

## Formation Spring Initiation + approfondissement : Applications web MVC et persistance avec Spring Data JPA

<b>Durée :</b>	5 jours
<b>Public :</b>	Développeurs Java
<b>Pré-requis :</b>	Maîtrise de la programmation orientée objets Java - Notions HTML/CSS/JS
<b>Objectifs :</b>	Construire des applications web en utilisant Spring Web et Spring Boot et gérer efficacement la couche de persistance de données
<b>Sanction :</b>	Attestation de fin de stage mentionnant le résultat des acquis
<b>Taux de retour à l'emploi:</b>	Aucune donnée disponible
<b>Référence:</b>	JAV101963-F
<b>Note de satisfaction des participants:</b>	4,47 / 5

### Comprendre les frameworks Spring

Panorama des frameworks Spring  
Spring Framework : inversion de contrôle et injection de dépendances  
Spring Web : présentation, usages  
Spring Boot : simplification de la configuration et du déploiement  
Organisation des dépendances Maven

**Atelier pratique : installation de l'environnement de développement et d'exécution - construction de la structure minimale d'un projet Spring Web MVC avec ou sans Spring Boot**

### Construire une application Spring Web MVC

Définition de contrôleurs et cycle de vie  
Gestion du mapping des actions  
Utilisation d'un framework de vues : JSP ou Thymeleaf ou Velocity Templates  
Composants graphiques, créations de formulaires  
Validation de formulaires  
Gestion des messages d'erreurs  
Listeners, Intercepteurs  
Internationalisation (i18n) dans une application Spring MVC  
Gestion de l'upload

**Atelier : Implémentation de plusieurs formulaires : création, validation, traitement**

### Persister des données en mémoire

Espaces de persistance : page, requête, session, application  
Manipulation de la session : ajout et récupération d'attributs  
Définition et utilisation de variables d'application  
Création et récupération de cookies

**Atelier : Utilisation des espaces de persistance disponibles**

## **Sécuriser l'accès avec Spring Security**

Aspects sécurité dans une application Java EE  
Spring Security : présentation, fonctions  
Architecture du framework  
Dépendances, configuration par annotations  
Mécanismes d'authentification (authenticationProvider)  
Configurer des rôles et gérer des autorisations

**Atelier : mettre en place Spring Security dans une application Web**

## **Optimiser le chargement des pages**

Gestion du chargement des pages  
Requêtes asynchrones :

- Communication (objet XMLHttpRequest)
- Envoi / traitement de la réponse
- Tags spécifiques

**Atelier : Implémentation d'actions asynchrones et chargements partiels de données**

## **Implémenter des web services REST**

Architecture REST : fonctionnement, méthodes HTTP  
Spring RestController : annotations, mappings  
Retour d'objets  
Ecriture de clients REST

**Atelier : écriture d'un @RestController et test avec Postman**

## **Configurer un projet Spring Boot pour intégrer Spring Data JPA**

Spring Data JPA : Présentation, fonctionnalités, dépendances Maven  
Configuration d'un projet Spring Boot  
Propriétés par défaut et paramétrage  
Gestion des logs avec Logback

**Atelier : Intégration de Spring Data JPA dans un projet, configuration des traces**

## **Réaliser le mapping des entités et des opérations**

Mapping des tables et gestion des clés primaires (simples, composées)  
Mapping des types de bases, propriétés des colonnes  
Gestion de la concurrence : optimistic (versioning), pessimistic  
Gestion des relations : OneToMany/ManyToOne, OneToOne, ManyToMany  
Paramétrage des cascades  
Gestion des collections : Map, Set, List,...

Mapping de l'héritage

Stratégies de chargement : Lazy ou Eager

**Atelier : Réalisation d'un schéma global de mapping d'une base de données, opérations CRUD (Create Read Update Delete)**

### **Ecrire des requêtes JP-QL ou SQL**

Interface JpaRepository et ses dérivées, ancêtres : méthodes disponibles

Nommage de méthodes pour une auto-génération des requêtes

Requêtes JPQL ou natives avec @Query : jointures, paramètres, fetch

Repository personnalisé et injection de l'EntityManager

Gestion des procédures stockées

**Atelier : Ecriture de repositories et test depuis des services ou des contrôleurs**

### **Maîtriser des concepts avancées**

Cache : fonctionnement, niveaux

Configuration du cache : @Cacheable

Mise en place d'une solution d'audit de tables (historique de modifications)

**Atelier : Implémentation d'une couche complète de persistance - mise en place d'un cache**