



CIPFP Mislata

Centre Integrat Públic
Formació Professional Superior

Boletín 4 PL/SQL: Ejercicios triggers

```
CREATE TABLE Clientes (
```

```
    id INT PRIMARY KEY,
```

```
    nombre VARCHAR(100),
```

```
    email VARCHAR(100),
```

```
    telefono VARCHAR(20)
```

```
);
```

```
CREATE TABLE Productos (
```

```
    codigo_producto VARCHAR(15) PRIMARY KEY,
```

```
    nombre VARCHAR(100),
```

```
    precio DECIMAL(10,2),
```

```
    cantidad_en_stock INT
```

```
);
```

```
CREATE TABLE Pedidos (
```

```
    codigo_pedido INT PRIMARY KEY,
```

```
    fecha_pedido DATE,
```

```
    fecha_entrega DATE,
```

```
    estado VARCHAR(20),
```

```
    total DECIMAL(10, 2) -- Nuevo campo para almacenar el total del pedido
```

```
);
```

```
CREATE TABLE Detalles_pedido (
```

```
    codigo_pedido INT,
```

```
    codigo_producto VARCHAR(15),
```

```
    cantidad INT,
```

```
    precio_unidad DECIMAL(10,2),
```

```
    entregado BOOLEAN DEFAULT FALSE,
```

```
    PRIMARY KEY (codigo_pedido, codigo_producto),
```

```
    FOREIGN KEY (codigo_pedido) REFERENCES Pedidos(codigo_pedido),
```

```
    FOREIGN KEY (codigo_producto) REFERENCES Productos(codigo_producto)
```



CIPFP Mislata

Centre Integrat Públic
Formació Professional Superior

);

```
INSERT INTO Clientes VALUES (1,'Juan Perez', 'juan@example.com', '123456789');
```

```
INSERT INTO Clientes VALUES (2,'María López', 'maria@example.com', '987654321');
```

```
INSERT INTO Productos (codigo_producto, nombre, precio, cantidad_en_stock) VALUES ('P1', 'Producto 1', 10.99, 100);
```

```
INSERT INTO Productos (codigo_producto, nombre, precio, cantidad_en_stock) VALUES ('P2', 'Producto 2', 20.50, 50);
```

```
INSERT INTO Pedidos VALUES (1,TO_DATE('2024-04-01', 'YYYY-MM-DD'), TO_DATE('2024-04-05', 'YYYY-MM-DD'), 'Pendiente');
```

```
INSERT INTO Pedidos VALUES (2,TO_DATE('2024-04-02', 'YYYY-MM-DD'), TO_DATE('2024-04-06', 'YYYY-MM-DD'), 'En Proceso');
```

```
INSERT INTO Detalles_pedido (codigo_pedido, codigo_producto, cantidad, precio_unidad) VALUES (1, 'P1', 2, 10.99);
```

```
INSERT INTO Detalles_pedido (codigo_pedido, codigo_producto, cantidad, precio_unidad) VALUES (1, 'P2', 3, 20.50);
```

```
INSERT INTO Detalles_pedido (codigo_pedido, codigo_producto, cantidad, precio_unidad) VALUES (2, 'P1', 1, 10.99);
```

EJERCICIO 1. Crea un trigger que se active después de insertar un nuevo cliente en la tabla Clientes. El trigger debe registrar la acción de inserción en una tabla llamada “auditoría de clientes”. Esa tabla tiene los siguientes campos:

auditoria_clientes(id_auditoria number, id_cliente number, accion varchar2(30), fecha date)

EJERCICIO 2. Crea un trigger que se active después de insertar o actualizar un detalle de pedido en la tabla Detalles_pedido.

El trigger debe actualizar automáticamente el campo cantidad_en_stock del producto correspondiente en la tabla Productos basado en la cantidad vendida en el detalle de pedido.

EJERCICIO 3. Crea un trigger que se active después de insertar o actualizar un detalle de pedido en la tabla Detalles_pedido. El trigger debe calcular automáticamente el total de un pedido sumando el precio total de todos los detalles de pedido asociados al pedido y actualizar el campo total en la tabla Pedidos.



CIPFP Mislata

Centre Integrat Públic
Formació Professional Superior

EJERCICIO 4. Crea un trigger que se active después de insertar o actualizar un detalle de pedido en la tabla Detalles_pedido. El trigger debe verificar si todos los productos de un pedido han sido entregados y, si es así, actualizar automáticamente el estado del pedido a "Entregado" en la tabla Pedidos.

EJERCICIO 5. Crea un trigger que, después de insertar un nuevo pedido en la tabla "Pedidos", actualice la cantidad en stock de todos los productos asociados a ese pedido en la tabla "Detalles_pedido". Utiliza un cursor para recorrer los detalles del pedido y actualizar la cantidad en stock de cada producto.