

## **Concepts Objets et programmation Java**

**Durée:** 5 jours

2250 €

17 au 21 mars

12 au 13 mai

1er au 5 septembre

13 au 17 octobre

8 au 12 décembre

### **Public:**

Développeurs, ingénieurs logiciels et architectes d'applications.

### **Objectifs:**

Apprendre le langage Java et assimiler les concepts objet. Utiliser les outils du JDK et les principales API de la Standard Edition 6. Ce cours s'appuie sur le développement d'une application complète mettant en oeuvre une architecture à 3 niveaux (présentation, métier et persistance).

### **Connaissances préalables nécessaires:**

Avoir une pratique de la programmation dans des langages structurés (C, Pascal, Basic, Fortran, Cobol, ...)

### **Programme:**

- Les concepts objet** : Programmation objet, les réutilisables.  
Principe de l'encapsulation. Attributs et méthodes.  
Accesseurs. Différence entre objet et classe. Instanciation.  
Conventions de nommage.
- Introduction à Java** : Philosophie de conception sous-jacente à Java. Les différentes éditions. Les API de la SE 6. Les fichiers sources, le byte-code et la JVM. Première application.
- Syntaxe java** : Les règles d'écritures. Types primitifs. Types abstraits. Déclaration des variables. Opérateurs. Structures de contrôle. Tableaux.
- Les packages** : Rôle des packages. Définir ses propres packages.
- Les classes** : Déclaration de classes, d'attributs et de méthodes.  
Constructeurs. Instanciation. Destruction des objets : le garbage collector.  
Accès aux attributs et méthodes. Les références : this et null.  
Surcharge des noms de méthodes.  
Membres et méthodes de classe : static.  
Les classes composées d'objets. Contrôle d'accès aux membres.
- Les énumérés** : Définition. Exemples.
- Les interfaces** : Définition et déclaration. Utilisation des interfaces.
- L'héritage** : Mécanisme d'héritage. Recherche de méthodes pour une classe dérivée.  
Héritage et instanciation. Conversions standards dans l'héritage.  
Le polymorphisme. Classes et méthodes abstraites.

## **Concepts Objets et programmation Java**

- Les classes internes** : Définition. Caractéristiques principales. Déclaration. Exemples.
- Les exceptions** : Définition. Graphes d'héritage. Gestion des exceptions.
- Les structures de données** : La classe Vector. La classe Stack. L'interface Enumeration. Structures de données ordonnées. Les collections.
- Les génériques** : Définition. Exemples.
- Auto Boxing et Auto UnBoxing** : Objectif. Exemples.
- Les annotations** : Définition. Annotations standards. Exemples.
- Présentation des principaux packages** : Java.lang et java.util. Internationalisation (i18n). Présentation des autres packages. Les entrées/sorties. Mode octet ou mode caractère. Canaux et filtres. Formatage. Sérialisation.
- Le multi-threading.** : Thread et runnable. Priorité et état. Gestion de la concurrence et synchronisation. L'API de concurrence.
- Présentation de JDBC** : Drivers JDBC. Les classes et interfaces en jeu. Les transactions. Le support de SQL.