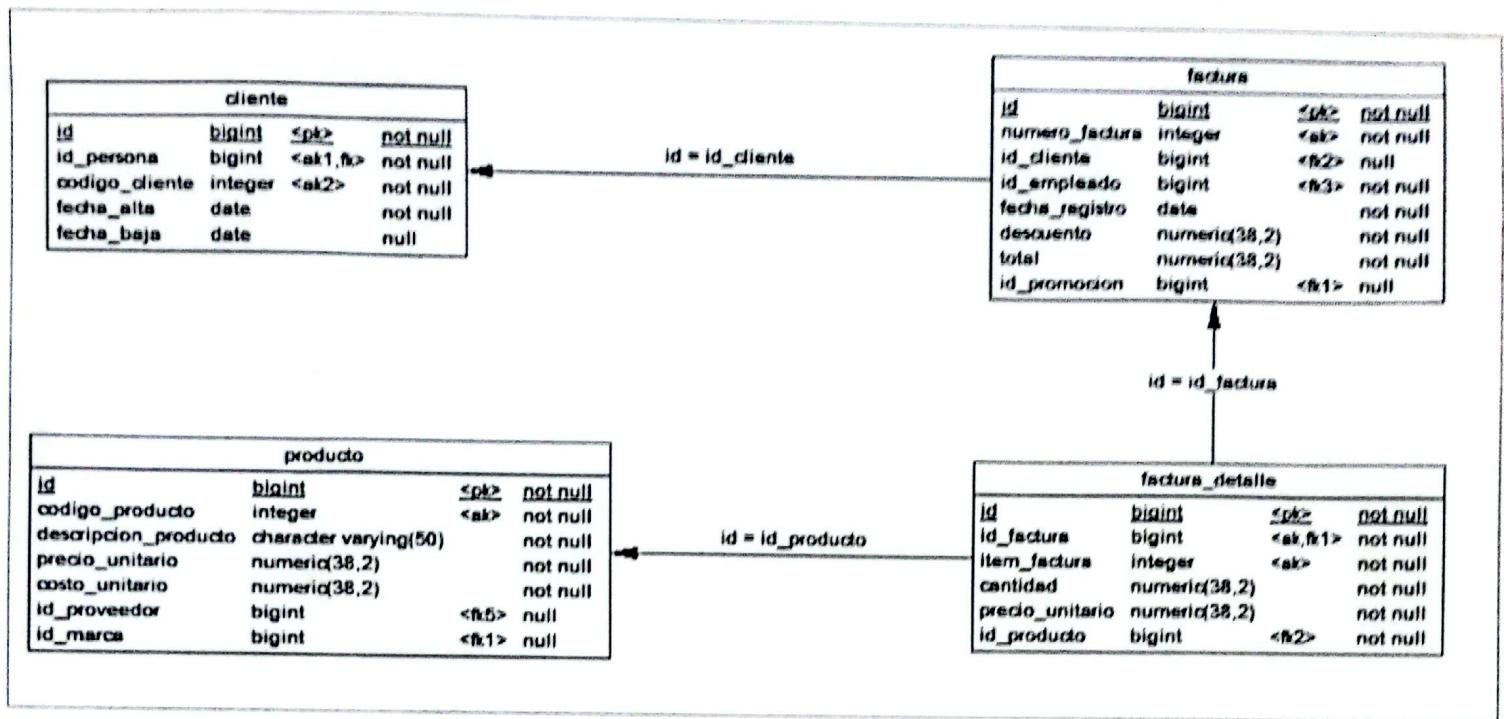


## BASES DE DATOS – EXAMEN FINAL - 10/02/2025

Dado el siguiente extracto del modelo de gestión:



### Ejercicio 1:

Codificar una función en PostgreSQL denominada *f\_productos\_mas\_vendidos()* que permita obtener los productos que figuran en más facturas, considerando solamente aquellos que están presentes mas de "n" veces (siendo n un parámetro de la función). La salida de la función deberá retornar registros del tipo: (producto integer, cantidad integer);

### Ejercicio 2:

Codificar una función en PostgreSQL denominada *f\_productos\_afines()* para obtener **grupos de tres artículos** que aparecen juntos en la misma factura más veces considerando un umbral mínimo a tener en cuenta a partir del cual se contabilizan. Por ejemplo:

Producto 1 Id	Cantidad producto 1	Producto 2 Id	Cantidad producto 1 y 2	Producto 3 Id	Cantidad producto 1, 2 y 3
230	68	231	37	233	37
46	76	230	7	233	6
46	76	231	6	233	5
230	68	233	49	303	5
231	69	233	44	303	5

En esta salida se utiliza como umbral 5 ocurrencias; de haber menos no se tiene en cuenta esa combinación. Puede verse que hay 37 facturas que contienen simultáneamente los productos 230, 231 y 233. La salida de la función debe retornar registros de acuerdo con lo mostrado en la tabla anterior. La función debe tener al umbral como parámetro.

### Ejercicio 3:

Codificar los scripts necesarios para agregar a la tabla *producto* una clave alternativa, conformada por el atributo *codigo\_nemo varchar(20)*, que admite valores nulos. Este atributo representa "códigos nemotécnicos" que se usan para identificar algunos productos, pero no todos los productos tienen este dato.

Al codificar el script considerar que la tabla tiene registros cargados y no se desean perder.

Implementar una alternativa de restricción de unicidad que pueda implementarse en PostgreSQL y una variante para poder implementarla en SQL Server.

### Ejercicio 4:

Codificar una sentencia SQL simple que permita mostrar la cantidad de facturas por cliente, incluyendo también la cantidad de aquellas facturas para las que no se haya registrado cliente (en general, las facturas que no referencian clientes son aquellas que se realizan a "Consumidor Final"). Para el caso de las facturas que tienen registrado cliente, mostrar el código; para el caso de facturas a consumidor final mostrar un "-1" en su lugar.

**Ejercicio 5:**

En el contexto de teoría de conjuntos, considerar dos tablas:

- **Tabla A:** representa el conjunto A.
- **Tabla B:** representa el conjunto B.

Cada tabla tiene un atributo clave primaria llamado ***id*** que identifica de manera única a cada elemento del conjunto.

Se desea obtener:

1. Los elementos que se encuentran en la **intersección** de A y B (elementos presentes en ambos conjuntos).
2. Los elementos de A incluyendo aquellos que también están en B (**conjunto A completo**).
3. Los elementos del conjunto A que **no forman parte de la intersección con B** (es decir, los elementos de A que no están presentes en B).

**Ejemplos de datos:** Basarse en las siguientes tablas para mostrar los resultados esperados de las consultas:

Tabla A	
<b>id</b>	<b>valor</b>
1	'X'
2	'Y'
3	'Z'

Tabla B	
<b>id</b>	<b>valor</b>
2	'Y'
4	'W'
5	'U'

**Instrucciones:**

1. Dibujar el diagrama de Venn correspondiente a los ejemplos de datos presentados.
2. Codificar las consultas SQL correspondientes a cada pregunta, y el resultado esperado.

### ALUMNOS LIBRES:

**Ejercicio 6:**

A partir del modelo físico del sistema de “Gestión”, presentado inicialmente, generar el correspondiente modelo conceptual de datos (MCD). Subrayando los atributos identificativos e indicando aquellos que pueden no tener valor. Incluir en el MCD las entidades que se corresponden con tablas que no se muestran en el modelo físico pero que se denota su existencia por las claves ajenas (FK).

**Ejercicio 7:**

En el ámbito del manejo de concurrencia, en qué consiste “Bloqueo optimista”? Describir como se implementa y como se realizan las transacciones utilizando esta metodología.