

# Normalización de Bases de Datos

Gestión y Arquitectura de Datos, Universidad de San Andrés

Si encuentran algún error en el documento o hay alguna duda, mandenme un mail a [rodriguezr@udesa.edu.ar](mailto:rodriguezr@udesa.edu.ar) y lo revisamos.

## 1. Introducción

La normalización es un proceso de diseño de bases de datos que reduce la redundancia y las anomalías de actualización. Se aplica a través de una serie de formas normales, cada una construyendo sobre la anterior.

## 2. Primera Forma Normal (1FN)

Una tabla está en Primera Forma Normal si:

- Cada columna contiene valores atómicos (no divisibles)
- No hay grupos repetitivos
- Existe una clave primaria

### 2.1. Ejemplo - Violación de 1FN

Tabla: Estudiantes (No normalizada)

ID	Nombre	Teléfonos	Materias
1	Juan	1234567, 7654321	Matemática, Física
2	María	2345678	Química, Biología, Física

### 2.2. Solución en 1FN

Tabla: Estudiantes

ID	Nombre	Teléfono
1	Juan	1234567
1	Juan	7654321
2	María	2345678

Tabla: EstudiantesMaterias

ID_Estudiante	Materia
1	Matemática
1	Física
2	Química
2	Biología
2	Física

### 3. Segunda Forma Normal (2FN)

Una tabla está en Segunda Forma Normal si:

- Está en 1FN
- Todos los atributos no clave dependen completamente de la clave primaria

#### 3.1. Ejemplo - Violación de 2FN

Tabla: VentasProductos (en 1FN pero no en 2FN)

ID_Venta	ID_Producto	Cantidad	Precio_Producto	Categoría_Producto
1	P1	2	100	Electrónica
1	P2	1	200	Hogar
2	P1	3	100	Electrónica

Aquí, Precio\_Producto y Categoría\_Producto solo dependen de ID\_Producto, no de la clave completa (ID\_Venta, ID\_Producto).

#### 3.2. Solución en 2FN

Tabla: Ventas

ID_Venta	ID_Producto	Cantidad
1	P1	2
1	P2	1
2	P1	3

Tabla: Productos

ID_Producto	Precio	Categoría
P1	100	Electrónica
P2	200	Hogar

## 4. Tercera Forma Normal (3FN)

Una tabla está en Tercera Forma Normal si:

- Está en 2FN
- No hay dependencias transitivas (atributos no clave que dependen de otros atributos no clave)

### 4.1. Ejemplo - Violación de 3FN

Tabla: Empleados (en 2FN pero no en 3FN)

ID_Empleado	Departamento	ID_Jefe	Nombre_Jefe
1	Ventas	J1	Ana López
2	IT	J2	Pedro Gómez
3	Ventas	J1	Ana López

Aquí, Nombre\_Jefe depende transitivamente de ID\_Empleado a través de ID\_Jefe.

### 4.2. Solución en 3FN

Tabla: Empleados

ID_Empleado	Departamento	ID_Jefe
1	Ventas	J1
2	IT	J2
3	Ventas	J1

Tabla: Jefes

ID_Jefe	Nombre
J1	Ana López
J2	Pedro Gómez

## 5. Ejemplo - TravelPro

### 5.1. Consigna

La empresa TravelPro gestiona reservas de clientes en hoteles internacionales. Actualmente, la información está almacenada en una única tabla con la siguiente estructura:

- IDCliente
- ClienteNombre
- ClienteEmail \*
- IDHotel
- HotelNombre
- IdCiudadHotel
- CiudadHotel
- IdPaisHotel
- NombrePaisHotel
- FechaCheckIn
- FechaCheckOut
- HabitaciónTipo
- IdGerenteHotel
- NombreGerenteHotel
- IdiomasGerenteHotel \*\*
- NivelIdiomaGerenteHotel \*\*\*

\* ClienteEmail: un cliente puede tener varios emails

\*\* IdiomasGerenteHotel: cada gerente puede hablar varios idiomas

\*\*\* NivelIdiomaGerenteHotel: indica el nivel de cada idioma que habla el gerente

Consideraciones:

- Un cliente puede alojarse varias veces en distintos hoteles (en distintas fechas)
- Cada hotel está ubicado en una única ciudad, y cada ciudad pertenece a un país

- Cada hotel tiene un único gerente asignado
- Cada reserva corresponde a un cliente, un hotel y queda registrada con su fecha de check-in y check-out

## 5.2. Análisis de la Clave Primaria

Para la tabla de reservas de TravelPro, la clave primaria mínima debe ser:

- IDCliente (identifica al cliente)
- IDHotel (identifica el hotel)
- FechaCheckIn (identifica cuándo comienza la reserva)

Esta combinación es necesaria porque:

- Un cliente puede tener múltiples reservas
- Un hotel puede tener múltiples reservas
- Un cliente puede reservar en el mismo hotel en diferentes fechas

## 5.3. Normalización a 3FN

### 5.3.1. 1FN - Eliminar atributos multivaluados

Tabla: Clientes\_Emails

IDCliente	ClienteEmail
1	cliente1@email.com
1	cliente1@alternativo.com
2	cliente2@email.com

Tabla: Gerentes\_Idiomas

IdGerenteHotel	Idioma	NivelIdioma
G1	Inglés	Avanzado
G1	Español	Intermedio
G2	Inglés	Básico

### 5.3.2. 2FN - Eliminar dependencias parciales

Tabla: Hoteles

IDHotel	HotelNombre	IdCiudadHotel	IdGerenteHotel	NombreGerenteHotel
H1	Hotel A	C1	G1	Juan Pérez
H2	Hotel B	C2	G2	María López

Tabla: Ciudades

IdCiudadHotel	CiudadHotel	IdPaisHotel
C1	Buenos Aires	P1
C2	Madrid	P2

Tabla: Paises

IdPaisHotel	NombrePaisHotel
P1	Argentina
P2	España

### 5.3.3. 3FN - Eliminar dependencias transitivas

Tabla: Reservas

IDCliente	IDHotel	FechaCheckIn	FechaCheckOut	HabitaciónTipo
1	H1	2024-01-01	2024-01-05	Simple
1	H2	2024-02-01	2024-02-03	Doble
2	H1	2024-03-01	2024-03-10	Suite

Tabla: Clientes

IDCliente	ClienteNombre
1	Juan García
2	María López

## 5.4. Explicación de la Normalización

- Se separaron los emails de clientes en una tabla aparte (1FN)
- Se separaron los idiomas de los gerentes en una tabla aparte (1FN)
- Se extrajo la información de hoteles, ciudades y países en tablas separadas (2FN)
- Se eliminaron las dependencias transitivas separando la información de clientes (3FN)

## 6. Ejercicio Práctico

Normalizar la siguiente tabla hasta 3FN:

ID_Pedido	Cliente	Ciudad_Cliente	Producto	Categoría	Precio	Stock
1	Juan	CABA	Laptop	Electrónica	1000	50
2	María	CABA	Mouse	Electrónica	20	100
3	Juan	CABA	Mouse	Electrónica	20	100

### 6.1. Solución

#### 6.1.1. 1FN - Eliminar atributos multivaluados

Ya está en 1FN (valores atómicos)

#### 6.1.2. 2FN - Eliminar dependencias parciales

Tabla: Pedidos

ID_Pedido	ID_Cliente	ID_Producto
1	C1	P1
2	C2	P2
3	C1	P2

Tabla: Productos

ID_Producto	Nombre	Categoría	Precio
P1	Laptop	Electrónica	1000
P2	Mouse	Electrónica	20

### 6.1.3. 3FN - Eliminar dependencias transitivas

Tabla: Pedidos (igual que en 2FN)

ID_Pedido	ID_Cliente	ID_Producto
1	C1	P1
2	C2	P2
3	C1	P2

Tabla: Productos

ID_Producto	Nombre	ID_Categoría
P1	Laptop	CAT1
P2	Mouse	CAT1

Tabla: Categorías

ID_Categoría	Nombre
CAT1	Electrónica

Tabla: Clientes

ID_Cliente	Nombre	Ciudad
C1	Juan	CABA
C2	María	CABA