.



5. SelectResultArea.GotoInsertMenu.SelectColumnnoption.



Practical No 8

Aim : Firing Queries on

Tables.SalesInformation

Open Application->Microsoft SQL Server 2008 R2->SQL Server Management Studio

1. Select Connect Tab->Database Engine->Select Server Name (local)

2. Expand 'Database' -> Expand 'SalesInformation' -> Expand Tables.

3. Fire following queries:

3.1. `SELECT [Branch_Key], [Branch_name], [Branch_Type] FROM [SalesInformation].[dbo].[BranchDim]`

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure, including the SalesInformation schema which contains tables like BranchDim, ItemDim, and SalesFact. The SQL Query window displays the following T-SQL code:

```
SELECT [Branch_Key], [Branch_name], [Branch_Type]
FROM [SalesInformation].[dbo].[BranchDim]
```

The Results pane shows the output of the query:

	Branch_Key	Branch_name	Branch_Type
1	B1	RJ	Vendor
2	B2	JR	Vendor

The Properties pane on the right provides connection details:

- Connection name: (local) (BSC-37\admin)
- Connection type: Local
- Connection status: Open
- Elapsed time: 00:00:00.031
- Finish time: 08-03-2019 09:52:20
- Name: (local)
- Rows returned: 2
- Start time: 08-03-2019 09:52:20
- State: Open
- Login name: BSC-37\admin
- Server name: (local)
- Session version: 10.50.1600
- Session Tracing ID: 53
- SPID: 53

3.2. `SELECT [Item_key], [Item_name], [Brand], [Type], [Supplier_Type] FROM [SalesInformation].[dbo].[ItemDim]`

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure, including the SalesInformation schema which contains tables like ItemDim, BranchDim, and SalesFact. The SQL Query window displays the following T-SQL code:

```
SELECT [Item_key], [Item_name], [Brand], [Type], [Supplier_Type]
FROM [SalesInformation].[dbo].[ItemDim]
```

The Results pane shows the output of the query:

	Item_key	Item_name	Brand	Type	Supplier_Type
1	I1	Laptop	LG	Accessories	VT
2	I2	Mouse	LG	Accessories	VT

The Properties pane on the right provides connection details:

- Connection name: (local) (BSC-37\admin)
- Connection type: Local
- Connection status: Open
- Elapsed time: 00:00:00.037
- Finish time: 08-03-2019 09:54:15
- Name: (local)
- Rows returned: 2
- Start time: 08-03-2019 09:54:15
- State: Open
- Login name: BSC-37\admin
- Server name: (local)
- Session version: 10.50.1600
- Session Tracing ID: 55
- SPID: 55

3.3. `SELECT [Location_key], [street], [city], [state], [country]` **SalesInformation**. [dbo]. [LocationDim]

The screenshot shows the Microsoft SQL Server Management Studio interface. The query window contains the following code:

```
SELECT [Location_key]
      ,[street]
      ,[city]
      ,[state]
      ,[country]
 FROM [SalesInformation].[dbo].[LocationDim]
```

The results pane displays the following data:

	Location_key	street	city	state	country
1	L1	LBS	Mumbai	Maha	India
2	L2	JM	Thane	Maha	India

The status bar at the bottom right indicates the query was executed successfully at 08-03-2019 09:57:38.

3.4. `SELECT [time_key], [item_key], [branch_key], [location_key], [dollars_sold], [units_sold]`
 FROM SalesInformation. [dbo]. [SalesFact]

The screenshot shows the Microsoft SQL Server Management Studio interface. The query window contains the following code:

```
SELECT [time_key]
      ,[item_key]
      ,[branch_key]
      ,[location_key]
      ,[dollars_sold]
      ,[units_sold]
 FROM [SalesInformation].[dbo].[SalesFact]
```

The results pane displays the following data:

	time_key	item_key	branch_key	location_key	dollars_sold	units_sold
1	T1	I1	B1	L1	100	3
2	T2	I2	B2	L2	200	5

The status bar at the bottom right indicates the query was executed successfully at 08-03-2019 09:58:08.

3.5. `SELECT [time_key], [day], [Day_Of_The_Week], [month] FROM [SalesInformation].[dbo].[TimeDim]`

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure, including the SalesInformation database and its tables like BranchDim, ItemDim, SalesFact, and TimeDim. The central pane displays the query:

```
SQLQuery20.sql - [C:\...\admin (61)*] [SQLQuery9.sql - ...C:\...\admin (58)*] [SQLQuery18.sql - ...C:\...\admin (57)*] [SQLQuery17.sql - ...C:\...\admin (55)*]
***** Script for SelectTopNRows command from SSMS *****
SELECT [time_key]
      ,[day]
      ,[Day_Of_The_Week]
      ,[month]
   FROM [SalesInformation].[dbo].[TimeDim]
```

The Results pane shows the output of the query:

	time_key	day	Day_Of_The_Week	month
1	T1	2018-02-01 00:00:00.000	Monday	2018-02-01 00:00:00.000
2	T2	2019-05-03 00:00:00.000	Tuesday	2019-03-03 00:00:00.000

The Properties pane on the right shows connection details. The status bar at the bottom indicates the session ID is 61.

3.6. `SELECT [SalesInformation].[dbo].[BranchDim].[Branch_Key]
 , [dollars_sold], [units_sold]
 FROM [SalesInformation].[dbo].[BranchDim],
 .[dbo].[SalesFact]
 WHERE [SalesInformation].[dbo].[BranchDim].[Branch_Key] =
 .[dbo].[SalesFact].[Branch_Key];`

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure, including the SalesInformation database and its tables like BranchDim, SalesFact, and SalesInfo. The central pane displays the query:

```
SQLQuery23.sql - [C:\...\admin (57)*] [SQLQuery22.sql - ...SC-37\...\admin (55)*] [SQLQuery21.sql - ...SC-37\...\admin (53)*]
SELECT [SalesInformation].[dbo].[BranchDim].[Branch_Key],
       ,[Branch_name],
       ,[dollars_sold],
       ,[units_sold]
  FROM [SalesInformation].[dbo].[BranchDim]
  JOIN [SalesInformation].[dbo].[SalesFact]
    ON [SalesInformation].[dbo].[BranchDim].[Branch_Key] =
       [SalesInformation].[dbo].[SalesFact].[Branch_Key];
```

The Results pane shows the output of the query:

	Branch_Key	Branch_name	dollars_sold	units_sold
1	B1	RJ	100	3
2	B2	JR	200	5

The Properties pane on the right shows connection details. The status bar at the bottom indicates the session ID is 57.

```

3.7.    SELECT[ S a l e s I n f o r m a t i o n ].[dbo].[ItemDim]
        .[Item_Key], [item_Name]
        , [Type], [dollars_sold], [units_sold] FRO
M [SalesInformation].[dbo].[ItemDim],
    [SalesInformation].[dbo].[SalesFact]
Where [SalesInformation].[dbo].[ItemDim].[Item_key]=
        [dbo].[SalesFact].[item_key];

```

The screenshot shows the Microsoft SQL Server Management Studio interface. The left pane displays the Object Explorer with the master database selected. The center pane contains a query window with the following SQL code:

```

SELECT[ S a l e s I n f o r m a t i o n ].[dbo].[ItemDim]
        .[Item_Key], [item_Name]
        , [Type], [dollars_sold], [units_sold] FRO
M [SalesInformation].[dbo].[ItemDim],
    [SalesInformation].[dbo].[SalesFact]
Where [SalesInformation].[dbo].[ItemDim].[Item_key]=
        [dbo].[SalesFact].[item_key];

```

The right pane shows the Properties window and the Results pane. The Results pane displays the query results:

Item_Key	item_Name	Type	dollars_sold	units_sold
11	Laptop	Accessories	100	3
12	Mouse	Accessories	200	5

A message at the bottom of the Results pane says "Query executed successfully."

```

3.8.    SELECT[ S a l e s I n f o r m a t i o n ].[dbo].[LocationDi
m].[Location_key]
        , [city], [item_Key], [dollars_sold], [units_sold] FR
OM [SalesInformation].[dbo].[LocationDim],
    [SalesInformation].[dbo].[SalesFact]
Where [SalesInformation].[dbo].[LocationDim].[Location_key]=
        [dbo].[SalesFact].[location_key];

```

Microsoft SQL Server Management Studio

File Edit View Query Debug Tools Window Community Help

New Query | Open | Save | Print | Execute | Script | Copy | Paste | Find | Replace | Properties | Column Widths | Results | Messages

Object Explorer

(local) SQL Server 10.50.1600 - BSC-37\admi

Database System Databases Database Snapshots ReportServer ReportServerTempDB Sales SalesInformation Database Diagrams Tables System Tables dbo.BranchDim dbo.ItemDim dbo.LocationDim dbo.SalesFact dbo.TimeDim Views Synonyms Programmability Service Broker Storage Security SalesProduct Security Server Objects Replication Management

SQLQuery4.sql - (SC-37\admin (60)) SQLQuery3.sql - (SC-37\admin (53)) SQLQuery2.sql - (SC-37\admin (58)) SQLQuery1.sql - (SC-37\admin (56))

```
***** Script for SelectTopNRows command from SSMS *****
SELECT [SalesInformation].[dbo].[LocationDim].[Location_key]
      ,[city]
      ,[item_Key]
      ,[dollars_sold]
      ,[units_sold]
  FROM [SalesInformation].[dbo].[LocationDim],[SalesInformation].[dbo].[SalesFact]
 where [SalesInformation].[dbo].[LocationDim].[Location_key]=
 [SalesInformation].[dbo].[SalesFact].[location_key];
```

Results Messages

	Location_key	city	item_Key	dollars_sold	units_sold
1	L1	Mumbai	I1	100	3
2	L2	Thane	I2	200	5

Query executed successfully.

(local) (10.50 RTM) | BSC-37\admin (56) | master | 00:00:00 | 2 rows

Properties

Current connection parameters

Aggregate Status

Connection fail Connection elapsed time 00:00:00.031 Finish time 08-03-2019 11:12:58 Name (local) Rows returned 2 Start time 08-03-2019 11:12:58 State Open

Connection

Connection name (local) (BSC-37\admin)

Connection Details

Connection elapsed time 00:00:00.031 Connection finished 08-03-2019 11:12:58 Connection row 2 Connection start 08-03-2019 11:12:58 Connection status Open Display name (local) Login name BSC-37\admin Server name (local) Server version 10.50.1600 Session Tracing 1 SPID 56

Name The name of the connection.

Output

Auto-attach to process '(1044) [SQL] (local)' on machine '(local)' succeeded.
The thread '(local) (56)' (0x1690) has exited with code 0 (0x0).
The thread '(local) (56)' (0x1690) has exited with code 0 (0x0).
The program '(1044) [SQL] (local)' has exited with code 0 (0x0).

Ready

Search the web and Windows

Ln 9 Col 61 Ch 55 INS

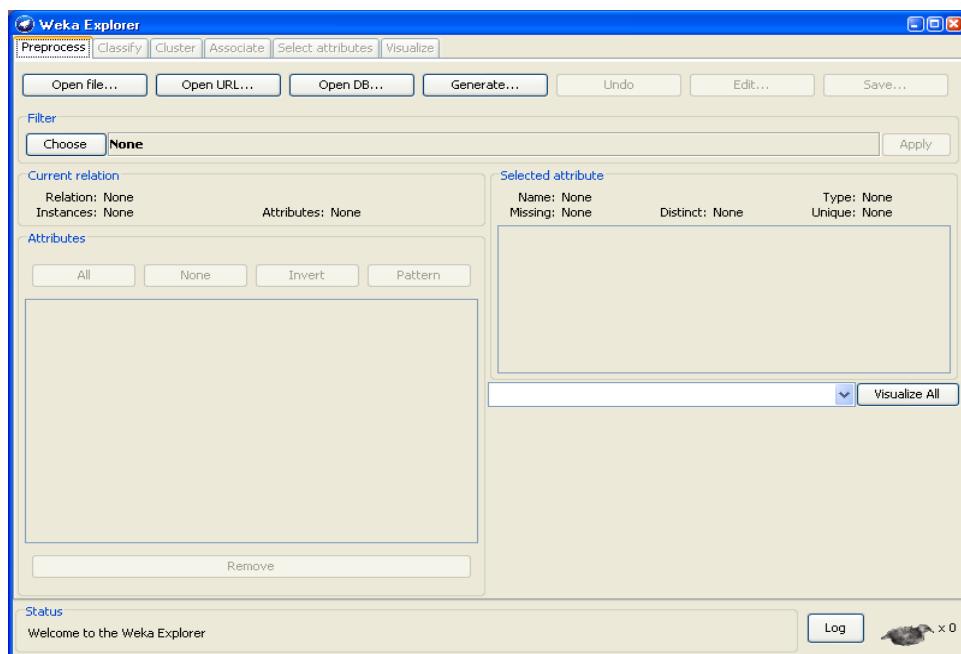
11:13 08-03-2019

PracticalNo-10

Aim:DataPreProcessing

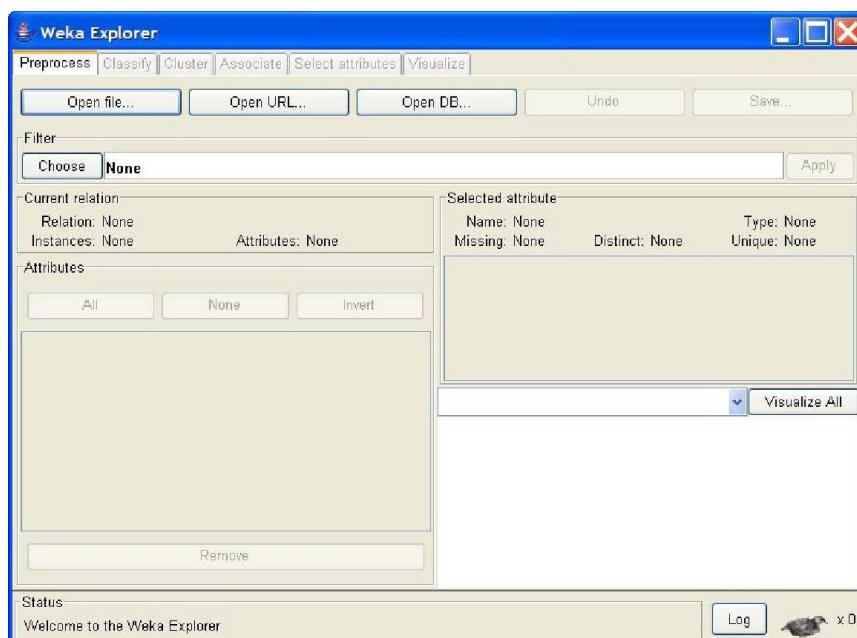
Solution:

Only the first tab, 'Preprocess', is active at the moment because there is no dataset open.



Openingfilefrom a localfilesystem

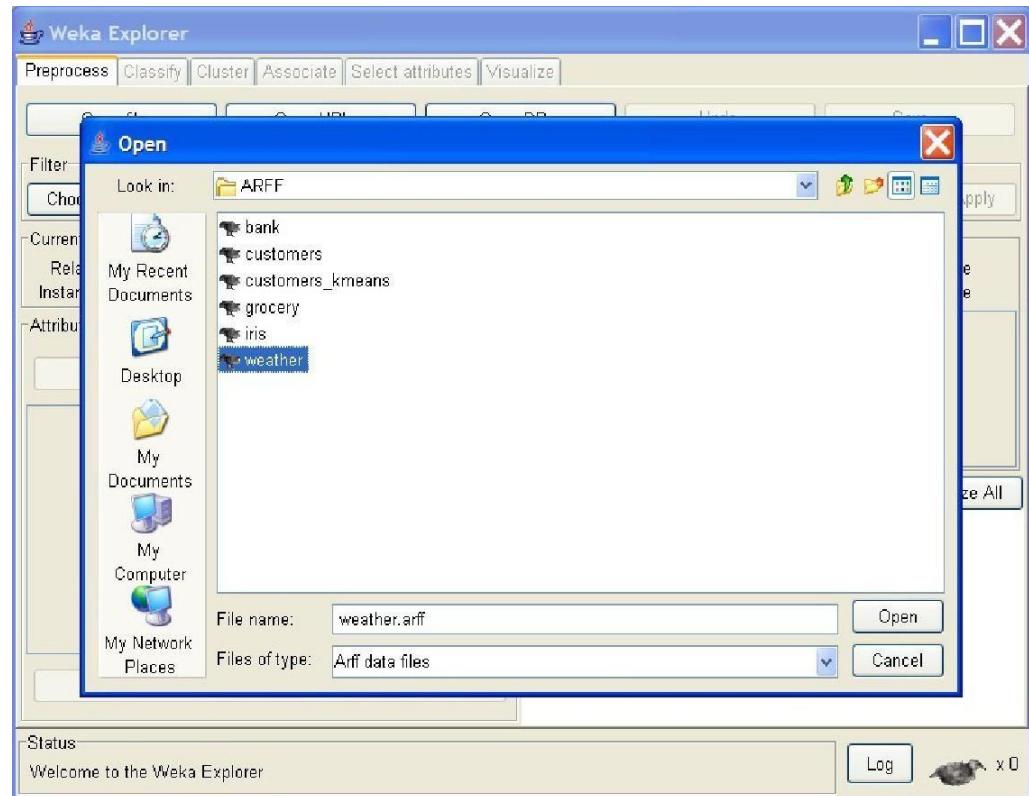
Click on 'Openfile...' button



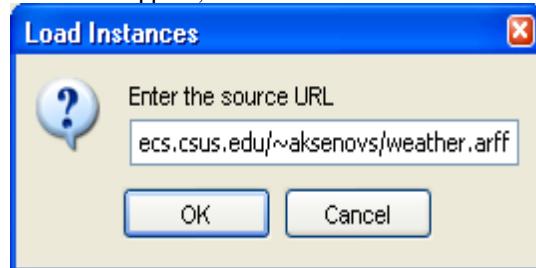
It brings up a dialog box allowing you to browse for the data file on the local filesystem, choose

“weather.arff”file.

Opening filefroma web site



Afilecanbeopenedfromawebiste.Suppose, that“weather.arff”isonthefollowingwebiste:



Reading data from database:

The screenshot shows a web browser window titled "CSC 288" displaying course assignments. The assignments listed are:

- Assignment 1 - Vocabulary and Case Study
- Assignment 2 - Classification
- Assignment 3 - Clustering
- Assignment 4 - Association Rules
- Assignment 5 - Credibility

Below the assignments, there are sections for "Paper Review" and "Term Project".

Under "Paper Review", the links are:

- Summary
- PowerPoint Presentation

Under "Term Project", the links are:

- Project Poposal
- Project Progress Report
- Project Progress Report Presentation

Below these sections, there are links for "WEKA Tutorial" and "WEKA Tutorial Presentation".

At the bottom of the page, there is a link to "weather.arff".

Below the browser window, there is a separate window titled "SQL-Viewer" showing the results of a query:

```
select * from stud
```

Row	name	hobby
1	dfgd	dfgd
2	dfgdf	dfgd
3	drg	dfg

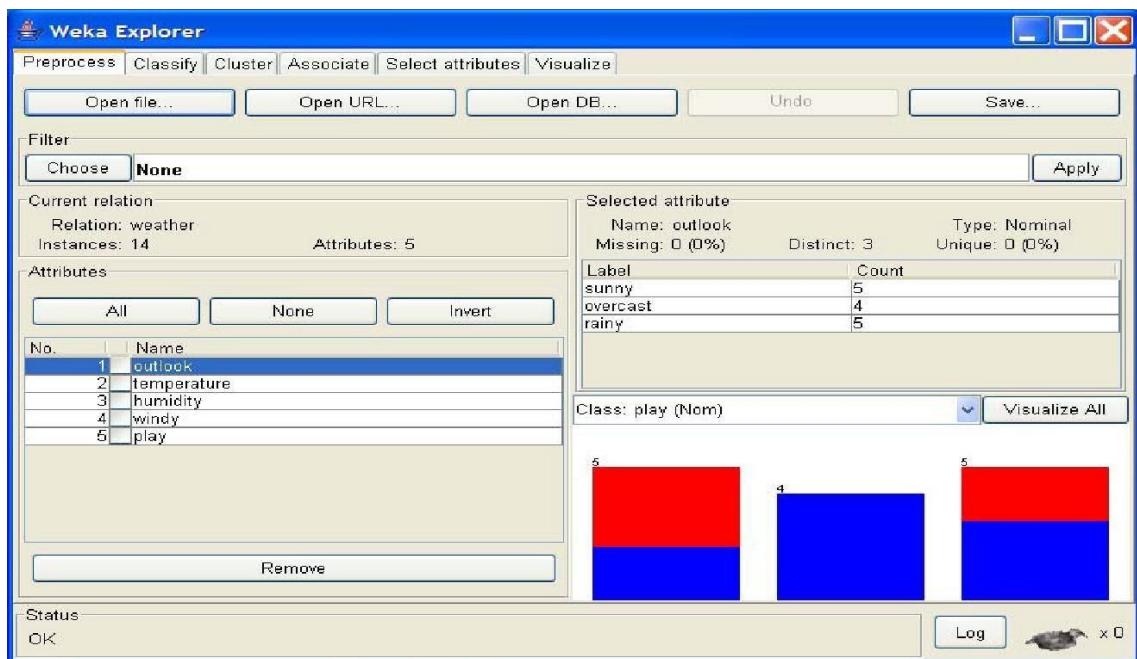
The "Info" pane shows the query history:

```
y: select * from stud  
is selected.  
y: select * from stud  
is selected.
```

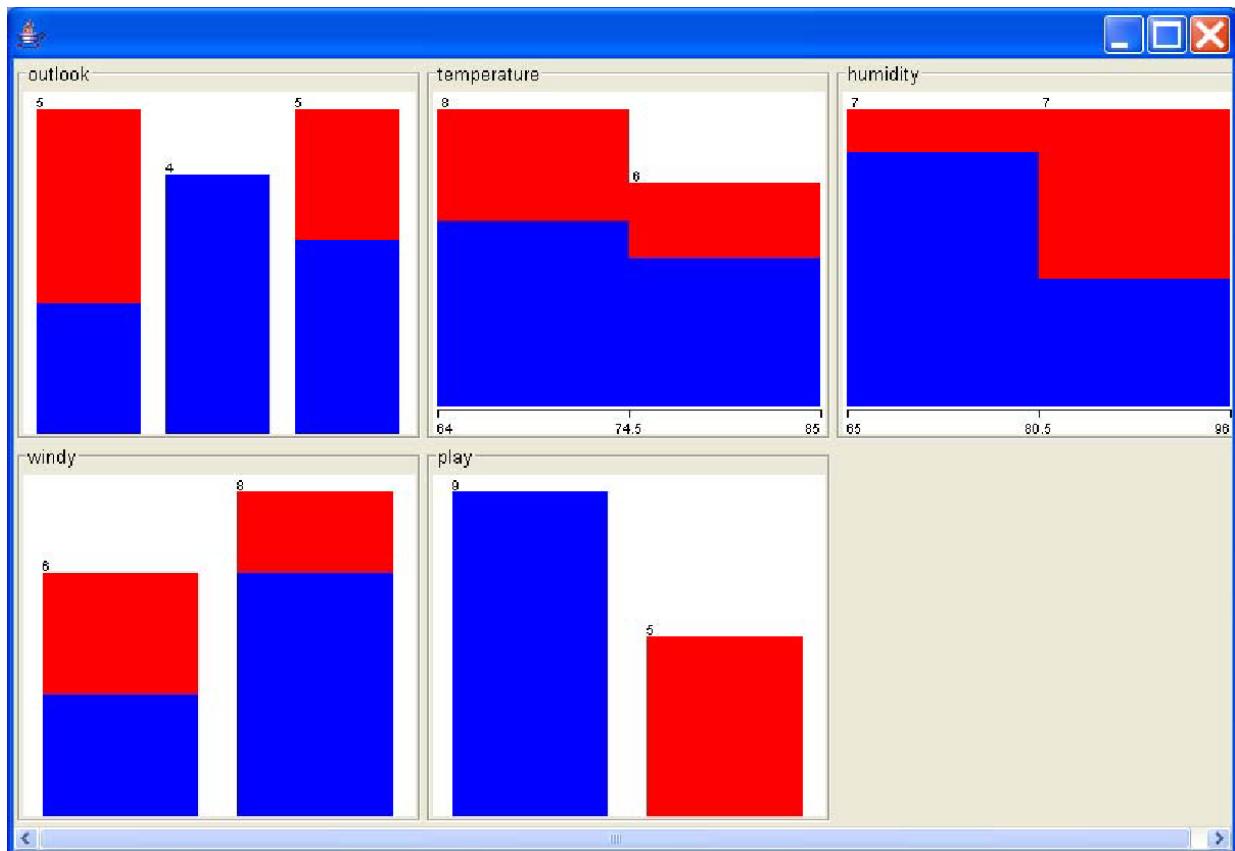
The "Current query" field shows "select * from stud".

Loading data

The most common and easiest way of loading data into WEKA is from ARFF file, using **File** button.



VisualizeAttributes:



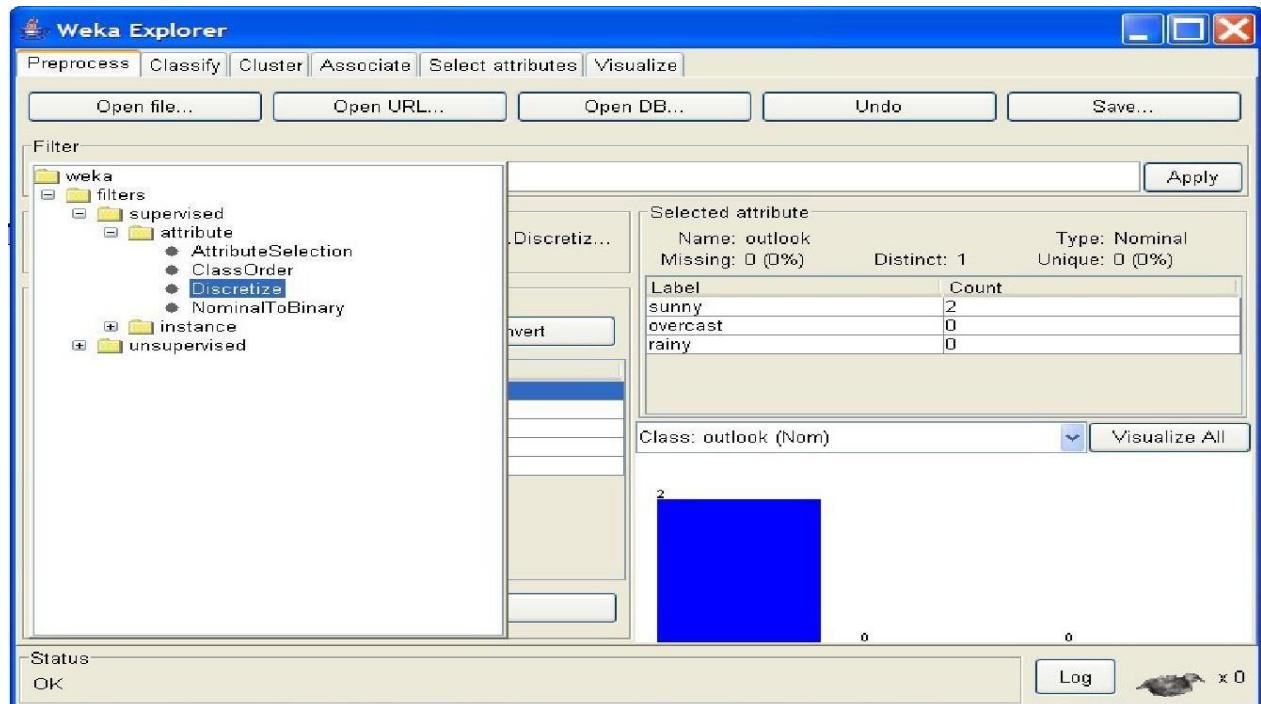
visualize all attributes by clicking on 'VisualizeAll' button.,

Practical No-11

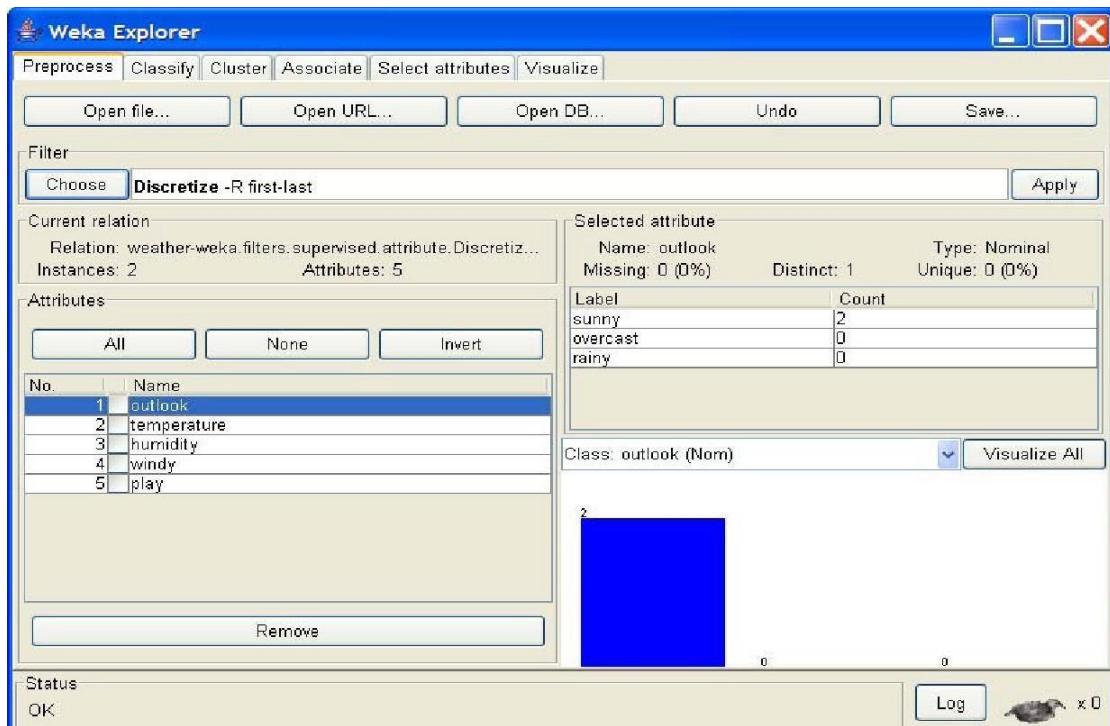
Aim: Data discretization.

Solution:

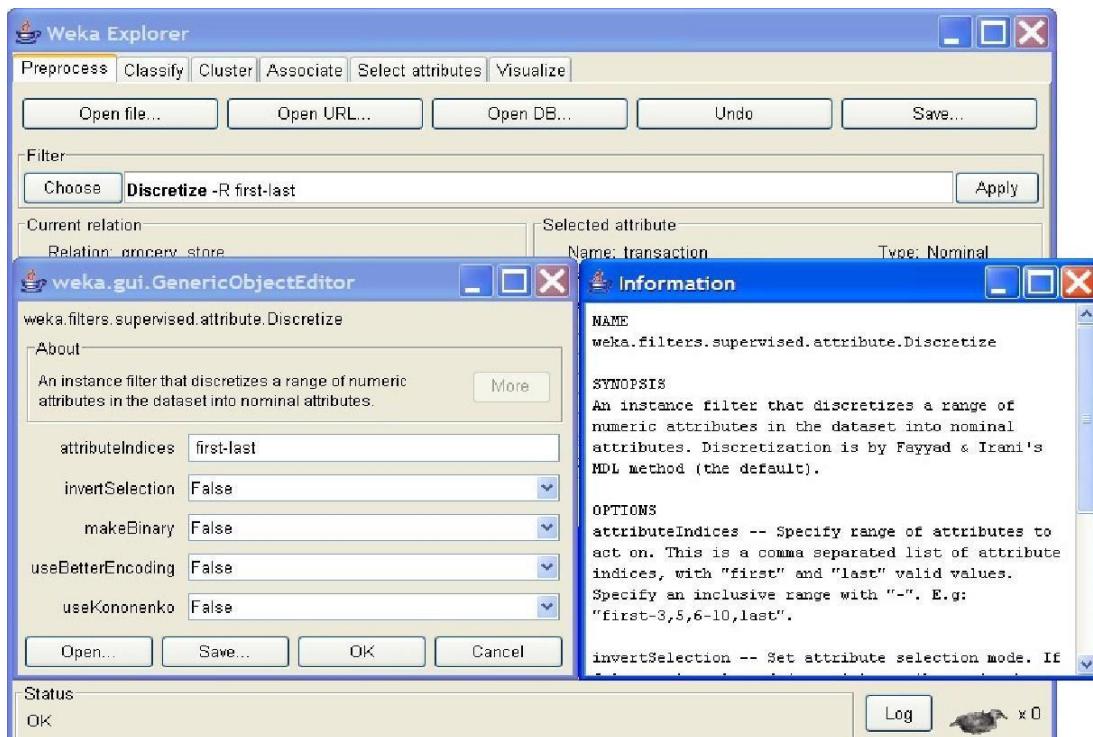
In 'Filters' window, click on the 'Choose' button. This will show pull-down menu with a list of available filters. Select Supervised [Attribute] Discretize and click on 'Apply' button.



The filter will convert Numeric values into Nominal.
The fields in the window change to reflect available options.



a 'GenericObjectEditor' dialog box comes up on your screen. This box lets you to choose the filter configuration options.

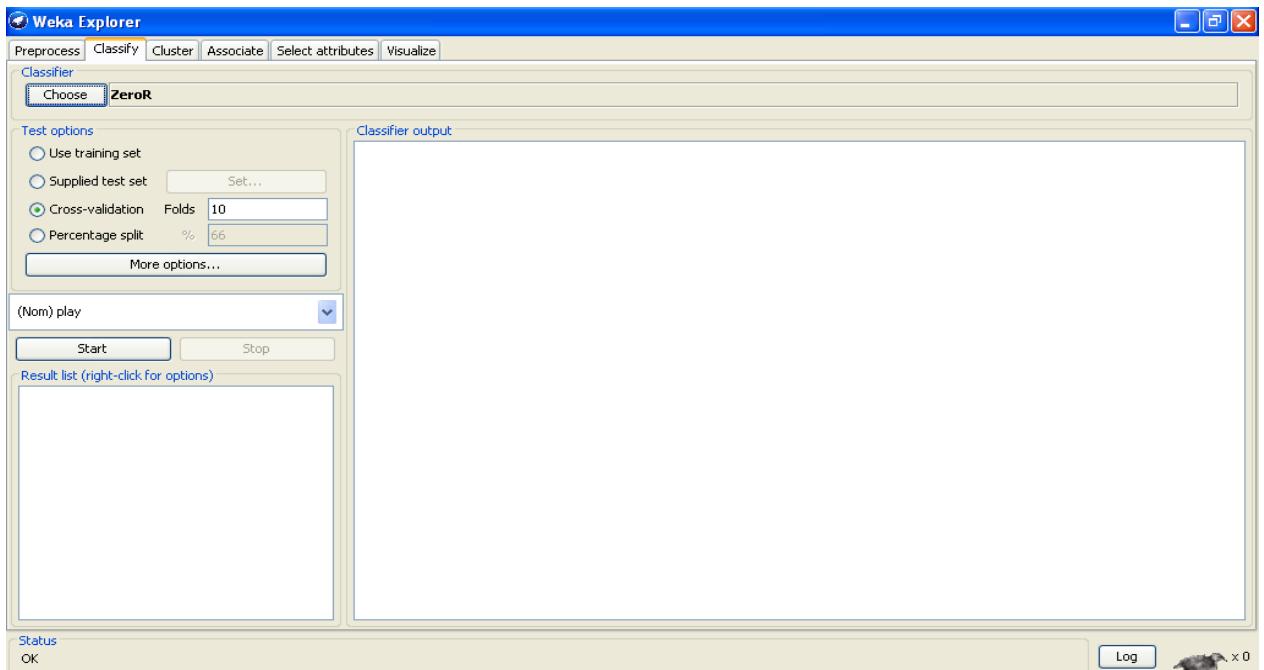


PracticalNo-12

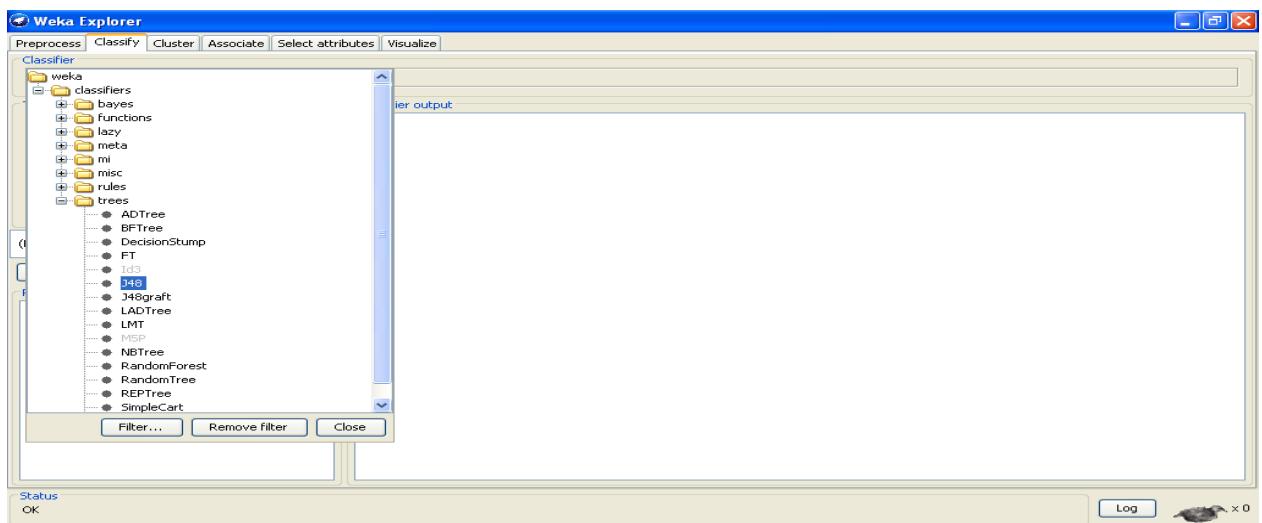
Aim:Classificationproblems.

Solution:

Onceyouhaveyourdatasetloaded,allthetabsareavailabletoyou.Clickonthe‘Classify’tab.



Clickon‘Choose’buttoninthe‘Classifier’boxjustbelowthetabsandselectC4.5.
WEKA Classifiers Trees J 4 8 .

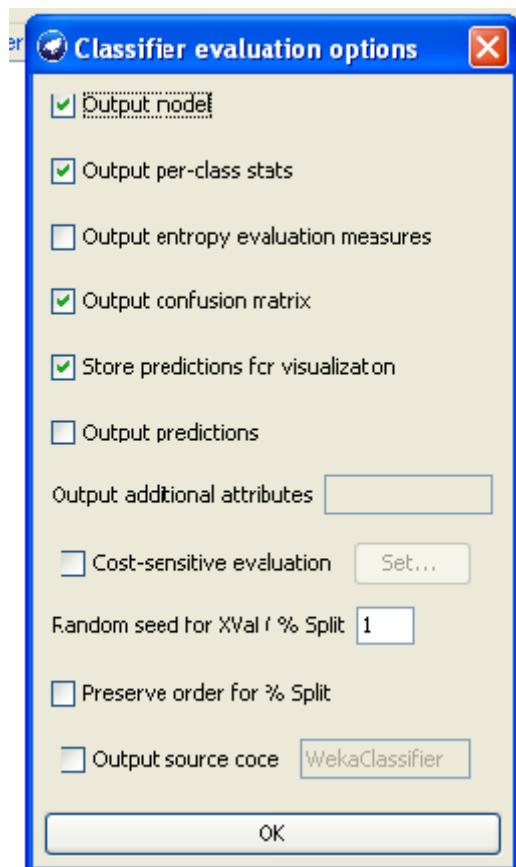


Check ‘Percentage split’ radio-button and keep it as default 66%. Click on
‘More...’button.

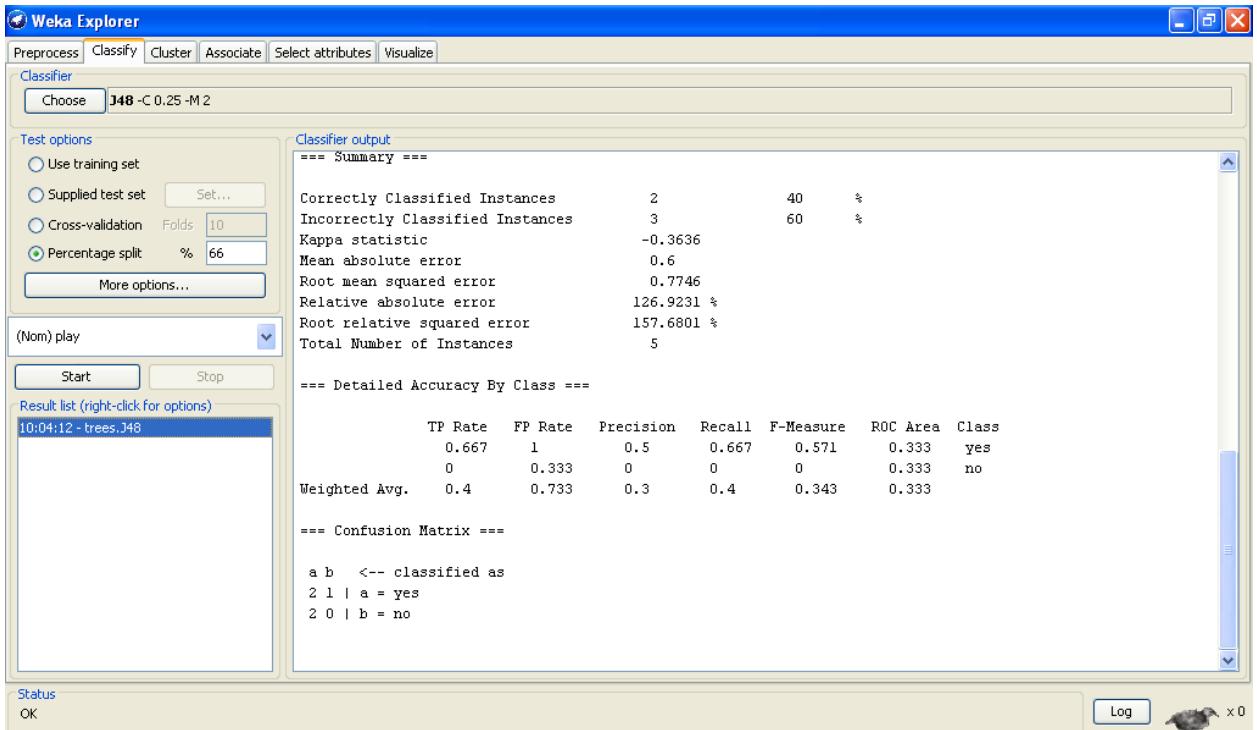
makesurethat theFollowing optionsarechecked:

1. Outputmodel.

2. Output per-class stats.
3. Output confusion matrix
4. Store predictions for visualization.
5. Set 'Random seed for Xval / % Split' to 1.

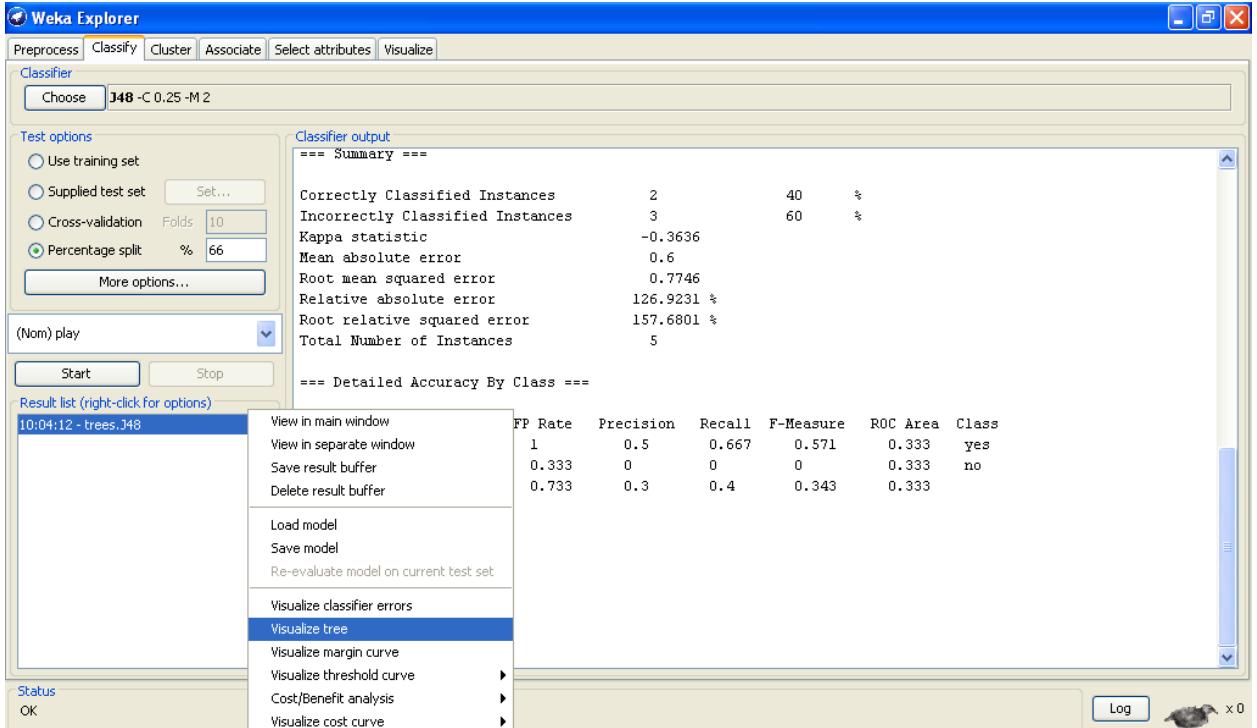


Once the options have been specified, you can run the classification algorithm. Click on the **Start** button.

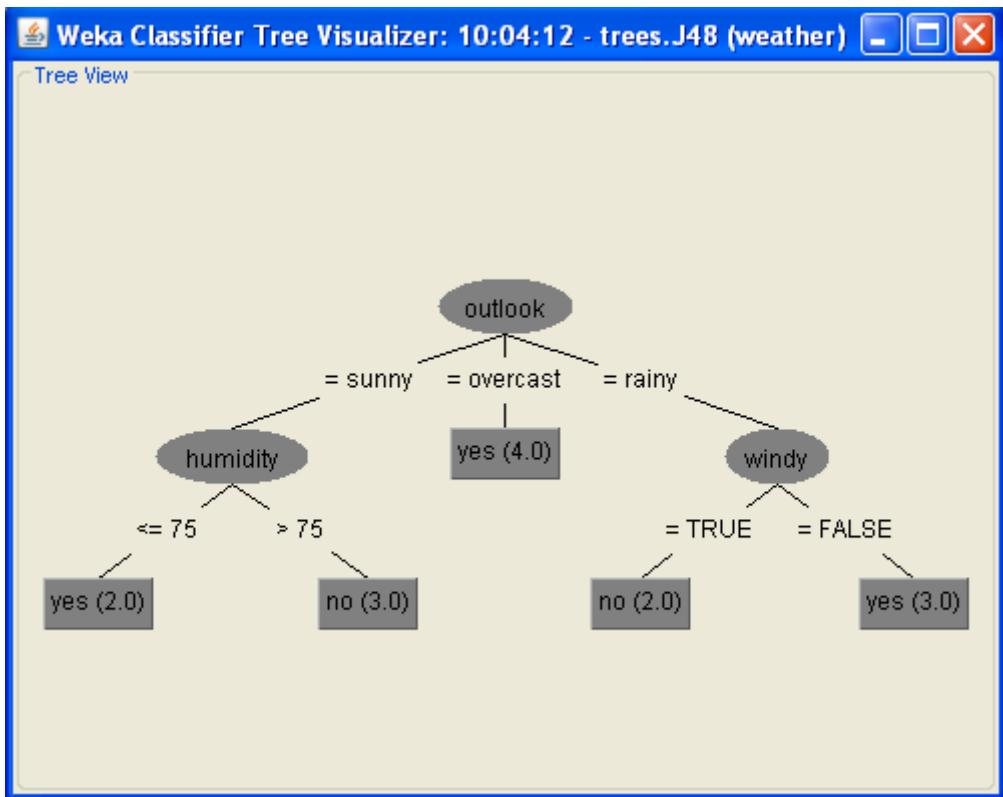


To see a graphical representation of the classification tree. Right-click on

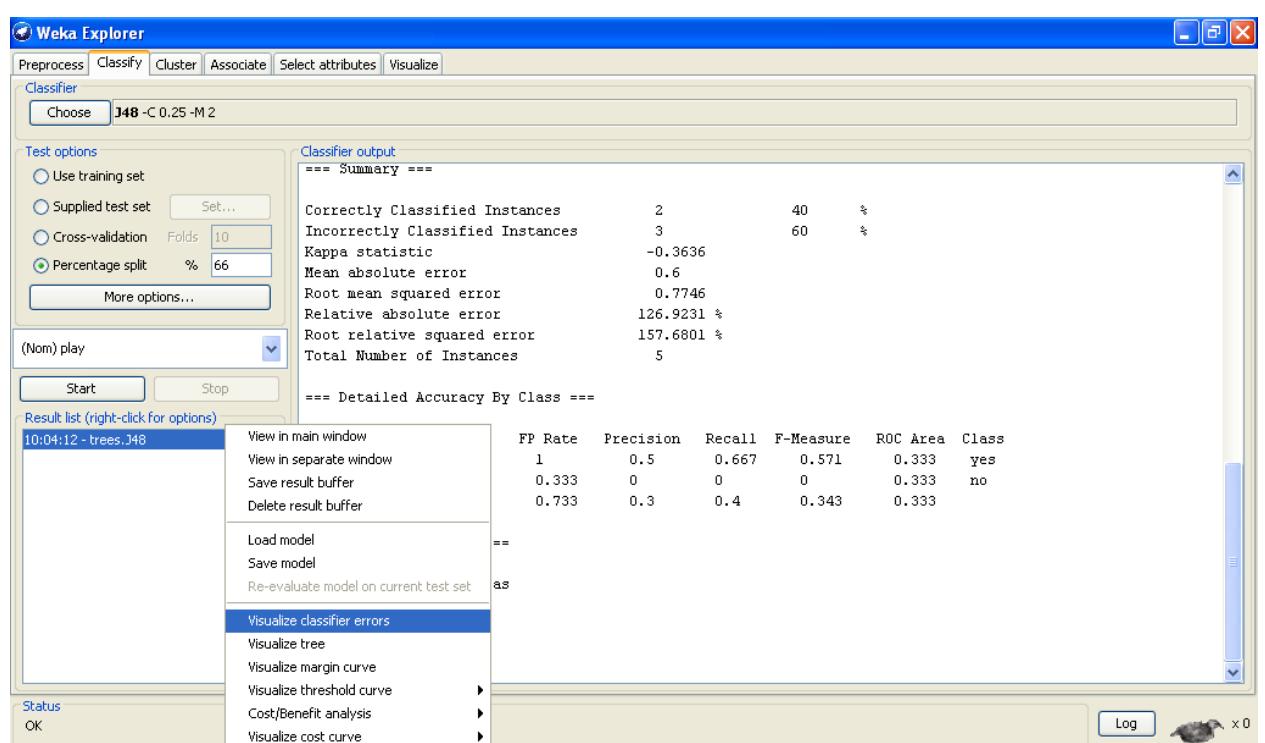
the entry in 'Result list' for which you would like to visualize a tree.



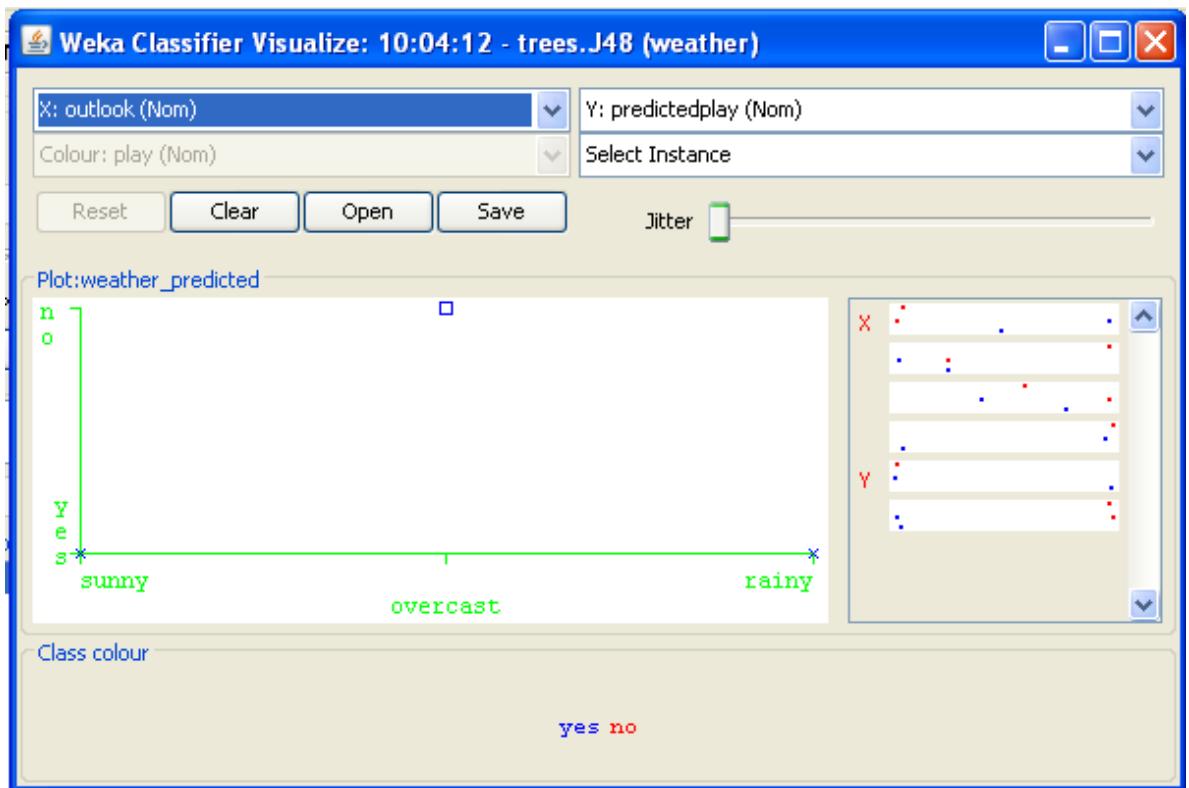
Select the item 'Visualize tree'; a new window comes up to the screen displaying the tree.



To visualize classification errors. Right-click on the entry in 'Result list' again and click 'Visualize errors' from the menu:



'Visualize' window displaying graph appears on the screen.

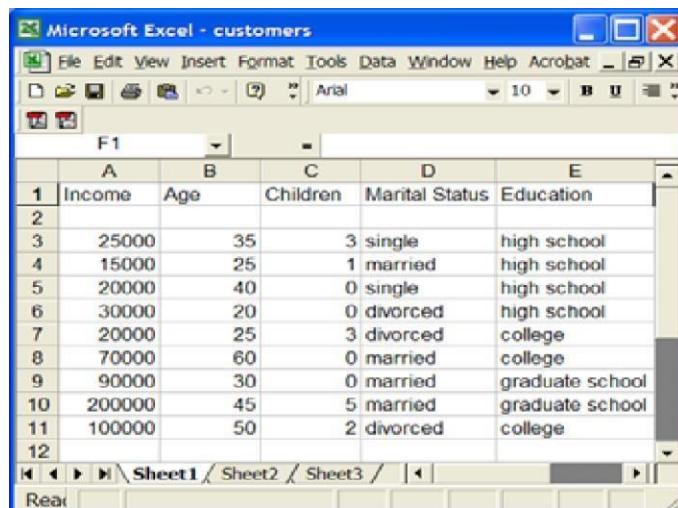


Practical N0-13

Aim: Clustering Analysis.

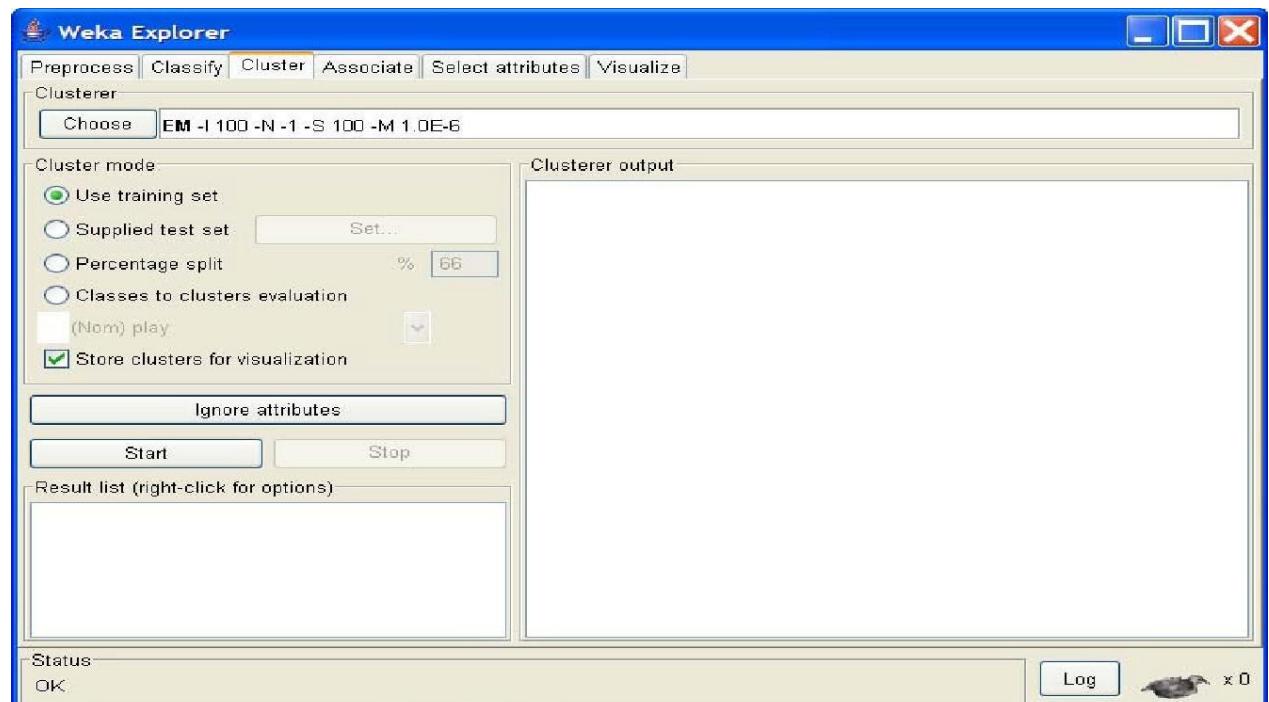
Solution:

We will use customer data [6] that is contained in "customers.arff" file and analyze it with k-means clustering scheme.

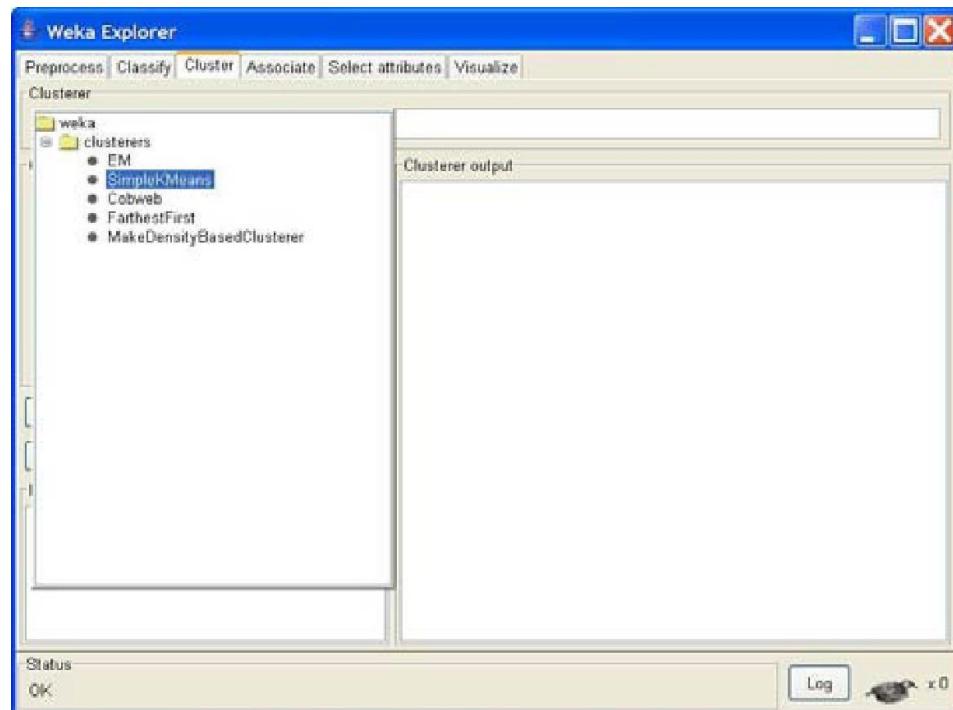
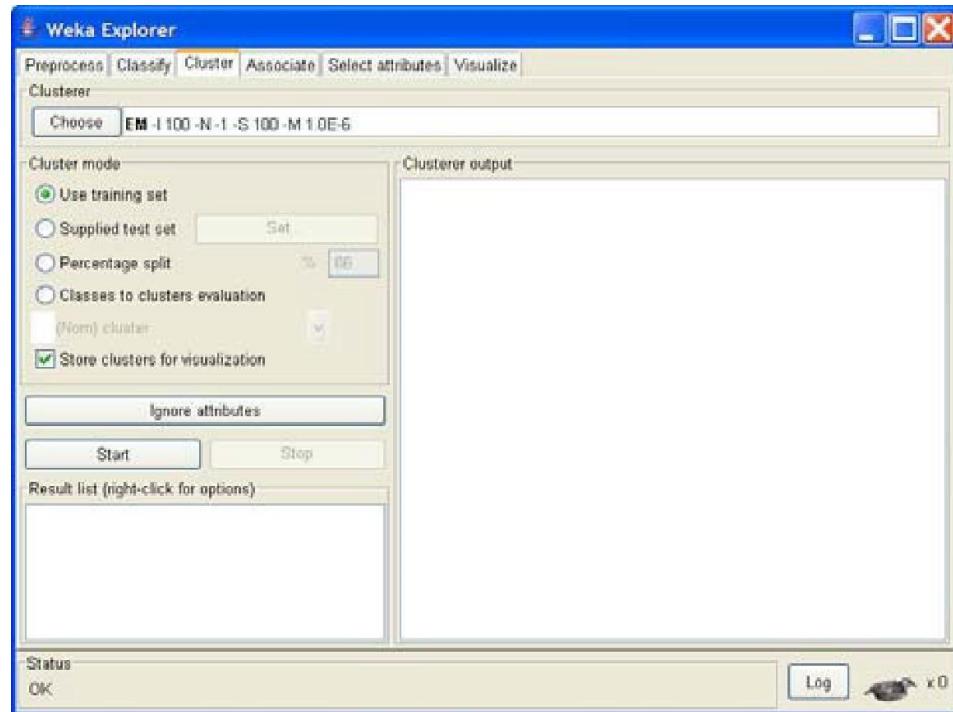


	A	B	C	D	E
1	Income	Age	Children	Marital Status	Education
2					
3	25000	35	3	single	high school
4	15000	25	1	married	high school
5	20000	40	0	single	high school
6	30000	20	0	divorced	high school
7	20000	25	3	divorced	college
8	70000	60	0	married	college
9	90000	30	0	married	graduate school
10	200000	45	5	married	graduate school
11	100000	50	2	divorced	college
12					

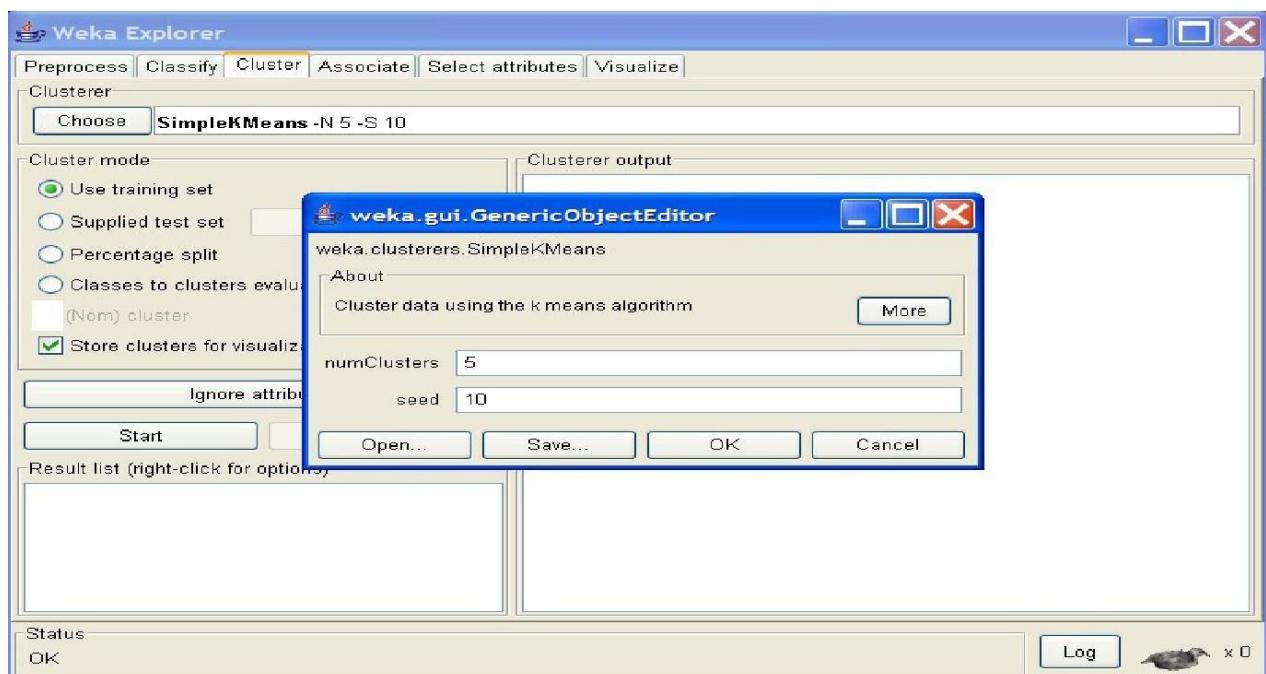
In 'Preprocess' window click on 'Open file...' button and select "customers.arff" file. Click 'Cluster' tab at top of WEKA Explorer window.



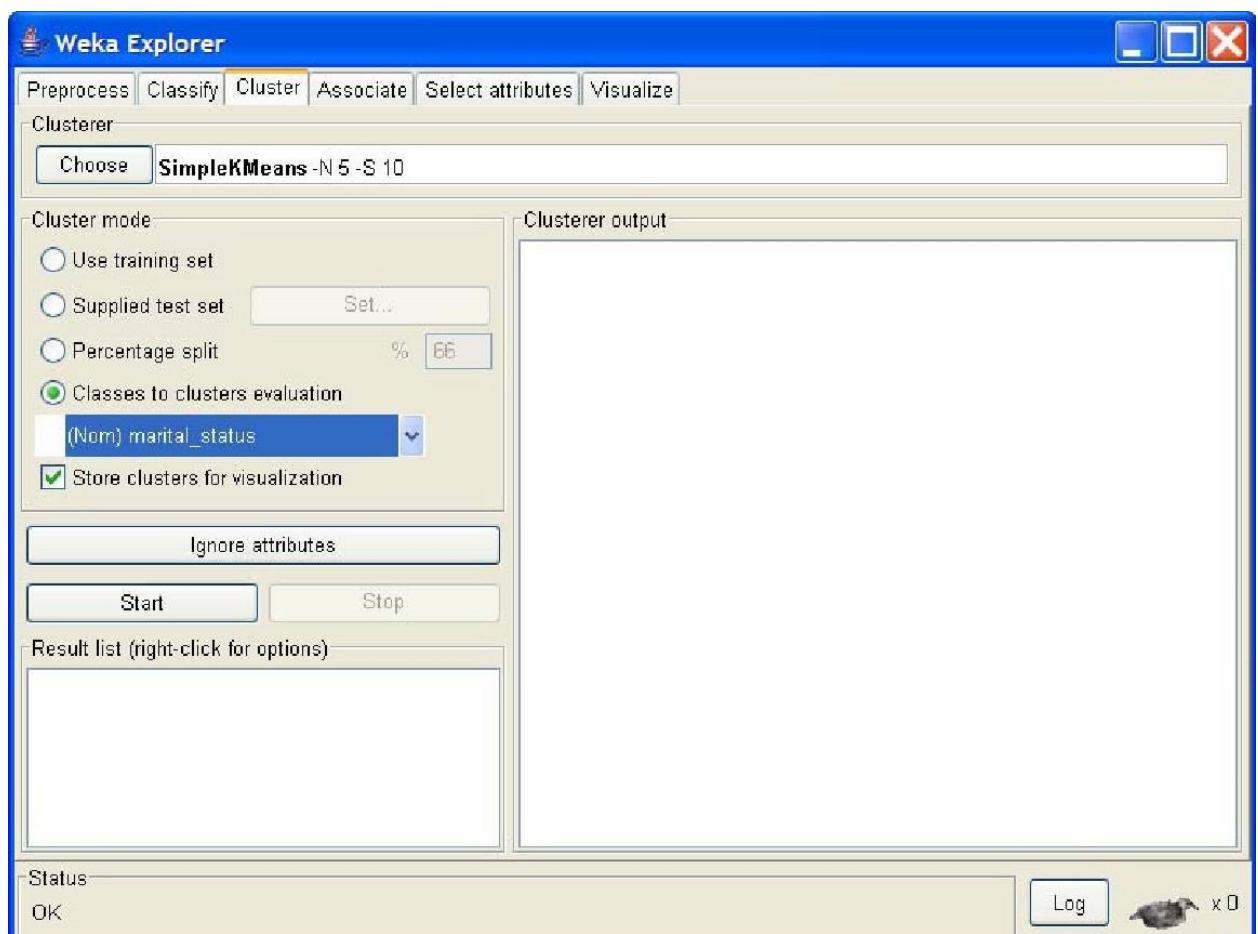
In the 'Clusterer' box click on 'Choose' button. In pull-down menu select WEKA Clusterers, and select the cluster scheme 'SimpleKMeans'. Some implementations of K-means do not allow numerical values for attributes.



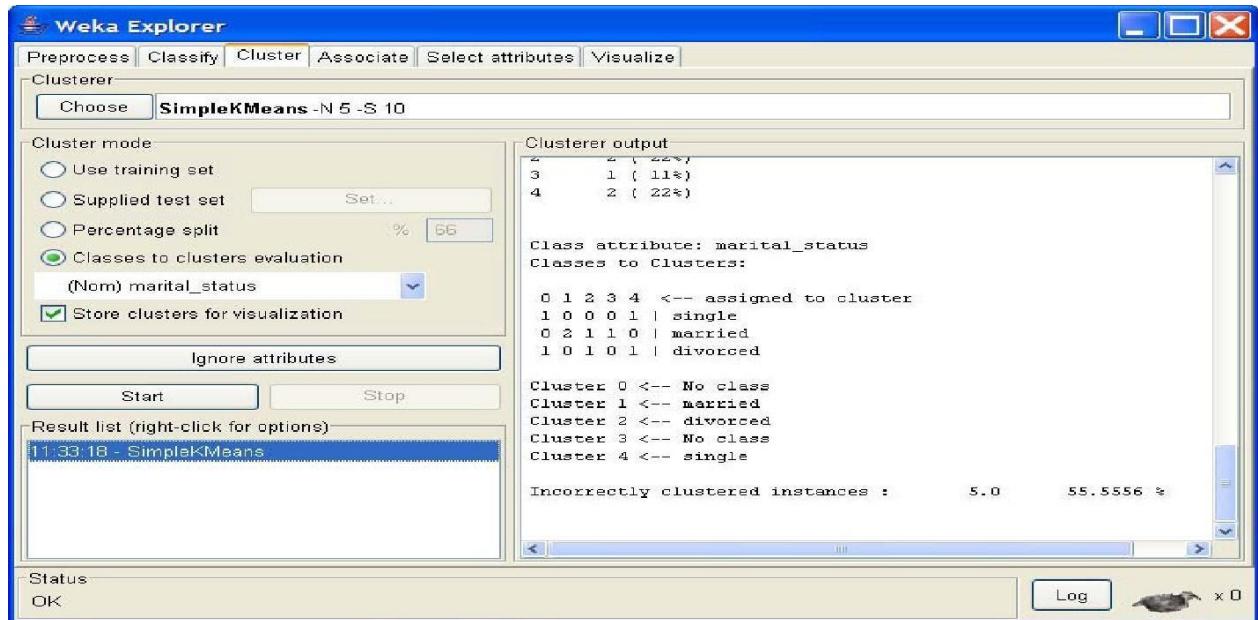
right-click on the algorithm "weak.gui.GenericObjectEditor" comes up to the screen. Set the value in "numClusters" to 5 (instead of default 2) because you have five clusters in your.arff file.



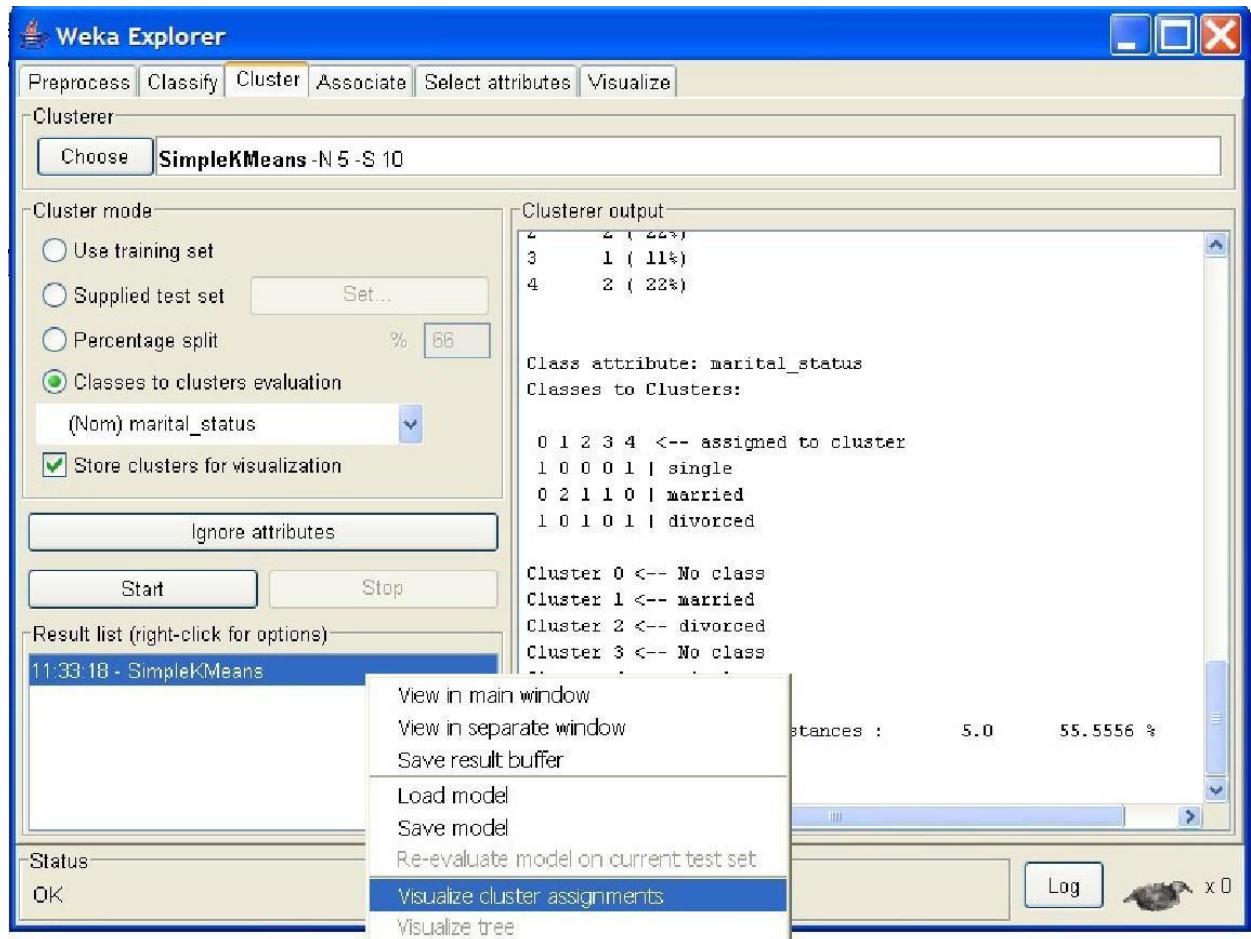
Click on ‘Classes to cluster evaluation’ radio-button in ‘Cluster mode’ box and select ‘marital_status’ in pull-down boxbelow.



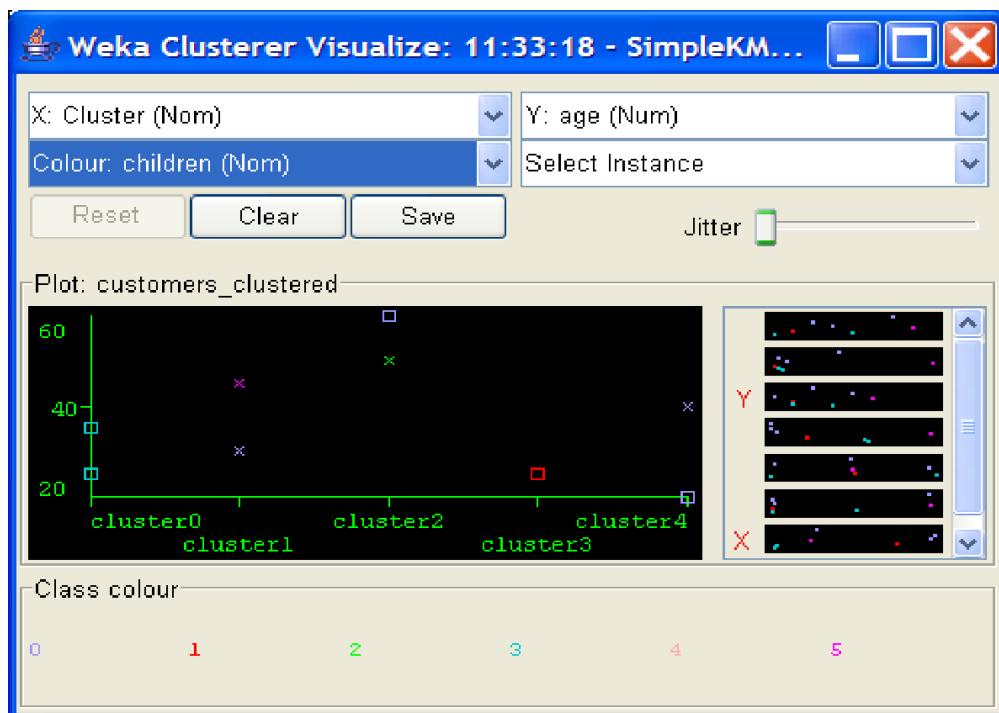
Click on the 'Start' button to execute the algorithm.



Right-click on the entry in the 'Result list' and select 'Visualize cluster assignments' in the pull-down window.



'WekaClustererVisualize' window.



there is a new attribute appeared in the file – ‘cluster’ that was added by WEKA. This attribute represents the clustering done by WEKA.

```

customers_kmeans - Microsoft Word
File Edit View Insert Format Tools Table Window Help Acrobat
Plain Text Courier New 10 100%
Plan Text Favorites Go
relation customers_clustered

@attribute Instance_number numeric
@attribute income numeric
@attribute age numeric
@attribute children {0,1,2,3,4,5}
@attribute marital_status {single,married,divorced}
@attribute education {high_school,college,graduate_school}
@attribute Cluster {cluster0,cluster1,cluster2,cluster3,cluster4}

@data
0,25000,35,3,single,high_school,cluster0
1,15000,25,1,married,high_school,cluster3
2,20000,40,0,single,high_school,cluster4
3,30000,20,0,divorced,high_school,cluster0
4,20000,25,3,divorced,college,cluster0
5,70000,60,0,married,college,cluster2
6,90000,30,0,married,graduate_school,cluster1
7,200000,45,5,married,graduate_school,cluster1
8,100000,50,2,divorced,college,cluster2

```

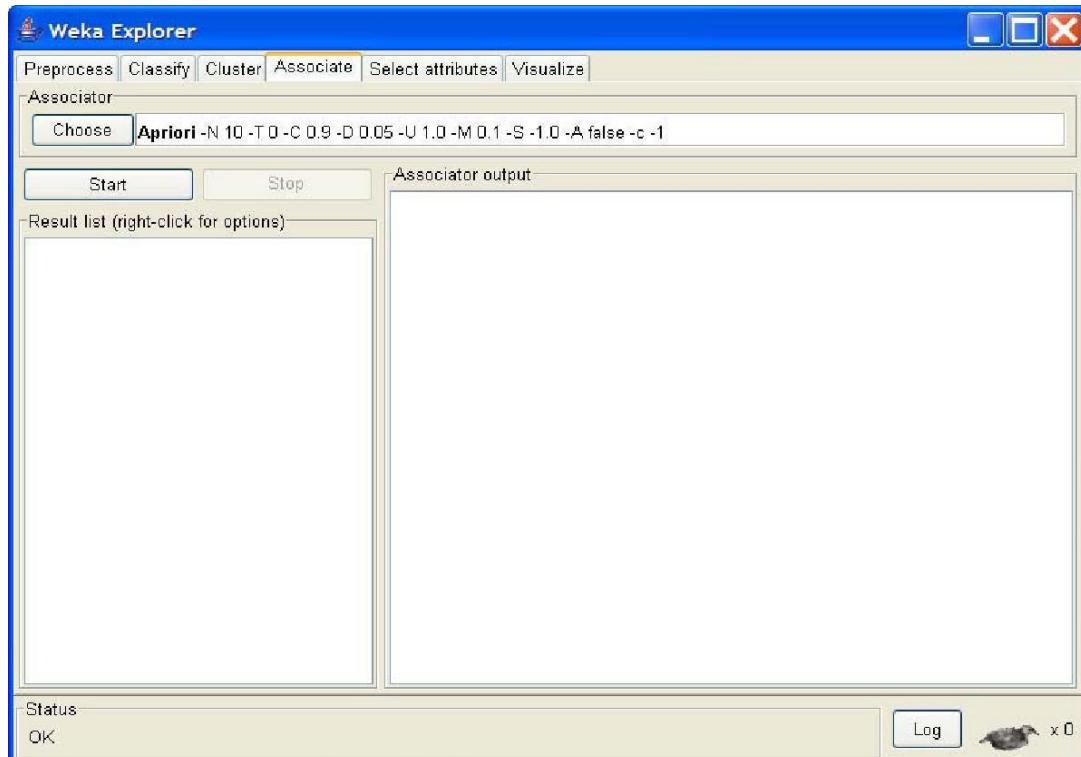
PracticalNo-14

Aim:AssociationRuleMining.

Solution:

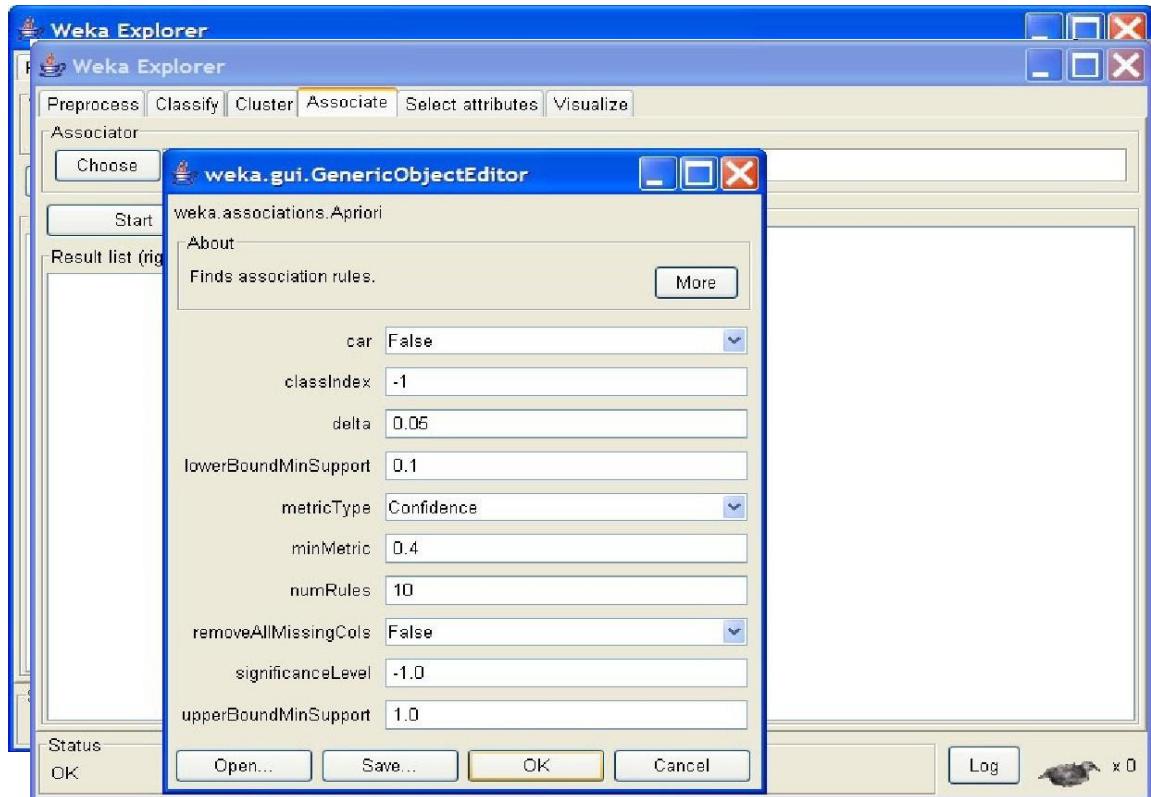
ChoosingAssociationScheme

Click ‘Associate’tab at the top of ‘WEKA Explorer’ window. It brings up interface for the ~~Apriori~~ algorithm.



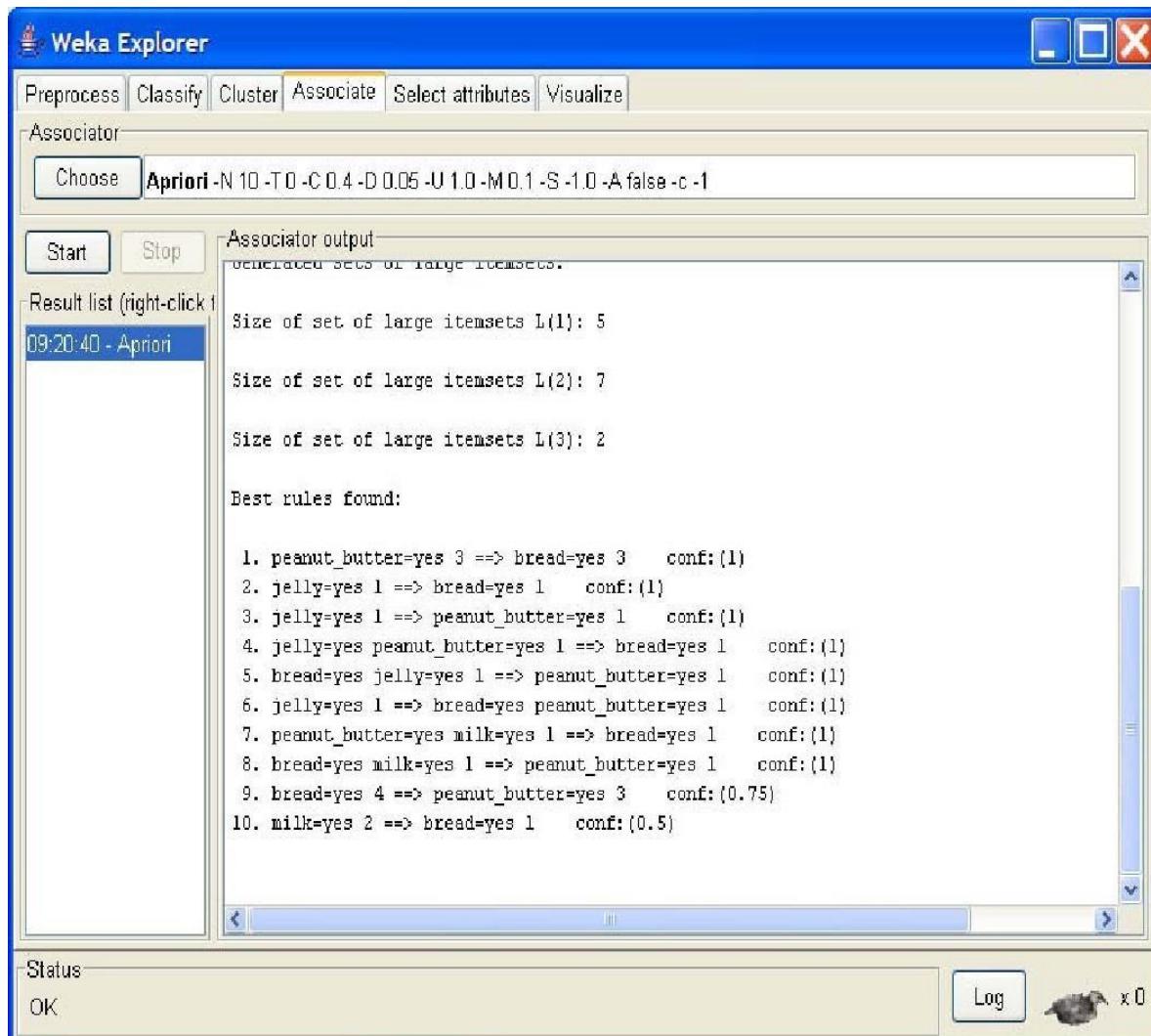
SettingTestOptions

Check the text field in the 'Associator' box at the top of the window



Right-click on the 'Associator' box, 'GenericObjectEditor' appears on your

screen
Click on the 'Start' button to execute the algorithm

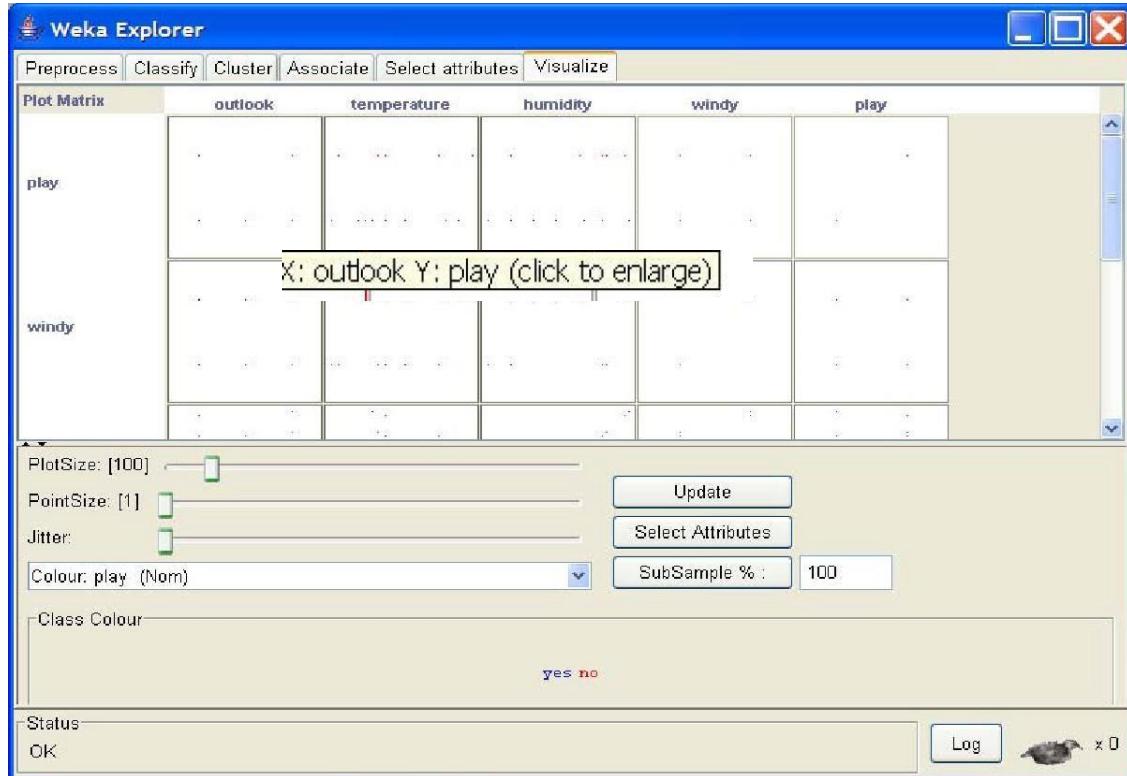


Practical No-15

Aim: Data Visualization

Solution:

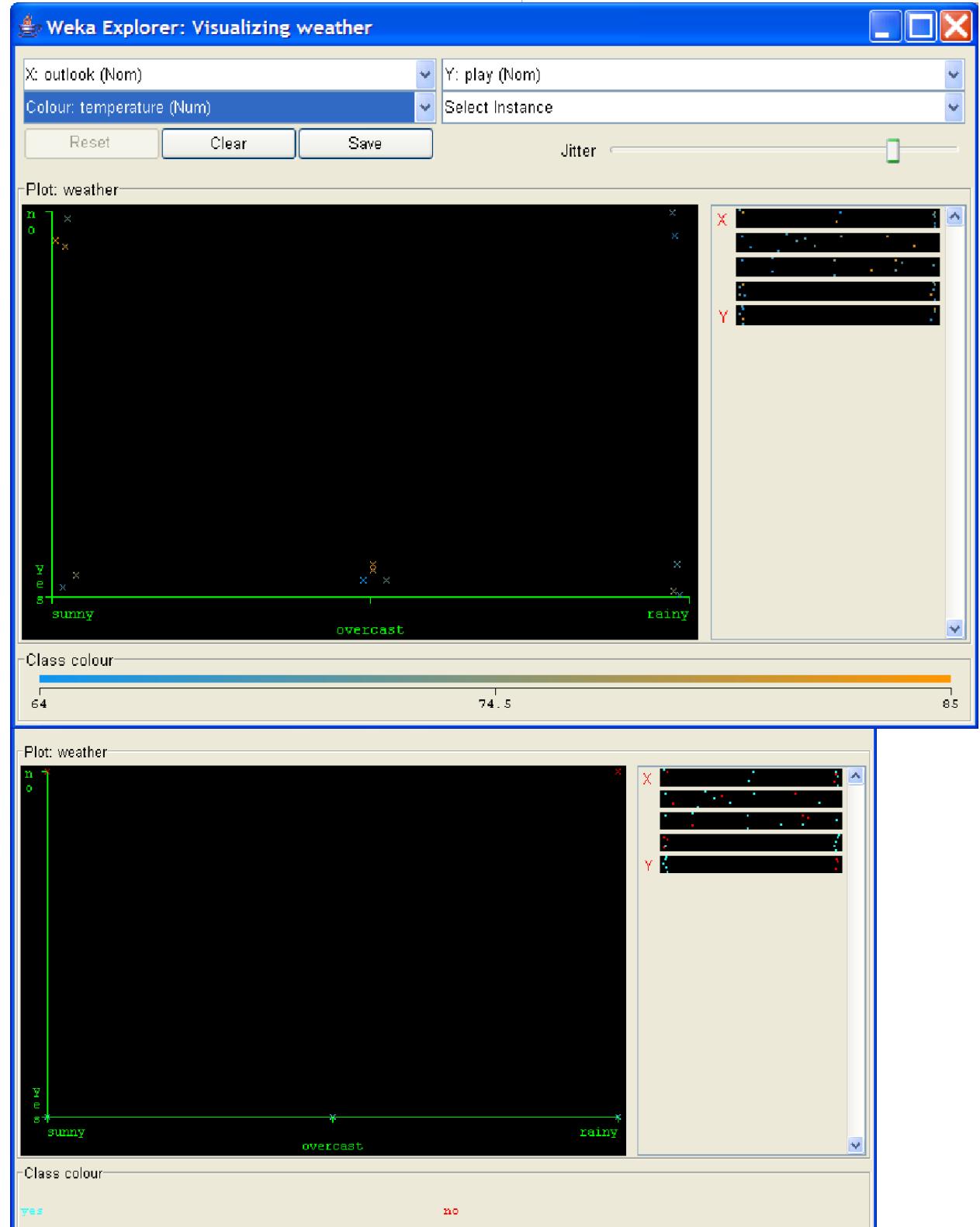
To open Visualizations screen, click 'Visualize' tab.
Select a square that corresponds to the attributes you would like to visualize.



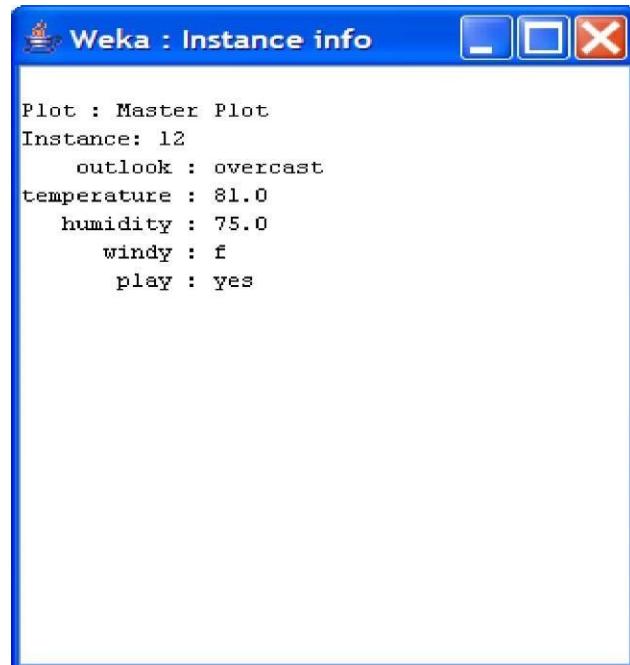
A 'Visualizing weather' window appears on the screen.

Changing the View

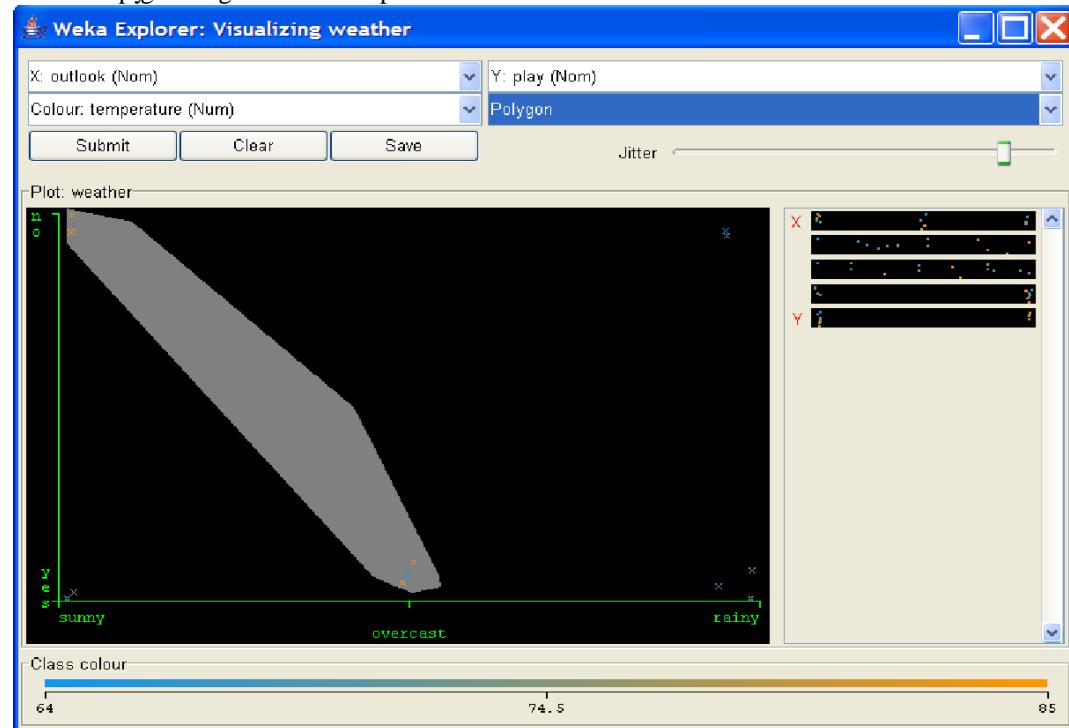
Keeps sliding 'Jitter', a random displacement given to all points in the plot, to the right, until you can spot concentration points



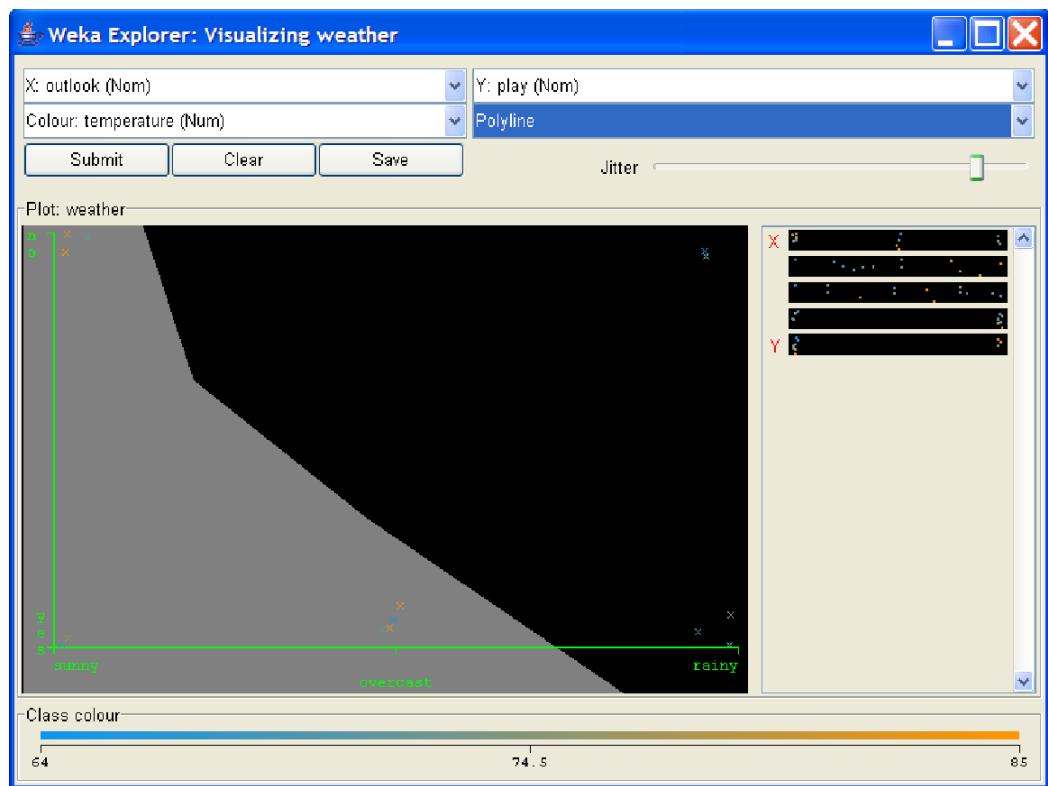
Selecting Instances: Click on an individual data point



3. **Polygon.** You can select several points by building a free-form polygon. Left-click on the graph to add vertices to the polygon and right-click to complete it.



4. **Polyline.** To distinguish the points on one side from the ones on another, you can build a polyline. Left-click on the graph to add vertices to the polyline and right-click to finish.



Rectangle. You can create a rectangle by dragging it around the points

