

R2: UNIDAD DIDÁCTICA 2

El siguiente esquema relacional representa el sistema de información de una ONG que se dedica a desarrollar una serie de actividades de cooperación en el extranjero. El sistema mantiene la información, para los próximos años, de las actividades que necesitan los países a los que ayuda la ONG, así como la planificación de los viajes de sus cooperantes.

Actividad(cod_act: char, nombre: char, descripción: char)

CP: {cod_act}

Único: {nombre}

VNN: {nombre}

Cooperante(dni: char, nombre: char, fecha_nac:date,
enseña: char, aprende: char)

CP: {dni}

CAj: {enseña}→Actividad(cod_act)

CAj: {aprende}→Actividad(cod_act)

VNN: {nombre, fecha_nac}

Único: {nombre}

País(cod_país: char, nombre: char, problemas: char)

CP: {cod_país}

VNN: {nombre}

Necesita(cod_país: char, cod_act: char, urgencia: entero)

CP: {cod_país, cod_act}

CAj: {cod_país}→País

CAj: {cod_act}→Actividad

VNN: {urgencia}

Viajar(dni:char, cod_país: char, fecha: date, coste: entero)

CP: {dni, cod_país}

CAj: {dni}→Cooperante

CAj: {cod_país}→País

VNN: {fecha, coste}

Único: {dni, fecha}

donde los atributos tienen el siguiente significado:

Actividad:

cod_act: código de la actividad

nombre: nombre de la actividad

descripción: descripción de la actividad

Cooperante :

dni: D.N.I. del cooperante

nombre: nombre del cooperante

fecha_nac: fecha de nacimiento del cooperante

enseña: actividad que enseña, es decir de la que es monitor

aprende: actividad que está aprendiendo

País:

cod_país: código del país

nombre: nombre del país

problemas: situación política del país

Necesita: el país de código *cod_país* necesita la actividad de código *cod_act* con grado de prioridad *urgencia* que puede valer entre 1 y 5 siendo 5 la urgencia más alta.

Viajar: el cooperante con D.N.I. *dni* va a viajar al país de código *cod_país* en la fecha *fecha*, y el viaje tiene un coste estimado de *coste* (en euros).

EXAMEN DE BASES DE DATOS Y SISTEMAS DE INFORMACIÓN (21/01/20)

1) Escribe en SQL las instrucciones que permitirían resolver las siguientes consultas:

- a. Obtener el código y el nombre de las actividades que no tienen descripción, ordenadas por nombre. (0'25 puntos)
- b. Obtener la cantidad de países que tienen alguna necesidad con urgencia 1. (0'25 puntos)
- c. Obtener el nombre de los cooperantes que no tienen previsto viajar a ningún país. (0'25 puntos)
- d. Obtener el dni y el nombre del cooperante o cooperantes más viejos que están aprendiendo alguna actividad. (0'75 puntos)
- e. Obtener el nombre y la fecha de nacimiento de los cooperantes que han nacido antes de 1980, cuyo nombre incluye la cadena 'Joan' y que van a viajar a un país que no necesita actividades con urgencia máxima. (0'75 puntos)
- f. Obtener el código y el nombre de las actividades que se necesitan en todos los países a los que va a viajar un cooperante que enseña esa actividad. No es necesario comprobar que se cumpla el antecedente. (0'75 puntos)
- g. Para todas las actividades que hay en la base de datos que son necesitadas por más de siete países, obtener el código de la actividad, el nombre de la actividad y la cantidad de cooperantes que la enseñan. (0'75 puntos)
- h. Obtener el dni, el nombre y el coste medio de sus viajes de los cooperantes que cumplen que ese coste medio es superior al coste medio de todos los viajes planificados y que enseñan una actividad que no es enseñada por ningún otro cooperante. (0'75 puntos)

2) DDL.

- a. Crea una vista para la siguiente consulta: para los cooperantes que enseñan y aprenden alguna actividad, el dni del cooperante, el nombre del cooperante y la cantidad de países a los que tienen planificados viajes. (No interesan los cooperantes que no tienen viajes planificados). (0'3 puntos)
- b. ¿Tiene sentido incluir en la instrucción anterior la opción WITH CHECK OPTION? (0'2 puntos)

EXAMEN DE BASES DE DATOS Y SISTEMAS DE INFORMACIÓN (21/01/20)

R2: UNIDAD DIDÁCTICA 2 SOLUCIONES

1) Escribe en SQL las instrucciones que permitirían resolver las siguientes consultas teniendo en cuenta lo siguiente:

a.

```
SELECT cod_act, nombre
FROM Actividad
WHERE descripción IS NULL
ORDER BY nombre;
```

b.

```
SELECT COUNT(DISTINCT N.cod_pais)
FROM Necesidad N
WHERE urgencia = 1;
```

c.

```
SELECT nombre
FROM Cooperante
WHERE dni NOT IN (SELECT dni
                  FROM Viajar);
```

= 0 =

```
SELECT nombre
FROM Cooperante C
WHERE NOT EXISTS (SELECT *
                  FROM Viajar V
                  C.dni = V.dni);
```

d.

```
SELECT C.dni, C.nombre
FROM Cooperante C
WHERE C.aprende IS NOT NULL
AND C.fecha_nac= (SELECT MIN(D.fecha_nac)
                  FROM Cooperante D
                  WHERE D.aprende IS NOT NULL);
```

e.

```
SELECT C.nombre
FROM Cooperante C
WHERE C.nombre LIKE '%Joan%' AND C.fecha_nac < '01/01/1980' AND
      C.dni IN (SELECT V.dni
                FROM Viajar V
                WHERE V.cod_pais NOT IN (SELECT N.cod_pais
                                         FROM Necesita N
                                         WHERE N.urgencia=5));
```

f.

```
SELECT A.cod_act, A.nombre
FROM Actividad A
WHERE NOT EXISTS (SELECT *
                  FROM País P
                  WHERE P.cod_pais IN
                        (SELECT V.cod_pais
                         FROM Viajar V, Cooperante C
                         WHERE V.dni=C.dni AND C.enseña=A.cod_act) AND
                  NOT EXISTS (SELECT *
                              FROM Necesita N
                              WHERE N.cod_pais=P.cod_pais AND
                                    N.cod_act=A.cod_act));
```

= 0 =

EXAMEN DE BASES DE DATOS Y SISTEMAS DE INFORMACIÓN (21/01/20)

```
SELECT A.cod_act, A.nombre
FROM Actividad A
WHERE NOT EXISTS (SELECT *
                  FROM País P, Viajar V, Cooperante C
                  WHERE P.cod_pais = V.cod_pais
                     AND V.dni=C.dni AND C.enseña=A.cod_act)AND
NOT EXISTS (SELECT *
            FROM Necesita N
            WHERE N.cod_pais=P.cod_pais AND
                  N.cod_act=A.cod_act));
```

= 0 =

```
SELECT A.cod_act, A.nombre
FROM Actividad A
WHERE NOT EXISTS (SELECT V.cod_pais
                    FROM Viajar V,Cooperante C
                    WHERE V.dni=C.dni AND C.enseña=A.cod_act)AND
NOT EXISTS (SELECT *
            FROM Necesita N
            WHERE N.cod_pais=P.cod_pais AND
                  N.cod_act=A.cod_act));
```

= 0 =

```
SELECT A.cod_act, A.nombre
FROM Actividad A
WHERE (SELECT COUNT(DISTINCT V.cod_pais)
      FROM Viajar V,Cooperante C
      WHERE V.dni=C.dni AND C.enseña=A.cod_act)
=
(SELECT COUNT(DISTINCT V.cod_pais)
 FROM Viajar V,Cooperante C
 WHERE V.dni=C.dni AND C.enseña=A.cod_act
 AND V.cod_pais IN (SELECT N.cod_pais
                   FROM Necesita N
                   WHERE N.cod_act=A.cod_act)
);
```

g.

```
SELECT A.cod_act, A.nombre, COUNT(C.dni) Cuántos_enseñan
FROM Actividad A LEFT JOIN Cooperante C ON A.cod_act=C.enseña
GROUP BY A.cod_act, A.nombre
HAVING A.cod_act IN (SELECT N.cod_act
                    FROM Necesita N
                    GROUP BY N.cod_act
                    HAVING COUNT(*)>7);
```

= 0 =

```
SELECT A.cod_act, A.nombre, (SELECT COUNT(C.dni)
                             FROM Cooperante C
                             WHERE A.cod_act=C.enseña) Cuántos_enseñan
FROM Actividad A
WHERE A.cod_act IN (SELECT N.cod_act
                   FROM Necesita N
                   GROUP BY N.cod_act
                   HAVING COUNT(*)>7);
```

= 0 =

EXAMEN DE BASES DE DATOS Y SISTEMAS DE INFORMACIÓN (21/01/20)

```
SELECT A.cod_act, A.nombre, COUNT(C.dni) Cuántos_enseñan
FROM Actividad A LEFT JOIN Cooperante C ON A.cod_act=C.enseña
WHERE (SELECT COUNT(*)
       FROM Necesita N
       WHERE N.cod_act= A.cod_act)>7
GROUP BY A.cod_act, A.nombre;
```

h.

```
SELECT C.dni, C.nombre, AVG(V.coste)
FROM Cooperante C, Viajar V
WHERE C.dni=V.dni AND C.enseña IS NOT NULL
GROUP BY C.dni, C.nombre, C.enseña
HAVING AVG(V.coste) > (SELECT AVG(W.coste) FROM Viajar W) AND
NOT EXISTS (SELECT *
            FROM Cooperante D
            WHERE C.dni<>D.dni AND
                  C.enseña=D.enseña)
```

= 0 =

```
SELECT C.dni, C.nombre, AVG(V.coste)
FROM Cooperante C, Viajar V
WHERE C.dni=V.dni AND C.enseña IS NOT NULL
AND NOT EXISTS (SELECT *
                FROM Cooperante D
                WHERE C.dni<>D.dni AND
                      C.enseña=D.enseña)
GROUP BY C.dni, C.nombre
HAVING AVG(V.coste) > (SELECT AVG(W.coste) FROM Viajar W);
```

2) DDL.

a.

```
CREATE VIEW Ejercicio2 (dni, nombre, cuántos_países)
AS
SELECT C.dni, C.nombre, COUNT(*)
FROM Cooperante C, Viajar V
WHERE C.dni=V.dni AND C.enseña IS NOT NULL AND
      C.aprende IS NOT NULL
GROUP BY C.dni, C.nombre;
```

= 0 =

```
CREATE VIEW Ejercicio2 (dni, nombre, cuántos_países)
AS
SELECT C.dni, C.nombre, (SELECT COUNT(*)FROM Viajar V
                        WHERE C.dni=V.dni)
FROM Cooperante C
WHERE C.enseña IS NOT NULL AND
      C.aprende IS NOT NULL
```

b.

No, no tiene sentido. Cada fila de la vista se genera a partir de una fila de la relación Cooperante y de varias filas de la relación Viajar por lo que esta vista no es actualizable.