

Object-Oriented Programming in Visual Basic .NET

Table of Contents

INTRODUCTION.....	INTRO-1
Prerequisites.....	INTRO-2
Installing the Practice Files	INTRO-3
Software Requirements	INTRO-3
Installation.....	INTRO-3
About the Author	INTRO-4
OVERVIEW OF OBJECTS.....	1-1
The Terminology of Objects.....	1-2
What Is an Object?	1-2
Classes	1-3
Class Interface	1-4
Properties.....	1-5
Methods	1-6
Messages.....	1-7
The Four Principles of Objects	1-8
Encapsulation	1-8
Inheritance	1-9
Polymorphism	1-10
Abstraction	1-11
Why Use Objects?	1-12
Reusability	1-12
Loosely Coupled Architecture	1-12
Implementation Safety	1-13
Black Box Development.....	1-13
Extensibility	1-14
Prototyping.....	1-14
Reinforcing the Basics	1-16
An Object-Oriented Starting Point.....	1-16
OBJECT-ORIENTED DESIGN: USE CASES	2-1
Object-Oriented Design	2-2
Top-Down Design and Functional Decomposition.....	2-2
The Object-Oriented Approach	2-4
Design Methodologies	2-7
Use Case Analysis.....	2-9

Use Case Analysis Example: Placing a Sales Order	2-10
The Benefits of Use Cases	2-17
The Risks of Use Cases	2-19
Guidelines for Creating a Use Case Diagram.....	2-23
Use Case Guidelines	2-23
Actor Guidelines.....	2-24
Association Guidelines.....	2-26
LAB 2: OBJECT-ORIENTED DESIGN: USE CASES	2-31
Lab 2 Overview.....	2-32
Write the Use Case Descriptions.....	2-33
Create the Use Case Diagram.....	2-37
OBJECT-ORIENTED DESIGN: THE CLASS MODEL	3-1
Elements of the Class Model	3-2
Class Element.....	3-2
Relationships	3-3
Common Class Design Mistakes.....	3-7
Design is Not Implementation!	3-7
Classes Are Not Entities	3-7
Class Models Are Not Everything	3-8
Creating the Class Model	3-9
Step 1: Identify the Classes	3-9
Step 2: Identify the Attributes.....	3-10
Step 3: Model the High-Level Interactions	3-11
The Object Sequence Diagram	3-14
Symbols and Notation.....	3-15
Creating an Object Sequence Diagram	3-17
Guidelines for Creating an OSD	3-23
LAB 3: OBJECT-ORIENTED DESIGN: THE CLASS MODEL	3-31
Lab 3 Overview.....	3-32
Create a Class Model	3-33
Create an Object Sequence Diagram for the Cash Withdrawal Use Case	3-37
OBJECT-ORIENTED DESIGN: REFINING THE DESIGN.....	4-1
Evaluating an Object-Oriented Design	4-2
Design Patterns	4-3
Definition	4-3

Design Pattern Example	4-3
Applying Design Patterns.....	4-5
Categorizing Design Patterns	4-6
Using Design Patterns in the Real World.....	4-11
Improving an Object-Oriented Design	4-14
What Is a “Good” Design?	4-14
IMPLEMENTING CLASSES IN VISUAL BASIC .NET	5-1
Creating a Class	5-2
Defining the Class.....	5-2
Defining Properties	5-4
Defining Methods	5-8
Shared Methods.....	5-10
Advanced Methods	5-13
Optional Parameters	5-13
Overloading Methods.....	5-14
Constructors.....	5-15
LAB 5: IMPLEMENTING CLASSES IN VISUAL BASIC .NET	5-25
Lab 5 Overview.....	5-26
Defining the Properties	5-27
Defining the Methods.....	5-33
INHERITANCE AND CLASS HIERARCHIES	6-1
The Power of Inheritance.....	6-2
Implementing Simple Inheritance	6-3
Abstract Classes.....	6-6
Defining an Abstract Class.....	6-8
Advanced Inheritance Concepts.....	6-12
Sealed Classes	6-12
Shadow Methods	6-13
Protected Methods.....	6-14
Interfaces.....	6-16
Defining an Interface.....	6-17
Implementing an Interface	6-17
LAB 6: INHERITANCE AND CLASS HIERARCHIES	6-22
Lab 6 Overview.....	6-23
Make the Report Class Virtual	6-24
Implement a Derived Class.....	6-26

IMPLEMENTING EVENTS AND DELEGATES.....	7-1
Event-Driven Applications.....	7-2
Delegates.....	7-4
Defining a Delegate	7-5
Using a Delegate	7-5
Events.....	7-8
Defining an Event.....	7-8
Raising the Event.....	7-9
Handling the Event.....	7-9
An Event-Raising Example	7-11
Advanced Event and Delegate Concepts.....	7-18
Shared Events	7-18
Multicast Delegates.....	7-19
LAB 7: IMPLEMENTING EVENTS AND DELEGATES.....	7-24
Lab 7 Overview.....	7-25
Create a Simple Chat Server.....	7-26
Monitor Long-Running Processes	7-34
HANDLING EXCEPTIONS.....	8-1
When Bad Things Happen to Good Programs	8-2
Error Handling.....	8-2
Exception Handling	8-3
Exception Handling.....	8-5
The Try Block.....	8-5
The Catch Block.....	8-6
The Finally Block.....	8-7
Putting Exceptions to Use.....	8-7
Exception Processing Guidelines	8-10
Exception Handling Performance	8-13
Custom Exceptions.....	8-17
System.Exception Class	8-17
Custom Exception Example	8-18
Unhandled Exceptions.....	8-20
LAB 8: HANDLING EXCEPTIONS	8-27
Lab 8 Overview.....	8-28
Build a Simple Try-Catch Block	8-29
Log Exceptions with a Custom Class	8-34

PUTTING OBJECTS TO USE IN THE BUSINESS LAYER	9-1
Physical Design	9-2
Research.....	9-3
Analysis and Rationalization	9-7
Implementation	9-8
Distributed Processing.....	9-10
What Is a Message Queue?	9-10
Using a Message Queue	9-13
Aspect-Oriented Programming	9-21
Call Interception	9-21
AOP Example: Method Logger	9-23
LAB 9: PUTTING OBJECTS TO USE IN THE BUSINESS LAYER	9-39
Lab 9 Overview.....	9-40
Role-Based Security for Methods	9-41
PUTTING OBJECTS TO USE IN WINDOWS APPLICATIONS.....	10-1
Visual Inheritance in Windows Forms.....	10-2
Creating the Base Form.....	10-2
Creating the Derived Form.....	10-5
Derived Controls	10-9
Creating an Inherited Control.....	10-9
LAB 10: PUTTING OBJECTS TO USE IN WINDOWS APPLICATIONS	10-18
Lab 10 Overview.....	10-19
Creating a Base Search Form	10-20
Create an Image Combo Box	10-29
PUTTING OBJECTS TO USE IN WEB APPLICATIONS.....	11-1
Visual Inheritance in Web Forms.....	11-2
The Goal of Visual Inheritance.....	11-3
ASP.NET Page Templates	11-4
Templates Explained	11-5
Server Controls.....	11-12
User Controls	11-12
Web Server Controls.....	11-15
LAB 11: PUTTING OBJECTS TO USE IN WEB APPLICATIONS	11-26
Lab 11 Overview.....	11-27
Creating a Dual List Box Control	11-28

Create an ASP.NET Page Template	11-39
INDEX.....	INDEX-1