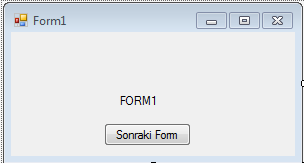
**Bolum10\_FormlarveKontroller**

**FORMLAR**

**Formlar\_Arası\_Gecis**

****

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

{

//this.Hide();

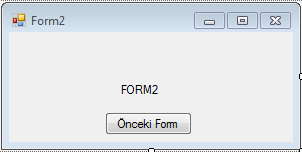
//this.Visible = false;

Form2 frm2 = new Form2();

//frm2.Show();

frm2.ShowDialog();

}



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

{

//this.Visible = false;

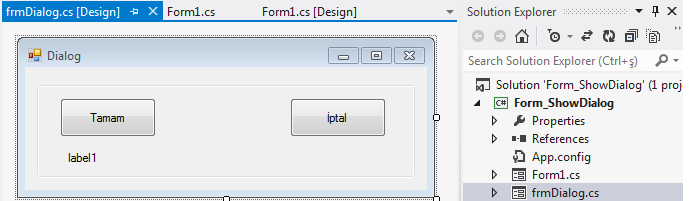
this.Close();

Form1 frm1 = new Form1();

frm1.Visible = true;

}

**Form\_ShowDialog**

****

private void btn\_tamam\_Click(object sender, EventArgs e)

{

this.DialogResult = System.Windows.Forms.DialogResult.OK;

this.Close();

}

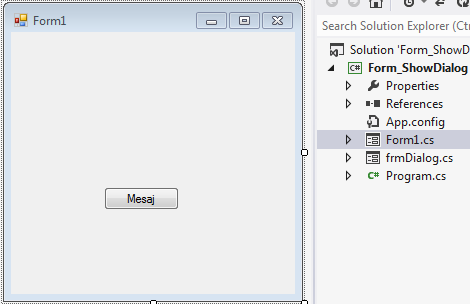
private void btn\_iptal\_Click(object sender, EventArgs e)

{

this.DialogResult = System.Windows.Forms.DialogResult.Cancel;

this.Close();

}



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

{

frmDialog diyalog\_formu = new frmDialog();

diyalog\_formu.label1.Text = "Silmek İstediğinize Emin misiniz?";

diyalog\_formu.ShowDialog();

if (diyalog\_formu.DialogResult == System.Windows.Forms.DialogResult.OK)

{

MessageBox.Show("Dosyalar Silindi");

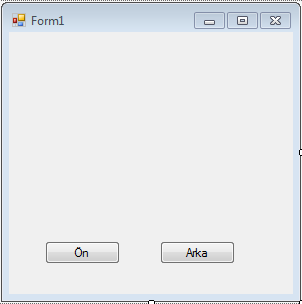
}

diyalog\_formu.Dispose();

}

**Form\_BringToFront\_SendToBack**

**2.formu ekle**

****

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

{

Form2 frm2 = new Form2();

frm2.Show();

frm2.BringToFront();

}

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

{

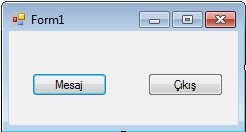
Form2 frm2 = new Form2();

frm2.Show();

frm2.SendToBack();

}

**Form\_AcceptButton\_CancelButton**



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

{

MessageBox.Show("AcceptButton");

}

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

{

Application.Exit();

}

**Form\_Acilis**

**using System.Diagnostics;**

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

{

Debug.Print("Activated Olayı");

}

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

{

Debug.Print("BindingContextChanged Olayı");

}

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

{

Debug.Print("Load Olayı");

}

private void Form1\_Paint(object sender, PaintEventArgs e)

{

Debug.Print("Paint Olayı");

}

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

{

Debug.Print("Shown Olayı");

}

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

{

Debug.Print("VisibleChanged Olayı");

}

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

{

Debug.Print("Validated Olayı");

}

**Form\_Kapanis**

private void Form1\_FormClosing(object sender, FormClosingEventArgs e)

{

Debug.Print("FormClosing Olayı");

}

private void Form1\_FormClosed(object sender, FormClosedEventArgs e)

{

Debug.Print("FormClosed Olayı");

}

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

{

Debug.Print("Deactivate Olayı");

}

**Form\_e\_cancel**

private void Form1\_FormClosing(object sender, FormClosingEventArgs e)

{

MessageBox.Show("Uygulamadan Çıkamazsınız!");

e.Cancel = true;

}

**Form\_CloseReason**

using System;

using System.Diagnostics;

using System.Windows.Forms;

namespace Form\_CloseReason

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void Form1\_FormClosing(object sender, FormClosingEventArgs e)

{

if (e.CloseReason == CloseReason.WindowsShutDown)

{

Process.Start("shutdown", "-a");

MessageBox.Show("Bilgisayarı Kapatma İşlemi İptal Edildi");

}

}

}

}

**Form\_KeyPress**

byte i;

private void Form1\_KeyPress(object sender, KeyPressEventArgs e)

{

i++;

if (e.KeyChar == 27) MessageBox.Show(i - 1 + " tuş tıklandı");

}

**Form\_KeyDown**

private void Form1\_KeyDown(object sender, KeyEventArgs e)

{

if (e.Control && e.KeyCode == Keys.C)

{

DialogResult cikis = MessageBox.Show

("Ctrl+C tuşlarına basıldı. Çıkmak istediğinize emin misiniz",

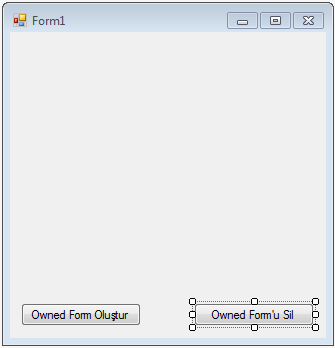
"Çıkış", MessageBoxButtons.OKCancel);

if (cikis == DialogResult.OK) Application.Exit();

}

}

**Form\_OwnedForm**

****

Form ownedForm1 = new Form();

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

{

this.AddOwnedForm(ownedForm1);

ownedForm1.Show();

}

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

{

this.RemoveOwnedForm(ownedForm1);

}

**LABEL KONTROLÜ**

**label\_ekle**

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

{

Label Label1 = new Label();

this.Controls.Add(Label1);

Label1.Text = "Güzel gören güzel düşünür. \r\n Güzel düşünen hayatından lezzet alır.";

Label1.Font = new Font("Comic Sans MS", 15, FontStyle.Italic);

Label1.ForeColor = Color.DarkBlue;

Label1.AutoSize = true;

}

**LİNKLABEL KONTROLÜ**

**LinkLabel\_Kontrolu**

****

private void linkLabel1\_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)

{

System.Diagnostics.Process.Start(e.Link.LinkData.ToString());

linkLabel1.LinkVisited = true;

}

private void linkLabel2\_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)

{

System.Diagnostics.Process.Start("mailto:volkanaktas@yahoo.com?subject=Soru Bankasi");

linkLabel2.LinkVisited = true;

}

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

{

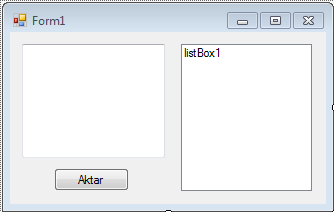
linkLabel1.Links.Add(0, 19, "www.volkanaktas.com");

linkLabel2.Links.Add(0, 24, "volkanaktas@yahoo.com");

}

**TEXTBOX KONTROLÜ**

**textbox2listbox**

****

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

{

for (int i = 0; i < textBox1.Lines.Length; i++)

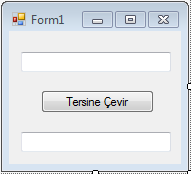
{

listBox1.Items.Add(textBox1.Lines[i]);

}

}

**textBox\_Reverse**

****

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

{

char[] str = textBox1.Text.ToCharArray();

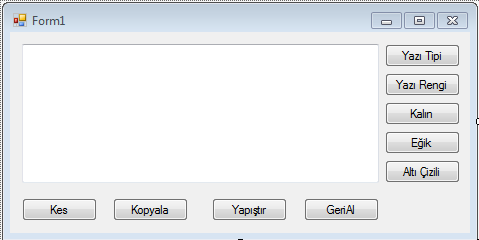
Array.Reverse(str);

string strReversed = new string(str);

textBox2.Text = strReversed;

}

**textBox\_Editor**

****

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

{

if (textBox1.SelectionLength > 0)

{

textBox1.Cut();

}

else

{

MessageBox.Show("Seçili Alan Yok");

}

}

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

{

if (textBox1.SelectionLength > 0)

{

textBox1.Copy();

}

else

{

MessageBox.Show("Seçili Alan Yok");

}

}

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

{

textBox1.Paste();

}

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

{

textBox1.Undo();

}

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

{

if (FontDialog1.ShowDialog() == DialogResult.OK)

textBox1.Font = FontDialog1.Font;

}

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

{

if (ColorDialog1.ShowDialog() == DialogResult.OK)

textBox1.ForeColor = ColorDialog1.Color;

}

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

{

textBox1.Font = new Font(Font, FontStyle.Bold);

}

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

{

textBox1.Font = new Font(Font, FontStyle.Italic);

}

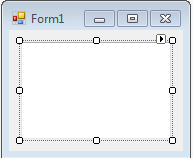
private void btnAltiCizili\_Click(object sender, EventArgs e)

{

textBox1.Font = new Font(Font, FontStyle.Underline);

}

**textBox\_AppendText**

****

int karakteradedi = 0;

private void textBox1\_KeyPress(object sender, KeyPressEventArgs e)

{

if (karakteradedi>=10)

{

if (karakteradedi % (10) == 0)

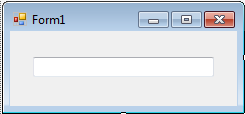
textBox1.AppendText(Environment.NewLine);

}

karakteradedi = karakteradedi + 1;

}

**textBox\_TextChanged**

****

private void textBox1\_TextChanged(object sender, EventArgs e)

{

CultureInfo CI = Thread.CurrentThread.CurrentCulture;

TextInfo TI = CI.TextInfo;

textBox1.Text = TI.ToTitleCase(textBox1.Text);

textBox1.Select(textBox1.Text.Length, 1);

}

**BUTON KONTROLÜ**

**Buton\_Basla\_Bitir**

**buton ekle**

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

{

if (button1.Text == "Başla")

button1.Text = "Bitir";

else

button1.Text = "Başla";

}

**Button\_Mouse\_Move\_Leave**

**buton ekle**

System.Drawing.Color renk;

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

{

button1.BackColor = renk;

}

private void button1\_MouseMove(object sender, MouseEventArgs e)

{

button1.BackColor = System.Drawing.Color.DarkGray;

}

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

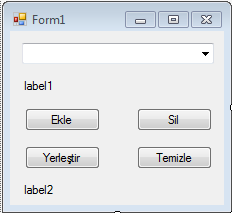
{

renk = button1.BackColor;

}

**COMBOBOX KONTROLÜ**

**comboBox\_Ekle\_Sil**

****

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

{

byte i;

for (i = 0; i <= 100; i++)

{

if (i % 2 == 0)

comboBox1.Items.Add(i);

}

label1.Text = comboBox1.Items.Count + " adet eleman var";

comboBox1.SelectedIndex = 0;

}

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

{

comboBox1.Items.Insert(0, -1);

comboBox1.SelectedIndex = 0;

label1.Text = comboBox1.Items.Count + " adet eleman var";

}

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

{

comboBox1.Items.Clear();

label1.Text = comboBox1.Items.Count + " adet eleman var";

}

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

{

if (comboBox1.SelectedIndex >= 0)

{

comboBox1.Items.RemoveAt(comboBox1.SelectedIndex);

}

else

comboBox1.Items.Remove(comboBox1.Items[0]);

comboBox1.SelectedIndex = 0;

label1.Text = comboBox1.Items.Count + " adet eleman var";

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

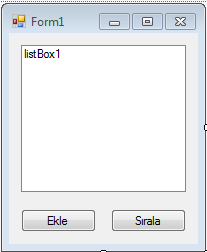
label2.Text = "Seçili eleman= " +

comboBox1.Items[comboBox1.SelectedIndex];

}

**LİSTBOX KONTROLÜ**

**listBox\_Sırala**

****

int i, sayi;

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

{

for (i = 1; i <= 10; i++)

{

sayi = Convert.ToInt32(Interaction.InputBox

(i + ". sayıyı giriniz."));

listBox1.Items.Add(sayi);

}

}

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

{

int l, m;

for (l = 0; l <= 9; l++)

{

for (m = l + 1; m <= 9; m++)

{

if (Int32.Parse(listBox1.Items[l].ToString()) >

Int32.Parse(listBox1.Items[m].ToString()))

{

int aradeger =

Int32.Parse(listBox1.Items[l].ToString());

listBox1.Items[l] = listBox1.Items[m];

listBox1.Items[m] = aradeger;

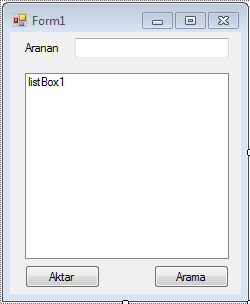
}

}

}

}

**listBox\_Arama**

****

int i;

Boolean bulundu = false;

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

{

for (i = 0; i <= 100; i++)

listBox1.Items.Add(i);

}

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

{

for (i = 1; i <= listBox1.Items.Count - 1; i++)

{

if (txtAranan.Text == listBox1.Items[i].ToString())

{

bulundu = true;

listBox1.SelectedIndex =

Int32.Parse(listBox1.Items[i].ToString());

break;

}

else bulundu = false;

}

if (bulundu == true)

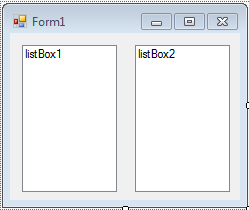
MessageBox.Show("Bulundu");

else

MessageBox.Show("Bulunamadı");

}

**listBox\_DragDrop**

****

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

{

byte i;

for (i = 1; i <= 10; i++)

listBox1.Items.Add(i);

}

private void listBox1\_MouseDown(object sender, MouseEventArgs e)

{

if (listBox1.Items.Count == 0) return;

string deger = listBox1.Items[listBox1.

IndexFromPoint(e.X, e.Y)].ToString();

if (DoDragDrop(deger, DragDropEffects.All) == DragDropEffects.All)

listBox1.Items.RemoveAt(listBox1.IndexFromPoint(e.X, e.Y));

}

private void listBox2\_DragDrop(object sender, DragEventArgs e)

{

if (e.Data.GetDataPresent(DataFormats.StringFormat))

listBox2.Items.Add(e.Data.GetData(DataFormats.StringFormat));

}

private void listBox2\_DragOver(object sender, DragEventArgs e)

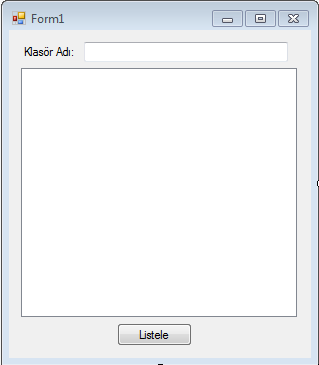
{

e.Effect = DragDropEffects.All;

}

**TREEVİEW\_KONTROLÜ**

**TreeView\_KlasorListesi**



using System;

using System.IO;

using System.Windows.Forms;

namespace TreeView\_KlasorListesi

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

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

{

string[] klasorler = Directory.GetDirectories(textBox1.Text);

treeView1.Nodes[0].Text = textBox1.Text;

TreeNode anadugum = treeView1.Nodes[0];

if (klasorler.Length != 0)

{

foreach (string aktifklasor in klasorler)

{

TreeNode dugum = new TreeNode(aktifklasor);

anadugum.Nodes.Add(dugum);

}

}

treeView1.ExpandAll();

}

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

{

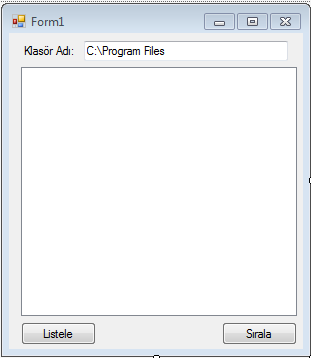
treeView1.Nodes.Add("Klasör Listesi");

}

}

}

**TreeView\_Sıralama**



using System;

using System.Collections;

using System.IO;

using System.Windows.Forms;

namespace TreeView\_Sıralama

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

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

{

treeView1.TreeViewNodeSorter = new twSorter();

//treeView1.Sort();

}

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

{

string[] klasorler = Directory.GetDirectories(textBox1.Text);

treeView1.Nodes[0].Text = textBox1.Text;

TreeNode anadugum = treeView1.Nodes[0];

if (klasorler.Length != 0)

{

foreach (string aktifklasor in klasorler)

{

TreeNode dugum = new TreeNode(aktifklasor);

anadugum.Nodes.Add(dugum);

}

}

treeView1.ExpandAll();

}

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

{

treeView1.Nodes.Add("Klasör Listesi");

}

}

public class twSorter : IComparer

{

public int Compare(object x, object y)

{

TreeNode tx = x as TreeNode;

TreeNode ty = y as TreeNode;

return string.Compare(ty.Text, tx.Text);

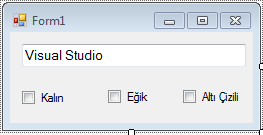
}

}

}

**CHECKBOX KONTROLÜ**

**CheckBox\_Yazıtipi**

****

private void chkKalin\_CheckedChanged(object sender, EventArgs e)

{

textBox1.Font = new Font(textBox1.Font.Name,

textBox1.Font.Size, textBox1.Font.Style ^ FontStyle.Bold);

}

private void chkEgik\_CheckedChanged(object sender, EventArgs e)

{

textBox1.Font = new Font(textBox1.Font.Name,

textBox1.Font.Size, textBox1.Font.Style ^ FontStyle.Italic);

}

private void chkAltiCizili\_CheckedChanged(object sender, EventArgs e)

{

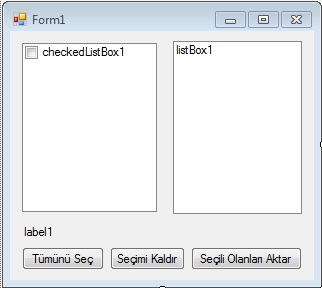
textBox1.Font = new Font(textBox1.Font.Name,

textBox1.Font.Size, textBox1.Font.Style ^ FontStyle.Underline);

}

**CHECKEDLİSTBOX\_KONTROLÜ**

**CheckedListBox\_Secim**

****

byte i;

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

{

for (i = 0; i <= checkedListBox1.Items.Count - 1; i++)

{

checkedListBox1.SetItemChecked(i, true);

}

label1.Text = checkedListBox1.CheckedItems.Count + " eleman seçili";

}

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

{

for (i = 1; i <= 10; i++)

checkedListBox1.Items.Add(i);

}

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

{

for (i = 0; i <= checkedListBox1.CheckedItems.Count - 1; i++)

listBox1.Items.Add(checkedListBox1.CheckedItems[i]);

}

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

{

for (i = 0; i <= checkedListBox1.Items.Count - 1; i++)

checkedListBox1.SetItemChecked(i, false);

label1.Text = checkedListBox1.CheckedItems.Count +

" eleman seçili";

}

private void checkedListBox1\_ItemCheck(object sender, ItemCheckEventArgs e)

{

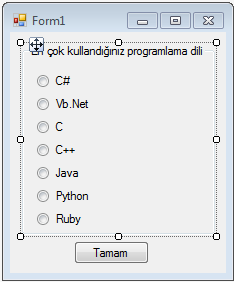
label1.Text = checkedListBox1.CheckedItems.Count - 1 +

" eleman seçili";

}

**RADİOBUTTON\_NESNESİ**

**RadioButton\_Diller**

****

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

{

if (radioButton1.Checked )

MessageBox.Show("En çok kullandığınız dil C#");

else if (radioButton2.Checked)

MessageBox.Show("En çok kullandığınız dil Vb.Net");

else if (radioButton3.Checked)

MessageBox.Show("En çok kullandığınız dil C");

else if (radioButton4.Checked)

MessageBox.Show("En çok kullandığınız dil C++");

else if (radioButton5.Checked)

MessageBox.Show("En çok kullandığınız dil Python");

else if (radioButton6.Checked)

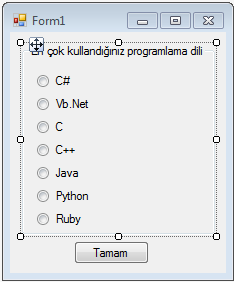
MessageBox.Show("En çok kullandığınız dil Ruby");

else if (radioButton7.Checked)

MessageBox.Show("En çok kullandığınız dil Delphi");

}

**RadioButton\_Diller\_foreach**

****

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

{

foreach (Control radio in groupBox1.Controls)

{

RadioButton rdb = (RadioButton)(radio);

if (rdb.Checked) MessageBox.Show

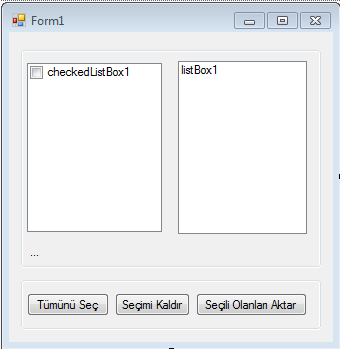
("En çok kullandığınız dil " + rdb.Text);

}

}

**GROUPBOX\_KONTROLU**

**GroupBox\_Kontrolu**

****

**GroupBox\_Kodla**

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

{

GroupBox groupBox1 = new GroupBox();

Label label1 = new Label();

label1.Location = new Point(20, 30);

label1.Text = "Adı Soyadı: ";

TextBox text1 = new TextBox();

text1.Text = "Volkan AKTAŞ";

text1.Location = new Point(150, 30);

Controls.Add(groupBox1);

groupBox1.FlatStyle = FlatStyle.Flat;

groupBox1.AutoSize = true;

groupBox1.Left = groupBox1.Padding.Left +

groupBox1.Margin.Left;

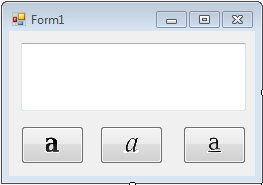
groupBox1.Controls.Add(label1);

groupBox1.Controls.Add(text1);

}

**IMAGELİST\_KONTROLU**

**ImageList\_Kontrolu**

****

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

{

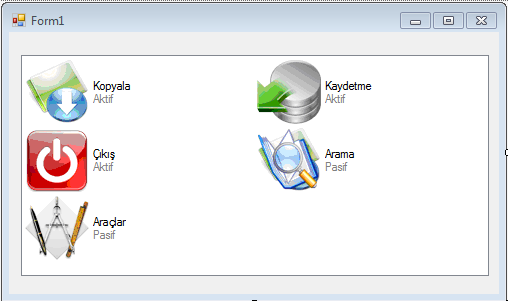
//button1.ImageIndex = 2;

//button1.Image = imageList1.Images[2];

}

**LİSTVİEW\_KONTROLU**

**listView\_Kontrolu**

****

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

{

ImageList imglist = new ImageList();

imglist.Images.Add(Bitmap.FromFile("./resimler/Dosya.ico"));

imglist.Images.Add(Bitmap.FromFile("./resimler/Klasor.ico"));

listView1.SmallImageList = imglist;

listView1.Columns.Add("İsim");

listView1.Columns.Add("Boyut");

listView1.Columns.Add("Tür");

ListViewItem item;

String[] klasorler = Directory.GetDirectories(textBox1.Text);

int boyut = klasorler.Length;

if (boyut != 0)

{

foreach (string aktifklasor in klasorler)

{

item = new ListViewItem(Path.GetFileName(aktifklasor), 1);

item.SubItems.Add("");

item.SubItems.Add("Klasör");

listView1.Items.Add(item);

}

String[] dosyalar = Directory.GetFiles(textBox1.Text);

boyut = dosyalar.Length;

foreach (string aktifdosya in dosyalar)

{

item = new ListViewItem(Path.GetFileName(aktifdosya), 0);

FileInfo dosya\_bilgisi = new FileInfo(aktifdosya);

item.SubItems.Add(Convert.ToString(dosya\_bilgisi.Length));

item.SubItems.Add("Dosya");

listView1.Items.Add(item);

}

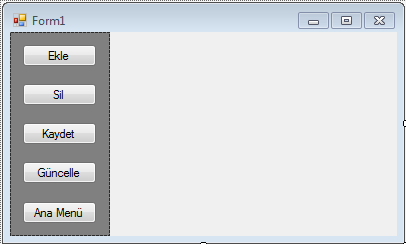
}

listView1.View = View.Details;

}

**PANEL\_KONTROLÜ**

**Panel\_OtomatikGizle**

****

int konum;

private void panel1\_MouseMove(object sender, MouseEventArgs e)

{

panel1.Left = konum;

}

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

{

konum = panel1.Left;

panel1.Left = panel1.Left - 90;

//panel1.Dock = DockStyle.Right

}

private void panel1\_MouseLeave(object sender, EventArgs e)

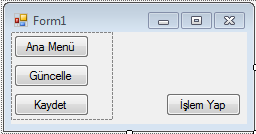
{

panel1.Left = konum - 90;

}

**FLOWLAYOUTPANEL\_KONTROLU**

**FlowLayoutPanel\_Kontrolu**

****

private void btnislem\_Click(object sender, EventArgs e){

//flowLayoutPanel1.FlowDirection = FlowDirection.TopDown;

//flowLayoutPanel1.WrapContents = true;

flowLayoutPanel1.AutoScroll = true;

}

**TABLELAYOUTPANEL\_KONTROLU**

**TableLayoutPanel\_Kontrolu**

private void Form1\_Paint(object sender, PaintEventArgs e)

{

TableLayoutPanel tlp = new TableLayoutPanel();

tlp.Location = new System.Drawing.Point(10,10);

tlp.Name = "TableLayoutPanel1";

tlp.Size = new System.Drawing.Size(200, 100);

tlp.BackColor = Color.DarkGray;

tlp.ColumnCount = 2;

tlp.RowCount = 2;

//tlp.ColumnStyles.Add(new RowStyle(SizeType.AutoSize));

tlp.ColumnStyles.Add(new ColumnStyle(SizeType.Absolute, 90F));

tlp.ColumnStyles.Add(new ColumnStyle(SizeType.Absolute, 90F));

tlp.RowStyles.Add(new RowStyle(SizeType.Percent, 50F));

tlp.RowStyles.Add(new RowStyle(SizeType.Percent, 50F));

Label label1 = new Label();

label1.Text = "Kullanıcı Adı:";

Label label2 = new Label();

label2.Text = "Şifre:";

TextBox textBox1 = new TextBox();

TextBox textBox2 = new TextBox();

tlp.Controls.Add(label1, 0, 0);

tlp.Controls.Add(textBox1, 1, 0);

tlp.Controls.Add(label2, 0, 1);

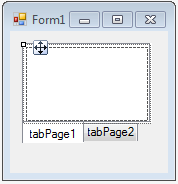
tlp.Controls.Add(textBox2, 1, 1);

this.Controls.Add(tlp);

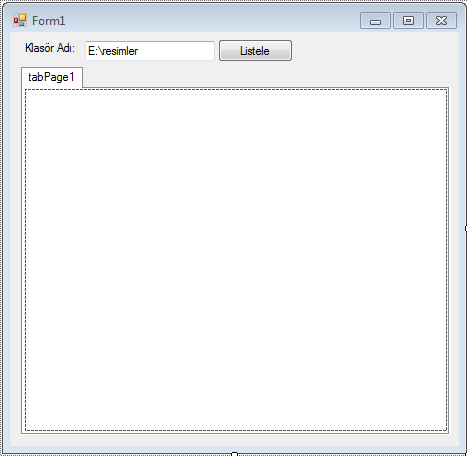
}

**TABCONTROL\_NESNESİ**

**TabControl\_Kontrolu**

****

**TabControl\_ResimListeleme**

****

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

{

int i = 0;

string[] klasorler = Directory.GetDirectories(textBox1.Text);

if (klasorler.Length != 0)

{

foreach (string aktifklasor in klasorler)

{

i++;

TabPage tp = new TabPage("tp" + i);

tabControl1.TabPages.Add(tp);

ListView lstv = new ListView();

lstv.Size = new System.Drawing.Size(420, 330);

lstv.View = View.LargeIcon;

ImageList imgl = new ImageList();

imgl.ImageSize = new Size(128, 128);

tabControl1.TabPages[i].Text = Path.GetFileName(aktifklasor);

string[] dosyalar = Directory.GetFiles(aktifklasor, "\*.jpg", SearchOption.AllDirectories);

imgl.Images.Clear();

foreach (string aktifdosya in dosyalar)

{

imgl.Images.Add(Image.FromFile(aktifdosya));

}

for (int j = 0; j < imgl.Images.Count; j++)

{

ListViewItem item = new ListViewItem();

item.ImageIndex = j;

lstv.Items.Add(item);

}

lstv.LargeImageList = imgl;

tabControl1.TabPages[i].Controls.Add(lstv);

}

}

tabControl1.TabPages.RemoveAt(0);

}

**TİMER KONTROLÜ**

**timer\_kontrolu\_saat**



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;

namespace timer\_kontrolu\_saat

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void Timer1\_Tick(object sender, EventArgs e)

{

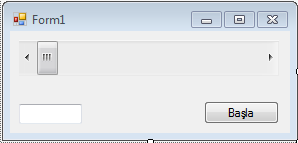
Label1.Text = string.Format("{0:hh:mm:ss}", DateTime.Now);

}

}

}

**timer\_kontrolu\_scrollbar**

****

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;

namespace timer\_kontrolu\_scrollbar

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

Boolean son=false;

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

{

timer1.Interval = hScrollBar1.Value;

}

private void hScrollBar1\_Scroll(object sender, ScrollEventArgs e)

{

timer1.Interval = hScrollBar1.Value;

if (hScrollBar1.Value == hScrollBar1.Maximum) son = true;

}

private void timer1\_Tick(object sender, EventArgs e)

{

if (son == false)

{

textBox1.Text = hScrollBar1.Value.ToString();

if (hScrollBar1.Value == 100)

{

son = true;

return;

}

hScrollBar1.Value = hScrollBar1.Value + 1;

}

else

{

textBox1.Text = hScrollBar1.Value.ToString();

if (hScrollBar1.Value == 1)

{

son = false;

return;

}

hScrollBar1.Value = hScrollBar1.Value - 1;

}

}

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

{

if (button1.Text == "Başla")

{

button1.Text = "Durdur";

timer1.Enabled = true;

}

else

{

button1.Text = "Başla";

timer1.Enabled = false;

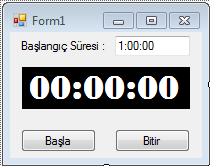
}

}

}

}

**timer\_kontrolu\_kronometre**



using System;

using System.Collections.Generic;

using System.Windows.Forms;

namespace timer\_kontrolu\_kronometre

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

int saat, dakika, saniye;

DateTime zaman, kalınanzaman;

Boolean duraklama = false;

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

{

if (btnBasla.Text == "Başla")

{

btnBasla.Text = "Dur";

if (duraklama == false)

{

zaman = Convert.ToDateTime(textBox1.Text);

saat = zaman.Hour;

dakika = zaman.Minute;

saniye = zaman.Second;

}

else

{

zaman = kalınanzaman;

saat = zaman.Hour;

dakika = zaman.Minute;

saniye = zaman.Second;

}

timer1.Start();

}

else

{

zaman = Convert.ToDateTime(label1.Text);

btnBasla.Text = "Başla";

timer1.Stop();

duraklama = true;

kalınanzaman = Convert.ToDateTime(label1.Text);

}

}

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

{

timer1.Enabled = false;

label1.Text = textBox1.Text;

}

private void timer1\_Tick(object sender, EventArgs e)

{

label1.Text = ((Convert.ToString(saat) + ":") + (Convert.ToString(dakika) + ":") + Convert.ToString(saniye));

if ((saniye == 0) && (dakika > 0))

{

dakika = dakika - 1;

saniye = 59;

}

else if ((saniye == 0) && (dakika == 0) && (saat > 0))

{

saniye = 59;

dakika = 59;

saat = saat - 1;

}

if ((dakika == 0) && (saat > 0))

{

dakika = 59;

saat = saat - 1;

}

if ((saat == 0) && (dakika == 0) && (saniye == 0))

{

timer1.Enabled = true;

MessageBox.Show("Süre Bitti");

}

saniye = saniye - 1;

}

}

}

**timer\_kontrolu\_baglantılı**

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;

namespace timer\_kontrolu\_baglantılı

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

byte i;

private void timer1\_Tick(object sender, EventArgs e)

{

i++;

if (i == 5)

{

MessageBox.Show("Timer1 saymayı bitirdi. Şimdi sıra Timer2 de.");

timer1.Enabled = false;

timer2.Enabled = true;

i = 0;

}

}

private void timer2\_Tick(object sender, EventArgs e)

{

i++;

if (i == 3)

{

MessageBox.Show("Timer2 de saymayı bitirdi.");

timer2.Enabled = false;

}

}

}

}

**timer\_console**

using System;

using System.Text;

using System.Timers;

namespace timer\_console

{

class Program

{

static void Main(string[] args)

{

Timer tmr = new Timer();

tmr.Elapsed += tmr\_Elapsed;

tmr.Interval = 1000;

tmr.Start();

Console.ReadLine();

}

static void tmr\_Elapsed(object sender, ElapsedEventArgs e)

{

Console.Clear();

Console.Write(string.Format("{0:hh:mm:ss}", DateTime.Now));

}

}

}

**timer\_Console\_Threading**

using System;

using System.Text;

using System.Threading;

namespace timer\_Console\_Threading

{

class Program

{

static void Main(string[] args)

{

TimerCallback tmrCb = new TimerCallback(Saat);

Timer tmr = new Timer(tmrCb, null, 0, 1000);

Console.ReadLine();

tmr.Dispose();

}

private static void Saat(Object state)

{

Console.Clear();

Console.Write(string.Format("{0:hh:mm:ss}", DateTime.Now));

}

}

}

**StopWatch\_Olcum**

static void Main(string[] args)

{

int[] toplam = new int[10000000];

Random rnd = new Random();

int rnd\_sayi = 0;

Stopwatch sw = new Stopwatch();

sw.Start();

for (int i = 0; i < 10000000; i++)

{

rnd\_sayi=rnd.Next(1,100);

if (rnd\_sayi < 50)

toplam[i] = rnd\_sayi;

}

sw.Stop();

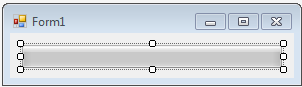
Console.WriteLine("Toplam süre: {0}",sw.Elapsed);

Console.ReadLine();

}

**PROGRESSBAR\_KONTROLU**

**ProgressBar\_Kontrolu**

****

private void timer1\_Tick(object sender, EventArgs e)

{

if (progressBar1.Value == progressBar1.Maximum)

progressBar1.Value = 1;

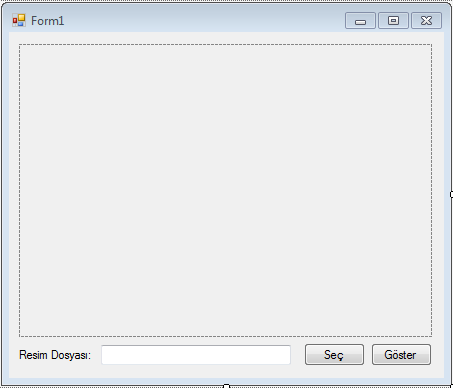
else

progressBar1.Value = progressBar1.Value + 1;

}

**PİCTUREBOX KONTROLÜ METODLARI**

**PictureBox\_ResimGoster**

****

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

{

openFileDialog1.FileName = "dosya seç";

openFileDialog1.Filter = "jpeg dosyaları (\*.jpeg) |\*.jpg|bmp dosyaları (\*.bmp)|\*.bmp|gif dosyaları (\*.gif)|\*.gif";

openFileDialog1.Title = "Lütfen Bir Resim Dosyası Seçin";

openFileDialog1.InitialDirectory =@"C:\Users\user\Pictures";

}

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

{

if (openFileDialog1.ShowDialog() == DialogResult.OK)

pictureBox1.Image = new Bitmap(openFileDialog1.OpenFile());

}

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

{

pictureBox1.Load(textBox1.Text);

}

**MENUSTRIP KONTROLU**

**Enabled**

**Checked**

**CheckOnClick**

**CONTENTMENUSTRIP KONTROLU**

**MenuKontrolleri\_Editor**

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

{

textBox1.Font = new Font(textBox1.Font.Name,

textBox1.Font.Size, textBox1.Font.Style ^ FontStyle.Bold);

}

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

{

textBox1.Font = new Font(textBox1.Font.Name,

textBox1.Font.Size, textBox1.Font.Style ^ FontStyle.Italic);

}

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

{

textBox1.Font = new Font(textBox1.Font.Name,

textBox1.Font.Size, textBox1.Font.Style ^ FontStyle.Underline);

}

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

{

if (textBox1.SelectionLength > 0)

textBox1.Cut();

else

MessageBox.Show("Seçili Alan Yok");

}

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

{

if (textBox1.SelectionLength > 0)

textBox1.Copy();

else

MessageBox.Show("Seçili Alan Yok");

}

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

{

textBox1.Paste();

}

**NOTİFYICON\_KONTROLU**

**NotifyIcon\_Kontrolu**

private void Form1\_FormClosing(object sender, FormClosingEventArgs e)

{

if (e.CloseReason == CloseReason.UserClosing)

{

e.Cancel = true;

this.Hide();

notifyIcon1.Visible = true;

notifyIcon1.ShowBalloonTip(1000);

}

}

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

{

notifyIcon1.ContextMenuStrip = contextMenuStrip1;

notifyIcon1.BalloonTipText = "Uygulama Hala Çalışıyor...";

notifyIcon1.BalloonTipTitle = "Örnek Uygulama";

notifyIcon1.BalloonTipIcon = ToolTipIcon.Info;

}

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

{

if (this.WindowState == FormWindowState.Minimized)

{

this.Hide();

notifyIcon1.Visible = true;

notifyIcon1.ShowBalloonTip(1000);

}

}

private void notifyIcon1\_DoubleClick(object sender, EventArgs e)

{

this.Show();

notifyIcon1.Visible = false;

}

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

{

this.Show();

this.WindowState = FormWindowState.Normal;

notifyIcon1.Visible = false;

}

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

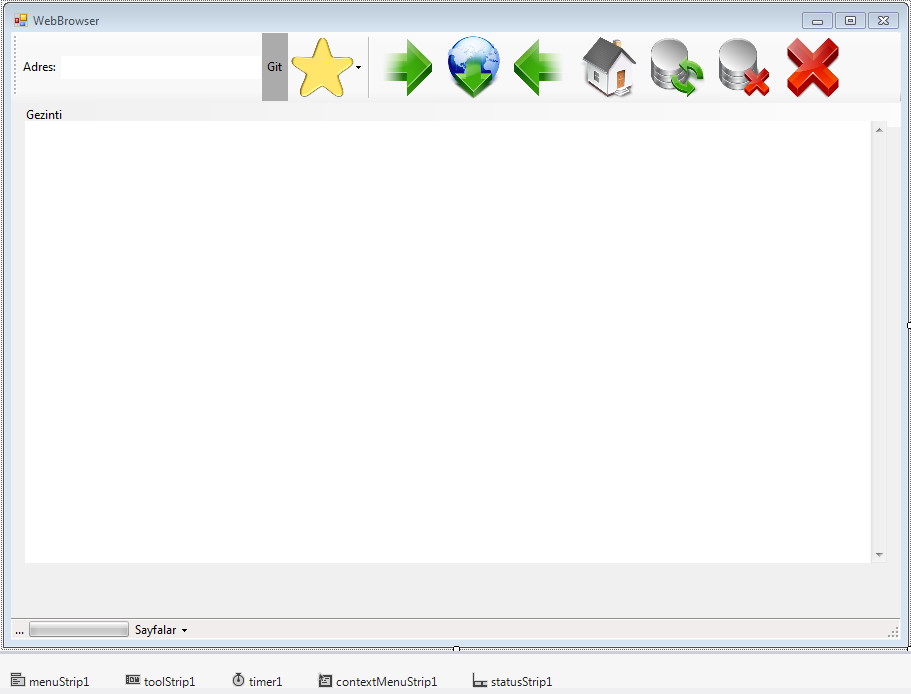
{

Application.Exit();

}

TOOLSTRIP KONTROLÜ

**webBrowser\_AracCubukları\_Ornegi**

****

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;

namespace webBrowser\_AracCubukları\_Ornegi

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

byte i;

private void tbSayfalar\_DropDownItemClicked(object sender, ToolStripItemClickedEventArgs e)

{

webBrowser1.Navigate(e.ClickedItem.Text);

}

private void tsSayfalar\_DropDownItemClicked(object sender, ToolStripItemClickedEventArgs e)

{

webBrowser1.Navigate(e.ClickedItem.Text);

}

private void webBrowser1\_DocumentCompleted(object sender, WebBrowserDocumentCompletedEventArgs e)

{

txtadres.Text = e.Url.ToString();

timer1.Stop();

toolStripStatusLabel1.Text = txtadres.Text + " sayfası yüklendi";

toolStripProgressBar1.Value = 0;

toolStripProgressBar1.Visible = false;

//tsSayfalar.Visible = true;

}

private void webBrowser1\_Navigating(object sender, WebBrowserNavigatingEventArgs e)

{

toolStripLabel1.Visible = true;

toolStripProgressBar1.Visible = true;

toolStripStatusLabel1.Text = txtadres.Text +

" adresi yüklenmeye başladı";

toolStripProgressBar1.Value = 0;

timer1.Start();

}

private void timer1\_Tick(object sender, EventArgs e)

{

toolStripStatusLabel1.Text = txtadres.Text + " Sayfa Yükleniyor ";

if (toolStripProgressBar1.Value == 100)

toolStripProgressBar1.Value = 0;

else

toolStripProgressBar1.Value = toolStripProgressBar1.Value + 10;

}

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

{

toolStripProgressBar1.Visible = false;

}

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

{

webBrowser1.GoBack();

}

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

{

webBrowser1.GoForward();

}

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

{

webBrowser1.GoHome();

}

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

{

webBrowser1.GoSearch();

}

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

{

webBrowser1.Stop();

}

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

{

Application.Exit();

}

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

{

i++;

if (!(txtadres.Text.StartsWith("http://")))

txtadres.Text = txtadres.Text.Insert(0, "http://");

webBrowser1.Navigate(txtadres.Text);

Cursor.Current = Cursors.Default;

ToolStripMenuItem yenimenu = new ToolStripMenuItem(String.Format(txtadres.Text, i));

tbSayfalar.DropDownItems.Add(yenimenu);

tsSayfalar.DropDownItems.Add(yenimenu);

}

}

}

**TOOLTİP KONTROLÜ METODLARI**

**toolTip\_Kontrolu**

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;

namespace toolTip\_Kontrolu

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

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

{

toolTip1.AutoPopDelay = 5000;

toolTip1.InitialDelay = 1000;

toolTip1.ReshowDelay = 500;

toolTip1.ToolTipIcon = ToolTipIcon.Warning;

toolTip1.ToolTipTitle = "Uyarı";

toolTip1.IsBalloon = true;

toolTip1.SetToolTip(txtAdSoyad,

"Bu alana yalnızca string bilgi giriniz!");

toolTip1.SetToolTip(txtNumara,

"Bu alana yalnızca rakam giriniz!");

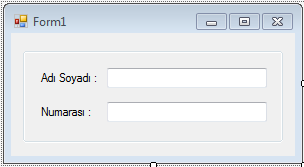
}

}

}

**ERRORPROVİDER KONTROLÜ**

**errorProvider\_Kontrolu**

****

Boolean rakamvar;

int rakam;

private void txtAdSoyad\_Validating(object sender, CancelEventArgs e)

{

errorProvider1.Clear();

rakamvar = int.TryParse(txtAdSoyad.Text, out rakam);

if (txtAdSoyad.Text == "")

{

errorProvider1.SetError(txtAdSoyad, "Ad Soyad boş bırakılamaz");

}

else

{

if (rakamvar == true)

errorProvider1.SetError(txtAdSoyad, "Ad Soyad alanına rakam girilemez");

}

}

private void txtNumara\_Validating(object sender, CancelEventArgs e)

{

errorProvider1.Clear();

rakamvar = int.TryParse(txtNumara.Text, out rakam);

if (txtNumara.Text == "")

{

errorProvider1.SetError(txtNumara, "Numara boş bırakılamaz");

}

else

{

if (rakamvar == false)

errorProvider1.SetError(txtNumara, "Numara alanına harf girilemez");

}

}

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

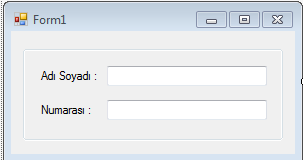
{

rakamvar = false;

}

**HELPPROVİDER KONTROLÜ METODLARI**

**helpProvider\_SetHelpString**

****

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

{

this.HelpButton = true;

this.MaximizeBox = false;

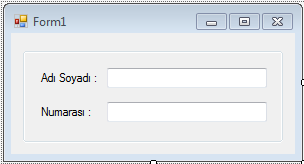
this.MinimizeBox = false;

helpProvider1.SetHelpString(txtAdSoyad,

"Bu kontrol ile ilgili yardim mevcut");

}

**helpProvider\_Kontrolu**

****

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

{

helpProvider1.HelpNamespace = Application.StartupPath + "\\yardim.chm";

helpProvider1.SetHelpKeyword(txtAdSoyad, "Numara");

helpProvider1.SetHelpNavigator(txtAdSoyad,

HelpNavigator.AssociateIndex);

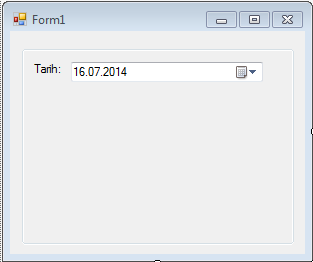
helpProvider1.SetShowHelp(txtAdSoyad, true);

}

//////////////////////////////////////////////////////400

**DATETİMEPİCKER\_KONTROLU**

**datetimePicker\_Kontrolu**

****

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

{

dateTimePicker1.Format = DateTimePickerFormat.Custom;

dateTimePicker1.CustomFormat = "hh:mm:ss – dd MMMM yyyy";

}

private void dateTimePicker1\_ValueChanged(object sender, EventArgs e)

{

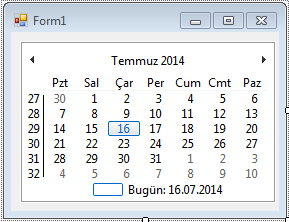
MessageBox.Show("Seçili Tarih : " +

dateTimePicker1.Value.ToLongDateString());

}

**MONTHCALENDAR\_KONTROLU**

**monthCalendar\_Kontrolu**

****

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

{

monthCalendar1.ShowToday = false;

monthCalendar1.ShowTodayCircle = false;

monthCalendar1.AddBoldedDate(DateTime.Parse("2012/10/4"));

monthCalendar1.AddBoldedDate(DateTime.Parse("2012/10/5"));

monthCalendar1.UpdateBoldedDates();

}

private void monthCalendar1\_DateSelected(object sender, DateRangeEventArgs e)

{

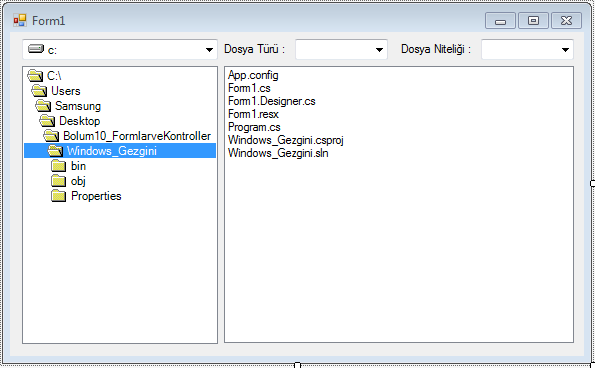
MessageBox.Show("Seçili Tarih: " +

monthCalendar1.SelectionRange.Start.ToString("dd MMMM yyyy"));

}

**FİLELİSTBOX KONTROLUNUN ÖZELLİKLERİ**

**Windows\_Gezgini**

****

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;

namespace Windows\_Gezgini

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void driveListBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

dirListBox1.Path = driveListBox1.Drive;

}

private void dirListBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

fileListBox1.Path = dirListBox1.Path;

}

private void fileListBox1\_DoubleClick(object sender, EventArgs e)

{

int n = fileListBox1.Path.IndexOf("\\");

if (fileListBox1.Path == fileListBox1.Path.Substring(n, 1))

System.Diagnostics.Process.Start

(fileListBox1.Path + fileListBox1.FileName);

else

System.Diagnostics.Process.Start

(fileListBox1.Path + "\\" + fileListBox1.FileName);

}

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

{

cmbDosyaTuru.SelectedIndex = 0;

cmbNitelik.SelectedIndex = 0;

fileListBox1.Pattern = cmbDosyaTuru.Text;

fileListBox1.Archive = true;

}

private void cmbNitelik\_SelectedIndexChanged(object sender, EventArgs e)

{

if (cmbNitelik.Text == "Arşiv Dosyaları")

fileListBox1.Archive = true;

if (cmbNitelik.Text == "Gizli Dosyalar")

fileListBox1.Hidden = true;

if (cmbNitelik.Text == "Sistem Dosyaları")

fileListBox1.System = true;

if (cmbNitelik.Text == "Yalnız Okunabilir Dosyalar")

fileListBox1.ReadOnly = true;

fileListBox1.Refresh();

}

private void cmbDosyaTuru\_SelectedIndexChanged(object sender, EventArgs e)

{

fileListBox1.Pattern = cmbDosyaTuru.Text;

}

}

}