

<b>EVALUACION</b>	<b>Solución Examen</b>	<b>FECHA</b>	<b>26/Ago/2016</b>
<b>MATERIA</b>	Bases de datos y Bases de datos 1		
<b>CARRERA</b>	AP – ATI		
<b>CONDICIONES</b>	- Puntos: 100 - Duración: 2 ½ horas - SIN material		

**IMPORTANTE: “Poner nombre del docente en la hoja de escrito.”**

### Ejercicio 1

Una empresa de entretenimientos para niños en edad escolar desea automatizar el manejo de la información de sus clientes y las asociaciones con las que trabaja.

Existen varias asociaciones juveniles, que tienen sus propias colonias de vacaciones. Cada asociación tiene varias colonias, pero cada colonia pertenece a una única asociación.

De cada asociación se conoce su nombre, que la identifica, la dirección y un teléfono de referencia.

De las colonias se conoce su código y ubicación; el código puede repetirse para las distintas asociaciones.

En las colonias trabajan varios líderes de grupos, de los cuales se conoce su C.I., nombre y teléfono.

Cada líder puede trabajar para varias colonias. Todos los líderes deben tener una certificación que los acredita como tales, interesa la fecha, el grado y la asociación que emitió el certificado. En caso de tener más de un certificado interesa sólo el más reciente.

Cada líder en una colonia coordina exactamente una actividad, pero puede ayudar en otras. Las actividades a su vez son desarrolladas por varios líderes de colonias.

De las actividades se conoce su identificador y una breve descripción de la misma. Estas pueden ser de los siguientes tipos: campamentos, deportes y juegos.

De los campamentos interesa la ubicación y la duración en días, de los deportes interesa el tipo, los accesorios necesarios y la cantidad de horas semanales de entrenamiento, de los juegos interesa el tipo de juego, una descripción de las características y la cantidad de participantes.

Cada colonia atiende a un conjunto de clientes, algunos de ellos asisten a más de una colonia. Nos interesa el número de cliente que lo identifica, nombre, C.I., teléfono y edad.

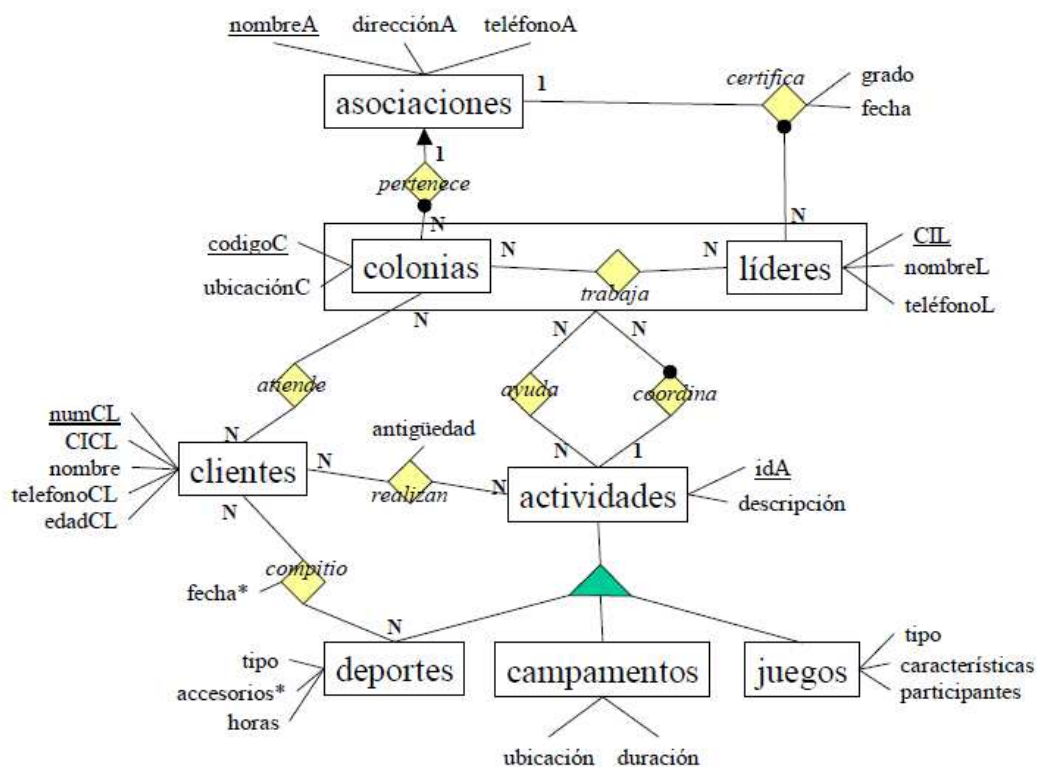
Los clientes realizan diversas actividades, interesando la antigüedad con que las realizan. En el caso de los deportes, interesan también las fechas en las que el cliente participó en competencias. Los clientes sólo realizan actividades de las disponibles en su colonia.

**Se pide:**

Modelo Entidad-Relación completo.

**(Máximo del Ejercicio 1 es 50 puntos)**

Solución:



---

## Ejercicio 2

**ESQUEMAS** (idEsquema, nomEsquema, fechaCreacion, cantTablas<sup>+</sup>).

Almacena la información de los esquemas creados. Cada esquema agrupa un conjunto de objetos como pueden ser tablas, índices y atributos. Para cada esquema se conoce su identificador, su nombre que lo describe y la fecha en que fue creado.

**TABLAS** (idTabla<sub>1</sub>, idEsquema<sub>2</sub>, nomTabla<sub>2</sub>, definicion).

Almacena la información de las tablas creadas en un esquema. Para cada una se conoce su identificador, el esquema al que pertenece, su nombre y su definición que es el texto SQL de las sentencias de creación de la misma.

**ATRIBUTOS** (idAtributo<sub>1</sub>, idTabla<sub>2</sub>, nomAtributo<sub>2</sub>, tipoAtributo, clave)

Almacena la información de los atributos de las tablas de un esquema. Para cada uno se conoce su identificador, la tabla a la que pertenece, su nombre, el tipo de dato admitido y si forma parte o no de la clave primaria de la tabla a la que pertenece (S por SI, N por No).

**INDICES** (idIndice<sub>1</sub>, idTabla<sub>2</sub>, nomIndice<sub>2</sub>, tipoIndice)

Contiene la información de los índices creados sobre las tablas de un esquema. Para cada uno se conoce su identificador, la tabla sobre la cual fue creada, su nombre y el tipo del índice (BTREE o HASH).

**ATRIBUTOS\_INDICES** (idIndice<sub>1</sub>, idAtributo<sub>1</sub>, posicion, orden)

Almacena la información de los atributos que componen los diferentes índices. Para cada atributo de cada índice se conoce la posición que ocupa dentro de la composición del índice. El primer atributo ocupa la posición 1 y así sucesivamente. También se sabe si los registros para dicho atributo en el índice se ordenan de forma ascendente (A) o descendente (D).

- a) Indique que retorna la siguiente consulta SQL

```
SELECT I.nomIndice, I.tipoIndice, count(AI.idAtributo)
FROM INDICES I natural join ATRIBUTOS_INDICES AI
GROUP BY I.idIndice, I.nomIndice, I.tipoIndice
HAVING count(*) > 5
```

(Máximo 5 puntos)

Solución: Nombre, tipo de los índices y cantidad de atributos que tienen, de aquellos índices que están definidos sobre más de 5 atributos. (Se asume que los índices son correctos).

- b) Indique que retorna la siguiente consulta SQL

```
SELECT T.nomTabla, E.nomEsquema
FROM (Esquemas E natural join Tablas T)
WHERE NOT EXISTS
(SELECT *
FROM Atributos A
WHERE A.idTabla = T.idTabla
and NOT EXISTS (SELECT *
FROM Atributos_Indices AI
WHERE AI.idAtributo = A.idAtributo))
```

(Máximo 5 puntos)

Solución: Nombre de las tablas y nombre de los esquemas al que pertenecen, tales que todos sus atributos pertenecen a algún índice definido sobre la tabla. (Se asume que los índices son correctos).

Escribir las consultas SQL (compatible con la versión de MS SQL Server utilizada durante el curso) para responder a las siguientes solicitudes:

- c) Listar los id de índice que tengan al menos 3 atributos

**Solución:**

```
SELECT idIndice
FROM Atributos_Indices
GROUP BY idIndice
HAVING COUNT(*) >= 3
```

- d) Listar los id de tabla que tienen un único índice

**Solución:**

```
SELECT idTabla
FROM Indices U
WHERE NOT EXISTS (SELECT *
                  FROM Indices D
                  WHERE U.IdTabla = D.IdTabla
                  AND U.IdIndice <> D.IdIndice)
```

- e) Listar para cada id de índice la cantidad de atributos en orden ascendente (A) que tienen.

**Solución:**

```
SELECT idIndice, COUNT(*)
FROM Atributos_Indices
WHERE orden = 'A'
GROUP BY idIndice
```

- f) Listar el id y nombre de esquema que tenga la mayor cantidad de tablas. (Si hubiera más de uno con la máxima cantidad, deben aparecer todos ellos en el resultado)

**Solución:**

```
SELECT idEsquema, nomEsquema
FROM Esquemas
WHERE cantTablas = (SELECT MAX(D.cantTablas)
                  FROM Esquemas D)
```

(Cada consulta (c-f) vale un máximo de 10p. Máximo puntaje del ejercicio 2 = 50 puntos).

Notas:

- En SQL debe evitarse en todos los casos el repetir datos que no aporten información útil en los resultados de las consultas.
- Se puede asumir que se cumple la integridad referencial pero no se puede asumir la participación total.