



ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ И НАУКИ

ГОРОДА МОСКВЫ

Государственное бюджетное профессиональное

образовательное учреждение города Москвы

«Колледж малого бизнеса № 4»

(ГБПОУ КМБ № 4)

Отчёт по лабораторной работе №4

Специальность: 09.02.07 Информационные системы и программирование

Форма обучения: очная

Студент: Межибор Ярослав Евгеньевич

Группа: ИПО-21.24

Проверил: Рыбаков Александр Сергеевич

Москва, 2025 г.

Цель:

Научиться делать приложения с нормальным главным меню, контекстными меню по правой кнопке мыши и стандартными диалогами для открытия или сохранения файлов.

Задачи:

- Сделать MenuStrip, добавить туда типичные пункты Файл, Правка и сделать горячие клавиши, чтобы всё работало как в нормальных программах.
- Сделать ContextMenuStrip, который вылезит в нужном месте по клику правой кнопкой.
- Интегрировать стандартные системные окна - OpenFileDialog для выбора файлов, SaveFileDialog для сохранения и ColorDialog, чтобы пользователь мог сам менять цвета в интерфейсе.
- Прописать код, чтобы при выборе цвета в диалоге или нажатии кнопки в меню реально менялся текст или настройки в программе.

Ход работы:

Упражнение 1:

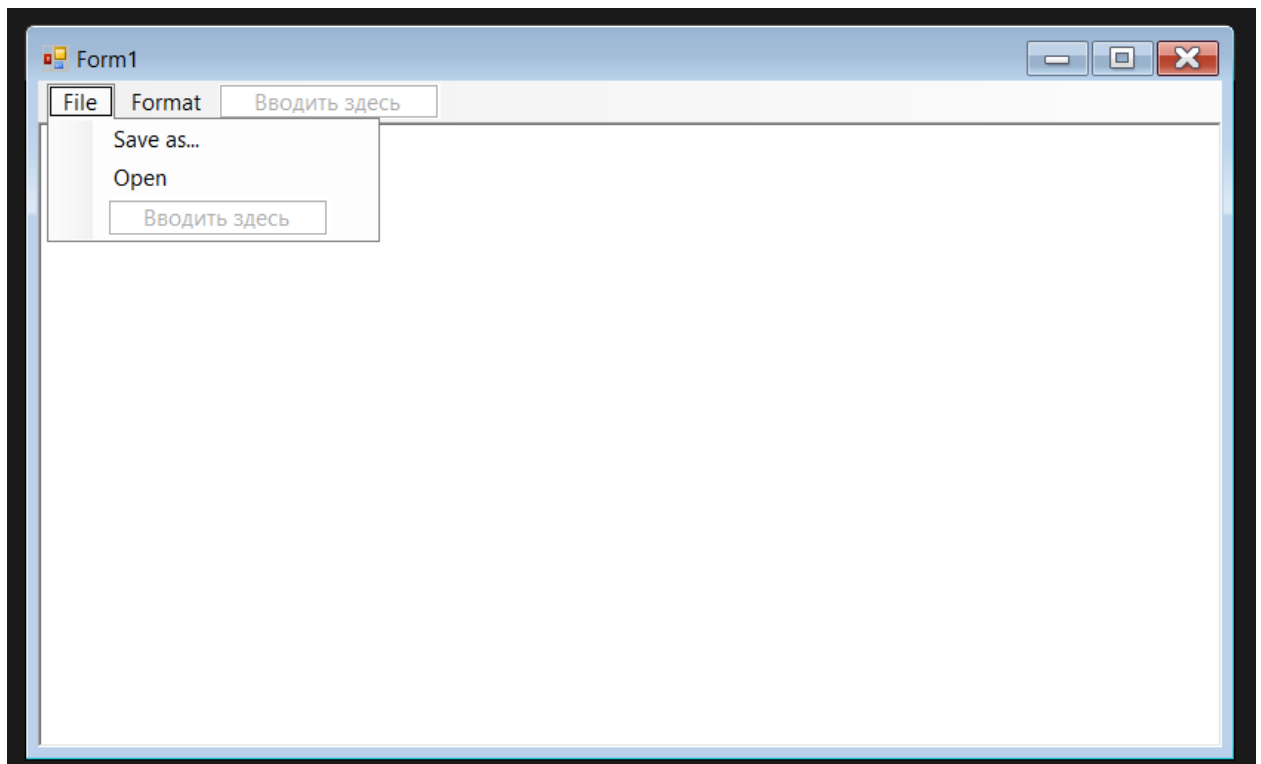


Рис. 1. Создание простой формы с выпадающим меню

```

private void сохранитьКакToolStripMenuItem_Click(object sender, EventArgs e)
{
    saveFileDialog1.Filter = "txt files (*.txt)|*.txt";
    if (saveFileDialog1.ShowDialog() ==
        System.Windows.Forms.DialogResult.OK
        && saveFileDialog1.FileName.Length > 0)
    {
        richTextBox1.SaveFile(saveFileDialog1.FileName,
            RichTextBoxStreamType.PlainText);
    }
}

```

Рис. 2. Создание кода в обработчике события, чтобы кнопка могла сохранять

Упражнение 2:

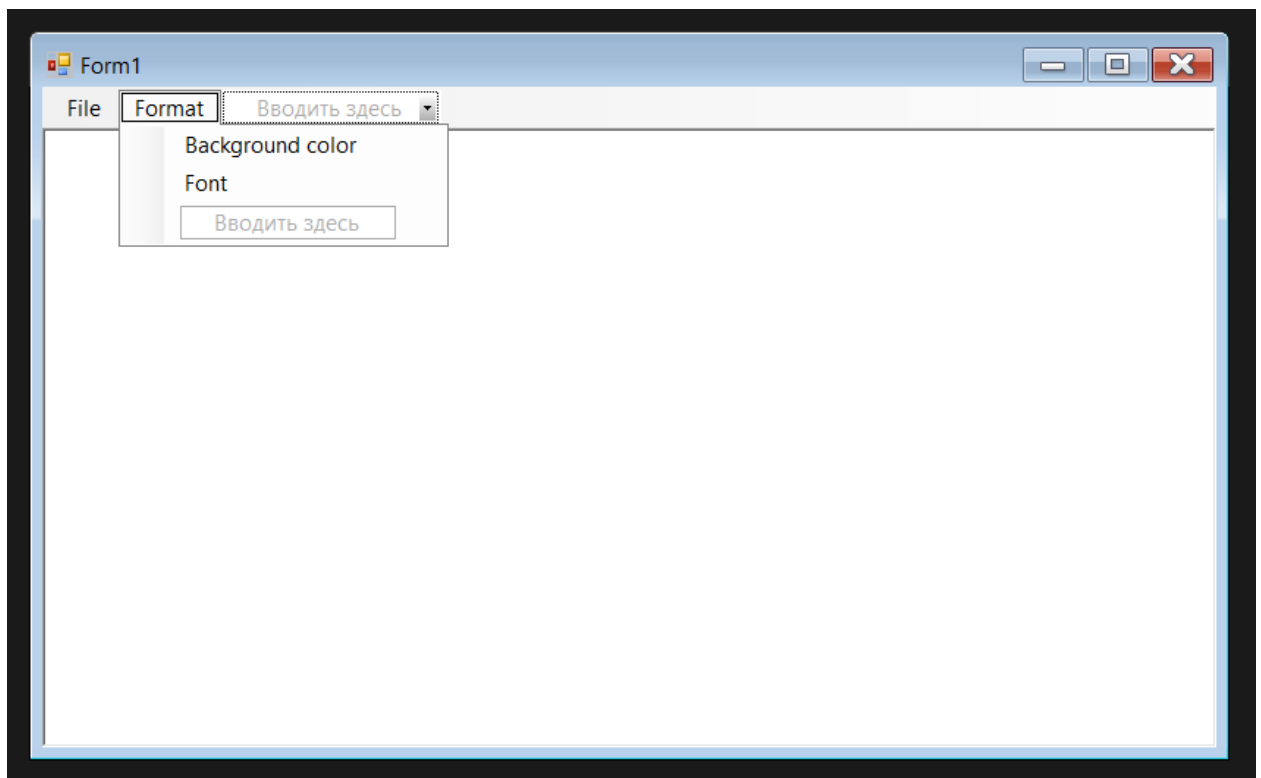


Рис. 3. Создание еще одного пункта меню с выпадающим списком

```

Ссылка: 1
private void цветФонаToolStripMenuItem_Click(object sender, EventArgs e)
{
    if (colorDialog1.ShowDialog() == DialogResult.OK)
    {
        richTextBox1.BackColor = colorDialog1.Color;
    }
}

```

Рис. 4. Обработчик события для кнопки, при нажатии которой выпадает меню, чтобы поменять цвет фона

Упражнение 3:

```
Ссылка: 1
private void шрифтToolStripMenuItem_Click(object sender, EventArgs e)
{
    if (fontDialog1.ShowDialog() == DialogResult.OK)
    {
        richTextBox1.Font = fontDialog1.Font;
    }
}
```

Рис. 5. Создание кода для обработчика события для кнопки при нажатии которой открывается меню, чтобы поменять шрифт

Упражнение 4:

```
Ссылка: 1
private void открытьToolStripMenuItem_Click(object sender, EventArgs e)
{
    Stream myStream = null;
    OpenFileDialog openFileDialog1 = new OpenFileDialog();
    openFileDialog1.InitialDirectory = @"c:\\";
    openFileDialog1.Filter = "txt files (*.txt)|*.txt|All files (*.*) | *.* ";
    openFileDialog1.FilterIndex = 2;
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        try
        {
            if ((myStream = openFileDialog1.OpenFile()) != null)
            {
                using (myStream)
                {
                    richTextBox1.LoadFile(openFileDialog1.FileName,
                                           RichTextBoxStreamType.PlainText);
                }
            }
        }
        catch (Exception ex)
        {
            MessageBox.Show("Error: Could not read file from disk: "
                             + ex.Message);
        }
    }
}
```

Рис. 6. Создание кода в обработчике события для кнопки при нажатии которой можно открыть файл

Дополнительное упражнение (от Рыбакова):

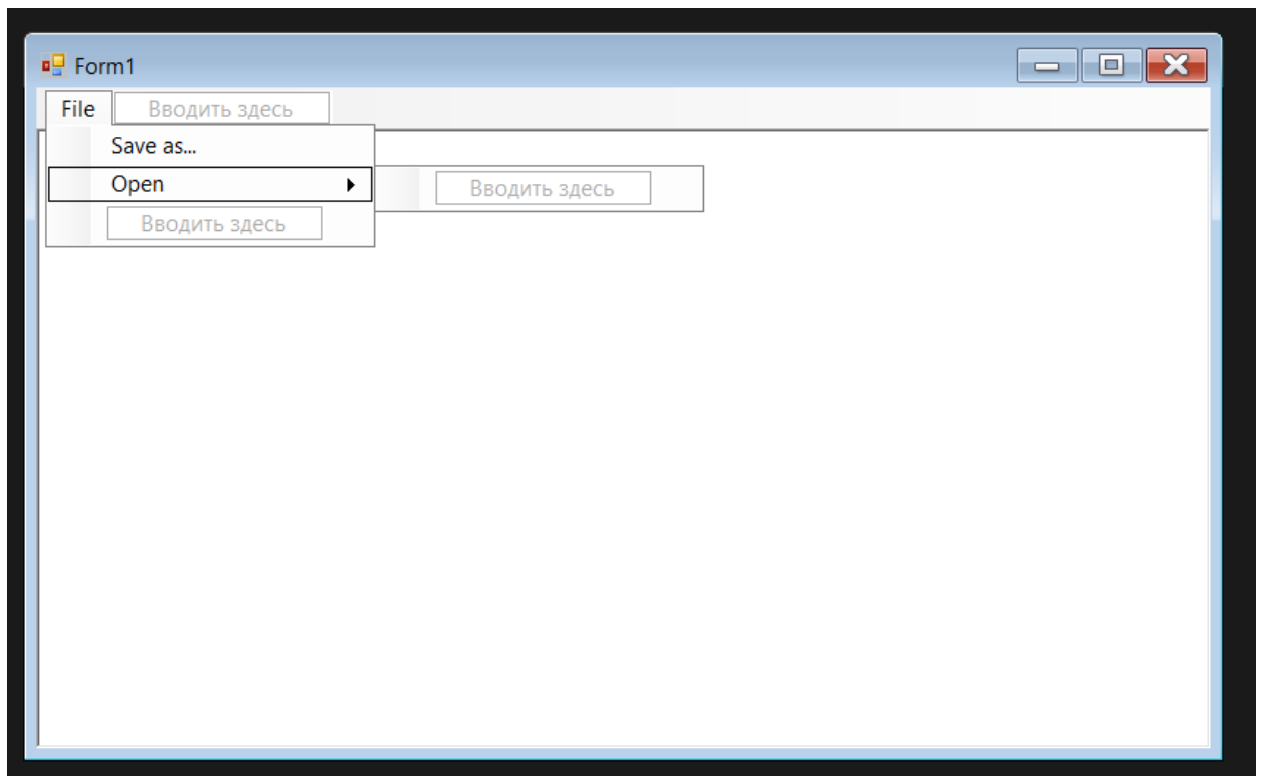


Рис. 7. Создание простого интерфейса формы

```

Ссылка: 1
private void saveAsToolStripMenuItem_Click(object sender, EventArgs e)
{
    // Проверяем, нажал ли пользователь ОК в окне сохранения
    if (saveFileDialog1.ShowDialog() == DialogResult.OK)
    {
        // Получаем путь к файлу, который выбрал пользователь
        string fileName = saveFileDialog1.FileName;

        // Пишем текст из нашего поля в файл
        System.IO.File.WriteAllText(fileName, richTextBox1.Text);

        MessageBox.Show("Файл успешно сохранен в формате .mylaba!");
    }
}

```

Рис. 8. Создание кода для кнопки Сохранить как...

Ссылка: 1

```
private void openToolStripMenuItem_Click(object sender, EventArgs e)
{
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        string fileName = openFileDialog1.FileName;

        // Читаем всё содержимое файла
        string readText = System.IO.File.ReadAllText(fileName);

        // Выводим этот текст в наше окно
        richTextBox1.Text = readText;
    }
}
```

Рис. 9. Создание кода для кнопки Открыть

DerferenceLinks	true
Filter	Мои Файлы (*.mylaba) *.mylaba Все файлы (*.*) *.*
FilterIndex	1

Рис. 10. Создание фильтра для формата файла

Вывод:

После этой лабы приложение стало похоже на реальный софт. Главное, что я понял: не нужно изобретать велосипед и рисовать свои окна для выбора папок или цветов - в Windows всё уже есть готовое, надо просто уметь это вызывать. Также разобрался, как связывать пункты меню с функциями, чтобы код не превращался в кашу, и как делать интерфейс интуитивно понятным для пользователя.