

# Wysiwyg printing ability on richtextbox

This allows you to print the contents of a RichTetBox in a WYSIWYG manner.

Create a new class that inherits RichTextBox and type in the code:

```
Imports System
Imports System.Windows.Forms
Imports System.Drawing
Imports System.Runtime.InteropServices
Imports System.Drawing.Printing

Public Class RichTextBoxEx
    Inherits RichTextBox
    <StructLayout(LayoutKind.Sequential)> Private Structure STRUCT_RECT
        Public left As Int32
        Public top As Int32
        Public right As Int32
        Public bottom As Int32
    End Structure

    <StructLayout(LayoutKind.Sequential)> Private Structure STRUCT_CHARRANGE
        Public cpMin As Int32
        Public cpMax As Int32
    End Structure

    <StructLayout(LayoutKind.Sequential)> Private Structure
STRUCT_FORMATRANGE
        Public hdc As IntPtr
        Public hdcTarget As IntPtr
        Public rc As STRUCT_RECT
        Public rcPage As STRUCT_RECT
        Public chrg As STRUCT_CHARRANGE
    End Structure

    <DllImport("user32.dll")> Private Shared Function SendMessage(ByVal hWnd
As IntPtr, ByVal msg As Int32, ByVal wParam As Int32, ByVal lParam As
IntPtr) As Int32
    End Function

    Private Const WM_USER As Int32 = &H400&
    Private Const EM_FORMATRANGE As Int32 = WM_USER + 57

    Public Function FormatRange(ByVal measureOnly As Boolean, _
        ByVal e As PrintPageEventArgs, _
        ByVal charFrom As Integer, _
        ByVal charTo As Integer) As Integer

        Dim cr As STRUCT_CHARRANGE
```

```
cr.cpMin = charFrom
cr.cpMax = charTo

Dim rc As STRUCT_RECT
rc.top = HundredthInchToTwips(e.MarginBounds.Top)
rc.bottom = HundredthInchToTwips(e.MarginBounds.Bottom)
rc.left = HundredthInchToTwips(e.MarginBounds.Left)
rc.right = HundredthInchToTwips(e.MarginBounds.Right)

Dim rcPage As STRUCT_RECT
rcPage.top = HundredthInchToTwips(e.PageBounds.Top)
rcPage.bottom = HundredthInchToTwips(e.PageBounds.Bottom)
rcPage.left = HundredthInchToTwips(e.PageBounds.Left)
rcPage.right = HundredthInchToTwips(e.PageBounds.Right)

Dim hdc As IntPtr
hdc = e.Graphics.GetHdc()

Dim fr As STRUCT_FORMATRANGE
fr.chrg = cr
fr.hdc = hdc
fr.hdcTarget = hdc
fr.rc = rc
fr.rcPage = rcPage

Dim wParam As Int32
If measureOnly Then
    wParam = 0
Else
    wParam = 1
End If

Dim lParam As IntPtr
lParam = Marshal.AllocCoTaskMem(Marshal.SizeOf(fr))
Marshal.StructureToPtr(fr, lParam, False)

Dim res As Integer
res = SendMessage(Handle, EM_FORMATRANGE, wParam, lParam)

Marshal.FreeCoTaskMem(lParam)

e.Graphics.ReleaseHdc(hdc)

Return res
End Function

Private Function HundredthInchToTwips(ByVal n As Integer) As Int32
    Return Convert.ToInt32(n * 14.4)
End Function

Public Sub FormatRangeDone()
```

```
        Dim lParam As New IntPtr(0)
        SendMessage(Handle, EM_FORMATRANGE, 0, lParam)
    End Sub

    <StructLayout(LayoutKind.Sequential)> Private Structure
STRUCT_CHARFORMAT
    Public cbSize As Integer
    Public dwMask As UInt32
    Public dwEffects As UInt32
    Public yHeight As Int32
    Public yOffset As Int32
    Public crTextColor As Int32
    Public bCharSet As Byte
    Public bPitchAndFamily As Byte
    <MarshalAs(UnmanagedType.ByValArray, SizeConst:=32)> _
    Public szFaceName As Char()
End Structure
End Class
```

And use this code to do the actual printing:

```
Private Sub ToolPrint_Click(sender As System.Object, e As System.EventArgs)
Handles ToolPrint.Click
Dim printDoc As New Printing.PrintDocument()
AddHandler printDoc.BeginPrint, AddressOf printDoc_BeginPrint
AddHandler printDoc.PrintPage, AddressOf printDoc_PrintPage
AddHandler printDoc.EndPrint, AddressOf printDoc_EndPrint
printDoc.Print()
End Sub
Private m_nFirstCharOnPage As Integer

Private Sub printDoc_BeginPrint(ByVal sender As Object, ByVal e As
System.Drawing.Printing.PrintEventArgs)
m_nFirstCharOnPage = 0
End Sub

Private Sub printDoc_PrintPage(ByVal sender As Object, ByVal e As
System.Drawing.Printing.PrintPageEventArgs)

m_nFirstCharOnPage = RTBe.FormatRange(False, e, m_nFirstCharOnPage,
RTBe.TextLength)

If (m_nFirstCharOnPage < RTBe.TextLength) Then
e.HasMorePages = True
Else
e.HasMorePages = False
End If
End Sub

Private Sub printDoc_EndPrint(ByVal sender As Object, ByVal e As
System.Drawing.Printing.PrintEventArgs)
```

RTBe.FormatRangeDone()  
End Sub

From:  
<https://wiki.plecko.hr/> - **Eureka Moment**

Permanent link:  
[https://wiki.plecko.hr/doku.php?id=development:net:wysiwyg\\_print\\_rtf](https://wiki.plecko.hr/doku.php?id=development:net:wysiwyg_print_rtf)

Last update: **2019/10/31 09:04**

