

Normalización

NORMALIZACIÓN

Para **mejorar el desempeño** de una base de datos, así como **evitar redundancia en la información** que contiene y, **en consecuencia**, generar condiciones para un **mejor diseño**, el analista de sistemas debe conocer las formas de normalización y condiciones en las que la desnormalización es recomendable.

La **normalización** es la transformación de las vistas de usuario complejas y del almacén de datos a un juego de estructuras de datos más pequeñas y estables. Además de ser más simples y estables, las estructuras de datos son más fáciles de mantener que otras estructuras de datos.

1er FORMA NORMAL

La primera regla de normalización se expresa generalmente en forma de dos indicaciones separadas.

- Todos los **atributos**, valores almacenados en las columnas, deben ser **indivisibles**.
No deben existir **grupos de valores repetidos**, cada intersección de renglón columna contiene un solo valor, no un conjunto de ellos.
- Todos los atributos son dependientes de la clave primaria.

Ejemplo: Supongamos que tienes en una tabla una columna Dirección para almacenar la dirección completa, dato que se compondría del nombre de la calle, el número exterior, el número interior (puerta), el código postal, el estado y la capital.

id	Nombre	Dirección	Teléfono	URL
1	Anaya	Jl: Luca	92199932	Anaya.com
2	Pericles	C/Luna # 20-28018 Tlaxcala	99299492	Pericles.com



Calle	Número	Puerta	CP	Población	Provincia
Luna	20		28018	Tlaxcala	Tlaxcala


Una tabla con esta estructura plantea problemas a la hora de recuperar información. Imagina que necesitas conocer todas las entradas correspondientes a una determinada población, o que quieres buscar a partir del código postal. Al ser la dirección completa una secuencia de caracteres de estructura libre no resultaría nada fácil.

1er FORMA NORMAL

En cuanto a la segunda indicación, se debe evitar la repetición de los datos de la población y provincia en cada una de las filas. Siempre que al muestrear la información de una tabla aparezcan datos repetidos, existe la posibilidad de crear una tabla independiente con ellos.

Si el diseño de nuestra base de datos cumple estas premisas, está preparada para pasar de la primera a la segunda forma normal.

id	Nombre	calle	Número	Puerta	CP	Estado	Capital	Teléfono	URL
1	Anaya	Jl: Luca	15	2	28917	Tepic	Nayarit	93488345	Anaya.com
2	Pericles	Luna	20		28120	San Blas	Nayarit	88238188	Pericles.com
3	Mieres	Tajin	12	1	28120	San Blas	Nayarit	94989982	Mieres.es



Estado		CP	PK
CP	Estado	Capital	
28917	Tepic	Nayarit	
28120	San Blas	Nayarit	
23009	Jean	Jaen	

2da FORMA NORMAL

- La segunda forma normal añade la necesidad de que no existan dependencias funcionales parciales. Esto significa que todos los valores de las columnas de una fila deben depender de la clave primaria de dicha fila, entendiendo por clave primaria los valores de todas las columnas que la formen, en caso de ser más de una.
- Ésta afecta exclusivamente a las tablas en las que la clave primaria está formada por los valores de dos o más columnas, debiendo asegurarse, en este caso, que todas las demás columnas son accesibles a través de la clave completa y nunca mediante una parte de esa clave.

3er FORMA NORMAL

En cuanto a la tercera forma normal, ésta indica que no deben existir dependencias transitivas entre las columnas de una tabla, lo cual significa que las columnas que no forman parte de la clave primaria deben depender sólo de la clave, nunca de otra columna no clave.

DESNORMALIZACIÓN

Las reglas de **normalización no consideran el rendimiento**. En algunos casos es necesario considerar la desnormalización para mejorar el rendimiento. ***La desnormalización es la duplicación intencionada de columnas en varias tablas, lo cual aumenta la redundancia de datos.***

La normalización crea más tablas al avanzar hacia formas normales más altas, sin embargo, a mayor número de tablas, mayor número de combinaciones al recuperar los datos; lo que contribuye a la ralentización de las consultas. Por esta razón, para mejorar la velocidad de determinadas consultas, se pueden anular las ventajas de la integridad de datos y devolver la estructura de los datos a una forma normal inferior.

Por lo tanto, pueden existir circunstancias fortuitas que permitan algún grado de desnormalización para incrementar la velocidad de procesamiento. Debemos tomar en cuenta que la ventaja de una mayor velocidad de procesamiento debe evaluarse cuidadosamente contra la desventaja de datos anómalos.

Ejemplo

Aplicar las reglas de normalización a la siguiente tabla:
ordenes (id_orden, fecha, id_cliente, nom_cliente, estado, num_art, nom_art, cant, precio)

Ordenes

Id_orden	Fecha	Id_cliente	Nom_cliente	Estado	Num_art	nom_art	cant	Precio
2301	23/02/11	101	Martin	Caracas	3786	Red	3	35,00
2301	23/02/11	101	Martin	Caracas	4011	Raqueta	6	65,00
2301	23/02/11	101	Martin	Caracas	9132	Paq-3	8	4,75
2302	25/02/11	107	Herman	Coro	5794	Paq-6	4	5,00
2303	27/02/11	110	Pedro	Maracay	4011	Raqueta	2	65,00
2303	27/02/11	110	Pedro	Maracay	3141	Funda	2	10,00

PRIMERA FORMAL NORMAL (1FN)

Al examinar estos registros, podemos darnos cuenta que contienen un grupo repetido para NUM_ART, NOM_ART, CANT y PRECIO. La 1FN prohíbe los grupos repetidos, por lo tanto tenemos que convertir a la primera forma normal.

Los pasos a seguir son:

1. Tenemos que eliminar los grupos repetidos.
2. Tenemos que crear una nueva tabla con la PK de la tabla base y el grupo repetido.

Los registros quedan ahora conformados en dos tablas que llamaremos ORDENES y ARTICULOS_ORDENES

ordenes (id_orden, fecha, id_cliente, nom_cliente, estado)

Articulos_ordenes (id_orden, num_art, nom_art, cant, precio)

PRIMERA FORMAL NORMAL (1FN)

ordenes (id_orden, fecha, id_cliente, nom_cliente, estado)

Ordenes

Id_orden	Fecha	Id_cliente	Nom_cliente	Estado
2301	23/02/11	101	Martin	Caracas
2302	25/02/11	107	Herman	Coro
2303	27/02/11	110	Pedro	Maracay

Articulos_ordenes (id_orden, num_art, nom_art, cant, precio)

Articulos_ordenes

Id_orden	Num_art	nom_art	cant	Precio
2301	3786	Red	3	35,00
2301	4011	Raqueta	6	65,00
2301	9132	Paq-3	8	4,75
2302	5794	Paq-6	4	5,00
2303	4011	Raqueta	2	65,00
2303	3141	Funda	2	10,00

SEGUNDA FORMAL NORMAL (2FN)

Ahora procederemos a aplicar la segunda formal normal, es decir, tenemos que eliminar cualquier columna no clave que no dependa de la clave primaria de la tabla. Los pasos a seguir son:

1. Determinar cuáles columnas que no son clave no dependen de la clave primaria de la tabla.
2. Eliminar esas columnas de la tabla base.
3. Crear una segunda tabla con esas columnas y la(s) columna(s) de la PK de la cual dependen.

La tabla ORDENES está en 2FN. Cualquier valor único de ID_ORDEN determina un sólo valor para cada columna. Por lo tanto, todas las columnas son dependientes de la clave primaria ID_ORDEN.

Por su parte, la tabla ARTICULOS_ORDENES no se encuentra en 2FN ya que las columnas PRECIO y NOM_ART son dependientes de NUM_ART, pero no son dependientes de ID_ORDEN. Lo que haremos a continuación es eliminar estas columnas de la tabla ARTICULOS_ORDENES y crear una tabla ARTICULOS con dichas columnas y la clave primaria de la que dependen.

SEGUNDA FORMAL NORMAL (2FN)

Las tablas quedan ahora de la siguiente manera.

Articulos_ordenes (id_orden, num_art, cant)

Articulos_ordenes

Id_orden	Num_art	cant
2301	3786	3
2301	4011	6
2301	9132	8
2302	5794	4
2303	4011	2
2303	3141	2

Articulos (num_art, nom_art, precio)

Articulos

Num_art	nom_art	Precio
3786	Red	35,00
4011	Raqueta	65,00
9132	Paq-3	4,75
5794	Paq-6	5,00
3141	Funda	10,00

TERCERA FORMAL NORMAL (3FN)

La tercera forma normal nos dice que tenemos que eliminar cualquier columna no clave que sea

dependiente de otra columna no clave. Los pasos a seguir son:

1. Determinar las columnas que son dependientes de otra columna no clave.
2. Eliminar esas columnas de la tabla base.
3. Crear una segunda tabla con esas columnas y con la columna no clave de la cual son dependientes.

Al observar las tablas que hemos creado, nos damos cuenta que tanto la tabla ARTICULOS, como la tabla ARTICULOS_ORDENES se encuentran en 3FN. Sin embargo la tabla ORDENES no lo está, ya que NOM_CLIENTE y ESTADO son dependientes de ID_CLIENTE, y esta columna no es la clave primaria.

TERCERA FORMAL NORMAL (3FN)

Para normalizar esta tabla, moveremos las columnas no clave y la columna clave de la cual dependen dentro de una nueva tabla CLIENTES. Las nuevas tablas CLIENTES y ORDENES se muestran a continuación.

ordenes (id_orden, fecha, id_cliente)

Ordenes

Id_orden	Fecha	Id_cliente
2301	23/02/11	101
2302	25/02/11	107
2303	27/02/11	110

Clientes (id cliente, nom cliente, estado)

Ordenes

Id_cliente	Nom_cliente	Estado
101	Martin	Caracas
107	Herman	Coro
110	Pedro	Maracay

Modelo de Datos Normalizado

Por lo tanto la base de datos queda de la siguiente manera:

ordenes (id_orden, fecha, id_cliente)

Clientes (id_cliente, nom_cliente, estado)

Articulos (num_art, nom_art, precio)

Articulos_ordenes (id_orden, num_art, cant)