

JPA Java Persistence API

Mise en oeuvre de la persistance avec Java Persistence API (JPA)

L'intérêt d'implémenter une couche de mapping objet-relationnel entre un modèle métier « objet » et 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.

Issue des travaux du groupe d'experts de la JSR 220 (EJB 3), JPA est une spécification clé de Java EE. Loin d'être un framework de plus, JPA établit une norme standard simplifiée pour la persistance afin de garantir une meilleure portabilité des applications Java EE. JPA reprend naturellement les meilleures pratiques de frameworks établis tels Hibernate ou TopLink. EJB 3 s'appuie sur JPA pour la persistance des entités.

Cette formation permet de comprendre et d'expérimenter JPA par la pratique. Elle est un pré-requis indispensable au cours EJB3.

Détails

- Code : JE-JPA
- Durée : 3 jours (21 heures)

Public

- Chefs de projets
- Architectes
- Développeurs
- Ingénieurs
- Consultants

Pré-requis

- Bonne pratique des applications web avec Java (Servlet/JSP)
- Bonne pratique de Java

Objectifs

- Connaître les concepts de la norme JPA,
- Définir et utiliser un modèle persistant,
- Connaître les problématiques liées à la persistance d'objets sur bases relationnelles,
- Utiliser le langage de requêtes objet JPQL,
- Maîtriser les techniques et compromis permettant de garantir de bonnes performances

Programme

La problématique de persistance

- Le mapping objet-relationnel
- Intérêt de JPA, positionnement
- JPA et les serveurs d'applications
- Historique des solutions, Hibernate, TopLink, EJB
- Java EE : historique
- Les serveurs Java EE

Le modèle de persistance

- Définition d'une entité persistante, concept de POJO, les attributs, les relations
- Entité et Identité
- Cycle de vie des entités
- Mise en place d'un mapping
- Mapping sans annotations
- Mapping par annotations
- Déclaration des attributs
- Représentation des différents types de relation entre entités
 - Classe embarquée
 - Relation 0..1
 - Relation 1..N et N..1
 - Relation N..M

- Manipulations CRUD des entités
- Gestion de la bidirectionnalité (ou relation inverse)
- Gestion de l'héritage et du polymorphisme
- Mapper une entité sur plusieurs tables

Travaux Pratiques

- Mise en place de la persistance d'un modèle illustrant les différentes configurations présentées

Manipulation des Entités

- Le concept d'Entity Manager
- Gestion des Transactions, JTA (java Transaction API)
- Les requêtes : requêtes nommées et requêtes dynamiques, JPQL (Java Persistence Query Language)
- Définition d'une requête
- Exécution d'une requête
- Requetes paramétrées
- Syntaxe JPQL

Travaux Pratiques

- Utilisation des fonctionnalités de requêtage de JPA

Performances

- Utilisation de Cascade
- Optimisation du chargement et Lazy Loading
- Le verrouillage Optimiste

Conclusion

- Panorama des implémentations
- Futur des serveurs d'application Java et de la persistance

Modalité

- Stage pratique en présentiel
- Stage pratique en distanciel
- Nombre de stagiaires minimum : 4
- Nombre de stagiaires maximum : 10

Méthodes pédagogiques

- Exposés
- Cas pratiques
- Echanges d'expérience

Profils des intervenants

- Toutes nos formations sont animées par des consultants-formateurs expérimentés et reconnus par leurs pairs.

Modalités d'évaluation

- Evaluation des acquis de la formation par le biais de cas pratiques et/ou mises en situation.
- Attestation de formation remise à chaque participant.

Démarche qualité

- Questionnaire d'évaluation de satisfaction à chaud complété par chaque participant à l'issue de la formation.

Moyens pédagogiques

- Salle équipée de PC (1 poste par stagiaire), vidéo-projecteur.
- Espace de pause.

Dernière mise à jour le 15/09/2022