

### 1. Generar el código SQL de las relaciones resultantes en el ejercicio 5\_5

```
CREATE TABLE CLIENTE (  
  id_cliente INT,  
  nombre_cliente VARCHAR(100) NOT NULL,  
  estado VARCHAR(30)  
);
```

```
CREATE TABLE ARTICULO (  
  no_articulo INT,  
  nombre_articulo VARCHAR(100) NOT NULL,  
  precio DECIMAL(10,2) NOT NULL  
);
```

```
CREATE TABLE ORDEN (  
  id_orden INT,  
  fecha DATE NOT NULL,  
  id_cliente INT  
);
```

```
CREATE TABLE DETALLE_ORDEN (  
  id_orden INT NOT NULL,  
  no_articulo INT NOT NULL,  
  cantidad INT NOT NULL  
);
```

### 2. Agregar las restricciones a los atributos generados usando ALTER

// Primero las llaves primarias

```
ALTER TABLE CLIENTE  
  ADD CONSTRAINT pk_cliente PRIMARY KEY (id_cliente);
```

```
ALTER TABLE ARTÍCULO  
  ADD CONSTRAINT pk_articulo PRIMARY KEY (no_articulo);
```

```
ALTER TABLE ORDEN
```

```
ADD CONSTRAINT pk_orden PRIMARY KEY (id_orden);
```

```
ALTER TABLE DETALLE_ORDEN  
ADD CONSTRAINT pk_detalle_orden PRIMARY KEY (id_orden, no_articulo);
```

// Ahora las FKs

```
ALTER TABLE orden  
ADD CONSTRAINT fk_orden_cliente FOREIGN KEY (id_cliente) REFERENCES  
CLIENTE(id_cliente) ON UPDATE CASCADE ON DELETE SET NULL;
```

```
ALTER TABLE DETALLE_ORDEN  
ADD CONSTRAINT fk_detalle_orden_orden FOREIGN KEY (id_orden) REFERENCES  
orden(id_orden) ON UPDATE CASCADE ON DELETE CASCADE;
```

### **3. Agregar 3 registros a cada tabla generada**

```
INSERT INTO CLIENTE (id_cliente, nombre_cliente, estado) VALUES (1, 'Carlos Pérez',  
'CDMX'), (2, 'Ana López', 'Jalisco'), (3, 'Luis García', 'Puebla');
```

```
INSERT INTO ARTICULO (no_articulo, nombre_articulo, precio) VALUES (100, 'Teclado',  
450.00), (101, 'Mouse', 250.00), (102, 'Monitor', 3200.00);
```

```
INSERT INTO ORDEN (id_orden, fecha, id_cliente) VALUES (10, '2025-10-25', 1), (11,  
'2025-10-25', 2), (12, '2025-10-25', 3);
```

```
INSERT INTO detalle_orden (id_orden, no_articulo, cantidad) VALUES (10, 100, 2), (10,  
101, 1), (11, 102, 1);
```

### **4. Actualizar el idCliente de algún registro de la tabla cliente. Explicar resultado**

```
UPDATE CLIENTE SET id_cliente = 5 WHERE id_cliente = 2;
```

Esto significa que vamos a cambiar el `id_cliente = 2` → `5` en la tabla `CLIENTE`. Como vimos en clase, es muy importante poner el `where` en la sentencia `UPDATE` para que no se cambien todos los datos de la columna (que en este caso no se podría porque es PK).

## 5. Borrar un registro de la tabla donde se almacena el detalle de la orden.

```
DELETE FROM DETALLE_ORDEN WHERE id_orden = 10 AND no_articulo = 101;
```

Borramos el registro de la PK de esta tabla (que en realidad son dos FKs porque es una relación de muchos a muchos). DETALLE\_ORDEN no tiene dependencias hacia otras tablas, así que el borrado se realiza sin problema. En este caso, no se propaga a otras tablas porque DETALLE\_ORDEN es la tabla hija. (Según tengo entendido, las restricciones ON DELETE afectan sólo cuando borras en las tablas padre, no hijas.)

## Referencias:

- Oracle Corporation. (2025). *MySQL 8.0 Reference Manual: 13.1.20.5 FOREIGN KEY Constraints*. Recuperado de <https://dev.mysql.com/doc/refman/8.0/en/create-table-foreign-keys.html>
- The PostgreSQL Global Development Group. (2025). *PostgreSQL 16 Documentation – Referential Integrity Constraints*. Recuperado de <https://www.postgresql.org/docs/current/ddl-constraints.html>
- Coronel, C., & Morris, S. (2019). *Database Systems: Design, Implementation, & Management* (13th ed.). Cengage Learning.