Desafío - Entendiendo cómo se comportannuestros clientes

--- - 0.- CREAR BASE DE DATOS jobs

--- - \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

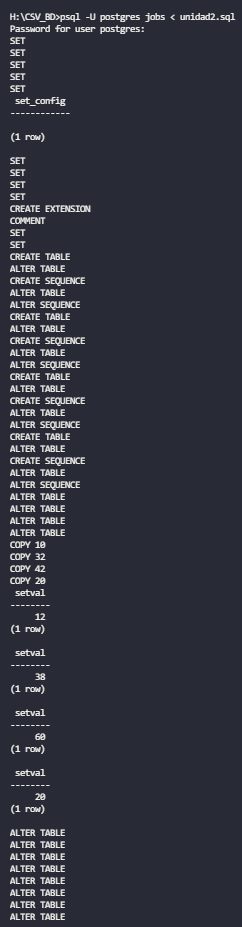
    CREATE DATABASE jobs

-- -- 1.- Cargar el respaldo de la base de datos unidad2.sql

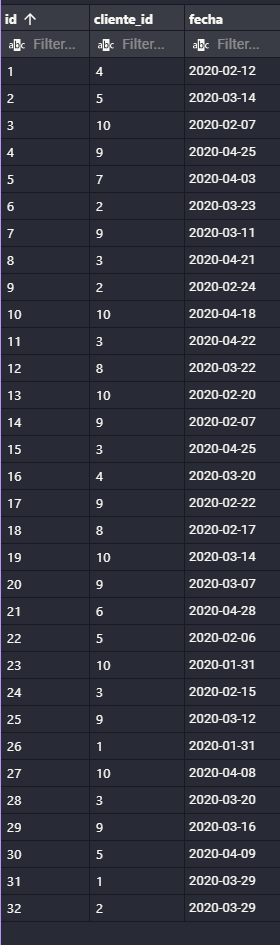
-- -- En el Terminal (CMD) se escribe la sgte sentencia

-- --  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

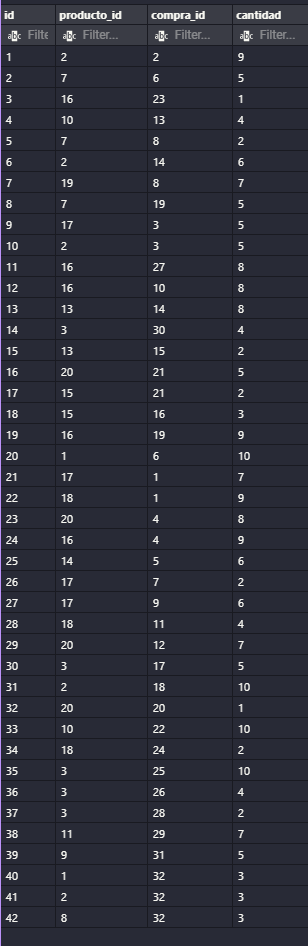
 psql -U postgres jobs < unidad2.sql



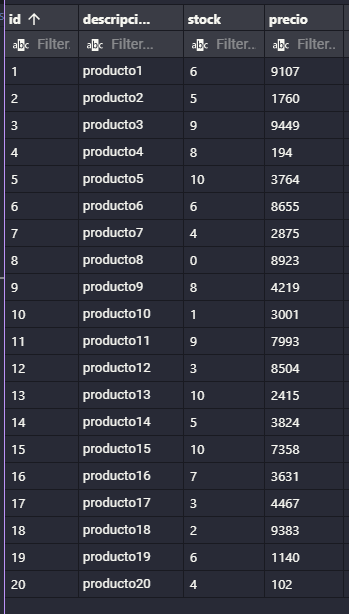
**REGISTRO DE LAS TABLAS DE LA BASE DE DATOS *Jobs***



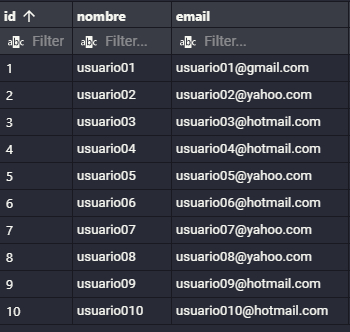
COMPRA



DETALLE\_DE\_COMPRA



PRODUCTO



CLIENTE

-- 2. El cliente usuario01 ha realizado la siguiente compra:

 -- -- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- ● producto: producto9

-- ● cantidad: 5

-- ● fecha: fecha del sistema

-- Mediante el uso de transacciones, realiza las consultas correspondientes para este

-- requerimiento y luego consulta la tabla producto para validar si fue efectivamente descontado en el stock

BEGIN TRANSACTION;

INSERT INTO compra (id,cliente\_id,fecha)

VALUES (33,1,'07-03-2022');

INSERT INTO detalle\_compra (id,producto\_id,compra\_id,cantidad)

VALUES (43,9,33,5);

UPDATE producto

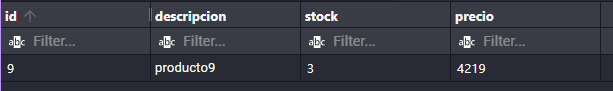
SET stock = stock - 5

WHERE id=9;

COMMIT;

RESULTADO // Antes stock=8 y ahora stock=3 (al descontar 5 en stock), para producto id=9

TABLA PRODUCTO



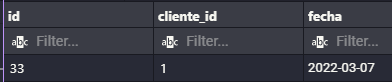
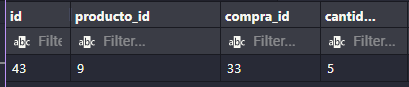
TABLA COMPRA (Se agrego compra id=33)

TABLA DETALLE \_COMPRA (Se agrego compra id=33)



-- -- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- 3. El cliente usuario02 ha realizado la siguiente compra:

-- -- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- ● producto: producto1, producto 2, producto 8

-- ● cantidad: 3 de cada producto

-- ● fecha: fecha del sistema

-- Mediante el uso de transacciones, realiza las consultas correspondientes para este

-- requerimiento y luego consulta la tabla producto para validar que si alguno de ellos

-- se queda sin stock, no se realice la compra

BEGIN TRANSACTION;

INSERT INTO compra (id,cliente\_id,fecha)

VALUES(34,2,'07-03-2022');

INSERT INTO detalle\_compra(id,producto\_id,compra\_id,cantidad)

VALUES(44,1,34,3);

UPDATE producto

SET stock = stock - 3 WHERE id=1;

SAVEPOINT a;

INSERT INTO detalle\_compra(id, producto\_id, compra\_id, cantidad)

VALUES (45,2,34,3);

UPDATE producto

SET stock = stock -3 WHERE id=2;

SAVEPOINT b;

INSERT INTO detalle\_compra(id, producto\_id, compra\_id, cantidad)

VALUES (46,8,34,3);

UPDATE producto

SET stock = stock -3 WHERE id=8;

ROLLBACK TO b;

COMMIT;

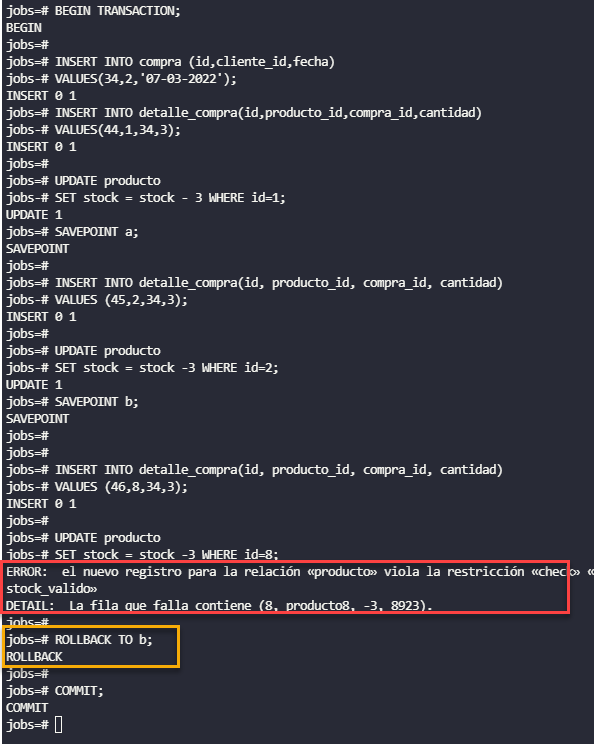
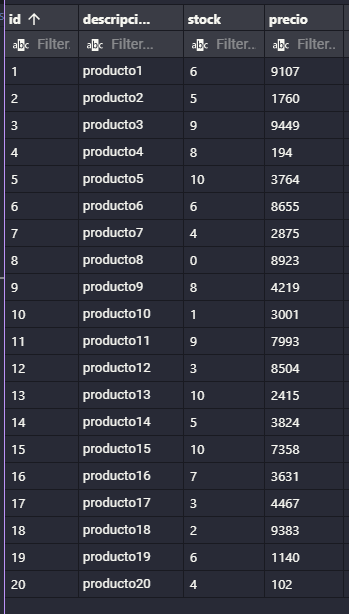


TABLA PRODUCTO **ANTES**



PRODUCTO

TABLA PRODUCTO  **DESPUES ,** (después de descontar 3 unidades al stock)

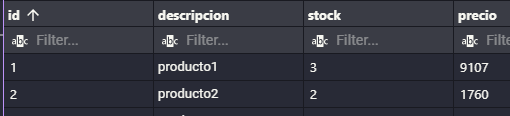
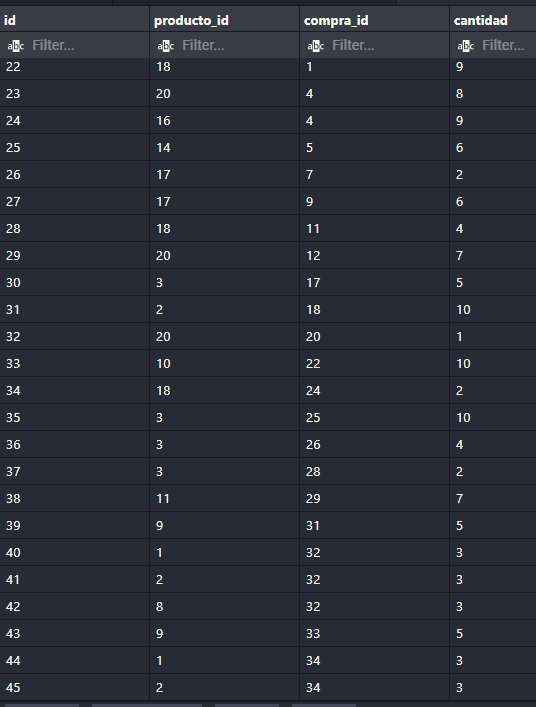
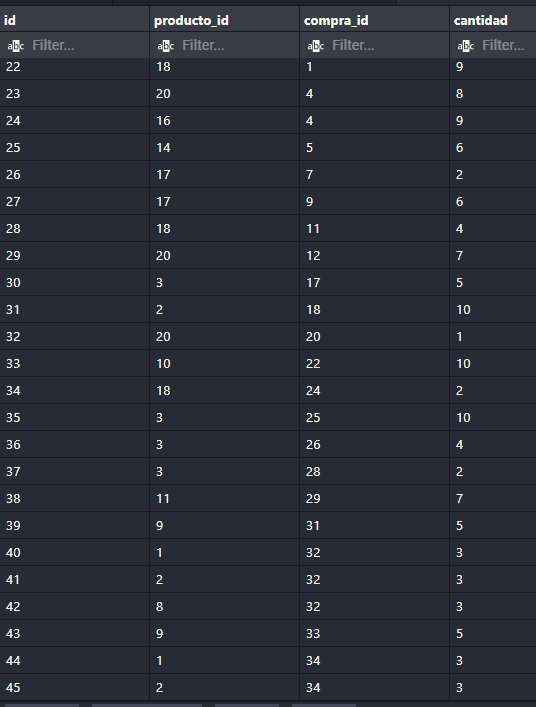
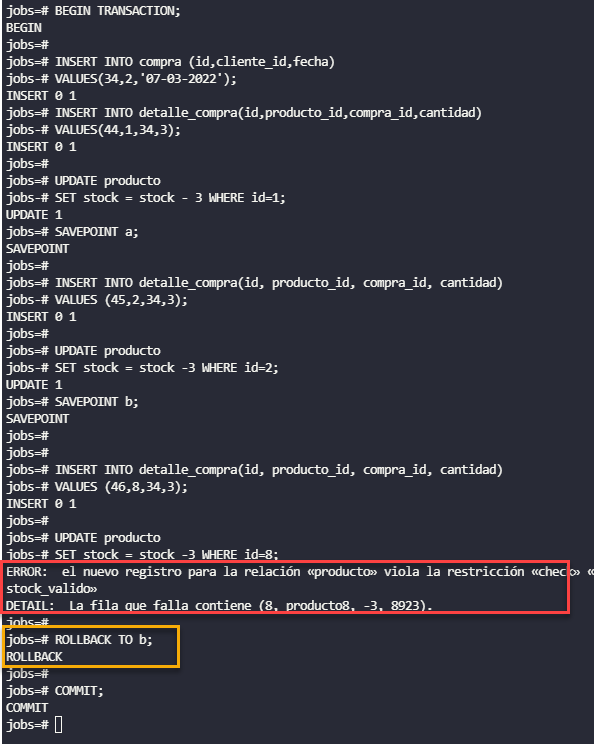


TABLA DETALLE\_COMPR (Se agregaron registros id=45 e id=46)



No se puede hacer UPDATE al producto id=8 por temer stock=0, por lo cual el script aplica la restricción



-- --- - \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- 4. Realizar las siguientes consultas:

-- --- - \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- a. Deshabilitar el AUTOCOMMIT

-- b. Insertar un nuevo cliente

-- c. Confirmar que fue agregado en la tabla cliente

-- d. Realizar un ROLLBACK

-- e. Confirmar que se restauró la información, sin considerar la inserción del

-- punto b

-- f. Habilitar de nuevo el AUTOCOMMIT

-- -- a.- Deshabilitar el AUTOCOMMIT

postgres=# \c jobs

You are now connected to database "jobs" as user "postgres".

jobs=# \set AUTOCOMMIT off

jobs=# \echo :AUTOCOMMIT

off

jobs=#

-- -- -- b.-

BEGIN TRANSACTION;

SAVEPOINT new\_client;

-- b. Insertar un nuevo cliente

INSERT INTO cliente (id,nombre,email)

VALUES (11,'usuario011','usuario011@hotmail.com');

 Confirmar que fue agregado en la tabla cliente

SELECT\*FROM cliente

ORDER BY id ASC;

 d. Realizar un ROLLBACK

ROLLBACK to new\_client;

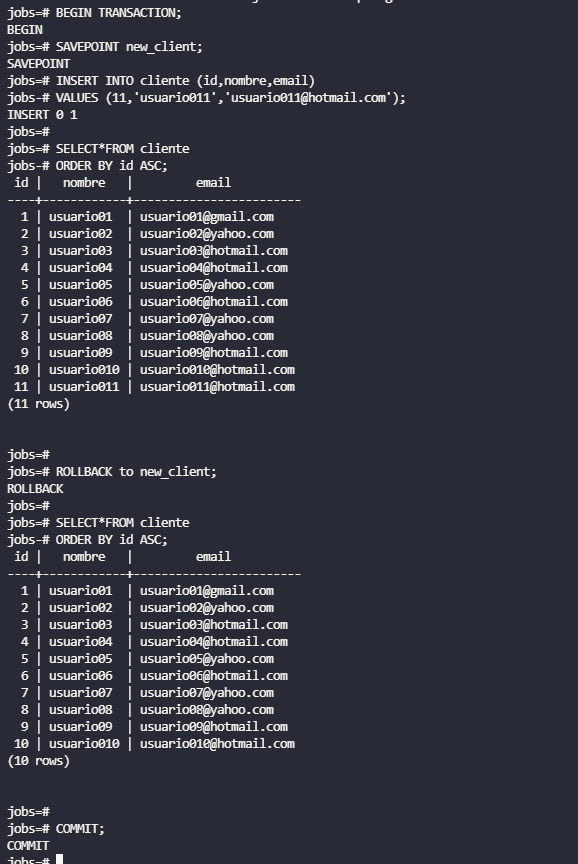
e. Confirmar que se restauró la información, sin considerar la inserción del punto b

SELECT\*FROM cliente

ORDER BY id ASC;

COMMIT;

//RESPUES por CMD



-- f. Habilitar de nuevo el AUTOCOMMIT

