# Visual Basic for Kids

2. The Visual Basic Environment

Review and Preview

In Class 1, we learned the important parts of a Visual Basic **project**. We saw that a project is built on a **form** using **controls**. By interacting with the controls using **events**, we get the computer to do assigned tasks via instructions we provide. In this second class, we will learn the beginning steps of building our own Visual Basic projects by looking at the different parts of the project and where they fit in the Visual Basic environment. Like Class 1, there are also a lot of new terms and skills to learn.

Parts of a Visual Basic Project

In Class 1, we saw that there are three major components in a Visual Basic project: the project itself, the form, and the controls. **Project** is the word used to encompass everything in a Visual Basic project. Other words used to describe a project are **application** or **program**. The **Form** is the window where you create the interface between the user and the computer. **Controls** are graphical features or tools that are placed on forms to allow user interaction (text boxes, labels, scroll bars, command buttons). Recall the form itself is a control. Controls are also referred to as **objects**. Pictorially, a project is:
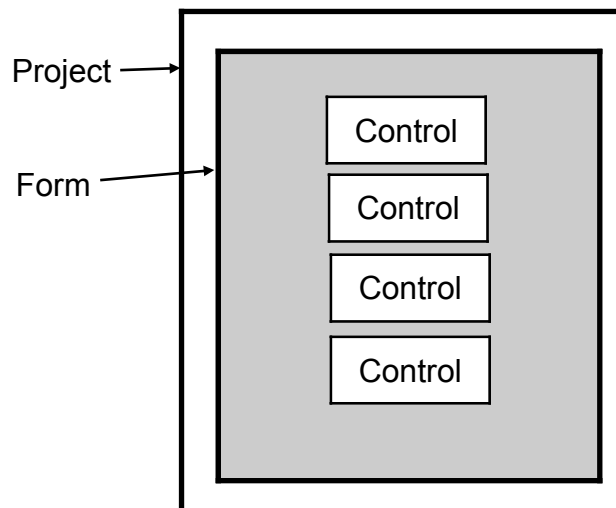
So, a project consists of a form containing several controls. In looking around your computer's file directory, you may find some files associated with a Visual Basic project. Two primary files are used to save a Visual Basic project. The project file will have an extension of **vbp** (in addition, there is sometimes a file with a vbw extension). The form file has an extension of **frm** (in addition, there is sometimes a form file with a frx extension). Look in your project folder (**VB4Projects**, **VB5Projects**, or **VB6Projects**) and see if you can find **Sample.vbp** and **Sample.frm** - those are the files for the project you opened in Class 1. You need to know these extensions for proper opening and saving of your projects.

An important concept concerning a Visual Basic project is that of a **property**. Every characteristic of a control (including the form itself) is specified by a property. Example properties include names, captions, sizes, colors, position on the form, and contents. For each control studied in this class, we will spend a lot of time talking about properties.

Also, in Class 1, we discovered that Visual Basic is an **event-driven** language. Visual Basic is governed by an **event processor**. That means that nothing happens in a Visual Basic project until some event occurs. Once an event is detected, the project finds a series of instructions related to that event, called an **event procedure**. That procedure is executed, then program control is returned to the event:
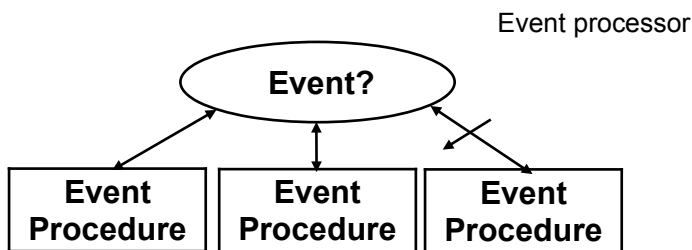
Event procedures are where we do the actual computer programming and are saved with the form in the file with the **frm** extension. These procedures are where we write BASIC language statements. You will learn a lot of programming and BASIC language in this class. The BASIC you will learn is very similar to the original BASIC used by Bill Gates and Paul Allen when starting Microsoft.

Parts of the Visual Basic Program

Visual Basic is more than just a computer language. It is a project building environment. Within this one environment, we can begin and build our project, run and test our project, eliminate errors (if any) in our project, and save our project for future use. With other computer languages, many times you need a separate text editor to write your program, something called a compiler to create the program, and then a different area to test your program. Visual Basic integrates each step of the project building process into one environment. Let's look at the parts of the Visual Basic environment.

Main Window

Start Visual Basic using the procedure learned in Class 1. Notice that several windows appear. The **Main Window** is used to control most aspects of the Visual Basic project building and running process.
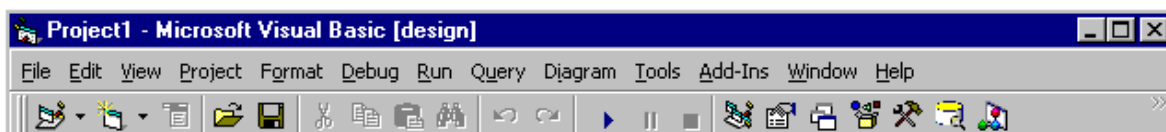
VB4:

VB5:

VB6:

The main window consists of the title bar, menu bar, and toolbar. The title bar indicates the project name and the current Visual Basic operating mode (design, break, run). The menu bar has drop-down menus from which you control the operation of the Visual Basic environment. The toolbar has buttons that provide shortcuts to some of the menu options. Look for the buttons we used in Class 1 to open a project, start a project, and stop a project.

Form Window

The **Form Window** is central to developing Visual Basic applications. It is where you develop your application.
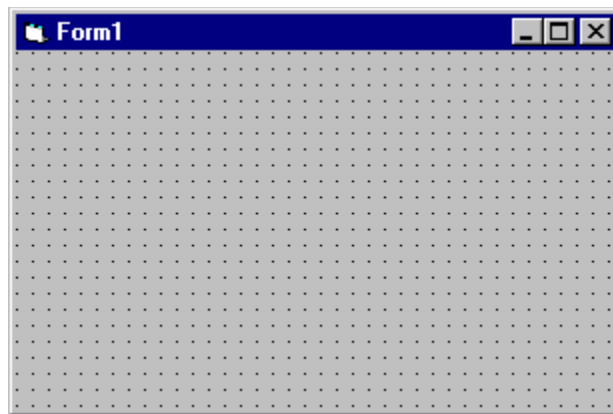
If the form window is not present on the screen:

    VB4:   Click **View** on the main menu, then **Form**.
    VB5, VB6:   Click **View** on the main menu, then **Object**.

As an alternate, if the window does not show up, press the **F7** function key while holding down <**Shift**>.

Toolbox Window

The **Toolbox Window** is the selection menu for controls used in your application. Many times, controls are also referred to as **tools**. So, three words are used to describe controls: objects, tools, and, most commonly, controls.

VB4:

Pointer	Picture Box
Label	Text Box
Frame	Command Button
Check Box	Option Button
Combo Box	List Box
Horizontal Scroll Bar	Vertical Scroll Bar
Timer	Drive
Directory	Files
Shape	Line
Image	Data Control
OLE	Common Dialog Box

VB5, VB6:

Pointer	▶	🖼	Picture Box
Label	A	abl	Text Box
Frame	xy	▭	Command Button
Check Box	☑	◉	Option Button
Combo Box	▤	▤	List Box
Horizontal Scroll Bar	◀▶	▲▼	Vertical Scroll Bar
Timer	⏱	▭	Drive
Directory	📁	📄	Files
Shape	🔷	╲	Line
Image	🖼	▥	Data Control
OLE	OLE		

If the toolbox window is not present on the screen, click **View** on the main menu, then **Toolbox**. See if you can identify controls we used in Class 1 with our **Sample** project.
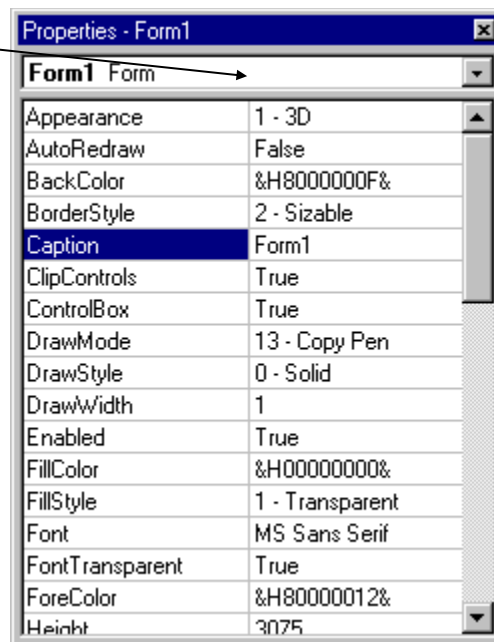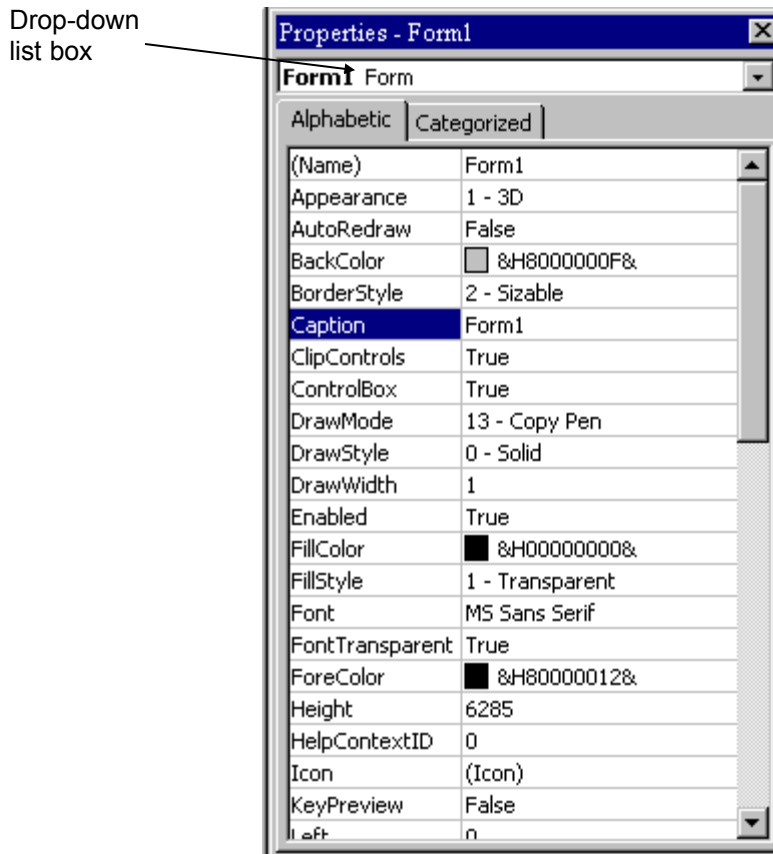
Properties Window

The **Properties Window** is used to establish initial property values for controls. The drop-down box at the top of the window lists all controls on the current form. Under this box are the available properties for the currently selected object.

VB4:

Drop-down
list box

Properties - Form1	☒
Form1 Form	▾
Appearance	1 - 3D
AutoRedraw	False
BackColor	&H8000000F&
BorderStyle	2 - Sizable
Caption	Form1
ClipControls	True
ControlBox	True
DrawMode	13 - Copy Pen
DrawStyle	0 - Solid
DrawWidth	1
Enabled	True
FillColor	&H00000000&
FillStyle	1 - Transparent
Font	MS Sans Serif
FontTransparent	True
ForeColor	&H80000012&
Height	3075

VB5, VB6: Two views are available: **Alphabetic** and **Categorized**. We will always used the Alphabetic view.

Drop-down
list box

Properties - Form1	☒
Form1 Form	▾

Alphabetic | Categorized

(Name)	Form1	▲
Appearance	1 - 3D	
AutoRedraw	False	
BackColor	☐ &H8000000F&	
BorderStyle	2 - Sizable	
Caption	Form1	
ClipControls	True	
ControlBox	True	
DrawMode	13 - Copy Pen	
DrawStyle	0 - Solid	
DrawWidth	1	
Enabled	True	
FillColor	■ &H00000000&	
FillStyle	1 - Transparent	
Font	MS Sans Serif	
FontTransparent	True	
ForeColor	■ &H80000012&	
Height	6285	
HelpContextID	0	
Icon	(Icon)	
KeyPreview	False	
Left	0	▼

If the properties window is not present on the screen:

>   **VB4**:   Click **View** on the main menu, then **Properties**.
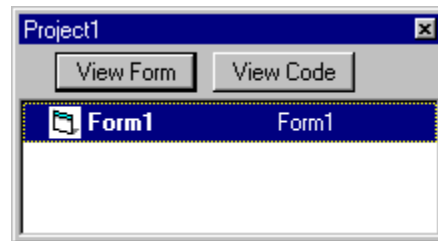>   **VB5, VB6**:   Click **View** on the main menu, then **Properties Window**.

As an alternate, if the window does not show up, press the **F4** function key. The properties window will only display when the form and its controls is also displayed.
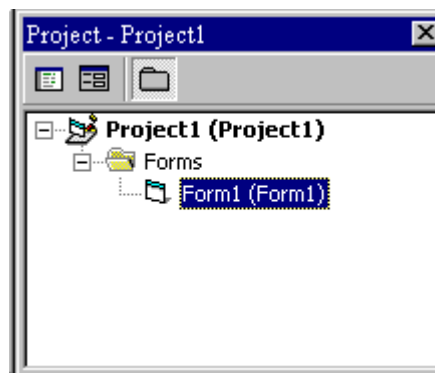
Project Window

The **Project Window** shows which form makes up your project. As you become a more proficient Visual Basic programmer, you will learn to develop projects with more than one form. In that case, all forms in your project are listed in this window. You can also obtain a view of the **Form** or **Code** windows (window containing the actual BASIC coding) from the Project window by clicking one of the displayed buttons. We will look at the code window in the next section.

VB4:

VB5, VB6:

If the project window is not present on the screen:

       VB4:   Click **View** on the main menu, then **Project**.

       VB5, VB6:   Click **View** on the main menu, then **Project Explorer**.

As an alternate, if the window does not show up, press the **R** key while holding down <**Ctrl**>.
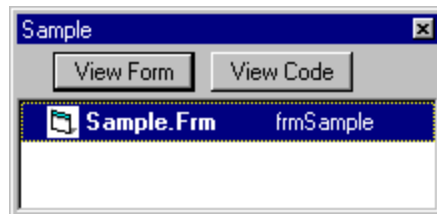
     You should be familiar with each of these windows and know where they are and how to locate them, if they are not displayed. Next, we'll revisit the project we used in Class 1 to illustrate some of the points we've covered here.
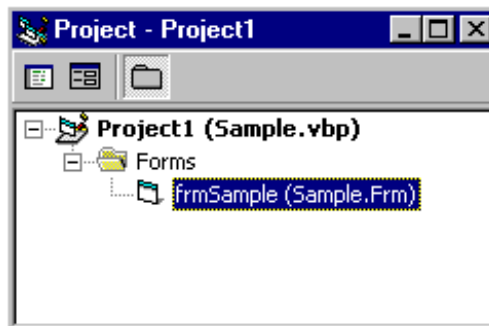
Moving Around in Visual Basic

Project Window

Open the project named **Sample** that we used in Class 1. Find and examine the **Project** window:
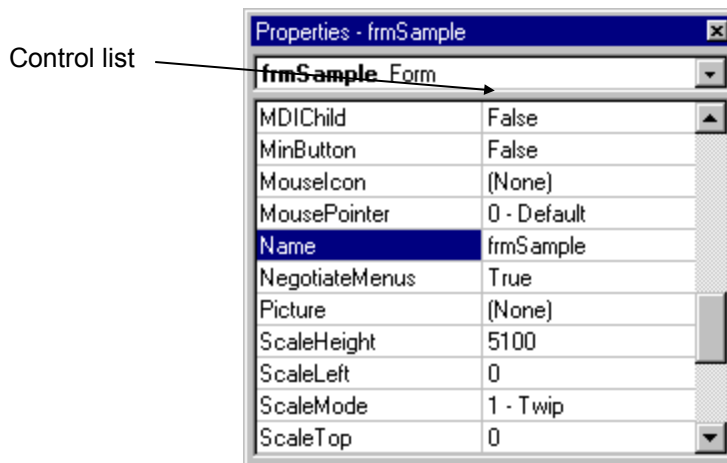
VB4:

VB5, VB6:

The project window indicates we have a project file saved as **Sample.vbp** which includes a single form saved as **Sample.frm**. Note, too, that the form is referred to by a name **frmSample**. Remember the **Properties** window? One of the most important properties we assign to a control is its **Name** - **frmSample** is the name property of our form. We always refer to controls by their name property, so it is critical that we are careful in how we name them (we will talk a lot about this in the next class). The name is so important here that it is listed in the project window.
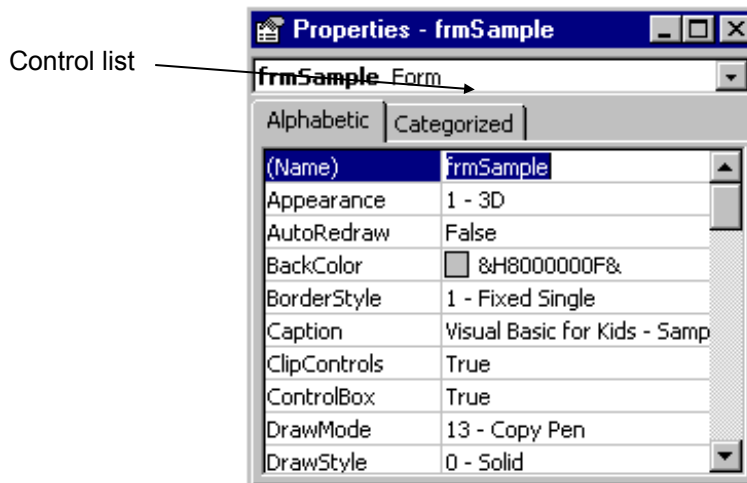
Properties Window

Now, find the **Properties** window. Remember it can only be shown when the form is displayed. So, you may have to make sure the form is displayed first. Review the steps that get the desired windows on your screen.

VB4:

Control list

Properties - frmSample	
frmSample Form	
MDIChild	False
MinButton	False
MouseIcon	(None)
MousePointer	0 - Default
Name	frmSample
NegotiateMenus	True
Picture	(None)
ScaleHeight	5100
ScaleLeft	0
ScaleMode	1 - Twip
ScaleTop	0

VB5, VB6:

Control list

| Properties - frmSample	
frmSample Form	
Alphabetic	Categorized
(Name)	frmSample
Appearance	1 - 3D
AutoRedraw	False
BackColor	&H8000000F&
BorderStyle	1 - Fixed Single
Caption	Visual Basic for Kids - Samp
ClipControls	True
ControlBox	True
DrawMode	13 - Copy Pen
DrawStyle	0 - Solid

The drop-down box at the top of the properties window is called the **control list**. It displays the name (the name property) of each control used in the project, as well as the type of control it is. Notice, as displayed, the current control is the

Form and it is named **frmSample**. The properties list is directly below this box.
In this list, you can scroll (using the scroll bar) through the properties for the
selected control. The property name is on the left side of the list and the current
property value is on the right side. Scroll through the properties for the form. Do
you see how many properties there are? You'll learn about many of these as you
continue through the course. Don't worry about them for now, though.

Click on the down arrow in the control list (remember that's the drop-down
box at the top of the properties window).
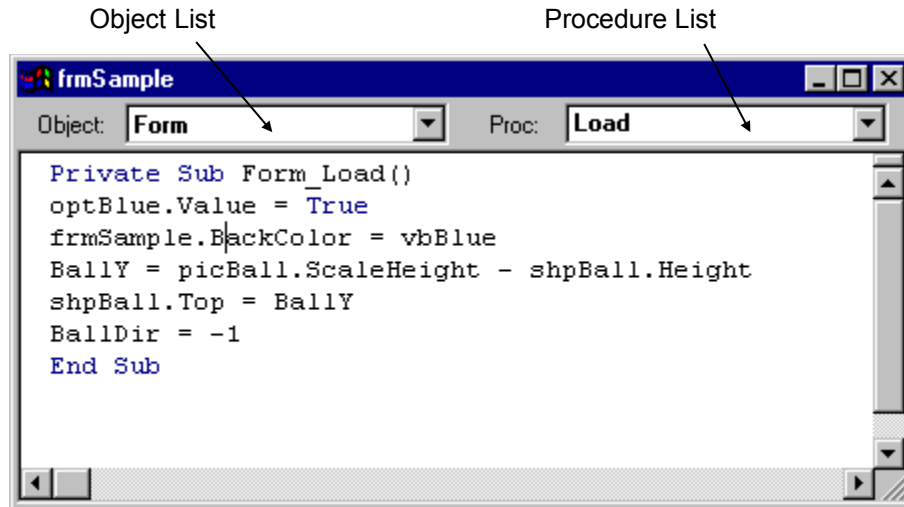
VB4:
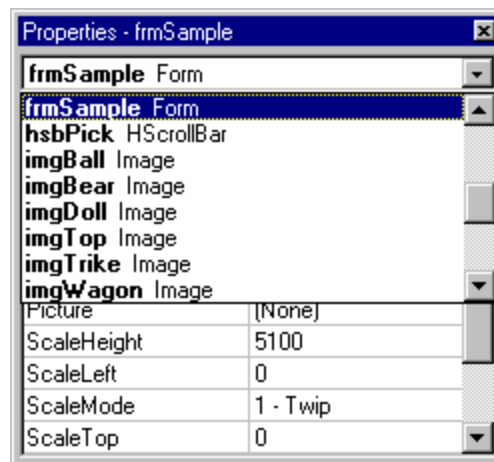
VB5, VB6:
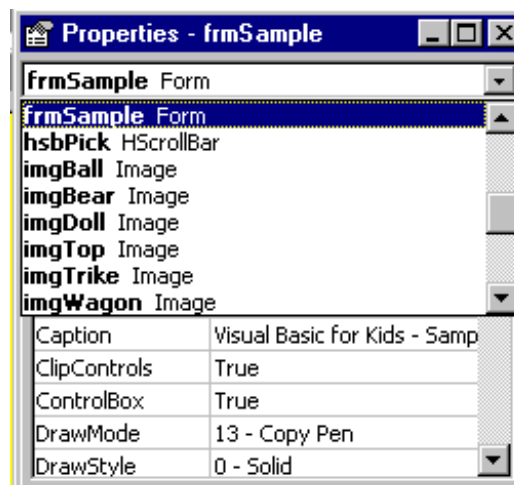
Scroll through the displayed list of all the controls on the form. There are a lot of them. Notice the assigned names and control types. Notice it's pretty easy to identify which control the name refers too. For example, **imgTrike** is obviously the **image** control holding a picture of a trike. That's the idea of proper control naming - making it easy to identify a control just by it's name. As we said, we'll spend time talking about control naming in the later classes.

Select a control and scroll through the properties for that control. Look at the properties for several controls. Notice every control has many properties. Most properties are assigned by default, that is the values are given to it by Visual Basic. We will change some properties from their default values to customize them for our use. We will look at how to change properties in Class 3.
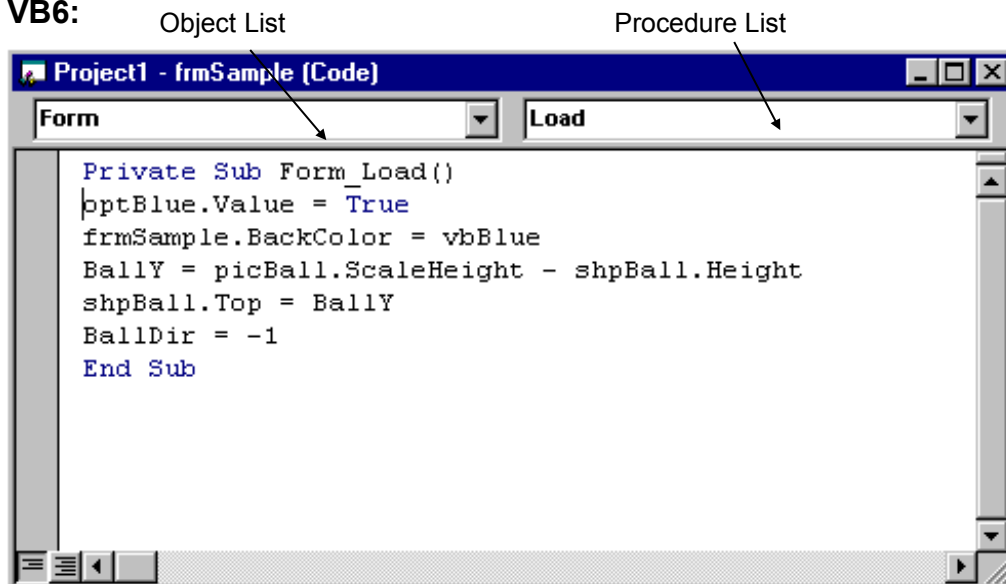
Code Window

Let's look at a new window. Recall Visual Basic is event-driven - when an event is detected, the project goes to the correct **event procedure**. Event procedures are used to tell the computer what to do in response to an event. They are where the actual computer programming (using the BASIC language) occurs. We view the event procedures in the **Code Window**. There are many ways to display the code window. One way is to use the appropriate button found in the project window. Another is to click **View** on the main menu, then **Code**. Or, as an alternate, press the **F7** function key. Find the code window for the **Sample** project.

VB4:

Object List   Procedure List

```
frmSample                                    _ □ ✕
Object: Form              ▼   Proc:  Load              ▼

Private Sub Form_Load()                          ▲
optBlue.Value = True
frmSample.BackColor = vbBlue
BallY = picBall.ScaleHeight – shpBall.Height
shpBall.Top = BallY
BallDir = -1
End Sub
                                                 ▼
◄  □                                           ►
```

VB5, VB6:

Object List   Procedure List

```
Project1 - frmSample (Code)                      _ □ ✕
Form                     ▼    Load                ▼

Private Sub Form_Load()                          ▲
optBlue.Value = True
frmSample.BackColor = vbBlue
BallY = picBall.ScaleHeight – shpBall.Height
shpBall.Top = BallY
BallDir = -1
End Sub

                                                 ▼
≡ ≣ ◄  □                                       ►
```
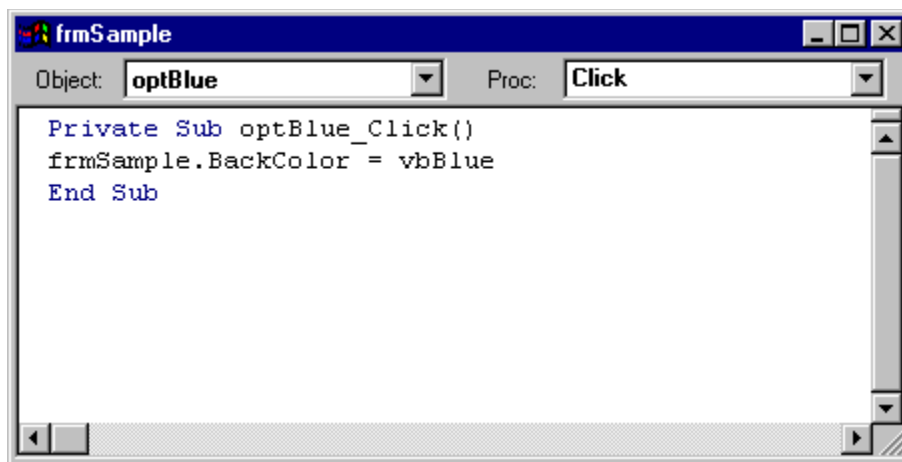
At the top of the code window are two boxes, the **object** (or control) **list** and the **procedure list**. The object list is similar to the control list in the properties window. It lists all objects on the form by name. Once an object, or control, name is selected in that list, the procedure list shows all possible event procedures for that control. As shown, the code window is displaying the **Load** event procedure for the **Form** control. (If the Form control is not selected in the object list, click the drop-down arrow, scroll until you find the word **Form** and select it). The Load event occurs when a project is first loaded into your computer. Under the object
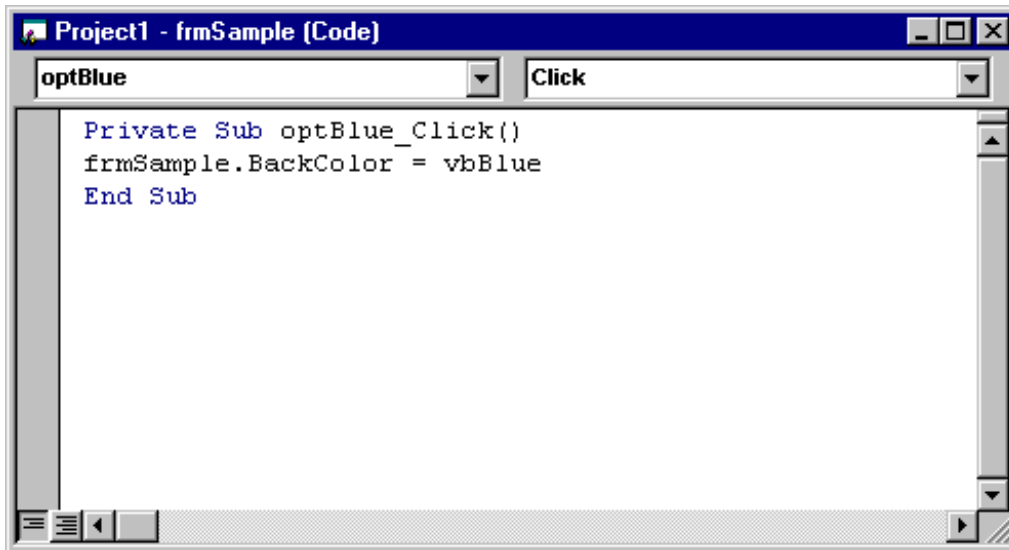
and procedure list boxes is the actual BASIC code for that event procedure. This code probably looks like a foreign language right now and don't worry - it should! In subsequent classes, you will start to learn BASIC and such code will become easy to read. Click on the drop-down arrow in the procedures list box. Notice all the other possible event procedures for the Form control. Fortunately, we don't have to write BASIC code for all of these procedures. We only write code for events we expect will happen when our project is running.

Click on the drop-down arrow in the objects list. Select **optBlue** as the object.

VB4:

```
frmSample
Object: optBlue          Proc: Click

 Private Sub optBlue_Click()
 frmSample.BackColor = vbBlue
 End Sub
```

VB5, VB6:

```
Project1 - frmSample (Code)                          _ □ ✕
optBlue                    ▼   Click                        ▼
      Private Sub optBlue_Click()                          ▲
      frmSample.BackColor = vbBlue
      End Sub
```

This is the **Click** event procedure for the control name **optBlue**. And even though you may not know any BASIC right now, you should be able for figure out what is going on here. Since we are careful in how we name controls, you should recognize this control to be the option button (one with a little circle) with the word **Blue** next to it (the word next to an option button is its **Caption** property). So, this event procedure is called whenever we click on the Blue option button. Notice the procedure has a single line of instruction (ignore the first and last line for now):

```
frmSample.BackColor = vbBlue
```

What this line of BASIC code says is set the **BackColor** property of the control named **frmSample** to Blue (represented by the word **vbBlue**). Pretty easy, huh?

Actually, most BASIC code is pretty easy to understand. So, don't be too worried. Select other objects in the object list and look at the corresponding event procedures. Look at the BASIC code. Start at the top line of code (again, ignore the top header line) and work down. Can you figure out what's going on even though you don't know any BASIC? I think you'll find you can in most cases. Writing BASIC code is primarily paying attention to lots of details. For the most

part, it's very logical and obvious. And, you're about to start writing your own code!

Summary

In this second class, we've learned the parts of the Visual Basic environment and how to move around in that environment. We've also learned some important new terms like **properties** and **event procedures**. You're now ready to build your first Visual Basic project. In the next class, you'll learn how to place controls on a form, move them around, and make them appear just like you want. And, you will learn the all-important step of how to put BASIC code in the event procedures.