

FUNDAMENTOS DE BASES DE DATOS

Examen Julio 2023

La duración del examen es de 3 horas.

Presentar la resolución del examen:

- Con las hojas numeradas y escritas de un solo lado.
- Con la cantidad de hojas entregadas en la primer hoja.
- Con cédula de identidad y nombre en cada hoja.
- Escrita a lápiz y en forma prolija.
- Comenzando cada ejercicio en una nueva hoja

Ejercicio 1 (25 puntos).

La biblioteca universitaria es un espacio dedicado a la gestión y facilitación de recursos educativos para estudiantes, profesores y personal de una universidad. Su objetivo principal es brindar acceso a una amplia variedad de libros y materiales de estudio, así como promover la investigación y el aprendizaje dentro de la comunidad universitaria.

En esta biblioteca, se cuenta con un catálogo de libros que abarca diferentes disciplinas académicas, incluyendo ciencias, humanidades, ingeniería, arte, entre otras. Cada libro se identifica mediante un número de identificación único llamado ISBN (International Standard Book Number) y está asociado a diversos atributos, como el título, autor, año de publicación, editorial y género.

Además, los libros pueden ser clasificados en diferentes categorías según su temática o campo de estudio. Estas categorías ayudan a los usuarios a encontrar libros relacionados y facilitan la organización de la biblioteca. Algunos ejemplos de categorías podrían ser literatura clásica, biología, historia contemporánea, programación, etc. Las categorías se identifican por su nombre.

Cada libro en la biblioteca puede pertenecer a una o más categorías, y no hay un límite estricto en cuanto al número de categorías a las que un libro puede pertenecer. Esto permite una mayor flexibilidad en la clasificación de los materiales de acuerdo con su contenido. En la mayoría de los casos, las categorías no son disjuntas. Es decir, un libro puede estar simultáneamente en varias categorías sin que exista exclusividad entre ellas. Las categorías pueden organizarse en una estructura jerárquica. Esto implica que las categorías más específicas se subordinan a categorías más generales. Por ejemplo, se puede tener una categoría superior llamada "Ciencias" y dentro de ella, categorías secundarias como "Biología", "Química" y "Física".

La biblioteca cuenta con ejemplares de los libros del catálogo. De cada ejemplar se conoce un código, que lo identifica parcialmente respecto al libro, y además la fecha en que fue comprado por la biblioteca.

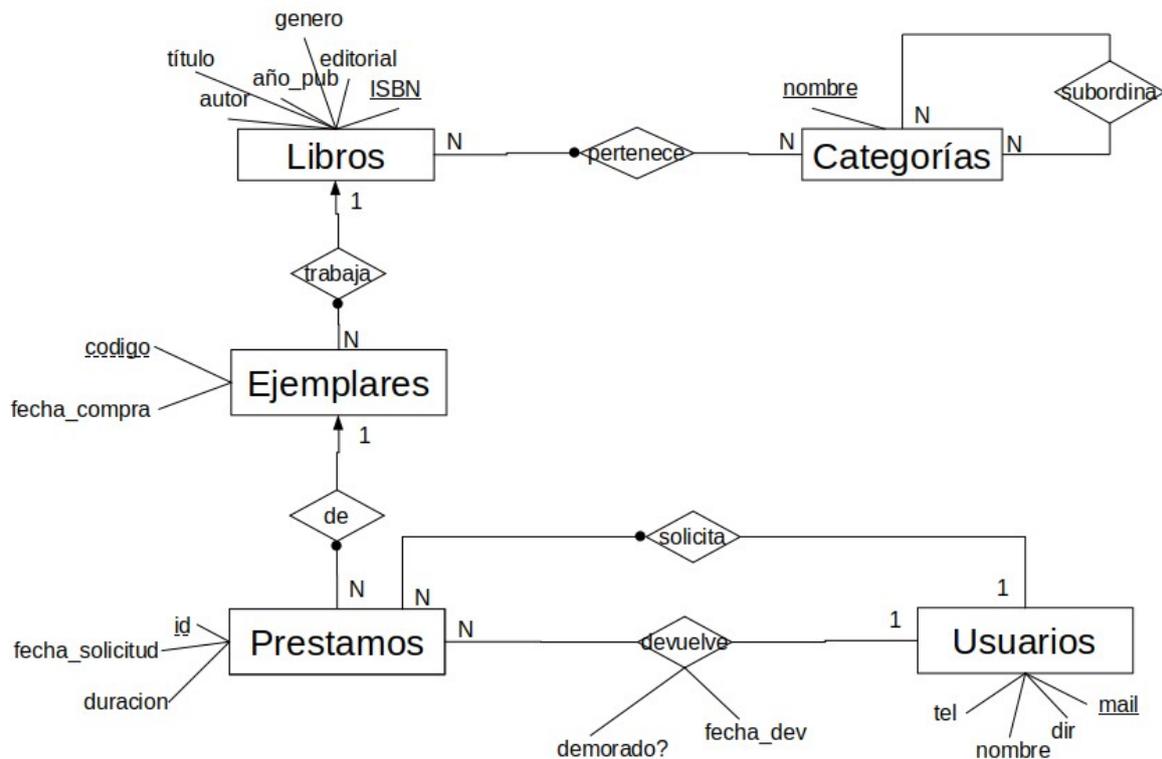
Los estudiantes, profesores y personal de la universidad pueden solicitar préstamos de ejemplares de libros, permitiéndoles llevar los materiales fuera de la biblioteca por un período determinado. Cada préstamo se registra en el sistema con un número de identificación relativo a un ejemplar específico de un libro. Los usuarios de la biblioteca se identifican por su correo electrónico y de ellos se conoce su nombre, dirección, y número de teléfono, para facilitar la gestión de los préstamos y mantener una comunicación efectiva.

Sobre la duración del préstamo se sabe que cada préstamo tiene una fecha de préstamo, que indica cuándo se llevó el ejemplar de libro. También se establece una fecha de vencimiento para cada préstamo. Esta fecha determina cuándo se espera que el ejemplar sea devuelto a la biblioteca. La duración del préstamo puede variar dependiendo de las políticas establecidas por la biblioteca. Por ejemplo, puede ser de 14 días, 30 días, etc.

Cuando un usuario devuelve un ejemplar prestado, se registra la fecha de devolución en el sistema. Se verifica si el ejemplar fue devuelto dentro del plazo establecido o si se ha producido alguna demora. En caso de retraso en la devolución, interesa registrar esto ya que la biblioteca puede aplicar multas o sanciones según sus políticas.

Se pide Modelo Entidad Relación completo.

SOLUCIÓN



RNE

- La relación subordina es irreflexiva.
- La relación subordina es asimétrica.
- Un usuario no puede solicitar un préstamo de un ejemplar que aún no fue comprado (la fecha de compra debe ser anterior a todas las fechas de solicitud de préstamos sobre ese ejemplar)
- Un usuario devuelve préstamos previamente solicitados por él.
- Un usuario no puede solicitar un préstamo de un ejemplar que no fue devuelto.

Ejercicio 2 (25 puntos)

Considere el esquema relacional de la BD de una tienda de venta y entrega de productos a domicilio:

Cientes (cli-id, nombre, dirección, barrio, correo electrónico, teléfono)

Esta tabla contiene información sobre los clientes de la empresa.

Productos (prod-id, nombre, categoría, precio)

Esta tabla contiene información sobre los productos disponibles.

Pedidos (ped-id, fecha, cli-id)

Esta tabla contiene información de los pedidos. Cada pedido se hace en una fecha por un cliente.

DetallePedido (ped-id, prod-id, cantidad)

Esta tabla detalla de los pedidos. Para cada producto pedido se conoce su cantidad.

Empleados (emp-id, nombre, cargo, salario)

Esta tabla contiene información de los empleados de la empresa.

$\pi_{cli_id}(Pedidos) \subseteq \pi_{cli_id}(Clientes)$
 $\pi_{ped_id}(DetallePedido) \subseteq \pi_{ped_id}(Pedidos)$
 $\pi_{prod_id}(DetallePedido) \subseteq \pi_{prod_id}(Productos)$

1. Resuelva en Álgebra Relacional las siguientes consultas

- a) Obtener el/los empleado/s con el salario más alto

$E_1 = \rho_{emp_id, salario \rightarrow eid1, s1}(Empleados)$
 $A = E_1 \times_{eid1 \neq emp_id \wedge s1 \geq salario} Empleados$
 $sol = \Pi_{eid1, sq}(A) \div \Pi_{salario} Empleados$

2. Resuelva en Cálculo Relacional las siguientes consultas

- a) Encontrar los clientes que han realizado pedidos de todos los productos disponibles de la categoría "Comestibles".

$\{\langle ped.cli_id \rangle / Pedidos(ped) \wedge (\forall p)(Productos(p) \wedge p.categoria = Comestibles \rightarrow (\exists dp)(DetallePedido(dp) \wedge dp.prod_id = p.prod_id \wedge ped.ped_id = dp.ped_id))\}$

- b) Obtener los productos que sólo han sido pedidos por clientes del barrio "Prado"

$\{\langle dp.prod_id \rangle / DetallePedido(dp) \wedge (\forall p)(Pedido(p) \wedge p.prod_id = dp.prod_id \rightarrow (\exists c) Clientes(c) \wedge p.cli_id = c.cli_id \wedge c.barrio = Prado \wedge \neg(\exists c2)(Cliente(c2) \wedge p.cli_id = c2.cli_id \wedge c2.barrio \neq Prado))\}$

3. Resuelva en SQL las siguientes consultas SIN utilizar subconsultas en el from, ni la cláusula with, ni vistas

- a) Obtener el promedio de precio de los productos vendidos en cada pedido, solo para los pedidos cuyo total es mayor a 1000

```

SELECT ped-id AVG (precio)
FROM Productos NATURAL JOIN DetallePedido
GROUP BY ped-id
HAVING SUM(precio*cantidad) > 1000

```

- b) Obtener el id de los clientes que entre el 1/1/2022 y el 1/2/2022 solo realizaron pedidos de productos de categoría "Almacen"

```

SELECT cli-id
FROM Pedidos p NATURAL JOIN DetallePedido dp NATURAL JOIN Productos pr
WHERE p.fecha BETWEEN '1/1/2022' AND '1/2/2022' AND
    pr.categoria = 'Almacen' AND
    NOT EXISTS (SELECT 1
        FROM Pedidos p1 NATURAL JOIN DetallePedido dp1
            NATURAL JOIN Productos pr1
        WHERE p1.cli-id = p.cli-id AND
            p1.fecha BETWEEN '1/1/2022' AND '1/2/2022' AND
            pr.categoria <> 'Almacen')

```

Ejercicio 3 (25 puntos).

Considere una tabla $R(A,B,C,D,E)$. En esta tabla se deberían cumplir las siguientes dependencias funcionales, pero no hay garantías de que la instancia actual de la base las cumplan:

$EC \rightarrow B$

$C \rightarrow A$

Sí hay garantías de que ningún atributo de ninguna tupla tiene valores nulos.

a) Escriba una consulta SQL que devuelva los valores de C que violan la segunda dependencia. Justifique.

```
SELECT r1.C
FROM R r1 JOIN R r2 ON r1.C=r2.C AND r1.A <> r2.A
```

$C \rightarrow A$ si y solo si para todo par de tuplas $t1$ y $t2$

$$t1[C] = t2[C] \Rightarrow t1[A] = t2[A]$$

La consulta planteada devuelve los valores de C que tienen más de un valor de A asociados.

b) Escriba una consulta SQL que devuelva los valores de los atributos de alguna de las claves tal que esa clave se viola para esos valores. Justifique

Primero calculo las claves

- nunca a la izquierda: E, C, D
- solo a la izquierda: A, B
- a ambos lados: -

E, C, D necesariamente están en todas las claves. Entonces calculamos $(E,C,D)^+$
 $(E,C,D)^+ = \{E, C, D, A, B\}$

Como (E,C,D) determina a todos los atributos, es una superclave, además es clave porque no le puedo quitar ningún atributo. Es la única clave porque solo podría agregar atributos haciendo que una posible superclave no sea minimal.

```
SELECT r1.E, r1.C, r1.D
FROM R r1 JOIN R r2 ON r1.E = r2.E AND
                    r1.D = r2.D AND
                    r1.C = r2.C AND
                    (r1.A <> r2.A OR
                    r1.B <> r2.B)
```

De esta forma retornamos los atributos C,D,E para los cuales o bien A o bien B no coinciden.

c) Escriba una descomposición de la tabla R con dos tablas de 3 atributos cada una, tal que **no tenga** join sin pérdida. Justifique.

Una descomposición de dos tablas $R1, R2$ tiene join sin pérdida si y solo si:

$$R1 \cap R2 \rightarrow R1 - R2 \text{ en } F^+ \text{ o } R1 \cap R2 \rightarrow R2 - R1 \text{ en } F^+$$

Por lo tanto hay que encontrar $R1$ y $R2$ tales que no se cumpla ninguna de las dos condiciones.

$R1(E,C,B)$

$$\Rightarrow R1 \cap R2 = \{C\}, R1 - R2 = \{E,B\}, R2 - R1 = \{A,D\}$$

$R2(C,A,D)$

$R1 \cap R2$ no determina a $\{E,B\}$ porque E forma parte de la única clave.

$R1 \cap R2$ no determina a $\{A,D\}$ porque D forma parte de la única clave.

d) Escriba una consulta SQL que devuelva las tuplas espurias para la descomposición mencionada en c).

La siguiente consulta devuelve las tuplas de R y las tuplas espurias.

```
SELECT r1.A, r2.B, r1.C, r2.D, r1.E  
FROM R r1 JOIN R r2 ON r1.C=r2.C
```

e) Determine en qué forma normal se encuentra R. Justifique.

La descomposición está en 1NF.

Mirando las dependencias funcionales y las claves hayadas en la parte b:

$EC \rightarrow B$ es una dependencia parcial porque EC no es superclave y B no es un atributo

$C \rightarrow A$ es una dependencia parcial porque C no es superclave y A no es un atributo

f) Encuentre una descomposición de R en BCNF siguiendo el algoritmo visto en el curso.

1. $EC \rightarrow B$ viola BCNF
R1(A,C,D,E), R2(E,C,B)
2. $C \rightarrow A$ viola BCNF sobre R1
R11(C,D,E), R12(A,C), R2(E,C,B)

Por lo tanto, siguiendo el algoritmo visto en el curso, la descomposición resultante es:

```
R11(C,D,E)  
R12(A,C)  
R2(E,C,B)
```

Ejercicio 4 (25 puntos)

Considerando las siguientes relaciones de la realidad planteada en el ejercicio de Consultas:

Clientes (cliente-id, nombre, dirección, barrio, correo electrónico, teléfono)

Productos (producto-id, nombre, categoría, precio)

Pedidos (pedido-id, fecha, cliente-id)

DetallePedido (pedido-id, producto-id, cantidad)

Empleados (empleado-id, nombre, cargo, salario)

y la siguiente información adicional:

Relación R	n _R	Dist. Unif. de valores	Índices
Productos Pr	25000	V(categoría, Pr) = 200	- Índice Primario sobre producto-id (IPPr) - Índice Sec. árbol b+ sobre categoría (ISCa)
Clientes C	400		- Índice Primario sobre cliente-id (IPC)
Pedidos Pe	5000		- Índice Primario sobre pedido-id (IPPe) - Índice Sec. árbol b+ sobre cliente (ISCI)
DetallePedido D	100000	La cantidad de unidades pedidas de cada producto varía entre 1 y 30.	- Índice Primario sobre pedido-id, producto-id (IPD)

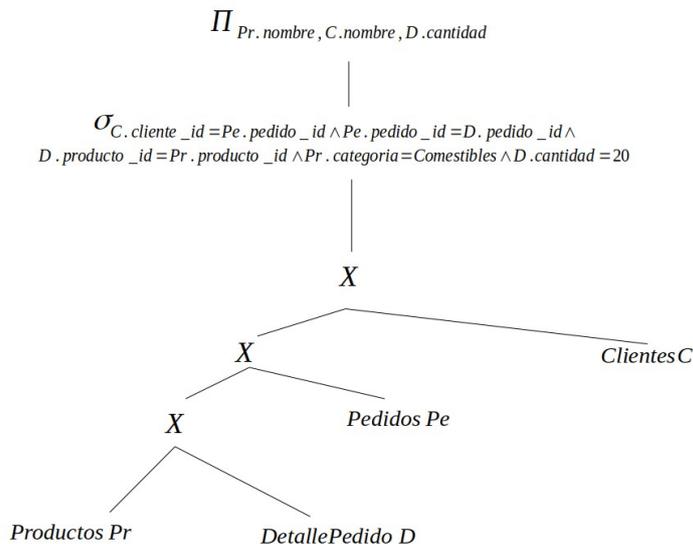
Dada la siguiente consulta SQL:

```

SELECT Pr.nombre, C.nombre, D.cantidad
FROM Productos Pr, DetallePedido D, Pedidos Pe, Clientes C
WHERE C.cliente-id = Pe.cliente-id and Pe.pedido-id=D.pedido-id and D.producto-id = Pr.producto-id
and Pr.categoría = "Comestibles" and D.cantidad = 20;
    
```

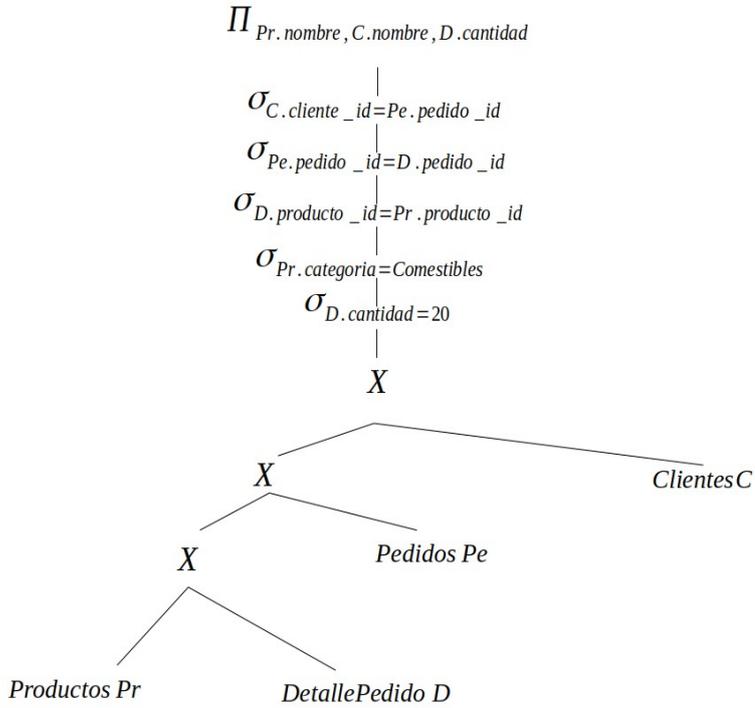
Se pide:

1. Dar el árbol canónico de la consulta.

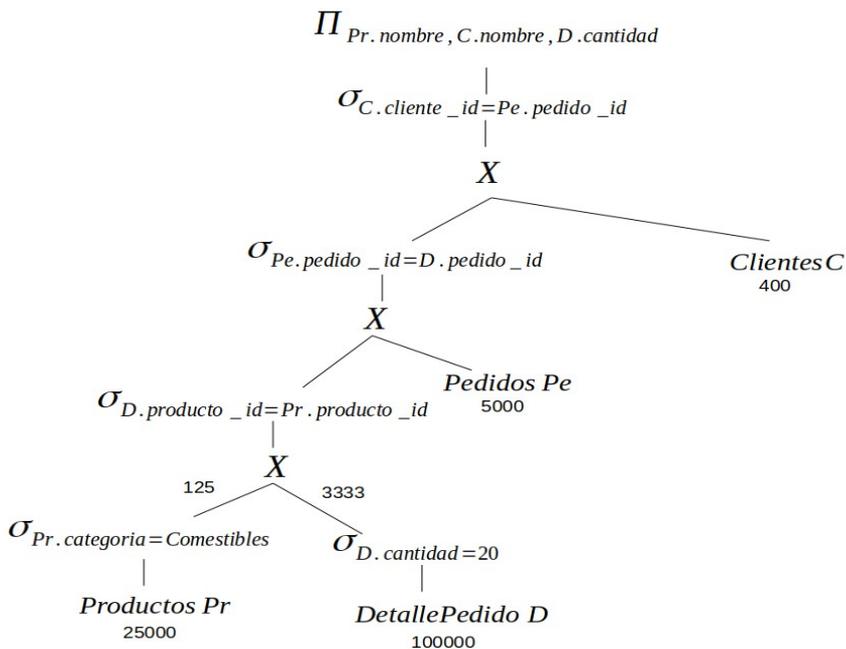


- Construir el plan lógico optimizado según las heurísticas, calculando los tamaños intermedios que sean necesarios. Mostrar cada paso aplicado.

Paso 1: cascada de selecciones

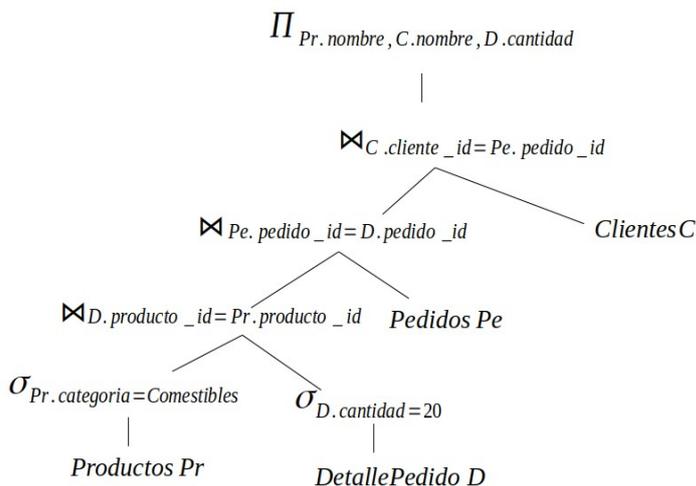


Paso 2: bajar las selecciones lo más posible

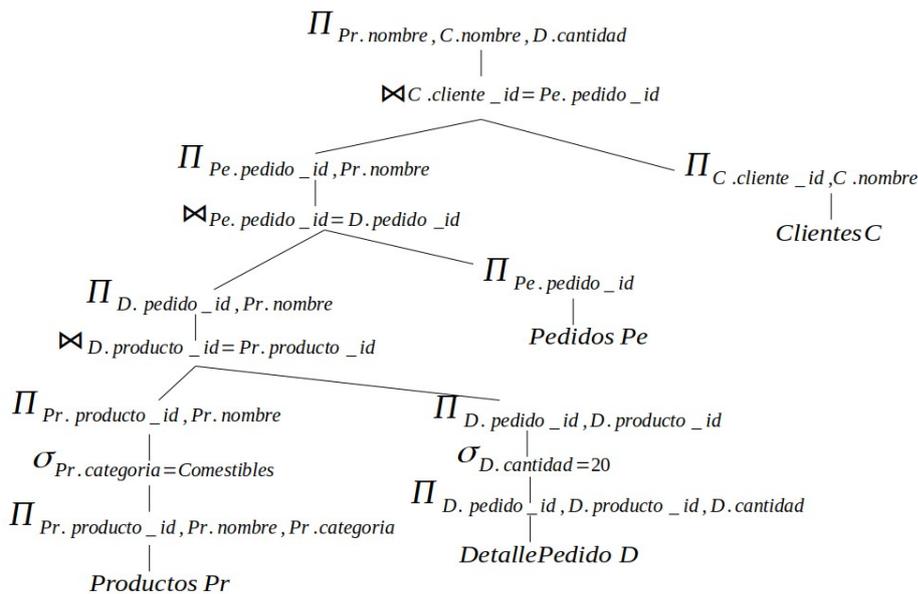


Paso 3: mover las hojas con menos tuplas a la izquierda sin perder joins.
Notar que este paso no genera cambios al árbol anterior.

Paso 4: cambiar las selecciones y productos cartesianos por joins



Paso 5: agregar proyecciones para quedarse solo con los atributos necesarios



3. Dar un plan físico correspondiente al plan dado en la parte 2, donde se utilicen los índices siempre que sea posible (para esta parte no considerar las proyecciones).

