



ETSI Informática (Gestión)

Administración de B.D.

Relación de Problemas 1

Normalización y SQL (Definición)

NOTA: Algunos ejercicios tienen errores que debe probar (para ver el tipo y mensaje del error) y corregir.

Se desea crear una BD para controlar el problema de las enfermedades en la ganadería, incluyendo enfermedades como la de las vacas locas (o encefalopatía espongiforme bovina) y la fiebre aftosa. Se tendrán en cuenta las siguientes consideraciones:

- De cada **res** se almacena su especie (BOVINA, PORCINA, OVINA...), un número de control (único para cada una dentro de su especie), raza, sexo, peso, fecha y granja de nacimiento, su padre, su madre, la granja en la que vive y un conjunto de enfermedades padecidas indicando si están ya curadas o no.
- También almacenaremos datos sobre **granjas** (nombre, dirección y ganadero propietario), **ganaderos** (DNI, nombre, teléfono y dirección) y **enfermedades** (nombre, síntomas, tratamiento y sinónimos).
- Cada granja pertenece a un único ganadero, pero un ganadero puede administrar varias granjas distintas.
- Debemos considerar el hecho de que una enfermedad tendrá un nombre principal, pero podrá tener otros nombres secundarios que actuarán como sinónimos.
- Cada síntoma puede tener asociado su gravedad (ligero, normal, grave, crónico...). El tratamiento exacto para una enfermedad dependerá de la gravedad con la que se presenten sus síntomas.
- Puede suponerse que todas las reses son de la misma especie que su padre y que su madre.

- 1.** Según esa información, escriba los atributos de la relación universal (tabla con todos los atributos de la base de datos), y calcule todas las dependencias que existan (funcionales, multivaluadas, jerárquicas y de reunión), entre las cuales deben estar las siguientes:

- a) DF : Especie, NumControl → FechaNacimiento
- b) DMV: NombreEnfermedad → Síntoma | NombreSinónimo
- c) DF : NombreEnfermedad, Síntoma, Gravedad → Tratamiento
- d) DF : Especie, NumControl, NombreEnfermedad → Curada
- e) DF : NombreGranja, DNIPropietario → DirecciónGranja
- f) DF : DirecciónGranja → DNIPropietario

NOTA: Razona si cada una de las dependencias es lógica y tiene un significado correcto.

- 2.** Divida esa relación universal en tablas normalizadas en 5FN, (o justifique porqué no se alcanza esa Forma Normal). Todas las tablas deben tener una llave primaria que no haya sido creada explícitamente para tal fin.

- 3.** Escriba los comandos SQL necesarios para crear dichas tablas, estableciendo todas las restricciones que considere convenientes y poniendo un nombre a cada una de esas restricciones. Entre esas restricciones debe incluir las siguientes:

- a) La granja en la que vive un animal tiene que ser conocida forzosamente.
- b) El sexo de cada res debe ser 'M' o 'H' (macho o hembra) pero puede ser desconocido.
- c) El atributo Curada para una res y una enfermedad debe valer 'S' o 'N' y debe conocerse obligatoriamente.
- d) Los progenitores de cada res pueden ser desconocidos (uno o ambos).
- e) De una res debe conocerse obligatoriamente la granja en la que vive, pero puede ignorarse su granja de nacimiento.
- f) El propietario de una granja debe conocerse siempre.
- g) No se podrá borrar un ganadero si es propietario de alguna granja.
- h) Dos ganaderos no pueden tener el mismo teléfono, aunque este atributo puede ser desconocido.
- i) El peso de cada animal tiene que ser superior a cero pero además, sabemos que el peso de una res bovina tiene que ser superior a 20 kg. y el de una res porcina tiene que ser superior a 3 kg.
- j) Si se borra una enfermedad hay que borrar todos los datos que se tengan de ella (síntomas, sinónimos...).
- k) Igualmente, si se borra una res hay que borrar la información que se tenga de dicha res respecto a sus enfermedades padecidas.
- l) Una enfermedad debe tener como mínimo un síntoma.
- m) Una res no puede ser progenitor de ninguno de sus dos progenitores (ni del resto de sus ancestros).

- 4.** Con las tablas recién creadas y vacías, ¿por qué no puede insertar información de un animal? ¿qué mensaje indica el SGBD si lo intenta?
- 5.** Inserte filas (con **INSERT**) en todas las tablas creadas. Intente incumplir las restricciones impuestas anteriormente y compruebe que no sea posible. Por ejemplo, intente introducir una animal bovino que pese 10 kg.
- 6.** Cambie algunos valores en las filas introducidas (con **UPDATE**) y compruebe que en este tipo de operaciones tampoco se pueden incumplir las restricciones.
- 7.** Borre (con **DELETE**) un ganadero que sea propietario de alguna granja. Observe el mensaje de error que debe recibir, ya que esa operación no debe permitirse.
- 8.** Borre una enfermedad y observe si realmente se ha borrado toda la información relacionada con ella (síntomas, sinónimos...).
- 9.** Borre la tabla de ganaderos con el comando **DROP TABLE Ganaderos;** ¿Por qué no puede borrarla? ¿Permite borrarla con el comando **DELETE Ganaderos?** ¿Se borra si le añade a la sentencia **DROP** anterior la cláusula **CASCADE CONSTRAINTS**? ¿Esa cláusula hace que se borren las granjas cuyos propietarios estaban en la tabla de ganaderos?
- 10.** Repita el ejercicio anterior con la tabla de enfermedades: ¿Hay alguna diferencia?
- 11.** Suponga que introducimos una dependencia funcional: NombreEnfermedad → Especie. ¿Qué cambios requiere la base de datos para reflejar esa dependencia funcional?
- 12.** Ahora, eliminamos la dependencia anterior e introducimos una dependencia multivaluada: NombreEnfermedad →→ Especie | Síntoma.

Esa DMV indica que una enfermedad puede afectar a muchas especies, pero en todas las especies los síntomas son los mismos. ¿Afecta algo esta DMV al conjunto de tablas?

- 13.** Supongamos que queremos añadir la siguiente DF:
NombreEnfermedad, Especie → Síntoma
Observe que esto nos obliga a eliminar la DMV del ejercicio anterior y posiblemente otras que sean contradictorias con la nueva. Según eso, calcule el nuevo elenco de tablas que necesitamos. Razona si esa DF afecta a la DF c) del ejercicio 1.