

# Formation Java Persistence API (JPA)

## Objectifs de la formation Java Persistence API

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. Plus concrètement après avoir participé à ce cours, vous serez en mesure de :

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

Cette formation permet de comprendre et d'expérimenter JPA par la pratique.

## À qui s'adresse cette formation ?

**Public :**

Ce cours est prévu notamment pour les architectes, chefs de projets, consultants, développeurs, ingénieurs.

### **Prérequis :**

Pour suivre ce cours dans les meilleures conditions possible, il vous faut connaître les bonnes pratiques des applications web avec Java (Servlet/JSP) ainsi que les bonnes pratiques de Java.

## **Contenu du cours Java Persistence API**

La problématique de persistance et les solutions apportées par JPA

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

Requêtes 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