

## Services REST en Java avec Jax-RS

Mettre en œuvre une architecture de services légère REST avec Jax-RS Jersey

REST (Representational State Transfer) est un style d'architecture SOA simplifiée afin de mettre en œuvre et de consommer des services web en bénéficiant de tous les avantages d'Internet : scalabilité, caching, etc.

Cette formation vous permettra de comprendre REST et de développer en Java, via la spécification Java EE 6 JAX-RS (JSR311), des solutions dites « RESTful ».

### Détails

- **Code** : AE-JRES
- **Durée** : 3 jours ( 21 heures )

#### Public

- Architectes
- Chefs de projets
- Développeurs
- Ingénieurs

#### Pré-requis

- Connaissance des principes de SOA
- Bonne pratique de Java

#### Objectifs

- Comprendre SOA à travers une implémentation simplifiée
- Être en mesure de développer et consommer des services web « RESTful » avec Java

### Programme

#### Introduction

- Les services REST
- Présentation de REST
- Un style architectural
- Les contraintes d'un système RESTful
- Modèle de maturité de Richardson
- Comparaison SOAP/REST

#### Rappels HTTP

- Les URI
- Structure d'une requête
- Méthode HTTP
- En-tête HTTP
- Structure d'une réponse
- Statut de réponse HTTP
- Requête sûre
- Requête idempotente
- Stateless vs Stateful
- Sécurisation HTTP

#### JAX-RS

- Les implémentations
- Notions de base : ressource
- Mise en place d'un service
- Déploiement d'un service

#### Jersey

- Déploiement dans différents environnements
- Création d'un projet

#### Les annotations JAX-RS

- Les annotations usuelles

- Les annotations pour extraire des infos
- Le modèle d'injection
- Conversion des types
- Validation

#### Gestion de contenu

- Conneg
- @Produces et @Consumes
- Le quality factor q
- Paramètre d'entrée
- Type de retour
- Entity provider
- ResponseBuilder
- Uri et UriBuilder
- Link et Link headers
- Gestion des exceptions

#### La représentation des ressources

- Le format des données

#### Support de XML

- XML
- JAX-B
- JAX-B et REST

#### JavaScript Object Notation

- Présentation de JSON

#### Utilisation avancée de JAX-RS

- Filtres
- Intercepteurs
- Dynamic binding

- Name Binding
- L'ordre des filtres et intercepteurs
- API Asynchrone

## Intégration avec les EJB

### Client Java RESTful

- Consommation de services REST
- Jersey Client
- L'API Client de JAX-RS
- Fonctionnalités clients avancées

### Sécurité avec JAX-RS

- Sécurité des services avec Java EE

- Sécurité des services avec JWT
- CORS

### La documentation d'une API

- WADL, RAML, Blue Print
- Swagger

### Tests des services REST

- Avec cURL
- Avec PostMan
- Avec RestAssured
- Avec SOAP-UI

### Conclusion

## Modalités

- **Type d'action** :Acquisition des connaissances
- **Moyens de la formation** :Formation présentielle – 1 poste par stagiaire – 1 vidéo projecteur – Support de cours fourni à chaque stagiaire
- **Modalités pédagogiques** :Exposés – Cas pratiques – Synthèse
- **Validation** :Exercices de validation – Attestation de stages