

SOLUTIONS MANUAL

Visual Basic® 2010

HOW TO PROGRAM



PAUL DEITEL
HARVEY DEITEL

Problem Solving
Integrated Debugging
Extensive VideoNotes
Late Objects Approach
Visual Basic® 2010
Windows® 7/Windows 8
User Interfaces • User Interface
Controls • Strings
• Methods • Files • File
Integrated Windows Services
• ASP • Database
Web Apps with ASP.NET
Graphics • Multimedia
Optional Online Chapters:
WPF GUI • WPF • WPF
WPF Graphics and Multimedia
WCF Web Services • Silverlight™
Collections • OO Design Case Study

Introduction to the Visual Basic Express 2008 IDE: Solutions

2

Seeing is believing.

—Proverb

Form ever follows function.

—Louis Henri Sullivan

Intelligence ... is the faculty of making artificial objects, especially tools to make tools.

—Henri-Louis Bergson

Objectives

In this chapter you'll learn:

- The basics of the Visual Studio Integrated Development Environment (IDE) that assists you in writing, running and debugging your Visual Basic programs.
- Visual Studio's help features.
- Key commands contained in the IDE's menus and toolbars.
- The purpose of the various kinds of windows in the Visual Studio 2008 IDE.
- What visual programming is and how it simplifies and speeds program development.
- To create, compile and execute a simple Visual Basic program that displays text and an image using the Visual Studio IDE and the technique of visual programming.

Self-Review Exercises

- 2.1 Fill in the blanks in each of the following statements:
- a) The technique of _____ allows you to create GUIs without writing any code.
ANS: visual programming.
 - b) A(n) _____ is a group of one or more projects that collectively form a Visual Basic program.
ANS: solution.
 - c) The _____ feature hides a window when the mouse pointer is moved outside the window's area.
ANS: auto-hide.
 - d) A(n) _____ appears when the mouse pointer hovers over an icon.
ANS: tool tip.
 - e) The _____ window allows you to browse solution files.
ANS: Solution Explorer
 - f) The properties in the **Properties** window can be sorted _____ or _____.
ANS: alphabetically, categorically.
 - g) A Form's _____ property specifies the text displayed in the Form's title bar.
ANS: Text.
 - h) The _____ allows you to add controls to the Form in a visual manner.
ANS: Toolbox.
 - i) Using _____ displays relevant help articles, based on the current context.
ANS: context-sensitive help.
 - j) The _____ property specifies how text is aligned within a Label's boundaries.
ANS: TextAlign.
- 2.2 State whether each of the following is *true* or *false*. If *false*, explain why.
- a) The title bar of the IDE displays the IDE's mode.
ANS: True.
 - b) The X box toggles auto-hide.
ANS: False. The pin icon toggles auto-hide. The X box closes a window.
 - c) The toolbar icons represent various menu commands.
ANS: True.
 - d) The toolbar contains icons that represent controls you can drag onto a Form.
ANS: False. The **Toolbox** contains icons that represent such controls.
 - e) Both Forms and Labels have a title bar.
ANS: False. Forms have a title bar but Labels do not (although they do have Label text).
 - f) Control properties can be modified only by writing code.
ANS: False. Control properties can be set using the **Properties** window.
 - g) PictureBoxes typically display images.
ANS: True.
 - h) Visual Basic files use the file extension .bas.
ANS: False. Visual Basic files use the file extension .vb.
 - i) A Form's background color is set using the BackColor property.
ANS: True.

Exercises

2.3 Fill in the blanks in each of the following statements:

- When an ellipsis button is clicked, a(n) _____ is displayed.
ANS: dialog. Dialogs are windows that facilitate user-computer communication.
- To save every file in a solution, select _____.
ANS: **File > Save All**.
- Using _____ help immediately displays a relevant help article.
ANS: context-sensitive.
- “GUI” is an acronym for _____.
ANS: graphical user interface.

2.4 State whether each of the following is *true* or *false*. If *false*, explain why.

- You can add a control to a Form by double clicking its control icon in the **Toolbox**.
ANS: True.
- The Form, Label and PictureBox have identical properties.
ANS: False. Each type of control has a different set of properties, although controls can have common properties.
- If your machine is connected to the Internet, you can browse the Internet from the Visual Studio IDE.
ANS: True.
- Visual Basic programmers usually create complex programs without writing any code.
ANS: False. Visual Basic programming usually involves a combination of writing a portion of the program code and having Visual Studio generate the remaining code.
- Sizing handles are visible during execution.
ANS: False. Sizing handles are present only in **Design** view when a Form or control is selected.

2.5 Some features that appear throughout Visual Studio perform similar actions in different contexts. Explain and give examples of how the ellipsis buttons, down-arrow buttons and tool tips act in this manner. Why do you think the Visual Studio IDE was designed this way?

ANS: An ellipsis button indicates that a dialog will be displayed when the button is clicked. The down-arrow button indicates that there are more options, both for toolbar icons and for items in the **Properties** window. Moving the mouse pointer over most icons displays the icon's name as a tool tip. These features make the Visual Studio IDE easier to learn and use.

2.6 Fill in the blanks in each of the following statements:

- The _____ property specifies which image a PictureBox displays.
ANS: Image.
- The _____ menu contains commands for arranging and displaying windows.
ANS: **Window**

2.7 Briefly describe each of the following IDE features:

- toolbar
ANS: A toolbar contains icons that, when clicked, execute a command.
- menu bar
ANS: A menu bar contains menus, which are groups of related commands.
- Toolbox**
ANS: The **Toolbox** contains controls used to customize forms.
- control

ANS: A control is a component, such as a `PictureBox` or `Label`. Controls are added to a Form.

e) Form

ANS: A Form represents the Windows Forms application that you are creating. The Form and controls collectively represent the program's GUI.

f) solution

ANS: A solution is a group of projects.

[*Note:* In the following exercises, you are asked to create GUIs using controls that we have not yet discussed in this book. The exercises give you practice with visual programming only—the programs do not perform any actions. You place controls from the **Toolbox** on a Form to familiarize yourself with what each control looks like. We have provided step-by-step instructions for you. If you follow these, you should be able to replicate the screen images we provide.]

2.8 (*Notepad GUI*) Create the GUI for the notepad as shown in Fig. 2.1.

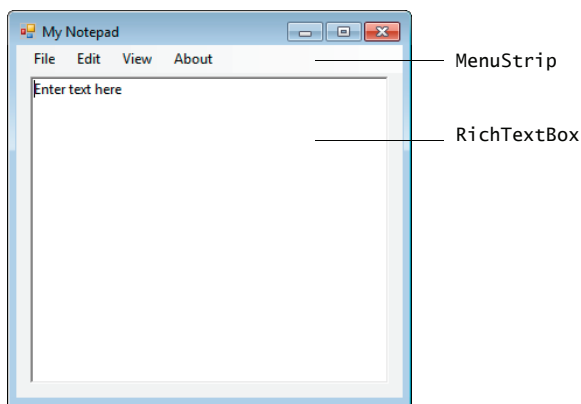


Fig. 2.1 | Notepad GUI.

- Manipulating the Form's properties.** Change the `Text` property of the Form to `My Notepad`. Change the `Font` property to 9pt Segoe UI.
- Adding a MenuStrip control to the Form.** Add a `MenuStrip` to the Form. After inserting the `MenuStrip`, add items by clicking the **Type Here** section, typing a menu name (e.g., `File`, `Edit`, `View` and `About`) and then pressing `Enter`.
- Adding a RichTextBox to the Form.** Drag this control onto the Form. Use the sizing handles to resize and position the `RichTextBox` as shown in Fig. 2.1. Change the `Text` property to `Enter text here`.

2.9 (*Calendar and Appointments GUI*) Create the GUI for the calendar as shown in Fig. 2.2.

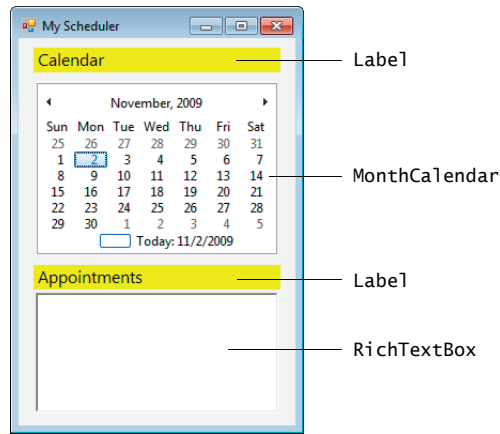


Fig. 2.2 | Calendar and appointments GUI.

- Manipulating the Form's properties.* Change the Text property of the Form to My Scheduler. Change the Font property to 9pt Segoe UI. Set the Form's Size property to 275, 400.
- Adding Labels to the Form.* Add two Labels to the Form. Both should be of equal size (231, 23; remember to set the AutoSize property to False) and should be centered in the Form horizontally, as shown. Set the Label's Text properties to match Fig. 2.2. Use 12-point font size. Also, set the BackColor property to Yellow.
- Adding a MonthCalendar control to the Form.* Add this control to the Form and center it horizontally in the appropriate place between the two Labels.
- Adding a RichTextBox control to the Form.* Add a RichTextBox control to the Form and center it below the second Label. Resize the RichTextBox accordingly.

2.10 (*Calculator GUI*) Create the GUI for the calculator as shown in Fig. 2.3.

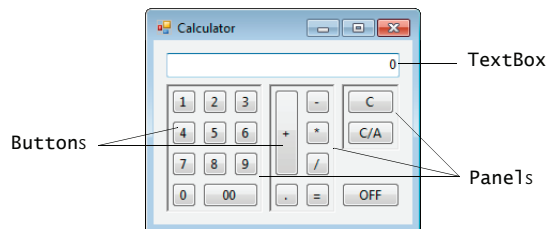


Fig. 2.3 | Calculator GUI.

- Manipulating the Form's properties.* Change the Text property of the Form to Calculator. Change the Font property to 9pt Segoe UI. Change the Size property of the Form to 258, 210.

- b) *Adding a TextBox to the Form.* Set the TextBox's Text property in the Properties window to 0. Stretch the TextBox and position it as shown in Fig. 2.3. Set the TextAlign property to Right—this right aligns text displayed in the TextBox.
- c) *Adding the first Panel to the Form.* Panel controls are used to group other controls. Add a Panel to the Form. Change the Panel's BorderStyle property to Fixed3D to make the inside of the Panel appear recessed. Change the Size property to 90, 120. This Panel will contain the calculator's numeric keys.
- d) *Adding the second Panel to the Form.* Change the Panel's BorderStyle property to Fixed3D. Change the Size property to 62, 120. This Panel will contain the calculator's operator keys.
- e) *Adding the third (and last) Panel to the Form.* Change the Panel's BorderStyle property to Fixed3D. Change the Size property to 54, 62. This Panel contains the calculator's C (clear) and C/A (clear all) keys.
- f) *Adding Buttons to the Form.* There are 20 Buttons on the calculator. Add a Button to the Panel by dragging and dropping it on the Panel. Change the Text property of each Button to the calculator key it represents. The value you enter in the Text property will appear on the face of the Button. Finally, resize the Buttons, using their Size properties. Each Button labeled 0–9, *, /, -, = and . should have a size of 23, 23. The 00 Button has size 52, 23. The OFF Button has size 54, 23. The + Button is sized 25, 64. The C (clear) and C/A (clear all) Buttons are sized 44, 23.

2.11 (*Alarm Clock GUI*) Create the GUI for the alarm clock as shown in Fig. 2.4.

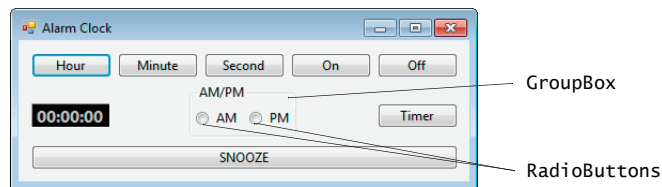


Fig. 2.4 | Alarm clock GUI.

- a) *Manipulating the Form's properties.* Change the Text property of the Form to Alarm Clock. Change the Font property to 9pt Segoe UI. Change the Size property of the Form to 438, 170.
- b) *Adding Buttons to the Form.* Add six Buttons to the Form. Change the Text property of each Button to the appropriate text. Align the Buttons as shown.
- c) *Adding a GroupBox to the Form.* GroupBoxes are like Panels, except that GroupBoxes display a title. Change the Text property to AM/PM, and set the Size property to 100, 50. Center the GroupBox horizontally on the Form.
- d) *Adding AM/PM RadioButtons to the GroupBox.* Place two RadioButtons in the GroupBox. Change the Text property of one RadioButton to AM and the other to PM. Align the RadioButtons as shown.
- e) *Adding the time Label to the Form.* Add a Label to the Form and change its Text property to 00:00:00. Change the BorderStyle property to Fixed3D and the BackColor to Black. Use the Font property to make the time bold and 12pt. Change the ForeColor to Silver (located in the Web tab) to make the time stand out against the black background. Position the Label as shown.

2.12 (Radio GUI) Create the GUI for the radio as shown in Fig. 2.5. [Note: The image used in this exercise is located in the examples folder for Chapter 2.]

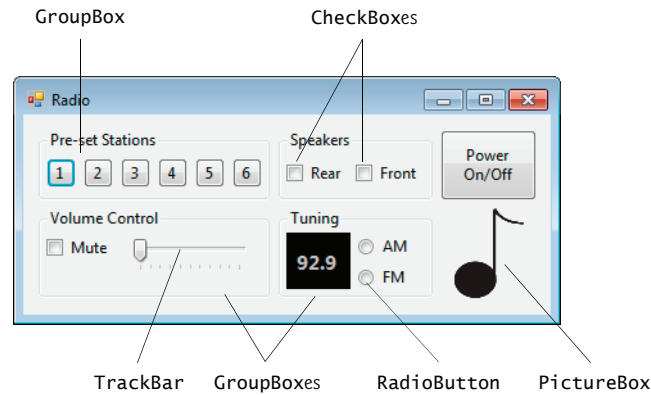


Fig. 2.5 | Radio GUI.

- Manipulating the Form's properties.** Change the Font property to 9pt Segoe UI. Change the Form's Text property to Radio and the Size to 427, 194.
- Adding the Pre-set Stations GroupBox and Buttons.** Set the GroupBox's Size to 180, 55 and its Text to Pre-set Stations. Add six Buttons to the GroupBox. Set each one's Size to 23, 23. Change the Buttons' Text properties to 1, 2, 3, 4, 5, 6, respectively.
- Adding the Speakers GroupBox and CheckBoxes.** Set the GroupBox's Size to 120, 55 and its Text to Speakers. Add two CheckBoxes to the GroupBox. Set the Text properties for the CheckBoxes to Rear and Front.
- Adding the Power On/Off Button.** Add a Button to the Form. Set its Text to Power On/Off and its Size to 75, 55.
- Adding the Volume Control GroupBox, the Mute CheckBox and the Volume TrackBar.** Add a GroupBox to the Form. Set its Text to Volume Control and its Size to 180, 70. Add a CheckBox to the GroupBox. Set its Text to Mute. Add a TrackBar to the GroupBox.
- Adding the Tuning GroupBox, the radio station Label and the AM/FM RadioButtons.** Add a GroupBox to the Form. Set its Text to Tuning and its Size to 120, 70. Add a Label to the GroupBox. Set its AutoSize to False, its Size to 50, 44, its BackColor to Black, its ForeColor to Silver, its font to 12pt bold and its TextAlign to MiddleCenter. Set its Text to 92.9. Place the Label as shown in the figure. Add two RadioButtons to the GroupBox. Set the Text of one to AM and of the other to FM.
- Adding the image.** Add a PictureBox to the Form. Set its BackColor to PeachPuff, its SizeMode to StretchImage and its Size to 55, 70. Set the Image property to Music-Note.gif (located in the examples folder for Chapter 2).