

Mis à jour le 17/05/2024

S'inscrire

Formation Java : Programmation avancée

4 jours (28 heures)

Présentation

Si vous maîtrisez les fonctionnalités basiques de Java, cette formation est faite pour vous. Vous y découvrirez les concepts avancés du langage comme la programmation réflexive ou encore la programmation concurrente.

Vous découvrirez comment tester, [débugger](#) et [optimiser](#) vos applications Java efficacement. De plus, nous vous présenterons les frameworks les plus populaires : Spring, Hibernate et JSF.

Notre formation Java avancé vous donnera les techniques nécessaires à la réalisation de programmes Java complexes. Vous apprendrez à optimiser et à sécuriser vos applications de la meilleure des manières.

Comme pour toutes nos formations, celle-ci vous présentera la dernière version de Java ([JDK 22](#)).

Objectifs

- Maîtriser les nouvelles fonctionnalités avancées du langage Java
- Savoir utiliser les principaux frameworks et bibliothèques Java
- Appliquer les techniques de POO pour construire des classes et créer des objets
- Développer des applications Java en utilisant des bases de données relationnelles
- Sécuriser son application Java et comprendre le chargement des classes
- Tester, débbugger et optimiser son application
- Comprendre et appliquer la programmation réflexive et concurrente

Public visé

- Développeurs
- Architectes techniques
- Chargé de développement des applications

- Chef de projet en informatique

Pré-requis

- Avoir suivi notre formation [Java](#)

Programme de notre formation Java : Programmation avancée

Rappel des nouveautés de Java 19

- Aperçu des modèles d'enregistrement
- Prise en charge de l'architecture de jeu d'instruction Linux/RISC-V
- Aperçu d'API de fonction et de mémoire étrangère
- Aperçu des threads virtuels
- Quatrième incubation d'API vectorielle
- Troisième aperçu de la correspondance de motifs pour switch

La programmation réflexive

- Qu'est-ce que la programmation réflexive ?
- La réflexivité sur Java
- Les annotations
- Obtenir l'objet d'une Class class
- Déterminer l'objet de la class
- Constructors Methods et fields
- L'API reflection
- L'instantiation et l'invocation dynamique

La programmation concurrente et le multi-threading

- Les threads et la programmation multithread
- Le cycle de vie d'un thread
- Les problèmes et les bénéfices du multithreading
- Le Multithreading avec GUI
- Semaphore et synchronisation entre les threads
- Interrompre les threads
- Aperçu des collections concurrentes
- Créer et gérer ses threads
- Gérer la concurrence

Java Management Extension (JMX)

- L'architecture de JMX
- Présentation de la couche instrumentation (les MBeans)

- Les différents concepts associés (Dynamic MBeans, Open MBeans, Model MBeans et MXBeans)
- La couche agent
- La couche services distribués
- La console d'administration (JConsole)
- Contrôler une JVM

Java Message Service (JMS)

- Pourquoi JMS ?
- JMS 1 et 2 API
- Gestion des messages (création, priorisation, demande de réponse, délai de livraison et d'expiration)
- Les types de messages et les objets message
- Les interfaces et les classes JMS
- Les différents types de communication (communication de point à point, publish subscribe, par queue...)
- Filtrer ou grouper les messages
- Sécuriser et Administrer JMS

Java Server Faces (JSF)

- Présentation des concepts JSF
- Le modèle MVC
- Configurer une application JSF
- Le cycle de vie d'une requête
- Facelets et Ajax

SQL et accès aux bases de données avec JDBC

- Rappel sur les bases de données relationnelles
- Différentes interfaces dans l'API JDBC
- Requête SELECT de base
- Clause WHERE
- Clause ORDER BY
- Fusionner les données de plusieurs tables : INNER JOIN
- Instruction INSERT
- Déclaration UPDATE
- Déclaration DELETE
- Installation de MySQL et de MySQL Connector/J
- Création d'une base de données dans MySQL

Les expressions Lambda

- Interfaces, expressions lambda et classes internes
- Interfaces
- Expressions lambda
- Classes internes
- Chargeurs de services

- Proxies
- Classe anonyme, concept d'expression lambda
- Manipuler les collections grâce aux lambda expressions
- Code SQL : Les Listeners

Introduction à JSTL

- Qu'est-ce que JSTL (Java Server page Standard Tag Library) ?
- Classifications des tags
- Fonctions principales de JSTL
- Utilisation des balises personnalisées
- Comment réduire son code ?
- Comment rendre son code plus performant avec JSTL ?

Servlets

- Qu'est-ce qu'un servlet ?
- Méthodes HTTP ; GET, POST, PUT, DELETE, TRACE, OPTIONS
- Cycle de vie des servlets
- Contexte configuration du servlet
- Développer une application utilisant des servlets

Mise en réseau

- Manipulation des URLs
- Lire un fichier sur un serveur Web
- Établissement d'un serveur simple à l'aide de sockets de type Stream
- Établissement d'un client simple à l'aide de sockets en continu
- Interaction client/serveur avec les connexions Stream Socket
- Datagrammes : Interaction client/serveur sans connexion
- Client/Serveur/Serveur multithreaded

Les bibliothèques

- Présentation de Java Standard Library
- Utiliser la machine virtuelle Java (JVM)
- Programmer plus simplement grâce à Java Standard Library
- Réutiliser des composants Java avec Apache Commons
- Manipuler des fichiers JSON, CSV et XML avec la bibliothèque Jackson

Débuter avec Spring

- Présentation de Spring
- Les capacités du Framework
- La programmation orientée aspect
- L'API REST

Découvrir Hibernate

- Présentation d'Hibernate
- Configuration du framework
- La gestion d'une occurrence
- L'héritage des classes
- Générer son code

Tester ses applications

- Les bonnes pratiques de test
- Les différents types de test
- Frameworks pour effectuer des tests unitaires
- Test Driven Development
- Troubleshooting

Sécuriser ses applications web

- Sécurité déclarative et programmatique
- Les rôles
- Authentification de base
- Authentification Digest
- Authentification par formulaire
- Formulaire de connexion standard
- Contraintes de sécurité
- Collection de ressources Web

Sociétés concernées

Cette formation s'adresse à la fois aux particuliers ainsi qu'aux entreprises, petites ou grandes, souhaitant former ses équipes à une nouvelle technologie informatique avancée ou bien à acquérir des connaissances métiers spécifiques ou des méthodes modernes.

Positionnement à l'entrée en formation

Le positionnement à l'entrée en formation respecte les critères qualité Qualiopi. Dès son inscription définitive, l'apprenant reçoit un questionnaire d'auto-évaluation nous permettant d'apprécier son niveau estimé sur différents types de technologies, ses attentes et objectifs personnels quant à la formation à venir, dans les limites imposées par le format sélectionné. Ce questionnaire nous permet également d'anticiper certaines difficultés de connexion ou de sécurité interne en entreprise (intraentreprise ou classe virtuelle) qui pourraient être problématiques pour le suivi et le bon déroulement de la session de formation.

Méthodes pédagogiques

Stage Pratique : 60% Pratique, 40% Théorie. Support de la formation distribué au format numérique à tous les participants.

Organisation

Le cours alterne les apports théoriques du formateur soutenus par des exemples et des séances de réflexions, et de travail en groupe.

Validation

À la fin de la session, un questionnaire à choix multiples permet de vérifier l'acquisition correcte des compétences.

Sanction

Une attestation sera remise à chaque stagiaire qui aura suivi la totalité de la formation.