

Chuleta de Normalización Relacional (1NF, 2NF, 3FN)

El objetivo de normalizar es eliminar la redundancia y evitar problemas al insertar, modificar o borrar datos (anomalías).

Ejemplo Práctico: Normalización de la tabla "VENDEDOR-VENTAS"

La Tabla Original (Sin Normalizar)

Partimos de una tabla que almacena la información de las ventas, incluyendo los datos del vendedor en cada fila. Esto causa una gran redundancia: los datos del vendedor (calle, ciudad) se repiten cada vez que realiza una venta.

| dniVendedor | calle | ciudad | codArticulo | cantidad |
|-------------|-----------|--------|-------------|----------|
| 123 | La Cámara | Avilés | 1 | 10 |
| 123 | La Cámara | Avilés | 2 | 3 |
| 456 | Uría | Oviedo | 1 | 4 |
| 678 | Begoña | Gijón | 3 | 7 |

VENDEDOR-VENTAS (dniVendedor, calle, ciudad, codArticulo, cantidad)

Fase 1: Primera Forma Normal (1FN)

El Objetivo

Identificar la **Clave Primaria (PK)** y asegurar que todos los valores de la tabla sean **atómicos**.

Análisis

La tabla ya está en 1FN. La clave primaria que identifica unívocamente cada venta (un vendedor vendiendo un artículo) es la combinación de `(dniVendedor, codArticulo)`.

Resultado (Tabla en 1FN)

VENTAS_1FN (dniVendedor, codArticulo, calle, ciudad, cantidad)

Fase 2: Segunda Forma Normal (2FN)

El Objetivo

Eliminar las **dependencias parciales**. Cada atributo no clave debe depender de la **clave primaria completa**.

Análisis

Buscamos atributos que no necesiten la PK completa `(dniVendedor, codArticulo)` para ser determinados:

- calle y ciudad: Para saber la calle o la ciudad de un vendedor, solo necesitamos su `dniVendedor`. No importa qué artículo haya vendido.
→ **Dependencia Parcial: {dniVendedor} → calle, ciudad**
- cantidad: Para saber la cantidad vendida, necesitamos saber qué vendedor (`dniVendedor`) vendió qué artículo (`codArticulo`). Depende de la clave completa. ¡Este atributo está bien!

Resultado (División de tablas)

Extraemos los datos del vendedor (`calle`, `ciudad`) a su propia tabla, ya que dependen parcialmente de la clave.

VENEDORES (dniVendedor, calle, ciudad)

VENTAS (dniVendedor*, codArticulo*, cantidad)

Fase 3: Tercera Forma Normal (3FN)

El Objetivo

Eliminar las **dependencias transitivas** (atributos no clave que dependen de otros atributos no clave).

Análisis

Revisamos las dos nuevas tablas que hemos creado:

- Tabla VENEDORES:** Los atributos `calle` y `ciudad` dependen directamente de la PK (`dniVendedor`). No hay dependencias transitivas. ✓ **Cumple 3FN.**
- Tabla VENTAS:** El único atributo no clave (`cantidad`) depende directamente de la PK (`dniVendedor, codArticulo`). No hay dependencias transitivas. ✓ **Cumple 3FN.**

El ejercicio original sugiere una dependencia transitiva entre `calle` y `ciudad`. Sin embargo, para que fuera transitiva, `ciudad` debería depender de `calle`, lo cual no es necesariamente cierto (dos calles "Mayor" pueden estar en ciudades distintas). Por lo tanto, el paso a 2FN ya soluciona el problema principal de redundancia.

Resultado Final (Tablas en 3FN)

El modelo final, descompuesto tras la 2FN, ya se encuentra en 3FN. Hemos separado claramente la información del vendedor de la información de la venta.

VENEDORES (dniVendedor, calle, ciudad)

// Guarda la información única de cada vendedor.

VENTAS (dniVendedor, codArticulo, cantidad)

// Tabla que relaciona vendedores con artículos y guarda la cantidad de cada venta.