

What is a Toolbox Control?

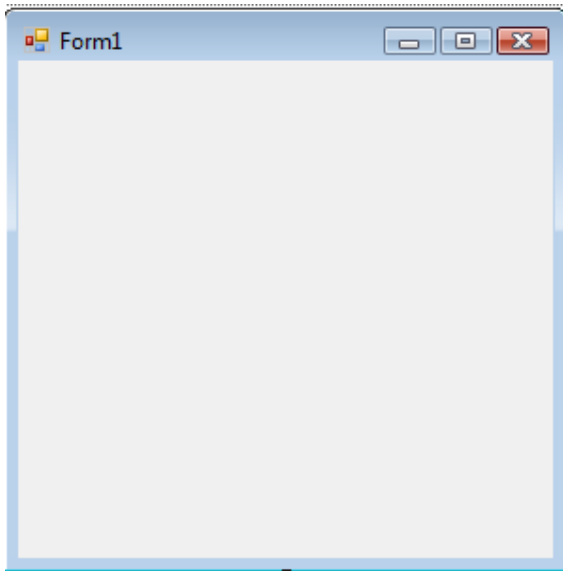
DocumentID : What_is_a_Control.PDF
Author : Michele Harris
Version : 1.0
Date : 07-08-2009

Contents

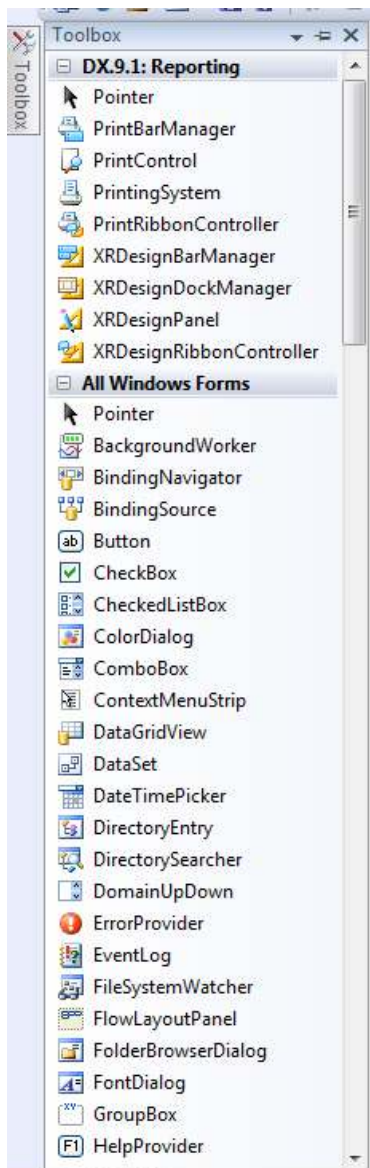
| | |
|--------------------------------------|---|
| ToolBox Controls: The Basics | 2 |
| Changing a Control's Properties..... | 4 |
| Running the Program..... | 7 |

ToolBox Controls: The Basics

Unlike some other programming languages, Visual Basic 2008 has a graphical component that allows you to add all sorts of controls visually instead of writing code for their every property. But what exactly are these controls? These controls are objects you'll add to your forms so that, for instance, a user may select an item from a list box; or so that a user can click OK; or so that the user can input data into a text box. These controls and their corresponding properties make up the style and format of your Windows form. To get the hang of using Controls, created a new Windows Form in Visual Basic Called MyForm. A new, blank form will appear on your screen. This is your canvas!



Visual Basic's controls can be accessed through the Toolbox. When using Visual Studio, you'll find the Toolbox on the upper left hand side of your screen. Hover your mouse over it, and a menu appears with dozens and dozens of controls to add.

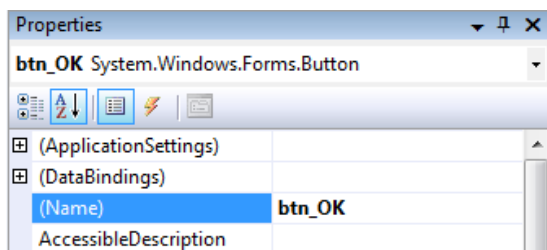


Under "All Windows Forms," double click the *button* option. A button should now appear on your form. Notice how you can easily drag and drop your button anywhere you like on the screen. Repeat this step—go back to the Toolbox and add a second button. Now go back to the Toolbox, and under "All Windows Forms," double click *TextBox*. Now that you have these three controls on your screen, feel free to experiment with them. For now, put them in any sort of arrangement you'd like. You'll begin to notice one handy feature of Visual Studio 2008—that a blue line appears to help you align your controls when you move them.

property, you'll need to locate the Properties Window, located on the bottom right side of your screen.

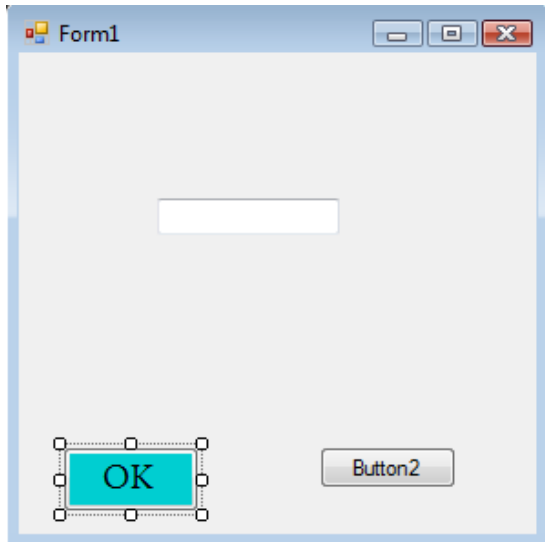
You'll notice that this list is ordered alphabetically, but any item enclosed in parentheses (such as your control's *Name*) are located at the top of the screen. Visual Studio automatically assigns a name to any control you create. You're free, however, to alter this name. Single click the TextBox on your form, and go back to the Properties window. Change the TextBox's name from *TextBox1* to *txt_hello*. Note: it's always a good practice to name your control based on what type of control it is. For instance, if we had a button whose operation is to exit a program, we might name that control *btn_exit*. Such a practice helps us keep track of our controls' operations much more easily.

Also, it's worth mentioning that changing the name of your control does *not* change its text and vice versa. For instance, go back to your new form, and single click Button1. Now find its Text property (you might need to scroll down), and change it from *Button1* to "OK." Now in the same Property Window, scroll up and note its name is still Button1. Change its name to *btn_OK*.

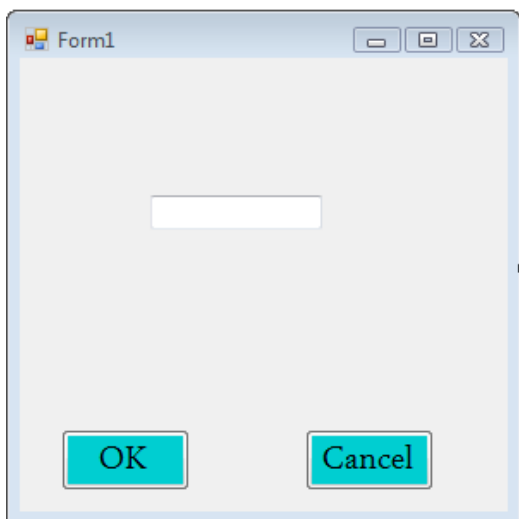


Now scroll down a little further. You'll note that you can change the color and font of your button, too! For this example, feel free to change these to your favorites! I changed *btn_OK*'s Forecolor to Dark Turquoise and its font to Arabic Typesetting. (Note: Depending on the font

size you choose, you may need to resize your button, as I have. You can perform this action on the form by clicking and extending one of the button's corners.)



Now click once on Button2 on your form. Go to the Properties window, and change its name to *btn_cancel* and its text to *Cancel*. And as before, format any of the other properties the way you'd like to.



Running the Program

Now Press F5 to start debugging. Note that you can type text in the text box and click the buttons. Sure, they don't *do* anything yet, but you did all this programming without typing a line of code yourself. Congratulations!

