Starting Visual Studio 2005
If this is your first time starting VS2005 after installation, you will probably see this screen. It is asking you which language you would always like to start with. It DOES NOT mean you are limited to that language, but simply that will be your default starting language each time. By the way, this is no big deal, it is very easy to switch languages.
Probably, the system will take several minutes to configure various features of the Studio system.
When configuration is complete, you will be shown this page. It is the main start page you’ll be taken to from now on. Notice there are no projects (solutions) listed in the window (arrow 1 in slide) (we have not built any yet!). You can click on the “Project …” link to either open an existing project (arrow 2 in slide) that does not show up in the history area, or start a new project (arrow 3 in slide). I’ll click on the “Create” version.
Notice that because during the initial configuration of Visual Studio we told it we wanted the default language to be Basic, that it opens the project definition page showing us the Basic capabilities.
We want to start a Visual Basic Windows program, so I clicked on “Windows Application” (should have already been highlighted) (arrow 1), and type the name I’ll use for the project (arrow 2).
Here we are at the design window. We’re ready to drag some objects onto the form. If you slide the mouse over the “Toolbox” button (arrow) it will expand (see next slide).
If you slide the mouse off this area the toolbox will close. If you want the toolbox to stay open (or docked), just click on the little pin (arrow 1). To see some common objects that we can put on the form, click on “Common Controls” (arrow 2).
Notice after we pinned the toolbox, the section that has the form moved over so we could see the entire form. Also, notice the different objects now available in the common controls area of the toolkit.
Adding a Button

1. Adding the button
2. Dragging the button

Here I’ve double-clicked on the button object in the toolkit (arrow 1), and then used the mouse to move the button on the form to where I want it (arrow 2).
Adding a Textbox

1. Adding the textbox
2. Dragging the textbox
3. Double-click to see code

I’ll do the same thing, and add a text box. Now, I want to write some code and attach it to the button, so when the user clicks on the button, the program will enter some text in the textbox. To do that, I’ll double-click on the button (arrow3).
The system switched to the code window and wrote some code for us! Since I double-clicked on the button, it wrote the subroutine to handle a single click (this is the default subroutine that it will write for us). I started typing the name of the textbox (textbox1) and as soon as I put the dot there (because I want a property of that object), the compiler is suggesting some properties (arrow).
Typing “te” takes the list to the word we’re looking for, so I could just hit the space bar and continue typing. Notice the yellow tooltip telling us what the selection will be/do.
Finishing the Command

So here I’ve finished typing the command. What will it do? The subroutine will only run when the button is clicked, so if the user clicks the button, the text “Phil” will be put in the textbox. Let’s see it happen.
Program Running!

Here's the running program!!
And after pressing the button, my name “magically” appears in the text box. To exit the program, I'll have to click the red “X” on the window to close it. By the way, if this is the primary window, and I close the window, it closes the application. If this is a “child” window, only that window is closed, and the application continues to run.
Here is the code window. How do I get back to the window designer to add another button? I can click on the “View Designer” button (arrow 1) or I can click on the “Form1.vb [Design]” tab (arrow 2).
Now I've added a new button, and changed the name of it to “btnExit”. You should always change object names AS SOON as you’ve made them. Why? I’ll show you in a couple of slides from now.
Changing Object Text

Now I’m changing the text to be shown on the face of the button to “E X I T”. We need to add some code to run when this button is pressed, so double click on the button.
Notice we’re taken to the code page for this form and the system wrote a subroutine to handle the click event associated with this button. Notice the name of the subroutine has the NEW name of the button in it.
I’ve shown two ways to close. If this is the only form (window), then either of these two lines will work. If there are other forms open from this application, then the first way (arrow 1) will only close this form. The second way (arrow 2) will force the application to close. You should only have one of these lines in the code. Let’s run the program.
Here it is running, and by the way, clicking on the “EXIT” button does what we expect – stops the running program.