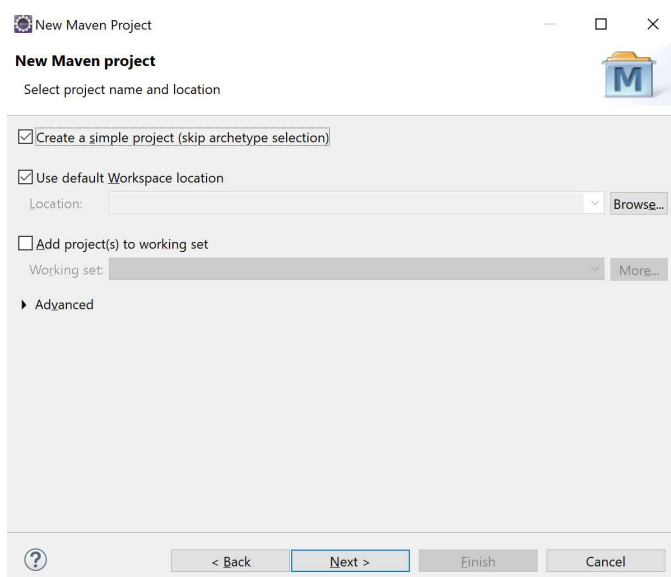


Outils et bibliothèques utilisés

- Mysql version
- Eclipse Version : 2019-12 (4.14.0)
- JDK : 1.12
- Hibernate : 5.4.1

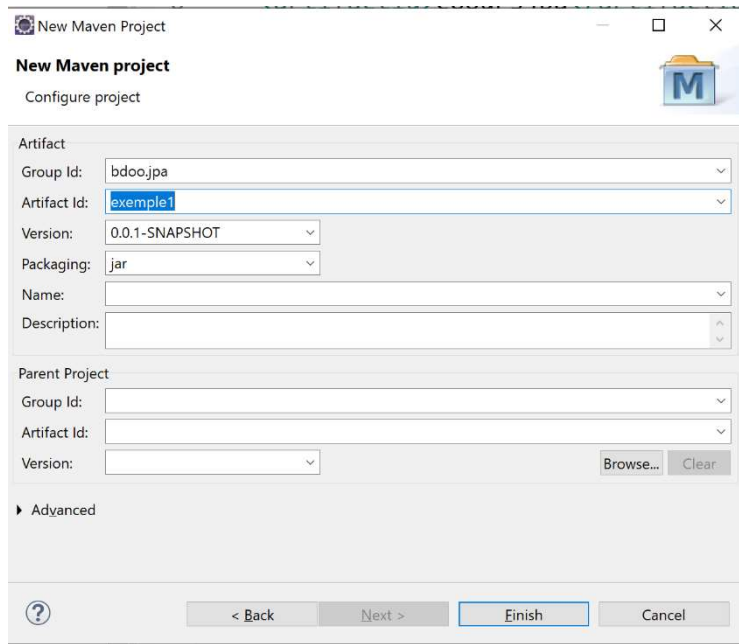
Création du projet

1. Créer un nouveau projet de type « Maven project »
2. Cocher la case « Créer a sample project »



Ce modèle créera un projet avec un fichier pom.xml vide sans aucune dépendance.

3. Renseigner le « group-id » et l' « artifact-id » de votre projet



le contenu du fichier pom.xml après la création du projet est le suivant :

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>bdoo.jpa</groupId>
  <artifactId>exemple1</artifactId>
  <version>0.0.1-SNAPSHOT</version>
</project>
```

Un fichier pom (Project Object Model) contient des informations sur le projet qui seront utilisées par Maven lors du processus de construction du projet

Configuration Maven

Nous allons ajouter des dépendances dans le fichier pom.xml pour que Maven télécharge bibliothèques nécessaires à notre projet

1. Avant la balise fermante </project> ajouter l'élément <dependencies></dependencies>
2. Dans l'élément « dependencies » ajouter les dépendances suivantes :

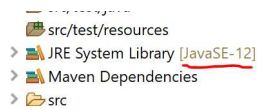
Bibliothèque	Group-id	Artifact-id	Version
Mysql	mysql	mysql-connector-java	8.0.19
HIbernate	org.hibernate	hibernate-entitymanager	5.4.12.Final

3. Dans le cas où le projet utilise une version de java inférieure à la version 8, juste avant l'élément <dependencies> ajoutez les propriétés suivantes :

```
<properties>
  <maven.compiler.target>1.X</maven.compiler.target>
  <maven.compiler.source>1.X</maven.compiler.source>
</properties>
```

Où X désigne la version du JDK installée sur votre machine est qui doit être >= 8

4. Enregistrez le fichier pom.xml
5. Cliquez avec le bouton droit de la souris sur le projet puis sélectionner la commande Maven/ Update project



Fichier pom.xml final

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>tests</groupId>
  <artifactId>test4</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <name>test4</name>

  <properties>
    <maven.compiler.target>1.12</maven.compiler.target>
    <maven.compiler.source>1.12</maven.compiler.source>
  </properties>
  <dependencies>
    <dependency>
      <groupId>org.hibernate</groupId>
      <artifactId>hibernate-entitymanager</artifactId>
      <version>5.4.12.Final</version>
    </dependency>
    <dependency>
      <groupId>mysql</groupId>
      <artifactId>mysql-connector-java</artifactId>
      <version>8.0.19</version>
    </dependency>
  </dependencies>
</project>
```

Configuration de la source de données

1. Ajoutez un dossier nommé META-INF dans src/main/resources
2. Dans ce dossier ajoutez un fichier nommé persistence.xml

Contenu du fichier persistence.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<persistence version="2.1" xmlns="http://xmlns.jcp.org/xml/ns/persistence"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/persistence
http://xmlns.jcp.org/xml/ns/persistence/persistence_2_1.xsd">
  <persistence-unit name="pu">
    <provider>org.hibernate.ejb.HibernatePersistence</provider>
```

```

        <properties>
            <property name="javax.persistence.jdbc.driver"
value="com.mysql.cj.jdbc.Driver"/>
            <property name="javax.persistence.jdbc.url"
value="jdbc:mysql://localhost/banque?serverTimezone=UTC"/>
            <property name="javax.persistence.jdbc.user" value="root"/>
            <property name="javax.persistence.schema-generation.database.action"
value="create"/>
        </properties>

    </persistence-unit>
</persistence>

```

Le nom de l'unité de persistance est « pu », il faut aussi mettre à jour les informations de connexion à la base de données : nom de la base de données, utilisateur et éventuellement le mot de passe.

la classe Entité Client

```

package test1.model;

import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.Id;

@Entity
public class Client {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private long id;

    public long getId() {
        return id;
    }

    public void setId(long id) {
        this.id = id;
    }

    private String nom;

    public String getNom() {
        return nom;
    }

    public void setNom(String nom) {
        this.nom = nom;
    }
}

```

Le programme principal

```

package test1;

import javax.persistence.EntityManager;
import javax.persistence.EntityManagerFactory;
import javax.persistence.EntityTransaction;

```

```
import javax.persistence.Persistence;

import test1.model.Client;

public class Main {

    public static void main(String[] args) {

        EntityManagerFactory emf =
Persistence.createEntityManagerFactory("pu");

        EntityManager em = emf.createEntityManager();
        Client cl = new Client();
        EntityTransaction tx = em.getTransaction();
        tx.begin();
        cl.setNom("client 1");
        em.persist(cl);
        cl = em.find(Client.class, 1L);
        System.out.println("Nom: " + cl.getNom());
        em.remove(cl);
        em.persist(cl);
        tx.commit();
        em.close();

    }

}
```