

# Développer en Java pour programmeurs C/C++

Cours Pratique de 5 jours - 35h

Réf : LJV - Prix 2024 : 2 970€ HT

Cours intensif qui permet d'acquérir une maîtrise approfondie du langage Java en appliquant les principes de la programmation Objet. Après avoir assimilé le "core language" et l'apport des différentes versions du langage, vous utiliserez les principales bibliothèques, zoomerez sur les concepts d'entrée/sortie, accéderez aux bases de données, notamment avec l'API JDBC.

## OBJECTIFS PÉDAGOGIQUES

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

Mettre en œuvre les principes de la Programmation Orientée Objet

Maîtriser la syntaxe du langage Java

Maîtriser les principales librairies standards Java

Maîtriser un environnement de développement intégré pour programmer en Java

## TRAVAUX PRATIQUES

Les exercices pratiques ont été conçus pour illustrer tous les éléments du langage et pour systématiquement mettre en œuvre les concepts de la Conception Orientée Objet : tous les exercices comportent une phase d'analyse/conception suivie d'une phase de programmation.

## LE PROGRAMME

dernière mise à jour : 07/2021

### 1) Les techniques Objet

- Les principes généraux de la modélisation et de la programmation Objet.
- L'abstraction et l'encapsulation : les interfaces. Les différentes formes d'héritage, le polymorphisme.
- Introduction à la modélisation UML : le modèle statique, le modèle dynamique, le modèle de coopération, les scénarios.

*Travaux pratiques : L'application des concepts à une étude de cas qui sera l'un des fils directeurs des exercices suivants.*

### 2) La programmation Objet avec Java

- Les bases de la syntaxe : variables, types, expressions, instructions, tableaux, structures de contrôle et autoboxing.
- Définition et instanciation des classes. Les champs, les méthodes, les constructeurs, les champs et méthodes statiques.
- Les méthodes à nombre variable d'arguments. Les aspects méthodologiques : la conception des classes.
- Les unités de compilation et packages : le contrôle de la visibilité des classes, le mécanisme d'import.
- Les différentes formes d'héritage : l'extension et l'implémentation.
- Les interfaces et l'implémentation des interfaces.
- Le polymorphisme et sa mise en œuvre.
- La construction de hiérarchies de classes.
- La définition des classes dérivées, les constructeurs, les références.
- La factorisation de code : les classes abstraites.
- L'utilisation simultanée de l'implémentation et de l'extension.
- Les classes abstraites.
- Les types génériques.

## PARTICIPANTS

Concepteur, développeur, ingénieur, chef de projets opérationnel.

## PRÉREQUIS

Bonnes connaissances en programmation C ou C++. Expérience requise en développement logiciel.

## 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.

- Les aspects méthodologiques : le regroupement des constantes, la spécification de services.

*Travaux pratiques* : Prise en main de l'environnement de développement et programmation d'un programme simple. Programmation de l'étude de cas. Conception et construction d'une hiérarchie de classes et d'interfaces. Mise en place du polymorphisme et de la généricité. Introduction des exceptions.

### 3) La conception d'interfaces graphiques (AWT, Swing)

- Concepts de base : principes de la visualisation et de la gestion des événements, quelques classes génériques.
- Visualisation des composants graphiques.
- Les conteneurs et les Layouts : BorderLayout, FlowLayout, GridLayout, GridBagLayout et CardLayout.
- Construction de conteneurs hiérarchiques.
- Quelques composants graphiques : labels, boutons, menus, zones de texte, boîtes à cocher, canevas.
- La gestion des événements. Les Listeners et Adapters.
- L'association de gestionnaires aux composants graphiques.
- Les particularités de Swing.

*Travaux pratiques* : Construction de petites applications ou d'une petite interface graphique pour quelques objets dans l'étude de cas. Construction d'un petit éditeur ou prise en compte des événements dans l'interface graphique de l'étude de cas.

### 4) Introduction à la programmation Web : les applets

- Les principes et les constituants du Web.
- Les Applets : principes, cycle de vie, la classe Applet.
- Intégration d'une Applet dans une page HTML, le passage des paramètres, les problèmes de sécurité.

*Travaux pratiques* : Construction d'une Applet.

### 5) Les entrées/sorties et quelques classes utilitaires

- Les I/O. La hiérarchie des classes d'entrée/sortie. Quelques classes de manipulation des systèmes de fichiers.
- Quelques classes d'entrée/sortie travaillant sur les flots de bytes, sur les flots de char. Les entrées/sorties clavier.
- La sérialisation.
- Les classes d'encapsulation des types.
- Les classes système.
- Les classes de conteneur.
- Les classes d'observateur.

### 6) La connexion aux bases de données : JDBC

- Le modèle de JDBC et les principes généraux.
- Rappels de SQL.
- Connexion à un SGBD.
- Exécution des requêtes et traitement des réponses.
- L'utilisation des requêtes précompilées.
- Les concepts transactionnels (A.C.I.D) : atomicité, cohérence, isolation, durabilité.

*Travaux pratiques* : Sauvegarde/consultation de données dans une BD.

## LES DATES

---

Nous contacter