

Задача 2: Опростен текстов редактор

1. Създайте в работната си папка нов проект на C# от тип Windows Application с име text_edit.
2. Добавете подходящите елементи, така че да се получи следния интерфейс (за по-удобно първо настройте размерите на отделните елементи и чак след това задайте размерите на формата):



3. Настройки на формата:
 - Стартова позиция – в центъра на екрана
 - Размери – 750 x 500 (включително за минимален и максимален)
 - MaximizeBox и MinimizeBox -> False (така изключваме съответните бутони)
 - Текст в заглавната лента – „Опростен текстов редактор“
 - Икона – създайте подходяща с <http://www.favicon.cc/>
4. Текстовото поле в центъра (RichTextBox):
 - **име** на полето (name) -> pole
 - **шрифт** - Microsoft Sans Serif, размер на символите – 14, първоначално без текст
 - **размери** – 730 x 420, ако не можете да го позиционирате добре, задайте му в свойство location – 2 за X и 40 за Y
 - **TabIndex** -> 0 (това ще е активния елемент при стартиране, при задаване на 1, 2, 3 и т.н се определя кои да са следващите активни елементи при натискане на TAB)

5. Падащия списък (**comboBox**):

- **име** на полето (name) -> comboBoxFont
- **шрифт** - Microsoft Sans Serif, размер на символите – 12
- **ширина** (Size->Width) -> 130, ако не можете да го позиционирате добре, задайте му в свойство location – 4 за X и 5 за Y
- **TabIndex** -> 1
- **елементи** – в свойство Items добавете на отделни редове избрани от Вас шрифтове (например Arial, Jokerman, Microsoft Sans Serif, Tahoma), за целта вижте наличните от свойство Font

6. Бутончетата (**Button**):

- **имена** (name) -> buttonBold, buttonItalic, buttonUnderline, за цветните бутони според избрания цвят, в примера са: buttonBlack, buttonIndigo, buttonDarkGreen, buttonDarkRed, buttonDarkBlue, buttonWhite, buttonLavender, buttonAliceBlue, buttonLavenderBlush и buttonBeige
- **шрифт** (за B, I, U) - Microsoft Sans Serif, размер на символите – 12, удебелен
- **текст** – B, I, U за първите три и без текст за цветните
- **позиция на текста** (TextAlign) за първите три -> MiddleCenter
- **фон** (BackColor) за цветните бутони – избирайте от раздела Web, това са именуваните цветове (промяната на цвят с код става най-лесно като се посочи името на цвета). В примера цветовете са съответно: Black, Indigo, DarkGreen, DarkRed, DarkBlue, White, Lavender, AliceBlue, LavenderBlush и Beige
- **размери** – 30 x 30, ако не можете да го позиционирате добре, задайте в свойство **location** – 5 за Y, а за X съответно -> 140, 170, 200, 355, 385, 415, 445, 475, 580, 610, 640, 670, 700
(Друг начин е за всяка от групите бутончета да позиционирате с мишката най-левия и най-десния където искате да бъдат, след което да маркирате цялата група с левия бутон на мишката като задържате Ctrl или Shift и да използвате командата Format / Horizontal Spacing / Make Equal)
- **TabIndex** -> от 2 до 14 в реда, в който са бутоните

7. Двата етикета (**Label**):

- **шрифт** - Microsoft Sans Serif, размер на символите – 10
- **текст** – “Цвят на текста: ” и „Фон: “
- **позиция на текста** (TextAlign) -> MiddleCenter
- **размери** – 120 x 35 за първия, 40 x 35 за втория, ако не можете да го позиционирате добре, задайте в свойство **location** – 2 за Y, а за X съответно -> 230 и 540
TabIndex -> 15 и 16 (понеже това не са активни елементи, те ще бъдат прескачани, но въпреки това е добре да им бъдат зададени индекси различни от останалите елементи)

8. Действия:

- При натискане на цветните бутони да се сменя цвета на текста или фона според това кой бутон е натиснат, например:

```
pole.ForeColor=Color.Black; //за черни букви
```

```
pole.BackColor=Color.White; //за бял фон
```

- При натискане на първите три бутона текста да става съответно удебелен, наклонен или подчертан, например:

```
pole.Font=new Font(pole.Font, FontStyle.Bold); //за удебеляване
```

- При избор на шрифт от падащия списък да се променя шрифта на текста. Това става със следния код:

```
var izbran=comboBoxFont.Text;
```

```
pole.Font=new Font(izbran, 14);
```

9. Възможности за подобряване (опитайте сами):

- Добавяне на help бутон и евентуално подходящ текст към него
- Добавяне на списъчно поле за промяна на размера на текста
- Смяна на външния вид на бутоните при натискане с мишката за да се показва кой е „включен“ в момента.
- Добавяне на бутон за изтриване на текста.
- Подобряване на кода, така че текста да може да е едновременно удебелен / подчертан / наклонен. (В горния вариант може да е включена само една от тези опции).
- Като бъдеща доработка – добавяне на лента с менюта или втора лента с бутони, даващи възможност за записване, зареждане, затваряне на текстов файл, копиране, изрязване, поставяне на текст, работа с части от текста и други възможности за които можете да се сетите 😊

10. Целия код, за който не може да се справи. Да напомним, че всяка от долните функции бива създадена автоматично при двукратно кликване с левия бутон на мишката върху желанния елемент, след което трябва да дописвате само това което е между двете фигурни скоби -> { и } :

```
void ButtonBlackClick(object sender, EventArgs e)
{
    pole.ForeColor=Color.Black;
}
void ButtonIndigoClick(object sender, EventArgs e)
{
    pole.ForeColor=Color.Indigo;
}
void ButtonDarkGreenClick(object sender, EventArgs e)
```

```
{
    pole.ForeColor=Color.DarkGreen;
}
void ButtonDarkRedClick(object sender, EventArgs e)
{
    pole.ForeColor=Color.DarkRed;
}
void ButtonDarkBlueClick(object sender, EventArgs e)
{
    pole.ForeColor=Color.DarkBlue;
}
void ButtonWhiteClick(object sender, EventArgs e)
{
    pole.BackColor=Color.White;
}
void ButtonLavenderClick(object sender, EventArgs e)
{
    pole.BackColor=Color.Lavender;
}
void ButtonAliceBlueClick(object sender, EventArgs e)
{
    pole.BackColor=Color.AliceBlue;
}
void ButtonLavenderBlushClick(object sender, EventArgs e)
{
    pole.BackColor=Color.LavenderBlush;
}
void ButtonBeigeClick(object sender, EventArgs e)
{
    pole.BackColor=Color.Beige;
}
void ButtonBoldClick(object sender, EventArgs e)
{
    pole.Font=new Font(pole.Font, FontStyle.Bold);
}
void ButtonItalicClick(object sender, EventArgs e)
{
    pole.Font=new Font(pole.Font, FontStyle.Italic);
}
void ButtonUnderlineClick(object sender, EventArgs e)
{
    pole.Font=new Font(pole.Font, FontStyle.Underline);
}
void ComboBoxFontSelectedIndexChanged(object sender, EventArgs e)
{
    var izbran=comboBoxFont.Text;
    pole.Font=new Font(izbran, 14);
}
```