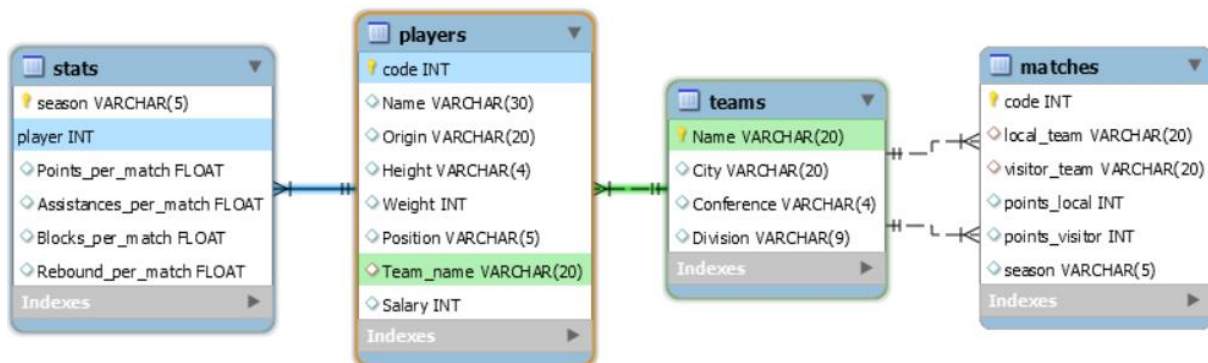


ACTIVIDAD 4.2 – CRUD CON OBJETOS DE HIBERNATE

1. CREAR BASE DE DATOS

Utiliza el fichero *nba.sql* para crear la base de datos en MySQL. Tomate unos minutos para entender el esquema:



2. MAPEA LA DB USANDO HIBERNATE

Configura un nuevo Proyecto y utiliza Hibernate para mapear la base de datos.

3. IMPLEMENTA EL PROGRAMA

Crea un programa con el siguiente menú:

1. Muestra un equipo por ID
2. Muestra un jugador por ID
3. Muestra los jugadores de un equipo existente
4. Crea un nuevo equipo
5. Crea un nuevo jugador asociado a un nuevo equipo
6. Crea un jugador en un equipo existente
7. Borra un jugador
8. Borra un equipo
9. Establece el salario de todos los jugadores de un equipo
10. Aumenta el salario de los mejores jugadores de un equipo en la temporada 07/08
11. Salir

Opción 1

Implementa el método *muestraEquipo()*. Preguntará por una ID y devolverá el equipo con la ID correspondiente por consola. Si no existe ningún equipo con esa ID, se deberá mostrar

un mensaje.

Opción 2

Implementa el método *muestraJugador()*. Este preguntará por la ID de un jugador y mostrará la información por pantalla. Si no existe el jugador, se deberá mostrar un mensaje.

Opción 3

Implementa el método *muestraJugadoresEnEquipo()*. Preguntará por la ID de un equipo y mostrará a los jugadores asociados a este. Si no existe un equipo con esa ID, se deberá mostrar un mensaje.

Opción 4

Implementa el método *crearEquipo()*. Este preguntará los datos necesarios para crear el equipo y crear un objeto persistente en la base de datos.

Opción 5

Implementa el método *crearJugadorYEquipo()*. Este preguntará por los datos necesarios para crear el jugador y el equipo y creará un objeto persistente en la base de datos.

Opción 6

Implementa el método *crearJugadorEnEquipo()*. Este preguntará por los datos necesarios para crear un jugador y la ID de un equipo. El objeto creado se hará persistente en la base de datos y si no existe ningún equipo con la ID introducida se deberá mostrar un mensaje.

Opción 7

Implementa el método *borrarJugador()*. Este preguntará por una ID para seleccionar el jugador indicado y borrarlo de la base de datos. Si no existe ningún jugador con esa ID, se deberá mostrar un mensaje.

Opción 8

Implementa el método *borrarEquipo()*. Este preguntará por el ID de un equipo y lo borrará de la base de datos. Si no existe ningún equipo con ese ID, un mensaje deberá ser mostrado. Si existen jugadores asociados a ese equipo, este no se puede borrar hasta que se hayan borrado los jugadores, se deberá mostrar un mensaje para confirmar la eliminación de los jugadores.

Opción 9

Implementa el método *establecerSalarioEquipo()*. Este preguntará por el ID de un equipo y modificará el salario de los jugadores de éste a la cantidad que se establezca por pantalla. Si no existe ningún equipo con esa ID se deberá mostrar un mensaje por pantalla.

Opción 10

Implementa el método *subirSalarioMejoresJugadoresEquipo()*. Este preguntara por la ID de un equipo, por el porcentaje del aumento que se va aplicar, y cual es la estadística a tener en cuenta. El salario de los jugadores con los máximos en esa estadística se debe incrementar. Si no existe ningún equipo con esa ID se deberá mostrar un mensaje por pantalla.

Por ejemplo, si la ID del equipo es 'Timberwolves', el porcentaje de aumento es un 10% y la estadística seleccionada es 'points per match', el nuevo salario del jugador 7 – Al Jefferson pasará a ser 110000.

Cuando implementes las funciones intenta crear funciones auxiliares para **reutilizar código en común**. Por ejemplo cuando borramos un equipo, la funcionalidad de borrar los jugadores se puede reutilizar.

Recuerda que la creación de una SessionFactory consume recursos, **créala solamente una vez**. **Las funciones no deben recibir Session como argumento**, cada función debe abrir su sesión, hacer un commit de las transacciones y cerrar la sesión.