

Fil rouge : corrigé séance 2

Module FIP INF 211

Gestion de la persistance des données (JPA)

Génération des classes entités.

Le nom des classes générées et le nom des attributs dépendent du nom des tables et des colonnes.

À titre d'information, le code de la classe *Entreprise* un peu arrangé pour en faciliter la lecture :

```
package eu.telecom_bretagne.cabinet_recrutement.data.model;

import java.io.Serializable;
import javax.persistence.*;
import java.util.Set;

/**
 * The persistent class for the entreprise database table.
 */
@Entity
public class Entreprise implements Serializable
{
    //-----
    private static final long serialVersionUID = 1L;

    @Id
    @SequenceGenerator(name = "ENTREPRISE_ID_GENERATOR", sequenceName = "ENTREPRISE_ID_SEQ", allocationSize = 1)
    @GeneratedValue(strategy = GenerationType.SEQUENCE, generator = "ENTREPRISE_ID_GENERATOR")
    private Integer id;

    private String nom;

    private String descriptif;

    @Column(name = "adresse_postale")
    private String adressePostale;

    // bi-directional many-to-one association to OffreEmploi
    @OneToMany(mappedBy = "entreprise")
    private Set<OffreEmploi> offresEmploi;
    //-----
    public Entreprise() {}
    //-----
    public Integer getId() { return this.id; }
    public String getNom() { return this.nom; }
    public String getDescriptif() { return this.descriptif; }
    public String getAdressePostale() { return this.adressePostale; }
    public Set<OffreEmploi> getOffresEmploi() { return this.offresEmploi; }
    //-----
    public void setId(Integer id) { this.id = id; }
    public void setNom(String nom) { this.nom = nom; }
    public void setDescriptif(String descriptif) { this.descriptif = descriptif; }
    public void setAdressePostale(String adressePostale) { this.adressePostale = adressePostale; }
    public void setOffresEmploi(Set<OffreEmploi> offresEmploi) { this.offresEmploi = offresEmploi; }
    //-----
}
```

Création des DAO

Implémentation complète du DAO de la classe entité Entreprise : EntrepriseDAO

```
package eu.telecom_bretagne.cabinet_recrutement.data.dao;

import java.util.List;

import javax.ejb.LocalBean;
import javax.ejb.Stateless;
import javax.persistence.EntityManager;
import javax.persistence.PersistenceContext;
import javax.persistence.Query;

import eu.telecom_bretagne.cabinet_recrutement.data.model.Entreprise;

/**
 * Session Bean implementation class EntrepriseDAO
 * @author Philippe TANGUY
 */
@Stateless
@LocalBean
public class EntrepriseDAO
{
    //-----
    @PersistenceContext
    EntityManager entityManager;
    //-----
    /**
     * Default constructor.
     */
    public EntrepriseDAO()
    {
        // TODO Auto-generated constructor stub
    }
    //-----
    public Entreprise persist(Entreprise entreprise)
    {
        entityManager.persist(entreprise);
        return entreprise;
    }
    //-----
    public Entreprise findById(Integer id)
    {
        return entityManager.find(Entreprise.class, id);
    }
    //-----
    public List<Entreprise> findAll()
    {
        Query query = entityManager.createQuery("select entreprise from Entreprise entreprise order by entreprise.id");
        List l = query.getResultList();

        return (List<Entreprise>)l;
    }
    //-----
    public Entreprise update(Entreprise entreprise)
    {
        entityManager.merge(entreprise);
        return findById(entreprise.getId());
    }
    //-----
    public void remove(Entreprise entreprise)
    {
        if(!entityManager.contains(entreprise)) // Si l'entité n'est pas dans un état "géré" (managed),
        {                                         // il est impossible de la supprimer directement, erreur "Entity must
                                                // be managed to call remove".
            entreprise = entityManager.merge(entreprise); // Il faut la "rattacher" au contexte de persistance par l'appel
        }                                                 // de la méthode merge de l'EntityManager.

        // L'entité était déjà attachée ou a été rattachée, on peut donc la supprimer...
        entityManager.remove(entreprise);
    }
    //-----
}
```

À noter, les requêtes JPQL sont ici des requêtes paramétrées.

CandidatureDAO

```
public List<Candidature> findBySecteurActiviteAndNiveauQualification(int idSecteurActivite, int idNiveauQualification)
{
    Query query = entityManager.createQuery("select c from Candidature c join c.secteursActivite secteur " +
                                           "where secteur.id = :idSA and c.niveauQualification.id = :idNQ " +
                                           "order by c.id desc");

    query.setParameter("idSA", idSecteurActivite);
    query.setParameter("idNQ", idNiveauQualification);
    List<Candidature> l = query.getResultList();
    return l;
}

public List<Candidature> findALL()
{
    Query query = entityManager.createQuery("select candidature from Candidature candidature " +
                                           "order by candidature.id desc");

    List l = query.getResultList();
    return (List<Candidature>)l;
}
```

OffreEmploiDAO

```
public List<OffreEmploi> findByEntreprise(int idEntreprise)
{
    Query query = entityManager.createQuery("select offreEmploi from OffreEmploi offreEmploi " +
                                           "where offreEmploi.entreprise.id = :idE " +
                                           "order by offreEmploi.id desc");

    query.setParameter("idE", idEntreprise);
    List<OffreEmploi> l = query.getResultList();
    return l;
}

public List<OffreEmploi> findBySecteurActiviteAndNiveauQualification(int idSecteurActivite, int idNiveauQualification)
{
    Query query = entityManager.createQuery("select oe from OffreEmploi oe join oe.secteursActivite secteurs " +
                                           "where secteurs.id = :idSA and oe.niveauQualification.id = :idNQ " +
                                           "order by oe.id desc");

    query.setParameter("idSA", idSecteurActivite);
    query.setParameter("idNQ", idNiveauQualification);
    List<OffreEmploi> l = query.getResultList();
    return l;
}

public List<OffreEmploi> findALL()
{
    Query query = entityManager.createQuery("select offreEmploi from OffreEmploi offreEmploi " +
                                           "order by offreEmploi.id desc");

    List l = query.getResultList();
    return (List<OffreEmploi>)l;
}
```

NiveauQualificationDAO

```
public List<NiveauQualification> findALL()
{
    Query query = entityManager.createQuery("select niveauQualification from NiveauQualification niveauQualification " +
                                           "order by niveauQualification.id");

    List l = query.getResultList();
    return (List<NiveauQualification>)l;
}
```

SecteurActiviteDAO

```
public List<SecteurActivite> findAll()
{
    Query query = entityManager.createQuery("select secteurActivite from SecteurActivite secteurActivite " +
                                           "order by secteurActivite.id");

    List l = query.getResultList();
    return (List<SecteurActivite>)l;
}
```