

SAMPLE CODE: DATABASE ACCESS AND UPDATING

EXAMPLE 1: USING A DATAGRIDVIEW BOUND TO A JOINED TABLE (MOST OF THE WORK WAS DONE IN THE DESIGNER)

```
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();
    }

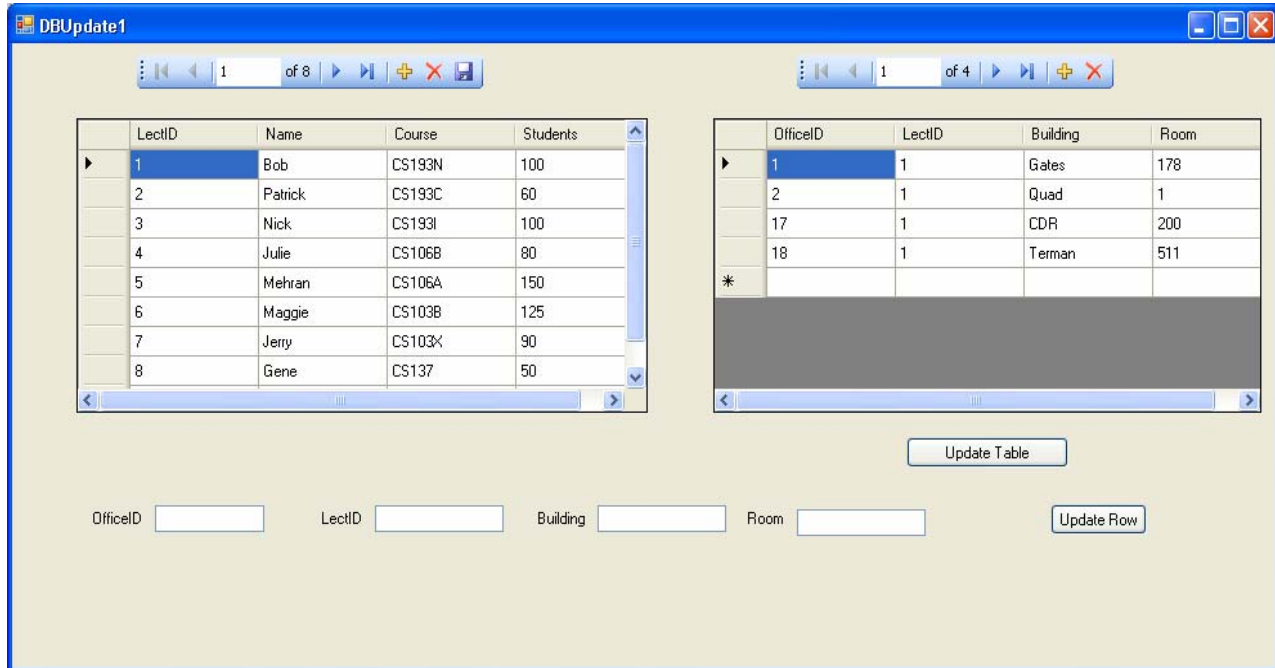
    private void Form1_Load(object sender, EventArgs e)
    {
        this.lecturersTableAdapter1.Fill(testSQL2DataSet.Lecturers);

        // TODO: This line of code loads data into the
        'testSQL2DataSet.LecturersRooms' table. You can move, or remove it, as needed.

        this.lecturersRoomsTableAdapter.Fill(this.testSQL2DataSet.LecturersRooms);
    }

    private void lecturersRoomsBindingSource_CurrentChanged(object sender,
                                                             EventArgs e)
    {
        DataRowView drv = (DataRowView) lecturersRoomsBindingSource.Current;
        TestSQL2DataSet.LecturersRoomsRow row =
            (TestSQL2DataSet.LecturersRoomsRow) drv.Row;
        TestSQL2DataSet.LecturersRow lectRow =
            testSQL2DataSet.Lecturers.FindByLectID(row.LectID);
        textBox1.Text = lectRow.Course;
    }
}
```

EXAMPLE 2: DATABASE UPDATES



```

private void Form1_Load(object sender, EventArgs e)
{
    // TODO: This line of code loads data into the
    'testSQL2DataSet.Offices' table. You can move, or remove it, as needed.
    this.officesTableAdapter.Fill(this.testSQL2DataSet.Offices);
    // TODO: This line of code loads data into the
    'testSQL2DataSet.Lecturers' table. You can move, or remove it, as needed.
    this.lecturersTableAdapter.Fill(this.testSQL2DataSet.Lecturers);
}

//This is the code for the Update Row button, which takes values from
textboxes.
//It looks up the row and uses its values as the original values in the
call to
//NewUpdateQuery, which pushes the new row out to the database. Then we
call Fill
//to reload the table. No need to clear it first here, since
ClearBeforeFill is set.
//NewUpdateQuery actually does the same thing as the Update method, but
it's an example
//of adding an additional query to the command collection.

private void btnUpdate_Click(object sender, EventArgs e)
{
    int rowsAffected;
    int officeID = Int32.Parse(tbOfficeID.Text);
    int lectID = Int32.Parse(tbLectID.Text);
    TestSQL2DataSet.OfficesRow row =
testSQL2DataSet.Offices.FindByOfficeID(officeID);
    if (row == null)
    {
        rowsAffected = 0;
    }
}

```

```
        MessageBox.Show("0 rows affected. Couldn't find row with
OfficeID " + officeID);
    }
    else
    {
        rowsAffected = officesTableAdapter.NewUpdateQuery(lectID,
tbBuilding.Text, tbRoom.Text, officeID, lectID, row.Building, row.Room, officeID);
        if (rowsAffected > 0)
        {
            MessageBox.Show(rowsAffected + " rows affected");
            officesTableAdapter.Fill(this.testSQL2DataSet.Offices);
        }
        else
        {
            MessageBox.Show("0 rows affected. Couldn't find row: " +
row.OfficeID + " " + row.LectID + " " + row.Building + " " + row.Room);
        }
    }
}

//This is the code for the Update Table button.

private void btnUpdateTable_Click(object sender, EventArgs e)
{
    officesTableAdapter.Update(testSQL2DataSet.Offices);
}
}
```