

Développement professionnel avec Java Spring (JSPRINGFR, 4 jours)

Description

Ce cours certifié SETC offre aux programmeurs Java une introduction au populaire framework Spring. Le cours débute avec les fondements de l'utilisation de Spring. Les participants vont être introduits aux fichiers de configuration de beans en XML. On commencera avec des objets simples pour continuer avec les collections, les constructeurs et autres configurations avancées. La programmation orientée aspect sera présentée, de même que quelques APIs de Spring tels que DAO, SpringVMC et JEE.

Tarifs

- Tarification: \$3,750/person
- Rabais de 10% lorsque vous inscrivez 3 personnes.

Plan de cours

Concepts d'architecture essentiels

Aperçu de l'architecture logique

La couche de données : JDBC et Hibernate

Le référentiel de données : Spring Data, les référentiels et JPA

Le modèle de conception MVC : De la demande à la réponse

Le chef d'orchestre : Exploration du rôle du contrôleur

De l'action à l'objet : Vues, modèles de vues, contrôleurs et modèles

Le rôle des modèles de vue : Que sont Thymeleaf et d'autres langages similaires

Spring Boot et Maven : Coordonner les éléments de base

Assembler le tout dans une conception d'entreprise

Démarrage de votre projet Spring

Utilisation de Spring Boot et Spring Initializr

Exploration du fichier pom.xml

Comprendre le processus de construction et les dépendances du runtime Java

Définition des propriétés de l'application : Propriétés du serveur, dialectes, pilotes, etc.

Création d'Entity Beans : Définition des mappages de tables, des ID et des relations entre entités

Création de référentiels de données avec Spring Data

Implémentation du service métier : Déclarer les interfaces de service

Création d'un premier contrôleur SOAP

Déploiement et exécution d'une application

Construire sur les bases

En faire plus avec les Entity Beans

Ajout de services métier à une application

Comprendre la gestion des erreurs de Spring

Configuration de la gestion des erreurs de Spring

Personnalisation du comportement de l'état de la réponse HTTP

Explorer le mappage d'URL et un contrôleur

Maîtriser la configuration de Spring

Configuration des ruches de ressources : Regroupements de ressources et actifs Web

Comprendre le traitement des modèles de vue

Création du répertoire des modèles

Modifications à apporter au fichier pom.xml

Exploration des options de configuration de Spring : XML et basé sur le code

Créer la classe de configuration : Définir le contexte de l'application

Ajout de paquets de ressources dans la configuration

Définition du modèle de vue dans la configuration : Le résolveur et le moteur

Mise en œuvre de la journalisation et exécution de l'application

Création de modèles Thymeleaf

A propos du traitement des modèles Thymeleaf

Comprendre le rôle de la classe View Model

Construire le premier modèle de vue

Création de modèles basés sur des modèles

Ajout de contenu aux modèles

Importation de paquets Java dans un modèle Thymeleaf

Utilisation d'objets Java dans un modèle Thymeleaf

Implémentation de la logique conditionnelle

Implémentation de segments répétés

Ajout d'un support de ressources à un modèle Thymeleaf

Ajout d'un support d'erreur à un modèle Thymeleaf

Assembler le tout

Introduction à la gestion des erreurs

Création d'un modèle Thymeleaf pour afficher les erreurs

Utilisation des conseils du contrôleur pour personnaliser la gestion des erreurs

Création de classes d'exceptions personnalisées

Utilisation de l'attribut `ExceptionHandler`

Configuration de Spring : Utilisation des résolveurs d'exceptions mappés

Tout mettre en place

Approfondissement de la validation et du formatage

Comprendre la nature distribuée de la validation des données

À propos de Spring et de son utilisation de la validation Hibernate

Implémentation de la validation de l'entité `Being`

Implémentation de la validation du modèle de la vue

Affichage des messages d'erreur dans une vue Thymeleaf

Création de formateurs personnalisés pour un formatage cohérent

Création de formateurs de conversion pour un formatage cohérent

Ajout de formateurs à la configuration de Spring

Explorer les intercepteurs et la sécurité

L'architecture des intercepteurs de Spring

Pourquoi les intercepteurs sont essentiels à l'architecture d'entreprise

Création de la classe `Interceptor`

Configuration de Spring : Ajout de l'intercepteur

Explorer la sécurité de Spring

Ajout de Spring Security à la configuration

Définition des ressources sécurisées et non sécurisées

Tout mettre en place