



zenika
ARCHITECTURE INFORMATIQUE

www.zenika.com

Repartez avec le livre



Durée
4 jours

Répartition
50% théorie
50% pratique

Pré-requis
Notion de développement Objet

Public
Architecte
Développeur
Chef de projet

Tarif (déjeuners inclus)
1790 € (HT)
Tarif (1 mois avant)
1490 € (HT)

Lieu
Paris 9ème

Sessions 2009
13 au 16 janvier
24 au 27 février
17 au 20 mars
14 au 17 avril
4 au 7 mai
15 au 18 juin
28 au 31 juillet
1 au 4 septembre
20 au 23 octobre
16 au 19 novembre
8 au 11 décembre

Intra-entreprise sur demande

Inscription et renseignements
+33(0)1.45.26.19.15
training@zenika.com
www.zenika.com

Java

Développement Java avec Eclipse

Objectifs

- Découvrir et mettre en oeuvre la programmation Objet avec Java
- Pouvoir modéliser et manipuler des objets du monde réel
- Savoir utiliser l'IDE Eclipse pour le développement d'applications Java
- Comprendre la structure et l'architecture des applications modernes

Contenu

Présentation du langage

- Présentation et historique du langage
- L'écosystème Java : Java SE, Java EE

L'environnement de développement

- Présentation du JDK
- Présentation d'Eclipse (Integrated Development Environment)
- Perspectives, vues et éditeurs
- Editer, compiler et exécuter du code
- Documenter le code avec Javadoc
- Importer des bibliothèques externes
- Assistants, raccourcis clavier, trucs & astuces
- Refactoring : formatage, gestion des imports, génération de code

Déclarations et contrôle d'accès (java.lang)

- Norme JavaBean et validité des identifiants
- Déclarer une classe, organisation en packages
- Déclarer des membres d'une classe
 - Types de données : types primitifs, références, tableaux
 - Déclarer une propriété
 - Déclarer une méthode : paramètres et type de retour
 - Notion de constructeur, processus d'instanciation
- Classes abstraites et interfaces
- Contrôle d'accès
 - Modificateurs de visibilité : public, protected, package, private
 - Autres modificateurs : final, static

Programmation objet (java.lang)

- Introduction à la conception objet, notions de classe et d'instance
- Introduction à la notation UML : diagrammes de classes et de séquences
- Principes de forte encapsulation et de couplage faible
- Notion de composition (« HAS-A »)
- Notions d'héritage (« IS-A »), de polymorphisme, de cast
- Surcharge (« overloading ») et redéfinition (« overriding »)
- Notion d'identité d'objet : equals et hashCode

Opérateurs (java.lang)

- Affectation
- Egalité et comparaison
- Opérateurs booléens : !, &, &&, |, ||, ^
- Instanceof

Structure de contrôle et exceptions (java.lang)

- Tests : if, switch
- Boucles : for, foreach, while, do-while
- Exceptions :
 - Notions d'exceptions
 - Déclaration dans les signatures des méthodes
 - Capture et traitement avec « try/catch »
 - Notion de bloc « finally »
 - Hiérarchie des exceptions, « checked » vs « runtime »

Tableaux et Collections (java.util)

- Les types de collections : List, Set, Map
- Comprendre l'API des collections : introduction aux types paramétrés (« generics »)
- Choisir la bonne collection
- Parcourir une collection ou un tableau : notion d'itérateur
- Trier une collection ou un tableau : Comparable et Comparator
- Rechercher dans une collection ou un tableau

Formatage et Internationalisation (java.text)

- Notion de Locale
- Formater les dates et les nombres avec DateFormat et NumberFormat
- Formater les messages avec MessageFormat et ChoiceFormat
- Externaliser les messages avec ResourceBundle

I/O Fichiers (java.io)

- Notions de flux textuel (Reader/Writer) et binaire (Streams)
- Hiérarchie des classes de java.io et notion d'adapter
- Ecriture et lecture d'un fichier texte sur disque avec BufferedReader et PrintWriter
- Notion de sérialisation, variables transientes
- Ecriture et lecture d'un graphe d'objets sur disque avec ObjectOutputStream et ObjectInputStream

I/O Réseau (java.io)

- Notion d'adresse réseau et de port
- Notion de socket
- Connexion à un serveur en TCP, lecture et écriture du flux
- Développement d'un serveur
- Notion de protocole, présentation de protocoles existants

JDBC (java.sql)

- Ouverture d'une connexion
- Effectuer une requête en écriture
- Effectuer une requête en lecture, notion de ResultSet
- Utiliser PreparedStatement
- Concept d'ORM (Hibernate, JPA)

Architecture et intégration

- Introduction aux Design Patterns
- Introduction aux Architectures N-tiers
- Présentation des principaux frameworks d'entreprise
- Notion d'usine logicielle et intégration continue