

Primer Parcial de Fundamentos de Bases de Datos

Setiembre 2023

Duración: 3 horas

Presentar la resolución del parcial:

- Con las hojas numeradas y escritas de un solo lado.
- Con la cantidad de hojas entregadas en la primer hoja.
- Con cédula de identidad y nombre en cada hoja.
- Escrita a lápiz y en forma prolija.

Ejercicio 1 (14 puntos)

Se desea modelar una base de datos con información sobre parques de diversiones.

Cada parque de diversión tiene un código que lo identifica, un nombre, capacidad máxima de visitantes, horarios de apertura (Ejemplo: viernes 8:00-18:00, sábado 10:00-22:00, etc.) y la ubicación (calle, número de puerta y ciudad). Los parques de diversión contienen distintas atracciones, las cuales poseen un código que las identifica dentro del parque. Además, tienen un nombre, tipo (montaña rusa, infantil, etc.) y altura mínima requerida para ingresar.

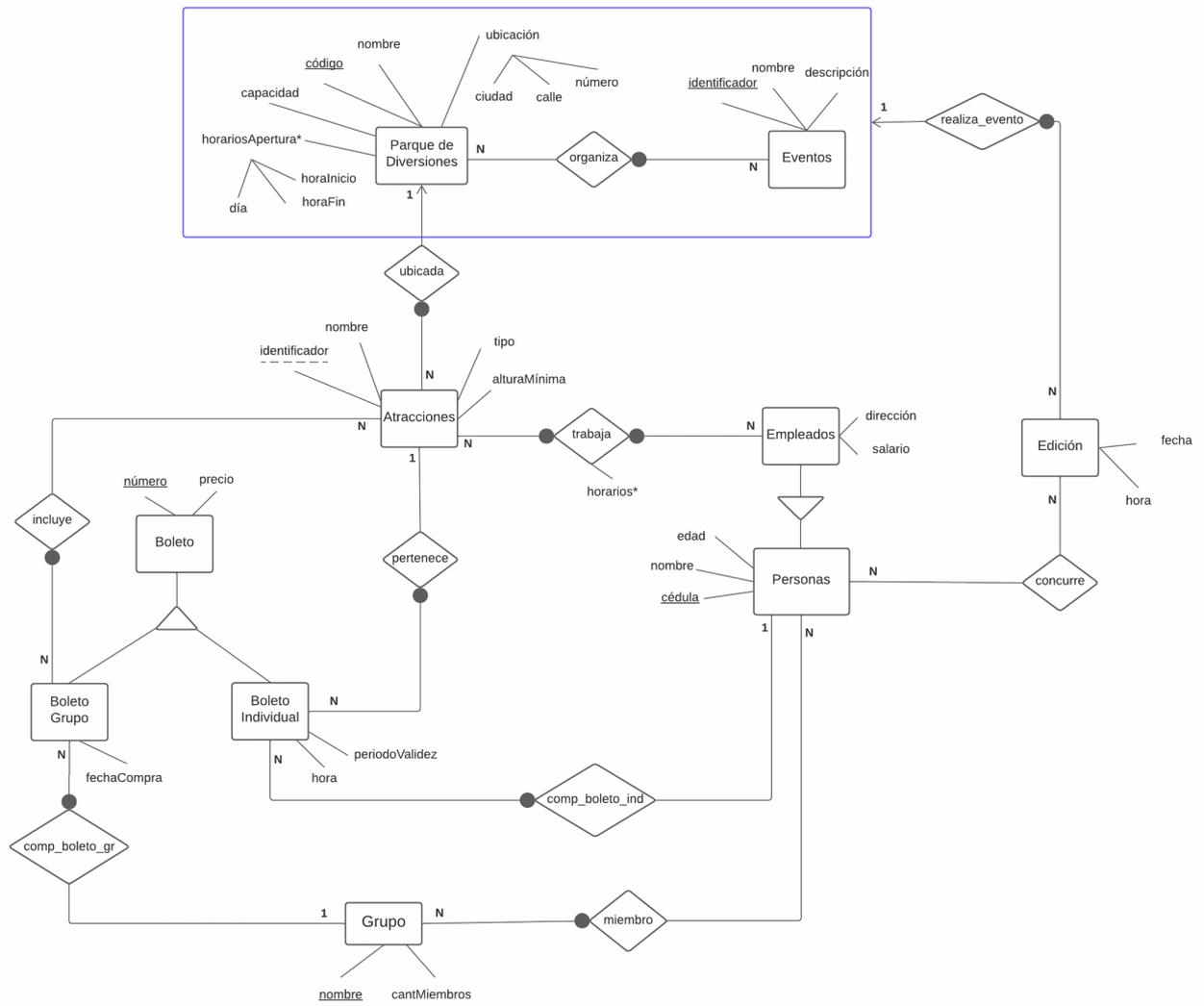
En las atracciones trabajan personas (empleados) y para cada una de ellas se quiere almacenar los horarios de trabajo en cada atracción (pudiendo una persona trabajar en varias atracciones). De las personas se conoce la cédula, el nombre y la edad. Además, de los empleados se conoce su dirección y salario. Existen grupos de personas, los cuales se identifican por su nombre y también interesa almacenar el total de miembros y las personas que integran cada grupo.

La entrada a los parques es gratuita, pero para acceder a una atracción es necesario comprar un boleto. Los boletos tienen un número que los identifica y un precio. Los boletos individuales los compra una persona y valen para una atracción, y tienen una hora y período de validez (Ejemplo: 15hs. – 2 horas). Estos boletos solo los pueden comprar personas mayores de 18 años. Por otro lado, también existen boletos grupales, que son comprados por los grupos, los cuales son válidos para un conjunto de atracciones dentro de un mismo parque. De estos boletos se conoce la fecha de compra.

Por último, los parques organizan distintos eventos (Ejemplo: conciertos, festivales, desfiles, etc.) de los cuales se conoce un identificador, el nombre y la descripción. Un mismo evento se puede repetir en el tiempo y realizarse en varios parques (Ejemplo: El desfile de Navidad se hace todos los años en el Parque 1, Parque 2 y Parque 3), siendo cada una de esas una edición diferente. De cada edición del evento, se quiere almacenar la fecha y hora en la cual se realiza. Se quiere registrar a qué edición del evento concurren las personas.

Se pide: Modelo Entidad-Relación completo del problema.

Solución:



RNEs:

1. Las atracciones asociadas a un Boleto de Grupo deben pertenecer al mismo Parque de Diversiones:

$$(\forall bg \in \text{BoletoGrupo})(\forall a_1 \in \text{Atracción})(\langle bg, a_1 \rangle \in \text{incluye} \wedge (\exists a_2 \in \text{Atracción})(\langle bg, a_2 \rangle \in \text{incluye}) \rightarrow \text{ubicada}(a_1) = \text{ubicada}(a_2))$$

2. Las personas que compran boletos individuales son mayores de edad (mayores de 18 años)

$$(\forall p \in \text{Persona})(\forall bi \in \text{BoletoIndividual})(\langle p, bi \rangle \in \text{compra_boleto_individual} \rightarrow \text{edad}(p) \geq 18)$$

3. El atributo *cantidadMiembros* se calcula como la suma de las instancias de Persona asociadas al Grupo mediante la relación **miembro**

$$(\forall g \in \text{Grupo})(\sum_{p \in \text{Persona}: \langle p, g \rangle \in \text{miembro}} 1 = \text{cantMiembros}(g))$$

4. Una persona no pueden estar en 2 ediciones distintas que sucedan al mismo tiempo.

$$(\forall p \in \text{Persona})(\forall e_1, e_2 \in \text{Edicion})(\langle p, e_1 \rangle \in \text{concorre} \wedge \langle p, e_2 \rangle \in \text{concorre} \wedge \text{fecha}(e_1) = \text{fecha}(e_2) \wedge \text{hora}(e_1) = \text{hora}(e_2)) \rightarrow \text{realiza_evento}(e_1) = \text{realiza_evento}(e_2))$$

5. $\text{Boleto} = \text{BoletoIndividual} \cup \text{BoletoGrupo}$

6. $\text{BoletoIndividual} \cap \text{BoletoGrupo} = \emptyset$

Ejercicio 2 (18 puntos)

Considere el siguiente esquema relacional de una agencia de viajes.

CLIENTES (ciCliente, nomCliente, telefono, direccion, edad)

Datos personales de los clientes.

PASAJES (codPasaje, ciCliente, destino, aerolinea, fechaSalida, fechaRegreso)

Datos de cada pasaje vendido a un cliente.

HOTELES (codHotel, nomHotel, ciudad, pais, web)

Datos de los hoteles con los que trabaja la agencia.

PAQUETES (codPaquete, codHotel, duracion, precio)

Datos de los paquetes que ofrece la agencia cuando se compra un pasaje.

PASAJE-PAQ (codPasaje, codPaquete, fecha-vta)

Datos de los paquetes que se vendieron a los clientes junto a los pasajes. Corresponde a los casos en que el cliente compró un paquete asociado al pasaje.

$$\begin{aligned}\Pi_{ciCliente}(PASAJES) &\subseteq \Pi_{ciCliente}(CLIENTES) \\ \Pi_{codHotel}(PAQUETES) &\subseteq \Pi_{codHotel}(HOTELES) \\ \Pi_{codPaquete}(PASAJE-PAQ) &\subseteq \Pi_{codPaquete}(PAQUETES) \\ \Pi_{codPasaje}(PASAJE-PAQ) &\subseteq \Pi_{codPasaje}(PASAJES)\end{aligned}$$

Nota: Ninguna de las tablas se encuentra vacía.

Se pide:

1. Resolver en Álgebra Relacional las siguientes consultas

- a) Devolver los nombres de los hoteles que solamente se encuentran en paquetes cuya duracion es mayor a 10 días.

$$\begin{aligned}A &= \Pi_{codHotel} (\sigma_{duracion \leq 10} (Paquetes)) \\ SOL &= \Pi_{nomHotel} ((\Pi_{codHotel} (Paquetes) - A) * Hoteles)\end{aligned}$$

- b) Devolver los códigos de los paquetes que fueron vendidos antes del 13/3/2020 al precio más bajo de los vendidos en ese período.

$$\begin{aligned}A &= \Pi_{codPaquete, precio} (\sigma_{fecha-vta \leq 13/3/2020} (Pasaje-Paq) * Paquetes) \\ B &= \rho_{codPaquete, precio \rightarrow CP, p} (A) \\ C &= \Pi_{CP} (\sigma_{p > precio} (B \times A)) \\ SOL &= \Pi_{codPaquete} (A) - C\end{aligned}$$

Otra solución:

$$A = \prod_{\text{codPaquete, precio}} (\sigma_{\text{fecha-vta} \leq 13/3/2020} (\text{Pasaje-Paq}) * \text{Paquetes})$$

$$B = \rho_{\text{codPaquete, precio} \rightarrow \text{CP, p}} (A)$$

$$C = \prod_{\text{codPaquete, CP}} (\sigma_{\text{precio} \leq p} (A \times B))$$

$$\text{SOL} = C \% \prod_{\text{codPaquete}} (A)$$

2. Resolver en Cálculo Relacional las siguientes consultas

- a) Devolver las parejas <nomHotel, nomCliente> tal que el cliente compró una estadia en el hotel.

$$\begin{aligned} \{ <u.\text{nomHotel}, t.\text{nomCliente}> / \text{Hoteles}(u) \wedge \text{Clientes}(t) \wedge \\ & (\exists x) (\text{Pasajes}(x) \wedge x.\text{ciCliente} = t.\text{ciCliente} \wedge \\ & (\exists w) (\text{Pasaje-Paq}(w) \wedge w.\text{codPasaje} = x.\text{codPasaje} \wedge \\ & (\exists v) (\text{Paquetes}(v) \wedge v.\text{codPaquete} = w.\text{codPaquete} \wedge \\ & v.\text{codHotel} = u.\text{codHotel})) \} \end{aligned}$$

- b) Devolver los destinos de los pasajes adonde fueron (o irán) todos los clientes de entre 20 y 30 años.

$$\begin{aligned} \{ <t.\text{destino}> / \text{Pasajes}(t) \wedge \\ & (\forall u) (\text{Clientes}(u) \wedge u.\text{edad} \geq 20 \wedge u.\text{edad} \leq 30 \\ & \rightarrow (\exists v) (\text{Pasajes}(v) \wedge v.\text{destino} = t.\text{destino} \wedge v.\text{ciCliente} = u.\text{ciCliente})) \\ & \wedge (\exists x) (\text{Clientes}(x) \wedge x.\text{edad} \geq 20 \wedge x.\text{edad} \leq 30) \} \end{aligned}$$

3. Resolver en SQL las siguientes consultas

- a) Devolver los códigos de paquetes que fueron vendidos a más de 30 personas menores de 18 años.

```
SELECT codPaquete
FROM Pasaje-Paq P1 NATURAL JOIN Pasajes P2 NATURAL JOIN Clientes C
WHERE C.edad < 18
GROUP BY codPaquete
HAVING COUNT(distinct ciCliente) > 30
```

- b) Devolver los nombres de clientes que compraron estancias en todos los hoteles de Viena ofrecidos por la agencia.

```
SELECT nomCliente
FROM Clientes C
WHERE NOT EXISTS
  (SELECT *
   FROM Hoteles H
   WHERE ciudad = 'Viena' AND
        NOT EXISTS
          (SELECT *
           FROM Pasaje-Paq P1 NATURAL JOIN Pasajes P2
           NATURAL JOIN Paquetes P3
           WHERE P2.ciCliente = C.ciCliente AND P3.codHotel = H.codHotel))
```

Otra solución:

```
SELECT nomCliente
FROM Clientes C
WHERE NOT EXISTS
  (SELECT *
   FROM Hoteles H
   WHERE H.ciudad = 'Viena' AND
        H.codHotel NOT IN
          (SELECT P3.codHotel
           FROM Pasaje-Paq P1 NATURAL JOIN Pasajes P2
           NATURAL JOIN Paquetes P3
           WHERE P2.ciCliente = C.ciCliente))
```

Ejercicio 3 (8 puntos)

Se desea construir una base de datos para un centro de investigación científica.

Existe un conjunto de investigadores, de los que se registra la ci (que los identifica), el nombre, el teléfono y sus áreas de investigación (que pueden ser varias). Por otro lado, se guardan los datos de los departamentos del centro, de los cuales se conocen un identificador, el nombre, un número de oficina y el investigador que lo dirige (que es solamente uno). Además, se tiene registro de los proyectos en curso, de los cuales se conoce un código (que los identifica), el título, la duración, el investigador responsable del proyecto y los departamentos que participan. Cada proyecto tiene un único investigador responsable.

Un investigador puede trabajar en más de un departamento y en más de un proyecto. Un investigador no puede tener asociada la misma área de investigación en más de un departamento. Es decir, si un investigador trabaja en un área lo hace en el contexto de un único departamento.

A continuación se presentan los nombres de los atributos de esta base de datos:

ci-inv	ci-director	cod-proy
nom-inv	id-depto	tit-proy
tel-inv	nom-depto	duracion
area-inv	ofic-depto	ci-responsable

Se pide:

a) Indicar, con **V** o **F**, cuáles de las siguientes dependencias funcionales se cumplen en la realidad descrita y cuáles no. En cada respuesta **Verdadera** mostrar qué parte de la descripción de la realidad la justifica o cómo se deduce la dependencia funcional.

1. ci-inv → nom-inv, tel-inv, area-inv **F**

2. id-depto → nom-depto, ofic-depto, ci-director **V**

“... los departamentos del centro, de los cuales se conocen un identificador, el nombre, un número de oficina y el investigador que lo dirige (que es solamente uno)”

3. cod-proy → tit-proy, duracion, ci-responsable **V**

“los proyectos en curso, de los cuales se conoce un código (que los identifica), el título, la duración, el investigador responsable del proyecto ... Cada proyecto tiene un único investigador responsable”

4. nom-inv, tel-inv, area-inv → ci-inv **F**

5. ci-inv, area-inv → id-depto **V**

“Un investigador no puede tener asociada la misma área de investigación en más de un departamento. Es decir, si un investigador trabaja en un área lo hace en el contexto de un único departamento.”

6. ci-inv → nom-inv, tel-inv **V**

“... investigadores, de los que se registra la ci (que los identifica), el nombre, el teléfono ...”

7. ci-inv → id-depto **F**

8. ci-inv, area-inv → ofic-depto **V**

Se deduce de las Dfs 5 y 2. Primero aplicando descomposicion en 2 y luego aplicando transitiva entre DF 5 y id-depto → ofic-depto

Otra forma de explicar que se cumple es calculando la clausura de los atributos de la izquierda bajo las Dfs que identificaron que se cumplen.

$\{ci-inv, area-inv\}^+ = \{ci-inv, area-inv, id-depto, nom-depto, ofic-depto, ci-director, \dots\}$

id-depto por la DF 5

nom-depto, ofic-depto, ci-director por la DF 2

Como ofic-depto $\in \{ci-inv, area-inv\}^+$, por definición de clausura de un conjunto de atributos se cumple la DF 8.

9. tit-proy \rightarrow ci-responsable F

10. ci-director \rightarrow id-depto F

a) Dada la siguiente instancia de una tabla de esta base de datos, decir si es una instancia válida o no, con respecto a la solución dada en la parte a), justificando.

ci-inv	nom-inv	tel-inv	area-inv	id-depto	nom-depto	ofic-depto
222222	Silvia Rodríguez	9934544	BD	4	AA	150
222222	Silvia Rodríguez	9934544	BD	8	AB	150
545454	Ramiro Suárez	9834200	ISE	6	AA	320

No es una instancia válida.

Justificación:

La instancia no cumple con la dependencia funcional ci-inv, area-inv \rightarrow id-depto, como se puede ver en la primera y segunda tupla de la tabla.

$t1[\{ci-inv, area-inv\}] = t2[\{ci-inv, area-inv\}]$

$t1[\{id-depto\}] \neq t2[\{id-depto\}]$