Introduction to the Visual Basic Express 2008 IDE
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 will 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.
OBJECTIVES

- 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.
2.1 Introduction
2.2 Overview of the Visual Studio 2008 IDE
2.3 Menu Bar and Toolbar
2.4 Navigating the Visual Studio IDE
2.5 Using Help
2.6 Using Visual Programming to Create a Simple Program that Displays Text and an Image
2.1 Introduction

• Visual Studio 2008 is Microsoft’s Integrated Development Environment (IDE) for creating, running and debugging programs.

• A simple Visual Basic program can be created by dragging and dropping predefined blocks into place through visual programming.
2.2 Overview of the Visual Studio 2008 IDE

- Start Microsoft Visual Basic 2008 Express Edition. (Fig. 2.1)
2.2 Overview of the Visual Studio 2008 IDE (Cont.)

- The **Recent Projects** section shows solutions you have been working on.

- **Getting Started** focuses on using the IDE for creating programs, learning Visual Basic, connecting to the developer community and providing development tools.

- **Visual Basic Express Headlines** and **MSDN: Visual Basic Express Edition** link to information about Visual Basic.
2.2 Overview of the Visual Studio 2008 IDE (Cont.)

- The IDE also has an internal web browser.
- To request a web page, enter its URL into the location bar (Fig. 2.2).
2.2 Overview of the Visual Studio 2008 IDE (Cont.)

- Select **File > New Project...** to create a new project (Fig. 2.3).
- Project **templates** are the project types users can create in Visual Basic.
  - A **Windows Forms application** has a **graphical user interface (GUI)**.
2.2 Overview of the Visual Studio 2008 IDE (Cont.)

Windows Forms Application (selected)

**Fig. 2.3** | **New Project** dialog.
2.2 Overview of the Visual Studio 2008 IDE (Cont.)

- Click **OK** to display the IDE in **Design view** (Fig. 2.4).

**Fig. 2.4 | Design view of the IDE.**
2.2 Overview of the Visual Studio 2008 IDE (Cont.)

- As you place controls on the Form, you’ll be able to modify their properties.
- Figure 2.5 shows where the Form’s title text can be modified.

Fig. 2.5 | Text box control for modifying a property in the Visual Studio IDE.
2.2 Overview of the Visual Studio 2008 IDE (Cont.)

• Figure 2.6 shows a dialog in which a control’s font properties can be modified.

Fig. 2.6 | Dialog for modifying a control’s font properties in the Visual Studio IDE.
2.3 Menu Bar and Toolbar

- Many commands are contained in menus (Fig. 2.7).
- The set of menus displayed depends on what you are currently doing in the IDE.

![Visual Studio menu bar.](image)

**Fig. 2.7** | Visual Studio menu bar.
### 2.3 Menu Bar and Toolbar (Cont.)

<table>
<thead>
<tr>
<th>Menu</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>File</strong></td>
<td>Commands for opening, closing, adding and saving projects, as well as printing project data and exiting Visual Studio.</td>
</tr>
<tr>
<td><strong>Edit</strong></td>
<td>Commands for editing programs, such as cut, copy, paste, undo, redo, delete, find and select.</td>
</tr>
<tr>
<td><strong>View</strong></td>
<td>Commands for displaying IDE windows and for adding toolbars.</td>
</tr>
<tr>
<td><strong>Project</strong></td>
<td>Commands for managing projects and their files.</td>
</tr>
<tr>
<td><strong>Build</strong></td>
<td>Commands for compiling Visual Basic programs.</td>
</tr>
<tr>
<td><strong>Debug</strong></td>
<td>Commands for debugging and running programs.</td>
</tr>
<tr>
<td><strong>Data</strong></td>
<td>Commands for interacting with databases.</td>
</tr>
<tr>
<td><strong>Format</strong></td>
<td>Commands for arranging and modifying a Form's controls.</td>
</tr>
<tr>
<td><strong>Tools</strong></td>
<td>Commands for accessing additional IDE tools and options.</td>
</tr>
<tr>
<td><strong>Window</strong></td>
<td>Commands for hiding, opening, closing and displaying IDE windows.</td>
</tr>
<tr>
<td><strong>Help</strong></td>
<td>Commands for accessing the IDE’s help features.</td>
</tr>
</tbody>
</table>

**Fig. 2.8** | Summary of Visual Studio 2008 IDE menus.
2.3 Menu Bar and Toolbar (Cont.)

- You can access common commands from the toolbar icons (Fig. 2.9).

![Standard Visual Studio toolbar.](image)

Fig. 2.9 | Standard Visual Studio toolbar.

© 2009 Pearson Education, Inc. All rights reserved.
2.3 Menu Bar and Toolbar (Cont.)

- Select **View > Toolbars** (Fig. 2.10).
- Each toolbar you select is displayed at the top of the Visual Studio window.

![Image of Visual Studio window with Toolbars]

**Fig. 2.10** | Adding the **Build** toolbar to the IDE.
2.3 Menu Bar and Toolbar (Cont.)

- Some icons contain a down arrow that you can click to display related or commands (Fig. 2.11).

Fig. 2.11 | IDE toolbar icon showing additional commands.
2.3 Menu Bar and Toolbar (Cont.)

• Positioning the mouse pointer over an icon highlights it and, after a brief pause, displays a description called a tool tip (Fig. 2.12).

Fig. 2.12 | Tool tip demonstration.
2.4 Navigating the Visual Studio IDE

• The IDE provides windows for accessing project files and customizing controls (Fig. 2.13).

![Diagram showing Toolbar icons for four Visual Studio windows.

Fig. 2.13 | Toolbar icons for four Visual Studio windows.
2.4 Navigating the Visual Studio IDE (Cont.)

- Visual Studio provides a space-saving feature called **auto-hide**.
  - When auto-hide is enabled, a tab appears along the edge of the IDE window (Fig. 2.14).

![Icon for hidden window (auto-hide enabled)](image)
2.4 Navigating the Visual Studio IDE (Cont.)

- Placing the mouse pointer over one of these icons displays that window (Fig. 2.15).

Fig. 2.15 | Displaying a hidden window when auto-hide is enabled.
2.4 Navigating the Visual Studio IDE (Cont.)

- To “pin down” a window, click its pin icon.
- When auto-hide is enabled, the pin icon is horizontal.
- When a window is “pinned down,” the pin is vertical (Fig. 2.16).

Fig. 2.16 | Disabling auto-hide (“pinning down” a window).
2.4 Navigating the Visual Studio IDE (Cont.)

2.4.1 Solution Explorer

• The Solution Explorer (Fig. 2.17) provides access to all of the solution’s files.

• The solution’s startup project is the one that runs when you select Debug > Start Debugging.

![Solution Explorer with an open project.](image)

Fig. 2.17 | Solution Explorer with an open project.
2.4 Navigating the Visual Studio IDE (Cont.)

- Clicking the **Solution Explorer’s Show All Files icon** displays all the files in the solution (Fig. 2.18).

![Solution Explorer showing plus boxes and minus boxes for expanding and collapsing the tree to reveal or hide project files.](image-url)

**Fig. 2.18** | **Solution Explorer** showing plus boxes and minus boxes for expanding and collapsing the tree to reveal or hide project files.
2.4 Navigating the Visual Studio IDE (Cont.)

- The plus and minus boxes can be clicked to expand and collapse the project tree.
  - Click the plus box to the left of **My Project** (Fig. 2.19).

![Solution Explorer](image)

*Fig. 2.19 | Solution Explorer expanding the My Project file after clicking its plus box.*
2.4 Navigating the Visual Studio IDE (Cont.)

– Click the minus box to collapse the tree from its expanded state (Fig. 2.20).

![Solution Explorer](image)

**Fig. 2.20** | *Solution Explorer* showing collapsed nodes after all minus boxes are clicked.
2.4 Navigating the Visual Studio IDE (Cont.)

2.4.2 Toolbox

Fig. 2.21 | Toolbox window displaying controls for the Common Controls group.
2.4 Navigating the Visual Studio IDE (Cont.)

2.4.3 Properties Window

• Select **View > Properties Window** (Fig. 2.22).
  
  – The **Properties** window allows you to modify a control’s properties visually, without writing code.
  
  – At the top of the **Properties** window is the **component selection drop-down list**.
2.4 Navigating the Visual Studio IDE (Cont.)

Fig. 2.22 | Properties window.
2.5 Using Help

• Some **Help menu** commands are summarized in Fig. 2.23.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>How Do I?</strong></td>
<td>Contains links to relevant topics, including how to upgrade programs and learn more about Web services, architecture and design, files and I/O, data, debugging and more.</td>
</tr>
<tr>
<td><strong>Search</strong></td>
<td>Finds help articles based on search keywords.</td>
</tr>
<tr>
<td><strong>Index</strong></td>
<td>Displays an alphabetized list of topics you can browse.</td>
</tr>
<tr>
<td><strong>Contents</strong></td>
<td>Displays a categorized table of contents in which help articles are organized by topic.</td>
</tr>
</tbody>
</table>

**Fig. 2.23** | **Help menu commands.**
2.5 Using Help (Cont.)

- To use context-sensitive help, click an item, then press the \textit{F1} key (Fig. 2.24).

\textbf{Fig. 2.24} | Using context-sensitive help.

© 2009 Pearson Education, Inc. All rights reserved.
2.5 Using Help (Cont.)

- The **Help** options can be set by selecting **Tools > Options**.
- Make sure that **Show all settings** is checked (Fig. 2.25).
- Select **Help**, then set the **Show Help using:** drop-down list to **External Help**.

![Options dialog displaying Help settings.](image)

*Fig. 2.25 | Options* dialog displaying **Help** settings.
• Figure 2.26 shows the result of a program as it executes.

**Fig. 2.26** | Simple program executing.
2.6 Using Visual Programming to Create a Simple Program… (Cont.)

- Select **File > New Project** (Fig. 2.27).
- From the template options, select **Windows Forms Application**.
  - Name the project **ASimpleProgram** and click **OK**.

![Fig. 2.27 | New Project dialog.](image-url)
2.6 Using Visual Programming to Create a Simple Program... (Cont.)

- Select **File > Save All** to display the **Save Project** dialog (Fig. 2.28).

Fig. 2.28 | **Save Project** dialog.
2.6 Using Visual Programming to Create a Simple Program… (Cont.)

– To set the project location, click the **Browse**… button (Fig. 2.29).

![Fig. 2.29 | Setting the project location in the Project Location dialog.](image)

© 2009 Pearson Education, Inc. All rights reserved.
2.6 Using Visual Programming to Create a Simple Program... (Cont.)

- When you first begin working in the IDE, it is in Design Mode.
  - The text in the Form’s title bar is determined by the Form’s Text property (Fig. 2.30).

![Fig. 2.30](image-url) Setting the Form’s Text property in the Properties window.

© 2009 Pearson Education, Inc. All rights reserved.
2.6 Using Visual Programming to Create a Simple Program... (Cont.)

- Press the *Enter* key when finished (Fig. 2.31).
- Click and drag sizing handles to resize the *Form*.

![A Simple Program](image)

**Fig. 2.31** | *Form* with enabled sizing handles.
– Click and drag one of the Form’s enabled sizing handles to make the Form larger (Fig. 2.32).

![Resized Form](image)
2.6 Using Visual Programming to Create a Simple Program... (Cont.)

- Clicking **BackColor** in the **Properties** window causes a down-arrow button to appear (Fig. 2.33).
- The arrow displays tabs for **Custom**, **Web** and **System**. Click the **Custom tab** to display the **palette**.

![Diagram of Properties Window](image)

**Fig. 2.33** | Changing the Form's **BackColor** property.
Once you select a color, the palette closes and the Form’s background color changes (Fig. 2.34).
2.6 Using Visual Programming to Create a Simple Program... (Cont.)

- Select **View > Toolbox** to display the set of controls.
- Double click the **Label** control in the **Toolbox** (Fig. 2.35).

*Fig. 2.35* | Adding a **Label** to the **Form**
2.6 Using Visual Programming to Create a Simple Program... (Cont.)

- Set the Label’s Text property to **Welcome to Visual Basic!**.
- Set the **AutoSize** property to **False** (Fig. 2.36) so that you can resize the Label on your own.

![AutoSize property](image)

*Fig. 2.36 | Changing the Label’s AutoSize property to False.*
2.6 Using Visual Programming to Create a Simple Program… (Cont.)

- Resize the Label so that the text fits.
- Center the Label control horizontally by selecting **Format > Center In Form > Horizontally** (Fig. 2.37).

![Diagram of GUI with resizing handles and centered label]

**Fig. 2.37** | GUI after the Form and Label have been customized.

© 2009 Pearson Education, Inc. All rights reserved.
2.6 Using Visual Programming to Create a Simple Program that Displays Text and an Image (Cont.)

• Select the **Font property**, which causes an **ellipsis button** to appear next to the value (Fig. 2.38).

![Properties window displaying the Label's properties.](image-url)

**Fig. 2.38 | Properties** window displaying the Label’s properties.

© 2009 Pearson Education, Inc. All rights reserved.
2.6 Using Visual Programming to Create a Simple Program... (Cont.)

• When the ellipsis button is clicked, the Font dialog (Fig. 2.39) is displayed.
  – Under Font, select Segoe UI.
  – Under Size, select 24 points and click OK.

Fig. 2.39 | Font dialog for selecting fonts, styles and sizes.
2.6 Using Visual Programming to Create a Simple Program... (Cont.)

• Select the **TextAlign** property (Fig. 2.40).
  
  – Set the TextAlign property to **MiddleCenter** to have the text to appear centered in the Label.

*Fig. 2.40* | Centering the Label's text.
2.6 Using Visual Programming to Create a Simple Program... (Cont.)

- The **PictureBox** control displays images.
- Locate the **PictureBox** in the **Toolbox** and double click it to add it to the **Form** (Fig. 2.41).

![Fig. 2.41 | Inserting and aligning a PictureBox.](image)
2.6 Using Visual Programming to Create a Simple Program… (Cont.)

• Locate the **Image property** (Fig. 2.42).

• No picture has been assigned, so the value of the Image property displays **(none)**.

---

**Fig. 2.42** | **Image property of the PictureBox.**

© 2009 Pearson Education, Inc. All rights reserved.
2.6 Using Visual Programming to Create a Simple Program... (Cont.)

• Click the ellipsis button to display the **Select Resource** dialog (Fig. 2.43).

![Select Resource dialog](image)

**Fig. 2.43** | Select Resource dialog to select an image for the PictureBox.
2.6 Using Visual Programming to Create a Simple Program... (Cont.)

• Click the **Import**... button.
• Locate `bug.png`, select it and click **OK**
• The resource named `bug` represents `bug.png` (Fig. 2.44).

![Select Resource dialog](image)

**Fig. 2.44** | **Select Resource** dialog displaying a preview of selected image.
2.6 Using Visual Programming to Create a Simple Program… (Cont.)

- Click **OK** to place the image in your program. Supported image formats include PNG, GIF, JPEG and BMP.
- To size the image to the PictureBox, change the **SizeMode property** to **StretchImage** (Fig. 2.45).

![SizeMode property](image)

**Fig. 2.45** | Scaling an image to the size of the **PictureBox**.

© 2009 Pearson Education, Inc. All rights reserved.
2.6 Using Visual Programming to Create a Simple Program… (Cont.)

• Resize the PictureBox, making it larger (Fig. 2.46).
• Select **File > Save All** to save the entire solution.

![PictureBox](image)

**Fig. 2.46** | PictureBox displaying an image.
2.6 Using Visual Programming to Create a Simple Program… (Cont.)

• Select **Debug > Start Debugging** to execute the program (Fig. 2.47).

Fig. 2.47 | Debugging a solution.

© 2009 Pearson Education, Inc. All rights reserved.
2.6 Using Visual Programming to Create a Simple Program... (Cont.)

• In run mode, the program is executing, and you can interact with only a few of the IDE features (Fig. 2.48).

Fig. 2.48 | IDE in run mode, with the running program in the foreground.