Phirio

Concepts objets et programmation Java

LJ001

Durée: 5 jours

Public :

Développeurs, ingénieurs logiciels et architectes d'applications.

Objectifs :

Apprendre le langage Java et assimiler les concepts objets. Utiliser les outils du JDK et les principales API. Maitriser la programmation objet, et la syntaxe Java. Tous les concepts sont illustrés par des travaux pratiques :soit sur des exemples de base, soit sur des exercices plus complets. Un projet global permettant de mettre en oeuvre l'ensemble des concepts abordés est réalisé tout au long de la formation au fur et à mesure de l'acquisition des concepts.

Connaissances préalables nécessaires :

Connaissance d'un langage de programmation structuré

Programme :

Les concepts objet

Programmation orientée objet, les réutilisables. Apports des langages objets.
Fondamentaux de la POO.
Définitions : Identité, classification, classe, héritage, polymorphisme, encapsulation
Principe de l'encapsulation, attributs et méthodes.
Relations entre objets.
Accesseurs. Différence entre objet et classe. Instanciation. Conventions de nommage.

Introduction à Java

Philosophie de conception sous-jacente à Java. Les différentes éditions.
Présentation JSE, du jdk. Les API.
Principe des fichiers sources, le bytecode et la JVM.
Le cycle de compilation.
Présentation des différents modes d'exécution. Contrôles lors de la compilation et de l'exécution

Atelier : réalisation d'une première application. Prise en main de l'environnement de développement.

Syntaxe java

Bases du langage.
Les règles d'écritures.
Etude des différents types de données, les déclarations de variables.
Présentation des types primitifs, des types objets et des types abstraits, conversion entre types numériques, les constantes.
Principaux opérateurs sur les types primitifs : arithmétiques, d'incrémentation, décrémentation, relationnels et booléens, binaires.
Opérateurs d'évaluation.
Etude des règles de priorité entre les opérateurs.
Traitement dec chaînes de caractères : concaténation, conversion, traitement.
Structures de contrôle : règles de définition d'une séquence d'instructions Java.
Présentation des structures de contrôle conditionnelles (if-else, switch) et itératives (while, do-while, for).
Tableaux : exemples de déclaration de tableau, création et initialisation. Tableaux multi-dimensionnels.

Atelier : mise en oeuvre sur des exemples simples

Les packages

Rôle des packages. La classe System. Définir ses propres packages.

Atelier : création et utilisation par import d'un package applicatif.

Les classes : l'objet vu par Java

Présentation des concepts orientés objet (classe, attribut, constructeur, héritage, ...)
Procédures de déclaration de classes, d'attributs et de méthodes.
Définition de constructeur et de l'instanciation.

Atelier : exercices de prise en main et manipulation de classes Java.

Mécanisme de 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.
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.

Bibliothèques standards

Principe de l'import.
Les structures de données.
La classe Vector. La classe Stack. L'interface Enumeration. Structures de données ordonnées.
Les génériques : définitions et exemples.
Les annotations : principe et mise en oeuvre

Apport des Design Pattern

Principes des solutions de conception cataloguées.
Méthodologie: définition des besoins techniques, des classes "types" du pattern, des collaborations entre classes.

Atelier : exemples de mise en oeuvre de patterns classiques.

Les classes internes

Définition de classe interne. Caractéristiques principales. Déclaration. Exemples.

Les exceptions et erreurs

Définition. Graphes d'héritage.
Présentation du mécanisme de gestion des exceptions, des différents types d'exception
Zoom sur les exceptions contrôlées.

Atelier : définition d'une nouvelle exception, déclenchement et traitement de l'exception générée.

Gestion des logs Java.