

Ejemplo 1: Sistema de Ventas

Registro de pedidos de clientes con detalles de productos comprados.

Tabla Original:

Pedidos

OrderID	CustomerName	CustomerEmail	ProductID	ProductName	Quantity	OrderDate
1	Alice	alice@email.com	P1	Laptop	2	2023-10-01
1	Alice	alice@email.com	P2	Mouse	1	2023-10-01
2	Bob	bob@email.com	P1	Laptop	1	2023-10-02

Problemas:

- Redundancia: *CustomerName*, *CustomerEmail*, y *ProductName* se repiten.
- Anomalías de actualización: Si Alice cambia su correo, hay que actualizar múltiples filas.
- Anomalías de inserción: No se puede agregar un producto sin vincularlo a un pedido.

Normalización Paso a Paso:

- 1) 1NF (Primera Forma Normal):
 - a) Asegurar atomicidad y eliminar grupos repetidos.
 - b) La tabla ya cumple 1NF porque cada fila representa un producto único en un pedido.
- 2) 2NF (Segunda Forma Normal):
 - a) Eliminar dependencias parciales.
 - b) *CustomerName* y *CustomerEmail* dependen solo de *OrderID*, no de la clave compuesta (*OrderID*, *ProductID*).
 - c) *ProductName* depende solo de *ProductID*.
- d) Tablas resultantes:
 - 1) Clientes:

CustomerID	CustomerName	CustomerEmail
1	Alice	alice@email.com
2	Bob	bob@email.com

2) Productos:

ProductID	ProductName
P1	Laptop
P2	Mouse

3) Pedidos:

OrderID	CustomerID	OrderDate
1	1	2023-10-01
2	2	2023-10-02

Detalles Pedido:

OrderID	ProductID	Quantity
1	P1	2
1	P2	1
2	P1	1

3) 3NF (Tercera Forma Normal):

- Eliminar dependencias transitivas.
- Ya está en 3NF: Todos los atributos no clave dependen directamente de la clave primaria.

Ejemplo 2: Sistema Universitario

Registro de matriculación de estudiantes en cursos.

Tabla Original: Matriculaciones

StudentID	StudentName	CourseID	CourseName	ProfessorID	ProfessorName	Grade
S1	John Doe	C1	Math 101	P1	Dr. Smith	A
S1	John Doe	C2	Physics 102	P2	Dr. Johnson	B
S2	Jane Roe	C1	Math 101	P1	Dr. Smith	C

Problemas:

- Redundancia: *StudentName*, *CourseName*, y *ProfessorName* se repiten.
- Anomalías de actualización: Si un profesor cambia su nombre, hay que actualizar todas sus filas.

Normalización Paso a Paso:

- 1) 1NF (Primera Forma Normal):
 - a) La tabla ya cumple 1NF (no hay grupos repetidos).
- 2) 2NF (Segunda Forma Normal):
 - a) Eliminar dependencias parciales.
 - 1) *StudentName* depende de *StudentID* (no de la clave compuesta (*StudentID*, *CourseID*)).
 - 2) *CourseName* y *ProfessorID* dependen de *CourseID*.
 - 3) *ProfessorName* depende de *ProfessorID*.
- 4) Tablas resultantes:

A) Estudiantes:

StudentID	StudentName
S1	John Doe
S2	Jane Roe

B) Cursos:

CourseID	CourseName	ProfessorID
C1	Math 101	P1
C2	Physics 102	P2

C) Profesores:

ProfessorID	ProfessorName
P1	Dr. Smith
P2	Dr. Johnson

D) Matriculaciones:

StudentID	CourseID	Grade
S1	C1	A
S1	C2	B
S2	C1	C

3) 3NF (Tercera Forma Normal):

a) Eliminar dependencias transitivas:

- 1) En *Cursos*, *ProfessorName* dependía de *ProfessorID*, no directamente de *CourseID*. Se resolvió al crear la tabla `Profesores`.

Ambos ejemplos evitan redundancias y anomalías, garantizando integridad y eficiencia.