#### The Birthday Program

The Birthday program uses a *DateTimePicker* control and a *Button* control to prompt the user for the date of his or her birthday . It then displays that information by using a message box . Give it a try now .

**Build the Birthday program**

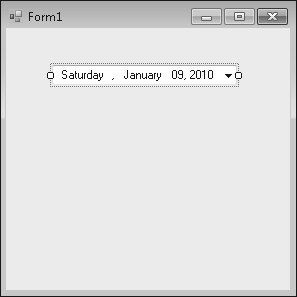
1. On the File menu, click New Project .

The New Project dialog box opens .

1. Create a new Visual Basic Windows Forms Application project named **MyBirthday** .

The new project is created, and a blank form appears in the Designer .

1. Click the *DateTimePicker* control in the Toolbox .
2. Draw a date/time picker object near the top of the form, as shown in the following screen shot:



The date/time picker object by default displays the current date, but you can adjust the displayed date by changing the object’s *Value* property . Displaying the date is a handy design guide—it lets you size the date/time picker object appropriately when you’re creating it .

1. Click the *Button* control in the Toolbox, and then add a button object below the date/time picker .

You’ll use this button to display your birth date and to verify that the date/time picker works correctly .

1. In the Properties window, change the *Text* property of the button object to **Show My Birthday** .

Now you’ll add a few lines of program code to a procedure associated with the button object . This is an event procedure because it runs when an event, such as a mouse click, occurs, or *fires,* in the object .

1. Double-click the button object on the form to display its default event procedure, and then type the following program statements between the *Private Sub* and *End Sub* statements in the *Button1\_Click* event procedure:

**MsgBox("Your birth date was " & DateTimePicker1.Text)**

**MsgBox("Day of the year: " & \_**

**DateTimePicker1.Value.DayOfYear.ToString())**

These program statements display two message boxes (small dialog boxes) with information from the date/time picker object . The first line uses the *Text* property of the date/time picker to display the birth date information that you select when using the object at run time . The *MsgBox* function displays the string value “Your birth date was” in addition to the textual value held in the date/time picker’s *Text* property . These two pieces of information are joined together by the string concatenation operator (&) .

The second and third lines collectively form one program statement and have been broken by the line continuation character (\_) because the statement was a bit too long to print in this book .

Program lines can be more than 65,000 characters long in the Visual Studio Code Editor, but it’s usually easiest to work with lines of 80 or fewer characters . You can divide long program statements among multiple lines by using a space and a line continuation character (\_) at the end of each line in the statement except for the last line . (You cannot use a line continuation character to break a string that’s in quotation marks, however .) I use the line continuation character in this exercise to break the second line of code into two parts .

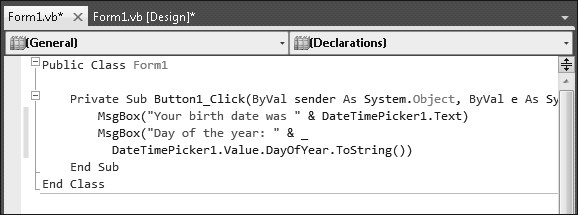
**Note** Starting in Visual Basic 2010, the line continuation character (\_) is optional . There are a few instances where the line continuation character is needed, but they are rare . In this book, I still use line continuation characters to make it clear where there are long lines, but you don’t have to include them .



The statement DateTimePicker1 .Value .DayOfYear .ToString() uses the date/time picker object to calculate the day of the year in which you were born, counting from January 1 . This is accomplished by the *DayOfYear* property and the *ToString* method, which converts the numeric result of the date calculation to a textual value that’s more easily displayed by the *MsgBox* function .

*Methods* are special statements that perform an action or a service for a particular object, such as converting a number to a string or adding items to a list box . Methods differ from properties, which contain a value, and event procedures, which execute when a user manipulates an object . Methods can also be shared among objects, so when you learn how to use a particular method, you’ll often be able to apply it to several circumstances . We’ll discuss several important methods as you work through this book .

After you enter the code for the *Button1\_Click* event procedure, the Code Editor looks similar to this:



**9.** Click the Save All button to save your changes to disk, and specify C:\Vb10sbs\Chap03 as the folder location .

Now you’re ready to run the Birthday program .

**Run the Birthday program**

**Tip**

The complete Birthday program is located in…opps I almost gave you the answer.

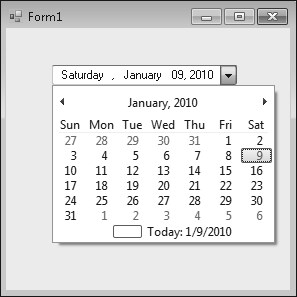


1. Click the Start Debugging button on the Standard toolbar .

The Birthday program starts to run in the IDE . The current date is displayed in the date/time picker .

1. Click the arrow in the date/time picker to display the object in Calendar view .

Your form looks like the following screen shot, but with a different date .



1. Click the Left scroll arrow to look at previous months on the calendar .

Notice that the text box portion of the object also changes as you scroll the date . The “today” value at the bottom of the calendar doesn’t change, however .

Although you can scroll all the way back to your exact birthday, you might not have the patience to scroll month by month . To move to your birth year faster, select the year value in the date/time picker text box and enter a new year .

1. Select the four-digit year in the date/time picker text box .

When you select the date, the date/time picker closes .

1. Type your birth year in place of the year that’s currently selected, and then click the arrow again .

The calendar reappears in the year of your birth .

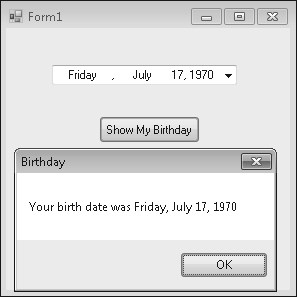
1. Click the scroll arrow again to locate the month in which you were born, and then click the exact day on which you were born .

If you didn’t know the day of the week on which you were born, now you can find out!

When you select the final date, the date/time picker closes, and your birth date is displayed in the text box . You can click the button object to see how this information is made available to other objects on your form .

1. Click the Show My Birthday button .

Visual Basic executes your program code and displays a message box containing the day and date of your birth . Notice how the two dates shown in the two boxes match:



1. Click OK in the message box .

A second message box appears, indicating the day of the year on which you were born—everything seems to work! You’ll find this control to be quite capable—not only does it remember the new date or time information that you enter, but it also keeps track of the current date and time, and it can display this date and time information in a variety of useful formats .

**Note**

To configure the date/time picker object to display times instead of dates, set the

object’s

*Format*

property to Time .



1. Click OK to close the message box, and then click the Close button on the form .

You’re finished using the *DateTimePicker* control for now .