

# ***Java Persistence API mit Hibernate***

***Seminarunterlage***

***Version: 6.03***



Dieses Dokument wird durch die ORDIX AG veröffentlicht.

Copyright ORDIX AG. Alle Rechte vorbehalten.

Alle Produkt- und Dienstleistungs-Bezeichnungen sind Warenzeichen oder eingetragene Warenzeichen der jeweiligen Firmen und beziehen sich auf Eintragungen in den USA oder USA-Warenzeichen.

Weitere Logos und Produkt- oder Handelsnamen sind eingetragene Warenzeichen oder Warenzeichen der jeweiligen Unternehmen.

Kein Teil dieser Dokumentation darf ohne vorherige schriftliche Genehmigung der ORDIX AG weitergegeben oder benutzt werden.

### **Adressen der ORDIX AG**

Die ORDIX AG besitzt folgende Geschäftsstellen

ORDIX AG  
Karl-Schurz-Straße 19a  
D-33100 Paderborn  
Tel.: (+49) 0 52 51 / 10 63 - 0  
Fax.: (+49) 01 80 / 1 67 34 90

ORDIX AG  
An der alten Ziegelei 5  
D-48157 Münster  
Tel.: (+49) 02 51 / 9 24 35 – 00  
Fax.: (+49) 01 80 / 1 67 34 90

ORDIX AG  
Welser Straße 9  
D-86368 Gersthofen  
Tel.: (+49) 08 21 / 507 492 – 0  
Fax.: (+49) 01 80 / 1 67 34 90

ORDIX AG  
Kreuzberger Ring 13  
D-65205 Wiesbaden  
Tel.: (+49) 06 11 / 7 78 40 – 00  
Fax.: (+49) 01 80 / 1 67 34 90

ORDIX AG  
Wikingerstraße 18-20  
D-51107 Köln  
Tel.: (+49) 02 21 / 8 70 61 – 0  
Fax.: (+49) 01 80 / 1 67 34 90

Internet: <http://www.ordix.de>

Email: [seminare@ordix.de](mailto:seminare@ordix.de)

## Inhaltsverzeichnis

<b>1</b>	<b>Java Persistence API .....</b>	<b>7</b>
1.1	Motivation .....	8
1.2	Persistenz Framework in Java .....	9
1.3	JPA Timeline .....	10
1.4	JPA 1.0 .....	11
1.5	JPA 2.0 .....	12
1.6	JPA 2.1 .....	13
1.7	JPA - Grundlagen .....	14
1.7.1	JPA Provider (Anbieter) .....	14
1.7.2	Beispiel: JPA Provider .....	15
1.7.3	DB Konfiguration via XML – <i>persistence.xml</i> .....	16
1.7.4	Java Persistenz API .....	17
1.7.5	EntityManager .....	18
1.7.6	Persistence Context .....	19
1.7.7	Typen von EntityManager .....	20
1.7.8	Java SE EntityManager .....	21
1.7.9	Entity Manager – Transaktionalität .....	22
1.7.10	EntityTransaction .....	23
1.7.11	Java EE EntityManager .....	24
1.7.12	EntityManager Interface .....	25
1.7.13	Beispiel: Arbeitsweise des Entity Manager .....	26
1.7.14	Entity Lebenszyklus (Zustände) .....	27
1.7.15	Detachment .....	29
1.7.16	JPA Konfiguration (Persistence Unit) .....	30
1.7.17	Persistence Konfiguration .....	31
1.7.18	Beispiel 1: Derby .....	33
1.7.19	Beispiel 2: hsqldb .....	34
1.7.20	Beispiel 3: MySQL .....	35
1.7.21	Entity .....	36
1.8	ORM – simple Mapping .....	38
1.8.1	ORM - JPA Annotationen .....	38
1.8.2	Default Mapping .....	39
1.8.3	Zu persistierende Felder .....	40
1.8.4	@Basic .....	41
1.8.5	Beispiel: @Basic .....	43
1.8.6	@Transient .....	44
1.8.7	Beispiel: @Transient .....	45
1.8.8	Entity Identität – Primärschlüssel .....	46
1.8.9	Einfacher Primärschlüssel @Id .....	47
1.8.10	Primärschlüssel - Generierte PK .....	48
1.8.11	Beispiel: GenerationType.AUTO .....	49
1.8.12	Beispiel: GenerationType.SEQUENCE .....	50
1.8.13	Komplexer Primärschlüssel .....	51
1.8.14	Primärschlüssel - @IdClass .....	52
1.8.15	Beispiel: @IdClass .....	53
1.8.16	Primärschlüssel - @EmbeddedId .....	54
1.8.17	Beispiel: @EmbeddedId .....	55
1.8.18	@Table .....	56
1.8.19	@Temporal .....	57
1.8.20	Beispiel: @Temporal .....	58
1.8.21	@Enumerated .....	59
1.8.22	Beispiel 1: @Enumerated .....	60
1.8.23	Beispiel 2: @Enumerated .....	61
1.8.24	@Column .....	62
1.8.25	@Column Parameter .....	63
1.8.26	@Lob .....	64
1.8.27	Beispiel: @Lob .....	65
1.8.28	Multi-table Mapping - @SecondaryTable .....	66

1.8.29	Beispiel: Multi-table Mapping .....	67
1.8.30	Eingebettete Objekte.....	68
1.8.31	Beispiel: Eingebettete Objekte .....	69
1.8.32	Eingebettete Objekte - @AttributeOverride .....	70
1.8.33	Beispiel: @AttributeOverride .....	71
1.8.34	Callbacks bei Entities .....	72
1.9	Entity Hibernate Special.....	73
1.9.1	Hibernate-Erweiterung - @DynamicInsert/Update.....	73
1.9.2	Hibernate-Erweiterung – Immutable .....	75
1.10	Entity Relationships.....	76
1.10.1	Motivation .....	76
1.10.2	Navigierbarkeit .....	77
1.10.3	Kardinalität .....	78
1.10.4	@OneToOne .....	79
1.10.5	@OneToMany .....	80
1.10.6	Beispiel: @OneToMany .....	81
1.10.7	@ManyToMany .....	82
1.11	Abfragen - Grundlagen.....	83
1.11.1	Motivation – Abfragen .....	84
1.11.2	Abfragen.....	85
1.11.3	JPQL.....	86
1.11.4	Typen von Abfragen .....	87
1.11.5	Beispiel 1 - <i>NativeQuery</i> .....	88
1.11.6	Beispiel 2 – <i>Query</i> .....	89
1.11.7	Beispiel 3 - <i>Query</i> mit Parameter .....	90
1.11.8	Beispiel 4 – <i>NamedQuery</i> .....	91
1.11.9	Fazit JPQL.....	92
1.12	Arbeiten mit JPQL.....	93
1.12.1	Anwendungsfälle .....	93
1.12.2	Selektionen.....	94
1.12.3	Beispiel Abfragen .....	95
1.12.4	JPQL <i>Expressions</i> (Pfade).....	96
1.12.5	Selektionen (DISTINCT).....	99
1.12.6	Multiple Iterationsvariablen .....	100
1.12.7	Multiple Iterationsvariablen - Join (INNER).....	101
1.12.8	Multiple Iterationsvariablen - Join (OUTER).....	102
1.12.9	<i>N+1 Select</i> Problem .....	103
1.12.10	<i>N+1 Select</i> lösen - Inner Join .....	104
1.12.11	<i>N+1 Select</i> Problem - Hibernate .....	105
1.12.12	Mengen einschränken.....	106
1.12.13	<i>Paging</i> .....	107
1.13	Was ist neu in JPA 2.0? .....	108
1.13.1	JPA 1.0 vs. JPA 2.0.....	108
1.13.2	Caching in JPA 1.x.....	109
1.13.3	Caching in JPA 2.x.....	110
1.13.4	Cache API - Second Level Cache.....	111
1.13.5	Neuigkeiten in JPA 2.0 – Mapping.....	112
1.13.6	Standard Properties in persistence.xml .....	115
1.13.7	Beispiel: Persistence.xml in JPA 1.0.....	116
1.13.8	Beispiel: Persistence.xml in JPA 2.0.....	117
1.14	Criteria API – Typesafe Queries .....	118
1.14.1	Motivation .....	118
1.14.2	Einführung in die Criteria-API.....	119
1.14.3	Motivation .....	121
1.14.4	JPA Criteria API .....	122
1.14.5	JPA Criteria API – Query-Komponenten.....	123
1.14.6	Begriffe .....	124
1.14.7	Expression<X> .....	125
1.14.8	Criteria Literale .....	126
1.14.9	Path<X> .....	128

1.14.10 Operatoren .....	129
1.14.11 Binäre Operatoren .....	130
1.14.12 Unäre Operatoren .....	131
1.14.13 String Manipulation.....	132
1.14.14 Date & Time Funktionen.....	133
1.14.15 Collections Manipulation .....	134
1.14.16 Vergleiche .....	135
1.14.17 Logische Operatoren.....	136
1.14.18 Criteria API – OO Abfragen.....	137
1.14.19 SELECT Klausel.....	138
1.14.20 SELECT DISTINCT .....	139
1.14.21 Multi SELECT .....	140
1.14.22 Multi SELECT – OO Alternative .....	141
1.14.23 CriteriaBuilder's construct .....	142
1.14.24 CriteriaBuilder Array .....	143
1.14.25 CriteriaBuilder TUPEL .....	144
1.14.26 Multi SELECT – Factory multiselect.....	145
1.14.27 JOIN Klausel .....	146
1.14.28 Criteria API – JOIN.....	147
1.14.29 Metamodell .....	148
1.14.30 Maven Konfiguration .....	149
1.14.31 Hibernate Metamodell Generator.....	150
1.14.32 Beispiel.....	151
1.14.33 Beispiel: Metamodell .....	152
1.14.34 Fazit.....	153
<b>2 Java Persistenz - Vererbung .....</b>	<b>155</b>
2.1 Einleitung.....	156
2.2 Strategien .....	157
2.3 Vererbungshierarchie.....	158
2.4 SINGLE_TABLE.....	159
2.5 TABLE_PER_CLASS.....	164
2.6 JOINED .....	167
2.7 Mapped Superclass .....	170
2.8 Transiente Klassen .....	174
2.9 Polymorphe Abfragen .....	175
2.10 Polymorphe Assoziationen.....	178
2.11 Fallstricke .....	179
<b>3 Java Transaction Api (JTA) .....</b>	<b>182</b>
3.1 Java Transaction API .....	183
3.2 Begriffe.....	184
3.2.1 Was ist eine Transaktion? .....	184
3.2.2 ACID-Eigenschaften.....	185
3.2.3 Isolation-Level .....	186
3.2.4 Grundlegende Probleme (ohne Transaktionen).....	187
3.2.5 Transaktionen bei Konkurrenzzugriff .....	188
3.2.6 Lösung der Probleme .....	189
3.2.7 Grenzen einer Transaktion.....	190
3.2.8 X/Open XA Protokoll (XA).....	191
3.2.9 Verteilte (XA) Transaktionen .....	192
3.3 Java EE Transaktionen.....	193
3.3.1 JTA Package .....	193
3.3.2 Transaktionen API (JTA, JTS) .....	194
3.3.3 EJBs und Transaktionen .....	195
3.3.4 Container-Managed Transactions (CMT).....	196
3.3.5 CMT – Transaktionsattribute .....	197
3.3.6 Transaktionsattribute Required .....	198
3.3.7 Transaktionsattribute Requires New .....	199
3.3.8 Weitere Transaktionsattribute .....	200

3.3.9	CMT – Transaktionsgrenzen .....	201
3.3.10	Bean-Managed Transactions (BMT) .....	202
3.3.11	(BMT) – UserTransaction Interface .....	203
3.3.12	BMT – Programmbeispiel .....	204
3.3.13	Fazit .....	205
<b>4</b>	<b>Caching .....</b>	<b>206</b>
4.1	Caching .....	207
4.2	Caching in Hibernate .....	208
4.2.1	Caching – First Level Cache .....	209
4.2.2	First Level Cache .....	209
4.2.3	Beispiel für First Level Cache .....	210
4.2.4	Cache Handling .....	211
4.2.5	Szenariobeschreibung .....	212
4.2.6	Funktionsweise .....	213
4.2.7	Laufzeitanalyse .....	215
4.3	Caches – Second Level Cache .....	216
4.3.1	Second Level Cache .....	216
4.3.2	Cache Provider .....	217
4.3.3	Cache Mode .....	218
4.3.4	Dirty Read .....	219
4.3.5	Nonrepeatable Reads .....	220
4.3.6	Unterstützte Strategien .....	221
4.3.7	Cache Handling .....	222
4.3.8	Konfiguration (Hibernate) .....	223
4.3.9	Konfiguration (JPA) .....	224
4.3.10	Cache Provider konfigurieren - ehcache.xml .....	225
4.3.11	Cache Strategie festlegen .....	226
4.3.12	Funktionsweise .....	227
4.3.13	Laufzeitanalyse .....	229
4.3.14	Nachteile/Stolperfallen/Sonstiges .....	230
4.3.15	EhCacheMonitor .....	231
4.4	Caches – Query Cache .....	232
4.4.1	Grundlagen .....	232
4.4.2	Konfiguration .....	233
4.4.3	Konfiguration .....	234
4.4.4	Funktionsweise .....	236
4.4.5	Laufzeitanalyse .....	238
4.4.6	Nachteile/Stolperfallen/Sonstiges .....	239
4.4.7	Zusammenfassung .....	240
<b>5</b>	<b>Validierung .....</b>	<b>241</b>
5.1	Allgemeines .....	242
5.2	Beispiel .....	243
5.3	Vordefinierte Constraints .....	244
5.4	Benutzerdefinierte Constraints .....	246
5.5	Kaskadierung .....	247
5.6	Validierungsgruppen – Definition .....	248
5.7	Validierungsgruppen – Anwendung .....	249
5.8	Implizite Validierung in JPA .....	250
5.9	Weitere Konfigurationsmöglichkeiten .....	251
5.10	Explizite Validierung .....	252