

Maîtriser la persistance des données avec JPA et Hibernate

Programme (Mis à jour le 31/01/2026)

Introduction aux ORM, JPA et Hibernate (1h)

- Rappels sur JDBC et ses limites
- Problématiques de l'impédance objet/relationnel
- Principes fondamentaux des ORM : rôle, avantages et fonctionnement
- Présentation de JPA : concepts clés, annotations principales, cycle de vie des entités
- Hibernate comme implémentation de référence de JPA
- Panorama des autres implémentations JPA (EclipseLink, OpenJPA)

Démarrage avec Hibernate et JPA (2h)

- Mise en place de l'environnement de travail
- Déploiement et connexion à une base de données relationnelle (MySQL ou équivalent)
- Configuration d'un projet JPA
- Présentation du persistence.xml
- Introduction aux mappings via annotations JPA
- Comparaison mappings XML (Hibernate natif) et annotations JPA
- Travaux pratiques : installation de l'environnement JPA / Hibernate avec Maven

Configuration et exploitation du moteur JPA (2h)

- Configuration fine du moteur JPA/Hibernate
- Gestion des unités de persistance
- Intégration et configuration des logs (SQL généré, paramètres, transactions)
- Analyse du comportement runtime d'Hibernate
- Travaux pratiques : configuration des APIs de logs et analyse des requêtes SQL générées par l'ORM

Le mapping JPA : les fondamentaux (8h)

- Introduction au mapping d'associations
- Gestion des associations @OneToOne
- Gestion des associations @ManyToOne
- Gestion des associations @OneToMany
- Gestion des associations @ManyToMany
- Gestion des collections
- Propagation des actions sur des grappes d'objets
- Travaux pratiques : divers exercices successifs permettant de mettre en œuvre un modèle JPA complet avec les différents types de relations proposées.

Requêtes avec JPA (4h)

- Introduction à JPQL
- Comparaison JPQL / HQL
- Requêtes JPQL simples et paramétrées
- Jointures et navigation dans les associations
- Fonctions d'agrégation et sous-requêtes
- Travaux pratiques : écriture de diverse requêtes JPQL
- Introduction à l'API Criteria
- Travaux pratiques : écriture de diverse requêtes Criteria
- Les requêtes natives et JPA

Référence

THIL3651

Durée

3 jours / 21 heures

Prix HT / stagiaire

1875€

Objectifs pédagogiques

- Comprendre les principes fondamentaux de l'ORM et les différences entre modèles objets et relationnels
- Maîtriser les concepts clés de l'API JPA et son cycle de vie des entités
- Configurer et utiliser Hibernate comme implémentation JPA
- Mettre en œuvre les mappings JPA pour gérer les relations entre entités
- Interagir avec la base de données via JPQL et l'API Criteria
- Intégrer JPA/Hibernate dans des applications Java concrètes en respectant les bonnes pratiques

Niveau requis

- Maîtriser le langage de programmation Java

Public concerné

- Concepteurs / Développeurs Java

Formateur

Les formateurs intervenants pour Themanis sont qualifiés par notre Responsable Technique Olivier Astre pour les formations informatiques et bureautiques et par Didier Payen pour les formations management.

Conditions d'accès à la formation

Délai : 3 mois à 1 semaine avant le démarrage de la formation dans la limite des effectifs indiqués

Moyens pédagogiques et techniques

Salles de formation (les personnes en situation de handicap peuvent avoir des besoins spécifiques pour suivre la formation. N'hésitez pas à nous contacter pour en discuter) équipée d'un ordinateur de dernière génération par stagiaire, réseau haut débit et vidéo-projection UHD

Documents supports de formation projetés
Apports théoriques, étude de cas concrets et exercices

Mise à disposition en ligne de documents supports à la suite de la formation

Dispositif de suivi de l'exécution de l'évaluation des résultats de la formation

Feuilles d'émargement (signature électronique privilégiée)

Evaluations formatives et des acquis sous forme de questions orales et/ou écrites (QCM)

- Travaux pratiques : mise en œuvre de requêtes natives

API JPA et transactions (4h)

- EntityManager et EntityManagerFactory
- Cycle de vie des entités persistantes
- Opérations CRUD
- Transactions JPA
- Bonnes pratiques d'intégration dans une application Java
- Travaux pratique : mise en œuvre d'un pattern efficient pour la gestion des transactions