

Parcial 2023

Nº de Alumno: _____

Apellido y Nombre: _____

BBDD1 Parcial

2da fecha 29/11/2023

Uso interno de la cátedra

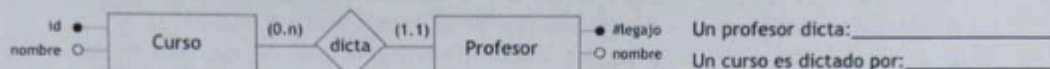
ER	Norm	AR	MySQL

Ayudante: _____

Entidad/Relación

Indique cómo interpreta las cardinalidades del siguiente modelo

(esto se usará para leer las cardinalidades de la solución que usted proponga)



Enunciado:

Se requiere una aplicación para manejar las reservas de las canchas de voley de un complejo recientemente inaugurado.

El complejo cuenta con varias canchas de las cuales se conoce el número y una descripción. De las reservas sobre estas canchas se conoce la fecha y la hora, registrándose además el usuario de la aplicación que la realizó y los dos equipos que participarán del partido. De los equipos se conoce un nombre, fecha de alta y los jugadores que conforman el equipo. Notar que un usuario puede reservar la misma cancha en diferentes fechas.

Por cada equipo participante de una reserva, se registra su resultado (si ganó o perdió). Tener en cuenta que en toda reserva se realiza un único partido.

Todos los usuarios deben registrarse con su dni, nombre, edad, correo electrónico y contraseña. Todos los jugadores son usuarios del sistema, de los que además se registra su grupo sanguíneo. Como la reserva solo puede realizarse por mayores de edad, tener en cuenta que quien hace la reserva no necesariamente es un jugador.

De los jugadores además se guarda una categoría. Esta categoría puede ir cambiando a lo largo del tiempo, por ello se quiere guardar un historial de fechas en que el jugador perteneció a una categoría específica. Tener en cuenta que podría volver a una categoría que haya poseído con anterioridad.

El pago por el uso de la cancha se realiza después del partido y la aplicación permite que cada equipo registrado realice el pago de su parte independientemente del otro. Del pago se debe guardar la fecha, el monto final (por equipo) y el medio de pago utilizado: tarjeta o efectivo. Si el pago es mediante tarjeta se debe registrar si fue mediante crédito o débito; si el pago es en efectivo se registra un descuento.

Actividades:

1. Realizar el modelo E/R
2. Realizar la transformación del modelo de E/R al modelo relacional.

parcial2daFecha-2023.txt

Modelo relacional

Cancha(#cancha, descripcion)

tiene(#reserva, #cancha)

Reserva(#reserva, fecha, hora)

posee(#reserva, #equipo, gano)

Equipo(#equipo, nombre, fecha_alta)

realizada (#reserva, #usuario)

Usuario(#usuario, dni, nombre, edad, contrasenia)

Jugador(#usuario, grupo_sanguineo)

pertenece(#usuario, #categoria)

Categoria(#categoria, nombre)

tiene_2(#historial_categoria, #usuario, #categoria)

Historial_Categoria(#historial_categoria, fecha_inicial, fecha_fin)

conforma (#equipo, #usuario)

realiza(#pago, #equipo, #reserva)

Pago(#pago, tipo_pago, debito, descuento)

Normalizacion

df1) #torneo → nombre_torneo

df2) #equipo → nombre_equipo, estadio_equipo

df3) #equipo, #torneo, año → puesto

df4) #torneo, año, puesto → #equipo

df5) #reglamentacion → descripcion

CC1 = (#torneo, #equipo, año, #auspiciante, #reglamentacion)

CC2 = (#torneo, puesto, año, #auspiciante, #reglamentacion)

Primero nos preguntamos, TORNEOS cumple con la definicion de BCNF?

TORNEOS no se encuentra en BCNF ya que existe al menos una df, ej df1 donde el detereminante {#torneo} no es superclave del esquema. Por lo tanto

particionamos por df1:

t1(#torneo, nombre_torneo)

t2(#torneo, estadio_equipo, nombre_equipo, año, #equipo, puesto, #reglamentacion, descripcion, #auspiciante)

No se pierde informacion ya que t1 interseccion t2 es {#torneo}, clave de t1.

Por validacion simple, en t1 vale df1 y en t2 valen df2, df3 y df4, por lo que no se pierden dfs.

t1 esta en BCNF ya que solo vale df1 en el esquema cuyo determinante {#torneo} es superclave de este esquema.

t2 no se encuentra en BCNF ya que existe al menos una df, ej df2 donde el determinante {#equipo} no es superclave del esquema. Por lo tanto particionamos por df2:

t3(#equipo, nombre_equipo, estadio_equipo)

t4(#torneo, año, #equipo, puesto, #reglamentacion, descripcion, #auspiciante)

No se pierde informacion ya que t3 interseccion t4 es {#equipo}, clave de t3.

Por validacion simple, en t3 vale df2 y en t4 valen df3 y df4, por lo que no se pierden dfs.

t3 esta en BCNF ya que solo vale df2 en el esquema cuyo determinante {#equipo} es superclave de este esquema.

t4 no se encuentra en BCNF ya que existe al menos una df, ej df3 donde el determinante {#equipo, #torneo, año} no es superclave del esquema. Por lo tanto particionamos por df3:

t5(#equipo, #torneo, año, puesto)

t6(#torneo, año, #equipo, #reglamentacion, descripcion, #auspiciante)

No se pierde informacion ya que t5 interseccion t6 es {#equipo, #torneo, año}, clave de t5.

Por validacion simple, en t5 vale df3 y df4, y en t6 vale df5.

Normalización

Algoritmo para analizar la pérdida de dfs.

Res = x

Mientras Res cambia

Para i= 1 to cant_de_ particiones_realizadas

Res = Res U((Res \cap Ri)* \cap Ri)

Algoritmo para encontrar X*

Result:= X

While (hay cambios en result) do

For (cada dependencia funcional Y->Z en F) do

if (Y \subseteq result) then result := result U Z

Se dispone del siguiente esquema que representa los resultados de diferentes torneos de fútbol.

TORNEOS (#torneo, nombre_torneo, año, #equipo, nombre_equipo, estadio_equipo, puesto, #reglamentacion, descripcion, #auspiciante)

- De cada torneo, se conoce su identificador (#torneo, único en el sistema) y un nombre. Un mismo torneo tiene diferentes ediciones, cada edición se realiza en un año determinado y el mismo torneo no puede repetirse el mismo año. En un año pueden realizarse varios torneos.
- Cada edición de un torneo tiene diferentes auspiciantes, identificados por #auspiciante (único en el sistema).
- En cada edición de un torneo participan varios equipos. De cada equipo se conoce su nombre, su estadio y su #equipo, que no se repite para diferentes equipos.
- Cada equipo finaliza una edición de un torneo en un puesto. Dos o más equipos no pueden finalizar en un mismo puesto.
- Además, se conoce un conjunto de reglamentaciones, identificadas por #reglamentación, aplicables a estos torneos.

Aplicar y explicar el proceso de normalización visto en la materia. Considerar que el esquema ya se encuentra en 1FN.

Álgebra Relacional

Dado el siguiente esquema:

EMPLEADO (#empleado, email, nombre, años_antigüedad)

PREMIO (#premio, descripción_premio)

RECIBE_PREMIO (#premio, #empleado, fecha_premio)

SANCION (#sancion, descripción_sancion)

RECIBE_SANCION (#sancion, #empleado, fecha_sancion, gravedad)

Hallar el #empleado, nombre y email de los empleados con más 10 años de antigüedad que hayan recibido todos los premios y que además no hayan recibido sanciones de ningún tipo.

MySQL

Se modificó el esquema anterior, presentado en el ejercicio de Álgebra Relacional, para agregar la siguiente tabla:

AUDITORÍA (#auditoria, #empleado, #sancion, prioridad)

Escriba un trigger para que con cada nueva sanción registrada a un empleado, si esta es de gravedad "Alta" se agregue una nueva tupla a la tabla AUDITORIA donde se registre la correspondiente información con la prioridad "Inmediata".

Algebra

$EMPLEADO_SIN_SANCION \leq \Pi \#empleado (EMPLEADO) - \Pi$
 $\#empleado(RECIBE_SANCION)$

$EMPLEADO_MAS_DIEZ \leq \sigma \text{ años_antiguedad} > 10 (EMPLEADO_SIN_SANCION$
 $|X| EMPLEADO)$

$PREMIOS \leq \Pi \#premio (PREMIO)$

$EMPLEADO_PREMIO \leq (\Pi \#empleado, \#premio RECIBE_PREMIO \%$
 $PREMIOS)$

$\Pi \#empleado, nombre, email (EMPLEADO_MAS_DIEZ |X| EMPLEADO_PREMIO)$

SQL

```
DELIMITER //
```

```
CREATE TRIGGER add_auditory  
AFTER INSERT ON RECIBE_SANCION  
FOR EACH ROW  
BEGIN  
    IF (NEW.gravedad = "Alta") THEN  
        INSERT INTO AUDITORIA (#empleado, #sancion, prioridad)  
        VALUES (NEW.#empleado, NEW.#sancion, "Inmediata")  
    END IF;  
END;  
  
DELIMITER ;
```