

Objectifs

Établir un mapping entre des objets java et des tables relationnelles. Créer, mettre à jour et supprimer des objets persistants. Maîtriser le langage de requêtes JPQL. Gérer des transactions

Participants

Développeurs, chefs de projet, architectes.

Prérequis

Bonnes connaissances du langage Java.

Pédagogie

La pédagogie est basée sur le principe de la dynamique de groupe avec alternance d'apports théoriques, de phases de réflexion collectives et individuelles, d'exercices, d'études de cas et de mises en situations observées. Formation / Action participative et interactive : les participants sont acteurs de leur formation notamment lors des mises en situation car ils s'appuient sur leurs connaissances, les expériences et mettront en oeuvre les nouveaux outils présentés au cours de la session.

Profil de l'intervenant

Consultant-formateur expert sur cette thématique. Suivi des compétences techniques et pédagogiques assurée par nos services.

Moyens techniques

Encadrement complet des stagiaires durant la formation. Espace d'accueil, configuration technique des salles et matériel pédagogique dédié pour les formations en centre. Remise d'une documentation pédagogique papier ou numérique à échéance de la formation.

Méthodes d'évaluation des acquis

Exercices individuels et collectifs durant la formation. Evaluation des acquis et attestation de fin de stage adressés avec la facture.

Programme

JPA - Architecture logicielle

Processus 2TUP

Modèle architecture logicielle en 5 couches

Modélisation de la couche métier et persistance : Domain Object, DAO et VO

Structuration avec les Patterns Method Factory, Abstract Factory

JPA - Architecture technique de JPA

Introduction à la problématique de persistance

Spécifications et implémentations

Fournisseur et contexte de persistance

Mise en place de l'environnement

Présentation des outils de la plate-forme Eclipse

Création d'un projet aux normes JPA

JPA -Fondamentaux du mapping d'objets persistants

Exemple d'application simple de persistance d'objets Java

Classe de persistance : l'annotation @Entity

Alternative : mapping xml

Opérations CRUD

Gestion des clés primaires

JPA - Mapping des relations unidirectionnelles

Associations unidirectionnelles

Associations unidirectionnelles avec tables de jointure

JPA - Mappings des relations bidirectionnelles

Associations bidirectionnelles sans table de jointure

Associations bidirectionnelles avec table de jointure

JPA - Mapping des Objets du domaine

Mapping des clés composites

Composants embarqués

JPA - Mapping d'une relation d'héritage

Une table par hiérarchie de classes

Une table par classe concrète

Une table par classe fille

JPA - Mapping des Collections et types objets complexes

Principe de fonctionnement des listes

Utilisation des collections d'éléments les différents types de collections

Ordonnancement d'une collection

Utilisation des maps

JPA - Effectuer ses requêtes avec JPQL

Le langage JPQL

Utilisation des objets Query

Utilisation des clauses et fonctions Chargement avec FETCH

Les requêtes polymorphiques

Les requêtes nommées

JPA - Gestion des transactions

Transactions locales

Transactions gérées par le conteneur avec JTA

Environnement managé et annotations

JPA - Cycle de vie des objets

Le cycle de vie d'un objet

Méthodes de passage d'un état à l'autre

JPA - L'API Criteria

Créer une requête par critère

Spécifier un select

Restriction du résultat

Trier les résultats

Associations

JPA - Utilisation avancée de JPA

Méthode de callback et listeners

Validation

Concurrence d'accès

Verrous

Gestion du cache

JPA - Nouveautés de JPA 2.0

Annotations

Verrous

Gestion du cache

Gestion des Maps

Identifiant dérivé

Objets embarqués

Mapping de Collections

OneToMany unidirectionnel

Mapping des listes ordonnées

Suppression des orphelins

Généralités sur les verrous

Verrou pessimiste

Utilisation de l'API de cache

Propriétés standards

API Criteria

JPA - Nouveautés de JPA 2.1

Convertisseurs

API Criteria : DELETE et UPDATE Procédures stockées

ConstructorResult

JPA - Spécificités des implémentations de JPA

Spécificités des implémentations de JPA : Hibernate, EclipseLink, OpenJPA