

## **Методическая разработка Текстового редактора в среде Visual Studio на языке C#**

**Тема:** Создание текстового редактора в среде Visual Studio на языке C#

**Цель работы:** Разработка текстового редактора в среде Visual Studio на языке C#

**Оборудование:** IBM – совместимые компьютеры.

**Место проведения:** Компьютерный класс.

**Техника безопасности:** См. инструкцию.

### **Ход работы**

Последовательность действий:

- 1) визуальное проектирование интерфейса (перенос на форму с Панели элементов необходимых визуальных и не визуальных элементов);
- 2) генерация заготовок методов обработки событий, связанных с элементами управления;
- 3) программирование методов обработки событий.

### **Постановка задачи**

Создать текстовый редактор с обязательными функциями работы с файлами «Открыть» и «Сохранить», а также функциями редактирования текста – Вырезать, Копировать, Вставить.

### **Выполнение**

1. Запустить MS Visual Studio.  
Создать проект по шаблону [Windows Forms Application](#)  
Сохранить проект под любым именем.
2. При разработке использовать следующие компоненты richTextBox, button, saveFileDialog, openFileDialog, menuStrip.
3. Форма должна иметь следующий вид (Рисунок 1)

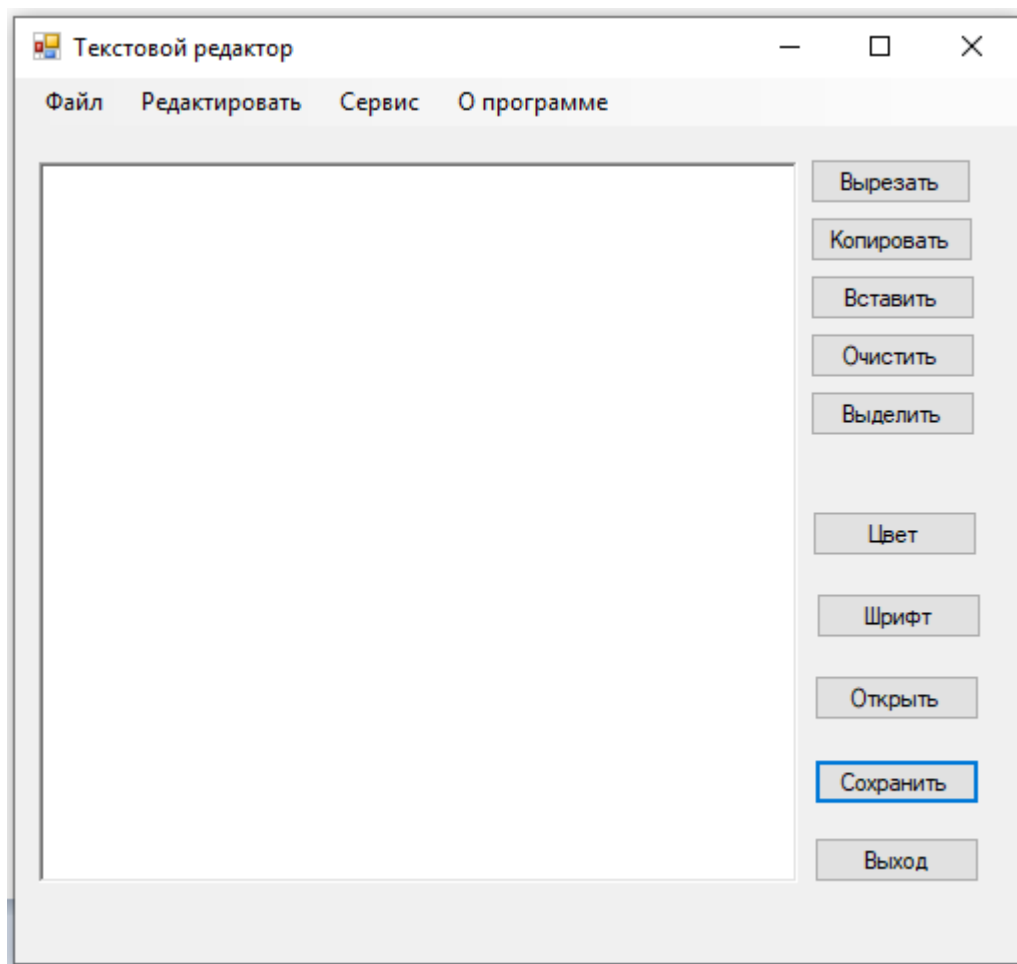


Рисунок 1

4. Для открытия файла используем следующий код

```
private void buttonOpen_Click_1(object sender, EventArgs e)
{
    if (openFileDialog1.ShowDialog() == DialogResult.OK) //Проверяем был ли выбран файл
    {
        richTextBox.Clear(); //Очищаем richTextBox
        openFileDialog1.Filter = "Text Files (*.txt)|*.txt"; //Указываем что нас интересуют
        только текстовые файлы
        string fileName = openFileDialog1.FileName; //получаем наименование файл и путь к
        нему.
        richTextBox.Text = File.ReadAllText(fileName, Encoding.GetEncoding(1251)); //Передаем
        содержимое файла в richTextBox
    }
}
```

5. Для сохранения файла используем следующий код

```
private void buttonSave_Click_1(object sender, EventArgs e)
{
    saveFileDialog1.Filter = "Text Files|*.txt"; //Задаем доступные расширения
    saveFileDialog1.DefaultExt = ".txt"; //Задаем расширение по умолчанию
}
```

```

    if (saveFileDialog1.ShowDialog() == DialogResult.OK) //Проверяем подтверждение
сохранения информации.
    {
        var name = saveFileDialog1.FileName; //Задаем имя файлу
        File.WriteAllText(name, richTextBox.Text, Encoding.GetEncoding(1251)); //Записываем
в файл содержимое richTextBox с кодировкой 1251
    }
    richTextBox.Clear();
}

```

6. Добавить модуль using System.IO;

7. Разработайте главное меню на основе компонента menuStrip (рисунки 2 – 4).

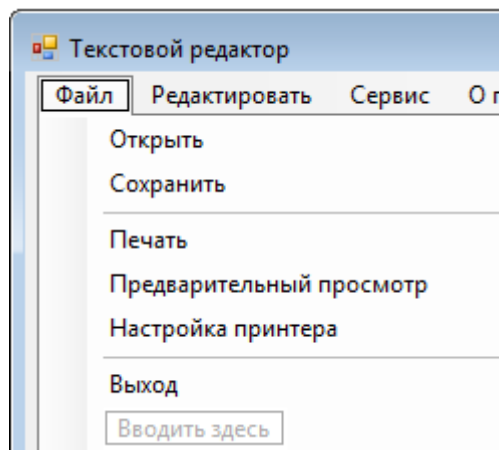


Рисунок 2

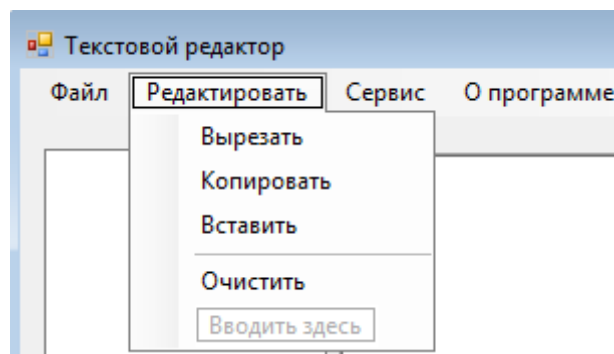


Рисунок 3

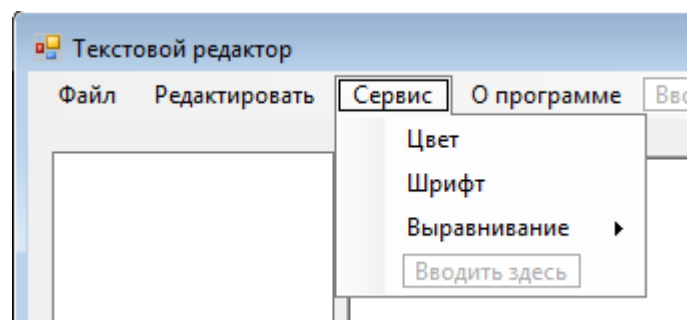


Рисунок 4

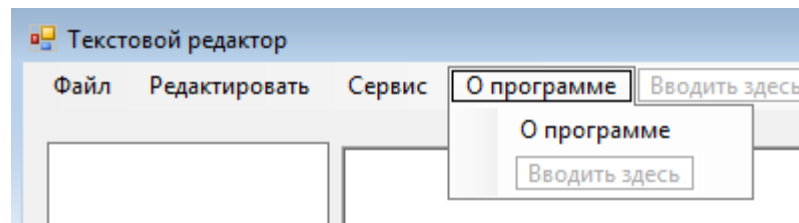


Рисунок 5

8. Для выполнения функций редактирования использовать функции обмена данных через буфер обмена (Cut, Paste, Copy).

9. В меню «Сервис» настроить выравнивание текста (Left, Center, Right ) по примеру кода

```
richTextBox.Select (); // выравнивание только выделенного текста
//richTextBox1.SelectAll(); //выделение всего текста
richTextBox1.SelectionAlignment = HorizontalAlignment.Center;
//richTextBox1.DeselectAll(); //Отмена выделения
```

10. Из Toolbox перенесите на форму элемент PrintDocument.

11. С помощью окна Properties для компонента printDocument1 добавьте обработчик события PrintPage и внутри него добавьте следующий код:

```
Font myFont = new Font("Tahoma", 12, FontStyle.Regular, GraphicsUnit.Pixel);
string Hello = "Hello World!";
e.Graphics.DrawString(Hello, myFont, Brushes.Black, 20, 20);
```

12. Откройте Form1 в режиме конструктора.

13. В главном меню Файл добавить три кнопки Печать, Настройка принтера, Предварительный просмотр (Рисунок 6).

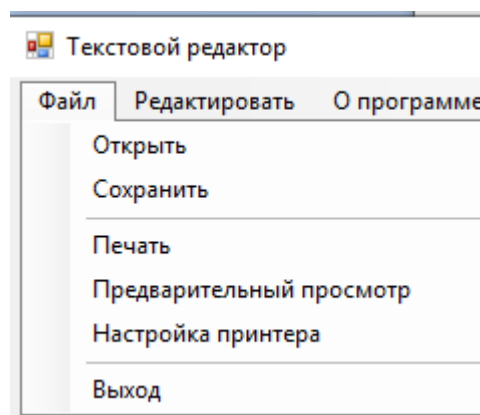


Рисунок 6

14. Из Toolbox перетащите в форму компоненты PageSetupDialog, PrintDialog и PrintPreviewDialog.

15. В окне Properties укажите свойству Document каждого компонента диалогового окна значение PrintDocument1.

16. Для элемента PrintDialog присвойте свойству AllowSomePages в значение True.

17. В конструкторе дважды щелкните кнопку Page Setup и добавьте следующий код:

```
pageSetupDialog1.ShowDialog();
```

18. В конструкторе дважды щелкните Print и добавьте следующий код:

```
if (printDialog1.ShowDialog() == DialogResult.OK)    printDocument1.Print();
```

19. В конструкторе дважды щелкните Print Preview и добавьте следующий код:

```
printPreviewDialog1.ShowDialog();
```

20. В пункте меню «О программе» добавить открытие формы AboutBox (Проект – Добавление нового элемента – Окно «О программе»). Заменить данные в окне можно по следующему примеру кода

```
public AboutBox1()
{
    InitializeComponent();
    this.Text = String.Format("О {0}", AssemblyTitle);
    this.labelProductName.Text = AssemblyProduct;
    this.labelVersion.Text = String.Format("Версия {0}",
AssemblyVersion);
    this.labelCopyright.Text = "Иванов И.И."; // AssemblyCopyright;
    this.labelCompanyName.Text = AssemblyCompany;
    this.textBoxDescription.Text = "Текстовый редактор";
//AssemblyDescription;
}
```

21. Добавить на форму компонент контекстного меню contextMenuStrip, зафиксировать его за компонентом richTextBox и продублировать функции редактирования текста.

22. Постройте и запустите приложение. Проверить работоспособность всех выполняемых функций в программе.

23. Полученное приложение показать преподавателю.