

Se pide desarrollar una aplicación en **Spring Boot** con **JpaRepository** para gestionar una plataforma de cursos online. Tenemos 3 entidades: Curso, DetalleCurso e Instructor. Donde un curso tiene sólo un detalle y cada detalle pertenece a un sólo curso. Un instructor podrá impartir varios cursos pero un curso sólo tendrá un instructor. La relación entre ellas es **bidireccional**. De cada entidad conocemos:

- **Curso:**
  - **id** (Long): (GenerationType.IDENTITY)
  - **titulo** (String)
  - **codigo** (String)
- **DetalleCurso:**
  - **id** (Long): (GenerationType.IDENTITY)
  - **duracionHoras** (Integer)
  - **nivel** (String).
- **Instructor:**
  - **id** (Long): (GenerationType.IDENTITY)
  - **nombre** (String)
  - **especialidad** (String)

**Implementar los siguientes métodos en los servicios correspondientes:**

- A) CRUD para todas la entidades
- B) Obtener todos los cursos con su detalle e instructor
- C) Buscar cursos por título.
- D) Obtener todos los cursos de un instructor, a partir del nombre del instructor.
- E) Buscar cursos cuyo título contenga un texto pasado por parámetro.
- F) Eliminar un curso por su ID que se recibirá por parámetro.
- G) Buscar los detalles de cursos dado su nivel
- H) Buscar los cursos cuyo título comience por una letra, dada por parámetro.
- I) Comprobar si existe el título de un curso.
- J) Obtener los instructores cuyo nombre termina en "EZ"
- K) Obtener los cursos que tengan una duración inferior a un valor pasado por parámetro.

**Parte2:** Probad los métodos del servicio necesarios:

- A) Crear los siguientes cursos de **Carlos Pérez**:  
Curso: "Spring Boot desde cero", código: SPR001, duración: 40 horas, nivel: Intermedio.  
Curso: "JPA Avanzado", código: JPA002, duración: 30 horas, nivel: Avanzado.
- B) Crear los siguientes cursos de **Laura Martínez**:  
Curso: "HTML y CSS Básico", código: WEB101, duración: 20 horas, nivel: Básico.  
Curso: "JavaScript Moderno", código: JS202, duración: 25 horas, nivel: Intermedio.  
Curso: "React con Hooks", código: REACT303, duración: 35 horas, nivel: Avanzado.

- C) Mostrar todos los cursos creados.
- D) Buscar los cursos que contengan “Java”.
- E) Buscar cursos que contengan “Python”.
- F) Actualizar el curso React con Hooks y cambiar la duración a 38 horas
- G) Mostrar los datos del curso React con Hooks.
- H) Eliminar el curso HTML y CSS Básico.
- I) Eliminar el curso inexistente ‘Laravel Profesional’.
- J) Mostrar todos los cursos de Laura Martínez

**Parte3:** Agregar una interfaz web con Thymeleaf para gestionar los cursos de forma visual. Para ello, añadir al proyecto un paquete controller para añadir el controlador. Implementar las siguientes funcionalidades.

Crear una página index.html donde se muestren los siguientes enlaces:

1. Buscar por título del curso
2. Buscar por nombre de Instructor
3. Buscar por nivel
4. Buscar por duración de horas

- B) Buscar por título del curso: URL: /buscarPorTitulo

Función: mostrar un formulario donde el usuario ingresa un texto y se buscan todos los cursos cuyo título contenga ese texto.

- C) Buscar por nombre de Instructor: URL: /buscarPorInstructor

Función: mostrar un formulario donde el usuario ingresa el nombre del instructor y se listan todos los cursos impartidos por ese instructor.

- D) Buscar por nivel del curso: URL: /buscarPorNivel

Función: mostrar un formulario donde el usuario selecciona un nivel (Básico, Intermedio, Avanzado) y se listan los cursos que tienen ese nivel

- E) Buscar por duración de horas. URL: /buscarPorDuracion

Función: mostrar un formulario donde el usuario ingresa un valor numérico y se listan los cursos cuya duración es menor que ese valor.