

HANSER



Craig Walls

Spring im Einsatz

ISBN: 978-3-446-42388-6

Weitere Informationen oder Bestellungen unter

<http://www.hanser.de/978-3-446-42388-6>

sowie im Buchhandel.

Register

Symbole

^(Operator) 59
-(Operator) 59
!(Operator) 61
?: (Operator) 61
[] (Operator) 64
*(Operator) 59
/(Operator) 59
&& (Operator) 100
%(Operator) 59
+(Operator) 59
==(Operator) 60
|| (Operator) 100
> und < (Operatoren) 60
>= und <= (Operatoren) 60

A

Abhängigkeitsinjektion 6, 10, 18

- AspectJ 115
- Konstruktorinjektion 8
- mit Java-Konfiguration 89

Ablauf 211

- Ablaufdaten 220, 222
- absichern 236
- Definieren eines Basisablaufs 222
- Informationen sammeln 226, 231
- Scoping von Daten 221
- Transitionen 219, 220
- Variablen deklarieren 220
- Zustände 215, 218

Ablaufdaten 220, 222

- conversation (Geltungsbereich) 221
- flash (Geltungsbereich) 221
- flow (Geltungsbereich) 221
- request (Geltungsbereich) 221
- sammeln 226, 231
- Scoping 221
- view (Geltungsbereich) 221

Ablaufzustand

- Aktionszustände 216
- Endzustände 216, 218
- Entscheidungszustände 216, 217
- Subflow-Zustände 216, 217

- View-Zustände 216

Accept (Header) 312

access (Attribut) 248, 253

Acegi Security 240 *siehe auch* Spring Security

ACID, Definition 157

ActiveMQ 339

- Message-Broker, ConnectionFactory einrichten 339
- Nachrichtenendpunkt deklarieren 340

ActiveMQConnectionFactory, brokerURL (Eigenschaft) 340

addAttachment() (Methode) 394

addInline() (Methode) 395

Adobe AIR 27

Adobe Flex 27

Advice 93

- after-returning (Advice) 103
- <aop:advisor> (Element) 101
- Around-Advices 112
- Before-Advice-Methode 103
- Grundlagen 97
- in Java geschrieben 97
- Transaktionen 172

Advice-Beans 91

After-Advice 14

afterPropertiesSet() (Methode) 41

after-returning (Advice) 101, 103, 116

@AfterReturning (Annotation) 111

after-throwing (Advice) 101, 103

@AfterThrowing (Annotation) 111

Aktionszustand 216

algorithm (Eigenschaft) 383

Amazon S3 208

Amazon Simple Storage 208

and (Operator) 61

Anfrage, Lebenszyklus 176

Anhang in E-Mail einfügen 394

annotatedClasses (Eigenschaft) 145

Annotation

- AnnotationSessionFactoryBean 145
- annotationsgetriebene 172
- @Around 112
- Around-Advices 112
- @AspectJ 33, 110

- @Async 401
- Autodiscovery 84
- @Autowired 76, 147, 291, 393
- Autowiring aktivieren 76
- @Bean 88
- @Before 111
- @Component 28, 84, 182, 353
- @Configuration 87
- @Controller 84, 182, 193
- @DeclareParents 114
- @Entity 145
- @Inject 80, 182
- Introductions 114
- JSR-303 29
- @ManagedAttribute 366
- @ManagedOperation 366
- @ManagedResource 366
- @MappedSuperclass 145
- @MessageDriven 350
- @Named 81
- @PathVariable 200, 302
- @Pointcut 110
- @PostAuthorize 265
- @PostConstruct 28
- @PostFilter 266
- @PreAuthorize 265
- @PreFilter 264
- @Qualifier 28, 78
- @Repository 84, 147, 153
- @RequestBody 314
- @RequestMapping 181, 197, 300, 303, 305, 306, 307, 312
- @RequestParam 193, 207
- @ResponseBody 311, 314
- @ResponseStatus 306, 307
- @RolesAllowed 263
- @Scheduled 399, 400
- @Secured 263
- @Service 85
- @Size 202
- Spring Expression Language 83
- @Transactional 153, 165, 173
- Transaktionen 172
- und Spring MVC 181
- @Valid 201, 207
- @Validation 199
- @Value 83, 379
- verschalten mit 76
- von Annotierungen gesteuerte MBeans 366, 367
- AnnotationAwareAspectJAutoProxyCreator (Klasse) 112
- AnnotationSessionFactoryBean (Klasse) 145
- Antwort
 - Metadaten extrahieren 319
 - Ressourcenzustand im Body der Antwort zurückgeben 311
- Anwendungskontext 18
 - über mehrere XML-Dateien aufteilen 191
- <aop:advisor> (Element) 101
- <aop:after> (Element) 101
- <aop:after-returning> (Element) 101
- <aop:after-throwing> (Element) 101
- <aop:around> (Element) 101
- <aop:aspect> (Element) 101, 104
- <aop:aspectj-autoproxy> (Element) 101, 112, 115
- AOP (Aspektorientierte Programmierung) 10, 15, 91, 165
 - Advices 93
 - AOP-Modul 23
 - Around-Advices 112
 - AspectJ 115
 - Aspekte 95
 - Aspekte deklarieren 101
 - Beispiel 12, 15
 - Einführung 92
 - Einweben von Aspekten 95
 - Grundlagen 92
 - in Spring 96
 - Joinpoints 94
 - Methoden absichern 262
 - Methoden-Joinpoints 98
 - Pointcuts 94
 - Spring AOP-Modul 23
 - Spring Security 240
 - Terminologie 93
 - Überblick 10
- <aop:before> (Element) 101
- <aop:config> (Element) 101, 103
- <aop:declare-parents> (Element) 101
- aop-Konfigurationsnamensraum 14
- aop (Namensraum) 33
- <aop:pointcut> (Element) 101
- <aop:pointcut> (Konfigurationselement) 104
- Apache CXF 290
- Apache Tiles 187
 - definition (Element) 197
 - Tile-Definitionen 188
- Apache Velocity 397
- ApplicationContext (Interface) 18, 98
- Application-Managed Entity-Manager 148, 149, 150
- Applikationskontext 10, 19
- Arithmetischer Operator 59
- Around-Advice 101
 - annotieren 112
 - @Around (Annotation) 112
 - deklarieren 104, 106
 - Vorteile gegenüber Before- und After-Advice 105
- @Around (Annotation) 112

- @Aspect (Annotation) 110
- @AspectJ 110
 - Annotation 33
 - Element 101
 - Parameter 113
- AspectJ 14, 28, 115
 - aop (Namensraum) 33
 - Aspekte injizieren 115
 - Filtern 86
 - im Vergleich mit Spring AOP 97
 - Pointcuts schreiben 98
- Aspekt 10, 15, 91, 95
 - Advices 93, 172
 - After-Advice 14
 - <aop:advisor> (Element) 101
 - <aop:after> (Element) 101
 - <aop:after-returning> (Element) 101
 - <aop:after-throwing> (Element) 101
 - <aop:around> (Element) 101
 - <aop:aspect> (Element) 101, 104
 - <aop:aspectj-autoproxy> (Element) 101, 112, 115
 - <aop:before> (Element) 101
 - <aop:config> (Element) 101, 103
 - <aop:declare-parents> (Element) 101
 - <aop:pointcut> (Element) 101
 - AspectJ 115
 - Before-Advice 14
 - Beispiel 12, 15
 - Cross-Cutting Concerns 11
 - einweben 95
 - erstellen 101
 - Grundlagen 92
 - injizieren 115
 - Introduction 108
 - Joinpoints 94
 - Pointcuts 94, 172
 - Proxy-Klasse 97
- Aspektorientierte Programmierung *siehe* AOP
- @Async (Annotation) 401
- Asynchrone Kommunikation 334
 - Entkopplung 338
 - garantierte Zustellung 339
 - JmsProxyFactoryBean 356
 - keine Wartezeit 338
 - Lingo 355
 - Unabhängigkeit vom Standort 338
 - Vorteile 337
- Asynchrone Methode 401
- Asynchrones Messaging 334
 - Java Message Service (JMS)
 - ActiveMQ einrichten 339
 - Architektur 335
- AsyncResult (Klasse) 402
- at inject 80 *siehe auch* @Inject (Annotation)
- AtomFeedHttpMessageConverter 313
- Attachment *siehe* Anhang
- Attribut
 - access 248, 253
 - arg-names 107
 - attributes 236
 - authentication-failure-url 245
 - authorities-by-username-query 257
 - auto-config 244
 - base-package 87
 - base-path 213
 - business-interface 390
 - cache 388
 - connectionFactory 351
 - content 198
 - cron 400
 - cssClass 204
 - data-source-ref 256
 - default-impl 109
 - defaultImpl 114
 - default-init-method 42
 - delegate-ref 109
 - destroy-method 41
 - else 217
 - expression 86, 217
 - factory-method 117
 - Factory-Methode 38
 - fixedDelay 400
 - fixedRate 400
 - flow-registry 213
 - group-authorities-by-username-query 257
 - group-search-base 259
 - group-search-filter 258
 - hash 259
 - header 312
 - id 34, 47
 - ignore-resource-not-found 380
 - ignore-unresolvable 380
 - implement-interface 109
 - init-method 41
 - jndi-name 130, 387, 390
 - key 52, 361
 - key-ref 52
 - Klasse 34
 - lazy-init 389
 - ldif 260
 - login 245
 - login-processing-url 245
 - logout-url 247
 - lookup-on-startup 388
 - Managed 360
 - mapping 179

- model 216
- name 171, 255
- objectName 367
- on 219
- on-attribute 219
- params 197
- password 255
- password-attribute 259
- path 203, 214
- path-type 248
- pattern 244, 248
- physicalName 341
- Pointcut 103
- pre-post-annotations 264
- primary 72
- properties-ref 380
- property 251
- proxy-interface 388
- ref 38, 103
- registration 368
- resource-ref 130, 387
- scope 39, 252
- secured-annotations 263
- start-state 224
- String (Attribut) 88
- system-properties-mode 380
- then 217
- to 219
- Transaktionsattribute 166
- type 86
- Typ-Matching 109
- über Interfaces definieren 365
- url 253
- use-expressions 248
- userPassword 259
- users-by-username-query 257
- user-search-base 259
- user-search-filter 258
- user-service-ref 255, 256
- value 37, 44, 52, 55, 114, 307
- value-ref 52
- view 216, 218
- audience (Beispiel) 101, 110, 113
- Auf Benachrichtigungen lauschen 374, 375
- Auflösung von Views 184
- Aufrufen von Remote-Methoden 369
- Ausdruck 54, 55
 - authentication 249
 - denyAll 249
 - hasAnyRole(list of roles) 249
 - hasIpAddress(IP Address) 249
 - hasRole(role) 249
 - isAnonymous() 249
 - isAuthenticated() 249
 - isFullyAuthenticated() 249
 - isRememberMe() 249
 - permitAll 249
 - principal 249
- Ausloggen 247
- authentication (Ausdruck) 249
- authentication-failure-url (Attribut) 245
- Authentifizierung 251, 254
 - bei Datenbank 256, 257
 - Datenbank 256
 - LDAP Authentifizierungsmanager 258
 - Lightweight Directory Access Protocol 258, 261
 - mit speicherresidentem User-Repository 255
 - Remember-me-Feature 261
- Authentifizierungsmanager 258
- authorities-by-username-query (Attribut) 257
- authorities (Eigenschaft) 251
- auto-config (Attribut) 244
- autodetect (Autowiring) 70, 74
- Autodiscovery 84
- @Autowired (Annotation) 76, 147, 291, 393
 - eigene Qualifier 79
 - @Inject 81
 - mehrdeutige Abhängigkeiten 78
 - optionales Autowiring 77
 - private (Schlüsselwort) 77
 - @Qualifier (Annotation) 78
- @Autowired-Annotation
 - Eigenschaften annotieren 77
 - und Konstruktoren 77
 - und Setter-Methoden 77
- Autowiring
 - autodetect 70
 - Autodetect 74
 - @Autowired (Annotation) 76
 - best-fit 70, 74
 - byname 70
 - default-autowire (Eigenschaft) 74
 - Eigenschaft 70
 - autodetect 70, 74
 - byName 70, 71
 - byType 70, 72
 - constructor 70, 73
 - @Inject (Annotation) 80
 - JAX-WS-Endpunkte 291
 - mehrdeutige Abhängigkeiten 78
 - mit Annotationen 76
 - mit Annotationen aktivieren 76
 - mit expliziter Verschaltung mischen 75
 - nach Konstruktor 70, 73
 - nach Typ 70
 - namensbasiertes 71

- null 75
- optionales 77
- @Qualifier (Annotation) 78
- standardbasiertes 80
- typenbasiertes 72

B

- base-package (Attribut) 87
- base-path (Attribut) 213
- Basic-Authentifizierung 247
- BasicDataSource
 - driverClassName (Eigenschaft) 131
 - initialSize (Eigenschaft) 131
 - maxActive (Eigenschaft) 131
 - maxIdle (Eigenschaft) 131
 - maxOpenPreparedStatements (Eigenschaft) 131
 - maxWait (Eigenschaft) 131
 - minEvictableIdleTimeMillis (Eigenschaft) 131
 - minIdle (Eigenschaft) 131
 - password (Eigenschaft) 131
 - Pool-Konfigurationseigenschaften 131
 - poolPreparedStatements (Eigenschaft) 131
 - url (Eigenschaft) 131
 - username (Eigenschaft) 131
- Basisauthentifizierung 247
- Bean *siehe auch* JavaBean
 - Advice-Beans 91
 - als HTTP-Services exportieren 287
 - andere referenzieren 44
 - aus Anwendungskontext laden 19
 - Autodiscovery 84
 - Autowiring 70
 - Bean-Managed Persistence 147
 - deklarieren 32
 - durch Konstruktor injizieren 35
 - Element 36
 - Entity Beans 147
 - erstellen 32
 - Exportieren als MBeans 368
 - Funktionalitäten mit Hessian und Burlap verfügbar machen 282
 - Geltungsbereich 39, 40
 - Hibernate-Session-Factorys 144
 - in Eigenschaften injizieren 42
 - initialisieren und zerstören 40
 - injizieren durch Konstruktoren 39
 - innere 46
 - innere injizieren 46
 - Konfiguration externalisieren 377
 - konfigurieren 32
 - Lebenszyklus 19
 - LocalContainerEntityManagerFactoryBean 149, 150
 - LocalEntityManagerFactoryBean 149
 - mit Factory-Methode erstellen 38, 39
 - nach ID referenzieren 56
 - nachrichtengetriebene 349
 - Namensraum 33
 - Prototyp 39
 - Singletons 39
 - Standalone JAX-WS-Endpunkte exportieren 292
 - verschalten 31, 67
 - @Bean (Annotation) 88
 - BeanFactory 18
 - Interfaces 18
 - Bean-Managed Persistence 147
 - BeanNameUrlHandlerMapping 181
 - BeanNameViewResolver 185
 - bean() (Pointcut-Bezeichner) 100
 - <beans> (Element) 33
 - BeanShell, mit Namensraum lang 33
 - BeansSingleton-Klassen 39
 - Bedingter Operator 61
 - Before-Advice 14
 - Before-Advice-Methode 103
 - @Before (Annotation) 111
 - Before- und After-Advice deklarieren 102
 - Beispiel
 - audience 110, 113
 - City 63
 - contestant 109, 114
 - Ein-Mann-Band 49
 - instrumentalist 42, 71, 72, 75, 76, 78, 79, 85
 - judge 116
 - juggler 88, 89
 - Juggler 34, 39
 - knife juggler 81
 - knight 7, 15
 - MindReader 106
 - Pizza-Bestellung 222
 - Definieren des Basisablaufs 222
 - poetic juggler 73
 - Poetischer Juggler 36, 38
 - Spring Idol 32, 67
 - stage 38
 - Ticket 40
 - Benachrichtigung 373, 374, 375
 - Benannter Parameter 139, 140
 - Berechtigung 252
 - Best-fit Autowiring 70, 74
 - BMP 147 *siehe auch* Bean-Managed Persistence
 - Boilerplate-Code eliminieren 15
 - brokerURL (Eigenschaft) 340
 - BufferedImageHttpMessageConverter 313
 - Burlap
 - auf Services zugreifen 285
 - Bean-Funktionalitäten verfügbar machen 282

- im Vergleich mit Hessian 282
 - Services exportieren 284
 - BurlapServiceExporter (Klasse) 284
 - business-interface (Attribut) 390
 - byName (Autowiring) 70
 - ByteArrayHttpMessageConverter 313
 - byType (Autowiring) 70
- C**
- cache (Attribut) 388
 - Caching von JNDI-Objekten 388
 - Callback-Methode 6
 - Callbacks 126
 - Catch-Block 124, 125
 - Caucho *siehe* Hessian
 - CCC (Cross-Cutting Concern) 11, 92
 - CCI 159 *siehe auch* Common Client Interface
 - CciDaoSupport Klasse 129
 - CciLocalTransactionManager 159
 - CciTemplate 127
 - Central Authentication System (CAS) 254
 - City (Beispiel) 63
 - ClassPathXmlApplicationContext 10, 18
 - Cloud-Speicherung, Amazon S3 208
 - CMP (Container-Managed Persistence) 147
 - CMT (Container-Managed Transaction) 158
 - Collection
 - auf Member mit SpEL zugreifen 63
 - Eigenschafts-Collections 52
 - <list> (Element) 49
 - Map-Collections 51
 - <map> (Element) 49
 - Member auswählen 65
 - Projektion 65
 - <props> (Element) 49
 - Selektion und Projektion 66
 - <set> (Element) 49
 - und Spring Expression Language 63
 - verschalten 48
 - commit() (Methode) 160, 161, 163
 - Common Client Interface 159
 - CommonsMultipartResolver 209
 - @Component (Annotation) 28, 84, 182, 353
 - @Configuration (Annotation) 87
 - connectionFactory (Attribut) 351
 - connectionFactory (Eigenschaft) 345, 354, 355, 356
 - ConnectionFactory, Java Message Service (JMS) 339
 - ConnectorServerFactoryBean 369, 370
 - constructor (Autowiring) 70
 - Container 17, 32
 - Abhängigkeitsinjektion 18
 - Container-Managed Entity-Manager 148
 - Container-Managed Persistence 147
 - Kern-Container 22
 - Lebenszyklus einer Spring-Bean 20
 - Container-Managed Entity-Manager 148, 150, 152
 - Container-Managed Persistence 147
 - ContainerSpring, Container 17
 - Containerverwaltete Transaktion 158
 - ContentNegotiatingViewResolver 185, 308, 309
 - View finden 310
 - Wahl der Medientypen beeinflussen 310
 - contestant (Beispiel) 109, 114
 - ContextLoaderListener (Interface) 191, 192
 - context (Namensraum) 366
 - <context:property-placeholder> 378
 - Element 380
 - Controller
 - Controller schreiben in Spring MVC 192
 - Front-Controller 177
 - Handler-Mappings 181
 - Hessian-Controller 284
 - Input bearbeiten 192
 - Nachrichtenkonvertierer 311
 - nicht-REST-konformer 300
 - REST-konformer 299
 - Spitter Homepage-Controller 182
 - Spring MVC 177
 - testen 183
 - @Controller (Annotation) 84, 182
 - ControllerBeanNameHandlerMapping 181
 - ControllerClassNameHandlerMapping 181
 - conversation (Geltungsbereich) 221
 - convertJmsAccessException() (Methode) 348
 - CosMailSenderImpl (Klasse) 391
 - createContainerEntityManagerFactory() (Methode) 148
 - createEntityManagerFactory() (Methode) 148
 - createMessage() (Methode) 346
 - create() (Methode) 390
 - createMimeMessage() (Methode) 394
 - credentials (Eigenschaft) 251
 - cron (Attribut) 400
 - Cron-Ausdruck 400
 - Cross-Cutting Concern 11 *siehe auch* CCC
 - CRUD-Operation 304
 - cssClass (Attribut) 204
- D**
- DAO 122 *siehe auch* Datenzugriffsobjekt
 - DaoAuthenticationProvider (Schnittstelle) 256
 - DAO-Support-Klasse 128
 - CciDaoSupport 129
 - JdbcDaoSupport 129
 - JdoDaoSupport 129
 - JpaDaoSupport 129
 - NamedParameterJdbcDaoSupport 129

- SimpleJdbcDaoSupport 129
 - SqlMapClientDaoSupport 129
 - und JDBC 140, 141
 - DataAccessException 125
 - dataSource (Eigenschaft) 141, 145, 150, 160, 387
 - data-source-ref (Attribut) 256
 - DataSourceTransactionManager (Klasse) 159, 160
 - DataSource von JNDI auslesen 385
 - Datei
 - Datei-Uploads 205
 - im Dateisystem abspeichern 207
 - in Amazon S3 speichern 208
 - Spring für Datei-Upload konfigurieren 209
 - Dateisystem, Dateien hochladen 207
 - Datenbank 121, 154
 - Authentifizierung 256, 257
 - Datensatz mit JDBC abfragen 135
 - Datensatz mit JDBC aktualisieren 134
 - Verbindung aufbauen 378
 - Datenbankeigenschaft 151
 - Datenpersistenz 154
 - Bean-Managed Persistence 147
 - Container-Managed Persistence 147
 - JDBC 133, 136, 141
 - mit der Java Persistence API 147, 154
 - mit Hibernate 144, 146
 - Schablonen 126
 - Templates 127
 - Datenquelle
 - BasicDataSource 130
 - gepoolt 130, 131
 - JDBC-Treiber-basiert 132
 - JNDI 129
 - mit SimpleJdbcTemplate 137
 - Datenzentrierung 338
 - Datenzugriff 121, 154
 - Callbacks 126
 - Datenquellen 129, 132
 - Ebenen 122
 - Exception-Hierarchie 123, 125
 - JDBC 133, 141
 - JDBC-Treiber-basierte Datenquelle 132
 - mit SimpleJdbcTemplate 137, 139
 - Schablonen 126
 - Templates 126, 127
 - unübersichtlicher Code 133, 136
 - Datenzugriffsebene 122
 - Datenzugriffsobjekt 122
 - Hibernate mit DAO-Klassen integrieren 146, 147
 - JPA-basiert 152
 - Supportklassen 128
 - Supportklassen mit JDBC 140, 141
 - @DeclareParents (Annotation), defaultImpl (Attribut) 114
 - DefaultAnnotationHandlerMapping 181
 - default-autowire (Eigenschaft) 74
 - default-destroy-method (Attribut) 42
 - Default-Endpoint einrichten 347
 - defaultImpl (Attribut) 114
 - DefaultMethodSecurityExpressionHandler (Klasse) 268
 - defaultView (Eigenschaft) 311
 - definitions (Eigenschaft) 188
 - Deklarieren
 - Around-Advice 104
 - Aspekte 101
 - Transaktionen 165
 - Transaktionen in XML 170
 - von Beans 32
 - von Variablen im Ablauf 220
 - delegate-ref (Attribut) 109
 - DelegatingFilterProxy (Klasse) 243
 - Delegation im Vergleich mit Aspekten 93
 - delete() (Methode) 316, 322
 - DELETE (Methode) 305
 - denyAll (Ausdruck) 249
 - Dependency Injection (DI)
 - Spring Security 240
 - und JNDI 386
 - Dependency Injection for Java (Spezifikation) 80
 - Designkonzept, Schablonenmethode 126
 - destination (Eigenschaft) 355, 356
 - destroy() (Methode) 20
 - details (Eigenschaft) 251
 - Dirty Read 168
 - DispatcherServlet 177, 184
 - Anwendungskontext aufteilen 191
 - Anwendungskontext laden 191
 - auf / mappen 178
 - für Datei-Upload konfigurieren 209
 - Handler-Mappings 181
 - HiddenHttpMethodFilter 330
 - Konfiguration 191
 - konfigurieren 178
 - mit Spring Web Flow 212
 - und HessianServlet 284
 - DisposableBean (Interface) 41
 - doInTransaction() (Methode) 164
 - driverClassName (Eigenschaft) 131
 - DriverManagerDataSource (Klasse) 132, 389
 - DRY-Prinzip (Don't Repeat Yourself) 104
- ## E
- Eager Fetching 142
 - EclipseLinkJpaVendorAdapter 151
 - Eigenschaft
 - algorithm 383
 - annotatedClasses 145

- authorities 251
- Autowiring 70
- Collections 48
- connectionFactory 345, 354, 355, 356
- credentials 251
- dataSource 141, 145, 150, 387
- Datenbank 151
- default-autowire (Eigenschaft) 74
- defaultView 311
- definitions 188
- destination 355, 356
- details 251
- Eigenschafts-Collections 52
- externalisieren 377
- extern verschlüsseln 382
- favorParameter 310
- favorPathExtension 310
- fehlende 379
- hibernateProperties 145
- host 391
- ignoreAcceptHeader 310
- injizieren 38, 42
- jdbcTemplate 141
- jdbc.url 379
- jpaVendorAdapter 151
- location 383
- mappingResources 145
- mediaTypes 309
- metadataStrategy 356
- mit @Autowired annotieren 77
- mit @Named qualifizieren 81
- nach ID referenzieren 56
- namespaceUrl 295
- null verschalten 53
- objectName 372
- packagesToScan 145
- password 383, 392
- persistenceUnitName 150
- p (Namensraum) 48
- port 391
- portName 295
- principal 251
- Property-Overrider 381
- PropertyPlaceholderConfigurer 378
- proxyInterface 372
- queueName (Eigenschaft) 354
- registrationBehaviorName 368
- registryHost 278
- registryPort 278
- server 361, 372
- service 355
- serviceInterface 280, 289, 355, 356
- serviceName 295
- serviceUrl 280, 289, 369
- session 392
- Systemeigenschaften 380
- systemEnvironment 64
- systemProperties 64
- useNotAcceptableStatusCode 311
- username 392
- velocityProperties 397
- verschiedene Bedeutungen des Begriffs 53
- Werte injizieren 43
- wsdlDocumentUrl 295
- Ein-Mann-Band (Beispiel) 49
- Einweben von Aspekten 95
- EJB 2.x, nachrichtengetriebene Bean 349
- EJB 3, nachrichtengetriebene Bean 349
- EJB (Enterprise JavaBeans) 3
- Element
 - action-state 217
 - ActiveMQ konfigurieren 340
 - After-Advice 101
 - <amq:queue> (Element) 341
 - <amq:topic> (Element) 341
 - <aop:advisor> (Element) 101
 - <aop:after> (Element) 101
 - <aop:after-returning> (Element) 101, 103
 - <aop:after-throwing> (Element) 101, 103
 - <aop:around> (Element) 101
 - <aop:aspect> (Element) 101, 104, 112
 - <aop:aspectj-autoproxy> (Element) 101, 112, 115
 - <aop:before> (Element) 101
 - <aop:config> (Element) 101
 - <aop:declare-parents> (Element) 101
 - <aop:pointcut> (Element) 101
 - authentication-manager (Element) 255, 258
 - authentication-provider (Element) 256, 258
 - Beans referenzieren 45
 - Before-Advice 101
 - <c:forEach> (Element) 195
 - constructor-arg (Element) 341
 - <context:component-scan> (Element) 182
 - <context:mbean-export> (Element) 366, 368
 - <context:mbean-server> 361
 - <context:property-placeholder> 378, 380
 - dataSource (Eigenschaft) 160
 - decision-state 217, 228
 - default-destroy-method 42
 - default-init-method (Attribut) 42
 - definition (Element) 197
 - destroy-method (Attribut) 41
 - end-state 218, 234, 235
 - entityManagerFactory (Eigenschaft) 162
 - evaluate 217, 220, 225, 229
 - Factory-Methode 38

- filter (Element) 243
- Filtern 85
- filter-name (Element) 243
- <flow:flow-executor> (Element) 213
- <flow:flow-registry> (Element) 213
- global-method-security (Element) 262, 268
- global-transition 220
- http (Element) 244
- id Attribut 34
- if 217
- init-method (Attribut) 41
- innere Beans 47
- input 218, 225, 232, 234
- input (Element) 205
- jdbc-user-service (Element) 256, 257
- <jee:jndi-lookup> (Element) 386
- <jms:listener-container> (Element) 351
- <jms:listener> (Element) 351
- key (Attribut) 52, 361
- key-ref (Attribut) 52
- Klassenattribut 34
- ldap-authentication-provider (Element) 258
- LocalContainerEntityManagerFactoryBean 150
- LocalEntityManagerFactoryBean 150
- null verschalten 53
- on-entry 233
- p (Namensraum) 48
- primary (Attribute) 72
- protect-pointcut (Element) 269
- ref (Attribut) 38, 56
- resource-ref Attribut 130
- scope (Attribut) 39
- <security:authentication> (Element) 251
- <security:authorize> (Element) 252, 253
- servlet-name (Element) 178
- set 221
- <sf:checkbox> (Element) 199
- <sf:form> (Element) 199, 328
- <sf:password> (Element) 199
- start-state (Attribut) 224
- subflow-state 217, 225
- <s:url> (Element) 190
- <task:annotation-driven> (Element) 399
- transactionManagerName (Eigenschaft) 162
- transition 219
- und MBeans 360
- user (Element) 255
- user-service (Element) 255
- <util:properties> (Element) 381
- value (Attribut) 37, 44, 52, 55
- var 220
- view-state 216
- Werte injizieren 43
- <wsdl:port> (Element) 295
- <wsdl:service> (Element) 295
- else (Attribut) 217
- Elvis-Operator 62
- E-Mail
 - Anhänge einfügen 394
 - Konstruktion 393
 - Mail-Sender 391
 - Mail-Sessions 392
 - mit reichhaltigem Inhalt 395
 - versenden 391
 - Vorlagen erstellen 396
- Endpunkt (destination)
 - Autowiring von JAX-WS-Endpunkten 291
 - definierter 335
 - JAX-WS 291
 - Standalone JAX-WS-Endpunkte exportieren 292
 - Standard setzen 347
 - Thema (topic) 336
- Endzustand 216, 218
- Enge Kopplung 7, 385
- Enterprise JavaBeans (EJB) 3
 - erzwingen die Verwendung ihrer Klassen oder Schnittstellen 5
 - jee (Namensraum) 33
 - stateless 390
 - verschalten 390
 - Versionen 148
- Enterprise-Services
 - E-Mail-Versand 391
 - JMX
 - MBeans als Spring-Beans exportieren 360
 - Verwaltung von Spring-Beans 359
 - Konfiguration, Mail-Sender 391
- @Entity (Annotation) 145
- Entity Beans 147
- Entity-Manager
 - application-managed 148, 149, 150
 - container-managed 148, 150, 152
- entityManagerFactory (Eigenschaft) 162
- EntityManagerFactory (Interface) 148, 152, 162
- Entscheidungszustand 216, 217
- EnvironmentStringPBCEConfig 383
- Exception 98
 - Catch-Block 124, 125
 - DataAccessException 125
 - Datenzugriff-Exception-Hierarchie 123, 125
 - Hibernate 124
 - ImageUploadException 207
 - InstanceAlreadyExistsException 368
 - JDBC Exception-Hierarchie gegen Spring Exception-Hierarchie 124
 - JmsException 343

- JMSEException 343
- MalformedURLException 277
- NullPointerException 57, 75
- plattformunabhängig 124
- RemoteAccessException 275
- RemoteException 275, 277
- SQLException 16, 123, 134
- UnsupportedOperationException 268
- URISyntaxException 321
- exchange() (Methode) 316, 325
- execute() (Methode) 164, 316
- Export, MBeans als Spring-Beans 360
- expression (Attribut) 86, 217
- Extensible Markup Language (XML)
 - Aspekte deklarieren 101
 - konfigurieren Spring with 33
 - mit Annotationen reduzieren 76
 - mit Java-basierter Konfiguration reduzieren 87
 - Transaktionen deklarieren 170
- Externalisierung 377
- Externe Eigenschaft 382

F

- FactoryBean (Schnittstelle) 144
- factory-method (Attribut) 39, 117
- Factory-Methode
 - Attribut 38
 - Beans erstellen 39
- failOnExisting 368
- Fallback-Objekt 389
- favorParameter (Eigenschaft) 310
- favorPathExtension (Eigenschaft) 310
- FileSystemResource (Klasse) 394
- FileSystemXmlApplicationContext 18
- FilterChainProxy (Klasse) 243
- Filtern (Element) 85
- FilterToBeanProxy (Klasse) 243
- fixedDelay (Attribut) 400
- fixedRate (Attribut) 400
- flash (Geltungsbereich) 221
- Flow
 - Flow-Executor 213
 - Flow-Registry 213, 214
 - Flow-Requests 214
- Flow-Executor 213
- flow (Geltungsbereich) 221
- FlowHandlerAdapter (Klasse) 214
- FlowHandlerMapping (Klasse) 214
- Flow-Registry 213, 214
- flow-registry (Attribut) 213
- Flow-Request 214
- format (Parameter) 310
- FormHttpMessageConverter 313, 321

Formular

- Datei-Upload-Feld 205
- Eingaben validieren 201
- Inhaltstyp 205
- Pfadvariablen 200
- REST-konformes 327
- Validierungsfehler 202
- Validierungsregeln 201
- verarbeiten 196
- versteckte Methodenfelder 328
- View definieren 199
- Formulareingabe
 - Validierungsregeln 201
 - verarbeiten 199
- Formularverarbeitung
 - Datei-Uploads 205
 - Validierungsfehler 202
- FreeMarkerViewResolver 185, 187
- Front-Controller, DispatcherServlet 177
- Funktion, querschnittliche 11 *siehe auch* CCC
- Future (Interface) 402

G

- Geltungsbereich 40, 221
 - Beans 39
- Gemini Blueprint 26
- Gepoolte Datenquelle 130, 131
- getAttribute() (Methode) 371
- getConnection() (Methode) 128
- getFirst() (Methode) 319
- getForEntity() (Methode) 316, 319
- getForObject() (Methode) 316, 318
- getHeaders() (Methode) 319
- getInstance() (Methode) 39
- getJdbcTemplate() (Methode) 128
- getLastModified() (Methode) 319
- get() (Methode) 319
- GET (Methode) 305, 317
- getObject() (Methode) 348
- getSimpleJdbcTemplate() (Methode) 141
- getStatusCode() (Methode) 320
- Globale Transition 219
- Groovy, mit Namensraum lang 33
- group-authorities-by-username-query (Attribut) 257
- group-search-base (Attribut) 259
- group-search-filter (Attribut) 258

H

- hasAnyRole(list of roles) (Ausdruck) 249
- hash (Attribut) 259
- hasIpAddress(IP Address) (Ausdruck) 249
- hasPermission() (Methode) 267
- hasRole() (Ausdruck) 253

- hasRole(role) (Ausdruck) 249
 - Header, Accept 312
 - headers, Accept 326
 - headers (Attribut) 312
 - headForHeaders() (Methode) 316
 - HEAD (Methode) 305
 - Hessian
 - auf Services zugreifen 285
 - Bean-Funktionalitäten verfügbar machen 282
 - Controller konfigurieren 284
 - im Vergleich mit Burlap 282
 - Services exportieren 283
 - HessianProxyFactoryBean 285
 - HessianServiceExporter (Klasse) 283
 - Hibernate
 - direkte Verknüpfung mit DAO-Klassen 146, 147
 - Eager Fetching 142
 - @Entity (Annotation) 145
 - Exception-Hierarchie 124
 - HibernateTemplate 127
 - Integration mit Spring 142, 147
 - Kaskaden 142
 - kontextbezogene Sessions 143
 - Lazy Loading 142
 - Schablonenklassen ausschließen 146, 147
 - Session-Factory deklarieren 144, 146
 - Transaktionen 159, 161
 - Überblick 143
 - und JNDI 387
 - HibernateDaoSupport (Klasse) 129
 - HibernateJpaVendorAdapter 151
 - Hibernate JPA Vendor-Adapter 151
 - hibernateProperties (Eigenschaft) 145
 - HibernateTemplate (Klasse) 127, 143
 - HibernateTransactionManager (Klasse) 159, 161
 - HiddenHttpMethodFilter 329
 - Hintergrundaufgabe 399
 - host (Eigenschaft) 391
 - HTTP-Client von Jakarta Commons 315
 - HttpInvoker 286 *siehe auch* HTTP-Invoker
 - HTTP-Invoker
 - auf Services zugreifen via HTTP 288
 - Beans als HTTP-Services exportieren 287
 - HttpInvokerServiceExporter 287
 - HttpMethod (Parameter) 325
 - HTTPS 249 *siehe auch* Hypertext Transfer Protocol Secure
 - HttpServletRequest 242
 - Hypertext Transfer Protocol
 - Methoden 304
 - Nachrichtenkonvertierer 311
 - Hypertext Transfer Protocol Secure 249, 250
- I**
- iBatis
 - SQL Maps 23
 - Transaktionen 159, 160
 - id (Attribut) 34, 47
 - Idempotenz 304
 - ignoreAcceptHeader (Eigenschaft) 310
 - ignoreExisting 368
 - ignore-resource-not-found (Attribut) 380
 - ignore-unresolvable (Attribut) 380
 - IllegalArgumentException 98
 - image (Parameter) 207
 - ImageUploadException 207
 - implement-interface (Attribut) 109
 - Inhalt 178
 - Initialisieren, Beans 40
 - Initialization on Demand Holder 39
 - InitializingBean (Interface) 41
 - initialSize Eigenschaft 131
 - init-method 20
 - @Inject (Annotation) 80, 81, 182
 - Injektion
 - AspectJ-Aspekte 115
 - durch Konstruktoren 35, 36, 38, 39
 - innere Beans injizieren 46
 - Setter-Injektion 42
 - Werte injizieren 43
 - Innere Beans 46
 - Input
 - Controller 192
 - Datei-Uploads 205
 - Installieren von Spring Web Flow 212, 215
 - InstanceAlreadyExistsException 368
 - instrumentalist (Beispiel) 42, 71, 72, 75, 76, 78, 79, 85
 - Integration, Hibernate mit DAO-Klassen 146, 147
 - Interface
 - ApplicationContext 18, 98
 - ContextLoaderListener 191, 192
 - DataSource 160
 - Definieren von MBean-Operationen und -Attribute über 365
 - DisposableBean 41
 - EntityManagerFactory 162
 - Future 402
 - InitializingBean 41
 - java.util.Collection 49, 50
 - java.util.Map 49, 51
 - java.util.Properties 49, 52, 64
 - java.util.Set 51
 - JpaDialect 162
 - Kodieren 46
 - lose Kopplung 46
 - MailSender 391

- MailSession 392
 - MessageCreator 346
 - MessageListener 350
 - MultipartResolver 209
 - NotificationListener 374
 - NotificationPublisherAware 373
 - org.hibernate.Session 144
 - Provider 81
 - Remote 277
 - Serializable 276
 - Transaction 161
 - TransactionCallback 164
 - TransactionDefinition 166
 - TransactionManager 163
 - UserTransaction 163
 - InterfaceBasedMBeanInfoAssembler 365
 - InternalResourceViewResolver 185, 186
 - Introduction 108
 - Annotationen 114
 - invoke() (Methode) 371
 - isAnonymous() (Ausdruck) 249
 - isAuthenticated() (Ausdruck) 249
 - isFullyAuthenticated() (Ausdruck) 249
 - ISOLATION_DEFAULT (Isolationsebene) 169
 - ISOLATION_READ_COMMITTED (Isolationsebene) 169
 - ISOLATION_READ_UNCOMMITTED (Isolationsebene) 169
 - ISOLATION_REPEATABLE_READ (Isolationsebene) 169
 - Isolationsebene 168, 169, 171
 - ISOLATION_SERIALIZABLE (Isolationsebene) 169
 - isRememberMe() (Ausdruck) 249
- J**
- J2EE Connector Architecture 159
 - Jakarta Commons
 - Database Connection Pooling 130, 131
 - REST-Client, HTTP-Client 315
 - JAR-Datei 22
 - JasperReportsViewResolver 185, 187
 - Jasypt 378, 382, 383
 - Java
 - Abhängigkeitsinjektion und Java-basierte Konfiguration 89
 - Entwicklung vereinfachen 4
 - jee (Namensraum) 33
 - jms (Namensraum) 33
 - Math-Klasse 57
 - mit Spring konfigurieren 87
 - POJOs 5, 6
 - Java-API,Boilerplate-Code 15
 - JavaBean *siehe auch* Bean
 - Abhängigkeitsinjektion 6, 10
 - Aspekte 10
 - Autodiscovery 84
 - deklarieren 32
 - EJB 3
 - Entity Beans 147
 - Grundlagen 3
 - Lebenszyklus 19
 - POJOs 5, 6
 - Spezifikation 3
 - Java Community Process, JSR-330 80
 - Java Database Connectivity 23 *siehe auch* JDBC (Java Database Connectivity)
 - Java Data Objects 148, 159
 - Java-Entwicklung durch Spring vereinfachen 4
 - java.io.File (Klasse) 207
 - JavaMailSenderImpl (Klasse) 391
 - Java Management Extensions 375
 - Benachrichtigungen 373, 375
 - JMX Messaging Protocol 369
 - JSR-160 369
 - und Remoting 369, 372
 - Java Message Service (JMS) 334, 341
 - Architektur 335
 - Default-Endpunkt einrichten 347
 - Einführung 334
 - JMS-Invoker 352
 - jms (Namensraum) 33
 - JmsTemplate 341
 - JmsTemplate102 344
 - nachrichtengetriebene POJOs 349
 - Remote Procedure Call (RPC) 352
 - Senden/Empfangen über konventionelles (nicht-Spring) 342
 - Templates, Nachrichten empfangen 347
 - Thema (topic) 336
 - Transaktionen 159
 - unübersichtlicher Code 342
 - Warteschlangen 336
 - Java Naming and Directory Interface (JNDI) 384, 386
 - Caching von JNDI-Objekten 388
 - EJBs verschalten 390
 - Fallback-Objekte 389
 - konventionelles 385
 - Lazy Loading 388
 - Mail-Sessions 392
 - Java Persistence API 147, 154
 - application-managed Entity-Manager 149, 150
 - container-managed Entity-Manager 150, 152
 - Hibernate JPA Vendor-Adapter 151
 - JPA-basierte Datenzugriffsobjekte 152
 - Transaktionen 159, 161
 - JavaServer Pages (JSP) 184
 - DispatcherServlet 177
 - Spring Security 241

- Tags für Spring Security 250
 - versteckte Methodenfelder 328
 - Java Transaction API 159, 162 *siehe auch* JTA
 - java.util.Collection (Interface) 49, 50
 - java.util.Map (Interface) 49, 51
 - java.util.Map (Namensraum) 64
 - java.util.Map (Schnittstelle) 140
 - java.util.Properties (Interface) 49, 52, 64
 - java.util.Set (Interface) 51
 - javax.inject (Package) 82
 - javax.sql.DataSource (Schnittstelle) 137
 - Jaxb2RootElementHttpMessageConverter 313
 - JAX-RPC 290
 - JAX-WS
 - Endpunkte 291
 - Proxy-Services auf Client-Seite 294
 - JaxWsPortProxyFactoryBean 294
 - JBoss 129
 - JCA 159 *siehe auch* J2EE Connector Architecture
 - JConsole und JMX 360
 - JDBC-Code, unübersichtlicher 133
 - JdbcDaoSupport Klasse 129, 140
 - JDBC (Java Database Connectivity) 23, 133, 141
 - Abfragen von Reihen in Datenbanken 135
 - Boilerplate-Code 15, 135
 - Datensatz in der Datenbank aktualisieren 134
 - JDBC Exception-Hierarchie gegen Spring Exception-Hierarchie 124
 - JdbcTemplate 127
 - SQLException 123
 - Templates 136
 - Transaktionen 159, 160
 - Treiber-basierte Datenquellen 132
 - und DAO-Support-Klassen 140, 141
 - unübersichtlicher Code 133, 136
 - JdbcTemplate 16, 127, 136
 - jdbcTemplate (Eigenschaft) 141
 - jdbc.url (Eigenschaft) 379
 - JDO 148 *siehe auch* Java Data Objects
 - JdoTemplate 127
 - JdoTransactionManager 159
 - jee (Namensraum) 33, 130
 - JMS-Code, unübersichtlicher 342
 - JmsException 343, 344
 - JMSException 343
 - JMS-Invoker 352
 - JmsInvokerProxyBean 354
 - JmsInvokerProxyFactoryBean 354
 - JmsInvokerServiceExporter 352, 354
 - jms (Namensraum) 33, 351
 - JmsServiceExporter 355, 356
 - JmsTemplate 341
 - Nachrichten verarbeiten 347
 - Nachrichten versenden 345
 - receive() (Methode) 347, 349
 - send() (Methode) 346, 347
 - Standard-Endpoint setzen 347
 - verschalten 345
 - Verschalten von Default-Endpoint 347
 - JmsTemplate102 344
 - JmsTransactionManager 159
 - JmsUtils (Klasse) 347
 - JMX 360 *siehe auch* Java Management Extensions
 - MBeans als Spring-Beans exportieren 360
 - Verwaltung von Spring-Beans 359
 - JMXConnectorServer 369
 - JMX Messaging Protocol 369
 - JMXMP 369 *siehe auch* JMX Messaging Protocol
 - JNDI 384
 - Datenquellen 129
 - Fallback-Objekte 389
 - <jee:jndi-lookup> (Element) 130
 - konventionelle APIs 385
 - Lazy Loading von Objekten 388
 - Mail-Session 392
 - Objekte cachen 388
 - Objekte injizieren 386
 - Objekte suchen 385
 - und EntityManagerFactory 152
 - jndi-name (Attribut) 130, 387, 390
 - Johnson, Rod 4
 - Joinpoint 94
 - mit Pointcuts auswählen 98
 - JpaDaoSupport Klasse 129
 - JPA (Java Persistence API), Transaktionen 161
 - JpaTransactionManager (Klasse) 159, 161
 - jpaVendorAdapter (Eigenschaft) 151
 - JRuby, mit Namensraum lang 33
 - JSR-160 369
 - JSR-250, @RolesAllowed (Annotation) 263
 - JTA (Java Transaction API) 158, 162
 - JtaTransactionManager (Klasse) 159, 162
 - judge (Beispiel) 116
 - juggler (Beispiel) 34, 39, 88, 89
- ## K
- Kaskaden 142
 - Kern-Container 22
 - key (Attribut) 52, 361
 - key-ref (Attribut) 52
 - Klasse
 - AnnotationAwareAspectJAutoProxyCreator 112
 - AnnotationSessionFactoryBean 145
 - AsyncResult 402
 - BurlapServiceExporter 284
 - CciDaoSupport 129

- CommonsMultipartResolver 209
- CosMailSenderImpl 391
- DAO-Support-Klassen 128
- DefaultMethodSecurityExpressionHandler 268
- DelegatingFilterProxy 243
- DriverManagerDataSource 132, 389
- FileSystemResource 394
- FilterChainProxy 243
- FlowHandlerAdapter 214
- FlowHandlerMapping 214
- HessianServiceExporter 283
- HibernateDaoSupport 129
- HibernateTemplate 143
- java.io.File 207
- JavaMailSenderImpl 391
- JdbcDaoSupport 129, 140
- JdbcTemplate 136
- JdoDaoSupport 129
- JmsUtils (Klasse) 348
- JpaDaoSupport 129
- Konfigurationsklassen definieren 88
- LinkedMultiValueMap 326
- LocalSessionFactoryBean 144
- Math 58
- MBeanExporter 360, 363
- MimeMessageHelper 394
- MultiValueMap 326
- NamedParameterJdbcDaoSupport 129, 140
- NamedParameterJdbcTemplate (Klasse) 136, 139
- Naming 279
- ParameterizedRowMapper 139
- PersistenceAnnotationBeanPostProcessor 153
- PersistenceExceptionTranslationPostProcessor 147
- ProviderManager 255
- RmiProxyFactoryBean 280
- RmiServiceExporter 278
- SimpleJdbcDaoSupport 129, 140
- SimpleJdbcTemplate 136, 137, 139, 140
- SimpleMetadataStrategy 356
- SingleConnectionDataSource 132
- SqlMapClientDaoSupport 129
- stringEncryptor 383
- TransactionTemplate 164
- Klassenattribut 34
- knife juggler (Beispiel), @Inject 81
- Kollisionen, MBean-Kollisionen 367, 368
- Komponente 10, 11, 15
- Konfiguration 209
 - auslagern 377
 - Datenquellen 129, 132
 - DispatcherServlet 191
 - eingebetteter LDAP-Server 260
 - externalisieren 377
 - Flow-Registry 213, 214
 - Hessian-Controller 284
 - Mail-Sender 391
 - Mail-Sender und Service-Bean 392
 - minimale Websicherheit 244
 - Property-Overrider 381
 - Property-Placeholder 378
 - RMI-Service 277
 - Spring mit Java 87
 - Spring Security 240
 - Spring Web Flow 212, 215
- Konfigurationsklasse 88
- Konfigurieren
 - Entity-Manager-Factories 148, 152
 - Message-Listener 351
 - Spring 32, 34
 - Spring mit XML 33
- Konstruktion, E-Mails 393
- Konstruktor
 - Autowiring 70, 73
 - Injektion 35
 - injizieren 37
 - injizieren durch 39
 - Konstruktorinjektion 8
 - Konstruktor-Pointcuts 115
 - Objektreferenzen injizieren 36, 38
- Konstruktorinjektion 8
- Kontextbezogene Sessions 143
- Kontext (Namensraum) 33
- Kopplung 7
 - enge 385
 - lose 6
- L**
- lang (Namensraum) 33
- lazy-init (Attribut) 389
- Lazy Loading 142, 388
- LDAP 258 *siehe auch* Lightweight Directory Access Protocol
- ldif (Attribut) 260
- Lebenszyklus von Beans 19
- le und ge (Operatoren) 60
- Lightweight Directory Access Protocol
 - Authentifizierung 258, 261
 - Authentifizierungsmanager 258
 - eingebetteter Server konfigurieren 260
 - Passwortvergleich 259
 - Remote-Server 260
- Lingo 355, 356
 - Service exportieren 355
- LinkedMultiValueMap (Klasse) 326
- LocalContainerEntityManagerFactoryBean 149, 150
- LocalEntityManagerFactoryBean 149

- LocalSessionFactoryBean (Klasse) 144
- location (Eigenschaft) 383
- login (Attribut) 245
- Login-Formular 244
- login-processing-url (Attribut) 245
- Logischer Operator 61
- logout-url (Attribut) 247
- lookup-on-startup (Attribut) 388
- Lose Kopplung 6, 46
- LTW (Load-Time Weaving) 95
- M**
- magician (Beispiel) 113
- Mail-Sender 391
 - konfigurieren 391
 - mit Service-Bean verschalten 392
- MailSender (Interface) 391
- Mail-Session 392
 - JNDI 392
 - Konfiguration, JNDI 392
- MailSession (Interface) 392
- MalformedURLException 277
- Managed (Attribut) 360
- @ManagedAttribute (Annotation) 366
- @ManagedOperation (Annotation) 366
- @ManagedResource (Annotation) 366
- Map 51
- @MappedSuperclass (Annotation) 145
- MappingJacksonHttpMessageConverter 313
- mappingResources (Eigenschaft) 145
- MarshallingHttpMessageConverter 313
- matches (Operator) 62
- Math-Klasse 58
- maxActive (Eigenschaft) 131
- maxIdle (Eigenschaft) 131
- maxOpenPreparedStatements (Eigenschaft) 131
- maxWait (Eigenschaft) 131
- MBeanExporter (Klasse) 360, 363
 - registrationBehaviorName (Eigenschaft) 368
 - server (Eigenschaft) 361
- MBean Info-Assembler 363, 365
 - InterfaceNameBasedMBeanInfoAssembler 365
 - MetadataMBeanInfoAssembler 366
 - MethodExclusionMBeanInfoAssembler 364
 - MethodNameBasedMBeanInfoAssembler 363
- MBeanProxyFactoryBean 371
- MBeans
 - Benachrichtigungen 373, 375
 - Definieren von Operationen und Attributen über Interfaces 365
 - dynamische 359
 - Exportieren von Remote-MBeans 369, 370
 - MBean-Server 360
 - Methoden über den Namen bereitstellen 363, 365
 - Model-MBeans 359
 - Open MBeans 359
 - Proxy 371
 - Proxying 372
 - Remoting 369, 372
 - Spring-Beans exportieren 360
 - Standard MBeans 359
 - Umgang mit Kollisionen 367, 368
 - von Annotierungen gesteuert 366, 367
 - Zugriff auf Remote-MBeans 370, 371
- MBean-Server 360
- MBeanServerConnection 370
- MBeanServerConnectionFactoryBean 370
- MBean-Server (MBean-Agent) 360
- MDPs 349 *siehe auch* Message-driven POJOs
- mediaTypes (Eigenschaft) 309
- Medientyp 309
- Mehrdeutige Abhängigkeit 78
- mergeTemplateIntoString() (Methode) 397
- Message-Broker
 - ActiveMQ 339
 - einrichten 339
- MessageCreator (Interface) 346
- @MessageDriven (Annotation) 350
- Message Driven Bean (MDB) 349
- Message-Listener (Interface) 350
 - erstellen 350
 - konfigurieren 351
- Messaging
 - ActiveMQ 339
 - asynchrones 334
 - Java Message Service (JMS), ActiveMQ einrichten 339
 - Remote Procedure Call (RPC), nachrichtenbasierte 352
 - Datenzentrierung 338
 - definierte Endpunkte 335
 - Java Message Service (JMS) 341
 - ActiveMQ einrichten 339
 - Architektur 335
 - konventionell (nicht-Spring) 342
 - Vorteile 337
 - Message-Listener 350
 - mit Java Message Service 334
 - nachrichtengetriebene POJOs (MDP) 349
 - Message-Listener 350
 - Publish-and-Subscribe-Modell 336
 - Standortunabhängigkeit 338
 - publish-subscribe 336
 - Publish-Subscribe-System 335
 - Punkt-zu-Punkt-Modell 336
 - Standortunabhängigkeit 338

- Queues 336
- Remote Procedure Call (RPC), nachrichtenbasierte 352
- Service exportieren 355
- synchrones 337
- Thema (topic) 336
- Warteschlange 336
- MetadataMBeanInfoAssembler 366
- metadataStrategy (Eigenschaft) 356
- method (Attribut) 103
- Methode
 - absichern 262
 - addAttachment() 394
 - addInline() 395
 - afterPropertiesSet() 20, 41
 - asynchrone 401
 - Callback-Methoden 6
 - commit() 160, 161, 163
 - convertJmsAccessException() 348
 - create() 390
 - createContainerEntityManagerFactory() 148
 - createEntityManagerFactory() 148
 - createMessage() 346
 - createMimeMessage 394
 - delete() 316, 322
 - DELETE 305, 306
 - destroy() 20
 - doInTransaction() 164
 - exchange() 316, 325
 - execute() 164, 316
 - Factory-Methode 38, 39
 - get() 319
 - GET 305
 - getAttribute() 371
 - getConnection() 128
 - getForEntity() 316, 319
 - getForObject() 316, 318
 - getInstance() 39
 - getJdbcTemplate() 128
 - getLastName() 319
 - getMethods() 319
 - getObject() 348
 - getSimpleJdbcTemplate() 141
 - getStatusCode 320
 - hasPermission() 267
 - HEAD 305
 - headForHeaders() 316
 - HiddenHttpMethodFilter 329
 - invoke() 371
 - mergeTemplateIntoString() 397
 - Methoden-Joinpoints 98
 - mit SpEL sichern 264
 - nach ID referenzieren 57
 - OPTIONS 305
 - optionsForAllow() 316
 - planen 399
 - POST 305, 306
 - post-authorizing 265
 - post-filtering 266
 - postForEntity() 316
 - postForLocation() 316, 324
 - postForObject() 316
 - postProcessAfterInitialization() 20
 - postProcessBeforeInitialization() 20
 - pre-authorizing 265
 - put() 316, 320
 - PUT 305
 - queryNames() 370
 - receive() 347
 - REST-Methoden 304
 - rollback() 160, 161, 163
 - Scheduling 400
 - send() 346, 347
 - setApplicationContext() 20
 - setAttribute() 371
 - setBeanFactory() 20
 - setBeanName() 20
 - setRollbackOnly() 164
 - Setter-Methode 44
 - Sicherheits-Pointcuts auf Methodenebene 269
 - toUpperCase() 57
 - TRACE 305
 - über den Namen bereitstellen 363, 365
 - versteckte Methodenfelder 328
- Methode.getHeaders() 319
- MethodExclusionMBeanInfoAssembler 364
- MethodNameBasedMBeanInfoAssembler 363
- MimeMessageHelper (Klasse) 394
- MindReader (Beispiel) 106
- minEvictableIdleTimeMillis (Eigenschaft) 131
- minIdle (Eigenschaft) 131
- Mockito 9, 184
- Model 177
- model (Attribut) 216
- Model View Controller (MVC) 23, 33, 176
 - siehe auch* MVC
- Modul 21, 24
 - AOP 23
 - DAO 23
 - JDBC 23
 - MVC 23
 - ORM 23
- MSMQ 26
- MultipartResolver (Interface) 209
- MultiValueMap (Klasse) 326
- MVC (Model View Controller) 23
- mvc (Namensraum) 33

N

Nachricht

- asynchrone 334
- mit JmsTemplate verarbeiten 347
- mit JmsTemplate versenden 345
- synchrone 334

Nachrichtenbasierte POJOs

- JmsInvokerProxyBean 354
- JmsInvokerProxyFactoryBean 354
- JmsInvokerServiceExporter 352, 354
- Remote Procedure Call (RPC) 352

Nachrichtsendpunkt 335

- für ActiveMQ deklarieren 340

Nachrichtengetriebene POJOs 349

- Message-Listener 350

Nachrichtenkonvertierer 311

name (Attribut) 255

@Named (Annotation) 81

NamedParameterJdbcDaoSupport (Klasse) 129, 140

NamedParameterJdbcTemplate (Klasse) 127, 136, 139

Namensbasiertes Autowiring 71

Namensraum

- aop 14, 33, 171
- Beans 33
- context 366, 378
- java.util.Map 64
- jee 33, 130
- jms 33, 351
- Kontext 33
- lang 33
- mvc 33, 179
- oxm 33
- p 48
- Spring Security Namensraum konfigurieren 241
- tx 33, 170
- util 33, 63

namespaceUrl (Eigenschaft) 295

NamingException 385

Naming (Klasse) 279

new, Schlüsselwort 19

Nonrepeatable Read 168

NotificationListener (Interface) 374

NotificationPublisherAware (Interface) 373

NullPointerException 57, 75

Nur-lesen-Transaktion 169

O

objectName (Attribut) 367

objectName (Eigenschaft) 372

Object Relational Mapping 23

siehe auch ORM

Object Request Broker 276

Object-zu-XML-Mapping 33

Objekt

- Caching 388
- Fallback 389
- injizieren 386
- Lazy Loading 388

Objektreferenz mit Konstruktor injizieren 36, 38

Objektrelationales Mapping 142

OC4 J-Container, Transaktionen 159

OC4JtaTransactionManager 159

on (Attribut) 219

on-exception (Attribut) 219

OpenID, Spring Security 241

OpenJpaVendorAdapter 151

Operator

- ^ 59
- - 59
- ! 61
- ?\ 61
- .^[] 65
- [] 64
- * 59
- / 59
- && 100
- % 59
- + 59
- == 60
- > 60
- >= 60
- || 100
- .\$[] 65
- and 61
- arithmetischer 59
- bedingter 61
- Elvis-Operator 62
- le und ge 60
- logischer 61
- matches 62
- not 61
- or 61
- reguläre Ausdrücke 62
- relationaler 60
- T() 57
- Vergleichsoperator 60
- optionsForAllow() (Methode) 316
- OPTIONS (Methode) 305
- ORB 276 *siehe auch* Object Request Broker
- org.hibernate.Session (Schnittstelle) 144
- org.springframework.jdbc.datasource (Paket) 132
- O/R-Mapping 23 *siehe auch* ORM
- ORM (Object Relational Mapping) 23
- or (Operator) 61
- OSGi Blueprint Container 26
- oxm (Namensraum) 33

- P**
- Package, javax.inject 82
 - packagesToScan (Eigenschaft) 145
 - Parameter
 - an Advices übergeben 106
 - benannt 139, 140
 - format 310
 - HttpMethod 325
 - in @AspectJ 113
 - ProceedingJoinPoint 105
 - ParameterizedRowMapper (Klasse) 139
 - params (Attribut) 197
 - Pareto-Prinzip 136
 - password (Attribut) 255
 - password-attribute (Attribut) 259
 - password (Eigenschaft) 131, 383, 392
 - Passwort mit LDAP vergleichen 259
 - path (Attribut) 203, 214
 - path-type (Attribut) 248
 - @PathVariable (Annotation) 200, 302
 - @Pattern (Annotation) 202
 - pattern (Attribut) 244, 248
 - Performance
 - tatsächliche 401
 - wahrgenommene 401
 - permitAll (Ausdruck) 249
 - PersistenceAnnotationBeanPostProcessor (Klasse) 153
 - PersistenceExceptionTranslationPostProcessor (Klasse) 147
 - PersistenceProvider (Schnittstelle) 148
 - persistenceUnitName (Eigenschaft) 150
 - persistence.xml-Datei 149
 - Persistenzebene, hinter Schnittstellen verstecken 123
 - persisting data 121
 - overview 121
 - Phantom Read 168
 - physicalName (Attribut) 341
 - Pizza-Lieferung (Beispiel)
 - Beenden des Ablaufs 231
 - Bestellung bilden 231
 - Kundendaten speichern 231
 - Kundeninformationen sammeln 226
 - Kunde suchen 229
 - Lieferbezirk prüfen 230
 - nach einer Telefonnummer fragen 228
 - Registrieren eines neuen Kunden 229
 - Zahlungsannahme 234
 - Placeholder-Variable 379
 - Plain-old Java Objects 5 *siehe auch* POJO
 - Plattformunabhängige Exceptions 124
 - p (Namensraum) 48
 - poetic juggler (Beispiel) 36, 38, 73
 - Pointcut 14, 94
 - <aop:pointcuts> (Element) 101, 104
 - bean() 100
 - Konstruktor-Pointcuts 115
 - Methodenebene 269
 - Pointcut-Bezeichner 98
 - schreiben 99
 - Transaktionen 172
 - @Pointcut (Annotation) 110
 - pointcut (Attribut) 103
 - Pointcut-Bezeichner 100
 - pointcut-ref (Attribut) 104
 - POJO 4, 5, 6, 17, 349 *siehe auch* Plain-old Java Objects
 - Message-Listener 350
 - nachrichtengetriebenes 349
 - poolPreparedStatements (Eigenschaft) 131
 - port (Eigenschaft) 391
 - portName (Eigenschaft) 295
 - @PostAuthorize (Annotation) 265
 - @PostConstruct (Annotation) 268
 - @PostFilter (Annotation) 266
 - postForEntity() (Methode) 316
 - postForLocation() (Methode) 316, 324
 - postForObject() (Methode) 316
 - POST (Methode) 305, 323
 - HiddenHttpMethodFilter 329
 - Objektantworten empfangen 323
 - Ressourcenstandort empfangen 324
 - postProcessAfterInitialization() (Methode) 20
 - postProcessBeforeInitialization() (Methode) 20
 - Prasanna, Dhanji R. 10
 - @PreAuthorize (Annotation) 265
 - @PreFilter (Annotation) 264
 - pre-post-annotations (Attribut) 264
 - primary (Attribut) 72
 - principal (Ausdruck) 249
 - principal (Eigenschaft) 251
 - private (Schlüsselwort), @Autowired-Annotation 77
 - Programmgesteuerte Transaktion 163
 - Projektion 65
 - PROPAGATION_MANDATORY (Verhalten) 167
 - PROPAGATION_NESTED (Verhalten) 167
 - PROPAGATION_NEVER (Verhalten) 167
 - PROPAGATION_NOT_SUPPORTED (Verhalten) 167
 - PROPAGATION_REQUIRED (Verhalten) 167
 - PROPAGATION_REQUIRES_NEW (Verhalten) 167
 - PROPAGATION_SUPPORTS (Verhalten) 167
 - Propagationsverhalten 166
 - propagation (Verhalten) 171
 - properties-ref (Attribut) 380
 - Property-Overrider 381
 - Property-Placeholder 379
 - PropertyPlaceholderConfigurer (Schnittstelle) 377
 - <props> (Element) 53

Prototyp, Beans 39
 Provider (Interface), @Inject (Annotation) 81
 ProviderManager (Klasse) 255
 proxy-interface (Attribut) 388
 proxyInterface (Eigenschaft) 372
 Proxy, MBeans 371, 372
 Pseudo-Implementierung 8, 24
 Publish-and-Subscribe-Modell
 - Nachrichtenmodell 336
 - Standortunabhängigkeit 338
 Publish-Subscribe-System 335
 Punkt-zu-Punkt-Modell 336
 - Standortunabhängigkeit 338
 put() (Methode) 316, 320
 PUT (Methode) 305
 - HiddenHttpMethodFilter 329

Q

@Qualifier (Annotation) 28, 78, 79
 Qualifier, eigener 79, 82
 Querschnittliche Funktion 11 *siehe auch* CCC
 queryNames() (Methode) 370
 Queue 336
 queueName (Eigenschaft) 354

R

receive() (Methode) 347, 349
 ref (Attribut) 38
 registration (Attribut) 368
 registrationBehaviorName (Eigenschaft) 368
 REGISTRATION_FAIL_ON_EXISTING 368
 REGISTRATION_IGNORE_EXISTING 368
 REGISTRATION_REPLACING_EXISTING 368
 Registrierungsformular 197
 registryHost (Eigenschaft) 278
 registryPort (Eigenschaft) 278
 Regulärer Ausdruck, Operatoren 62
 Relationaler Operator 60
 Remember-me-Feature 261
 RemoteAccessException 275
 RemoteException 277
 Remote (Interface) 277
 Remote Method Invocation (RMI) 275, 276, 277
 Remote Procedure Call (RPC) 274
 - Burlap 282
 - Hessian 282
 - HTTP-Invoker 286
 - JMS-Invoker 352
 - JmsInvokerProxyBean 354
 - JmsInvokerProxyFactoryBean 354
 - JmsInvokerServiceExporter 352, 354
 - JmsProxyFactoryBean 356
 - JmsServiceExporter 355, 356

- Lingo 355
 - Modelle 274
 - nachrichtenbasierte 352
 - Remote Method Invocation (RMI) 276
 Remote-Service 274
 - Hessian und Burlap 282
 - HTTP-Invoker 286
 - Remote Method Invocation (RMI) 276
 - Übersicht 274, 276
 Remoting 274 *siehe auch* Remote-Service
 replaceExisting 368
 Repositories 122 *siehe auch* Datenzugriffsobjekte
 @Repository (Annotation) 84, 147, 153
 Repräsentation
 - REST-Ressourcen 308
 - verhandeln 308
 Request
 - abfangen 247, 250
 - HiddenHttpMethodFilter 329
 - HttpServletRequest 242
 - HTTPS erzwingen 249, 250
 - in Spring MVC 176, 177
 - Medientypen feststellen 309
 - Metadaten aus Antwort extrahieren 319
 - Objektantworten empfangen 323
 - Ressourcenstandort empfangen 324
 - Ressourcenzustand im Body des Requests zurückgeben 314
 - sichern 242, 250
 - View finden 310
 - Wahl der Medientypen beeinflussen 310
 @RequestBody (Annotation) 314
 request (Geltungsbereich) 221
 @RequestMapping (Annotation) 181, 193, 197, 300, 303, 307, 312
 @RequestParam (Annotation) 193, 207
 ResourceBundleViewResolver 185
 ResourceHttpMessageConverter 313
 resource-ref (Attribut) 130, 387
 @ResponseBody (Annotation) 311, 314
 ResponseEntity 319
 @ResponseStatus (Annotation) 306, 307
 Ressource
 - auslesen 318
 - austauschen 325
 - DELETE-Request 322
 - GET-Request 317
 - POST-Request 323
 - PUT-Request 320
 - Repräsentation verhandeln 308
 - repräsentieren 308
 - ressourcenorientierte URLs 302
 - Ressourcenstandort empfangen 324

- Ressourcenzustand im Body der Antwort zurückgeben 311
- Ressourcenzustand im Body des Requests zurückgeben 314
- REST 298
- REST-Clients 315
- REST-konforme 298
- REST-konformer Controller 299
- REST
 - Clients schreiben 315
 - Controller 299
 - Einführung 298
 - Formulare 327
 - Methoden 304
 - nicht-REST-konformer Controller 300
 - Repräsentation verhandeln 308
 - Ressourcen austauschen 325
 - Ressourcen repräsentieren 308
 - REST-konforme URLs 301
 - RestTemplate 316, 317
 - Spring-Support für 299
 - Verben 304
- REST-Client
 - GET-Request 317
 - PUT-Requests 320
 - schreiben 315
- REST-konforme Ressource 298
- REST-konformer URL 301
- RestTemplate 315, 316, 317
- RmiProxyFactoryBean (Klasse) 280
- RmiRegistryFactoryBean 369
- RmiServiceExporter (Klasse) 278, 369
- @RolesAllowed (Annotation) 263
- rollback() (Methode) 160, 161, 163
- Rollback-Regel 170, 171
- RPC 274 *siehe auch* Remote Procedure Call
- RssChannelHttpMessageConverter 313

- S**
- Schablone, Datenzugriff 126
- Schablonenklassen, Beziehung zu DAO-Support-Klassen 128
- Schablonenmethode 126
- @Scheduled (Annotation) 399, 400
- Scheduling 377, 399, 400 *siehe auch* Zeitplanung
- Schnittstelle
 - EntityManagerFactory 148, 152
 - FactoryBean 144
 - java.util.Map 140
 - javax.sql.DataSource 137
 - PersistenceProvider 148
 - Persistenzebene verstecken 123
 - SessionFactory 147
- scope (Attribut) 39, 40, 252
- @Secured (Annotation) 263
- secured-annotations (Attribut) 263
- send() (Methode) 346, 347
- Serializable (Interface) 276
- server (Eigenschaft) 361, 372
- Server, MBean-Server 360
- Service
 - auf Hessian-/Burlap-Services zugreifen 285
 - Beans als HTTP-Services exportieren 287
 - Burlap-Service exportieren 284
 - Hessian-Service exportieren 283
 - JAX-WS auf Client-Seite 294
 - remote 274
 - RMI-Service exportieren 277
 - RMI-Service konfigurieren 277
 - RMI-Service verschalten 279
 - Webservices veröffentlichen und weiterverarbeiten 289
 - Zugriff via HTTP 288
- @Service (Annotation) 85
- service (Eigenschaft) 355
- serviceInterface (Eigenschaft) 280, 289, 355, 356
- serviceName (Eigenschaft) 295
- Service-oriented Architecture (SOA) 289
- Serviceorientierte Architektur 289
- serviceUrl (Eigenschaft) 280, 289, 369
- Servlet
 - DispatcherServlet 177
 - Servlet-Filter 243
- Servlet-Filter 243
- <servlet-name> 178
- Session-Bean 5
- session (Eigenschaft) 392
- Session-Factorys 144, 146
- SessionFactory (Schnittstelle) 147
- setApplicationContext() (Methode) 20
- setAttribute() (Methode) 371
- setBeanFactory() (Methode) 20
- setBeanName() (Methode) 20
- setRollbackOnly() (Methode) 164
- Setter-Injektion 42
 - vs. Konstruktoreninjektion 37
- Setter-Methode 44, 77
- Sicherheit 239, 242, 304 *siehe auch* Spring Security
 - Abläufe sichern 236
 - minimale 244
- SimpleJaxWsServiceExporter 291, 292
- SimpleJdbcDaoSupport (Klasse) 129, 140
- SimpleJdbcTemplate (Klasse) 16, 127, 136, 137, 139, 140
- SimpleMetadataStrategy (Klasse) 356
- Simple Storage Service 208
- SimpleUrlHandlerMapping 181
- SingleConnectionDataSource (Klasse) 132

- Singleton-Bean 19, 39
- @Size (Annotation) 202
- SOA 289 *siehe auch* Service-oriented Architecture
- SourceHttpMessageConverter 313
- SpEL 54 *siehe auch* Spring Expression Language
- Spitter (Beispiel) 137, 154, 163, 171
 - Controller 192
 - Element 393
 - hochgeladene Dateien von Usern 206
 - Homepage-Controller 182
 - Homepage-View 188, 191
 - JmsInvokerServiceExporter 352
 - Messaging mit JMS 345
 - mit nachrichtengetriebenen POJO 351
 - Remote-Services 274
 - und MBeans 360, 375
 - und REST 300
- Spring
 - Abhängigkeitsinjektion 6, 10
 - Advice-Beans 91
 - Anwendungskontext 18, 192
 - AOP-Unterstützung 96
 - Applikationskontext 10, 18, 19
 - Arbeiten mit Datenbanken 154
 - Arbeit mit Datenbanken 121
 - Aspektorientierte Programmierung (AOP) 10, 15
 - Beans 19
 - Beans deklarieren 32
 - Beans verschalten 31, 67
 - Boilerplate-Code eliminieren 15
 - Container 17
 - Datenzugriff-Exception-Hierarchie 123, 125
 - Datenzugriffsebene 122
 - Exception-Hierarchie gegen JDBC Exception-Hierarchie 124
 - Flexibilität 6
 - für Datei-Upload konfigurieren 209
 - Geschichte 3
 - Grundlagen 3, 4
 - Integration mit der Java Persistence API 147, 154
 - Integration mit Hibernate 142, 147
 - JDBC-Unterstützung 23
 - JmsTemplate 341
 - Kern-Container 22
 - konfigurieren 32, 34
 - Laufzeit-AOP 97
 - Messaging mit JMS 334
 - mit Java konfigurieren 87
 - Modul 21, 24
 - MVC 23 *siehe auch* Spring MVC
 - MVC-Framework 23
 - programmgesteuerte 163
 - Sicherheit 239
 - Spring AOP 96
 - Spring Batch 25
 - Spring Dynamic Modules 26
 - Spring Faces 29
 - Spring Flex 27
 - Spring Framework 21, 24
 - Spring Integration 25
 - Spring JavaScript 29
 - Spring LDAP 26
 - Spring.NET 26
 - Spring Rich Client 26
 - Spring Roo 27
 - Spring Security 25, 240
 - Spring Social 25
 - Spring Web Flow 24, 211
 - Spring Web Services 24
 - Support für REST 299
 - Testmodul 24
 - Transaktionen 155, 163
 - Transaktionen deklarieren 165
 - Transaktions-Support 158
 - und POJOs 5, 6
 - vereinfachte Java-Entwicklung 4
 - Verschaltung 9
 - Verschaltung von EJBs 390
 - was ist neu in Spring 2.5 28
 - was ist neu in Spring 3.0 29
 - XML-Konfiguration 33
- Spring AOP 96
 - Methoden-Joinpoints 98
- Spring Batch 25
- SpringBeanAutowiringSupport 291
- Spring-Beans
 - Export als MBeans 360, 368
 - Verwalten mit JMX 359
- Spring-DM 26
- Spring Dynamic Modules 26
- Spring Expression Language 54, 217
 - and (Operator) 61
 - Annotation 83
 - auf Collection-Member zugreifen 63
 - Beans nach ID referenzieren 56
 - Collection-Member auswählen 65
 - Collections 63
 - le- und ge-Operatoren 60
 - literale Werte 55
 - Methoden sichern 264
 - not (Operator) 61
 - ^ (Operator) 59
 - - (Operator) 59
 - ! (Operator) 61
 - ?: (Operator) 61
 - [] (Operator) 64

- * (Operator) 59
 - / (Operator) 59
 - % (Operator) 59
 - + (Operator) 59
 - == (Operator) 60
 - > (Operator) 60
 - or (Operator) 61
 - sicherheitsspezifische Ausdrücke 248
 - Spring Security 248, 249
 - systemEnvironment (Eigenschaft) 64
 - systemProperties (Eigenschaft) 64
 - T() (Operator) 57
 - Typen 57
 - und @Value (Annotation) 83
 - Spring Faces 29
 - Spring Flex 27
 - Spring Framework 23, 24, 96
 - Spring Integration 25
 - Spring JavaScript 29
 - Spring LDAP 26
 - Spring MVC 23, 175, 176, 210
 - Annotationen 181
 - Controller 177
 - Controller-Input bearbeiten 192
 - Controller schreiben 180, 192
 - DispatcherServlet 177
 - Formulare verarbeiten 196
 - Grundlagen 176
 - mit Apache Tiles 187
 - Requests 176, 177
 - Spring Web Flow 212
 - Spring.NET 26
 - Spring-Portfolio 24, 29
 - Spring Rich Client 26
 - Spring Roo 27
 - Spring Security 25, 239, 240
 - Acegi Security 240
 - Aspekt-orientierte Programmierung 240
 - Ausloggen 247
 - Authentifizierung 251, 254, 255
 - Authentifizierung bei Datenbank 256, 257
 - Authentifizierung mit speicherresidentem UserRepository 255
 - Basic-Authentifizierung 247
 - Berechtigungen 252
 - Dependency Injection 240
 - Elemente auf View-Ebene sichern 250
 - HTTPS erzwingen 249, 250
 - JavaServer Pages (JSP) 241
 - konfigurieren 240
 - Lightweight Directory Access Protocol 258, 261
 - Login-Formular 244
 - Methoden 262
 - minimale Websicherheit 244
 - Module 240
 - Namensraum konfigurieren 241
 - OpenID 241
 - Passwortvergleich mit LDAP 259
 - Pointcuts auf Methodenebene 269
 - Remember-me-Feature 261
 - Requests abfangen 247, 250
 - Servlet-Filter 243
 - sicherheitsspezifische Ausdrücke 248
 - Spring Expression Language 249
 - Verschlüsselungsstrategie 259
 - Web-Requests sichern 242, 250
 - Spring Security 2.0 29
 - Spring Security-Modul 240
 - Spring Social 25
 - Spring Web Flow 24, 211, 212, 213, 215 *siehe auch* Flows
 - Installation 212
 - Spring Web Flow 2.0 29
 - Spring Web Services 24
 - SQLException 16, 134
 - SqlMapClientDaoSupport (Klasse) 129
 - SqlMapClientTemplate 127
 - stage (Beispiel) 38
 - StandardPBEStrategy 383
 - start-state (Attribut) 224
 - String (Attribute), Nachteile 88
 - stringEncryptor (Klasse) 383
 - Struts 6
 - Subflow-Zustand 216, 217
 - Synchrone Kommunikation 334, 337, 338
 - Kopplung 338
 - Synchrones Messaging 337
 - Systemeigenschaft 380
 - systemEnvironment (Eigenschaft) 64
 - systemProperties (Eigenschaft) 64
 - system-properties-mode (Attribut) 380
- ## T
- Tapestry 6
 - Task, Zeitplanung 377
 - Template 15, 126
 - ausschließen von Schablonenklassen mit Hibernate 146, 147
 - CciTemplate 127
 - Datenzugriff 127
 - E-Mail-Vorlagen erstellen 396
 - HibernateTemplate 127
 - Java Message Service (JMS)
 - Default-Endpunkt 347
 - Nachrichten empfangen 347
 - Nachrichten versenden 345
 - JDBC 136

- JdbcTemplate 16, 127
 - JdoTemplate 127
 - NamedParameterJdbcTemplate 127
 - SimpleJdbcTemplate 16, 127
 - SqlMapClientTemplate 127
 - Test
 - Controller 183
 - Pseudo-Implementierung 24
 - Testmodul 24
 - Unit-Test 7
 - Thema (topic) 336
 - then (Attribut) 217
 - Ticket (Beispiel) 40
 - Tile-Definition 188
 - TilesViewResolver 185, 187
 - Tiles, Views auflösen 187
 - Timeout 169
 - timeout (Attribut) 171
 - to (Attribut) 219
 - Tomcat 129
 - T() (Operator) 57
 - TopLinkJpaVendorAdapter 151
 - toUpperCase() (Methode), nach ID referenzieren 57
 - TRACE (Methode) 305
 - @Transactional (Annotation) 153, 165, 173
 - TransactionCallback (Schnittstelle) 164
 - TransactionDefinition (Interface) 166
 - Transaction (Interface) 161
 - transaction-manager (Attribut) 172
 - TransactionManager (Interface) 163
 - transactionManagerName (Eigenschaft) 162
 - Transaktion 155
 - annotationsgetriebene 172
 - Attribute 166
 - containerverwaltete 158
 - Definition 159
 - deklarieren 165
 - Dirty Reads 168
 - Grundlagen 156
 - Hibernate 161
 - Isolationsebenen 168, 171
 - Java Transaction API 158
 - JDBC 160
 - JPA 161
 - JTA 162
 - Nonrepeatable Reads 168
 - nur lesende 169
 - Phantom Reads 168
 - programmgesteuerte 163
 - Propagationsverhalten 166
 - propagation (Verhalten) 171
 - read-only 169, 171
 - Rollback-Regeln 170, 171
 - Support in Spring 158
 - timeout 171
 - Timeout 169
 - Transaktionsattribute 166
 - Transaktionsmanager 159
 - tx (Namensraum) 33
 - Transaktionsattribut 166
 - Isolationsebene 171
 - propagation (Verhalten) 171
 - read-only 169, 171
 - Rollback-Regeln 170, 171
 - timeout 171
 - Transaktionsmanager 159, 160, 161
 - Transition 219, 220
 - <tx:advice> (Konfigurationselement) 171
 - <tx:annotation-driven> (Konfigurationselement) 172
 - <tx:attributes> (Konfigurationselement) 171
 - <tx:method> (Konfigurationselement) 171
 - tx (Namensraum) 33, 165, 170, 172
 - type (Attribut) 86
 - Typenbasiertes Autowiring 72
 - Typ, in Spring Expression Language 57
- ## U
- Uniform Resource Locator (URL)
 - eingebettete Parameter 302
 - Kennzeichen REST-konformer URLs 301
 - REST-konforme 301
 - Unit-Test 386
 - schreiben 7
 - UnsupportedOperationException 268
 - Unübersichtlicher Code 133, 136, 342
 - URISyntaxException 321
 - url (Attribut) 253
 - UrlBasedViewResolver 185
 - url (Eigenschaft) 131
 - URLs 301 *siehe auch* Uniform Resource Locator
 - use-expressions (Attribut) 248
 - useNotAcceptableStatusCode (Eigenschaft) 311
 - username (Eigenschaft) 131, 392
 - userPassword (Attribut) 259
 - users-by-username-query (Attribut) 257
 - user-search-base (Attribut) 259
 - user-search-filter (Attribut) 258
 - user-service-ref (Attribut) 255, 256
 - UserTransaction (Interface) 163
 - util (Namensraum) 33, 63
- ## V
- @Valid (Annotation) 199
 - Validierung
 - Fehler 202
 - Formulareingabe 201

- Regeln 201
 - Validierungsfehler, Formular 202
 - @Value (Annotation) 83, 207, 379
 - value (Attribut) 37, 44, 52, 55, 114, 307
 - value-ref (Attribut) 52
 - Variable
 - deklarieren im Ablauf 220
 - Placeholder-Variable 379
 - VelocityEngine 397
 - VelocityEngineFactoryBean 397
 - VelocityLayoutViewResolver 185, 187
 - velocityProperties (Eigenschaft) 397
 - VelocityViewResolver 185, 187
 - Verbindung mit Datenbank aufbauen 378
 - Vererbung im Vergleich mit Aspekten 93
 - Vergleichsoperator 60
 - Verschalten 9
 - Autowiring 70
 - Autowiring und explizite Verschaltung mischen 75
 - Beans 31, 38, 67
 - Collections 48
 - Definition 31
 - Eigenschaften 48
 - Eigenschafts-Collections 52
 - Enterprise JavaBeans 390
 - Funktionsweise 10
 - innere Beans 47
 - JmsTemplate 345
 - JNDI-Objekte 384
 - Mail-Sender mit Service-Bean 392
 - Map-Collections 51
 - mit Annotationen 76
 - mit Spring Expression Language 54
 - nach ID referenzieren 56
 - null-Element 53
 - -ref (Suffix) 48
 - RMI-Service 279
 - von Default-Endpunkt in JmsTemplate 347
 - von EJBs in Spring 390
 - Verschlüsselung 259, 382, 383
 - View
 - Auflösung 184
 - Berechtigungen 252
 - Elemente auf View-Ebene sichern 250
 - finden 310
 - Formular-Views 197, 199
 - Homepage-View 188, 191
 - interner 186
 - mit Apache Tiles 187
 - rendern 195
 - view (Attribut) 216, 218
 - view (Geltungsbereich) 221
 - View-Resolver 184
 - BeanNameViewResolver 185
 - ContentNegotiatingViewResolver 185, 308
 - FreeMarkerViewResolver 185, 187
 - InternalResourceViewResolver 185
 - JasperReportsViewResolver 185, 187
 - ResourceBundleViewResolver 185
 - TilesViewResolver 185, 187
 - UrlBasedViewResolver 185
 - VelocityLayoutViewResolver 185, 187
 - VelocityViewResolver 185, 187
 - XmlViewResolver 185
 - XsltViewResolver 185, 187
 - View-Zustand 216
 - VisualVM, JMX 360
- W**
- Warteschlange (queue) 336
 - was ist neu in Spring 3.0? 29
 - WEB-INF directory 178
 - WebLogicJtaTransactionManager 159
 - WebLogic, Transaktionen 159
 - Webrequest, Spring Model-View-Controller (MVC) 176
 - Webservice
 - JAX-WS auf Client-Seite 294
 - JAX-WS-Endpunkte 291
 - veröffentlichen und weiterverarbeiten 289
 - WebSphere 129
 - WebSphereUowTransactionManager 159
 - WebWork 6
 - Wert
 - injizieren 43
 - literaler 55
 - Wiring 9 *siehe auch* Verschalten
 - wsdlDocumentUrl (Eigenschaft) 295
- X**
- XFire 290
 - XML 9 *siehe auch* Extensible Markup Language
 - Aspekte 14
 - XmlAwareFormHttpMessageConverter 313
 - XmlViewResolver 185
 - XmlWebApplicationContext 18
 - XsltViewResolver 185, 187
- Z**
- Zeitplanung für Tasks 377
 - Zerstören, Beans 40