

Fundamentos de Bases de Datos.

Práctica 9.

Profesora: Dra. Amparo López Gaona

alg@ciencias.unam.mx

Laboratorio: Lic. Carlos Augusto Escalona Navarro

caen@ciencias.unam.mx

25 de noviembre de 2021

Se dan a conocer especificaciones de entrega para la práctica 9.

1. Formas Normales

El objetivo del diseño de las bases de datos relacionales es la generación de un conjunto de esquemas relacionales que nos permita almacenar la información sin redundancias, pero que también nos permita recuperar fácilmente esa información. Un enfoque es el diseño de esquemas que se encuentren en una forma normal adecuada para la solución del caso de uso.

1.1. Dependencias Funcionales

Una dependencia funcional es un tipo de restricción o regla de negocio que constituye una generalización del concepto de clave. Permiten expresar hechos sobre la empresa que se modela con la base de datos. Desempeñan un papel fundamental en la diferenciación entre los buenos diseños de bases de datos y los malos.

Considérese el esquema de una relación R y sean $\alpha \subseteq R$ y $\beta \subseteq R$. La dependencia funcional

$$\alpha \rightarrow \beta$$

se cumple para el esquema R si, en cualquier relación legal $r(R)$, para todos los pares de tuplas $t1$ y $t2$ de r tales que $t1[\alpha] = t2[\alpha]$, también ocurre que $t1[\beta] = t2[\beta]$.

Las dependencias funcionales se utilizarán de dos maneras:

1. Para probar las relaciones y ver si son legales según un conjunto dado de dependencias funcionales. Si una relación r es legal según el conjunto F de dependencias funcionales, se dice que r satisface F .

2. Para especificar las restricciones del conjunto de relaciones legales. Así, sólo habrá que preocuparse por las relaciones que satisfagan un conjunto dado de dependencias funcionales. Si uno desea restringirse a las relaciones del esquema R que satisfagan el conjunto F de dependencias funcionales, se dice que F se cumple en R .

Se dice que algunas dependencias funcionales son triviales porque las satisfacen todas las relaciones. Por ejemplo, $A \rightarrow A$ la satisfacen todas las relaciones que impliquen al atributo A . La lectura literal de la definición de dependencia funcional deja ver que, para todas las tuplas $t1$ y $t2$ tales que $t1[A] = t2[A]$, se cumple que $t1[A] = t2[A]$. De manera parecida, $AB \rightarrow A$ la satisfacen todas las relaciones que impliquen al atributo A . En general, una dependencia funcional de la forma $\alpha \rightarrow \beta$ es trivial si $\beta \subseteq \alpha$.

Dado un esquema relacional R , una dependencia funcional f de R está implicada lógicamente por un conjunto de dependencias funcionales F de R si cada ejemplar de la relación $r(R)$ que satisface F satisface también f . Es decir, dado un conjunto F de dependencias funcionales, se puede probar que se cumplen otras dependencias funcionales determinadas. Se dice que esas dependencias funcionales están implicadas lógicamente por F .

1.2. Axiomas de Armstrong

Los axiomas, o reglas de inferencia, proporcionan una técnica más sencilla para el razonamiento sobre las dependencias funcionales. Se pueden utilizar las tres reglas siguientes para hallar las dependencias funcionales implicadas lógicamente. Aplicando estas reglas repetidamente, se puede hallar todo $F+$, dado F .

- Regla de la reflexividad. Si α es un conjunto de atributos y $\beta \subseteq \alpha$, entonces se cumple que $\alpha \rightarrow \beta$.
- Regla de la aumentatividad. Si se cumple que $\alpha \rightarrow \beta$ y γ es un conjunto de atributos, entonces se cumple que $\gamma\alpha \rightarrow \gamma\beta$.
- Regla de la transitividad. Si se cumple que $\alpha \rightarrow \beta$ y también se cumple que $\beta \rightarrow \gamma$, entonces se cumple que $\alpha \rightarrow \gamma$.
- Regla de la unión. Si se cumple que $\alpha \rightarrow \beta$ y que $\alpha \rightarrow \gamma$, entonces se cumple que $\alpha \rightarrow \beta\gamma$.
- Regla de la descomposición. Si se cumple que $\alpha \rightarrow \beta\gamma$, entonces se cumple que $\alpha \rightarrow \beta$ y que $\alpha \rightarrow \gamma$.
- Regla de la pseudotransitividad. Si se cumple que $\alpha \rightarrow \beta$ y que $\gamma\beta \rightarrow \delta$, entonces se cumple que $\alpha\gamma \rightarrow \delta$.

Los axiomas de Armstrong son correctos porque no generan dependencias funcionales incorrectas. Son completos, porque, para un conjunto dado F de dependencias funcionales, permiten generar todo F^+ .

Sea α un conjunto de atributos. Al conjunto de todos los atributos determinados funcionalmente por α bajo un conjunto F de dependencias funcionales se le denomina cierre de α bajo F ; se denota mediante α^+ .

```
F+ = F
repeat
for each dependencia funcional f de F+
aplicar las reglas de reflexividad y de aumentatividad a f
añadir las dependencias funcionales resultantes a F+
for each pareja de dependencias funcionales f1 y f2 de F+
if f1 y f2 pueden combinarse mediante la transitividad
añadir la dependencia funcional resultante a F+
until F+ no cambie más
```

1.3. Primera Forma Normal

Se dice que el esquema de una relación R está en la primera forma normal (1FN) si los dominios de todos los atributos de R son atómicos.

1.4. Forma Normal de Boyce-Codd

Mediante las dependencias funcionales se pueden definir varias formas normales que representan «buenos» diseños de bases de datos. Una de las formas normales más deseables que se pueden obtener es la forma normal de Boyce-Codd (FNBC). Un esquema de relación R está en FNBC respecto a un conjunto de dependencias funcionales F si, para todas las dependencias funcionales de F^+ de la forma $\beta \rightarrow \alpha$, donde $\alpha \subseteq R$ y $\beta \subseteq R$, se cumple al menos una de las siguientes condiciones:

- $\alpha \rightarrow \beta$ es una dependencia funcional trivial (es decir, $\beta \subseteq \alpha$).
- α es una super llave del esquema R .

1.5. Tercera Forma Normal

FNBC exige que todas las dependencias no triviales sean de la forma $\alpha \rightarrow \beta$ donde α es una super llave. 3FN relaja ligeramente esta restricción permitiendo dependencias funcionales no triviales cuya parte izquierda no sea una super llave. Un esquema de relación R está en tercera forma normal (3FN) respecto a un conjunto F de dependencias funcionales si, para todas las dependencias funcionales de F^+ de la forma $\alpha \rightarrow \beta$, donde $\alpha \subseteq R$ y $\beta \subseteq R$, se cumple al menos una de las siguientes condiciones:

1. $\alpha \rightarrow \beta$ es una dependencia funcional trivial.
2. α es una super llave de R .

3. Cada atributo A de $\beta - \alpha$ está contenido en alguna clave candidata de R

2. Actividad

Para esta práctica vamos a trabajar con un esquema llamado geográfico, el cual contiene la información de los estados, distritos federales, distritos locales, municipios y secciones que existen en un país.

- Lo primero que vamos a realizar es hacer el restore con el archivo llamado geográfico_sin_normalizar.backup.
- Una vez recreado este archivo procederemos a realizar las siguientes consultas, para la solución de estas vamos a tomar el tiempo que se tarda en resolver la ejecución el sistema manejador de base de datos:
 1. Obtener el nombre y la abreviatura de cada uno de los estados que existan en la tabla de geografico_completo, no deben aparecer estados repetidos.
 2. Obtener el nombre del estado y cabecera distrital federal, que exista en la tabla de geografico_completo, no deben aparecer cabeceras distritales repetidas.
- Realizaremos 3 inserciones, 3 actualizaciones y 1 borrado:
 1. Insertar un estado llamado 'mi estado'.
 2. Insertar 2 distritos federales, distritos locales y municipios diferentes.
 3. Insertar 2 secciones para cada uno de los distritos federales.
 4. Actualizar el nombre del estado de 'AGUASCALIENTES' para que ahora se llame 'AGUASCALIENTES NUEVO'.
 5. Actualizar una sección para el estado de 'TAMAULIPAS' con cabecera distrital federal 'RIO BRAVO'.
 6. Actualizar los registros de las primeras dos inserciones para que no tengan ningún campo en null.
 7. Borrar todos los registros repetidos en tu tablas geografico_completo.
- Realizar un reporte donde hagas un análisis sobre la base de datos no normalizada. En ese mismo reporte deberás desarrollar cómo es que se aplica la **tercera forma normal**, por lo que debes poner todos los pasos hasta obtener tu nueva base de datos normalizada, para esto cuentas con las siguientes dependencias funcionales:
 1. nombre_estado \rightarrow abreviatura.
 2. cabecera_distrital_federal \rightarrow nombre_estado.

3. cabecera_distrital_local → nombre_estado.
 4. nombre_municipio → nombre_estado.
 5. nombre_estado, sección, tipo → cabecera_distrital_federal, cabecera_distrital_local, nombre_municipio.
- Dentro del reporte que se pide en el punto anterior, deberás agregar el **diagrama relacional** de la base de datos no normalizado y el nuevo diagrama con tu base de datos normalizada.
 - Una vez terminada tu normalización deberás crear las tablas resultantes de esta por medio de sentencias DDL.
 - Crear todas las llaves primarias y foráneas de las tablas por medio de **constraint**.
 - **Punto extra:** Cada una de estas tablas resultantes de la normalización deberán tener la información correspondiente de la tabla geografico_completo.

3. Entregables

Para la entrega de los dos primeros puntos deberás crear un archivo llamado practica9.sql, el cual contendrá la sintaxis DML que utilizaste para la solución de estos.

Para los puntos tres y cuatro deberás entregar un reporte llamado normalizacion.pdf, el cual contendrá la solución de estos puntos.

Para los últimos dos puntos deberás entregar un archivo llamado normalizacion.sql, el cual contendrá la solución de estos puntos.