



INVESTIGACIÓN SOBRE NORMALIZACIÓN DE BASES DE DATOS

ALUMNA: STEFANY AUSENCIO LOPEZ

GRUPO: 3TIDSM-G1

MATERIA: BASE DE DATOS

PROFESOR: ING. JOSE LUIS HERRERA GALLARDO

FECHA: 16/07/2025

1. Introducción

La normalización de bases de datos es una técnica utilizada en el diseño de bases de datos relacionales con el objetivo de **eliminar la redundancia, mejorar la integridad y optimizar la estructura** de las tablas. Este proceso se basa en una serie de **formas normales**, que son reglas que se aplican paso a paso para mejorar el modelo de datos.

Las formas normales más importantes son:

- **Primera Forma Normal (1FN)**: organiza los datos en valores atómicos.
- **Segunda Forma Normal (2FN)**: elimina las dependencias parciales.
- **Tercera Forma Normal (3FN)**: elimina las dependencias transitivas.

Esta investigación explica cada una de ellas con **el mismo ejemplo de pedidos de productos**, para observar cómo mejora la base de datos en cada etapa.

2. Desarrollo

2.1. Primera Forma Normal (1FN)

¿Qué condiciones debe cumplir una tabla para estar en 1FN?

Una tabla está en **Primera Forma Normal** cuando:

- Todos los campos contienen **valores atómicos** (indivisibles).
- No hay listas ni múltiples valores en una misma celda.
- Cada columna tiene un solo tipo de dato y cada fila representa una entidad única.

¿Por qué aplicar la 1FN?

Porque facilita el acceso, filtrado y manipulación de los datos. Si una celda contiene varios valores, no se puede consultar ni ordenar fácilmente.

Ejemplo de una tabla que no cumple con 1FN:

ID_Pedido	Cliente	Productos	Cantidades
1	Lucía	Lápiz, Borrador	2, 1
2	Juan	Cuaderno, Lápiz, Borrador	1, 3, 2

Problemas:

- Las columnas **Productos** y **Cantidades** contienen varios valores en una sola celda.
- Es difícil saber cuántos lápices se vendieron o hacer una suma total por producto.

Transformación para cumplir con 1FN:

ID_Pedido	Cliente	Producto	Cantidad
1	Lucía	Lápiz	2
1	Lucía	Borrador	1
2	Juan	Cuaderno	1
2	Juan	Lápiz	3
2	Juan	Borrador	2

¿Por qué ahora cumple con 1FN?

Cada celda contiene un solo valor atómico. Ahora es posible realizar búsquedas y operaciones sobre productos específicos sin complicaciones.

2.2. Segunda Forma Normal (2FN)

¿Qué condiciones debe cumplir una tabla para estar en 2FN?

Una tabla está en **Segunda Forma Normal** si:

- Cumple con la **1FN**.
- Todos los campos no clave **dependen completamente** de la clave primaria.
- **No existen dependencias parciales** (un campo depende solo de una parte de una clave compuesta).

¿Qué es una dependencia parcial?

Una **dependencia parcial** ocurre cuando un campo depende de **solo una parte de una clave compuesta**. Esto genera **repetición innecesaria** y puede provocar **inconsistencias** al actualizar los datos.

Ejemplo de transformación desde 1FN a 2FN:

Tabla actual (en 1FN):

ID_Pedido	Cliente	Producto	Cantidad
1	Lucía	Lápiz	2
1	Lucía	Borrador	1
2	Juan	Cuaderno	1
2	Juan	Lápiz	3
2	Juan	Borrador	2

Clave primaria: (ID_Pedido, Producto)

Problema: El campo **Cliente** depende solo de **ID_Pedido**, no de toda la clave eso es la **dependencia parcial**.

Transformación a 2FN:

Tabla Pedidos:

ID_Pedido	Cliente
1	Lucía
2	Juan

Tabla Detalle_Pedido:

ID_Pedido	Producto	Cantidad
1	Lápiz	2
1	Borrador	1
2	Cuaderno	1
2	Lápiz	3
2	Borrador	2

¿Por qué ahora cumple con 2FN?

Se separó la información que dependía parcialmente (Cliente) en una tabla distinta. Ahora, todos los campos dependen completamente de su clave primaria.

2.3. Tercera Forma Normal (3FN)

¿Qué condiciones debe cumplir una tabla para estar en 3FN?

Una tabla está en **Tercera Forma Normal** si:

- Cumple con la **2FN**.
- No existen **dependencias transitivas** entre campos no clave.
- Todos los atributos no clave deben depender **solo de la clave primaria**.

¿Qué es una dependencia transitiva?

Una **dependencia transitiva** se produce cuando:

- Campo A (clave primaria) determina a campo B (no clave), y
 - Campo B determina a campo C (otro campo no clave).
- Esto implica que **campo C depende indirectamente** de la clave primaria.

Ejemplo de transformación desde 2FN a 3FN:

Supongamos que se agrega la columna **PrecioUnitario**:

ID_Pedido	Producto	Cantidad	PrecioUnitario
1	Lápiz	2	5.00
1	Borrador	1	3.00
2	Cuaderno	1	20.00
2	Lápiz	3	5.00
2	Borrador	2	3.00

Problema: El **PrecioUnitario** depende del **Producto**, no del pedido.

Eso crea una **dependencia transitiva**: ID_Pedido > Producto > PrecioUnitario

Transformación a 3FN:

Tabla Productos:

Producto	PrecioUnitario
Lápiz	5.00
Borrador	3.00
Cuaderno	20.00

Tabla Detalle_Pedido (ajustada):

ID_Pedido	Producto	Cantidad
1	Lápiz	2
1	Borrador	1
2	Cuaderno	1
2	Lápiz	3
2	Borrador	2

¿Por qué ahora cumple con 3FN?

El campo **PrecioUnitario** ya no depende indirectamente de la clave.

Cada tabla contiene atributos que dependen **directamente y exclusivamente** de su clave primaria.

3. Conclusión

El proceso de normalización permite organizar correctamente la información en una base de datos para que:

- No haya **datos repetidos innecesarios**.
- Se eviten **errores al insertar, eliminar o modificar**.
- Las tablas estén organizadas de forma **eficiente y lógica**.

Con cada forma normal aplicada:

- **1FN**: se garantiza la atomicidad de los datos.
- **2FN**: se eliminan dependencias parciales.
- **3FN**: se eliminan dependencias transitivas.

A través del ejemplo utilizado (pedidos con productos), se pudo observar cómo una tabla mal estructurada puede transformarse en un modelo de datos limpio, escalable y fácil de mantener.

4. Bibliografías

1. Helenclu. (s. f.). *Descripción de la normalización de la base de datos - Microsoft 365 Apps*. Microsoft Learn. <https://learn.microsoft.com/es-es/office/troubleshoot/access/database-normalization-description>
2. Castañeda, M. P. (s. f.). *Normalización de bases de datos*. https://programas.cuaed.unam.mx/repositorio/moodle/pluginfile.php/872/mod_resource/content/7/Contenido/index.html
3. Más Tecnología. (2024, 13 marzo). *⚠ NORMALIZACIÓN de bases de datos (1FN, 2FN y 3FN) | Videotutorial* [Video]. YouTube. <https://www.youtube.com/watch?v=m7kpSO6kqY8>
4. TodoCode. (2025, 26 junio). *NORMALIZACIÓN en BASES DE DATOS / MASTER CLASSS ¡Con Ejercicio!* [Video]. YouTube. <https://www.youtube.com/watch?v=kvt2wE-q-yY>