

## JAVA

# Les bases de la Programmation Java EE

5 jours (35h00) | JAVA001 | Num form : form-9 | Perfectionnement / Avancé

INFORMATIQUE / DÉVELOPPEMENT / JAVA

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

- Maîtriser la syntaxe et les principales APIs du langage Java
- Savoir programmer en Java en toute autonomie
- Pouvoir manipuler les Frameworks les plus utilisés du langage tels que JUnit, log4J ou i18n
- Être en mesure de manipuler des données à l'aide de JDBC
- Savoir appliquer le paradigme de la programmation fonctionnelle
- Découvrir les nouveautés de Java 9

## Niveau requis :

- Disposer d'une expérience d'un langage de programmation (C, C++, VB...)
- Connaître les principes de la programmation orientée objet
- Connaissance des concepts de bases de données relationnelles et du langage SQL

## Public concerné :

- Développeurs
- Chargés de développement d'applications informatiques

## Programme :

### PRÉSENTATION DE L'ÉCOSYSTÈME JAVA

- Introduction à Java et à ses tiers techniques J2SE/J2E/J2ME et la JVM
- Présentation des composants techniques de Java (JDK) ainsi que des principales APIs
- Les outils Java : JavaDoc, Jar et co
- Panorama des principaux IDE : Eclipse, NetBeans et IntelliJ

**L'IDE ECLIPSE**

- Notions de workspace, vues et perspectives
- Présentation des différents wizard
- Extensions avec Eclipse Marketplace
- Les accès aux bases de données et outils serveurs dans Eclipse

**LES BASES DU LANGAGE : LA SYNTAXE**

- Déclaration de variables
- Les opérateurs
- Initialisation
- Instructions de contrôle
- Boucles et itérations
- Notions de visibilité et de variable de classe Vs. variable d'instance

**LES CONCEPTS ORIENTÉS OBJET EN JAVA**

- La classe et ses attributs / méthodes (nombre d'arguments fixe et variable)
- Les constructeurs
- Les références
- L'encapsulation, les imports et packages
- Les interfaces
- Les classes abstraites
- L'héritage
- Unboxing et l'autoboxing

**GESTION DES ERREURS AVEC LES EXCEPTIONS**

- Qu'est-ce que qu'une exception ?
- Les différentes exceptions en Java
- Les blocs try/catch/finally
- Gestion locale ou centralisée
- Créer ses propres exceptions métier

**BIBLIOTHÈQUES STANDARDS (COLLECTION, DATE, ENTRÉES/SORTIES)**

- Gestion des tableaux et collections
- Les dates et calendriers
- Les entrées / sorties de fichiers
- La sérialisation

**ACCÈS AUX SGBDR AVEC JDBC**

- Présentation de la librairie JDBC
- Connexion à une base de données, exécution d'une requête et itération sur les résultats avec les interfaces Connection, Statement et ResultSet
- Requêtes paramétrées et requêtes stockées

**FICHIERS DE PROPRIÉTÉS ET L'API D'INTERNATIONALISATION (I18N)**

- Accès et manipulation des fichiers .properties
- Comment faire des applications en plusieurs langues
- Resource Bundle

**API POUR LES FICHIERS DE LOGS**

- Générer des Logs avec Log4j
- Filtrer les Logs par niveau d'importance
- Configurer la / les destinations où sont générés les logs

**AMÉLIORER LA QUALITÉ D'UNE APPLICATION GRÂCE AUX TESTS UNITAIRES**

- Créer des tests unitaires avec JUnit
- Créer des suites de tests avec JUnit
- Générer des rapports
- Introduction à la notion de couverture de code et exemples de frameworks
- Options avancées de la perspective Debug

**PROGRAMMATION FONCTIONNELLE ET LAMBDA EXPRESSIONS**

- Introduction à la programmation fonctionnelle
- Exemples de Lambda Expressions
- Référentiels et méthodes

**INTRODUCTION AUX NOUVEAUTÉS JAVA 9**

- La modularisation via le projet Jigsaw
- Fabriques pour les collections
- Améliorations de la classe Optional et de l'API Stream
- Un shell Java : REPL jShell