

UNIVERSIDAD NACIONAL
AUTÓNOMA DE MÉXICO

FUNDAMENTOS DE BASES DE
DATOS

Tarea 3: Modelo Relacional

Almeida Rodríguez Jerónimo
418003815

Figueroa Sandoval Gerardo Emiliano
315241774

Ibarra Moreno Gisselle
315602193



1. Preguntas de Repaso.

- (a) ¿Qué es una **relación** y qué características tiene?

Es un modelo de organización y gestión de bases de datos consistente en el almacenamiento de datos en tablas compuestas por filas, o tuplas, y columnas o campos. Surge como solución a la creciente variedad de los datos que integran las data warehouses y podemos resumir el concepto como una colección de tablas (relaciones).

Estas tablas, pueden ser construídas de diversas maneras:

- Creando un conjunto de tablas iniciales y aplicar operaciones de normalización hasta conseguir el esquema más óptimo.
- Convertir el diagrama e-r a tablas y posteriormente aplicar también operaciones de normalización hasta conseguir el esquema óptimo.

- (b) ¿Qué es un **esquema de relación**?

Es el que contiene la definición de una estructura (generalmente relaciones o tablas de una base de datos), es decir, determina la identidad de la relación y qué tipo de información podrá ser almacenada dentro de ella; en otras palabras, el esquema contiene los metadatos de la relación. Todo esquema constará de:

- Nombre de la relación (su identificador).
- Nombre de los atributos (o campos) de la relación y sus dominios; el dominio de un atributo o campo define los valores permitidos para el mismo, equivalente al tipo de dato por ejemplo character, integer, date, string..

- (c) ¿Qué es una **llave primaria**?, ¿qué es una **llave candidata**?, ¿qué es una **llave mínima**?, ¿qué es una **super llave**?

- **Llave primaria:** Una llave primaria es un conjunto de uno o más atributos de una tabla, que tomados colectivamente nos permiten identificar un registro como único, es decir, en una tabla podemos saber cual es un registro en específico sólo con conocer la llave primaria.
- **Llave candidata:** Una llave candidata de una relación es el atributo o conjunto de atributos que tienen la propiedad de identificar unívocamente a una tupla dentro de la relación. Las llaves constituyen el mecanismo de direccionamiento a nivel de tuplas básico en un sistema relacional, es decir, es el único modo, garantizado por el sistema, de localizar alguna tupla específica.

- **Llave mínima:**

- **Super llave:**

- (d) ¿Qué restricciones impone una **llave primaria** y una **llave foránea** al modelo de datos relacional?
- (e) Investiga cómo se traducen las **categorías** (presentes en el **modelo E/R**) al **modelo relacional**. Proporciona un ejemplo.

2. Modelo Relacional.

3. Lectura.

1. Regla de Información:

Toda la Información en la base de datos está representada de una manera única, cómo valores en tablas.

En general, toda la información se guarda en tablas.

2. Regla del Acceso Garantizado:

Está garantizado que cada dato (valor atómico) pueda ser accedido lógicamente por medio de una combinación de nombre de tabla, valor de la llave primaria y el nombre de la columna.

Esta regla lo que busca es garantizar el acceso a la información de manera única por medio de un conjunto de “coordenadas” que, basadas en el nombre de la tabla, la llave primaria y el nombre de la columna, permite que cada dato pueda recuperarse de manera única.

3. Tratamiento Sistemático de los Valores 'NULL':

Los valores NULL son usados en SMBDR para representar información faltante o desconocida de una manera sistemática, independientemente del tipo de dato. Son distintos del carácter de cadena vacía, de caracteres blancos y de cualquier número.

Implementa el uso de valores NULL, que en general significan que el valor es desconocido. Usualmente se trata cómo operar el vacío (\emptyset) en teoría de conjuntos: cualquier cosa operada con NULL devuelve NULL; aunque en algunos casos, si se intenta concatenar con una cadena, devuelve la nueva cadena.

4. Catálogo Dinámico en Línea Basado en el Modelo Relacional:

En el nivel lógico, la descripción de la base de datos está representada al mismo nivel que los datos ordinarios. De esta manera, los

usuarios autorizados pueden usar el mismo lenguaje relacional que se usa para datos regulares.

En general requiere que el sistema sea autodescriptivo.

5. Regla del Sublenguaje de Datos Comprensivos:

Un sistema relacional puede soprtar varios lenguajes de programación y modos de uso terminal, pero al menos uno de ellos debe tener expresiones que sean expresables por medio de una sitáxis bien definida y puede soportar lo siguiente:

- *Definición de datos.*
- *Visibilización de la definición.*
- *Manipulación de datos*
- *Restricciones de integridad.*
- *Autorizaciones.*
- *Bordes de transacciones (inicio, commit, etc.).*

Esta restricción requiere que el sistema tenga un lenguaje que pueda manipular (recuperar, insertar, borrar, etc.) la información dentro de la base de datos.

6. Regla de la vista actualizada:

Todas las vistas que se pueden actualizar las puede actualizar el sistema.

Esta regla requiere que los distintos usuarios puedan ver la estructura de las bases de datos.

Es una de las reglas más difíciles de implementar, de tal manera que ningún SMBD la satisface completamente. En SQL Server, las vistas se actualizan solamente si no se actualiza más de una tabla en un momento dado.

7. Insert, Delete y Actualizar a Alto Nivel:

La habilidad de manejar una relación base o derivada cómo un único operando no solo al recuperar la información sino también para insertar, borrar y actualizar información.

Esta regla establece que la información debe ser manipulada cómo conjuntos, lo que ayuda a garantizar la consistencia de la base de datos.

8. Independencia Física de Datos:

Los programas de aplicación y actividades terminales se mantienen funcinando cuándo hay cambios en la representación de almacenamiento o métodos de acceso.

El objetivo es que el almacenamiento de los datos sea independiente del programa que los requiere pero asegurando que pueden ser accedidos de la misma manera.

9. Independencia Lógica de Datos:

Los programas de aplicación y actividades terminales se mantienen lógicamente funcionales cambios que preserven la información se haga en alguna tabla.

Establece que la manera específica de acceder a la base de datos no debe afectar la habilidad del usuario de manipular los datos.

10. Independencia de Integridad:

Las restricciones de integridad específicas de una base relacional deben poder ser definidas en el lenguaje de el lenguaje relacional y deben poder ser almacenadas en el catálogo, no en los programas de aplicación. Protege la información de datos inválidos.

Esta regla intenta mantener la integridad de la base de datos, particularmente en lo relacionado a todo aquello que se puede acceder por medio de una llave primaria o foránea.

11. Independencia de Distribución:

El lenguaje del SMBD debe permitir que los programas de aplicación y actividades terminales permanezcan lógicamente funcionales y accesibles ya sea que los datos estén físicamente centralizados o distribuidos.

El objetivo de esta regla es mantener una base de datos “unificada” aunque la información se encuentre físicamente repartida en dispositivos diferentes. Esto significa que a pesar de estar separada cumple (idealmente) con las propiedades que una base de datos debe cumplir.

12. Regla de No-Subversión:

Si un sistema relacional permite el uso de algún lenguaje de programación de un nivel más bajo, este lenguaje no puede ser usado para traspasar las reglas de integridad y restricciones establecidas por el lenguaje relacional.

El objetivo es que no haya manera de acceder a la información fuera de la establecida por el SDBDR. Esto con el objetivo de mantener la integridad del sistema.

La importancia de estas reglas radica en mantener un sistema que a lo largo de su ejecución sea capaz de mantener consistencia en la información que almacena además de establecer una manera de manipular y acceder a la información de manera consistente y confiable.

Cómo establece el artículo, tanto la regla 6 cómo la 12 no se pueden cumplir enteramente por los SMDBR actuales.