

# Primer Parcial de Fundamentos de Bases de Datos

Setiembre 2016

## Indicaciones Generales:

- La duración de la prueba es de **tres (3)** horas.
- En la prueba **NO** se permite consultar material alguno.
- Empezar cada ejercicio en una hoja nueva.
- Escribir con lápiz y de un solo lado de las hojas.
- Numerar todas las hojas. Incluir en cada hoja la cédula y el nombre. En la primer hoja, incluir la cantidad de hojas que se entregan.

## Ejercicio 1. (1 punto)

Se desea modelar la base de datos que contendrá los datos correspondientes a una competencia de baile. En esta competencia baila una pareja por vez y es evaluada por un jurado. En primer lugar, interesa registrar los datos de todas las personas que participan de la misma, estos son CI, nombre, dirección y teléfono. Algunas de las personas son jurados o bailarines. De los jurados interesan todos sus títulos (bailarín, periodista, etc.) y si es suplente o titular. No se permiten jurados que participen en la competencia de baile.

De los ritmos musicales que se proponen en la competencia se conoce un código que los identifica y su nombre (por ejemplo salsa, merengue, reguetón, etc.). Por otro lado, interesa registrar la fecha y hora de cada baile para cada ritmo musical, en el cual participa una sola pareja. Nunca hay dos bailes simultáneos del mismo ritmo, aunque puede haberlos de ritmos diferentes. Por otra parte, todas las parejas bailan al menos un ritmo.

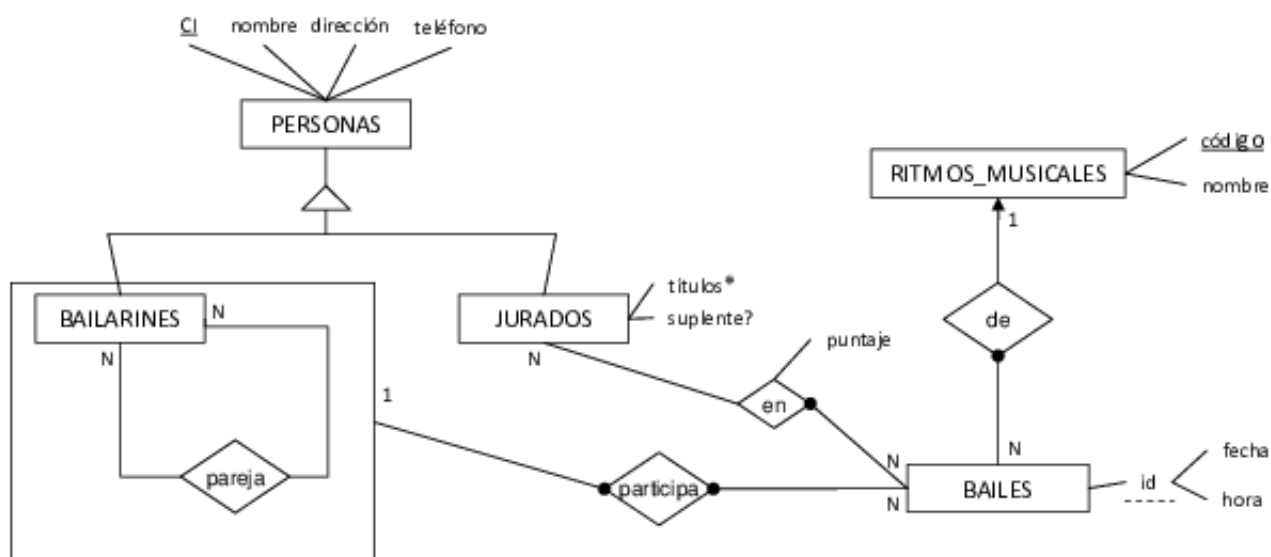
Tenga en cuenta que un bailarín dado, nunca puede participar de dos bailes simultáneos.

Además, se registra cuáles fueron los jurados que participaron en la evaluación de cada baile y el puntaje que éstos asignaron.

**Se Pide:** Realizar el Modelo Entidad Relación completo.

Debe incluir el diagrama entidad relación usando los símbolos vistos en el curso y las restricciones no estructurales.

**Solución.**



**RNE:**

- Un bailarín no puede formar pareja con sí mismo.  
 $\forall x \in Bailarines. \langle x, x \rangle \notin pareja$
- Las parejas, son simétricas.  
 $\forall x \in Bailarines. \forall y \in Bailarines. (\langle x, y \rangle \in pareja \rightarrow \langle y, x \rangle \in pareja)$
- Un jurado no puede participar como bailarín.  
 $Jurados \cap Bailarines \equiv \emptyset$
- Todos los bailes de un bailarín en el mismo ritmo, son con la misma pareja.

$$\forall x \in Bailarines. \forall y \in Bailarines. \\
\neg \exists r \in Ritmos\_Musicales. ((\exists b \in Bailes. \langle x, y, b \rangle \in participa \wedge \langle b, r \rangle \in De) \wedge \\
\exists z \in Bailarines. (\exists b_1 \in Bailes. \langle x, z, b_1 \rangle \in participa \wedge \langle b_1, r \rangle \in De \wedge z \neq y))$$

- Un bailarín no puede participar de dos bailes simultáneos.

$\forall x \in \text{Bailarines}. \neg \exists b, b_1 \in \text{bailes}.$

$(b \neq b_1 \wedge b \wedge \text{fecha}(b) = \text{fecha}(b_1) \wedge \text{hora}(b) = \text{hora}(b_1) \wedge$

$\exists y, z \in \text{Bailarines}(\langle \langle x, y \rangle, b \rangle \in \text{participa} \wedge \langle \langle x, z \rangle, b_1 \rangle \in \text{participa}))$

**Ejercicio 2.** (1 punto)

Considere la realidad descrita en la siguiente base de datos de locales de maquinas:

Los datos de cada local son: su identificación (#local), barrio, dirección, el promedio diario de fichas que vende y el valor de la ficha; y se encuentran en el esquema relación:

**LOCALES** ( #local, barrio, dirección, promFichas, valorFicha )

Los datos de cada juego son: su identificación (#juego), su origen y su tipo (ej.: deportivo, bélicos, etc.); y se encuentran en el esquema relación:

**JUEGOS** ( #juego, origen, tipo)

Los datos sobre cada maquina son: su identificación (#maquina), la identificación del juego que tiene y la identificación del local en el que se encuentra; y se encuentran en el esquema relación:

**MAQUINAS** ( #maquina, #juego, #local )

En esta base de datos se cumplen las siguientes restricciones de inclusión y no hay tablas vacías:

$$\begin{aligned} \Pi_{\#local}(MAQUINAS) &\subseteq \Pi_{\#local}(LOCALES) \\ \Pi_{\#juego}(MAQUINAS) &\subseteq \Pi_{\#juego}(JUEGOS) \end{aligned}$$

Con la información almacenada en los esquemas relacionales anteriores:

- (a) Resolver mediante operaciones de **álgebra relacional** la siguiente consulta:

*Obtener el barrio y la dirección de los locales que no tienen juegos de tipo “Rol”.*

**Solución.**

$$A = \Pi_{\#local}(MAQUINAS * (\sigma_{tipo=ROL}(JUEGOS)))$$

Locales que tienen juegos de tipo ROL.

$$SOL = \Pi_{barrio,direccion}(((\Pi_{\#local}LOCALES) - A) * LOCALES)$$

- (b) Resolver mediante una expresión del **cálculo relacional de tuplas** la siguiente consulta:

*Obtener el número y origen de los juegos que están en todos los locales que tienen un promedio diario de fichas vendidas mayor a 20. En caso de no existir locales con esta característica el resultado de la consulta debe ser vacío.*

**Solución.**

$$\begin{aligned} &\{t.\#juego, t.origen / JUEGOS(t) \wedge \\ &\quad (\forall l)(LOCALES(l) \wedge l.promFichas > 20 \\ &\quad \rightarrow \\ &\quad (\exists m)(MAQUINAS(m) \wedge m.\#juego = t.\#juego \wedge m.\#local = l.\#local) \\ &\quad ) \wedge \\ &\quad (\exists l)(LOCALES(l) \wedge l.promFichas > 20) \\ &\} \end{aligned}$$

- (c) Resolver la siguiente consulta en **SQL** sin utilizar vistas ni subconsultas en el FROM:

*Obtener el número y tipo de los juegos que no están en los locales con el menor valor de la ficha.*

**Solución.**

```
SELECT J.#juego, J.tipo
FROM JUEGOS J
WHERE NOT EXISTS
  (SELECT *
   FROM MAQUINAS M natural join LOCALES L
```

```

WHERE M.#juego = J.#juego
      AND L.valorFicha = (SELECT min(valorFicha)
                          FROM LOCALES
                          )
    )

```

- (d) Dada la siguiente expresión del cálculo relacional de tuplas obtener una expresión en álgebra relacional o en SQL que sea equivalente. En caso de dar una expresión en SQL no se permite utilizar vistas ni subconsultas en el FROM

$$\{t.\#juego, t.origen / JUEGOS(t) \wedge t.tipo = Accion \wedge \neg(\exists m)(MAQUINAS(m) \wedge m.\#juego = t.\#juego \wedge (\exists l)(LOCALES(l) \wedge l.\#local = m.\#local \wedge l.barrio = Centro))\}$$

**Solución.**

La consulta Obtiene los números y origen de los juegos de tipo “Acción” que no están en locales ubicados en el Centro (barrio Centro).

**Álgebra Relacional**

$$A = \Pi_{\#juego}(\sigma_{tipo=Accopm} JUEGOS)$$

Números de los juegos de tipo Acción

$$B = \Pi_{\#juego}((MAQUINAS * (\sigma_{barrio=Centro} LOCALES)) * A)$$

Números de los los juegos de tipo Acción que están en locales del Centro

$$SOL = \Pi_{\#juego, origen}((A - B) * JUEGOS)$$

**SQL**

```

SELECT J.#juego , J. origen
FROM JUEGOS J
WHERE J. tipo=Accion
AND J.#juego NOT in
      (SELECT M.#juego
       FROM MAQUINAS M natural join LOCALES L
       WHERE L. barrio=Centro
      )

```

- (e) Dadas las siguientes expresiones del álgebra relacional:

$$A = \Pi_{\#local}(\sigma_{barrio=Centro} LOCALES)$$

$$B = \Pi_{\#juego, \#local}(MAQUINAS) \% A$$

$$RES = \Pi_{\#juego, origen}(B * JUEGOS)$$

Obtener la instancia del resultado (RES) si la instancia de las tablas de la base de datos son las siguientes:

	#local	barrio	direccion	promFichas	valorFicha
LOCALES	1	Centro	18 de julio 1111	35	50
	2	Carrasco	Avenida Italia 1111	21	60
	3	Centro	18 de julio 2222	10	55
	4	Aguada	Marmaraja 1111	48	70
	5	Belvedere	Garzon 1111	44	50
	6	Centro	18 de julio 3333	10	50

JUEGOS

#juego	origen	tipo
1	China	Aventura
2	UK	Accion
3	Usa	Rol

MAQUINAS

#maquina	#juego	#local
1	1	1
2	1	2
3	1	3
4	1	4
5	1	5
6	1	6
7	2	1
8	2	3
9	2	4
10	2	6
11	3	1

**Solución.**

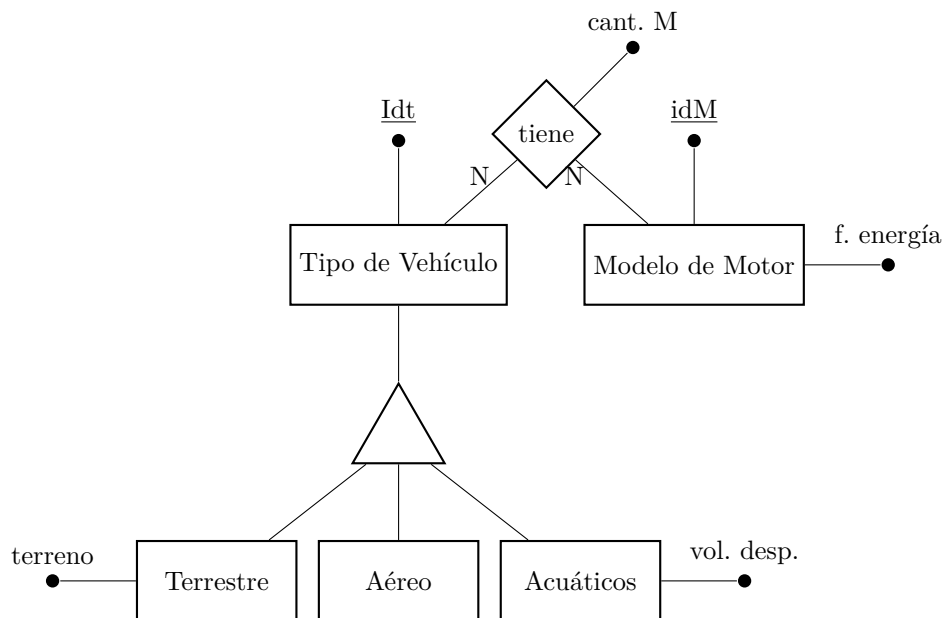
Número y origen de los juegos que están en todos los locales del Centro.

#juego	origen
1	China
2	UK

**Ejercicio 3.** (1 punto)

La FING quiere construir una base de datos de tipos de vehículos para transporte de carga o personas.

El Diagrama entidad relación de la base hasta el momento es el siguiente



$$Terrestre \cap Aereo \equiv \emptyset$$

$$Terrestre \cap Acuatico \equiv \emptyset$$

$$Acuatico \cap Aereo \equiv \emptyset$$

$$Tipo\ de\ Vehiculos \equiv Terrestre \cup Aereo \cup Acuatico$$

De los tipos de vehículos se conoce una identificación y se dividen en terrestres, aéreos y acuáticos. Para los terrestres se conoce sobre qué tipo de terreno pueden circular (Cualquiera, Pavimento, Rieles, Nieve, etc). De los acuáticos, se conoce el volumen de desplazamiento.

De los motores, se conoce una identificación y el tipo de energía que utiliza (eléctrica, diesel, gas, nuclear, etc.).

Por otra parte, cada vehículo puede tener más de un motor (Ej: autos híbridos, barcos) o ninguno (Ej: Bicicleta, planeador). Para cada vehículo se conoce la cantidad de motores que tiene para cada modelo de motor.

- (a) Construya una representación relacional de la realidad. Debe describir las tablas, las dependencias de inclusión y las dependencias funcionales que se deben cumplir.

Justifique su respuesta.

**Solución.**

A continuación se detalla para cada estructura de datos, cuál es la representación relacional y la justificación de por qué se elige esa representación.

**Categorización** El punto más complejo a resolver es cómo representar la categorización. Habría dos posibilidades obvias:

**Por Joins** Creando una tabla por cada entidad y sub entidad. Esto, dado el caso, admite una variación que es crear sólo las subentidades y definir la entidad principal como una vista. El mayor inconveniente de estas soluciones es la performance en el acceso a los datos. Por otra parte, la solución permite la máxima flexibilidad a la hora de imponer las restricciones de integridad, dado que muchas de ellas (en particular las funcionales) se imponen a nivel de tabla (y nunca de parte de una tabla.)

**Única tabla con atributo de tipo** . Si bien, la solución puede tener buena performance en el acceso, no es así en el manejo de los nulos, o la definición de las restricciones de integridad. Otro problema que se puede presentar es que puedan existir restricciones de integridad que no se puedan imponer en forma adecuada y deban implementarse posteriormente mediante triggers, checks o mecanismos similares.

En base a este mínimo análisis, y ante todo por la flexibilidad que plantea, se elige como solución la primera: Por Joins. La solución entonces es la siguiente<sup>1</sup>:

$$\begin{aligned} V_{Terrestre}(IdT, terreno) &: \{IdT \rightarrow terreno\} \\ V_{Aereo}(IdT) & \\ V_{Acuatico}(IdT, volDesp) &: \{IdT \rightarrow volDesp\} \\ TVehiculo &\equiv \Pi_{IdT}(V_{Terrestre}) \cup \Pi_{IdT}(V_{Acuatico}) \cup V_{Aereo} \end{aligned}$$

Dado que se presenta la solución de la la vista para la categoría principal, no son necesarias dependencias de inclusion.

Las dependencias funcionales en cada tabla, surgen del hecho de que los atributos de la izquierda de cada dependencia son determinantes en el MER.

**Entidad Modelo de Motor** Esta estructura no presenta mayores problemas, teniendo únicamente una dependencia funcional que surge de la indicación del atributo determinante.

$$MMotor(IdM, fEnergia) : \{IdM \rightarrow fEnergia\}$$

**Relación tiene** La primer observación que hay que realizar es la cardinalidad de la relación, que en este caso es  $N : N$ . Es por esto, que se representará con una tabla propia de la siguiente forma:

$$\begin{aligned} V_{tieneM}(IdT, IdM, CantM) &: IdT, IdM \rightarrow CantM \\ \Pi_{IdT}(V_{tieneM}) &\subseteq \Pi_{IdT}(TVehiculo) \\ \Pi_{IdM}(V_{tieneM}) &\subseteq \Pi_{IdM}(MMotor) \end{aligned}$$

No hay más dependencias de inclusión dado que no hay totalidades en el MER.

- (b) Considere ahora que la subentidad correspondiente a los vehículos terrestres tuvo la siguiente modificación:



Haga las modificaciones necesarias en su respuesta de la parte anterior para representar este cambio.

**Solución.**

Dada la representación elegida para la categorización en la parte anterior, las modificaciones necesarias quedan asociadas a la tabla  $V_{Terrestre}$ .

Es necesario agregar al atributo en la tabla y además una dependencia funcional que lo establezca como clave. De esta forma, la tabla queda definida como sigue:

$$V_{Terrestre}(IdT, IdTerr, terreno) : \{IdT \rightarrow terreno, IdTerr \rightarrow IdT, IdT \rightarrow IdTerr\} \quad (1)$$

- (c) Considere ahora la tabla **Dueño**(idTerr, propietario, fecha), cuyo contenido es el identificador del vehículo terrestre, el nombre de su propietario, y la fecha en que el propietario compró el terreno. Cuál es el resultado de la siguiente consulta?

```
Select count (*)
From Dueño as R1, Dueño as R2
Where R1.IdTerr = R2.IdTerr
      and R1.propietario = R2.propietario and R1.fecha<>R2.fecha
```

Justifique su respuesta.

**Solución.**

La consulta pide la cantidad de tuplas del producto de *Dueños* consigo misma, tal que coinciden en el terreno y el propietario pero tienen diferente la fecha. Esto significa que en la tabla en cuestión, hay al menos dos tuplas diferentes que coinciden en el terreno y el propietario y eso es imposible porque terreno y propietario forman la clave primaria de la tabla.

Es por esto que el resultado de la consulta es **0**.