## Week 8 – ADO level 1

In the lecture we covered ADO level 1, a very simple way to embed SQL within our allocation so that we can use an attached database. In order to do this we will create a database within visual studio as it has ‘lite’ version of SQL server for testing/development purposes.

I could give you a project which already contained the Database but let’s be honest where would the fun in that be! So we will create the database and run SQL script to create and populate the database. This is based on old DB we used with the first year students to teach them oracle SQL. There are deliberately some problems with the way the database has been constructed. These won’t affect what we are doing today.

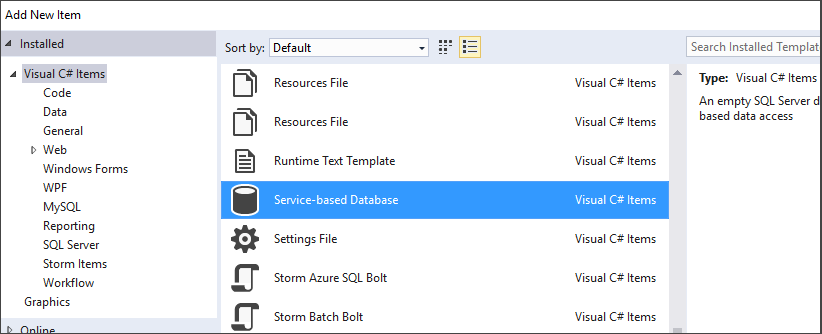
The simple application we are going to build will list the students on a particular program. A student can be selected which will open another form showing their mark if no mark is available then the user is informed that the student has no marks

Step one

Create a new windows form project called Week9StudentDBForm

In the project explorer select the project right click and select ADD - > **Add New Item**

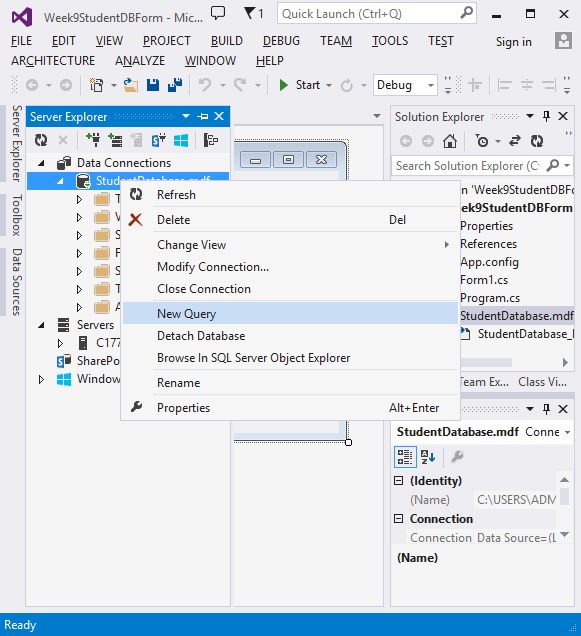
In the list of possible template items, find and select Server-Based database.



Name the database **StudentDatabase**, and then select the **Add** button.

In the solutions explorer, right click on the database and select open.

You should now see the database in the **Server Explorer**.



Right click on the database and select new query.

This will open a new area in which you can write SQL.

Copy and paste all the statements from SQLforStudentDatabase.txt into this area and then execute the script. Little green triangle on the top the sql edit screen.

It should create a database with the following structure

class

Subject

Staffmember

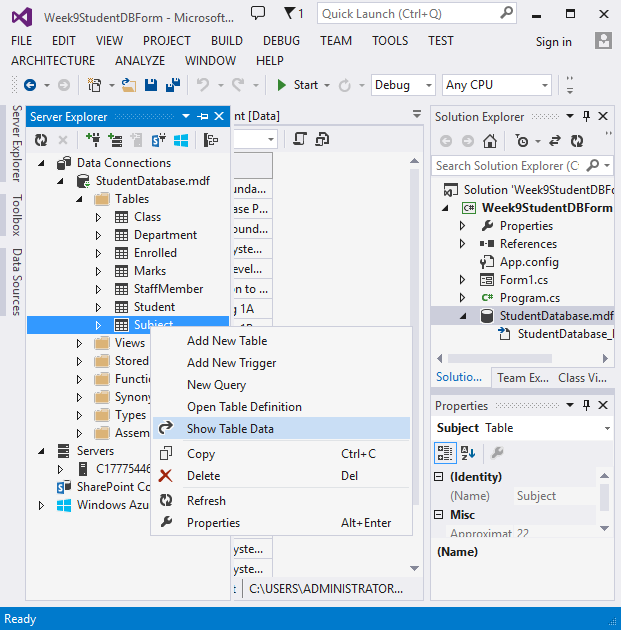
Department

Marks

Enrolled

Student

The script will run, with errors. You can ignore these errors but we need to check the tables contain the information.



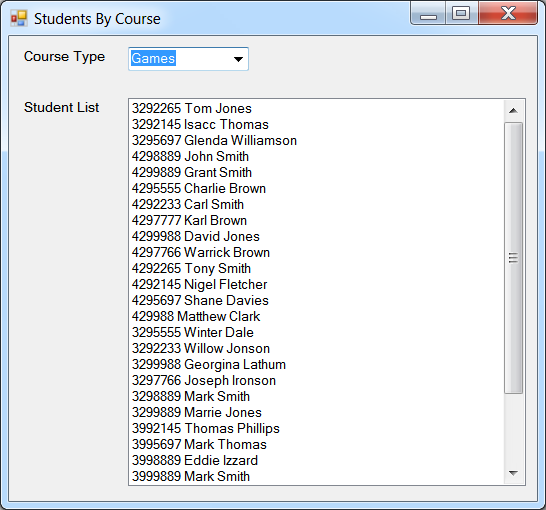
You should now be able to right click on the database in the server explorer and select properties, this will give you the connection string to the DB.

You now need to set up a connection string for your project. In the solutions explorer right click on the project and select properties.

You now need to select settings and on the type select connection string.

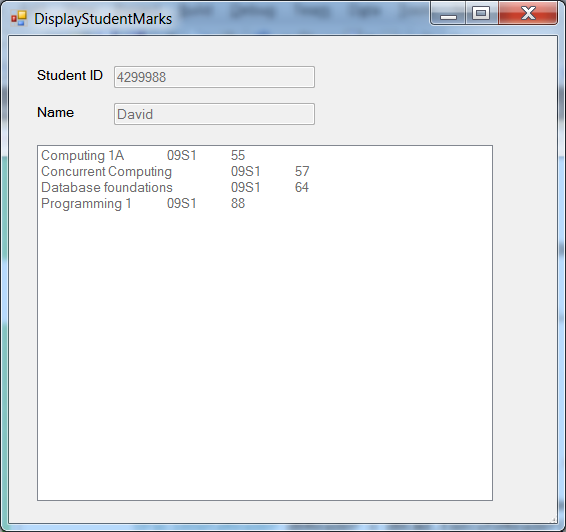
Paste the connection string you got from your database property in to the value section. CHANGE THE NAME OF THIS PROPERTY TO CONNECTION

What the application will look like



Selecting a course will display different set of students.

Selecting a student will case the application to open a new form.



You will need the following include statements

using System.Configuration;

using System.Data.sqlClient;

In your code you will need to establish a connection.

new SqlConnection(Properties.Settings.Default.connection);

NOTE THE NAME CONNECTION MUST MATCH WHAT YOU HAD IN THE PROPERTIES

OK you are MSc student so should be able to complete the rest without too much help so here are just a couple of pointers

The SQL code you will need is as follows.

TO LIST THE COURSES:

Select distinct major from student

TO SELECT THE STUDENTID and NAMES for a particular programe:

select studentid, stuname from student where major = '" + majorString + "'"

TO SELECT A THE SUBJECT MARKSPERIOD AND MARK for a particular student:

select sname, period, mark from marks join subject using (SUBJECTID) where studentid =

This is the solution for the first form only, please refers to this only if you cannot see how to do the solution yourself. I do not plan to make the second form solution available. You need to complete this yourself.

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Configuration;

using System.Data.SqlClient;

namespace Week9StudentDBForm

{

public partial class Form1 : Form

{

SqlConnection con;

public Form1()

{

con = new SqlConnection(Properties.Settings.Default.connection);

InitializeComponent();

}

private void Form1\_Load(object sender, EventArgs e)

{

con.Open();

string sqlSelectCouses = "Select distinct major from student";

SqlCommand dbCmd = new SqlCommand(sqlSelectCouses, con);

SqlDataReader dbReader = dbCmd.ExecuteReader();

while (dbReader.Read())

{

string row = dbReader.GetString(0);

cmb\_courseList.Items.Add(row);

}

dbCmd.Dispose();

dbReader.Close();

}

private void cmb\_courseList\_SelectedIndexChanged(object sender, EventArgs e)

{

string selected = (string)cmb\_courseList.SelectedItem;

string sqlStudentList = "select studentid, stuname from student where major = '" + selected.Trim() + "'";

lst\_studentList.Items.Clear();

SqlCommand dbCmd = new SqlCommand(sqlStudentList, con);

SqlDataReader dbReader = dbCmd.ExecuteReader();

while (dbReader.Read())

{

int studentID = dbReader.GetInt32(0);

string studentName = dbReader.GetString(1);

lst\_studentList.Items.Add(studentID.ToString() + " " + studentName);

}

dbCmd.Dispose();

dbReader.Close();

}

private void lst\_studentList\_SelectedIndexChanged(object sender, EventArgs e)

{

string selected = (string)lst\_studentList.SelectedItem;

string[] values = selected.Split();

DisplayStudentMarks studentMarksForm = new DisplayStudentMarks(values[0], values[1]);

this.Hide();

studentMarksForm.ShowDialog();

this.Show();

}

}

}