Creating MDI Applications

MDI stands for multiple document interface. An MDI lets us open windows within a parent container window. The multiple document interface MDI allows us to create an application that maintains multiple forms with a single container form. Application such as Microsoft Excellent Microsoft word for windows use multiple versions of windows were predominant. The multiple document nature allowed user to open more than one file at a time, without having to open several copies of the program itself. This not only saves time, but is also saves memory. Visual Basic applications can have only one MDI form, which contains all the child forms.

An MDI form is similar to an ordinary form with one restriction we cannot place a control directly on a MDI form unless the control has an align property (such as a picture box control) or has no visible interface (such as timer control). The multiple document interfaces can be designed for document centre applications. This application allows the user to open many similar document at the same time.

While using the word processors accessed multiple documents simultaneously. To provide the user similar facility have to create Multiple Document Interface (MDI)applications, which will display multiple forms in the same window. It acts as a container for all the forms. The components of the MDI applications are the parent Form and Child Form. Parent Form is also known as MDI Form.

The characteristics of MDI application

1. There can be only one MDI form in the application.
2. There may be as many child forms as user want.
3. All the child forms can be displayed only within the MDI form's workspace.
4. When a child form is minimized, the icon of it will be displayed the bottom of the MDI form and not on the task bar.
5. The form controls like label, text box cannot be placed on the MDI form except the picture box control.
6. Menus can be created on the MDI form.

Features of Child Form

1. All the Child Forms gets invoked within the MDI Form.
2. If menu is added in the Child Form, it gets displayed on the MDI form.
3. After minimizing a Child Form its icon and the title bar gets displayed within the MDI form.
4. Again, after maximizing the child from its caption gets displayed along with the caption of MDI from in the title bar of the form.
5. When the MDI Form is minimized all the child Forms are also minimized.

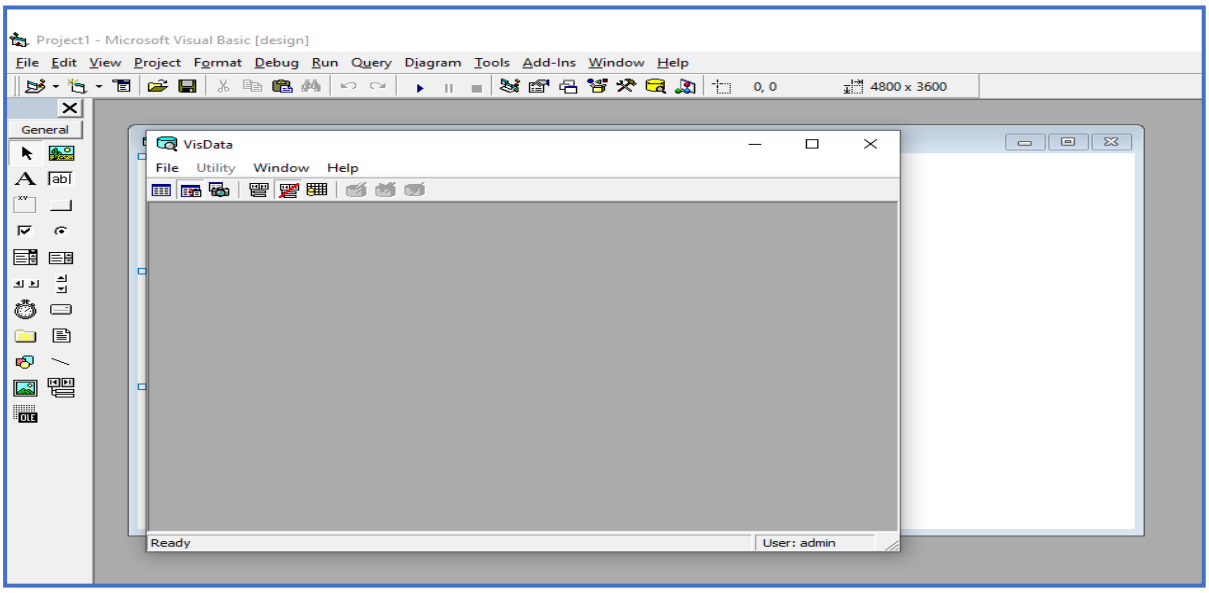
SDI	MDI
1. SDI stands for Single Document Interface.	1. MDI stands for Multiple Document Interface.
2. In the previous version of Visual Basic the IDE was designed as a single document interface.	2. In Visual Basic 6.0 the IDE is in a multiple document interface format.
3. In a single document interface, each window is a free floating window that is contained within a main window and can move anywhere on the screen as long as Visual Basic is the current application	3. In a multiple document interface the windows associated with the project will stay within a single container known as the parent. Code and form based windows will stay within the main container form.
4. With SDI option, windows can only be docked beneath the menu bar.	4. With the MDI option, windows can be docked to any side of the parent window.

Database Handling:

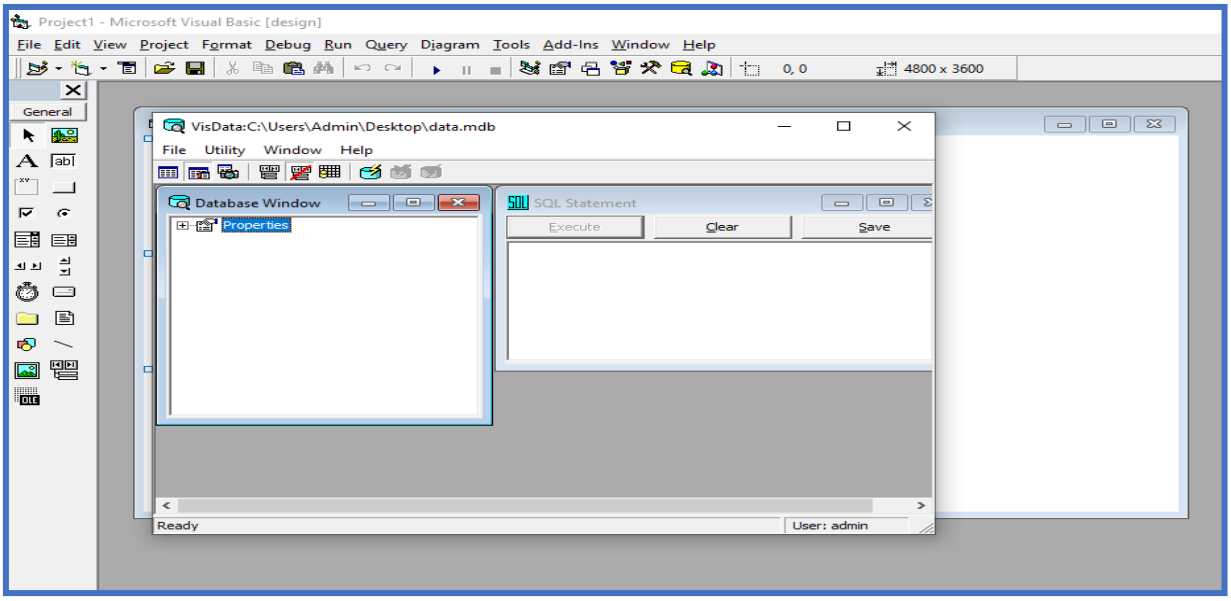
Creating the Database

Let's create an application, which will handle the data of Microsoft Access data source. Firstly, you have to create the database. You can create the database after opening Microsoft Access; otherwise, you can create from Visual Basic directly.

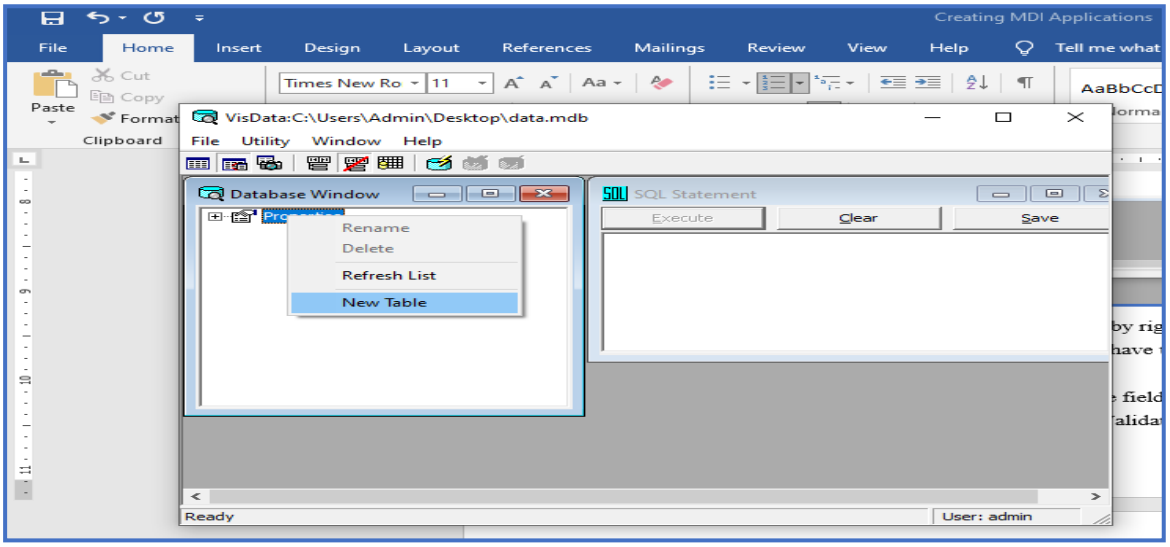
1. Start with a new project.
2. Click on Visual Data Manager from the Add-Ins menu.



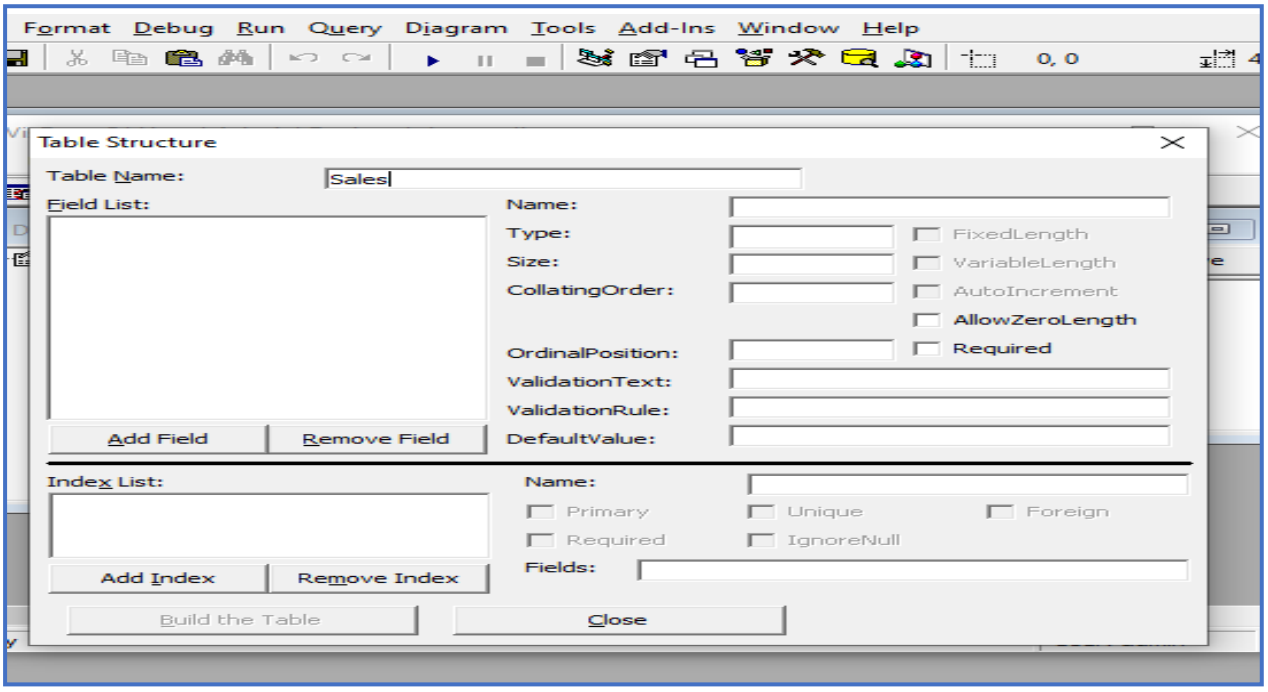
3. The VisData dialog gets invoked. Click on New from the File menu and then on Microsoft Access ->Version 7.0.
4. The Select Microsoft Access Database to Create dialog appears which prompts to enter the file name. Provide a file name and click on Save



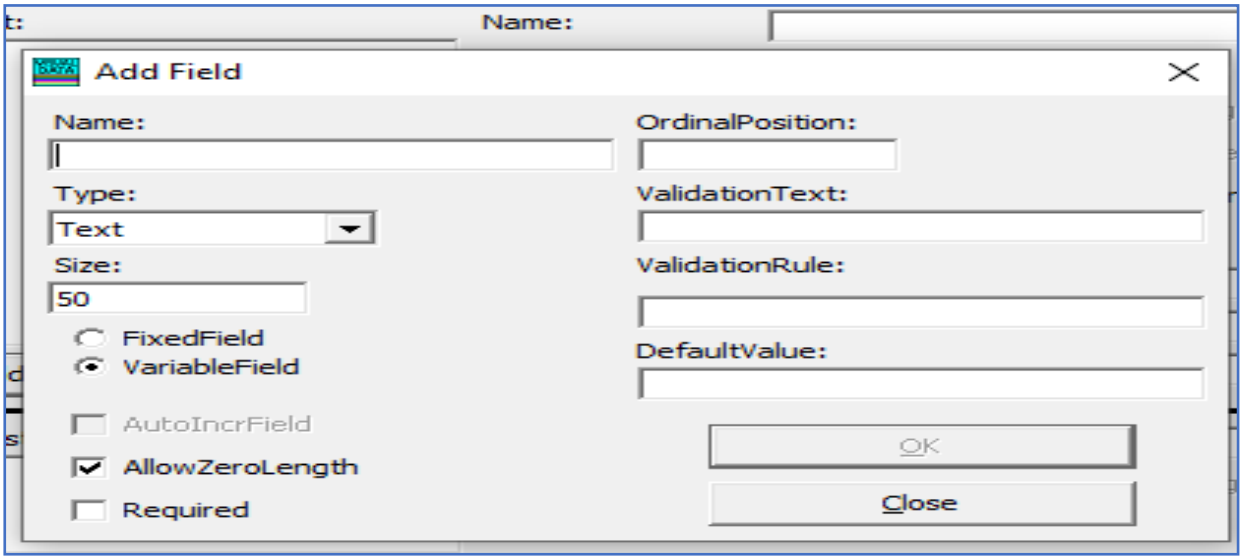
5. Select properties from the Database Window and by right clicking on it click on new Table option.



6. The Table Structure dialog gets invoked. Firstly, have to provide a table name. Then click on Add field button to add field for the table.

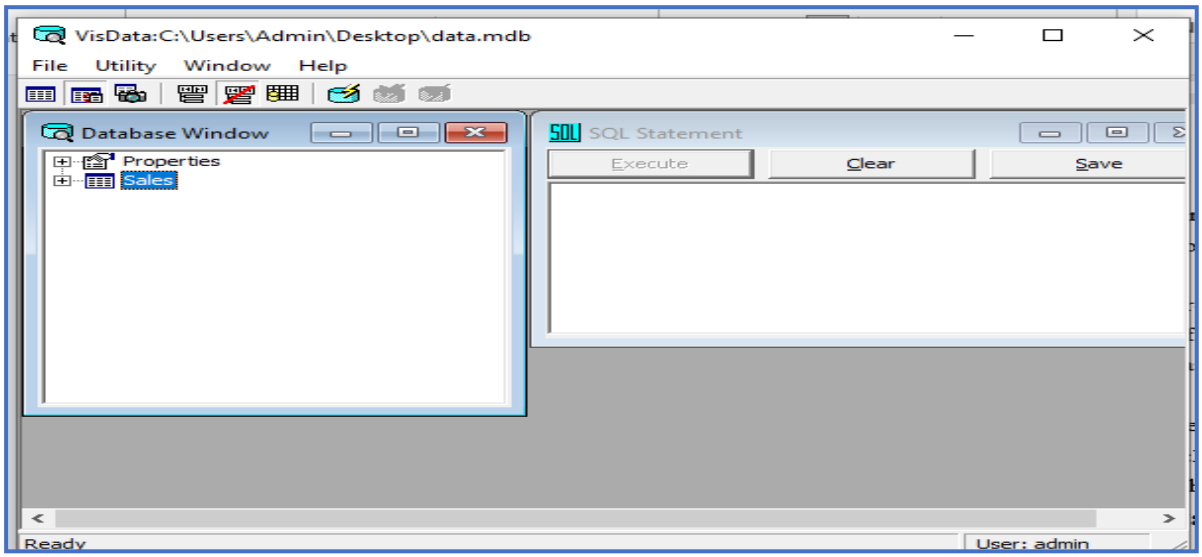


7. The Add field dialog appears. Here have to set the fields information. Provide the Field name, select the type and then set the size. More ever can set the ValidationRule, Validation Text etc.



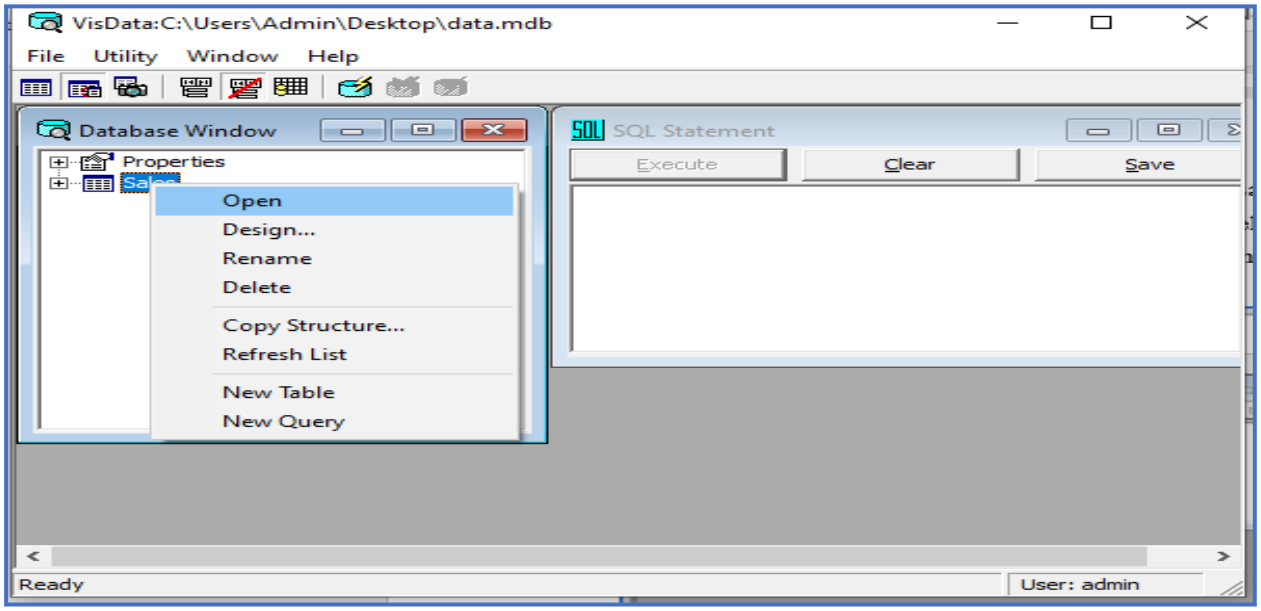
Name	Type	Size
P_name	Text	30
P_desc	Text	40
P_mfd	Date/Time	8
P_cost	Single	4

8. After entering each field details click on Ok. Later click on Close to close the Add field dialog. It will Observe that all the fields appeared in the Field List box. Now to build the table click on Build the table. The table gets listed in the database Window.

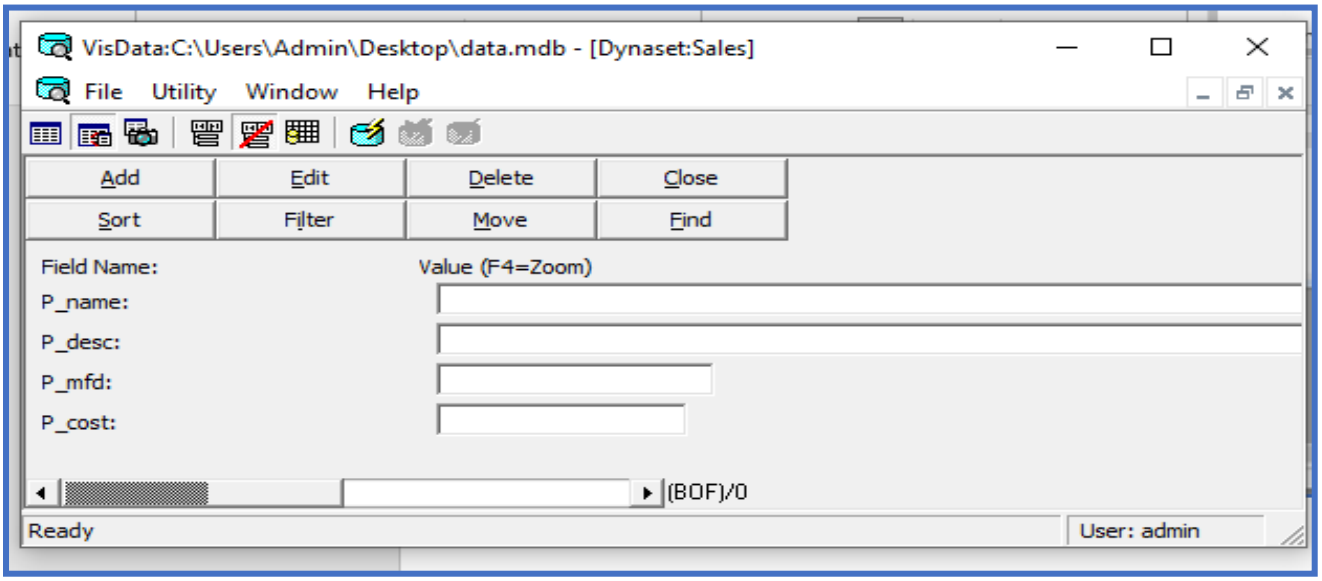


9. Your next work is to add the records in the table. From the Database Window select data table by clicking on it. Holding the mouse on it press the right button of the mouse. Click on Open from the Popup menu.

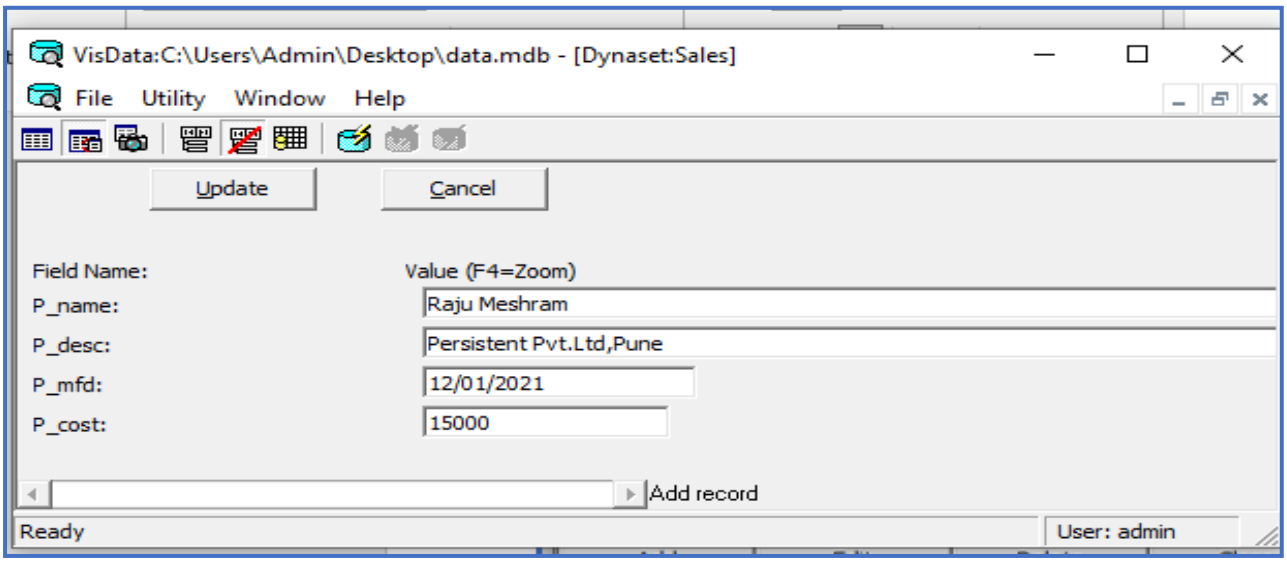
10. It invokes the dialog. Note that the dialog contains different buttons to perform the necessary work.



11. So, to add the records click on Add button.



12. Provide the data and click on Update to update the database.



13. Repeat the procedure to add more records. Then click on Close button to close the dialog.
14. So, database is ready. To exit click on Exit from the File menu.

Accessing the database by using the Data Control

Data Controls

The data control is an excellent tool for rapidly developing a database information retrieval system. The data control gives us access to database without any programming. With very little code, we can get a useful view in to stored information. We can set a few properties of the control and the use regular controls such as text boxes to display the values of the fields in the database. This is no code approach to database programming, which is implemented quite nicely in Visual Basic. This is practically a drop in place process whereby we can place a data control, a data control, a DBGrid, and a command button on a form and be able to access our database I a form and be able to access our database in a matter of minutes. The data control is a professional quality tool that can meet most application need for data access. A data control is used to create applications that display, edit, and update information from numerous types of existing databases, including Microsoft Access, dBase, Foxpro, and paradox.

The data control can perform the following tasks without writing any code.

1) Connect to a local or external database.
2) Open a specified database.
3) Attach data field to bound controls where values can be displayed and changed.
4) Add new records or update a database with the data displayed on the bound controls.
5) Close the database.

Data control has three parts.

1. Two inner arrows that allows us to move forward and backward through a database table with displaying one record at a time.
2. Outer arrows that helps us to move to the first and last record in a database table.
3. The middle area that displays whatever information that we want to display

To access data from the database, have to set few properties of the Data Control. Now follow thesesteps to create the application.

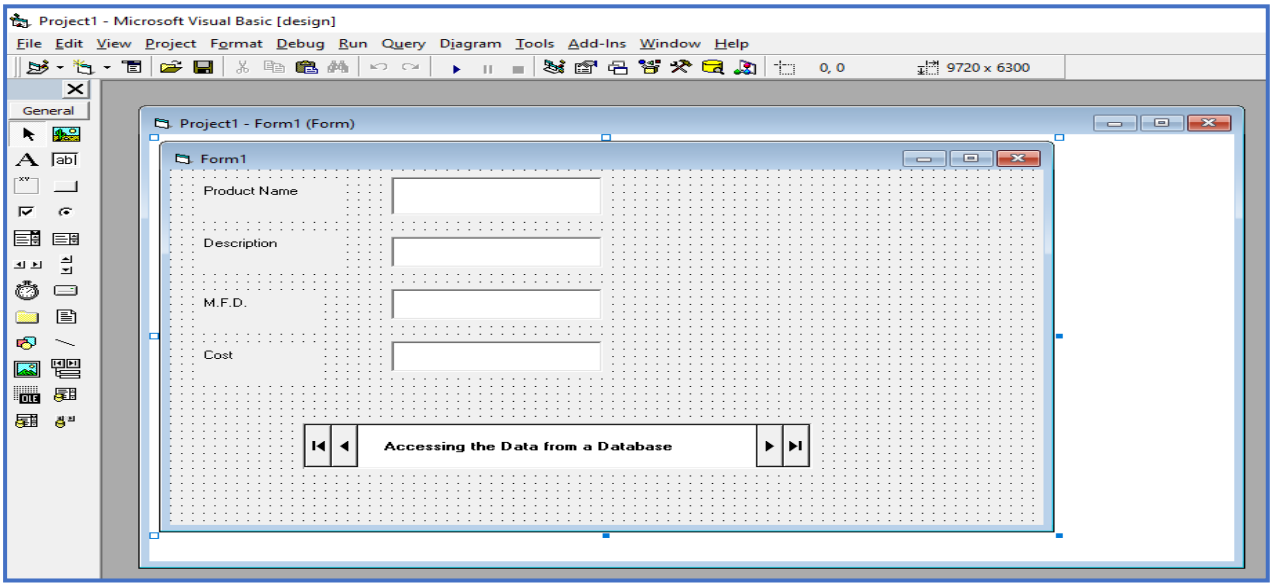
1.Firstly add a Data Control in the Form. Change the Caption property as “Accessing the data from a Database”.

2.Select Data1.From the Properties Window click on the ellipsis button of Database name property. Previously created the database named as data, in the default folder of Visual basic i.e. VB98.Select the database data and click on Open. The filename with its full path appears.

3.To set the Record Source property. Click on the drop-down arrow. our table name was data. So will find that the data table is added in the list. Click on data, to set the Record Source property as data. The Data Control accesses records one at a time unable to display it.

4.To display the records use the textbox control. Our table has four fields. So, add four textbox and four labels.

5.Form view given below and accordingly arrange the caption of Labels, Data Control and set their position.

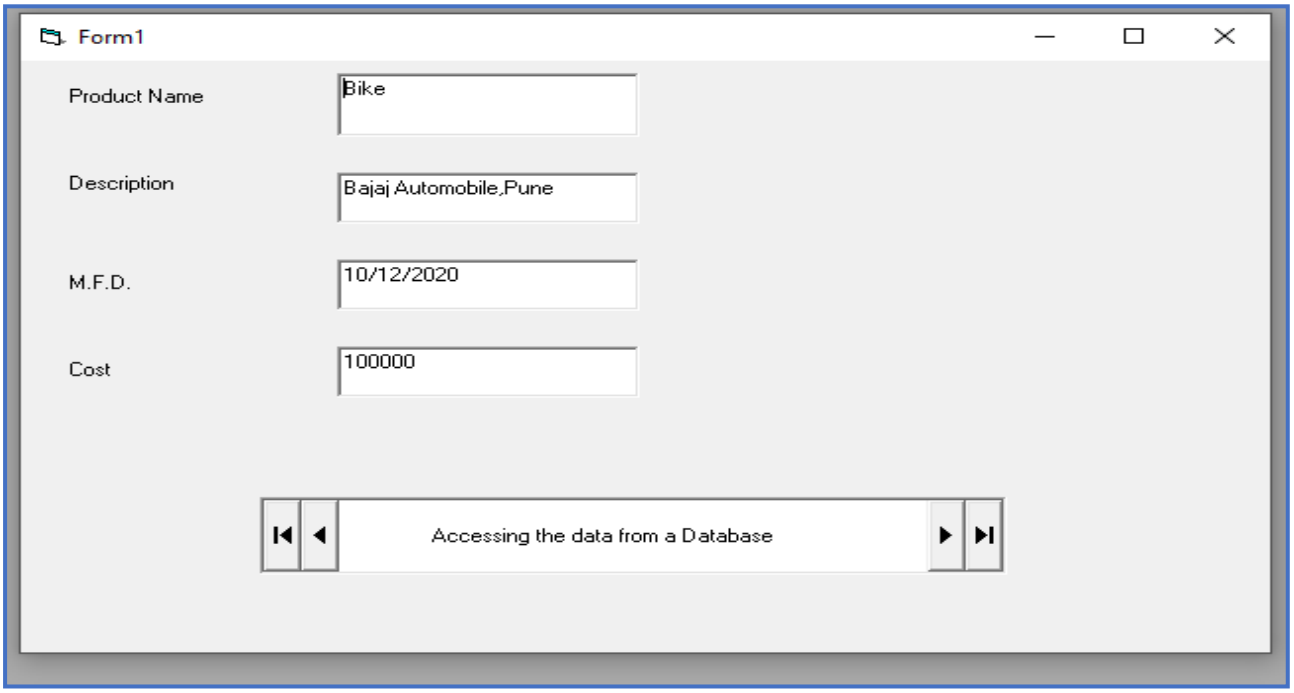


6. Remove the Text property of all the textboxes. Then select Text1, click on the drop-down arrow of Data Source property. Our data source is Data1 and will find in the list Data1.To set the DataSource property as data1, click on it.

7. Then click on the DataField property of Text1.it will display the list of fields. Click on P_name field. This will display the record of P_name field in Text1.

8. Repeat the procedure and set the Datafield property of Text2 as P_desc,Text3 as P_mfd and Text4 as P_cost.

9. Run the application and will observe that the textboxes are displaying the first record of the table data. To navigate between the records, use the navigate buttons of Data1.



Using ADO Data Control.

The visual database tool are not the only possibility for developing database application with Active data object. One of the new controls added to the toolbox by the data project type is the ADO Data control, which is equivalent to the data control. With the ADO data control we can connect to more databases of different types. The simplest way to use the ADO in projects is to add the ADO data control to form and use similar way to using the data control. To use the ADO Data control with standard EXE project must add this control to the tool box.

Open the project menu and select component to open the component dialog box and check the Microsoft ADO control 6.0 (OLEDB) option.

The general characteristics of ADO

1. Ease of use.
2. High performance.
3. Programmatic control of cursors.
4. Ability to return multiple result sets from a single query.
5. Excellent error trapping.

Comparing ADO, DAO and RDO

ADO is successor to DAO and RDO functionally ADO2.0 is most similar to DAO and there is generally a similar mapping between the two models.

The major benefits offered by ADO as compared to DAO and RDO are.

1. It contains fewer objects as compared to the DAO and RDO models.
2. It contains more number of properties and methods as compared to DAO / RDO
3. Much of the functionality contained in the DAO and RDO models is considered into single object, making it much simpler object model.

4. Unlike DAO and RDO although ADO objects are hierarchical, they are also creatable outside hierarchy.

ADO Library

To use ADO with Microsoft Visual Basic you also need to establish a reference to the ADO type library. Select references from the Project menu check the box for "Microsoft Active X Data Objects 2.0 Library" and then click "Ok" ADO objects, methods and properties will then be accessible.

To add ADO library to the project

Project -> References -> Microsoft Active X data Object 2.0 Library

1. Connection

This object represents a single session with the selected data source. It maintains connection information about the data provider. This object is similar to the RDODC connection object in most respects. The connection maintains properties to indicate the cursor type, connect string, query time out, connection timeout, and other ADO specific properties such as default database and isolation level.

2. Command

The command object is used to save a query definition in our application. This is useful when developer uses the same or when he performs the same action query this object specifies the data definition or data manipulation statement to be executed. since ADO supports both data definition and data manipulation, the developer will still be able to create database using ADO. It supports a number of new properties used to describe the type and suppose of the query therefore helping ADO optimize the operation being performed.

3. Recordset

This object represents a set or rows fetched from the database as a result of some command. The user can control the type of cursor being used on the record, the way in which they should be sorted and filtered as well as look at the definitions of each field in the recordset. This similar to the recordset object and RDO.

4. Field

This part is dependent and always exists as part of a recordset. We can traverse the fields collection to retrieve information object each field in the recordset. The recordset feature a fields collection to contains all its field objects. It is equivalent to RDO‟s object, but contain more information. Richer data type support is also available in ADO. We can populate our recordset object with fields by using the append method.

5. Parameter

This object is part of the command object and it allows us to fill in values in a query. This is similar to

ADO object Model

ADO uses a new database connection framework called OLEDB which allows faster more flexible access to multiple data providers, and ADO wraps it all into one; easy to use interface. This means that we can write database applications that can easily scale from single user databases such as access to complex client server system such as Oracle, SQL server or almost any other database that is either an OLEDB provider, or ODBC provider.

ADO provides the means for us to perform the following sequence of actions.

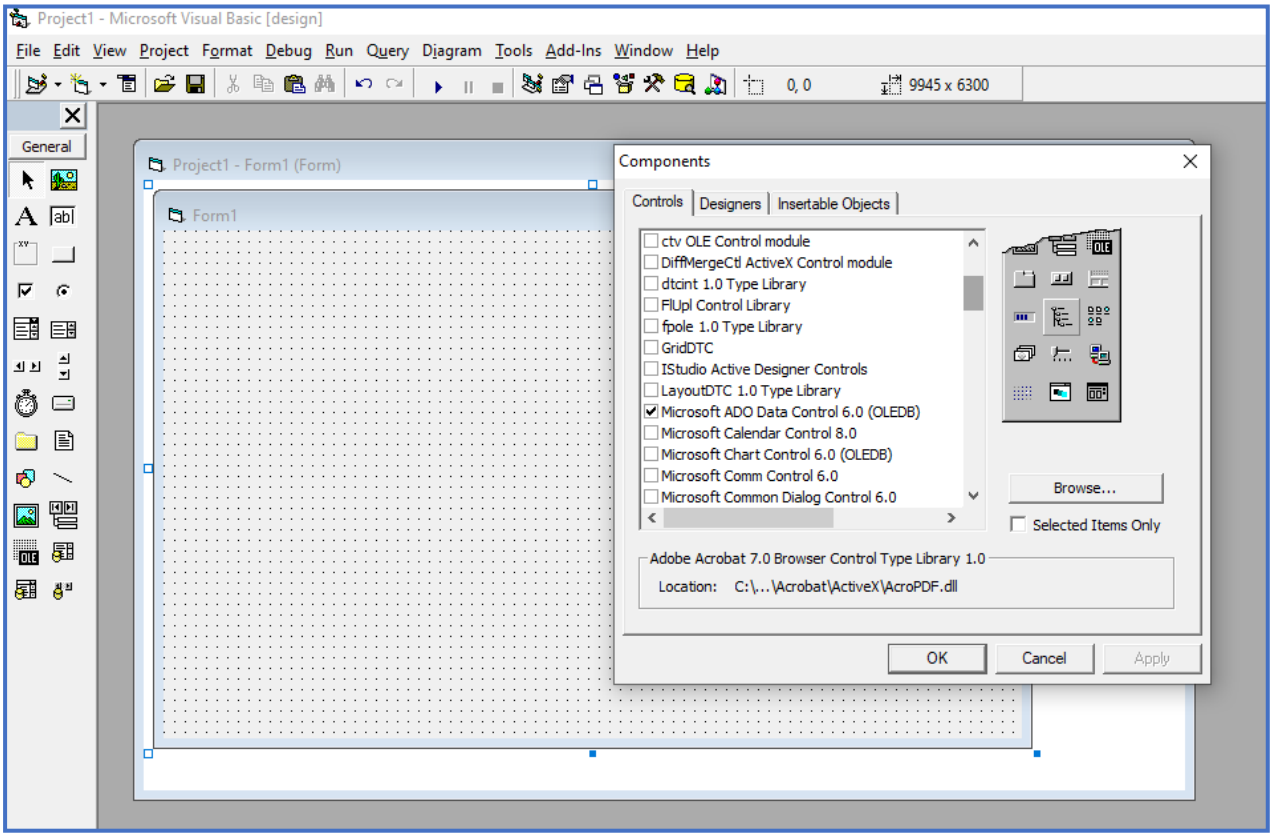
1. Connect to a data source.
2. Specify a commands to gain access to the data source optionally with variable parameter or optionally optimized for performance.
3. Execute the command.
4. If the command causes the data to be returned in the form of rows in a table, store the rows in a cache that we can easily examine manipulate or change.

5. If appropriate, update the data source with changes from the cache of rows.
6. Provides a general means to detect errors.

It can access the database with ADO Data Control. To set few properties of that control.

Now follow these steps, which will guide to access the database with ADO Data control.

1. Start with a new project.
2. Firstly to add the ADO Data control. To do so click on Components from the project menu.

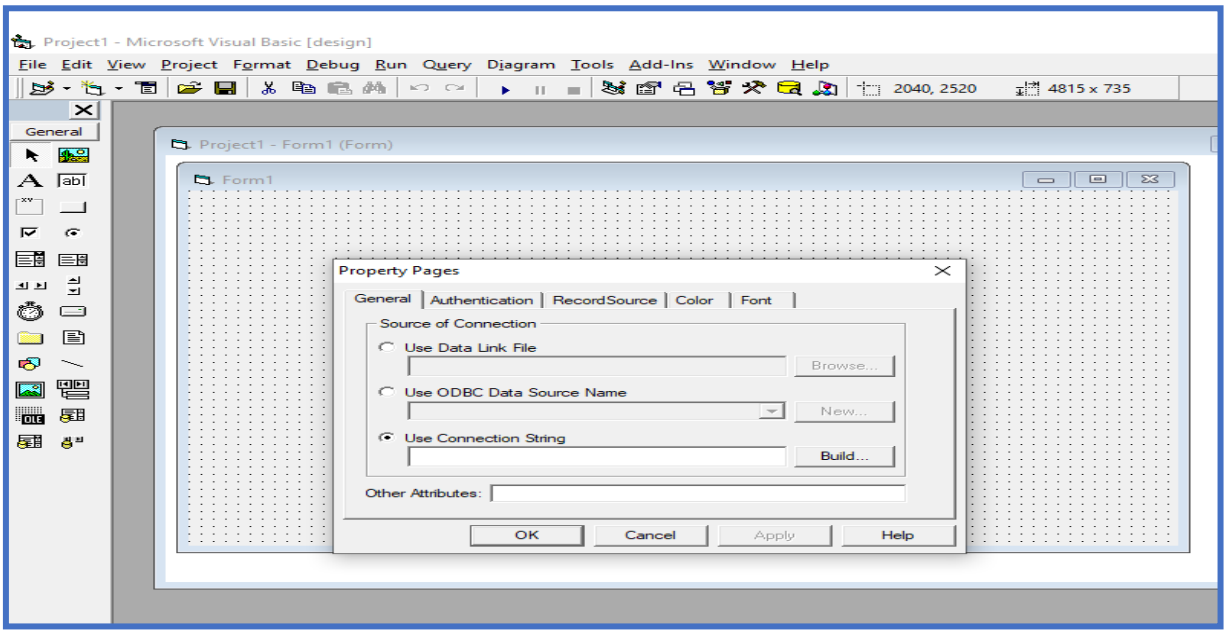


3. Select the option “Microsoft ADO Data Control 6.0(OLEDB)” Click on Apply and then on OK.ADO Data Control gets added in the Toolbox.
4. Add the control in the Form. Let’s access the data from the database, which created previously.
5. Now to specify the data source from where it will retrieve the data. It can specify the table or can write a SQL query for accessing the Data. So, to specify the data source right click on the ADO Data Control.
6. Click on ADODC Properties option from the popup menu. The Property pages dialog gets invoked. It can connect the data source in three different ways. In the General tab of the Property pages will find three options,
7. Use DataLink File, Use ODBC Data Source Name and Use Connection String. The utility of the options is given below.

Use Data Link File: When are using a link file(.udl) for specifying the connection choose this option. To mention the path of the .udl file in the given field.

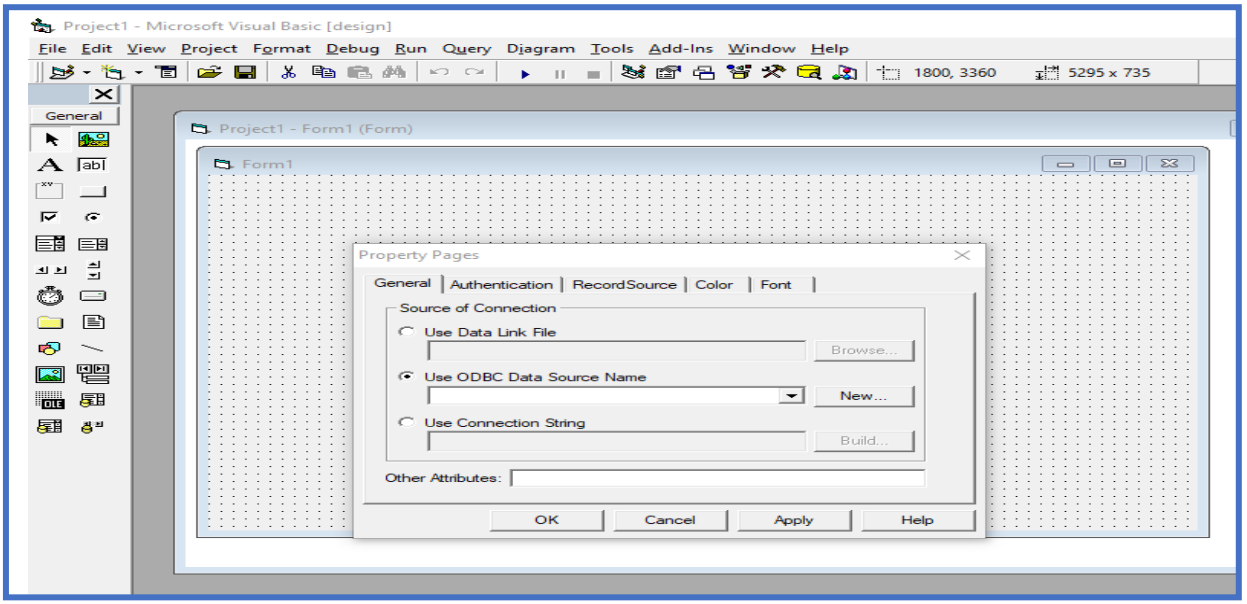
Use ODBC Data Source Name:

When are connecting using the ODNC DSN then use this option. If have already created the ODBC DSN then select the radio button “Use ODBC Data Source Name” and then select the DSN from the list. Again, using the New---button can create a new ODBC DSN.To create have to click on New button, which invokes the Create new Data Source wizard.



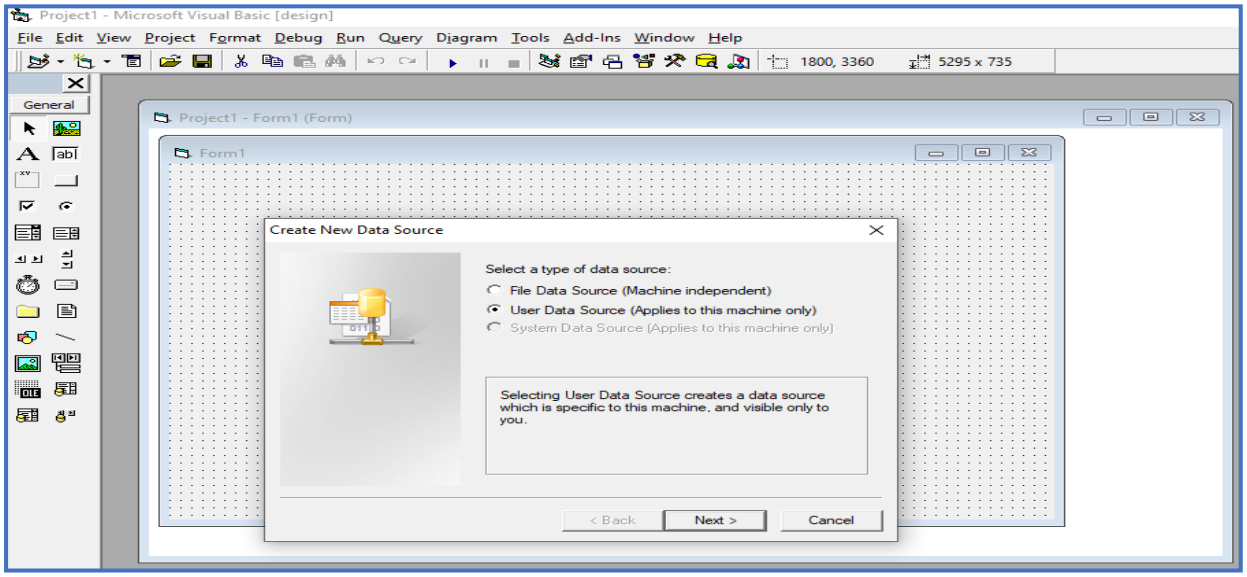
Use Connection String: -

1. When are going to use the OLE DB provider to connect the database select this option. To specify the connection, click on the Build button. Which invokes the Data Link Properties dialog.
2. In application connect using the ODBC DSN. So select the option “use ODBC Data Source Name”. To create a new ODBC DSN so click on the new button.
3. It invokes the Create new Data Source wizard. In the first step have to select the data source type.



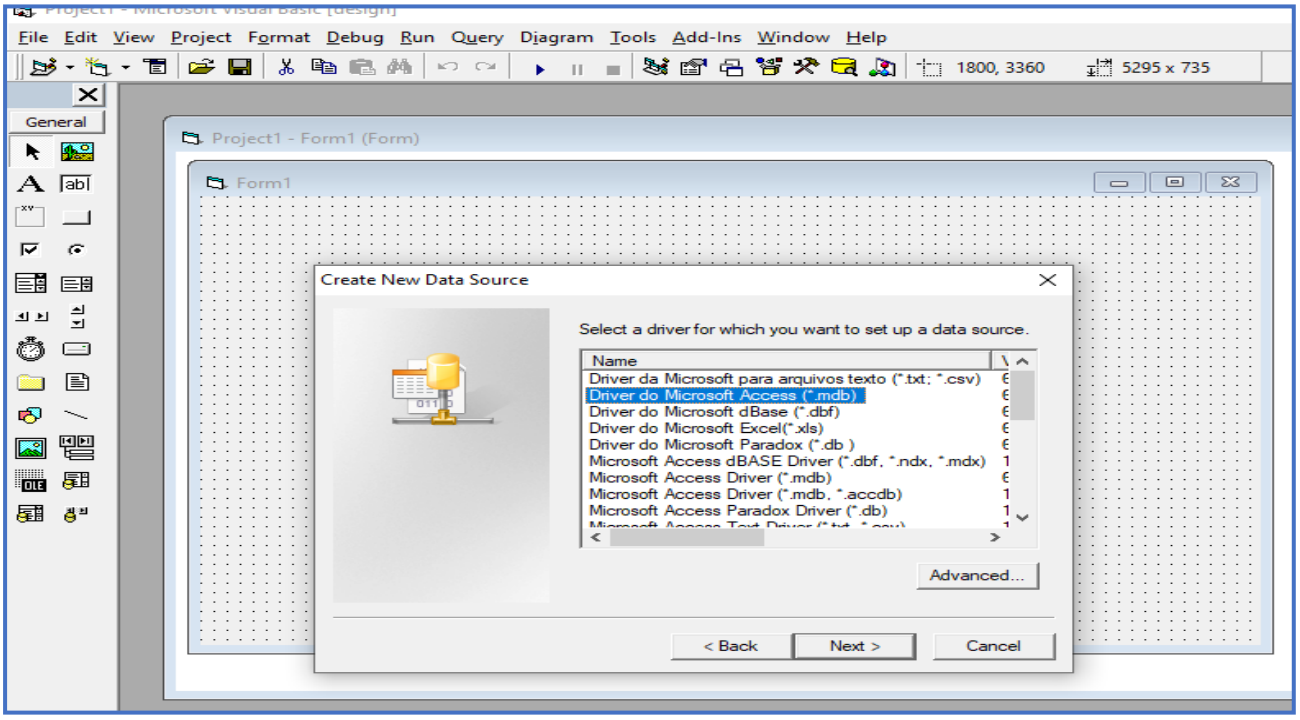
Let’s select” System Data Source “Click on Next to proceed.

1. In the next step have to select a driver. To created the table in Access so select “Driver do Microsoft Access(*mdb) and click on next.



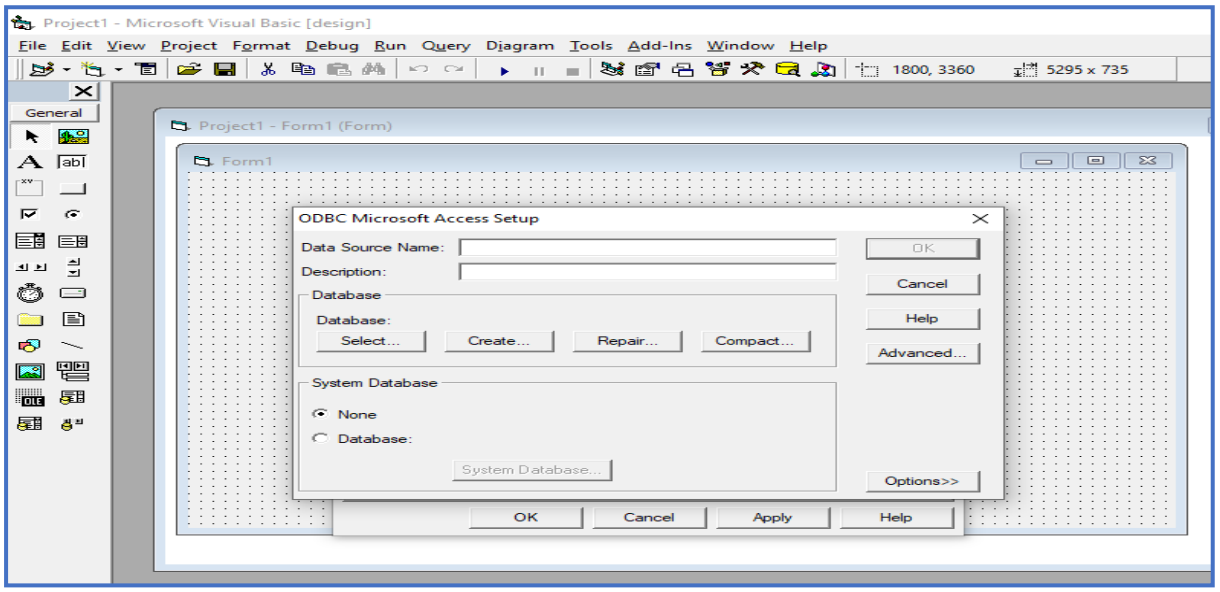
This is the last step of the wizard. Click on Finish button.

2. The ODBC Microsoft Access Setup appears.
3. Let's set the Data Source Name as "ADO Data".

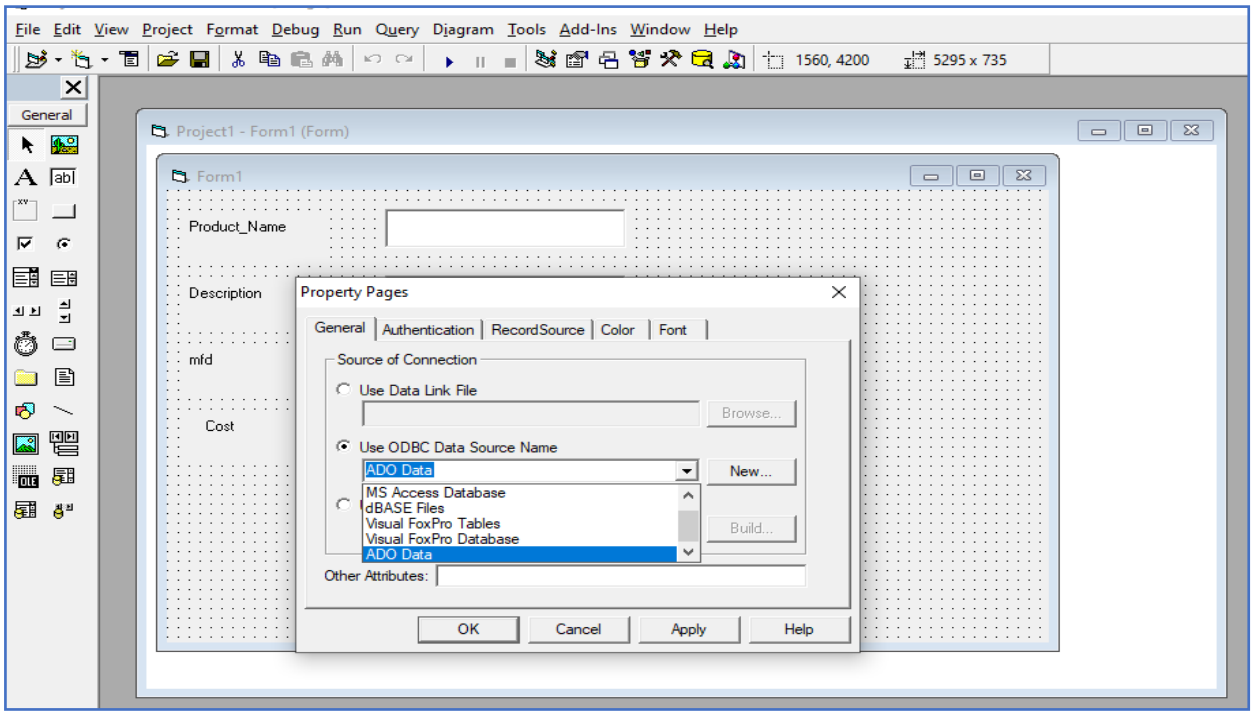


4. To select the database. Here it will use the Data.mdb, which have created previously. Click on Select button.

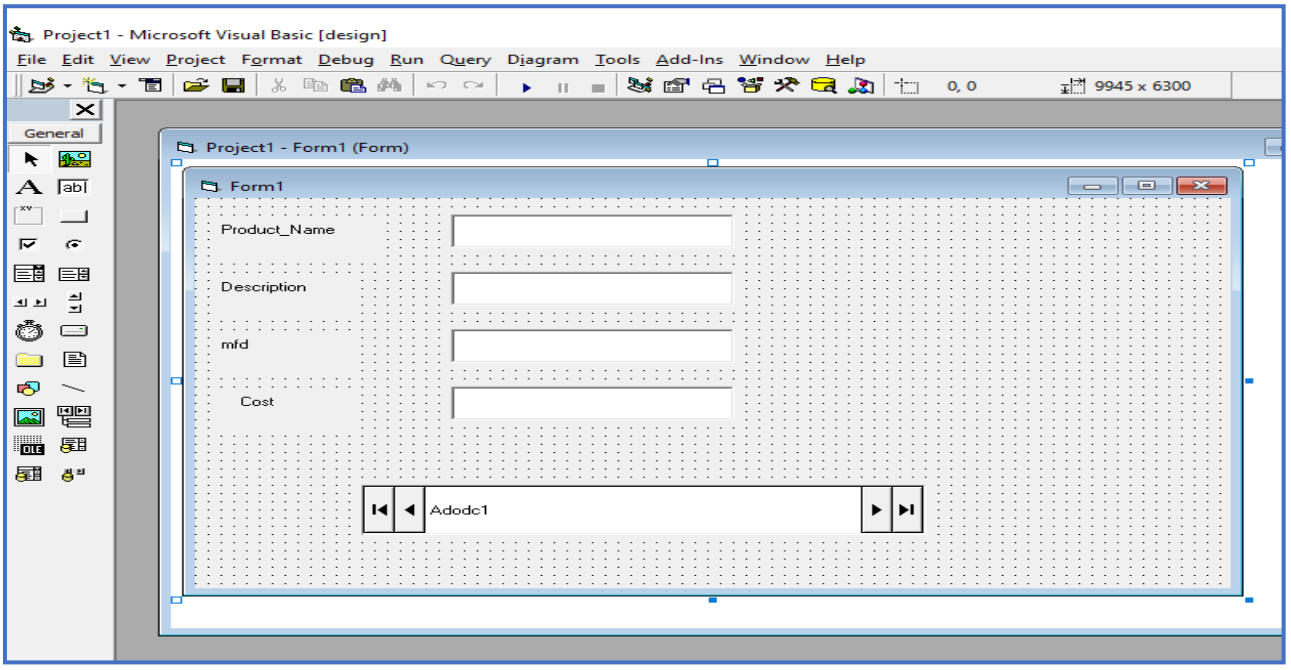
   The select Database dialog gets invoked



5. The Database name with its path appears next to Database in the ODBC Microsoft Access Setup dialog. Click on OK to close the dialog.
6. The Property Pages dialog appears again. The Data source name gets added in the list. Select the DSN" ADO Data". Click on apply.



7. To set the Record Source, so click on the RecordSource tab.



Working with Advance Data Controls –

In Visual Basic 6.0 find a set of ActiveX controls specially designed for handling the database. The different ActiveX controls are:

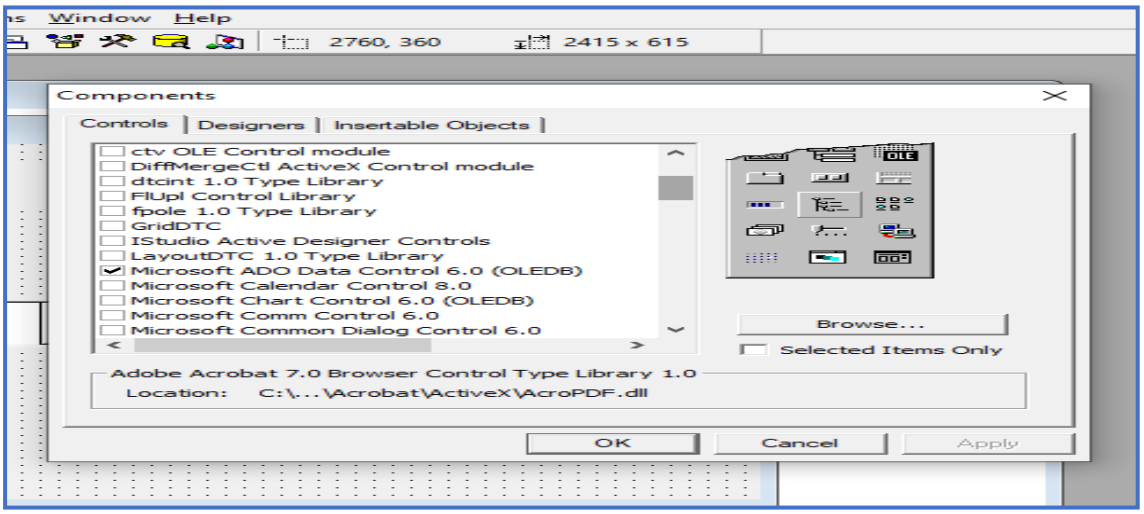
- DataList Control
- Data Combo Control
- DataGrid Control
- MSHFlexGrid Control

Datalist Control:

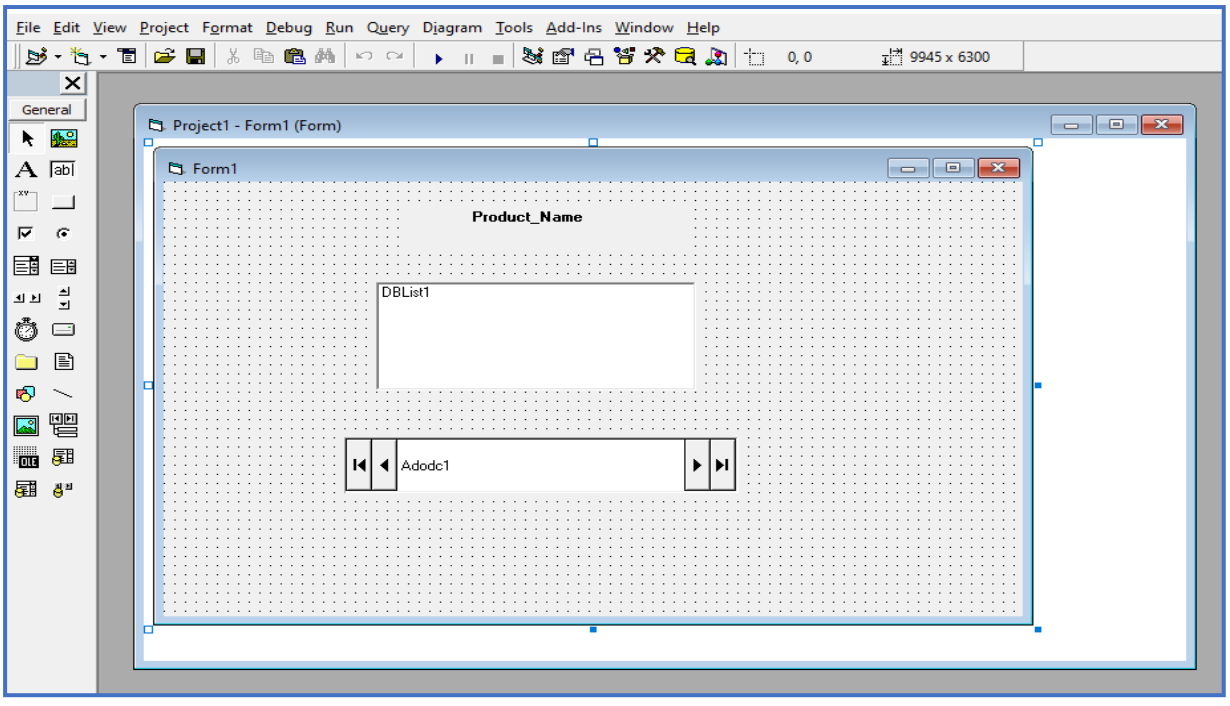
DataList Control is mainly used to display values from the selected field of the data source to which it is connected. This is an ideal control used while creating database application and generally used to display values from the table.

Now follow the steps and create a simple application, which will display the values in the DataList control.

1. Start with a new Standard EXE project. Firstly, to add the ADO Data control and the DataList control in the Toolbox. So, click on Components from the project Menu.
2. Select the option" Microsoft ADO data Control 6.0(OLEDB)and "Microsoft DataList Control 6.0(OLEDB)". Click on apply and then on Ok to close the components dialog.



3. Observe that along with the DataList control DataCombo control is also added in the Toolbox.
4. Add an ADO Data Control and a DataList control in the Form. Add a Label control and change the Caption property of Label1 as "Product name", set the Font size as 10 and style as Bold.

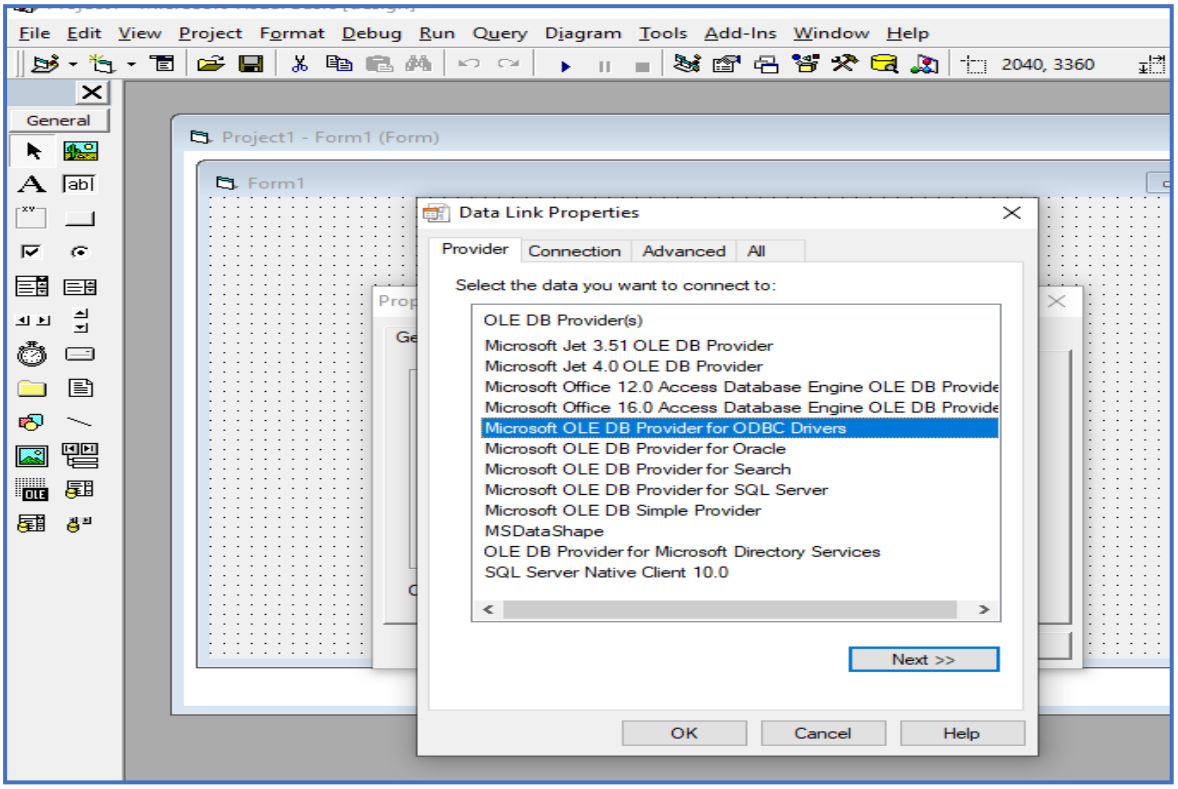


5. At run time ADO Data Control need not appear, so set its Visible property as False.

6. At this time also use the same data database, which created previously. In the DataList control can display the values of any field.

7. Initially to connect the ADO data Control with the data source. Open the property Pages dialog of ADODC1.Before connected using the ODBC DSN. This time select “Use Connection String” Click on Build button.

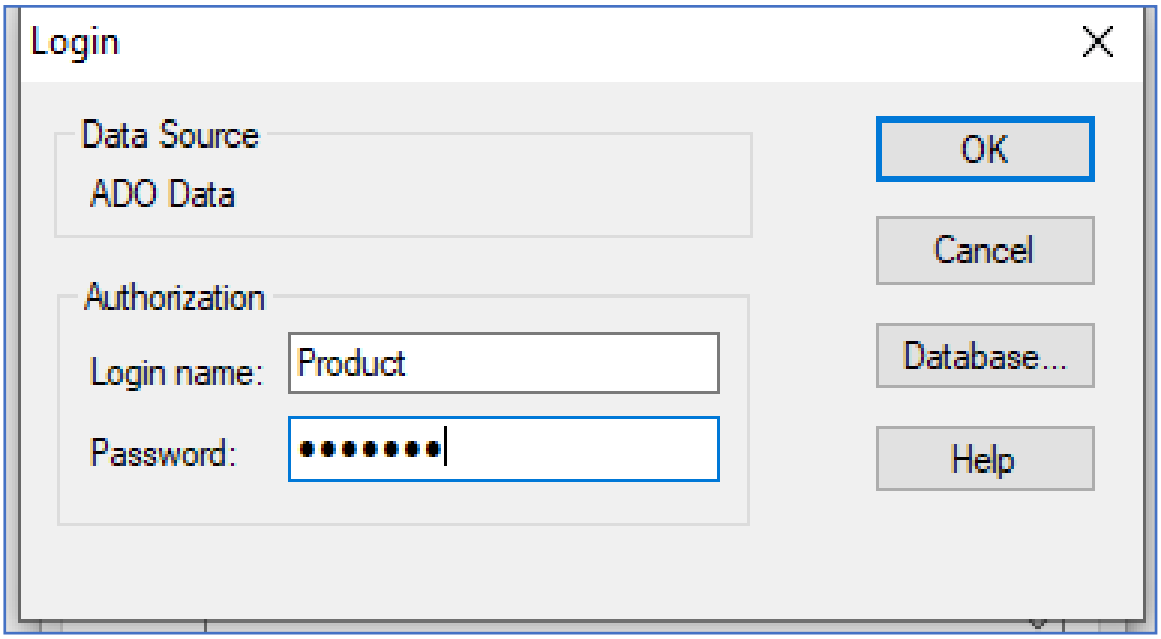
8. The Data Link properties dialog appears. Select “Microsoft OLE DB Provider for ODBC Drivers” and click on next>>.



9. Previously created the DSN. You can select the option.” Use data source name” and can select the DSN from the list. Otherwise, can select the option “Use connection string”, Click on Build button to continue.

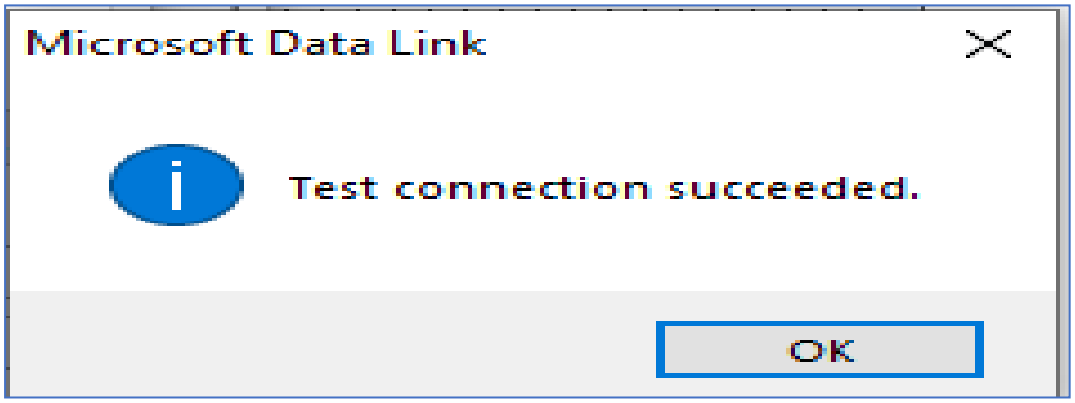
10. It invokes the Select Data Source dialog. It contains two tabs.” File Data Source “and “Machine Data Source” Data source is in the same machine, so open the machine data Source page by clicking on it.

11. While Doing the application created the DSN” ADO Data”. You find it added in the list.



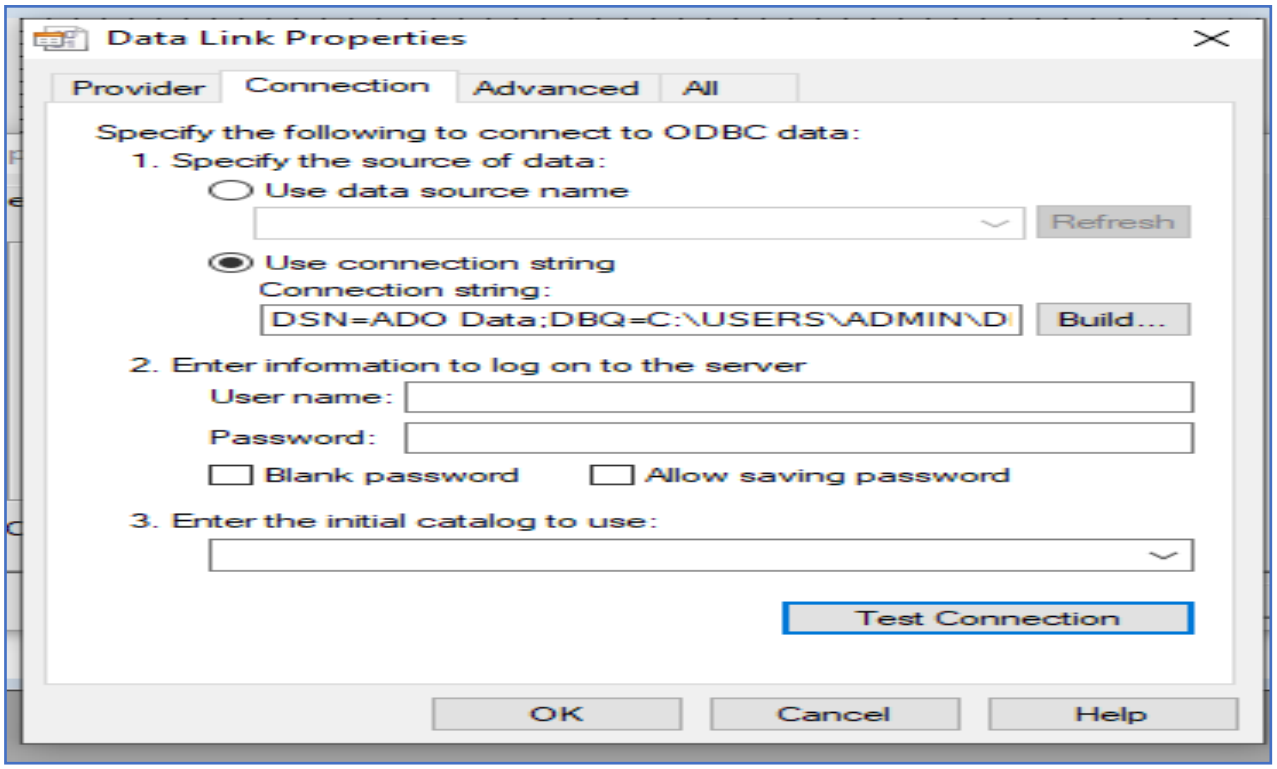
12. Select “ADO Data” and click on OK to close the dialog. A Login dialog gets displayed. If required can specify a Login name and a password. After click on OK to return to Data Link Properties dialog.

13. The connection details appear in the Connection String field. To test the connection, click on Test Connection button. The “Test connection succeeded” message gets displayed.

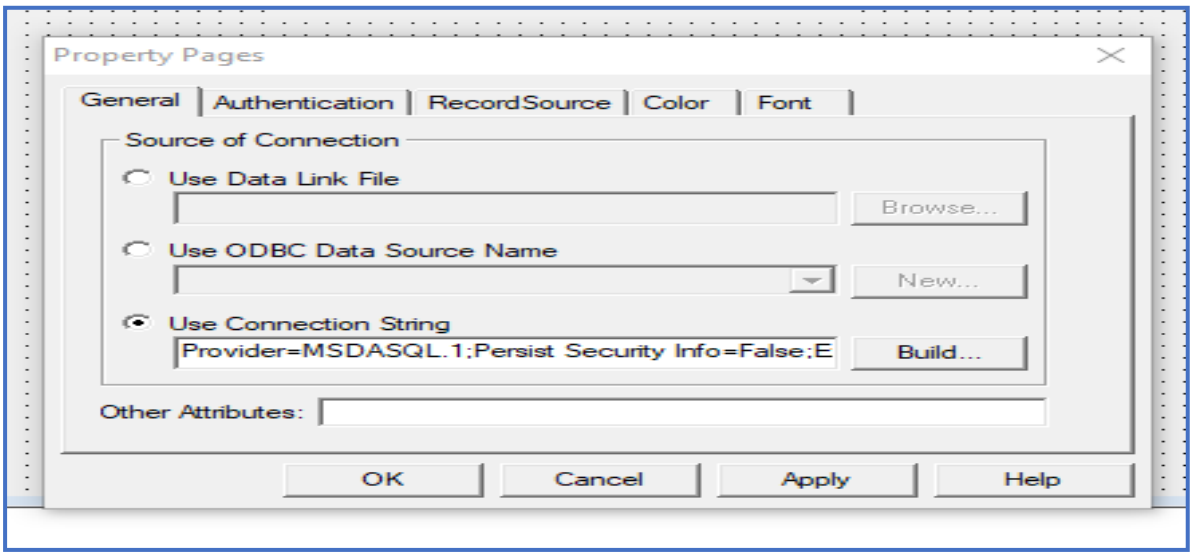


14. Click on Ok to close the message box.

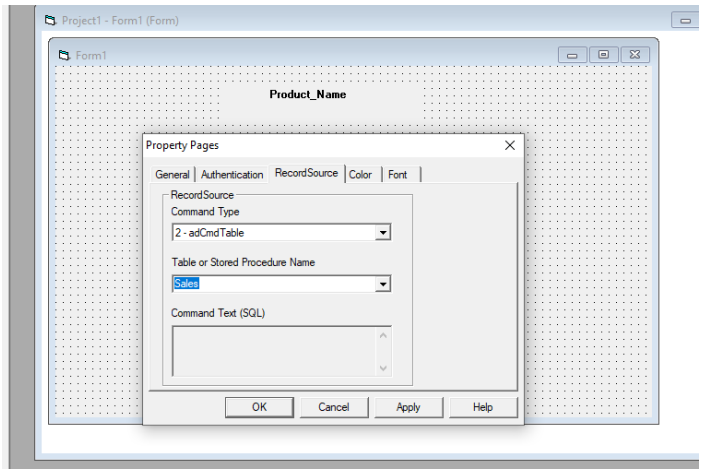
15. Again click on OK to close the data link properties dialog.



16. The connection detail appears in the Use Connection string field.



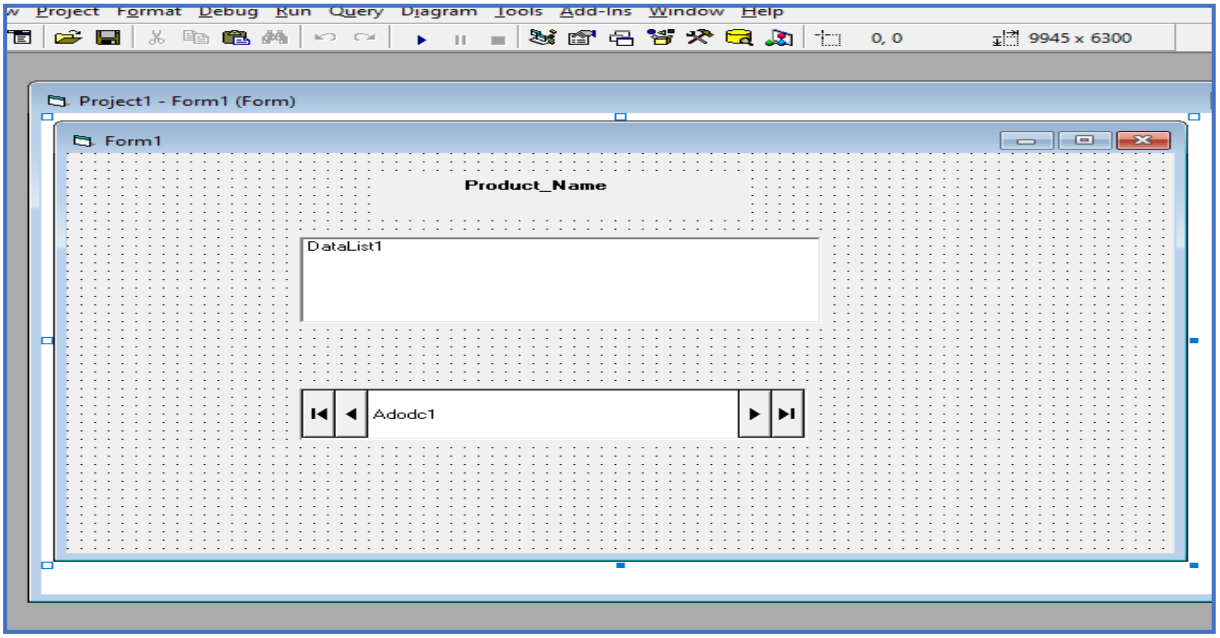
17. Open the Record Source tab by clicking on it. Choose from the Command Type list and select the table name from the Table or stored procedure name list.

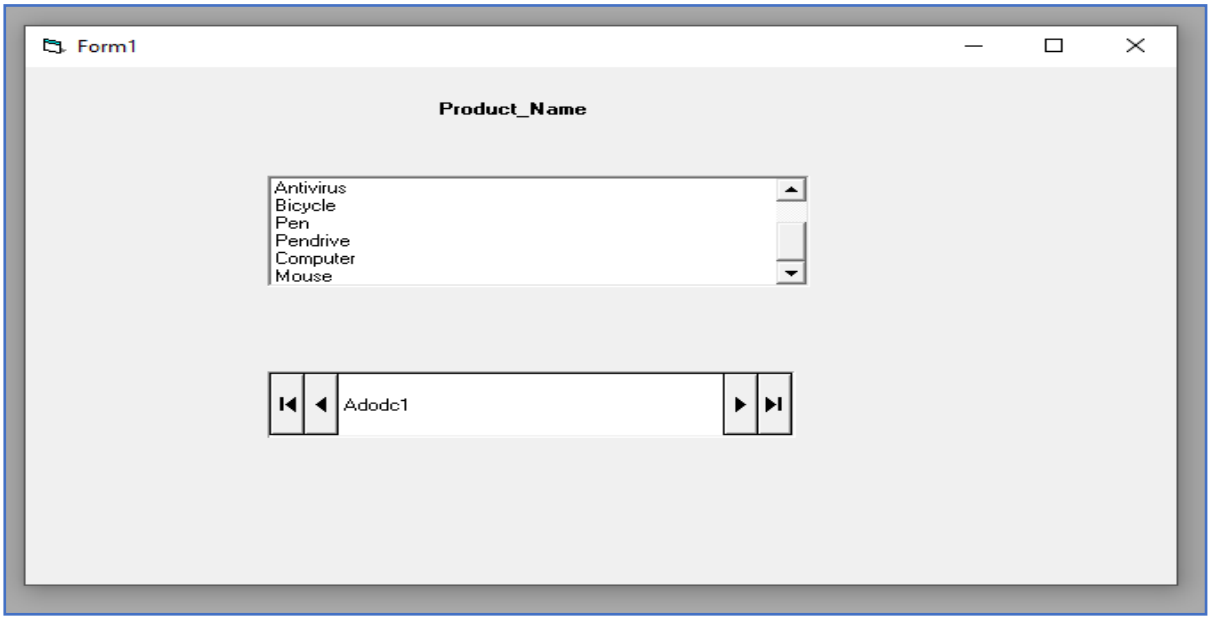


18. Click on Apply and then on OK to close the dialog.

19. After setting the properties of ADO Data control, next work is to bind the DataList control with ADO Data Control. You can do that by setting the RowSource and ListField property of DataList control.

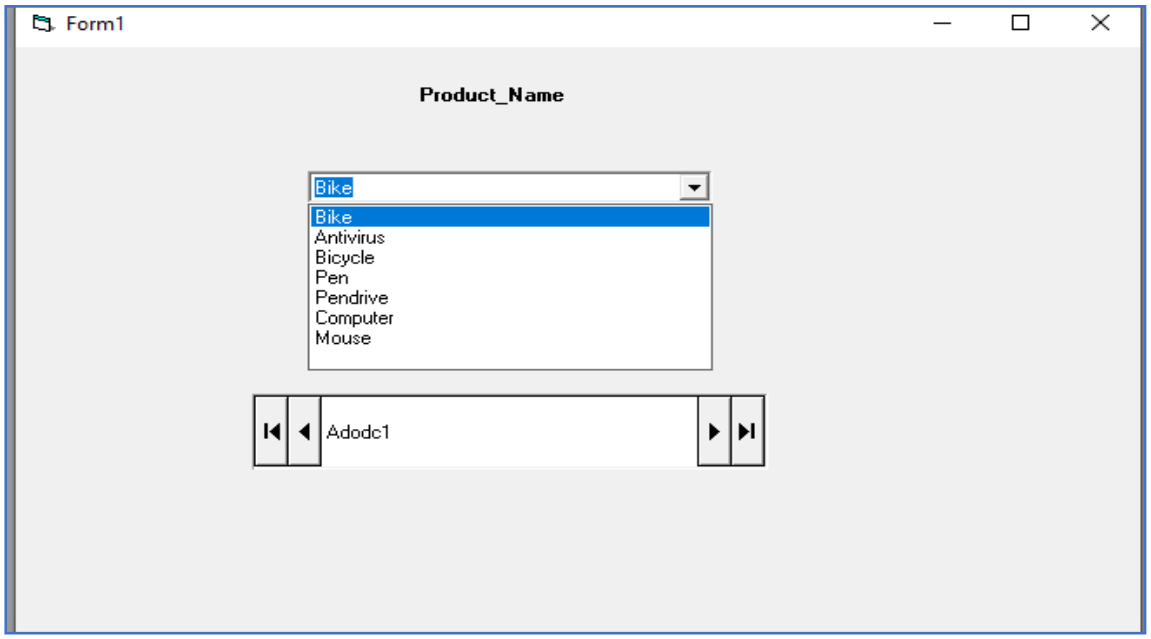
20. Select the DataList control and specify the RowSource property as ADODC1





DataCombo Control:

It can display the field values in the Data Combo control. Instead of DataList control add DataCombo control and repeat the same procedure, it has done while working with DataList control. The only difference is it lets to type and edit values, which is not possible in DataList control.

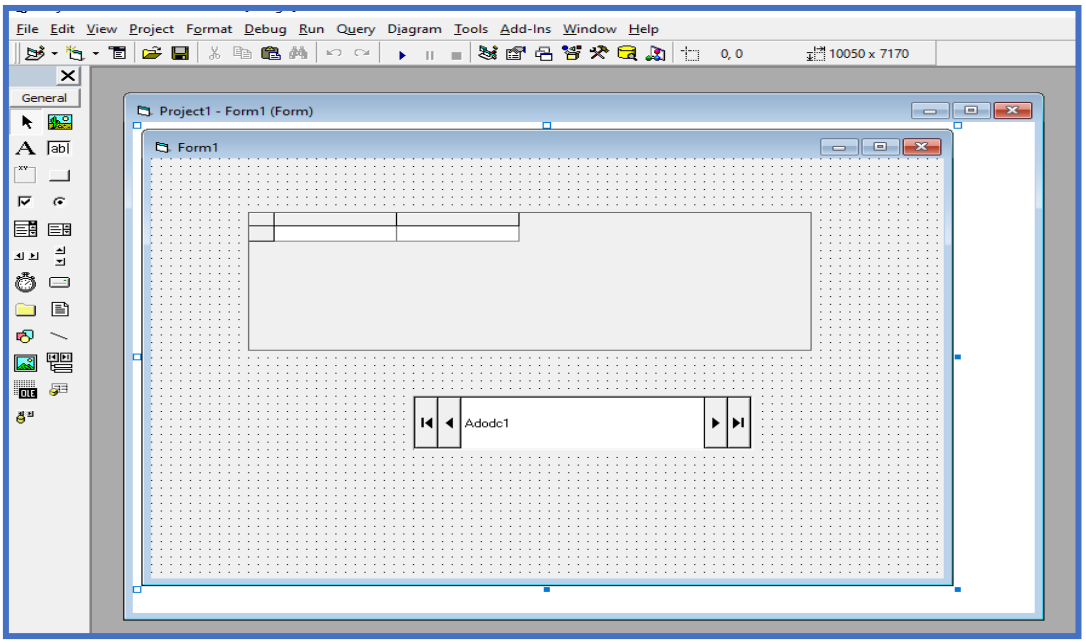


DataGrid Control:

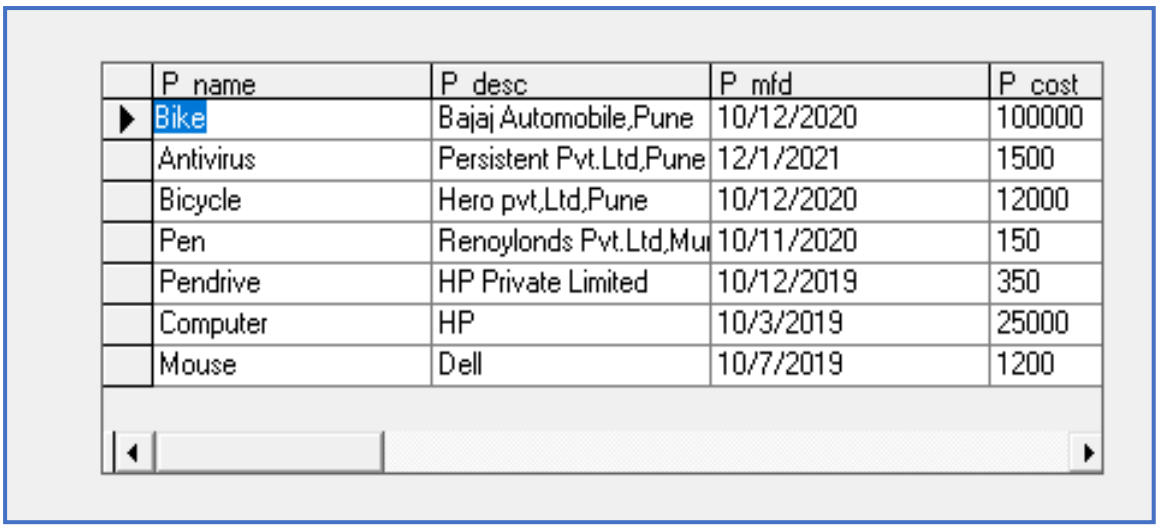
This control displays the field values in a tabular format. Each row of the data retrieved represents a record and each column represents the field of the record. The Column headers of the DataGrid control automatically displays the columns of the data source. Using the DataGrid control can set the options of adding, deleting and editing records.

Create an application where the DataGrid control will display the records of the data table.

1. Start with a new project.
2. To add the controls in the Toolbox click on Components from the Project menu. Select the option "Microsoft DataGrid Control 6.0(OLEDB) and "Microsoft ADO Data Control 6.0(OLEDB). Click on Apply and then on OK to close the dialog.
3. Add the controls in the Form. Firstly, set the Connection String property or ODBC DSN of ADODC1 with the same database.
4. Set the Record Source property of ADODC1.Again to hide ADO Data Control at runtime change the Visible property as False.
5. To bind the DataGrid control with ADO Control set the Data Source property of DataGrid1 as ADODC1.



6.Execute the application and will find the record of the table are displayed in a grid format.



P_name	P_desc	P_mfd	P_cost
Bike	Bajaj Automobile,Pune	10/12/2020	100000
Antivirus	Persistent Pvt.Ltd,Pune	12/1/2021	1500
Bicycle	Hero pvt,Ltd,Pune	10/12/2020	12000
Pen	Renoylonds Pvt.Ltd,Mu	10/11/2020	150
Pendrive	HP Private Limited	10/12/2019	350
Computer	HP	10/3/2019	25000
Mouse	Dell	10/7/2019	1200

Setting the properties of the DataGrid control

DataGrid control has several properties, which allows to change the appearance and behaviour of the control. It can set the properties at design time through Properties window or by using the Property Pages dialog. To set the properties through Property pages dialog.

1. Right click on the DataGrid control and click on the Properties option from the Popup menu. Many tabs are available using which can customize the settings of DataGrid control.
2. From the General tab can specify a Caption for the DataGrid control. Moreever can set other properties like add, delete, update etc. Just check the required options. By checking the Enabled option can specify whether want the scrollbars to be enabled or disabled. Similarly, can specify the headLines, Rowheight, Appearance etc.
3. Next tab is Keyboards, which provides some navigational properties. Check the option “Allow Arrows” that makes the arrow keys function. So, at run time the user can move from one cell to another or from one row to another.
4. To move the cell pointer from one row to another check the option “Wrap Cellpointer.
5. You can select any of the tab action from the Tab Action list.
6. You can select a column and change its Caption or DataField property from the Columns tab.

7. Layout tab contains various options, which helps in setting the grid layout.
8. To make the grid control colourful can change the BlackColor and ForeColor.Here Backcolor is changed to light yellow and ForeColor to red. After opening the Color tab can select the color from the Color Palette. Otherwise click on Edit Custom Color. It invokes Color dialog from where can choose a color.
9. Use the Font tab to set the Font properties. From the Splits tab set the split properties and to change the formats use the Format tab.
10. Click on Apply and then on Ok to close the dialog. Now have changed the properties of DataGrid control. Once again execute the application and observe the changes are reflected. It can add, delete, edit, move the cursor using the arrow keys.

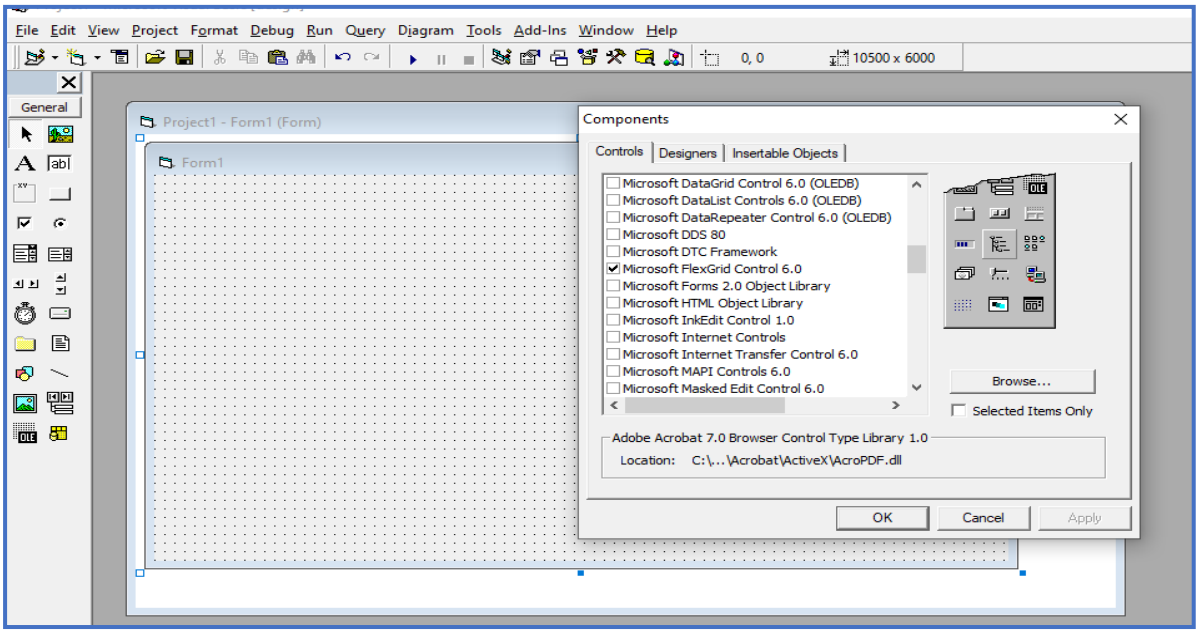
MSHFlexGrid Control

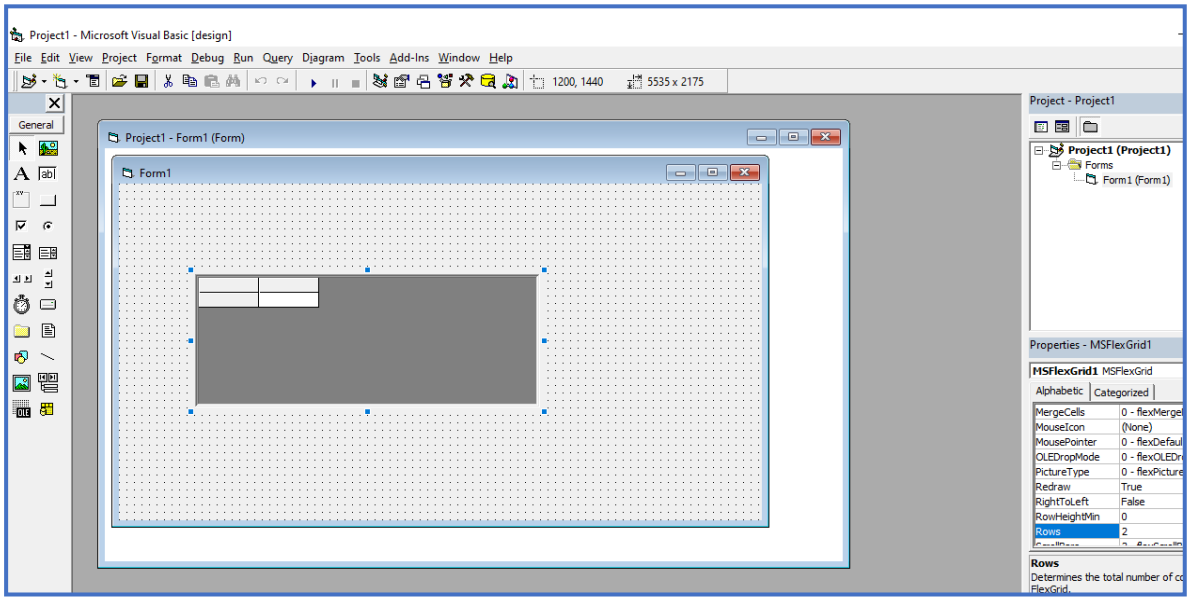
Another custom ActiveX control commonly used in data handling is MSHFlexGrid control. This also displays the data in a tabular format and in addition it allows to format, merge and sort tables. It can display images.

Now to display the records in MSHFlexGrid control follow these steps.

1.Start with a new project. Firstly, have to add the MSHFlexGrid control in the ToolBox. 2.To add click on Components from the Project menu.

3.Check the option “Microsoft hierarchical FlexGrid Control 6.0(OLEDB)” and “Microsoft ADO Data Control 6.0(OLEDB)”. Click on Apply and then on OK.

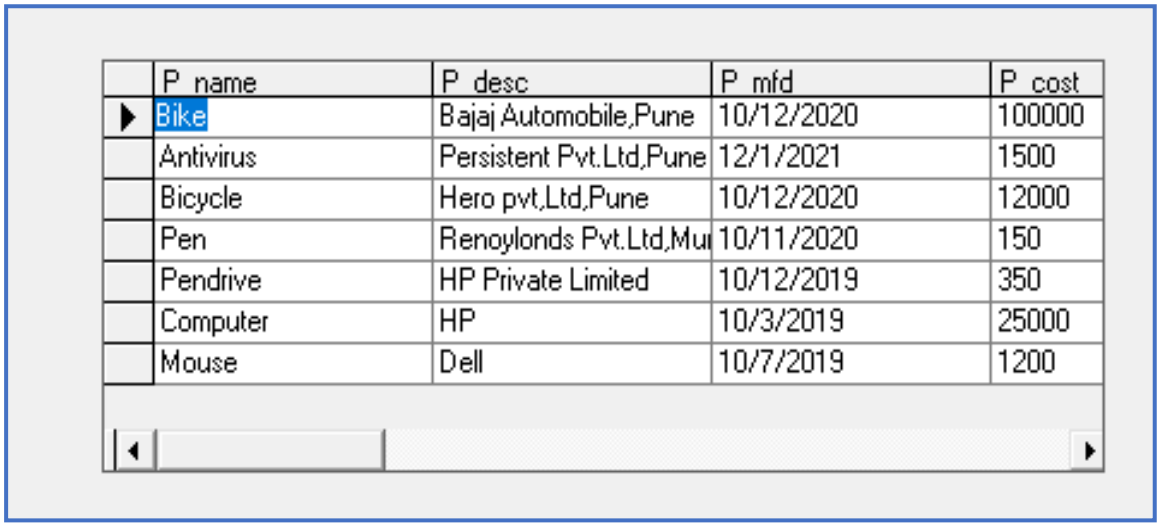




4. Add the controls in the Form. Set the Visible property of ADODC1 as False.
5. This time let's create a new database named as Books. Create a table with the following fields according to the specified structure. Set the table name as Book.

Field name	Type	Size
Year Published	Long	4
Bookid	Text	10
Bookname	Text	40
Publisher	Text	40
ISBN	Text	20

6. Add some records in the table.
7. Now set the Connection String property or ODBC DSN of ADODC1 with the books database.
8. Set the RecordSource property of ADODC1.
9. To bind the MSHFlexGrid control with ADO Data Control set the DataSource property of MSHFlexGrid control as ADODC1.
10. Run the application and check the run time view.



P_name	P_desc	P_mfd	P_cost
Bike	Bajaj Automobile,Pune	10/12/2020	100000
Antivirus	Persistent Pvt.Ltd,Pune	12/1/2021	1500
Bicycle	Hero pvt,Ltd,Pune	10/12/2020	12000
Pen	Renoylonds Pvt.Ltd,Mu	10/11/2020	150
Pendrive	HP Private Limited	10/12/2019	350
Computer	HP	10/3/2019	25000
Mouse	Dell	10/7/2019	1200

Setting the properties of MSHFlexGrid Control

- The records are displayed in the MSHFlexGrid control. You can modify the properties of the control by opening the Property Pages. To invoke, right-click on MSHFlexGrid control and click on Properties from the Popup menu.
- It will find seven tabs there. Do necessary changes. For example, from the General tab can set the Scrollbars, Fill Style etc.
- Click on Apply and then on Ok to close the dialog.
- It might happen that want to display the records in order. For example, want to display the records year wise. So, need to sort on P_desc field. This control provides the facility of sorting. So let's do sorting on P_desc coloum.