

Контрол PictureBox

Представя изображение от тип **Image**.

Свойства

Image изображение;

SizeMode преоразмеряване и позициониране на изображението: **Normal**, **StretchImage**, **AutoSize** и **CenterImage**.

Събития

Click <L> в контрола; подразбиращо се събитие при <L?> в изглед **Designer**.

Метод **Image.FromFile** създава **Image** обект от файла **filename**.

```
public static Image FromFile( string filename );
```

Пример: Изобразяване на албум със снимки



```
using System;
using System.Drawing;
using System.Collections;
using System.ComponentModel;
using System.Windows.Forms;
using System.Data;
using System.IO;

namespace PictureBox
{
    public class Form1 : System.Windows.Forms.Form
    {
        private System.Windows.Forms.PictureBox imagePictureBox;
        private System.Windows.Forms.Label promptLabel;
        private int imageNum = -1;
        private System.ComponentModel.Container components = null;

        public Form1()
        {
            InitializeComponent();
        }

        protected override void Dispose( bool disposing ) { ... }
    }
}
```

```
#region Windows Form Designer generated code
private void InitializeComponent()
{
    ...
    this.imagePictureBox = new System.Windows.Forms.PictureBox();
    this.promptLabel = new System.Windows.Forms.Label();
    // imagePictureBox
    this.imagePictureBox.Name = "imagePictureBox";
    this.imagePictureBox.SizeMode =
        System.Windows.Forms.PictureBoxSizeMode.StretchImage;
    this.imagePictureBox.Click +=
        new System.EventHandler(this.imagePictureBox_Click);
    // promptLabel
    this.promptLabel.ForeColor = System.Drawing.Color.Red;
    this.promptLabel.Name = "promptLabel";
    this.promptLabel.Text = "Натиснете бутона на мишката, " +
        "за да видите снимките от албума";
    this.promptLabel.TextAlign =
        System.Drawing.ContentAlignment.MiddleCenter;
}
#endregion
```

```
// Form1
this.Controls.Add(this.promptLabel);
this.Controls.Add(this.imagePictureBox);
this.Name = "Form1";
this.Text = "35 години ФКСУ";
}
#endregion

[STAThread]
static void Main()
{
    Application.Run(new Form1());
}

private void imagePictureBox_Click(object sender, System.EventArgs e)
{
    imageNum = (imageNum + 1) % 3;
    imagePictureBox.Image = Image.FromFile(
        Directory.GetCurrentDirectory() + "\\fksu" + imageNum + ".tif");
}
}
```

Работа с менюта

Меню – групира

- а) **меню-елементи (команди):**
 - с клавиш за достъп (**Alt**+клавиш);
 - с маркер за отметка;
- б) **подменюта.**

Видове менюта

- а) с меню-елементи;
- б) контекстно меню (<R>).

Slide 1

S1

ag

SB; 19.5.2004 г.

Контрол **MainMenu**

Изобразява меню с меню-елементи. Менюто се свързва с формата, като неговото име се присвоява на свойството **Menu** на формата. Методът **Menu.CloneMenu** създава копие на структурата на менюто – използва се за модифициране на структурата на менюто.

Свойства

MenuItems списък от меню-елементите, които съдържа.

Контрол **ContextMenu**

Изобразява меню с често използвани елементи при натискане десния бутон на мишката. Свързва се с друг контрол, като неговото име се присвоява на свойството **ContextMenu** на другия контрол.

Свойства

SourceControl дава контрола, който изобразява контекстното меню.

Клас **MenuItem**

Изобразява елемент от **MainMenu** или **ContextMenu**.

Свойства

(Name) име на елемент от меню;
Checked **True** – изобразява маркер за проверка, **False** – скрива маркера.
DefaultItem **True** – подразбира се елемент, изобразява се с удебелен шрифт; <L²> меню-родител се изпълнява командата, съответстваща на подразбиращия се елемент;
Enabled определя достъпност (**True**) /недостъпност (**False**) – елементът се изобразява в светло сиво;

Index позиция на елемента в менюто-родител;

MenuItems списък на елементите на подменюто за даден меню-елемент;

RadioCheck изобразява радио бутон, когато е **True** и **Checked** е **True**;

Shortcut определя клавишна комбинация за избор на меню-команда;

Text изобразява текст за меню-елемента; чрез символа & определя клавиш за достъп;

Visible определя дали елементът ще бъде изобразен в менюто; по-общо е да се използва свойството **Enabled**.

Събития

Click <L> меню-елемент или клавишна комбинация; подразбира се събитие при <L²> в изглед **Designer**.

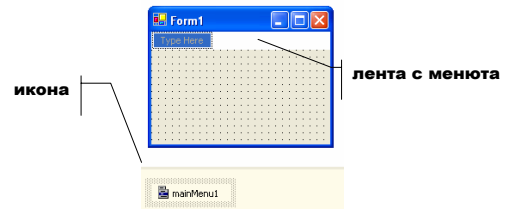
Popup изобразява меню при динамично установяване свойствата на меню-елемент, зависещи от текущия контекст и условия;

PopupSelect изобразява меню-елемент, който все още не е натиснат (напр. при преминаване над него с клавишите стрелки).

Създаване на меню чрез **Designer**:

1. Създаване на главно меню

Toolbox ⇒ <L²> **MainMenu**



2. Установяване свойства на **mainMenu1**

(Name) име_на_клас_главно_меню (тип **MainMenu**)

3. Добавяне на менюта към главното меню

<L> **Type Here** ⇒ текст за `име_на_меню`



4. Свойства на меню

(Name) `име_на_клас_меню` (тип `MenuItem`)
 Text `име_на_меню`
 (след името се поставят ... за поява на диалогов прозорец)

5. Добавяне на подменюта към меню

<L> `име_на_меню`
 <L> **Type Here** ⇒ текст за `име_на_подменю`

6. Свойства подменю

(Name) `име_на_клас_подменю` (тип `MenuItem`)
 Text `име_на_подменю`

7. Клавиш за достъп

& пред подчертания символ

&Файл Файл (Alt+Ф)

Свойство `Shortcut` определя клавиша (Alt по подразбиране).

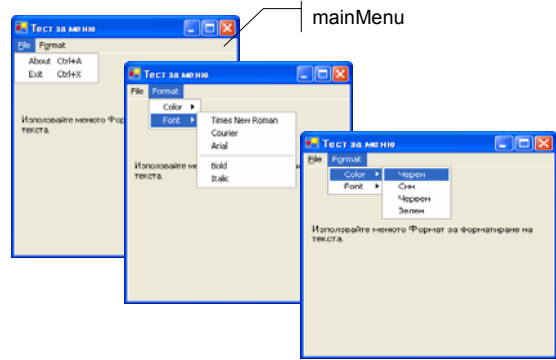
8. Добавяне на разделител

<R> `меню-елемент` ⇒ **Insert Separator**
 или – за текст на меню-елемент

9. Премахване на меню-елемент

<L> `меню-елемент` ⇒ **<Delete>**

Пример: Меню-лента с две менюта: File и Format.



```
using System;
using System.Drawing;
using System.Collections;
using System.ComponentModel;
using System.Windows.Forms;
using System.Data;
```

```
namespace Menus
{
    public class MenuTest : System.Windows.Forms.Form
    {
        private System.Windows.Forms.MainMenu mainMenu;
        private System.Windows.Forms.MenuItem fileMenuItem;
        private System.Windows.Forms.MenuItem aboutMenuItem;
        private System.Windows.Forms.MenuItem exitMenuItem;
        private System.Windows.Forms.MenuItem formatMenuItem;
        private System.Windows.Forms.MenuItem colorMenuItem;
        private System.Windows.Forms.MenuItem blackMenuItem;
        private System.Windows.Forms.MenuItem blueMenuItem;
        private System.Windows.Forms.MenuItem redMenuItem;
        private System.Windows.Forms.MenuItem greenMenuItem;
```

```
private System.Windows.Forms.MenuItem fontMenuItem;
private System.Windows.Forms.MenuItem timesMenuItem;
private System.Windows.Forms.MenuItem courierMenuItem;
private System.Windows.Forms.MenuItem arialMenuItem;
private System.Windows.Forms.MenuItem separatorMenuItem;
private System.Windows.Forms.MenuItem boldMenuItem;
private System.Windows.Forms.MenuItem italicMenuItem;
private System.Windows.Forms.Label displayLabel;
private System.ComponentModel.Container components = null;
public MenuTest()
{
    InitializeComponent();
}
protected override void Dispose( bool disposing ) { ... }
```

```
#region Windows Form Designer generated code
private void InitializeComponent()
{
    ...
    this.mainMenu = new System.Windows.Forms.MainMenu();
    this.fileMenuItem = new System.Windows.Forms.MenuItem();
    this.aboutMenuItem = new System.Windows.Forms.MenuItem();
    this.exitMenuItem = new System.Windows.Forms.MenuItem();
    this.formatMenuItem = new System.Windows.Forms.MenuItem();
    this.colorMenuItem = new System.Windows.Forms.MenuItem();
    this.blackMenuItem = new System.Windows.Forms.MenuItem();
    this.blueMenuItem = new System.Windows.Forms.MenuItem();
    this.redMenuItem = new System.Windows.Forms.MenuItem();
    this.greenMenuItem = new System.Windows.Forms.MenuItem();
    this.fontMenuItem = new System.Windows.Forms.MenuItem();
    this.timesMenuItem = new System.Windows.Forms.MenuItem();
    this.courierMenuItem = new System.Windows.Forms.MenuItem();
    this.arialMenuItem = new System.Windows.Forms.MenuItem();
    this.separatorMenuItem = new System.Windows.Forms.MenuItem();
    this.boldMenuItem = new System.Windows.Forms.MenuItem();
    this.italicMenuItem = new System.Windows.Forms.MenuItem();
    this.displayLabel = new System.Windows.Forms.Label();
    this.SuspendLayout();
```

```
// mainMenu
this.mainMenu.MenuItems.AddRange(
    new System.Windows.Forms.MenuItem[] {
        this.fileMenuItem, this.formatMenuItem});
// fileMenuItem
this.fileMenuItem.Index = 0;
this.fileMenuItem.MenuItems.AddRange(
    new System.Windows.Forms.MenuItem[] {
        this.aboutMenuItem, this.exitMenuItem});
this.fileMenuItem.Text = "&File";
// aboutMenuItem
this.aboutMenuItem.Shortcut = System.Windows.Forms.Shortcut.CtrlA;
this.aboutMenuItem.Text = "&About";
this.aboutMenuItem.Click +=
    new System.EventHandler(this.aboutMenuItem_Click);
// exitMenuItem
this.exitMenuItem.Shortcut = System.Windows.Forms.Shortcut.CtrlX;
this.exitMenuItem.Text = "E&xit";
this.exitMenuItem.Click +=
    new System.EventHandler(this.exitMenuItem_Click);
```

```
// formatMenuItem
this.formatMenuItem.MenuItems.AddRange(
    new System.Windows.Forms.MenuItem[] {
        this.colorMenuItem, this.fontMenuItem});
this.formatMenuItem.Text = "F&ormat";
// colorMenuItem
this.colorMenuItem.MenuItems.AddRange(
    new System.Windows.Forms.MenuItem[] {
        this.blackMenuItem, this.blueMenuItem,
        this.redMenuItem, this.greenMenuItem});
this.colorMenuItem.Text = "Color";
// blackMenuItem
this.blackMenuItem.RadioCheck = true;
this.blackMenuItem.Text = "Black";
this.blackMenuItem.Click +=
    new System.EventHandler(this.blackMenuItem_Click);
// blueMenuItem
this.blueMenuItem.RadioCheck = true;
this.blueMenuItem.Text = "Blue";
this.blueMenuItem.Click +=
    new System.EventHandler(this.blueMenuItem_Click);
```

```
// redMenuItem
this.redMenuItem.RadioCheck = true;
this.redMenuItem.Text = "Red";
this.redMenuItem.Click +=
    new System.EventHandler(this.redMenuItem_Click);
// greenMenuItem
this.greenMenuItem.RadioCheck = true;
this.greenMenuItem.Text = "Green";
this.greenMenuItem.Click +=
    new System.EventHandler(this.greenMenuItem_Click);
// fontMenuItem
this.fontMenuItem.MenuItems.AddRange(
    new System.Windows.Forms.MenuItem[] {
        this.timesMenuItem, this.courierMenuItem, this.arialMenuItem,
        this.separatorMenuItem, this.boldMenuItem, this.italicMenuItem});
this.fontMenuItem.Text = "Font";
// timesMenuItem
this.timesMenuItem.Text = "Times New Roman";
this.timesMenuItem.Click +=
    new System.EventHandler(this.timesMenuItem_Click);
```

```
// courierMenuItem
this.courierMenuItem.Text = "Courier";
this.courierMenuItem.Click +=
    new System.EventHandler(this.courierMenuItem_Click);
// arialMenuItem
this.arialMenuItem.Text = "Arial";
this.arialMenuItem.Click +=
    new System.EventHandler(this.arialMenuItem_Click);
// separatorMenuItem
this.separatorMenuItem.Text = "-";
// boldMenuItem
this.boldMenuItem.Text = "Bold";
this.boldMenuItem.Click +=
    new System.EventHandler(this.boldMenuItem_Click);
// italicMenuItem
this.italicMenuItem.Text = "Italic";
this.italicMenuItem.Click +=
    new System.EventHandler(this.italicMenuItem_Click);
```

```
// displayLabel
this.displayLabel.Location = new System.Drawing.Point(8, 80);
this.displayLabel.Name = "displayLabel";
this.displayLabel.Text = "Използвайте менюто Формат " +
    " за форматиране на текста.";
// MenuTest
this.Controls.Add(this.displayLabel);
this.Menu = this.mainMenu;
this.Name = "MenuTest";
this.Text = "Тест за меню";
this.ResumeLayout(false);
}
#endregion
[STAThread]
static void Main()
{
    Application.Run(new MenuTest());
}
```

```
private void aboutMenuItem_Click(object sender, System.EventArgs e)
{
    MessageBox.Show("Пример за използване на меню", "Относно",
        MessageBoxButtons.OK, MessageBoxIcon.Information);
}
private void exitMenuItem_Click(object sender, System.EventArgs e)
{
    Application.Exit();
}
private void ClearColor()
{
    blackMenuItem.Checked=false;
    blueMenuItem.Checked=false;
    redMenuItem.Checked=false;
    greenMenuItem.Checked=false;
}
private void blackMenuItem_Click(object sender, System.EventArgs e)
{
    ClearColor();
    displayLabel.ForeColor=Color.Black;
    blackMenuItem.Checked=true;
}
```

```
private void blueMenuItem_Click(object sender, System.EventArgs e)
{
    ClearColor();
    displayLabel.ForeColor=Color.Blue;
    blueMenuItem.Checked=true;
}
private void redMenuItem_Click(object sender, System.EventArgs e)
{
    ClearColor();
    displayLabel.ForeColor=Color.Red;
    redMenuItem.Checked=true;
}
private void greenMenuItem_Click(object sender, System.EventArgs e)
{
    ClearColor();
    displayLabel.ForeColor=Color.Green;
    greenMenuItem.Checked=true;
}
```

```
private void ClearFont()
{
    timesMenuItem.Checked=false;
    courierMenuItem.Checked=false;
    arialMenuItem.Checked=false;
}
private void timesMenuItem_Click(object sender, System.EventArgs e)
{
    ClearFont();
    displayLabel.Font = new Font("Times New Roman", 14,
                                displayLabel.Font.Style);
    timesMenuItem.Checked=true;
}
private void courierMenuItem_Click(object sender, System.EventArgs e)
{
    ClearFont();
    displayLabel.Font=new Font("Courier", 14, displayLabel.Font.Style);
    courierMenuItem.Checked=true;
}
```

```
private void arialMenuItem_Click(object sender, System.EventArgs e)
{
    ClearFont();
    displayLabel.Font=new Font("Arial", 14, displayLabel.Font.Style);
    arialMenuItem.Checked=true;
}
private void boldMenuItem_Click(object sender, System.EventArgs e)
{
    displayLabel.Font = new Font(displayLabel.Font.FontFamily, 14,
                                displayLabel.Font.Style^FontStyle.Bold);
    boldMenuItem.Checked=true;
}
private void italicMenuItem_Click(object sender, System.EventArgs e)
{
    displayLabel.Font = new Font(displayLabel.Font.FontFamily, 14,
                                displayLabel.Font.Style^FontStyle.Italic);
    italicMenuItem.Checked=true;
}
}
```

Пример: Допълваме формата за записване на студенти в избираеми дисциплини с меню Файл и контекстни менюта за текстовите контроли (Изчистване на текста) и за формата (Запазване на студент и Изчисти).

```
// Добавяне на менюта във формата-родител
private System.Windows.Forms.MainMenu mainMenu;
private System.Windows.Forms.MenuItem fileItem;
private System.Windows.Forms.MenuItem newItem;
private System.Windows.Forms.MenuItem openItem;
private System.Windows.Forms.MenuItem saveItem;
private System.Windows.Forms.MenuItem menuItem5;
private System.Windows.Forms.MenuItem printItem;
private System.Windows.Forms.MenuItem menuItem7;
private System.Windows.Forms.MenuItem exitItem;
// Инициализиране на менютата
this.mainMenu = new System.Windows.Forms.MainMenu();
this.fileItem = new System.Windows.Forms.MenuItem();
this.newItem = new System.Windows.Forms.MenuItem();
this.openItem = new System.Windows.Forms.MenuItem();
this.saveItem = new System.Windows.Forms.MenuItem();
this.menuItem5 = new System.Windows.Forms.MenuItem();
this.printItem = new System.Windows.Forms.MenuItem();
this.menuItem7 = new System.Windows.Forms.MenuItem();
this.exitItem = new System.Windows.Forms.MenuItem();
```

```
// Свойства и добавяне на манипулатори
// mainMenu
this.mainMenu.MenuItems.AddRange(
    new System.Windows.Forms.MenuItem[] {this.fileItem});
// fileItem
this.fileItem.MenuItems.AddRange(new System.Windows.Forms.MenuItem[] {
    this.newItem, this.openItem, this.saveItem, this.menuItem5, this.printItem,
    this.menuItem7, this.exitItem});
this.fileItem.Text = "&Файл";
// newItem
this.newItem.Shortcut = System.Windows.Forms.Shortcut.CtrlN;
this.newItem.Text = "&Нов";
// openItem
this.openItem.Shortcut = System.Windows.Forms.Shortcut.CtrlO;
this.openItem.Text = "&Отваряне";
// saveItem
this.saveItem.Shortcut = System.Windows.Forms.Shortcut.CtrlS;
this.saveItem.Text = "&Запазване на студент";
// menuItem5
this.menuItem5.Index = 3;
this.menuItem5.Text = ".";
```

```
// printItem
this.printItem.Enabled = false;
this.printItem.Shortcut = System.Windows.Forms.Shortcut.CtrlP;
this.printItem.Text = "&Печат";
// menuItem7
this.menuItem7.Index = 5;
this.menuItem7.Text = "-";
// exitItem
this.exitItem.Shortcut = System.Windows.Forms.Shortcut.AltF4;
this.exitItem.Text = "&Изход";
// StudentForm
this.Menu = this.mainMenu;
// Създаване на манипулатор на събитието Click за меню Нов
this.newItem.Click += new System.EventHandler(this.newClick);
// Създаване на манипулатор на събитието Click за меню Изход
this.exitItem.Click += new System.EventHandler(this.exitClick);
```

```
// Реализиране на манипулатор при избор на меню Нов
private void newClick(object sender, System.EventArgs e)
{
    Reset(); // Изчистване на контролите
    printItem.Enabled=true; // Менюто Печат е достъпно
}
// Реализиране на манипулатор при избор на меню Изход
private void exitClick(object sender, System.EventArgs e)
{
    Close();
}
// Добавяне на контекстно меню за изчистване на текста за текстовите
// контроли (въвеждане на собствено и фамилно име)
private System.Windows.Forms.ContextMenu textBoxMenu;
private System.Windows.Forms.MenuItem textBoxClearItem;
// Инициализиране на контекстното меню
this.textBoxMenu = new System.Windows.Forms.ContextMenu();
this.textBoxClearItem = new System.Windows.Forms.MenuItem();
```

```
// Свойства на контекстното меню textBoxMenu
this.textBoxMenu.MenuItems.AddRange(
    new System.Windows.Forms.MenuItem[] { this.textBoxClearItem});
// textBoxClearItem
this.textBoxClearItem.Text = "Изчистване на текста";
// Свързване на контекстното меню с текстови контроли
this.firstName.ContextMenu = this.textBoxMenu;
this.lastName.ContextMenu = this.textBoxMenu;
// Създаване на манипулатор на събитието Click за контекстното меню
this.textBoxClearItem.Click +=
    new System.EventHandler(this.textBoxClearClick);
// Реализиране на манипулатор при избор на контекстното меню
private void textBoxClearClick(object sender, System.EventArgs e)
{
    if(textBoxMenu.SourceControl.Equals(firstName))
        firstName.Clear();
    else
        lastName.Clear();
}
```

```
// Създаване и свързване на контекстно меню за формата с най-често
// използваните команди Запазване на студент и Изчисти
public StudentForm()
{
    InitializeComponent();
    Reset();
// Създаване на елементите на контекстно меню
MenuItem[] formMenuItemList=new MenuItem[2];
// Копиране на първия елемент от главното меню чрез CloneMenu
formMenuItemList[0]=saveItem.CloneMenu();
// Създаване на втори елемент чрез готовия манипулатор
// за бутона Изчисти
MenuItem clearItem=new MenuItem("&Изчисти",
    new System.EventHandler(clearClicked));
formMenuItemList[1]=clearItem;
// Създаване и свързване на контекстното меню към формата чрез
// свойството ContextMenu
ContextMenu formMenu = new ContextMenu(formMenuItemList);
this.ContextMenu=formMenu;
}
```

Контроли OpenFileDialog и SaveFileDialog

Изобразяват диалогов прозорец за отваряне/запазване на файл.

Свойства

- AddExtension** true/false – дава/установява автоматично добавяне на разширение към името на файла;
- DefaultExt** дава/установява подразбиращо се разширение към името на файла;
- FileName** дава/установява името на файла, избран в диалоговия прозорец;
- InitialDirectory** дава/установява началната директория;
- Title** дава/установява заглавието на диалоговия прозорец;

- ValidateNames** true/false – дава/установява проверка за валидни файлови имена за Win32;
- OverwritePrompt (SaveFileDialog)** true/false – дава/установява изобразяване на съобщение, ако вече съществува определеното име на файла при SaveAs прозорец.

Методи

- OpenFile** отваря избрания файл, определен чрез свойството FileName, в режим само за четене (OpenFileDialog)/ четене и запис (SaveFileDialog) и връща потока Stream;
- ShowDialog** показва диалогов прозорец и връща DialogResult.OK (натиснат бутон ОК) или DialogResult.Cancel в противен случай.

Контрол RichTextBox

Изобразява контрол за въвеждане и редактиране на текст. Текстът се въвежда директно или се зарежда от текстов файл в Rich Text Format или Plain формат; позволява форматиране на символи и параграфи.

Методи

```
public int Find( string str )
```

Търси низа **str** в текста на контрола. Връща мястото на намерения текст или -1 при неуспешно търсене.

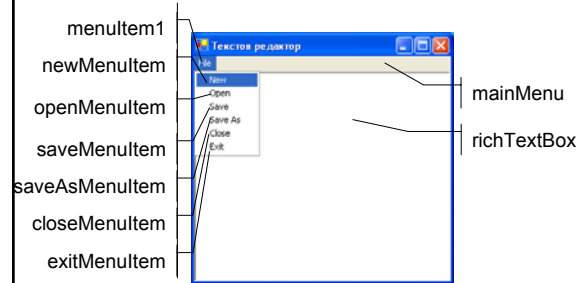
```
public void LoadFile (Stream data, RichTextBoxStreamType fileType);
```

Зарежда съдържанието на съществуващ поток с данни **data** от тип **fileType** (PlainText, RichText, UnicodePlainText) в контрола.

```
public void SaveFile (string path, RichTextBoxStreamType fileType);
```

Запазва съдържанието на контрола във файл с определено име **path** и тип **fileType**.

Пример: Текстов редактор – меню File с подменюта за обработка на файлове.



```
using System;
using System.Drawing;
using System.Collections;
using System.ComponentModel;
using System.Windows.Forms;
using System.Data;
namespace FileMenu
{
    public class Form1 : System.Windows.Forms.Form
    {
        private System.Windows.Forms.MainMenu mainMenu;
        private System.Windows.Forms.MenuItem menuItem1;
        private System.Windows.Forms.OpenFileDialog openFileDialog;
        private System.Windows.Forms.MenuItem newMenuItem;
        private System.Windows.Forms.RichTextBox richTextBox;
        private System.Windows.Forms.MenuItem openMenuItem;
        private System.Windows.Forms.SaveFileDialog saveFileDialog;
        private System.Windows.Forms.MenuItem saveMenuItem;
        private System.Windows.Forms.MenuItem saveAsMenuItem;
        private System.Windows.Forms.MenuItem exitMenuItem;
        private System.Windows.Forms.MenuItem closeMenuItem;
        bool fileNew;
        private System.ComponentModel.IContainer components = null;
```

```
public Form1()
{
    InitializeComponent();
    fileNew=true;
}

protected override void Dispose( bool disposing ) { ... }

#region Windows Form Designer generated code
private void InitializeComponent()
{
    ...
    this.mainMenu = new System.Windows.Forms.MainMenu();
    this.menuItem1 = new System.Windows.Forms.MenuItem();
    this.newMenuItem = new System.Windows.Forms.MenuItem();
    this.openMenuItem = new System.Windows.Forms.MenuItem();
    this.saveMenuItem = new System.Windows.Forms.MenuItem();
    this.openFileDialog = new System.Windows.Forms.OpenFileDialog();
    this.richTextBox = new System.Windows.Forms.RichTextBox();
    this.saveFileDialog = new System.Windows.Forms.SaveFileDialog();
    this.saveAsMenuItem = new System.Windows.Forms.MenuItem();
    this.exitMenuItem = new System.Windows.Forms.MenuItem();
    this.closeMenuItem = new System.Windows.Forms.MenuItem();
}
```

```
// mainMenu
this.mainMenu.MenuItems.AddRange
( new System.Windows.Forms.MenuItem[] { this.menuItem1});
// menuItem1
this.menuItem1.MenuItems.AddRange
( new System.Windows.Forms.MenuItem[] {this.newMenuItem,
this.openMenuItem, this.saveMenuItem, this.saveAsMenuItem,
this.closeMenuItem, this.exitMenuItem});
this.menuItem1.Text = "&File";
// newMenuItem
this.newMenuItem.Text = "&New";
this.newMenuItem.Click +=
new System.EventHandler(this.newMenuItem_Click);
// openMenuItem
this.openMenuItem.Text = "&Open";
this.openMenuItem.Click +=
new System.EventHandler(this.openMenuItem_Click);
// saveMenuItem
this.saveMenuItem.Text = "&Save";
this.saveMenuItem.Click +=
new System.EventHandler(this.saveMenuItem_Click);
```

```
// openFileDialog
this.openFileDialog.DefaultExt = "txt";
this.openFileDialog.Title = "Отваряне на файл";
// richTextBox
this.richTextBox.Dock =
System.Windows.Forms.DockStyle.Fill;
this.richTextBox.Name = "richTextBox";
this.richTextBox.Text = "";
// saveFileDialog
this.saveFileDialog.Title = "Запазване на файл";
// saveAsMenuItem
this.saveAsMenuItem.Text = "Save &As";
this.saveAsMenuItem.Click +=
new System.EventHandler(this.saveAsMenuItem_Click);
// exitMenuItem
this.exitMenuItem.Text = "E&xit";
this.exitMenuItem.Click +=
new System.EventHandler(this.exitMenuItem_Click);
// closeMenuItem
this.closeMenuItem.Text = "&Close";
this.closeMenuItem.Click +=
new System.EventHandler(this.closeMenuItem_Click);
```



```
// Form1
this.Controls.Add(this.richTextBox);
this.Menu = this.mainMenu;
this.Name = "Form1";
this.Text = "Текстов редактор";
}
#endregion
[STAThread]
static void Main()
{
    Application.Run(new Form1());
}
private void newMenuItem_Click(object sender, System.EventArgs e)
{
    richTextBox.Text=String.Empty;
    fileNew=true;
}
}
```

```
private void openMenuItem_Click(object sender, System.EventArgs e)
{
    if(openFileDialog.ShowDialog()==DialogResult.OK)
    {
        richTextBox.LoadFile(openFileDialog.FileName,
            RichTextBoxStreamType.PlainText);
        fileNew=false;
    }
}
private void saveMenuItem_Click(object sender, System.EventArgs e)
{
    if(!fileNew)
    {
        saveFileDialog.FileName=openFileDialog.FileName;
        richTextBox.SaveFile(saveFileDialog.FileName,
            RichTextBoxStreamType.PlainText);
    }
    else
    {
        if((saveFileDialog.ShowDialog() == DialogResult.OK))
        {
            richTextBox.SaveFile(saveFileDialog.FileName,
                RichTextBoxStreamType.PlainText);
        }
        fileNew=false;
    }
}
```

```
private void saveAsMenuItem_Click(object sender, System.EventArgs e)
{
    if((saveFileDialog.ShowDialog() == DialogResult.OK))
    {
        richTextBox.SaveFile(saveFileDialog.FileName,
            RichTextBoxStreamType.PlainText);
    }
}
private void exitMenuItem_Click(object sender, System.EventArgs e)
{
    Application.Exit();
}
private void closeMenuItem_Click(object sender, System.EventArgs e)
{
    richTextBox.Text=String.Empty;
    fileNew=false;
}
}
```

Контрол LinkLabel

Изобразява връзки към други обекти (файлове и Web страници). Представява подчертан текст. При преминаване над контрола указателят на мишката се променя – ръка.

Свойства

- ActiveLinkColor** определя цвета на активната връзка при избор (червен);
- LinkColor** определя цвета на връзката преди да бъде избрана (син);
- LinkVisited** true – връзката се появява, ако е посетена и цветът ѝ се променя, определен от свойството VisitedLinkColor; подразбира се false;
- VisitedLinkColor** определя цвета на посетена връзка (пурпурен);

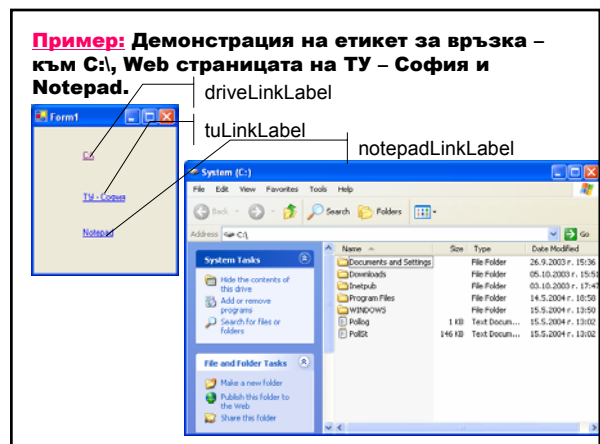
Text изобразяван текст;

UseMnemonic true – символът & в свойството Text действа като клавиш за достъп;

Links списък на LinkLabel.Link обекти – връзките в LinkLabel.

Събития

LinkClicked <L> връзка; подразбиращо се събитие при <L²> в изглед Designer.



Класът `System.Diagnostics.Process` осигурява достъп до локални и отдалечени процеси; позволява стартиране и спиране на локални системни процеси.

```
public static Process Start (string fileName);
public static Process Start (string fileName, string arguments);
```

Стартира процес, определен чрез име `fileName` на документ или файл и установява аргументите `arguments` на командния ред.

```
using System;
using System.Drawing;
using System.Collections;
using System.ComponentModel;
using System.Windows.Forms;
using System.Data;

namespace LinkLabelDemo
{
    public class Form1 : System.Windows.Forms.Form
    {
        private System.Windows.Forms.LinkLabel driveLinkLabel;
        private System.Windows.Forms.LinkLabel tuLinkLabel;
        private System.Windows.Forms.LinkLabel notepadLinkLabel;
        private System.ComponentModel.IContainer components = null;
        public Form1()
        {
            InitializeComponent();
        }

        protected override void Dispose (bool disposing) { ... }
    }
}
```

```
#region Windows Form Designer generated code
private void InitializeComponent()
{
    ...
    this.driveLinkLabel = new System.Windows.Forms.LinkLabel();
    this.tuLinkLabel = new System.Windows.Forms.LinkLabel();
    this.notepadLinkLabel = new System.Windows.Forms.LinkLabel();
    // driveLinkLabel
    this.driveLinkLabel.Name = "driveLinkLabel";
    this.driveLinkLabel.Text = "C:\\";
    this.driveLinkLabel.LinkClicked +=
        new System.Windows.Forms.LinkLabelLinkClickedEventHandler
            (this.driveLinkLabel_LinkClicked);
    // tuLinkLabel
    this.tuLinkLabel.Name = "tuLinkLabel";
    this.tuLinkLabel.Text = "ТУ - София";
    this.tuLinkLabel.LinkClicked +=
        new System.Windows.Forms.LinkLabelLinkClickedEventHandler
            (this.tuLinkLabel_LinkClicked);
}
```

```
// notepadLinkLabel
this.notepadLinkLabel.Name = "notepadLinkLabel";
this.notepadLinkLabel.Text = "Notepad";
this.notepadLinkLabel.LinkClicked +=
    new System.Windows.Forms.LinkLabelLinkClickedEventHandler
        (this.notepadLinkLabel_LinkClicked);
// Form1
this.Controls.Add(this.notepadLinkLabel);
this.Controls.Add(this.tuLinkLabel);
this.Controls.Add(this.driveLinkLabel);
this.Name = "Form1";
this.Text = "Form1";
}
#endregion

[STAThread]
static void Main()
{
    Application.Run(new Form1());
}
```

```
private void driveLinkLabel_LinkClicked(object sender,
    System.Windows.Forms.LinkLabelLinkClickedEventArgs e)
{
    driveLinkLabel.LinkVisited=true;
    System.Diagnostics.Process.Start("C:\\";
}
private void tuLinkLabel_LinkClicked(object sender,
    System.Windows.Forms.LinkLabelLinkClickedEventArgs e)
{
    tuLinkLabel.LinkVisited=true;
    System.Diagnostics.Process.Start("IExplore", "http://www.tu-sofia.bg");
}
private void notepadLinkLabel_LinkClicked(object sender,
    System.Windows.Forms.LinkLabelLinkClickedEventArgs e)
{
    notepadLinkLabel.LinkVisited=true;
    System.Diagnostics.Process.Start("notepad.exe");
}
}
```

Контрол **TreeView**

Изобразява йерархично представяне на възли в дърво.

Дърво (клас `TreeView`) е колекция от възли, организирани йерархично.

Възел (клас `TreeNode`) е обект, който съдържа данни и връзки към други възли.

Създаване на дърво

1. **TreeView** ⇒ свойство **Nodes** ⇒ **TreeNode Editor**



2. Чрез код

а) създаване и добавяне на корен на дървото

```
<дървоTreeView>.Nodes.Add(new TreeNode(<етикет_на_корен>);
```

б) избор на възел-родител

```
<дървоTreeView>.Nodes(индекс_на_възел-родител);
```

в) добавяне на възел-наследник

```
<дървоTreeView>.Nodes(индекс_на_възел-родител).Add(
    new TreeNode(<етикет_на_наследник>);
```

Свойства на контрола **TreeView**

- CheckBoxes** true – определя изображаване на прозорец за проверка; подразбира се false;
- ImageList** определя колекцията **ImageList** от **Image** обекти за иконите за възлите;
- Nodes** списък от колекцията **TreeNode**; съдържа методи **Add** (добавя възел), **Clear** (изтрива колекцията) и **Remove** (изтрива определен възел); изтриването на възел-родител води до изтриването на всички негови наследници;

Събития за контрола **TreeView**

- AfterSelect** промяна на избран възел; подразбиращо се събитие при <L2> в изглед **Designer**.

Свойства на класа **TreeNode**

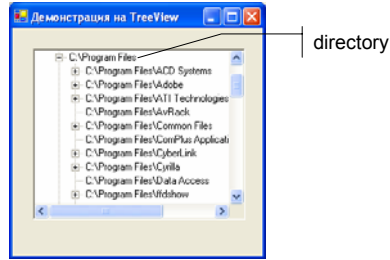
- Checked** true – определя проверка за възела; трябва свойството на родителското дърво **TreeView** да се установи в true;
- FirstNode** първи възел в колекцията **Nodes**;
- FullPath** път на възела, започвайки от корена на дървото;
- ImageIndex** индекс на изображението, което се изображава при деселектиране на възела;
- LastNode** последен възел в колекцията **Nodes**;
- NextNode** следващ възел в дървото;

- Nodes** колекцията **TreeNode** в текущия възел; съдържа методи **Add** (добавя възел), **Clear** (изтрива колекцията) и **Remove** (изтрива определен възел); изтриването на възел-родител води до изтриването на всички негови наследници;
- PrevNode** предшестващ възел в дървото;
- SelectedImageIndex** индекс на изображението при избор на възела;
- Text** изображаван текст в контрола **TreeView**.

Методи на класа **TreeNode**

- Collapse** пропада възел;
- Expand** разширява възел;
- ExpandAll** разширява всички наследници на възел;
- GetNodeCount** връща броя на възлите-наследници.

Пример: Изобразява дървовидната файлова структура на управляващото устройство C:\.



```
using System;
using System.Drawing;
using System.Collections;
using System.ComponentModel;
using System.Windows.Forms;
using System.Data;
using System.IO;

namespace TreeView
{
    public class Form1 : System.Windows.Forms.Form
    {
        private System.Windows.Forms.TreeView directory;
        private System.ComponentModel.IContainer components = null;
        public Form1()
        {
            InitializeComponent();
        }
        protected override void Dispose( bool disposing ) { ... }
    }
}
```

```
#region Windows Form Designer generated code
private void InitializeComponent()
{
    ...
    this.directory = new System.Windows.Forms.TreeView();
    this.SuspendLayout();
    // directory
    this.directory.ImageIndex = -1;
    this.directory.Name = "directory";
    this.directory.SelectedImageIndex = -1;
    // Form1
    this.Controls.Add(this.directory);
    this.Name = "Form1";
    this.Text = "Демонстрация на TreeView";
    this.Load += new System.EventHandler(this.Form1_Load);
}
#endregion

[STAThread]
static void Main()
{
    Application.Run(new Form1());
}
```

```
public void PopulateTreeView( string directoryValue,TreeNode parentNode)
{
    string[] directoryArray = Directory.GetDirectories(directoryValue);
    try
    {
        if (directoryArray.Length != 0)
        {
            foreach (string d in directoryArray)
            {
                TreeNode myNode=new TreeNode(d);
                parentNode.Nodes.Add(myNode);
                PopulateTreeView(d, myNode);
            }
        }
    }
    catch(UnauthorizedAccessException e)
    {
        parentNode.Nodes.Add("Достъпът не е разрешен");
    }
}

private void Form1_Load(object sender, System.EventArgs e)
{
    directory.Nodes.Add("C:\");
    PopulateTreeView("C:\", directory.Nodes[0]);
}
}
```

Контрол ListView
Изобразява списък от елементи (клас ListViewItem), като пред елементите изобразява икони по различни начини.

Дефиниране на икони за елементите
1. Toolbox => ImageList => свойство Images => Image Collection Editor => Add

2. **ListView** ⇒
 свойство **SmallImageList** ⇒ обект **ImageList**
 свойство **LargeImageList** ⇒ обект **ImageList**
3. Икони за елементите **ListViewItem** –
 в кода се установява свойството **ImageIndex**

Свойства на контрола ListView

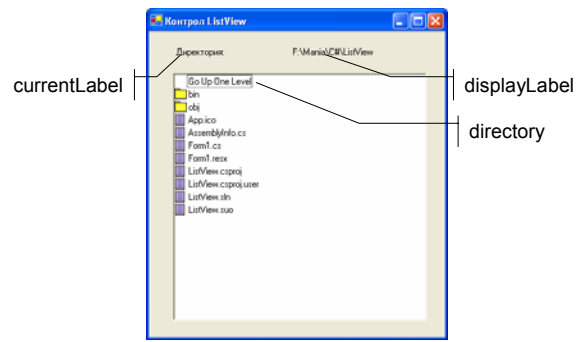
- Activation** определя начина за активиране на елемент: **OneClick (<L>)**, **TwoClick (<L²>)**, елементът променя цвета си при активиране) и **Standard (<L²>)**;
- CheckBoxes** определя изобразяване на елемента с проверка (**true**); подразбира се **false**;
- LargeImageList** използван списък от изображения за големите икони;
- Items** връща колекция от **ListViewItem** в контрола;
- MultiSelect** многократен избор (**true/false**);
- SelectedItems** списък от текущо избраните елементи;

- SmallImageList** използван списък от изображения за малките икони;
- View** определя вида на елементите в списъка: **LargeIcon**, **SmallIcon**, **List** и **Details**.

Събития за контрола ListView

- ItemActivate** активиране на елемент от контрола; не определя кой точно елемент е избран.

Пример: Изобразява файлове и директории чрез ListView



```
using System;
using System.Drawing;
using System.Collections;
using System.ComponentModel;
using System.Windows.Forms;
using System.Data;
using System.IO;
namespace ListView
{
    public class Form1 : System.Windows.Forms.Form
    {
        private System.Windows.Forms.ListView directory;
        string currentDirectory=Directory.GetCurrentDirectory();
        private System.Windows.Forms.Label displayLabel;
        private System.Windows.Forms.ImageList fileFolder;
        private System.Windows.Forms.Label currentLabel;
        private System.ComponentModel.IContainer components;
        public Form1()
        {
            InitializeComponent();
        }
        protected override void Dispose( bool disposing ) { ... }
    }
}
```

```
#region Windows Form Designer generated code
private void InitializeComponent()
{
    ...
    this.components = new System.ComponentModel.Container();
    System.Resources.ResourceManager resources =
        new System.Resources.ResourceManager(typeof(Form1));
    this.directory = new System.Windows.Forms.ListView();
    this.fileFolder =
        new System.Windows.Forms.ImageList(this.components);
    this.displayLabel = new System.Windows.Forms.Label();
    this.currentLabel = new System.Windows.Forms.Label();
    // directory
    this.directory.LargeImageList = this.fileFolder;
    this.directory.Name = "directory";
    this.directory.SmallImageList = this.fileFolder;
    this.directory.View = System.Windows.Forms.View.List;
    this.directory.Click += new System.EventHandler(this.directory_Click);
    // fileFolder
    this.fileFolder.ImageStream =
        ((System.Windows.Forms.ImageListStreamer)
            (resources.GetObject("fileFolder.ImageStream")));
    this.fileFolder.TransparentColor = System.Drawing.Color.Transparent;
}
```

```
// displayLabel
this.displayLabel.Name = "displayLabel";

// currentLabel
this.currentLabel.Name = "currentLabel";
this.currentLabel.Text = "Директория:";

// Form1
this.Controls.Add(this.currentLabel);
this.Controls.Add(this.displayLabel);
this.Controls.Add(this.directory);
this.Name = "Form1";
this.Text = "Контрол ListView";
this.Click += new System.EventHandler(this.directory_Click);
this.Load += new System.EventHandler(this.Form1_Load);
}
#endregion
[STAThread]
static void Main()
{
    Application.Run(new Form1());
}
```

```
private void directory_Click(object sender, System.EventArgs e)
{ // Ако е избрана директория
if(directory.SelectedItems.Count != 0)
{ // Ако е избран първият елемент (винаги е Go Up One Level)
if(directory.Items[0].Selected)
{ DirectoryInfo directoryObject = new DirectoryInfo(currentDirectory);
// Ако не е коренът на дървото
if(directoryObject.Parent != null)
LoadFilesInDirectory (directoryObject.Parent.FullName);
} else
{ // Преглеждане на структурата на дървото
string chosen=directory.SelectedItems[0].Text;
// Проверка за избор на директория или файл
if(Directory.Exists(currentDirectory+"\\"+chosen))
{ if(currentDirectory=="C:\\"")
LoadFilesInDirectory(currentDirectory+chosen);
else LoadFilesInDirectory(currentDirectory+"\\"+chosen);
}
}
displayLabel.Text=currentDirectory;
}
}
```

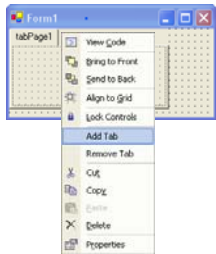
```
public void LoadFilesInDirectory(string currentDirectoryValue)
{ try
{ directory.Items.Clear();
directory.Items.Add("Go Up One Level");
// При избор на елемента Go Up One Level
currentDirectory=currentDirectoryValue;
DirectoryInfo newCurrentDirectory = new DirectoryInfo(currentDirectory);
DirectoryInfo[] directoryArray = newCurrentDirectory.GetDirectories();
FileInfo[] fileArray = newCurrentDirectory.GetFiles();
foreach(DirectoryInfo d in directoryArray)
{ ListViewItem newDirectoryItem = directory.Items.Add(d.Name);
newDirectoryItem.ImageIndex=0; // икона за директория
}
foreach(FileInfo file in fileArray)
{ ListViewItem newFileItem = directory.Items.Add(file.Name);
newFileItem.ImageIndex = 1; // икона за файл
}
}
catch(UnauthorizedAccessException)
{ MessageBox.Show("Внимание: Няма достатъчно привилегии",
"Attention", 0, MessageBoxIcon.Warning); }
}
```

```
private void Form1_Load(object sender, System.EventArgs e)
{
    Image folderImage = Image.FromFile(currentDirectory+"\\Folder.bmp");
    Image fileImage = Image.FromFile(currentDirectory+"\\File.bmp");
    fileFolder.Images.Add(folderImage);
    fileFolder.Images.Add(fileImage);
    LoadFilesInDirectory(currentDirectory);
    displayLabel.Text=currentDirectory;
}
}
```

Контрол TabControl

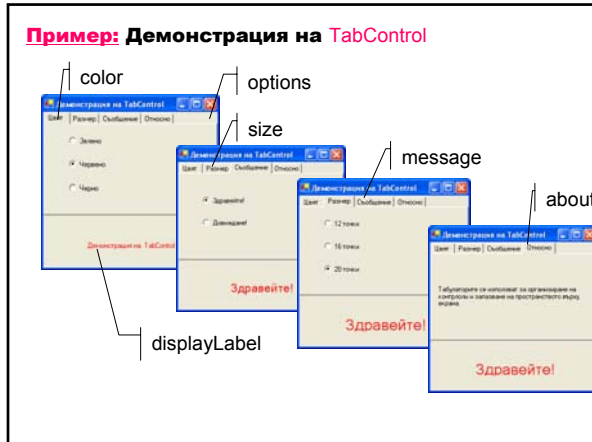
Изобразява табулирани прозорци – съдържа много контроли. Състои се от обекти **TabPage**, които съдържат контролите.

<R> TabControl ⇒ AddTab



Свойства	
ImageList	списък от изображенията, представени от табулатора;
ItemSize	размер на табулатора;
MultiLine	табулаторът изобразява много линии (true);
SelectedIndex	индекс на избрания TabPage ;
SelectedTab	избраният TabPage ;
TabCount	брой табулатори;
TabPage	колекция от TabPage в TabControl .
Събития	
SelectedIndexChanged	при промяна на SelectedIndex .

Пример: Демонстрация на TabControl



```
using System;
using System.Drawing;
using System.Collections;
using System.ComponentModel;
using System.Windows.Forms;
using System.Data;

namespace TabControl
{
    public class Form1 : System.Windows.Forms.Form
    {
        private System.Windows.Forms.TabControl options;
        private System.Windows.Forms.TabPage color;
        private System.Windows.Forms.RadioButton green;
        private System.Windows.Forms.RadioButton red;
        private System.Windows.Forms.RadioButton black;
        private System.Windows.Forms.Label displayLabel;
        private System.Windows.Forms.RadioButton size12;
        private System.Windows.Forms.RadioButton size16;
        private System.Windows.Forms.RadioButton size20;
        private System.Windows.Forms.TabPage message;
    }
}
```

```
private System.Windows.Forms.RadioButton hello;
private System.Windows.Forms.RadioButton goodBye;
private System.Windows.Forms.TabPage about;
private System.Windows.Forms.Label messageLabel;
private System.ComponentModel.Container components = null;
public Form1()
{
    InitializeComponent();
}
protected override void Dispose( bool disposing ) { ... }
#region Windows Form Designer generated code
private void InitializeComponent()
{
    ...
    this.options = new System.Windows.Forms.TabControl();
    this.color = new System.Windows.Forms.TabPage();
    this.green = new System.Windows.Forms.RadioButton();
    this.red = new System.Windows.Forms.RadioButton();
    this.black = new System.Windows.Forms.RadioButton();
    this.displayLabel = new System.Windows.Forms.Label();
    this.size = new System.Windows.Forms.RadioButton();
    this.size12 = new System.Windows.Forms.RadioButton();
}
```

```
this.size16 = new System.Windows.Forms.RadioButton();
this.size20 = new System.Windows.Forms.RadioButton();
this.message = new System.Windows.Forms.TabPage();
this.hello = new System.Windows.Forms.RadioButton();
this.goodBye = new System.Windows.Forms.RadioButton();
this.about = new System.Windows.Forms.TabPage();
this.messageLabel = new System.Windows.Forms.Label();
this.options.SuspendLayout();

// options
this.options.Controls.Add(this.color);
this.options.Controls.Add(this.size);
this.options.Controls.Add(this.message);
this.options.Controls.Add(this.about);
this.options.Name = "options";

// color
this.color.Controls.Add(this.green);
this.color.Controls.Add(this.red);
this.color.Controls.Add(this.black);
this.color.Name = "color";
this.color.Text = "Цвят";

// size
this.size.Controls.Add(this.size20);
this.size.Controls.Add(this.size16);
this.size.Controls.Add(this.size12);
this.size.Name = "size";
this.size.Text = "Размер";

// size12
this.size12.Name = "size12";
this.size12.Text = "12 точки";
this.size12.CheckedChanged +=
    new System.EventHandler(this.size12_CheckedChanged);

// size16
this.size16.Name = "size16";
this.size16.Text = "16 точки";
this.size16.CheckedChanged +=
    new System.EventHandler(this.size16_CheckedChanged);

// size20
this.size20.Name = "size20";
this.size20.Text = "20 точки";
this.size20.CheckedChanged +=
    new System.EventHandler(this.size20_CheckedChanged);
```

```
// green
this.green.Name = "green";
this.green.Text = "Зелено";
this.green.CheckedChanged +=
    new System.EventHandler(this.green_CheckedChanged);

// red
this.red.Name = "red";
this.red.Text = "Червено";
this.red.CheckedChanged +=
    new System.EventHandler(this.red_CheckedChanged);

// black
this.black.Name = "black";
this.black.Text = "Черно";
this.black.CheckedChanged +=
    new System.EventHandler(this.black_CheckedChanged);

// displayLabel
this.displayLabel.Name = "displayLabel";
this.displayLabel.Text = "Демонстрация на TabControl";
this.displayLabel.TextAlign =
    System.Drawing.ContentAlignment.MiddleCenter;
```

```
// size
this.size.Controls.Add(this.size20);
this.size.Controls.Add(this.size16);
this.size.Controls.Add(this.size12);
this.size.Name = "size";
this.size.Text = "Размер";

// size12
this.size12.Name = "size12";
this.size12.Text = "12 точки";
this.size12.CheckedChanged +=
    new System.EventHandler(this.size12_CheckedChanged);

// size16
this.size16.Name = "size16";
this.size16.Text = "16 точки";
this.size16.CheckedChanged +=
    new System.EventHandler(this.size16_CheckedChanged);

// size20
this.size20.Name = "size20";
this.size20.Text = "20 точки";
this.size20.CheckedChanged +=
    new System.EventHandler(this.size20_CheckedChanged);
```

```
// message
this.message.Controls.Add(this.goodBye);
this.message.Controls.Add(this.hello);
this.message.Location = new System.Drawing.Point(4, 22);
this.message.Name = "message";
this.message.Text = "Съобщение";

// hello
this.hello.Name = "hello";
this.hello.Text = "Здравейте!";
this.hello.CheckedChanged +=
    new System.EventHandler(this.hello_CheckedChanged);

// goodBye
this.goodBye.Name = "goodBye";
this.goodBye.Text = "Довиждане!";
this.goodBye.CheckedChanged +=
    new System.EventHandler(this.goodBye_CheckedChanged);

// about
this.about.Controls.Add(this.messageLabel);
this.about.Name = "about";
this.about.Text = "Относно";
```

```
// messageLabel
this.messageLabel.Name = "messageLabel";
this.messageLabel.Text = "Табулаторите се използват за "
    + "организиране на контроли и запазване на "
    + "пространството върху екрана.";

// Form1
this.Controls.Add(this.displayLabel);
this.Controls.Add(this.options);
this.Name = "Form1";
this.Text = "Демонстрация на TabControl ";
}
#endregion
[STAThread]
static void Main()
{
    Application.Run(new Form1());
}
private void black_CheckedChanged(object sender, System.EventArgs e)
{
    displayLabel.ForeColor=Color.Black;
}
```

```
private void red_CheckedChanged(object sender, System.EventArgs e)
{
    displayLabel.ForeColor=Color.Red;
}
private void green_CheckedChanged(object sender, System.EventArgs e)
{
    displayLabel.ForeColor=Color.Green;
}
private void size12_CheckedChanged(object sender, System.EventArgs e)
{
    displayLabel.Font = new Font(displayLabel.Font.Name, 12);
}
private void size16_CheckedChanged(object sender, System.EventArgs e)
{
    displayLabel.Font = new Font(displayLabel.Font.Name, 16);
}
private void size20_CheckedChanged(object sender, System.EventArgs e)
{
    displayLabel.Font=new Font(displayLabel.Font.Name,20);
}
```

```
private void hello_CheckedChanged(object sender, System.EventArgs e)
{
    displayLabel.Text = "Здравейте!";
}
private void goodBye_CheckedChanged(object sender,
    System.EventArgs e)
{
    displayLabel.Text = "Довиждане!";
}
}
```