



Jaime Navas Carmona

Hibernate



Las tablas que tengo son Jugador, Técnica y Equipo.

Un jugador tiene un id, nombre y una posición.

```
public class Jugador {  
    private int idJugador;  
    private String nombre;  
    private String posicion;  
  
    public Jugador(){  
  
    }  
    public Jugador(String nombre, String posicion){  
        this.nombre = nombre;  
        this.posicion = posicion;  
    }  
}
```

Una técnica tiene un id, nombre, tipo y al jugador que pertenece.

```
public class Tecnica {  
    private int idTecnica;  
    private String nombre;  
    private String tipo;  
    private int idJugador;  
  
    public Tecnica(){  
  
    }  
    public Tecnica(String nombre, String tipo){  
        this.nombre = nombre;  
        this.tipo = tipo;  
    }  
}
```



Un equipo tiene un id, nombre y país.

```
public class Equipo {  
    private int idEquipo;  
    private String nombre;  
    private String pais;  
  
    public Equipo(){  
  
    }  
    public Equipo(String nombre, String pais){  
        this.nombre = nombre;  
        this.pais = pais;  
    }  
}
```

**Hay una parte comentada en el código que son las relaciones entre las tablas que me daban fallo por eso las he comentado funciona correctamente todo.**

1 equipo tiene muchos jugadores y 1 jugador tiene muchas técnicas.



## 1.- Crear jugador

```
Elige una opcion
1
Nombre del jugador
jaime
Posicion del jugador
portero
Hibernate:
    insert
    into
        Jugador
        (Nombre, Posicion)
    values
        (?, ?)
Se ha creado el jugador
```

	idJugador	Nombre	Posicion
▶	3	pepe	suplente
	4	ivan	defensa
	5	jaime	portero
●	NULL	NULL	NULL



## 2.- Leer Jugador

```
2
ID del jugador que quieres leer: 5
Hibernate:
    select
        j1_0.idJugador,
        j1_0.Nombre,
        j1_0.Posicion
    from
        Jugador j1_0
    where
        j1_0.idJugador=?
Jugador: nombre=jaime posicion= portero
```

	idJugador	Nombre	Posicion
▶	3	pepe	suplente
	4	ivan	defensa
	5	jaime	portero
●	NULL	NULL	NULL



### 3.-Actualizar jugador

```
Elige una opcion
3
ID del jugador que quieres actualizar: 5
Nuevo nombre: alejandro
Nueva posición: delantero
Hibernate:
    update
      Jugador
    set
      Nombre=?,
      Posicion=?
    where
      idJugador=?
Jugador actualizado: nombre=alejandro posicion= delantero
```

	idJugador	Nombre	Posicion
▶	3	pepe	suplente
	4	ivan	defensa
	5	alejandro	delantero
⬇	NULL	NULL	NULL



#### 4.- Eliminar jugador

```
Elige una opcion
4
ID del jugador a eliminar: 5
Hibernate:
    delete
    from
        Jugador
    where
        idJugador=?
Jugador eliminado: nombre=alejandro posicion= delantero
Menu de opciones
```

	idJugador	Nombre	Posicion
▶	3	pepe	suplente
	4	ivan	defensa
✱	NULL	NULL	NULL



## 5.-Listar jugador

```
5
Hibernate:
    select
        j1_0.idJugador,
        j1_0.Nombre,
        j1_0.Posicion
    from
        Jugador j1_0
    order by
        j1_0.idJugador
Jugador ID: 3 - Nombre: pepe - Posición: suplente
Jugador ID: 4 - Nombre: ivan - Posición: defensa
```

	idJugador	Nombre	Posicion
▶	3	pepe	suplente
	4	ivan	defensa
•	NULL	NULL	NULL





## 6.-Filtrar por equipo

```
Elige una opcion
6
Introduce el nombre del equipo: Zeus
Hibernate:
    select
        e1_0.idEquipo,
        e1_0.Nombre,
        e1_0.Pais
    from
        Equipo e1_0
    where
        e1_0.Nombre like replace(?, '\\', '\\\\')
    order by
        e1_0.idEquipo
Equipo ID: 3 - Nombre: Zeus - Pais: Japón
```

	idEquipo	Nombre	Pais
▶	1	Raimon	Japón
	2	Instituto Imperial	Japón
	3	Zeus	Japón
	4	Génesis	Japón
	5	Kirkwood	Japón