

PROCEDIMIENTOS / FUNCIONES / TRIGGERS (DISPARADORES)

Sobre la Base de Datos **Jardinería**, realiza las siguientes actividades:

1. Escribe un procedimiento que reciba el nombre de un país como parámetro de entrada y realice una consulta sobre la tabla cliente para obtener todos los clientes que existen en la tabla de ese país.
2. Escribe un procedimiento que reciba como parámetro de entrada una forma de pago, que será una cadena de caracteres (Ejemplo: PayPal, Transferencia, etc) Y devuelva como salida el pago de máximo valor realizado para esa forma de pago. Deberá hacer uso de la tabla pagos.
3. Escribe un procedimiento que reciba como parámetro de entrada una forma de pago, que será una cadena de caracteres (Ejemplo: PayPal, Transferencia, etc). Y devuelva como salida los siguientes valores teniendo en cuenta la forma de pago seleccionada como parámetro de entrada:
 - El pago de máximo valor.
 - El pago de mínimo valor.
 - El valor medio de los pagos realizados.
 - La suma de todos los pagos.
 - El número de pagos realizados para esa forma de pago.
4. Función: `calcular_precio_total_pedido`.
Dado un código de pedido la función debe calcular la suma total del pedido. Tenga en cuenta que un pedido puede contener varios productos diferentes y varias cantidades de cada producto.
 - Parámetros de entrada: `codigo_pedido` (INT)
 - Parámetros de salida: El precio total del pedido (FLOAT)
5. Función: `calcular_suma_pedidos_cliente`.
Dado un código de cliente la función debe calcular la suma de todos los pedidos realizados por el cliente. Deberá hacer uso de la función del apartado anterior `calcular_precio_total_pedido`.
 - Parámetros de entrada: `codigo_cliente` (INT)
 - Parámetros de salida: El suma total de todos los pedidos del cliente (FLOAT)
6. Función: `calcular_suma_pagos_cliente`
Dado un código de cliente la función debe calcular la suma total de los pagos realizados por ese cliente.
 - Parámetros de entrada: `codigo_cliente` (INT)
 - Parámetros de salida: El suma total de todos los pagos del cliente (FLOAT)

7. Procedimiento: calcular_pagos_pendientes

Deberá calcular los pagos pendientes de todos los clientes. Para saber si un cliente tiene algún pago pendiente deberemos calcular cuál es la cantidad de todos los pedidos y los pagos que ha realizado. Si la cantidad de los pedidos es mayor que la de los pagos entonces ese cliente tiene pagos pendientes.

Deberá insertar en una tabla llamada clientes_con_pagos_pendientes los siguientes datos:

- id_cliente
- suma_total_pedidos
- suma_total_pagos
- pendiente_de_pago

8. Escriba una función llamada cantidad_total_de_productos_vendidos que reciba como parámetro de entrada el código de un producto y devuelva la cantidad total de productos que se han vendido con ese código. Escriba una sentencia SQL que realice una llamada a la función realizada para comprobar que se ejecuta correctamente.

9. Escriba un procedimiento llamado obtener_numero_empleados que reciba como parámetro de entrada el código de una oficina y devuelva el número de empleados que tiene. Escriba una sentencia SQL que realice una llamada al procedimiento realizado para comprobar que se ejecuta correctamente.

10. Crea una tabla que se llame notificaciones que tenga las siguientes columnas:

- id (entero sin signo, autoincremento y clave primaria)
- fecha_hora: marca de tiempo con el instante del pago (fecha y hora)
- total: el valor del pago (real)
- codigo_cliente: código del cliente que realiza el pago (entero)

Escriba un trigger que nos permita llevar un control de los pagos que van realizando los clientes. Los detalles de implementación son los siguientes:

Nombre: trigger_notificar_pago

- Se ejecuta sobre la tabla pago.
- Se ejecuta después de hacer la inserción de un pago.
- Cada vez que un cliente realice un pago (es decir, se hace una inserción en la tabla pago), el trigger deberá insertar un nuevo registro en una tabla llamada notificaciones.

Escriba algunas sentencias SQL para comprobar que el trigger funciona correctamente.