|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ПРАВИТЕЛЬСТВО САНКТ-ПЕТЕРБУРГАКОМИТЕТ ПО НАУКЕ И ВЫСШЕЙ ШКОЛЕ Санкт-Петербургское государственное бюджетное профессиональное образовательное учреждение «ПЕТРОВСКИЙ КОЛЛЕДЖ» (СПб ГБПОУ «Петровский колледж») | | | | |
| Отделение информационно-промышленных технологий и судостроения | | | | |
| Практическая работа №9 | | | | |
|  | | | | |
| **Обзор управляющих элементов и получение навыков работы с ними** | | | | |
| Выполнил: | | | |
| Семенов Сергей Андреевич | | | |
| студент(ка): | | 3902 | группы |
|  | |  |  |
| Специальность: | | 09.02.03 Программирование в компьютерных системах | |
| Преподаватель: | |  | |
|  | Спиридонова Т. С.  Бережков А. В. | | | |

|  |
| --- |
| Санкт-Петербург, 2022 |
|  |
|  |

# Задание

Цель: изучение класса Control, основных свойств и методов стандартных элементов управления,

изучение класса CommonDialog.

I. Изучение класса Control

1. Создайте приложение TestMenuToolboxStatus.

2. Перетащите элемент управления MenuStrip на форму, что позволит вводить текст меню

непосредственно в элементы меню и создайте следующее меню (рис.1).

Рис. 1 Простое меню на форме

3. При двойном щелчке на пункте меню Visual Studio автоматически сгенерирует оболочку для

обработчика события Click и перейдет в окно кода.

4. Для тестирования меню создайте обработчик пункта меню "Мое действие", который будет сообщать,

что выбран именно этот пункт меню.

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

{

MessageBox.Show("Пункт меню");

}

5. Реализуйте следующее: элемент управления TextBox содержит в себе простейшее контекстное меню,

дублирующее действия подменю Edit (рис. 2):

Рис. 2. Контекстное меню элемента TextBox

Для этого:

5.1. Перетащите элемент управления contextMenuStrip.

5.2. Создайте пункты Cut, Copy, Paste, SelectAll и обработчики для них:

private void toolStripMenuItem2\_Click(object sender, EventArgs e)

{

textBox1.Cut();

}

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

{

textBox1.Copy();

}

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

{

textBox1.Paste();

}

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

{

textBox1.SelectAll();

}

5.3. Определите, где будет появляться контекстное меню. Элемент TextBox имеет свойство

ContextMenu, где мы укажем Name контекстного меню ( в нашем случае это contextMenu1)

6. Перетащите элемент управления StatusStrip на форму. Откоройте выпадающий список этого

объекта на форме и выберите StatusLable:

Рис. 3. Добавляем метку в строку состояния

7. Для отображения информации о пункте меню, на который наведена мышка необходимо:

7.1. В дизайнере формы выделить пункт, перейти на вкладку Properties и открыть окно событий, нажав

кнопку На данной вкладке необходимо выделить событие MouseEnter и в поле ввода сделать

двойной щелчок.

7.2. Система сгенерирует код обработчика, который привести к виду:

private void моеДействиеToolStripMenuItem\_MouseEnter(object sender,

EventArgs e)

{

toolStripStatusLabel1 = "Будет выведено сообщение";

}

После компиляции получим следующее:

Рис. 4. Вывод сообщения в строке состояния

7.3. Для изменения текста при потере фокуса мыши на исходный, необходимо воспользоваться

событием "MouseLeave", которое генерируется, когда мышь перемещается (покидает) с пункта меню.

Обработчик данного события примет следующий вид:

private void моеДействиеToolStripMenuItem\_MouseLeave(object sender,

EventArgs e)

{

toolStripStatusLabel1.Text = "";

}

7.4. Самостоятельно сформировать соответствующие события MouseEnter и MouseLeave для всех

подпунктов меню и создать для них обработчики.

8. Подписать всем компонентам на форме ToolTips.

ІІ. UseListAndCombo

1. Выполните последовательность действий:

- разместите на форме: textBox1, button1, comboBox1;

- реализуйте метод нажатия на кнопку следующим образом:

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

{

comboBox1.Items.Add(textBox1.Text);

}

Это даст возможность добавить в выпадающий список введенное значение в поле ввода.

2. Самостоятельно: добавить еще две кнопки и listBox1. При нажатии на кнопки реализовать

добавление/исключение выбранного значения выпадающего списка в listBox1 (перенос осуществлять с

удалением значения из соотв. списка)

Рис. 5.

ІІІ. Изучение диалоговых окон.

1.Создайте приложение TestDialogs.

2. На главной форме проекта разместить компонент RichTextBox, полосу главного меню (используя

Insert Standard Items), и панель инструментов toolStrip, на которой дублируются пункты меню:

Рис. 6. Пример формы

3. Для работы с файловыми потоками в коде формы blank подключаем пространство имен System.IO:

using System.IO;

4.Перетащите на форму OpenFileDialog и приведите к виду:

Рис.7. Свойства OpenFileDialog

5. В обработчике события нажатия соответствующего пункта меню прописать:

OpenFileDialog fd = new OpenFileDialog();

fd.Filter = "Text format (\*.txt)|\*.txt|Rich Text (\*.rtf)|\*.rtf";

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

{

// если выбрали текст

if (fd.FilterIndex == 1)

richTextBox1.LoadFile(fd.FileName,RichTextBoxStreamType.PlainText);

else

richTextBox1.LoadFile(fd.FileName, RichTextBoxStreamType.RichText);

}

6. В обработчике события для SaveFileDialog:

SaveFileDialog fd = new SaveFileDialog();

fd.Filter = "Text format (\*.txt)|\*.txt|Rich Text (\*.rtf)|\*.rtf";

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

{

// если выбрали текст

if (fd.FilterIndex == 1)

richTextBox1.SaveFile(fd.FileName,RichTextBoxStreamType.PlainText);

else

richTextBox1.SaveFile(fd.FileName, RichTextBoxStreamType.RichText);

}

7. По аналогии создать обработчики диалогов ColorDialog и FontDialog:

ColorDialog d = new ColorDialog();

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

{

richTextBox1.SelectionColor = d.Color;

}

FontDialog d = new FontDialog();

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

{

richTextBox1.SelectionFont = d.Font;

}

8. В обработчике события закрытия формы:

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

{

if (MessageBox.Show("Вы уверены?", "Вопрос", MessageBoxButtons.YesNo,

MessageBoxIcon.Question, MessageBoxDefaultButton.Button2) !=

DialogResult.Yes)

{

e.Cancel = true;

}

}

# Листинг

Главная форма

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 Pract9\_semenov\_39\_02

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void contextMenuStrip1\_Opening(object sender, CancelEventArgs e)

{

}

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

{

}

private void cutToolStripMenuItem1\_Click(object sender, EventArgs e)

{

textBox1.Cut();

}

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

{

textBox1.Copy();

}

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

{

textBox1.Paste();

}

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

{

textBox1.SelectAll();

}

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

{

MessageBox.Show("Hello, hello");

}

private void copyToolStripMenuItem1\_Click(object sender, EventArgs e)

{

textBox1.Copy();

}

private void pasteToolStripMenuItem1\_Click(object sender, EventArgs e)

{

textBox1.Paste();

}

private void cutToolStripMenuItem2\_Click(object sender, EventArgs e)

{

textBox1.Cut();

}

private void cutToolStripMenuItem2\_Click\_1(object sender, EventArgs e)

{

textBox1.Cut();

}

private void contextMenuStrip1\_Opening\_1(object sender, CancelEventArgs e)

{

}

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

{

comboBoxChoice.Items.Add(textBox2.Text);

textBox2.Text = "";

}

private void selectAllToolStripMenuItem1\_Click(object sender, EventArgs e)

{

textBox1.SelectAll();

}

private void toolStripStatusLabel1\_Click(object sender, EventArgs e)

{

}

private void toolStripStatusLabel1\_MouseEnter(object sender, EventArgs e)

{

toolStripStatusLabel1.Text = "Мое сообщение!";

}

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

{

listBoxItems.Items.Add(comboBoxChoice.SelectedItem.ToString());

}

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

{

listBoxItems.Items.Remove(comboBoxChoice.SelectedItem.ToString());

}

private void button1\_Click\_1(object sender, EventArgs e)

{

this.Hide();

FormDialogWindow fr = new FormDialogWindow();

fr.ShowDialog();

}

}

}

Форма диалогового окна

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.IO;

namespace Pract9\_semenov\_39\_02

{

public partial class FormDialogWindow : Form

{

public FormDialogWindow()

{

InitializeComponent();

}

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

{

this.Hide();

Form1 form1 = new Form1();

form1.ShowDialog();

}

private void openFileDialog1\_FileOk(object sender, CancelEventArgs e)

{

}

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

{

OpenFileDialog fd = new OpenFileDialog();

fd.Filter = "Text format (\*.txt)|\*.txt|Rich Text (\*.rtf)|\*.rtf";

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

{

// если выбрали текст

if (fd.FilterIndex == 1)

richTextBox1.LoadFile(fd.FileName, RichTextBoxStreamType.PlainText);

else

richTextBox1.LoadFile(fd.FileName, RichTextBoxStreamType.RichText);

}

}

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

{

richTextBox1.Text = "";

}

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

{

SaveFileDialog fd = new SaveFileDialog();

fd.Filter = "Text format (\*.txt)|\*.txt|Rich Text (\*.rtf)|\*.rtf";

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

{

// если выбрали текст

if (fd.FilterIndex == 1)

richTextBox1.SaveFile(fd.FileName, RichTextBoxStreamType.PlainText);

else

richTextBox1.SaveFile(fd.FileName, RichTextBoxStreamType.RichText);

}

}

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

{

ColorDialog d = new ColorDialog();

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

{

richTextBox1.SelectionColor = d.Color;

}

}

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

{

FontDialog d = new FontDialog();

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

{

richTextBox1.SelectionFont = d.Font;

}

}

private void closeToolStripMenuItem1\_Click(object sender, EventArgs e)

{

if (MessageBox.Show("Вы уверены?", "Вопрос", MessageBoxButtons.YesNo,

MessageBoxIcon.Question, MessageBoxDefaultButton.Button2) !=

DialogResult.Yes)

{

}

else

{

this.Close();

}

}

}

}