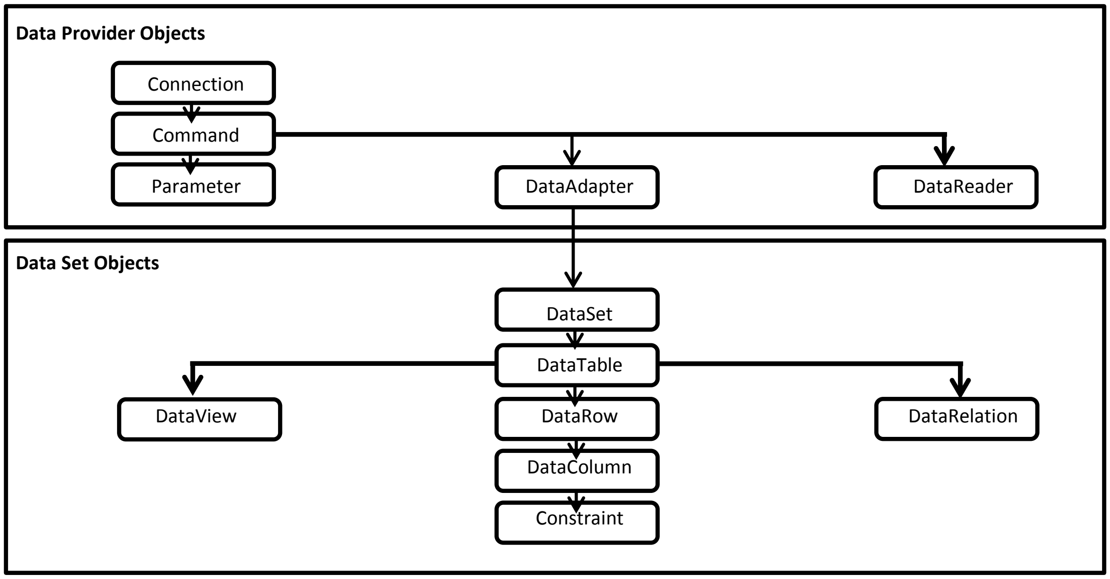
ADO.NET stands for ActiveX Data Objects for .NET framework.

ActiveX Data Object framework is a universal database framework that can be used to various data sources. ADO.NET architecture is divided into two distinct data objects.



ADO.NET is implemented in various namespaces

1. System.Data.Oledb
2. System.Data.Odbc
3. System.Data.Sqlclient
4. System.Data.OracleClient

	Name	Implementations
1.	System.Data.Oledb	**OledbConnection, OledbCommand, OledbDataAdapter, OledbDataReader**
2.	System.Data.Odbc	**OdbcConnection, OdbcCommand, OdbcDataAdapter, OdbcDataReader**
3.	System.Data.Sqlclient	**SqlConnection, SqlCommand, SqlDataAdapter, SqlDataReader**
4.	System.Data.OracleClient	**OracleConnection, OracleCommand, OracleDataAdapter, OracleDataReader**

Comment [S1]: UnManaged Data providers (We cannot collect garbage collection hence it is called “UnManaged Data Provider”.

Comment [S2]: UnManaged Data providers

Database	Provider Name

MS-Access	Microsoft.JET.OLEDB.3.51(Access97)
	Microsoft.JET.OLEDB.4.0(Access-XP or Higher Versions)

SQL Server	SQLOLEDB or SQLOLEDB1
Oracle	MSDAORA.1

1. Data Provider Objects

These are connection oriented and automatically implemented connection pooling. (Automatically disconnects the idle user after a stipulated interval of time in order to make a room for other users to access the database)

1. **Connection**

This object corresponds to a database connection. It automatically implements connection pooling that is disconnects from idle users after a stipulated time interval.
The following are the members of connection objects.

Name	Type	Description
ConnectionState	Property	Determines the state of the connection 1 for open 0 for close
Close()	Method	Closes the connection with the database.
ServerVersion	Property	Returns the current database server version.
Open()	Method	Open the connection to data source to make the successful entity in to the database.
ConnectionString	Property	Specifies the database connection string parameter like server name, username, password etc.
Database	Property	Gets the name of the database.(Works in SQL only)
DataSource	Property	Gets the server name or filename

2. **Command**

This object corresponds to an SQL statement (Insert, Update, Delete, and Select) that can be executed through connection object.

Name	Type	Description
Cancel()	Method	Cancels the command execution.
CommandText	Property	Gets the SQL statement that can be Insert, Update, Select and Delete.
CommandType	Property	Specifies the type of the command to be executed. It can be any one of the following. • CommandType.StoreProcedure Execute the store procedure. • CommandType.TableDirect Execute the given table name only for Microsoft products like SQLServer, Access. • CommandType.TableDirect(DEFAULT) Executes the valid SQL statements.
Connection	Property	Specifies the connection object to which the command is to be executed.
ExecuteNonQuery()	Method	Execute the SQL statement that do not result return. (Delete, Insert, Update)
ExecuteReader()	Method	Execute the SQL statement that return results.(Select)
ExecuteScalar()	Method	Execute the SQL statement and returns the data value present in the 1st column of 1st row in the given table.
Parameter	Property	Gets the collection of parameters for the command object.

3. **DataAdapter**

This object provides essential link between the data provider and data set objects.
The members of data adapter are as follows.

Name	Type	Description
Fill(DataSet,TableName)	Method	Fills the data into a dataset by retrieving data from database table.
SelectCommand	Property	Specifies the source SQL select command.

UpdateCommand	Property	Specifies the source SQL update command.
DeleteCommand	Property	Specifies the source SQL delete command.
InsertCommand	Property	Specifies the source SQL insert command.

4. **DataReader**

This object corresponds to read only, forward only record set associated with a database table.
The members of data adapter are as follows.

Name	Type	Description
Close()	Method	Closes the DataReader.
FieldCount	Property	Returns the number of columns in the table.
GetName(index)	Method	Returns the name of the column for the given index.
GetValue(index)	Method	Returns the data value for the given column index.
HasRows	Property	Determines if the DataReader contains any rows.
Read()	Method	Advances the DataReader to the next record in the database table

2. **Data Set Objects**

These objects are disconnected database object that corresponds to a data source. These objects are available under **System. Data** namespace.

1. **DataSet**

This object corresponds to a set of database tables or relations that exist in a particular database.

Name	Type	Description
Name	Method	Commits the changes made to the dataset.
AcceptChanges()	Method	Clears all the data present in the data set.
Clear()	Method	Clears all the data present in the dataset.
GetXML()	Method	Returns the XML data representation of the data set.
GetXMLSchema()	Method	Returns the XML schema representation of the data set.
HasChanges()	Method	Gets a value indicating if the dataset has any changes.
Load()	Method	Loads values from a data source.
ReadXML()	Method	Reads an XML file into the dataset using XML reader.
ReadXMLSchema()	Method	Reads an XML schema into the dataset.
RejectChanges	Property	Rollbacks all the changes may to the data set.
Relations	Property	Gets the collection of relations on this data set.
Table	Property	Get the collection of table on this data set.
WriteXML()	Method	Writes the data set out as an XML file.
WriteXMLSchema()	Method	Writes the dataset out an XMLSchema file.
GetChanges()	Method	Gets a copy of changes made to the data set since the last load of data.

2. **DataTable**

This object corresponds to a table associated with a data source (MS Access, Any Databases)

Name	Type	Description
AcceptChanges()	Method	Clears all the data present in the datatable.
ChildRelation	Property	Gets the collection of child relation on this table.

Clear()	Method	Clear all the data present in the datatable.
Load()	Method	Loads values from a datatable.
GetChangess()	Method	Gets the copy of changes made to the data table.
NewRow()	Method	Creates a new blank row.
ParentRelations	Property	Gets the collection of parent relations on this table.
ReadXML()	Method	Reads an XML file into the datatable.
ReadXMLSchema()	Method	Reads an XML schema into the datatable.
Rows	Property	Gets the collection of rows on this table.
TableName	Property	Returns the name of the table.
WriteXML()	Method	Writes the datatable out as an XML file.
WriteXMLSchema()	Method	Writes the datatable out an XMLSchema file.

3. DataRow

This object corresponds to a row in a database table.

Name	Type	Description
AcceptChanges()	Method	Commits the changes made to the datarow.
BeginEdit()	Method	Starts a new edit operation on the datarow.
CancelEdit	Method	Cancels the current edit operation in progress.
Delete()	Method	Deletes the row.
GetChildRows()	Method	Gets the collection of child row based on the data relation name.
GetParentRow()	Method	Gets the parent row based on the data relation name.
GetParentRow()	Method	Gets the collection of parent rows based on the data relation name.
IsNull()	Method	Determines if the row contains any null values.
Item	Property	Gets or Sets the data stored in the column specified by its index.

4. DataColumn

This object corresponds to a field or a column in a database table.

Name	Type	Description
AllowDBNull	Property	Determines if the null values are allowed for the column.
ColumnName	Property	Returns the name of the column.
DataType	Property	Returns the datatype of the column.
Table	Property	Returns the name of the table associated with the column.
Unique	Property	Gets or sets a value that determines if the value for each row in column is unique.

5. DataView

This object corresponds a view or read only record set that can be used to manipulate with the dataview.

Name	Type	Description
AddNew()	Method	Adds a new row.
Count	Propery	Returns the number of row on the data view.
Delete()	Methd	Delete the row at the specified index.
Find()	Method	Finds a row in the data view.
Item	Property	Gets a row of data from the specified table.
Table	Property	Gets the name of the table.

6. DataRelation

This object corresponds to a logical relationship that corresponds to master and detail table in a data table.

Name	Type	Description

ChildColumns	Property	Gets the child columns
ChildKeyConstraint	Property	Gets the foreign key constraint.
ChildTable	Property	Gets the name of the child table.
DataSet	Property	Returns the name of the dataset to which this relation these associated.
ParentColumns	Property	Gets an array of parent columns on this table.
ParentKeyConstraint	Property	Gets the primary key or unique constraints for the column.
ParentTable	Property	Gets the name of the parent table.
RelationName	Property	Gets the name of the relation.

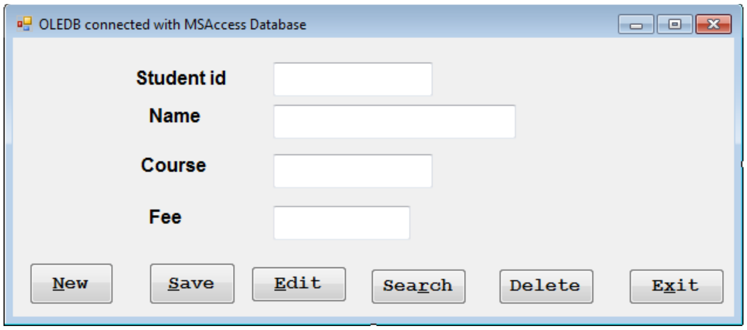
7. **Constraint**

This object corresponds to constraint define for a column in the database table.

Name	Type	Description
Name	Type	Gets the name of the constraint.
ConstraintName	Property	Gets the name of the constraint.
Table	Property	Gets the name of the table to which this constraint associated.

Create an interface to connect to MS Access using OLEDB

Step1: Create an interface following



Step 2:

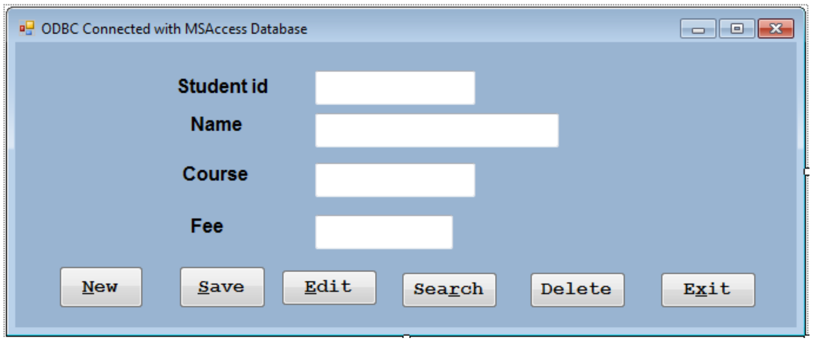
```
OleDbConnection con = new OleDbConnection();
      OleDbCommand cmd = new OleDbCommand();
      OleDbDataReader dr;

public void connectDB()
      {
         try
         {
            //con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" + @"E:\C#
PROJECT\DataBase\Student_MSA.mdb;User Id=Admin; password=;";
            String path = Application.StartupPath;
            con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" + path + @"\Student_MSA.mdb;User Id=Admin;
password=;";
            con.Open();
            cmd.Connection = con;
         }
         catch (Exception ex)
         {
            MessageBox.Show("The error is.." + ex.Message);
         }
      }
//clear
         foreach (Control ctl in this.Controls)
            if (ctl is TextBox)
               ctl.Text = " ";
         textBox1.Focus();
public void TextColor()
      {
         //TextColor
         textBox1.BackColor = Color.Aqua;
         textBox1.ForeColor = Color.Black;
         textBox2.BackColor = Color.Aqua;
         textBox2.ForeColor = Color.Black;
         textBox3.BackColor = Color.Aqua;
         textBox3.ForeColor = Color.Black;
         textBox4.BackColor = Color.Aqua;
         textBox4.ForeColor = Color.Black;

      }
//New
         clearFields();
         TextColor();
//Save
         try
         {
            connectDB();
            cmd.CommandText = "Insert into Student(sid,sname,course,fee)values(" + textBox1.Text + ",'" + textBox2.Text + "','" +
textBox3.Text + "'," + textBox4.Text + " );";
            cmd.ExecuteNonQuery();
            MessageBox.Show(" ONE STUDENT RECORD IS SAVED", "SAVE", MessageBoxButtons.OK, MessageBoxIcon.Information);
         }
         catch (Exception ex)
         {
            MessageBox.Show("The error is..." + ex.Message);
         }
         finally
```

```
            {
                con.Close();
                clearFields();
            }
  //Edit
            try
            {
                connectDB();
                cmd.CommandText = "update Student set sid=" + textBox1.Text + ",sname='" + textBox2.Text + "',course='" + textBox3.Text
+ "',fee=" + textBox4.Text + " where sid=" + textBox1.Text + ";";
                cmd.ExecuteNonQuery();
                MessageBox.Show(" ONE STUDENT INFO MODIFIED", "Modify", MessageBoxButtons.OK, MessageBoxIcon.Information);
            }
            catch (Exception ex)
            {
                MessageBox.Show("The(Edit) error is.." + ex.Message);
            }
            finally
            {
                con.Close();
                clearFields();
            }

  //Search
            try
            {
                connectDB();
                cmd.CommandText = "Select * from Student where sid=" + textBox1.Text;
                dr = cmd.ExecuteReader();
                dr.Read();
                if (dr.HasRows == true)
                {
                    textBox1.Text = dr["sid"].ToString();
                    textBox2.Text = dr["sname"].ToString();
                    textBox3.Text = dr["course"].ToString();
                    textBox4.Text = dr["fee"].ToString();
                }
                else
                {
                    MessageBox.Show(" No Such info available", "Find Failed", MessageBoxButtons.OK, MessageBoxIcon.Information);
                    clearFields();
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show("The (Search) Error is.." + ex.Message);
            }
            finally
            {
                con.Close();
            }
//DELETE
            DialogResult response;
            response = MessageBox.Show("Are you sure?", "Delete student", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
            if (response == DialogResult.Yes)
            {
```

```
        try
        {
            connectDB();
            cmd.CommandText = "Delete from student where sid=" + int.Parse(textBox1.Text);
            cmd.ExecuteNonQuery();
            MessageBox.Show(" ONE STUDENT INFO DELETED", "DELETE", MessageBoxButtons.OK, MessageBoxIcon.Information);
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.Message);
        }
        finally
        {
            con.Close();
            clearFields();
        }
     }
 //Exit
     this.Close();
```

Create an Interface to connect to MS Access using ODBC

Step1: Create an interface following



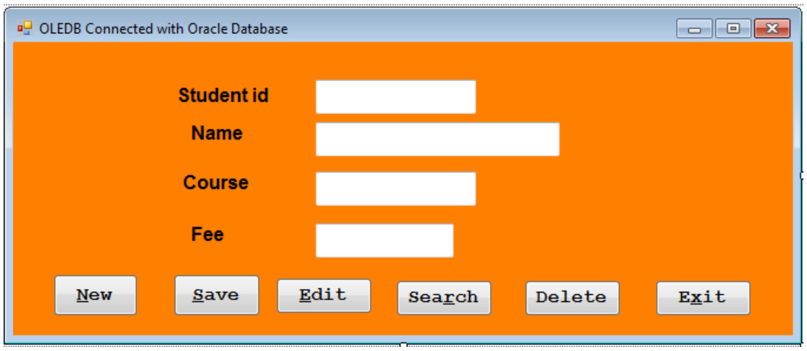
Step 2:

```
//Header
OdbcConnection con = new OdbcConnection();
OdbcCommand cmd = new OdbcCommand();
OdbcDataReader dr;

//Method at global section
public void connectDB()
        {
            try
            {
                con.ConnectionString = "dsn=msaccessdsn;User Id=Admin; password=;";
                //String path = Application.StartupPath;
                //con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" + path + @"\Student_MSA.mdb;User
Id=Admin; password=;";
                con.Open();
                cmd.Connection = con;
            }
            catch (Exception ex)
            {
                MessageBox.Show("The error is.." + ex.Message);
            }
        }
        public void clearFields()
        {
            foreach (Control ctl in this.Controls)
                if (ctl is TextBox)
                    ctl.Text = " ";
            textBox1.Focus();
        }
        public void TextColor()
        {
            textBox1.BackColor = Color.Aqua;
            textBox1.ForeColor = Color.Black;
            textBox2.BackColor = Color.Aqua;
            textBox2.ForeColor = Color.Black;
            textBox3.BackColor = Color.Aqua;
            textBox3.ForeColor = Color.Black;
            textBox4.BackColor = Color.Aqua;
            textBox4.ForeColor = Color.Black;

        }
//New
            clearFields();
            TextColor();
//Save
            try
            {
                connectDB();
                cmd.CommandText = "Insert into Student(sid,sname,course,fee)values(" + textBox1.Text + ",'" + textBox2.Text + "','" +
textBox3.Text + "'," + textBox4.Text + " );";
                cmd.ExecuteNonQuery();
                MessageBox.Show(" ONE STUDENT RECORD IS SAVED", "SAVE", MessageBoxButtons.OK, MessageBoxIcon.Information);
            }
            catch (Exception ex)
            {
                MessageBox.Show("The error is..." + ex.Message);
            }
            finally
            {
                con.Close();
                clearFields();
            }
//Edit
            try
            {
                connectDB();
                cmd.CommandText = "update Student set sid=" + textBox1.Text + ",sname='" + textBox2.Text + "',course='" + textBox3.Text
+ "',fee=" + textBox4.Text + " where sid=" + textBox1.Text + ";";
                cmd.ExecuteNonQuery();
                MessageBox.Show(" ONE STUDENT INFO MODIFIED", "Modify", MessageBoxButtons.OK, MessageBoxIcon.Information);
            }
            catch (Exception ex)
            {
                MessageBox.Show("The(Edit) error is.." + ex.Message);
            }
            finally
            {
                con.Close();
                clearFields();
            }
//Search
            try
            {
                connectDB();
                cmd.CommandText = "Select * from Student where sid=" + textBox1.Text;
                dr = cmd.ExecuteReader();
                dr.Read();
                if (dr.HasRows == true)
                {
                    textBox1.Text = dr["sid"].ToString();
                    textBox2.Text = dr["sname"].ToString();
                    textBox3.Text = dr["course"].ToString();
                    textBox4.Text = dr["fee"].ToString();
                }
                else
                {
                    MessageBox.Show(" No Such info available", "Find Failed", MessageBoxButtons.OK, MessageBoxIcon.Information);
                    clearFields();
                }
     //DELETE
            DialogResult response;
            response = MessageBox.Show("Are you sure?", "Delete student", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
            if (response == DialogResult.Yes)
            {
                try
                {
                    connectDB();
                    cmd.CommandText = "Delete from student where sid=" + int.Parse(textBox1.Text);
                    cmd.ExecuteNonQuery();
                    MessageBox.Show(" ONE STUDENT INFO DELETED", "DELETE", MessageBoxButtons.OK, MessageBoxIcon.Information);
                }
                catch (Exception ex)
                {
                   MessageBox.Show(ex.Message);
                }
                finally
                {
                   con.Close();
                   clearFields();
                }
             }
//Exit
             this.Close();
```

Create an interface to connect to oracle using OLEDB

```
 create table student
 (
        sid       varchar2(15)constraint sid_pk primary key,constraint sid_ck check(sid like 'S%'),
        sname   varchar2(20),
        course   varchar(8) check(course in('M.C.A','M.B.A','M.Sc')),
        fee        number(10,2));
```

Step1: Create an interface following



Step2:

```
        OleDbConnection con = new OleDbConnection();
        OleDbCommand cmd = new OleDbCommand();
        OleDbDataReader dr;
        public void connectDB()
        {
            try
            {
                con.ConnectionString = "Provider=MSDAORA.1;User Id=nagendra;Password=password";
                con.Open();
                cmd.Connection = con;
            }
            catch (Exception ex)
            {
                MessageBox.Show("The (ConnectDb)serror is.." + ex.Message);
            }
        }
        public void clearFields()
        {
            foreach (Control ctl in this.Controls)
                if (ctl is TextBox)
                    ctl.Text = " ";
            textBox1.Focus();
        }
        public void TextColor()
        {
            textBox1.BackColor = Color.Aqua;
            textBox1.ForeColor = Color.Black;
            textBox2.BackColor = Color.Aqua;
            textBox2.ForeColor = Color.Black;
            textBox3.BackColor = Color.Aqua;
            textBox3.ForeColor = Color.Black;
            textBox4.BackColor = Color.Aqua;
            textBox4.ForeColor = Color.Black;

        }
//New
            clearFields();
            TextColor();
//Save
            try
            {
                connectDB();
                cmd.CommandText ="Insert into student values('"+ textBox1.Text + "','" + textBox2.Text + "','" + textBox3.Text + "'," +
textBox4.Text + ")";
                cmd.ExecuteNonQuery();
                MessageBox.Show("ONE STUDENT RECORD IS SAVED", "SAVE", MessageBoxButtons.OK, MessageBoxIcon.Information);
            }
            catch (Exception ex)
            {
                MessageBox.Show("The error(save) is..." + ex.Message);
            }
            finally
            {
                con.Close();
                clearFields();
            }
```

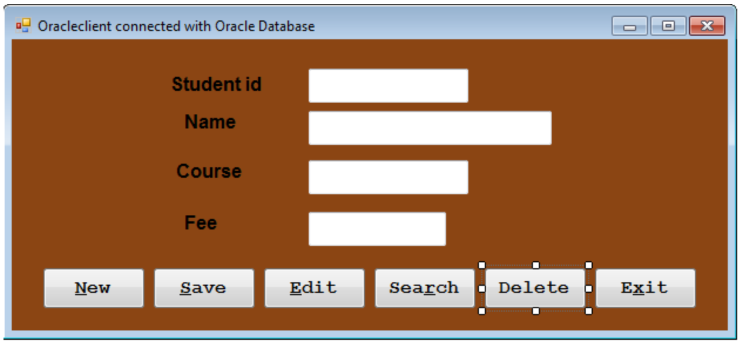
```
 //Exit
      this.Close();

//Search
      try
      {
        connectDB();
        cmd.CommandText = "Select * from Student where sid='" + textBox1.Text+"'";
        dr = cmd.ExecuteReader();
        dr.Read();
        if (dr.HasRows == true)
        {
          textBox1.Text = dr["sid"].ToString();
          textBox2.Text = dr["sname"].ToString();
          textBox3.Text = dr["course"].ToString();
          textBox4.Text = dr["fee"].ToString();
        }
        else
        {
          MessageBox.Show(" No Such info available", "Find Failed", MessageBoxButtons.OK, MessageBoxIcon.Information);
          clearFields();
        }
      }
      catch (Exception ex)
      {
        MessageBox.Show("The (Search) Error is.." + ex.Message);
      }
      finally
      {
        con.Close();
      }
//Edit
      try
      {
        connectDB();
        cmd.CommandText = "update student set sid='" + textBox1.Text + "',sname='" + textBox2.Text + "',course='" +
textBox3.Text + "',fee=" + textBox4.Text + " where sid='" + textBox1.Text+"'";
        cmd.ExecuteNonQuery();
        MessageBox.Show(" ONE STUDENT INFO MODIFIED", "Modify", MessageBoxButtons.OK, MessageBoxIcon.Information);
      }
      catch (Exception ex)
      {
        MessageBox.Show("The(Edit) error is.." + ex.Message);
      }
      finally
      {
        con.Close();
        clearFields();
      }
//DELETE
      DialogResult response;
      response = MessageBox.Show("Are you sure?", "Delete student", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
      if (response == DialogResult.Yes)
      {
        try
        {
            connectDB();
            //cmd.CommandText = "Delete from student where sid=" + int.Parse(textBox1.Text)+"'";
            cmd.CommandText = "Delete from student where sid='" + textBox1.Text + "'";
            cmd.ExecuteNonQuery();
            MessageBox.Show(" ONE STUDENT INFO DELETED", "DELETE", MessageBoxButtons.OK, MessageBoxIcon.Information);
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.Message);
        }
        finally
        {
            con.Close();
            clearFields();
        }
    }
```

Create an interface to connect to oracle using System.Data.OracleClient

Note: The System.Data.OracleClient namespace will not be available by default in the windows application. It should be added to the project explicitly by using project> Add Reference in that select System.Data.OracleClient and click ok button.

```
create table student
(
        sid         varchar2(15)constraint sid_pk primary key,constraint sid_ck check(sid like 'S%'),
        sname    varchar2(20),
        course    varchar(8) check(course in('M.C.A','M.B.A','M.Sc')),
        fee         number(10,2));
```

Step1: Create an interface following

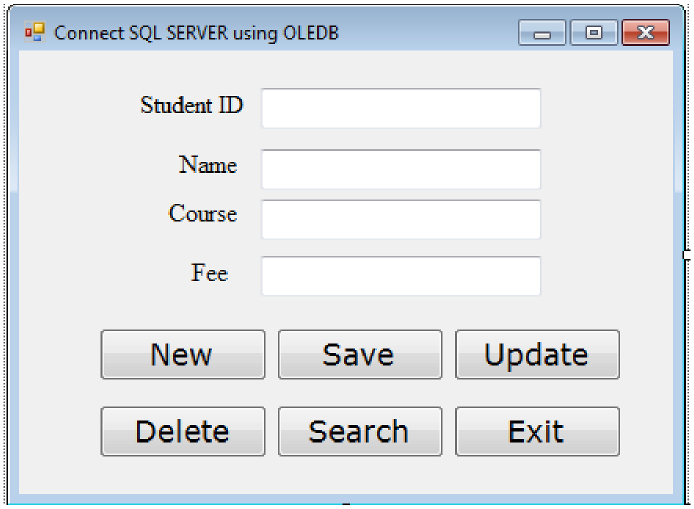


Step 2:

```
OracleConnection con = new OracleConnection();
OracleCommand cmd = new OracleCommand();
OracleDataReader dr;

public void connectDB()
        {
            try
            {
                con.ConnectionString = "Server= ORCL; User Id= nagendra; Password=password;";
                con.Open();
                cmd.Connection = con;
            }
            catch (Exception ex)
            {
                MessageBox.Show("The (ConnectDb)serror is.." + ex.Message);
            }
        }
        public void clearFields()
        {
            foreach (Control ctl in this.Controls)
                if (ctl is TextBox)
                    ctl.Text = " ";
            textBox1.Focus();
        }
        public void TextColor()
        {
            textBox1.BackColor = Color.Aqua;
            textBox1.ForeColor = Color.Black;
            textBox2.BackColor = Color.Aqua;
            textBox2.ForeColor = Color.Black;
            textBox3.BackColor = Color.Aqua;
            textBox3.ForeColor = Color.Black;
            textBox4.BackColor = Color.Aqua;
            textBox4.ForeColor = Color.Black;

        }
//New
            clearFields();
            TextColor();
//Save
            try
            {
                connectDB();
                cmd.CommandText = "Insert into student values('" + textBox1.Text + "','" + textBox2.Text + "','" + textBox3.Text + "'," +
textBox4.Text + ")";
                cmd.ExecuteNonQuery();
                MessageBox.Show("ONE STUDENT RECORD IS SAVED", "SAVE", MessageBoxButtons.OK, MessageBoxIcon.Information);
            }
            catch (Exception ex)
            {
                MessageBox.Show("The error(save) is..." + ex.Message);
            }
            finally
            {
                con.Close();
                clearFields();
            }
//close
            this.Close();
//Edit
            try
            {
                connectDB();
                cmd.CommandText = "update student set sid='" + textBox1.Text + "',sname='" + textBox2.Text + "',course='" +
textBox3.Text + "',fee=" + textBox4.Text + " where sid='" + textBox1.Text + "'";
                cmd.ExecuteNonQuery();
                MessageBox.Show(" ONE STUDENT INFO MODIFIED", "Modify", MessageBoxButtons.OK, MessageBoxIcon.Information);
            }
            catch (Exception ex)
            {
                MessageBox.Show("The(Edit) error is.." + ex.Message);
            }
            finally
            {
                con.Close();
                clearFields();
            }
//Search
            try
            {
                connectDB();
                cmd.CommandText = "Select * from Student where sid='" + textBox1.Text + "'";
                dr = cmd.ExecuteReader();
                dr.Read();
                if (dr.HasRows == true)
                {
                    textBox1.Text = dr["sid"].ToString();
                    textBox2.Text = dr["sname"].ToString();
                    textBox3.Text = dr["course"].ToString();
                    textBox4.Text = dr["fee"].ToString();
                }
                else
                {
                    MessageBox.Show(" No Such info available", "Find Failed", MessageBoxButtons.OK, MessageBoxIcon.Information);
                    clearFields();
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show("The (Search) Error is.." + ex.Message);
            }
            finally
            {
                con.Close();
            }
//DELETE
            DialogResult response;
            response = MessageBox.Show("Are you sure?", "Delete student", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
            if (response == DialogResult.Yes)
            {
                try
                {
```

```
        connectDB();
        //cmd.CommandText = "Delete from student where sid=" + int.Parse(textBox1.Text)+"'";
        cmd.CommandText = "Delete from student where sid='" + textBox1.Text + "'";
        cmd.ExecuteNonQuery();
        MessageBox.Show(" ONE STUDENT INFO DELETED", "DELETE", MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
    finally
    {
        con.Close();
        clearFields();
    }
}
```

Create an interface to connect to sql server using OLEDB

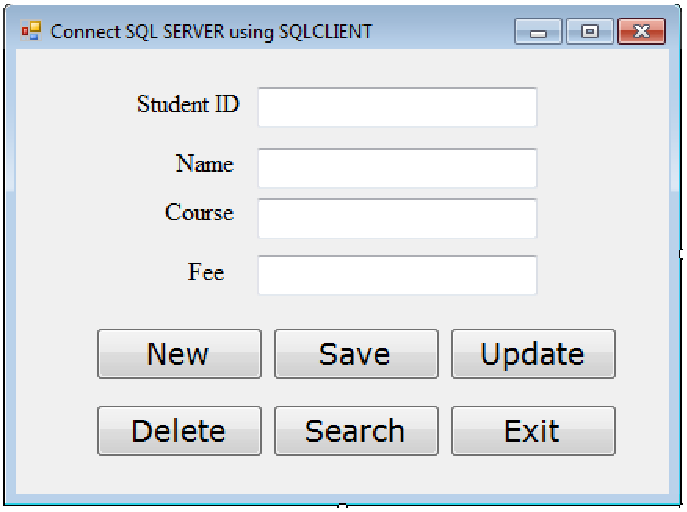


```
--Create login id and with password.
CREATE LOGIN nagendra
    WITH PASSWORD = 'password';
GO

-- Creates a database user for the login created above.
CREATE USER StudentDB FOR LOGIN nagendra;
GO


create table student
```

```
(
      sid   int,
      sname varchar(20),
      course      varchar(8),
      fee   float
  );
```

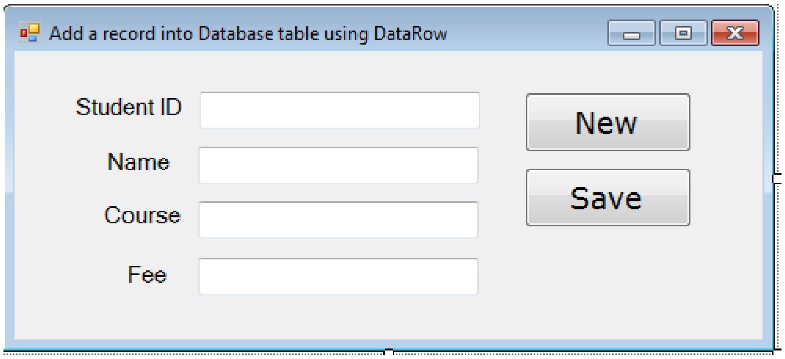
```
OleDbConnection con = new OleDbConnection();
        OleDbCommand cmd = new OleDbCommand();
        OleDbDataReader dr;
public void connectDB()
        {
            try
            {
                con.ConnectionString = "Provider=MSDAORA.1;User Id=sss;Password=password";
                con.Open();
                cmd.Connection = con;
            }
            catch (Exception ex)
            {
                MessageBox.Show("The (ConnectDb)serror is.." + ex.Message);
            }
        }
        public void clearFields()
        {
            foreach (Control ctl in this.Controls)
                if (ctl is TextBox)
                    ctl.Text = " ";
            textBox1.Focus();
        }
        public void TextColor()
        {
            textBox1.BackColor = Color.Aqua;
            textBox1.ForeColor = Color.Black;
            textBox2.BackColor = Color.Aqua;
            textBox2.ForeColor = Color.Black;
            textBox3.BackColor = Color.Aqua;
            textBox3.ForeColor = Color.Black;
            textBox4.BackColor = Color.Aqua;
            textBox4.ForeColor = Color.Black;

        }

//New
            clearFields();
            TextColor();
//Save
            try
            {
                connectDB();
                cmd.CommandText = "Insert into student values('" + textBox1.Text + "','" +
textBox2.Text + "','" + textBox3.Text + "'," + textBox4.Text + ")";
                cmd.ExecuteNonQuery();
                MessageBox.Show("ONE STUDENT RECORD IS SAVED", "SAVE", MessageBoxButtons.OK,
MessageBoxIcon.Information);
            }
            catch (Exception ex)
            {
                MessageBox.Show("The error(save) is..." + ex.Message);
            }
            finally
            {
                con.Close();
                clearFields();
            }
  //Exit
            this.Close();
//Search
            try
            {
                connectDB();
                cmd.CommandText = "select * from Student where sid='" + textBox1.Text + "'";
                dr = cmd.ExecuteReader();
                dr.Read();
                if (dr.HasRows == true)
                {
                    textBox1.Text = dr["sid"].ToString();
                    textBox2.Text = dr["sname"].ToString();
                    textBox3.Text = dr["course"].ToString();
                    textBox4.Text = dr["fee"].ToString();
                }
                else
                {
                    MessageBox.Show(" No Such info available", "Find Failed", MessageBoxButtons.OK, 
MessageBoxIcon.Information);
                    clearFields();
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show("The (Search) Error is.." + ex.Message);
            }
            finally
            {
                con.Close();
            }
//Edit
            try
            {
                connectDB();
                cmd.CommandText = "update student set sid='" + textBox1.Text + "',sname='" + 
textBox2.Text + "',course='" + textBox3.Text + "',fee=" + textBox4.Text + " where sid='" + 
textBox1.Text + "'";
                cmd.ExecuteNonQuery();
                MessageBox.Show(" ONE STUDENT INFO MODIFIED", "Modify", MessageBoxButtons.OK, 
MessageBoxIcon.Information);
            }
            catch (Exception ex)
            {
                MessageBox.Show("The(Edit) error is.." + ex.Message);
            }
            finally
            {
                con.Close();
                clearFields();
            }
//DELETE
            DialogResult response;
            response = MessageBox.Show("Are you sure?", "Delete student", MessageBoxButtons.YesNo, 
MessageBoxIcon.Question);
            if (response == DialogResult.Yes)
            {
                try
                {
                    connectDB();
                    //cmd.CommandText = "Delete from student where sid=" + 
int.Parse(textBox1.Text)+"'";
                    cmd.CommandText = "Delete from student where sid='" + textBox1.Text + "'";
                    cmd.ExecuteNonQuery();
                    MessageBox.Show(" ONE STUDENT INFO DELETED", "DELETE", MessageBoxButtons.OK, 
MessageBoxIcon.Information);
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.Message);
                }
                finally
                {
                    con.Close();
                    clearFields();
                }
            }
```

Create an interface to connect to sql server using SQLClient



```
using System.Data.SqlClient;
SqlConnection con = new SqlConnection();
        SqlCommand cmd = new SqlCommand();
        SqlDataReader dr;


        public void connectDB()
        {
            try
            {
                con.ConnectionString = "server=localhost; Initial Catalog=Test; User
Id=nagendra2;Password=password";
                con.Open();
                cmd.Connection = con;
            }
            catch (Exception ex)
            {
                MessageBox.Show("The (ConnectDb)serror is.." + ex.Message);
            }
        }

        public void clearFields()
        {
            foreach (Control ctl in this.Controls)
                if (ctl is TextBox)
                    ctl.Text = " ";
            textBox1.Focus();
        }
        public void TextColor()
        {
            textBox1.BackColor = Color.Aqua;
            textBox1.ForeColor = Color.Black;
            textBox2.BackColor = Color.Aqua;
            textBox2.ForeColor = Color.Black;
            textBox3.BackColor = Color.Aqua;
            textBox3.ForeColor = Color.Black;
            textBox4.BackColor = Color.Aqua;
            textBox4.ForeColor = Color.Black;

        }
  //New
            clearFields();
            TextColor();
//Save
            try
            {
                connectDB();
                cmd.CommandText = "Insert into student values('" + textBox1.Text + "','" +
textBox2.Text + "','" + textBox3.Text + "'," + textBox4.Text + ")";
                cmd.ExecuteNonQuery();
                MessageBox.Show("ONE STUDENT RECORD IS SAVED", "SAVE", MessageBoxButtons.OK,
MessageBoxIcon.Information);
            }
            catch (Exception ex)
            {
                MessageBox.Show("The error(save) is..." + ex.Message);
            }
            finally
            {
                con.Close();
                clearFields();
            }
//Edit
            try
            {
                connectDB();
                cmd.CommandText = "update student set sid='" + textBox1.Text + "',sname='" +
textBox2.Text + "',course='" + textBox3.Text + "',fee=" + textBox4.Text + " where sid='" +
textBox1.Text + "'";
                cmd.ExecuteNonQuery();
                MessageBox.Show(" ONE STUDENT INFO MODIFIED", "Modify", MessageBoxButtons.OK,
MessageBoxIcon.Information);
            }
            catch (Exception ex)
            {
                MessageBox.Show("The(Edit) error is.." + ex.Message);
            }
            finally
            {
                con.Close();
                clearFields();
            }
//DELETE
            DialogResult response;
            response = MessageBox.Show("Are you sure?", "Delete student", MessageBoxButtons.YesNo,
MessageBoxIcon.Question);
            if (response == DialogResult.Yes)
            {
                try
                {
                    connectDB();
                    //cmd.CommandText = "Delete from student where sid=" +
int.Parse(textBox1.Text)+"'";
                    cmd.CommandText = "Delete from student where sid='" + textBox1.Text + "'";
                    cmd.ExecuteNonQuery();
                    MessageBox.Show(" ONE STUDENT INFO DELETED", "DELETE", MessageBoxButtons.OK,
MessageBoxIcon.Information);
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.Message);
                }
                finally
                {
                    con.Close();
                    clearFields();
                }
            }
//Search
            try
            {
                connectDB();
                cmd.CommandText = "select * from Student where sid='" + textBox1.Text + "'";
                dr = cmd.ExecuteReader();
                dr.Read();
                if (dr.HasRows == true)
                {
                    textBox1.Text = dr["sid"].ToString();
                    textBox2.Text = dr["sname"].ToString();
                    textBox3.Text = dr["course"].ToString();
                    textBox4.Text = dr["fee"].ToString();
                }
                else
                {
                    MessageBox.Show(" No Such info available", "Find Failed", MessageBoxButtons.OK,
MessageBoxIcon.Information);
                    clearFields();
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show("The (Search) Error is.." + ex.Message);
            }
            finally
            {
                con.Close();
            }
//Exit
            this.Close();
```

Create an interface to add a record into the database table using DataRow



```
using System.Data.SqlClient;
SqlConnection con = new SqlConnection();
        SqlCommand cmd = new SqlCommand();
        SqlDataAdapter da=new SqlDataAdapter();
        DataSet ds=new DataSet();


        public void connectDB()
        {
            try
            {
                con.ConnectionString = "server=localhost; Initial Catalog=Test; User
Id=nagendra2;Password=password";
                con.Open();
                cmd.Connection = con;
            }
            catch (Exception ex)
            {
                MessageBox.Show("The (ConnectDb)serror is.." + ex.Message);
            }
        }

        public void clearFields()
        {
            foreach (Control ctl in this.Controls)
                if (ctl is TextBox)
                    ctl.Text = " ";
            textBox1.Focus();
        }
        public void TextColor()
        {
            textBox1.BackColor = Color.Aqua;
            textBox1.ForeColor = Color.Black;
            textBox2.BackColor = Color.Aqua;
            textBox2.ForeColor = Color.Black;
            textBox3.BackColor = Color.Aqua;
            textBox3.ForeColor = Color.Black;
            textBox4.BackColor = Color.Aqua;
            textBox4.ForeColor = Color.Black;

        }
   //New
            clearFields();
            TextColor();
//Save
            connectDB();
            cmd.CommandText="select * from student";
            da.SelectCommand=cmd;
            da.Fill(ds,"student");
            SqlCommandBuilder builder=new SqlCommandBuilder(da);
            DataRow dr=ds.Tables["student"].NewRow();
            dr["sid"]=textBox1.Text;
            dr["sname"]=textBox2.Text;
            dr["course"]=textBox3.Text;
            dr["fee"]=textBox4.Text;
            ds.Tables["student"].Rows.Add(dr);
            da.Update(ds,"student");
            con.Close();
            MessageBox.Show("One Record saved");
```

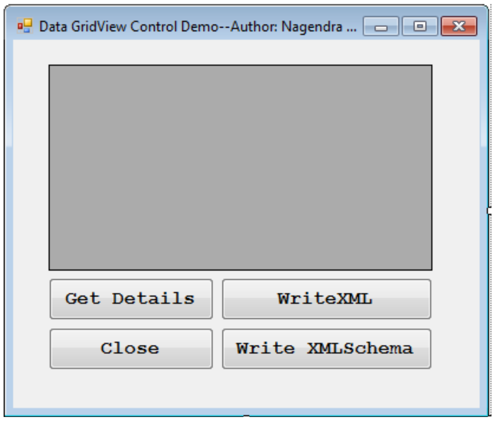
DataGridView control

This control is used to view the data in a tabular format.

Property	Description
Name	The property that uniquely identifies DataGridView.
Column	Gets the collection of columns on this DataTable.
Datamembers	Specifies source database name from which the data is to be populated.
DataSource	Specifies the source dataset name to which the database is associated.

Create an interface to connect to oracle database and retrieve all the rows using a DataGridView control.

Step1: Create an interface following


Data GridView Control Demo--Author: Nagendra ...
Get Details
WriteXML
Close
Write XMLSchema

```
using System.Data.OracleClient;

OracleConnection con;
        OracleCommand cmd;
        OracleDataAdapter da;
        DataSet ds;
        public void connectDB()
        {
            try
            {
                con = new OracleConnection("Server= ORCL; User Id= nagendra; Password=password;");
                cmd = new OracleCommand("select * from student", con);
                da = new OracleDataAdapter();
            }
            catch (Exception ex)
            {
                MessageBox.Show("The (ConnectDb)serror is.." + ex.Message);
            }
        }
//GetDetails
            try
            {
                connectDB();
                con.Open();
                da.SelectCommand = cmd;
                ds = new DataSet();
                da.Fill(ds, "Student");
                dataGridView1.DataMember = "Student";
                dataGridView1.DataSource = ds;
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.Message);
            }
            finally
            {
                con.Close();
            }
//close
            this.Close();
//Write XML
            try
            {
                connectDB();
                con.Open();
                da.SelectCommand = cmd;
                ds = new DataSet();
                da.Fill(ds, "Student");
                ds.WriteXml(@"E:\Nagendra Data\C# Course Content\Class #10 C#-
ADO.NET\Test\XMLStudent.xml");
                MessageBox.Show("XML File created!");
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.Message);
            }
            finally
            {
                con.Close();
            }
//Write XML Schema
```

```
             try
            {
                connectDB();
                con.Open();
                da.SelectCommand = cmd;
                ds = new DataSet();
                da.Fill(ds, "Student");
                ds.WriteXmlSchema(@"E:\Nagendra Data\C# Course Content\Class #10 C#-
ADO.NET\Test\XMLStudentSchema.xsd");
                 //contains only list
                ds.WriteXmlSchema(@"E:\Nagendra Data\C# Course Content\Class #10 C#-
ADO.NET\Test\XMLStudentSchema.xml");
                 //contains structure that includes attribute and its data type
                MessageBox.Show("XML Schema File created!");
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.Message);
            }
            finally
            {
                con.Close();
            }
```

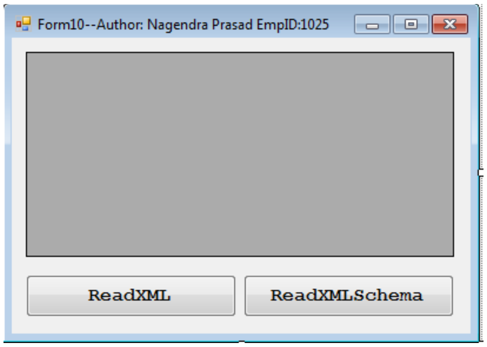
Note: Dataset is under System.Data namespace.

Reading data from XML file.
Create an interface to read data from an XML file
Step1: create XML file
Project> Add New Item > XML file > change the name to Student.xml. click Add Button.

Step 2: **Enter the following sample data**

```
<student>
        <stid>
                <sno>1001</sno>
                <name>person1</name>
                <course>MCA</course>
                <fee>3500</fee>
        </stid>
        <stid>
                <sno>1002</sno>
                <name>person2</name>
                <course>MCA</course>
                <fee>3500</fee>
        </stid>
        <stid>
                <sno>1003</sno>
                <name>person3</name>
                <course>MCA</course>
                <fee>3500</fee>
        </stid>
        <stid>
                <sno>1004</sno>
                <name>person4</name>
                <course>MCA</course>
                <fee>3500</fee>
        </stid>
</student>
```

Step 3: Create the following interface



Step 4: Enter the events of the above form

```
//ReadXML
            try
            {
                DataSet ds = new DataSet();
                ds.ReadXml(@"E:\Nagendra Data\C# Course Content\Class #10 C#-
ADO.NET\Test\Student1.xml");
                dataGridView1.DataSource=ds;
                dataGridView1.DataMember="Stid";

            }
            catch (Exception ex)
            {

            }
//ReadXMLSchema
            try
            {
                DataSet ds = new DataSet();
                ds.ReadXmlSchema(@"E:\Nagendra Data\C# Course Content\Class #10 C#-
ADO.NET\Test\Student1.xml");
                dataGridView1.DataSource = ds;
                dataGridView1.DataMember = "Stid";
            }
            catch (Exception ex)
            {
            }
```

Data View

A Predefined object that is used to retrieve the data for viewing purpose.

RowFilter: Specifies the expression that can be appended after the where clause of a select statement.

Link Label

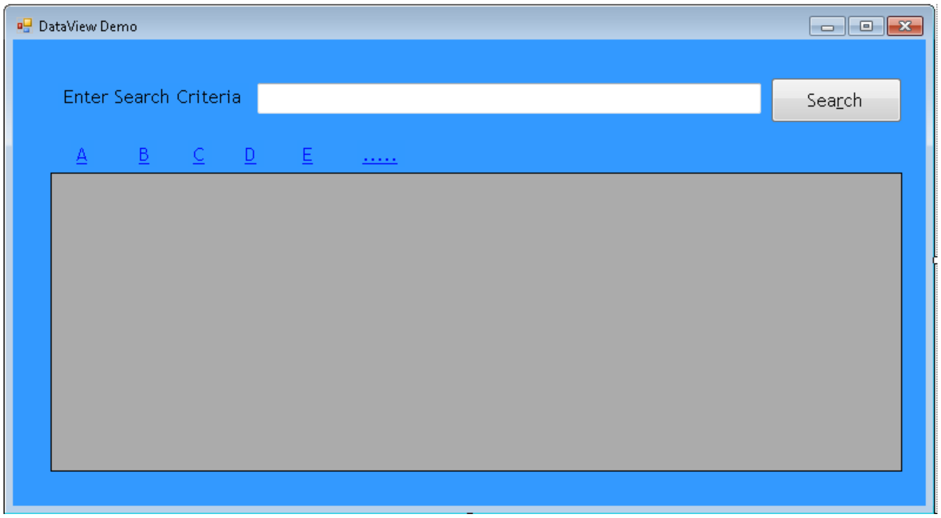
This control is used to provide a hyperlink style control in windows application.

Properties

Name The property that uniquely identifies link label.
Active Link Color Specifies the color of the link label, when the user clicks the link label.
LinkColor Determines the color of the link label. By default true.
Text Specifies the text to be set for the link label.

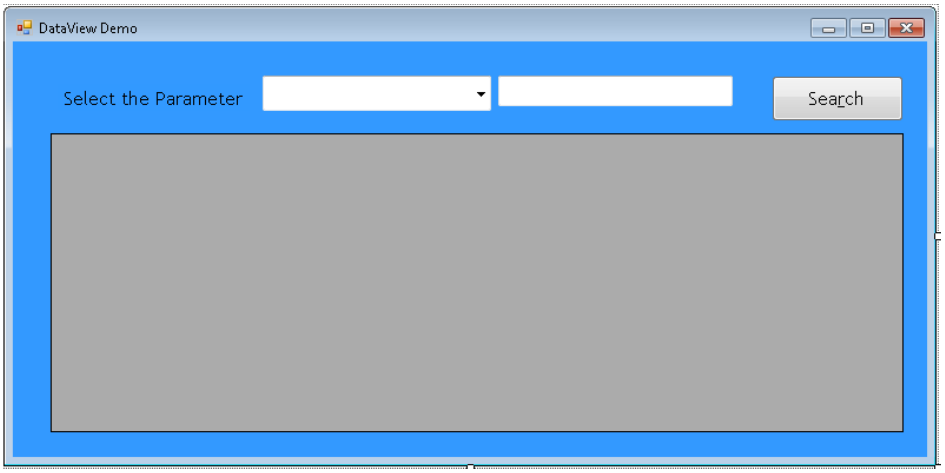
LinkClicked() This event occurs when the user clicks the link label.

Create an interface to filter the records from emp table using data view.



```
using System.Data.OleDb;

DataSet ds = new DataSet();
        DataView dv;
//A's Search Result
            dv.RowFilter = " Ename like 'a%'";
            dataGridView1.DataSource = dv;
//B's Search Result
            dv.RowFilter = " Ename like 'b%'";
            dataGridView1.DataSource = dv;
  //C's Search Result
            dv.RowFilter = " Ename like 'c%'";
            dataGridView1.DataSource = dv;
//D's Search Result
            dv.RowFilter = " Ename like 'd%'";
            dataGridView1.DataSource = dv;
//E's Search Result
            dv.RowFilter = " Ename like 'e%'";
            dataGridView1.DataSource = dv;
            dataGridView1.DataSource = dv;
  //Search
            dv.RowFilter ="Ename='"+ textBox1.Text+"'";
            dataGridView1.DataSource = dv;
//for all others
            label1.Visible = true;
            textBox1.Visible = true;
            Search_bt.Visible = true;
            textBox1.Focus();
//At the loading time
            try
            {
                OleDbConnection con = new OleDbConnection("Provider=MSDAORA.1; User Id=scott;
Password=tiger;");
                OleDbDataAdapter da = new OleDbDataAdapter("Select * from emp", con);
                da.Fill(ds, "emp");
                dv = ds.Tables["emp"].DefaultView;
                dataGridView1.DataSource = dv;
            }
            catch (Exception ex)
            {
                MessageBox.Show("The load(Error) is..." + ex.Message, "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error);
            }
```

Create an interface to modify the above program by having the customized search



```
using System.Data.OleDb;
  DataSet ds = new DataSet();
        DataView dv;
//Load
             try
             {

                 OleDbConnection con = new OleDbConnection("Provider=MSDAORA.1; User Id=scott;
Password=tiger;");
                 OleDbDataAdapter da = new OleDbDataAdapter("select * from emp", con);
                 da.Fill(ds, "emp");
                 dv = ds.Tables["emp"].DefaultView;
                 foreach (DataColumn dc in ds.Tables["emp"].Columns)
                     comboBox1.Items.Add(dc.ColumnName.ToString());
             }
             catch (Exception ex)
             {
                 MessageBox.Show(ex.Message, "Error", MessageBoxButtons.OK, MessageBoxIcon.Information);
             }
//Search
             int i = comboBox1.SelectedIndex;
             if (i == 1)
             {
                 string columnname = comboBox1.SelectedItem.ToString();
                 string expr = columnname + " Like '" + textBox1.Text + "%'";
                 dv.RowFilter = expr;
                 dataGridView1.DataSource = dv;
             }
             if (i == 2)
             {
                 string columnname = comboBox1.SelectedItem.ToString();
                 string expr = columnname + " Like'" + textBox1.Text + "%'";
                 dv.RowFilter = expr;
                 dataGridView1.DataSource = dv;
             }
```