

Normalización en Bases de Datos

UPEMOR Base de Datos

Introducción a la Normalización

- **Definición:**

La normalización es el proceso de organizar los datos en una base de datos para reducir la redundancia y dependencia de datos.

- **Objetivo:**

Crear un diseño eficiente y libre de anomalías, haciendo que las tablas sean más fáciles de mantener, actualizar y consultar.

Problemas que Resuelve la Normalización

- **Redundancia de datos:**

Almacenar la misma información en múltiples lugares, lo que puede generar inconsistencias.

- **Anomalías de actualización:**

Problemas al modificar, insertar o eliminar datos debido a la duplicación.

- **Incoherencia de los datos:**

Cuando los mismos datos se encuentran en diferentes lugares y no se actualizan al mismo tiempo.

Las Formas Normales (FN)

Las formas normales son reglas o estándares para estructurar los datos de manera eficiente. Existen **cinco formas normales principales**:

1. **Primera Forma Normal (1FN)**
2. **Segunda Forma Normal (2FN)**
3. **Tercera Forma Normal (3FN)**
4. **Cuarta Forma Normal (4FN)**
5. **Quinta Forma Normal (5FN)**

En esta presentación, nos enfocaremos en las 1FN, 2FN y 3FN.

Primera Forma Normal (1FN)

Regla:

Una tabla está en 1FN si:

1. Todos los atributos contienen sólo valores atómicos (es decir, no se permiten listas ni conjuntos de valores).
2. Cada valor en una columna debe ser único (no puede haber filas duplicadas).

Ejemplo de Violación de 1FN:

Antes de aplicar 1FN:

ID	Nombre	Teléfonos
1	Juan	1234, 5678
2	María	9876, 5432

Primera Forma Normal (1FN)

Problema:

- La columna **Teléfonos** tiene múltiples valores por fila, lo cual no cumple con la regla de valores atómicos.

Después de aplicar 1FN (separa los valores múltiples en filas diferentes):

Después de 1FN:

ID	Nombre	Teléfono
1	Juan	1234
1	Juan	5678
2	María	9876
2	María	5432

Primera Forma Normal (1FN)

Explicación:

- Ahora, la tabla cumple con 1FN porque cada celda contiene un solo valor atómico.
- Los valores duplicados en la columna **ID** son ahora coherentes con la estructura de la base de datos, ya que **Juan y María** tiene dos números de teléfono, y esto está reflejado con dos filas separadas pero con el mismo **ID**.

Segunda Forma Normal (2FN)

Regla:

Una tabla está en 2FN si:

1. Está en 1FN.
2. Todos los atributos no clave dependen completamente de la clave primaria (sin dependencias parciales).

Ejemplo:

Antes de 2FN:

ID	Producto	Cantidad	Precio
1	Producto1	10	20
2	Producto2	5	30

Segunda Forma Normal (2FN)

Dependencia parcial: **Producto** depende de la clave parcial **ID**, ya que el precio no depende del ID, sino del producto.

Después de 2FN (dividiendo en dos tablas):

- Tabla 1:

ID	Producto	Cantidad
1	Producto1	10
2	Producto2	5

- Tabla 2:

Producto	Precio
Producto1	20
Producto2	30

Tercera Forma Normal (3FN)

Regla:

Una tabla está en 3FN si:

1. Está en 2FN.
2. No tiene dependencias transitivas (es decir, los atributos no clave dependen directamente de la clave primaria, no a través de otros atributos).

Ejemplo:

Antes de 3FN:

ID	Nombre	Ciudad	Código Postal
1	Juan	Madrid	28001
2	María	Barcelona	08001

Tercera Forma Normal (3FN)

Dependencia transitoria: **Ciudad** depende de **Código Postal**.

Después de 3FN (dividiendo en dos tablas):

- Tabla 1:

ID	Nombre	Ciudad
1	Juan	Madrid
2	María	Barcelona

- Tabla 2:

Ciudad	Código Postal
Madrid	28001
Barcelona	08001

Cuarta y Quinta Forma Normal (4FN y 5FN)

- **Cuarta Forma Normal (4FN):**

Una tabla está en 4FN si no tiene dependencias multivaluadas, es decir, no puede contener más de un conjunto de valores relacionados con un solo atributo.

- **Quinta Forma Normal (5FN):**

Una tabla está en 5FN si no tiene dependencias de unión. Es decir, todos los datos de la tabla deben ser independientes de las uniones de otras tablas.

Nota:

Estas formas normales son menos comunes en bases de datos tradicionales, pero son importantes en bases de datos muy complejas.

Beneficios de la Normalización

- **Reducción de la redundancia:**
Minimiza la duplicación de datos.
- **Mejora de la consistencia:**
Asegura que los datos sean consistentes y no contradictorios.
- **Optimización del almacenamiento:**
Reduce el uso de espacio al eliminar duplicados.
- **Facilita el mantenimiento:**
Hace que sea más fácil modificar y actualizar los datos sin causar inconsistencias.

Desventajas de la Normalización

- **Mayor complejidad de consultas:**

Las consultas pueden volverse más complejas debido a las relaciones entre tablas.

- **Desempeño en consultas:**

Las bases de datos altamente normalizadas pueden experimentar un rendimiento más bajo en ciertas consultas debido a la necesidad de realizar más uniones.

Conclusión

- La normalización es una técnica clave en el diseño de bases de datos que ayuda a reducir la redundancia y las anomalías.
- Es importante encontrar un equilibrio entre normalización y desempeño, ya que en algunos casos, la desnormalización puede ser útil para mejorar la eficiencia de las consultas.