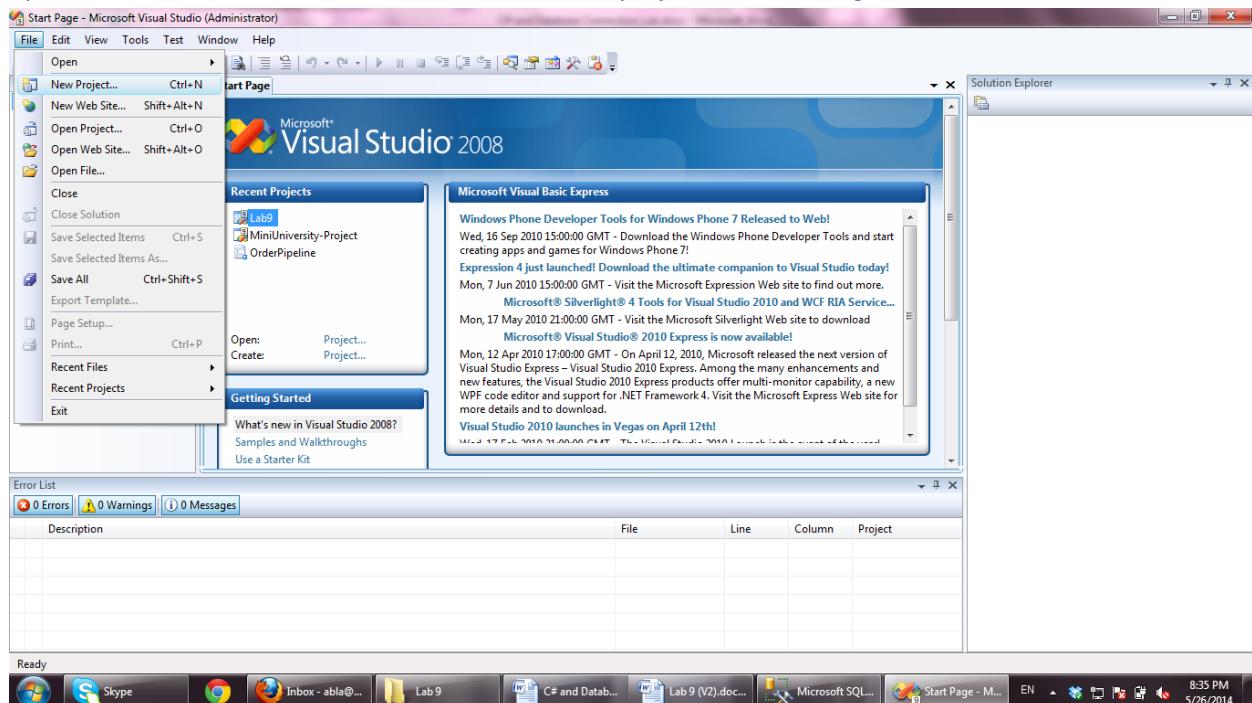


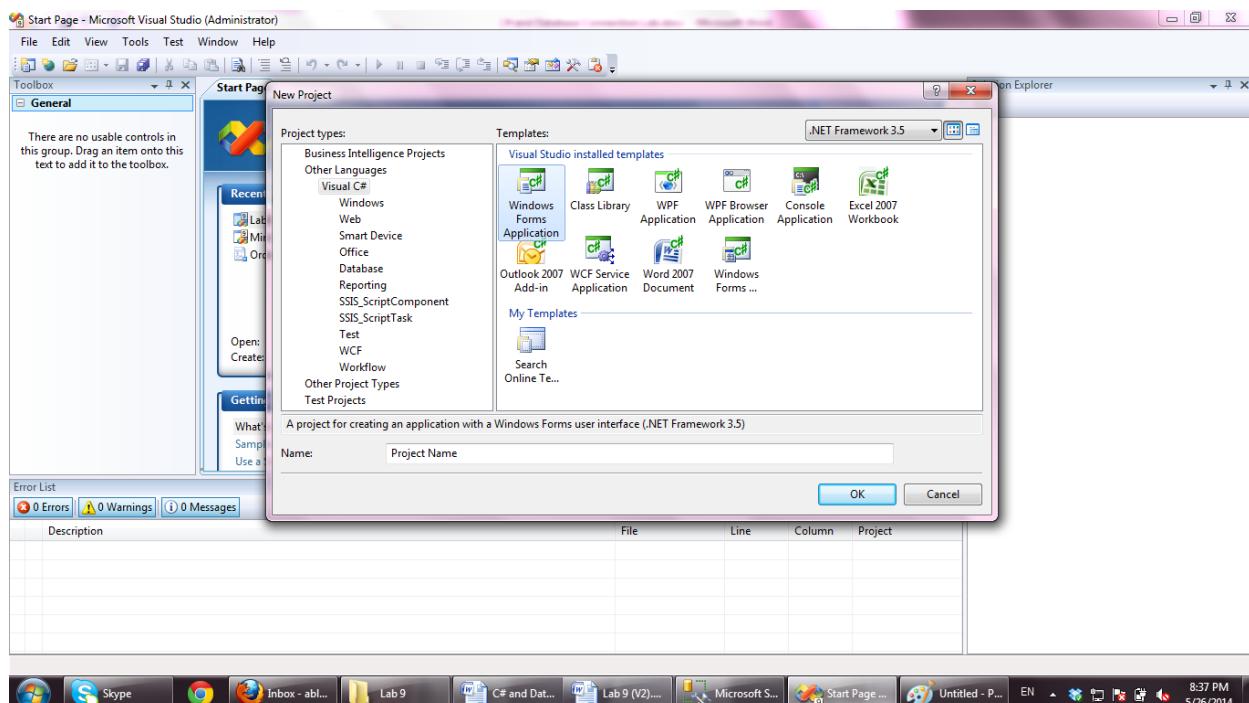
Start

Step by Step to finish your Project: (Steps explained in the last lab in details):

1. Open SQL Server and connect on your server name.
2. Create new database with your project name
3. New query on this database and execute the Script generated from your physical model using power designer
4. Insert data in these tables at least 5 rows in each table by insert query statement or manually.
5. Keep SQL Server opened
6. Open Visual Studio and from file menu select “new project” as following

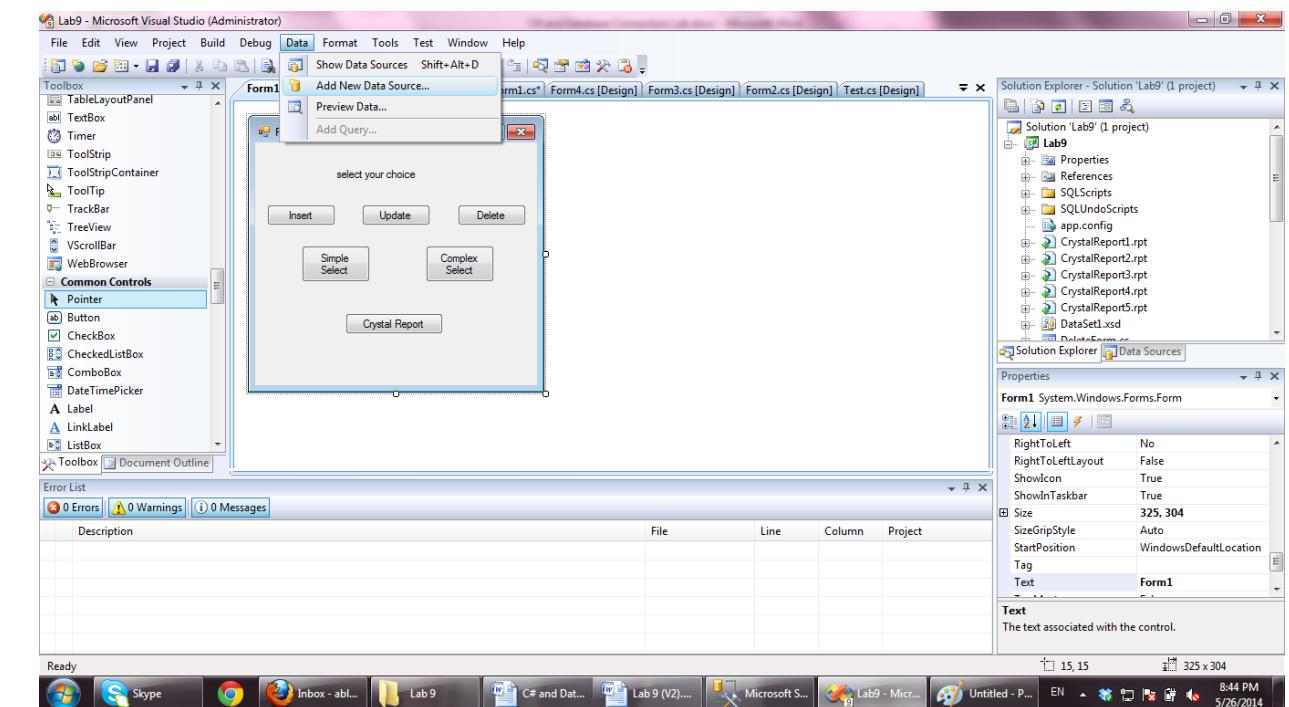


7. Choose the type of the project as “Windows Application Form” and change the name to your project name as following

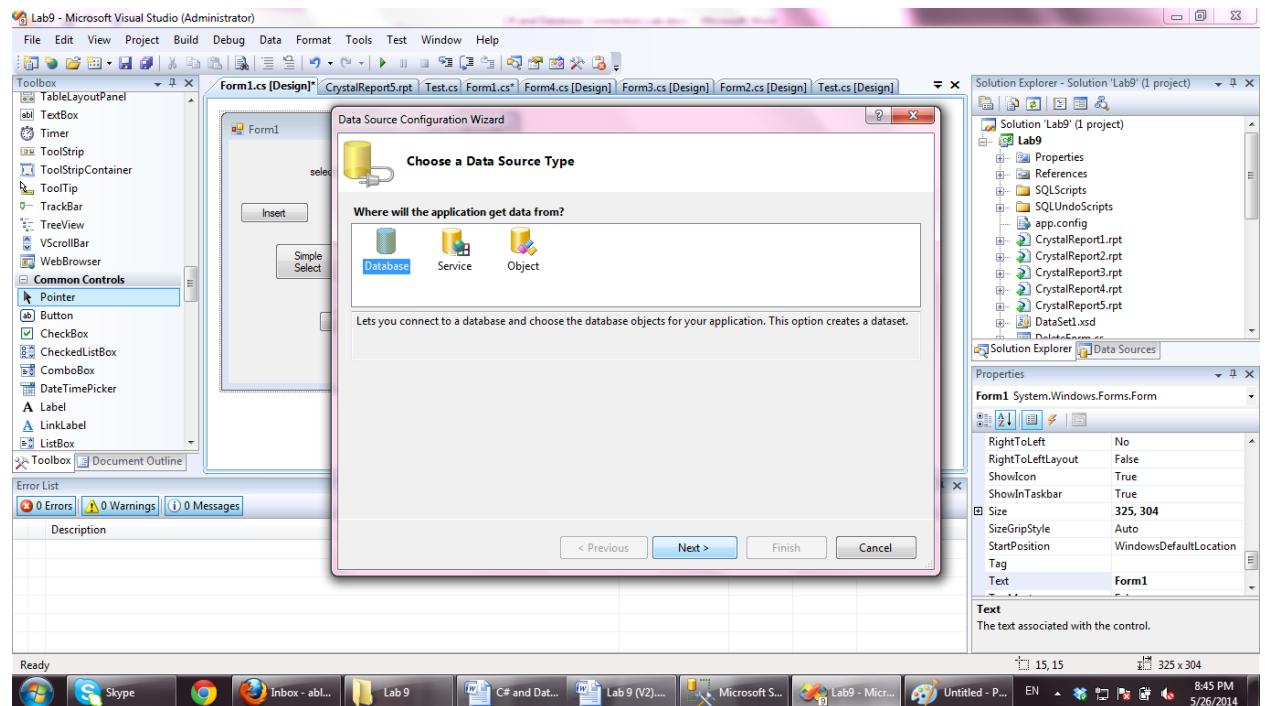


8. Create new connection to your database in SQL by following these steps:

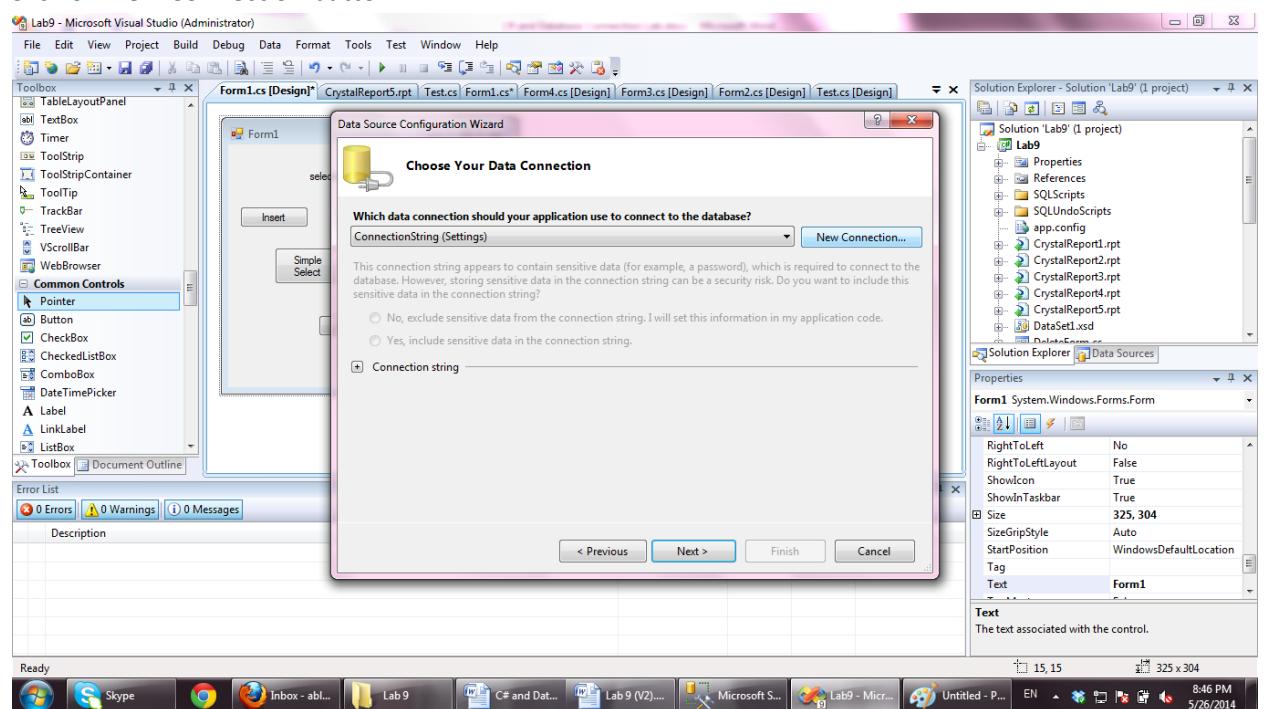
- From the Data menu select “Add new Data Source”



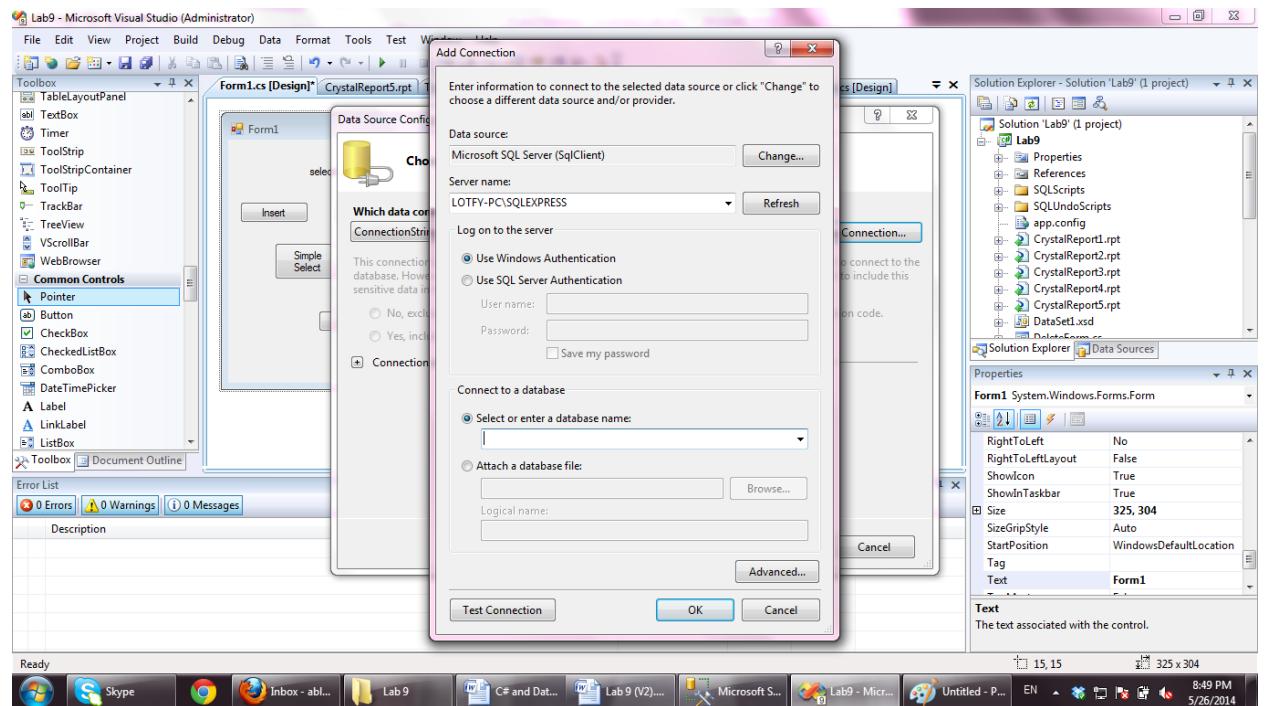
- Choose Database and click Next



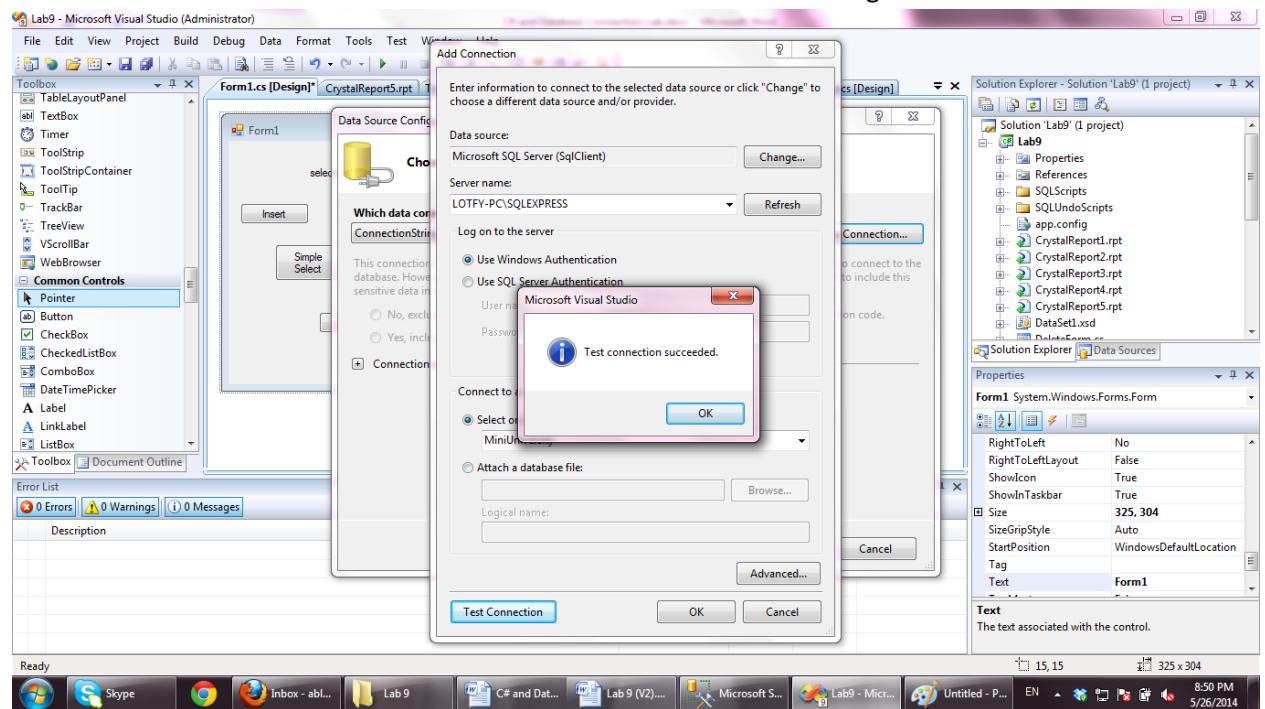
c. Click on New Connection button



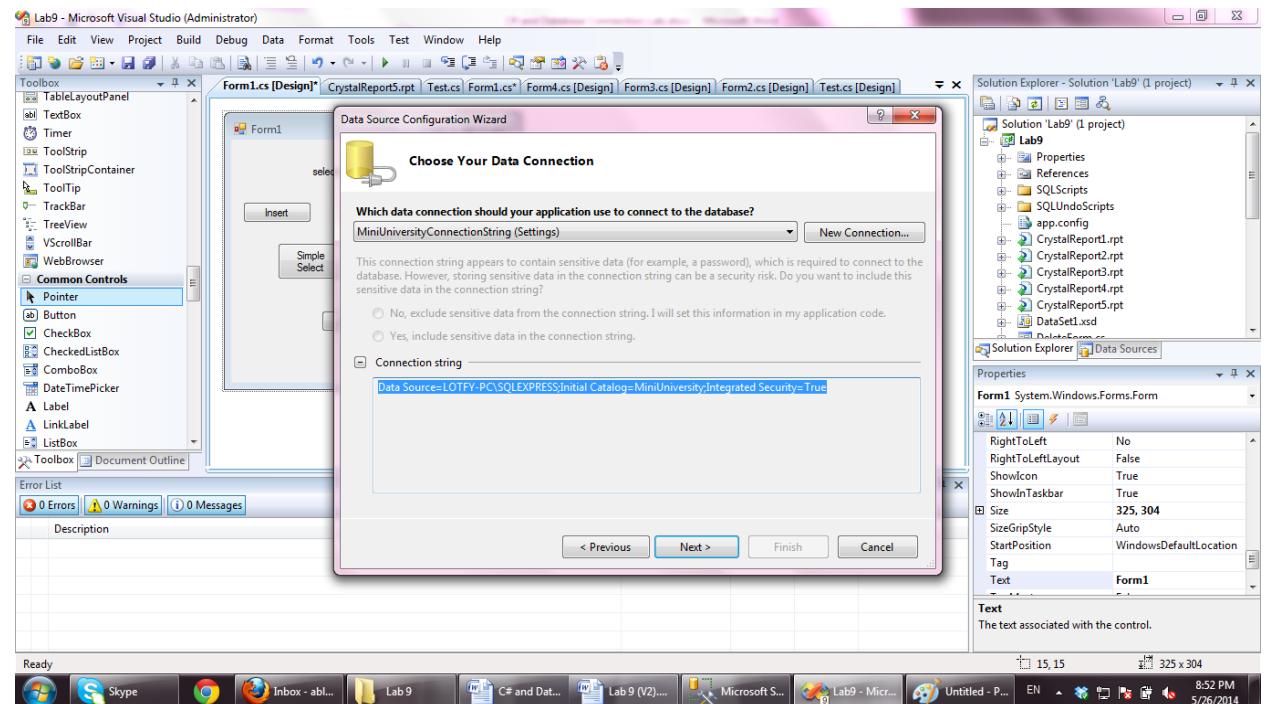
d. Write the server name as it is from the SQL Server and then choose the database that you will use in the project which is the database you already created in step 2 and 3



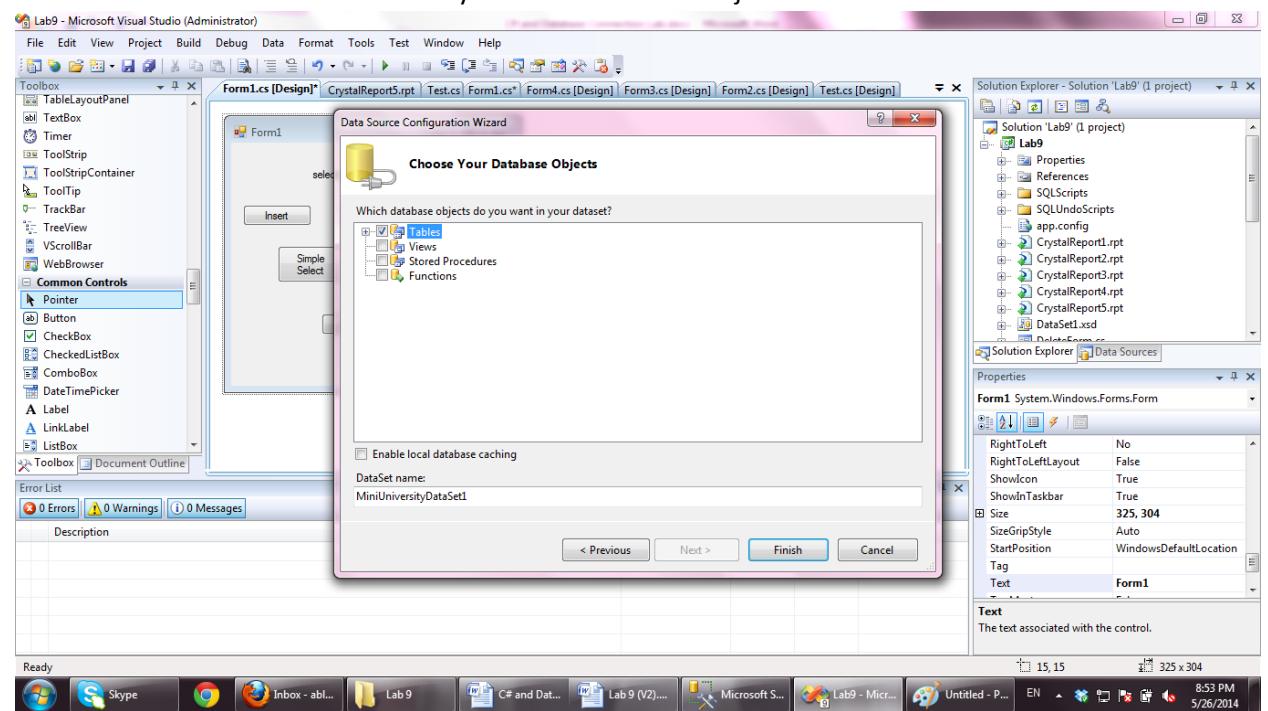
- e. Click on the “Test Connection” button to check the connection is working well



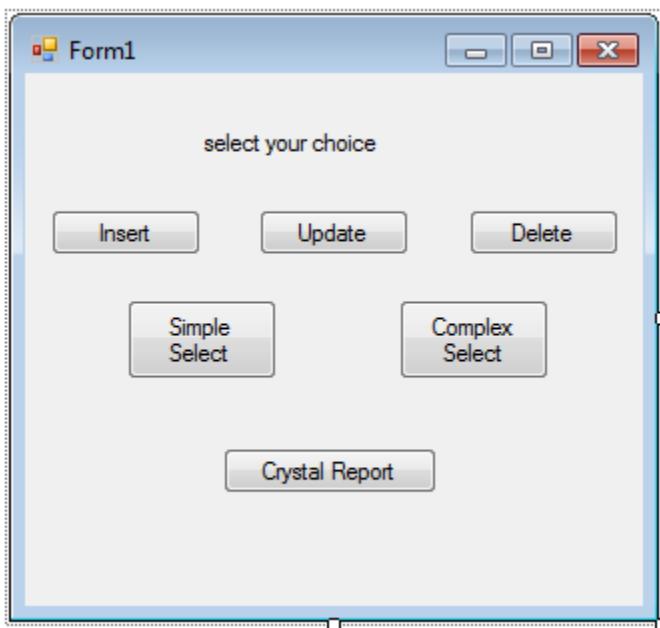
- f. Open the + sign beside the word “Connection String” and copy the connection string of your database and save it in a Text file.



- g. Click next and then choose Tables only from the database objects and then click Finish.

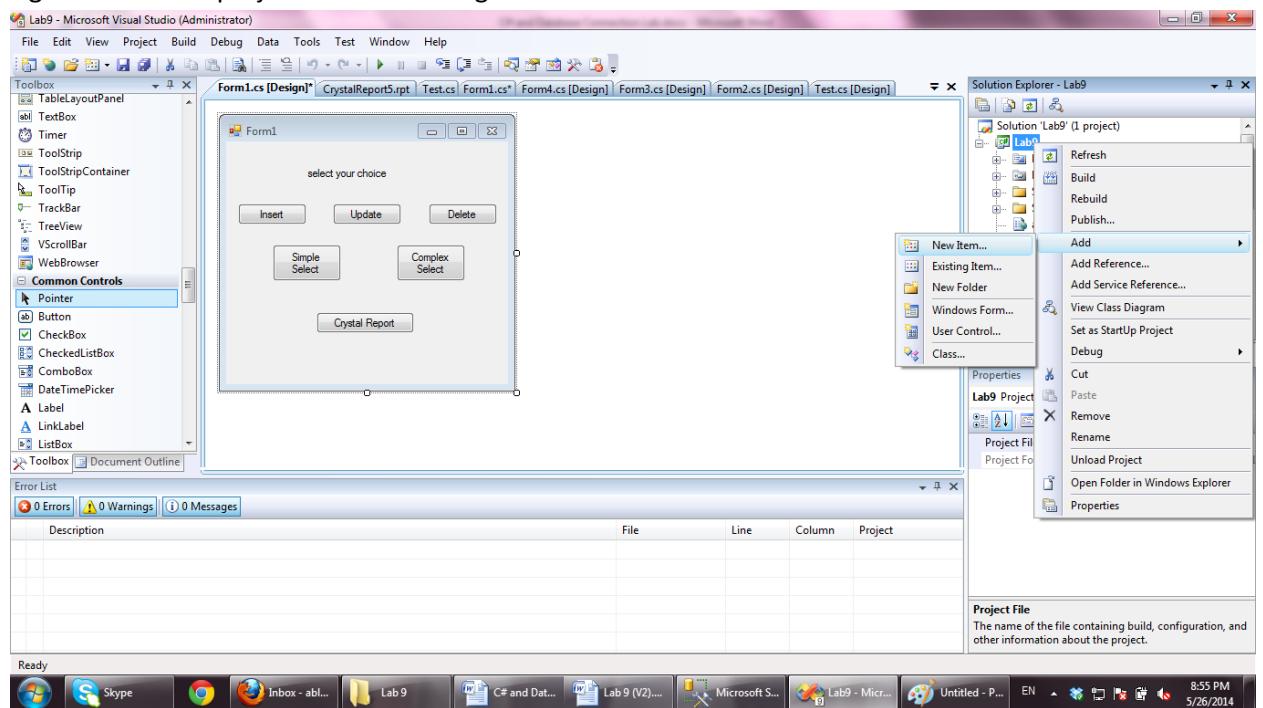


9. Design your main form as a master form that I can use it to navigate to all other forms required in the project and back to it again from any other form (you can do that by using buttons for example as shown in the picture)

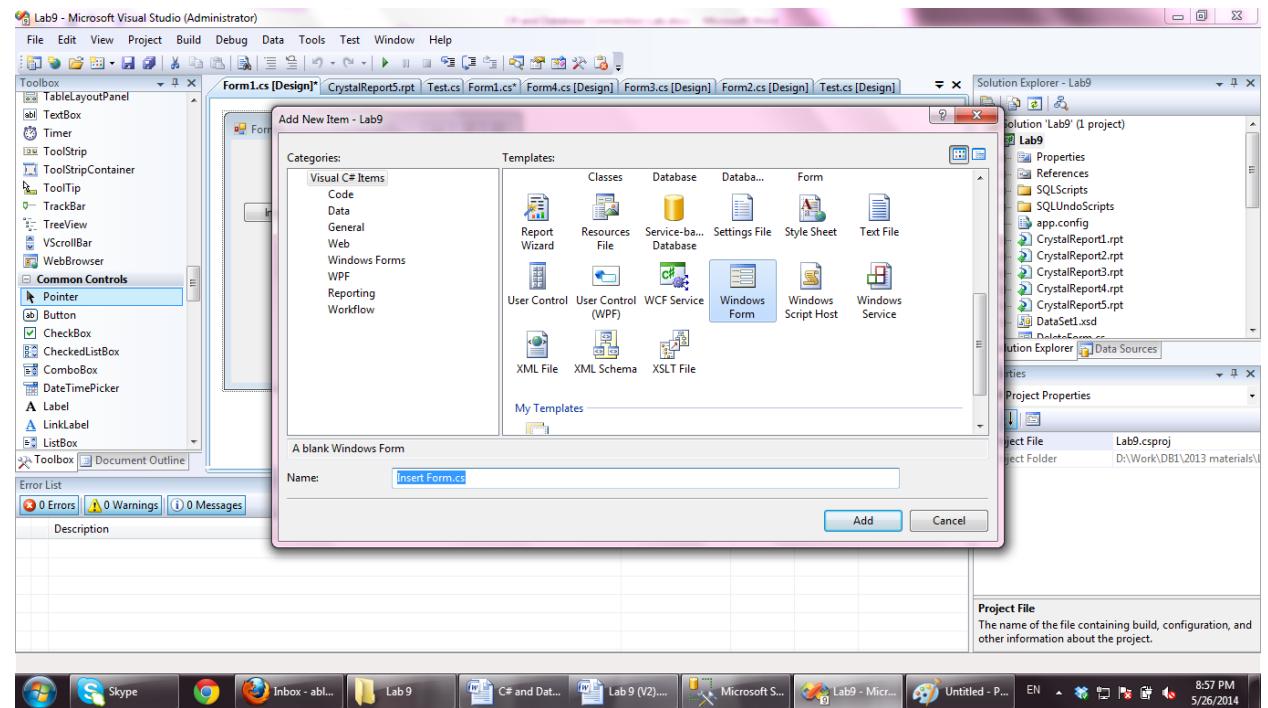


10. To do the Insert Function you should add new form to the project by follow these steps:

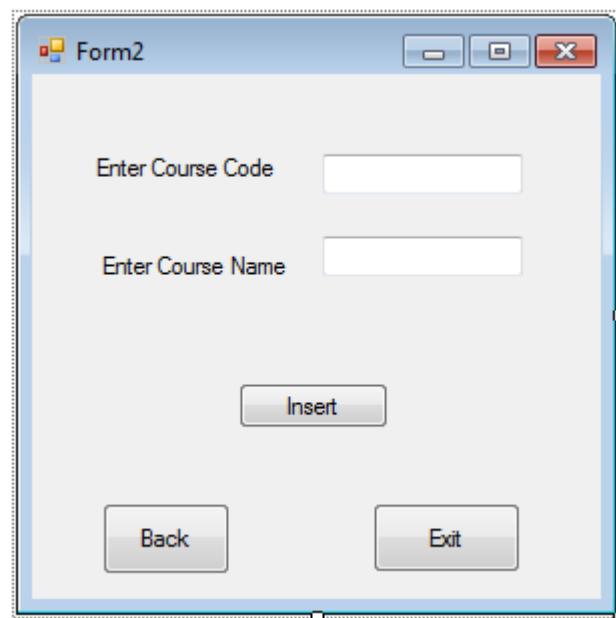
- Right click on the project name in the right side and select Add→New Item



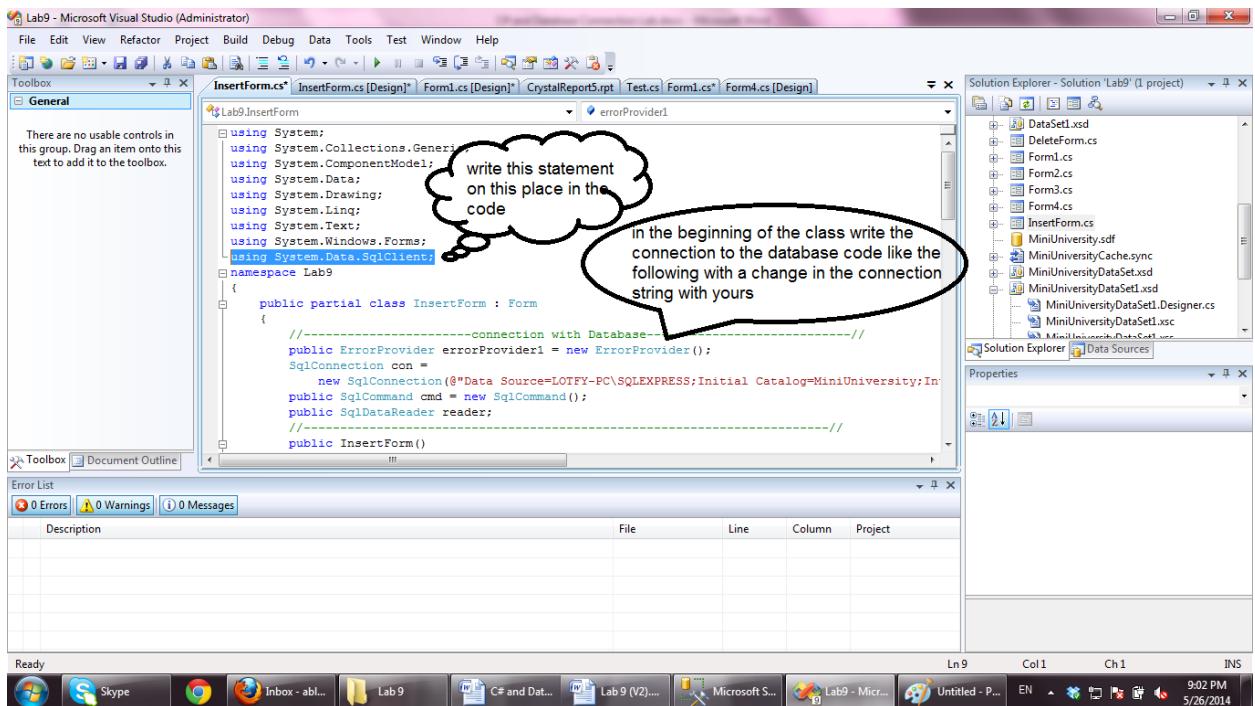
- Choose windows form and change its name to the form that you want to design which is here "InsertForm"



- c. Select a table from your project tables that you will apply the insert function on it and design the form with number of labels and text boxes equals to the number of the attributes included in that table. (Example here will be on a table called Course and include only two attributes). You will need to add a Back button and Exit button in each form to be able to navigate through forms without closing the run each time.



- d. Double Click on the Insert button and write the following code by these same steps:
- Write the using statement and the connection string code like this picture



And this is the code that is in the image

```
//-----connection with Database----- //
```

```

public ErrorProvider errorProvider1 = new ErrorProvider();
SqlConnection con = new SqlConnection(@"Data Source=LOTFY-
PC\SQLEXPRESS;Initial Catalog=MiniUniversity;Integrated
Security=True");
public SqlCommand cmd = new SqlCommand();
public SqlDataReader reader;
//----- //
```

ii. In the function of the Insert button write the following code by changing the Command with yours (According to the table you choose to insert data in it)

```

if (textBox1.Text.Trim() == "" || textBox1.Text.Trim() == "")
{
    errorProvider1.SetError(textBox1, " Please Enter Valid Course Code and
a course name ");
    errorProvider1.BlinkStyle = ErrorBlinkStyle.AlwaysBlink;
}
else
{
    con.Open();
    errorProvider1.Clear();
    cmd.Connection = con;

    SqlCommand myCommand = new SqlCommand("insert into course values ('" +
    textBox1.Text.ToString() + "','" + textBox2.Text.ToString() + "')", con);

    int success = myCommand.ExecuteNonQuery();
    if (success == 1)
        MessageBox.Show(success + " row has been inserted ");
    con.Close();
}

```

- iii. Back to the Design of the form and Double Click on the Back button and write the following code (Form 1 here is my master form)

```
Form1 I = new Form1();  
this.Hide();  
I.Show();
```

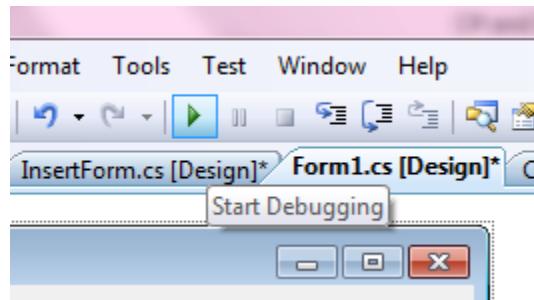
- iv. Back to the Design of the form and Double Click on the Exit button and write the following code

```
Application.Exit();
```

- v. To execute the insert form we should go back to the master form (Form 1 here) and double click on the insert button and call the insert form. So when this button clicked the insert form will appear and then we can use it.

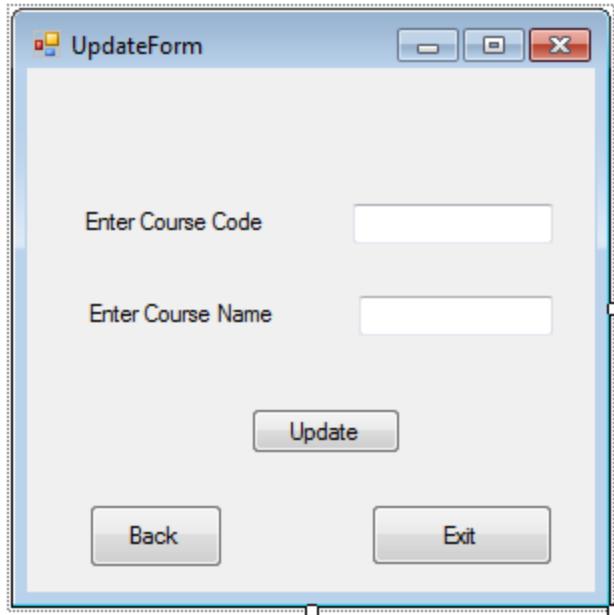
```
InsertForm I = new InsertForm();  
this.Hide();  
I.Show();
```

- e. To test your work working well or not you should click on the green small arrow on the toolbar like this image



11. To do the Update Function you should add new form to the project by follow these steps:

- a. Right click on the project name in the right side and select Add→New Item
- b. Choose windows form and change its name to the form that you want to design which is here “UpdateForm”
- c. Select a table from your project tables that you will apply the Update function on it and design the form with number of labels and text boxes equals to the number of the attributes included in that table that you want to update in it. (Example here will be on a table called Course and include only two attributes and we will update the course name using the course code). You will need to add a Back button and Exit button in each form to be able to navigate through forms without closing the run each time.



- d. Double Click on the Update button and write the following code by these same steps:
- Write the using statement and the connection string code like that in the insert form.
 - In the function of the Update button write the following code by changing the Command with yours (According to the table you choose to update data in it)

```

if (textBox1.Text.Trim() == "" || textBox2.Text.Trim() == "")
{
    errorProvider1.SetError(textBox1, " Please Enter Valid Course Code ");
    errorProvider1.BlinkStyle = ErrorBlinkStyle.AlwaysBlink;
}
else
{
    con.Open();
    errorProvider1.Clear();
    cmd.Connection = con;
    SqlCommand myCommand = new SqlCommand("update Course set Name = '" +
        textBox2.Text.ToString() + "' Where CrsCode ='" + textBox1.Text.ToString() +
        "'", con);
    int success = myCommand.ExecuteNonQuery();
    MessageBox.Show(success + " row has been updated");
    con.Close();
}

```

- Back to the Design of the form and Double Click on the Back button and write the following code (Form 1 here is my master form)

```

Form1 I = new Form1();
this.Hide();
I.Show();

```

- Back to the Design of the form and Double Click on the Exit button and write the following code

```
Application.Exit();
```

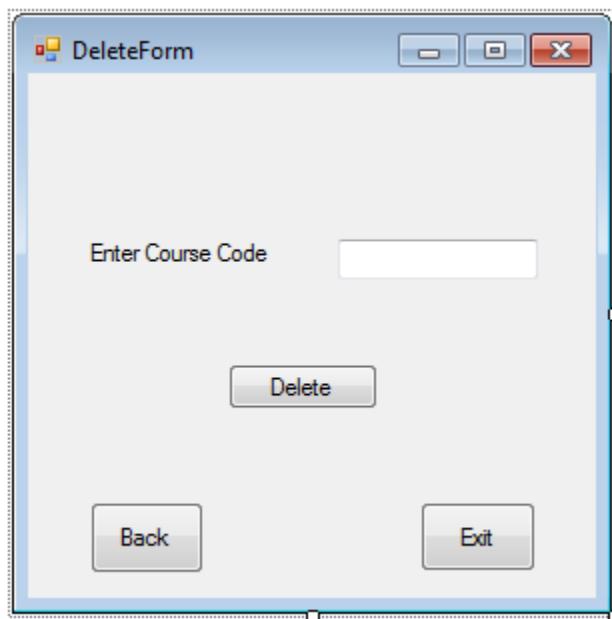
- iii. To execute the Update form we should go back to the master form (Form 1 here) and double click on the update button and call the update form. So when this button clicked the update form will appear and then we can use it.

```
UpdateForm I = new UpdateForm();
this.Hide();
I.Show();
```

- iv. To test your work working well or not you should click on the green small arrow on the toolbar

12. To do the Delete Function you should add new form to the project by follow these steps:

- a. Right click on the project name in the right side and select Add→New Item
- b. Choose windows form and change its name to the form that you want to design which is here “DeleteForm”
- c. Select a table from your project tables that you will apply the Delete function on it and design the form with a label and text box for the primary key of this table. (Example here will be on a table called Course and include the course code attribute which is the primary key). You will need to add a Back button and Exit button in each form to be able to navigate through forms without closing the run each time.



- d. Double Click on the Delete button and write the following code by these same steps:
 - i. Write the using statement and the connection string code like that in the insert form.
 - ii. In the function of the Delete button write the following code by changing the Command with yours (According to the table you choose to Delete data from it)

```
con.Open();
errorProvider1.Clear();
cmd.Connection = con;
SqlCommand myCommand = new SqlCommand("delete from Course where CrsCode=
'" + textBox1.Text + "'", con);
```

```

int success = myCommand.ExecuteNonQuery();
MessageBox.Show(success + " row has been Deleted");
con.Close();

```

- Back to the Design of the form and Double Click on the Back button and write the following code (Form 1 here is my master form)

```

Form1 I = new Form1();
this.Hide();
I.Show();

```

- Back to the Design of the form and Double Click on the Exit button and write the following code

```

Application.Exit();

```

- To execute the Delete form we should go back to the master form (Form 1 here) and double click on the delete button and call the delete form. So when this button clicked the delete form will appear and then we can use it.

```

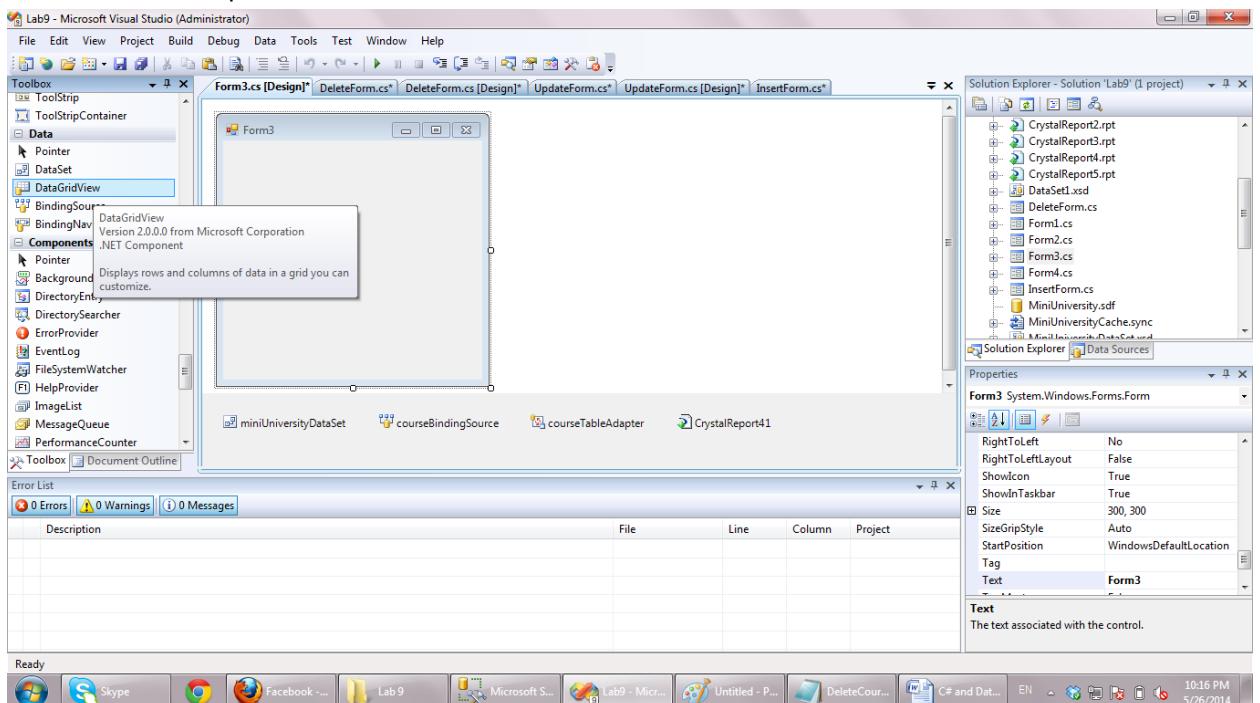
DeleteForm I = new DeleteForm();
this.Hide();
I.Show();

```

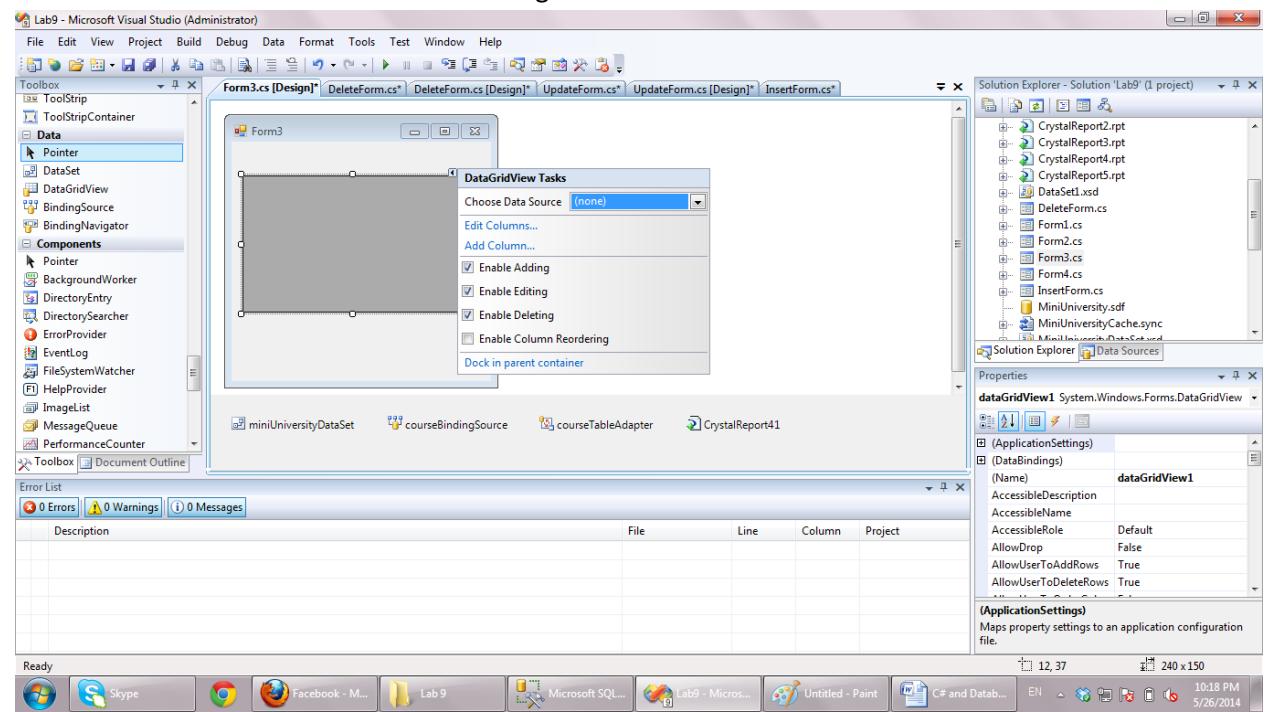
- To test your work working well or not you should click on the green small arrow on the toolbar

- To do the Simple Select Function you should add new form to the project by follow these steps:

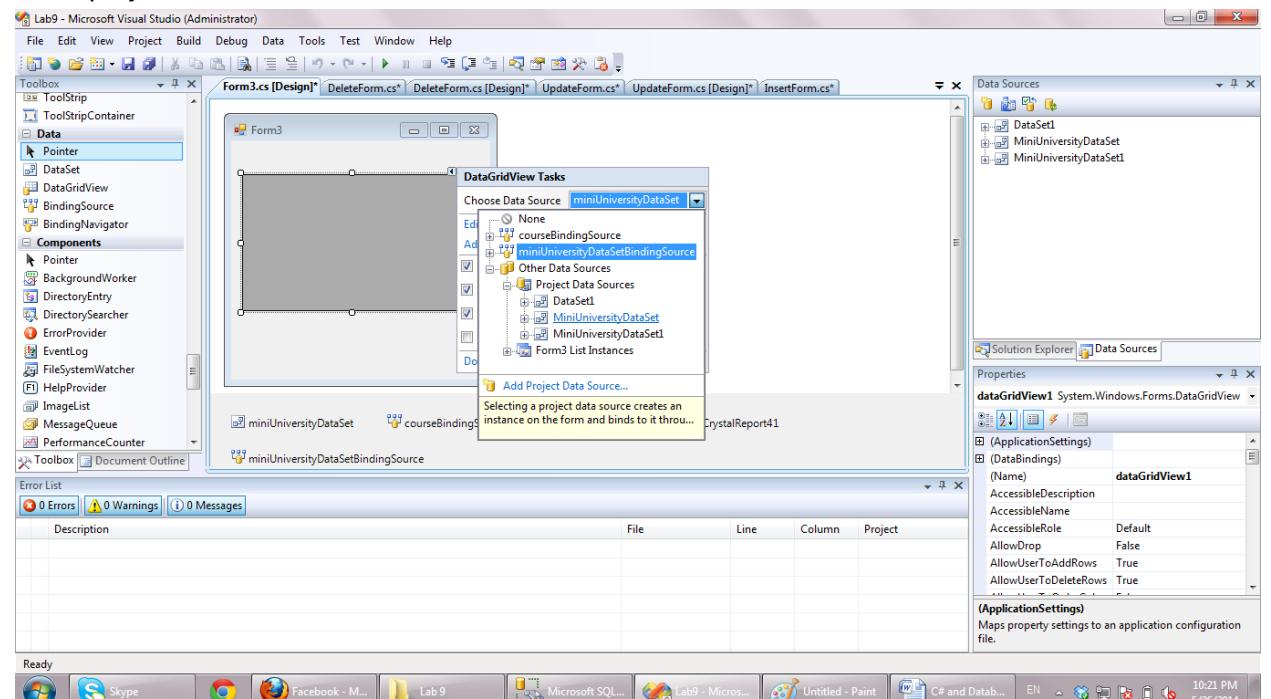
- Right click on the project name in the right side and select Add→New Item
- Choose windows form and change its name to the form that you want to design which is here “SimpleSelectForm”
- From the left side from the Data part in the toolbox select the “DataGridView” and put it on the form and then put the two buttons of Back and Exit on the form also.



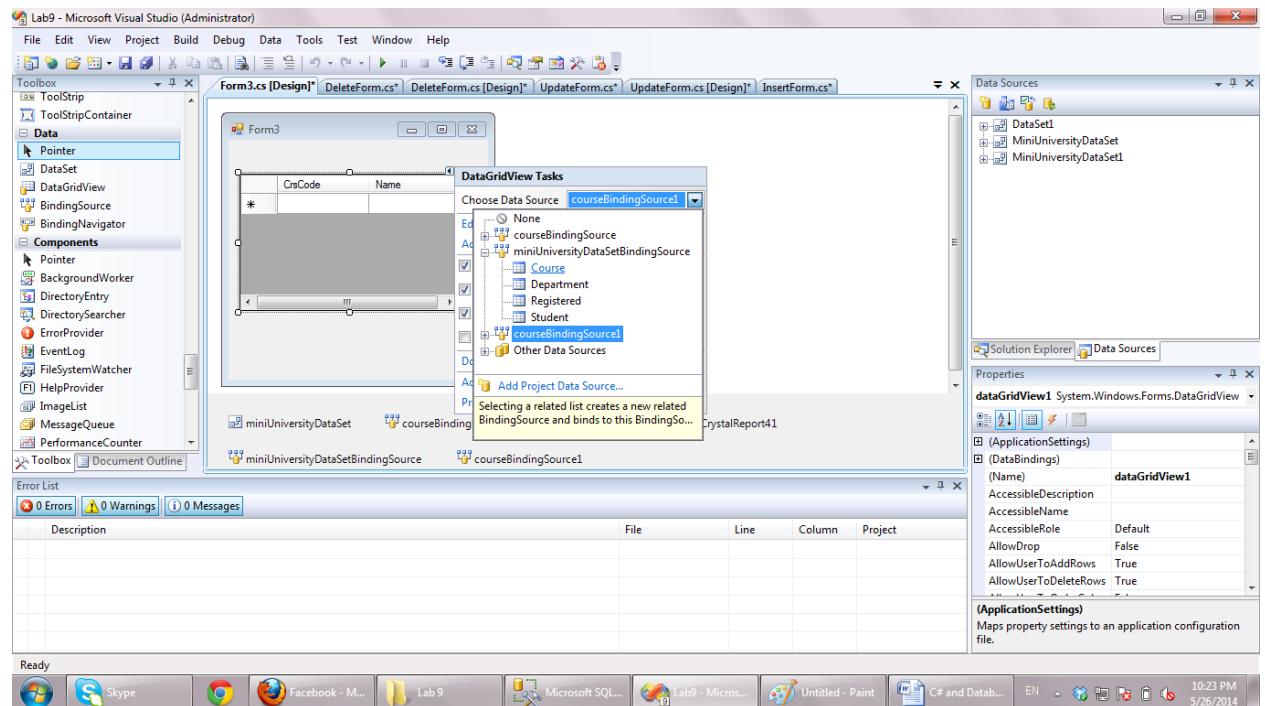
- d. Then from the small black arrow beside the grid view and choose a data source



- e. Select from the ProjectDataSources your data source that we already made in the beginning of the project



- f. Open the + sign beside it and choose the table that you want to select all of its attributes and its data in that grid view (our example here will be the Course table)



- g. Back to the Design of the form and Double Click on the Back button and write the following code (Form 1 here is my master form)

```
Form1 I = new Form1();
this.Hide();
I.Show();
```

- h. Back to the Design of the form and Double Click on the Exit button and write the following code

```
Application.Exit();
```

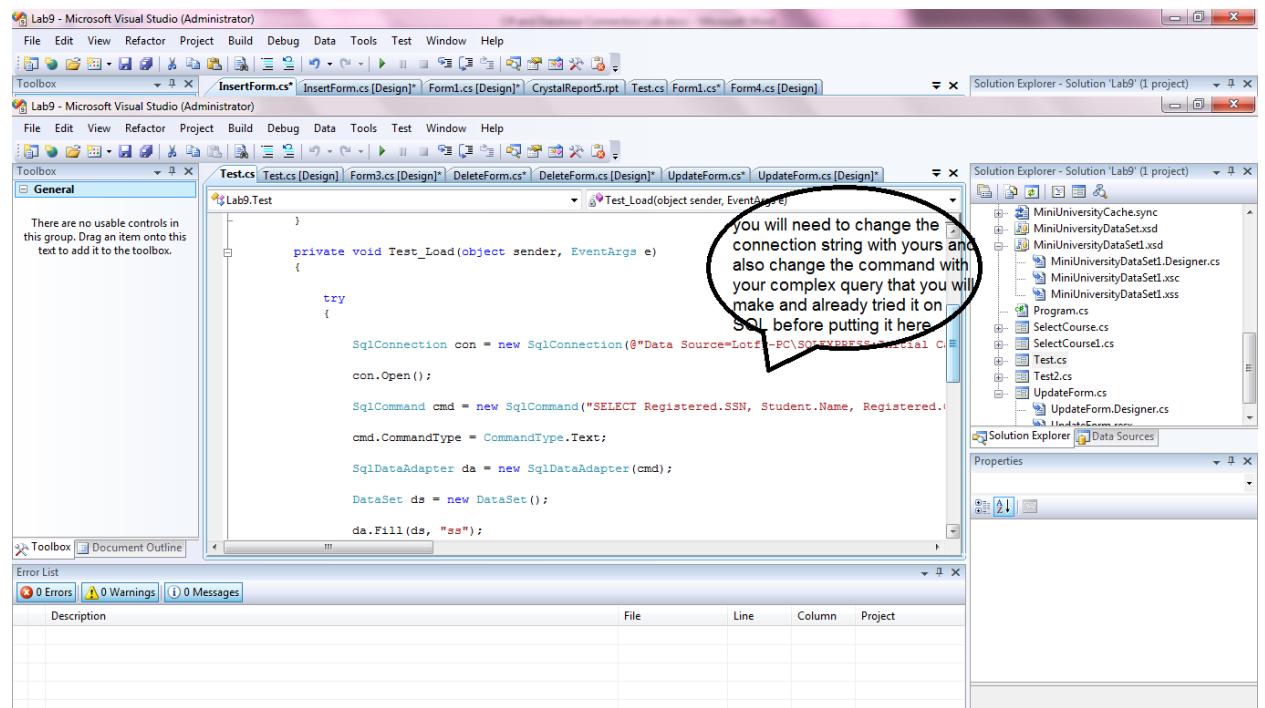
- i. To execute the Simple Select form we should go back to the master form (Form 1 here) and double click on the Simple Select button and call the Simple Select form. So when this button clicked the Simple Select form will appear and then we can use it.

```
SimpleSelectForm I = new SimpleSelectForm();
this.Hide();
I.Show();
```

- j. To test your work working well or not you should click on the green small arrow on the toolbar

14. To do the Complex Select Function you should add new form to the project by follow these steps:

- Right click on the project name in the right side and select Add→New Item
- Choose windows form and change its name to the form that you want to design which is here “ComplexSelectForm”
- From the left side from the Data part in the toolbox select the “DataGridView” and put it on the form and then add the two buttons of Back and Exit on the form also.
- Right click on the form name in the right side of the project window and choose “View Code” and write the following code in the function called “ComplexSelectForm_load” as the following image



And here is the code in the image

```
try
{
    SqlConnection con = new SqlConnection(@"Data Source=Lotfy-PC\SQLEXPRESS;Initial Catalog=MiniUniversity;Integrated Security=True"); //connection name
    con.Open();
    SqlCommand cmd = new SqlCommand("SELECT Registered.SSN, Student.Name, Registered.CrsCode, Course.Name AS Course, Registered.Semester, Registered.Year, Student.City FROM Course INNER JOIN Registered ON Course.CrsCode = Registered.CrsCode INNER JOIN Student ON Registered.SSN = Student.SSN", con);
    cmd.CommandType = CommandType.Text;
    SqlDataAdapter da = new SqlDataAdapter(cmd);
    DataSet ds = new DataSet();
    da.Fill(ds, "ss");
    dataGridView1.DataSource = ds.Tables["ss"];
}
catch
{
    MessageBox.Show("No Record Found");
}
```

- e. Back to the Design of the form and Double Click on the Back button and write the following code (Form 1 here is my master form)

```
Form1 I = new Form1();
this.Hide();
I.Show();
```

- f. Back to the Design of the form and Double Click on the Exit button and write the following code

```
Application.Exit();
```

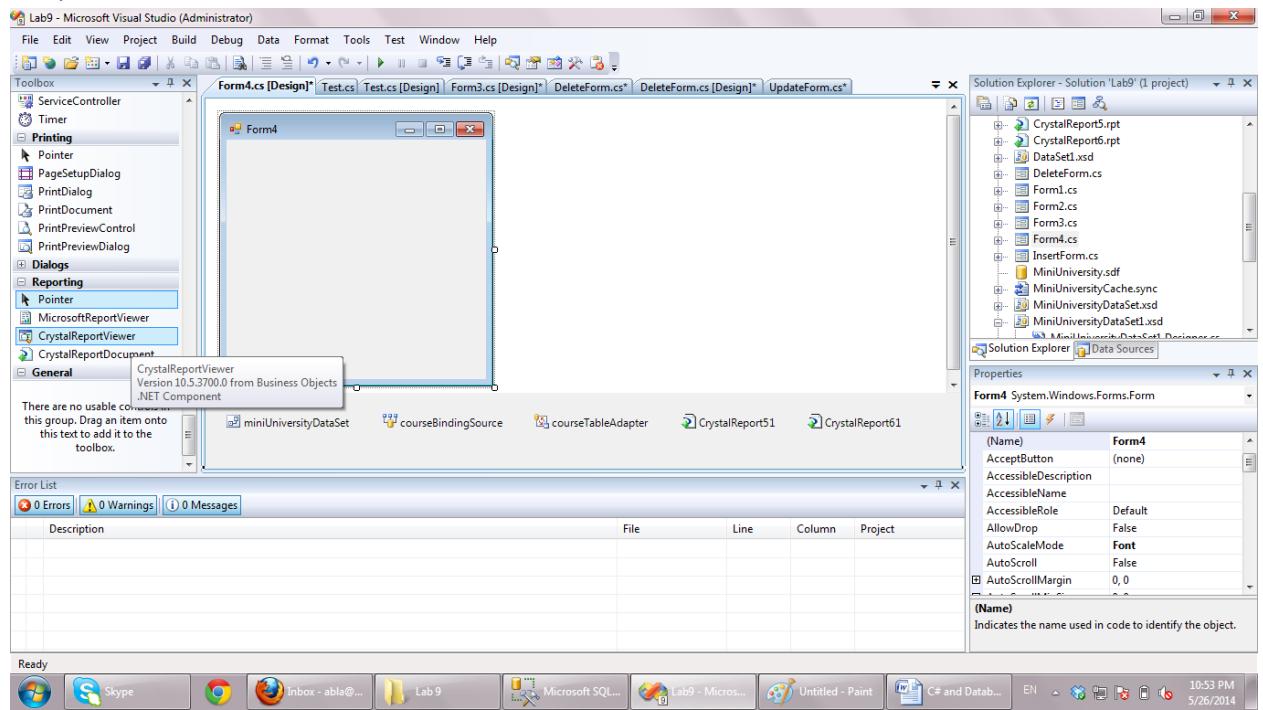
- g. To execute the Complex Select form we should go back to the master form (Form 1 here) and double click on the Complex Select button and call the Complex Select form. So when this button clicked the Complex Select form will appear and then we can use it.

```
ComplexSelectForm I = new ComplexSelectForm();
this.Hide();
I.Show();
```

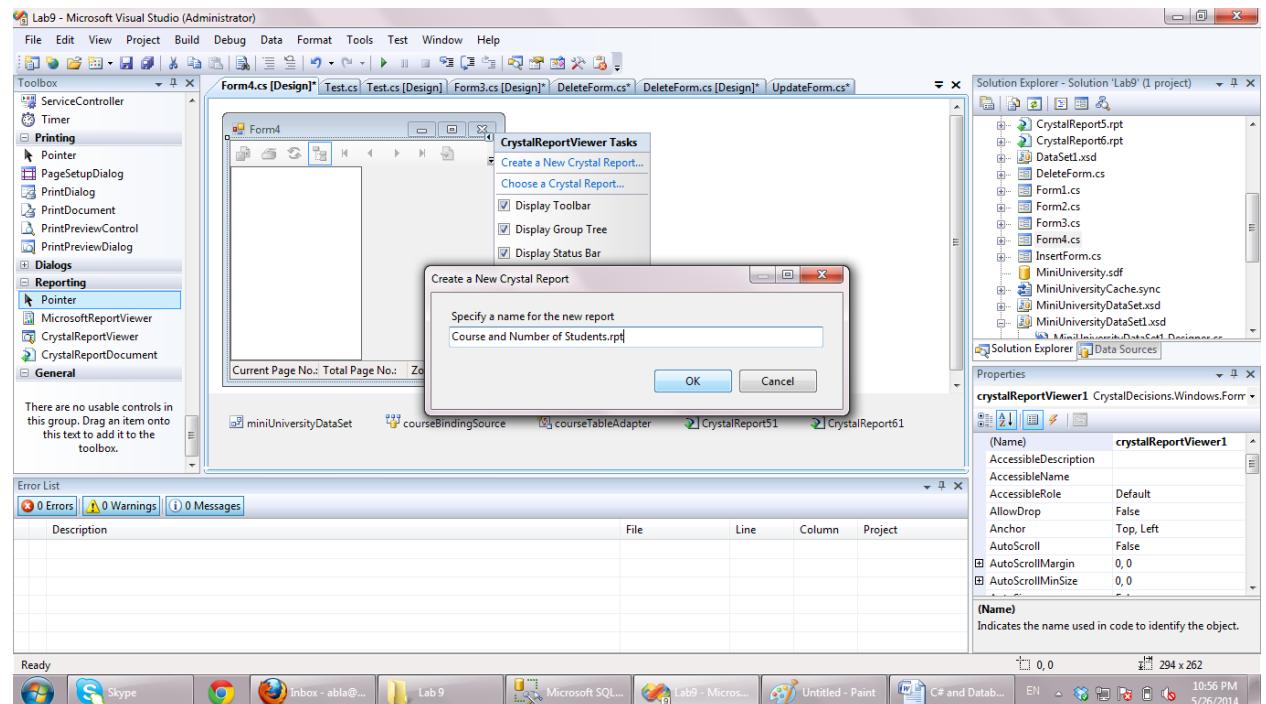
- h. To test your work working well or not you should click on the green small arrow on the toolbar

15. To do the Crystal Report you should first specify what kind of data that you need to do some statistics on it and view it as a report. (Example here will be number of students registered in each course and filtering on the total number of all students registered in all courses)

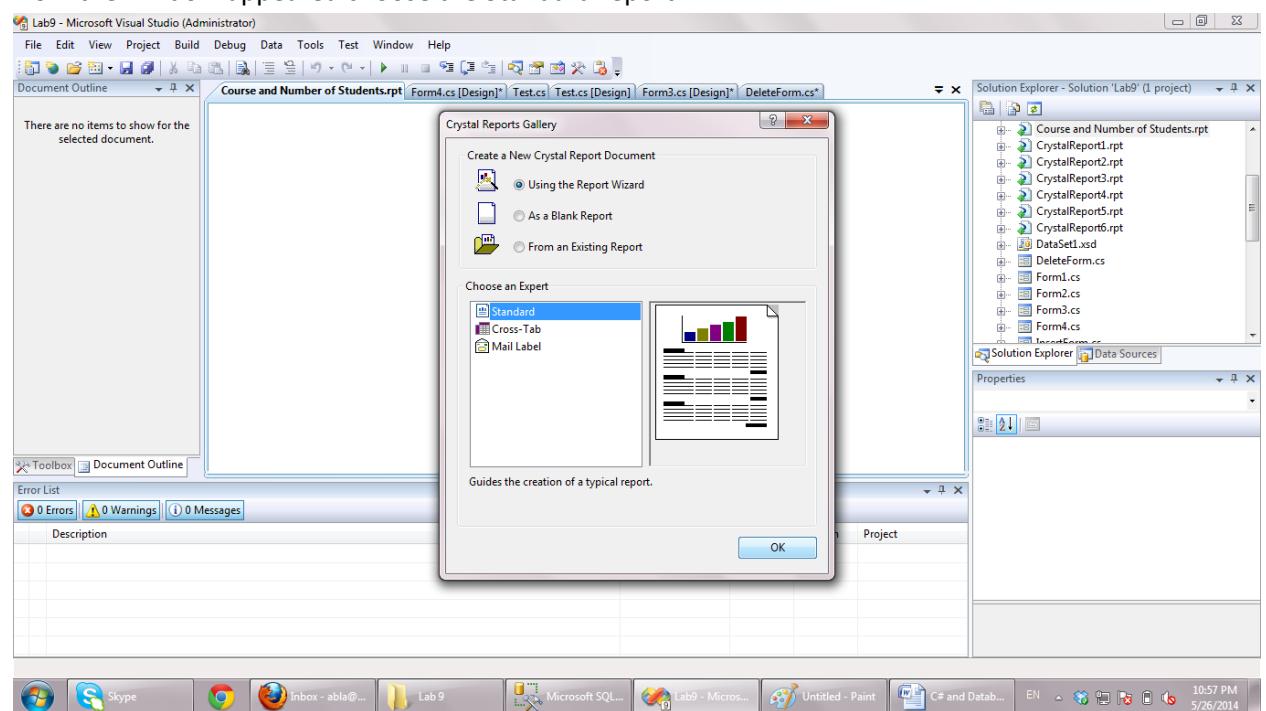
- Right click on the project name in the right side and select Add→New Item
- Choose windows form and change its name to the form that you want to design which is here “CrystalReportForm”
- From the left side from the Reporting part in the toolbox select the “CrystalReportViewer” and put it on the form.



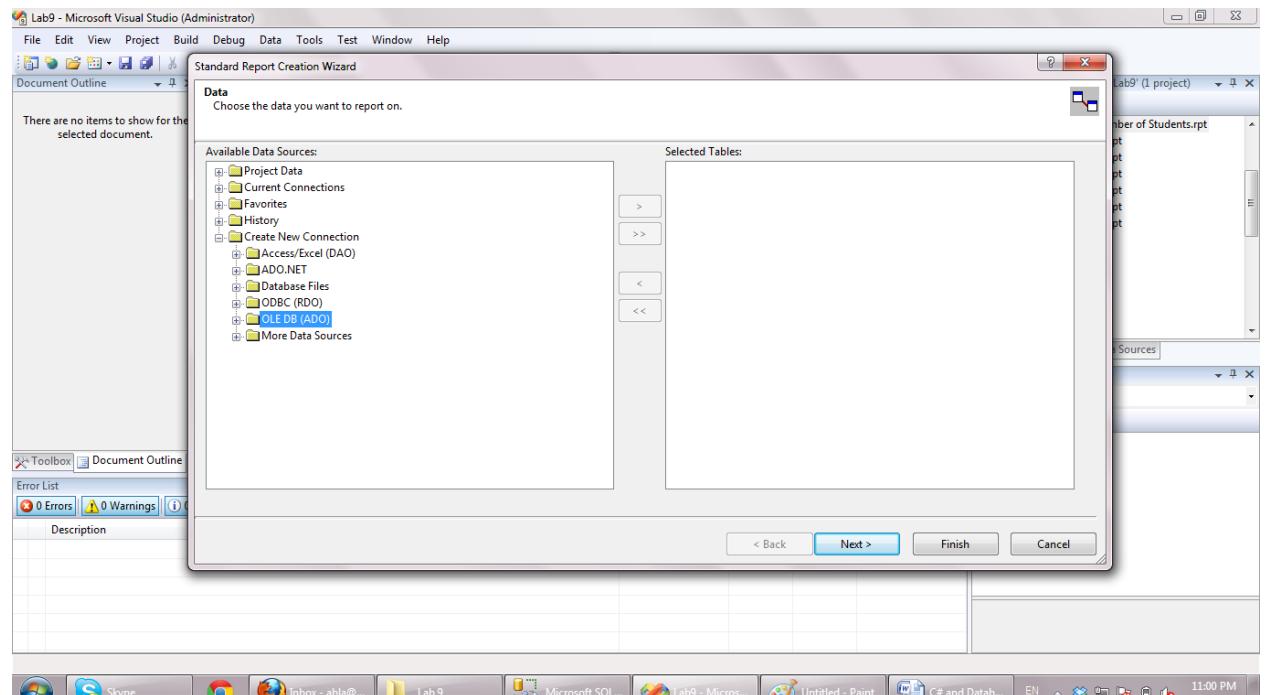
- From the small black arrow in the crystal report viewer choose “Create a new Crystal Report”
- Name the crystal report with a valuable name and then click Ok



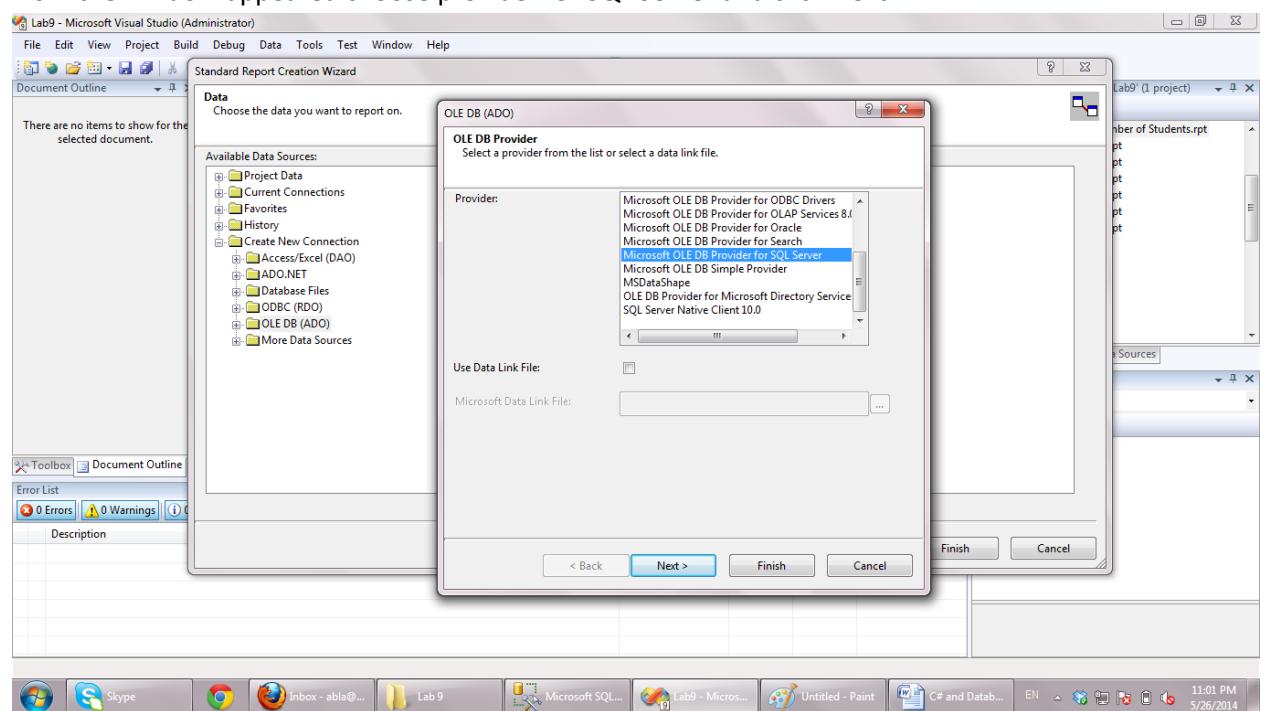
- f. From the window appeared choose the Standard report



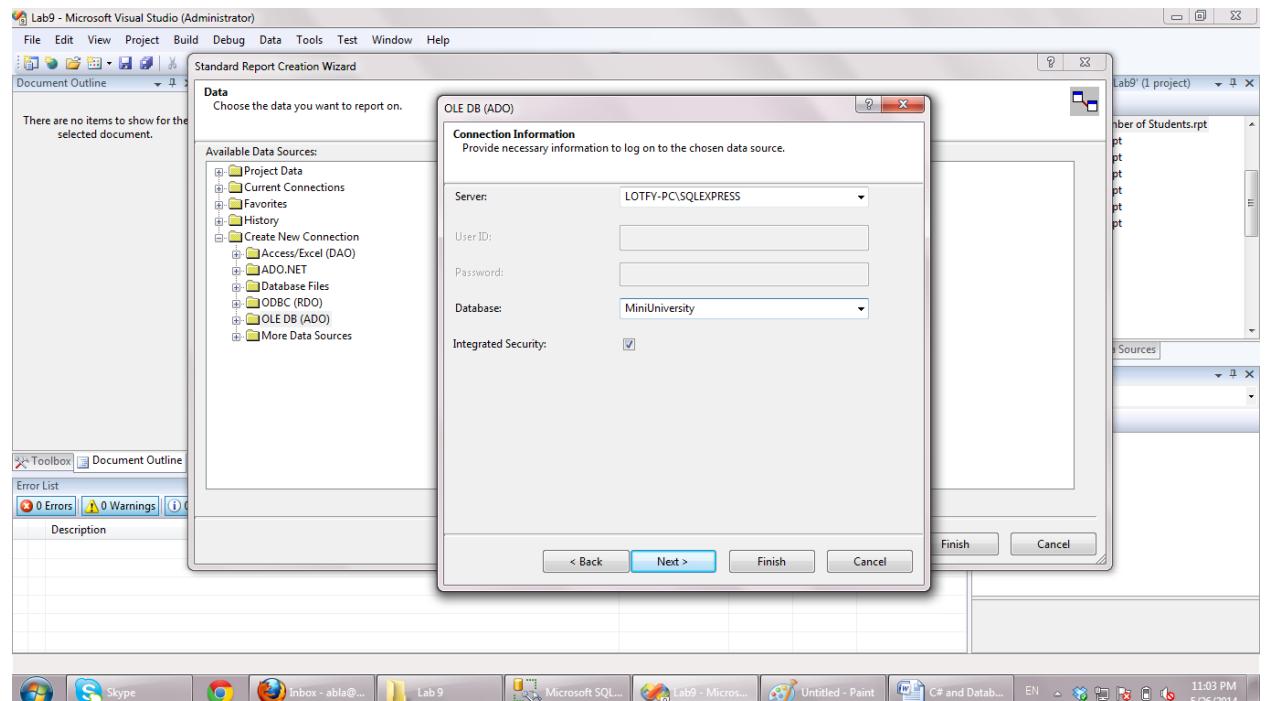
- g. From the window appeared open the + sign beside "Create New Connection" and then open the + sign beside "OLE DB (ADO)"



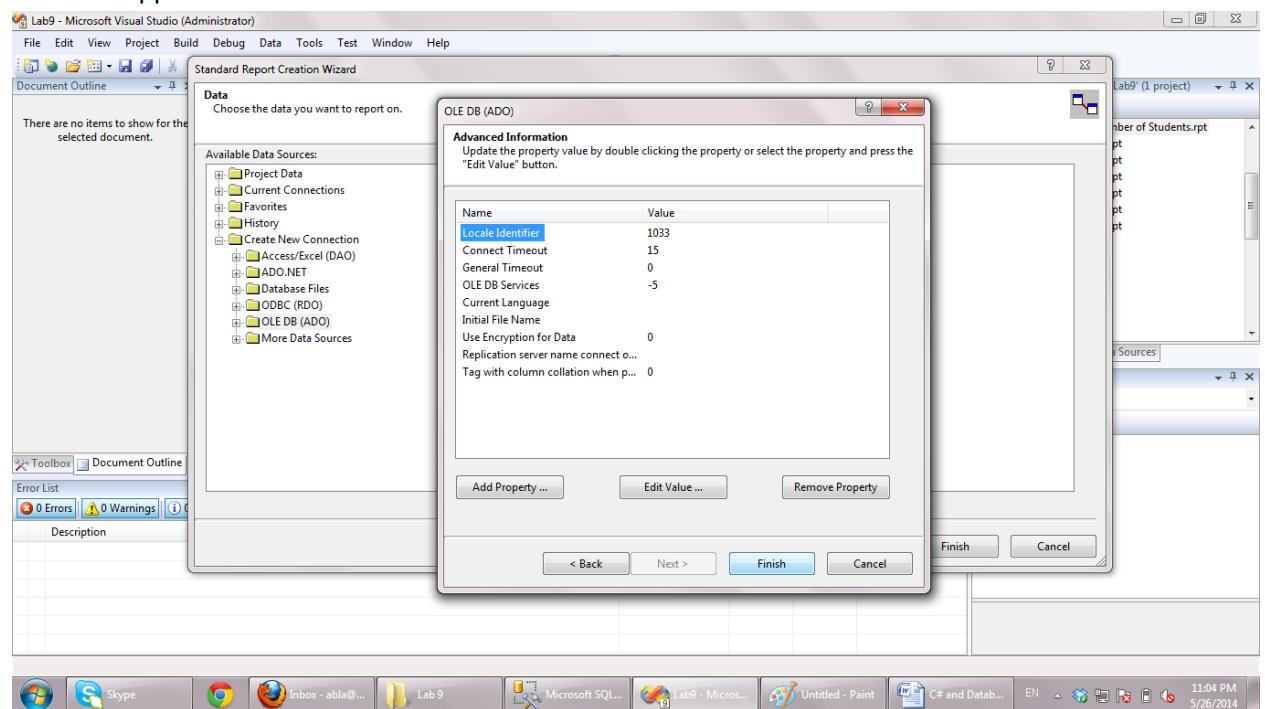
- h. From the window appeared choose provider for SQL Server and click Next



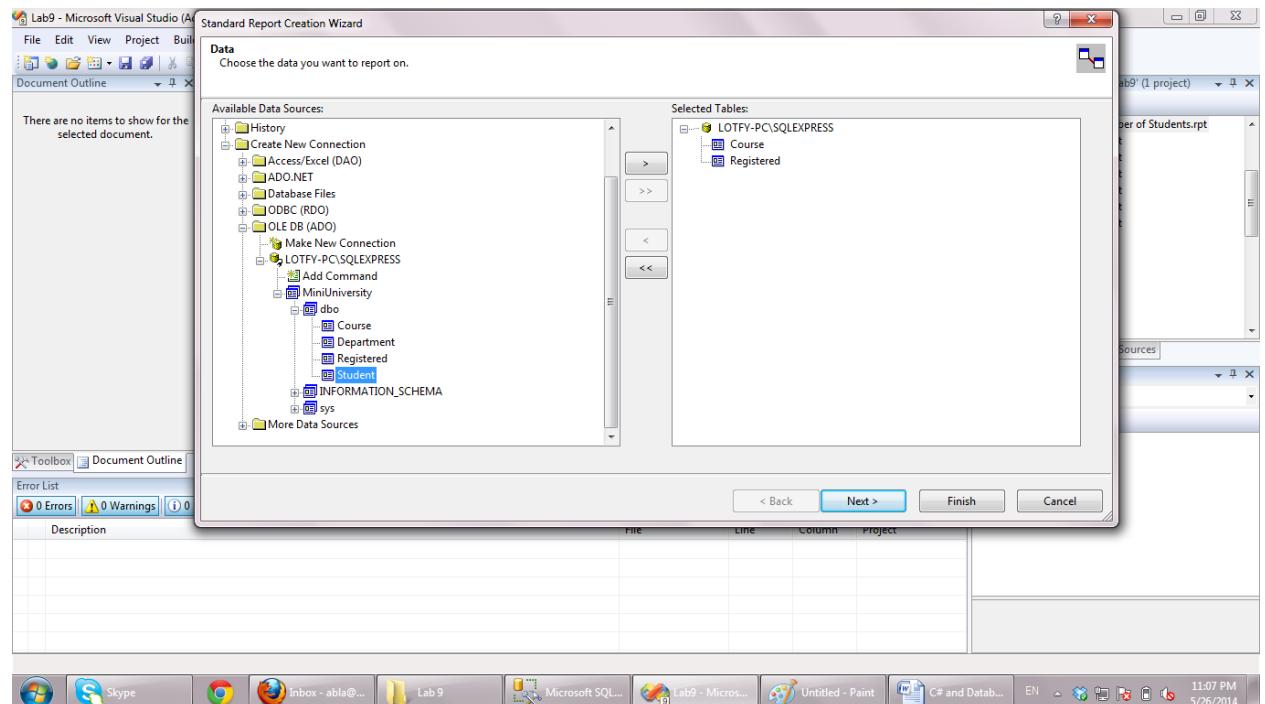
- i. From the appeared window write the server name of the SQL Server and then check the checkbox of “Integrated Security” and then Choose the database that you already work on it in the project and then click Next



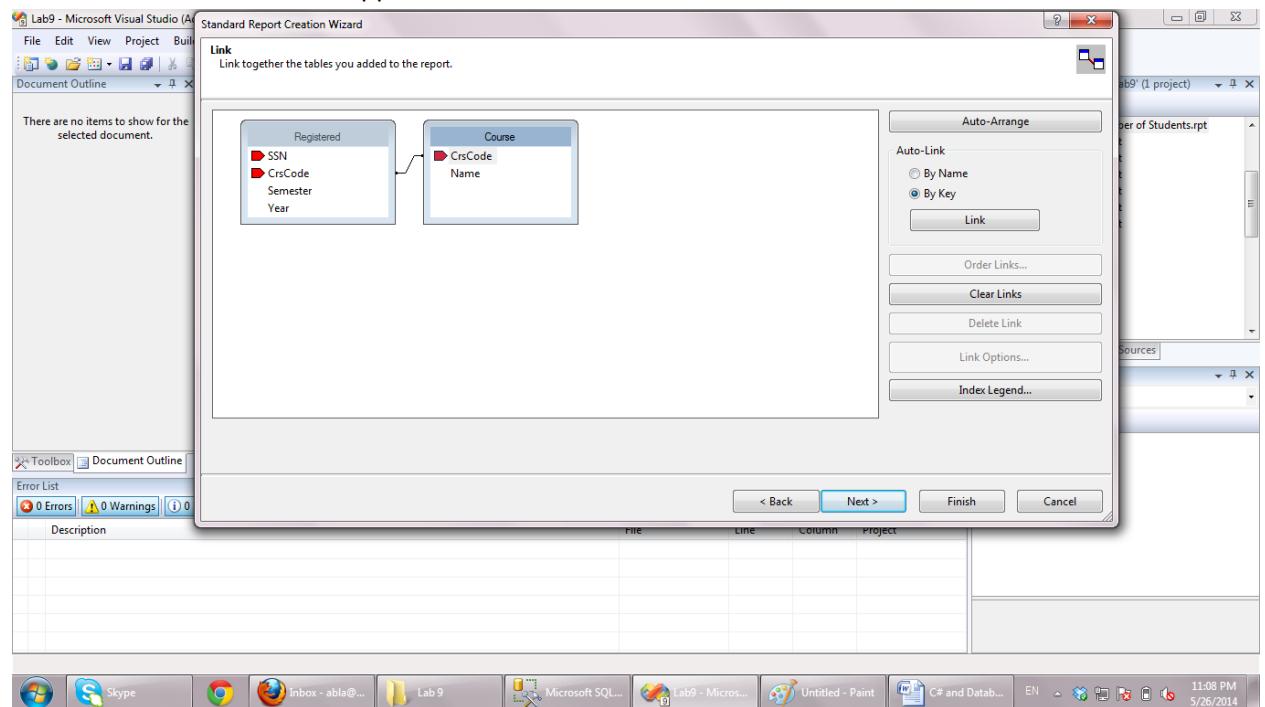
- j. From the appeared window click Finish



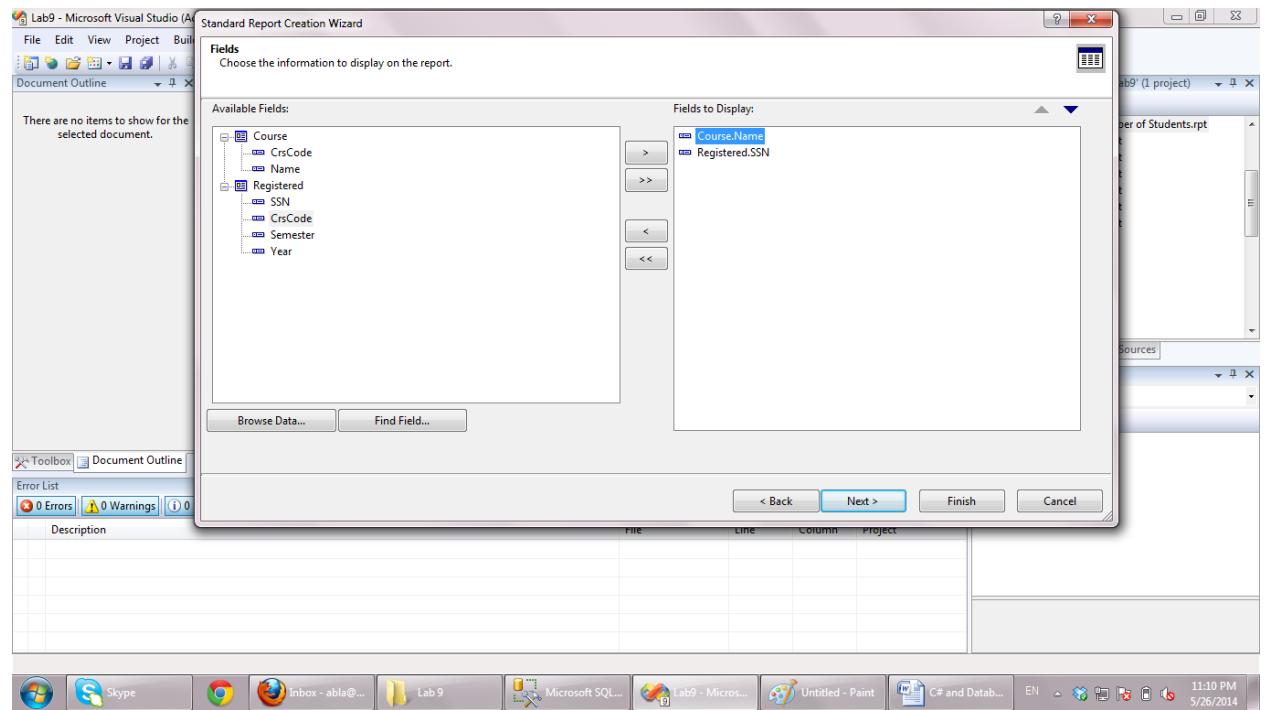
- k. Open the + sign beside the new connection and then open the + sign beside the dbo and then choose the tables that will be included in your report that you will extract data from them and then press the button to bring them from the left side to be included in the right side under "Selected Tables". (Example here will be tables Course, Registered) and then click Next



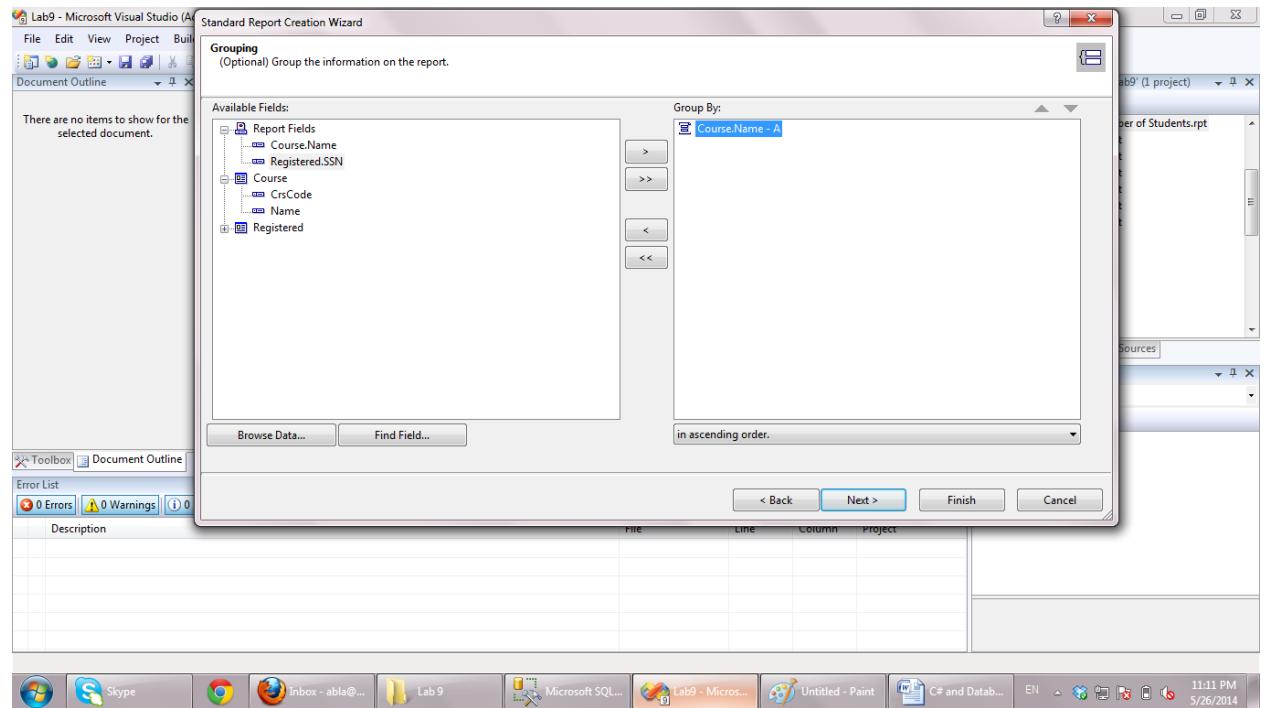
- i. The link between tables will appear like that



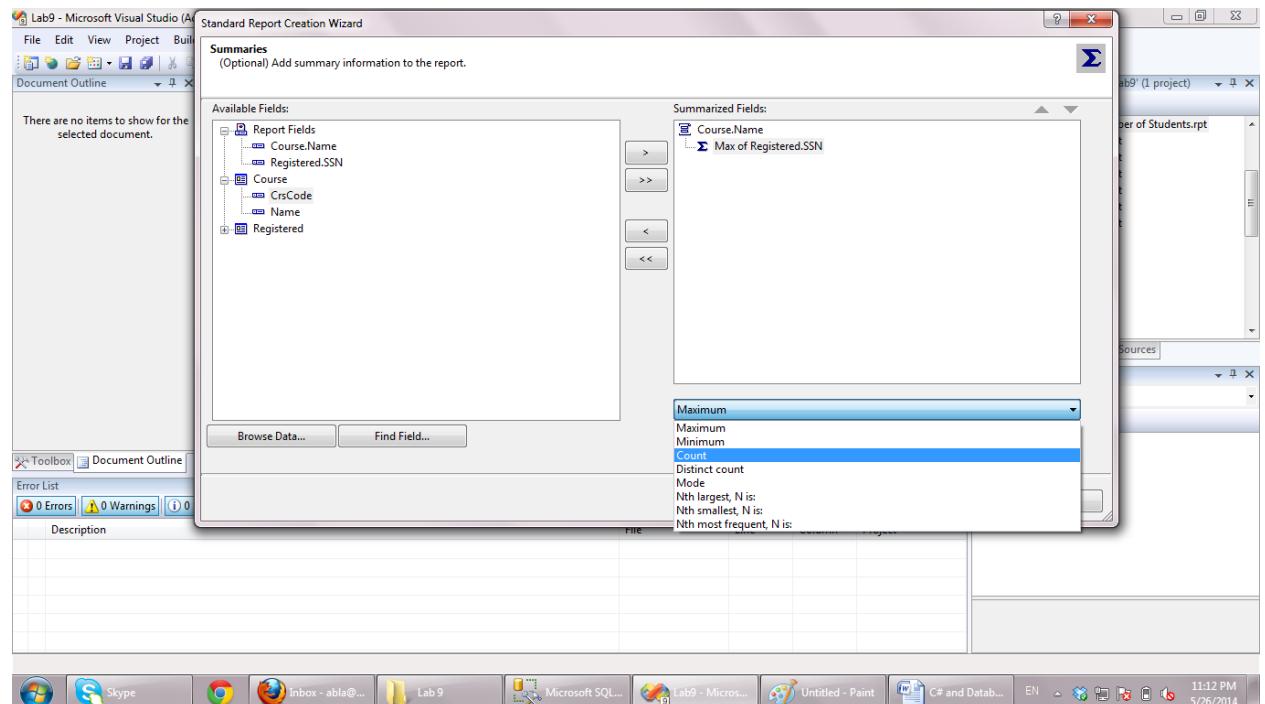
- m. Then we will select the fields to be displayed in the report to be included from the left side to the right side (Here will be the Course Name and SSN) and then click Next



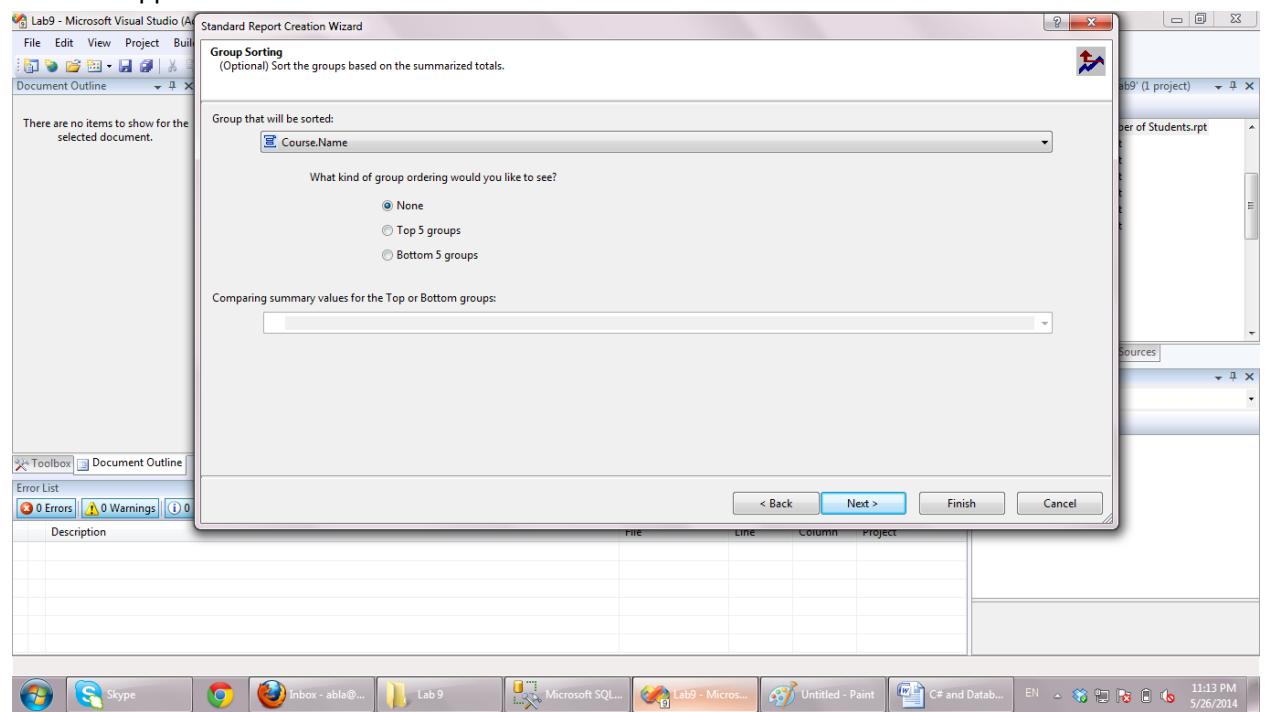
- n. Choose the field or fields that you need to do Group by on as we will use an aggregate function (we should do group by when using aggregation as you study in SQL) and then click Next



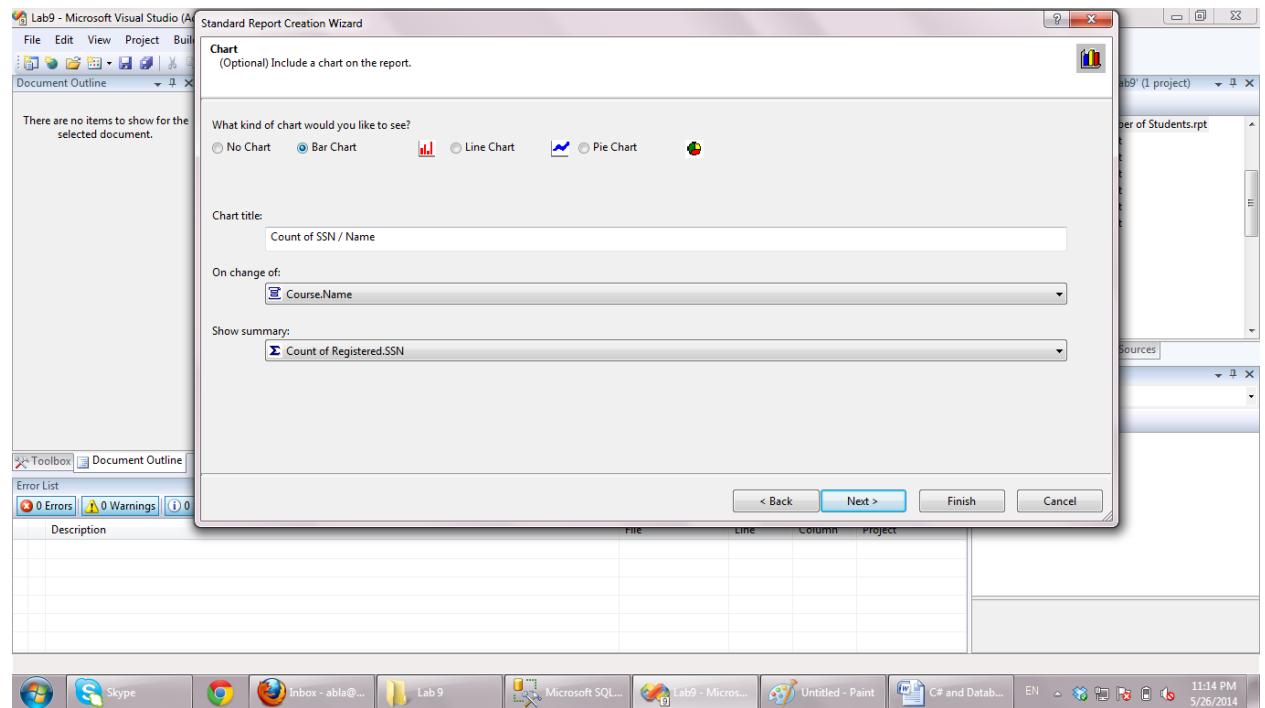
- o. Select the Summarized Fields which are the fields or attributes that you will apply aggregation on it (Here will be the SSN as we will apply the Count function on it) and then click Next



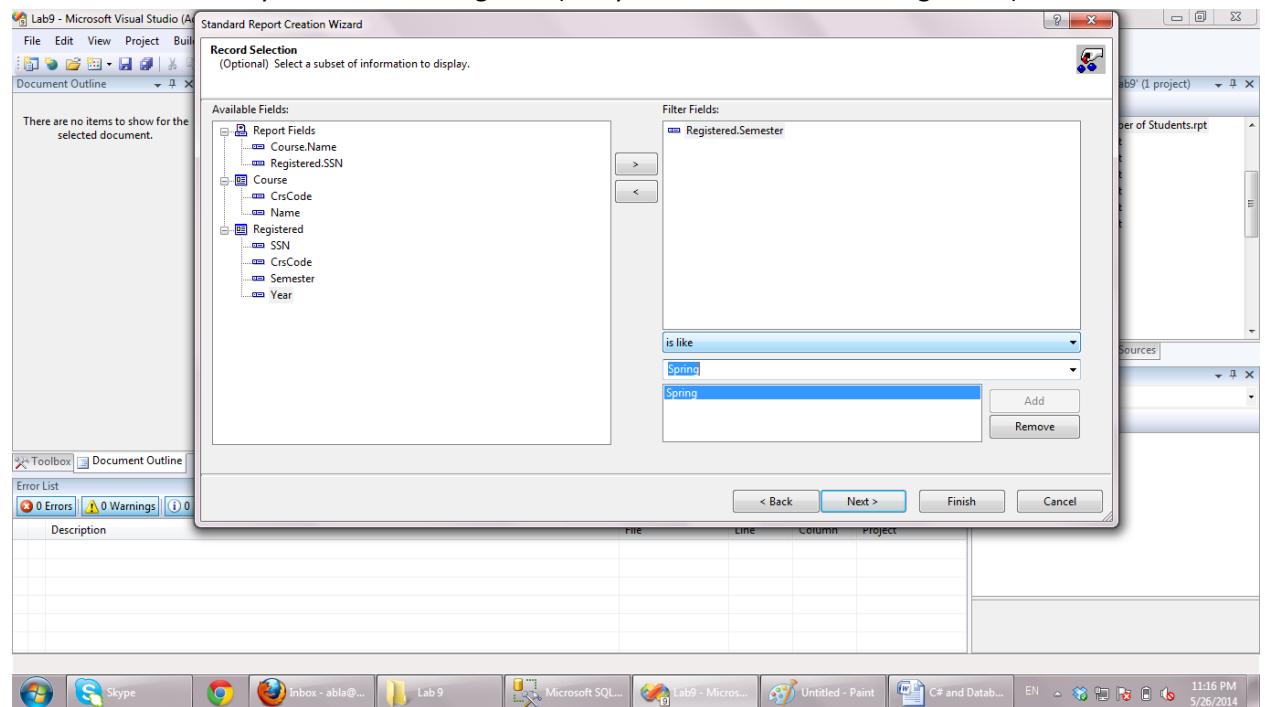
p. From the appeared window click Next



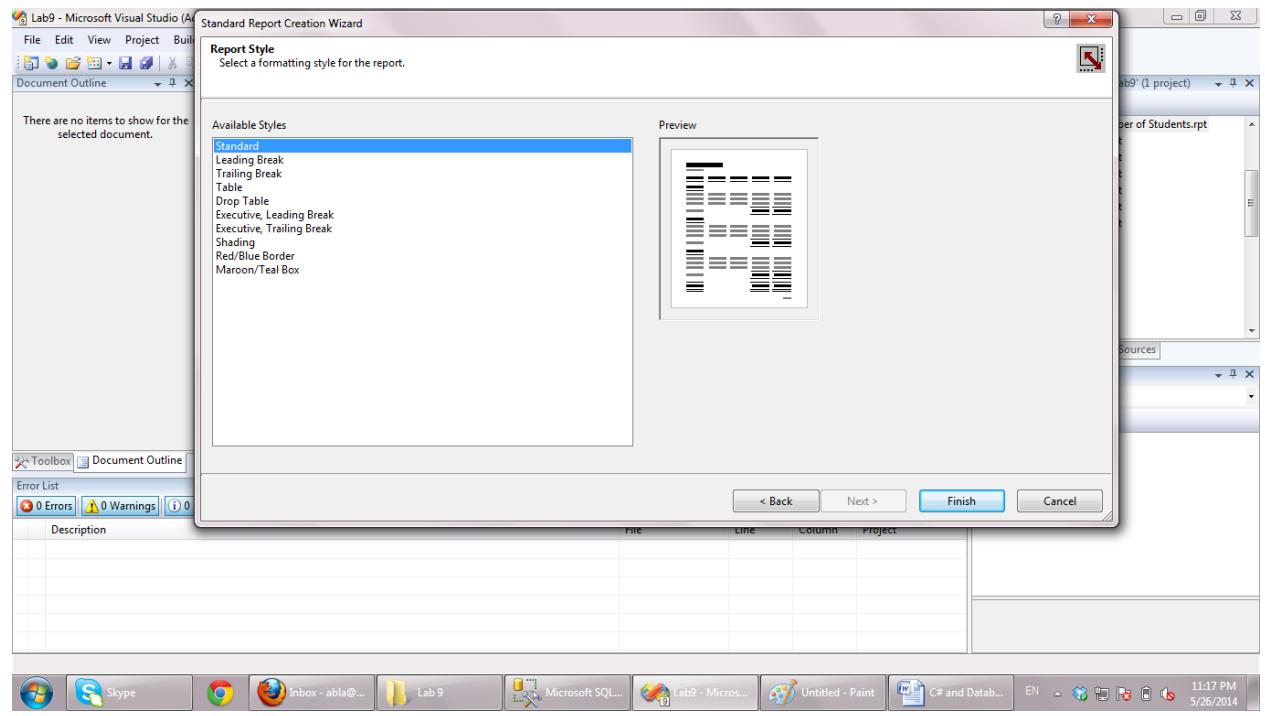
q. Select the type of chart that you will use in your report (it is your choice to show your data in a chart or not) and click Next



- r. Select the fields that you will do filtering on it (It is your choice to use filtering or not)



- s. Select the report style that you want and then click Finish



- t. To execute the Crystal Report form we should go back to the master form (Form 1 here) and double click on the Crystal report button and call the Crystal Report form. So when this button clicked the Crystal Report form will appear and then we can use it.

```
CrystalReportForm I = new CrystalReportForm();
this.Hide();
I.Show();
```

- u. To test your work working well or not you should click on the green small arrow on the toolbar

The End