



Jakarta EE Recipes

A Problem-Solution Approach

—

Josh Juneau

Apress®

Jakarta EE Recipes

A Problem-Solution Approach

Josh Juneau

Apress®

Jakarta EE Recipes: A Problem-Solution Approach

Josh Juneau
Hinckley, IL, USA

ISBN-13 (pbk): 978-1-4842-5586-5
<https://doi.org/10.1007/978-1-4842-5587-2>

ISBN-13 (electronic): 978-1-4842-5587-2

Copyright © 2020 by Josh Juneau

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Managing Director, Apress Media LLC: Welmoed Spahr
Acquisitions Editor: Jonathan Gennick
Development Editor: Laura Berendson
Coordinating Editor: Jill Balzano

Cover image designed by Freepik (www.freepik.com)

Distributed to the book trade worldwide by Springer Science+Business Media New York, 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com. Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a **Delaware** corporation.

For information on translations, please e-mail rights@apress.com, or visit <http://www.apress.com/rights-permissions>.

Apress titles may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Print and eBook Bulk Sales web page at <http://www.apress.com/bulk-sales>.

Any source code or other supplementary material referenced by the author in this book is available to readers on GitHub via the book's product page, located at www.apress.com/9781484255865. For more detailed information, please visit <http://www.apress.com/source-code>.

Printed on acid-free paper

This book is dedicated to my wife, Angela, and my five children, Kaitlyn, Jacob, Matthew, Zachary, and Lucas. You are my joy and inspiration. It is also dedicated to the many Java and Jakarta EE developers worldwide. I hope that these recipes can lead you to developing the sophisticated solutions of tomorrow.

—Josh Juneau

Table of Contents

About the Authorxxxiii

About the Technical Reviewerxxxv

Acknowledgmentsxxxvii

Introductionxxxix

Chapter 1: Servlets and JavaServer Pages..... 1

 1-1. Developing a Servlet..... 3

 Problem 3

 Solution 3

 How It Works 5

 1-2. Packaging, Compiling, and Deploying a Servlet 8

 Problem 8

 Solution 8

 How It Works 9

 1-3. Registering a Servlet Without Web.xml..... 11

 Problem 11

 Solution 11

 How It Works 13

 1-4. Displaying Dynamic Content with a Servlet..... 14

 Problem 14

 Solution 14

 How It Works 17

 1-5. Handling Requests and Responses 17

 Problem 17

 Solution 17

 How It Works 20

TABLE OF CONTENTS

1-6. Listening for Servlet Container Events 22

 Problem 22

 Solution 22

 How It Works 24

1-7. Reading and Writing with Nonblocking I/O 26

 Problem 26

 Solution 26

 How It Works 31

1-8. Pushing Resources from a Server to a Client..... 34

 Problem 34

 Solution 34

 How It Works 36

1-9. Creating a Simple JSP Page 37

 Problem 37

 Solution 37

 How It Works 38

1-10. Embedding Java into a JSP Page 40

 Problem 40

 Solution 40

 How It Works 41

1-11. Separating Business Logic from View Code 43

 Problem 43

 Solution 43

 How It Works 44

1-12. Yielding or Setting Values 46

 Problem 46

 Solution 46

 How It Works 48

1-13. Invoking a Function in a Conditional Expression	50
Problem	50
Solution	50
How It Works	53
1-14. Creating a JSP Document.....	55
Problem	55
Solution	55
How It Works	56
1-15. Embedding Expressions in EL.....	58
Problem	58
Solution	58
How It Works	61
1-16. Accessing Parameters in Multiple Pages	65
Problem	65
Solution	65
How It Works	67
1-17. Creating a Custom JSP Tag.....	68
Problem	68
Solution	68
How It Works	71
1-18. Including Other JSPs into a Page	74
Problem	74
Solution	74
How It Works	75
1-19. Creating an Input Form for a Database Record	76
Problem	76
Solution	76
How It Works	80

TABLE OF CONTENTS

1-20. Looping Through Database Records Within a Page 82

 Problem 82

 Solution 82

 How It Works 86

1-21. Handling JSP Errors..... 87

 Problem 87

 Solution 87

 How It Works 89

1-22. Disabling Scriptlets in Pages..... 91

 Problem 91

 Solution 91

 How It Works 91

1-23. Ignoring EL in Pages..... 92

 Problem 92

 Solution #1 92

 Solution #2 92

 Solution #3 92

 How It Works 93

Chapter 2: JavaServer Faces Fundamentals 95

2-1. Writing a Simple JSF Application..... 96

 Problem 96

 Solution #1 96

 Solution #2 100

 How It Works 102

2-2. Writing a Controller Class 105

 Problem 105

 Solution 105

 How It Works 111

2-3. Building Sophisticated JSF Views with Components	114
Problem	114
Solution	114
How It Works	122
2-4. Displaying Messages in JSF Pages	125
Problem	125
Solution	125
How It Works	128
2-5. Updating Messages Without Recompiling	131
Problem	131
Solution	131
How It Works	134
2-6. Navigation Based upon Conditions	134
Problem	134
Solution	135
How It Works	142
2-7. Validating User Input.....	144
Problem	144
Solution	144
How It Works	149
2-8. Evaluation of Page Expressions Immediately	151
Problem	151
Solution	152
How It Works	154
2-9. Passing Page Parameters to Methods.....	155
Problem	155
Solution	155
How It Works	159

TABLE OF CONTENTS

2-10. Operators and Reserved Words in Expressions 161

 Problem 161

 Solution 161

 How It Works 163

2-11. Creating Bookmarkable URLs 165

 Problem 165

 Solution 165

 How It Works 167

2-12. Displaying Lists of Objects 169

 Problem 169

 Solution 169

 How It Works 173

2-13. Developing with HTML5..... 175

 Problem 175

 Solution 175

 How It Works 176

2-14. Creating Page Templates 177

 Problem 177

 Solution 177

 How It Works 179

2-15. Applying Templates..... 181

 Problem 181

 Solution 181

 How It Works 189

2-16. Adding Resources into the Mix 193

 Problem 193

 Solution 194

 How It Works 197

2-17. Handling Variable-Length Data 198

 Problem 198

 Solution 198

 How It Works 201

2-18. Invoking Controller Class Actions on Life-Cycle Phase Events	206
Problem	206
Solution	206
How It Works	206
Chapter 3: Advanced JavaServer Faces	209
Component and Tag Primer.....	210
Common Component Tag Attributes	214
Common JavaScript Component Tags	215
Binding Components to Properties	216
3-1. Creating an Input Form.....	217
Problem	217
Solution	217
How It Works	221
3-2. Invoking Actions from Within a Page	224
Problem	224
Solution	224
How It Works	228
3-3. Displaying Output	231
Problem	231
Solution	231
How It Works	235
3-4. Adding Form Validation.....	240
Problem	240
Solution #1	241
Solution #2	242
Solution #3	243
How It Works	245
3-5. Validating Input with Ajax	249
Problem	249
Solution	249
How It Works	255

TABLE OF CONTENTS

3-6. Submitting Pages Without Page Reloads..... 260

 Problem 260

 Solution 260

 How It Works 261

3-7. Making Partial-Page Updates..... 262

 Problem 262

 Solution 262

 How It Works 263

3-8. Applying Ajax Functionality to a Group of Components 264

 Problem 264

 Solution 264

 How It Works 270

3-9. Custom Processing of Ajax Functionality..... 270

 Problem 270

 Solution 271

 How It Works 273

3-10. Listening for System-Level Events 275

 Problem 275

 Solution 275

 How It Works 277

3-11. Listening for Component Events..... 278

 Problem 278

 Solution 278

 How It Works 279

3-12. Developing a Page Flow 280

 Problem 280

 Solution 280

 How It Works 285

3-13. Broadcasting Messages from the Server to All Clients	289
Problem	289
Solution	289
How It Works	291
3-14. Programmatically Searching for Components	293
Problem	293
Solution #1	293
Solution #2	294
How It Works	295
Chapter 4: Eclipse Krazo	299
4-1. Configure an Application for the Eclipse Krazo Framework	300
Problem	300
Solution	300
How It Works	303
4-2. Making Data Available for the Application	304
Problem	304
Solution #1	305
Solution #2	310
How It Works	316
4-3. Writing a Controller Class	317
Problem	317
Solution	317
How It Works	319
4-4. Using a Model to Expose Data to a View	322
Problem	322
Solution	322
How It Works	324

TABLE OF CONTENTS

4-5. Utilizing CDI for Exposing Data	325
Problem	325
Solution	325
How It Works	327
4-6. Supplying Message Feedback to the User	328
Problem	328
Solution	328
How It Works	332
4-7. Inserting and Updating Data.....	332
Problem	332
Solution	333
How It Works	335
4-8. Applying a Different View Engine.....	336
Problem	336
Solution #1	336
Solution #2	337
How It Works	339
Chapter 5: JDBC with Jakarta EE	341
5-1. Obtaining Database Drivers and Adding Them to the CLASSPATH	342
Problem	342
Solution	342
How It Works	344
5-2. Connecting to a Database.....	344
Problem	344
How It Works	349
5-3. Handling Database Connection Exceptions	353
Problem	353
Solution	353
How It Works	353

5-4. Simplifying Connection Management.....	354
Problem	354
Solution	354
How It Works	359
5-5. Querying a Database	360
Problem	360
Solution	360
How It Works	361
5-6. Performing CRUD Operations	363
Problem	363
Solution	363
How It Works	367
5-7. Preventing SQL Injection	368
Problem	368
Solution	369
How It Works	373
5-8. Utilizing Java Objects for Database Access.....	377
Problem	377
Solution	377
How It Works	384
5-9. Calling PL/SQL Stored Procedures.....	386
Problem	386
Solution	386
How It Works	387
5-10. Querying and Storing Large Objects.....	388
Problem	388
Solution	388
How It Works	392

TABLE OF CONTENTS

5-11. Querying with a REF_CURSOR..... 393

 Problem 393

 Solution 393

 How It Works 394

Chapter 6: Object-Relational Mapping..... 395

6-1. Creating an Entity 397

 Problem 397

 Solution 397

 How It Works 400

6-2. Mapping Data Types 403

 Problem 403

 Solution 403

 How It Works 405

6-3. Creating a Persistence Unit 407

 Problem 407

 Solution 407

 How It Works 408

6-4. Using Database Sequences to Create Primary Key Values 411

 Problem 411

 Solution 411

 How It Works 414

6-5. Generating Primary Keys Using More Than One Attribute 416

 Problem 416

 Solution #1 416

 Solution #2 420

 How It Works 422

6-6. Defining a One-to-One Relationship 426

 Problem 426

 Solution 426

 How It Works 429

6-7. Defining One-to-Many and Many-to-One Relationships	430
Problem	430
Solution	430
How It Works	433
6-8. Defining a Many-to-Many Relationship	436
Problem	436
Solution	436
How It Works	438
6-9. Querying with Named Queries	441
Problem	441
Solution	441
How It Works	443
6-10. Performing Validation on Entity Fields	443
Problem	443
Solution	444
How It Works	445
6-11. Generating Database Schema Objects Automatically	447
Problem	447
Solution	447
How It Works	447
6-12. Mapping Date-Time Values	454
Problem	454
Solution	454
How it Works	455
6-13. Using the Same Annotation Many Times	456
Problem	456
Solution	456
How It Works	457

TABLE OF CONTENTS

Chapter 7: Jakarta NoSQL..... 459

7-1. Configuring for Jakarta NoSQL..... 460

 Problem 460

 Solution 460

 How It Works 463

7-2. Writing a Query for a Document Database 464

 Problem 464

 Solution 465

 How It Works 465

7-3. Inserting, Updating, and Deleting from a Document-Oriented Database..... 467

 Problem 467

 Solution 467

 How It Works 469

7-4. Working with a Key-Value Database..... 472

 Problem 472

 Solution 472

 How It Works 473

Chapter 8: Enterprise JavaBeans 475

8-1. Obtaining an Entity Manager 476

 Problem 476

 Solution #1 476

 Solution #2 477

 How It Works 477

8-2. Developing a Stateless Session Bean 478

 Problem 478

 Solution #1 478

 Solution #2 480

 How It Works 484

8-3. Developing a Stateful Session Bean	487
Problem	487
Solution	487
How It Works	493
8-4. Utilizing Session Beans with JSF	495
Problem	495
Solution	495
How It Works	499
8-5. Persisting an Object	501
Problem	501
Solution	501
How It Works	501
8-6. Updating an Object	502
Problem	502
Solution	502
How It Works	502
8-7. Returning Data to Display in a Table	503
Problem	503
Solution #1	503
Solution #2	505
How It Works	507
8-8. Creating a Singleton Bean	508
Problem	508
Solution	509
How It Works	511
8-9. Scheduling a Timer Service	513
Problem	513
Solution #1	513
Solution #2	514
How It Works	515

TABLE OF CONTENTS

8-10. Performing Optional Transaction Life-Cycle Callbacks 518

 Problem 518

 Solution 518

 How It Works 519

8-11. Ensuring a Stateful Session Bean Is Not Passivated 521

 Problem 521

 Solution 521

 How It Works 521

8-12. Denoting Local and Remote Interfaces..... 522

 Problem 522

 Solution 522

 How It Works 522

8-13. Processing Messages Asynchronously from Enterprise Beans 524

 Problem 524

 Solution 524

 How It Works 525

Chapter 9: Java Persistence Query Language 527

9-1. Querying All Instances of an Entity..... 528

 Problem 528

 Solution #1 528

 Solution #2 528

 How It Works 529

9-2. Setting Parameters to Filter Query Results 531

 Problem 531

 Solution #1 531

 Solution #2 532

 How It Works 532

9-3. Returning a Single Object..... 534

 Problem 534

 Solution 534

 How It Works 535

9-4. Creating Native Queries.....	535
Problem	535
Solution #1	536
Solution #2	536
How It Works	537
9-5. Querying More Than One Entity	539
Problem	539
Solution #1	539
Solution #2	540
How It Works	541
9-6. Calling JPQL Aggregate Functions.....	545
Problem	545
Solution	545
How It Works	545
9-7. Invoking Database Stored Procedures Natively.....	547
Problem	547
Solution	547
How It Works	547
9-8. Joining to Retrieve Instances from Multiple Entities	548
Problem	548
Solution	548
How It Works	549
9-9. Joining to Retrieve All Rows Regardless of Match	550
Problem	550
Solution	550
How It Works	551
9-10. Applying JPQL Functional Expressions.....	552
Problem	552
Solution	552
How It Works	553

TABLE OF CONTENTS

9-11. Forcing Query Execution Rather Than Cache Use 555

 Problem 555

 Solution 555

 How It Works 556

9-12. Performing Bulk Updates and Deletes..... 556

 Problem 556

 Solution 556

 How It Works 557

9-13. Retrieving Entity Subclasses 560

 Problem 560

 Solution 560

 How It Works 561

9-14. Joining with ON Conditions 562

 Problem 562

 Solution 562

 How It Works 563

9-15. Processing Query Results with Streams 564

 Problem 564

 Solution 564

 How It Works 564

9-16. Converting Attribute Data Types 566

 Problem 566

 Solution 566

 How It Works 567

Chapter 10: Bean Validation 569

10-1. Validating Fields with Built-in Constraints..... 570

 Problem 570

 Solution #1 570

 Solution #2 571

 How It Works 572

10-2. Writing Custom Constraint Validators	574
Problem	574
Solution	574
How It Works	576
10-3. Validating at the Class Level.....	577
Problem	577
Solution	578
How It Works	580
10-4. Validating Parameters	581
Problem	581
Solution	581
How It Works	582
10-5. Constructor Validation	583
Problem	583
Solution	583
How It Works	583
10-6. Validating Return Values	584
Problem	584
Solution	584
How It Works	585
10-7. Defining a Dynamic Validation Error Message.....	585
Problem	585
Solution	585
How It Works	586
10-8. Manually Invoking the Validator Engine.....	588
Problem	588
Solution	588
How It Works	588

TABLE OF CONTENTS

10-9. Grouping Validation Constraints 589

 Problem 589

 Solution 589

 How It Works 591

Chapter 11: Contexts and Dependency Injection 593

11-1. Injecting a Contextual Bean or Other Object..... 594

 Problem 594

 Solution 594

 How It Works 596

11-2. Binding a Bean to a Web View 598

 Problem 598

 Solution 598

 How It Works 601

11-3. Allocating a Specific Bean for Injection 602

 Problem 602

 Solution 603

 How It Works 605

11-4. Determining Scope of a Bean 606

 Problem 606

 Solution 606

 How It Works 609

11-5. Injecting Non-bean Objects 610

 Problem 610

 Solution 610

 How It Works 613

11-6. Ignoring Classes 615

 Problem 615

 Solution #1 615

 Solution #2 615

 How It Works 616

11-7. Disposing of Producer Fields	617
Problem	617
Solution	617
How It Works	617
11-8. Specifying an Alternative Implementation at Deployment Time	617
Problem	617
Solution	618
How It Works	618
11-9. Injecting a Bean and Obtaining Metadata	619
Problem	619
Solution	619
How It Works	619
11-10. Invoking and Processing Events	620
Problem	620
Solution	620
How It Works	624
11-11. Intercepting Method Invocations	625
Problem	625
Solution	626
How It Works	629
11-12. Bootstrapping Java SE Environments	630
Problem	630
Solution	630
How It Works	631
11-13. Enhancing Business Logic of a Method	632
Problem	632
Solution	632
How It Works	634

Chapter 12: Java Message Service..... 637

12-1. Creating JMS Resources 638

 Problem 638

 How It Works 642

12-2. Creating a Session..... 645

 Problem 645

 Solution 645

 How It Works 646

12-3. Creating and Sending a Message..... 647

 Problem 647

 How It Works 649

12-4. Receiving Messages..... 652

 Problem 652

 How It Works 654

12-5. Filtering Messages 655

 Problem 655

 Solution 655

 How It Works 658

12-6. Inspecting Message Queues..... 658

 Problem 658

 Solution 658

 How It Works 660

12-7. Creating Durable Message Subscribers 660

 Problem 660

 Solution 660

 How It Works 665

12-8. Delaying Message Delivery 667

 Problem 667

 Solution 667

 How It Works 667

Chapter 13: RESTful Web Services	669
13-1. Developing a RESTful Web Service.....	671
Problem	671
Solution #1	671
Solution #2	672
How It Works	675
13-2. Consuming and Producing with REST	678
Problem	678
Solution	678
How It Works	681
13-3. Writing a Jakarta RESTful Web Services Client	683
Problem	683
Solution	683
How It Works	684
13-4. Filtering Requests and Responses	690
Problem	690
Solution	690
How It Works	691
13-5. Processing Long-Running Operations Asynchronously	694
Problem	694
Solution	694
How It Works	696
13-6. Pushing One-Way Asynchronous Updates from Servers	698
Problem	698
Solution	698
How It Works	701
13-7. Receiving Server-Sent Events As a Client	703
Problem	703
Solution	703
How It Works	704

Chapter 14: WebSockets and JSON 705

14-1. Creating a WebSocket Endpoint 706

 Problem 706

 Solution 706

 How It Works 707

14-2. Sending Messages to a WebSocket Endpoint..... 707

 Problem 707

 Solution 707

 How It Works 710

14-3. Building a JSON Object..... 712

 Problem 712

 Solution 712

 How It Works 714

14-4. Writing a JSON Object to Disk 715

 Problem 715

 Solution 715

 How It Works 716

14-5. Reading JSON from an Input Source 716

 Problem 716

 Solution 716

 How It Works 717

14-6. Converting Between JSON and Java Objects 718

 Problem 718

 Solution 718

 How It Works 720

14-7. Custom Mapping with JSON-B 722

 Problem 722

 Solution 722

 How It Works 723

14-8. Replacing a Specified Element in a JSON Document.....	725
Problem	725
Solution	725
How It Works	726
Chapter 15: Security	729
15-1. Setting Up Application Users and Groups in GlassFish.....	730
Problem	730
Solution	730
How It Works	733
15-2. Performing Basic Web Application Authorization.....	734
Problem	734
How It Works	737
15-3. Developing a Programmatic Login Form with Custom Authentication Validation.....	741
Problem	741
Solution	741
How It Works	758
15-4. Authentication with the Security API Using Database Credentials	760
Problem	760
Solution	760
How It Works	767
15-5. Managing Page Access Within a JSF Application	770
Problem	770
Solution	770
How It Works	772
15-6. Configuring LDAP Authentication Within GlassFish/Payara	773
Problem	773
Solution	773
How It Works	775

TABLE OF CONTENTS

15-7. Configuring Custom Security Certificates Within GlassFish/Payara 776

 Problem 776

 Solution 776

 How It Works 777

Chapter 16: Concurrency and Batch..... 779

16-1. Creating Resources for Processing Tasks Asynchronously in an Application Server 780

 Problem 780

 How It Works 783

16-2. Configuring and Creating a Reporter Task..... 784

 Problem 784

 Solution 784

 How It Works 789

16-3. Running More Than One Task Concurrently..... 790

 Problem 790

 Solution 790

 How It Works 793

16-4. Utilizing Transactions Within a Task..... 794

 Problem 794

 Solution 795

 How It Works 796

16-5. Running Concurrent Tasks at Scheduled Times 797

 Problem 797

 Solution 797

 How It Works 800

16-6. Creating Thread Instances..... 801

 Problem 801

 Solution 801

 How It Works 803