

Contents

| | |
|---|--------------|
| <i>About the Authors</i> | <i>xxi</i> |
| <i>About the Technical Reviewer</i> | <i>xxiii</i> |
| <i>Acknowledgments</i> | <i>xxv</i> |
| <i>Introduction</i> | <i>xxvii</i> |

| | |
|--|-----------------|
| <i>Chapter 1 ADO.NET Basics</i> | <i>1</i> |
|--|-----------------|

| | |
|---|-----------|
| <i>Overview of Microsoft Data Access Technologies</i> | <i>2</i> |
| ODBC | <i>2</i> |
| DAO | <i>4</i> |
| MFC ODBC and DAO Classes | <i>5</i> |
| RDO | <i>5</i> |
| OLE DB | <i>5</i> |
| ADO | <i>6</i> |
| ADO.NET | <i>6</i> |
| <i>What Is ADO.NET?</i> | <i>7</i> |
| <i>Why ADO.NET Was Designed</i> | <i>8</i> |
| <i>Advantages of ADO.NET</i> | <i>9</i> |
| Single Object-Oriented API | <i>9</i> |
| Managed Code | <i>10</i> |
| XML Support | <i>10</i> |
| Visual Data Components | <i>11</i> |
| Performance and Scalability | <i>11</i> |
| <i>Comparing ADO.NET and ADO</i> | <i>11</i> |
| Connections and Disconnected Data | <i>11</i> |
| Recordsets vs. DataSets | <i>12</i> |
| XML Support | <i>12</i> |
| <i>Overview of ADO.NET Namespaces and Classes</i> | <i>13</i> |
| <i>Understanding ADO.NET Components</i> | <i>15</i> |
| The Connection Object | <i>16</i> |
| The Command Object | <i>17</i> |
| The Command Builder | <i>19</i> |
| The DataAdapter Object | <i>19</i> |
| DataSet Structure | <i>20</i> |
| DataSets in DataViews | <i>21</i> |

| | |
|--|----|
| <i>Writing ADO.NET Applications</i> | 22 |
| Choosing a .NET Data Provider | 22 |
| Adding Namespace References | 23 |
| Establishing a Connection | 24 |
| Creating a Command or DataAdapter Object | 25 |
| Filling Data to a DataSet or DataReader Object | 25 |
| Displaying Data | 25 |
| Closing the Connection | 26 |
| <i>Creating a Simple ADO.NET Application</i> | 26 |
| <i>Using a DataSet to Read Data</i> | 28 |
| <i>Summary</i> | 31 |
| Chapter 2 Data Components in Visual Studio .NET | 33 |
| <i>Creating an ADO.NET Project</i> | 33 |
| <i>Using the Server Explorer</i> | 34 |
| Adding a New Connection | 35 |
| Managing and Viewing Data | 36 |
| <i>Using Visual Data Components</i> | 37 |
| Understanding Data Connections | 38 |
| Understanding Connection Strings | 40 |
| Working with SQL DataAdapters | 40 |
| <i>Using DataSet and DataView Components</i> | 51 |
| Understanding Typed DataSets in VS .NET | 51 |
| Understanding the DataView | 54 |
| <i>Using the Data Form Wizard</i> | 57 |
| Data Form Wizard: Looking under the Hood | 66 |
| Understanding MyDS.xsd | 67 |
| Understanding DataForm1.vb | 67 |
| <i>Summary</i> | 71 |
| Chapter 3 ADO.NET in Disconnected Environments | 73 |
| <i>Understanding the ADO.NET Architecture</i> | 75 |
| <i>Exploring the ADO.NET Class Hierarchy</i> | 76 |
| <i>Choosing a Data Provider</i> | 80 |
| <i>Understanding ADO.NET Disconnected Classes</i> | 81 |
| The System.Data Namespace | 81 |
| The System.Data.Common Namespace | 82 |
| <i>Working with DataTables</i> | 83 |
| The DataColumn | 85 |
| The DataRow | 94 |

| | |
|--|-----|
| The DataRelation | 99 |
| The DataTable | 100 |
| More DataTable Operations | 106 |
| The Row and Column Collections | 115 |
| The DataRow States and Versions | 117 |
| <i>Using DataSet, DataView, and DataViewManager</i> | 120 |
| The DataSet: The Heart of ADO.NET | 121 |
| Typed and Untyped DataSets | 127 |
| The DataView | 128 |
| The DataViewManager | 132 |
| <i>Summary</i> | 133 |
| Chapter 4 ADO.NET in Connected Environments | 135 |
| <i>Understanding the Generic Data Provider Model</i> | 135 |
| <i>Importing a Namespace</i> | 137 |
| <i>Exploring the System.SqlClient Namespace</i> | 137 |
| <i>The Connection: Connecting to a Data Source</i> | 139 |
| Creating a Connection | 139 |
| Understanding the Connection Properties and Methods | 140 |
| Opening and Closing a Connection | 142 |
| Understanding Connection Pooling | 145 |
| Using the CreateCommand and ChangeDatabase Methods | 149 |
| <i>The Command: Executing SQL Statements</i> | 150 |
| Creating a Command Object | 151 |
| Creating and Using OleDbCommand | 153 |
| Using the CommandType Enumeration | 154 |
| Calling a Stored Procedure | 155 |
| Using TableDirect | 157 |
| Executing a Command | 158 |
| Using Other Command Methods | 160 |
| <i>The DataReader: Walking through the Data</i> | 160 |
| Initializing a DataReader | 161 |
| Understanding DataReader Properties and Methods | 161 |
| Reading with the DataReader | 162 |
| Interpreting Batches of Queries | 164 |
| Understanding Command Behaviors | 165 |
| <i>The DataAdapter: Adapting to Your Environment</i> | 166 |
| Constructing a DataAdapter Object | 166 |
| Understanding DataAdapter Properties | 168 |
| Understanding DataAdapter Methods | 170 |
| Filling the DataSet | 170 |
| Adding a DataTable to a DataSet | 172 |

| | |
|--|-----|
| Looking at a DataAdapter Example | 172 |
| Filling the DataAdapter from a Recordset | 175 |
| Updating the Database Using the Update Method | 176 |
| Table and Column Mapping | 178 |
| <i>CommandBuilder: Easing the Work of Programmers</i> | 181 |
| Creating a CommandBuilder Object | 182 |
| Using SqlCommandBuilder Members | 182 |
| Using SqlCommandBuilder | 182 |
| <i>Staying within the Parameters</i> | 184 |
| <i>The DataSet in Connected Environments</i> | 188 |
| Filling a DataSet from Multiple Tables | 188 |
| Using a SQL Statement to Read Data from Two or More Tables | 190 |
| Adding, Updating, and Deleting Data through the DataSet | 190 |
| Accepting and Rejecting Changes through the DataSet | 191 |
| Saving Changed Data Only | 192 |
| Using a DataSet vs. Using a DataReader | 193 |
| Fetching Data Based on Wildcard Characters | 194 |
| <i>The DataView in Connected Environments</i> | 194 |
| Creating Multiple Views | 195 |
| Using Transactions in ADO.NET | 200 |
| Using Concurrency in ADO.NET | 201 |
| Understanding Rollback, Commit, and Savepoints | 204 |
| Executing Batches | 206 |
| <i>Summary</i> | 208 |
| Chapter 5 Handling ADO.NET Events | 209 |
| Working with Connection Events | 211 |
| Working with DataAdapter Events | 215 |
| Working with DataSet Events | 220 |
| Working with DataTable Events | 221 |
| Working with XmlDataDocument Events | 226 |
| Working with DataView and DataViewManager Events | 229 |
| Summary | 231 |
| Chapter 6 Integrating XML with ADO.NET | 233 |
| Understanding Microsoft .NET and XML | 233 |
| Using the System.Xml Namespace | 234 |
| Using the System.Xml.Schema Namespace | 235 |
| Using the System.Xml.Serialization Namespace | 236 |
| Using the System.Xml.XPath Namespace | 236 |

| | |
|--|-----|
| Using the System.Xml.Xsl Namespace | 236 |
| Using the Document Object Model Interfaces | 237 |
| Looking at the XML .NET Architecture | 237 |
| Adding a System.Xml Namespace Reference | 238 |
| <i>Reading XML</i> | 238 |
| Getting Node Information | 239 |
| Moving to a Content Node | 242 |
| Using the GetAttributes of a Node | 243 |
| Searching for a Node | 244 |
| Closing the Document | 245 |
| <i>Understanding the XmlWriter Classes</i> | 246 |
| Setting XmlWriter Properties | 247 |
| Writing XML Items | 247 |
| Seeing XmlWriter in Action | 248 |
| Using the Close Method | 250 |
| <i>Using the XmlConvert Class</i> | 251 |
| <i>Understanding the DOM Implementation</i> | 251 |
| Using the XmlNode Class | 252 |
| Using the XmlDocument Class | 252 |
| Saving a Document | 253 |
| Using the XmlDocumentFragment Class | 253 |
| Using the XmlElement Class | 254 |
| Adding Attributes to a Node | 259 |
| <i>Understanding Transformation and XSLT</i> | 260 |
| Using the Transform Method | 261 |
| Looking at an Example | 261 |
| <i>Using ADO.NET and XML Together</i> | 262 |
| Reading XML Using a DataSet | 262 |
| Writing XML Using a DataSet | 264 |
| Using XmlDataDocument and XML | 267 |
| Displaying XML Data in a DataSet Format | 270 |
| Saving Data from a DataSet to XML | 271 |
| XmlDataDocument: Looking under the Hood | 272 |
| <i>Navigating in XML</i> | 272 |
| Using the XPathNavigator Class | 273 |
| Using XPathNavigator Move Methods | 274 |
| Searching Using XPathNavigator | 276 |
| <i>Working with Schemas</i> | 277 |
| Generating a New Schema | 277 |
| Adding Schema from a Database Objects | 283 |
| Generating a Typed DataSet from a Schema | 285 |
| <i>Summary</i> | 286 |

| | |
|--|---------|
| Chapter 7 Data Binding and Windows Forms | |
| Data-Bound Controls | 287 |
| <i>Understanding Data Binding</i> | 287 |
| Using the Binding Class | 288 |
| Understanding the BindingManagerBase Functionality | 293 |
| Understanding BindingContext | 296 |
| Building a Record Navigation System | 296 |
| <i>Working with Complex Data-Bound Controls</i> | 301 |
| The Role of Control Class in Data Binding | 301 |
| Using the ListControl Class | 301 |
| ListControl DataBinding-Related Events | 302 |
| Data Binding in ComboBox and ListBox Controls | 304 |
| Data Binding in a DataGrid Control | 306 |
| Deleting Data Binding | 307 |
| <i>The DataGrid: Super Data-Bound Control</i> | 307 |
| Understanding the DataGrid Class Members | 309 |
| Exploring the DataGrid Helper Objects | 312 |
| Understanding the DataGrid and DataGrid Column Styles | 313 |
| Seeing DataGridTableStyle and DataGridColumnStyle in Action | 321 |
| Seeing HitTest in Action | 324 |
| Implementing Custom Sorting in a DataGrid | 334 |
| Building a DataGrid Record Navigation System | 335 |
| Implementing Search in a DataGrid | 338 |
| Inserting, Updating, and Deleting Data through DataGrids | 342 |
| Summary | 344 |
| Chapter 8 Constraints and Data Relations | 345 |
| <i>Understanding Constraints</i> | 345 |
| Implementing the Constraint Classes | 346 |
| Implementing the ForeignKeyConstraint Class | 346 |
| <i>Understanding Data Relations</i> | 353 |
| <i>Understanding the DataRelation and DataRelationCollection Objects</i> | 354 |
| Summary | 358 |
| Chapter 9 ADO.NET Exception Handling | 359 |
| <i>Why Use Exception Handling?</i> | 360 |
| Using Try...Catch Statements | 361 |
| Using Try...Catch...Finally Statements | 362 |

| | |
|---|-----|
| Nesting Try...Catch Statements | 363 |
| Using Try with Multiple Catch Statements | 363 |
| Defining Custom Filters and Error Messages | 364 |
| <i>Understanding the Exception Classes</i> | 365 |
| Using the Exception Class: Mother of All Exceptions | 365 |
| Understanding Other Exception Classes | 366 |
| <i>Exception Handling in ADO.NET</i> | 366 |
| Recognizing SQL Server 2000 Errors | 368 |
| Using SQL Server Error Logs | 370 |
| Using the SqlException Class | 371 |
| Using the SqlError and SqlErrorCollection Classes | 373 |
| <i>Summary</i> | 374 |

Chapter 10 Working with the ODBC

| | |
|---|-----|
| .NET Data Provider | 375 |
| <i>Using the ODBC .NET Data Provider</i> | 375 |
| Installing the ODBC .NET Data Provider | 376 |
| Understanding the ODBC .NET Data Provider | 377 |
| <i>Accessing Access Databases</i> | 378 |
| <i>Accessing MySQL Databases</i> | 380 |
| <i>Accessing MySQL Databases</i> | 381 |
| <i>Accessing Text File Databases</i> | 382 |
| Exporting an Access Table to a Text File | 383 |
| Accessing a Text File | 385 |
| <i>Accessing Excel Databases</i> | 389 |
| <i>Implementing Two .NET 1.1 Updates</i> | 391 |
| <i>Summary</i> | 391 |

Chapter 11 Stored Procedures and Views

| | |
|---|-----|
| <i>Working with Stored Procedures</i> | 393 |
| Creating a Stored Procedure | 393 |
| Executing a Stored Procedure from VS .NET | 397 |
| Executing a Stored Procedure Programmatically | 399 |
| <i>Understanding SQL Injection</i> | 405 |
| <i>Working with Views</i> | 406 |
| Creating a View | 406 |
| Executing a View from VS .NET | 409 |
| Retrieving Data from a View Programmatically | 409 |
| Filling a DataSet from Multiple Views | 411 |
| <i>Summary</i> | 411 |

| | |
|--|-----|
| Chapter 12 Oracle, SQLXML, and Other .NET Data Providers | 413 |
| <i>Using Oracle .NET Data Providers</i> | 413 |
| Adding a Reference to the Oracle .NET Data Provider | 415 |
| Using the Oracle .NET Data Provider Classes | 416 |
| <i>Using the SQLXML and SQL XML .NET Data Providers</i> | 426 |
| Using the FOR XML Keyword | 427 |
| Using the SqlXmlAdapter Object | 428 |
| Using the SqlXmlParameter Object | 429 |
| Using the SqlXmlCommand Object | 430 |
| <i>Using a MySQL Database and the .NET Data Provider</i> | 435 |
| <i>Summary</i> | 435 |
| Chapter 13 Developing a Custom Data Provider | 437 |
| <i>Overview of the Data Provider Interfaces</i> | 438 |
| Understanding the Connection Object | 439 |
| Understanding the Command Object | 440 |
| Understanding the Reader Object | 441 |
| Understanding the Adapter Object | 443 |
| <i>Building a Custom Data Provider</i> | 446 |
| Creating the PipedDataConnection Object | 446 |
| Creating the PipedDataCommand Object | 449 |
| Creating the PipedDataReader Object | 453 |
| Creating the PipedDataAdapter Object | 460 |
| Testing the PipedDataProvider Application | 462 |
| <i>Debugging Your Custom Data Provider</i> | 464 |
| <i>Summary</i> | 464 |
| Chapter 14 Developing Database Web Applications Using ASP.NET | 465 |
| <i>Introducing ASP.NET</i> | 465 |
| ASP.NET Platform Requirements | 466 |
| ASP.NET Language Support | 466 |
| Installing ASP.NET | 466 |
| ASP.NET Editors | 466 |
| ASP.NET: An Evolution of ASP | 467 |
| Web Forms and Web Services | 468 |

| | |
|--|-----|
| <i>Developing Your First ASP.NET Web Application</i> | 468 |
| Creating a Web Application Project | 469 |
| Adding Web Controls to a Web Form | 472 |
| Setting Control Properties | 474 |
| Using Document Outline | 475 |
| Writing Code on the Button Click Event Handler | 477 |
| <i>Creating Your First ADO.NET Web Application</i> | 478 |
| <i>Using ASP.NET Server-Side Controls</i> | 480 |
| HTML Server Controls | 481 |
| Validation Controls | 482 |
| User Controls | 483 |
| Server Controls and the .NET Framework Library | 483 |
| Why Are Web Forms Controls Called Server-Side Controls? | 485 |
| Adding Server-Side Controls to a Web Form | 485 |
| <i>Understanding Data Binding in ASP.NET</i> | 487 |
| Data-Bound Controls | 488 |
| DataGrid and DataList Controls | 490 |
| Paging in the DataGrid Control | 499 |
| Enabling Paging Programmatically | 500 |
| Adding, Editing, and Deleting Data in Web Forms | 502 |
| Adding Data | 505 |
| Updating Data | 506 |
| Deleting Data | 507 |
| <i>Creating a Guest Book in ASP.NET</i> | 507 |
| Default Web Form: MyGuestBook.aspx | 508 |
| Adding Forms to the Guest Book | 511 |
| Compiling and Running the Guest Book Project | 514 |
| <i>Summary</i> | 516 |
| <i>Chapter 15 Using ADO.NET in XML Web Services</i> | 517 |
| <i>Exploring Web Services and the .NET Framework Library</i> | 518 |
| <i>Creating a Web Service in VS .NET</i> | 518 |
| <i>Testing Your Web Service</i> | 521 |
| <i>Creating the Web Service Consumer</i> | 524 |
| Adding Data to a Database Through a Web Service | 529 |
| Adding More Functionality to the Web Service | 534 |
| <i>Summary</i> | 535 |

| | |
|---|------------|
| Chapter 16 ASP.NET Server Controls and Data Binding..... | 537 |
| <i>Data Binding in Web Forms, Revisited</i> | <i>537</i> |
| Read-Only Data Binding | 537 |
| Simple and Complex Data Binding | 538 |
| The Roles of the Control and Page Classes in Data Binding | 539 |
| <i>Data Binding in Simple Web Forms Controls</i> | <i>540</i> |
| Using the Button Control | 540 |
| Using the Calendar Control | 542 |
| Understanding Web Server Control Templates | 546 |
| Using the Repeater Control | 552 |
| Understanding the Role of the ListControl Class | 555 |
| Using the ListBox Control | 556 |
| Using the DropDownList Control | 558 |
| Data Binding in Other Simple Controls | 560 |
| <i>Data Binding in Complex Web Forms Controls</i> | <i>562</i> |
| Understanding the BaseDataList Class | 562 |
| Using the DataKeyCollection Class | 563 |
| Using the DataList Web Server Control | 564 |
| Using the DataGrid Web Server Control | 582 |
| <i>Summary</i> | <i>608</i> |
| Chapter 17 Building Real-World Web Applications | 609 |
| <i>Introducing the Job Board Application Specifications</i> | <i>609</i> |
| <i>Understanding the Database Schema</i> | <i>610</i> |
| <i>Developing the Online Job Board Application</i> | <i>611</i> |
| Creating the Login.aspx Page | 612 |
| Creating the Register.aspx Page | 615 |
| Creating the Contact.aspx Page | 617 |
| Creating the PostJob.aspx Page | 618 |
| Creating the PostResume.aspx Page | 620 |
| Creating the DisplayData.aspx and Resumes.aspx Pages | 623 |
| Creating the dnjHeader.aspx Page | 632 |
| Running the Application | 633 |
| <i>Improving and Modifying the Application</i> | <i>635</i> |
| <i>Summary</i> | <i>636</i> |

| | |
|---|-----|
| Chapter 18 Object-Relational Mapping in .NET | 637 |
| <i>Why Not Just Use a DataSet?</i> | 638 |
| <i>Understanding Object Databases</i> | 639 |
| <i>Why Object Relational Mapping?</i> | 640 |
| <i>Creating the Basic Design of the Video Store Application</i> | 640 |
| Understanding the Basic Architecture | 641 |
| Creating the Database Design of the Video Store | 641 |
| Creating the Data Object | 643 |
| Creating the Mapped Object | 644 |
| Creating the Data Access Component | 648 |
| Building the VideoTape Test Case | 652 |
| Using the User Component | 654 |
| Creating the User Test Case | 659 |
| Managing Dependencies | 661 |
| Creating the Check In/Check Out Component | 663 |
| Seeing How It Works | 664 |
| <i>Making Design Tradeoffs</i> | 665 |
| Understanding Logic in Stored Procedures | 665 |
| Understanding Component Isolation | 666 |
| Creating a Workaround on Part of the Problem | 667 |
| Creating the VideoCheckInCheckOut Test Case | 670 |
| <i>Summary</i> | 673 |
| Chapter 19 Mapped Objects: Performance | |
| <i>Considerations and Data Binding</i> | 675 |
| <i>Using Data Binding and Mapped Objects</i> | 675 |
| Directly Accessing a Bound Object's Properties | 678 |
| Using Logic with Bound Objects | 680 |
| <i>Understanding Problematic Master-Detail Relationships</i> | 684 |
| Working with Highly Nested Structures | 685 |
| Working with a Much Larger Tree | 691 |
| A Final Note on Tree Optimization | 693 |
| <i>Working with Large Master-Detail Relationships</i> | 694 |
| Implementing Lazy Loading | 695 |
| Implementing an Incremental Lazy Loading Scheme | 700 |
| Loading the Detail When the Master Is Loaded | 705 |
| <i>Using a Near-Zero-Impact Caching Mechanism</i> | 706 |
| <i>Summary</i> | 708 |

| | |
|---|---------|
| Chapter 20 COM Interoperability and ADO.NET | 709 |
| <i>Exploring the .NET Interop Tools</i> | 710 |
| Using Visual Studio .NET | 710 |
| Using the Type Library Importer and Exporter (Tlbimp.exe) | 711 |
| Using the Type Library Exporter (Tlbexp.exe) | 712 |
| <i>Using the ADO Recordset in ADO.NET</i> | 712 |
| <i>Using ADOX with ADO.NET</i> | 716 |
| <i>Accessing OLAP Server Data with ADO.NET</i> | 718 |
| <i>Summary</i> | 724 |
| Chapter 21 Messaging | 725 |
| <i>Introducing Types of Queues</i> | 725 |
| Using User Queues | 726 |
| Using System Queues | 726 |
| <i>Installing Message Queuing</i> | 726 |
| <i>Using the System.Messaging Namespace</i> | 727 |
| <i>Working with Message Queues</i> | 729 |
| Using the MessageQueue Properties | 729 |
| Managing Queues Using VS .NET | 731 |
| Retrieving Available Queues | 732 |
| Filtering Queues Using MessageQueueCriteria | 734 |
| Creating and Deleting Message Queues Programmatically | 735 |
| <i>Creating a Sample Application</i> | 736 |
| Creating a MessageQueue Instance | 738 |
| Setting a Queue's Path | 739 |
| Sending and Receiving Messages | 740 |
| Walking Through Simple Messaging Application | 741 |
| <i>Working with Messages</i> | 745 |
| Creating and Removing Messages | 745 |
| Setting Queue and Message Priorities | 746 |
| <i>Understanding Transactional Messaging</i> | 747 |
| Using the MessageQueueTransaction Class | 747 |
| Creating Transactional Queues | 748 |
| Sending and Receiving Transactional Messages | 748 |
| <i>Encrypting Messages</i> | 750 |
| <i>Summary</i> | 752 |

| | |
|---|------------|
| Chapter 22 SQL Server and ADO.NET: | |
| Notes on Performance | 753 |
| <i>Improving SQL Server Performance</i> | <i>753</i> |
| Using Indexes | 753 |
| Choosing Which Columns to Index | 757 |
| Understanding the SQL Server Profiler | 758 |
| Optimizing Automatically with the SQL Server Profiler | 760 |
| Optimizing Your Transact-SQL Queries for Performance | 762 |
| Avoiding Excessive Stored Procedure Recompilation | 765 |
| Avoiding Locks | 766 |
| Designing Your Tables for Optimal Performance | 766 |
| Checking the Size of Database Pages | 767 |
| Understanding Denormalization | 768 |
| Understanding the Dangers of Database Design Slippage | 768 |
| <i>Understanding ADO.NET Performance Issues</i> | |
| <i>and Using Best Practices</i> | <i>769</i> |
| Selecting Data Providers | 769 |
| Writing a Generic Data Access Component | 770 |
| Choosing a Development Mode: Designer vs. Manual | 775 |
| Retrieving Data: DataReader, XmlReader, and DataSet | 775 |
| Updating a Data Source: Command vs. DataSet | 776 |
| Saving Data and Using GetChanges | 777 |
| Retrieving Data with the SELECT Statement | 777 |
| Using Stored Procedures | 777 |
| Using Connection Strings and Pooling | 777 |
| <i>Summary</i> | <i>778</i> |
| Appendix A Relational Databases: | |
| Some Basic Concepts | 779 |
| <i>Understanding Normalization</i> | <i>780</i> |
| First Normal Form (1NF) | 780 |
| Second Normal Form (2NF) | 782 |
| Third Normal Form (3NF) | 784 |
| Boyce-Codd Normal Form (BCNF) | 785 |
| Fourth Normal Form (4NF) | 785 |
| Fifth Normal Form (5NF or PJ/NF) | 786 |
| <i>Introducing Sets, Cursors, and ADO.NET</i> | <i>786</i> |

| | |
|--|-----|
| <i>Using Locking</i> | 788 |
| Isolation Levels | 788 |
| Locking Modes | 790 |
| <i>Summary</i> | 792 |
| <i>Appendix B Commonly Used SQL Statements</i> | 793 |
| <i>Understanding SQL References</i> | 793 |
| Using the SELECT Statement | 796 |
| Using the UPDATE Statement | 802 |
| Using the DELETE Statement | 802 |
| Using the CREATE TABLE Statement | 803 |
| Using the DROP TABLE Statement | 803 |
| Using the TRUNCATE TABLE Statement | 804 |
| Using the INSERT Statement | 805 |
| Using Joins and Aliases | 806 |
| <i>Understanding Views</i> | 807 |
| <i>Using SQL Server's SELECT...FOR XML Clause</i> | 809 |
| <i>Summary</i> | 810 |
| <i>Appendix C ADO.NET Frequently Asked Questions</i> | 811 |
| <i>Summary</i> | 845 |
| <i>Index</i> | 847 |