Rebecca Chacko
BCIS 4720
12/6/2022

Update from Database

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.Sql;
using System.Data.SqlClient;
using System.Xml.Linq;

namespace Website1
{
   public partial class Update : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {
         //Logical test to see if user is logged in. If user is not logged in, bounce back to login.
         if (!IsPostBack)
         {
            //Set textbox visibility to false.
            txtFname.Visible = false;
            txtLname.Visible = false;
            txtYOB.Visible = false;
            txtPass1.Visible = false;
            txtConfPass.Visible = false;
         }
```

Rebecca Chacko
BCIS 3680
12/6/2022

```
//Set up our objects (connection, command, and datareader)
//Declare a SQLConnection
SqlConnection conRegisterCon = new SqlConnection(SqlDataSource1.ConnectionString);

//Declare a SQLCommand
SqlCommand cmdInsert = new SqlCommand(SqlDataSource1.SelectCommand);

//Declare datareader
SqlDataReader drLoginReader;

//Configure command with connection and add parameter value.
cmdInsert.Connection = conRegisterCon;
cmdInsert.Parameters.AddWithValue("@Email", Session["User"].ToString());

//Open Connection
conRegisterCon.Open();

//Execute datareader
drLoginReader = cmdInsert.ExecuteReader(CommandBehavior.CloseConnection);

//While loop to fill controls on page.
while (drLoginReader.Read())
{
   lblFName.Text = drLoginReader.GetString(2).Trim();
   lblLName.Text = drLoginReader.GetString(3).Trim();
   lblYOB.Text = drLoginReader.GetString(4).Trim();
   lblPass.Text = drLoginReader.GetString(4).Trim();
   lblConfPass.Text = drLoginReader.GetString(4).Trim();
```

```
        //Populate labels
        lblFName.Text = "";
        lblLName.Text = "";
        lblPass.Text = "";
        lblConfPass.Text = "";
        lblYOB.Text = "";
    }

    //Close Connection
    conRegisterCon.Close();
}
protected void BtnUpdateInfo_Click(object sender, EventArgs e)
{
    //Set up objects (connection, command)
    SqlConnection conUpdateCon = new SqlConnection(SqlDataSource1.ConnectionString);
    SqlCommand cmdUpdate = new SqlCommand(SqlDataSource1.UpdateCommand);

    //Configure command with connection and add parameter value.
    cmdUpdate.Connection = conUpdateCon;
    cmdUpdate.Parameters.AddWithValue("Email", Session["User"].ToString());

    //Load appropriate parameter values. Use an if statement for all parameter values
    //other than email. If checked then pull from textbox, if not pull from label.
    if (cbFname.Checked == true && txtFname.Text != "")
    {
        cmdUpdate.Parameters.AddWithValue("Fname", txtFname.Text);
    }
```

```
else
{
   cmdUpdate.Parameters.AddWithValue("Fname", txtFname.Text);
}
if (cbLname.Checked == true && txtLname.Text != "")
{
   cmdUpdate.Parameters.AddWithValue("LName", lblLName.Text);
}
else
{
   cmdUpdate.Parameters.AddWithValue("Lname", lblLName.Text);
}
if (cbPass.Checked == true && txtConfPass.Text != txtConfPass.Text)
{
   cmdUpdate.Parameters.AddWithValue("Password", txtConfPass.Text);
}
else
{
   cmdUpdate.Parameters.AddWithValue("Password", lblPass.Text);
}
if (cbYOB.Checked == true && txtYOB.Text != "")
{
   cmdUpdate.Parameters.AddWithValue("YOB", Convert.ToInt32(txtYOB.Text));
}
else
{
   cmdUpdate.Parameters.AddWithValue("YOB", Convert.ToInt32(lblYOB.Text));
}
```

```
    //Open our connection.
    conUpdateCon.Open();

    //Execute nonquery
    cmdUpdate.ExecuteNonQuery();

    //Close connection.
    conUpdateCon.Close();

    //Redirect to self
    Response.Redirect("Update.aspx");
}

protected void CbFname_CheckedChanged(object sender, EventArgs e)
{
    //Tell app what to show if checkbox is not checked.
    if (cbFname.Checked == false)
    {
        txtFname.Visible = false;
    }
    //Tell app what to do if checkbox is checked
    else
    {
        txtFname.Visible=true;
    }
}
protected void CbLname_CheckedChanged(object sender, EventArgs e)
```

```
{
   //Tell app what to show if checkbox is not checked.
   if(cbLname.Checked == false)
   {
      txtLname.Visible = false;
   }
   //Tell app what to do if checkbox is checked
   else
   {
      txtLname.Visible=true;
   }
}

protected void CbYOB_CheckedChanged(object sender, EventArgs e)
{
   //Tell app what to show if checkbox is not checked.
   if(cbYOB.Checked == false)
   {
      txtYOB.Visible = false;
   }
   //Tell app what to do if checkbox is checked
   else
   {
      txtYOB.Visible=true;
   }
}

protected void CbPass_CheckedChanged(object sender, EventArgs e)
```

```
        {
            //Tell app what to show if checkbox is not checked.
            if(cbPass.Checked == false)
            {
                txtPass1.Visible = false;
                txtConfPass.Visible = false;
            }
            //Tell app what to do if checkbox is checked
            else
            {
                txtPass1.Visible=true;
                txtConfPass.Visible=true;
            }
        }
    }
}
```