**Convert temperatures by using a Do loop**

1. On the File menu, click New Project. The New Project dialog box opens.

**2.** Create a new Visual Basic Windows Forms Application project named My Celsius Conversion. The new project is created, and a blank form opens in the Designer. This time, you’ll place all the code for your program in the Form1\_Load event procedure so that Visual Basic immediately prompts you for the Fahrenheit temperature when you start the application. You’ll use an InputBox function to request the Fahrenheit data, and you’ll use a MsgBox function to display the converted value.

**3.** Double-click the form. The Form1\_Load event procedure appears in the Code Editor.

**4**. Type the following program statements in the Form1\_Load event procedure:

Dim FTemp, Celsius As Single

Dim strFTemp As String

Dim Prompt As String = "Enter a Fahrenheit temperature."

Do

strFTemp = InputBox(Prompt, "Fahrenheit to Celsius")

If strFTemp <> "" Then

FTemp = CSng(strFTemp) Celsius = Int((FTemp + 40) \* 5 / 9 - 40)

MsgBox(Celsius, , "Temperature in Celsius")

End If

Loop While strFTemp <> ""

End

**Tip**. Be sure to include the End statement at the bottom of the Form1\_Load event procedure. When the user has had his or her fill of converting temperatures, this is how the program terminates.

This code handles the calculations for the project. The first line declares two single-precision variables, FTemp and Celsius, to hold the Fahrenheit and Celsius temperatures, respectively. The second line declares a string variable named strFTemp that holds a string version of the Fahrenheit temperature. The third line declares a string variable named Prompt, which will be used in the InputBox function, and assigns it an initial value. The Do loop repeatedly prompts the user for a Fahrenheit temperature, converts the number to Celsius, and then displays it on the screen by using the MsgBox function.

The value that the user enters in the input box is stored in the strFTemp variable. The InputBox function always returns a value of type String, even if the user enters numbers. Because we want to perform mathematical calculations on the entered value, strFTemp must be converted to a number. The CSng function is used to convert a string into the Single data type. CSng is one of many conversion functions you can use to convert a string to a different data type. The converted single value is then stored in the FTemp variable.

The loop executes until the user clicks the Cancel button or until the user presses ENTER or clicks OK with no value in the input box. Clicking the Cancel button or entering no value returns an empty string (“”). The loop checks for the empty string by using a While conditional test at the bottom of the loop. The program statement:

Celsius = Int((FTemp + 40) \* 5 / 9 - 40)

handles the conversion from Fahrenheit to Celsius in the program. This statement employs a standard conversion formula, but it uses the Int function to return a value that contains no decimal places to the Celsius variable. (Everything to the right of the decimal point is discarded.) This cutting sacrifices accuracy, but it helps you avoid long, unsightly numbers such as 21.11111, the Celsius value for 70 degrees Fahrenheit.

**5.** Click the Save All button on the Standard toolbar to save your changes. Now you’ll try running the program.

**6**. Click the Start Debugging button on the Standard toolbar. The program starts, and the InputBox function prompts you for a Fahrenheit temperature.

**7.** Type 212

**8.** Click OK. The temperature 212 degrees Fahrenheit is converted to 100 degrees Celsius, as shown in this message box.

**9**. Click OK. Then type 72 in the input box, and click OK again. The temperature 72 degrees Fahrenheit is converted to 22 degrees Celsius.

**10.** Click OK, and then click Cancel in the input box. The program closes, and the development environment returns.

**11.** Get Ticked-Off