Chapter 1
Object-Oriented System Development with VB .NET

Objectives
In this chapter, you will:
• Learn about OO development and VB .NET
• Understand object-oriented concepts
• Recognize the benefits of OO development
• Preview the approach this book uses to teach you OO development

Introduction
• Object-oriented information system development involves analysis, design, and implementation of information systems using:
  – Object-oriented programming languages
  – Object-oriented technologies
  – Object-oriented techniques
• Object-oriented information system development is usually referred to as “OO” or as “the OO approach”

Understanding OO Development and VB .NET
• The object-oriented approach defines a system as a collection of objects that work together to accomplish tasks
  – The objects can carry out actions when asked
  – Each object maintains its own data

• When developing business systems, OO means using an:
  – Object-oriented approach to system analysis (OOA)
  – Object-oriented approach to system design (OOD)
  – Object-oriented approach to programming (OOP)

• The procedural approach defines a system as a set of procedures that interact with data
  – The data are maintained in files separate from the procedures
  – When the procedure executes, data files are created or updated
Object-Oriented Programming

- Object-oriented programming started in the 1960s with the development of the Simula programming language
- A major milestone in the history of OO was the development of the SmallTalk programming language in the early 1970s
- Additional object-oriented programming languages include Objective-C, Eiffel, and most notably C++

Object-Oriented Programming

- In 1995, Sun Microsystems introduced Java as a pure OO language, which has:
  - Syntax similar to C++
  - Features that make it appropriate for Internet applications
- Microsoft immediately released a version of Java called J++

Object-Oriented Programming

- Microsoft recently released a more direct competitor to Java as part of the .NET framework, named C#
- With the release of VB .NET, Microsoft hopes to dominate OO and Web-based development

The Microsoft .NET Framework and VB .NET

- The .NET framework has two main components:
  - .NET common language runtime
  - .NET framework class library
- The .NET Common Language Runtime (CLR) manages code at execution
- Developers can use a variety of programming languages that the CLR environment can compile and execute

The Microsoft .NET Framework and VB .NET

- The .NET framework class library provides reusable classes of objects that work with the CLR
- A programmer using any of the .NET programming languages can use these classes
- VB .NET is a full-blown OO programming language that shares the common language runtime and .NET framework class library with the other .NET languages
Object-Oriented Analysis and Design

- The standard object-oriented analysis and design modeling notation is the Unified Modeling Language (UML)
- UML assumes a model-driven approach to analysis and design

Object-Oriented Analysis and Design

- The system development life cycle (SDLC) is a project management framework
- It defines project phases and activities within phases
- The phases typically are named:
  - Planning
  - Analysis
  - Design
  - Implementation
  - Support

Object-Oriented Analysis and Design

- OO developers usually follow an iterative approach to analysis, design, and implementation
- Prototyping and joint application development (JAD) are usually part of OO development
  - Prototyping: creating a working model of one or more parts of a system to give users a chance to see and evaluate something concrete
  - During JAD sessions, key system stakeholders and decision makers work together to define system requirements and designs

Object-Oriented Analysis and Design

- The following are also required when using OO development, as they are in traditional system development:
  - Project management
  - Interviewing and data collection
  - User interface design
  - Testing
  - Conversion techniques

Understanding Object-Oriented Concepts

- Object-oriented development assumes that a system is a collection of objects that interact to accomplish tasks
Objects, Attributes, and Methods

- An object is a thing that has attributes and behaviors
- A GUI object uses graphics, such as a button or label, to represent part of a system
- A GUI object has attributes, which are characteristics that have values
- GUI objects also have behaviors or methods, which describe what an object can do

Objects, Attributes, and Methods

- OO systems also contain problem domain objects, which are specific to a business application
- For example: a business system that processes orders includes:
  - Customer objects
  - Order objects
  - Product objects
- Problem domain objects also have attributes and methods

Object Interactions and Messages

- Objects interact by sending messages to each other, asking another object to invoke, or carry out, one of its methods
- Objects interacting by sending messages to carry out tasks is the main concept of OOA and OOD

Encapsulation and Information Hiding

- Encapsulation: an object has attributes and methods combined into one unit
- By combining attributes and methods, the programmer does not need to know the internal structure of the object to send messages to it
- Information hiding: Using encapsulation to hide the internal structure of objects, protecting them from corruption

Encapsulation and Information Hiding

- Each object also has a unique identity
- An object’s identity must be known for sending a message to it
- The object’s identity is usually stored as a memory address
- Persistent objects are those that are defined as available for use over time

Classes, Instances, and Associations

- The class defines what all objects of the class represent
- Objects can be referred to as instances of the class
- When an object is created for the class, it is common to say the class is instantiated
- The terms “instance” and “object” are often used interchangeably
Objects maintain association relationships among themselves.
Some association relationships are one-to-one, and some associations are one-to-many.
UML refers to the number of associations as the multiplicity of the association.

Inheritance, one class of objects takes on characteristics of another class and extends them.
For example:
- An object belonging to the Customer class might also be something more general, such as a person.
- If the Person class is already defined, the Customer class can be defined by extending the Person class to take on more specific attributes and methods required of a customer.

In the previous example:
- The Person class is a superclass.
- The Customer class is the subclass.

The result of extending general classes into more specific subclasses is referred to as a generalization/specialization hierarchy.
It is also called an inheritance hierarchy.
Polymorphism: the way different objects can respond in their own way to the same message.
Classes are polymorphic if their instances can respond to the same message.
Recognizing the Benefits of OO Development

- The two main reasons why the object-oriented approach is being used in information system development are:
  - Naturalness
  - Reuse

Objects Are More Natural

- Naturalness: people usually think about their world in terms of objects
- When people discuss system requirements, it is natural to define the classes of objects involved
- OOA, OOD, and OOP all involve modeling classes of objects – so the focus remains on objects throughout the development process

Classes of Objects Can Be Reused

- The ability to reuse classes and objects is an important benefit of object-oriented development
- Classes and objects can be invented once and used many times
- Object-oriented programming languages come with class libraries that contain predefined classes most programmers need
  - Programmers use these classes to create their own objects

Learning OO Development

- This book provides a comprehensive guide to OO system development, including OOA, OOD, and OOP

Introducing Three-Tier Design

- The book is organized according to an approach to OO development called three-tier design
- Three-tier design requires that the collection of objects that interact in an OO system be separated into three categories of classes:
  - Problem domain classes
  - GUI classes
  - Data access classes

Introducing Three-Tier Design

- Problem domain classes are the classes of objects specific to the business application
- GUI classes define the objects that make up the user interface to the application
- Data access classes work with the database management system to store information about objects for later use
- The core of this book is organized according to the three tiers of OO development
Part 1: Object-Orientation and VB .NET Fundamentals

- Part 1 covers OO concepts and introduces the VB .NET programming language
- This part includes chapters 1, 2, 3, 4 and 5

Part 2: Developing Problem Domain Classes

- Part 2 shows how to use VB .NET to create new problem domain classes that are specific to the business system being developed
- Part 2 includes chapters 6, 7, 8 and 9

Part 3: Developing GUI Classes

- Part 3 describes how to create graphical user interface classes with which the user can interact
- The GUI classes in turn interact with problem domain classes
- Part 3 includes chapters 10, 11 and 12

Part 4: Developing Data Access Classes

- Part 4 covers the third tier—data access classes
- Data access classes are used to manage database interactions and achieve object persistence
- Part 4 includes chapters 13 and 14

Part 5: Deploying the Three-Tier Application

- Part 5 shows how GUI classes, problem domain classes, and data access classes function together as three tiers to create a complete client-server system
- Part 5 includes chapters 15 and 16

Summary

- Object-oriented information system development includes object-oriented analysis (OOA), object-oriented design (OOD), and object-oriented programming (OOP)
- Object-oriented (OO) systems are viewed as collections of interacting objects that accomplish tasks
- The Microsoft .NET framework is a computing platform that simplifies development of OO applications
Summary

• A model-driven approach using UML diagrams defines requirements and designs prior to programming
• The benefits of OO development include naturalness and reuse
• This text is organized into five parts and explains the three-tier design approach to OO development