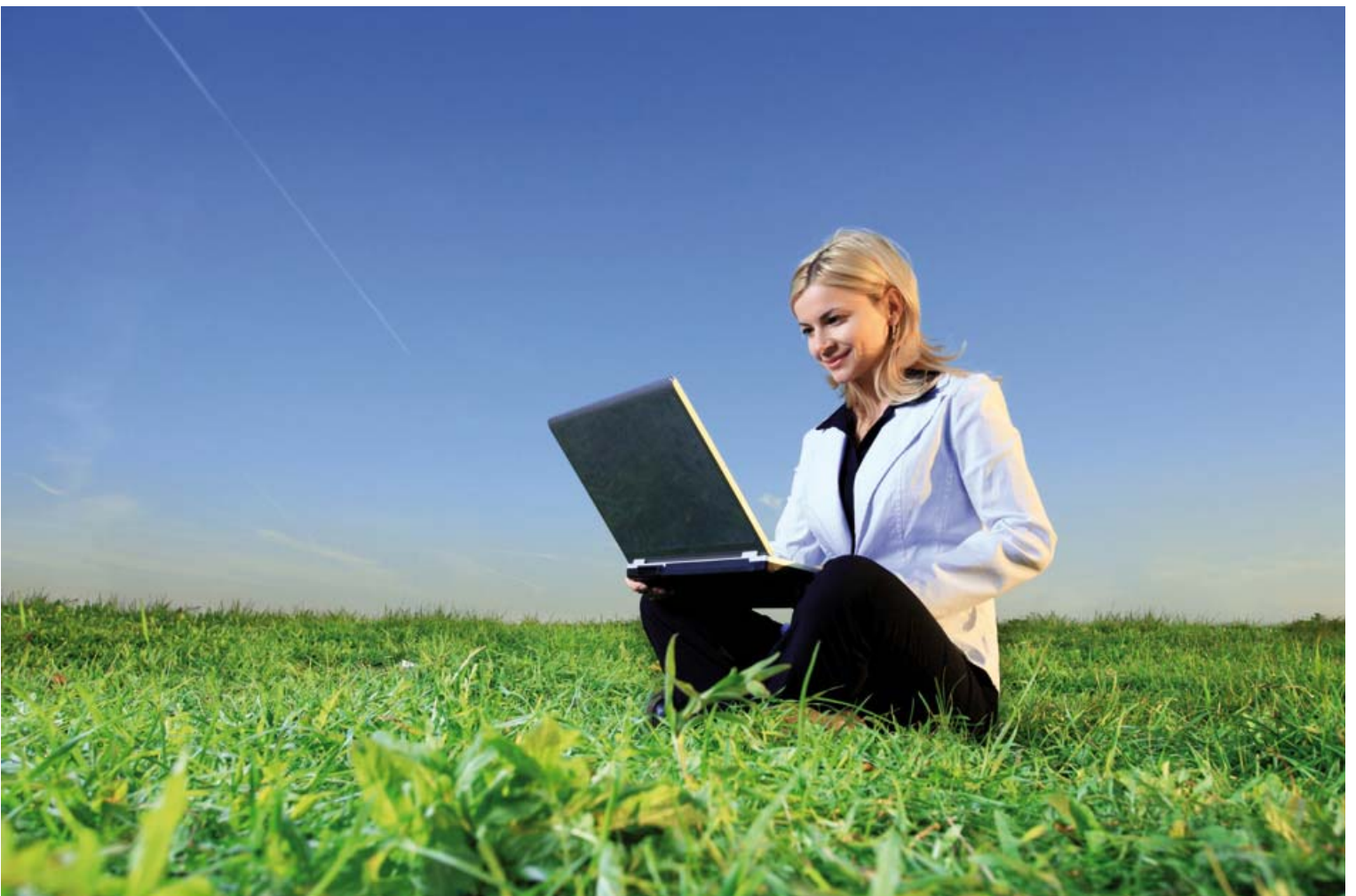


ORDIX AG

Entwickeln mit dem Spring-Framework

Seminarunterlagen

Version: 2.02



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
Westernmauer 12-16
D-33098 Paderborn
Tel.: (+49) 0 52 51 / 10 63 - 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
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
Wikingerstr. 18-20
D-51107 Köln
Tel.: (+49) 02 21 / 8 70 61 - 0
Fax.: (+49) 01 80 / 1 67 34 90

ORDIX AG
Marlene-Dietrich-Str. 5
D-89231 Neu-Ulm
Tel.: (+49) 07 31 / 9 85 88 -
Fax.: (+49) 01 80 / 1 67 34 90

Sie können die ORDIX AG von der ganzen Welt aus durch folgende Internet Adresse kontaktieren:

<http://www.ordix.de>

Sie können uns weiterhin über die Email-Adressen
training@ordix.de oder **info@ordix.de** kontaktieren

Inhaltsverzeichnis

1. Einführung.....	6
1.1 Spring	7
1.2 Warum Spring?	8
1.3 Die Säulen von Spring	9
1.4 Objektnetze in OO-Systemen	10
1.5 Objektnetze mit Dependency Injection	12
1.6 DI - Dependency Injection	13
1.7 Das Problem „Cross-Cutting-Concern“	14
1.8 AOP - Aspect Oriented Programming	15
2. Dependency Injection.....	17
2.1 Objektnetze in OO-Systemen	18
2.2 Das Konzept von Dependency Injection.....	21
2.3 Das Framework	22
2.4 Die BeanFactory und der ApplicationContext.....	23
2.5 Dependency Injection mit Spring 2	24
2.6 Spring-Konfiguration	25
2.6.1 Beispiel für eine Konfiguration.....	26
2.7 Zugriff auf Spring-Beans	27
2.8 Spring-Beans und Factories	28
2.9 Lebenszyklus einer Spring-Bean	29
2.10 Komplexe Datentypen übergeben	30
2.11 Autowiring	31
2.12 Geschäftsobjekte mit Annotations konfigurieren	32
2.13 Nur Vorteile?	33
2.14 Zusammenfassung	34
3. ServiceLocator	35
3.1 ServiceLocator.....	36
3.2 ServiceLocator-Interface	37
3.3 ServiceLocatorFactory.....	38
3.4 Spring-Konfiguration	39
3.5 Laden von Beans.....	40
4. Entwicklungswerkzeuge & die Praxis.....	41
4.1 XML-Editor.....	42
4.2 Spring-IDE für Eclipse.....	43
4.3 BeanDoc.....	46
4.4 Spring im Kreuzfeuer	47
4.5 Umsetzung in die Praxis	48
5. Aspect Oriented Programming (AOP).....	49
5.1 Konzept	50
5.2 Begriffe im Zusammenhang mit AOP	51
5.3 AOP - Aspect Oriented Programming	52
5.4 AOP - Übersicht.....	53
5.5 AOP mit Spring.....	54
5.6 Advice Interfaces	55
5.7 Interessante Advices	56

5.8 Aspekte implementieren	57
5.9 Beispiel eines Aspekts mit dem MethodInterceptor	58
5.10 Beispiel eines Aspekts mit ThrowsAdvice	59
5.11 Spring-Aspekte mit AspectJ-Pointcuts	60
5.12 Die AspectJ-Notation	61
5.13 Konfiguration von Aspekten mit XML-Schema	62
5.14 Zusammenfassung	63
6. Persistenz	64
6.1 Persistenz	65
6.2 Springs DAO-Unterstützung	66
6.3 Datenzugriff mit JDBC	67
6.4 Datenzugriffsschablonen	68
6.5 Datenbank-Exceptions	69
6.6 ORM - Datenzugriff	70
6.7 Spring und Hibernate 3	71
6.8 HibernateTemplate	72
6.9 DAOs auf Basis von Hibernate erstellen	74
6.10 Kontextabhängige Sessions in Hibernate	75
6.11 Kontextabhängige Sessions oder HibernateTemplate?	76
6.12 Zusammenfassung	77
7. Transaktionen	78
7.1 Transaktionen	79
7.2 Globale Transaktionen	80
7.3 Lokale Transaktionen	81
7.4 Transaktionsattribute	82
7.5 TransactionDefinition - Propagation	83
7.6 TransactionDefinition - Isolation	84
7.7 TransactionDefinition - Lesemodus, Timeout, Rollback	85
7.8 Deklarative Transaktionsverwaltung	86
7.9 XML-deklarierte Transaktionen	87
7.10 Deklaration über Annotations	89
7.11 Programmatisches Transaktionsmanagement	91
7.12 Spring-Transaktionsmanager	92
7.13 JDBC-Transaktionsmanager	93
7.14 Hibernate-Transaktionsmanager	94
7.15 PlatformTransactionManager	95
7.16 TransactionDefinition - Beispiel	96
7.17 Deklaratives TM vs. Programmatisches TM	97
7.18 Transaktionen, Spring und die Praxis	98
7.19 Zusammenfassung	99
8. Integration	100
8.1 Spring, EJBs und ihr Doppelleben	101
8.2 EJB-Integration mit Spring	102
8.3 Was spricht für einen Application-Server?	103
8.4 Was spricht gegen einen Application-Server?	104
8.5 Spring oder EJBs?	105
8.6 Spring und EJBs	106
8.7 Wiring von EJBs in Spring	107
8.8 Spring EJB 3 und Pitchfork	109

8.9 Fazit.....112