

Desafío - Always Music

- Para realizar este desafío debes haber estudiado previamente todo el material disponibilizado correspondiente a la unidad.
- Desarrollo desafío:
 - El desafío se debe desarrollar de manera Grupal.

Capítulos

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

- Instalación y configuración de la librería pg
- Propiedades del objeto result y la asincronía en una consulta
- Ingresando y consultando datos
- Actualizando y eliminando datos

Descripción

La escuela de música Always Music es reconocida en la ciudad por graduar a grandes músicos de reconocimiento mundial, sin embargo, a pesar de lo mucho que ha crecido con el tiempo, nunca dejaron de utilizar excel como base de datos y han decidido cambiar esto por un desarrollo personalizado.

En este desafío deberás desarrollar una aplicación en Node que realice consultas a PostgreSQL con el paquete “pg” para:

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

Ya que el caso se trata de un proceso de desarrollo, la interacción la debes realizar con argumentos por la línea de comandos o a través de una API REST en Express.

Antes de iniciar este desafío deberás crear una base de datos y una tabla con las siguientes columnas:

- Nombre
- Rut
- Curso
- Nivel

Usa las siguientes imágenes como referencia de lo que debes lograr en el desarrollo de este desafío.

Nuevo estudiante:

```
$ node index.js nuevo 'Brian May' '12.345.678-9' guitarra 7
Estudiante Brian May agregado con éxito
```

Consulta:

```
$ node index.js consulta
Registro actual [
  {
    nombre: 'Brian May',
    rut: '12.345.678-9',
    curso: 'guitarra',
    nivel: '7'
  }
]
```

Editar estudiante:

```
$ node index.js editar 'Brian May' '12.345.678-9' guitarra 10
Estudiante Brian May editado con éxito
```

Consultar por rut:

```
$ node index.js rut '12.345.678-9'
[
  {
    nombre: 'Brian May',
    rut: '12.345.678-9',
    curso: 'guitarra',
    nivel: '10'
  }
]
```

Eliminar registro de estudiante:

```
$ node index.js eliminar '12.345.678-9'
Registro de estudiante con rut 12.345.678-9 eliminado
```

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. Hacer todas las consultas con un JSON como argumento.
3. Hacer las consultas con texto parametrizado.
4. Liberar a un cliente al concluir su consulta si se usa client de pool.
5. Capturar los posibles errores en todas las consultas.
6. Retornar por consola un mensaje de error en caso de haber problemas de conexión.
7. Obtener el registro de los estudiantes registrados en formato de arreglos.