Contents

INTRODUCTION  XXXIII
   Real Code for Real Programmers  xxxiv
   How This Book Is Organized  xxxv
   Conventions  xxxviii
   About the Web Site  xxxix
   About the Authors  xxxix

ACKNOWLEDGMENTS  XLI

PART 1
THE HYPERTEXT MARKUP LANGUAGE  2

CHAPTER 1
DESIGNING WEB PAGES WITH HTML 4.0  4
   1.1 The HyperText Markup Language  5
   1.2 HTML 4.0 and Other HTML Standards  7
   1.3 Steps to Publish a Document on the Web  9
       Create the Document  9
Contents

10
12
12
13
14
16
17
17
22
25

CHAPTER 2

BLOCK-LEVEL ELEMENTS IN HTML 4.0

2.1 Headings

2.2 Basic Text Elements

Basic Paragraphs
Paragraphs with White Space Preserved
Indented Quotations
Addresses

2.3 Numbered, Bulleted, and Indented Lists

Numbered Lists
Bulleted Lists
Definition Lists

2.4 Tables

The Basic Table Structure
Defining Table Rows
Table Headings and Data Cells
Grouping Table Contents

2.5 Fill-Out Forms

2.6 Miscellaneous Block-Level Elements

2.7 Summary
CHAPTER 3
TEXT-LEVEL ELEMENTS IN HTML 4.0 58
3.1 Physical Character Styles 59
3.2 Logical Character Styles 64
3.3 Specifying Hypertext Links 67
3.4 Embedded Images 70
  Animated GIFs 71
  The IMG Element 71
3.5 Client-Side Image Maps 75
3.6 Embedding Other Objects in Documents 79
  Embedded Applets 80
  Embedded Video, Audio, and Other Formats with Plug-ins 82
  Embedded ActiveX Controls 83
  Embedded Scrolling Text Banners 85
3.7 Controlling Line Breaks 86
3.8 Summary 87

CHAPTER 4
FRAMES 88
4.1 Frame Document Template 90
4.2 Specifying Frame Layout 91
4.3 Specifying the Content of Frame Cells 96
  Examples 98
4.4 Targeting Frame Cells 100
  Predefined Frame Names 103
4.5 Solving Common Frame Problems 103
  Bookmarking Frames 104
  Printing Frames 104
  Updating Multiple Frame Cells Simultaneously 105
  Preventing Your Documents from Being Framed 108
  Creating Empty Frame Cells 109
4.6 Inline Frames 109
4.7 Summary 113
CHAPTER 5
CASCADING STYLE SHEETS  114

5.1 Specifying Style Rules  116
5.2 Using External and Local Style Sheets  118
   External Style Sheets  119
   The STYLE Element and JavaScript Style Sheets  120
   Inline Style Specification  121
5.3 Selectors  121
   HTML Elements  122
   User-Defined Classes  123
   User-Defined IDs  124
   Anchor Pseudoclasses  124
5.4 Cascading: Style Sheet Precedence Rules  125
5.5 Font Properties  126
5.6 Foreground and Background Properties  132
5.7 Text Properties  135
5.8 Properties of the Bounding Box  139
   Margins  140
   Borders  141
   Padding  142
   Bounding Box Display Types  143
5.9 Images and Floating Elements  143
5.10 List Properties  146
5.11 Standard Property Units  147
   Lengths  147
   Colors  147
5.12 Layers  148
   Specifying Layers with the LAYER and ILAYER Elements  149
   Specifying Layers with Style Sheets  153
5.13 Summary  157
PART 2
JAVA PROGRAMMING  158

CHAPTER 6
GETTING STARTED WITH JAVA  160

6.1 Unique Features of Java  162
   Java Is Web-Enabled and Network Savvy  162
   Java Is Cross-Platform  166
   Java Is Simple  168
   Java Is Object Oriented  169
   Java Is Rich with Powerful Standard Libraries  170

6.2 Myths About Java  171
   Java Is Only for the Web  172
   Java Is Cross-Platform  172
   Java Is Simple  174
   Java Is Object Oriented (the One True Way of Programming)  174
   Java Is the Programming Language for All Software Development  175

6.3 Java Versions  175
   Which Version Should You Use?  177
   Whichever Version You Use  177

6.4 Getting Started: Nuts and Bolts  178
   Install Java  178
   Install a Java-Enabled Browser  179
   Bookmark or Install the On-Line Java API  180
   Optional: Get an Integrated Development Environment  180
   Create and Run a Java Program  181

6.5 Some Simple Java Programs  182
   The Basic Hello World Application  182
   Command-Line Arguments  183
   The Basic Hello World (Wide Web) Applet  183
   Applet Customization Parameters  185

6.6 Summary  187
Contents

CHAPTER 7
OBJECT-ORIENTED PROGRAMMING IN JAVA  190
  7.1  Instance Variables  191
  7.2  Methods  194
  7.3  Constructors and the “this” Reference  196
       Static Initialization Blocks  199
  7.4  Destructors  199
  7.5  Overloading  200
  7.6  Public Version in Separate File  204
  7.7  Javadoc  209
       Javadoc Tags  211
       Javadoc Command-Line Arguments  213
  7.8  Inheritance  216
  7.9  Interfaces and Abstract Classes  221
  7.10 Packages, Classpath, and JAR Archives  230
       The CLASSPATH  233
  7.11 Modifiers in Declarations  236
       Visibility Modifiers  236
       Other Modifiers  238
  7.12 Summary  239

CHAPTER 8
BASIC JAVA SYNTAX  242
  8.1  Rules of Syntax  243
  8.2  Primitive Types  245
       Primitive-Type Conversion  247
  8.3  Operators, Conditionals, Iteration  248
       Arithmetic Operators  248
       Conditionals  249
       Loops  255
  8.4  The Math Class  259
       Constants  259
Contents

General-Purpose Methods 259
Trigonometric Methods 261
BigInteger and BigDecimal 261

8.5 Input and Output 263
  Printing to Standard Output 263
  Printing to Standard Error 265
  Reading from Standard Input 265

8.6 Execution of Non-Java Programs 266

8.7 Reference Types 273
  Java Argument-Passing Conventions 275
    The instanceof Operator 275

8.8 Strings 277
  String Methods 278
    Constructors 284

8.9 Arrays 284
  Two-Step Array Allocation 285
    One-Step Array Allocation 286
    Multidimensional Arrays 287

8.10 Vectors 288
  Constructors 289
    Methods 289

8.11 Example: A Simple Binary Tree 291

8.12 Exceptions 296
  Basic Form 296
    Multiple Catch Clauses 299
      The Finally Clause 300
    Thrown Exceptions 300
      Unchecked Exceptions 302

8.13 Summary 303
CHAPTER 9
APPLETS AND BASIC GRAPHICS  304

9.1  What Are Applets?   305
9.2  Creating an Applet  306
     Template for Applets  307
     Template for HTML   307
9.3  An Example Applet   309
     Redrawing Automatically  311
     Reloading Applets During Development  311
9.4  The Applet Life Cycle  312
9.5  Other Applet Methods   314
9.6  The HTML APPLET Element  320
9.7  Reading Applet Parameters  322
     Reading Applet Parameters: An Example   323
9.8  HTML OBJECT Element  326
9.9  The Java Plug-In   328
9.10 Graphical Applications  331
9.11 Graphics Operations  332
     Drawing Operations  333
     Colors and Fonts   336
     Drawing Modes   336
     Coordinates and Clipping Rectangles  337
9.12 Drawing Images   337
     Loading Applet Images from Relative URLs  338
     Loading Applet Images from Absolute URLs   340
     Loading Images in Applications  342
9.13 Preloading Images   344
9.14 Controlling Image Loading: Waiting for Images and Checking Status   348
9.15 Summary   355
CHAPTER 10
JAVA 2: GRAPHICS IN JAVA 2  358
10.1 Getting Started with Java 2D  360
   Useful Graphics2D Methods  363
10.2 Drawing Shapes  366
   Shape Classes  367
10.3 Paint Styles  371
   Paint Classes  372
   Tiled Images as Fill Patterns  375
10.4 Transparent Drawing  378
10.5 Using Local Fonts  381
10.6 Stroke Styles  383
   Stroke Attributes  384
10.7 Coordinate Transformations  390
   Shear Transformations  393
10.8 Other Capabilities of Java 2D  394
10.9 Summary  395

CHAPTER 11
HANDLING MOUSE AND KEYBOARD EVENTS  398
11.1 Handling Events with a Separate Listener  400
   Drawing Circles  402
11.2 Handling Events by Implementing a Listener Interface  404
11.3 Handling Events with Named Inner Classes  406
11.4 Handling Events with Anonymous Inner Classes  407
11.5 The Standard Event Listeners  409
11.6 Behind the Scenes: Low-Level Event Processing  415
11.7 A Spelling-Correcting Textfield  418
11.8 A Whiteboard Class  421
   A Better Whiteboard  423
11.9 Summary  425
Contents

CHAPTER 12
LAYOUT MANAGERS  426

12.1 The FlowLayout Manager  428
  FlowLayout Constructor Options  429
  Other FlowLayout Methods  429

12.2 The BorderLayout Manager  430
  BorderLayout Constructor Options  432
  Other BorderLayout Methods  432

12.3 The GridLayout Manager  433
  GridLayout Constructor Options  434
  Other GridLayout Methods  435

12.4 The CardLayout Manager  436
  CardLayout Constructor Options  440
  Other CardLayout Methods  440

12.5 GridBagLayout  441
  The GridBagConstraints Object  442
  Example  444
  GridBagLayout Constructor Options  448
  Other GridBagLayout Methods  448

12.6 The BoxLayout Manager  449
  BoxLayout Constructor Options  452
  Other BoxLayout Methods  453

12.7 Turning Off the Layout Manager  454

12.8 Effective Use of Layout Managers  455
  Use Nested Containers  456
  Turn Off the Layout Manager for Some Containers  459
  Adjust the Empty Space Around Components  461

12.9 Summary  464
CHAPTER 13

AWT COMPONENTS 466

13.1 The Canvas Class 468
   Creating and Using a Canvas 469
   Example: A Circle Component 469
13.2 The Component Class 472
13.3 Lightweight Components in Java 1.1 479
13.4 The Panel Class 482
   Default LayoutManager: FlowLayout 482
   Creating and Using a Panel 483
   Example: Using a Panel for Grouping 483
13.5 The Container Class 485
13.6 The Applet Class 487
13.7 The ScrollPane Class 487
   Creating and Using a ScrollPane 487
   Example: ScrollPane with 100-Button Panel 488
13.8 The Frame Class 489
   Default LayoutManager: BorderLayout 489
   Creating and Using a Frame 490
   Frame Examples 491
   A Closeable Frame 492
   Menus 493
   Other Useful Frame Methods 495
13.9 Serializing Windows 497
   Writing a Window to Disk 497
   Reading a Window from Disk 497
   Example: A Saveable Frame 498
13.10 The Dialog Class 501
    Creating and Using a Dialog 501
    Example: A Quit Confirmation Dialog 502
13.11 TheFileDialog Class 504
Contents

Example: Displaying Files in a TextArea 504

13.12 The Window Class 507
  Default LayoutManager: BorderLayout 507
  Creating and Using a Window 507

13.13 Handling Events in GUI Controls 508
  Decentralized Event Processing 509
  Centralized Event Processing 511

13.14 The Button Class 512
  Constructors 513
  Example: Applet with Three Buttons 513
  Other Button Methods 514
  Handling Button Events 515

13.15 The Checkbox Class 518
  Constructors 519
  Example: Checked Checkboxes 519
  Other Checkbox Methods 520
  Handling Checkbox Events 521

13.16 Check Box Groups (Radio Buttons) 521
  Constructors 522
  Example: Check Boxes vs. Radio Buttons 522
  Other CheckboxGroup and Checkbox Methods 523
  Handling CheckboxGroup Events 524

13.17 Choice Menus 524
  Constructor 525
  Example: Simple Choices 525
  Other Choice Methods 526
  Handling Choice Events 527

13.18 List Boxes 529
  Constructors 529
  Example: Single and Multiple List Selections 529
  Other List Methods 531
  Handling List Events 533
CHAPTER 14
BASIC SWING 562

14.1  Getting Started with Swing 564
  Differences Between Swing and the AWT 564
Contents

14.2 The JApplet Component  572
14.3 The JFrame Component  574
14.4 The JLabel Component  576
   New Features: Images, Borders, and HTML Content  576
   JLabel Constructors  577
   Useful JLabel Methods  578
14.5 The JButton Component  581
   New Features: Icons, Alignment, and Mnemonics  581
   HTML in Button Labels  582
   JButton Constructors  582
   Useful JButton (AbstractButton) Methods  582
14.6 The JPanel Component  585
   JPanel Constructors  585
   New Feature: Borders  585
   Useful BorderFactory Methods  586
14.7 The JSlider Component  590
   New Features: Tick Marks and Labels  590
   JSlider Constructors  590
   Useful JSlider Methods  591
14.8 The JColorChooser Component  594
   Constructors  595
   Useful JColorChooser Methods  595
14.9 Internal Frames  598
   JInternalFrame Constructors  598
   Useful JInternalFrame Methods  598
14.10 The JOptionPane Component  602
   Useful JOptionPane Methods  602
14.11 The JToolBar Component  607
   JToolBar Constructors  609
   Useful JToolBar Methods  609
14.12 The JEditorPane Component  614
   Following Hypertext Links  615
Contents xvii

JEditorPane Constructors 616
Useful JEditorPane Methods 616
Implementing a Simple Web Browser 618
HTML Support and JavaHelp 621

14.13 Other Simple Swing Components 622
The JCheckBox Component 622
The JRadioButton Component 623
The JTextField Component 625
The JTextArea Component 625
The JFileChooser Component 625

14.14 Summary 626

CHAPTER 15
ADVANCED SWING  628

15.1 Using Custom Data Models and Renderers 630
15.2 JList 631
JList with a Fixed Set of Choices 631
JLists with Changeable Choices 636
JList with Custom Data Model 639
JList with Custom Renderer 646
15.3 JTree 650
Simple JTree 650
JTree Event Handling 654

15.4 JTable 664
Simple JTable 664
Table Data Models 669
Table Cell Renderers 674
Table Event Handling 676
15.5 Swing Component Printing 680
Printing Basics 681
The Role of Double Buffering 683
CHAPTER 16
CONCURRENT PROGRAMMING WITH JAVA THREADS 698
16.1 Starting Threads 700
   Mechanism 1: Put Behavior in a Separate Thread Object 700
   Mechanism 2: Put Behavior in the Driver Class, Which Must Implement Runnable 703
16.2 Race Conditions 706
16.3 Synchronization 709
   Synchronizing a Section of Code 709
   Synchronizing an Entire Method 710
   Common Synchronization Bug 710
16.4 Creating a Multithreaded Method 712
16.5 Thread Methods 717
   Constructors 718
   Constants 719
   Methods 719
   Stopping a Thread 725
16.6 Thread Groups 727
   Constructors 727
   Methods 727
16.7 Multithreaded Graphics and Double Buffering 729
   Redraw Everything in paint 730
   Implement the Dynamic Part as a Separate Component 734
   Have Routines Other Than paint Draw Directly 735
   Override update and Have paint Do Incremental Updating 737
   Use Double Buffering 743
16.8 Animating Images 748
16.9 Timers 753
    Constructor 757
    Other Timer Methods 757
16.10 Summary 759

CHAPTER 17
NETWORK PROGRAMMING 760
17.1 Implementing a Client 762
    Example: A Generic Network Client 765
17.2 Parsing Strings by Using StringTokenizer 768
    The StringTokenizer Class 768
    Constructors 769
    Methods 769
    Example: Interactive Tokenizer 770
17.3 Example: A Client to Verify E-Mail Addresses 771
17.4 Example: A Network Client That Retrieves URLs 774
    A Class to Retrieve a Given URI from a Given Host 775
    A Class to Retrieve a Given URL 777
    UriRetriever Output 778
17.5 The URL Class 779
    Reading from a URL 779
    Other Useful Methods of the URL Class 781
17.6 WebClient: Talking to Web Servers Interactively 783
17.7 Implementing a Server 791
    Example: A Generic Network Server 793
    Connecting NetworkClient and NetworkServer 797
17.8 Example: A Simple HTTP Server 797
    ThreadedEchoServer: Adding Multithreading 802
17.9 RMI: Remote Method Invocation 804
    Steps to Build an RMI Application 805
    A Simple Example 806
Contents

A Realistic Example: A Server for Numeric Integration 811
A Realistic Example of the Four Required Classes 813
Compiling and Running the System for the Realistic Example 817
Enterprise RMI Configuration 819
Compiling and Running the System for an Enterprise RMI Configuration 822
RMI Applet Example 825

17.10 Summary 828

PART 3
SERVER-SIDE PROGRAMMING 830

CHAPTER 18
HTML FORMS 832

18.1 How HTML Forms Transmit Data 833
18.2 The FORM Element 838
18.3 Text Controls 843
   Textfields 843
   Password Fields 845
   Text Areas 846
18.4 Push Buttons 848
   Submit Buttons 849
   Reset Buttons 852
   JavaScript Buttons 853
18.5 Check Boxes and Radio Buttons 854
   Check Boxes 854
   Radio Buttons 855
18.6 Combo Boxes and List Boxes 857
18.7 File Upload Controls 860
18.8 Server-Side Image Maps 862
   IMAGE—Standard Server-Side Image Maps 863
   ISMAP—Alternative Server-Side Image Maps 865
18.9 Hidden Fields 867
18.10 Grouping Controls 868
18.11 Tab Order Control 870
18.12 Summary 871

CHAPTER 19
SERVER-SIDE JAVA: SERVLETS 872

19.1 The Advantages of Servlets Over “Traditional” CGI 874
- Efficient 874
- Convenient 875
- Powerful 875
- Portable 875
- Secure 876
- Inexpensive 876

19.2 Server Installation and Setup 876
- Obtain Servlet and JSP Software 877
- Bookmark or Install the Servlet and JSP API Documentation 878
- Identify the Classes to the Java Compiler 878
- Package Your Classes 879
- Configure the Server 879
- Compile and Install Your Servlets 879
- Invoke Your Servlets 880

19.3 Basic Servlet Structure 880
- A Servlet That Generates Plain Text 882
- A Servlet That Generates HTML 883
- Simple HTML-Building Utilities 885

19.4 The Servlet Life Cycle 887
- The init Method 887
- The service Method 888
- The doGet, doPost, and doXxx Methods 889
- The SingleThreadModel Interface 889
The destroy Method 890

19.5 An Example Using Initialization Parameters 890

19.6 The Client Request: Form Data 893
  Reading Form Data from CGI Programs 893
  Reading Form Data from Servlets 894
  Example: Reading Three Explicit Parameters 894
  Filtering Query Data 897

19.7 The Client Request: HTTP Request Headers 898
  Reading Request Headers from Servlets 899
  Example: Making a Table of All Request Headers 900
  HTTP 1.1 Request Headers 903
  Sending Compressed Web Pages 906

19.8 The Servlet Equivalent of the Standard CGI Variables 908

19.9 The Server Response: HTTP Status Codes 911
  Specifying Status Codes 912
  HTTP 1.1 Status Codes 913
  A Front End to Various Search Engines 919

19.10 The Server Response: HTTP Response Headers 924
  Setting Response Headers from Servlets 924
  HTTP 1.1 Response Headers 926
  Persistent Servlet State and Auto-Reloading Pages 932

19.11 Cookies 941
  Benefits of Cookies 942
  Some Problems with Cookies 943
  The Servlet Cookie API 944
  Examples of Setting and Reading Cookies 947
  Basic Cookie Utilities 951
  Finding Cookies with Specified Names 951
  Creating Long-Lived Cookies 953

19.12 Session Tracking 953
  The Need for Session Tracking 953
The Session Tracking API  955
Terminating Sessions  959
A Servlet Showing Per-Client Access Counts  960
19.13 Summary  962

CHAPTER 20
JAVASERVER PAGES  964
20.1 JSP Overview  965
20.2 Advantages of JSP  967
   Versus Active Server Pages (ASP) or ColdFusion  967
   Versus PHP  967
   Versus Pure Servlets  967
   Versus Server-Side Includes (SSI)  968
   Versus JavaScript  968
20.3 JSP Scripting Elements  968
   Expressions  969
   Scriptlets  972
   Declarations  975
   Predefined Variables  977
20.4 The JSP page Directive  979
   The import Attribute  979
   The contentType Attribute  982
   The isThreadSafe Attribute  983
   The session Attribute  984
   The buffer Attribute  984
   The autoflush Attribute  984
   The extends Attribute  984
   The info Attribute  985
   The errorPage Attribute  985
   The isErrorPage Attribute  985
   The language Attribute  985
Contents

XML Syntax for Directives 986

20.5 Including Files and Applets in JSP Documents 986
   The include Directive: Including Files at Page Translation Time 987
   Including Files at Request Time 989
   Including Applets for the Java Plug-In 992
   The jsp:fallback Element 995

20.6 Using JavaBeans with JSP 999
   Basic Bean Use 1001
   Example: StringBean 1003
   Setting Bean Properties 1005
   Sharing Beans 1011

20.7 Defining Custom JSP Tags 1015
   The Components That Make Up a Tag Library 1015
   Defining a Basic Tag 1019
   Assigning Attributes to Tags 1022
   Including the Tag Body 1026
   Optionally Including the Tag Body 1031
   Manipulating the Tag Body 1034
   Including or Manipulating the Tag Body Multiple Times 1038
   Using Nested Tags 1042

20.8 Integrating Servlets and JSP 1049
   Forwarding Requests 1049
   Example: An On-Line Travel Agent 1053
   Forwarding Requests From JSP Pages 1062

20.9 Summary 1062

CHAPTER 21
USING APPLETS AS FRONT ENDS TO
SERVER-SIDE PROGRAMS 1064

21.1 Sending Data with GET and Displaying the Resultant Page 1066

21.2 A Multisystem Search Engine Front End 1067
21.3 Using GET and Processing the Results Directly
(HTTP Tunneling) 1071
  Reading Binary or ASCII Data 1072
  Reading Serialized Data Structures 1073
21.4 A Query Viewer That Uses Object Serialization
and HTTP Tunneling 1075
21.5 Using POST and Processing the Results Directly
(HTTP Tunneling) 1083
21.6 An Applet That Sends POST Data 1086
21.7 Bypassing the HTTP Server 1091
21.8 Summary 1091

CHAPTER 22
JDBC 1092
22.1 Basic Steps in Using JDBC 1094
  Load the Driver 1094
  Define the Connection URL 1095
  Establish the Connection 1096
  Create a Statement 1096
  Execute a Query 1097
  Process the Results 1097
  Close the Connection 1098
22.2 Basic JDBC Example 1098
22.3 Some JDBC Utilities 1105
22.4 Applying the Database Utilities 1114
22.5 An Interactive Query Viewer 1120
  Query Viewer Code 1122
22.6 Prepared Statements (Precompiled Queries) 1127
22.7 Summary 1131
CHAPTER 23
XML PROCESSING WITH JAVA  1132
  23.1 Parsing XML Documents with DOM Level 2  1134
     Installation and Setup  1134
     Parsing  1135
  23.2 DOM Example: Representing an XML Document as a JTree  1137
  23.3 Parsing XML Documents with SAX 2.0  1149
     Installation and Setup  1149
     Parsing  1150
  23.4 SAX Example 1: Printing the Outline of an XML Document  1152
  23.5 SAX Example 2: Counting Book Orders  1158
  23.6 Transforming XML with XSLT  1164
     Installation and Setup  1164
     Translating  1165
  23.7 XSLT Example 1: XSLT Document Editor  1169
  23.8 XSLT Example 2: Custom JSP Tag  1179
  23.9 Summary  1187

PART 4
JAVASCRIPT  1188

CHAPTER 24
JAVASCRIPT: ADDING DYNAMIC CONTENT TO WEB PAGES  1190
  24.1 Generating HTML Dynamically  1193
     Compatibility with Multiple Browsers  1197
  24.2 Monitoring User Events  1198
  24.3 Mastering JavaScript Syntax  1200
     Dynamic Typing  1201
     Function Declarations  1201
     Objects and Classes  1202
     Arrays  1208
24.4 Using JavaScript to Customize Web Pages 1209
   Adjusting to the Browser Window Size 1209
   Determining Whether Plug-Ins Are Available 1213
24.5 Using JavaScript to Make Pages Dynamic 1215
   Modifying Images Dynamically 1215
   Moving Layers 1223
24.6 Using JavaScript to Validate HTML Forms 1228
   Checking Values Individually 1229
   Checking Values When Form Is Submitted 1231
24.7 Using JavaScript to Store and Examine Cookies 1237
24.8 Using JavaScript to Interact with Frames 1242
   Directing a Particular Frame to Display a URL 1242
   Giving a Frame the Input Focus 1246
24.9 Accessing Java from JavaScript 1246
   Calling Java Methods Directly 1247
   Using Applets to Perform Operations for JavaScript 1248
   Controlling Applets from JavaScript 1252
24.10 Accessing JavaScript from Java 1256
   Example: Matching Applet Background with Web Page 1259
   Example: An Applet That Controls HTML Form Values 1260
   Methods in the JSObject Class 1271
24.11 Summary 1272

CHAPTER 25
JAVASCRIPT QUICK REFERENCE 1274
25.1 The Array Object 1275
   Constructors 1275
   Properties 1276
   Methods 1276
   Event Handlers 1279
Contents

25.2 The Button Object 1279
    Properties 1279
    Methods 1280
    Event Handlers 1280

25.3 The Checkbox Object 1281
    Properties 1281
    Methods 1282
    Event Handlers 1282

25.4 The Date Object 1283
    Constructors 1283
    Properties 1283
    Methods 1283
    Event Handlers 1286

25.5 The Document Object 1286
    Properties 1286
    Methods 1288
    Event Handlers 1289

25.6 The Element Object 1289
    Properties 1289
    Methods 1290
    Event Handlers 1291

25.7 The FileUpload Object 1291
    Properties 1292
    Methods 1292
    Event Handlers 1292

25.8 The Form Object 1293
    Properties 1293
    Methods 1293
    Event Handlers 1294

25.9 The Function Object 1294
    Constructor 1294
Contents

25.18 The Math Object 1307
   Properties 1307
   Methods 1308
   Event Handlers 1310
25.19 The MimeType Object 1310
   Properties 1311
   Methods 1311
   Event Handlers 1311
25.20 The Navigator Object 1311
   Properties 1311
   Methods 1314
   Event Handlers 1314
25.21 The Number Object 1314
   Constructor 1315
   Properties 1315
   Methods 1315
   Event Handlers 1317
25.22 The Object Object 1317
   Constructors 1318
   Properties 1318
   Methods 1318
   Event Handlers 1319
25.23 The Option Object 1319
   Constructors 1319
   Properties 1319
   Methods 1320
   Event Handlers 1320
25.24 The Password Object 1320
   Properties 1320
   Methods 1321
   Event Handlers 1321
25.25 The Plugin Object 1322
  Properties 1322
  Methods 1322
  Event Handlers 1322
25.26 The Radio Object 1323
  Properties 1323
  Methods 1323
  Event Handlers 1324
25.27 The RegExp Object 1324
  Constructors 1324
  Properties 1326
  Methods 1327
  Event Handlers 1328
  Special Patterns in Regular Expressions 1328
25.28 The Reset Object 1329
  Properties 1330
  Methods 1330
  Event Handlers 1330
25.29 The Screen Object 1331
  Properties 1331
  Methods 1332
  Event Handlers 1332
25.30 The Select Object 1332
  Properties 1333
  Methods 1334
  Event Handlers 1334
25.31 The String Object 1335
  Constructor 1335
  Properties 1335
  Methods 1335
  Event Handlers 1340
Contents

25.32 The Submit Object 1340
  Properties 1340
  Methods 1341
  Event Handlers 1341

25.33 The Text Object 1342
  Properties 1342
  Methods 1342
  Event Handlers 1343

25.34 The Textarea Object 1343
  Properties 1344
  Methods 1344
  Event Handlers 1344

25.35 The Window Object 1345
  Properties 1345
  Methods 1349
  Event Handlers 1354
  An Example of the open Method 1355

25.36 Summary 1358

INDEX 1360
In late 1995, Marty Hall proposed a new course for the part-time graduate program in Computer Science at the Johns Hopkins University. The idea was to bring together the major Web-related topics in a single course dubbed “Distributed Development on the World Wide Web,” with Java technology as a unifying theme. Students would look at HTML, Java, HTTP, CGI programming, and JavaScript, with lots of hands-on projects and no exams. Little did Marty know what he was getting himself into. By the time the first section was offered in the summer of 1996, the Java tidal wave had swept through the university and the companies that the students represented. Shortly after enrollment opened, the class was filled. There were more students on the waiting list than in the course. Marty got frantic phone calls from students insisting that they absolutely had to be in the course. Several local companies called, asking for on-site courses. What fun!

However, when Marty went shopping for texts over the next semester or two, he got a rude surprise. Despite the availability of good books in most of the individual areas he wanted to cover, Marty found that he needed three, four, or even five separate books to get good coverage of the overall material. Similarly, for his day job, Marty was constantly switching back and forth among the best of the huge stack of books he had accumulated and various on-line references. Surely there was a better way. Shouldn’t it be possible to fit 85 percent of what professional programmers use in about 35 percent of the space, and get it all in one book?

That was the genesis of the first edition of Core Web Programming. The book was very popular, but the industry has been rapidly moving since the book’s release. Browsers moved from HTML 3.2 to 4.0. The Java 2 platform was released, providing greatly improved performance and graphics libraries suitable for commercial-quality
Introduction

applications. JSP 1.0 came along, resulting in an explosion of interest in both servlets and JSP as an alternative to CGI and to proprietary solutions like ASP and ColdFusion. XML burst upon the scene. The server equalled or even surpassed the desktop as the biggest application area for the Java programming language.

Wow. And demand has only been growing since then. Although readers were clamoring for a new edition of the book, it was just too much for Marty to handle alone. Enter Larry Brown, with broad development and teaching experience in Java and Web technologies, and with particular expertise in the Java Foundation Classes, multithreaded programming, RMI, and XML processing with Java. Larry teamed up with Marty to totally update the existing material to HTML 4, CSS/1, HTTP 1.1, and the Java 2 platform; to replace the CGI sections with chapters on servlets 2.2 and JSP 1.1; and to add completely new sections on Swing, Java 2D, and XML processing with JAXP, DOM Level 2, SAX 2.0, and XSLT. They even got a little bit of sleep along the way.

We—Marty and Larry—hope you find the result enjoyable and useful!

Real Code for Real Programmers

This book is aimed at serious software developers. If you are looking for a book that shows you how to use a browser, lists the current hottest Web sites, and pontificates about how Web-enabled applications will revolutionize your business, you’ve come to the wrong place. If you’re already a programmer of some sort and want to get started with HTML, XML, Java applets, desktop applications in Java, servlets, JavaServer Pages, and JavaScript as quickly as possible, this is the book for you. We illustrate the most important approaches and warn you of the most common pitfalls. To do so, we include plenty of working code: over 250 complete Java classes, for instance. We try to give detailed examples of the most important and frequently used features, summarize the lesser-used ones, and refer you to the API (available on-line) for a few of the rarely used ones.
A word of caution, however. Nobody becomes a great developer just by reading. You have to write some real code too. The more, the better. In each chapter, we suggest that you start by making a simple program or a small variation of one of the examples given, then strike off on your own with a more significant project. Skim the sections you don’t plan on using right away, then come back when you are ready to try them out.

If you do this, you should quickly develop the confidence to handle the real-world problems that brought you here in the first place. You should be able to balance the demand for the latest features in Web pages with the need for multiplatform support. You should be comfortable with frames, style sheets, and layered HTML. You should be able to make portable stand-alone graphical applications. You should have no qualms about developing Web interfaces to your corporate database through JDBC. You should be able to connect these applications to remote systems over the network. You should understand how to easily distribute computation among multiple threads, or even spin it off to separate systems by using RMI. You should understand HTTP 1.1 well enough to use its capabilities to enhance the effectiveness of your pages. You should be able to spin off complex server-side behaviors into JavaBeans components or custom JSP tag libraries. You should be able to use JavaScript to validate HTML forms or to animate Web pages. You should get a raise. A big one, preferably.

How This Book Is Organized

This book is divided into four parts: HTML, Java programming, server-side programming, and JavaScript.

Part 1: The HyperText Markup Language
Web pages are created with HTML, the HyperText Markup Language. HTML lets you mix regular text with special tags that describe the content, layout, or appearance of the text. These tags are then used by Web browsers like Netscape Navigator or Microsoft Internet Explorer to format the page. This first part of the book covers the following topics in HTML.

- HTML 4.01. Full coverage of all the elements in the latest official HTML standard. Hypertext links, fonts, images, tables, client-side image maps, and more.
- Major Netscape and Internet Explorer extensions. Forwarding pages, using custom colors and font faces, embedding audio, video, and ActiveX components.
Introduction

- **Frames.** Dividing the screen into rectangular regions, each associated with a separate HTML document. Borderless frames. Floating frames. Targeting frame cells from hypertext links.
- **Cascading style sheets.** Level-one style sheets for customizing fonts, colors, images, text formatting, indentation, lists, and more.

**Part 2: Java Programming**
Java is a powerful general-purpose programming language that can be used to create stand-alone programs as well as ones that are embedded in Web pages. The following Java topics are covered.

- **Unique features of Java.** What’s different about Java? The truth about Java myths and hype.
- **Object-oriented programming in Java.** Variables, methods, constructors, overloading, and interfaces. Modifiers in class declarations. Packages, the CLASSPATH, and JAR files.
- **Java syntax.** Primitive types, operators, strings, vectors, arrays, input/output and the Math class.
- **Java 2D.** Creating professional, high-quality 2D graphics. Creating custom shapes, tiling images, using local fonts, creating transparent shapes, and transforming coordinates.
- **Mouse and keyboard events.** Processing events. Event types, event listeners, and low-level event handlers. Inner classes. Anonymous classes.
- **AWT components.** Canvas, Panel, Applet,ScrollPane, Frame, Dialog, FileDialog, and Window. Component and Container. Buttons, check boxes, radio buttons, combo boxes, list boxes, textfields, text areas, labels, scrollbars, and pop-up menus. Saving and loading windows with object serialization.
- **Advanced Swing.** JList, JTree, and JTable. Using custom data models and renderers. Printing Swing components. Updating Swing components in a thread-safe manner.
• **Multithreaded programming.** Threads in separate or existing objects. Synchronizing access to shared resources. Grouping threads. Multithreaded graphics and double buffering. Animating images. Controlling timers.

• **Network programming.** Clients and servers using sockets. The `URL` class. Implementing a generic network server. Creating a simple HTTP server. Invoking distributed objects with RMI.

**Part 3: Server-Side Programming**

Programs that run on a Web server can generate dynamic content based on client data. Servlets are Java technology’s answer to CGI programming and JSP is Java’s answer to Active Server Pages or ColdFusion. The following server-side topics are discussed.


• **JavaServer Pages (JSP).** The benefits of JSP. JSP expressions, scriptlets, and declarations. Using JavaBeans components with JSP. Creating custom JSP tag libraries. Combining servlets and JSP.

• **Using applets as servlet front ends.** Sending GET and POST data. HTTP tunneling. Using object serialization to exchange high-level data structures between applets and servlets. Bypassing the HTTP server altogether.

• **Java Database Connectivity (JDBC).** The seven basic steps in connecting to databases. Some utilities that simplify JDBC usage. Formatting a database result as plain text or HTML. An interactive graphical query viewer. Precompiled queries.

• **XML processing with Java.** Representing an entire XML document by using the Document Object Model (DOM) Level 2. Responding to individual XML parsing events with the Simple API for XML Parsing (SAX) 2.0. Transforming XML with XSLT. Hiding vendor-specific details with the Java API for XML Processing (JAXP).

**Part 4: JavaScript**

JavaScript is a scripting language that can be embedded in Web pages and interpreted as the pages are loaded. The final part covers the following JavaScript topics.
Introduction

- **JavaScript syntax.** Fields, methods, functions, strings, objects, arrays, and regular expressions.
- **Customizing Web pages.** Adapting to different browsers, JavaScript releases, and screen sizes.
- **Making pages dynamic.** Animating images. Manipulating layers. Responding to user events.
- **Validating HTML forms.** Checking form entries as they are changed. Checking data when form is submitted.
- **Handling cookies.** Reading and setting values. The `Cookie` object.
- **Controlling frames.** Sending results to specific frames. Preventing documents from being framed. Updating multiple frame cells. Giving frame cells the focus automatically.
- **Integrating Java and JavaScript.** LiveConnect and the `JSObject` class.
- **JavaScript quick reference.** Major classes in JavaScript 1.2. All fields, methods, and event handlers. `Document, Window, Form, Element, String, Math, RegExp`, and so forth.

Conventions

Throughout the book, concrete programming constructs or program output is presented in a monospaced font. For example, when abstractly describing Java programs that can be embedded in Web pages, we refer to “applets,” but when we refer to `Applet` we are talking about the specific Java class from which all applets are derived.

User input is indicated in boldface, and command-line prompts are either generic (`Prompt>`) or indicate the operating system to which they apply (`Unix>`). For instance, the following indicates that “Some Output” is the result when “java SomeProgram” is executed.

```
Prompt> java SomeProgram
Some Output
```

Important standard techniques are indicated by specially marked entries, as in the following example.

**Core Approach**

*Pay particular attention to items in “Core Approach” sections. They indicate techniques that should always or almost always be used.*

Notes and warnings are called out in a similar manner.
About the Web Site

The book has a companion Web site at

http://www.corewebprogramming.com/

This free site includes:

- Documented source code for all examples shown in the book; this code can be downloaded for unrestricted use.
- On-line versions of all HTML pages, Java applets, and JavaScript examples.
- Links to all URLs mentioned in the text of the book.
- Information on book discounts.
- Reports on Java short courses.
- Book additions, updates, and news.
- A free Ronco combination paring knife and e-commerce tool. OK, maybe not.

About the Authors

Marty Hall is a Senior Computer Scientist in the Research and Technology Development Center at the Johns Hopkins University Applied Physics Lab, where he specializes in the application of Java and Web technology to customer problems. He also teaches Java and Web programming in the Johns Hopkins part-time graduate program in Computer Science, where he directs the Distributed Computing and Web Technology concentration areas. When he gets a chance, he also teaches industry short courses on servlets, JavaServer Pages, and other Java technology areas. He is the author of Core Servlets and JavaServer Pages and the first edition of Core Web Programming. Marty can be reached at the following address:

Research and Technology Development Center
The Johns Hopkins University Applied Physics Laboratory
11100 Johns Hopkins Road
Laurel, MD 20723
hall@corewebprogramming.com

Larry Brown is a Senior Network Engineer at the Naval Surface Warfare Center, Carderock Division, where he specializes in developing and deploying network and Web solutions in an enterprise environment. He is also a Computer Science faculty member at the Johns Hopkins University, where he teaches server-side program-
Introduction

ming, distributed Web programming, and Java user interface development for the part-time graduate program in Computer Science. Larry can be reached at the following address:

Naval Surface Warfare Center, Carderock Division
9500 MacArthur Boulevard
West Bethesda, MD 20817
brown@corewebprogramming.com
Many people have helped us out with this book. Without their assistance, we would still be on the fourth chapter. Those that provided valuable technical feedback, pointed out errors, and gave useful suggestions include Don Aldridge, Chris Bennett, Camille Bell, Pete Clark, Maria Dimalanta, Nguyen-Khoa Duy, Denise Evans, Amy Karlson, Paul McNamee, Toddi Norum, Walter Pasquini, Rich Slywczak, Bob Tinker, and Kim Topley. This book would not be a success without their contributions. Mary Lou “Eagle Eye” Nohr spotted our errant commas, awkward sentences, typographical errors, and grammatical inconsistencies. She improved the result immensely. We hope that we learned from her advice. Vanessa Moore produced the final version; she did a great job despite our last-minute changes and crazy travel schedules. Ralph Semmel and Julie Wessel both provided supportive work environments and flexible schedules. Greg Doench of Prentice Hall believed in a second edition and encouraged us to write the book. Thanks to all.

Most of all, I–Marty–thank B.J., Lindsay, and Nathan for their patience with my long hours and funny schedule. I–Larry–thank Lee for her loving support and patience while I disappeared to the computer room every weekend.

God has blessed us both with great families.