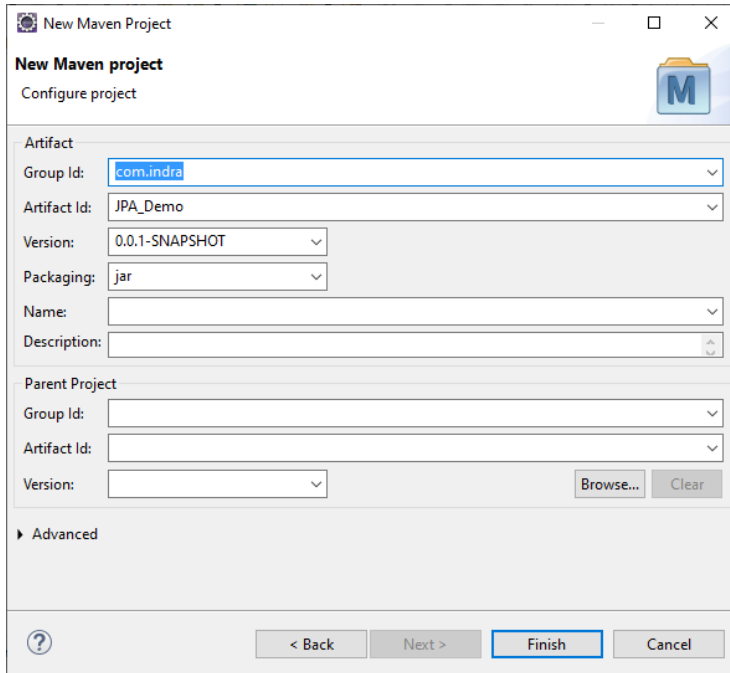


## GUIA I – JPA

### 1) Crear new Project MAVEN



New Maven Project

Configure project

Artifact

Group Id: com.indra

Artifact Id: JPA\_Demo

Version: 0.0.1-SNAPSHOT

Packaging: jar

Name:

Description:

Parent Project

Group Id:

Artifact Id:

Version:

Browse... Clear

Advanced

< Back Next > Finish Cancel

### 2) Configurar las dependencias en el pom.xml

```
<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>com.empresa</groupId>
  <artifactId>JPA_Demo</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <dependencies>
    <dependency>
      <groupId>org.hibernate.orm</groupId>
      <artifactId>hibernate-core</artifactId>
      <version>6.0.0.Alpha8</version>
    </dependency>
    <dependency>
      <groupId>mysql</groupId>
      <artifactId>mysql-connector-java</artifactId>
      <version>8.0.29</version>
    </dependency>
  </dependencies>
```

</project>

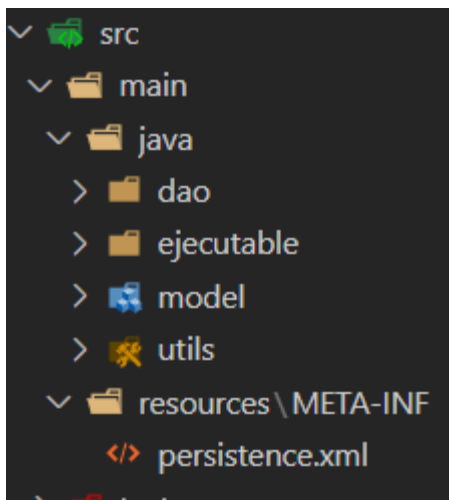
### 3) En MySQL crear la table “personas”

```
CREATE TABLE `indra2022`.`personas` ( `id` INT NOT NULL, `nombre` VARCHAR(45) NULL,
`apellido` VARCHAR(45) NULL, `fecha_nacimiento` DATETIME NULL, `cedula` VARCHAR(45) NULL,
PRIMARY KEY (`id`));
```

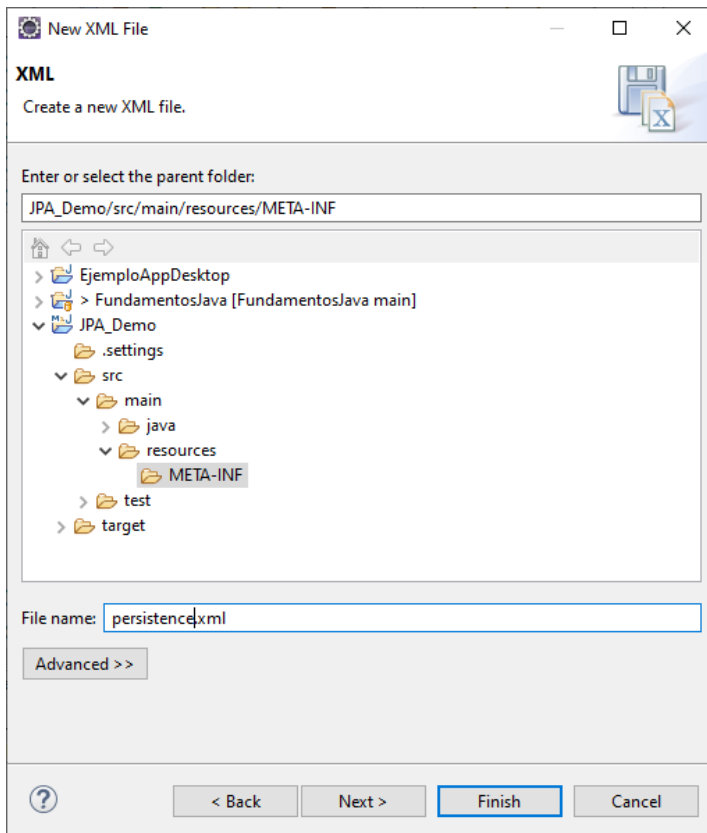
#### Y luego insertar datos

```
INSERT INTO `indra2022`.`personas` (`id`, `nombre`, `apellido`, `fecha_nacimiento`, `cedula`)
VALUES ('1', 'Alejandra', 'Buquete', '1990-11-22 00:00:00', '1234');INSERT INTO
`indra2022`.`personas` (`id`, `nombre`, `apellido`, `fecha_nacimiento`, `cedula`) VALUES ('2',
'Daniel', 'Ridolfo', '1989-10-07 00:00:00', '5678');INSERT INTO `indra2022`.`personas` (`id`,
`nombre`, `apellido`, `fecha_nacimiento`, `cedula`) VALUES ('3', 'Natalia', 'Perez', '1987-03-12
00:00:00', '9012');INSERT INTO `indra2022`.`personas` (`id`, `nombre`, `apellido`,
`fecha_nacimiento`, `cedula`) VALUES ('4', 'Sebastian', 'Cabinas', '1992-09-12 00:00:00', '3478');
```

### 4) En Eclipse crear la arquitectura de packages:



### 5) Crear el archivo persistence.xml



```
<?xml version="1.0" encoding="UTF-8"?>
<persistence version="1.0"
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_1_0.xsd">
<persistence-unit name="JPA_Demo" transaction-type="RESOURCE_LOCAL">
<provider>org.hibernate.jpa.HibernatePersistenceProvider</provider>
<class>model.Persona</class>
<properties>
<property name="hibernate.connection.username" value="root"/>
<property name="hibernate.connection.password" value="1234"/>
<property name="hibernate.connection.driver_class"
value="com.mysql.jdbc.Driver"/>
<property name="hibernate.connection.url"
value="jdbc:mysql://localhost:3306/indra2022"/>
<property name="hibernate.cache.provider_class"
value="org.hibernate.cache.NoCacheProvider"/>
</properties>
</persistence-unit>
</persistence>
```

6) Crear la clase Persona en el package model:

7) `package model;`

```
8)
9) import java.io.Serializable;
10) import java.util.Date;
11)
12) import javax.persistence.*;
13)
14) @Entity
15) @Table(name="personas")
16) public class Persona implements Serializable{
17)     private static final long serialVersionUID=-1L;
18)     @Id
19)     @Column(name="id")
20)     @GeneratedValue(strategy=GenerationType.AUTO)
21)     private Long id=null;
22)
23)     @Column(name="nombre")
24)     private String nombre;
25)
26)     @Column(name="apellido")
27)     private String apellido;
28)
29)     @Column(name="fecha_nacimiento")
30)     @Temporal(TemporalType.DATE)
31)     private Date fechaNacimiento;
32)
33)     @Column(name="cedula")
34)     private String cedula;
35)     //setters and getters
36)
37)     public Long getId() {
38)         return id;
39)     }
40)
41)     public void setId(Long id) {
42)         this.id = id;
43)     }
44)
45)     public String getNombre() {
46)         return nombre;
47)     }
48)
49)     public void setNombre(String nombre) {
50)         this.nombre = nombre;
51)     }
52)
```

```

53)     public String getApellido() {
54)         return apellido;
55)     }
56)
57)     public void setApellido(String apellido) {
58)         this.apellido = apellido;
59)     }
60)
61)     public Date getFechaNacimiento() {
62)         return fechaNacimiento;
63)     }
64)
65)     public void setFechaNacimiento(Date fechaNacimiento) {
66)         this.fechaNacimiento = fechaNacimiento;
67)     }
68)
69)     public String getCedula() {
70)         return cedula;
71)     }
72)
73)     public void setCedula(String cedula) {
74)         this.cedula = cedula;
75)     }
76)
77) }
78)

```

7) Crear la clase JpaUtils en el package utils (esta clase tendrá la factoria de entidades)

```

package utils;

import javax.persistence.*;

public class JpaUtil {
    private static final EntityManagerFactory emf;

    static{
        try{
            emf=Persistence.createEntityManagerFactory("JPA_Demo"); // esta
declarado en el persistence.xml
        } catch (Throwable ex) {
            System.err.println("Initial SessionFactory creation failed." + ex );
            throw new ExceptionInInitializerError(ex);
        }
    }
}

```

```

        public static EntityManagerFactory getEntityManagerFactory(){
            return emf;
        }
    }
}

```

8) Crear la clase DaoPersona en el package dao (esta clase tendrá el comportamiento de ABM)

```

package dao;

import javax.persistence.*;

import model.Persona;
import utils.JpaUtil;

public class DaoPersona {

    public static Persona find(Long id){
        EntityManager
em=JpaUtil.getEntityManagerFactory().createEntityManager();
        Persona persona=null;
        try{
            persona=em.find(Persona.class, id);
        }catch(Exception ex){
            System.out.println("upss!! ha ocurrido un error");
            ex.printStackTrace();
        }
        finally{
            em.close();
        }
        return persona;
    }

    public static void create(Persona persona){
        EntityManager
em=JpaUtil.getEntityManagerFactory().createEntityManager();
        em.getTransaction().begin();
        try{
            em.persist(persona);
            em.getTransaction().commit();
        }catch(Exception ex){

```

```

        em.getTransaction().rollback();
        System.out.println("Ha ocurrido un error al guardar");
    }finally{
        em.close();
    }
}

    public static void update(Persona persona){
        EntityManager
em=JpaUtil.getEntityManagerFactory().createEntityManager();
        EntityTransaction tx=em.getTransaction();
        tx.begin();
        try{
            em.merge(persona);
            tx.commit();
            System.out.println("Actualizacion exitosa");
        }catch(Exception ex){
            tx.rollback();
            System.out.println("Ha ocurrido un erro al actualizar");
            ex.printStackTrace();
        }
        finally{
            em.close();
        }
    }

    public static void delete(Long id){
        EntityManager
em=JpaUtil.getEntityManagerFactory().createEntityManager();
        em.getTransaction().begin();
        try{
            Persona p=DaoPersona.find(id);
            em.remove(p);
            em.getTransaction().commit();

        }catch(Exception ex){ ex.printStackTrace();
        }
        finally{
            em.close();
        }
    }

}

```

9) Crear la clase Inicio con la rutina main, para ejecutar nuestro código, colocarla en el package ejecutable

```
package ejecutable;

import dao.DaoEmpleado;
import dao.DaoPersona;
import model.Persona;

public class Inicio {

    public static void main(String[] args) {
        Persona p=DaoPersona.find(3L);

        System.out.println("-----PERSONA de base de datos-----");
        System.out.println(p);
        System.out.println("-----Cambios-----");

        p.setNombre("Nadia");
        p.setApellido("Alvarez");

        DaoPersona.update(p);

    }

}
```