

Mise en oeuvre de la persistance avec Hibernate

L'intérêt d'implémenter une couche de mapping objet-relationnel pour découpler un modèle métier « objet » d'une base de données relationnelle n'est probablement plus à démontrer : le code qui contient les requêtes SQL est isolé dans un endroit unique, permettant une adhérence réduite au schéma de la base ; le modèle métier peut respecter les principes de la programmation objet, utiliser l'héritage et la navigation.

Librairie plébiscitée, Hibernate est l'origine de nombre des concepts repris par le groupe d'experts de la JSR 220 (EJB 3) pour établir la spécification JPA. C'est aujourd'hui une des implémentations open source de référence de cette spécification. Outil riche et fiable, Hibernate offre un vaste éventail d'options de configuration permettant d'établir un mapping performant adapté aux besoins spécifiques d'une application."

Programme

Objectifs

- Définition de la persistance objet
- Classes, fichier de mapping, configuration, cycle de vie des objets
- Mappage des associations unidirectionnelle/bidirectionnelle et des collections

L'architecture d'Hibernate

- Vue en couche
- Cycle de vie des instances
- Intégration JNDI, JDBC, JTA
- Gestion des sessions

Configuration

- APIs permettant la configuration par programmation
- Ouverture des sessions
- Gestion des connexions JDBC
- Dialectes SQL, cache de second niveau et cache de requêtes,

- Fichier de configuration XML
- Intégration à un serveur d'application Java EE

Réalisation de classes persistantes

- Les Plain Ordinary Java Object (POJO)
- Mise en oeuvre de l'héritage
- Redéfinition des méthodes equals et hashCode

Les bases du Object Relational Mapping

- Les déclarations de mapping
- Les types utilisés par Hibernate
- Plusieurs mappings pour une même classe
- Les annotations JAVA 5

Etablir la persistance des collections

- Les types de collections persistantes
- Les clés étrangères, les éléments et les index d'une collection
- Associations *.* et 1..*

Etablir la persistance des associations

- Associations unidirectionnelles
- Associations bidirectionnelles
- Mappings complexes

Etablir la persistance des relations d'héritage

- Différentes stratégies
- Utilisation d'un discriminant
- Gestion du polymorphisme
- Comparaison des méthodes, avantages et inconvénients, dans quels cas les utiliser ?

Hibernate Query Language (HQL) et gestion des objets persistants

- Stocker et charger un objet
- Syntaxe du langage HQL, les clauses from, select, where, order by, group by
- Jointures, agrégation, requêtes polymorphiques
- Les requêtes imbriquées
- Les bonnes pratiques
- Externalisation de requêtes nommées
- Chargement tardif des propriétés
- Gestion du cache de second niveau
- Le cache de requêtes

Détails

Code :

JP-HIB1

Durée :

3 jours

Objectifs :

- Etre capable de développer une application Java EE (web, EJB) en réalisant la persistance des objets avec le framework Hibernate
- Connaître et assimiler un framework ORM (Object Relational Mapping) dans une architecture n-tiers
- Maîtriser les concepts d'Hibernate
- Savoir utiliser les bonnes pratiques de développement Hibernate
- Savoir utiliser les annotations, les outils Hibernate.

Public :

- Architectes techniques
- Concepteurs, développeurs
- Chefs de projets, équipe projet

Pré-requis :

Très Bonne pratique de Java

Tarif : 1290 € H.T. par personne (déjeuner inclus)