

## Java : Les bases

### Programmer en Java et maîtriser ses concepts objets

Cette formation constitue un point d'entrée pour toutes les personnes qui souhaitent s'orienter vers le développement Java. Java est à la fois un langage de programmation et une plate-forme d'exécution.

Il est la pierre de base du développement d'applications basées sur la technologie Java EE (WebSphere, WebLogic, JBoss, Tomcat,...).

Notre formation met l'accent sur l'approche objet : la compréhension de ses principes fondateurs et la (bonne) mise en pratique avec Java.

Nous pensons qu'un développeur Objet développe plus vite parce qu'il code moins (et mieux).

La plate forme Java offre un grand nombre d'API que nous ne pouvons raisonnablement pas aborder en cinq jours.

Nous avons pris le parti de présenter les APIs nécessaires pour utiliser une base de données : JDBC, un outil qui sera certainement essentiel pour écrire une première application de gestion.

A la demande, nous pouvons adapter cette formation pour utiliser un autre IDE que Eclipse dans les Travaux Pratiques : Oracle NetBeans, IntelliJ IDEA, IBM RAD ...

#### Détails

- Code : JA-BAS
- Durée : 5 jours ( 35 heures )

#### Public

- Développeurs
- Ingénieurs

#### Pré-requis

#### Objectifs

- Développer une application en Java
- Assimiler les concepts Objet par la pratique de Java
- Manipuler les fichiers
- Savoir gérer les accès concurrents
- Utiliser une base de données avec Java

#### Programme

##### La plate-forme Java

- Historique
- La machine virtuelle
- Compilation et exécution
- JRE et JDK

##### IDE (Eclipse)

- Concepts et organisation
- Création d'un projet
- Création d'une classe
- Compiler, exécuter et mettre au point un programme Java

##### Syntaxe, les bases

- Commentaires
- Identificateurs
- Littéraux
- Variables
- Types primitifs
- Opérateurs de base
- Chaînes de caractères
- Branchements conditionnels
- Boucles
- Fonctions et procédures

- Tableaux
- Packages et Imports
- Classpath et archives java (jar)
- Visibilité

##### Concepts objets, les bases

- Concepts objet : les origines
- Problème avec la programmation structurée
- Concepts objet : principe, propriétés, messages
- Concepts de classe, d'instances
- Encapsulation
- Mise en œuvre avec Java
- L'explication sur les chaînes de caractère
- Marqueurs des attributs
- Marqueurs des méthodes
- Constructeurs
- Utilisation du constructeur, instantiation
- Le mot clé this
- Getters et setters
- Bonnes et mauvaises pratiques usuelles

##### Concepts objets avancés

- Abstraction, objets et classes

- Héritage et redéfinition
- Classe abstraite et méthode abstraite
- Comprendre le polymorphisme
- Le mot clé super
- Surcharge de méthode
- Comprendre le polymorphisme (encore)
- Relation entre les classes
  - Association
  - Agrégation
  - Composition
  - Dépendance
- Réutilisation
- Mise en œuvre avec Java

## Syntaxe et concepts avancés

- Précédence des opérateurs
- Instruction switch
- Break et continue
- Caractères spéciaux
- Tableaux multidimensionnels
- La classe System
- Types énumérés
- Interfaces
- Transtypage (cast)
- Le mot clé instanceof
- Collections
- Wrapper de primitifs
- Auto-boxing et auto-unboxing
- Types génériques
- Collections génériques
- Iterateurs
- Boucle sur un Iterable
- Méthode « varargs »
- Comparable et Comparator
- Classe interne

## Exceptions

- principes, utilisation
- Checked exception, Error exception, Runtime exception
- Pile d'appel
- Interception
- Gestion des exceptions
- Propager l'exception
- Traitement de l'exception
- Le bloc finally

## Entrées/Sorties

- Entrées/Sorties
- Sorties

- Package java.io
- Flux fichiers
- Flux de données
- Flux objet
- Sérialisation

## Introspection et réflexion

- Annotations
- Principe, utilisation
- Création d'annotations
- Annotation de code
- Quelques annotations importantes

## Logging

- Présentation
- APIs de logging
- Composants Log4j
- Level, Logger, Appenders, Layouts
- Configuration

## Multitâches

- Motivation
- Thread simple
- Runnable
- Partage de ressources
- Gestion des threads
- Bonnes pratiques

## JDBC

- Architecture et drivers JDBC
- Bases de données et drivers
- Chargement du driver
- Connexion à la base
- Exécution de requêtes SQL
- Parcours du ResultSet
- Requête paramétrée
- Persistance d'objet
- Principe du mapping objet/relationnel
- Illustration avec Hibernate

## Architecture et conception

- Modèle MVC
- Design Patterns
  - Singleton
  - Façade
  - Observateur (Observer)
  - Etat
  - Fabrique abstraite
  - Visiteur
  - Inversion de contrôle

## Modalités

- **Type d'action** :Acquisition des connaissances
- **Moyens de la formation** :Formation présentielle – 1 poste par stagiaire – 1 vidéo projecteur – Support de cours fourni à chaque stagiaire
- **Modalités pédagogiques** :Exposés – Cas pratiques – Synthèse
- **Validation** :Exercices de validation – Attestation de stages

