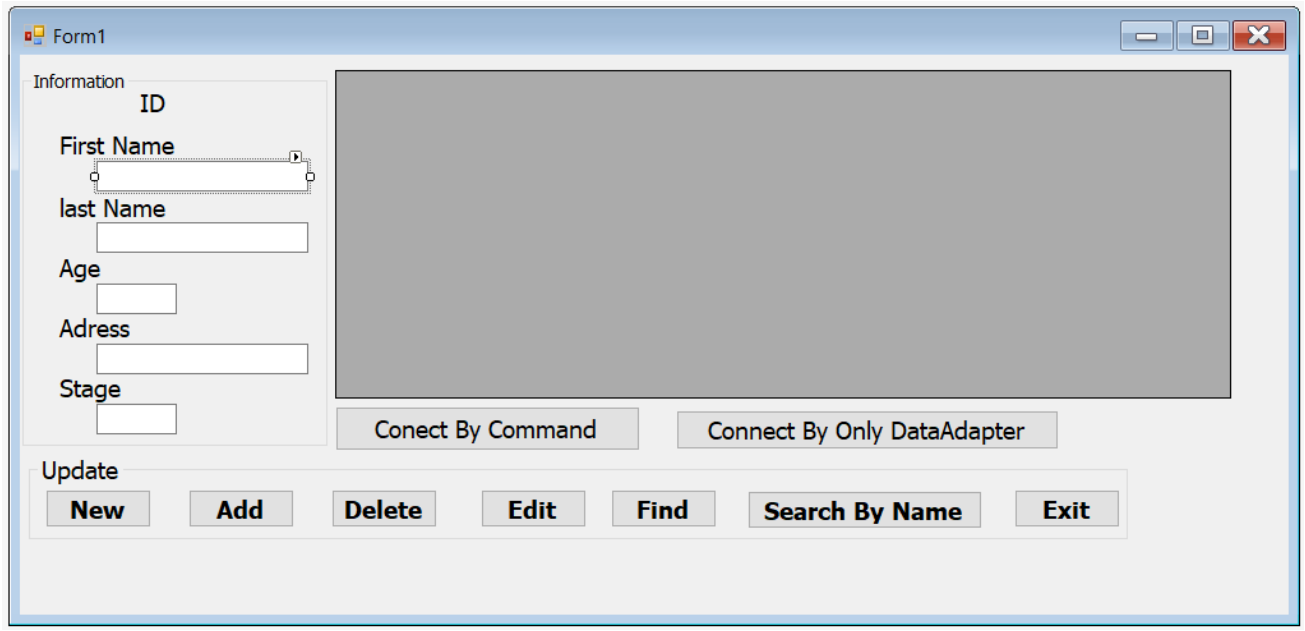
The main object in ADO.Net

1.	Connection	Use to connect database with App
2.	Command	Used to run Sql database command (select, Insert, Update, Delete) on dataabse
3.	DataReader	Use to **read only** from data source on **Connected Mode**
4.	DataSet	Collection of object contents tables and views such as database, use to collect data in memory , and we can use in **Disconnected Mode**
5.	DataTable	DataTable, which represents one table of in-memory relational data, can be created and used independently, or can be used by other .NET
6.	DataAdapte	DataAdapter serves as a bridge between a DataSet and a data source for retrieving and saving data. The DataAdapter provides this bridge by mapping Fill, which changes the data in the DataSet to match the data in the data source, and Update, which changes the data in the data source to match the data in the DataSet.
7.	Import System.Data.OleDb	The.NET Framework Data Provider for OLE DB describes a collection of classes used to access an OLE DB data source in the managed space.

Connect Database

View Data in Data Grid View tool



Step 1: Use Namespace

```
Imports System.Data.OleDb
Public Class Form1

    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load

    End Sub
End Class
```

Step 2: Creta connection

Uss **OleDbConnection** instance `to connect Access Database with App`

An instance of the OleDbConnection class in .NET Framework is supported the OLEDB Data Provider. The OleDbConnection instance takes Connection String as argument and pass the value to the Constructor statement. When the connection is established, SQL Commands may be executed, with the help of the Connection Object, to retrieve or manipulate data in the database.

```
Imports System.Data.OleDb
Public Class Form1

Dim Con As New OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0; Data Source=
D:\MyDb.accdb")

    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    End Sub
End Class
```

Provider=Microsoft.ACE.OLEDB.12.0; Data	Source=Students.accdb
Provider used to connect with Access 2007 an up	Data base path and name

Step 3: Use `OleDbDataAdapter` , `DataTable`

```
Imports System.Data.OleDb
Public Class Form1

    Dim Con As New OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0; Data Source=
D:\MyDb.accdb")

    Dim Da As OleDbDataAdapter
    Dim Dt As New DataTable

    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    End Sub
End Class
```

Step 4: Load data from database to Grid View in Form

```
Imports System.Data.OleDb
Public Class Form1
    Dim Con As New OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0; Data Source=
D:\MyDb.accdb")
    Dim Da As OleDbDataAdapter
    Dim Dt As New DataTable
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Da = New OleDbDataAdapter("Select * From Student", Con)
        Da.Fill(Dt)
        DGV.DataSource = Dt
    End Sub
End Class
```

Explaining code

`Da = New OleDbDataAdapter("Select * From Student", Con)`
Use to bring data from database by SQL select statement, with con connection

`Da.Fill(Dt)`
Full data table within Data from sql stetment

`DGV.DataSource = Dt`	Assange data to DataGridView tool

Connect By Using Command

```
        Dim com As New OleDbCommand()
        com.CommandType = CommandType.Text
        com.CommandText = "select * from Student"
        com.Connection = Con
        Dim daSet As New OleDbDataAdapter(com)

        daSet.Fill(Dt)
        DGV.DataSource = Dt
```

Explaining code

`Dim com As New OleDbCommand()`	
`com.CommandType = CommandType.Text`	command Type
`com.CommandText = "select * from Student"`	command Text
`com.Connection = Con`	Connect command with con connection
`Dim daSet As New OleDbDataAdapter(com)`	Use to bring data from database by SQL select statement in command Text

Clear Data in textbox (New)

```
Private Sub btnNew_Click(sender As Object, e As EventArgs) Handles btnNew.Click
        txtFirstName.Text = ""
        TxtLastName.Text = ""
        txtAge.Text = ""
        TxtAdress.Text = ""
        TxtStage.Text = ""
    End Sub
```

Insert Data to Database (Insert)

```
    Private Sub btnAdd_Click(sender As Object, e As EventArgs) Handles btnAdd.Click
        Dim Cmd As New OleDbCommand

        Try
            Cmd = New OleDbCommand("Insert Into Student (FristName , LastName, Age,
            Adress, Stage) Values ('" & txtFirstName.Text & "','" & TxtLastName.Text
            & "','" & txtAge.Text & "', '" & TxtAdress.Text & "','" & TxtStage.Text &
            "')", Con)

            Con.Open()
            Cmd.ExecuteNonQuery()
            Con.Close()
            MsgBox("Student added successfully ", MsgBoxStyle.Information)
            Dt.Clear()
```

```
            btnConnectByCommand_Click(sender, e)
            btnNew_Click(sender, e)
        Catch ex As Exception
            MsgBox("Some errors was occured !", MsgBoxStyle.Critical)
        End Try
    End Sub
```

Explaining code

Cmd = New OleDbCommand("Insert Into Student (FristName , LastName, Age, Adress, Stage) Values ('" & txtFirstName.Text & "','" & TxtLastName.Text & "','" & txtAge.Text & "', '" & TxtAdress.Text & "','" & TxtStage.Text & "')", Con)	Sql Statement to add data to table students Data from TextBox
Con.Open()	Open Connection with database
Cmd.ExecuteNonQuery()	Execute Sql command
Con.Close()	Close the database Connection
Dt.Clear()	Delete all previous data in data table
btnConnectByCommand_Click(sender, e)	Run "Connect By Only DataAdapter" Button to view data with a new record
btnNew_Click(sender, e)	Run "New" Button to clear all text box after add data

Delete from Database (Delete)

```
Private Sub btnDelete_Click(sender As Object, e As EventArgs) Handles btnDelete.Click
        Dim Cmd As New OleDbCommand
        Dim Input As String
        Try
            Input = InputBox("Enter The ID of Student To Delete !", "Delete")
             Cmd = New OleDbCommand("Delete * From Student where ID=" & Input & "",
             Con)
            Con.Open()
            Cmd.ExecuteNonQuery()
            Con.Close()
            MsgBox("Student Deleted Successfully", MsgBoxStyle.Information)
            Dt.Clear()
            btnConnectByCommand_Click(sender, e)
        Catch ex As Exception
            MsgBox("Some errors was occured !", MsgBoxStyle.Critical)
        End Try
    End Sub
```

Explaining code

`Input = InputBox("Enter The ID of Student To Delete !", "Delete")`	Use input box to enter students id
`Cmd = New OleDbCommand("Delete * From Student where ID=" & Input & "", Con)`	Use SQl Delete statements to delete data

Find from Database (Find)

```
Private Sub btnFind_Click(sender As Object, e As EventArgs) Handles btnFind.Click
        Dim Cmd As New OleDbCommand
        Dim Dr As OleDbDataReader
        Dim Input As String
        Try
            Input = InputBox("Enter The ID of Student To Search !", "Search")
            Cmd = New OleDbCommand("select * from Student where ID=" & Input & "", Con)
            Con.Open()
            Dr = Cmd.ExecuteReader
            While Dr.Read()
                LblID.Text = Dr(0)
                txtFirstName.Text = Dr(1)
                TxtLastName.Text = Dr(2)
                txtAge.Text = Dr(3)
                TxtAdress.Text = Dr(4)
                TxtStage.Text = Dr(5)
            End While
            Dr.Close()
            Con.Close()
        Catch ex As Exception
            MsgBox("Some errors was occured !", MsgBoxStyle.Critical)
        End Try
    End Sub
```

Explaining Code

`Dim Cmd As New OleDbCommand`	New command
`Dim Dr As OleDbDataReader`	Provides a way of reading a forward-only stream of data rows from a data source.
`Dim Input As String` `Input = InputBox("Enter The ID of Student To Search !", "Search")`	Input name
`Cmd = New OleDbCommand("select * from Student where ID=" & Input & "", Con)`	Use sql Select to `Find information by ID`
`Dr = Cmd.ExecuteReader`	Return the result of command `Execute in the (Data Reader DR)`

`While Dr.Read()`
`LblID.Text = Dr(0)`
`txtFirstName.Text = Dr(1)`
	This part of the code used to loop for all data in (DR), then Assign data to text box sequentially

`TxtLastName.Text = Dr(2)`
`txtAge.Text = Dr(3)`
`TxtAdress.Text = Dr(4)`
`TxtStage.Text = Dr(5)`
`End While`	

`Dr.Close()`	Close Data reader after finishing gathering data

Search By Name

In this part can find all students have same name, then saved in data table then set data table to Data Grid View

```
Private Sub btnSearchByName_Click(sender As Object, e As EventArgs) Handles
btnSearchByName.Click
        Dim Cmd As New OleDbCommand
        Dim Dr As OleDbDataReader
        Dim DtN As New DataTable

        DtN.Columns.Add("ID")
        DtN.Columns.Add("FirstName")
        DtN.Columns.Add("LastName")
        DtN.Columns.Add("Age")
        DtN.Columns.Add("Adress")
        DtN.Columns.Add("Stage")

        Dim Input As String
        Try
            Input = InputBox("Enter The ID of Student To Search !", "Search")
            Cmd = New OleDbCommand("select * from Student where FristName='" & Input
            & "'", Con)
            Con.Open()
            Dr = Cmd.ExecuteReader
            While Dr.Read()
                DtN.Rows.Add(Dr(0), Dr(1), Dr(2), Dr(3), Dr(4), Dr(5))
            End While
            Dr.Close()
            Con.Close()
            DGV.DataSource = DtN
        Catch ex As Exception
            MsgBox("Some errors was occured !", MsgBoxStyle.Critical)
        End Try
    End Sub
```

Explaining Code

`Dim DtN As New DataTable`	Define New temp DataTable

`DtN.Columns.Add("ID")`
`DtN.Columns.Add("FirstName")`
`DtN.Columns.Add("LastName")`
`DtN.Columns.Add("Age")`
`DtN.Columns.Add("Adress")`
`DtN.Columns.Add("Stage")`
	Add Columns to dataTable as in our real table

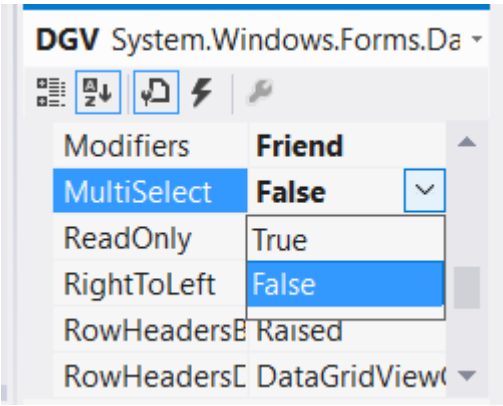
`Input = InputBox("Enter The ID of Student To Search !", "Search")`	Input name
`Cmd = New OleDbCommand("select * from Student where FristName='" & Input & "'", Con)`	Used sql statement to find all names by filter FirstName field
`DtN.Rows.Add(Dr(0), Dr(1), Dr(2), Dr(3), Dr(4), Dr(5))`	Add row to the new datatable (DtN) depend on data in Dr sequentially
`DGV.DataSource = DtN`	Assange DtN to data grid view

Update

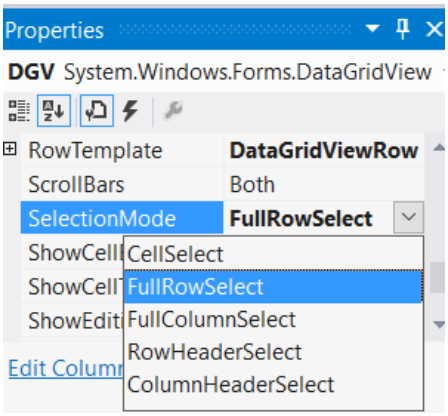
In this section, we explain how to edit data by enable click on grid view tool to get data, and put data selected is then added to the corresponding text box.

First

A. Stop MultiSelected on DataGridView



B. Go to section on mode: chose **FullRowSelect**



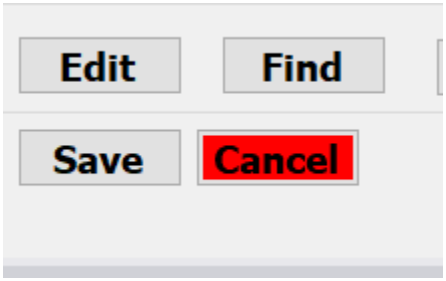
C. Add below code to (`CellClick`) Event

```
    Private Sub DGV_CellClick(sender As Object, e As DataGridViewCellEventArgs)
Handles DGV.CellClick
        If EidtFalg = 1 Then

            Label7.Text = DGV.CurrentRow.Cells(0).Value.ToString
            txtFirstName.Text = DGV.CurrentRow.Cells(1).Value.ToString
            TxtLastName.Text = DGV.CurrentRow.Cells(2).Value.ToString
            txtAge.Text = DGV.CurrentRow.Cells(3).Value.ToString
            TxtAdress.Text = DGV.CurrentRow.Cells(4).Value.ToString
            TxtStage.Text = DGV.CurrentRow.Cells(5).Value.ToString
        End If

    End Sub
```

Explaining Code

Get the data in the selected Row and cell the put it in the text box

D. Add this buttons (Save, Cancel)



Cancel Button: used to Cancel edit mod

```
    Private Sub btnCancel_Click(sender As Object, e As EventArgs) Handles
btnCancel.Click
        EidtFalg = 0
        btnSave.Visible = False
        btnCancel.Visible = False
```

```
        btnNew_Click(sender, e)
    End Sub
```

Save Button : used to save changes into database

```
  Private Sub btnSave_Click(sender As Object, e As EventArgs) Handles
btnSave.Click
        Dim Cmd As New OleDbCommand

        Try
            Cmd = New OleDbCommand("  UPDATE  Student set FristName ='" &
            txtFirstName.Text & "', LastName = '" & TxtLastName.Text & "', Age ='"
            & txtAge.Text & "', Adress = '" & TxtAdress.Text & "', Stage ='" &
            TxtStage.Text & "'  WHERE id = " & Label7.Text, Con)
                Con.Open()
            Cmd.ExecuteNonQuery()
            Con.Close()
            MsgBox("Student updated successfully ", MsgBoxStyle.Information)
            Dt.Clear()
            btnConnectByCommand_Click(sender, e)
            btnNew_Click(sender, e)

            EidtFalg = 0
            btnSave.Visible = False

        Catch ex As Exception
            MsgBox("Some errors was occured !", MsgBoxStyle.Critical)
        End Try
    End Sub
```