

```
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
10 namespace WpfApp1
11 {
12     /// <summary>
13     /// MyDocumentViewer.xaml 的互動邏輯
14     /// </summary>
15     public partial class MyDocumentViewer : Window
16     {
17         Color fontColor = Colors.Blue;
18         Color backgroundColor = Colors.LightYellow;
19         public MyDocumentViewer()
20         {
21             InitializeComponent();
22             fontColorPicker.SelectedColor = fontColor;
23             backgroundColorPicker.SelectedColor =      ↗
                backgroundColor;
24
25             foreach (FontFamily fontFamily in      ↗
                Fonts.SystemFontFamilies)
26             {
27                 fontFamilyComboBox.Items.Add(fontFamily);
28             }
29             fontFamilyComboBox.SelectedIndex = 8;
30
31             fontSizeComboBox.ItemsSource = new List<Double>()      ↗
                { 8, 9, 10, 12, 20, 24, 32, 40, 50, 60, 80,      ↗
                90 };
32             fontSizeComboBox.SelectedIndex = 3;
33         }
34
35         private void NewCommand_Executed(object sender,      ↗
            ExecutedRoutedEventArgs e)
36         {
37             // 在這裡實現「新建」的操作，例如打開一個新文件、清空文檔      ↗
            等
38             MyDocumentViewer myDocumentViewer = new      ↗
                MyDocumentViewer();
39             myDocumentViewer.Show();
40         }
```

```
41     private void OpenCommand_Executed(object sender,      ↗
        ExecutedRoutedEventArgs e)
42     {
43         OpenFileDialog fileDialog = new OpenFileDialog();
44         fileDialog.Filter = "RTF 檔案|*.rtf|HTML 檔案|      ↗
            *.html|所有檔案|*.*";
45
46         if (fileDialog.ShowDialog() == true)
47         {
48             string fileName = fileDialog.FileName;
49             string fileExtension = Path.GetExtension      ↗
                (fileName);
50
51             TextRange range = new TextRange      ↗
                (rtbEditor.Document.ContentStart,
                rtbEditor.Document.ContentEnd);
52
53             if (fileExtension.Equals(".rtf",      ↗
                StringComparison.OrdinalIgnoreCase))
54             {
55                 // RTF 格式
56                 using (FileStream fileStream = new      ↗
                    FileStream(fileName, FileMode.Open))
57                 {
58                     range.Load(fileStream,      ↗
                        DataFormats.Rtf);
59                 }
60             }
61             else if (fileExtension.Equals(".html",      ↗
                StringComparison.OrdinalIgnoreCase))
62             {
63                 // HTML 格式
64                 using (FileStream fileStream = new      ↗
                    FileStream(fileName, FileMode.Open))
65                 {
66                     range.Load(fileStream,      ↗
                        DataFormats.Html);
67                 }
68             }
69             else
70             {
71                 MessageBox.Show("不支援的檔案格式");
72             }
73         }
74     }
75
76     private void SaveCommand_Executed(object sender,      ↗
```

```
ExecutedRoutedEventArgs e)
77     {
78         SaveFileDialog fileDialog = new SaveFileDialog();
79         fileDialog.Filter = "RTF 檔案|*.rtf|HTML 檔案|
            *.html|所有檔案|*.*";
80
81         if (fileDialog.ShowDialog() == true)
82         {
83             string fileName = fileDialog.FileName;
84             string fileExtension = Path.GetExtension
            (fileName);
85
86             TextRange range = new TextRange
            (rtbEditor.Document.ContentStart,
            rtbEditor.Document.ContentEnd);
87
88             if (fileExtension.Equals(".rtf",
            StringComparison.OrdinalIgnoreCase))
89             {
90                 // RTF 格式
91                 using (FileStream fileStream = new
            FileStream(fileName, FileMode.Create))
92                 {
93                     range.Save(fileStream,
            DataFormats.Rtf);
94                 }
95             }
96             else if (fileExtension.Equals(".html",
            StringComparison.OrdinalIgnoreCase))
97             {
98                 // HTML 格式
99                 using (FileStream fileStream = new
            FileStream(fileName, FileMode.Create))
100                 {
101                     range.Save(fileStream,
            DataFormats.Html);
102                 }
103             }
104             else
105             {
106                 MessageBox.Show("不支援的檔案格式");
107             }
108         }
109     }
110
111     private void rtbEditor_SelectionChanged(object
```

```

sender, RoutedEventArgs e)
113     {
114         //判斷選中的文字是否為粗體，並同步更新boldButton的狀態
115         object property =
116             rtbEditor.Selection.GetPropertyValue
117             (TextElement.FontWeightProperty);
118         boldButton.IsChecked = (property is FontWeight &&
119             (FontWeight)property == FontWeights.Bold);
120
121         //判斷選中的文字是否為斜體，並同步更新italicButton的狀態
122         property = rtbEditor.Selection.GetPropertyValue
123             (TextElement.FontStyleProperty);
124         italicButton.IsChecked = (property is FontStyle
125             && (FontStyle)property == FontStyles.Italic);
126
127         //判斷選中的文字是否有底線，並同步更新underlineButton的狀態
128         property = rtbEditor.Selection.GetPropertyValue
129             (Inline.TextDecorationsProperty);
130         underlineButton.IsChecked = (property !=
131             DependencyProperty.UnsetValue && property ==
132             TextDecorations.Underline);
133
134         //判斷選中的文字的字體，同步更新fontFamilyComboBox的狀態
135         property = rtbEditor.Selection.GetPropertyValue
136             (TextElement.FontFamilyProperty);
137         fontFamilyComboBox.SelectedItem = property;
138
139         //判斷所選中的文字的字體大小，同步更新fontSizeComboBox
140         //的狀態
141         property = rtbEditor.Selection.GetPropertyValue
142             (TextElement.FontSizeProperty);
143         fontSizeComboBox.SelectedItem = property;
144
145         //判斷所選中的文字的字體色彩，同步更新fontColorPicker的狀態
146         SolidColorBrush? foregroundProperty =
147             rtbEditor.Selection.GetPropertyValue
148             (TextElement.ForegroundProperty) as
149             SolidColorBrush;
150         if (foregroundProperty != null)
151         {
152             fontColorPicker.SelectedColor =
153                 foregroundProperty.Color;
154         }
155     }

```

```
140     }
141
142     private void fontColorPicker_SelectedColorChanged      ↗
        (object sender,                                     ↗
         RoutedPropertyChangedEventArgs<Color?> e)
143     {
144         fontColor = (Color)e.NewValue;
145         SolidColorBrush fontBrush = new SolidColorBrush   ↗
            (fontColor);
146         rtbEditor.Selection.ApplyPropertyValue             ↗
            (TextElement.ForegroundProperty, fontBrush);
147     }
148
149     private void fontFamilyComboBox_SelectionChanged      ↗
        (object sender,                                     ↗
         System.Windows.Controls.SelectionChangedEventArgs ↗
         e)
150     {
151         if (fontFamilyComboBox.SelectedItem != null)
152         {
153             rtbEditor.Selection.ApplyPropertyValue         ↗
                (TextElement.FontFamilyProperty,           ↗
                 fontFamilyComboBox.SelectedItem);
154         }
155     }
156
157     private void fontSizeComboBox_SelectionChanged(object ↗
        sender,                                             ↗
        System.Windows.Controls.SelectionChangedEventArgs ↗
        e)
158     {
159         if (fontSizeComboBox.SelectedItem != null)
160         {
161             rtbEditor.Selection.ApplyPropertyValue         ↗
                (TextElement.FontSizeProperty,             ↗
                 fontSizeComboBox.SelectedItem);
162         }
163     }
164
165     private void clearButton_Click(object sender,         ↗
        RoutedEventArgs e)
166     {
167         rtbEditor.Document.Blocks.Clear();
168     }
169
170     private void BackgroundColorPicker_SelectedColorChanged(object ↗
        BackgroundColorPicker_SelectedColorChanged(object ↗
```

```

        sender, RoutedPropertyChangedEventArgs<Color?> e)
171     {
172         if (backgroundColorPicker.SelectedColor.HasValue)
173         {
174             SolidColorBrush backgroundBrush = new SolidColorBrush
175                 SolidColorBrush
176                 (backgroundColorPicker.SelectedColor.Value);
177             rtbEditor.Background = backgroundBrush;
178         }
179     }
180 }

```