

# Contents

<b>Acknowledgments</b>	<b>xi</b>
<b>Chapter 1: Programming with Visual C++ 2005</b>	<b>1</b>
<b>The .NET Framework</b>	<b>2</b>
<b>The Common Language Runtime (CLR)</b>	<b>2</b>
<b>Writing C++ Applications</b>	<b>3</b>
<b>Learning Windows Programming</b>	<b>4</b>
Learning C++	5
The C++ Standards	5
Console Applications	6
Windows Programming Concepts	6
<b>What Is the Integrated Development Environment?</b>	<b>8</b>
Components of the System	9
The Editor	9
The Compiler	9
The Linker	9
The Libraries	9
<b>Using the IDE</b>	<b>10</b>
Toolbar Options	10
Dockable Toolbars	12
Documentation	12
Projects and Solutions	13
Defining a Project	13
Debug and Release Versions of Your Program	19
Dealing with Errors	23
Setting Options in Visual C++ 2005	27
Creating and Executing Windows Applications	28
Creating an MFC Application	28
Building and Executing the MFC Application	30
Creating a Windows Forms Application	31
<b>Summary</b>	<b>35</b>

## **Chapter 2: Data, Variables, and Calculations** **37**

---

### **The Structure of a C++ Program** **38**

Program Comments 44

The #include Directive — Header Files 45

Namespaces and the Using Declaration 46

The main() Function 46

Program Statements 47

Whitespace 49

Statement Blocks 49

Automatically Generated Console Programs 50

### **Defining Variables** **51**

Naming Variables 51

Keywords in C++ 52

Declaring Variables 52

Initial Values for Variables 53

### **Fundamental Data Types** **54**

Integer Variables 54

Character Data Types 55

Integer Type Modifiers 56

The Boolean Type 57

Floating-Point Types 57

Fundamental Types in ISO/ANSI C++ 58

Literals 59

Defining Synonyms for Data Types 60

Variables with Specific Sets of Values 60

Specifying the Type for Enumeration Constants 62

### **Basic Input/Output Operations** **62**

Input from the Keyboard 62

Output to the Command Line 63

Formatting the Output 64

Escape Sequences 65

### **Calculating in C++** **67**

The Assignment Statement 67

Understanding Lvalues and Rvalues 68

Arithmetic Operations 68

The const Modifier 70

Constant Expressions 70

Program Input 71

Calculating the Result 71

Displaying the Result 73

Calculating a Remainder 73

---

Modifying a Variable	73
The Increment and Decrement Operators	74
The Sequence of Calculation	77
Operator Precedence	77
<b>Variable Types and Casting</b>	<b>78</b>
Rules for Casting Operands	78
Casts in Assignment Statements	79
Explicit Casts	80
Old-Style Casts	81
The Bitwise Operators	81
The Bitwise AND	82
The Bitwise OR	84
The Bitwise Exclusive OR	85
The Bitwise NOT	86
The Bitwise Shift Operators	86
<b>Understanding Storage Duration and Scope</b>	<b>88</b>
Automatic Variables	88
Positioning Variable Declarations	91
Global Variables	91
Static Variables	94
<b>Namespaces</b>	<b>95</b>
Declaring a Namespace	96
Multiple Namespaces	98
<b>C++/CLI Programming</b>	<b>99</b>
C++/CLI Specific: Fundamental Data Types	99
C++/CLI Output to the Command Line	104
C++/CLI Specific — Formatting the Output	104
C++/CLI Input from the Keyboard	107
Using <code>safe_cast</code>	108
C++/CLI Enumerations	109
Specifying a Type for Enumeration Constants	111
Specifying Values for Enumeration Constants	111
<b>Summary</b>	<b>112</b>
<b>Exercises</b>	<b>113</b>
<b>Chapter 3: Decisions and Loops</b>	<b>115</b>
<b>Comparing Values</b>	<b>115</b>
The if Statement	117
Nested if Statements	118
The Extended if Statement	120
Nested if-else Statements	122

# Contents

---

Logical Operators and Expressions	124
Logical AND	125
Logical OR	125
Logical NOT	126
The Conditional Operator	127
The switch Statement	129
Unconditional Branching	132
<b>Repeating a Block of Statements</b>	<b>132</b>
What Is a Loop?	132
Variations on the for Loop	135
Using the continue Statement	139
Floating-Point Loop Counters	143
The while Loop	143
The do-while Loop	146
Nested Loops	147
<b>C++/CLI Programming</b>	<b>150</b>
The for each Loop	153
<b>Summary</b>	<b>156</b>
<b>Exercises</b>	<b>157</b>
<b>Chapter 4: Arrays, Strings, and Pointers</b>	<b>159</b>
<b>Handling Multiple Data Values of the Same Type</b>	<b>160</b>
Arrays	160
Declaring Arrays	161
Initializing Arrays	164
Character Arrays and String Handling	166
String Input	167
Multidimensional Arrays	169
Initializing Multidimensional Arrays	170
<b>Indirect Data Access</b>	<b>172</b>
What Is a Pointer?	172
Declaring Pointers	173
The Address-Of Operator	173
Using Pointers	174
The Indirection Operator	174
Why Use Pointers?	174
Initializing Pointers	176
Pointers to char	177
The sizeof Operator	181
Constant Pointers and Pointers to Constants	183

---

Pointers and Arrays	185
Pointer Arithmetic	185
Using Pointers with Multidimensional Arrays	190
Pointer Notation with Multidimensional Arrays	191
<b>Dynamic Memory Allocation</b>	<b>192</b>
The Free Store, Alias the Heap	192
The new and delete Operators	193
Allocating Memory Dynamically for Arrays	194
Dynamic Allocation of Multidimensional Arrays	196
<b>Using References</b>	<b>197</b>
What Is a Reference?	197
Declaring and Initializing References	197
<b>C++/CLI Programming</b>	<b>198</b>
Tracking Handles	199
Declaring Tracking Handles	199
CLR Arrays	200
Sorting One-Dimensional Arrays	205
Searching One-Dimensional Arrays	206
Multidimensional Arrays	209
Arrays of Arrays	213
Strings	216
Joining Strings	217
Modifying Strings	220
Searching Strings	222
Tracking References	225
Interior Pointers	225
<b>Summary</b>	<b>228</b>
<b>Exercises</b>	<b>230</b>
<b>Chapter 5: Introducing Structure into Your Programs</b>	<b>231</b>
<b>Understanding Functions</b>	<b>232</b>
Why Do You Need Functions?	233
Structure of a Function	233
The Function Header	233
The Function Body	234
The return Statement	235
Using a Function	235
Function Prototypes	235
<b>Passing Arguments to a Function</b>	<b>239</b>
The Pass-by-value Mechanism	240
Pointers as Arguments to a Function	241

# Contents

---

Passing Arrays to a Function	243
Passing Multi-Dimensional Arrays to a Function	245
References as Arguments to a Function	247
Use of the const Modifier	249
Arguments to main()	250
Accepting a Variable Number of Function Arguments	252
<b>Returning Values from a Function</b>	<b>254</b>
Returning a Pointer	254
A Cast Iron Rule for Returning Addresses	256
Returning a Reference	258
A Teflon-Coated Rule: Returning References	260
Static Variables in a Function	260
<b>Recursive Function Calls</b>	<b>262</b>
Using Recursion	264
<b>C++/CLI Programming</b>	<b>265</b>
Functions Accepting a Variable Number of Arguments	266
Arguments to main()	267
<b>Summary</b>	<b>268</b>
<b>Exercises</b>	<b>269</b>
<b>Chapter 6: More about Program Structure</b>	<b>271</b>
<b>Pointers to Functions</b>	<b>271</b>
Declaring Pointers to Functions	272
A Pointer to a Function as an Argument	275
Arrays of Pointers to Functions	277
<b>Initializing Function Parameters</b>	<b>277</b>
<b>Exceptions</b>	<b>279</b>
Throwing Exceptions	281
Catching Exceptions	282
Exception Handling in the MFC	283
<b>Handling Memory Allocation Errors</b>	<b>284</b>
<b>Function Overloading</b>	<b>285</b>
What Is Function Overloading?	286
When to Overload Functions	288
<b>Function Templates</b>	<b>288</b>
Using a Function Template	289
<b>An Example Using Functions</b>	<b>291</b>
Implementing a Calculator	291
Analyzing the Problem	292
Eliminating Blanks from a String	294
Evaluating an Expression	295

---

Getting the Value of a Term	298
Analyzing a Number	299
Putting the Program Together	302
Extending the Program	304
Extracting a Substring	305
Running the Modified Program	307
<b>C++/CLI Programming</b>	<b>308</b>
Understanding Generic Functions	309
Defining Generic Functions	309
Using Generic Functions	310
A Calculator Program for the CLR	315
Removing Spaces from the Input String	316
Evaluating an Arithmetic Expression	316
Obtaining the Value of a Term	318
Evaluating a Number	318
Extracting a Parenthesized Substring	319
<b>Summary</b>	<b>320</b>
<b>Exercises</b>	<b>321</b>
<b>Chapter 7: Defining Your Own Data Types</b>	<b>323</b>
<b>The struct in C++</b>	<b>324</b>
What Is a struct?	324
Defining a struct	324
Initializing a struct	325
Accessing the Members of a struct	325
Intellisense Assistance with Structures	329
The struct RECT	330
Using Pointers with a struct	330
Accessing Structure Members through a Pointer	332
The Indirect Member Selection Operator	332
<b>Data Types, Objects, Classes and Instances</b>	<b>332</b>
First Class	334
Operations on Classes	334
Terminology	335
<b>Understanding Classes</b>	<b>335</b>
Defining a Class	336
Access Control in a Class	336
Declaring Objects of a Class	336
Accessing the Data Members of a Class	337
Member Functions of a Class	339

# Contents

---

Positioning a Member Function Definition	341
Inline Functions	342
<b>Class Constructors</b>	<b>343</b>
What Is a Constructor?	343
The Default Constructor	345
Assigning Default Parameter Values in a Class	347
Using an Initialization List in a Constructor	350
<b>Private Members of a Class</b>	<b>350</b>
Accessing private Class Members	353
The friend Functions of a Class	354
Placing friend Function Definitions Inside the Class	356
The Default Copy Constructor	356
<b>The Pointer this</b>	<b>358</b>
<b>const Objects of a Class</b>	<b>360</b>
const Member Functions of a Class	361
Member Function Definitions Outside the Class	362
<b>Arrays of Objects of a Class</b>	<b>363</b>
<b>Static Members of a Class</b>	<b>364</b>
Static Data Members of a Class	365
Static Function Members of a Class	367
<b>Pointers and References to Class Objects</b>	<b>368</b>
Pointers to Class Objects	368
References to Class Objects	371
Implementing a Copy Constructor	371
<b>C++/CLI Programming</b>	<b>372</b>
Defining Value Class Types	373
The ToString() Function in a Class	376
Literal Fields	377
Defining Reference Class Types	378
Class Properties	381
Defining Scalar Properties	382
Trivial Scalar Properties	384
Defining Indexed Properties	388
More Complex Indexed Properties	392
Static Properties	393
Reserved Property Names	394
initonly Fields	394
Static Constructors	396
<b>Summary</b>	<b>396</b>
<b>Exercises</b>	<b>397</b>

---

<b>Chapter 8: More on Classes</b>	<b>399</b>
<b>Class Destructors</b>	<b>399</b>
What Is a Destructor?	399
The Default Destructor	400
Destructors and Dynamic Memory Allocation	402
<b>Implementing a Copy Constructor</b>	<b>405</b>
<b>Sharing Memory Between Variables</b>	<b>407</b>
Defining Unions	408
Anonymous Unions	409
Unions in Classes and Structures	410
<b>Operator Overloading</b>	<b>410</b>
Implementing an Overloaded Operator	411
Implementing Full Support for an Operator	414
Overloading the Assignment Operator	418
Fixing the Problem	418
Overloading the Addition Operator	423
Overloading the Increment and Decrement Operators	426
<b>Class Templates</b>	<b>427</b>
Defining a Class Template	428
Template Member Functions	430
Creating Objects from a Class Template	431
Class Templates with Multiple Parameters	434
<b>Using Classes</b>	<b>436</b>
The Idea of a Class Interface	436
Defining the Problem	436
Implementing the CBox Class	437
Comparing CBox Objects	438
Combining CBox Objects	439
Analyzing CBox Objects	441
Defining the CBox Class	445
Adding Data Members	446
Defining the Constructor	447
Adding Function Members	448
Adding Global Functions	453
Using Our CBox Class	455
<b>Organizing Your Program Code</b>	<b>458</b>
Naming Program Files	460
<b>C++/CLI Programming</b>	<b>461</b>
Overloading Operators in Value Classes	461
Overloading the Increment and Decrement Operators	467
Overloading Operators in Reference Classes	467
<b>Summary</b>	<b>470</b>
<b>Exercises</b>	<b>471</b>

<b>Chapter 9: Class Inheritance and Virtual Functions</b>	<b>473</b>
<b>Basic Ideas of OOP</b>	<b>473</b>
<b>Inheritance in Classes</b>	<b>475</b>
What Is a Base Class?	475
Deriving Classes from a Base Class	476
<b>Access Control Under Inheritance</b>	<b>479</b>
Constructor Operation in a Derived Class	482
Declaring Class Members to be Protected	486
The Access Level of Inherited Class Members	489
<b>The Copy Constructor in a Derived Class</b>	<b>490</b>
<b>Class Members as Friends</b>	<b>495</b>
Friend Classes	496
Limitations on Class Friendship	496
<b>Virtual Functions</b>	<b>497</b>
What Is a Virtual Function?	499
Using Pointers to Class Objects	501
Using References With Virtual Functions	503
Incomplete Class Definitions	504
Pure Virtual Functions	505
Abstract Classes	505
Indirect Base Classes	508
Virtual Destructors	511
<b>Casting Between Class Types</b>	<b>516</b>
<b>Nested Classes</b>	<b>516</b>
<b>C++/CLI Programming</b>	<b>520</b>
Inheritance in C++/CLI Classes	520
Interface Classes	526
Defining Interface Classes	527
Classes and Assemblies	531
Visibility Specifiers for Classes and Interfaces	531
Access Specifiers for Class and Interface Members	531
Functions Specified as new	536
Delegates and Events	536
Declaring Delegates	537
Creating Delegates	537
Unbound Delegates	541
Creating Events	545
Destructors and Finalizers in Reference Classes	549
Generic Classes	551
Generic Interface Classes	554
Generic Collection Classes	555
<b>Summary</b>	<b>561</b>
<b>Exercises</b>	<b>562</b>

---

<b>Chapter 10: Debugging Techniques</b>	<b>565</b>
<b>Understanding Debugging</b>	<b>565</b>
Program Bugs	567
Common Bugs	567
<b>Basic Debugging Operations</b>	<b>568</b>
Setting Breakpoints	570
Advanced Breakpoints	572
Setting Tracepoints	572
Starting Debugging	573
Inspecting Variable Values	576
Viewing Variables in the Edit Window	577
Changing the Value of a Variable	577
<b>Adding Debugging Code</b>	<b>578</b>
Using Assertions	578
Adding Your Own Debugging Code	580
<b>Debugging a Program</b>	<b>585</b>
The Call Stack	585
Step Over to the Error	587
<b>Testing the Extended Class</b>	<b>591</b>
Finding the Next Bug	593
<b>Debugging Dynamic Memory</b>	<b>593</b>
Functions Checking the Free Store	594
Controlling Free Store Debug Operations	595
Free Store Debugging Output	596
<b>Debugging C++/CLI Programs</b>	<b>602</b>
Using the Debug and Trace Classes	602
Generating Output	603
Setting the Output Destination	604
Indenting the Output	604
Controlling Output	605
Assertions	607
<b>Summary</b>	<b>611</b>
<b>Chapter 11: Windows Programming Concepts</b>	<b>613</b>
<b>Windows Programming Basics</b>	<b>614</b>
Elements of a Window	614
Windows Programs and the Operating System	616
Event-Driven Programs	617
Windows Messages	617
The Windows API	617
Windows Data Types	618
Notation in Windows Programs	619

# Contents

---

<b>The Structure of a Windows Program</b>	<b>620</b>
The WinMain() Function	621
Specifying a Program Window	623
Creating a Program Window	625
Initializing the Program Window	627
Dealing with Windows Messages	627
A Complete WinMain() Function	631
Message Processing Functions	632
The WindowProc() Function	633
Decoding a Windows Message	633
Ending the Program	636
A Complete WindowProc() Function	636
A Simple Windows Program	637
<b>Windows Program Organization</b>	<b>638</b>
<b>The Microsoft Foundation Classes</b>	<b>640</b>
MFC Notation	640
How an MFC Program Is Structured	640
The Application Class	641
The Window Class	642
Completing the Program	642
The Finished Product	643
<b>Using Windows Forms</b>	<b>645</b>
<b>Summary</b>	<b>647</b>
<b>Chapter 12: Windows Programming with the Microsoft Foundation Classes</b>	<b>649</b>
<b>The Document/View Concept in MFC</b>	<b>650</b>
What Is a Document?	650
Document Interfaces	650
What Is a View?	651
Linking a Document and Its Views	652
Document Templates	652
Document Template Classes	653
Your Application and MFC	653
<b>Creating MFC Applications</b>	<b>655</b>
Creating an SDI Application	657
The Output from the MFC Application Wizard	660
Viewing Project Files	662
Viewing Classes	663
The Class Definitions	664
Creating an Executable Module	667

Running the Program	668
How the Program Works	669
Creating an MDI Application	671
Running the Program	672
<b>Summary</b>	<b>674</b>
<b>Exercises</b>	<b>674</b>

---

**Chapter 13: Working with Menus and Toolbars** **677**

<b>Communicating with Windows</b>	<b>677</b>
Understanding Message Maps	678
Message Handler Definitions	679
Message Categories	681
Handling Messages in Your Program	682
How Command Messages Are Processed	682
<b>Extending the Sketcher Program</b>	<b>683</b>
<b>Elements of a Menu</b>	<b>684</b>
Creating and Editing Menu Resources	684
Adding a Menu Item to the Menu Bar	684
Adding Items to the Element Menu	686
Modifying Existing Menu Items	687
Completing the Menu	687
<b>Adding Handlers for Menu Messages</b>	<b>688</b>
Choosing a Class to Handle Menu Messages	690
Creating Menu Message Functions	690
Coding Menu Message Functions	692
Adding Members to Store Color and Element Mode	692
Initializing the New Class Data Members	694
Running the Extended Example	696
Adding Message Handlers to Update the User Interface	697
Coding a Command Update Handler	697
Exercising the Update Handlers	699
<b>Adding Toolbar Buttons</b>	<b>700</b>
Editing Toolbar Button Properties	701
Exercising the Toolbar Buttons	703
Adding Tooltips	703
<b>Summary</b>	<b>704</b>
<b>Exercises</b>	<b>705</b>

## **Chapter 14: Drawing in a Window** **707**

---

### **Basics of Drawing in a Window** **707**

The Window Client Area	708
The Windows Graphical Device Interface	709
What Is a Device Context?	709
Mapping Modes	709

### **The Drawing Mechanism in Visual C++** **711**

The View Class in Your Application	711
The OnDraw() Member Function	711
The CDC Class	712
Displaying Graphics	713
Drawing in Color	717

### **Drawing Graphics in Practice** **721**

#### **Programming the Mouse** **725**

Messages from the Mouse	725
WM_LBUTTONDOWN	726
WM_MOUSEMOVE	726
WM_LBUTTONUP	727
Mouse Message Handlers	727
Drawing Using the Mouse	729
Getting the Client Area Redrawn	731
Defining Classes for Elements	732
The CElement Class	736
The CLine Class	737
Calculating the Enclosing Rectangle for a Line	742
The CRectangle Class	742
The CCircle Class	744
The CCurve Class	746
Completing the Mouse Message Handlers	747

#### **Exercising Sketcher** **753**

Running the Example	754
Capturing Mouse Messages	755

#### **Summary** **756**

#### **Exercises** **757**

## **Chapter 15: Creating the Document and Improving the View** **759**

---

### **What Are Collection Classes?** **759**

Types of Collection	760
The Type-Safe Collection Classes	761

Collections of Objects	761
The CArray Template Class	761
Helper Functions	763
The CList Template Class	763
The CMap Template Class	768
The Typed Pointer Collections	771
The CTypedPtrList Template Class	771
CTypedPtrList Operations	771
<b>Using the CList Template Class</b>	<b>773</b>
Drawing a Curve	774
Defining the CCurve Class	775
Implementing the CCurve Class	777
Exercising the CCurve Class	778
<b>Creating the Document</b>	<b>779</b>
Using a CTypedPtrList Template	779
Implementing the Document Destructor	780
Drawing the Document	781
Adding an Element to the Document	783
Exercising the Document	784
<b>Improving the View</b>	<b>785</b>
Updating Multiple Views	785
Scrolling Views	787
Logical Coordinates and Client Coordinates	789
Dealing with Client Coordinates	790
Using MM_LOENGLISH Mapping Mode	792
<b>Deleting and Moving Shapes</b>	<b>793</b>
<b>Implementing a Context Menu</b>	<b>794</b>
Associating a Menu with a Class	795
Choosing a Context Menu	797
Identifying a Selected Element	798
Exercising the Pop-Ups	800
Checking the Context Menu Items	800
Highlighting Elements	802
Drawing Highlighted Elements	806
Exercising the Highlights	807
Servicing the Menu Messages	807
Deleting an Element	807
Moving an Element	808
Getting the Elements to Move Themselves	811
Exercising the Application	814
<b>Dealing with Masked Elements</b>	<b>814</b>
<b>Summary</b>	<b>815</b>
<b>Exercises</b>	<b>816</b>

<b>Chapter 16: Working with Dialogs and Controls</b>	<b>817</b>
<b>Understanding Dialogs</b>	<b>817</b>
<b>Understanding Controls</b>	<b>818</b>
Common Controls	820
<b>Creating a Dialog Resource</b>	<b>820</b>
Adding Controls to a Dialog Box	820
Testing the Dialog	822
<b>Programming for a Dialog</b>	<b>822</b>
Adding a Dialog Class	823
Modal and Modeless Dialogs	824
Displaying a Dialog	824
Code to Display the Dialog	826
Code to Close the Dialog	827
<b>Supporting the Dialog Controls</b>	<b>828</b>
Initializing the Controls	828
Handling Radio Button Messages	830
<b>Completing Dialog Operations</b>	<b>831</b>
Adding Pen Widths to the Document	832
Adding Pen Widths to the Elements	832
Creating Elements in the View	833
Exercising the Dialog	834
<b>Using a Spin Button Control</b>	<b>835</b>
Adding the Scale Menu Item and Toolbar Button	835
Creating the Spin Button	835
The Controls' Tab Sequence	838
Generating the Scale Dialog Class	838
Dialog Data Exchange and Validation	840
Initializing the Dialog	840
Displaying the Spin Button	841
<b>Using the Scale Factor</b>	<b>842</b>
Scaleable Mapping Modes	842
Setting the Document Size	844
Setting the Mapping Mode	844
Implementing Scrolling with Scaling	846
Setting Up the Scrollbars	847
<b>Working with Status Bars</b>	<b>848</b>
Adding a Status Bar to a Frame	848
Defining the Status Bar Parts	850
Updating the Status Bar	851

---

<b>Using a List Box</b>	<b>852</b>
Removing the Scale Dialog	852
Creating a List Box Control	853
Creating the Dialog Class	854
Displaying the Dialog	855
<b>Using an Edit Box Control</b>	<b>856</b>
Creating an Edit Box Resource	856
Creating the Dialog Class	858
The CString Class	858
Adding the Text Menu Item	859
Defining a Text Element	860
Implementing the CText Class	861
The CText Constructor	861
Drawing a CText Object	862
Moving a CText Object	862
Creating a Text Element	863
<b>Summary</b>	<b>864</b>
<b>Exercises</b>	<b>865</b>
<b>Chapter 17: Storing and Printing Documents</b>	<b>867</b>
<hr/>	
<b>Understanding Serialization</b>	<b>867</b>
<b>Serializing a Document</b>	<b>868</b>
Serialization in the Document Class Definition	868
Serialization in the Document Class Implementation	869
The Serialize() Function	870
The CArchive Class	870
Functionality of CObject-Based Classes	872
The Macros Adding Serialization to a Class	872
How Serialization Works	873
How to Implement Serialization for a Class	874
<b>Applying Serialization</b>	<b>874</b>
Recording Document Changes	874
Serializing the Document	876
Serializing the Element Classes	877
The Serialize() Functions for the Shape Classes	879
<b>Exercising Serialization</b>	<b>881</b>
<b>Moving Text</b>	<b>882</b>
<b>Printing a Document</b>	<b>884</b>
The Printing Process	885
The CPrintInfo Class	886

# Contents

---

<b>Implementing Multipage Printing</b>	<b>888</b>
Getting the Overall Document Size	889
Storing Print Data	890
Preparing to Print	891
Cleaning Up After Printing	892
Preparing the Device Context	893
Printing the Document	894
Getting a Printout of the Document	898
<b>Summary</b>	<b>898</b>
<b>Exercises</b>	<b>899</b>
<b>Chapter 18: Writing Your Own DLLs</b>	<b>901</b>
<b>Understanding DLLs</b>	<b>901</b>
How DLLs Work	903
Run-Time Dynamic Linking	904
Contents of a DLL	906
The DLL Interface	906
TheDllMain() Function	906
DLL Varieties	906
MFC Extension DLL	906
Regular DLL — Statically Linked to MFC	907
Regular DLL — Dynamically Linked to MFC	907
<b>Deciding What to Put in a DLL</b>	<b>907</b>
<b>Writing DLLs</b>	<b>908</b>
Writing and Using an Extension DLL	908
UnderstandingDllMain()	910
Adding Classes to the Extension DLL	911
Exporting Classes from the Extension DLL	912
Building a DLL	913
Using the Extension DLL in Sketcher	914
Files Required to Use a DLL	915
Exporting Variables and Functions from a DLL	916
Importing Symbols into a Program	917
Implementing the Export of Symbols from a DLL	917
Using Exported Symbols	918
<b>Summary</b>	<b>920</b>
<b>Exercises</b>	<b>920</b>

---

<b>Chapter 19: Connecting to Data Sources</b>	<b>921</b>
<b>Database Basics</b>	<b>921</b>
<b>A Little SQL</b>	<b>924</b>
Retrieving Data Using SQL	924
Choosing Records	925
Joining Tables Using SQL	926
Sorting Records	929
<b>Database Support in MFC</b>	<b>929</b>
MFC Classes Supporting ODBC	930
<b>Creating a Database Application</b>	<b>931</b>
Registering an ODBC Database	931
Generating an MFC ODBC Program	933
Snapshot versus Dynaset Recordsets	935
Understanding the Program Structure	936
Understanding Recordsets	937
Understanding the Record View	941
Creating the View Dialog	943
Linking the Controls to the Recordset	946
Exercising the Example	948
<b>Sorting a Recordset</b>	<b>948</b>
Modifying the Window Caption	949
<b>Using a Second Recordset Object</b>	<b>950</b>
Adding a Recordset Class	950
Adding a View Class for the Recordset	954
Creating the Dialog Resource	954
Creating the Record View Class	955
Customizing the Recordset	957
Adding a Filter to the Recordset	958
Defining the Filter Parameter	958
Initializing the Record View	960
Accessing Multiple Table Views	961
Switching Views	961
Enabling the Switching Operation	963
Handling View Activation	965
Viewing Orders for a Product	966
<b>Viewing Customer Details</b>	<b>967</b>
Adding the Customer Recordset	967
Creating the Customer Dialog Resource	968
Creating the Customer View Class	968
Adding a Filter	970
Implementing the Filter Parameter	972

# Contents

---

Linking the Order Dialog to the Customer Dialog	973
Exercising the Database Viewer	976
<b>Summary</b>	<b>976</b>
<b>Exercises</b>	<b>977</b>
<b>Chapter 20: Updating Data Sources</b>	<b>979</b>
<b>Update Operations</b>	<b>979</b>
CRecordset Update Operations	980
Checking that Operations are Legal	981
Record Locking	982
Transactions	982
CDatabase Transaction Operations	983
<b>A Simple Update Example</b>	<b>984</b>
Customizing the Application	986
<b>Managing the Update Process</b>	<b>988</b>
Implementing Update Mode	990
Enabling and Disabling Edit Controls	991
Changing the Button Label	992
Controlling the Visibility of the Cancel Button	993
Disabling the Record Menu	994
Expediting the Update	996
Implementing the Cancel Operation	997
<b>Adding Rows to a Table</b>	<b>999</b>
The Order Entry Process	1000
Creating the Resources	1001
Creating the Recordsets	1002
Creating the Recordset Views	1002
Adding Controls to the Dialog Resources	1006
Implementing Dialog Switching	1010
Creating an Order ID	1014
Storing the New Order ID	1014
Creating the New Order ID	1015
Initiating ID Creation	1017
Storing the Order Data	1019
Setting Dates	1020
Selecting Products for an Order	1021
Adding a New Order	1023
<b>Summary</b>	<b>1028</b>
<b>Exercises</b>	<b>1029</b>

---

<b>Chapter 21: Applications Using Windows Forms</b>	<b>1031</b>
<b>Understanding Windows Forms</b>	<b>1031</b>
<b>Understanding Windows Forms Applications</b>	<b>1032</b>
Modifying the Properties of a Form	1034
How the Application Starts	1035
<b>Customizing the Application GUI</b>	<b>1035</b>
Adding Controls to a Form	1036
Adding Menus	1037
Adding Submenus	1038
Adding a Tab Control	1040
Using GroupBox Controls	1042
Using Button Controls	1044
Using the WebBrowser Control	1047
Operation of the Winning Application	1048
Adding a Context Menu	1049
Creating Event Handlers	1049
Event Handlers for Menu Items	1050
Adding Members to the Form1 Class	1050
Handling the Play Menu Event	1052
Handling Events for the Limits Menu	1056
Creating a Dialog Box	1056
Adding a List to a ListBox	1058
Handling the Dialog Button Events	1060
Controlling the State of the ListBox Objects	1061
Creating the Dialog Object	1061
Using the Dialog Box	1062
Validating the Input	1063
Handler the Reset Menu Item Event	1067
Adding the Second Dialog	1068
Getting the Data from the Dialog Controls	1070
Disabling Input Controls	1073
Updating the Limits Menu Item Handlers	1074
Implementing the Help   About Menu Item	1075
Handling a Button Click	1076
Responding to the Context Menu	1079
The Logic for Dealing with the Choose Menu Item	1080
Creating the Dialog Form	1080
Developing the Dialog Class	1081
Handling the Click Event for the ChooseMenu	1083
<b>Summary</b>	<b>1086</b>
<b>Exercises</b>	<b>1087</b>

# Contents

---

<b>Chapter 22: Accessing Data Sources in a Windows Forms Application</b>	<b>1089</b>
<b>Working with Data Sources</b>	<b>1089</b>
<b>Accessing and Displaying Data</b>	<b>1090</b>
<b>Using a DataGridView Control</b>	<b>1091</b>
<b>Using a DataGridView Control in Unbound Mode</b>	<b>1093</b>
<b>Customizing a DataGridView Control</b>	<b>1099</b>
Customizing Header Cells	1101
Customizing Non-Header Cells	1101
Setting Up the Data	1102
Setting Up the Control	1103
Setting Up the Column Headers	1105
Formatting a Column	1106
Customizing Alternate Rows	1108
Dynamically Setting Cell Styles	1108
<b>Using Bound Mode</b>	<b>1114</b>
<b>The BindingSource Component</b>	<b>1115</b>
<b>Using the BindingNavigator Control</b>	<b>1120</b>
<b>Binding to Individual Controls</b>	<b>1123</b>
<b>Working with Multiple Tables</b>	<b>1127</b>
<b>Summary</b>	<b>1129</b>
<b>Exercises</b>	<b>1129</b>
<b>Appendix A: C++ Keywords</b>	<b>1131</b>
<b>Appendix B: ASCII Codes</b>	<b>1133</b>
<b>Index</b>	<b>1139</b>