

Java, programmation avancée

Cours Pratique de 5 jours - 35h

Réf : JAP - Prix 2024 : 2 970CHF HT

L'objectif principal est de présenter un panorama des principales bibliothèques concernant la gestion des threads, les communications via le réseau, l'administration et la supervision des applications. Le cours présente aussi les principes architecturaux qui sous-tendent la construction compositionnelle d'applications.

OBJECTIFS PÉDAGOGIQUES

À l'issue de la formation l'apprenant sera en mesure de :

Mettre en œuvre la programmation concurrente à base de threads

Mettre en œuvre différentes techniques de communication (Socket, RMI, JMS)

Administrer une application Java via JMX, surveiller la JVM

Mettre en œuvre la programmation réflexive et à base d'annotations

TRAVAUX PRATIQUES

Ce cours illustre les principes exposés par la construction progressive d'une application en Java.

LE PROGRAMME

dernière mise à jour : 09/2021

1) La programmation concurrente

- Les concepts de la programmation multithread : le modèle d'activités de Java (Runnable et Thread).
- La création/destruction des threads. Ordonnement des threads.
- La synchronisation des threads. Le verrouillage des méthodes et des instructions (synchronized). Les moniteurs.
- Quelques problèmes du multithread : l'interblocage (caractérisation, évitement, prévention, détection), la famine.
- Les extensions du modèle introduites dans Java.5 (Callable<T>, Future<T>, ExecutorService. Les nouvelles collections.
- Le modèle Fork/Join de Java 7. Les extensions de Java 8 (CompletableFuture).
- Les divers outils de gestion de la concurrence : les verrous partagés/exclusifs, les sémaphores, les barrières cycliques.

Travaux pratiques : Construction d'une application multithread avec contraintes de concurrence.

2) La communication par socket

- Rappels sur les principaux concepts réseaux.
- La communication en mode connecté.
- Le modèle client/serveur. Serveur séquentiel versus serveur concurrent. Utilisation de la sérialisation.
- La programmation en mode non connecté. Le modèle Peer to Peer.

Travaux pratiques : Programmation de l'étude de cas en mode client-serveur (séquentiel et concurrent).

3) La communication par invocation de méthode distante : RMI

- Les principes généraux des ORB (Object Request Broker).
- Le modèle RMI (les concepts, les interfaces classes de base).

PARTICIPANTS

Développeurs, ingénieurs, chefs de projets proches du développement.

PRÉREQUIS

Bonnes connaissances du langage Java. Expérience requise en programmation Java.

COMPÉTENCES DU FORMATEUR

Les experts qui animent la formation sont des spécialistes des matières abordées. Ils ont été validés par nos équipes pédagogiques tant sur le plan des connaissances métiers que sur celui de la pédagogie, et ce pour chaque cours qu'ils enseignent. Ils ont au minimum cinq à dix années d'expérience dans leur domaine et occupent ou ont occupé des postes à responsabilité en entreprise.

MODALITÉS D'ÉVALUATION

Le formateur évalue la progression pédagogique du participant tout au long de la formation au moyen de QCM, mises en situation, travaux pratiques...

Le participant complète également un test de positionnement en amont et en aval pour valider les compétences acquises.

MOYENS PÉDAGOGIQUES ET TECHNIQUES

- Les moyens pédagogiques et les méthodes d'enseignement utilisés sont principalement : aides audiovisuelles, documentation et support de cours, exercices pratiques d'application et corrigés des exercices pour les stages pratiques, études de cas ou présentation de cas réels pour les séminaires de formation.
- À l'issue de chaque stage ou séminaire, ORSYS fournit aux participants un questionnaire d'évaluation du cours qui est ensuite analysé par nos équipes pédagogiques.
- Une feuille d'émargement par demi-journée de présence est fournie en fin de formation ainsi qu'une attestation de fin de formation si le stagiaire a bien assisté à la totalité de la session.

MODALITÉS ET DÉLAIS D'ACCÈS

L'inscription doit être finalisée 24 heures avant le début de la formation.

ACCESSIBILITÉ AUX PERSONNES HANDICAPÉES

Vous avez un besoin spécifique d'accessibilité ? Contactez Mme FOSSE, référente handicap, à l'adresse suivante psh-accueil@orsys.fr pour étudier au mieux votre demande et sa faisabilité.

- Le service de nommage.
 - Le processus de développement du client et du serveur.
 - Les contraintes de sécurité et de chargement de classes.
- Travaux pratiques : Programmation de l'étude de cas à l'aide de RMI.*

4) La communication par messages : JMS

- Les principes généraux de la communication par messages.
- Le modèle de base (les concepts de JMS, les interfaces et les classes).
- Les différentes formes de messages.
- La communication en mode point à point.
- La communication en mode publish/subscribe.

Travaux pratiques : Programmation de l'étude de cas à l'aide de JMS.

5) L'administration des applications : JMX

- Le modèle JMX (Java Management eXtension) : les concepts, les interfaces et les classes correspondantes.
- Les MBeans et MBeanServers.
- La mise en place d'une couche d'administration.
- La console d'administration (JConsole).
- La communication à l'aide des adaptateurs et des connecteurs.

Travaux pratiques : Mise en place d'une couche d'administration-supervision pour l'étude de cas.

6) La programmation réflexive

- Les objectifs et les principes.
- La découverte dynamique des informations relatives à une classe ou à un objet.
- L'instanciation et l'invocation dynamique.
- La réflexivité et les annotations en Java 5.

Travaux pratiques : Programmation de l'étude de cas en mode client-serveur (séquentiel et concurrent).

7) Un panorama des extensions de Java 5 à...

- Les types : les génériques, les énumérations, l'autoboxing/autounboxing, les records.
- Les lambda-expressions, les interfaces fonctionnelles.
- Les streams de Java 8.
- Les modules de Java 9.

LES DATES

CLASSE À DISTANCE

2024 : 27 mai, 30 sept., 18 nov.