Learning VB.Net

Tutorial 04 – understanding controls : dialogs, menus

Hello everyone... welcome to vb.net tutorials. These are going to be very basic tutorials about using the language to create simple applications, hope you enjoy it. If you have any notes about it, please send them to <u>notes@mka-soft.com</u> I will be happy to receive them. Finally if you find these tutorials are useful, it would be nice from you to send a small donation via PayPal to <u>donation@mka-soft.com</u>.

tutorial posted on 2009-MAY-16.

Hello everyone, today's tutorial is about Menus and Dialogs. So, let us start designing a simple Notpad application. Please notice that this tutorial focuses only on the controls not the code. More about the code in the next tutorials.

🖳 Start Page - Microsoft	Visual Basic 2008 Express Editio	n		
File Edit Unit Tarls	118a danı - 11ada			2
Toolbox <u>T</u> emplates: Gener: Visual Studio) installed templates			· · · · ·
There controls Drag an text to text to	Class Library WPF WPF Brov Application Application	vser Console ion Application		
A project for cre <u>N</u> ame:	ating an application with a Windows user in	iterface (.NET Framework 3.5)		
			ОК	Cancel
	Learn VB What's New? Beginner Developer Learning Cent Download Additional Content MSDN Forums	er		
🐉 start 🔰 🖳 Star	t Page - Microsof			🐲 👔 🕵 🧐 💼 - 3:48 AM

Create a new project and name it: My Notepad



Rename the form title to My Notepad by modifying the Text property of the form.



Drop a text box on the form.

🖳 My Notepad - Microsoft	Visual Basic 2008 Express Edition				JX
File Edit View Project	Build Debug Data Tools Window Help				
1 🔂 😂 🖽 • 🖬 🗶 🕺 X	B (1) (1) (1) (1) (1) (1) (1) (1) (1) (1)		R 😤 🖻 🛠 🗳	-	
Toolbox 🗸 🗣 🗙	Form1.vb [Design]* Start Page	• ×	Solution Explorer	-	ųΧ
PrintPreviewControl		1			
	My Notepad		🛐 My Notepad		
Process	0		My Project		
💷 ProgressBar			Form1.vb		
PropertyGrid	0				
RadioButton					
🙀 RichTextBox					
🛃 SaveFileDialog					
🔎 SerialPort					
ServiceController			Solution Explorer	Data Source	s
SplitContainer			25000000		
+ + Splitter			Propercies		4 X
는 StatusStrip			TextBox1 System.	Vindows.Forms.	Text -
C TabControl			8≣ 2↓ 🗐 🖋		
TableLayoutPanel			🖽 MaximumSize	0, 0	^
abl TextBox	E1	5	MaxLength	32767	_
🕐 Timer			MinimumSize Madifiana	0, 0 Eviced	_
💷 ToolStrip			Multiline	True	~
			PasswordChar		~
kan ToolTip			Multiline		
0— TrackBar			Controls whether the	text of the edit	81
TreeView			control can span mor	e man one line.	
Ready					-
🛃 start 📃 🖾 My No	otepad - Microso		* 💓	B 9 🕀 3:50	D AM

The problem with text boxes are that you can only use it to write a single line. To solve this, modify the text box property **Multiline** to **True**. This will allow you to write multiple lines in the text box and modify its height. Resize the text box to take the full size of the window





Run the application, and try to write some text. You can see it works fine.

hello, this is a test	
hi	

Next try to resize the window. Now you see there is a problem. The text box does not resize itself to match window size.

🖳 My Notepad - Microsoft	Visual Basic 2008 Express Edition		_ 2 🗙
File Edit View Project	Build Debug Data Tools Window Help		
👬 💕 🗃 • 🖬 🕼 👗	🛍 🚨 📓 🗏 😫 🔊 • (* • 🗐 • 🖫 1) II II 🖬 🗐 🗐	🧟 🖀 🕺 🏷 💪 🚍 🖕
Toolbox 🗸 🗣 🗙	Start Page Form1.vb [Design]	• x	Solution Explorer 🛛 🗸 🕂 🗙
😑 All Windows Forms 🔥 🛑	· · · · · · · · · · · · · · · · · · ·		
🕨 Pointer	Mar Notopad		My Notepad
🐺 BackgroundWorker	a my notepad		🚽 🔤 My Project
🚰 BindingNavigator			Form1.vb
📅 BindingSource			
ab Button			
CheckBox			
CheckedListBox			
🗾 ColorDialog			
E ComboBox	ģ	¢.	
🔄 ContextMenuStrip			Solution Explorer
🔁 DataGridView			Properties - 4 ×
🗗 DataSet			TextBox1 System, Windows, Forms, Text -
T DateTimePicker			
😼 DirectoryEntry			ē Ž↓ 💷 💉 💷
🖳 DirectorySearcher	0	b	AccessibleDescript
DomainUpDown	Re		AccessiblePole Default
ErrorProvider			AllowDrop False
EventLog			Anchor Top, Left 💌
			AutoCompleteCust
FlowLayoutPanel			Anchor
FolderBrowserDialog			Defines the edges of
🔚 FontDialog 🛛 😵			
Ready			
🛃 start 📃 🖾 My No	tepad - Microso		💓 👔 🖉 🧶 🎒 3:51 AM

To solve this select the text box and change the **Anchor** property. This property specifies how should the text box resize itself when its parent container resize itself (the window). The anchor side specifies how far the edge of the control should be from the border of the outer window (or any other control). Specifying the Anchor to be **Top**, **Left**, **Right**, **Bottom** means that whenever the size of the window changes, the text box changes itself so that its sides keeps the same distance from window border. Try changing the anchor settings and test the application and resizing the window.

🕮 My Notepad - Microsoft Visual Basic 2008 Express Edition	
File Edit View Project Build Debug Data Tools Window Help	
🕼 😂 🖬 • 📓 🧶 🛝 🐘 🐘 📓 🗏 🔊 • 🕫 • 🗐 • 🐘 📗 💷 📬 📮	
Toolbox - 4 × Start Page Form1.vb [Design] - ×	Solution Explorer 🛛 🗸 🕂 🗙
😨 DirectorySearcher	
DomainUpDown	📴 My Notepad
ErrorProvider	My Project
P EventLog	Form1.vb
👼 FileSystemWatcher 💼	
FlowLayoutPanel	
G FolderBrowserDialog	
FontDialog	
C GroupBox	
F1 HelpProvider	Solution Explorer
Si HScrollBar	
🗊 ImageList	Properties - 4 X
A Label	TextBox1 System. Windows. Forms. Text -
A LinkLabel	81 2↓ □ 🧭 📰
E ListBox	AccessibleDescript
22 ListView	AccessibleName
#- MaskedTextBox	AccessibleRole Default
A MenuStrip	AllowDrop False
	AutoCompleteCust (Collection)
Version 2.0.0.0 from Microsoft Corporation	Anchor
.NET Component	Defines the edges of the container to
Displays application commands and options grouped by	which a certain control is bound. When a
Ready	
Start My Notepad - Microso	

Next we start working with menus. Search for a control called **MenuStrip** and drop it on the window.

📴 My Notepad - Microsoft Visual Basic 2008 Express Edition	E 8 🛛
File Edit View Project Build Debug Data Format Tools Window Help	
🛐 😂 🖬 • 🔜 🥥 X 🐚 🖄 🗟 🖄 • 🕫 • 🖓 • 🖓 • 🖓 - 🕄 🕨 💷 🖼 📮	🔩 😤 🖄 🛠 🚨 🖕
Toolbox • 4 × Start Page Form1.vb [Design]* • ×	Solution Explorer 🛛 👻 🕂 🗙
😨 DirectorySearcher	
DomainUpDown	My Notepad
ErrorProvider	My Project
EventLog	E Form1.vb
FileSystemWatcher	
FlowLayoutPanel	
FolderBrowserDialog	
FontDialog	
GroupBox	
F1 HelpProvider	Solution Explorer Data Sources
SI HScrollBar	
🗊 ImageList	Properties 🗸 🗸 X
A Label	MenuStrip1 System.Windows.Forms.Me -
A LinkLabel	
E ListBox	AccessibleBole Default
232 ListView	AllowDrop False
#_ MaskedTextBox	AllowItemReorder False
E MenuStrip	AllowMerge True
MessageQueue	Anchor Top, Left
	AutoSize True 💌
NotifyIcon	Anchor
1 NumericUpDown	which a certain control is bound. When a
	J
Ready 10 0,	0 <u>∓</u> 292 × 24
🛃 start 🕺 🖳 My Notepad - Microso	🏕 💓 🕵 🧐 💼 3:52 AM

The control appears under the window as **MenuStrip1** which is the name of the control. The actual menu appears on the form itself. You can create the menu quickly by just start typing. Try typing **File.**

UrrectorySearcher	🔡 My Notepad	My Notepad	20
ErrorProvider EventLog FileSystemWatcher FlowLayoutPanel FolderBrowserDialog FontDialog	File Type Here	- 🥶 My Project	

Notice that when you write a menu entry, you can extend it horizontally, and vertically.

C DirectorySearcher	🔛 My Notepad		👘 💕 🐔 🖆 🖷
ErrorProvider EventLog FileSystemWatcher FlowLayoutPanel FolderBrowserDialog FontDialog GroupBox	File Type Here	Type Here	Form1.vb

Under the file menu add the New, Open, Save, and Close menu items.

Next you will write the code to handle the events for these menu items. Now in the workspace just double click the **New** menu item.

🖾 My Notepad - Microsof	t Visual Basic 2008 Express Edition	
File Edit View Project	Build Debug Data Tools Window Help	
👬 🗃 🖬 • 🖬 🕼 🐰	B B B E E P • C • Ø • B • ■ ■ E Ø Ø €	R 🔁 🖄 🛠 🚨 🗸
Toolbox 👻 🕈 🗙	Form1.vb* Start Page Form1.vb [Design]*	Solution Explorer - 7 ×
🖻 General	Empty V	
There are no usable controls in this group. Drag an item onto this text to add it to the toolbox.	Public Class Form1 Private Sub NewToolStripMenuItem_Click(B) I End Sub End Class	My Notepad My Project Form1.vb Solution Explorer Data Sources Properties • 4 × © 2 2 1 1
Ready	Ln 4	Col 9 Ch 9 INS
📑 start 🔰 🛄 My N	otepad - Microso	🛛 🐲 👔 🥦 🖗 👜 3:53 AM

As with the previous tutorial you get the code editor. Write down the following:

TextBox1.Text = ""

This will clear the text box and allow you to write next text. Add the following code to the close menu item.

End

This will close the application. The code should look like this:

Notice that I haven't change controls names (the menu items name property). You can change the name to a more friendly one as we did in the previous tutorial. Also the NewToolStripMenuItem.Click and CloseToolStripMenuItem.Click are on the same line as Handles.

Run the application, try to write some text, then select File->New. Then try File->Close. Next we see how to save text.

🕮 My Notepad - Microsof	t Visual Basic 2008 Express Edition		
File Edit View Project	Build Debug Data Format Tools Window Help		
i 📴 🗃 • 🖬 🕔 🐰	ⓑⓑⓑ]≣≌ ♥・♡→₽+╚ ▶ ॥ ॥ % [] %	2 2 2 3	🖅 🖕
Toolbox 👻 🕂 🗙	Start Page Form1.vb* Form1.vb [Design]*	Solution Explorer	→ ‡ X
All Windows Forms			12
Common Controls Containers	🔣 My Notepad	My Notepad	
Menus & Toolbars	File	My Project	
🗄 Data			
± Components			
Printing			
🖻 Dialogs			
Pointer			
ScolorDialog			
FolderBrowserDialog		Solution Evoloror	Data Camaral
FontDialog			Data Sources
🛃 OpenFileDialog		Properties	~ ↓ ×
🛃 SaveFileDialog		SaveFileDialog1 S	ystem.Windows.Forn 👻
WPF Interoperability		•= A [=] #	
Visual Basic PowerP			-
🗁 General		CheckPathExists	True
There are no usable		DefaultExt	raise.
controls in this group.		DereferenceLinks	True
Drag an item onto this		FileName	
toolbox.		Filter	~
	MenuStrip1	FileName The file first shown in last one selected by I	the dialog box, or the the user.
Ready			
🥞 start 🔰 🖳 My N	lotepad - Microso	* ()	🐉 🧐 🏚 3:54 AM

Search for a control called **SaveFileDialog** and drop it on the form. You won't see any visual change to the form itself. You can see there is save file dialog available under the window. This control allows you to specify where to save files on the file system and it looks like the window below:



Let us modify the filter property of the dialog. Click on **SaveFileDialog1** to display its properties.

Now change the **Filter** property to be like that:

Set it to be: Text files |*.txt

This property prevents the user form mistakenly saving the file in formats other than text. The **Text files** part is displayed to the end user, while the ***.txt** is used to filter the files and make sure you only select or overwrite text (ending with .txt) files.

The next step is to write the code to save the text written in your application into the disk. Add the code in the handler of the **Save** menu item by double clicking it then typing:

```
SaveFileDialog1.ShowDialog()
If SaveFileDialog1.FileName = "" Then
        Exit Sub
End If
' this part saves the file
FileSystem.FileOpen(1, SaveFileDialog1.FileName, OpenMode.Output)
FileSystem.Print(1, TextBox1.Text)
FileSystem.FileClose(1)
```

The code is explained as follows:

SaveFileDialog1.ShowDialog(): this instructs the application to show the Save dialog on the screen so that you can specify the file name. When you select a name for the file, you can get it from the control using the **FileName** property.

```
If SaveFileDialog1.FileName = "" Then
    Exit Sub
End If
```

The code above checks if the **FileName** is not specified, in other words if you pressed the cancel button when the dialog is shown, the **FileName** will be empty. So in this case no saving will happen and the execution path of the code exits the subroutine.

```
FileSystem.FileOpen(1, SaveFileDialog1.FileName, OpenMode.Output)
FileSystem.Print(1, TextBox1.Text)
FileSystem.FileClose(1)
```

This part saves the text into the file. The SaveFileDialog1.FileName property allows you to get the name of the file. TextBox1.Text gets the text from the text box.

My Notepad (Ru	nning) - Microso Project Build De	ft Visual Basic bug Tools W	: 2008 Express Edit indow Help	ion				_ 2 ×
🔄 🐸 🖬 - 属	Save As						? 🗙	
Form1.vb Form (General)	Save in:	Desktop	ts	~	O Ø E		1	- + X ≥ = ad
E Pr: He	My Recent Documents	S My Computer	Places					ject vb
- End	Desktop							
 	My Documents							orer ☐Data Sources ↓ ↓ ×
Watch	My Computer							
Name	My Network	File name: Save as type:	test Text files				Save Cancel	,
Watch Immediat	e Window					0		
Ready					Ln 23	Col 1	Ch 1	INS
🛃 start 🔰 🚺	💌 My Notepad (Rur	ning 🔣 M	ly Notepad				🦛 🐧) 🥦 🧶 👜 3:58 AM

Run the application, write some text, and then select **File->save**. Save your file into the desktop. You can open this text file via the standard notepad as shown below:



🖳 My Notepad - Microsoft	Visual Basic 2008 Express Edition	
File Edit View Project	Build Debug Data Format Tools Window Help	
🔚 🧉 • 🖬 🐠 🐰	🛍 🖄 🗏 🖹 🧐 • (* • 💭 • 🖏 🕨 💷 🗐 📮 📬	💀 🖻 🖄 🛠 🗳 🗖 🖕
Toolbox 🗸 🗣 🗙	Start Page Form1.vb* Form1.vb [Design]*	Solution Explorer 🛛 🗸 🕂 🗙
All Windows Forms Common Controls		🖹 🗿 🔊 🗉
+ Containers		🔤 📴 My Project
Menus & Toolbars	File	E Form1.vb
± Data + Components		
+ Printing		
Dialogs		
Pointer		
ColorDialog		
FontDialog		Solution Explorer Data Sources
		Properties - I X
3 SaveFileDialog		OpenSileDialant Susters Windows Farr
WPF Interoperability		openFileDialog1 System. windows.For •
🗄 Visual Basic PowerP		₩ 2. 8
🖯 General		CheckFileExists True
		CheckPathExists True
There are no usable		DefaultExt
Drag an item onto this		DereferenceLinks True
text to add it to the		FileName UpenFileDialog1
COOIDOX,	🔄 MenuStrip1 🔠 SaveFileDialog1 🔠 Ope <u>p#g</u> Dialog1	FileName
	· Innonentering Communication	The file first shown in the dialog box, or the last one selected by the user.
Ready		
🛃 start 🔰 🖳 My No	itepad - Microso	🐲 💓 🥩 🧶 🖗 3:59 AM

Now we work on the **File->Open** part. Search for the **OpenFileDialog** control and drop it on the form. Change **FileName** property and remove all the text from it. And change the filter property to: **Text files|*.txt**. and finally go to the **File->Open** event handler by double clicking the Open menu item, And then add the following code:

```
OpenFileDialog1.ShowDialog()
If OpenFileDialog1.FileName = "" Then
     Exit Sub
End If
' this part loads the file
Dim Tmp As String
Tmp = ""
FileSystem.FileOpen(1, OpenFileDialog1.FileName, OpenMode.Input)
Do While Not FileSystem.EOF(1)
      Tmp = Tmp & FileSystem.LineInput(1)
      If Not FileSystem.EOF(1) Then
            Tmp = Tmp \& Chr(13) \& Chr(10)
      End If
Loop
FileSystem.FileClose(1)
TextBox1.Text = Tmp
```

The **OpenFileDialog** works very similar to the **SaveFileDialog** so there is no much to explain about it. After testing this part, we will work with the font. We want to add the ability to change the size and type of the font in the text box to make it easier to read. So add menu entries **Edit**, and **Font**, then add the **FontDialog** control.

🖳 My Notepad - Microsoft	Visual Basic 2008 Express Edition	
File Edit View Project F	auld Debug Data Format Tools Window Help	N 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
Toolbox • 4 ×	Start Page Form1.vb* Form1.vb [Design]*	Solution Explorer
All Windows Forms		
+ Controls	📰 My Notepad	🛐 My Notepad
+ Menus & Toolbars		My Project
+ Data		Form1.vb
+ Components	Font Type Here	
Printing	Type Here	
🖻 Dialogs		
Repointer		
💽 ColorDialog		
🔄 FolderBrowserDialog		
🕢 FontDialog		Solution Explorer Data Sources
🔠 OpenFileDialog		Properties
🛃 SaveFileDialog		
WPF Interoperability		
+ Visual Basic PowerP		8. 2.↓ E
🖃 General		
There are no usable controls in this group. Drag an item onto this text to add it to the		
toolbox.	MenuStript to SaveFileDialog1	1
	Energenpt Energenebrangt Energenption	
	FontDialog1	
Ready		
🛃 start 🔰 🖳 My Not	repad - Microso	🐲 👔 🕏 🧶 🖗 4:05 AM

Next double click the **Font** menu item to built its handler.

FontDialog1.ShowDialog()
TextBox1.Font = FontDialog1.Font

Try to run the application and select the Font menu item

ont	Font style:	Size:	
Microsoft Sans Gerif	Regular	8	OK
Microsoft Sans Serif Microsoft Sans Serif Microsoft Sans Serif Palatino Linotype Shuti Shuti O Sylfaen Symbol	Regular Italic Bold Bold Italic	9 10 11 12 14 16	Cancel
Effects Strikeout Underline	Sample	YyZz	
	Script:		
	Western	~	

Now let us improve out menu now. Right now the menu cannot be accessed using the keyboard. You can make it accessible using the Alt key and some letter. For example click once on the file menu item. Now you can change the text displayed on the menu item. Modify it to be &File. This will have the effect of adding the ability to access the menu item using the Alt+F combination. Perform the same operation for other menu items to be &New, &Open, &Close, &Save, &Edit. The letter after the & symbol is always the access key.



Try to run the application, then pressing Alt+F, then O to show the open file dialog as a test to see if it works.

🖳 My Notepad - Microsoft	Visual Basic 2008 Express Edition	_ # X
File Edit View Project	Build Debug Data Tools Window Help	
🔚 🗃 🖬 • 🖬 🖉 🛛 X	🛍 🛍 📓 🗏 🔊 • 🕫 - 💷 - 🖳 🕨 💷 🖼 📮	i 🗟 😤 🖄 🋠 🖾 🖕
Toolbox 🗸 🗸 🗙	Start Page Form1.vb* Form1.vb [Design]*	x Solution Explorer v 4 X
🗄 All Windows Forms		
🗄 Common Controls		My Notepad
Containers		My Project
+ Menus & Toolbars	Edit Type Here	Form1.vb
± Data	New	
Components Printing	Open	
- Dialogs	Save	
Pointer		
	Close F Type Here	
EolderBrowserDialog	Type Here	
		Solution Explorer Data Sources
3 OpenFileDialog		Properties 🗸 🗸 🗸
SaveFileDialog		CloseToolStripMenuItem System.Wir -
+ WPF Interoperability		8 AL 8
+ Visual Basic PowerP		
General		PidetToLeft No
There are no usable		RightToLeftAutoMi False
controls in this group.		ShortcutKeyDispla
Drag an item onto this text to add it to the		ShortcutKeys None 🗙
toolbox.		Modifiers:
	Menustripi 🖾 Saverlieblaiogi 🔠 Opennieblaiogi	S Ctri Shift D olt
	AT Fontulaiog1	
Ready		Reset
🛃 start 🔰 💐 My No	tepad - Microso	🛹 😧 🔊 🗐 🗐 4:07 AM

Next we will add shortcut keys. This is easy, click the menu item once to show its properties. Change the **ShortcutKeys**, by marking the **Ctrl** check box and selecting the **X** button for the **Close** menu item.



Repeat the same step for other menu items





Run the application and press **Ctrl+O** and you will see the **Open File Dialog** directly.

Next try to make dividing lines between menu items. To do so, write the text – (the minus sign) in the text part of the menu item. Notice that you can drag and change the position of the menu items, so try to position those dividing lines to be something similar to the above.

Now we start defining the **Copy**, **Cut**, and **Paste** commands. So, add these menu items under **Edit**, and write the code for each one of these:

```
1- for the copy
```

Clipboard.SetText(TextBox1.SelectedText)

2- for the cut

```
Clipboard.SetText(TextBox1.SelectedText)
TextBox1.SelectedText = ""
```

3- for the paste

TextBox1.SelectedText = Clipboard.GetText

Run the application and test the copy, cut, and paste.



File Edit View Project Build Debug Data Format Tools Window Help Toolbox Image: Control ima	🕮 My Notepad - Microsof	t Visual Basic 2008 Express Edition	
Image: MenuStrip1 Image: SaveFileDialog1 Image: FontDialog1	Wy Notepad - Microsoft File Edit View Project Prior Project Project Toolbox Project Project Toolbox Project Project Hall Windows Forms Prommon Controls Project Containers Prontainers Project Data Printing Printing Dialogs Printeroperability Visual Basic PowerP General There are no usable controls in this group. Drag an item onto this text to add it to the toolbox.	t Visual Basic 2008 Express Edition Build Debug Data Format Tools Window Help Start Page Form1.vb* Form1.vb [Design]* × × File Edit Help Type Here Type Here Type Here	Solution Explorer Data Sources Properties
		E MenuStrip1 E SaveFileDialog1 E OpenFileDialog1	
	Ready		
Sterrt Wildroson	🛃 start 🛛 🛄 My N	lotepad - Microso	🐲 👔 🛒 🧶 🚇 4:15 AM

Now let us just add an **About** menu item, under the **help** menu item.

Add the following code in the **About** menu item.

MsgBox("This application is a test of making a notepad application out of VB.NET", MsgBoxStyle.OkOnly, "About My Notepad")

The final code should look like this:

```
Public Class Form1
    Private Sub NewToolStripMenuItem Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
         NewToolStripMenuItem.Click
TextBox1.Text = ""
    End Sub
    Private Sub CloseToolStripMenuItem Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
                   CloseToolStripMenuItem.Click
    End Sub
    Private Sub SaveToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles _
                   SaveToolStripMenuItem.Click
         SaveFileDialog1.ShowDialog()
If SaveFileDialog1.FileName = "" Ther
             Exit Sub
         End If
           this part saves the file
         FileSystem.FileOpen(1, SaveFileDialog1.FileName, OpenMode.Output)
FileSystem.Print(1, TextBox1.Text)
         FileSystem.FileClose(1)
    End Sub
    Private Sub OpenToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles _____
OpenToolStripMenuItem.Click
OpenFileDialog1.ShowDialog()
         If OpenFileDialog1.FileName = "" Then
Exit Sub
         End If
         ' this part loads the file
        Dim Tmp As String
Tmp = ""
         FileSystem.FileOpen(1, OpenFileDialog1.FileName, OpenMode.Input)
         Do While Not FileSystem.EOF(1)
   Tmp = Tmp & FileSystem.LineInput(1)
   If Not FileSystem.EOF(1) Then
                  Tmp = Tmp & Chr(13) & Chr(10)
             End If
         Loop
         FileSystem.FileClose(1)
TextBox1.Text = Tmp
    End Sub
    Private Sub FontToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
                   FontToolStripMenuItem.Click
         FontDialog1.ShowDialog()
TextBox1.Font = FontDialog1.Font
    End Sub
    Private Sub CopyToolStripMenuItem Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
         CopyToolStripMenuItem.Click
Clipboard.SetText(TextBox1.SelectedText)
    End Sub
    Private Sub CutToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
                  CutToolStripMenuItem.Click
         Clipboard.SetText(TextBox1.SelectedText)
         TextBox1.SelectedText =
    End Sub
    Private Sub PasteToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
         PasteToolStripMenuItem.Click
TextBox1.SelectedText = Clipboard.GetText
    End Sub
    Private Sub AboutToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
                    AboutToolStripMenuItem.Click
          MsgBox("This application is a test of making a notepad application out of VB.NET", MsgBoxStyle.OkOnly, ________
"About My Notepad")
    End Sub
End Class
```

Notice that some lines are too long so I divided them to multiple lines. In visual basic, writing a command on multiple line requires you to add the underscore symbol (_) to tell the compiler that these two (or more) lines are actually one long line. Try to run the application and test it.

So in this tutorial you tested how to work with Menus, and Dialogs. You can download the example for this tutorial from the web site.

The next tutorial start discussing how to write code in detail and speaking about main blocks of code, and how to work with variables. If you have any notes, or suggestion, let me know by sending emails to <u>notes@mka-soft.com</u>.