

## CUE: TRANSACCIONALIDAD: OPERACIONES TRANSACCIONALES

### REBOUND EXERCISE: TRANSACCIONES EN NODE-POSTGRES CON UN POOL DE CONEXIONES

Para resolver este ejercicio, anteriormente debe haber revisado la lectura y los videos del CUE: Transaccionalidad: Operaciones Transaccionales.

#### EJERCICIO:

Tomando como referencia los **EXERCISES** de este CUE, en los que se crea una tabla **Cuentas**, y se opera con transacciones(tx) del banco, se creará una nueva tabla "Históricos de Movimientos", que se llamará `historicos_tx`; esto motivado a que se migrarán todos los datos de la tabla `Cuentas`, a la tabla `cuentas_nva`. No debe existir duplicidad en ambas tablas, por lo que se migra por completo, o se devuelven los datos al estado original.

Contamos con la siguiente estructura:

```
1 CREATE TABLE "cuentas" (  
2     "id" SERIAL,  
3     "nombre" VARCHAR(50) NOT NULL,  
4     "balance" DEC(15,2) NOT NULL,  
5     PRIMARY KEY ("id")  
6 );  
7  
8 CREATE TABLE "cuentas_nva" (  
9     "id" SERIAL,  
10    "nombre" VARCHAR(50) NOT NULL,  
11    "balance" DEC(15,2) NOT NULL,  
12    PRIMARY KEY ("id")  
13 );
```

Los pasos a seguir son:

1. Crear un script en node para una tabla en la base de datos.
2. Insertar registros en la tabla `cuentas`.
3. Realizar transacciones con: `Begin`, `Commit`, y `Rollback` con `try catch`.