

Canchas de Paddle

Parte 1: Normalización y Desnormalización

Dada la siguiente estructura se pide realizar DER / Estructura de tablas, normalizado, desnormalizando si fuera necesario, justificando las decisiones tomadas. Si surgieran alternativas de diseño, compararlas.

Identificador de la cancha de paddle

Nombre de la cancha ("Gattiker", "Lasaigues", "Nerone", "Sanz", "Brea")

Código de color

Descripción de color

Tiene iluminación

Inicio del partido

Fin del partido

Jugadores que participaron del partido (2 o 4)

Identificador del jugador

Nombre

Apellido

Domicilio

Fecha nacimiento

Código de paleta con la que juega cada jugador

Nombre de la paleta

Grosor de la paleta en mm

Código de color

Descripción de color

Código del constructor de la paleta

Nombre del constructor de la paleta

Domicilio del constructor de la paleta

Tener en cuenta los siguientes requerimientos:

- **Se necesita saber qué jugadores estuvieron en qué canchas.**
- **Qué paleta usó un jugador en un partido.**
- **De qué color era la cancha en un partido.**
- **Se necesita poder reservar una cancha, no se puede reservar si existe un partido en ese horario.**
- **Saber las canchas disponibles para un horario determinado.**
- **Las canchas que no tienen iluminación, solo aceptan partidos en el horario de 12:00 a 18:00.**

Observaciones:

- Un jugador puede usar una paleta por partido.
- Un jugador tiene una paleta, pero puede cambiarla.
- Los colores de las canchas varían con el tiempo.
- Una cancha que no tiene iluminación, nunca la tendrá ya que se hizo en un lugar que no es factible agregarla.
- La cancha queda reservada por un jugador, ya que nos interesaría poder llamarlo si sucede algo.

Parte 2: ORM

Dado el DER generado en la Parte 1, se pide:

- Realizar una implementación en Java del modelo de datos generado, en clases¹².
- Agregar las JPA Annotations necesarias para realizar el mapeo entre las clases modeladas y el modelo realizado en la Parte 1.
- Conectar el modelo Java con una Base de Datos Relacional. (Recomendamos MySQL³)
- El proyecto debe ser capaz de, al correr, generar las tablas correspondientes en la base de datos, de acuerdo con el DER, sin tablas adicionales ni faltantes y con las relaciones bien formadas.

¹ Como base, recomendamos usar el siguiente repositorio:
<https://github.com/dds-utn/jpa-proof-of-concept-template> (branch *master*, en las otras branches hay ejemplos)

² Como referencia, se recomienda el siguiente apunte:
<https://docs.google.com/document/d/1dYvrVLRbFE9qwuKj5biz9oRBaRzj-K6ujlKOXNan02s/edit?ts=57e1f2b8#heading=h.kkyach7i1h8n>

³ Se puede descargar el servidor [aquí](#) y el cliente para acceder manualmente a la base [aquí](#)