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.Data.OleDb;

namespace SinanScrumTable

{

public partial class Form1 : Form

{

OleDbCommand komut = new OleDbCommand();

OleDbConnection baglanti = new OleDbConnection();

int oncekisayi = 0;

int oncekisayi2 = 0;

int objepanel2 = 0;

int objepanel3 = 0;

int objepanel4 = 0;

object dragsource = null;

public Form1()

{

InitializeComponent();

}

void VeritabaninaBaglan()

{

baglanti.ConnectionString = "Provider=Microsoft.ACE.Oledb.12.0;Data Source=Scrum.accdb";

komut.Connection = baglanti;

baglanti.Open();

}

void VeritabaniBaglantisiniKapat()

{

baglanti.Close();

}

private void btnStoryEkle\_Click(object sender, EventArgs e)

{

DialogResult result = MessageBox.Show("Yeni Story eklensin mi?", "Onay", MessageBoxButtons.YesNo);

if (result == System.Windows.Forms.DialogResult.Yes)

{

VeritabaninaBaglan();

komut.CommandText = "INSERT into Stories (Story) VALUES('" + txtYeniStory.Text + "')";

komut.ExecuteNonQuery();

MessageBox.Show("Kayıt başarıyla eklendi! Listenin güncellenmesi için uygulamayı kapatıp açın.");

VeritabaniBaglantisiniKapat();

StoryListele();

}

}

public class ComboboxItem

{

public string Text { get; set; }

public object Value { get; set; }

public override string ToString()

{

return Text;

}

}

public class MyPictureBox : PictureBox

{

private string storyid;

private string catergoryid;

public string StoryId { get { return storyid; } set { storyid = value; } }

public string CategoryId { get { return catergoryid; } set { catergoryid = value; } }

}

public class MyRichTextBox : RichTextBox

{

private string storyid;

private string catergoryid;

private int cat1count;

private int cat2count;

private int cat3count;

public int Cat1Count { get { return cat1count; } set { cat1count = value; } }

public int Cat2Count { get { return cat2count; } set { cat2count = value; } }

public int Cat3Count { get { return cat3count; } set { cat3count = value; } }

}

Color RenkSec()

{

Random rastgele = new Random();

int sayi = rastgele.Next(1,5);

while (oncekisayi == sayi)

{

sayi = rastgele.Next(1, 5);

}

oncekisayi = sayi;

if (sayi == 1) { return System.Drawing.Color.LightCyan; }

if (sayi == 2) { return System.Drawing.Color.Pink; }

if (sayi == 3) { return System.Drawing.Color.PaleGoldenrod; }

if (sayi == 4) { return System.Drawing.Color.LightGreen; }

else { return System.Drawing.Color.Thistle; }

}

Image TaskResimAl()

{

Random rastgele = new Random();

int sayi = rastgele.Next(1, 5);

while (oncekisayi2 == sayi)

{

sayi = rastgele.Next(1, 5);

}

oncekisayi2 = sayi;

if (sayi == 1) { return Image.FromFile(@"resimler\t1.png"); }

if (sayi == 2) { return Image.FromFile(@"resimler\t2.png"); }

if (sayi == 3) { return Image.FromFile(@"resimler\t3.png"); }

if (sayi == 4) { return Image.FromFile(@"resimler\t4.png"); }

else { return Image.FromFile(@"resimler\t5.png"); }

}

void StoryListele()

{

ComboboxItem item = new ComboboxItem();

komut.CommandText = "SELECT Id, Story FROM Stories";

DataTable tablom = new DataTable();

OleDbDataAdapter verilericek = new OleDbDataAdapter(komut);

verilericek.Fill(tablom);

int rtBoyut = 0;

for (int i = 0; i < tablom.Rows.Count; i++)

{

item.Text = tablom.Rows[i][1].ToString();

item.Value = tablom.Rows[i][0].ToString();

//cbStroy.Items.Add(item);

cbStroy.Items.Add(new { Text = tablom.Rows[i][1].ToString(), Value = tablom.Rows[i][0].ToString() });

cbStroy.DisplayMember = "Text";

cbStroy.ValueMember = "Value";

cbStroy.SelectedIndex = 0;

MyRichTextBox rTBox = new MyRichTextBox();

rTBox.Text = tablom.Rows[i][1].ToString();

rTBox.BackColor = RenkSec();

rTBox.Cat1Count = 0;

rTBox.Cat2Count = 0;

rTBox.Cat3Count = 0;

rTBox.Location = new System.Drawing.Point(5, rtBoyut);

rTBox.Name = "rTBox" + item.Value;

rTBox.Size = new System.Drawing.Size(117, 82);

rTBox.TabIndex = 0;

panel1.Controls.Add(rTBox);

rtBoyut += 90;

}

}

Point TaskYerlestir(int taskcategoryid, int satirY, string name)

{

int yerlestirY = 0;

int yerlestirX = 0;

if (taskcategoryid == 1)

{

int bolum = 0;

bolum = objepanel2 / 2;

yerlestirY = (35 \* bolum) + ((bolum + 1) \* 5);

bolum = 0;

if (objepanel2 % 2 == 1) { yerlestirX = ((bolum+1) \* 51) + ((bolum + 1) \* 10); }

else { yerlestirX = 5; }

}

if (taskcategoryid == 2)

{

int bolum = 0;

bolum = objepanel3 / 2;

yerlestirY = (35 \* bolum) + ((bolum + 1) \* 5);

bolum = 0;

if (objepanel3 % 2 == 1) { yerlestirX = ((bolum+1) \* 51) + ((bolum + 1) \* 10); }

else { yerlestirX = 5; }

}

if (taskcategoryid == 3)

{

int bolum = 0;

bolum = objepanel4 / 2;

yerlestirY = (35 \* bolum) + ((bolum + 1) \* 5);

bolum = 0;

if (objepanel4 % 2 == 1) { yerlestirX = ((bolum+1) \* 51) + ((bolum + 1) \* 10); }

else { yerlestirX = 5; }

}

yerlestirY += satirY;

return new Point(yerlestirX, yerlestirY);

}

void TaskTasi(int taskcategoryid, int satirY, string name, MyPictureBox a, int objesay)

{

int yerlestirY = 0;

int yerlestirX = 0;

int bolum = 0;

bolum = objesay / 2;

yerlestirY = (35 \* bolum) + ((bolum + 1) \* 5);

bolum = 0;

if (objesay % 2 == 1) { yerlestirX = ((bolum+1) \* 51) + ((bolum + 1) \* 10); }

else { yerlestirX = 5; }

yerlestirY += satirY;

a.Location = new Point(yerlestirX, yerlestirY);

}

void TaskleriListele2()

{

ComboboxItem item = new ComboboxItem();

komut.CommandText = "SELECT Id, TaskName, TaskCategoryId, TaskStoryId FROM Tasks";

DataTable tablom = new DataTable();

OleDbDataAdapter verilericek = new OleDbDataAdapter(komut);

verilericek.Fill(tablom);

int rtBoyut = 5;

int satirY = 5;

int yerlestirY = 5;

int sayac = -1;

for (int i = 0; i < tablom.Rows.Count; i++)

{

item.Text = tablom.Rows[i][1].ToString();

item.Value = tablom.Rows[i][0].ToString();

string tsid = tablom.Rows[i][3].ToString();

for (int x = 0; x < cbStroy.Items.Count; x++)

{

sayac++;

if ((cbStroy.Items[x] as dynamic).Value == tsid) { break; }

}

satirY = (80 \* sayac);

cbTasks.Items.Add(new { Text = tablom.Rows[i][1].ToString(), Value = tablom.Rows[i][0].ToString() });

cbTasks.DisplayMember = "Text";

cbStroy.ValueMember = "Value";

cbTasks.SelectedIndex = 0;

object p = null;

for (int a = panelgizli.Controls.Count - 1; a >= 0; a--)

{

if (panelgizli.Controls[a].Name == item.Text.Replace(" ", "\_"))

{

p = panelgizli.Controls[a];

}

}

var pictureBoxx = p as PictureBox;

if (tablom.Rows[i][2].ToString() == "1") {

panel2.Controls.Add(pictureBoxx);

pictureBoxx.Location = TaskYerlestir(1, satirY, pictureBoxx.Name);

objepanel2++;

}

if (tablom.Rows[i][2].ToString() == "2") {

panel3.Controls.Add(pictureBoxx);

pictureBoxx.Location = TaskYerlestir(2, satirY, pictureBoxx.Name);

objepanel3++;

}

if (tablom.Rows[i][2].ToString() == "3")

{

panel4.Controls.Add(pictureBoxx);

pictureBoxx.Location = TaskYerlestir(3, satirY, pictureBoxx.Name);

objepanel4++;

}

sayac = -1;

//rtBoyut += 80;

}

}

void TaskleriListele()

{

int rtBoyut = 5;

int satirY = 0;

int yerlestirY = 5;

int sayac = -1;

OleDbDataAdapter odat = new OleDbDataAdapter("SELECT Id, Story FROM Stories", baglanti);

DataSet ds = new DataSet();

odat.Fill(ds);

for (int z = 0; z < ds.Tables[0].Rows.Count; z++)

{

ComboboxItem item = new ComboboxItem();

komut.CommandText = "SELECT Id, TaskName, TaskCategoryId, TaskStoryId FROM Tasks WHERE TaskStoryId=" + ds.Tables[0].Rows[z][0].ToString();

DataTable tablom = new DataTable();

OleDbDataAdapter verilericek = new OleDbDataAdapter(komut);

verilericek.Fill(tablom);

for (int i = 0; i < tablom.Rows.Count; i++)

{

item.Text = tablom.Rows[i][1].ToString();

item.Value = tablom.Rows[i][0].ToString();

string tsid = tablom.Rows[i][3].ToString();

cbTasks.Items.Add(new { Text = tablom.Rows[i][1].ToString(), Value = tablom.Rows[i][0].ToString() });

cbTasks.DisplayMember = "Text";

cbStroy.ValueMember = "Value";

cbTasks.SelectedIndex = 0;

MyPictureBox pictureBoxx = new MyPictureBox();

pictureBoxx.StoryId = tablom.Rows[i][3].ToString();

pictureBoxx.CategoryId = tablom.Rows[i][2].ToString();

pictureBoxx.Image = TaskResimAl();

int bolum = 0;

bolum = objepanel2 / 2;

if (tablom.Rows[i][2].ToString() == "1")

{

if (objepanel2 % 2 == 0) { yerlestirY = (bolum \* 35) + ((bolum + 1) \* 5); }

else { }

}

pictureBoxx.Location = new System.Drawing.Point(rtBoyut, yerlestirY);

pictureBoxx.Name = item.Text.Replace(" ", "\_");

//"pictureBox" + i.ToString();

pictureBoxx.Size = new System.Drawing.Size(51, 35);

pictureBoxx.SizeMode = System.Windows.Forms.PictureBoxSizeMode.StretchImage;

pictureBoxx.TabIndex = 3;

pictureBoxx.TabStop = false;

pictureBoxx.Paint += new PaintEventHandler((sender, e) =>

{

e.Graphics.TextRenderingHint = System.Drawing.Text.TextRenderingHint.ClearTypeGridFit;

string text = pictureBoxx.Name.Replace("\_", " ");

SizeF textSize = e.Graphics.MeasureString(text, Font);

PointF locationToDraw = new PointF();

locationToDraw.X = (pictureBoxx.Width / 2) - (textSize.Width / 2);

locationToDraw.Y = (pictureBoxx.Height / 2) - (textSize.Height / 2);

e.Graphics.DrawString(text, Font, Brushes.Black, locationToDraw);

});

pictureBoxx.Invalidate();

pictureBoxx.MouseDown += new System.Windows.Forms.MouseEventHandler((sender, e) =>

{

dragsource = pictureBoxx;

pictureBoxx.DoDragDrop(pictureBoxx.Image, DragDropEffects.All);

});

MyRichTextBox FoundStory = panel1.Controls.Find("rTBox" + tsid, true).FirstOrDefault() as MyRichTextBox;

if (tablom.Rows[i][2].ToString() == "1")

{

panel2.Controls.Add(pictureBoxx);

pictureBoxx.Location = TaskYerlestir(1, satirY, pictureBoxx.Name);

objepanel2++;

FoundStory.Cat1Count++;

}

if (tablom.Rows[i][2].ToString() == "2") { panel3.Controls.Add(pictureBoxx); pictureBoxx.Location = TaskYerlestir(2, satirY, pictureBoxx.Name); objepanel3++; FoundStory.Cat2Count++; }

if (tablom.Rows[i][2].ToString() == "3") { panel4.Controls.Add(pictureBoxx); pictureBoxx.Location = TaskYerlestir(3, satirY, pictureBoxx.Name); objepanel4++; FoundStory.Cat3Count++; }

sayac = -1;

}

satirY += 90;

objepanel2 = 0;

objepanel3 = 0;

objepanel4 = 0;

}

}

public class Mysource

{

object source { get; set; }

}

void TaskKategoriListele()

{

ComboboxItem item = new ComboboxItem();

komut.CommandText = "SELECT Id, TaskCategory FROM TaskCategories";

DataTable tablom = new DataTable();

OleDbDataAdapter verilericek = new OleDbDataAdapter(komut);

verilericek.Fill(tablom);

for (int i = 0; i < tablom.Rows.Count; i++)

{

cbTaskKategorileri.Items.Add(new { Text = tablom.Rows[i][1].ToString(), Value = tablom.Rows[i][0].ToString() });

cbTaskKategorileri.SelectedIndex = 0;

}

cbTaskKategorileri.DisplayMember = "Text";

cbTaskKategorileri.ValueMember = "Value";

}

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

{

VeritabaninaBaglan();

cbTasks.Hide();

StoryListele();

TaskKategoriListele();

TaskleriListele();

VeritabaniBaglantisiniKapat();

}

private void btnTaskEkle\_Click(object sender, EventArgs e)

{

if (txtYeniTask.Text.Trim() == "" || txtTaskYapacakKisi.Text.Trim() == "" || (cbStroy.SelectedItem as dynamic).Value == "" || (cbTaskKategorileri.SelectedItem as dynamic).Value == "") { MessageBox.Show("Formu eksiksiz doldurunuz!"); return; }

DialogResult result = MessageBox.Show("Yeni Task eklensin mi?", "Onay", MessageBoxButtons.YesNo);

if (result == System.Windows.Forms.DialogResult.Yes)

{

VeritabaninaBaglan();

komut.CommandText = "INSERT into Tasks (TaskName, TaskAuthor, TaskDate, TaskCategoryId, TaskStoryId) VALUES('" + txtYeniTask.Text + "','" + txtTaskYapacakKisi.Text + "','" + dtTaskTarihi.Value + "'," + (cbTaskKategorileri.SelectedItem as dynamic).Value + "," + (cbStroy.SelectedItem as dynamic).Value + ")";

komut.ExecuteNonQuery();

MessageBox.Show("Kayıt başarıyla eklendi! Listenin güncellenmesi için uygulamayı kapatıp açın.");

VeritabaniBaglantisiniKapat();

//YeniTaskEkle(XmlReadMode,(cbTaskKategorileri.SelectedItem as dynamic).Value,(cbStroy.SelectedItem as dynamic).Value)

}

}

private void Form1\_DragEnter(object sender, DragEventArgs e)

{

e.Effect = DragDropEffects.Move;

Application.DoEvents();

}

private void panel2\_DragEnter(object sender, DragEventArgs e)

{

panel2.BackColor = Color.White;

e.Effect = DragDropEffects.Move;

}

private void panel2\_DragLeave(object sender, EventArgs e)

{

panel2.BackColor = Color.Transparent;

}

private void panel3\_DragEnter(object sender, DragEventArgs e)

{

panel3.BackColor = Color.White;

e.Effect = DragDropEffects.Move;

}

private void panel3\_DragLeave(object sender, EventArgs e)

{

panel3.BackColor = Color.Transparent;

}

private void panel4\_DragEnter(object sender, DragEventArgs e)

{

panel4.BackColor = Color.White;

e.Effect = DragDropEffects.Move;

}

private void panel4\_DragLeave(object sender, EventArgs e)

{

panel4.BackColor = Color.Transparent;

}

void Resetle(MyPictureBox a, int kategori)

{

int TasiId = 0;

for (int x = 0; x < cbTasks.Items.Count; x++)

{

if ((cbTasks.Items[x] as dynamic).Text == a.Name.Replace("\_", " ")) { TasiId = Convert.ToInt32((cbTasks.Items[x] as dynamic).Value); }

}

VeritabaninaBaglan();

komut.CommandText = "UPDATE Tasks SET TaskCategoryId=" + kategori + " WHERE Id=" + TasiId;

komut.ExecuteNonQuery();

VeritabaniBaglantisiniKapat();

MyRichTextBox FoundStory = panel1.Controls.Find("rTBox" + a.StoryId, true).FirstOrDefault() as MyRichTextBox;

if (a.CategoryId == "1") { FoundStory.Cat1Count--; }

if (a.CategoryId == "2") { FoundStory.Cat2Count--; }

if (a.CategoryId == "3") { FoundStory.Cat3Count--; }

int sayac = -1;

for (int x = 0; x < cbStroy.Items.Count; x++)

{

sayac++;

if ((cbStroy.Items[x] as dynamic).Value == a.StoryId) { break; }

}

if (kategori == 1) { panel2.Controls.Add(a); TaskTasi(kategori, (90 \* sayac), a.Name, a, FoundStory.Cat1Count); FoundStory.Cat1Count++; a.CategoryId = "1"; }

if (kategori == 2) { panel3.Controls.Add(a); TaskTasi(kategori, (90 \* sayac), a.Name, a, FoundStory.Cat2Count); FoundStory.Cat2Count++; a.CategoryId = "2"; }

if (kategori == 3) { panel4.Controls.Add(a); TaskTasi(kategori, (90 \* sayac), a.Name, a, FoundStory.Cat3Count); FoundStory.Cat3Count++; a.CategoryId = "3"; }

}

void YeniTaskEkle(MyPictureBox a, int kategori)

{

int TasiId = 0;

for (int x = 0; x < cbTasks.Items.Count; x++)

{

if ((cbTasks.Items[x] as dynamic).Text == a.Name.Replace("\_", " ")) { TasiId = Convert.ToInt32((cbTasks.Items[x] as dynamic).Value); }

}

VeritabaninaBaglan();

komut.CommandText = "UPDATE Tasks SET TaskCategoryId=" + kategori + " WHERE Id=" + TasiId;

komut.ExecuteNonQuery();

VeritabaniBaglantisiniKapat();

MyRichTextBox FoundStory = panel1.Controls.Find("rTBox" + a.StoryId, true).FirstOrDefault() as MyRichTextBox;

if (a.CategoryId == "1") { FoundStory.Cat1Count--; }

if (a.CategoryId == "2") { FoundStory.Cat2Count--; }

if (a.CategoryId == "3") { FoundStory.Cat3Count--; }

int sayac = -1;

for (int x = 0; x < cbStroy.Items.Count; x++)

{

sayac++;

if ((cbStroy.Items[x] as dynamic).Value == a.StoryId) { break; }

}

if (kategori == 1) { panel2.Controls.Add(a); TaskTasi(kategori, (90 \* sayac), a.Name, a, FoundStory.Cat1Count); FoundStory.Cat1Count++; a.CategoryId = "1"; }

if (kategori == 2) { panel3.Controls.Add(a); TaskTasi(kategori, (90 \* sayac), a.Name, a, FoundStory.Cat2Count); FoundStory.Cat2Count++; a.CategoryId = "2"; }

if (kategori == 3) { panel4.Controls.Add(a); TaskTasi(kategori, (90 \* sayac), a.Name, a, FoundStory.Cat3Count); FoundStory.Cat3Count++; a.CategoryId = "3"; }

}

private void panel3\_DragDrop(object sender, DragEventArgs e)

{

panel3.BackColor = Color.Transparent;

var a = dragsource as MyPictureBox;

if (a.Parent.Name == "panel3") { return; }

Resetle(a, 2);

}

private void panel2\_DragDrop(object sender, DragEventArgs e)

{

panel2.BackColor = Color.Transparent;

var a = dragsource as MyPictureBox;

if (a.Parent.Name == "panel2") { return; }

Resetle(a, 1);

}

private void panel4\_DragDrop(object sender, DragEventArgs e)

{

panel4.BackColor = Color.Transparent;

var a = dragsource as MyPictureBox;

if (a.Parent.Name == "panel4") { return; }

Resetle(a, 3);

}

private void button1\_Click(object sender, EventArgs e)

{

this.Close();

}

}

}