

Classe `Personne` ayant un attribut `Adresse`.

Déroulement

- Créer le projet Gradle (voir fiche Démonstration 1)
- Création du fichier contenant la configuration de la base de données

```
/src/main/resources/META-INF/persistence.xml
```

```
<?xml version="1.0" encoding="UTF-8"?>
<persistence version="2.0"
  xmlns="http://java.sun.com/xml/ns/persistence" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/persistence http://java.sun.com/xml/ns/persistence/persistence_2_0.xsd">
  <persistence-unit name="MySQL_UIP" transaction-type="RESOURCE_LOCAL">
    <provider>org.hibernate.jpa.HibernatePersistenceProvider</provider>
    <class>fr.eni.jpa.entity.Adresse</class>
    <class>fr.eni.jpa.entity.Personne</class>
    <properties>
      <property name="javax.persistence.jdbc.driver" value="com.mysql.jdbc.Driver" />
      <property name="javax.persistence.jdbc.url" value="jdbc:mysql://localhost:3306/javaavance" />
      <property name="javax.persistence.jdbc.user" value="java" />
      <property name="javax.persistence.jdbc.password" value="avance" />
      <property name="hibernate.show_sql" value="false" />
      <property name="hibernate.hbm2ddl.auto" value="create" />
    </properties>
  </persistence-unit>
</persistence>
```

- Création de la classe `Adresse`

fr.eni.jpa.entity.Adresse

```
@Entity
@Table(name = "Adresse0T0")

public class Adresse {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int id;

    private String codePostal;
    private String ville;

    public Adresse() {
    }
}
```

- Création de la classe `Personne`

fr.eni.jpa.entity.Personne

```
@Entity
@Table(name = "Personne0T0")
public class Personne {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int id;

    private String nom;
    private String prenom;

    @OneToOne(cascade=CascadeType.ALL, fetch=FetchType.EAGER)
    // @Basic(fetch=FetchType.EAGER)
    private Adresse adresse;

    public Personne() {
    }
}
```

- Création de la classe `DAOUtil`

Cette classe permet de gérer les `EntityManager` utilisés dans les classes DAO.

Dans cet exemple, nous utilisons un unique `EntityManager` (déclaré `static`) car nous sommes dans une application client lourd dans laquelle il n'y aura pas d'accès multiples.

fr.eni.jpa.dao.DAOUtil

```
public class DAOUtil {
    private static EntityManagerFactory emf;
    private static EntityManager em;

    static{
        emf = Persistence.createEntityManagerFactory("Mysql_UP");
        em = emf.createEntityManager();
    }

    public static EntityManager getEntityManager(){
        return em;
    }

    public static void close(){
        if (em != null )
            em.close();
        if (emf != null )
            emf.close();
    }
}
```

- Création de la classe `PersonneDAO`

fr.eni.jpa.dao.PersonneDAO

// TODO

- Création de la classe de test

fr.eni.jpa.main.TestPersonne

// TODO :

- add personne x2 (avec une adresse)
- update codePostal
- findAll personnes
- findAll adresse des personnes
- delete personne

- Exécution

```
liste des personnes (nom) : L
brun Leu
iond Lebl
des personnes (adresse) : Liste
[id=1, codePostal=44100, ville=Nantes] Adresse
[id=2, codePostal=33000, ville=Bordeaux] Adresse L
n de p1 : suppressio
```