

# Contents

<b>About the Authors</b>	<b>v</b>
<b>Acknowledgments</b>	<b>vii</b>
<b>Introduction</b>	<b>xxv</b>
<b>The Right Way to Do Web Development Approach</b>	<b>xxv</b>
<b>How This Book Is Structured</b>	<b>xxvi</b>
<b>Conventions</b>	<b>xxix</b>
<b>Source Code</b>	<b>xxx</b>
<b>Errata</b>	<b>xxx</b>
<b>p2p.wrox.com</b>	<b>xxx</b>
<b>Part I: JSP Fundamentals</b>	<b>1</b>
<b>Chapter 1: Getting Started with JavaServer Pages</b>	<b>3</b>
<b>Creating Applications for the Internet</b>	<b>4</b>
Limitations of the basic Web server model	5
Dynamic HTML generation via CGI	5
Shortcomings of CGI	7
Improving Java-based CGI: servlets	7
<b>Summary</b>	<b>28</b>
<b>Exercises</b>	<b>29</b>
<b>Chapter 2: JSP Basics 1: Dynamic Page Creation for Data Presentation</b>	<b>31</b>
<b>The Anatomy of a JSP Page</b>	<b>31</b>
Directives	32
XML-compatible syntax	33
Template data	33
Action	34
Scripting elements	35
Handling HTML form submission with JSP	36
Web site personalization	60
<b>Summary</b>	<b>70</b>
<b>Exercises</b>	<b>71</b>

<b>Chapter 3: JSP Basics 2: Generalized Templating and Server Scripting</b>	<b>73</b>
<b>Scripting Elements for Java Code Embedding</b>	<b>74</b>
Scripting elements	75
<b>Creating a Simple Web Storefront</b>	<b>80</b>
Attaching attributes to implicit objects	86
Rendering the list of categories	88
Rendering the list of products in a given category	89
<b>Adding a Shopping Cart to a Catalog</b>	<b>90</b>
<b>Creating the Shopping Cart</b>	<b>95</b>
Decoding incoming request parameters	97
Rendering order information	98
Rendering the Return to Shopping hyperlink	98
Shopping cart limitations	99
Overcoming the shopping cart limitations	100
Sessions and JSPs	107
Rendering the shopping cart using a session attribute	111
<b>Summary</b>	<b>114</b>
<b>Exercises</b>	<b>115</b>
<b>Chapter 4: CSS, JavaScript, VBScript, and JSP</b>	<b>117</b>
<b>Code Elements That Execute on the Client Side</b>	<b>118</b>
Cascading Stylesheets	119
JavaScript	120
VBScript	121
<b>User Preference Implementation</b>	<b>121</b>
<b>Creating a User-Customizable DHTML Menu</b>	<b>132</b>
<b>Summary</b>	<b>143</b>
<b>Exercises</b>	<b>144</b>
<b>Chapter 5: JSP and EL</b>	<b>145</b>
<b>EL and Its Vital Role in JSP</b>	<b>146</b>
<b>EL Named Variables</b>	<b>146</b>
<b>Applying EL</b>	<b>148</b>
Using EL expressions inline with template data	148
Using EL expressions in attribute values	148
<b>Coercion: Automatic Type Conversion</b>	<b>157</b>
Boxing and unboxing	157
Coercion to a string	157
Coercion to a number	158

---

Coercion to a character	158
Best attempt to “do the right thing” without error	158
<b>Accessing Object Properties and Collections</b>	<b>162</b>
<b>Implicit EL Objects in JSP 2.0</b>	<b>167</b>
<b>User-Supplied Functions within EL</b>	<b>173</b>
Namespace and EL functions	173
Static methods of a Java class	173
<b>Summary</b>	<b>177</b>
<b>Exercises</b>	<b>178</b>
<b>Chapter 6: JSP Tag Libraries and JSTL</b>	<b>179</b>
<hr/>	
<b>The Vital Role of JSP Tag Libraries</b>	<b>180</b>
<b>The JSP Standard Tag Library</b>	<b>181</b>
<b>Anatomy of a Tag Library</b>	<b>182</b>
The Tag Library Descriptor	182
The taglib map in the web.xml deployment descriptor	183
Locating JSTL and understanding tag library packaging	184
JSTL tags	184
<b>Summary</b>	<b>210</b>
<b>Exercises</b>	<b>211</b>
<b>Chapter 7: JSP Directives</b>	<b>213</b>
<hr/>	
<b>Directive Basics</b>	<b>214</b>
Directives as instructions for the container	214
Alternative XML syntax for directives	214
Available JSP directives	215
<b>The page Directive</b>	<b>217</b>
The language attribute	217
The extends attribute	217
The import attribute	218
The session attribute	218
The info attribute	219
The isELIgnored attribute	219
The isErrorPage attribute	219
The errorPage attribute	220
The contentType attribute	220
<b>The taglib Directive</b>	<b>223</b>
Two general usage forms for the taglib directive	224
Attributes of the taglib directive	225
How It Works	227

# Contents

---

<b>The include Directive</b>	<b>231</b>
The include performed at translation time	231
<b>Summary</b>	<b>239</b>
<b>Exercises</b>	<b>239</b>
<b>Chapter 8: JSP Standard Actions</b>	<b>241</b>
<hr/>	
<b>JSP Standard Actions Are Built-in Tags</b>	<b>242</b>
<b>Actions for Working with JavaBeans</b>	<b>242</b>
The <jsp:useBean> standard action	243
How <jsp:useBean> and JSTL <c:set> Differ	258
<b>Including JSP Output via &lt;jsp:include&gt;</b>	<b>259</b>
<b>Transferring Control Between JSPs</b>	<b>264</b>
The <jsp:forward> standard action	264
<b>Specifying Parameters for Other Actions</b>	<b>265</b>
The <jsp:param> standard action	265
<b>Working with Plug-ins</b>	<b>266</b>
The <jsp:plugin> standard action	267
The <jsp:params> standard action	267
The <jsp:fallback> standard action	268
<b>Standard Actions Specific to Tag Files</b>	<b>273</b>
<b>Summary</b>	<b>274</b>
<b>Exercises</b>	<b>274</b>
<b>Chapter 9: JSP and JavaBeans</b>	<b>275</b>
<hr/>	
<b>Anatomy of a JavaBean</b>	<b>276</b>
JavaBean Properties	277
JavaBean methods	280
Common JavaBean packaging	281
<b>How JavaBeans and EJBs Differ</b>	<b>301</b>
<b>Summary</b>	<b>301</b>
<b>Exercises</b>	<b>302</b>
<b>Chapter 10: Error Handling</b>	<b>303</b>
<hr/>	
<b>Understanding the Origin of Errors</b>	<b>303</b>
Errors in Java language coding	304
Errors in JSP directives and actions	309
JSTL errors and EL errors	326
User data-input errors	327
Errors found in JSP template data	327

---

<b>Summary</b>	<b>328</b>
<b>Exercises</b>	<b>328</b>
<b>Chapter 11: Building Your Own Custom JSP Tag Library</b>	<b>329</b>
<b>What Is a Tag File?</b>	<b>329</b>
<b>A Simple Tag File: Displaying Today's Date</b>	<b>330</b>
<b>Advantages of Tag Files</b>	<b>331</b>
Code reuse	331
Hiding complexity	331
Separation of concerns	331
Simplicity	332
Flexible packaging	332
<b>Developing Tag Files</b>	<b>332</b>
Scope and implicit objects	332
Using directives in tag files	335
Body processing	341
Attributes	345
<b>Packaging Tag Files</b>	<b>355</b>
Java custom actions versus tag file custom actions	358
<b>Summary</b>	<b>359</b>
<b>Exercises</b>	<b>359</b>
<b>Chapter 12: Advanced Dynamic Web Content Generation</b>	<b>361</b>
<b>Data Validation in Web Applications</b>	<b>362</b>
Server-side validation and efficient usage of resources	363
Client-side data validation	364
The need for server-side validation	365
Common client-side validation scenarios	365
Operation of client-side validation	365
Dynamic generation of client-side JavaScript code	379
Dynamic generation of XML using JSP	386
<b>Summary</b>	<b>392</b>
<b>Exercises</b>	<b>393</b>
<b>Chapter 13: Internationalization and Localized Content</b>	<b>395</b>
<b>About Internationalization-Ready Applications</b>	<b>396</b>
<b>Internationalization and Localization</b>	<b>396</b>
The unique i18n requirements of a Web-based JSP application	397

# Contents

---

<b>Building on the Java Platform's i18n Capabilities</b>	<b>399</b>
The concept of a locale	399
Maintaining locale information	400
<b>Summary</b>	<b>433</b>
<b>Exercises</b>	<b>434</b>
<b>Chapter 14: JSP Debugging Techniques</b>	<b>435</b>
<hr/>	
<b>The Science of Debugging</b>	<b>436</b>
<b>Catching Bugs at Compile Time</b>	<b>437</b>
<b>Using a Debugging System versus a Production System</b>	<b>441</b>
<b>Using System.out.println() to Instrument Code</b>	<b>442</b>
<b>Using a Logging System</b>	<b>445</b>
Logging through the servlet container	445
Logging with the JDK logger	447
Logging with Log4j	454
Logging with tag libraries	460
<b>Debugging with Tools</b>	<b>464</b>
Setting breakpoints	464
Examining variables and setting watchpoints	467
Stepping through code	468
Remote debugging	468
<b>Debugging Code in Production Environments</b>	<b>469</b>
Debugging when the JSP is precompiled	470
Debugging under load	470
Adding contextual information to log files	471
Adding log filters	474
<b>Finding the Intermittent Problem</b>	<b>474</b>
Adding information as response comments	474
Snoop application	475
<b>Avoiding Concurrency Issues</b>	<b>479</b>
<b>Summary</b>	<b>479</b>
<b>Exercises</b>	<b>479</b>
<b>Part II: JSP and Modern Web Server Software Development</b>	<b>481</b>
<hr/>	
<b>Chapter 15: JSPs and Servlets</b>	<b>483</b>
<hr/>	
<b>A JSP Is a Servlet</b>	<b>484</b>
Anatomy of a servlet	488
The lifecycle of a servlet	492

---

Describing servlets to containers in the deployment descriptor	495
The servlet declaration	497
Using servlets when JSP is available	503
Specifying initialization parameters	504
Accessing initialization parameters within the ControllerServlet	506
Custom forwarding of incoming requests via the controller servlet	508
The forwarding targets	509
<b>Summary</b>	<b>510</b>
<b>Exercises</b>	<b>510</b>
<b>Chapter 16: The Role of JSP in the Wider Context: Web Applications</b>	<b>511</b>
<hr/>	
<b>What Is a Web Application?</b>	<b>512</b>
<b>Directory Structure for a Web Application</b>	<b>512</b>
<b>The Deployment Descriptor</b>	<b>513</b>
Deployment descriptor elements for a JSP developer	513
<b>Using a Web Archive</b>	<b>523</b>
<b>Development and Deployment Strategies</b>	<b>523</b>
<b>Packaging and Deploying for Tomcat</b>	<b>524</b>
Introduction to Web application security	528
<b>Summary</b>	<b>529</b>
<b>Exercises</b>	<b>530</b>
<b>Chapter 17: Model View Controller</b>	<b>533</b>
<hr/>	
<b>What Is MVC?</b>	<b>534</b>
<b>MVC and Web Applications</b>	<b>534</b>
<b>Summary</b>	<b>546</b>
<b>Exercise</b>	<b>547</b>
<b>Chapter 18: Web Frameworks</b>	<b>549</b>
<hr/>	
<b>What Is a Framework?</b>	<b>549</b>
<b>Why Frameworks Are Good</b>	<b>550</b>
<b>Types of Frameworks</b>	<b>551</b>
Application frameworks	551
Persistence frameworks	552
Utility frameworks	553
<b>WebWork</b>	<b>554</b>
Components of WebWork	554

# Contents

---

<b>The Spring Framework</b>	<b>567</b>
Main components of the Spring MVC	567
An example using Spring	569
<b>Summary</b>	<b>589</b>
<b>Exercises</b>	<b>590</b>
<b>Chapter 19: Struts Framework</b>	<b>591</b>
<hr/>	
<b>Introducing Struts</b>	<b>592</b>
A Struts walkthrough	593
Controller: actions and forms	594
Model	603
View	604
Example application	624
<b>Summary</b>	<b>641</b>
<b>Exercises</b>	<b>641</b>
<b>Chapter 20: Layout Management with Tiles</b>	<b>643</b>
<hr/>	
<b>Introduction to Tiles</b>	<b>643</b>
What is the Tiles framework?	644
Tiles terminology	645
Installing Tiles	646
<b>Tile Scope</b>	<b>650</b>
<b>Definitions</b>	<b>650</b>
<b>Tiles Tags</b>	<b>652</b>
insert	653
definition	654
put	654
putList	655
add	656
get	656
getAsString	656
useAttribute	657
importAttribute	657
initComponentDefinitions	658
<b>Passing Parameters to Tiles</b>	<b>658</b>
<b>Advanced Tiles</b>	<b>659</b>
Definitions and inheritance	659
Nesting tiles	664

---

<b>Tiles and Struts</b>	<b>671</b>
Configuring Tiles with Struts	671
Tiles definitions as action forwards	671
Passing values from Struts to Tiles	673
<b>Summary</b>	<b>678</b>
<b>Exercises</b>	<b>679</b>
<b>Chapter 21: JavaServer Faces</b>	<b>681</b>
<hr/>	
<b>Configuring a JSF Project</b>	<b>681</b>
<b>Getting Started with JSF</b>	<b>683</b>
Using backing beans	685
Managed beans	687
Controlling navigation	687
Preventing direct access to Faces JSP files	690
<b>JSF Lifecycle</b>	<b>690</b>
Restore view	692
Apply request values	692
Process validations	692
Update model values	692
Invoke application	693
Render response	693
<b>Validating Data</b>	<b>693</b>
Using standard validators	697
Creating your own validation	697
<b>Converting Data</b>	<b>705</b>
Standard converters	711
Implementing the Converter interface	713
<b>Handling Events with Listeners</b>	<b>715</b>
<b>JSF versus Struts</b>	<b>718</b>
<b>Summary</b>	<b>719</b>
<b>Exercises</b>	<b>719</b>
<b>Chapter 22: JSP in J2EE</b>	<b>721</b>
<hr/>	
<b>Overview of J2EE</b>	<b>721</b>
What is J2EE?	722
What does a container do?	724
<b>J2EE Technologies</b>	<b>724</b>
J2EE APIs	725
J2EE platform architecture	727
JSP development scenarios	728

# Contents

---

<b>Using JSP with J2EE Components</b>	<b>729</b>
Servlets	730
JNDI	735
JDBC	738
<b>EJB</b>	<b>741</b>
Benefits of EJB	742
Restrictions on EJB	742
Types of EJB	743
Session beans	743
Entity beans	744
Message-driven beans	745
Timer Service	745
<b>Summary</b>	<b>746</b>
<b>Exercises</b>	<b>746</b>
<b>Chapter 23: Access to Databases</b>	<b>747</b>
<hr/>	
<b>Introduction to Databases</b>	<b>748</b>
Connecting to a database	749
Downloading and Installing MySQL	752
JDBC APIs	753
<b>Building Applications</b>	<b>774</b>
Data characteristics	774
Transactions	775
Object-relational mapping	777
<b>Different Types of Applications</b>	<b>777</b>
Simple applications	778
Using JSP and JDBC	778
More complex applications	782
<b>Using Hibernate</b>	<b>782</b>
Installing Hibernate	784
<b>Summary</b>	<b>823</b>
<b>Chapter 24: Security</b>	<b>825</b>
<hr/>	
<b>Areas of Security</b>	<b>825</b>
Authentication	826
Authorization	828
Data integrity	830
Confidentiality	830
<b>Data Integrity and Confidentiality</b>	<b>830</b>
Implementing SSL	831

<b>Authentication</b>	<b>837</b>
HTTP BASIC authentication	837
Form-based authentication	843
Client-certificate authentication	849
<b>Authorization</b>	<b>855</b>
Container	855
Programmatic	855
<b>Summary</b>	<b>860</b>
<b>Exercises</b>	<b>860</b>
<b>Chapter 25: Performance</b>	<b>861</b>
<b>Performance Concepts</b>	<b>861</b>
What to measure	862
The user's perspective	863
How to measure performance	863
What to do after performance testing	865
<b>Measuring Performance Using JMeter</b>	<b>866</b>
Installing JMeter	866
JMeter concepts	866
<b>Performance Tuning Tips</b>	<b>872</b>
Development-time measures	872
Deployment-time measures	875
<b>Summary</b>	<b>883</b>
<b>Chapter 26: Best Practices and Tools</b>	<b>885</b>
<b>Development Methodologies</b>	<b>886</b>
Waterfall and iterative methodologies	886
Rational Unified Process	888
Extreme Programming	888
Test-driven development	889
Feature-driven development	890
<b>Development Tools</b>	<b>890</b>
Version-control tools	890
Build tools	890
Testing tools	891
Logging tools	891
Tools for enforcing coding standards	891
<b>Apache Ant</b>	<b>892</b>
Installing Ant	892
Ant concepts	893
Additional resources	903

# Contents

---

<b>CVS</b>	<b>903</b>
Installing CVS	903
CVS concepts	903
Connecting to a CVS server	904
Checking in code	905
Checking out code	906
Comparing changes across revisions	906
Additional resources	907
<b>JUnit</b>	<b>907</b>
Installing JUnit	907
JUnit concepts	908
Additional resources	915
<b>HttpUnit</b>	<b>915</b>
Installing HttpUnit	915
HttpUnit concepts	916
Additional resources	919
<b>Best Practices for Web Development</b>	<b>919</b>
Follow good JSP coding practices	919
Separate application logic and presentation	920
Use design patterns where appropriate	920
Use frameworks for developing applications	921
Early testing and integration	921
<b>Summary</b>	<b>921</b>
<b>Exercises</b>	<b>922</b>
<b>Part III: Spreading Your New Wings: Applying JSP in the Real World</b>	<b>923</b>
<b>Chapter 27: JSP Project I: Personalized Portal</b>	<b>925</b>
<b>The Portal Project</b>	<b>925</b>
Introduction to RSS	926
Introduction to Web services	929
<b>Project Features</b>	<b>930</b>
Use cases	931
Initial analysis	931
<b>The Application Design</b>	<b>932</b>
Designing the persistent store	932
Defining the key entity objects	935
Designing the Web site control flow	944

---

<b>The Development Environment</b>	<b>945</b>
Directory structure	946
Build scripts	947
Struts	949
Tiles	950
Log4j	951
Rome	953
Apache Axis	956
<b>The Application</b>	<b>958</b>
View	959
Controller	966
Running the portal application	975
<b>Summary</b>	<b>981</b>
<b>Exercises</b>	<b>981</b>
<b>Chapter 28: JSP Project II: Shopping Cart Application</b>	<b>983</b>
<b>The Bookstore Project</b>	<b>984</b>
Project features	984
Use cases	984
Initial analysis	985
<b>Application Design</b>	<b>986</b>
Database design	986
Defining the key entity objects	989
<b>The Development Environment and Its Configuration</b>	<b>1008</b>
Directory structure	1009
Tomcat	1010
Ant	1010
Database	1016
Hibernate	1016
Struts, Validator, and Tiles	1022
web.xml	1023
<b>The Application</b>	<b>1024</b>
Application layout: Tiles	1024
Cart status	1026
Presenting a list of categories	1028
Presenting a list of books	1030
Adding a book to the shopping cart	1034
Managing the shopping cart	1036
Checkout	1042
<b>Summary</b>	<b>1055</b>
<b>Exercises</b>	<b>1055</b>

## **Part IV: Appendixes** **1057**

### **Appendix A: JSP Syntax Reference** **1059**

---

#### **Scoping** **1060**

- page scope 1060
- request scope 1061
- session scope 1061
- application scope 1061

#### **Implicit Objects** **1062**

- The request object 1062
- The response object 1062
- The pageContext object 1063
- The session object 1063
- The application object 1064
- The out object 1064
- The config object 1065
- The page object 1065
- The exception object 1065

#### **Directives** **1066**

- The page directive 1066
- The taglib directive 1067
- The include directive 1067
- Tag file directives 1068

#### **Actions** **1070**

- Standard actions 1070
- Other JSP standard actions 1074

#### **Scripting Elements** **1075**

- Declaration scripting elements 1075
- Scriptlets 1075
- Expression scripting elements 1076

### **Appendix B: JSP Expression Language Reference** **1077**

---

#### **EL Expressions** **1077**

- EL expressions in-line with template data 1077
- EL expressions in attribute values 1078

#### **Accessing Arrays, Maps, Object Properties, and Collections** **1078**

- Object properties access 1078
- Array member access 1078
- Java map access 1078
- Java collection access 1079

---

<b>Elements of EL Expressions</b>	<b>1079</b>
Named variables	1079
Literals	1079
Operators	1080
Operator precedence	1081
Functions	1082
<b>Type Conversions</b>	<b>1083</b>
Coercion: automatic type conversion	1083
<b>Implicit Objects</b>	<b>1084</b>
The pageContext implicit object	1084
The pageScope implicit object	1085
The requestScope implicit object	1085
The sessionScope implicit object	1085
The applicationScope implicit object	1086
The param implicit object	1086
The paramValues implicit object	1086
The header implicit object	1087
The headerValues implicit object	1087
The cookies implicit object	1087
The initParam implicit object	1087

---

**Appendix C: JSTL Reference** **1089**

<b>JSTL Core Tags</b>	<b>1089</b>
catch	1090
choose	1091
if	1091
import	1092
forEach	1092
forEachTokens	1093
out	1094
otherwise	1095
param	1095
redirect	1095
remove	1096
set	1096
url	1097
when	1098
<b>JSTL XML Tags</b>	<b>1098</b>
choose	1099
forEach	1100
if	1100

# Contents

---

otherwise	1101
out	1101
param	1102
parse	1102
set	1103
transform	1104
when	1105
<b>JSTL Formatting Tags</b>	<b>1105</b>
bundle	1106
formatDate	1106
formatNumber	1107
message	1108
param	1109
parseDate	1109
parseNumber	1110
requestEncoding	1111
setBundle	1111
setLocale	1112
setTimeZone	1112
timeZone	1113
<b>JSTL SQL Tags</b>	<b>1113</b>
dateParam	1114
param	1114
query	1115
setDataSource	1116
transaction	1116
update	1117
<b>JSTL Functions</b>	<b>1118</b>
contains	1119
containsIgnoreCase	1119
endsWith	1119
escapeXml	1120
indexOf	1120
join	1120
length	1120
replace	1120
split	1121
startsWith	1121
substring	1121

substringAfter	1121
substringBefore	1122
toLowerCase	1122
toUpperCase	1122
trim	1122

---

<b>Appendix D: Exercise Solutions</b>	<b>1123</b>
---------------------------------------	-------------

<b>Index</b>	<b>1213</b>
--------------	-------------

