**TITULO: TALLER DE PW - SQL**

**TEMA: MANEJO DE PROCEDIMIENTOS ALMACENADOS Y TRIGGERS EN SQL**

**OBJETIVO:** El estudiante crea procedimientos almacenados y triggers que actúan sobre la BBDD actual

**Nombre estudiante \_\_\_\_\_\_ Curso PW\_\_\_\_\_\_\_ Fecha Entrega \_\_\_\_\_\_\_\_\_\_\_\_**

**CURSORES**

Los cursores nos permiten almacenar un conjunto de filas de una tabla en una estructura de datos que podemos ir recorriendo de forma secuencial.

En base de datos un cursor es un mecanismo el cual nos permite procesar fila por fila el resultado de una consulta.

Es importante mencionar que en MySQL los cursores solo podrán ser utilizados dentro de stored procedures.

**Los cursores tienen la propiedad:**

Nonscrollable: sólo pueden ser recorridos en una dirección y no podemos saltarnos filas.

Cuando declaramos un cursor dentro de un procedimiento almacenado debe aparecer antes de las declaraciones de los manejadores de errores (HANDLER) y después de la declaración de variables locales.

Operaciones con cursores

Las operaciones que podemos hacer con los cursores son las siguientes:

**DECLARE**

El primer paso que tenemos que hacer para trabajar con cursores es declararlo. La sintaxis para declarar un cursor es:

DECLARE cursor\_name CURSOR FOR select\_statement

**OPEN**

Una vez que hemos declarado un cursor tenemos que abrirlo con OPEN.

OPEN cursor\_name

**FETCH**

Una vez que el cursor está abierto podemos ir obteniendo cada una de las filas con FETCH. La sintaxis es la siguiente:

FETCH [[NEXT] FROM] cursor\_name INTO var\_name [, var\_name] ...

Cuando se está recorriendo un cursor y no quedan filas por recorrer se lanza el error NOT FOUND, que se corresponde con el valor SQLSTATE ‘02000’. Por eso cuando estemos trabajando con cursores será necesario declarar un handler para manejar este error.

DECLARE CONTINUE HANDLER FOR NOT FOUND ...

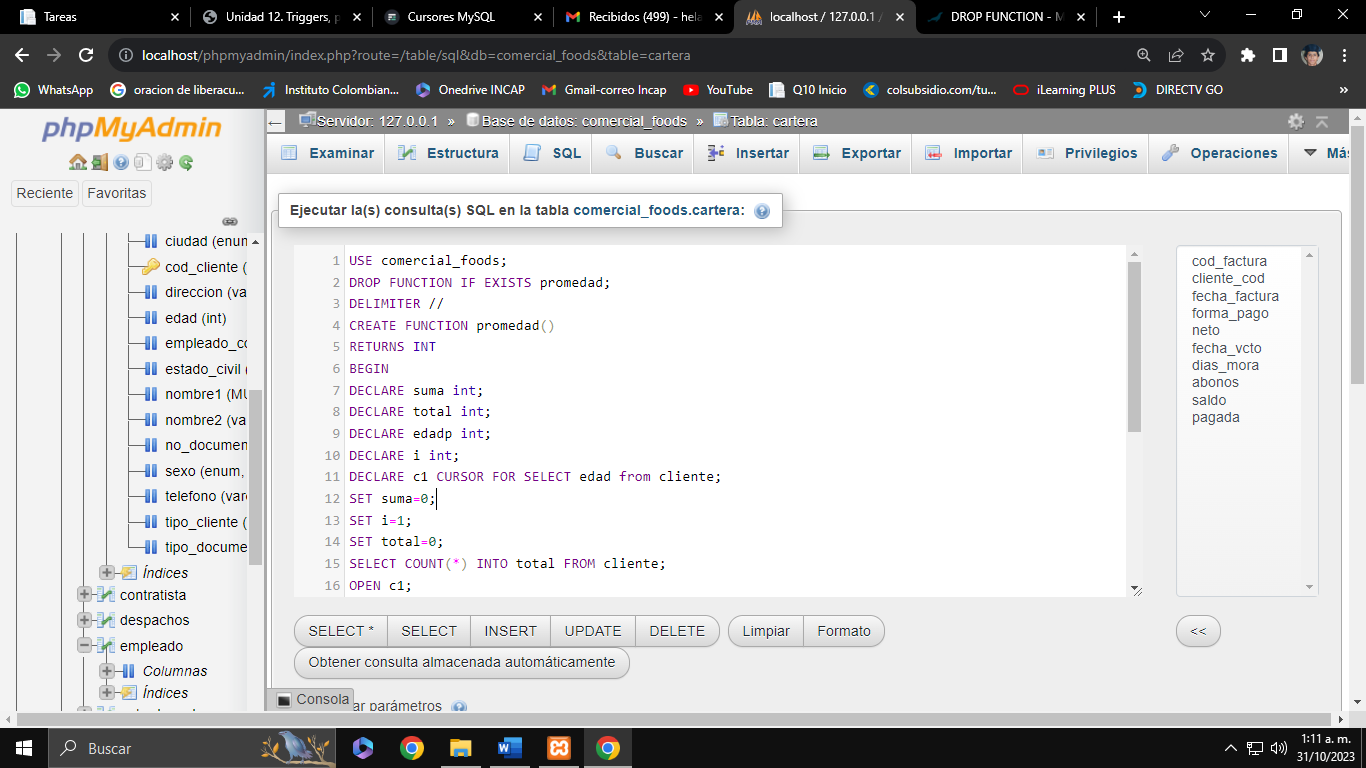
**CLOSE**

Cuando hemos terminado de trabajar con un cursor tenemos que cerrarlo.

CLOSE cursor\_name

**EJEMPLO**

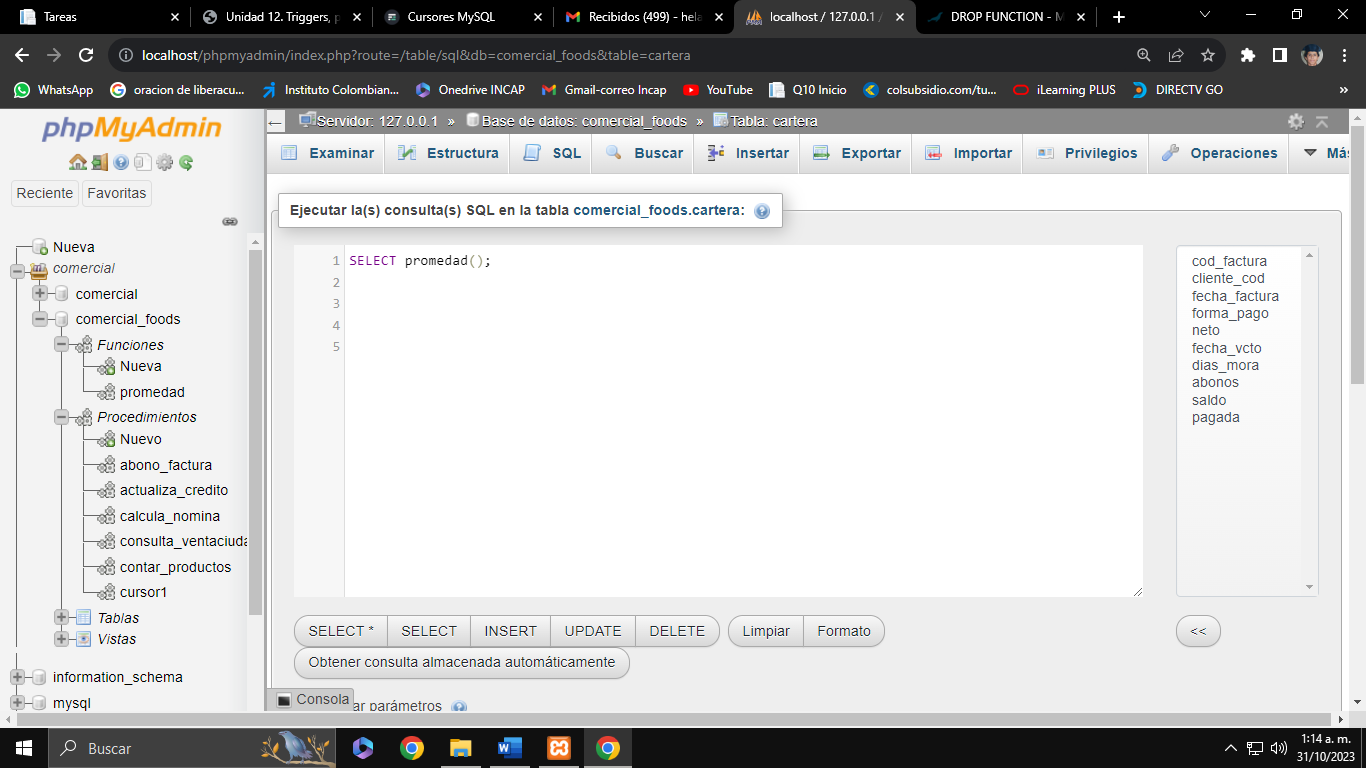
Vamos a crear una función con cursores que permita evaluar el promedio de edad de los clientes de la empresa



Una captura de pantalla de una computadora

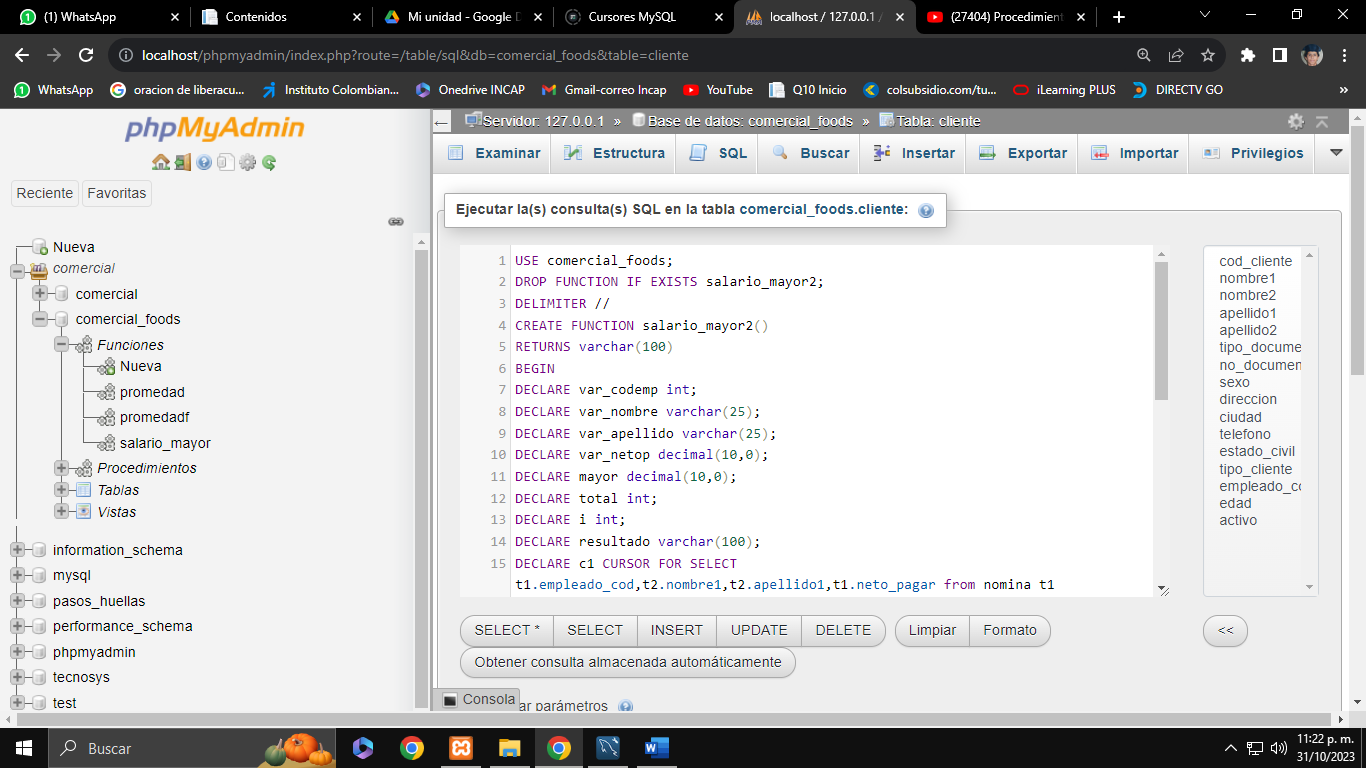
Descripción generada automáticamente

**Para ejecutar:**



**EJEMPLO**

Vamos a crear una función con cursores que permita evaluar el mayor salario neto pagado de la tabla empleado



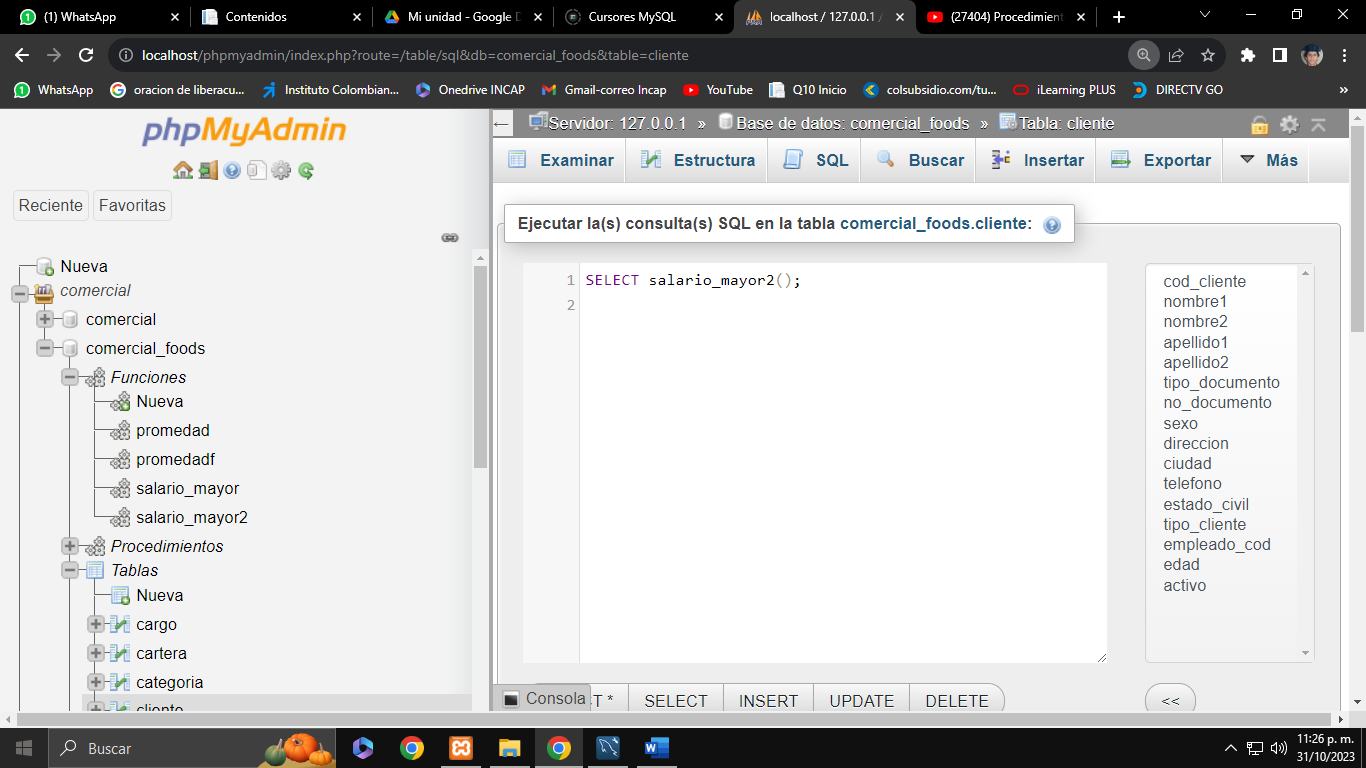
Una captura de pantalla de una computadora

Descripción generada automáticamente

Una captura de pantalla de una computadora

Descripción generada automáticamente

Para ejecutar:



EJEMPLO

Vamos a crear un procedimiento con cursores para inactivar a los clientes en tabla clientes que tengan facturas vencidas con muchos días de mora en cartera

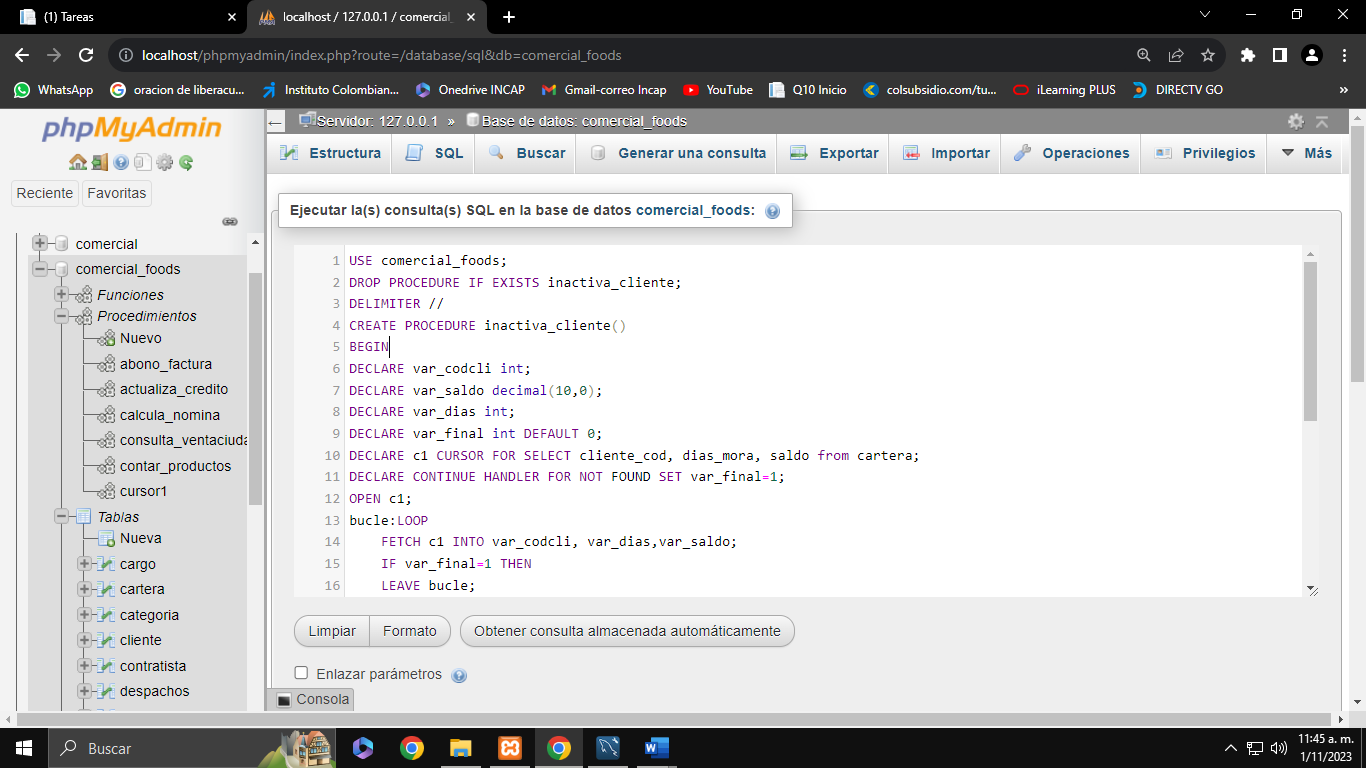
**ANTES HAGAMOS LO SIGUIENTE:**

Ahora vamos a crear en la tabla Cartera la llave foránea (constraint) del campo cliente\_cod con la tabla Cliente en campo cod\_cliente

Una captura de pantalla de una computadora

Descripción generada automáticamente

Escribe este procedure cursor:



Una captura de pantalla de una computadora

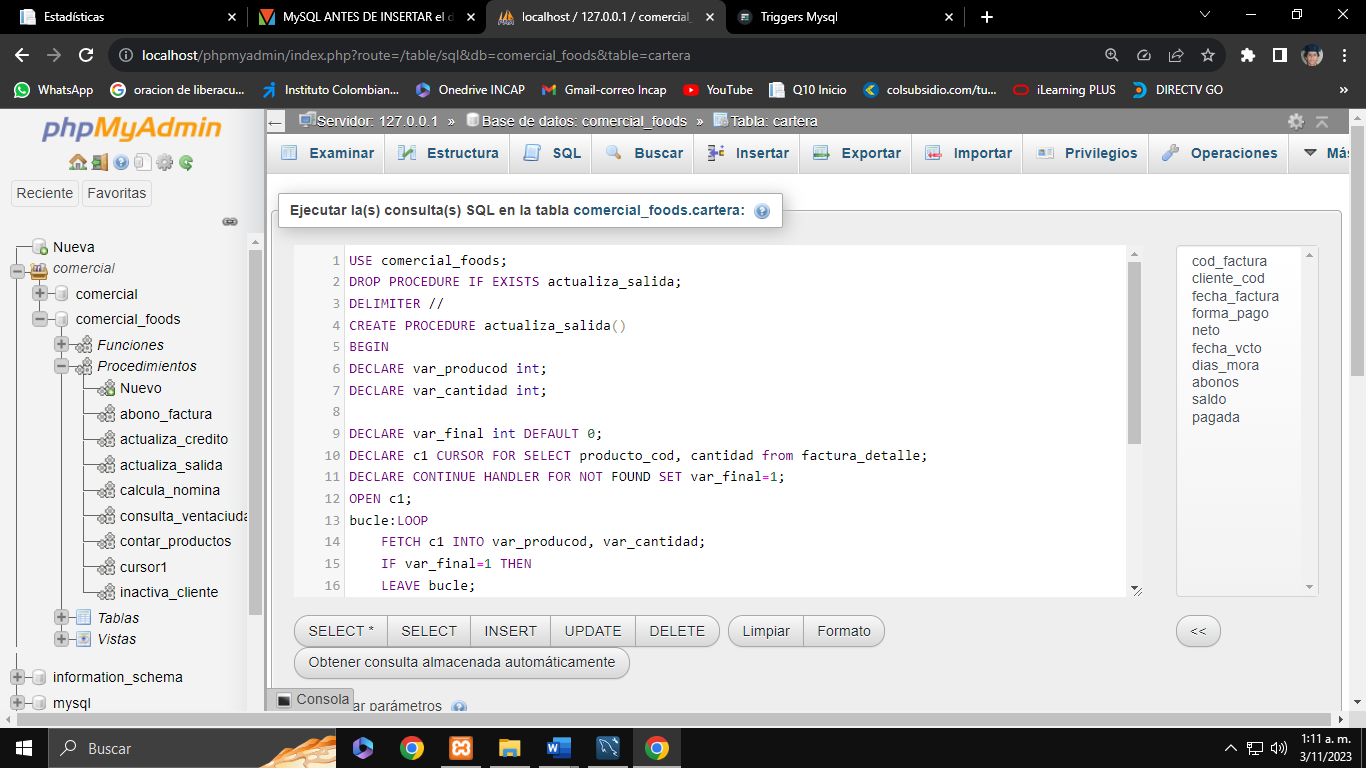
Descripción generada automáticamente

**Para ejecutar:**

Call inactiva\_cliente();

EJEMPLO

Vamos a crear un procedimiento con cursores para actualizar las existencias de los productos con base en las salidas de ventas en tabla **factura\_detalle**



Una captura de pantalla de una computadora

Descripción generada automáticamente

**Para ejecutar:**

Call actualiza\_salida();

**ESTUDIANTE REALIZA ESTE EJERCICIO**

EJEMPLO

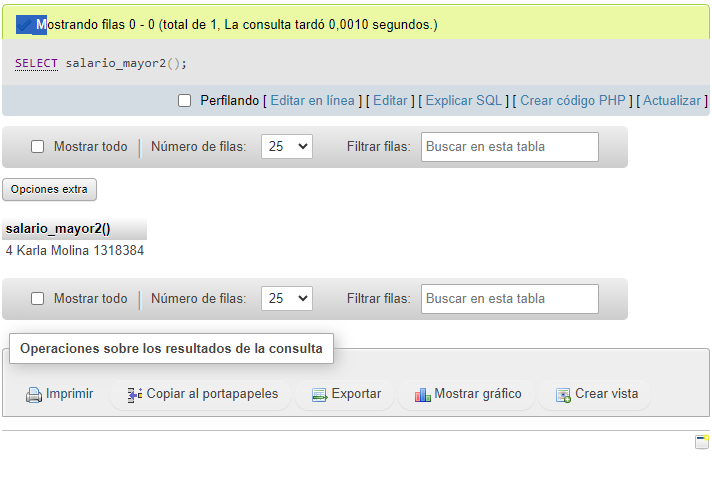
Vamos a crear un procedimiento con cursores para actualizar las existencias de los productos con base en las entradas de compras en tabla **entrada\_detalle.**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente



Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente