**Funciones Utilizadas en los Archivos**

Santiago Alfonso Mendez Pulido. C.C. 1016951011

Yanith Juliana Angarita Chavez. C.C. 1049394392

Ficha: 2996234

Instructor: Mauricio Demetrio Estupiñan

Explicación de las funciones utilizadas en el archivo DDL

1. DROP DATABASE IF EXISTS TECNO;

RTA: Esta función elimina la base de datos TECNO si ya existe y evita errores si vuelves a crear la base de datos y ya existía antes.

2. CREATE DATABASE TECNO;

RTA: Esta función crea una nueva base de datos llamada TECNO. Es el espacio donde se guardarán las tablas, procedimientos y datos.

3. USE TECNO;

RTA: Indica que queremos trabajar con la base de datos TECNO. Todas las acciones siguientes se aplicarán sobre ella.

4. CREATE TABLE

RTA: Crea una nueva tabla en la base de datos. Se usa para definir la estructura que tendrán los datos, como el nombre de las columnas y sus tipos de datos.

5. INT(10)

RTA: Especifica que la columna será de tipo número entero. El número entre paréntesis indica la visualización (no limita el tamaño del número).

6. PRIMARY KEY

RTA: Establece que esa columna será la clave principal de la tabla. No puede repetirse ni ser nula, y sirve para identificar de forma única cada fila.

7. AUTO\_INCREMENT

RTA: Hace que el valor de esa columna se incremente automáticamente en 1 cada vez que se agrega un nuevo registro. Muy común para campos ID.

8. VARCHAR(100)

RTA: Crea una columna que puede almacenar texto de hasta 100 caracteres. Es ideal para nombres o descripciones.

9. DOUBLE

RTA: Define un tipo de dato numérico con decimales. Se usa normalmente para representar precios o cantidades que no son enteras.

10. CREATE TABLE producto (...)

RTA: Crea una tabla llamada producto con campos para guardar el nombre, precio y el código del fabricante al que pertenece el producto.

11. codigo\_fabricante INT(10)

RTA: Es una columna numérica que referencia al código del fabricante. Permite establecer la relación entre productos y fabricantes.

12. ALTER TABLE producto ADD CONSTRAINT fk\_nombre FOREIGN KEY (...)

RTA: Modifica la tabla producto para añadir una clave foránea (foreign key). Esto conecta producto.codigo\_fabricante con fabricante.codigo, creando una relación entre las dos tablas. Garantiza que no se pueda insertar un producto con un fabricante que no exista.

13. REFERENCES fabricante(codigo)

RTA: Especifica a qué tabla y columna se refiere la clave foránea. En este caso, dice que producto.codigo\_fabricante debe coincidir con fabricante.codigo.

Explicación de las funciones utilizadas en el archivo DML

1. CALL insertar\_fabricante(...)

RTA: Ejecuta un procedimiento que agrega un nuevo fabricante.

1. CALL insertar\_producto(...)

RTA: Ejecuta un procedimiento que agrega un nuevo producto con su precio y fabricante.

1. SELECT

RTA: Permite consultar datos de una o varias tablas.

1. AS

RTA: Sirve para poner un nombre alternativo a columnas o resultados.

1. ROUND(valor, decimales)

RTA: Redondea un número a los decimales que se indiquen.

1. TRUNCATE(valor, decimales)

RTA: Corta los decimales sin redondear. Solo deja la parte entera si se usa con 0.

1. UPPER(texto)

RTA: Convierte un texto a mayúsculas.

1. LOWER(texto)

RTA: Convierte un texto a minúsculas.

1. LEFT(texto, n)

RTA: Toma los primeros n caracteres de un texto.

1. SUBSTRING(texto, inicio)

RTA: Devuelve una parte del texto desde cierta posición.

1. CONCAT(texto1, texto2, ...)

RTA: Junta varios textos en uno solo.

1. TRIM(texto)

RTA: Elimina espacios en blanco al inicio y al final del texto.

1. LENGTH(texto)

RTA: Devuelve cuántos caracteres tiene un texto.

1. ORDER BY

RTA: Ordena los resultados según una o más columnas. Puede ser ascendente o descendente.

1. LIMIT n

RTA: Muestra solo una cantidad específica de resultados.

1. OFFSET n

RTA: Omite una cantidad de resultados antes de empezar a mostrar.

1. IN (...)

RTA: Comprueba si un valor está dentro de una lista.

1. BETWEEN a AND b

RTA: Filtra valores que estén entre a y b, incluyendo ambos.

1. NOT

RTA: Niega una condición. Ejemplo: muestra los datos que NO cumplen con algo.

1. LIKE

RTA: Se usa para buscar coincidencias parciales en texto. Ejemplo: 'S%' busca los que empiezan con S.

1. JOIN

RTA: Une dos tablas que tienen columnas relacionadas.

1. LEFT JOIN

RTA: Une dos tablas, mostrando todos los datos de la izquierda, aunque no tengan coincidencia en la derecha.

1. GROUP BY

RTA: Agrupa los resultados por una o varias columnas.

1. COUNT(\*)

RTA: Cuenta cuántas filas hay en total.

1. COUNT(DISTINCT columna)

RTA: Cuenta cuántos valores diferentes hay en una columna.

Explicación de las funciones utilizadas en el archivo de procedimientos

1. DROP PROCEDURE IF EXISTS insertar\_fabricante;

RTA: Elimina el procedimiento si ya existe, para poder crearlo de nuevo sin errores.

1. DELIMITER //

RTA: Cambia el delimitador temporalmente, para poder escribir varias instrucciones dentro de un procedimiento.

1. CREATE PROCEDURE insertar\_fabricante(p\_nombre)

RTA: Crea un procedimiento que inserta un fabricante nuevo usando el valor que se le pasa como parámetro.

1. INSERT INTO fabricante(nombre) VALUES (p\_nombre);

RTA: Inserta el valor recibido en la tabla fabricante.

1. CREATE PROCEDURE insertar\_producto(nombre, precio\_produc, cod\_fabricante)

RTA: Crea un procedimiento que permite registrar un nuevo producto con nombre, precio y fabricante.

1. INSERT INTO producto(nombre, precio, codigo\_fabricante) VALUES (...);

RTA: Guarda el producto en la tabla con los valores recibidos por parámetro.

1. DROP PROCEDURE IF EXISTS convertir\_a\_euros;

RTA: Borra ese procedimiento si ya existe. En este caso, está declarado pero vacío.

1. DELIMITER ;

RTA: Restaura el delimitador normal ; al finalizar la definición del procedimiento.