

## Spring : Développer avec Spring

### Objectif :

Spring est un framework open source Java EE pour les applications 3-tiers, dont il facilite le développement et les tests. Spring s'appuie principalement sur l'intégration de trois concepts clés : l'inversion de contrôle ou injection de dépendance, la programmation orientée aspect et une couche d'abstraction qui permet d'intégrer d'autres Frameworks et bibliothèques avec une plus grande facilité.

Les participants à cette formation Spring apprendront les bases du framework Spring et seront sensibilisés à la facilité de mettre en œuvre des tests unitaires et d'intégration avec ce Framework.

Le premier atelier permet aux participants de mettre en place l'IDE et d'effectuer une première configuration Spring simple. Les ateliers suivants s'adosent à une application web modélisant une médiathèque en ligne. Les stagiaires doivent alors compléter les sources fournis, effectuer les configurations Spring adéquates et mettre au point les tests validant la spécification. L'atelier suivant permet d'intégrer la suite en testant toutes les couches de l'application à une procédure de build. Enfin, avec le dernier atelier, les participants mettent en pratique les concepts de l'AOP.

### Participants :

Cette formation Spring s'adresse aux développeurs JAVA

### Pré-requis :

Bonnes connaissances du langage Java

### Travaux pratiques :

Les nombreux travaux pratiques proposés lors de cette formation Spring fourniront aux participants une expérience pratique du développement d'applications avec Spring.

**Durée : 3 jours (21 heures)    Référence : SFSG**

### Contenu de la formation :

#### Introduction à Spring

- Concepts de conteneur léger
- Vue d'ensemble et exemples d'utilisation
- Pattern "Inversion de Contrôle (IoC) ; Injection de dépendance"
- Tests unitaires en isolation
- Approche MVC avec Spring MVC

**ATELIER :**

*Mise en place de l'IDE Eclipse, 1ère configuration Spring, Tests via Junit*

### **Mise en oeuvre de Spring**

Les Beans, BeanFactory et ApplicationContext  
Modes singleton ou prototype  
Gestion des propriétés, "collaborators"  
Méthodes d'injection de dépendance  
Configuration de Beans spécifiques à Spring, cycle de vie  
Définition de Bean abstrait et héritage

**ATELIER :**

*Mini-application médiathèque en ligne : Différentes types de configurations de bean, Complétion de code et écriture d'une classe de test*

### **Spring et l'accès aux données (JDBC et Hibernate)**

Pattern DAO avec JDBC et les Classes abstraites de Spring  
Configuration de la SessionFactory d'Hibernate dans le contexte d'application Spring  
IoC : Hibernate Template et Hibernate Callback  
Implémentation DAO avec les APIs Hibernate  
Démarcation de transactions par programmation et déclaration

**ATELIER :**

*Implémentation de la couche DAO de la mini-application avec Hibernate, Démarcation des transactions des services métier*

### **Support de Spring pour les Tests**

Concepts de l'eXtreme programming  
Tests unitaires et test d'intégration  
Indépendance de Spring vis à vis des outils de test  
Apports de Spring : Cache de l'ApplicationContext, Injection de données de test, et rollback des transactions

**ATELIER :**

*Mise en place des suites de tests d'intégration pour la mini-application. Intégration de l'exécution des tests dans la procédure de build*

### **Programmation orientée aspect (AOP)**

Concepts de l'AOP  
Terminologie utilisée : aspect, joinpoint, advice, shortcut, weaving  
Les différents types d' « advice » : around, before, throws, after  
Objectifs et fonctionnalités AOP de Spring  
Intégration avec le framework AspectJ

**ATELIER :**

*Mise en place d'un aspect profilable permettant de faire du profiling d'application*