



VISUAL BASIC CREATING A TEXT EDITOR



By [thecodingguys](#)

This will teach you how to make a fully functional WordPad like program.

[Thecodingguys.net](#)

6/28/2012

© 2012 [thecodingguys.net](#)

CONTENTS

<u>ABOUT</u>	PAGE 1
<u>LICENSE</u>	PAGE 1
<u>PART A – DESIGN</u>	PAGE 2
<u>SAVE WORK – DIALOG</u>	PAGE 4
<u>OPEN WORK –DIALOG</u>	PAGE 5
<u>THE MAIN MENU</u>	PAGE 6
<u>THE TOOL BAR</u>	PAGE 7
<u>THE STATUS STRIP</u>	PAGE 9
<u>THE CONTEXT MENU STRIP</u>	PAGE 9
<u>THE RICHTEXTBOX</u>	PAGE 10
<u>THE TIMER</u>	PAGE 10
<u>PART B – CODING</u>	PAGE 11
<u>CREATING REGIONS</u>	PAGE 11
<u>MAIN MENU - FILE</u>	PAGE 12
<u>MAIN MENU – EDIT</u>	PAGE 13
<u>MAIN MENU – TOOLS</u>	PAGE 14
<u>TOOL BAR</u>	PAGE 15
<u>TOOL BAR – BOLD, ITALIC, UNDERLINE, STRIKETHROUGH</u>	PAGE 17
<u>TOOL BAR – ALIGN LEFT, RIGHT, CENTER, WORD CASE, AND ZOOM</u>	PAGE 18
<u>TOOL BAR – FONT SIZE AND FONT FAMILY</u>	PAGE 19
<u>RIGHT CLICK MENU</u>	PAGE 21
<u>YOU'RE DONE</u>	PAGE 22

**THIS PAGE HAS
BEEN LEFT BLANK**

(OK, I'll be honest I couldn't get the page numbers to work in Microsoft Office 2010, ☺)

ABOUT

©2012 thecodingguys.net

Published: 28, June 2012

Now that you have learned the basics of VB it is time to create a small project. We will create a text editor similar to windows WordPad. Since this will be a windows form application I hope you guys know how to navigate the IDE because I will not be teaching you that. I am sure you can find some useful resources on the Microsoft website.

I will split this up in two parts Part A will be design and Part B will be coding.

The source code and tutorial files for this can be found at <http://www.thecodingguys.net/downloads>

LICENSE

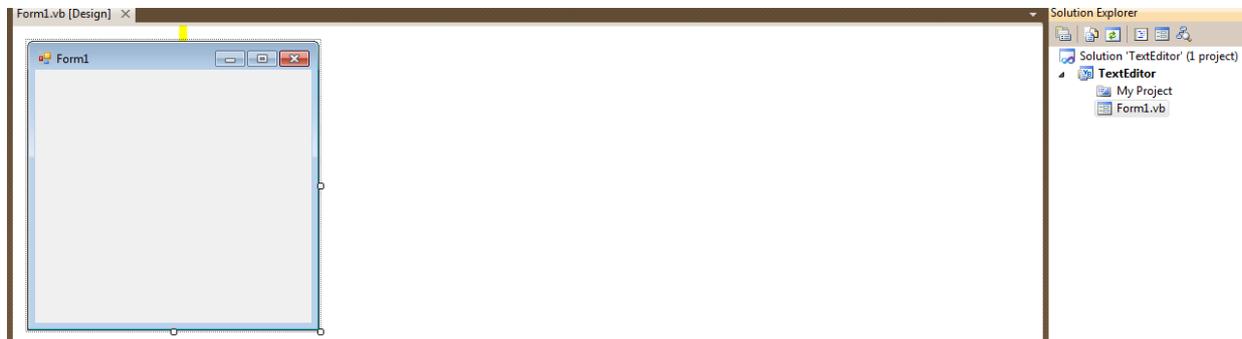
This work is licensed under the creative commons Attribution-[NonCommercial-NoDerivs 3.0 Unported](http://creativecommons.org/licenses/by-nc-nd/3.0/).

- You may not alter, transform, or build upon this work.
- You may not use this work for commercial purposes.
- You are free to copy, distribute and transmit the work

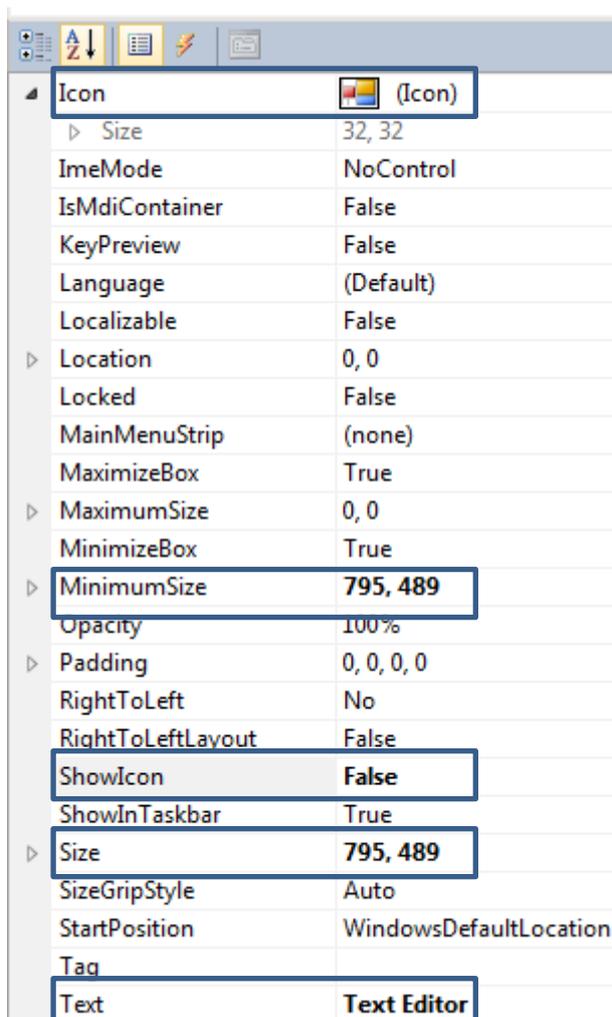
©2012 thecodingguys.net

PART A - DESIGN

Let's get started go ahead and create a new VB Windows Form Application. Name it TextEditor.



Once you create the form you get this, on the right in the Solution Explorer rename **Form1.vb** to **TextEditor.vb**. Next resize the form so it is around **795, 489**. Next make sure that the **minimum size** is also **795, 489**. Then change the **Form1 display name** to **Text Editor** and either choose an icon for your form or remove the icon. Here are the highlighted options to change.



It's totally up to you if you want to change the icon, I personally do not want to show the icon. Next we need to insert the controls for our form. We need to insert the main controls and rename them. In the tables below are the controls to insert and rename them to their new name. Rename them correctly as I have here. To rename a control click on it and in the properties scroll to the top until you see **(name)** and change it.

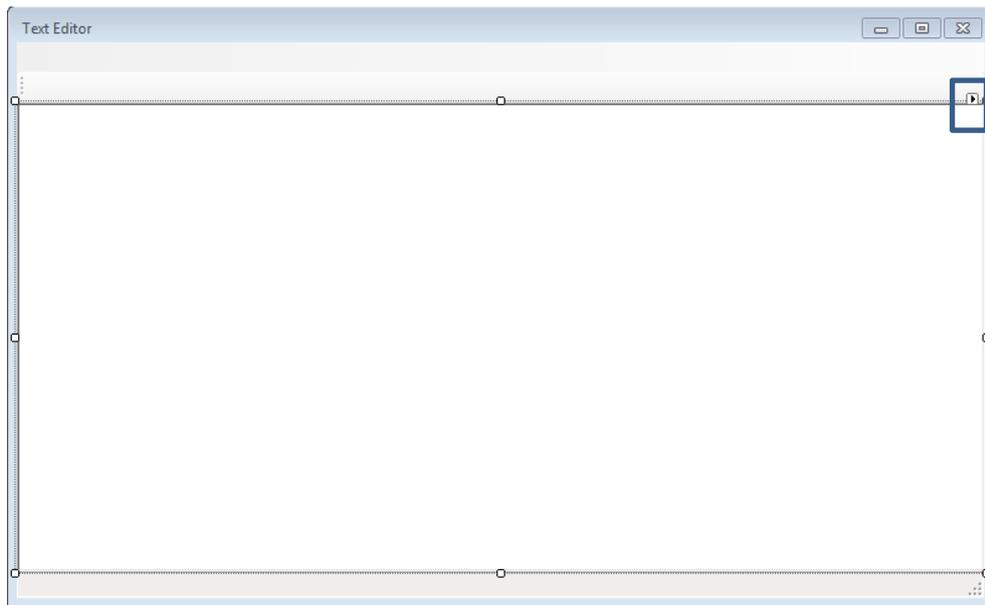
General Items to Insert

ITEM TO INSERT	RENAME IT TO
MenuStrip	mainMenu
ToolStrip	Tools
Richtextbox	Document
StatusStrip	Status

Dialogs to Insert (These dock at the bottom)

ITEM TO INSERT	RENAME IT TO
ContextMenuStrip	rcMenu
SaveFileDialog	saveWork
OpenFileDialog	openWork
Timer	Timer

The form should look like this:



Save Work Dialog

Click the save Work Dialog and apply these following settings highlighted:

(ApplicationSettings)	
(Name)	saveWork
AddExtension	True
AutoUpgradeEnabled	True
CheckFileExists	False
CheckPathExists	True
CreatePrompt	False
DefaultExt	
DereferenceLinks	True
FileName	
Filter	Text Files *.txt RTF Files *.rtf
FilterIndex	1
GenerateMember	True
InitialDirectory	
Modifiers	Friend
OverwritePrompt	True
RestoreDirectory	False
ShowHelp	False
SupportMultiDottedExtens:	False
Tag	
Title	Save Work
ValidateNames	True

Filter: The filter is just the formats you can save the work in. Copy this code;

Text Files |*.txt|RTF Files|*.rtf

Title: The title is the title of the save work dialog. You see this right at the top of the dialog.

Open Work Dialog

Click the open work dialog and apply these following settings highlighted:

(ApplicationSettings)	
(Name)	openWork
AddExtension	True
AutoUpgradeEnabled	True
CheckFileExists	True
CheckPathExists	True
DefaultExt	
DereferenceLinks	True
FileName	
Filter	Text Files (*.txt) RTF Files (*.rtf)
FilterIndex	1
GenerateMember	True
InitialDirectory	
Modifiers	Friend
Multiselect	False
ReadOnlyChecked	False
RestoreDirectory	False
ShowHelp	False
ShowReadOnly	False
SupportMultiDottedExtension	False
Tag	
Title	Open Work
ValidateNames	True

Filter: Is what I said above. This time it will only search for text files and rtf files to look for.

Title: Like above.

FileName: The default is OpenFileDialog, make sure this is blank!

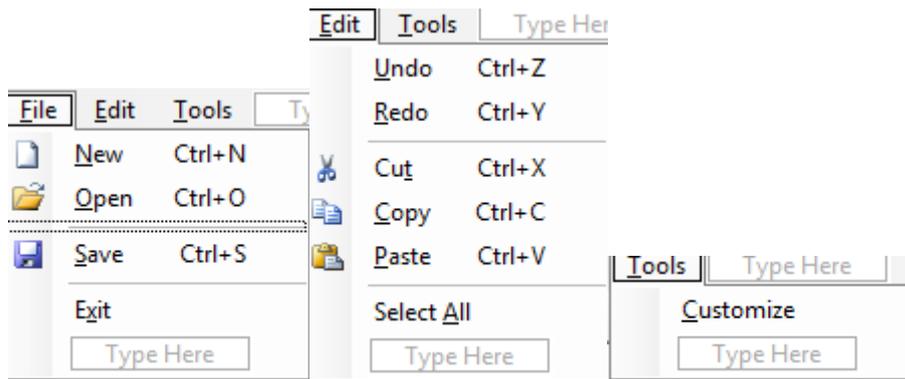
The Main Menu

OK next we will work our way down with the controls. First look at the image on page 4 you see that arrow in the upper right hand corner? We will use this to insert default controls. Click on the Menu Strip and you will see that arrow show, and then click **insert standard items**.

When you do you will get these items.



OK now on the menu bar we need to delete a few things. Click **Help** and delete it (press delete on your keyboard) then under **Tools** delete **Options**. Then under File delete **Print**, **Save As** and **Print Preview**. Now on the main menu bar you should only have, File > Edit and Tools and underneath them it should look like this:



Now what we need to do is name each of these items, yes there are a lot of things to rename! OK rename **File** to **mMFile**, rename **Edit** to **mMEdit** and rename **Tools** to **mMTools**. Next we need to rename the sub items under the menu. In the table below are the names for each new item.

UNDER FILE	UNDER EDIT	UNDER TOOLS
mMNew	mMUndo	mMCustomize
mMOpen	mMRedo	
mMSave	mMCut	
mMExit	mMCopy	
	mMPaste	
	mmSelectAll	

The Tool Bar

Now let's focus on the toolbar, click it and on the upper right hand side you should see that arrow again. Click the arrow and then click **insert standard items**. The tool bar should look like this:



Delete the Help Icon and the Printer Icon. Next on the menu bar see that drop down arrow? Click it and insert 11 buttons and two Combo Boxes, lay everything out so it looks like this:



Next we need to start editing these buttons. Click the first button that we inserted and change the **DisplayStyle** to Text, the **font** to **Arial** and style to bold, the **display text** to **B**. See the image below.

DisplayStyle	Text
DoubleClickEnabled	False
Enabled	True
Font	Arial, 9pt, style=Bold
ForeColor	ControlText
GenerateMember	True
Image	System.Drawing.Bitmap
ImageAlign	MiddleCenter
ImageScaling	SizeToFit
ImageTransparentColor	Magenta
Margin	0, 1, 0, 2
MergeAction	Append
MergeIndex	-1
Modifiers	Friend
Overflow	AsNeeded
Padding	0, 0, 0, 0
RightToLeft	No
RightToLeftAutoMirrorImage	False
Size	23, 22
Tag	
Text	B
TextAlign	MiddleCenter

We need to do this for the next 10 buttons. Design it out so it looks like this:



OK for those of you who are lost, these are our buttons for Bold, Italic Underline, and Strikethrough. Then we got L, C, and R. (Align text) Then we got **A** and **a**, which is case-text, and last we got + and - which is zoom in.

Now for the combo boxes you want to change these settings:

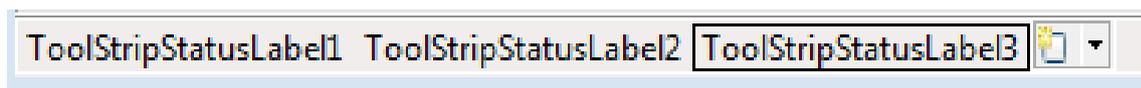
DropDownStyle	DropDownList
DropDownWidth	180
Enabled	True
FlatStyle	Standard

For the first Combo Box change its width to **180** and for the second change combo Box its width to **80**. OK now we need to rename the tool bar controls below in the table is the new names for each item. It goes in order from left to right.

tbNew	tbUnderline
tbOpen	tbStrike
tbSave	tbAlignLeft
tbCut	tbAlignCentre
tbCopy	tbAlignRight
tbPaste	tbUpper
tbBold	tbLower
tbItalic	tbZoom
	tbZoomOut
	tbSelectFont
	tbSelectSize

The Status Strip

OK nearly there! Click the status strip and click the drop down arrow and insert a new label. Then Insert another Label, then another one. The status strip should look like this:



Then click the middle label which says **ToolStripStatusLabel2** in the properties towards the bottom you should see something called **Spring**, set it to **true**. The Status Strip should expand out looking like this:



Next click on the last Status Strip Label, and sets its name to **lblZoom** set its display text to **zoom**. Then for the first label set its name to **lblCharCount** and its text to **0**. For the middle one leave its name as default and change its display text to nothing. It should all look like this.

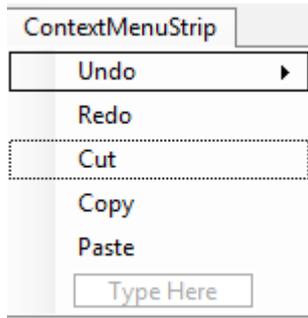


The Context Menu Strip

OK now click on the Context Menu Strip (rcMenu). And add the following values.

Undo, Redo, Cut, Copy, Paste,

It should look like this:



OK next we need to rename this, rename them to the following:

rcUndo, rcRedo, rcCut, rcCopy, rcPaste.

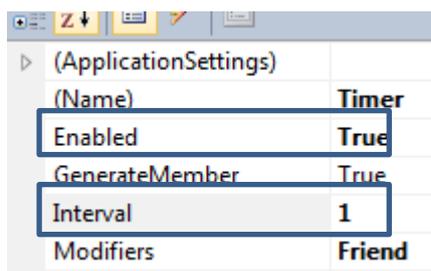
The Richtextbox

Nearly done, click on the rich textbox and then in the properties find **ContextMenuStrip** and select the **rcMenu** this will link them both.



The Timer

OK Last thing! Hopefully I have not forgotten anything. Click the Timer, in the properties make sure it is enabled and set its interval to 1.



We are now done with the design.

PART B – CODING

It is now time to code, Hit F7 or got to View > Code. You will see some text like this:

```
Public Class TextEditor  
End Class
```

At the top of Public Class you want the following code:

```
Imports System.Drawing.Text  
Imports System.IO
```

Creating Regions

OK remember that thing about regions? I find it easier to organize code using regions. So First we will use regions to organize the code then we will make some methods.

OK so the regions we want are one for the editor and timer, one for the menu bar, one for the toolbar, one for methods, and one for the right click menu. Go ahead and make the regions your code should look like this:

```
#Region "Editor and Timer"  
#End Region  
  
#Region "Main Menu Bar"  
#End Region  
  
#Region "Toolbar"  
#End Region  
  
#Region "Methods"  
#End Region  
  
#Region "RightClickMenu"  
#End Region
```

OK let me explain this code is going to go in the methods section (remember with methods or procedures we can reuse code). Then we can call these methods we make. First let's make some more regions under methods. Now the regions we want to make under method are going to be called, File, Edit, Tools. So it all should look like this:

```
#Region "Methods"  
  
#Region "File"  
  
#End Region  
  
#Region "Edit"  
  
#End Region  
  
#Region "Tools"  
  
#End Region  
  
#End Region
```

OK Now under **File** we want the following Code:

Main Menu - File

File > New. Creates a new Document

```
'CLEAR THE RICHTEXTBOX - MAKES NEW DOCUMENT  
Private Sub NewDocument()  
    Document.Clear()  
End Sub
```

File > Open. Using the openWork Dialog that we made this code opens the file we select as plain text.

```
'OPEN WORK AS PLAIN TEXT FILE  
Private Sub Open()  
    If openWork.ShowDialog = Windows.Forms.DialogResult.OK Then  
        Document.LoadFile(openWork.FileName, _  
                           RichTextBoxStreamType.PlainText)  
    End If  
End Sub
```

File > Save. Using the saveWork Dialog that we made this code saves our work.

```
'SAVE WORK AS PLAIN TEXT FILE
Private Sub Save()
    If saveWork.ShowDialog = Windows.Forms.DialogResult.OK Then
        Try
            Document.SaveFile(saveWork.FileName, _
                RichTextBoxStreamType.PlainText)
        Catch ex As Exception
            MessageBox.Show(ex.Message)
        End Try
    End If
End Sub
```

File > Exit. This code Exits the program.

```
'EXIT THE PROGRAM
Private Sub ExitApplication()
    Me.Close()
End Sub
```

Main Menu – Edit

Now we want the code for the Edit Menu. This code is for Cut, Copy, Paste, etc. We will be able to reuse this code for our ToolBar and Our Right Click Menu.

In the region Edit, put this code:

Edit > Undo. Code for the Undo Tab.

```
Private Sub Undo()
    Document.Undo()
End Sub
```

Edit > Redo

```
Private Sub Redo()
    Document.Redo()
End Sub
```

Edit > Cut

```
Private Sub Cut()
    Document.Cut()
End Sub
```

Edit > Copy

```
Private Sub Copy()
    Document.Copy()
End Sub
```

Edit > Paste

```
Private Sub Paste()  
    Document.Paste()  
End Sub
```

Edit > Select All

```
Private Sub SelectAll()  
    Document.SelectAll()  
End Sub
```

This code is pretty straight forward we are just calling the richtextbox (Document) methods. There are tons too choose from more can be found [here](#).

Main Menu – Tools

OK now we want the code for tools section and for the customize option. The code is the following.

```
Private Sub Customize()  
  
Dim ColorPicker As New ColorDialog()  
  
    ColorPicker.AllowFullOpen = True  
    ColorPicker.FullOpen = True  
    ColorPicker.AnyColor = True  
  
    If ColorPicker.ShowDialog = Windows.Forms.DialogResult.OK Then  
        mainMenu.BackColor = ColorPicker.Color  
        Tools.BackColor = ColorPicker.Color  
        Status.BackColor = ColorPicker.Color  
    End If  
  
End Sub
```

OK In this code we make our method customize to call later. Then we make a new variable ColorPicker and set it as new colorDialog. We set its properties. Then if it shows and the user selects a colour apply that colour to the Main Menu, Tool Bar, and Status Strip.

OK now that we have made all the methods, we need to apply these to the main menu items. So go back to design view click on File and then double click New. You should see this code:

```
Private Sub mMNew_Click(sender As System.Object, e As System.EventArgs) Handles mMNew.Click  
  
End Sub
```

Now all we need to do is call the method we made.

```
Private Sub mMNew_Click(sender As System.Object, e As System.EventArgs) Handles mMNew.Click
    NewDocument()
End Sub
```

OK now you need to do this for all the MainMenu bar items. Here is all the code:

Under File

```
Private Sub mMNew_Click(sender As System.Object, e As System.EventArgs) Handles mMNew.Click
    NewDocument()
End Sub

Private Sub mMOpen_Click(sender As System.Object, e As System.EventArgs) Handles mMOpen.Click
    Open()
End Sub

Private Sub mMSave_Click(sender As System.Object, e As System.EventArgs) Handles mMSave.Click
    Save()
End Sub

Private Sub mMExit_Click(sender As System.Object, e As System.EventArgs) Handles mMExit.Click
    ExitApplication()
End Sub
```

Under Edit

```
Private Sub mMUndo_Click(sender As System.Object, e As System.EventArgs) Handles mMUndo.Click
    Undo()
End Sub

Private Sub mMRedo_Click(sender As System.Object, e As System.EventArgs) Handles mMRedo.Click
    Redo()
End Sub

Private Sub mMCut_Click(sender As System.Object, e As System.EventArgs) Handles mMCut.Click
    Cut()
End Sub

Private Sub mMCopy_Click(sender As System.Object, e As System.EventArgs) Handles mMCopy.Click
    Copy()
End Sub

Private Sub mMPaste_Click(sender As System.Object, e As System.EventArgs) Handles
mMPaste.Click
    Paste()
End Sub

Private Sub mMSelectAll_Click(sender As System.Object, e As System.EventArgs) Handles
mMSelectAll.Click
    SelectAll()
End Sub
```

Under Tools

```
Private Sub mMCustomize_Click(sender As System.Object, e As System.EventArgs) Handles mMCustomize.Click
    Customize()
End Sub
```

OK make sure you move all this to the region “Main Menu”

If you go ahead and test your program these should work.

Tool Bar

OK now it is time to work on the toolbar. Now the first 6 icons are all ok because we already have made the method, we just need to call it.

Go to designer view click on the first icon and you should see this:

```
Private Sub tbNew_Click(sender As System.Object, e As System.EventArgs) Handles tbNew.Click
    End Sub
```

All we need to do is call the NewDocument() method. So it should look like this:

```
Private Sub tbNew_Click(sender As System.Object, e As System.EventArgs) Handles tbNew.Click
    NewDocument()
End Sub
```

Now do this for the rest of the 5 icons. The code is below.

```
Private Sub tbOpen_Click(sender As System.Object, e As System.EventArgs) Handles tbOpen.Click
    Open()
End Sub

Private Sub tbSave_Click(sender As System.Object, e As System.EventArgs) Handles tbSave.Click
    Save()
End Sub

Private Sub tbCut_Click(sender As System.Object, e As System.EventArgs) Handles tbCut.Click
    Cut()
End Sub

Private Sub tbCopy_Click(sender As System.Object, e As System.EventArgs) Handles tbCopy.Click
```

```

Copy()
End Sub

Private Sub tbPaste_Click(sender As System.Object, e As System.EventArgs) Handles tbPaste.Click
Paste()
End Sub

```

OK now make sure all this code is in the region toolbar. Now we need to work on the editing tools, such as Bold, Italic, Underline. Here is the code for all that.

Tool Bar – Bold

```

Dim bfont As New Font(Document.Font, FontStyle.Bold)
Dim rfont As New Font(Document.Font, FontStyle.Regular)

If Document.SelectedText.Length = 0 Then Exit Sub
If Document.SelectionFont.Bold Then
Document.SelectionFont = rfont
Else
Document.SelectionFont = bfont
End If

```

Tool Bar – Italic

```

Dim Ifont As New Font(Document.Font, FontStyle.Italic)
Dim rfont As New Font(Document.Font, FontStyle.Regular)

If Document.SelectedText.Length = 0 Then Exit Sub
If Document.SelectionFont.Italic Then
Document.SelectionFont = rfont
Else
Document.SelectionFont = Ifont
End If

```

Tool Bar – Underline

```

Dim Ufont As New Font(Document.Font, FontStyle.Underline)
Dim rfont As New Font(Document.Font, FontStyle.Regular)

If Document.SelectedText.Length = 0 Then Exit Sub
If Document.SelectionFont.Underline Then
Document.SelectionFont = rfont
Else
Document.SelectionFont = Ufont
End If

```

Tool Bar – Strikethrough

```

Dim Sfont As New Font(Document.Font, FontStyle.Strikeout)

```

```
Dim rfont As New Font(Document.Font, FontStyle.Regular)

If Document.SelectedText.Length = 0 Then Exit Sub
If Document.SelectionFont.Strikeout Then
    Document.SelectionFont = rfont
Else
    Document.SelectionFont = Sfont
End If
```

Tool Bar - Align Left

```
Document.SelectionAlignment = HorizontalAlignment.Left
```

Tool Bar - Align Centre

```
Document.SelectionAlignment = HorizontalAlignment.Center
```

Tool Bar - Align Right

```
Document.SelectionAlignment = HorizontalAlignment.Right
```

Tool Bar - Uppercase

```
Document.SelectedText = Document.SelectedText.ToUpper()
```

Tool Bar - Lowercase

```
Document.SelectedText = Document.SelectedText.ToLower()
```

Tool Bar - Zoom In

```
If Document.ZoomFactor = 63 Then

    Exit Sub
Else

    Document.ZoomFactor = Document.ZoomFactor + 1
End If
```

Tool Bar – Zoom Out

```
If Document.ZoomFactor = 1 Then
    Exit Sub

Else

    Document.ZoomFactor = Document.ZoomFactor - 1
End If
```

OK now we are done with the toolbar it's time to focus on the two Combo Boxes the first one is going to get list of Installed fonts and the second will get the font size. This is where a **For Next** statement will come useful.

In the region editor and timer insert the following code:

Tool Bar – Get Font Size

```
Private Sub TextEditor_Load(sender As Object, e As System.EventArgs) Handles Me.Load
    For fntSize = 10 To 75

        tbSelectSize.Items.Add(fntSize)
    Next
End Sub
```

Basically when the form loads this will print out numbers 10 – 75, I mean you could do it by hand but you might be there for some time.....

Tool Bar – Get Installed Fonts

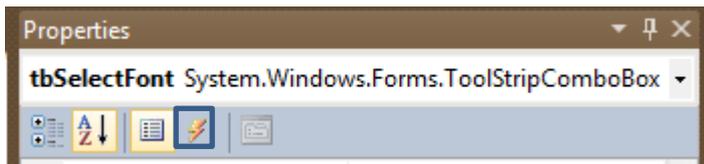
Just below the Get font size code insert this code. This will get the list of all system fonts.

```
Dim fonts As New InstalledFontCollection()

    For fntFamily As Integer = 0 To fonts.Families.Length - 1
        tbSelectFont.Items.Add(fonts.Families(fntFamily).Name)
    Next
```

OK now the combo Boxes are done, they won't actually do anything what we want to do is in the same region **editor** we want to add this code:

We want to use the event **SelectedIndexChanged** for the combo Boxes this basically means when the option changes. Go to design view click on the first combo box then in the properties at the top click events. (It is has a lightning icon, see the image below)



Click in then scroll down until you find **SelectedIndexChanged**, double click inside it and you should get this code:

```
Private Sub tbSelectFont_SelectedIndexChanged(sender As System.Object, e As System.EventArgs)
    Handles tbSelectFont.SelectedIndexChanged

    End Sub
```

Next inside that you would to put this code:

```
    Dim ComboFonts As System.Drawing.Font

    Try
        ComboFonts = Document.SelectionFont
        Document.SelectionFont = New System.Drawing.Font(fontSelection.Text,
        Document.SelectionFont.Size, Document.SelectionFont.Style)
    Catch ex As Exception
        MsgBox(ex.Message)
    End Try
```

Now when you highlight some font in the document and then scroll to change the font, it will. Now let's do the same for the Font Size. Click on the other Combox box and find its event **SelectedIndexChanged**. Double click it and between the private sub and end sub insert this code:

```
        Document.SelectionFont = New Font(tbSelectSize.SelectedItem.ToString,
        CInt(tbSelectSize.SelectedItem.ToString), Document.SelectionFont.Style)
```

The whole code should look like this:

```
Private Sub tbSelectSize_SelectedIndexChanged(sender As System.Object, e As System.EventArgs)
    Handles tbSelectSize.SelectedIndexChanged

        Document.SelectionFont = New Font(tbSelectSize.SelectedItem.ToString,
        CInt(tbSelectSize.SelectedItem.ToString), Document.SelectionFont.Style)

    End Sub
```

Nearly there, now we need to focus on the status bar. We have very two simple codes for these. The first one will count the characters in the current document and the second will count the zoom factor. Go to design view then double click the Timer. You should get this code:

```
Private Sub Timer_Tick(sender As System.Object, e As System.EventArgs) Handles Timer.Tick
    End Sub
```

Now the first code is the character counter we want this code;

```
lblCharCount.Text = "Characters in the current document: " & Document.TextLength.ToString()
```

This is very simple we are saying the text for label “Char Count” should be the number of characters in the document. We use ToString() to convert Integers to strings.

Next for the zoom factor

```
lblZoom.Text = Document.ZoomFactor.ToString()
```

This is basically same as above. Now when you run the text editor you should see the status strip update every time you type and zoom.

Right Click Menu – rcMenu

Woo! Nearly finished, now all we need to do is click the rcMenu (right click menu) and apply the codes for that. The code is very simple we just need to call the methods we made. So the whole code is this:

```
Private Sub UndoToolStripMenuItem_Click(sender As System.Object, e As System.EventArgs)
Handles UndoToolStripMenuItem.Click
    Undo()
End Sub

Private Sub RedoToolStripMenuItem_Click(sender As System.Object, e As System.EventArgs)
Handles RedoToolStripMenuItem.Click
    Redo()
End Sub

Private Sub CutToolStripMenuItem_Click(sender As System.Object, e As System.EventArgs)
Handles CutToolStripMenuItem.Click
    Cut()
End Sub

Private Sub CopyToolStripMenuItem_Click(sender As System.Object, e As System.EventArgs)
```

```
Handles CopyToolStripMenuItem.Click
    Copy()
End Sub

Private Sub PasteToolStripMenuItem_Click(sender As System.Object, e As System.EventArgs)
Handles PasteToolStripMenuItem.Click
    Paste
End Sub
```

You see this is why methods are useful we can reuse code a lot of times. Make sure you place this code in the region "Right Click Menu"

Document - Link

One last thing go to design view click on the richtextbox go to the events and find **LinkClicked** event. Double click it and you should see this code:

```
Private Sub Document_LinkClicked(sender As System.Object, e As
System.Windows.Forms.LinkClickedEventArgs) Handles Document.LinkClicked

End Sub
```

Put this code between the private and end sub.

```
System.Diagnostics.Process.Start(e.LinkText)
```

This will basically make links clickable.

You're Done

I hope you guys enjoyed this tutorial, like always [feedback](#) and criticism is welcome. I hope I have not missed anything out.

It takes a very long time to make such document please respect our copyright and spread the word about us, it really helps.

For updated and other tutorials, see:

<http://www.thecodingguys.net/downloads>

And

<http://www.thecodingguys.net>

Enjoyed it?

Follow me on:



IF YOU FOUND THIS USEFUL, PLEASE SHARE IT!