MENU ITEM BINDING

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Data;

public partial class menu : System.Web.UI.Page

{

logic lg = new logic();

Menu menu1 = new Menu();

protected void Page\_Load(object sender, EventArgs e)

{

DataSet ds = lg.selectvenue();

MenuItem mi = new MenuItem("venues");

menu1.Items.Add(mi);

foreach (DataRow parent in ds.Tables["venue"].Rows)

{

MenuItem mn = new MenuItem((string)parent["venue\_name"]);

mn.Value = Convert.ToString(parent["venue\_id"]);

mi.ChildItems.Add(mn);

}

Panel1.Controls.Add(menu1);

Panel1.DataBind();

menu1.MenuItemClick += new MenuEventHandler(menu1\_MenuItemClick);

}

protected void menu1\_MenuItemClick(object sender, MenuEventArgs e)

{

int vid=Convert.ToInt32(e.Item.Value);

Response.Redirect("showvenue.aspx?id="+vid);

}

}

ADDING VENUE

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.IO;

public partial class addvenue : System.Web.UI.Page

{

logic lg = new logic();

protected void Page\_Load(object sender, EventArgs e)

{

}

protected void btn\_addvenue\_Click(object sender, EventArgs e)

{

lg.vename=TextBox1.Text;

lg.venimage = GetPhoto();

if (lg.addven\_click(lg))

{

Label3.Text = "data inserted";

}

else

{

Label3.Text = "data not inserted";

}

}

public string GetPhoto()

{

string strPhoto;

try

{

//Call the function to convert file uploaded to base64string

strPhoto = ConvertoBase64(ref FileUpload1);

if (string.IsNullOrEmpty(strPhoto))

{

if (!string.IsNullOrEmpty((string)Session["UserPhoto"]))

{

strPhoto = (string)Session["UserPhoto"];

//imgCBSELogo.Visible = true;

return strPhoto;

}

}

else

{

Session.Add("UserPhoto", strPhoto);

//imgCBSELogo.Visible = true;

}

}

catch (Exception ex)

{

throw ex;

}

return strPhoto;

}

public string ConvertoBase64(ref System.Web.UI.WebControls. FileUpload upldCtrl)

{

using (MemoryStream objMemoryStream = new MemoryStream())

{

string imgContentType = upldCtrl.PostedFile.ContentType;

// Convert Image to byte[]

switch (imgContentType)

{

case "image/jpeg":

System.Drawing.Image.FromStream(upldCtrl.PostedFile.InputStream).Save(objMemoryStream, System.Drawing.Imaging.ImageFormat.Jpeg);

break;

case "image/gif":

System.Drawing.Image.FromStream(upldCtrl.PostedFile.InputStream).Save(objMemoryStream, System.Drawing.Imaging.ImageFormat.Gif);

break;

case "image/bmp":

System.Drawing.Image.FromStream(upldCtrl.PostedFile.InputStream).Save(objMemoryStream, System.Drawing.Imaging.ImageFormat.Bmp);

break;

case "image/png":

System.Drawing.Image.FromStream(upldCtrl.PostedFile.InputStream).Save(objMemoryStream, System.Drawing.Imaging.ImageFormat.Png);

break;

}

byte[] objImageBytes = objMemoryStream.ToArray();

string base64String = Convert.ToBase64String(objImageBytes);

return base64String;

}

}

}

SHOW VENUE

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Data;

public partial class showvenue : System.Web.UI.Page

{

logic lg = new logic();

DataSet ds = new DataSet();

protected void Page\_Load(object sender, EventArgs e)

{

lg.v = Convert.ToInt32(Request.QueryString["id"]);

ds = lg.shvenue(lg);

Label1.Text = Convert.ToString(ds.Tables["venue"].Rows[0][0]);

Session["IMAGE"] =ds.Tables["venue"].Rows[0][1];

Image1.ImageUrl = "~/showimage.aspx";

}

}

SHOW IMAGE

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.IO;

using System.Data;

public partial class showimage : System.Web.UI.Page

{

logic lg = new logic();

DataSet ds = new DataSet();

protected void Page\_Load(object sender, EventArgs e)

{

string strBase64 = (string)Session["IMAGE"];

if (string.IsNullOrEmpty(strBase64))

return;

byte[] bImage = Convert.FromBase64String(strBase64);

MemoryStream ms = new MemoryStream(bImage, 0, bImage.Length);

// Convert byte[] to Image

ms.Write(bImage, 0, bImage.Length);

Response.Buffer = true;

Response.BinaryWrite(bImage);

ms.Dispose();

//byte[] imgbytes = (byte[])Session["IMAGE"];

//MemoryStream ms = new MemoryStream(imgbytes, 0, imgbytes.Length);

//ms.Write(imgbytes, 0, imgbytes.Length);

//Response.Buffer = true;

//Response.BinaryWrite(imgbytes);

//ms.Dispose();

}

private void ShowImage()

{

//try

//{

// ////Session.Add("DJImage", null);

// //string strBase64 = (string)Session["image"];

// //byte[] imageBytes = Convert.FromBase64String(strBase64);

// //MemoryStream ms = new MemoryStream(imageBytes, 0, imageBytes.Length);

// //// Convert byte[] to Image

// //ms.Write(imageBytes, 0, imageBytes.Length);

// //Response.Buffer = true;

// //Response.BinaryWrite(imageBytes);

// //ms.Dispose();

// byte[] imgbytes = (byte[])Session["image"];

// Response.BinaryWrite(imgbytes);

// //Image newImage;

// //MemoryStream stream = new MemoryStream(imgbytes);

// //newImage = Image.FromStream(stream);

//}

//catch (Exception objException)

//{

//}

}

}