

## TAREA 1

```
CREATE VIEW vista_bajo_stock AS  
SELECT *  
FROM products  
WHERE stock < 10;
```

## TAREA 2

```
CREATE TRIGGER insert_ventas  
AFTER INSERT ON ventas  
FOR EACH ROW  
BEGIN  
    INSERT INTO auditoria (id_producto, cantidad_vendida)  
    VALUES ( NEW.id_producto, NEW.cantidad_vendida);  
END;
```

Las palabras NEW y OLD se emplean para referirse a las filas afectadas por el disparador, es decir, a las filas de la tabla sobre la que se activa, para referirse al estado de esa fila, antes (OLD) o después (NEW) de haber actuado el disparador.

### TAREA 3

```
CREATE PROCEDURE realizar_venta(IN p_id_producto INT, IN p_cantidad INT)
BEGIN
    DECLARE stock_actual INT;
    DECLARE msg_error VARCHAR(255);

    -- Iniciar transacción
    START TRANSACTION;

    -- Obtener el stock actual del producto
    SELECT stock INTO stock_actual FROM productos WHERE id = p_id_producto FOR UPDATE;

    -- Verificar si hay suficiente stock
    IF stock_actual < p_cantidad THEN
        SET msg_error = 'Stock insuficiente para realizar la venta';
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = msg_error;
    ELSE
        -- Actualizar el stock del producto
        UPDATE productos SET stock = stock - p_cantidad WHERE id = p_id_producto;

        -- Insertar el registro de la venta
        INSERT INTO ventas (id_producto, cantidad) VALUES (p_id_producto, p_cantidad);

        -- Confirmar transacción
        COMMIT;
    END IF;
END;
```

El procedimiento **realizar\_venta** registra una venta, pero solo si hay suficiente stock disponible del producto. Primero inicia con una transacción y bloquea la fila del producto para evitar conflictos. Luego, verifica si el stock es suficiente y en caso de no serlo saltará un error y no se realizará la venta en este caso. Si hay stock se actualizaría el inventario restando el número de productos vendidos, se registrará la venta en la tabla de ventas y por último se confirma la transacción con el COMMIT del final.