

JAVA

Java EE 7: Front-end Web Application Development

5 jours (35h00) | JAVA003 | Num form : form-11 | Perfectionnement / Avancé

INFORMATIQUE / DÉVELOPPEMENT / JAVA

À l'issue de ce stage vous serez capable de :

- Développer et exécuter une application de la technologie EJB
- Créer des applications de la technologie Java EE avec la plate-forme Java EE 7
- Identifier les services fournis par un serveur d'applications
- Intégrer, déployer et déboguer des applications d'entreprise
- Créer et utiliser les annotations Java
- Sélectionner le bon profil Java EE pour une application donnée
- Concevoir des applications pour utiliser l'injection de dépendances
- Utiliser des IDE et des serveurs d'applications pour le développement de Java EE

Niveau requis :

- Connaissance de base des concepts de bases de données et de la syntaxe SQL
- Expérience avec Java SE ou certification de programmeur Java
- Comprendre les principes orientés objet
- Programmation Java SE 8
- Savoir créer des pages Web optimisées en HTML, CSS et JavaScript

Public concerné :

- Développeurs d'applications
- Développeurs JEE
- Intégrateurs Système

Programme :

Plate-forme Java, Enterprise Edition

- La plate-forme Java EE
- Les besoins des développeurs d'applications d'entreprise Spécifications de Java EE
- Une comparaison des services et des bibliothèques Le profil Web de Java EE
- Les niveaux et couches d'applications Java EE

Outils de développement et applications d'entreprise

- L'objectif d'un serveur d'applications
- Lancer et arrêter WebLogic Server Propriétés des composants Java EE
- Le processus de développement d'une application Java EE
- Configuration et intégration des applications Java EE

JavaBeans, annotations et journalisation

- Les fonctionnalités de Java SE utilisées dans les applications Java EE
- Création de composants POJO JavaBeans
- Utilisation de la journalisation
- Utilisation des annotations Java fréquentes
- Développement d'annotations personnalisées
- Le rôle des annotations dans les applications Java EE

Architecture Web de Java EE

- Le modèle requête-réponse HTTP
- Différences entre les composants Java Servlets, JSP et JSF
- La superposition des applications et le modèle MVC
- Éviter les problèmes de sécurité liés au thread dans les composants Web
- Utiliser le langage d'expression

Développer des servlets

- L'API Servlet
- Les API de requête et de réponse
- Définir des en-têtes de réponse
- Deux méthodes pour créer un corps de réponse
- Téléchargement de fichiers à l'aide d'un servlet
- Transfert du contrôle et transmission des données
- Utilisation de l'API de gestion des sessions

Développement avec les JavaServer Pages

- Le rôle de JSP en tant que mécanisme de présentation Création de pages d'affichage JSP
- Traitement des données à partir de servlets dans une page JSP Utilisation des bibliothèques de balises

Services Web JAX-RS

- La nécessité des services Web Conception d'un service Web RESTful
- Créer des méthodes qui suivent les règles prescrites du comportement de la méthode HTTP
- Créer une ressource JAX-RS et des classes d'applications
- Consommer des types de requêtes et d'autres paramètres
- Produire et consommer des données complexes sous forme de codes d'état HTTP XML

Clients Java RESTful

- Clients pré-JAX-RS 2 : HttpURLConnection et l'API client Jersey
- L'API client JAX-RS 2

Applications HTML5 avec JavaScript et AJAX

- Manipulation de DOM HTML avec JavaScript
- Clients RESTful avec JavaScript (AJAX)
- Limites des clients JavaScript
- La stratégie Same-Origin et le CORS

WebSocket et l'API Java pour le traitement JSO

- Limites des services Web
- Explication de WebSocket
- Création de WebSockets avec Java
- WebSocket côté client avec JavaScript
- WebSocket côté client avec Java
- Consommation de JSON avec Java
- Production de JSON avec Java

Mise en œuvre d'une stratégie de sécurité

- Sécurité gérée par des conteneurs
- Rôles et responsabilités des utilisateurs
- Créer une stratégie de sécurité basée sur les rôles
- L'API de sécurité

Modèles de composants POJO et EJB-Lite

- Le rôle des composants EJB dans les applications Java EE
- Les avantages des composants EJB
- Caractéristiques opérationnelles des beans de sessions sans état et avec état
- Création de beans de sessions
- Création de clients de beans de sessions

L'API Java Persistence

- Le rôle de l'API Java Persistence dans les applications Java EE
- Bases du mapping objet-relationnel
- Les éléments et l'environnement du composant d'une entité
- Le cycle de vie et les caractéristiques opérationnelles des composants d'entités

Mise en œuvre d'une stratégie de transaction

- Sémantique de la transaction
- Comparaison de la portée des transactions de programmation et déclaratives
- Utilisation de JTA pour délimiter les transactions par programmation
- Mise en œuvre d'une stratégie de transactions gérées par des conteneurs