Chapter 1

Visual Studio

Advanced Programming Using Visual Basic 2005

Objectives

- Distinguish the features of the Visual Studio IDE and the .NET Framework
- Identify and understand the purpose of each of the files listed in the Solution Explorer
- Understand what happens at compile time
- Set and display data from the application’s assembly attributes
- Create an MDI project with a parent form, child forms, a toolbar, status bar, context menus and ToolTips
- Use the singleton design pattern to create a class that should be instantiated only once
Visual Basic 2005

- Major update to the development environment and the VB language
  - More powerful and easier to use
  - Expanded IntelliSense, smart tags, wizards and Project Designer
  - Resources stored in a Resources folder — can hold graphic and audio files, text strings, and foreign language conversions
  - Form Designer uses new snap lines and spacing guides
  - Partial classes, code snippets, Click Once deployment all simplify creating and deploying .NET applications

The .NET Framework

- A platform for developing and running applications and XML Web services
- Framework components
  - Common language runtime
  - Class libraries
  - ASP.NET

Common Language Runtime (CLR)

- Manages execution of code
  - Integrates components developed in different languages
  - Handles errors across languages
  - Provides security
  - Manages storage and destruction of objects
Managed code and Metadata

- Code compiled to run in the CLR
- Data that describe data
  - data types
  - members
  - references
- Portable executable (PE) file

Managed data
- CLR manages data storage
- CLR garbage collector removes objects no longer in use

Class Library
- Stores all classes and interfaces of the .NET language
- Stored in a library known as the .NET Framework class library
- Namespaces – sections within the library that contain classes, structures, enumerations, delegates, interfaces

See Table 1.1 in text
Common Language Specification (CLS)

- Published standards that specify how a language that interacts with the CLR should behave
- Programs should be CLS-compliant

NET Framework Developer’s Guide
Contains the rules for CLS compliance

Class Library - Types

- Refer to classes, structures, enumerations, delegates, interfaces, data types
- Any element used in the As clause
  - Dim anyName As someType
- Can be Value type or Reference type
  - Value type — Actual value stored in variable
    - secondValueInteger = firstValueInteger
  - Two separate variables (two copies of value)
  - Reference type — Variable stores a reference (address)
    - secondForm = firstForm
  - Both variables reference the same location

Using .NET CLR to Compile Programs

The steps in compiling and executing a program
Using the .NET CLR
Assemblies
- Basic units of code that may be a single PE file or multiple files
- The **assembly manifest**
  - Metadata about the version
  - Table describing files needed
  - Assembly reference list of external files needed
- Only the needed assemblies are loaded into memory at execution time
- Assemblies are reference objects

Attributes
- Tags containing information about parts of a program
- Part of the metadata in a .NET assembly
- Reflection – examining the metadata in an assembly’s attributes

The Reference Collection
- Used to connect a VB project to external components
- Two types
  - Assemblies
    - Reference to another project; is called a project-to-project reference
  - COM objects
    - Components written in previous versions of VB or other non-CLS-compliant languages
ASP.NET

- ASP.NET 2.0 is the newest version of Active Server Pages (ASP)
- Web development environment that compiles applications written in .NET compatible languages
- Uses CLR and managed code features
- Makes Web development easier by providing development and debugging support for Web applications and Web Services similar to that for Windows applications

Temporary Projects

- New feature for VS 2005
  - By default, projects are created as a temporary project, stored in a temporary folder on the hard drive
  - When you compile, the compiler saves the files in a temporary folder
  - You can test a project without saving it
  - No clutter on the hard drive; but can lose a project easier
  - Can choose to save on exiting or select to always save new projects: Choose Tools/Options/Projects and Solutions and select Save new projects when created

Setting Environment Options

- Options dialog box has new features — Select Tools/Options and select the check box for Show all settings
Selecting Development Settings

- When installing VS 2005, select the option for a VB programmer, which creates the default settings profile.
- If sharing a computer, the defaults may need to be reset.
  - Choose Tools/Import and Export Settings.
- Current settings can be saved or can be overwritten.

Alignment Guides for Windows Forms

- Windows form designer includes snap lines used to align controls.
  - Blue snap lines appear when edges of controls are aligned.
  - Red snap lines appear when text baselines of controls are aligned.
- Visible grid does not appear by default.
  - Can choose to display the grid in the Options dialog box, which turns off the snap lines.

The Solution Explorer

- Click the Show All Files button to view the hidden files and folders.
- New My Project folder holds several files for project configuration, including AssemblyInfo.vb.
- The solution name does not display by default.
  - Select Tools/Options/Projects and Solutions and check Always show solution.
After compiling a project and clicking the Refresh button, the bin\Debug folder holds the files needed to run the program in the debugger.

The Resources folder holds the project resources, such as graphic and sound files and text strings.
VS 2005 New Image Files

- Microsoft is including many new graphic files with VS 2005
- File: VS2005ImageLibrary.zip holds standard Windows graphics that you can use in your applications

Form Designer Generated Code

- Partial classes (types) allow a class to be split into two or more separate files
- The VB form designer splits the form’s class into two parts, separating the designer-generated code from the developer-written code

The Project Designer

- An expansion of the Project Properties dialog box
- To open, double-click on the My Project folder or select Project/ProjectName Properties
Deploying Windows Applications

- Target computer must have the .NET Framework or the .NET Framework Redistributable installed (free download)
- Choose from several methods for deploying Windows applications
  - XCopy deployment
  - Windows Installer Technology
  - Click-Once deployment
  - Third-party installer products

The My Object

- New feature in VB 2005
- Provides easy access to
  - Information about applications
  - Resources
  - The computer on which the application is executing
  - The current user

Assembly Information

- The Assembly Information dialog box with values supplied by default
- Entries can be edited to chosen values
Assembly Information Example

- Assembly information displays in labels in the program's output.

Displaying Attributes

- Display the Version tab to view the attributes.
- Point to the filename and pause, the attributes pop up automatically.

Menus, Toolbars, and Status Bars

- New components for menus, context menus, toolbars and status bars:
  - MenuStrip, ContextMenuStrip, ToolStrip, and StatusStrip
- Old components can still be used
- New components have more capabilities
- Includes designers which simplify creating menus, toolbars and status bars:
  - Smarter than the old ones
  - Give each element a meaningful name
  - Simplify setting properties for the controls
Creating Menus with MenuStrips

- When added to a form, the component appears in the component tray and the words “Type Here” appear.
- Drop down the list for a new menu item and make a selection from the list.

Items Collection Editor

- Use to add and remove menu items, reorder items and set properties.

Creating Toolbars with ToolStrips

- Typically hold shortcut buttons to menu items.
- Similar to MenuStrips.
Creating Status Bars with StatusStrips

- Generally appear at the bottom of a form
- Supply information to the user
- Share many of the same characteristics as MenuStrips

Displaying the Date and Time

- Use the properties and methods of the DateTime structure to retrieve and format the current date and time
- The **Now property** holds the system date and time in a numeric format that can be used for calculations
- Methods are used to format the date and time format
  - Examples
    - `Today.ToShortDateString`
    - `Now.ToShortTimeString`

MDI Applications

- Main form (parent) and sub-forms (children)
- Child forms must stay inside the boundaries of the parent form
- If a parent form closes all the children “leave” too
Adding a Window Menu

- Parent form should have a Window menu
  - Lists the open child windows and allows the user to switch between and arrange windows

Singleton Design Pattern

- Only allow a single instance of a form
- OOP technique: Use singleton design pattern
- To create a singleton pattern
  - Define a property that returns an instance of the class
  - Declare a shared private variable to hold an instance of the class
- To instantiate the single instance
  ```vbs
  Dim aChildOneForm As ChildOneForm = ChildOneForm.Instance()
  With aChildOneForm
    .MdiParent = Me
    .Show()
    .Focus()
  End With
  ```

Splash Screen Forms and Templates

- The initial screen normally displays while a program is loading
- Adds a professional touch
- Should display for as long as it takes for the rest of the application to load
- Code can be written to make the splash screen appear longer using threads
  - Allow multiple processes to execute simultaneously
- The Splash Screen template can be used for a splash screen
Class Diagrams

- Helps to visualize the classes in projects

Select the project name and click on the View Class Diagram button in the Solution Explorer to create or view a class diagram.

Class Diagram Menu

- The Class Diagram menu appears when a class diagram displays in the main Document window.

Customizing a Class Diagram

- It may be easier to visualize the organization of an application without any extras; click a shape and press Delete
- Shapes can also be moved and resized
- Clarify a class diagram by hiding some members