

Programação Java em Ambiente Distribuído

**Ênfase no Mapeamento Objeto-Relacional
com JPA, EJB e Hibernate**

Douglas Rocha Mendes

Copyright © 2011 da Novatec Editora Ltda.

Todos os direitos reservados e protegidos pela Lei 9610 de 19/02/1998.
É proibida a reprodução desta obra, mesmo parcial, por qualquer processo,
sem prévia autorização, por escrito, do autor e da Editora.

Editor: Rubens Prates

Revisão gramatical: Marta Almeida de Sá

Capa: Victor Bittow

Editoração eletrônica: Camila Kuwabata e Carolina Kuwabata

ISBN: 978-85-7522-262-1

Histórico de impressões:

Outubro/2011 Primeira edição

Novatec Editora Ltda.

Rua Luís Antônio dos Santos 110
02460-000 – São Paulo, SP – Brasil

Tel.: +55 11 2959-6529

Fax: +55 11 2950-8869

E-mail: novatec@novatec.com.br

Site: www.novatec.com.br

Twitter: twitter.com/novateceditora

Facebook: facebook.com/novatec

LinkedIn: linkedin.com/in/novatec

Dados Internacionais de Catalogação na Publicação (CIP)
(Câmara Brasileira do Livro, SP, Brasil)

Mendes, Douglas Rocha
Programação Java em ambiente distribuído :
ênfase no mapeamento objeto-relacional com JPA,
EJB e Hibernate / Douglas Rocha Mendes. --
São Paulo : Novatec Editora, 2011.

ISBN 978-85-7522-262-1

1. Análise de sistemas 2. Java (Linguagem de
programação) 3. Sistemas distribuídos -
Desenvolvimento I. Título.

11-07649

CDD-005.133

Índices para catálogo sistemático:

1. Java : Linguagem de programação :
Computadores : Processamento de dados
005.133
OGF20111003

Sumário

Agradecimentos.....	13
Sobre o autor	14
Prefácio	16
Capítulo 1 ■ Framework collection.....	18
1.1 Introdução	18
1.2 Vantagens de se usar o framework collection	19
1.3 Interfaces e classes do framework collection	20
1.3.1 Interface Collection	20
1.3.2 Interface Set.....	25
1.3.3 Classe HashSet	29
1.3.4 Classe LinkedHashSet.....	30
1.3.5 Interface SortedSet.....	32
1.3.6 Classe TreeSet	33
1.3.7 Interface List.....	38
1.3.8 Classe ArrayList	39
1.3.9 Classe LinkedList	44
1.3.10 Classe Vector	45
1.3.11 Classe Stack.....	46
1.3.12 Interface Queue.....	47
1.3.13 Interface Map	48
1.3.14 Classe HashMap.....	50
1.3.15 Classe LinkedHashMap	51
1.3.16 Classe Hashtable	55
1.3.17 Classe Properties	55
1.3.18 Interface SortedMap	58
1.3.19 Classe TreeMap	58
1.4 Interfaces auxiliares do framework collection.....	60
1.4.1 Interface Iterable	60
1.4.2 Interface Iterator	62
1.4.3 Interface ListIterator	64
1.4.4 Interface Comparable.....	66

1.4.5 Interface Comparator.....	70
1.5 Classes utilitárias do framework collection.....	73
1.5.1 Classe Collections	73
1.5.2 Classe Arrays.....	73
1.6 Exercícios.....	74
1.6.1 Exercício 1.....	74
1.6.2 Exercício 2.....	76
1.6.3 Exercício 3.....	78
1.6.4 Exercício 4.....	80
1.6.5 Exercício 5.....	81
1.6.6 Exercício 6.....	83
Capítulo 2 ■ Genéricos – Tipos Parametrizados.....	86
2.1 Introdução	86
2.2 Por que usar genéricos?	88
2.2.1 Benefícios do uso dos genéricos.....	88
2.2.2 Definição de Type Erasure	92
2.2.3 Relação entre casting (coerção) e genéricos	92
2.3 Criando tipos Genéricos.....	94
2.4 Tipos genéricos e herança	97
2.5 Usando curingas	99
2.5.1 Restrição de uso do método add() em método com curinga	100
2.5.2 Curinga limitado (Bounded Wildcard)	102
2.6 Criando e usando métodos genéricos.....	107
2.7 Criando e usando classes genéricas	110
2.8 Exercícios	112
2.8.1 Exercício 1.....	112
2.8.2 Exercício 2	114
Capítulo 3 ■ Annotations	116
3.1 Introdução	116
3.1.1 O que são annotations	117
3.1.2 Considerações sobre annotations.....	117
3.1.3 Usando annotations	118
3.1.4 Meta annotations @Target e @Retention	118
3.1.5 Reflexão computacional	120
3.2 Annotations built-ins	120
3.2.1 @Deprecated	121
3.2.2 @Override	121
3.2.3 @SuppressWarnings.....	123
3.2.4 @Documented	124
3.2.5 @Inherited.....	124

Sumário	9
3.3 Classificação das annotations	125
3.3.1 Criando e usando annotations do tipo marker	125
3.3.2 Criando e usando annotations do tipo single value	129
3.3.3 Criando e usando annotations do tipo full value	133
3.4 Exercícios.....	141
3.4.1 Exercício 1.....	141
3.4.2 Exercício 2.....	142
3.4.3 Exercício 3.....	143
Capítulo 4 ■ JDBC	145
4.1 Introdução	145
4.1.1 A linguagem SQL.....	145
4.2 Conectando-se a um banco de dados com JDBC.....	146
4.2.1 Comandos utilizados para uma conexão	149
4.3 Comandos DDL (Data Definition Language).....	152
4.4 Comandos DML (Data Manipulation Language).....	155
4.4.1 Usando comandos DML e o framework collection.....	158
4.5 Usando a interface Statement	166
4.5.1 Método executeUpdate()	168
4.5.2 Método executeQuery()	168
4.5.3 Método execute()	168
4.6 Usando a interface PreparedStatement	172
4.7 Usando a interface CallableStatement	177
4.7.1 Usando o método registerOutParameter()	180
4.7.2 Vantagens do uso de stored procedures	183
4.8 Comandos DCL (Data Control Language)	184
4.9 Usando JDBC com comandos em lote.....	190
4.10 Usando o padrão de projeto DAO.....	196
Capítulo 5 ■ JPA e EJB	199
5.1 Introdução	199
5.1.1 Especificação JPA.....	201
5.1.2 Vantagens do uso da especificação JPA	203
5.1.3 EJB – Enterprise JavaBeans.....	203
5.2 Exemplo de uma aplicação usando JPA	204
5.2.1 Bean de entidade.....	205
5.2.2 Criando banco de dados e sequences.....	209
5.2.3 Bean de sessão	211
5.3 Annotations estratégicas	224
5.3.1 Mapeando a chave primária – @Id.....	224
5.3.2 Mapeando uma chave composta – @idClass.....	224
5.3.3 Mapeando atributos transientes – @Transient	231

5.3.4 Annotation @Basic	231
5.3.5 Enumeration FetchType (LAZY e EAGER).....	231
5.3.6 Mapeando datas com a annotation @Temporal	241
5.3.7 Mapeamento múltiplo com @SecondaryTable	241
5.3.8 Mapeamento múltiplo com @SecondaryTables.....	247
5.4 Relacionamento entre tabelas.....	252
5.4.1 One-to-one unidirecional	253
5.4.2 One-to-one bidirecional	259
5.4.3 One-to-many unidirecional	262
5.4.4 One-to-many bidirecional	264
5.4.5 Many-to-one unidirecional.....	274
5.4.6 Many-to-many bidirecional	281
5.4.7 Many-to-many bidirecional sem o uso do atributo mappedBy	291
5.4.8 Many-to-many unidirecional.....	295
5.5 Usando a interface Map na especificação JPA.....	303
5.6 Utilizando herança na especificação JPA.....	308
5.6.1 Conceito de herança	309
5.6.2 Conceito de polimorfismo	310
5.6.3 Consulta polimórfica	310
5.6.4 Estratégia SINGLE_TABLE	312
5.6.5 Estratégia TABLE_PER_CLASS.....	332
5.6.6 Estratégia JOINED	348
5.7 Análise de desempenho entre as estratégias	362
5.7.1 Análise de desempenho para inserção.....	363
5.7.2 Análise de desempenho para seleção.....	364
5.7.3 Análise de desempenho para deleção	366
5.7.4 Programas utilizados para os testes	368
5.8 Herança de classes que não representam entidades	387
5.9 Considerações finais sobre o mapeamento de herança	389
5.10 Interfaces Query e EntityManager e o EJB QL.....	390
5.10.1 Interfaces Query e EntityManager.....	391
5.10.2 Pesquisa paginada com a interface Query	395
5.10.3 Linguagem EJB QL.....	401
5.10.4 Queries nativas (native).....	423
5.10.5 Querys identificadas (named)	433
5.10.6 Queries nativas e identificadas (named)	434
5.11 Diferença entre um bean de sessão @Stateless e @Stateful	435
5.12 Auxílio para a preparação do ambiente	441
5.12.1 Instalando e ajustando os bancos de dados	441
5.12.2 Instalando e configurando o servidor de aplicação.....	442
5.12.3 Criando um projeto do tipo EJB.....	443
5.12.4 Relacionando o projeto EJB com o servidor de aplicação	447
5.12.5 Relacionando o projeto EJB com o framework JUnit	449
5.12.6 Criando um projeto do tipo EAR	451

Sumário	11
5.13 Exercícios	453
5.13.1 Exercício 1.....	453
5.13.2 Exercício 2	455
5.13.3 Exercício 3.....	457
5.13.4 Exercício 4.....	459
5.13.5 Exercício 5	463
Capítulo 6 ■ Teste de software com JUnit.....	467
6.1 Qualidade versus Testes	467
6.1.1 Cobertura dos testes	469
6.2 Atividades do processo de teste de software	469
6.2.1 Teste unitário e teste unitário integrado.....	469
6.2.2 Teste de sistema e teste integrado de sistema	470
6.2.3 Teste de aceitação por parte do usuário.....	470
6.2.4 Teste de aceitação por parte da área de suporte.....	470
6.3 Benefícios dos testes de unidade	471
6.4 Por que testar?.....	472
6.5 Pressão no processo de testes	473
6.6 Usando o framework JUnit.....	473
6.7 Annotations usadas no framework JUnit.....	477
6.7.1 Annotation @Before	477
6.7.2 Annotation @BeforeClass	481
6.7.3 Annotation @Test	486
6.7.4 Annotation @After	487
6.7.5 Annotation @AfterClass.....	487
6.7.6 Annotation @Ignore	487
6.7.7 Annotation @RunWith e annotation @SuiteClasses	487
6.8 Classe Assert	488
6.9 Exercícios	488
6.9.1 Exercício 1.....	488
6.9.2 Exercício 2	490
Índice remissivo	492