```
using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Data.OleDb;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;

public partial class New_Invoice : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }

    /*~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
        Function name: GridView1_SelectedIndexChanged(object
           sender, EventArgs e)
        Purpose: Displays the selected invoice in GridView1 in
           DetailsView1.
        Author: Stuart Crome
        Created on: June 7, 2011
        Modified by: Stuart Crome
        Modified: June 10, 2011
        Parameters: object sender, EventArgs e
        ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~*/
    protected void GridView1_SelectedIndexChanged(object sender, EventArgs e)
    {
        DetailsView1.PageIndex = GridView1.SelectedIndex;
    }

    /*~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
        Function name: DetailsView1_ItemInserted(object sender,
           DetailsViewInsertedEventArgs e)
        Purpose: Checks whether the new item inserted checks
           correctly against the contract. A data set is created
           and varibles are created to check values of the fee
           max, the hourly rates and the contract end and start
           dates.
        Author: http://msdn.microsoft.com/en-us/library/aa288452
          (v=vs.71).aspx
        Created on: June 10, 2011
        Modified by: Stuart Crome
        Modified: June 12, 2011
        Parameters: object sender, DetailsViewInsertedEventArgs e
        ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~*/
    protected void DetailsView1_ItemInserted(object sender,
DetailsViewInsertedEventArgs e)
    {
        string oledbConnectString = "Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=" +
@"C:\Documents and Settings\Stuart Crome\My Documents\Visual Studio
2008\WebSites\Box_v4\App_Data\BoX Contractual Payment System6_1_2011\" +
"BoX Contractual Payment System_6_1.accdb;";
```

```
        using (OleDbConnection connection = new
OleDbConnection(oledbConnectString))
        {
            try
            {
                connection.Open();
                //Response.Write("you are connected <br />");
                // create the DataSet
                DataSet ds = new DataSet();

                // create the adapter and fill the DataSet
                OleDbDataAdapter adapter =
                new OleDbDataAdapter("SELECT [tblContracts].[Contract_StartDate],
[tblContracts].[Contract_EndDate], [tblContracts].[Contract_HourlyRate],
[tblContracts].[Contract_FeeMax],[tblInvoices].[Invoice_StartDate],
[tblInvoices].[Invoice_EndDate], [tblInvoices].[Invoice_HourlyRate],
[tblInvoices].[Invoice_Total] FROM [tblContracts], [tblInvoices] WHERE
[tblContracts].[Contract_ID] = [tblInvoices].[Contract_ID]", connection);
                adapter.Fill(ds);
                DataTable dt = ds.Tables[0];
                foreach (DataRow dr in dt.Rows)
                {
                    string cStart = (dr["Contract_StartDate"].ToString());
                    string cEnd = (dr["Contract_EndDate"].ToString());
                    string cHourly = (dr["Contract_HourlyRate"].ToString());
                    string cFeeMax = (dr["Contract_FeeMax"].ToString());
                    string iStart = (dr["Invoice_StartDate"].ToString());
                    string iEnd = (dr["Invoice_EndDate"].ToString());
                    string iHourly = (dr["Invoice_HourlyRate"].ToString());
                    string iTotal = (dr["Invoice_Total"].ToString());

                    DateTime cStartDate = new DateTime();
                    DateTime cEndDate = new DateTime();
                    Decimal cHourlyRate;
                    Decimal cMax;

                    DateTime iStartDate = new DateTime();
                    DateTime iEndDate = new DateTime();
                    Decimal iHourlyRate;
                    Decimal iInvoiceTotal;

                    cStartDate = DateTime.Parse(cStart);
                    cEndDate = DateTime.Parse(cEnd);
                    cHourlyRate = Decimal.Parse(cHourly);
                    cMax = Decimal.Parse(cFeeMax);
                    iStartDate = DateTime.Parse(iStart);
                    iEndDate = DateTime.Parse(iEnd);
                    iHourlyRate = Decimal.Parse(iHourly);
                    iInvoiceTotal = Decimal.Parse(iTotal);

                    if ((iStartDate < cStartDate) || (iEndDate > cEndDate))
                    {
                        tbxMessage.Text += "The invoice dates you entered are
beyond the scope of contract dates\r\n";
                    }
                    if (cHourlyRate != iHourlyRate)
                    {
                        tbxMessage.Text += "The invoice hourly rate does not match
the contract hourly rate\r\n";
                    }
```

```
                    if (iInvoiceTotal > cMax)
                    {
                        tbxMessage.Text += "The invoice total you entered exceeds
the contract fee max\r\n";
                    }
                }

            }
            catch (Exception ex)
            {
                Response.Write("Failed to connect to data source");
                Response.Write(ex);
            }
            finally
            {
                connection.Close();
            }
        }
    }
    protected void btnExit_Click(object sender, EventArgs e)
    {
        Response.Write("<script>window.close()</script>");
    }
}
```