

EJERCICIOS RESUELTOS

Presentamos a continuación una serie de ejercicios de consulta sobre la base de datos formada por las tablas de PROVEEDORES, COMPONENTES, ARTICULOS y ENVIOS. En dicha base de datos está almacenada la siguiente información:

PROVEEDORES

P#	PNOMBRE	CATEGORIA	CIUDAD
P1	CARLOS	20	SEVILLA
P2	JUAN	10	MADRID
P3	JOSE	30	SEVILLA
P4	INMA	20	SEVILLA
P5	EVA	30	CACERES

ENVIOS

P#	C#	T#	CANTIDAD
P1	C1	T1	200
P1	C1	T4	700
P2	C3	T1	400
P2	C3	T2	200
P2	C3	T3	200
P2	C3	T4	500
P2	C3	T5	600
P2	C3	T6	400
P2	C3	T7	800
P2	C5	T2	100
P3	C3	T1	200
P3	C4	T2	500
P4	C6	T3	300
P4	C6	T7	300
P5	C2	T2	200
P5	C2	T4	100
P5	C5	T4	500
P5	C5	T7	100
P5	C6	T2	200
P5	C1	T4	100
P5	C3	T4	200
P5	C4	T4	800
P5	C5	T5	400
P5	C6	T4	500

COMPONENTES

C#	CNOMBRE	COLOR	PESO	CIUDAD
C1	X3A	ROJO	12	SEVILLA
C2	B85	VERDE	17	MADRID
C3	C4B	AZUL	17	MALAGA
C4	C4B	ROJO	14	SEVILLA
C5	VT8	AZUL	12	MADRID
C6	C30	ROJO	19	SEVILLA

ARTICULOS

T#	TNOMBRE	CIUDAD
T1	CLASIFICADORA	MADRID
T2	PERFORADORA	MALAGA
T3	LECTORA	CACERES
T4	CONSOLA	CACERES
T5	MEZCLADORA	SEVILLA
T6	TERMINAL	BARCELONA
T7	CINTA	SEVILLA

PROVEEDORES .- Representa los datos de proveedores de componentes para la fabricación de artículos y su ciudad de residencia.

COMPONENTES.- Indica la información de piezas utilizadas en la fabricación de diferentes artículos, indicándose el lugar de fabricación de dichos componentes.

ARTICULOS.- Información sobre los diferentes artículos que se fabrican y el lugar de montaje del mismo.

ENVIOS.- Suministros realizados por los diferentes proveedores de determinadas cantidades de componentes asignadas para la elaboración del artículo correspondiente.

Para resolver cada consulta, debes seguir los siguientes pasos:

- Lee atentamente el enunciado, hasta que hayas creído entender qué se pide.

- b) Obtén manualmente los resultados para los datos de ejemplo.
- c) Intenta representar la consulta mediante una expresión algebraica relacional. (Pueden existir numerosas expresiones algebraicas relacionales equivalentes que respondan a la misma consulta).
- c) Expresa la consulta mediante una sentencia SELECT de SQL.
- d) Pueden existir distintas formas de escribir una consulta SQL que resuelva el apartado anterior. ¿Se te ocurre alguna otra forma de expresión de la sentencia en SQL para representar la consulta?

No te fíes de los datos particulares del ejemplo; puede que obtengas el mismo resultado que la solución correcta, pero que sea errónea tu consulta para otros datos diferentes. Tu solución debe trabajar correctamente para cualquier relación consistente que pueda almacenar nuestra base de datos.

Para cada ejercicio se aporta una posible solución pero es posible que existan varias que sean también correctas.

EJERCICIOS

1. Obtener todos los detalles de todos los artículos de *CACERES*.
2. Obtener todos los valores de *P#* para los proveedores que abastecen el artículo *TI*.
3. Obtener la lista de pares de atributos (*COLOR*, *CIUDAD*) de la tabla componentes eliminando los pares duplicados.
4. Obtener de la tabla de artículos los valores de *T#* y *CIUDAD* donde el nombre de la ciudad acaba en D o contiene al menos una E.
5. Obtener los valores de *P#* para los proveedores que suministran para el artículo *TI* el componente *CI*.
6. Obtener los valores de *TNOMBRE* en orden alfabético para los artículos abastecidos por el proveedor *PI*.
7. Obtener los valores de *C#* para los componentes suministrados para cualquier artículo de *MADRID*.
8. Obtener todos los valores de *C#* de los componentes tales que ningún otro componente tenga un valor de peso inferior.
9. Obtener los valores de *P#* para los proveedores que suministren los artículos *TI* y *T2*.
10. Obtener los valores de *P#* para los proveedores que suministran para un artículo de *SEVILLA* o *MADRID* un componente *ROJO*.
11. Obtener, mediante subconsultas, los valores de *C#* para los componentes suministrados para algún artículo de *SEVILLA* por un proveedor de *SEVILLA*.
12. Obtener los valores de *T#* para los artículos que usan al menos un componente que se puede obtener con el proveedor *PI*.
13. Obtener todas las ternas (*CIUDAD*, *C#*, *CIUDAD*) tales que un proveedor de la primera ciudad suministre el componente especificado para un artículo montado en la segunda ciudad.
14. Repetir el ejercicio anterior pero sin recuperar las ternas en los que los dos valores de ciudad sean los mismos.

15. Obtener el número de suministros, el de artículos distintos suministrados y la cantidad total de artículos suministrados por el proveedor *P2*.
16. Para cada artículo y componente suministrado obtener los valores de *C#*, *T#* y la cantidad total correspondiente.
17. Obtener los valores de *T#* de los artículos abastecidos al menos por un proveedor que no viva en MADRID y que no esté en la misma ciudad en la que se monta el artículo.
18. Obtener los valores de *P#* para los proveedores que suministran al menos un componente suministrado al menos por un proveedor que suministra al menos un componente *ROJO*.
19. Obtener los identificadores de artículos, *T#*, para los que se ha suministrado algún componente del que se haya suministrado una media superior a 320 artículos.
20. Seleccionar los identificadores de proveedores que hayan realizado algún envío con Cantidad mayor que la media de los envíos realizados para el componente a que corresponda dicho envío.
21. Seleccionar los identificadores de componentes suministrados para el artículo 'T2' por el proveedor 'P2'.
22. Seleccionar todos los datos de los envíos realizados de componentes cuyo color no sea 'ROJO'.
23. Seleccionar los identificadores de componentes que se suministren para los artículos 'T1' y 'T2'.
24. Seleccionar el identificador de proveedor y el número de envíos de componentes de color 'ROJO' llevados a cabo por cada proveedor.
25. Seleccionar los colores de componentes suministrados por el proveedor 'P1'.
26. Seleccionar los datos de envío y nombre de ciudad de aquellos envíos que cumplan que el artículo, proveedor y componente son de la misma ciudad.
27. Seleccionar los nombres de los componentes que son suministrados en una cantidad total superior a 500.
28. Seleccionar los identificadores de proveedores que residan en Sevilla y no suministren más de dos artículos distintos.
29. Seleccionar los identificadores de artículos para los cuales todos sus componentes se fabrican en una misma ciudad.
30. Seleccionar los identificadores de artículos para los que se provean envíos de todos los componentes existentes en la base de datos.
31. Seleccionar los códigos de proveedor y artículo que suministran al menos dos componentes de color 'ROJO'.
32. Propón tu mismo consultas que puedan realizarse sobre esta base de datos de ejemplo. Intenta responderla, y si te parece un problema interesante o no estás seguro de su solución, puedes exponerlos en la clases prácticas para su resolución en grupo.

SOLUCIONES A LOS EJERCICIOS

1. Obtener todos los detalles de todos los artículos de *CACERES*.

$\sigma_{\text{ciudad}='CACERES'}(\text{Artículos})$

SELECT *
FROM artículos
WHERE ciudad = 'CACERES';

T#	TNOMBRE	CIUDAD
T3	LECTORA	CACERES
T4	CONSOLA	CACERES

2. Obtener todos los valores de *P#* para los proveedores que abastecen el artículo *T1*.

$\pi_{p\#}(\sigma_{t\#='T1'}(\text{Envíos}))$

SELECT DISTINCT p#
FROM envíos
WHERE t#='T1';

P#
P1
P2
P3

3. Obtener la lista de pares de atributos (*COLOR*, *CIUDAD*) de la tabla *componentes* eliminando los pares duplicados.

$\pi_{\text{color, ciudad}}(\text{Componentes})$

SELECT DISTINCT color, ciudad
FROM componentes;

COLOR	CIUDAD
AZUL	MADRID
AZUL	MALAGA
ROJO	SEVILLA

4. Obtener de la tabla de artículos los valores de *T#* y *CIUDAD* donde el nombre de la ciudad acaba en *D* o contiene al menos una *E*.

$\pi_{t\#, ciudad}(\sigma_{(\text{ciudad LIKE '%D'}) \text{ OR } (\text{ciudad LIKE '%E\%'})}(\text{Artículos}))$

SELECT t#, ciudad
FROM artículos
WHERE ciudad LIKE '%D' OR
ciudad LIKE '%E%';

T#	CIUDAD
T1	MADRID
T3	CACERES
T4	CACERES
T5	SEVILLA
T6	BARCELONA

5. Obtener los valores de *P#* para los proveedores que suministran para el artículo *T1* el componente *C1*.

$\pi_{p\#}(\sigma_{(t\#='T1') \text{ AND } (c\#='C1')}(\text{Envíos}))$

SELECT p#
FROM envíos
WHERE t#='T1' AND c#='C1';

P#
P1

6. Obtener los valores de *TNOMBRE* en orden alfabético para los artículos abastecidos por el proveedor *P1*.

$$\pi_{\text{tnombre}}(\text{Artículos} * (\sigma_{\text{p\#}='P1'}(\text{Envíos})))^1$$

```
SELECT tnombre
FROM artículos a, envios e
WHERE e.p\#='P1' AND e.t\#=a.t\#
ORDER BY tnombre;
```

TNOMBRE
CLASIFICADORA
CONSOLA

```
SELECT tnombre
FROM artículos NATURAL JOIN envios
WHERE p\#='P1'
ORDER BY nombre;
```

7. Obtener los valores de *C#* para los componentes suministrados para cualquier artículo de *MADRID*.

$$\pi_{\text{c\#}}(\text{Envíos} * (\sigma_{\text{Ciudad}='MADRID'}(\text{Artículos})))$$

```
SELECT DISTINCT c\#
FROM envios
WHERE t\# IN ( SELECT t\#
FROM artículos
WHERE ciudad='MADRID');
```

C#
C1
C3

8. Obtener todos los valores de *C#* de los componentes tales que ningún otro componente tenga un valor de peso inferior.

$$\pi_{\text{c\#}}(\text{Componentes} * \text{peso}=\text{p} (\sum() \oplus \text{MIN}(\text{Peso}) \text{P}(\text{Componentes})))$$

```
SELECT c\#
FROM componentes
WHERE peso = (SELECT MIN(peso)
FROM componentes );
```

C#
C1
C5

9. Obtener los valores de *P#* para los proveedores que suministren los artículos *T1* y *T2*.

$$(\pi_{\text{p\#}}(\sigma_{\text{t\#}='T1'}(\text{Envíos}))) \cap (\pi_{\text{p\#}}(\sigma_{\text{t\#}='T2'}(\text{Envíos})))$$

```
SELECT p\#
FROM envios
WHERE t\#='T1'
INTERSECT
SELECT p\#
FROM envios
WHERE t\#='T2';
```

P#
P2
P3

¹ Por comodidad usaremos como simbolo del join *, si es un join externo por la izquierda /*, si es un join externo por la derecha */ y si es un join externo completo */*

10. Obtener los valores de P# para los proveedores que suministran para un artículo de SEVILLA o MADRID un componente ROJO.

$$\pi_{p\#} ((\sigma_{\text{Color}='ROJO'}(\text{Componentes}) * (\text{Envíos} * \pi_{t\#} ((\sigma_{(\text{ciudad}='MADRID' \text{ OR } \text{Ciudad}='SEVILLA')} (\text{Artículos}))))))$$

```
SELECT p#
FROM envios e, componentes c, artículos a
WHERE e.c#=c.c# AND e.t#=a.t# AND
      c.color='ROJO' AND
      a.ciudad IN('SEVILLA', 'MADRID');
```

P#
P1
P4

```
SELECT p#
FROM componentes c JOIN2 (envios e
      NATURAL JOIN artículos a) ON c.c#=e.c#
WHERE c.color='ROJO' AND
      a.ciudad IN('SEVILLA', 'MADRID');
```

11. Obtener, mediante subconsultas, los valores de C# para los componentes suministrados para algún artículo de SEVILLA por un proveedor de SEVILLA.

$$\pi_{c\#} ((\sigma_{\text{Ciudad}='SEVILLA'}(\text{Proveedores}) * \pi_{c\#,p\#} (\text{Envíos} * (\sigma_{\text{Ciudad}='SEVILLA'}(\text{Artículos})))))$$

```
SELECT c#
FROM envios
WHERE t# IN ( SELECT t#
              FROM artículos
              WHERE ciudad='SEVILLA')
AND p# IN ( SELECT p#
            FROM proveedores
            WHERE ciudad='SEVILLA');
```

C#
C6

12. Obtener los valores de T# para los artículos que usan al menos un componente que se puede obtener con el proveedor P1.

$$\pi_{t\#} (\text{Envíos} * \pi_{c\#} (\sigma_{p\#='P1'}(\text{Envíos})))$$

```
SELECT DISTINCT t#
FROM envios
WHERE c# IN ( SELECT DISTINCT c#
              FROM envios
              WHERE p#='P1');
```

T#
T1
T4

13. Obtener todas las ternas (CIUDAD, C#, CIUDAD) tales que un proveedor de la primera ciudad suministre el componente especificado para un artículo montado en la segunda ciudad.

² Observese que se ha hecho un JOIN en lugar de un NATURAL JOIN para que se realice sólo por el atributo c# y evitar que use para el join el atributo ciudad

Articulo_ciudad (p#, c#, ciudad) $\leftarrow \pi_{p\#,c\#,ciudad}$ (Envíos * Artículos)
 $\pi_{ciudad, c\#, ciudad}$ (Proveedores * Articulo_ciudad)

*SELECT p.ciudad ,e.c#, a.ciudad
 FROM envios e, proveedores p , artículos a
 WHERE e.p#=p.p# AND e.t#=a.t#;*

*SELECT p.ciudad ,e.c#, a.ciudad
 FROM proveedores p JOIN (envios e NATURAL
 JOIN artículos a) ON p.p#=e.p#;*

CIUDAD	C#	CIUDAD
SEVILLA	C1	MADRID
SEVILLA	C1	CACERES
MADRID	C3	MADRID
MADRID	C3	MALAGA
MADRID	C3	CACERES
MADRID	C3	CACERES
MADRID	C3	SEVILLA
MADRID	C3	BARCELONA
MADRID	C3	SEVILLA
MADRID	C5	MALAGA
SEVILLA	C3	MADRID
SEVILLA	C4	MALAGA
SEVILLA	C6	CACERES
SEVILLA	C6	SEVILLA
CACERES	C2	MALAGA
CACERES	C2	CACERES
CACERES	C5	CACERES
CACERES	C5	SEVILLA
CACERES	C6	MALAGA
CACERES	C1	CACERES
CACERES	C3	CACERES
CACERES	C4	CACERES
CACERES	C5	SEVILLA

14. Repetir el ejercicio anterior pero sin recuperar las ternas en los que los dos valores de ciudad sean los mismos.

Articulo_ciudad (p#, c#, ciudad) $\leftarrow \pi_{p\#,c\#,ciudad}$ (Envíos * Artículos)
 $\pi_{ciudad, c\#, ciudad}$ ($\sigma_{ciudad \neq ciudad}$ (Proveedores * Articulo_ciudad))

*SELECT p.ciudad ,c#, a.ciudad
 FROM envios e, proveedores p , artículos a
 WHERE e.p#=p.p# AND
 e.t#=a.t# AND
 p.ciudad <> a.ciudad;*

*SELECT p.ciudad ,e.c#, a.ciudad
 FROM proveedores p JOIN (envios e NATURAL JOIN
 artículos a) ON p.p#=e.p# and p.ciudad <>a.ciudad;*

CIUDAD	C#	CIUDAD
SEVILLA	C1	MADRID
SEVILLA	C1	CACERES
MADRID	C3	MALAGA
MADRID	C3	CACERES
MADRID	C3	CACERES
MADRID	C3	SEVILLA
MADRID	C3	BARCELONA
MADRID	C3	SEVILLA
MADRID	C5	MALAGA
SEVILLA	C3	MADRID
SEVILLA	C4	MALAGA
SEVILLA	C6	CACERES
CACERES	C2	MALAGA
CACERES	C5	SEVILLA
CACERES	C6	MALAGA

15. Obtener el número de suministros, el de artículos distintos suministrados y la cantidad total de artículos suministrados por el proveedor P2.

$(\sum()) \oplus \text{Count}(*)$ Número de Suministros $\oplus \text{Count}(\text{Distinct } t\#)$ Artículos Suministrados
 $\oplus \text{Sum}(\text{Cantidad})$ Total Artículos Suministrados ($\sigma_{p\#='P2'}$ (Envíos))

```

SELECT COUNT(*) Número de Suministros,
COUNT (DISTINCT t#) Artículos Suministrados,
SUM(cantidad) Total Artículos Suministrados
FROM envios
WHERE p#='P2';
  
```

Número de Suministros	Artículos Suministrados	Total Artículos Suministrados
8	7	3200

16. Para cada artículo y componente suministrado obtener los valores de C#, T# y la cantidad total correspondiente.

$\sum(c\#, t\#) \oplus \text{Sum}(\text{Cantidad})$ Total (Envíos)

```

SELECT c#,t#, SUM(cantidad) Total
FROM envios
GROUP BY c#,t#;
  
```

C#	T#	Total
C1	T1	200
C1	T4	800
C2	T2	200
C2	T4	100
C3	T1	600
C3	T2	200
C3	T3	200
C3	T4	700
C3	T5	600
C3	T6	400
C3	T7	800
C4	T2	500
C4	T4	800
C5	T2	100
C5	T4	500
C5	T5	400
C5	T7	100
C6	T2	200
C6	T3	300
C6	T4	500

17. Obtener los valores de T# de los artículos abastecidos al menos por un proveedor que no viva en MADRID y que no esté en la misma ciudad en la que se monta el artículo.

$\text{Prov_no_Madrid}(p\#, \text{ciudadp}) \leftarrow \pi_{p\#, \text{ciudad}} (\sigma_{\text{ciudad} \neq \text{'MADRID'}} (\text{Proveedores}))$
 $\pi_{t\#} (\sigma_{(\text{ciudad} \neq \text{ciudadp})} (\text{Artículos} * (\text{Envíos} * \text{Prov_no_Madrid})))$

```

SELECT DISTINCT e.t#
FROM envios e,articulos a
WHERE e.t# = a.t# AND EXISTS
(SELECT *
FROM proveedores p
WHERE p.ciudad!= a.ciudad AND
p.p# = e.p# AND
p.ciudad!= 'MADRID' );
  
```

T#
T1
T2
T3
T4
T5
T7

18. Obtener los valores de $P\#$ para los proveedores que suministran al menos un componente suministrado al menos por un proveedor que suministra al menos un componente **ROJO**.

$$\pi_{p\#}(\text{Envios} * (\pi_{c\#}(\text{Envios} * (\pi_{p\#}(\text{Envios} * (\sigma_{\text{Color}='ROJO'}(\text{Componentes})))))))$$

```
SELECT distinct p#
FROM envios
WHERE c# IN
```

```
(SELECT c#
FROM envios
WHERE p# IN
( SELECT p#
FROM envios NATURAL JOIN componentes
WHERE color='ROJO' ));
```

P#
P1
P2
P3
P4
P5

19. Obtener los identificadores de artículos, $T\#$, para los que se ha suministrado algún componente del que se haya suministrado una media superior a 320 artículos.

$$\pi_{t\#}(\text{Envios} * (\pi_{c\#}(\sigma_{A > 320}(\sum(c\#) \oplus \text{AVG}(\text{cantidad}) A(\text{Envios}))))))$$

```
SELECT DISTINCT t#
FROM envios
WHERE c# IN ( SELECT c#
FROM envios
GROUP BY c#
HAVING AVG(cantidad) > 320 );
```

T#
T1
T2
T3
T4
T5
T6

20. Seleccionar los identificadores de proveedores que hayan realizado algún envío con Cantidad mayor que la media de los envíos realizados para el componente a que corresponda dicho envío.

$$\pi_{p\#}(\sigma_{(\text{Cantidad} > A)}(\text{Envios} * (\sum(c\#) \oplus \text{AVG}(\text{cantidad}) A(\text{Envios}))))$$

```
SELECT distinct p#
FROM envios a
WHERE cantidad > (SELECT AVG(cantidad)
FROM envios b
WHERE b.c#=a.c#);
```

P#
P1
P2
P5

21. Seleccionar los identificadores de componentes suministrados para el artículo 'T2' por el proveedor 'P2'.

$$\pi_{c\#}(\sigma_{t\#='T2' \text{ AND } p\#='P2'}(\text{Envios}))$$

Nótese que es un error grave dar como solución la expresión:

$$\pi_{c\#}(\sigma_{t\#='T2'}(\text{Envios})) \cap \pi_{c\#}(\sigma_{p\#='P2'}(\text{Envios}))$$

```
SELECT c#
FROM envios
WHERE t# = 'T2' AND p# = 'P2';
```

C#
C3
C5

22. Seleccionar todos los datos de los envíos realizados de componentes cuyo color no sea 'ROJO'.

$$\pi_{\text{Envíos.*}}(\text{Envíos*}(\sigma_{\text{Color} \neq \text{'ROJO'}}(\text{Componentes})))$$

```
SELECT e.*
FROM envios e, componentes c
WHERE e.c# = c.c# AND color <> 'ROJO';
```

```
SELECT e.*
FROM envios e NATURAL JOIN componentes
WHERE color <> 'ROJO';
```

P#	C#	T#	CANTIDAD
P5	C2	T2	200
P5	C2	T4	100
P2	C3	T1	400
P3	C3	T1	200
P2	C3	T2	200
P2	C3	T3	200
P2	C3	T4	500
P5	C3	T4	200
P2	C3	T5	600
P2	C3	T6	400
P2	C3	T7	800
P2	C5	T2	100
P5	C5	T4	500
P5	C5	T5	400

23. Seleccionar los identificadores de componentes que se suministren para los artículos 'T1' y 'T2'.

$$(\pi_{c\#}(\sigma_{t\#='T1'}(\text{Envíos}))) \cap (\pi_{c\#}(\sigma_{t\#='T2'}(\text{Envíos})))$$

Nótese que es un error grave dar como solución:

$$\pi_{c\#}(\sigma_{t\#='T1' \text{ AND } t\#='T2'}(\text{Envíos}))$$

```
SELECT c#
FROM envios
WHERE t# = 'T1'
INTERSECT
SELECT c#
FROM envios
WHERE t# = 'T2';
```

C#
C3

24. Seleccionar el identificador de proveedor y el número de envíos de componentes de color 'ROJO' llevados a cabo por cada proveedor.

$$\sum_{p\#} (p\#) \oplus \text{COUNT}(*), A(\text{Envíos} * (\sigma_{\text{Color} = \text{'ROJO'}}(\text{Componentes})))$$

```
SELECT p#, count(*) A
FROM envios
WHERE c# IN (SELECT c#
FROM componentes
WHERE color = 'ROJO')
GROUP BY p#;
```

P#	A
P1	2
P3	1
P4	2
P5	4

25. Seleccionar los colores de componentes suministrados por el proveedor 'P1'.

$$\pi_{\text{color}}(\text{Componentes} * (\sigma_{p\# = \text{'P1'}}(\text{Envíos})))$$

```
SELECT DISTINCT color
FROM componentes
WHERE c# IN (SELECT DISTINCT c#
FROM envios
WHERE p# = 'P1');
```

COLOR
ROJO

26. Seleccionar los datos de envío y nombre de ciudad de aquellos envíos que cumplan que el artículo, proveedor y componente son de la misma ciudad.

$\pi_{c\#, t\#, p\#, cantidad, ciudad} (\text{Artículos} * (\text{Proveedores} * (\text{Envíos} * \text{Componentes})))$

Nota: como no se indica nada en *, el join con proveedor se realiza por p# y ciudad; el join con artículo por el t# y ciudad

```
SELECT e.*, c.ciudad
FROM envios e, componentes c,
      articulos a, proveedores p
WHERE e.t# = a.t# AND e.c# = c.c# AND
      e.p# = p.p# AND p.ciudad=c.ciudad AND
      p.ciudad = a.ciudad;
```

P#	C#	T#	CANTIDAD	C.CIUDAD
P4	C6	T7	300	Sevilla

```
SELECT e.*, .ciudad
FROM articulos NATURAL JOIN
      (proveedores NATURAL JOIN
      (envios e NATURAL JOIN componentes));
```

27. Seleccionar los nombres de los componentes que son suministrados en una cantidad total superior a 500.

$\pi_{cnombre} (\text{Componentes} * (\sigma_{(S > 500)} (\sum (c\#) \oplus \text{SUM}(cantidad) S (\text{Envíos}))))$

```
SELECT DISTINCT cnombre
FROM componentes
WHERE c# IN ( SELECT c#
              FROM envios
              GROUP BY c#
              HAVING SUM(cantidad) > 500);
```

CNOMBRE
C30
C4B
VT8
X3A

28. Seleccionar los identificadores de proveedores que residan en Sevilla y no suministren más de dos artículos distintos.

$\pi_{p\#} (\sigma_{ciudad='SEVILLA'} (\text{Proveedores})) \cap \pi_{p\#} (\sigma_{(c \leq 2)} (\sum (p\#) \oplus \text{COUNT}(\text{DISTINCT } t\#) C (\text{Envíos})))$

```
(SELECT p#
FROM proveedores
WHERE ciudad='SEVILLA')
MINUS
(SELECT p#
FROM envios
GROUP BY p#
HAVING COUNT(DISTINCT t#) > 2);
```

P#
P1
P3
P4

Puede parecer que la sentencia SQL es muy diferente de la expresión algebraica relacional. En realidad son equivalentes aplicando la siguiente propiedad del álgebra de conjuntos: $A - B = A \cap \sim B$ (donde $\sim B$ representa el complementario de B).

29. Seleccionar los identificadores de artículos para los cuales todos sus componentes se fabrican en una misma ciudad.

$$\pi_{t\#} (\sigma_{(C=1)} (\sum (t\#) \oplus \text{COUNT}(\text{DISTINCT ciudad})^C (\text{Componentes} * \text{Envíos})))$$

```
SELECT t#
FROM envios e,componentes c
WHERE e.c# = c.c#
GROUP BY t#
HAVING COUNT(DISTINCT ciudad) = 1;
```

T#
T6

```
SELECT t#
FROM envios NATURAL JOIN componentes
GROUP BY t#
HAVING COUNT(DISTINCT ciudad) = 1;
```

30. Seleccionar los identificadores de artículos para los que se provean envíos de todos los componentes existentes en la base de datos.

$$\pi_{t\#} (\text{Envíos} \div \pi_{c\#} (\text{Componentes}))$$

```
SELECT t#
FROM Envíos
GROUP BY t#
HAVING COUNT(DISTINCT c#) = (SELECT COUNT(*)
FROM Componentes);
```

T#
T4

31. Seleccionar los códigos de proveedor y artículo que suministran al menos dos componentes de color 'ROJO'.

$$\pi_{p\#,t\#} (\sigma_{(C>1)} (\sum (p\#, t\#) \oplus \text{COUNT}(*))^C (\text{Envíos} * (\sigma_{\text{color}='ROJO'} (\text{Componentes}))))$$

```
SELECT p#, t#
FROM envios e,componentes c
WHERE e.c# = c.c# AND c.color='ROJO'
GROUP BY p#, t#
HAVING COUNT (*) > 1;
```

P#	T#
P5	T4

```
SELECT p#, t#
FROM envios NATURAL JOIN componentes
WHERE .color='ROJO'
GROUP BY p#, t#
HAVING COUNT (*) > 1;
```