

Durée : 5 jours soit 35 heures

Référence : IF-JAVAC

Public visé :

- Développeurs d'applications JAVA

Pré-requis :

Pour suivre ce stage, il est nécessaire de maîtriser la programmation orientée objet en Java ou d'avoir suivi la formation référencée "JAVA" ou équivalente.

Objectifs pédagogiques :

- Approfondir certains aspects avancés du langage
- Comprendre et utiliser les expressions lambda
- Comprendre les principes de base de la nouvelle API
- Maîtriser la communication par socket
- Comprendre la communication distribuée : RMI
- Maîtriser l'accès aux bases de données avec JDBC
- Comprendre les principes de SWING et utiliser les principaux conteneurs

Modalités pédagogiques :

Session dispensée en présentiel ou téléprésentiel, selon la modalité inter-entreprises ou intra-entreprises sur mesure.

La formation est animée par un(e) formateur(trice) durant toute la durée de la session et présentant une suite de modules théoriques clôturés par des ateliers pratiques validant l'acquisition des connaissances. Les ateliers peuvent être accompagnés de Quizz.

L'animateur(trice) présente la partie théorique à l'aide de support de présentation, d'animation réalisée sur un environnement de démonstration.

En présentiel comme en téléprésentiel, l'animateur(trice) accompagne les participants durant la réalisation des ateliers.

Moyens et supports pédagogiques :

Cadre présentiel

Salles de formation équipées et accessibles aux personnes à mobilité réduite.

- Un poste de travail par participant
- Un support de cours numérique ou papier (au choix)
- Un bloc-notes + stylo
- Vidéoprojection sur tableau blanc
- Connexion Internet
- Accès extranet pour partage de documents et émargement électronique

Cadre téléprésentiel

Session dispensée via notre solution iClassroom s'appuyant sur Microsoft Teams.

- Un compte Office 365 par participant
- Un poste virtuel par participant
- Un support numérique (PDF ou Web)
- Accès extranet pour partage de documents et émargement électronique

Modalités d'évaluation et suivi :

Avant

Afin de valider le choix d'un programme de formation, une évaluation des prérequis est réalisée à l'aide d'un questionnaire en ligne ou lors d'un échange avec le formateur(trice) qui validera la base de connaissances nécessaires.

Pendant

Après chaque module théorique, un ou des ateliers pratiques permettent la validation de l'acquisition des connaissances. Un Quizz peut accompagner l'atelier pratique.

Après

Un examen de certification si le programme de formation le prévoit dans les conditions de l'éditeur ou du centre de test (TOSA, Pearson Vue, ENI, PeopleCert)

Enfin

Un questionnaire de satisfaction permet au participant d'évaluer la qualité de la prestation.

Description / Contenu

Module 1 : Quelques aspects avancés du langage

- Les Inner Classes
- Les classes anonymes
- Les nouvelles boucles for
- L'auto-boxing, auto-unboxing
- Les types énumérés
- La généricité
- Les annotations

Module 2 : Les expressions lambda

- Notion de classe anonyme imbriquée, améliorations apportées par le concept d'expression lambda.

- Les aspects syntaxiques (déclaration, implémentation, passage de paramètres, portée des variables...).
- Le concept de "foncteur" à travers les interfaces "fonctionnelles"
- Utiliser les lambda-expressions pour manipuler les collections.

Module 3 : Nouvelle API Date et Time

- Les difficultés engendrées par les API de Date
- Les principes de base de la nouvelle API.
- La gestion du temps machine avec java.time.Instant et java.time.Duration
- Le temps humain avec java.time.LocalDate, java.time.LocalDateTime, java.time.LocalTime



Module 4 : La communication par socket

- Rappels sur les principaux concepts réseaux
- La communication en mode connecté
- Le modèle client/serveur
- Utilisation de la sérialisation

Module 5 : La communication distribuée : RMI

- Les principes généraux
- Les classes de base
- Le service de nommage
- Le processus de développement du client et du serveur

Module 6 : Accès aux bases de données avec JDBC

- Utilisation de l'API JDBC, des pilotes JDBC
- PreparedStatement et CallableStatement
- Requêtes
- Les transactions
- Utilisation des RowSet
- Le pattern DAO

Module 7 : Introduction à SWING

- Présentation et origine de Swing
- Evolutions des toolkits graphiques Java
- Les principes de base

Module 8 : Principaux conteneurs SWING

- JFrame et JDialog : fenêtres et boîtes de dialogue
- JPanel : conteneur générique
- JScrollPane : conteneur scrollable

Module 9 : Principaux layouts SWING

- Notions de Layout
- Layouts standard

Module 10 : Principaux composants SWING

- JLabel : intégration texte simple ou HTML
- Composants texte : JTextField, JTextArea
- Boutons : JButton, JComboBox

Module 11 : Internationalisation d'une application

- Définir la locale
- Gérer des messages dans des ResourceBundle
- Formater des dates et des nombres

Module 12 : Le multi-tâche

- Lancer un thread en Java
- Problématiques de la programmation multitâche (synchronisation...)
- Créer des variables atomiques
- Utiliser des Read-Write Locks
- Collections Thread-safe
- Concurrent Synchronizers (Semaphore, Phaser...)
- Executors et ThreadPools