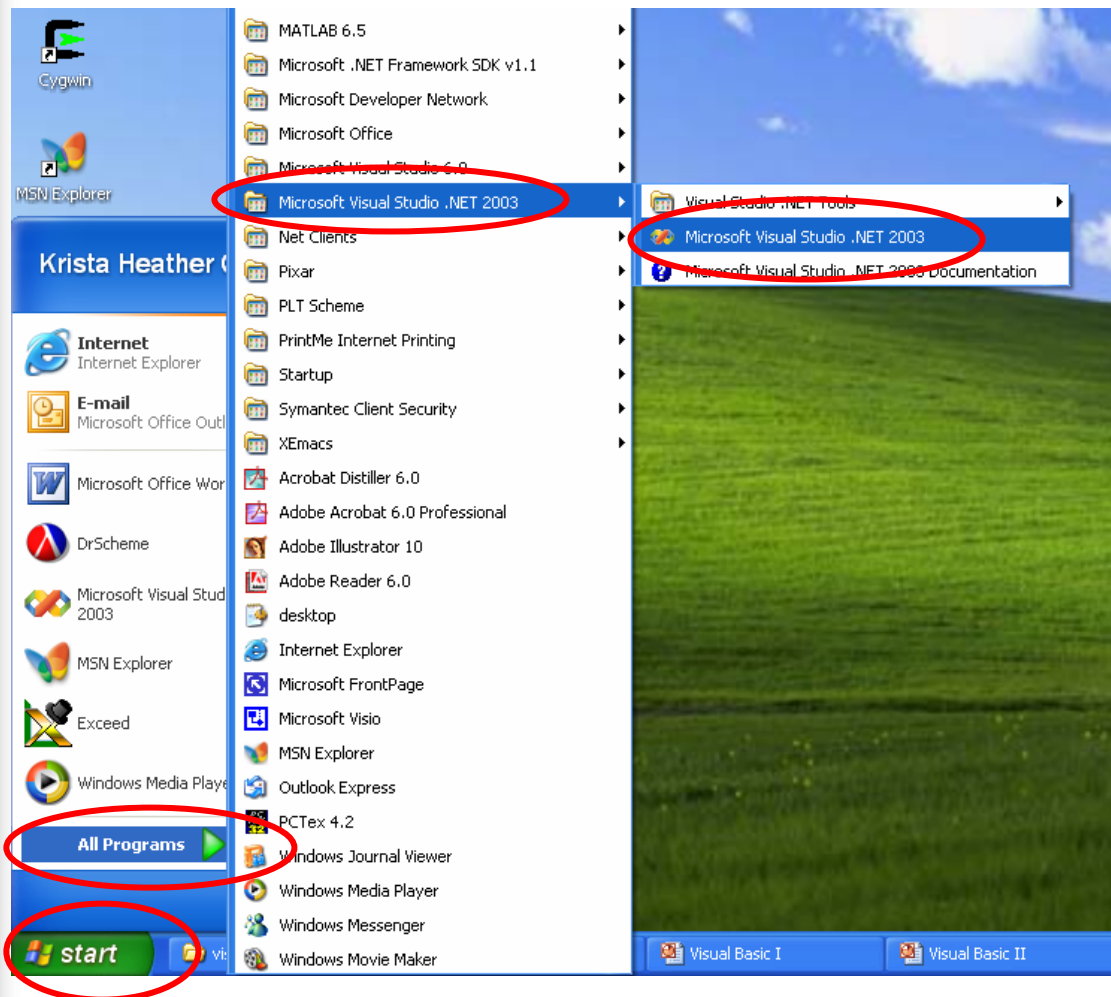


# Let's Get Visual

A decorative horizontal bar consisting of a series of vertical rectangular segments in various colors including black, blue, teal, light blue, and yellow, arranged in a slightly wavy pattern across the width of the slide.

Start creating GUIs.

# Start Up Visual Studio .NET



1.) Go to Start

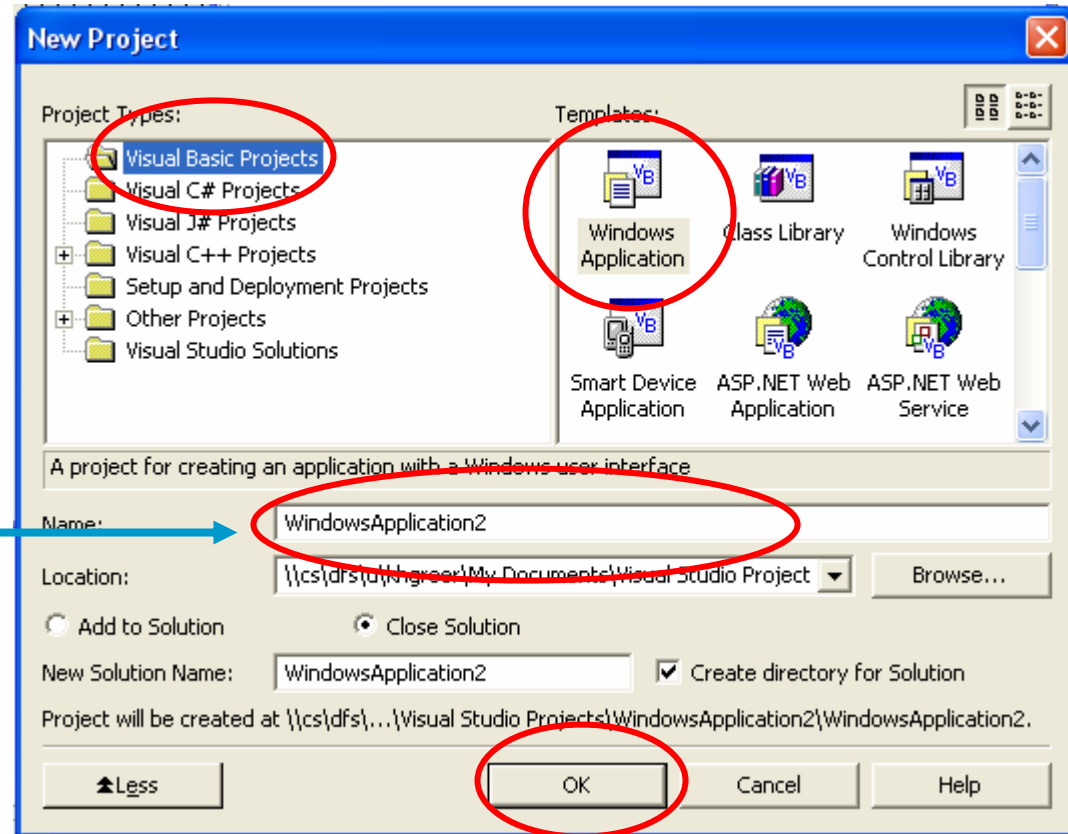
2.) Go to All Programs

3.) Go to Microsoft Visual Studio .NET 2003

4.) Go to Microsoft Visual Studio .NET 2003



But what now? Open an new project by going to File on the menu bar.

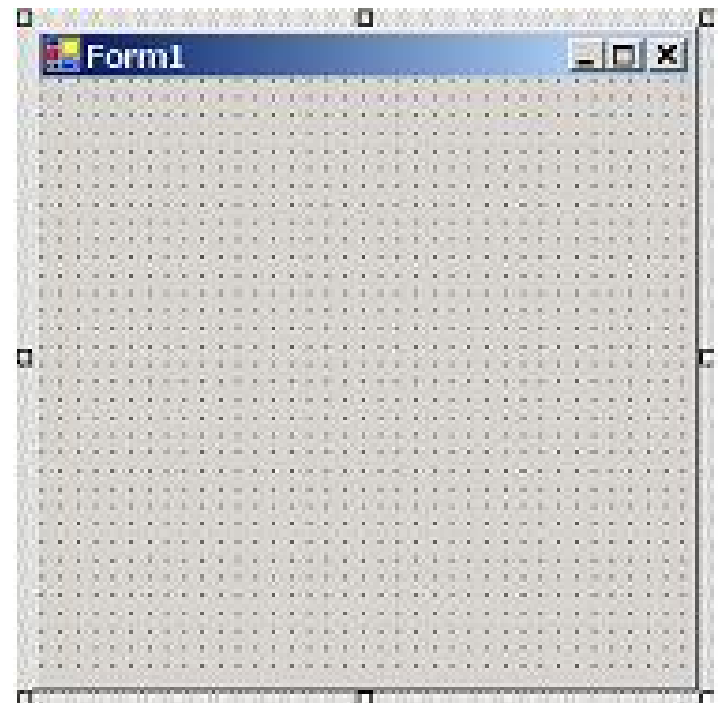


Name your project Inventory.

Select Windows Application and hit OK.

# Projects Appear as Forms

- Projects appear in the Visual Basic workspace as forms like the one to the right.
- Your first job is to take the different objects like labels, buttons, and textboxes and place them in the form.



# Adding Objects to a Form

The screenshot shows the Microsoft Visual Basic .NET IDE in Design mode. The title bar reads "test - Microsoft Visual Basic .NET [design] - Form1.vb [Design]\*". The menu bar includes File, Edit, View, Project, Build, Debug, Data, Format, Tools, Window, and Help. The toolbar contains various icons for file operations and development tools. The Toolbox on the left is expanded to show the "Windows Forms" category, listing various controls such as Pointer, Label, LinkLabel, Button, TextBox, MainMenu, CheckBox, RadioButton, GroupBox, PictureBox, Panel, DataGrid, ListBox, CheckedListBox, ComboBox, ListView, TreeView, TabControl, DateTimePicker, and MonthCalendar. A red arrow points to the "Toolbox" button in the top-left corner of the Toolbox. Another red arrow points to the "TextBox" control in the Toolbox. On the right, a form is displayed with a grid background. A "TextBox1" control has been added to the form, and a red arrow points from the "TextBox" in the Toolbox to this control. Two blue callout boxes provide instructions: the top one says "Double-click the Toolbox button to display the list of objects you can use in your project." and the bottom one says "Click on the object you want and drag the object to the spot on the form you want. Then, let go. You can still relocate your object by clicking and dragging."

test - Microsoft Visual Basic .NET [design] - Form1.vb [Design]\*

File Edit View Project Build Debug Data Format Tools Window Help

Debug

Toolbox

Data

Components

Windows Forms

- Pointer
- Label
- LinkLabel
- Button
- TextBox
- MainMenu
- CheckBox
- RadioButton
- GroupBox
- PictureBox
- Panel
- DataGrid
- ListBox
- CheckedListBox
- ComboBox
- ListView
- TreeView
- TabControl
- DateTimePicker
- MonthCalendar

Double-click the Toolbox button to display the list of objects you can use in your project.

Click on the object you want and drag the object to the spot on the form you want. Then, let go. You can still relocate your object by clicking and dragging.

# Give Your Objects Names

The image shows a Windows Forms application designer interface. The main window is titled "Testing Testing 1-2-3" and contains a form with a blue background and a grid pattern. The form has the word "Tester" centered at the top and a text box below it with the label "Name:". The text box is selected, and its properties are displayed in the Properties window on the right. The Properties window shows various properties for the text box, including "Lines", "Behavior", "Configurations", "Data", "Layout", and "Focus". The "Name" property is highlighted in blue and circled in red, with a red line pointing to it from the text below.

Property	Value
Lines	String[] Array
RightToLeft	No
ScrollBars	None
Text	
TextAlign	Left
Behavior	
AcceptsReturn	False
AcceptsTab	False
AllowDrop	False
AutoSize	True
CharacterCasing	Normal
ContextMenu	(none)
Enabled	True
HideSelection	True
ImeMode	NoControl
MaxLength	32767
Multiline	False
PasswordChar	
ReadOnly	False
TabIndex	1
TabStop	True
Visible	True
WordWrap	True
Configurations	
(DynamicProperties)	
Data	
(DataBindings)	
Tag	
Layout	
(Name)	<b>nameTextBox</b>
Locked	False
Modifiers	<b>Friend</b>
Focus	
CausesValidation	True
Layout	

Make sure you give your objects good names that you can use later for programming.



# Assignment: Inventory GUI

*A college bookstore receives cartons of textbooks. For every shipment, each carton contains the same number of textbooks. The inventory manager wants to use a computer to calculate the total number of textbooks that arrive at the bookstore for each shipment. The inventory manager will enter the number of cartons received and the fixed number of textbooks in each carton. Then, the application will calculate the total number of textbooks for that shipment.*

## **Create a GUI with:**

- a title in a label
- a text box to take in the number of cartons
- a text box to take in the number of books per carton
- a text box to print out the total number of books
- a label for each text box
- a button to calculate the total inventory

*Remember...it should be readable and nice to look at, just like your webpage.*

# Let's Get Visual!

A decorative horizontal bar consisting of a series of vertical rectangular segments in various colors including black, blue, teal, light blue, yellow, and grey, arranged in a slightly wavy pattern across the width of the slide.

Let's start using the visual  
basic objects in mathematical  
programs.

# Using Information from Objects

- A user is going to come along and put stuff into your VB program, like numbers into the text boxes of Inventory.
- You need to be able take that information and do stuff with it, like multiply two numbers together in Inventory.



# Taking Stuff Out of Boxes

1. Start by double-clicking on the button that you created for Inventory. This should open up the code used to program your project.
2. It should say something like:

```
Private Sub btnCalculate_Click(. . .)
```

```
End Sub
```

3. Inside this **subroutine** of your program, you need to be able to get information from the text box. You do this by using the name of that textbox like this:

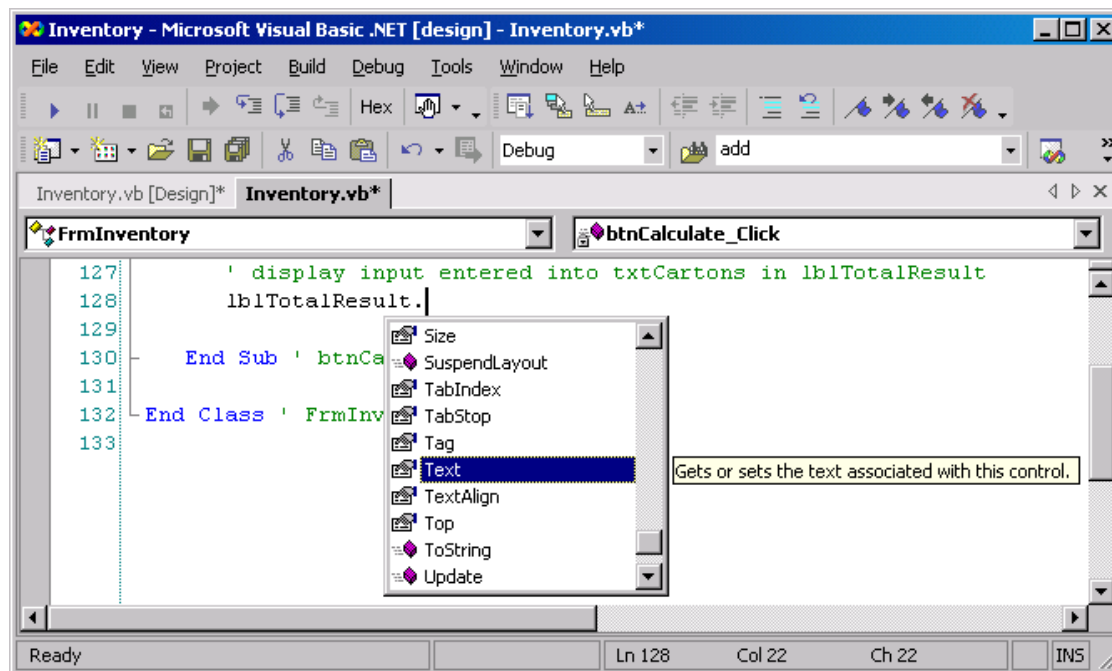
```
Val (textbox1.Text)
```



4. This takes the text inside textbox1, and then turns it into a numerical value.

# A Little Menu Might Pop Up...

- While you're typing the “\*.Text” part, the Intellisense menu might pop up.
- This lists all of the Properties that you can set.
- Either just type “\*.Text” or select Text from the menu.



# Doing Stuff With It

So now we have the information the user has put into your program. What should we do with it?

In the case of the Inventory program, we want to multiply together the two numbers the user puts in. So all we do is multiply the numerical values, like this:

```
Val(textBox1.Text) * Val(textBox2.Text)
```



# Math in Visual Basic

<u>Operation</u>	<u>Operator</u>	<u>VB Expression</u>
Addition	+	f + 7
Subtraction	-	p - c
Multiplication	*	b * m
Division (float)	/	x / y
Division (integer)	\	v \ u
Exponentiation	^	q ^ p
Negative	-	-e

# Putting Stuff Back In

So now you have the result of your computations and you want the user to be able to see it. You need to display the information in another text box.



Do you remember when you reset the contents of the text box to 0 in the Properties menu? You can set the contents of the text box in the code. This is how its done:

```
textBox3.Text = Val(textBox1.Text) * Val(textBox2.Text)
```

# Assignment: Inventory Program

- Use the GUI at right, or the similar one you created earlier.
- Double-click the button to show the code.
- Tell the button to calculate the total number of items when clicked.

