

TRABAJO PRACTICO – LENGUAJES DE CONSULTA

Ej 1.– Sea la siguiente BD:

PROVEEDORES (nro_p, nom_p, categoria, ciud_p)

ITEMS (nro_i, descripcion_i, ciud_i)

PEDIDOS (nro_p, nro_c, nro_i, cantidad, precio)

CLIENTES (nro_c, nom_c, ciud_c)

a) Listar los proveedores de Córdoba

```
SELECT Nro_P, Nom_P
```

```
FROM PROVEEDORES
```

```
WHERE CIUD_P = 'CORDOBA';
```

b) Listar los proveedores que proveen el ítem 1

```
SELECT P.Nro_P, P.Nom_P
```

```
FROM ITEMS AS I, PEDIDOS AS PD, PROVEEDORES AS P
```

```
WHERE I.Descripcion_I = 'I1'
```

```
AND PD.Nro_I = I.Nro_I
```

```
AND P.Nro_P = PD.Nro_P;
```

c) Listar los clientes que solicitan ítems provistos por el proveedor 1

```
SELECT C.NRO_C, C.Nom_C
```

```
FROM CLIENTES AS C, PEDIDOS AS PD, PROVEEDORES AS P
```

```
WHERE P.Nom_P = 'P1'
```

```
AND PD.Nro_P = P.Nro_P
```

```
AND C.Nro_C = PD.Nro_C;
```

d) Listar los clientes que solicitan algún ítem provistos por proveedores con categoría mayor que 4

```
SELECT C.Nro_C, C.Nom_C
```

```

FROM CLIENTES AS C
WHERE C.Nro_C IN (
SELECT DISTINCT PD.Nro_C
FROM PEDIDOS AS PD, PROVEEDORES AS P
WHERE P.Categoria > 4
AND PD.Nro_P = P.Nro_P
);

```

o bien :

```

SELECT DISTINCT CL.nro_c, CL.nom_c
FROM clientes CL
WHERE EXISTS (
SELECT *
FROM pedidos PE, proveedores PR
WHERE PE.nro_p = PR.nro_p
AND CL.nro_c = PE.nro_c
AND PR.categoria > 4
)

```

e) Listar los ítems pedidos por clientes de Rosario.

```

SELECT I.Nro_I, I.Descripcion_I
FROM ITEMS AS I
WHERE I.Nro_I IN (
SELECT PD.Nro_I
FROM CLIENTES AS C, PEDIDOS AS PD
WHERE C.Ciud_C = 'Rosario'
AND PD.Nro_C = C.Nro_C
);

```

o bien :

```
SELECT IT.nro_i, IT.descripcion_i  
FROM items IT  
WHERE EXISTS (  
SELECT *  
FROM clientes CL, pedidos PE  
WHERE CL.nro_c = PE.nro_c  
AND IT.nro_i = PE.nro_i  
AND CL.ciud_c = 'rosario'  
);
```

f) Listar los pedidos en los cuales un cliente de rosarios solicita artículos fabricados en Mendoza.

```
SELECT PR.nom_p, CL.nom_c, IT.descripcion_i, PE.cantidad, PE.precio  
FROM proveedores AS PR, clientes AS CL, items AS IT, pedidos AS PE  
WHERE PR.nro_p = PE.nro_p  
AND CL.nro_c = PE.nro_c  
AND IT.nro_i = PE.nro_i  
AND CL.ciud_c = 'rosario'  
AND IT.ciud_i = 'mendoza';
```

g) Listar los pedidos en los que el cliente 23 solicita ítems no solicitados por el cliente 30

```
SELECT PR1.nom_p, CL1.nom_c, IT1.descripcion_i, PE1.cantidad, PE1.precio  
FROM proveedores PR1, clientes CL1, items IT1, pedidos PE1  
WHERE PR1.nro_p = PE1.nro_p  
AND CL1.nro_c = PE1.nro_c  
AND IT1.nro_i = PE1.nro_i  
AND PE1.nro_c = 23  
AND PE1.nro_i not in (
```

```

SELECT IT2.nro_i

FROM items IT2, pedidos PE2

WHERE IT2.nro_i = PE2.nro_i

AND PE2.nro_c = 30

);

```

h) Listar las ciudades en la forma (ciudad1, ciudad2) tales que un proveedor en la ciudad1 provea ítems solicitados por clientes en ciudad2

```

SELECT DISTINCT P.Ciud_P AS Ciudad1, C.Ciud_C AS Ciudad2

FROM PROVEEDORES AS P, CLIENTES AS C, PEDIDOS AS PD

WHERE P.Nro_P = PD.Nro_P

AND C.Nro_C = PD.Nro_C;

```

i) Listar los números de proveedores cuya categoría sea mayor que la de todos los proveedores que proveen el ítem "cuaderno"

```

SELECT P.Nro_P, P.Nom_P

FROM PROVEEDORES AS P

WHERE P.Categoria > (

SELECT MAX(P.Categoria)

FROM PROVEEDORES AS P, PEDIDOS as PD, ITEMS AS I

WHERE I.Descripcion_I = 'Cuaderno'

AND PD.Nro_I = I.Nro_I

AND P.Nro_P = PD.Nro_P

);

```

j) Listar los clientes que han pedido dos o más ítems distintos.

```

SELECT DISTINCT PD1.Nro_C, C.Nom_C

FROM PEDIDOS AS PD1, PEDIDOS AS PD2, CLIENTES AS C

WHERE PD1.Nro_C = PD2.Nro_C

AND PD1.Nro_I <> PD2.Nro_I

```

AND C.Nro_C = PD1.Nro_C;

k) Listar los proveedores que proveen a todos los clientes de Córdoba una cantidad mayor que el promedio de las cantidades pedidas por los clientes de Rosario.

SELECT P.Nro_P, P.Nom_P

FROM PROVEEDORES AS P, PEDIDOS AS PD, CLIENTES AS C

WHERE C.Ciud_C = 'Cordoba'

AND PD.Nro_C = C.Nro_C

AND PD.Cantidad > (

SELECT AVG(SUMA)

FROM TEMP

)

AND P.Nro_P = PD.Nro_P

GROUP BY P.Nro_P, P.Nom_P

HAVING COUNT(C.Nro_C) = (

SELECT COUNT(C2.Nro_C)

FROM CLIENTES C2 , PEDIDOS PD2

WHERE C2.Ciud_C = 'Cordoba' AND

C2.Nro_C = PD2.Nro_C AND

PD2.Nro_P = P.Nro_P

);

TABLA TEMP

SELECT C2.Nro_C, SUM(PD2.Cantidad) AS Suma

FROM PEDIDOS AS PD2, CLIENTES AS C2

WHERE PD2.Nro_C = C2.Nro_C AND

C2.Ciud_C = 'Rosario'

GROUP BY C2.Nro_C;

Ej 2.- Una oficina gubernamental desea construir un complejo habitacional, para lo cual elaboro la siguiente Base de Datos:

TRAMO (c_ciudadA, c_ciudadB, Distancia)

CIUDADES (c_ciudad, nombre, cant_escuelas, cant_fabricas)

Para decidir donde instalarlo, desea conocer los siguientes datos:

a) Las ciudades alcanzables desde la ciudad con mayor cantidad de fabricas, recorriendo no más de dos tramos, c/u de los cuales no puede tener mas de 10km de longitud.

SELECT CB.Nombre

FROM CIUDADES CA, TRAMO T, CIUDADES CB

WHERE CA.Cant_Fabricas = (

SELECT MAX(Cant_Fabricas)

FROM CIUDADES

)

AND T.Distancia <= 10

AND T.C_CiudadA = CA.C_Ciudad

AND CB.C_Ciudad = T.C_CiudadB

AND CB.C_Ciudad <> CA.C_Ciudad

UNION

SELECT CB.Nombre

FROM CIUDADES CA, TRAMO T1, TRAMO T2, CIUDADES CB

WHERE CA.Cant_Fabricas = (

SELECT MAX(Cant_Fabricas)

FROM CIUDADES

)

AND T1.Distancia <= 10

AND T2.Distancia <= 10

AND T1.C_CiudadA = CA.C_Ciudad

AND T1.C_CiudadB = T2.C_CiudadA

AND CB.C_Ciudad = T2.C_CiudadB

AND CB.C_Ciudad <> CA.C_Ciudad;

b) Las ciudades con mas de 10 fabricas que estén conectadas con todas las demás en forma directa, siempre que ningún tramo supere los 50km.

SELECT DISTINCT C_Ciudad, C.Nombre

FROM CIUDADES AS C, TRAMO AS T

WHERE C.Cant_Fabricas > 10 AND

C.C_Ciudad = T.C_CiudadA AND

T.Distance < 50

GROUP BY C.C_CIUDAD, C.Nombre

HAVING COUNT(C.C_Ciudad) = (

SELECT COUNT (C1.C_CIUDAD) – 1

FROM CIUDADES C1

);

c) Los pares de ciudades, de la forma (ciudad1, ciudad2), que son alcanzables a través de, a lo sumo, otra ciudad, llámese a esta, ciudad3 (esto es, encontrar los caminos ciudad1–ciudad3–ciudad2).

SELECT CA.Nombre, CB.Nombre

FROM CIUDADES AS CA, TRAMO AS T1, TRAMO AS T2, CIUDADES AS CB

WHERE T1.C_CiudadA = CA.C_Ciudad

AND T1.C_CiudadB = T2.C_CiudadA

AND CB.C_Ciudad = T2.C_CiudadB

AND CB.C_Ciudad <> CA.C_Ciudad;

Ej 3.– Dada la BD:

FRECUENTA (nombre–pers, nombre–bar)

SIRVE (nombre–bar, nombre–cerveza)

GUSTA (nombre–pers, nombre–cerveza)

- Encontrar las personas que frecuentan un bar que sirven una cerveza que les gusta.

```
SELECT DISTINCT F.Nombre_Pers
FROM FRECUENTA AS F, SIRVE AS S, GUSTA AS G
WHERE S.Nombre_Cerveza = G.Nombre_Cerveza
AND F.Nombre_Bar = S.Nombre_Bar
AND F.Nombre_Pers = G.Nombre_Pers;
```

- Encontrar a las personas que beben en el mismo bar que las personas a las que les gusta la cerveza Quilmes

```
SELECT DISTINCT FAux.Nombre_Pers
FROM FRECUENTA AS F, SIRVE AS S, GUSTA AS G, Frecuenta AS FAux
WHERE G.Nombre_Cerveza = 'Quilmes'
AND S.Nombre_Cerveza = G.Nombre_Cerveza
AND F.Nombre_Bar = S.Nombre_Bar
AND F.Nombre_Pers = G.Nombre_Pers
AND FAux.Nombre_Pers <> F.Nombre_Pers
AND FAux.Nombre_Bar = F.Nombre_Bar;
```

- Encontrar las personas que beben en el mismo bar que aquellas a las que le gusta una marca de cerveza que sirva dicho bar y que le guste a Juan Pérez.

```
SELECT DISTINCT FAux.Nombre_Pers
FROM FRECUENTA AS F, SIRVE AS S, GUSTA AS G, Frecuenta AS FAux
WHERE G.Nombre_Cerveza IN (
SELECT Nombre_Cerveza
FROM GUSTA
WHERE Nombre_Pers = 'Juan Perez'
)
AND S.Nombre_Cerveza = G.Nombre_Cerveza
AND F.Nombre_Bar = S.Nombre_Bar
AND F.Nombre_Pers = G.Nombre_Pers
```

AND FAux.Nombre_Pers <> F.Nombre_Pers

AND FAux.Nombre_Bar = F.Nombre_Bar;

- Encontrar las personas que frecuentan solamente bares que sirven alguna cerveza que les gusta (asumir que cada persona frecuenta al menos un bar y le gusta al menos una cerveza)

SELECT DISTINCT G.Nombre_Pers

FROM FRECUENTA AS F, SIRVE AS S, GUSTA AS G

WHERE F.Nombre_Pers = G.Nombre_Pers AND

F.Nombre_Bar = S.Nombre_Bar AND

G.Nombre_Cerveza = S.Nombre_Cerveza

GROUP BY G.Nombre_Pers

**HAVING COUNT(F.Nombre_Bar) = (SELECT COUNT(F1.Nombre_Bar) FROM FRECUENTA F1
WHERE G.Nombre_Pers = F1.Nombre_Pers);**

- Encontrar las personas que no frecuentan ningún bar que sirven cerveza que les gusta.

SELECT DISTINCT F.Nombre_Pers

FROM FRECUENTA AS F

WHERE NOT EXISTS(

SELECT FAux.Nombre_Pers

FROM FRECUENTA AS FAux, SIRVE AS S, GUSTA AS G

WHERE S.Nombre_Cerveza = G.Nombre_Cerveza

AND FAux.Nombre_Bar = S.Nombre_Bar

AND FAux.Nombre_Pers = G.Nombre_Pers

);

Ej 4.- Sea la BD:

PERSONAS (tipo_doc, num_doc, nomyap, dir, tel, fnac, sexo)

PROGENITOR (tipo_doc, num_doc, tipo_doc_hijo, num_doc_hijo)

- Listar para cada Juan Pérez los tipo y numero de documento, nombre y apellido y teléfono de todos sus hijos.

SELECT DISTINCT PH.*

FROM PERSONAS AS PH, PERSONAS AS PP, PROGENITOR AS P

WHERE PP.NomYAp = 'Juan Perez'

AND P.Tipo_Doc = PP.Tipo_Doc

AND P.Nro_Doc = PP.Num_Doc

AND PH.Tipo_Doc = P.Tipo_Doc_Hijo

AND PH.Num_Doc = P.Nro_Doc_Hijo;

- Idem a, de:
- todos sus hermanos (los hijos de su madre y/o su padre)

SELECT DISTINCT PH.*

FROM PERSONAS AS PH, PERSONAS AS PP, PROGENITOR AS P1, PROGENITOR AS P2

WHERE PP.NomYAp = 'Juan Perez'

AND P1.Tipo_Doc_Hijo = PP.Tipo_Doc

AND P1.Nro_Doc_Hijo = PP.Num_Doc

AND P2.Tipo_Doc = P1.Tipo_Doc

AND P2.Nro_Doc = P1.Nro_Doc

AND PH.Tipo_Doc = P2.Tipo_Doc_Hijo

AND PH.Num_Doc = P2.Nro_Doc_Hijo

AND PH.NomYAp <> PP.NomYAp;

- su madre

SELECT DISTINCT PM.*

FROM PERSONAS AS PM, PERSONAS AS PP, PROGENITOR AS P

WHERE PP.NomYAp = 'Juan Perez'

AND P.Tipo_Doc_Hijo = PP.Tipo_Doc

AND P.Nro_Doc_Hijo = PP.Num_Doc

AND PM.Tipo_Doc = P.Tipo_Doc

AND PM.Num_Doc = P.Nro_Doc

AND PM.Sexo = 'F';

- su abuelo materno

SELECT DISTINCT PA.*

**FROM PERSONAS AS PM, PERSONAS AS PP, PERSONAS AS PA, PROGENITOR AS P1,
PROGENITOR AS P2**

WHERE PP.NomYAp = 'Juan Perez'

AND P1.Tipo_Doc_Hijo = PP.Tipo_Doc

AND P1.Nro_Doc_HIjo = PP.Num_Doc

AND PM.Tipo_Doc = P1.Tipo_Doc

AND PM.Num_Doc = P1.Nro_Doc

AND PM.Sexo = 'F'

AND P2.Tipo_Doc_Hijo = PM.Tipo_Doc

AND P2.Nro_Doc_HIjo = PM.Num_Doc

AND PA.Tipo_Doc = P2.Tipo_Doc

AND PA.Num_Doc = P2.Nro_Doc

AND PA.Sexo = 'M';

- todos sus nietos

SELECT DISTINCT PN.*

**FROM PERSONAS AS PH, PERSONAS AS PP, PERSONAS AS PN, PROGENITOR AS P1,
PROGENITOR AS P2**

WHERE PP.NomYAp = 'Juan Perez'

AND P1.Tipo_Doc = PP.Tipo_Doc

AND P1.Nro_Doc = PP.Num_Doc

AND PH.Tipo_Doc = P1.Tipo_Doc_Hijo

AND PH.Num_Doc = P1.Nro_Doc_Hijo

AND P2.Tipo_Doc = PH.Tipo_Doc

AND P2.Nro_Doc = PH.Num_Doc

AND PN.Tipo_Doc = P2.Tipo_Doc_Hijo

AND PN.Num_Doc = P2.Nro_Doc_Hijo;

Diego Balseiro Bases de Datos

Sebastián Bromberg

Martín Mirad

Página 9