

Atelier REST n°3

JAX_RS CRUD

Objectifs

Notre objectif est de créer un service web RESTFul qui fournira un exemple d'opérations CRUD (Créer, lire, mettre à jour et supprimer).

Développement du service web RESTFul

Nous visons à créer un service web de gestion des employés dans une entreprise.

Employe
-cin:String -nom:String -prenom:String
ajouterEmploye(Employe employe) modifierEmploye(Employe employe) chercherEmploye(String cin) afficherListeEmployes() supprimerEmploye(String cin)

1. Créez un projet **maven** de packaging **war**.
2. Créez l'entité "**Employe**" comme suit:

```

import javax.xml.bind.annotation.XmlRootElement;

@XmlRootElement
public class Employe {

    private int cin;
    private String nom;
    private String prenom;

    public Employe() {
        super();
        // TODO Auto-generated constructor stub
    }

    public Employe(int cin, String nom, String prenom) {
        super();
        this.cin = cin;
        this.nom = nom;
        this.prenom = prenom;
    }

    public int getCin() {
        return cin;
    }
    public void setCin(int cin) {
        this.cin = cin;
    }

    public String getNom() {

```

- au niveau de l'entité "**Employe**", générez les deux méthodes equals et hashCode.

*equals () est utilisée dans la plupart des collections pour déterminer si une collection contient un élément donné.

* hashCode () est utilisée pour déterminer où stocker et où chercher l'objet en interne.

3. Créez les ressources REST qui constituent le service web selon les descriptions suivantes:

a) **displayEmployeesList**

Requête GET /employees
Réponse <pre>[{ "cin": 123, "nom": "Foulen", "prenom": "Ben Foulen" }]</pre> Status 200OK

b) addEmployee

Requête POST /employees Content-Type: application/json <pre>{ "cin": 124, "nom": "Ben Ahmed", "prenom": "Skander" }</pre>
Réponse add successful Status 200OK

- testez la méthode avec un outil de test des requêtes HTTP(Advanced REST client, poster, postman...)

c) updateEmployee

Requête PUT /employees Content-Type: application/xml <pre><employee> <cin> 123 </cin> <prenom>ahmed</prenom> <nom>ben salah</nom> </employee></pre>
Réponse update successful Status 200OK

d) searchEmployee

Requête GET /employees/123
Réponse <pre><employee> <cin> 123 </cin> <nom> ahmed </nom> <prenom> salah </prenom> </employee></pre> Status 200OK

e) deleteEmployee

Requête DELETE /employees/123
Réponse true Status 200OK

4. Renvoyez la requête de suppression:

Requête DELETE /employees/123
Réponse false Status 200OK

- a. Que remarquez-vous?
- b. Comment résoudre ce problème?