

TP 2 : Persistance avec Hibernate

Objectif du TP

Ce TP a pour objectif de mettre en pratique les concepts de persistance des objets avec Hibernate. Les étudiants apprendront à configurer Hibernate, à effectuer des opérations CRUD, à gérer les relations (1-1, 1-n, n-n), à appliquer les stratégies d'héritage et à comprendre la persistance via les fichiers de configuration XML.

Partie 1 : Configuration du projet Hibernate

1. Créez un projet Maven nommé `TP\_Hibernate`.
2. Ajoutez les dépendances suivantes dans le fichier `pom.xml` :
 - hibernate-core
 - mysql-connector-java ou ojdbc11
3. Configurez le fichier `hibernate.cfg.xml` avec les paramètres de connexion à la base de données.
4. Créez un package `model` pour les entités et un package `dao` pour la couche d'accès aux données.

Partie 2 : Opérations CRUD

1. Créez une entité `Etudiant` avec les attributs suivants :
 - id (clé primaire) – nom – prenom – email
2. Implémentez les opérations CRUD dans une classe `EtudiantDAO` :
 - createEtudiant(Etudiant e)
 - updateEtudiant(Etudiant e)
 - deleteEtudiant(int id)
 - getEtudiant(int id)
 - getAllEtudiants()
3. Testez ces méthodes dans une classe `Main`.

Partie 3 : Gestion des relations

1. Créez une entité `Departement` (id, nom).
2. Ajoutez une relation 1-n entre `Departement` et `Etudiant` (un département contient plusieurs

étudiants).

3. Créez une entité `Projet` (id, titre) et établissez une relation n-n entre `Etudiant` et `Projet`.
4. Testez les insertions et les requêtes pour vérifier les relations.

Partie 4 : Héritage

1. Créez une classe `Personne` (id, nom, prenom).
2. Créez deux sous-classes :
 - `Enseignant` (specialite)
 - `Etudiant` (email)
3. Implémentez deux stratégies d'héritage :
 - Stratégie SINGLE_TABLE via annotations JPA
 - Stratégie JOINED via fichiers XML
4. Testez les insertions et les requêtes pour chaque stratégie.

Partie 5 : Persistance avec fichier XML

1. Créez un fichier `Etudiant.hbm.xml` pour le mappage XML de la classe `Etudiant`.
2. Configurez Hibernate pour lire les entités à partir du fichier XML.
3. Vérifiez la cohérence des résultats entre l'approche annotations et XML.

Partie 6 : Optimisation et performance

1. Étudiez les stratégies de chargement :
 - Lazy loading vs Eager loading
2. Configurez un cache de premier et de second niveau :
 - Utilisez EHCACHE ou Caffeine
3. Analysez les performances lors de l'exécution de requêtes massives.

Partie 7 (Optionnelle) : Requêtes HQL et critères

1. Écrivez des requêtes HQL pour :
 - Sélectionner tous les étudiants d'un département donné
 - Rechercher un étudiant par nom
 - Supprimer les étudiants sans projet
2. Implémentez les mêmes requêtes avec l'API Criteria.