

Spring - Architecture Java

Format :

Présentiel et/ou Distanciel

Durée :

3 jours / 21 h

Référence :

DEV-SPR-0203

Type de formation :

Formation qualifiante

Public :

Développeurs, Chefs de projets.

Personnes en situation de handicap :

Vous êtes en situation de handicap et vous souhaitez faire une formation ?

Merci de bien vouloir nous contacter en amont afin d'étudier ensemble vos besoins et les solutions les plus adaptées.

Objectifs de développement des compétences :

Fournir connaissances théoriques et pratiques permettant d'être un leader technique Architecte Java.

Pré-requis :

Pratique développement java/j2ee.

Compétences et méthodes pédagogiques :

Les prestations de formation sont assurées par des formateurs professionnels qui utilisent des moyens pédagogiques adaptés.

Dans le cadre de session intra entreprise, possibilité de travailler sur vos projets afin de répondre à vos besoins spécifiques.

Supports de cours pédagogiques imprimés et/ou numérisés.

Répartition du temps (environ) :

Théorique 45%, Pratique 55%

Modalités d'évaluation :

Questionnaire d'auto-positionnement:

Un questionnaire d'auto-positionnement est adressé aux stagiaires en amont de la formation afin de l'adapter aux besoins et attentes des participants.

Évaluation à chaud par le biais de travaux pratiques.

- Exercices, tests d'évaluations (QUIZZ ou QCM ...).

Moyens techniques et pédagogiques :

Salle(s) de cours équipée(s) des moyens audiovisuels avec le matériel adapté à la formation (si besoin, ordinateur par stagiaire).

Modalité et délai d'accès à la formation :

Toutes nos formations sont réalisées à la demande et en fonction des souhaits de nos clients.

Nous pouvons également réaliser des formations sur-mesure à partir de programmes existants ou en construisant un programme spécifique à partir de vos objectifs.

Merci donc de bien vouloir nous contacter par courriel ou par téléphone afin de définir ensemble les dates et modalités de formation souhaitées.

Tarif :

Nous contacter pour devis personnalisés.

Programme de la formation

1. Spring : Comprendre le conteneur léger

Notion clé : injection de dépendance

Conteneur 'léger' spring / conteneur 'lourd' j2ee

Fichier ApplicationContext.xml

Cycle de vie d'un bean Spring/Scope singleton, prototype

Interfaces de callback/Les post-processors

Injection par setter/Injection par constructeur

2. Spring : architecture

Approche modulaire des librairies Spring 3 (jars)

Noyau : Spring Core au coeur de l'injection

Accès données : Spring DAO, Spring ORM

Applications web : Spring Web, Spring MVC

Nouveauté Spring 3 : Spring Expression Language (spEL)

Tests avec Spring

3. Spring : paramétrage conteneur

Propriétés simples (String, int,...)/Propriétés de type collection

Méthodes d'injection (setter, constructeur)

Auto-câblage (Autowiring) byName, byType

Optimisation : autowiring annotation (context:annotation-config)

Optimisation : auto-découverte (context:component-scan)

Annotations @Component, @Service, @Controller

Classe abstraite (abstract=true)/Héritage (parent=true)

4. Spring : Programmation orientée aspect

Notion clé : Interception/Pointcut, Advice, JoinPoint

Notion de Tissage, Tisseur/Présentation framework AOP AspectJ

Comparaison tisseurs Spring AOP / AspectJ

5. Spring : accès données JDBC

Les classes de la couche d'abstraction

Mise en oeuvre design pattern DAO

Classes fournies : JdbcTemplate, JdbcDaoSupport

Gestion des exceptions : gamme d'exceptions Spring

Exécuter des requêtes de lecture, écriture/Annotation @Repository

6. Spring : Gestion des transactions

Concept de transaction/Gérer les transactions avec Spring

Transactions programmatiques et déclaratives

Annotation @Transaction Management

7. Spring : Création d'écrans web avec Spring MVC

Cinématique d'une requête Spring MVC : 6 étapes clés

Configuration Spring MVC : 4 étapes clés

Comprendre le DispatcherServlet

Création de contrôleurs Spring MVC 3/Annotations @Controller
Formulaires Spring MVC : balises clés/validation (@Valid, jsr 303)
Formulaires Spring MVC : conversion/internationalisation
Intégration Spring avec templates Tiles

8. Spring : intégration avec JSF

Rappels JSF 2 : cycle de vie, facelets, @nnotations
Faire d'un Managed bean JSF un bean Spring
ContextLoaderListener
Injection bean service dans le contrôleur Spring

9. Spring : intégration Spring/JPA Hibernate

10. Tests avec Spring

11. Sécurité j2ee avec Spring Security

Nous contacter :

Dominique Odillard : 05 24 61 30 79

Version mise à jour le 11/06/2020