

## Desafío - Always Music 2.0

- Para realizar este desafío debes haber estudiado previamente todo el material disponibilizado correspondiente a la unidad.
- Una vez terminado el desafío, comprime la carpeta que contiene el desarrollo de los requerimientos solicitados y sube el `.zip` en el LMS.
- Puntaje total: 10 puntos.
- Desarrollo desafío:
  - El desafío se debe desarrollar de manera Grupal.

### Capítulos

El desafío está basado en el siguiente capítulo de la lectura:

- Pooling.
- Mi primera consulta con Pool.
- Consultas con texto parametrizado.
- JSON como argumento de una consulta.
- Prepared Statement.
- Captura de errores.

### Descripción

La escuela de música Always Music solicitó hacer unas pruebas con el avance del desarrollo del sistema de gestión con base de datos PostgreSQL, se dieron cuenta que no se podían hacer varias consultas de forma simultánea y que al intentar hacer una consulta errónea, no recibían ningún error, dejando la posibilidad de creer que un estudiante fue registrado y que esto no sea así.

En este desafío deberás ocupar la clase Pool definiendo sus diferentes propiedades, capturar los posibles errores en el proceso de conexión con la base de datos y realizar las siguientes consultas, usando textos parametrizados y Prepared Statement:

- Agregar un nuevo estudiante.
- Consultar los estudiantes registrados.
- Consultar estudiante por rut.
- Actualizar la información de un estudiante.
- Eliminar el registro de un estudiante.

## Requerimientos

1. Realizar la conexión con PostgreSQL, utilizando la clase Pool y definiendo un máximo de 20 clientes, 5 segundos como tiempo máximo de inactividad de un cliente y 2 segundos de espera de un nuevo cliente. **(2 Puntos)**
2. Hacer todas las consultas con un JSON como argumento definiendo la propiedad name para el Prepared Statement. **(2 Puntos)**
3. Hacer las consultas con texto parametrizado. **(1 Punto)**
4. Liberar a un cliente al concluir su consulta. **(1 Punto)**
5. Capturar los posibles errores en todas las consultas. **(2 Puntos)**
6. Retornar por consola un mensaje de error en caso de haber problemas de conexión. **(1 Punto)**
7. Obtener el registro de los estudiantes registrados en formato de arreglos. **(1 Punto)**