Run-Time Errors

If your project halts during execution, it is called a *run-time error* or an *exception*. Visual Basic displays a dialog box and highlights the statement causing the problem.

Statements that cannot execute correctly cause run-time errors. The statements are correctly formed Basic statements that pass the syntax checking; however, the statements fail to execute due to some serious issue. You can cause run-time errors by attempting to do impossible arithmetic operations, such as calculate with nonnumeric data, divide by zero, or find the square root of a negative number.

In Chapter 3 you will learn to catch exceptions so that the program does not come to a halt when an error occurs.

Logic Errors

When your program contains logic errors, your project runs but produces incorrect results. Perhaps the results of a calculation are incorrect or the wrong text appears or the text is okay but appears in the wrong location.

Beginning programmers often overlook their logic errors. If the project runs, it must be right—right? All too often, that statement is not correct. You may need to use a calculator to check the output. Check all aspects of the project output: computations, text, and spacing.

For example, the Hello World project in this chapter has event procedures for displaying "Hello World" in English and in Spanish. If the contents of the two procedures were switched, the program would work but the results would be incorrect.

The following code does not give the proper instructions to display the message in Spanish:

```
Private Sub SpanishButton_Click
   ' Display the Hello World Message in Spanish.

  MessageLabel.Text = "Hello World"
End Sub
```

Project Debugging

If you talk to any computer programmer, you will learn that programs don't have errors, but that programs get "bugs" in them. Finding and fixing these bugs is called ***debugging***.

For syntax errors and run-time errors, your job is easier. Visual Basic displays the Editor window with the offending line highlighted. However, you must identify and locate logic errors yourself.

VB has a very popular feature: edit-and-continue. If you are able to identify the run-time error and fix it, you can continue project execution from that location by clicking on the *Continue* button, pressing F5, or choosing *Debug / Continue*. You also can correct the error and restart from the beginning.

The Visual Studio IDE has some very helpful tools to aid in debugging your projects. The debugging tools are covered in Chapter 4.