

ENERO 26 2022



Bases de Datos
Grado en Ingeniería Informática
Universidad de La Laguna
26 de enero de 2022

Una cadena de tiendas tiene una base de datos para gestionar los artículos que vende. En el esquema de base de datos utilizado los atributos se abrevian según el siguiente convenio:

Atributo	Significado
CA	Código de artículo: A1, A2, ...
CAT	Categoría: limpieza, higiene personal, menaje, alimentación, ...
CT	Código de Tienda: T1, T2, ...
DNI	DNI del comprador
F	Fecha de compra: '03-11-2019', '10-11-2021', ...
NU	Número de unidades: 5, 2, 1, ...
PR	Precio en Euros de un artículo

Las tablas utilizadas son:

ARTÍCULOS (CA, CAT)

SIGNIFICADO: El artículo CA es de categoría CAT.

CLAVE PRIMARIA: (CA)

TIENDAS (CT, CA, PR)

SIGNIFICADO: La tienda con código CT dispone del artículo CA a un precio de PR euros.

CLAVE PRIMARIA: (CT, CA) **CLAVE AJENA:** (CA)

VENTAS (DNI, CT, CA, NU, F)

SIGNIFICADO: La persona con dni DNI, ha comprado en la tienda CT, NU unidades del artículo CA, en la fecha F.

CLAVE PRIMARIA: (DNI, CT, CA, F) **CLAVE AJENA:** (CT, CA)

1) Responder en álgebra relacional a las siguientes consultas:

- Tiendas con artículos de las categorías C1 o C2.
- Personas que han comprado el artículo A1 y A2 en una misma tienda.
- Tienda que tiene el artículo A1 con menor precio.
- Personas que han comprado al menos un artículo en cada tienda.
- Personas que han comprado en alguna tienda todos los artículos de alguna categoría.

2) Responder en cálculo relacional de t-uplas a las siguientes consultas:

- Personas que un mismo día han comprado al menos 2 artículos distintos en una misma tienda.
- Artículos que sólo están disponibles en la tienda T1.
- Artículo más barato de la tienda T1.
- Personas que han comprado un mismo artículo en cada tienda.
- Tiendas que en un mismo día han vendido al menos una unidad de cada uno de sus artículos disponibles.

3) Responder en SQL a las siguientes consultas:

- Número de productos de la tienda T1 con un precio inferior a 30 euros.
- Tiendas que no disponen de artículos de la categoría C1.
- Personas que han comprado en una misma tienda más de 10 unidades de productos iguales o distintos.
- Tiendas que en un mismo día han vendido artículos de todas las categorías.
- Tiendas tales que al menos el 40% de sus artículos cuestan menos de 30 euros.

4) Responder en SQL a las siguientes peticiones:

- Elimina en la tabla VENTAS la clave primaria.
- Incrementa en un 10% el precio de todos los artículos de la tienda T1 que pertenecen a la categoría C1.
- Crea una tabla que almacene todas las ventas anteriores al 1 de enero de 2020 de la base de datos.
- Impide que los precios de un mismo artículo puedan variar según la tienda.
- Impide que una tienda pueda vender de un mismo producto más de 100 unidades en un mismo día.

a)

$P(CT) (S((CAT = 'C1') \vee (CAT = 'C2')) (TIENDAS * ARTICULOS))$

b)

$A=B=VENTAS$

$P(DNI) (S((A.CA = 'A1' \wedge B.CA = 'A2') \wedge (A.CT = B.CT)) (A \times B))$

c)

$A = B = P(CT, PR) (S(CA = 'A1') (TIENDAS))$

$C = P(CT) (S(A.PR \geq B.PR) (A \times B))$

$P(CT) (A - C)$

d)

$T = P(CT) (TIENDAS)$

$B = P(DNI, CT) (VENTAS)$

B/T

e)

Ejercicio 2

a)

$\text{dom}(v) = \text{dom}(v1) = \text{ventas}$

$\{t \mid \exists v (t[DNI] = v[DNI]) \wedge \exists v2 ((v[DNI] = v2[DNI]) \wedge (v[CT] = v2[CT]) \wedge (v[CA] \neq v2[CA]) \wedge (v[F] = v2[F]))\}$

b)

$\text{dom}(i) = \text{dom}(i1) = \text{TIENDAS}$

$\text{dom}(a) = \text{artículos}$

$\{t \mid \exists a (t[ca] = a[ca]) \wedge \exists i (i[ca] = a[ca] \wedge i[ct] = 't1') \wedge \neg \exists i1 (i[ct] \neq i1[ct]) \wedge (i[ca] \neq i1[ca]) \}$

c)

d)

e)

Ejercicio 3

a)

$\text{SELECT COUNT}(CA)$
 FROM TIENDAS
 $\text{WHERE PR} < 30$

b)

SELECT CT
 FROM TIENDAS
 $\text{WHERE CT NOT IN (SELECT CT}$

```
FROM TIENDAS NATURAL JOIN ARTICULOS
WHERE CAT=C1)
```

c)

```
SELECT DNI
FROM VENTAS
GROUP BY DNI, CT
HAVING SUM(NU) >= 10
```

d)

```
SELECT CT
FROM VENTAS NATURAL JOIN ARTICULOS
GROUP BY CT, F
HAVING COUNT(DISTINCT CAT) = (SELECT COUNT(DISTINCT CAT)
                                FROM ARTICULOS
                                )
```

e)

```
SELECT
```

Ejercicio 4

a)

```
ALTER TABLE VENTAS
DROP PRIMARY KEY
```

b)

```
UPDATE TIENDAS
PR *= 0.10
WHERE CT = 'T1' AND CA IN (SELECT CA
                            FROM ARTICULOS
                            WHERE CAT = 'C1')
```

c)

```
CREATE TABLE PROVE
AS (SELECT *
    FROM VENTAS
    WHERE F < 1/01/2020)
```

d)

```
ALTER TABLE TIENDAS
CHECK ((SELECT COUNT(PR)
        FROM TIENDAS
        GROUP BY CA) = 1)
```

e)

FEBRERO 3

Daniel Deniz García

Una empresa organizadora de carreras tiene una base de datos para gestionar las competiciones que ofrece. En el esquema de base de datos utilizado los atributos se abrevian según el siguiente convenio:

Atributo	Significado
CDC	Ciudad donde se realiza la carrera
D	Distancia en kilómetros de una carrera: 10.000, 21.097, ...
DNI	DNI de la persona
F	Fecha de la carrera: '15-05-2020', '04-12-2021', ...
IC	Identificador de carrera: I1, I2, ...
PR	Precio en Euros de la inscripción en una carrera
S	Sexo: Hombre o Mujer
T	Tiempo en minutos que una persona invirtió en una carrera. Nulo si no la acabó.

Las tablas utilizadas son:

No acabó

PERSONAS (DNI, S)

SIGNIFICADO: La persona con DNI es de sexo S.

CLAVE PRIMARIA: (DNI)

CARRERAS (IC, CDC, F, D, PR)

SIGNIFICADO: La prueba IC se celebra en la ciudad CDC, en la fecha F, tiene una distancia de D kilómetros y el precio de la inscripción es de PR euros.

CLAVE PRIMARIA: (IC)

PARTICIPANTES (DNI, IC, T)

SIGNIFICADO: La persona con dni DNI, participó en la carrera IC e hizo un tiempo de T minutos. Si el valor de T es nulo la persona no terminó la carrera.

CLAVE PRIMARIA: (DNI, IC) **CLAVES AJENAS:** (DNI), (IC)

1) Responder en álgebra relacional a las siguientes consultas:

- ☒ a) Personas que han corrido la carrera I1 pero no la I2.
- ☒ b) Personas que han corrido una misma distancia en varias ciudades.
- ☒ c) Persona que ganó la carrera I1.
- ☒ d) Personas que han corrido al menos una vez en cada ciudad en la que se organizan carreras.
- ☒ e) Personas que han corrido en alguna ciudad todas las carreras que se han organizado en ella.

2) Responder en cálculo relacional de t-uplas a las siguientes consultas:

- a) Personas que han corrido en al menos 2 carreras celebradas un mismo día.
- b) Personas que siempre corren la misma distancia.
- c) Personas que han ganado alguna carrera.
- d) Personas que han ganado todas las carreras en las que han participado.
- e) Personas que han ganado todas las carreras en las que han participado celebradas en una misma ciudad.

3) Responder en SQL a las siguientes consultas:

- a) Número de corredores que han terminado la competición I1.
- b) Mostrar para cada carrera el mejor tiempo realizado.
- c) Corredores que siempre que participan en una carrera la terminan.
- d) Ciudad en la que se han organizado el mayor número de carreras.
- e) Carreras tales que al menos el 30% de sus participantes son mujeres.

4) Responder en SQL a las siguientes peticiones:

- a) Crea una vista que muestre para cada corredor y cada carrera en la que haya participado el tiempo medio por km.
- b) Elimina las personas que no han participado en ninguna carrera.
- c) Impide que los precios de inscripción en cualquier carrera sean superiores a 50 euros.
- d) Fuerza a que todos los tiempos en una carrera dada sean distintos.
- e) Impide que una ciudad pueda ofrecer más de 5 carreras en un mismo año.

Ejercicio 1

a)

$A = P(\text{DNI}) \text{ S}(\text{IC} = 'I1') (\text{PARTICIPANTES})$

$B = P(\text{DNI}) \text{ S}(\text{IC} = 'I2') (\text{PARTICIPANTES})$

$A - B$

b)

$A = B = P(\text{DNI}, \text{CDC}, \text{D}) \text{ S}(\text{PARTICIPANTES} * \text{CARRERAS})$

$P(\text{DNI}) \text{ S}((A.DNI = B.DNI) \wedge (A.D = B.D) \wedge (A.CDC \neq B.CDC)) (A \times B)$

c)

$A = B = P(\text{DNI}, \text{T}) \text{ S}(\text{IC} = 'I1') (\text{PARTICIPANTES})$

$C = P(A.DNI) \text{ S}(A.T > B.T) (A \times B)$

$(P(\text{DNI}) (A)) - C$

d)

$A = P(\text{CDC}) (\text{CARRERAS})$

$B = P(\text{DNI}, \text{CDC}) (\text{CARRERAS} * \text{PARTICIPANTES})$

B/A

e)

Ejercicio 2

a)

$\text{dom}(c1) = \text{dom}(c2) = \text{carreras}$

$\text{dom}(p1) = \text{dom}(p2) = \text{participantes}$

$\{t \mid \text{Ex } p1 (t[\text{DNI}] = p1[\text{DNI}]) \wedge \text{Ex } p2 (p1[\text{DNI}] = p2[\text{DNI}]) \wedge \text{Ex } c1, c2 (p1[\text{IC}] = c1[\text{ic}] \wedge (p2[\text{ic}] = c2[\text{IC}]) \wedge (c1[\text{IC}] \neq c2[\text{IC}]))\}$

b)

c)

d)

e)

Ejercicio 3

a)

```
SELECT COUNT(DNI)
FROM PARTICIPANTES
WHERE IC = '11' AND T IS NOT NULL
```

b)

```
SELECT IC, MIN(T)
FROM PARTICIPANTES
GROUP BY IC
```

c)

```
SELECT UNIQUE DNI
FROM PARTICIPANTES
WHERE T IN (SELECT T
            FROM PARTICIPANTES
            WHERE T IS NOT NULL)
```

d)

```
SELECT CDC
FROM CARRERAS
GROUP BY CDC
HAVING MAX(COUNT(IC))
```

e)

```
SELECT IC
FROM CARRERAS PARTICIPANTES
GROUP BY IC
HAVING COUNT(DNI) * 0.3 >= (SELECT COUNT(DNI)
                           FROM PARTICIPANTES NATURAL JOIN PERSONAS
                           WHERE S = 'MUJER' )
```

Ejercicio 4

a)

```
CREATE VIEW V1  
AS (SELECT DNI, IC, D/T  
    FROM PARTICIPANTES NATURAL JOIN CARRERAS  
    )
```

b)

```
DELETE FROM PARTICIPANTES  
WHERE DNI IN(SELECT DNI  
    FROM PARTICIPANTES  
    WHERE T IS NULL)
```

c)

```
ALTER TABLE CARRERAS  
ADD CONSTRAINT C1  
CHECK PR <= 50
```

d)

e)

```
ALTER TABLE CARRERAS  
ADD CONSTRAINT C1  
CHECK NOT EXIST (SELECT *  
    FROM CARRERAS  
    GROUP BY CDC, YEAR(F)  
    HAVING COUNT(IC) > 5)
```




Universidad
de La Laguna

Bases de Datos
Grado en Ingeniería Informática
Universidad de La Laguna
26 de enero de 2022

Una cadena de tiendas tiene una base de datos para gestionar los artículos que vende. En el esquema de base de datos utilizado los atributos se abrevian según el siguiente convenio:

Atributo	Significado
CA	Código de artículo: A1, A2, ...
CAT	Categoría: limpieza, higiene personal, menaje, alimentación, ...
CT	Código de Tienda: T1, T2, ...
DNI	DNI del comprador
F	Fecha de compra: '03-11-2019', '10-11-2021', ...
NU	Número de unidades: 5, 2, 1, ...
PR	Precio en Euros de un artículo

Las tablas utilizadas son:

ARTÍCULOS (CA, CAT)

SIGNIFICADO: El artículo CA es de categoría CAT.

CLAVE PRIMARIA: (CA)

TIENDAS (CT, CA, PR)

SIGNIFICADO: La tienda con código CT dispone del artículo CA a un precio de PR euros.

CLAVE PRIMARIA: (CT, CA) **CLAVE AJENA:** (CA)

VENTAS (DNI, CT, CA, NU, F)

SIGNIFICADO: La persona con dni DNI, ha comprado en la tienda CT, NU unidades del artículo CA, en la fecha F.

CLAVE PRIMARIA: (DNI, CT, CA, F) **CLAVE AJENA:** (CT, CA)

1) Responder en álgebra relacional a las siguientes consultas:

- Tiendas con artículos de las categorías C1 o C2.
- Personas que han comprado el artículo A1 y A2 en una misma tienda.
- Tienda que tiene el artículo A1 con menor precio.
- Personas que han comprado al menos un artículo en cada tienda.
- Personas que han comprado en alguna tienda todos los artículos de alguna categoría.

2) Responder en cálculo relacional de t-uplas a las siguientes consultas:

- Personas que un mismo día han comprado al menos 2 artículos distintos en una misma tienda.
- Artículos que sólo están disponibles en la tienda T1.
- Artículo más barato de la tienda T1.
- Personas que han comprado un mismo artículo en cada tienda.
- Tiendas que en un mismo día han vendido al menos una unidad de cada uno de sus artículos disponibles.

3) Responder en SQL a las siguientes consultas:

- Número de productos de la tienda T1 con un precio inferior a 30 euros.
- Tiendas que no disponen de artículos de la categoría C1.
- Personas que han comprado en una misma tienda más de 10 unidades de productos iguales o distintos.
- Tiendas que en un mismo día han vendido artículos de todas las categorías.
- Tiendas tales que al menos el 40% de sus artículos cuestan menos de 30 euros.

4) Responder en SQL a las siguientes peticiones:

- Elimina en la tabla VENTAS la clave primaria.
- Incrementa en un 10% el precio de todos los artículos de la tienda T1 que pertenecen a la categoría C1.
- Crea una tabla que almacene todas las ventas anteriores al 1 de enero de 2020 de la base de datos.
- Impide que los precios de un mismo artículo puedan variar según la tienda.
- Impide que una tienda pueda vender de un mismo producto más de 100 unidades en un mismo día.

Ejercicio 1

a)

$P(CT) \ S(CAT = 'C1' \vee CAT = 'C2') \ (TIENDAS * ARTICULOS)$

b)

A = B = VENTAS

$P(A.DNI) \ S((A.DNI = B.DNI) \wedge (A.CT = B.CT) \wedge (A.CA = A1) \wedge (B.CA = 'A2'))(A \times B)$

c)

A = B = P(CT, PR) S(CA = A1) (TIENDAS)

$P(A.CT) \ S((A.CT \neq B.CT) \wedge (A.PR < B.PR)) (A \times B)$

d)

A = P(CT) (TIENDAS)

B = P(DNI, CT) (VENTAS)

B/A

e)

Ejercicio 2

a)

$\text{dom}(v1) = \text{dom}(v2) = \text{ventas}$

$\{t \mid \exists v1, v2 (t[DNI] = v1[DNI]) \wedge (v1[DNI] = v2[DNI]) \wedge (v1[F] = v2[F]) \wedge (v1[CT] = v2[CT]) \wedge (v1[CA] = v2[CA]) \}$

b)

c)

d)

e)

Ejercicio 3

a)

```
SELECT COUNT(CA)
FROM TIENDAS
WHERE CT = 'T1' AND PR < 30
```

b)

```
SELECT CT
FROM TIENDAS
WHERE CA NOT IN (SELECT CA
                  FROM ARTICULOS
                  WHERE CAT = 'C1')
```

c)

```
SELECT DNI
FROM VENTAS
GROUP BY DNI, CT
HAVING SUM(NU) > 10
```

d)

```
SELECT CT
```

```
FROM VENTAS NATURAL JOIN ARTICULOS
GROUP BY CT, F
HAVING COUNT(UNIQUE CAT) = (SELECT COUNT(UNIQUE CAT)
                             FROM ARTICULOS
                             )
```

```
e)
SELECT CT
FROM TIENDAS
GROUP BY CT
HAVING COUNT(CA) * 0.4 > (SELECT COUNT(CA)
                          FROM TIENDAS
                          WHERE PR < 30)
```

Ejercicio 4

```
a)
CREATE VIEW V1
AS (SELECT DNI, IC, D/T
    FROM PARTICIPANTES NATURAL JOIN CARRERAS
    )
```

```
b)
DELETE FROM PARTICIPANTES
WHERE DNI IN(SELECT DNI
             FROM PARTICIPANTES
             WHERE T IS NULL)
```

```
c)
ALTER TABLE CARRERAS
ADD CONSTRAINT C1
CHECK PR <= 50
```

```
d)
```

```
e)
ALTER TABLE CARRERAS
ADD CONSTRAINT C1
CHECK NOT EXIST (SELECT *
                 FROM CARRERAS
                 GROUP BY CDC, YEAR(F)
                 HAVING COUNT(IC) > 5)
```