

## Spring Core

Le framework Spring est devenu un incontournable des développements logiciels Java. L'écosystème Spring, bien que rendu plus accessible avec le module Spring Boot, est aujourd'hui très vaste et constitue une galaxie de modules (MVC, Rest, Data, Batch, Cloud...).

Pourtant tous ces modules s'appuient sur un même noyau fondamental, le module Spring Core, et des concepts relativement simples mais puissants.

Cette formation est destinée à toute personne désirant acquérir les bases fondamentales de Spring pour intégrer un projet, qu'il soit ancien (configuré en XML), en migration ou bien moderne (configuration zéro XML).

L'accent est mis sur la mise en pratique des concepts à travers de nombreux exercices. Cette formation est proposée sur 2 jours mais peut être allongée sur 3 jours pour un public novice en Java et en conception objet (contactez l'équipe [formation@oxiane.lu](mailto:formation@oxiane.lu) pour plus d'informations).

Cette formation correspond aux deux premiers jours de notre formation plus complète Spring essentiel de 4 jours et constitue le prérequis à d'autres formations telles que Spring Boot.

### Détails

- Code : JF-SPRC
- Durée : 2 jours ( 14 heures )

#### Public

- Architectes
- Chefs de projets
- Développeurs

#### Pré-requis

- Bonne connaissance de Java

### Objectifs

- Maîtriser les concepts fondamentaux de Spring Core
- Comprendre les configurations XML, Java et par annotations
- Devenir opérationnel sur un projet Spring, ancien ou moderne

### Programme

#### Introduction

- Historique
- Motivations
- Inversion de contrôle
- Le contrôle des dépendances
- Contrôle directe des dépendances
- Injection de dépendances

#### Conteneur léger

- Métadonnées de configuration
- Utilisation du conteneur
- Bénéfices de l'injection de dépendance
- TP 1 : *Expérimenter un assemblage d'objets*
- A quoi sert Spring ?
- Evolutions du Framework

#### Présentation du Conteneur IoC

- Fichier de configuration de Spring
- Les schémas et namespaces XML
- Vers les annotations
- BeanFactory et ApplicationContext
- Mise en place du contexte d'application

#### Déclaration des beans

- Exemple d'assemblage

- Exemple de déclaration de beans
- Déclaration des beans sans XML
- L'annotation @Configuration
- TP 2 : *Mettre en œuvre Spring*

#### Injection et instanciation des beans

- Injection par setter
- Injection (instanciation) par constructeur
- Instanciation par « factory method » statique
- Instanciation par « factory bean »
- L'interface FactoryBean
- Bean anonyme (créé à la volée)
- Instanciation lazy
- TP 3 : *Instanciation et injection des beans*

#### L'autowiring

- Les modes d'autowiring
- L'annotation @Autowired
- Candidats à l'autowiring
- Réduire le nombre de candidats à l'autowiring
- @Qualifier spécifiques
- TP 4 : *Expérimenter l'autowiring*

#### Autres injections

- Injection de collections

- @Autowired avec des collections
- Dépendance optionnelle/requise
- Autres annotations d'injection
- *TP 5 : Injection de collection et dépendance optionnelle*
- Injection de valeur et @Value
- Autres sources de valeurs
- Spring Expression Language (SpEL)
- Éléments de syntaxe de SpEL
- *TP 6 : Injection de valeur et expérimenter SpEL*

### Décomposer la configuration

- @Import, @ImportResource
- Autres particularités des configurations XML

### Le scope des Beans

- Le scope « prototype »
- Autres scopes
- *TP 7 : Comprendre les scopes*

### Déclaration de beans par annotation

- Découverte des @Component
- Composants stéréotypés
- Les annotations
- *TP 8 : Déclaration de beans par annotation*

### Profils d'environnement

- Profils d'environnement : création

- Profils d'environnement : activation
- L'annotation @Conditional
- *TP 9 : Utiliser les profils d'environnement (bean sous condition)*

### Le cycle de vie des beans

- Initialisation et destruction du bean
- « init-method » et « destroy-method »
- @PostConstruct et @PreDestroy
- Les interfaces « Aware »
- L'interface BeanPostProcessor
- Déclaration des BeanPostProcessor
- Le cycle de vie en résumé
- *TP 10 : Comprendre le cycle de vie des beans*

### Les contextes d'application

- Les implémentations d'ApplicationContext
- Les ressources
- Récupération des ressources
- Utilisation des ressources
- L'internationalisation
- *TP 11 : Mettre en œuvre les ressources*

### Contexte d'application et événements

- Exemple d'événement applicatif
- *TP 12 : Mettre en œuvre les événements applicatifs*

## Modalités

- **Type d'action** :Acquisition des connaissances
- **Moyens de la formation** :Formation présentielle – 1 poste par stagiaire – 1 vidéo projecteur – Support de cours fourni à chaque stagiaire
- **Modalités pédagogiques** :Exposés – Cas pratiques – Synthèse
- **Validation** :Exercices de validation – Attestation de stages