

- Les outils de développement.
- L'environnement Eclipse.
- Dépendances Maven.
- Les "deployment assembly".
- Les tests unitaires (avec JUnit et JeeUnit).
- Le fichier domain.
- xml.
- JNDI, la standardisation proposée par Java EE 6.
- Le moniteur transactionnel.
- Le serveur Web.
- Le fonctionnement modulaire : OSGI.
- Travaux pratiques Installer l'IDE et le serveur.

L'accès aux données avec JPA

- Les principes et le fonctionnement du mapping.
- La manipulation des instances persistées.
- Le Lazy loading.
- Le metamodel.
- Les interfaces entityManager, Query et Criteria.
- La gestion de l'entityManager par le conteneur.
- Travaux pratiques Mise en place d'entités persistantes dans un projet Java.
- Mise en place de la couche d'accès aux données avec JPA au sein de l'application fil rouge.

Les EJB Session

- Déclaration d'une classe en tant qu'EJB Session.
- La gestion du cycle de vie (
 - Stateless.
 -).
- Les apports de Java EE 6.
- L'injection de dépendances.
- Injection hors CDI.
- Les interceptions.
- Transactions.
- La sécurité.
- Les timers.
- Le remoting.
- Travaux pratiques Mise en place de services avec les EJB Sessions.

Les Messages Driven Beans (MDB)

- Le dialogue asynchrone entre les applications.
- L'API JM.
- Les concepts.
- Les éléments JMS : connectionFactory, connection, session, destination, message, messageProducer, messageConsumer.
- La configuration sous Glassfish.
- Les fonctionnalités avancées.
- Travaux pratiques Mise en place d'un système de notification d'applications clientes.

Le Web et Java Server Faces

- JSF 2.
- 0 et la configuration par annotations.
- Les concepts (MVP, Presenter, View).
- Les facettes, taglibs, jsf-el.
- La validation JSR303.
- Mise en oeuvre d'Ajax.
- Travaux pratiques Mise en oeuvre de JSF et d'Ajax au sein d'une application Web.

Le Web et Java Server Faces

- Les possibilités offertes.
- Les API JAX-WS et JAX-RS.
- Utilisation d'un Custom Serializer dans les deux cas.
- Travaux pratiques Développement d'une façade Web exposant les services de l'exercice développé.