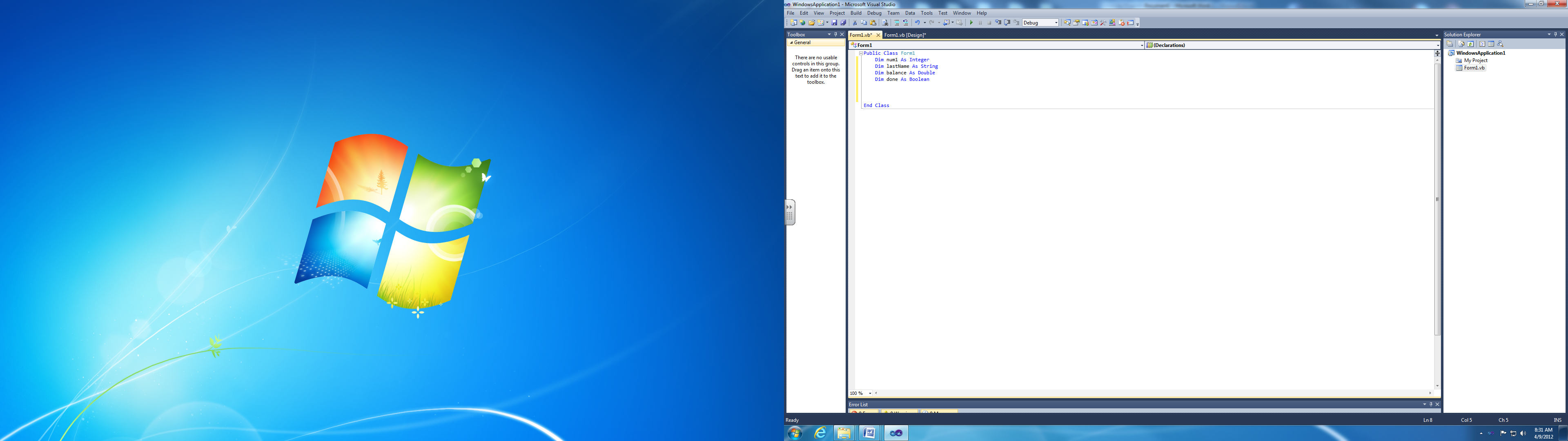
**Computer Programming - Variables**

Variables hold a value (text or number) in a word that can be called upon at any point in your program. It can be assigned at program start or provided by the user. It allows for interaction with the application.

**Declaring Variables**

Any variable that is going to be used in our programs should be declared at the top of the code editor, just below where it says “Public Class Form1”.



(see image at right)

**Syntax**

**DIM** Variable\_Name **as** Type

The *Variable\_name* can be anything, but should be descriptive of the input that it will hold. For example, if you want to save the user’s First name in a variable to be used later in the program, you could save it as f**irstName**.

The *type* part of the declaration tells what kind of data we are dealing with. Some options are:

String – Holds a text string (up to 2 billion characters)

Integer – Holds whole numbers only. (Ex: 5, 99, -7)

Single – Hold decimal numbers (Ex: 3.14, 8.23 1.01)

Boolean – Holds a *true* or *false* value

Char – Holds a single alphanumeric character (Ex: g, 9, B, $)

**Assigning values to Variable**

You can assign a value to a variable when you declare it.

Dim first as String = “Bill”

Dim num1 as Integer = 17

Dim pi as Single = 3.14

Dim Done as Boolean = True

Dim firstInital as Char = "W"

**User Assignment of Value**

We can use a textbox to elicit input from the user to give a value to a variable. Visual Basic has an event called TextChanged that is tied to text boxes that allows the programmer to make assignment of that input to a variable.

For example, if I had a text box that requested the user to enter their first name, I could assign whatever value they inputted to that variable with the following statement:

First=Textbox1.text

If I asked for a number from the user, it is slightly different. Since the textbox is ready to accept words or letters, we must convert that string to a number. Let’s assume we declared a variable called age as Integer, and we used a textbox to ask the user for their age. To assign their input to the variable age, we do this:

Age=Val(textbox1.text)

**Using and Outputting Variables**

Now that we have a value tied to a variable, we want to use it for something. We can either output it to a label or message box, or we can use it as part of a formula. We use the *&* symbol when we want to transition between static text (text surrounded by " ") and a variable. You can use several *&*s if you have several things to output.

Ex: Msgbox(“My name is “ & First &” “&Last)

This will output:

My name is Bill Dixon

\*Notice that we had to put a space in between the first and last variables.

**Using Variables in Formulas** - For the following examples assume:

Dim x as Integer = 5

Dim y as Integer = 2

|  |  |
| --- | --- |
| **Code** | **Result** |
| x = 7 + 6 | x = 13 |
| x = x + 1 | x = 6 (value of x is increased by 1) |
| x = x - 1 | x = 4 (value of x is reduced by 1) |
| x = x + y | x = 5 + 2, thus x = 7 |
| x = y \*3 | x = 6 |
| x = Val(Textbox1.text) \* 2 | (user enters 4 in textbox1) 🡪 x = 8 |

Often, we want to use the user's input to calculate something. For example, if we wanted to calculate our paycheck for the week, we could do this:

Dim pay, hourlyWage, hoursWorked as Single

Let's assume that the user entered hourlyWage and hoursWorked into two textboxes. To calculate it:

hourlyWage = Val(Textbox1.text)

hoursWorked = Val(Textbox2.text)

pay = hourlyWage \* hoursWorked (multiplies wage by hours worked and saves the result in *pay*)

Label1.text = pay (outputs *pay* into label)