

Forum:[Hack Forums](#)

Thread: [\[Tutorial\]Beginners Guide to Vb.Net\[Tutorial\]](#)

SpencerXZX - October 23, 2011, 2:17 pm

This tutorial was made assuming that you know how to open Visual Basic, and how to create a new form.

So, Lets get started!

Controls:

Adding a Tool (Control) to your form

Let's start by adding a textbox to our form. With the tools displayed, do the following:

- Locate the textbox tool.
- Doubleclick the icon
- A textbox is added to your form.

The textbox gets added to the top left position of your form. To move it down, hold your mouse over the textbox and drag to a new position.

Notice the small squares around the textbox. These are sizing handles. Move your mouse over one of them. The mouse pointer turns into an extended line with arrowheads. Hold your left mouse button down and drag outwards. The textbox is resized. Play around with the sizing handles until you're happy with the size of your textbox. One thing you will notice is that you can't make the size any higher, but you can make it wider. The reason why you can't make it any higher is because the default action of a textbox is to have it contain only a single line of text. If it's only going to contain one line of text, Microsoft reasoned, there's no reason why you should be able to change its height. A textbox can only be made higher if it's set to contain multiple lines of text. You'll see how to do this soon.

- Create two more textboxes by double clicking on the textbox icon in the toolbar (Or Right-Click on the selected textbox and choose Copy. Then Right-Click on the Form and choose Paste.)
- Resize them to the same size as your first one.
- Line them up, one below the other with a space in between.

Let's add some labels near the textboxes, so that your users will know what they are for.

- Locate the label control in the control box.
- Double click the label icon
- A new label is added to your form

Click on the label to select it. Now hold your left mouse button down on the label. Keep it held down and drag it to the left of the textbox.

Create two more labels, and position them to the left of the textboxes.

To see what your Form looks like as a programme, click Debug > Start Debugging from the menu bar. Or press F5 on your keyboard.

To stop the programme from running, you can do one of the following:

- Click the Red X at the top right of your Form
- Click Debug > Stop Debugging from the menu bar

All right, we're getting somewhere. We now have a form with textboxes and labels, something that looks like a form people can fill in. But those labels are not exactly descriptive, and our textboxes have the default text in them. So how can we enter our own text for the labels, and get rid of that default text for the textboxes?

To do those things, we need to discuss something called a Property

Properties:

I'm sure you've noticed the area to the bottom right of the design environment, the area with the textboxes in a grid that has those daunting names like AccessibleDescription, AccessibleName, AccessibleRole. That's the Properties box.

Click anywhere on the form that is not a label or a textbox, somewhere on the form's grey areas. The form should have the little sizing handles now, indicating that the form is selected.

If your Properties box says "Textbox1 Textbox" or "Label1 Label" then you haven't yet selected the Form. Click away from the textbox or label until the Properties box reads "Form1." What you are looking at is a list of the properties that a form has: Name, BackColor, Font, Image, Text, etc. Just to the right of these properties are the values for them. These values are the default values, and can be changed. We're going to change the value of the Text property.

First, you might want to display the list of Properties in a more accessible form. You can display the list properties alphabetically. To do that, click the Alphabetic icon at the top of the Properties box.

This will make the properties easier to find.

Before we change anything in the Properties box, let's clear up what we mean by "Property". Those controls you added to the form (textboxes and labels), and the form itself, are called control objects. You can think of controls as things, something solid that you can pick up and move about. Controls (things) have properties. If your television were a control, it too would have properties: an On/Off button property, a colour property, a volume property, and a ... well, what other properties would your television have? Think about it. The properties of your television will have values. The On/Off button would have just two values - On or Off. The volume property could have a range of values, from zero to ten, for

example. If the value of the volume property was set to ten, the loudest value, then you'd probably have some very angry neighbours!

In VB.NET, you can change a property of a control from the Properties Box. (You can also change a property using code, which you'll do quite a lot.) If we go back to our Form object, and the properties and values it has, we can see how to change them using the Properties Box. We'll change only one of these values for now - the value of the Text property. So, locate the property "Text" in the property box.

Text is a Property of Form1. Don't be confused by the word "Form1" next to the word "Text". All this means is that the current value of the Text property is set to the word "Form1". This is the default.

To change this to something of your own, do this:

- Click inside the area next to "Text", and delete the word "Form1" by hitting the backspace key on your keyboard
- When "Form1" has been deleted, type the words My First Form.
- Click back on the form itself (the one with the labels and textboxes), or hit the return key on your keyboard
- The words "My First Form" will appear as white text on a blue background at the top of the form.

As you can see, your new text has gone at the top of the form, in white on the blue background. So the Text Property of a form is for setting the caption you want to display in the title bar at the top. Changing the values of some properties is fairly easy. We'll now change the Text properties of our labels, and the Text properties of our Textboxes.

Click on Label1 so that it has the sizing handles, and is therefore selected. Examine the Property

box for the Label.. You can see that the Label control has quite a few different properties to the Form control.

Think back to your television as a control. It will have different buttons and knobs to your DVD Player control. A label has different “buttons and knobs” to the Form’s “buttons and knobs”. But the Label has a lot of properties that are the same. The Text property of a Label does what you’d expect it to do: adds text to your label. We’ll do that now:

- With label1 selected, click inside the area next to "Text", and delete the word "Label1" by hitting the backspace key on your keyboard
- Type in the words "First Name"
- Click back onto the grey form, or hit the return key on your keyboard
- Label1 has now changed its text caption to read "First Name"
- If you've made a typing error, go back to the first step above and try again.

Now, change the Text property of the other two labels. Change them to these values:

Label2: Last Name

Label3: Telephone Number

What you should notice is that the labels resize themselves, after you press the enter key to commit the changes. You may need to drag your labels to the left a bit.

Now, that Form looks a little squashed at the moment. Is there anything we can do to make it bigger? Well, it just so happens there is.

The Form can be resized just like the Label and the textboxes. Click anywhere on the form that is not a textbox or a label. If you look closely around the Form’s edges, you’ll notice our old friends the sizing handles. To make the form bigger, just stretch it like you did the labels and the textboxes. Play around with the size of the form until you’re happy with it. You can now reposition and resize the textboxes and labels so that things don’t look too squashed.

Now, go ahead and press F5, or start debugging to look at your newly made programme, were making some progress.

Variables:

Time to get our hands dirty with a bit of programming. There’s no putting it off any longer, I’m afraid! We’ll create a new project for this section.

So, if you already have your VB NET software open, you can get rid of the current project by clicking File from the menu bar. From the drop down menu, choose Close Project. You will be returned to the Start Page. Click File > New Project from the menu bars. When you get the dialogue box popping up, choose Windows Application at the top. Then change the Name from WindowsApplication to Variables.

What are Variables?

Why are we discussing variables? And what is a variable?

With Visual Basic, and most programming languages, what you are doing is storing things in the computer’s memory, and manipulating this store. If you want to add two numbers together, you put the numbers into storage areas and “tell” Visual Basic to add them up. But you can’t do this without variables.

So a variable is a storage area of the computer’s memory. Think of it like this: a variable is an empty cardboard box. Now, imagine you have a very large room, and in this room you have a whole lot of empty cardboard boxes. Each empty cardboard box is a single variable. To add two numbers together, write the first number on a piece of paper and put the piece of paper into an empty box. Write the second number on a piece of paper and put this second piece of paper in a

different cardboard box.

Now, out of all your thousands of empty cardboard boxes two of them contain pieces of paper with numbers on them. To help you remember which of the thousands of boxes hold your numbers, put a sticky label on each of the two boxes. Write "number1" on the first sticky label, and "number2" on the second label. What have we just done? Well, we've created a large memory area (the room and the cardboard boxes), and we've set up two of the boxes to hold our numbers (two variables). We've also given each of these variables a name (the sticky labels) so that we can remember where they are.

Now examine this code:

```
Code Dim number1 As Integer
```

```
Dim number 2 As Integer
```

```
number1 = 3
```

```
number2 = 5
```

That's code from Visual Basic Net. It's VB's way of setting up (or declaring) variables. Here's a breakdown of the first line of code (Dim number1 As Integer):

Dim:

Short for Dimension. It's a type of variable. You declare (or "tell" Visual Basic) that you are setting up a variable with this word. We'll meet other types of variables later, but for now just remember to start your variable declarations with Dim.

number1:

This is the cardboard box and the sticky label all in one. This is a variable. In other words, our storage area. After the Dim word, Visual Basic is looking for the name of your variable. You can call your variable almost anything you like, but there are a few reserved words that VB won't allow. It's good practice to give your variables a name appropriate to what is going in the variable.

As Integer:

We're telling Visual Basic that the variable is going to be a number (integer). We'll meet alternatives to Integer later.

Number1 = 3:

The equals sign is not actually an equals sign. The = sign means assign a value of. In other words, here is where you put something in your variable. We're telling Visual Basic to assign a value of 3 to the variable called number1. Think back to the piece of paper going into the cardboard box. Well, this is the programming equivalent of writing a value on a piece of paper.

Find Part 2 here: <https://hackforums.net/showthread.php?tid=1851264>

Perplexity - October 23, 2011, 2:18 pm

(Oct 23, 2011 02:17 PM)The Raptor Wrote: [\[CITE\]](#) This tutorial was made assuming that you know how to open Visual Basic, and how to create a new form.

So, Lets get started!

Controls:

Adding a Tool (Control) to your form

Let's start by adding a textbox to our form. With the tools displayed, do the following:

Is it me or is there something significant missing? 0_0

Tedd - October 23, 2011, 2:18 pm

(Oct 23, 2011 02:18 PM)Perplexity Wrote: [\[CITE\]](#) Is it me or is there something significant missing? O_OI think it is just you. His tutorial obviously covers the beginning steps of visual basic :)

SpencerXZX - October 23, 2011, 2:21 pm

I copied this from my other tutorial, and Hackforums cant use the bullets I have, so it cuts it off, Im working on it.

[EDIT] fixed it. [EDIT]

Penthos - October 23, 2011, 2:34 pm

(Oct 23, 2011 02:21 PM)The Raptor Wrote: [\[CITE\]](#) I copied this from my other tutorial, and Hackforums cant use the bullets I have, so it cuts it off, Im working on it.

[EDIT] fixed it. [EDIT]

Although its a basic guide aimed at people only just thinking of vb. I thought the guide was written well and liked the cardboard box example. Lol.

I think that will help a LOT of noobs understand the very basics of vb.

I would like to suggest you add some pics so users not familiar with the layout can see what you doing, apart from that keep it up ^^

I always have time for those that have time for others ;)

SpencerXZX - October 23, 2011, 5:18 pm

Added a part 2. Follow the link at bottom of part 1.

Perplexity - October 23, 2011, 5:59 pm

(Oct 23, 2011 05:57 PM)TheProgrammer Wrote: [\[CITE\]](#) Good Tut Keep Up The Good Work!

/shortt

Is it me or are you posting on every damn thread just to raise your post count?
Your posts mean nothing and are nothing more than spam.

Menalix - October 23, 2011, 6:12 pm

(Oct 23, 2011 05:59 PM)Perplexity Wrote: [\[CITE\]](#) Is it me or are you posting on every damn thread just to raise your post count?

Your posts mean nothing and are nothing more than spam.
And he readed your comment and was like:

?

They means a lot! POSTS ARE THE BEST WAY TO SHOW YOU'RE A EPIC L337 SICK CODER, NO ONE WILL NOTICE IT'S SPAM BITCH!

Oh damn you noticed. god damn it, i will never be good at coding now whatever: fab fab fab fab fab fab.

SpencerXZX - October 23, 2011, 6:18 pm

I only have 37 posts, but yet I dont need to prove myself by spamming every post here. If you want to show you know something, show it. In my opinion, post count doesnt matter, its what you know that matters, and what your willing to contribute to the community. Sure, I might not be an "Epic

Leet" Coder, but I understand the basics of Visual Basic, and would like to share my knowledge with the public.