
Programme de Formation

Spring Framework : Développer et déployer des applications d'entreprise performantes

Description

Formation pratique pour maîtriser Spring et Spring Boot dans le développement d'applications d'entreprise : configuration, persistance, API REST, sécurité et déploiement.

Points Forts

Apprenez à concevoir des applications robustes et évolutives grâce à des ateliers pratiques combinant Spring MVC, persistance de données, API REST et sécurité JWT. Développez vos compétences opérationnelles avec des cas concrets d'entreprise.



Public visé

- Développeurs Java
- Architectes Java SE / Java EE
- Chefs de projets techniques
- Ingénieurs logiciels
- Consultants techniques spécialisés en développement d'applications d'entreprise



Objectifs pédagogiques

- Comprendre le fonctionnement de Spring
- Connaître les bonnes pratiques de développement avec Spring
- Savoir configurer les composants d'une application avec Spring
- Implémenter un service d'accès aux données
- Gagner en efficacité avec Spring Boot
- Sécuriser l'application

Organisation

Durée : 35 heures

Mode d'organisation : Mixte

Contenu pédagogique



Description

Jour 1 :

Comprendre le fonctionnement de Spring

- Rappel sur le Framework Spring, enjeux et objectifs
- Spring Boot, extension de Spring
- Installation de Spring Tools Suite (Eclipse)
- Maven ou Gradle
- Exemples de travaux pratiques : les participants réalisent un exercice et doivent installer la pile JEE nécessaire sur leur poste de développement, installer et configurer STS (Spring Tools Suite) pour améliorer la productivité.

Gagner en efficacité avec Spring Boot



- Identifier les principales dépendances du framework (data, web, ...)
- Identifier les dépendances complémentaires utiles (drivers pour les bases de données, outils de build ou d'automatisation comme Lombok, ...)
- Enjeux liés à la gestion et cycle de vie des dépendances (versions, vulnérabilités, gestionnaires d'artefacts privés en entreprises)
- Les modes de configuration de Spring Boot (gestion des environnements, fichier properties ou yaml)
- Exemples de travaux pratiques : Trouver les dépendances utilisées à partir de start.spring.io, lancer l'exécution d'une première application Spring Boot.

Jour 2 :

Savoir configurer les composants d'une application avec Spring

- Annotations Spring
- Auto-Configuration, custom configurations
- Développement en couches : contrôleur, service, persistance
- Injection de dépendances
- Exemples de travaux pratiques : Création d'une application SpringBoot en utilisant les annotations spécifiques, Création d'un CommandLineRunner pour inspecter les Beans créés Création d'un service simple, pour gérer l'injection dans un contrôleur.

Connaître les bonnes pratiques de développement avec Spring

- Comparaison rapide des modèles blocking I/O (Spring MVC) ou non blocking I/O (WebFlux)
- Principe MVC : Modèle Vue Contrôleur
- Ajouter Thymleaf aux dépendances
- Premier template
- Exemples de travaux pratiques : Création d'un contrôleur simple pour router vers un template de base. Modification d'un template.

Jour 3 :

Utiliser un moteur de template pour industrialiser la production des pages

- Syntaxe Thymleaf : conditions, itérations
- Passer le modèle à la vue
- Récupérer les données de la vue
- Exemples de travaux pratiques : Création d'un template conditionnel et itératif, passage du modèle à la vue : création d'un formulaire, récupération des données de la vue : implémentation d'un POST.

Implémenter un service d'accès aux données

- ORM Pattern
- Entities / Repositories avec différents types de requêtes (Spring Data fonctionnal queries methods, JPQL, native queries, appel de procédures stockées)
- Utilisation des couches : service, persistance
- Exemples de travaux pratiques : Création d'une simple table dans une base de données relationnelles, Création d'un service pour ajouter, modifier, supprimer des lignes dans la table.

Jour 4 :

Gérer les relations entre les entités et utiliser les dépôts de données

- Relations : ManyToOne, OneToMany, ManyToMany, OneToOne
- Récupération des entités : lazy ou eager
- Custom repositories
- Exemples de travaux pratiques : A partir du schéma simple précédent, ajouter des entités afin de répondre à une fonctionnalité donnée, créer un Custom Repository afin de répondre à une exigence de récupération de donnée non standard.

Travailler avec des bases orientées Documents

- Rappel sur le No-SQL
- Installation d'un serveur de données No-SQL,
- Création des entités
- Manipulation des collections
- Exemples de travaux pratiques : A partir du schéma simple précédent : configuration d'une application Spring Boot avec les dépendances nécessaires, création d'un « domaine » simple, exploitation d'un domaine en REST avec Postman, exécution d'une requête « géo-centrée ».

Jour 5 :

Mettre en place une architecture REST et produire une API

- Retour sur l'architecture RESTful
- Configuration des endpoints d'API
- HTTP Requests / HTTP Responses / HTTP Statuses
- Implémentation du pattern DTO
- Mode synchrone ou asynchrone
- Aller plus loin avec la Programmation Orientée Aspects (AOP)
- Exemples de travaux pratiques : Création d'un contrôleur de type REST et définition des routes, utilisation des décorateurs pour récupérer les données en entrée des routes, création d'un DTO pour transformer les données en entrée ou en sortie.

Sécuriser l'application

- Objectifs et principes
- Authentification
- Autorisations
- Exemples de travaux pratiques : Création d'un contrôleur pour l'enregistrement d'un nouvel utilisateur, configuration des routes sans authentification, protection des routes avec une authentification de type JWT et Bearer.

Prérequis

- Bonnes connaissances du langage Java
- Un questionnaire de positionnement sera réalisé au préalable pour adapter la formation à vos besoins

Modalités pédagogiques

- Exercices concrets
- Cas pratiques
- Quiz d'évaluation des connaissances

Moyens et supports pédagogiques

- Mise à disposition de postes de travail
- Vidéoprojecteur, tableau blanc, supports numériques
- Documentation pédagogique numérique



Modalités d'évaluation et de suivi

- Positionnement en amont de la formation :
 - Un quiz de consolidation des pré-requis sera administré en amont de la formation
- Suivi « pendant » :
 - Feuilles de présence
 - Exercices pratiques
 - Évaluation « fin de formation »
 - Évaluation des acquis en fin de formation
 - Formulaires d'évaluation de la formation
- Évaluation à froid :
 - Suivi post-formation : Questionnaire de satisfaction à j+30