



VISUAL BASIC I

**CC111
INTRODUCTION TO COMPUTERS**



Intended Learning Objectives



- Familiar with the **Generation of computer Languages.**
- High level of languages **translation**
- Familiar with the **Visual Basic Environment.**

What is Visual Basic?

- High Level **4th Generation** Programming Language
- **Object** and **Event** Driven
- **Visual** - Windows Based
- Integrated Development Environment (**IDE**)

What is VB Used For?

- Most Popular Programming Language
- Stand alone programs
- Customized specialized Applications
 - E- Commerce
 - Web based shopping forms and inquiries
- User friendly Interface to applications

Example Order Entry Screen

The screenshot shows a window titled "Order Entry Screen" with the following fields and data:

- Cust No. 32455
- Date 25-Jan-96
- Customer Name/Addr: Carlos Ortega, 123 Main, Anytown, USA
- Order No. 123-555

Below the form is a table with the following data:

Prod No	Qty	Description	Price	Price
11101	1	Ergo Mouse	\$59.00	\$59.00
66203	1	Laser 600 Printer	\$750.00	\$750.00
48024	2	Disk Drive 1024 mb	\$829.00	\$1,658.00
51032	2	RAM Expansion 32 mb	\$550.00	\$1,100.00
		Total		\$3,567.00

Buttons: Show Products, Exit

Annotations:

- Red arrow pointing to the Customer Name/Addr field.
- Red arrow pointing to the Order No. field.
- Red arrow pointing to the first row of the table.
- Red arrow pointing to the Show Products button.
- Red arrow pointing to the Exit button.

Also enables user to click on buttons to initiate processing steps.

Allows user to use a mouse to click on boxes for text entry.

Used by a computer equipment retail outlet to record data regarding customer orders.

Generations of Computer Languages

- **1st** - Machine language - 0110 0011 1000
- **2nd** - Procedure-oriented languages
 - ▣ FORTRAN - 1954 for scientists and engineers
 - ▣ COBOL - 1959 for business applications
 - ▣ C - 1972 - for UNIX operating systems
- **3rd** - Object-oriented languages
- **4th** - Event-driven languages. Example: **VB**
- **5th** - Natural languages

Compiler Vs Interpreter

- Higher Languages are translated to Machine Language by:

- **Interpreter**

- Translates instructions to machine code line-by-line.
- BASIC, Quick Basic, **Visual Basic**

- **Compiler**

- Translates the entire program to machine code before running it.
- Fortran, C, C++, **Visual Basic** is also a compiler

What are the Objects in VB?

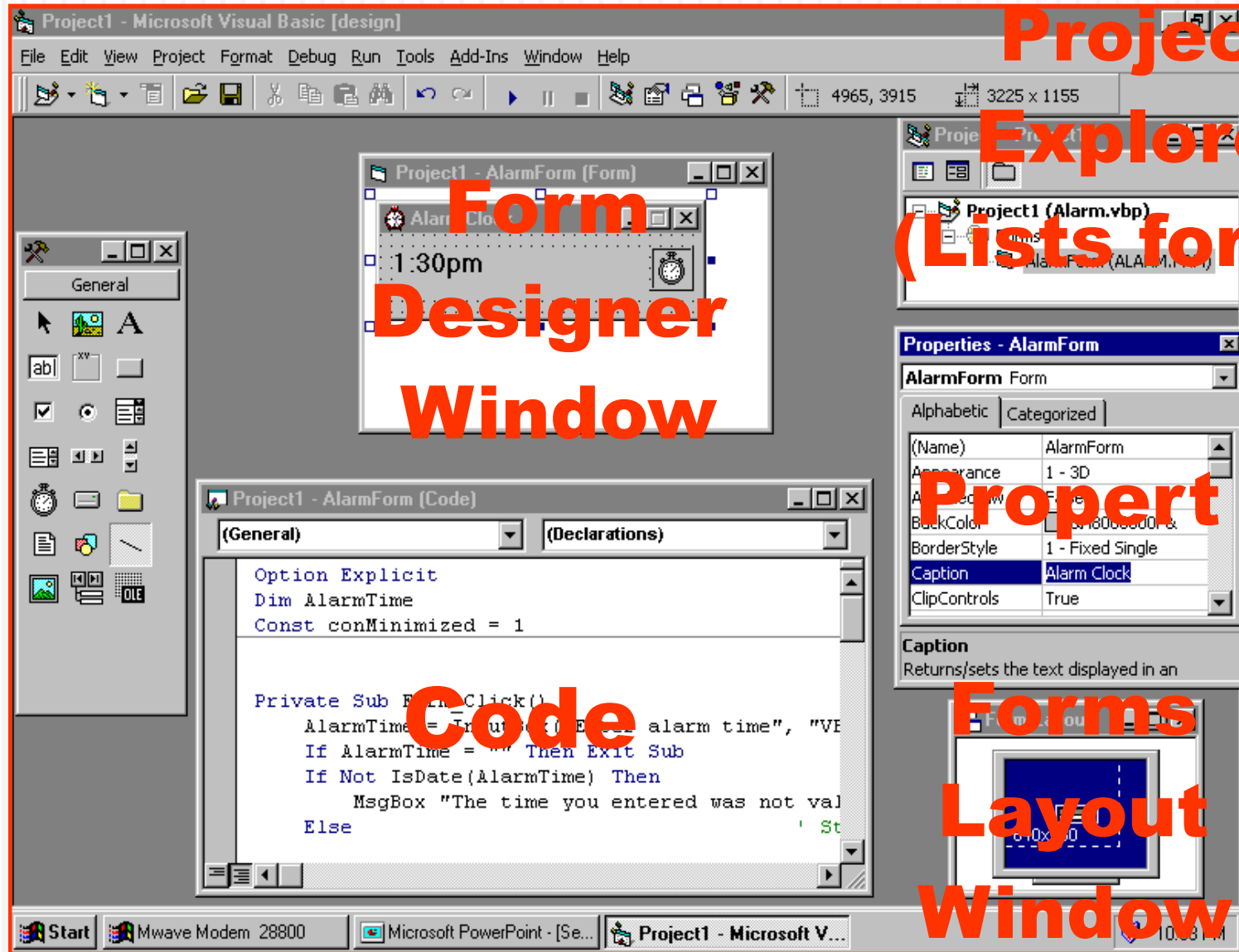
- **Pre-programmed** Code for:
 - Command Buttons
 - Labels
 - Pictures
 - Text Boxes
- Have both data and procedures wrapped together

The screenshot shows a Windows-style window titled "Form1" with a standard title bar (minimize, maximize, close buttons). The form content is titled "Price and Quantity Information" and contains the following elements:

- A text box containing the word "Apples".
- A label "Cost Price:" followed by a text box containing "0.25".
- A label "Quantity In Stock:" followed by a text box containing "64".
- A label "Markup:" followed by a text box containing "0.33".
- A label "Quantity Received:" followed by a text box containing "0".
- A label "Retail Price:" followed by a text box containing "\$0.33".
- A label "Total Quantity:" followed by a text box containing "64".
- Two buttons at the bottom: "Save" on the left and "Close" on the right.

Three red arrows originate from the text in the list on the left and point to specific elements in the form: one points to the "Apples" text box, one points to the "Cost Price:" label and its text box, and one points to the "Close" button.

What is the Integrated Development Environment (IDE)?



Visual Basic Environment



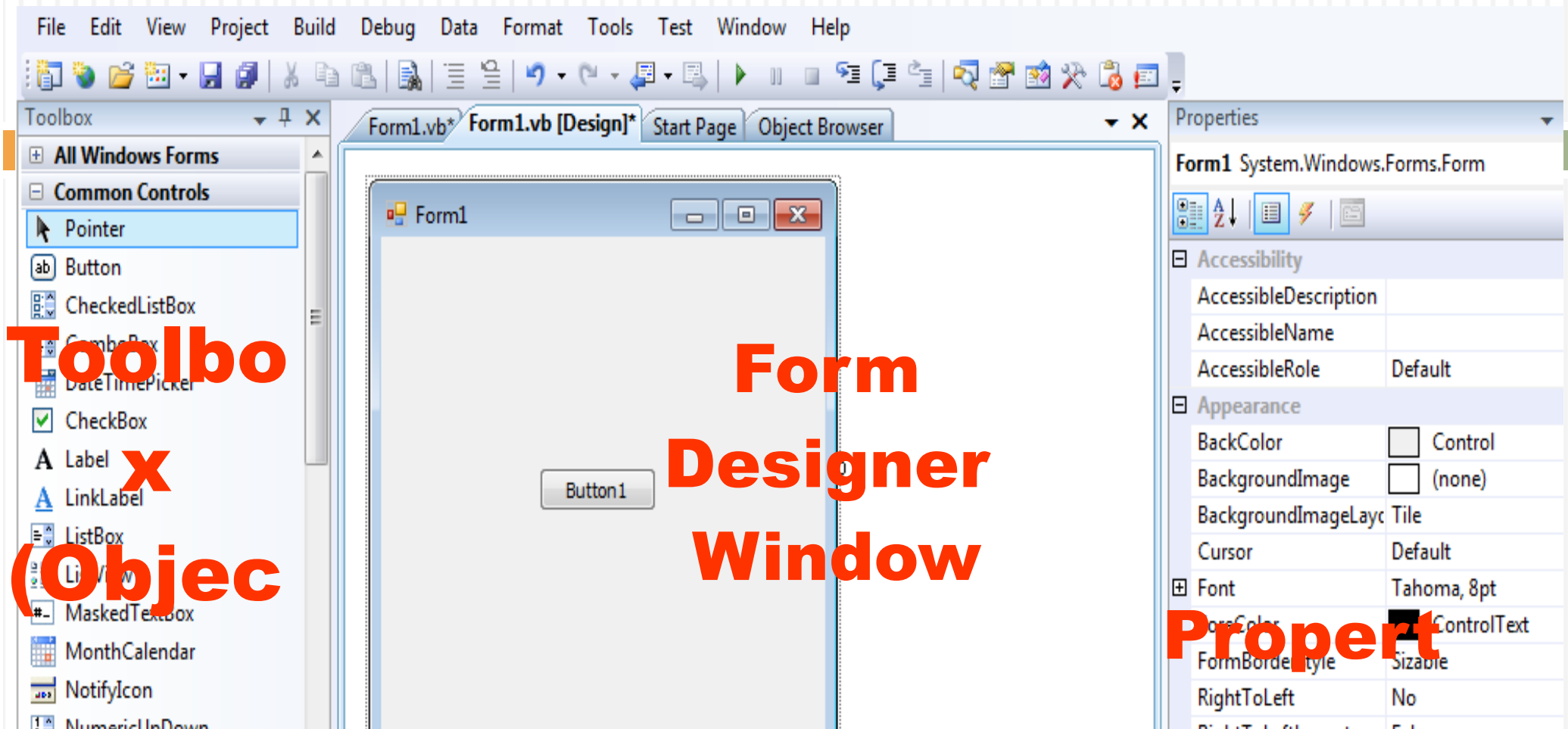
GUI (Graphical User Interface) - forms and windows that the user sees

Property - a characteristic or attribute of an object such as color and size

Event - a user action such as clicking a button

Code Editor window - editor specially designed to help you when writing code

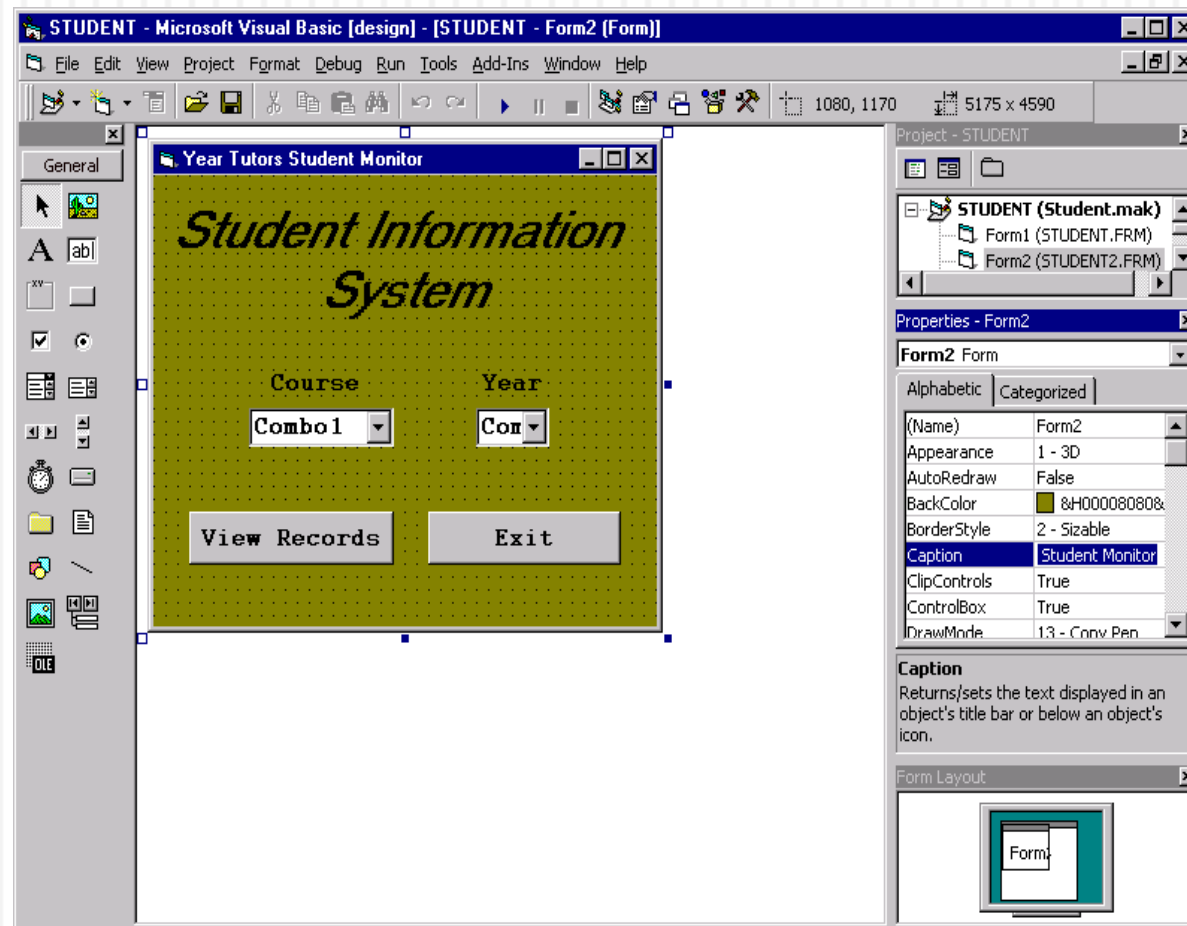
Visual Studio 2008 IDE Environment



- The **Form Designer** Window show the GUI application
- The **Toolbox** is used to drag and drop components to your forms
- The **Properties window** show the value of the properties of each selected component(Object)

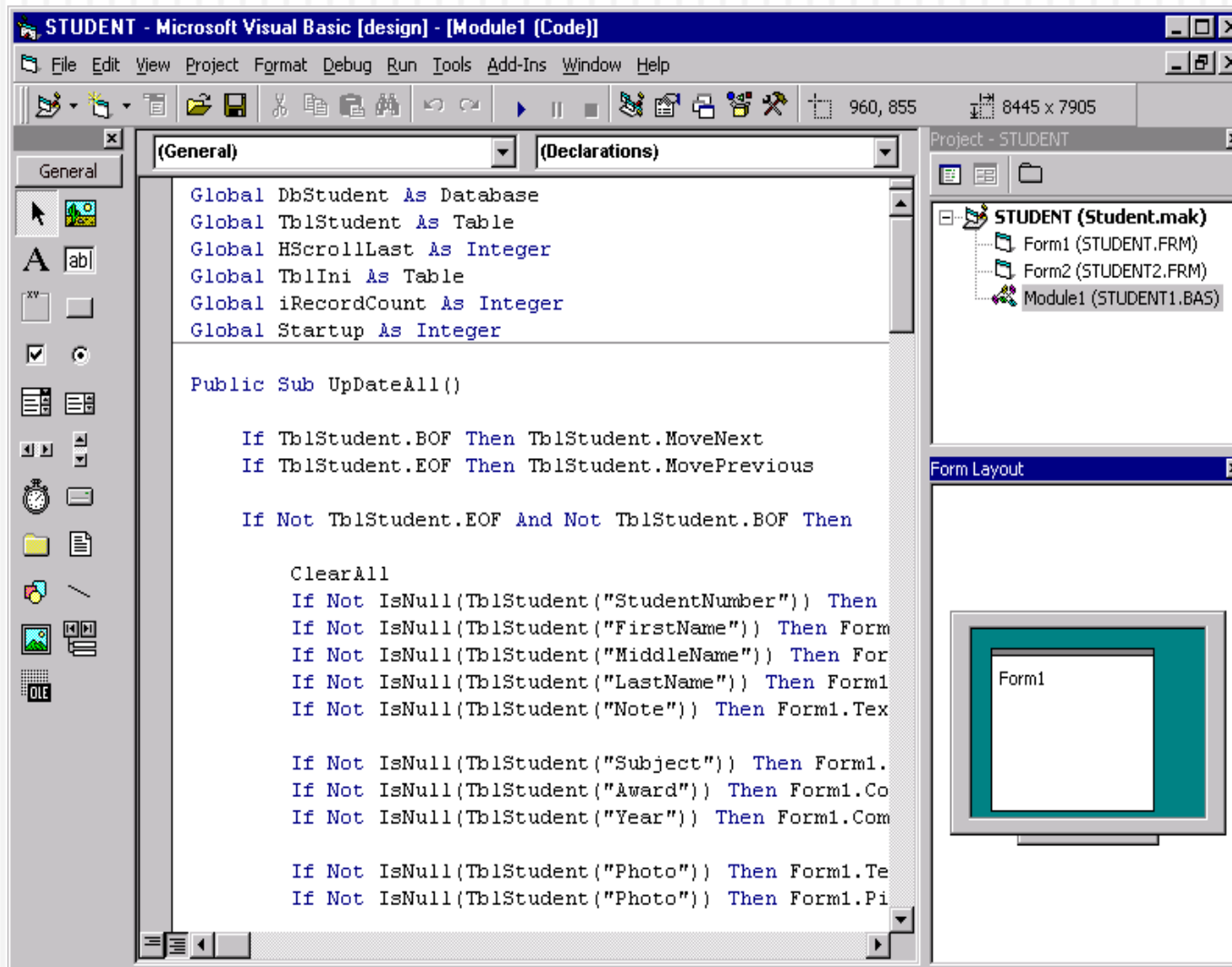
Visual Basic Environment

- Components of the VB design environment (Form view)

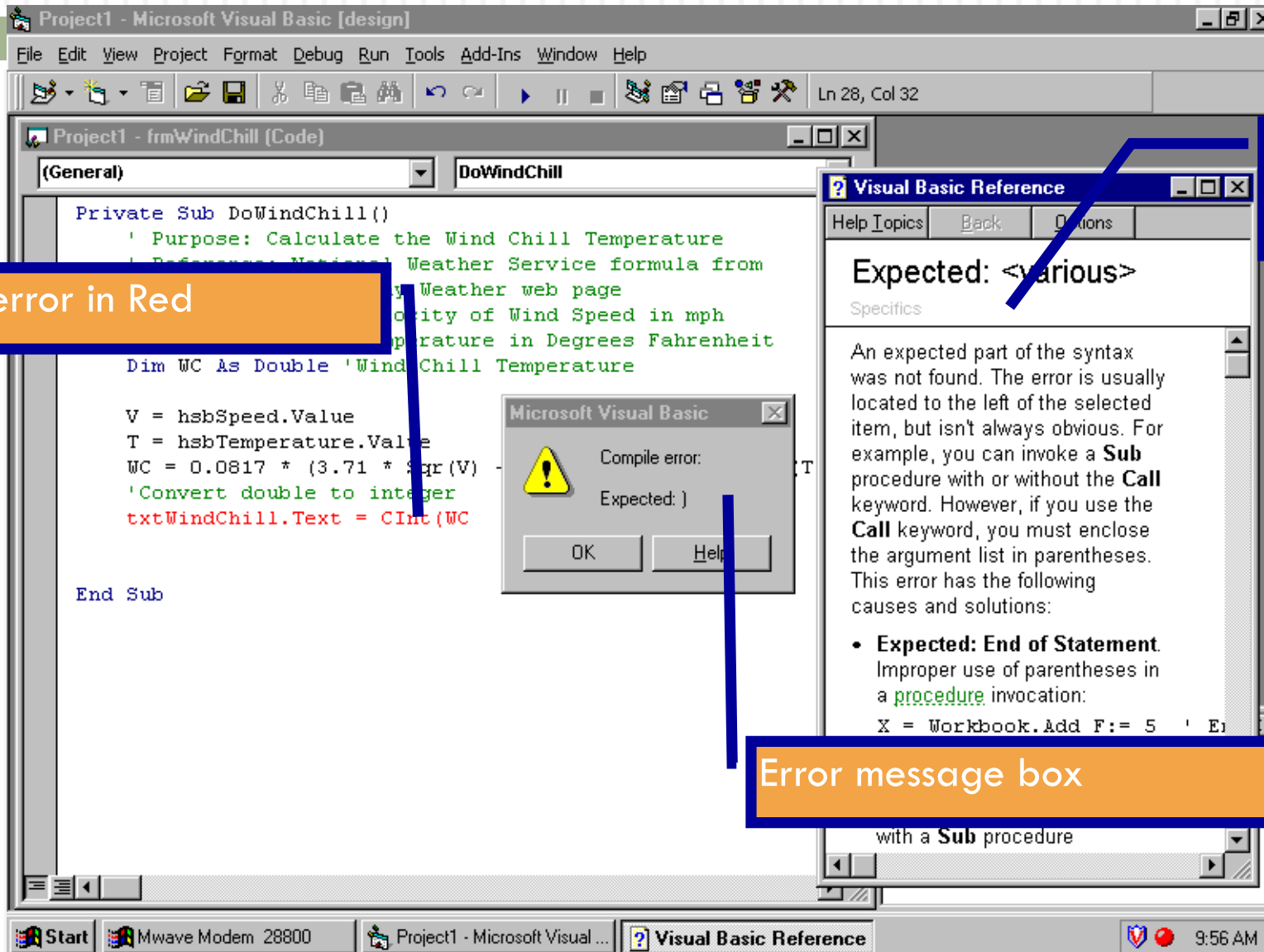


Visual Basic Environment

- Components of the VB design environment (Code view)



The Code Editor Window



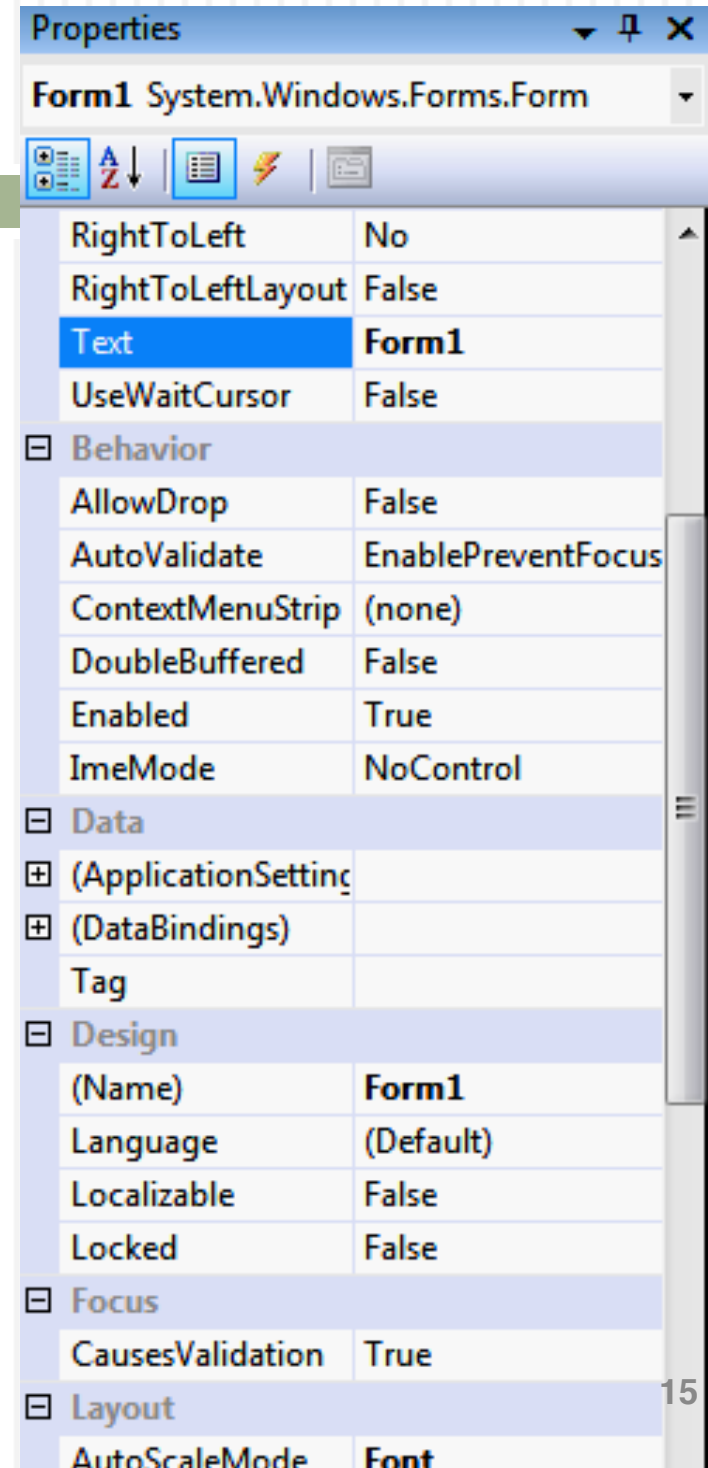
Syntax error in Red

Help window

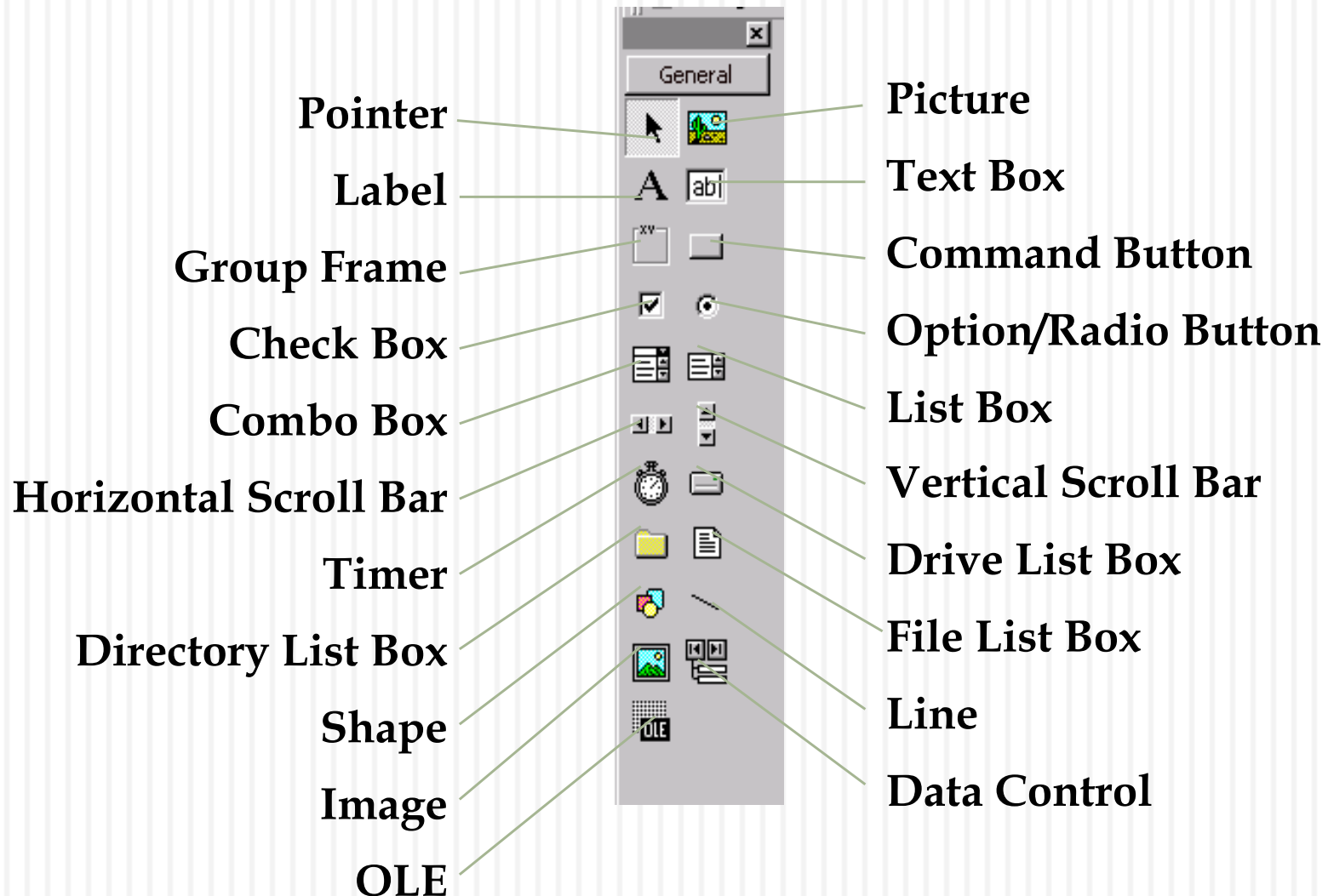
Error message box

Properties Window

- Used to set how a control looks and behaves
- Holds its default values



Control Toolbox



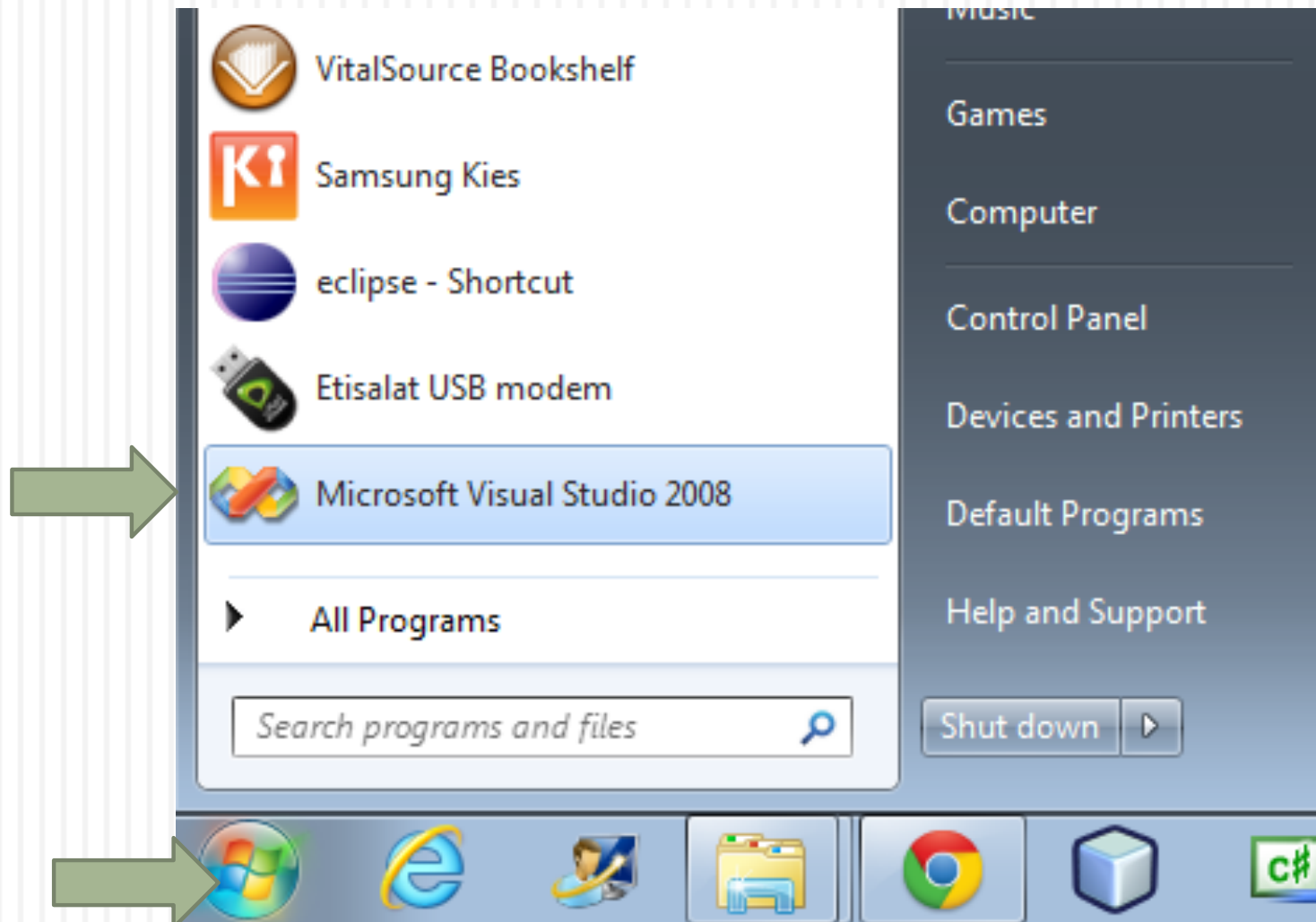
Building Your First Application

- There are **three** primary steps in building a Visual Project:
 - Place (or draw) **controls** on the form.
 - Assign **properties** to the controls.
 - Write **event procedures** for the controls.

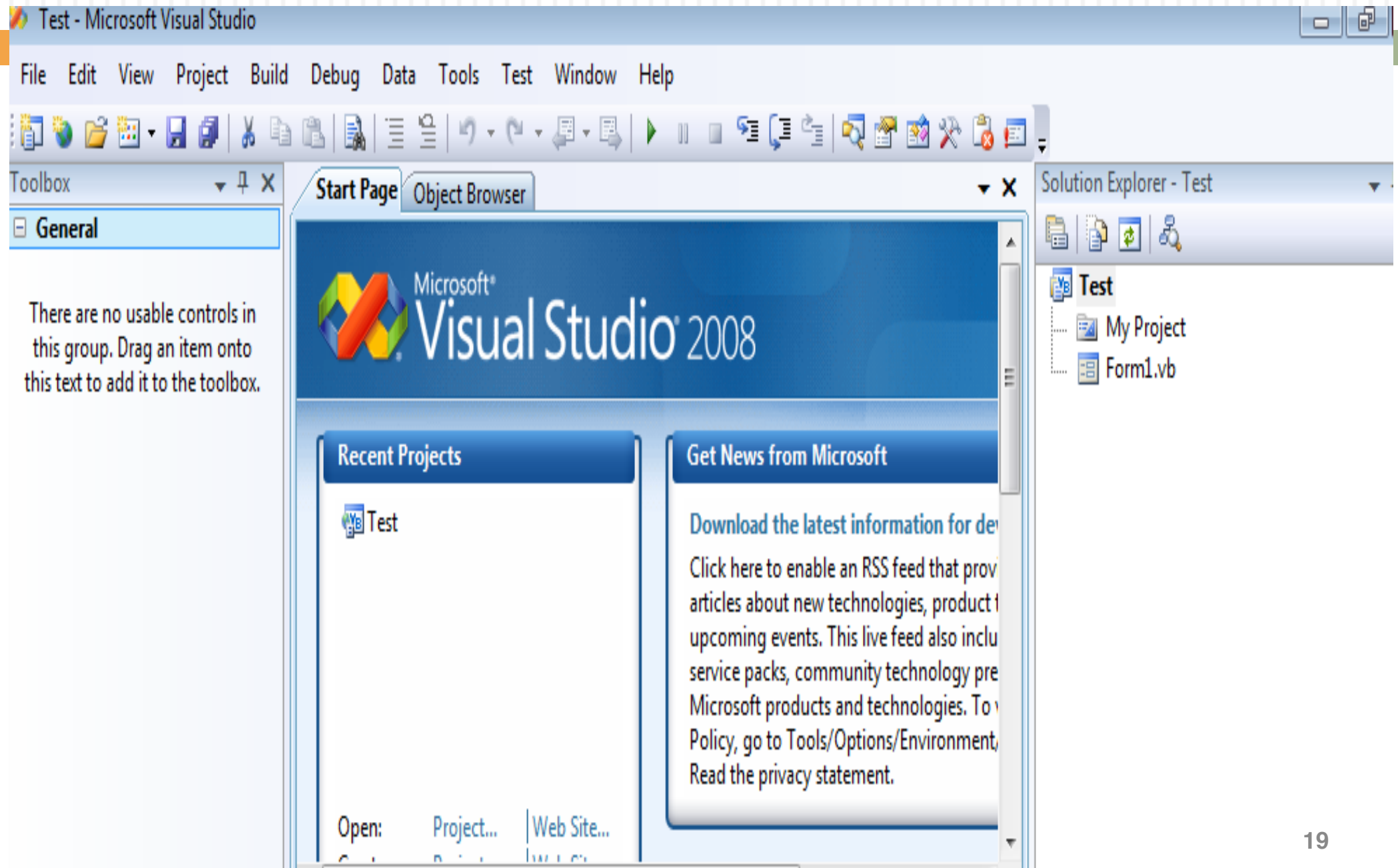


[WelcomeApplication.exe](#)

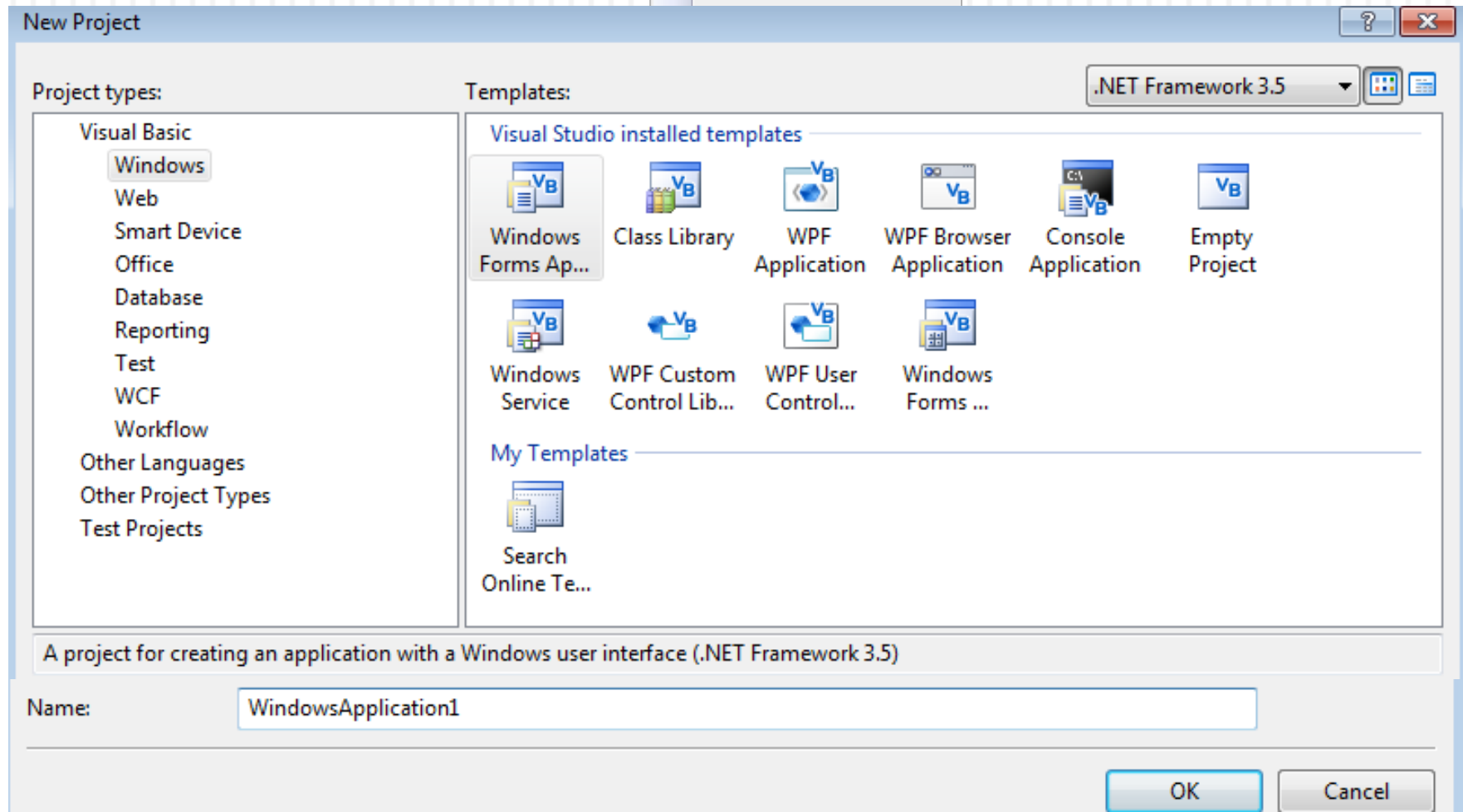
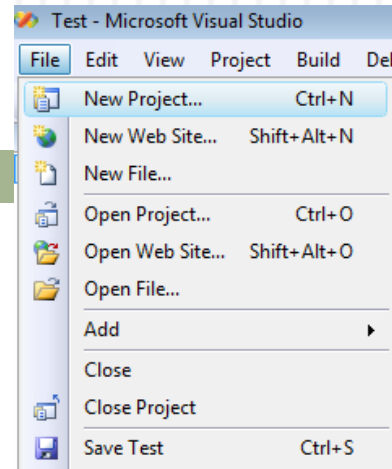
Starting Visual Basic



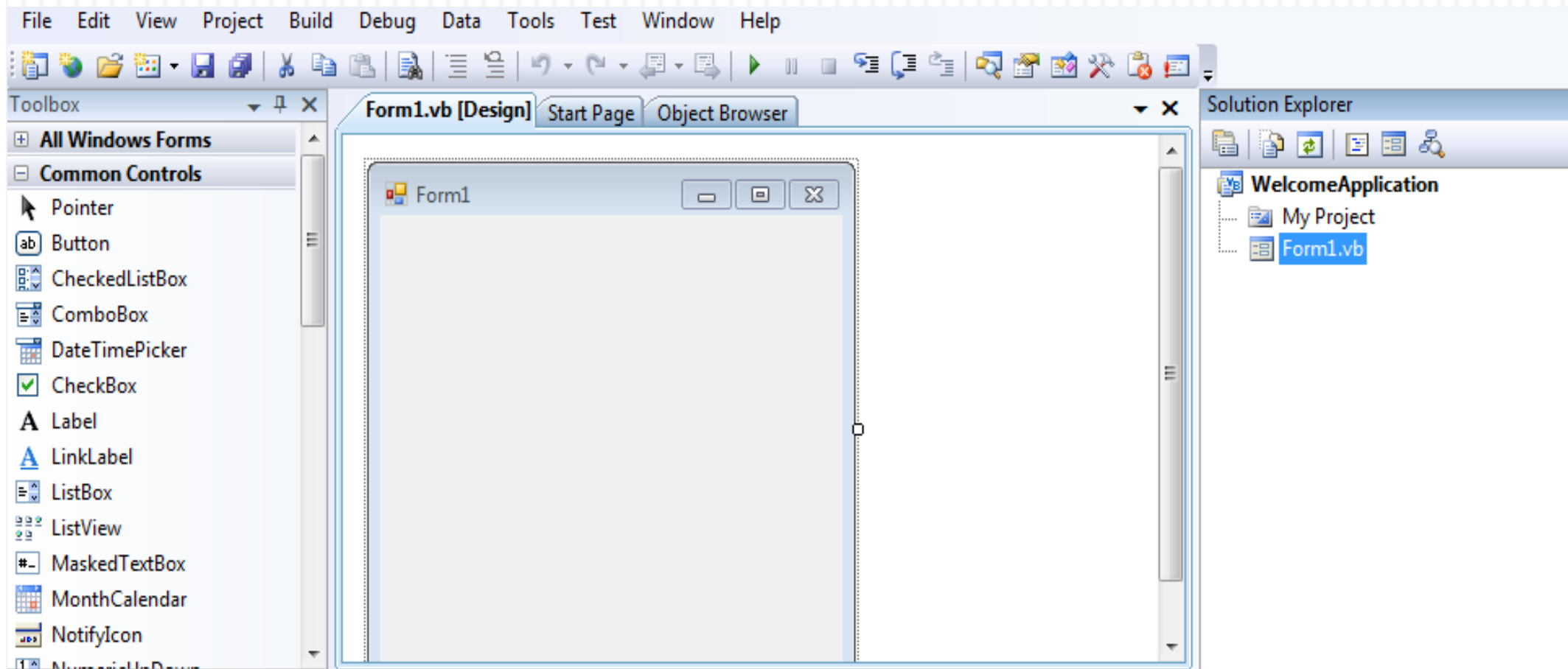
Starting Visual Studio 2008



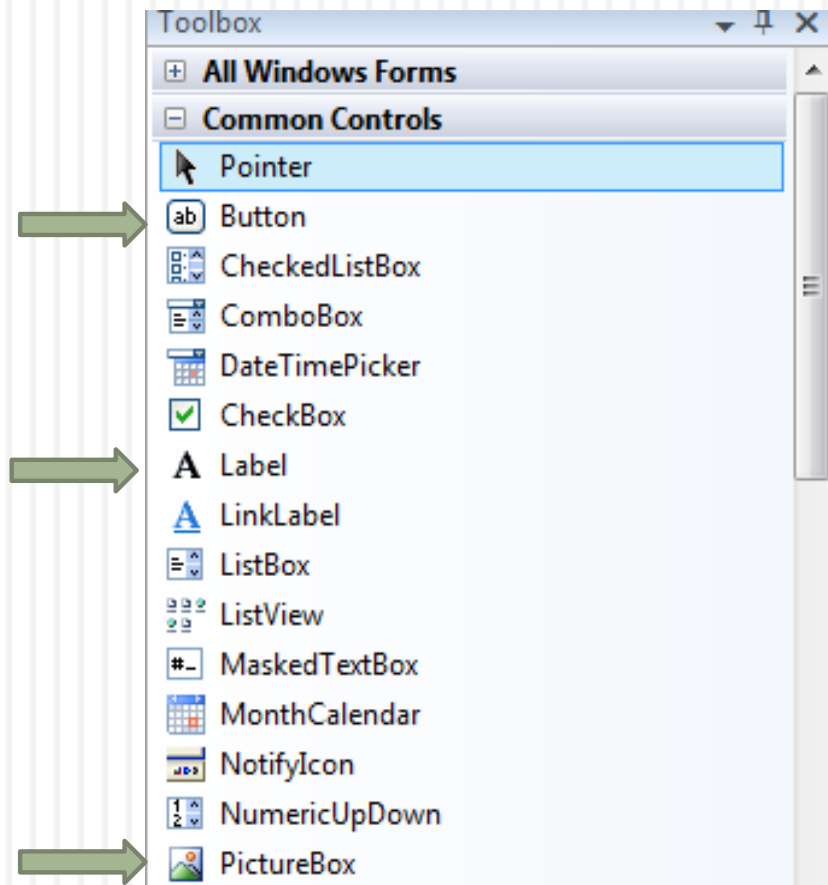
Creating New Project



Project initial interface



Step 1 :Draw your controls





Step 2 : Assign Properties

Changing the Form Title

The image shows the Visual Studio IDE with a Windows Form in Design mode. The form's title bar displays 'Form1'. The design surface contains a dashed rectangle, a 'Display Message' button, and a 'Label1'. The Properties window on the right shows the 'Text' property of the form is set to 'Form1', indicated by a green arrow.

Form1 System.Windows.Forms.Form	
AccessibleDescription	
AccessibleName	
AccessibleRole	Default
[-] Appearance	
BackColor	<input type="checkbox"/> Control
BackgroundImage	<input type="checkbox"/> (none)
BackgroundImageLayout	Tile
Cursor	Default
[+] Font	
ForeColor	<input checked="" type="checkbox"/> ControlText
FormBorderStyle	Sizable
RightToLeft	No
RightToLeftLayout	False
Text	Form1
UseWaitCursor	False

Changing the Form Title.....

Properties

Form1 System.Windows.Forms.Form

Accessibility

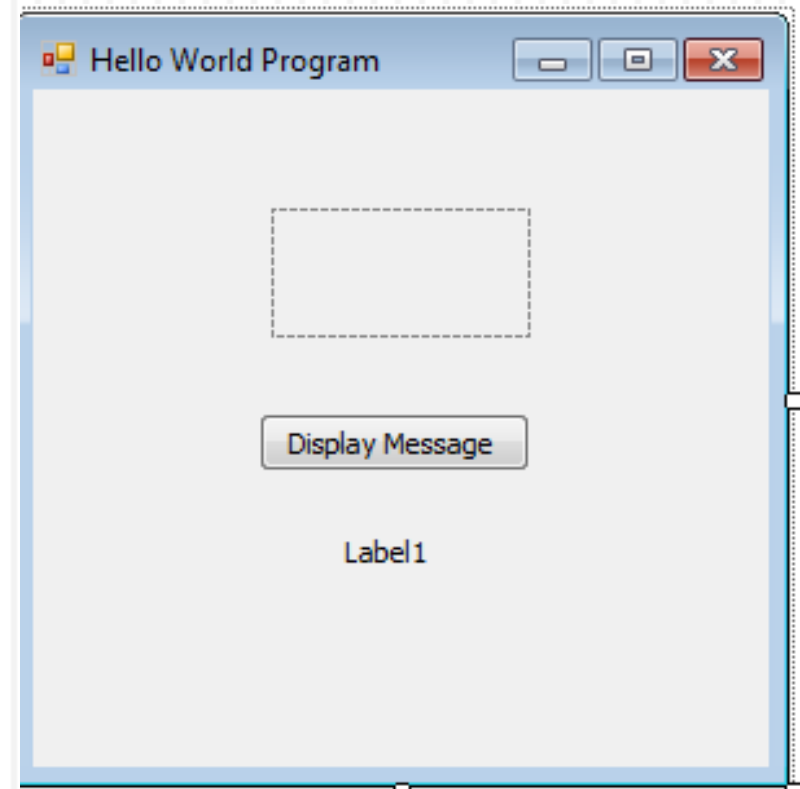
AccessibleDescription	
AccessibleName	
AccessibleRole	Default

Appearance

BackColor	<input type="checkbox"/> Control
BackgroundImage	<input type="checkbox"/> (none)
BackgroundImageLayout	Tile
Cursor	Default

Font

Font	Tahoma, 8pt
ForeColor	<input checked="" type="checkbox"/> ControlText
FormBorderStyle	Sizable
RightToLeft	No
RightToLeftLayout	False
Text	Hello World Program



Edit Image Properties



PictureBox1 System.Windows.Forms.PictureBox

Accessibility

AccessibleDescription	
AccessibleName	
AccessibleRole	Default


Appearance

BackColor	<input type="checkbox"/> Control
BackgroundImage	<input type="checkbox"/> (none)
BackgroundImageLayout	Tile
BorderStyle	None
Cursor	Default

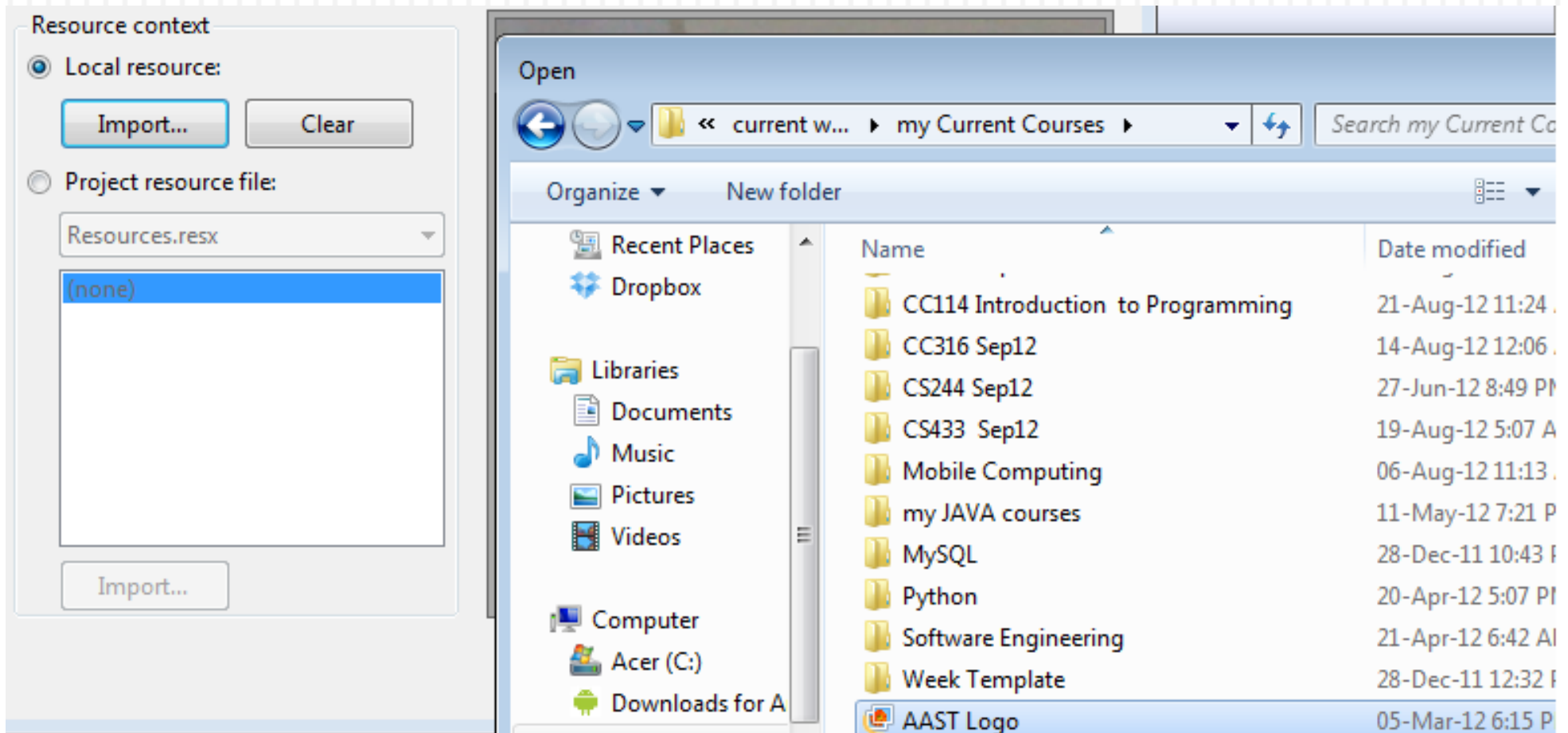
 **+** Image  **System.Drawing.Bitmap**

UseWaitCursor	False
---------------	-------

Asynchronous

+ ErrorImage  System.Drawing.Bitmap



Browse for an Image



Edit the SizeMode property of the PictureBox

PictureBox1 System.Windows.Forms.PictureBox

Asynchronous


ErrorImage	 System.Drawing.Bitmap
ImageLocation	
InitialImage	 System.Drawing.Bitmap
WaitOnLoad	False

Behavior

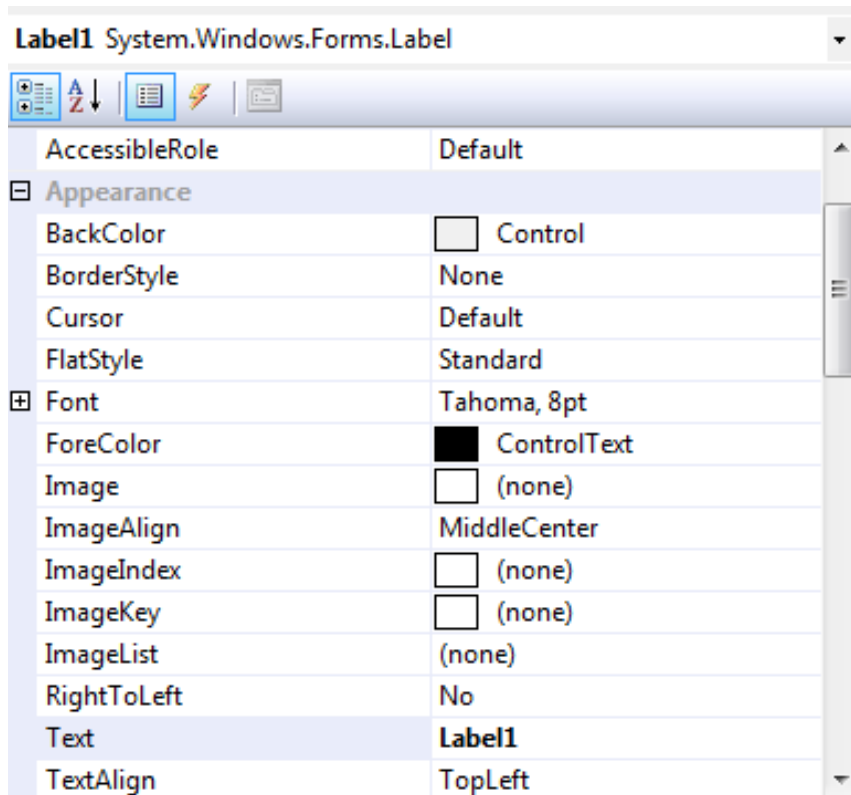
ContextMenuStrip	(none)
Enabled	True
SizeMode	StretchImage
Visible	True

Data

(ApplicationSettings)	
(DataBindings)	
Tag	

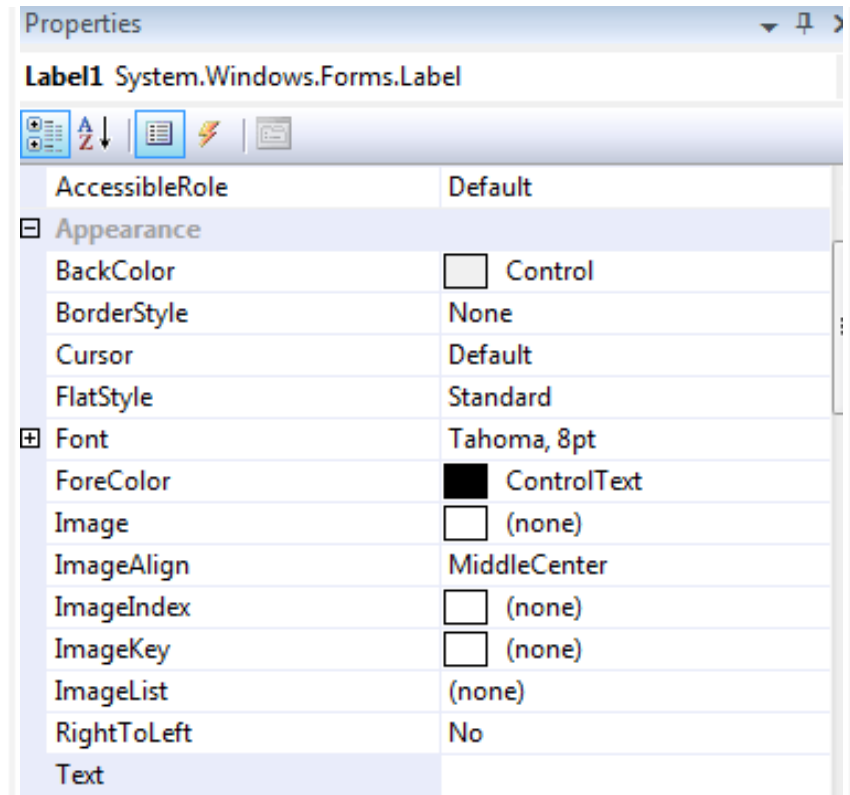


Edit the Text Property of Label1



Label1 System.Windows.Forms.Label

AccessibleRole	Default
Appearance	
BackColor	<input type="checkbox"/> Control
BorderStyle	None
Cursor	Default
FlatStyle	Standard
Font	
ForeColor	<input checked="" type="checkbox"/> ControlText
Image	<input type="checkbox"/> (none)
ImageAlign	MiddleCenter
ImageIndex	<input type="checkbox"/> (none)
ImageKey	<input type="checkbox"/> (none)
ImageList	(none)
RightToLeft	No
Text	Label1
TextAlign	TopLeft

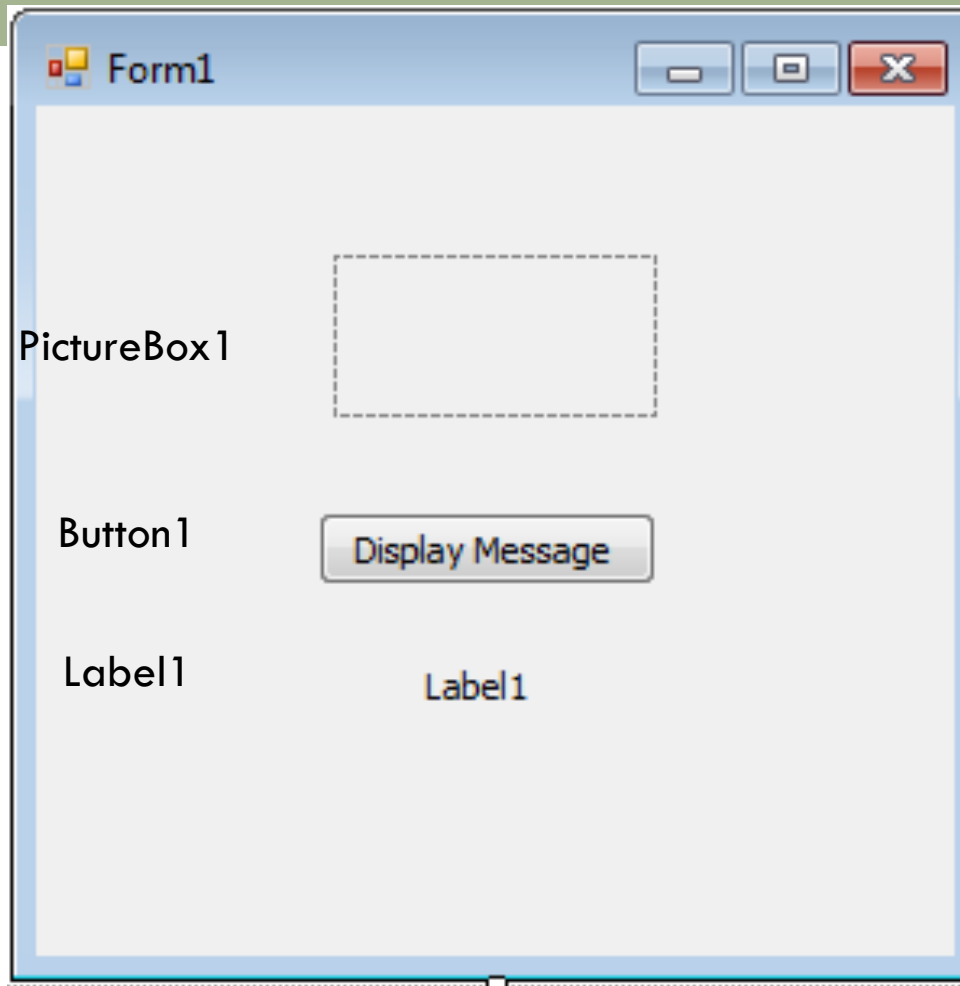


Label1 System.Windows.Forms.Label

AccessibleRole	Default
Appearance	
BackColor	<input type="checkbox"/> Control
BorderStyle	None
Cursor	Default
FlatStyle	Standard
Font	
ForeColor	<input checked="" type="checkbox"/> ControlText
Image	<input type="checkbox"/> (none)
ImageAlign	MiddleCenter
ImageIndex	<input type="checkbox"/> (none)
ImageKey	<input type="checkbox"/> (none)
ImageList	(none)
RightToLeft	No
Text	

The Text Property is changed to “ “

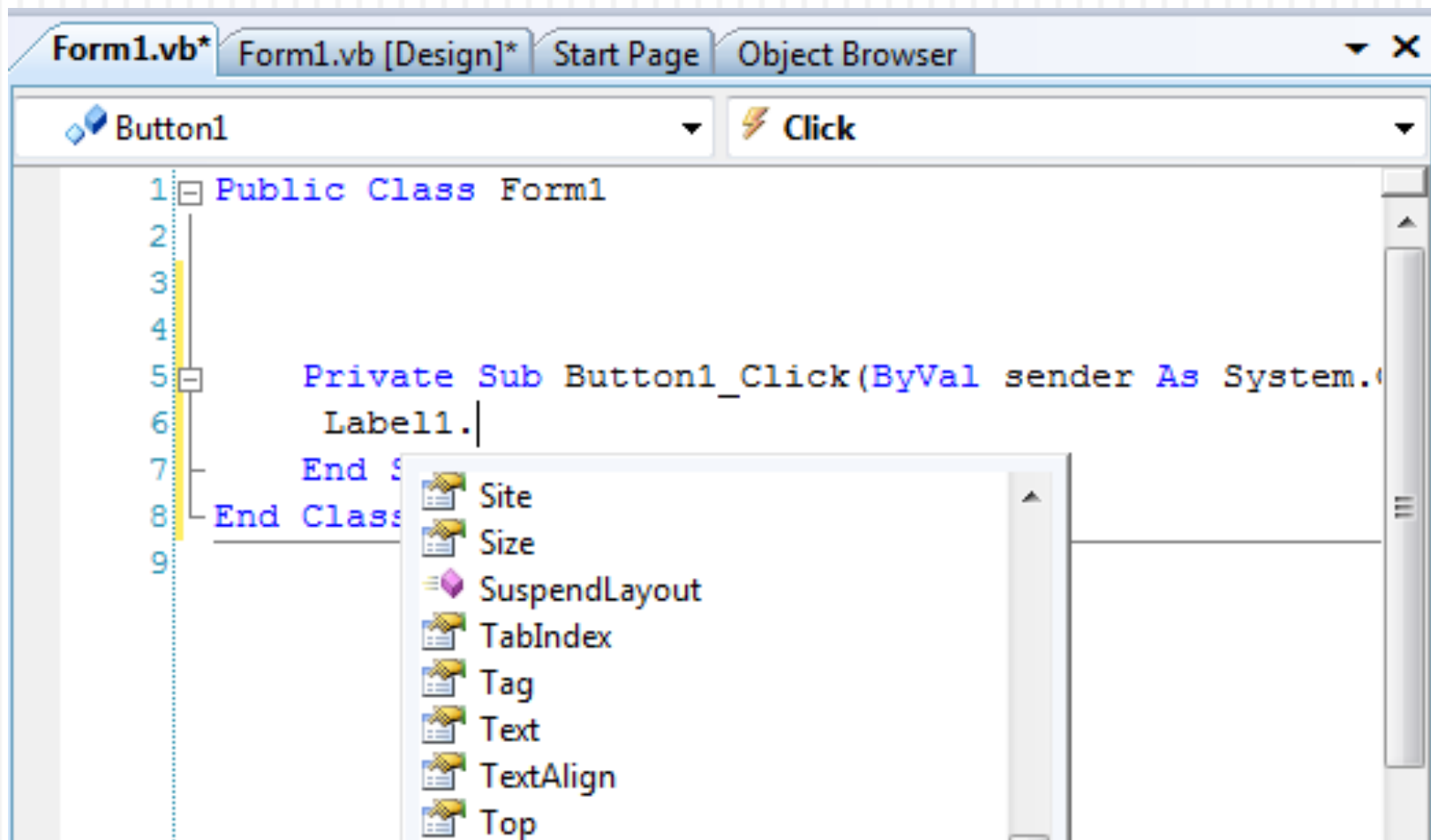
Accessing components properties from code



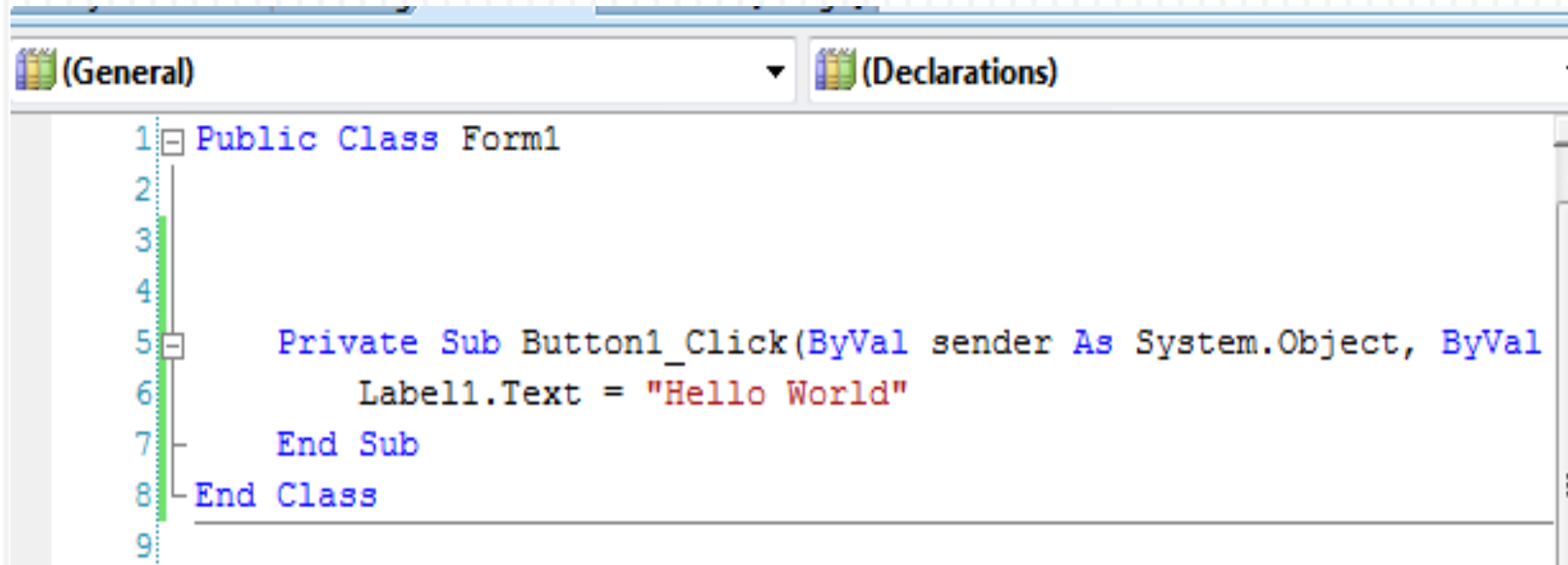
Each Component has a name, you access its property by: **componentName.PropertyName**

Step 3: Writing the Code (Events)

Double click on the Button then write your code



Writing the Code (Events)



The screenshot shows a code editor window with two tabs: "(General)" and "(Declarations)". The "(Declarations)" tab is active, displaying the following code:

```
1 Public Class Form1
2
3
4
5     Private Sub Button1_Click(ByVal sender As System.Object, ByVal
6         Label1.Text = "Hello World"
7     End Sub
8 End Class
9
```


Running the Application

