

NORMALIZACIÓN

La normalización de base de datos es una técnica de modelado consistente en aplicar una serie de reglas a las relaciones con el fin de:

- Evitar redundancia
- Simplificar la actualización de datos
- Garantizar la integridad referencial

Requerimientos

Para que una tabla sea considerada una relación tiene que cumplirse lo siguiente:

- Cada tabla tiene que tener un nombre único
- No pueden haber dos filas iguales
 - No se permiten duplicados
- Todos los datos en una columna deben ser del mismo tipo

Conceptos

A continuación vamos a definir una serie de conceptos fundamentales para comprender las formas normales que explicaremos después:

Dependencia Funcional: Es una conexión entre uno o más atributos
 $\text{DNI} \rightarrow \text{Nombre y apellidos}$

Dependencia Funcional Reflexiva: Si “Y” está incluido en “X” entonces $X \rightarrow Y$
Si dirección y nombre están incluidos en DNI entonces con el DNI se puede recuperar la dirección y el nombre

Dependencia Funcional Aumentativa: Si $X \rightarrow Y$ entonces $XZ \rightarrow YZ$
 $\text{DNI} \rightarrow \text{Nombre}$
 $\text{DNI, Dirección} \rightarrow \text{Nombre, Dirección}$

Dependencia Funcional Transitiva: Si $X \rightarrow Y$ y $Y \rightarrow Z$ entonces $X \rightarrow Z$
 $\text{Fecha de nacimiento} \rightarrow \text{Edad}$
 $\text{Edad} \rightarrow \text{Conducir}$
 $\text{Fecha de nacimiento} \rightarrow \text{Edad} \rightarrow \text{Conducir}$

Ejemplo:

ID (Pk)	Nombre	Puesto	Salario	Emails
1	Juan Pérez	Jefe de área	3000	juan@test.com;jefel@test.com
2	José Sánchez	Administrativo	1500	jsanchez@test.com
3	Ana Díaz	Administrativo	1500	adiaz@test.com;admin@test.com

Para explicar las formas normales vamos a poner una tabla de ejemplo de Empleados. La siguiente tabla muestra la información de una empresa cuyos puestos de trabajo están regulados por el Estado, de modo que el salario de cada empleado depende del puesto. Datos empleados: ID, nombre, puesto, salario e email, siendo ID la clave primaria.

1FN (Primera forma normal)

Una tabla está en primera forma normal si:

Sus atributos contienen valores atómicos (esto quiere decir que tienen que ser indivisibles)

Tabla empleados:

ID (Pk)	Nombre	Puesto	Salario	Emails
1	Juan Pérez	Jefe de área	3000	juan@test.com;jefel@test.com X
2	José Sánchez	Administrativo	1500	jsanchez@test.com
3	Ana Díaz	Administrativo	1500	adiaz@test.com;admin@test.com X

En el ejemplo podemos ver que no se cumple para el atributo «Emails»

Para solucionarlo existen 2 opciones:

1. Duplicar registros con valores repetidos:

Se elimina el atributo «Emails» que violaba la condición

Se incluye un nuevo atributo «Email» que sí sea indivisible. Por lo que se añade el atributo email a la clave primaria

La nueva clave primaria será «ID + Email» Ya que vamos a suponer que un mismo correo puede pertenecer a más de una persona. A veces ocurre, que hay dos personas que comparten email, o te facilitan el email de la organización.

Tabla empleados:

ID (PK)	Nombre	Puesto	Salario	Email (PK)
1	Juan Pérez	Jefe de área	3000	juan@test.com
1	Juan Pérez	Jefe de área	3000	jefe1@test.com
2	José Sánchez	Administrativo	1500	jsanchez@test.com
3	Ana Díaz	Administrativo	1500	adiaz@test.com
3	Ana Díaz	Administrativo	1500	admin@test.com

2. Separar atributo «Email» en otra tabla:

Se crea una nueva tabla Empleados que no contenga el atributo «Email»

Tabla empleados:

ID (PK)	Nombre	Puesto	Salario
1	Juan Pérez	Jefe de área	3000
2	José Sánchez	Administrativo	1500
3	Ana Díaz	Administrativo	1500

Se crea una nueva tabla EMAILS con clave primaria ID-Email. Las tablas Emails y Empleados se relacionan por el campo ID. La clave primaria de esta nueva tabla es ID+Email.

Tabla empleados_emails:

ID (PK)	Email (PK)
1	juan@test.com
1	jefe1@test.com
2	jsanchez@test.com
3	adiaz@test.com

3	admin@test.com
---	----------------

2FN (Segunda forma normal)

Una tabla está en 2FN si:

Está en 1FN.

Todos los atributos que no son clave primaria tienen dependencia funcional completa con respecto a todas las claves existentes en el esquema. Para recuperar un atributo no clave, se necesita acceder por la clave completa, no por una subclave

Las 2FN aplican a las relaciones con claves primarias compuestas por dos o más atributos.

Cogemos la primera solución obtenida en la 1FN:

Tabla empleados:

ID (PK)	Nombre	Puesto	Salario	Email (PK)
1	Juan Pérez	Jefe de área	3000	juan@test.com
1	Juan Pérez	Jefe de área	3000	jefe1@test.com
2	José Sánchez	Administrativo	1500	jsanchez@test.com
3	Ana Díaz	Administrativo	1500	adiaz@test.com
3	Ana Díaz	Administrativo	1500	admin@test.com

En la tabla de Empleados se pueden ver las dependencias de los atributos:

**ID → Nombre,
puesto, salario**
Puesto → Salario

Observamos que los atributos nombre, puesto y salario dependen únicamente del campo ID, por lo que **no cumple la 2FN**.

Para solucionarlo:

→ Actuar sobre los atributos con dependencias incompletas:

Eliminar los atributos con dependencias incompletas

Crear nueva tabla con los atributos y la clave de la que depende

Tabla empleados:

ID (PK)	Nombre	Puesto	Salario
1	Juan Pérez	Jefe de área	3000
2	José Sánchez	Administrativo	1500
3	Ana Díaz	Administrativo	1500

Tabla empleados-emails:

ID (PK)(FK)	Email (PK)
1	juan@test.com
1	jefe1@test.com
2	jsanchez@test.com
3	adiaz@test.com
3	admin@test.com

Se llega a la misma solución que con la 1FN.

3FN (Tercera forma normal)

Una tabla está en 3FN si:

Está en 2FN

Todos los atributos que no son clave primaria no dependen transitivamente de ésta

Por tanto hay que buscar dependencias funcionales entre atributos que no estén en la clave.

Tabla empleados:

ID (PK)	Nombre	Puesto	Salario
1	Juan Pérez	Jefe de área	3000
2	José Sánchez	Administrativo	1500
3	Ana Díaz	Administrativo	1500

Las dependencias transitivas son:

ID →
Puesto
Puesto →
Salario

Observamos como la dependencia Puesto – Salario tiene dependencia transitiva con la clave primaria.

Para solucionarlo:

→ Actuar sobre los atributos con dependencias transitivas

Separar en una tabla adicional los atributos que tienen dependencia transitiva con la clave (Salario) y establecer como Pk el campo que define la transitividad (Puesto).

Tabla puestos:

Puesto (PK)	Salario
Jefe de área	3000
Administrativo	1500

Se añade el campo «Puesto» como *Foreign Key*:

Tabla empleados:

ID (PK)	Nombre	Puesto (FK)
1	Juan Pérez	Jefe de área
2	José Sánchez	Administrativo
3	Ana Díaz	Administrativo

POR TANTO SOLUCIÓN:

Tabla puestos:

Puesto	Salario
Jefe de área	3000
Administrativo	1500

Tabla empleados:

ID (PK)	Nombre	Puesto
1	Juan Pérez	Jefe de área
2	José Sánchez	Administrativo
3	Ana Díaz	Administrativo

Tabla empleados_emails:

ID (PK)	Email (PK)
1	juan@test.com
1	jefe1@test.com
2	jsanchez@test.com
3	adiaz@test.com
3	admin@test.com