№ 9Создание WPFтекстового редактора

**Задание**

Создать многодокументный текстовый редактор на основе шаблона WPF. Требования:

1) Должен поддерживать динамический интерфейс. Панели инструментов, меню, основное окно и др. элементы должны изменяться пропорционально размерам окна. Используйте вложенные контейнеры компоновки (Grid, StrackPanel, DockPanel и т.д.) и свойства.

2) Содержит меню (

<Menu>

<MenuItem Header="\_File" x:Name="File">

<MenuItem Header=" \_Save" x:Name="Save"/>...)

и дублирующую панель инструментов: ***File***  ***New***, ***Save, Open*** (используйте стандартные диалоги открытия OpenFileDialog и сохранения текстового файла SaveFileDialog), ***Edit*** ***Copy, Paste*** (поддержка работы с буфером обмена, кроме того ЭУ RichTextBox поддерживает методы Copy(), Paste(), Undo(), Cut(), SpellCheck, форматы, выделения, параграфы, блоки, стили, поиск и замену и т.д), Font и ***Color*** (реализует стандартные диалоги настройки шрифта и выбора цвета для **выделенного фрагмента**) а также Combox (содержит перечисление из 4-6 шрифтов) и Slider (устанавливает размер шрифта) и три ToggleButton, которые устанавливают FontStyles жирный, курсив и подчеркивание.

<ToolBar>

<Button x:Name="\_Save" ToolTip="Save" Command="Edit.Save">

<Image Source="files/3.jpg" Height="20" Width="20" />

</Button>

<ToggleButton> ...

Добавьте строку состояния с информацией о числе символов (слов) и slider -ом масштабирования.

Можно использовать Ribbon control из библиотеки RibbonControlsLibrary.dll

Все действия редактора должны быть реализованы через Command.

3) В области ввода текста (проще использовать ЭУ RichTextBox) должны поддерживаться drag-and-drop (например, перетаскивание файла в редактор или текстового блока).

4) При создании нового окна к заголовку окна последовательно добавляется номер созданного окна и присваивается имя по умолчанию (например: Window1 Window2 и т.д.). При загрузке файла в окно – в заголовке отображается полный путь и имя

файла. Кроме этого, окно должно содержать контекстное меню с командами : ***Close*** (текущее окно) и ***Copy, Delete*** и т.д.

5) Добавьте поддержку двух языков интерфейса (например, русский и английский). Должна поддерживаться динамическая локализация на основе Resource Dictionary.

6) Подгрузить в редактор свой курсор и нарисованную свою иконку для приложения.

7) Создайте **стиль** для вашего редактора и примените его.

Пример:

**Стандартные классы диалогов**

OpenFileDialog или SaveFileDialog используются для указания местоположения и имени файла, а затем текст загружается или сохраняется с помощью объекта TextRange, который мы получаем непосредственно из RichTextBox. Он обеспечивает доступ к физическому файлу. Файл загружается и сохраняется в формате RTF, но вы можете указать один из других форматов.

using Microsoft.Win32;

OpenFileDialog dlg = new OpenFileDialog();

dlg.Filter = "Rich Text Format (\*.rtf)|\*.rtf|All files (\*.\*)|\*.\*";

if (dlg.ShowDialog() == true)

{

FileStream fileStream = new FileStream(dlg.FileName, FileMode.Open);

TextRange range = new TextRange(rtbEditor.Document.ContentStart, rtbEditor.Document.ContentEnd);

range.Load(fileStream, DataFormats.Rtf);

}

SaveFileDialog dlg = new SaveFileDialog();

dlg.Filter = "Rich Text Format (\*.rtf)|\*.rtf|All files (\*.\*)|\*.\*";

if (dlg.ShowDialog() == true)

{

FileStream fileStream = new FileStream(dlg.FileName, FileMode.Create);

TextRange range = new TextRange(rtbEditor.Document.ContentStart, rtbEditor.Document.ContentEnd);

range.Save(fileStream, DataFormats.Rtf);

}