



Développer une application Java/JEE avec Spring

Référence : SPRING-CORE-04

Durée : 4 jour(s)

Formation certifiante

Présentation

Dans le cadre de leur partenariat, Zenika vous invite à vous joindre aux formations SpringSource officielles animées par un consultant certifié par SpringSource. N'hésitez pas à nous contacter pour plus d'information.

Objectifs

- Utiliser le Framework Spring pour développer des applications Java.
- Utiliser l'injection de dépendance pour configurer des applications.
- Tester des applications basées sur Spring.
- Effectuer la configuration Spring avec du XML, des annotations et des classes Java.
- Utiliser Hibernate et JDBC avec Spring pour les accès aux bases de données relationnelles.
- Utiliser le support de transaction de Spring.
- Utiliser la programmation orientée aspect (POA) pour ajouter du comportement aux objets.
- Développer une application Web avec Spring MVC.
- Utiliser Spring Security pour sécuriser les applications Web.
- Utiliser Spring avec RMI, HttpInvoker, et JMS pour les communications externes.
- Monitorer des applications Spring avec JMX.

Répartition: 40% Théorie, 60% Pratique

Public: Architecte, Développeur, Chef de projet

Pré-requis: Connaissance de Java

Programme

Introduction à Spring

- Le contexte d'application Spring
- configuration XML
- Travailler avec des fabriques déjà existantes
- Travailler avec plusieurs fichiers de configuration

- Configuration d'Hibernate avec Spring
- Gestion des exceptions

Comprendre le cycle de vie du conteneur Spring

- Phases de d'initialisation, d'utilisation et de destruction d'un contexte Spring
- Initialiser, utiliser, et détruire les phases
- Travailler avec les post-processeurs
- Externaliser des paramètres de configuration dans des fichiers propriétés
- Notion de portée (bean scope)

Gestion des transactions avec Spring

- Déclarer un gestionnaire de transaction dans Spring
- L'annotation @Transactionnal
- Configuration des niveaux d'isolation
- Configuration de la propagation
- Transactions et tests d'intégration

Bonnes pratiques de configuration

- Définition des Beans par héritage
- Beans internes
- p et autres namespaces utiles
- Injection de dépendances des collections
- Spring Expression Language (SpEL)

Spring dans une application web

- Configurer Spring dans une application Web
- Introduction à Spring MVC
- Définition des contrôleurs Spring MVC avec des annotations
- Gestion des vues avec Spring MVC)
- Namespace mvc

Injection de dépendance avec les annotations et des classes Java

- Autowiring et scanning des composants
- Annotations stéréotypes
- Configuration basée sur Java
- Faire cohabiter les différents styles de configuration
- Quand utiliser les configurations par XML, annotations et Java

Spring security

- Les problèmes résolus par Spring Security
- Configuration de l'authentification
- Interception des URLs
- La librairie de tag Spring Security pour les pages JSP
- Personnaliser la chaîne de filtre de Spring Security

Tester une application basée sur Spring

- Support pour JUnit dans Spring
- La mise en cache du contexte d'application et l'annotation @DirtiesContext

Remoting

- Problème lié au RMI traditionnel
- Utiliser Spring Remoting à la place de RMI
- Utiliser Spring HttpInvoker pour les appels distants sur HTTP

Programmation orientée aspect (POA)

- Quels problèmes résout la POA ?
- Différences entre Spring AOP et AspectJ
- Définir des expressions de pointcut
- Comment utiliser les différents types d'advice : around, before, after

JMS

- Introduction à JMS
- Configurer les ressources JMS avec Spring
- Envoyer et recevoir des messages en utilisant Spring JmsTemplate et les message listener containers

Accès aux données et JDBC avec Spring

- Comment Spring s'intègre avec les technologies d'accès aux données existantes
- Hiérarchie d'exceptions d'accès aux données (DataAccessException)
- Namespace JDBC
- Simplifier les accès JDBC avec le JdbcTemplate


JMX

- Introduction à JMX
- Configurer Spring pour exporter des MBeans automatiquement
- Exporter un Spring bean en un MBean

Intégration de Spring avec Hibernate

- Introduction au mapping objet/relationnel
- Bénéfices de l'utilisation de Spring et d'Hibernate

OFFERT EN INTER-ENTREPRISE

 Le petit déjeuner croissants, jus d'orange, café)

 Le déjeuner

 Une qualification téléphonique si nécessaire avec l'un de nos consultants

 Examen pour la certification offert!

Tel: +33(0)1 45 26 19 15
Fax : +33(0)1 75 43 49 92
Email : training@zenika.com

