

# Configuración de Hibernate

---

Este documento explica paso a paso cómo se configuró Hibernate en el proyecto **Sistema de Reserva de Canchas Deportivas**, utilizando Maven y la base de datos **MariaDB**.

---

## 1 Dependencias Maven

Agrega las siguientes dependencias en tu archivo `pom.xml`:

```
<dependencies>
  <!-- Hibernate Core -->
  <dependency>
    <groupId>org.hibernate</groupId>
    <artifactId>hibernate-core</artifactId>
    <version>5.6.15.Final</version>
  </dependency>

  <!-- Conector MariaDB -->
  <dependency>
    <groupId>org.mariadb.jdbc</groupId>
    <artifactId>mariadb-java-client</artifactId>
    <version>3.1.2</version>
  </dependency>

  <!-- JPA API -->
  <dependency>
    <groupId>jakarta.persistence</groupId>
    <artifactId>jakarta.persistence-api</artifactId>
    <version>2.2.3</version>
  </dependency>
</dependencies>
```

Con Maven, ya no es necesario descargar los `.jar` manualmente, Maven los gestionará por ti.

---

## 2 Archivo de Configuración: `hibernate.cfg.xml`

Ubicado en el directorio `resources/`.

```
<!DOCTYPE hibernate-configuration PUBLIC "-//Hibernate/Hibernate Configuration DTD
3.0//EN"
    "http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">
<hibernate-configuration>
  <session-factory>
    <!-- Configuración de la conexión -->
    <property
name="hibernate.connection.driver_class">org.mariadb.jdbc.Driver</property>
```

```

        <property
name="hibernate.connection.url">jdbc:mariadb://localhost:3306/centro_deportivo</pr
operty>

        <property name="hibernate.connection.username">root</property>
        <property name="hibernate.connection.password"></property>

        <!-- Dialecto -->
        <property
name="hibernate.dialect">org.hibernate.dialect.MariaDBDialect</property>

        <!-- Mostrar consultas SQL en consola -->
        <property name="hibernate.show_sql">true</property>

        <!-- Actualizar el esquema -->
        <property name="hibernate.hbm2ddl.auto">update</property>

        <!-- Clases mapeadas -->
        <mapping class="com.centrodeportivo.model.Cliente"/>
        <mapping class="com.centrodeportivo.model.Cancha"/>
        <mapping class="com.centrodeportivo.model.Reserva"/>
    </session-factory>
</hibernate-configuration>

```

### 3 Anotaciones en las Clases Modelo

Cliente.java

```

@Entity
@Table(name = "Cliente")
public class Cliente {

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

    @Column(nullable = false)
    private String nombre;

    @Column(nullable = false)
    private String telefono;

    // Getters y setters
}

```

Cancha.java

```

@Entity
@Table(name = "Cancha")

```

```
public class Cancha {  
  
    @Id  
    @GeneratedValue(strategy = GenerationType.IDENTITY)  
    private int id;  
  
    @Column(nullable = false)  
    private String tipo;  
  
    @Column(nullable = false)  
    private boolean disponible;  
  
    // Getters y setters  
}
```

## Reserva.java

```
@Entity  
@Table(name = "Reserva")  
public class Reserva {  
  
    @Id  
    @GeneratedValue(strategy = GenerationType.IDENTITY)  
    private int id;  
  
    @ManyToOne  
    @JoinColumn(name = "cliente_id")  
    private Cliente cliente;  
  
    @ManyToOne  
    @JoinColumn(name = "cancha_id")  
    private Cancha cancha;  
  
    @Temporal(TemporalType.TIMESTAMP)  
    private Date fecha;  
  
    // Getters y setters  
}
```

---

## 4 Creación del **SessionFactory**

```
import org.hibernate.SessionFactory;  
import org.hibernate.cfg.Configuration;  
  
SessionFactory sessionFactory = new  
Configuration().configure().buildSessionFactory();
```

## 5 Uso de Sesiones y Transacciones

```
import org.hibernate.Session;
import org.hibernate.Transaction;

Session session = sessionFactory.openSession();
Transaction tx = session.beginTransaction();

// Operaciones: save, update, delete

tx.commit();
session.close();
```

---

## ☒ ¡Hibernate Configurado Correctamente con Maven y MariaDB!

Con esta configuración puedes realizar operaciones CRUD fácilmente y manejar transacciones de forma segura utilizando Maven y conectándote a una base de datos MariaDB.