

## DÉVELOPPERMENT JAVA AVEC HIBERNATE

<b>Durée</b>	<b>3 jours</b>	<b>Référence Formation</b>	<b>4-JA-HIB</b>
--------------	----------------	----------------------------	-----------------

### Objectifs

Appréhender tous les rouages essentiels d'Hibernate afin d'écrire une couche d'accès aux données performantes.

### Participants

Développeur.

### Pré-requis

Connaissance du langage Java et de l'API JDBC

### Moyens pédagogiques

Réflexion de groupe et apports théoriques du formateur

Travail d'échange avec les participants sous forme de réunion-discussion

Utilisation de cas concrets issus de l'expérience professionnelle

Validation des acquis par des questionnaires, des tests d'évaluation, des mises en situation et des jeux pédagogiques.

Remise d'un support de cours.

### PROGRAMME

#### Problématique du mapping objet relationnel

- Stratégies de mapping
- Associations
- Héritage
- Panorama des solutions aujourd'hui disponibles

#### Installation et configuration du framework Hibernate

- Installation
- Fichiers de configuration
- Configuration pour accès au SGBD MySQL
- Création, suppression, mise à jour, recherche d'objets Java
- Exemple d'application simple de persistance d'objets Java

#### Définition des mappings

- Création du fichier de mapping
- Classe simple
- Classe sérialisable
- Classe avec attribut Date ou Calendar
- Génération par Xdoclets

#### Gestion des associations et de l'héritage

- Différents types d'associations
- Gestion des collections
- Héritage: une table par hiérarchie de classes
- Héritage: une table par sous-classe
- Héritage: une table par sous-classe concrète

#### Le langage HQL (Hibernate Query Language)

- Requêtes simples

- Clause WHERE
- Gestion des collections
- Gestion du polymorphisme
- Gestion des sous-requêtes
- Requêtes avec jointures

#### **Hibernate dans le contexte JEE**

- Hibernate et applications web
- Hibernate et EJB (Enterprise Java beans)

#### **Utilisation avancée**

- Gestion des transactions
- Gestion du cache
- Lazy loading