

五專資工三甲

# 視窗程式設計

第六次作業

學號：5B0G0007

姓名：尤苡恬

## 作業說明：

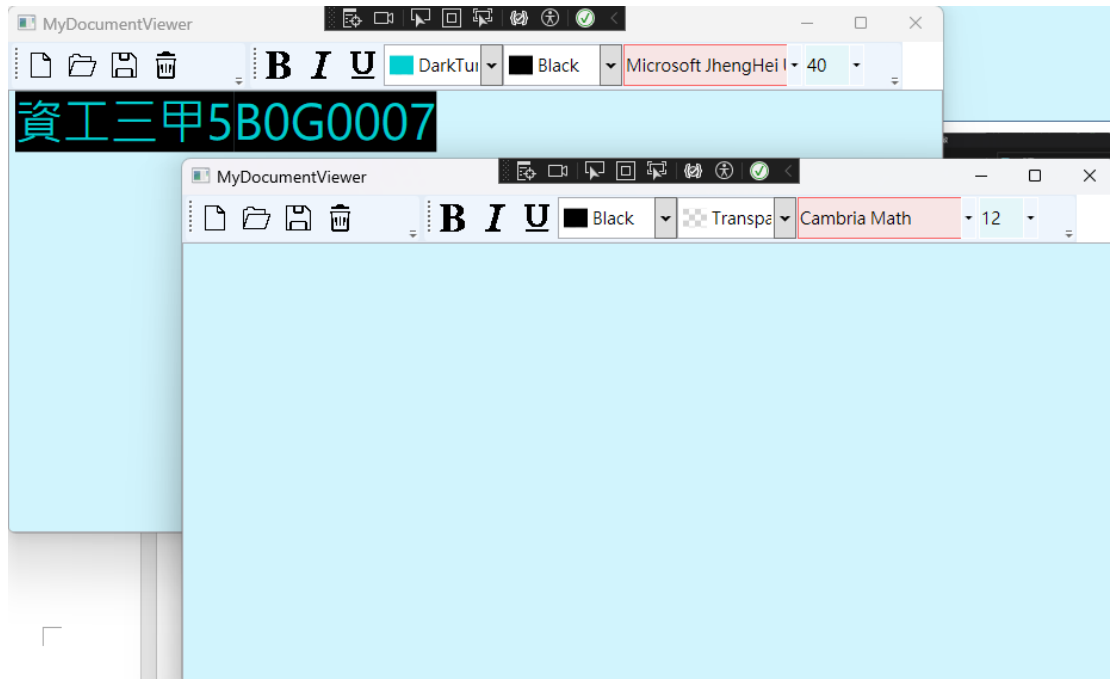
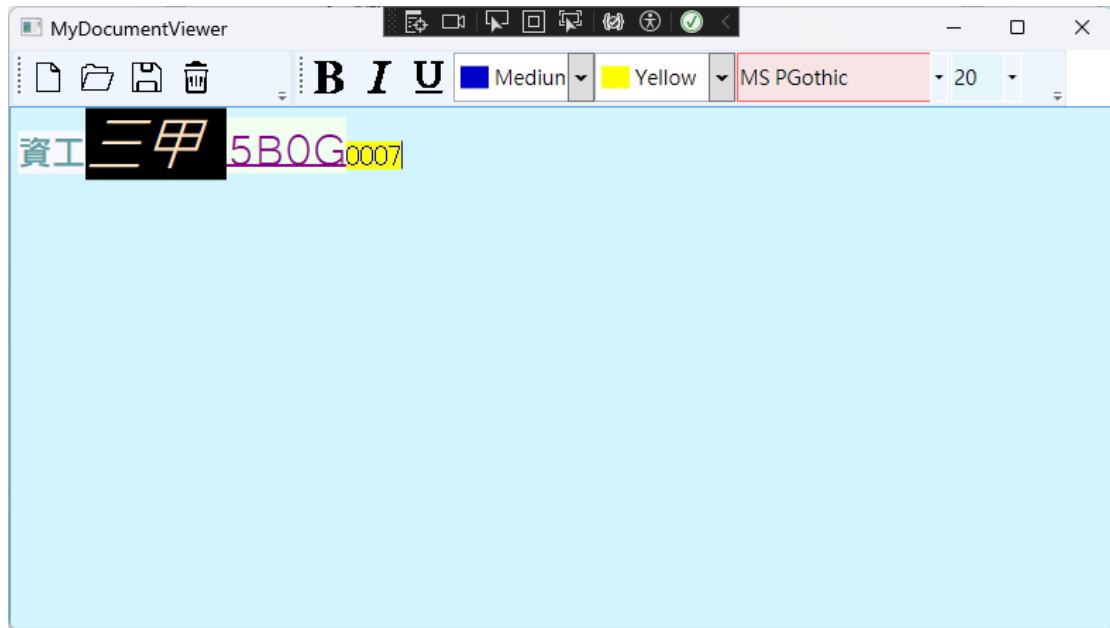
### 作業 6：記事本

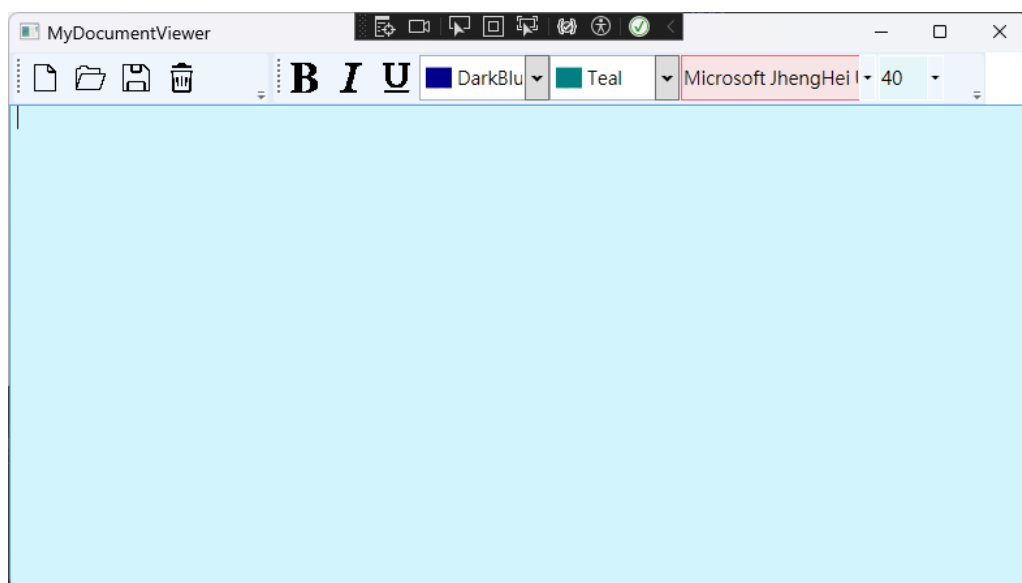
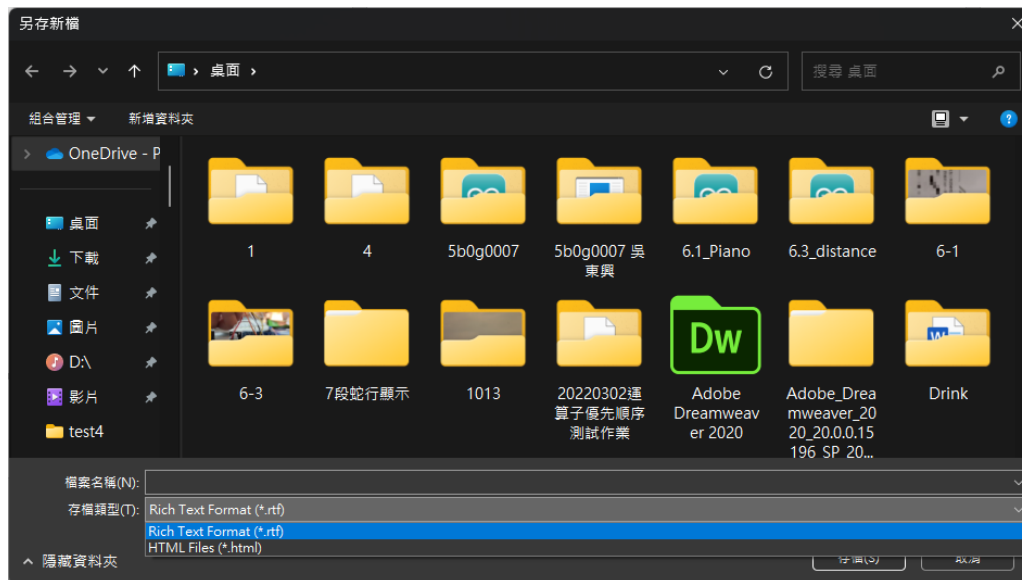
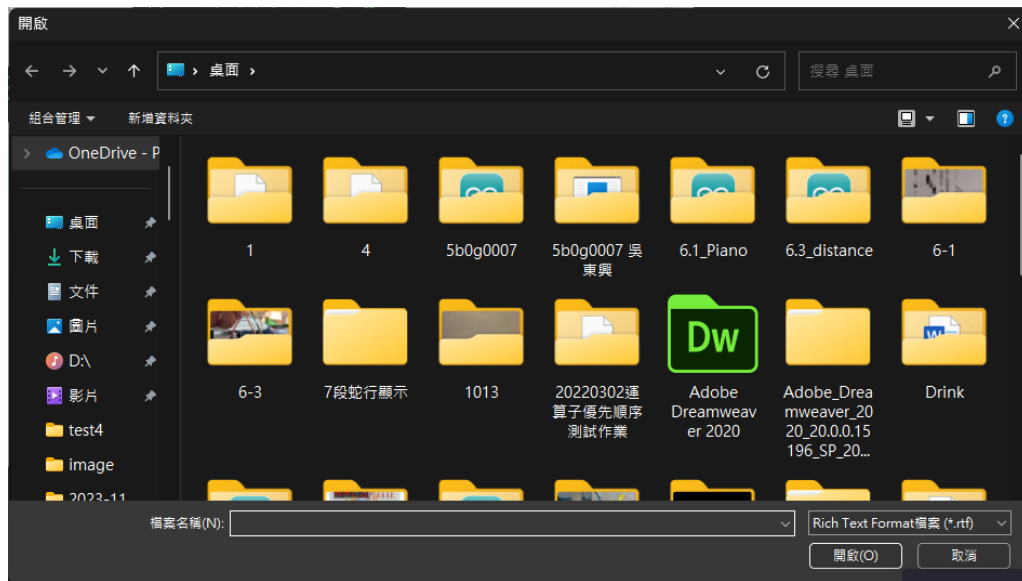
請依照上課範例，完成一個精簡版的 WPF 記事本應用程式，你的程式要具備以下功能：

- 開新檔案
- 開啟舊檔
- 儲存檔案成為 rtf 與 **html 格式(新增功能)**
- 改變字型與字體大小
- 改變字體外觀(粗體、斜體、底線)
- 變換字體顏色與**背景顏色(新增功能)**
- 清除全部內容

## 執行程式：







程式碼：

```
1 <Window x:Class="_2023_WpfApp4.MainWindow"
2     xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
3     xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
4     xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
5     xmlns:mc="http://schemas.openxmlformats.org/markup-
    compatibility/2006"
6     xmlns:local="clr-namespace:_2023_WpfApp4"
7     mc:Ignorable="d"
8     Title="2023小作家" Height="450" Width="800">
9     <Grid>
10        <Button x:Name="openButton" Content="開啟新文件"
            HorizontalAlignment="Left" Margin="110,85,0,0"
            VerticalAlignment="Top" Height="75" Width="140" FontSize="18"
            Click="openButton_Click"/>
11
12    </Grid>
13 </Window>
14
```

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6 using System.Windows;
7 using System.Windows.Controls;
8 using System.Windows.Data;
9 using System.Windows.Documents;
10 using System.Windows.Input;
11 using System.Windows.Media;
12 using System.Windows.Media.Imaging;
13 using System.Windows.Navigation;
14 using System.Windows.Shapes;
15
16 namespace _2023_WpfApp4
17 {
18     /// <summary>
19     /// Interaction logic for MainWindow.xaml
20     /// </summary>
21     public partial class MainWindow : Window
22     {
23         public MainWindow()
24         {
25             InitializeComponent();
26         }
27
28         private void openButton_Click(object sender, RoutedEventArgs e)
29         {
30             MyDocumentViewer myDocumentViewer = new MyDocumentViewer();
31             myDocumentViewer.Show();
32         }
33     }
34 }
35
```

```
1 using Microsoft.Win32;
2 using System;
3 using System.Collections.Generic;
4 using System.IO;
5 using System.Windows;
6 using System.Windows.Documents;
7 using System.Windows.Input;
8 using System.Windows.Media;
9 using System.Xaml;
10
11 namespace _2023_WpfApp4
12 {
13     /// <summary>
14     /// MyDocumentViewer.xaml 的互動邏輯
15     /// </summary>
16     public partial class MyDocumentViewer : Window
17     {
18         Color fontColor = Colors.Black;
19         Color fontBackgroundColor = Colors.Transparent;
20         public MyDocumentViewer()
21         {
22             InitializeComponent();
23             fontColorPicker.SelectedColor = fontColor;
24             fontBackgroundColorPicker.SelectedColor = fontBackgroundColor;
25
26             foreach (FontFamily fontFamily in Fonts.SystemFontFamilies)
27             {
28                 fontFamilyComboBox.Items.Add(fontFamily);
29             }
30             fontFamilyComboBox.SelectedIndex = 8;
31
32             fontSizeComboBox.ItemsSource = new List<Double>() { 8, 9, 10, 12, 20, 24, 32, 40, 50, 60, 80, 90 };
33             fontSizeComboBox.SelectedIndex = 3;
34         }
35
36         private void NewCommand_Executed(object sender, ExecutedRoutedEventArgs e)
37         {
38             // 在這裡實現「新建」的操作，例如打開一個新文件、清空文檔等
39             MyDocumentViewer myDocumentViewer = new MyDocumentViewer();
40             myDocumentViewer.Show();
41         }
42
43         private void OpenCommand_Executed(object sender, ExecutedRoutedEventArgs e)
44         {
45             OpenFileDialog openFileDialog = new OpenFileDialog();
46             openFileDialog.Filter = "Rich Text Format檔案|.rtf|所有檔案|*.*";
```

```
47         if (fileDialog.ShowDialog() == true)
48         {
49             TextRange range = new TextRange
                                     (rtbEditor.Document.ContentStart,
                                     rtbEditor.Document.ContentEnd);
50
51             using (FileStream fileStream = new FileStream
                                     (fileDialog.FileName, FileMode.Open))
52             {
53                 range.Load(fileStream, DataFormats.Rtf);
54             }
55         }
56     }
57
58     private void SaveCommand_Executed(object sender, RoutedEventArgs e)
59     {
60         SaveFileDialog saveFileDialog = new SaveFileDialog();
61         saveFileDialog.Filter = "Rich Text Format (*.rtf)|*.rtf|HTML
                                     Files (*.html)|*.html";
62         if (saveFileDialog.ShowDialog() == true)
63         {
64             string selectedFileType = Path.GetExtension
                                     (saveFileDialog.FileName).ToLowerInvariant();
65             if (selectedFileType == ".rtf")
66             {
67                 SaveAsRtf(saveFileDialog.FileName);
68             }
69             else if (selectedFileType == ".html")
70             {
71                 SaveAsHtml(saveFileDialog.FileName);
72             }
73         }
74     }
75
76     private void SaveAsRtf(string fileName)
77     {
78         using (FileStream fs = new FileStream(fileName,
                                     FileMode.Create))
79         {
80             TextRange range = new TextRange
                                     (rtbEditor.Document.ContentStart,
                                     rtbEditor.Document.ContentEnd);
81             range.Save(fs, DataFormats.Rtf);
82         }
83     }
84
85     private void SaveAsHtml(string fileName)
86     {
```



```

...owner\Desktop\writer\WpfApp1\MyDocumentViewer.xaml.cs 3
87      using (FileStream fs = new FileStream(fileName, ➤
           FileMode.Create))
88      {
89          TextRange range = new TextRange ➤
           (rtbEditor.Document.ContentStart, ➤
           rtbEditor.Document.ContentEnd);
90          range.Save(fs, DataFormats.Html);
91      }
92  }
93
94  private void rtbEditor_SelectionChanged(object sender, ➤
           RoutedEventArgs e)
95  {
96      //判斷選中的文字是否為粗體，並同步更新boldButton的狀態
97      object property = rtbEditor.Selection.GetPropertyValue ➤
           (TextElement.FontWeightProperty);
98      boldButton.IsChecked = (property is FontWeight && (FontWeight) ➤
           property == FontWeights.Bold);
99
100     //判斷選中的文字是否為斜體，並同步更新italicButton的狀態
101     property = rtbEditor.Selection.GetPropertyValue ➤
           (TextElement.FontStyleProperty);
102     italicButton.IsChecked = (property is FontStyle && (FontStyle) ➤
           property == FontStyles.Italic);
103
104     //判斷選中的文字是否有底線，並同步更新underlineButton的狀態
105     property = rtbEditor.Selection.GetPropertyValue ➤
           (Inline.TextDecorationsProperty);
106     underlineButton.IsChecked = (property != ➤
           DependencyProperty.UnsetValue && property == ➤
           TextDecorations.Underline);
107
108     //判斷選中的文字的字體，同步更新fontFamilyComboBox的狀態
109     property = rtbEditor.Selection.GetPropertyValue ➤
           (TextElement.FontFamilyProperty);
110     fontFamilyComboBox.SelectedItem = property;
111
112     //判斷所選中的文字的字體大小，同步更新fontSizeComboBox的狀態
113     property = rtbEditor.Selection.GetPropertyValue ➤
           (TextElement.FontSizeProperty);
114     fontSizeComboBox.SelectedItem = property;
115
116     //判斷所選中的文字的字體色彩，同步更新fontColorPicker的狀態
117     SolidColorBrush? foregroundProperty = ➤
           rtbEditor.Selection.GetPropertyValue ➤
           (TextElement.ForegroundProperty) as SolidColorBrush;
118     if (foregroundProperty != null)
119     {
120         fontColorPicker.SelectedColor = foregroundProperty.Color;

```

```
121     }
122     //判斷所選中的文字的字體色彩，同步更新fontBackgroundColorPicker的狀態
123     SolidColorBrush? backgroundProperty =
        rtbEditor.Selection.GetPropertyValue
        (TextElement.BackgroundProperty) as SolidColorBrush;
124
125     if (backgroundProperty != null)
126     {
127         fontBackgroundColorPicker.SelectedColor =
            backgroundProperty.Color;
128     }
129 }
130
131 private void fontColorPicker_SelectedColorChanged(object sender,
    RoutedPropertyChangedEventArgs<Color?> e)
132 {
133     fontColor = (Color)e.NewValue;
134     SolidColorBrush fontBrush = new SolidColorBrush(fontColor);
135     rtbEditor.Selection.ApplyPropertyValue
        (TextElement.ForegroundProperty, fontBrush);
136 }
137
138
139 private void fontBackgroundColorPicker_SelectedColorChanged(object
    sender, RoutedPropertyChangedEventArgs<Color?> e)
140 {
141     fontBackgroundColor = (Color)e.NewValue;
142     SolidColorBrush backgroundBrush = new SolidColorBrush
        (fontBackgroundColor);
143     rtbEditor.Selection.ApplyPropertyValue
        (TextElement.BackgroundProperty, backgroundBrush);
144 }
145
146 private void fontFamilyComboBox_SelectionChanged(object sender,
    System.Windows.Controls.SelectionChangedEventArgs e)
147 {
148     if (fontFamilyComboBox.SelectedItem != null)
149     {
150         rtbEditor.Selection.ApplyPropertyValue
            (TextElement.FontFamilyProperty,
            fontFamilyComboBox.SelectedItem);
151     }
152 }
153
154 private void fontSizeComboBox_SelectionChanged(object sender,
    System.Windows.Controls.SelectionChangedEventArgs e)
155 {
156     if (fontSizeComboBox.SelectedItem != null)
157     {
```

```
...owner\Desktop\writer\WpfApp1\MyDocumentViewer.xaml.cs 5
158         rtbEditor.Selection.ApplyPropertyValue          ↗
            (TextElement.FontSizeProperty,               ↗
            fontSizeComboBox.SelectedItem);
159     }
160 }
161
162 private void clearButton_Click(object sender, RoutedEventArgs e)
163 {
164     rtbEditor.Document.Blocks.Clear();
165 }
166 }
167 }
168
```

```
1 <Window x:Class="_2023_WpfApp4.MyDocumentViewer"
2     xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
3     xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
4     xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
5     xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
6     xmlns:xctk="http://schemas.xceed.com/wpf/xaml/toolkit"
7     xmlns:local="clr-namespace:_2023_WpfApp4"
8     mc:Ignorable="d"
9     Title="MyDocumentViewer" Height="450" Width="800">
10 <Window.CommandBindings>
11     <CommandBinding Command="ApplicationCommands.New"
12         Executed="NewCommand_Executed"/>
13     <CommandBinding Command="ApplicationCommands.Open"
14         Executed="OpenCommand_Executed"/>
15     <CommandBinding Command="ApplicationCommands.Save"
16         Executed="SaveCommand_Executed"/>
17 </Window.CommandBindings>
18 <DockPanel>
19     <ToolBarTray Height="40" VerticalAlignment="Top"
20         HorizontalAlignment="Left" DockPanel.Dock="Top"
21         Background=#FFE3ECF6">
22         <ToolBar x:Name="applicationToolBar" Height="40" Width="200">
23             <Button x:Name="newButton"
24                 Command="ApplicationCommands.New" Width="35" Height="35">
25                 <Image Width="23" Height="23" Source="/new.png"/>
26             </Button>
27             <Button x:Name="openButton"
28                 Command="ApplicationCommands.Open" Width="35"
29                 Height="35">
30                 <Image Width="23" Height="23" Source="/open.png"/>
31             </Button>
32             <Button x:Name="saveButton"
33                 Command="ApplicationCommands.Save" Width="35"
34                 Height="35">
35                 <Image Width="23" Height="23" Source="/save.png"/>
36             </Button>
37             <Button x:Name="clearButton" Width="35" Height="35"
38                 Click="clearButton_Click">
39                 <Image Width="23" Height="23" Source="/clear.png"/>
40             </Button>
41         </ToolBar>
42         <ToolBar x:Name="editingToolBar" Width="550">
43             <ToggleButton x:Name="boldButton"
44                 Command="EditingCommands.ToggleBold" Height="35"
45                 Width="35">
46                 <Image Width="23" Height="23" Source="/bold.png"/>
47             </ToggleButton>
48             <ToggleButton x:Name="italicButton"
```

```

        Command="EditingCommands.ToggleItalic" Height="35"
        Width="35">
36         <Image Width="23" Height="23" Source="/italic.png"/>
37     </ToggleButton>
38     <ToggleButton x:Name="underlineButton"
        Command="EditingCommands.ToggleUnderline" Height="35"
        Width="35">
39         <Image Width="23" Height="23" Source="/underlined.png"/
        >
40     </ToggleButton>
41     <xctk:ColorPicker x:Name="fontColorPicker" Width="100"
        Height="35" VerticalAlignment="Center"
        HorizontalAlignment="Left" DisplayColorAndName="True"
        SelectedColorChanged="fontColorPicker_SelectedColorChange
        d" />
42     <xctk:ColorPicker x:Name="fontBackgroundColorPicker"
        Width="100" Height="35" VerticalAlignment="Center"
        HorizontalAlignment="Left" DisplayColorAndName="True"
        SelectedColorChanged="fontBackgroundColorPicker_SelectedC
        olorChanged" />
43     <ComboBox x:Name="fontFamilyComboBox" Height="35"
        Width="150" VerticalAlignment="Center"
        HorizontalAlignment="Left" BorderBrush="#FFFA6666"
        Background="#FFF7E5E5"
        SelectionChanged="fontFamilyComboBox_SelectionChanged"/>
44     <ComboBox x:Name="fontSizeComboBox" Height="35" Width="50"
        VerticalAlignment="Center" HorizontalAlignment="Left"
        Background="#FFE4F6F9"
        SelectionChanged="fontSizeComboBox_SelectionChanged"/>
45     </ToolBar>
46 </ToolBarTray>
47 <RichTextBox x:Name="rtbEditor"
        SelectionChanged="rtbEditor_SelectionChanged"
        Background="#FFD1F4FD">
48     <FlowDocument/>
49 </RichTextBox>
50 </DockPanel>
51 </Window>
52

```