

**PROGRAMA DE ESTUDIOS**

# **DESARROLLO DE SISTEMAS DE INFORMACIÓN**

**DISEÑO E IMPLEMENTACIÓN  
DE BASE DE DATOS**

**Tema**

**DISEÑO LÓGICO - NORMALIZACIÓN**

## DISEÑO LÓGICO: NORMALIZACIÓN

### 1. ¿Qué es la normalización de una base de datos?

En la construcción e implementación de una Base de Datos existen varias etapas y la primera de ellas es la etapa de Diseño. A su vez, dividiremos el Diseño en cuatro:

**Diseño Conceptual**, en el que relevamos y comprendemos los datos utilizados en la empresa y sus sistemas. En él definimos el esquema conceptual.

**Diseño Lógico**, en el cual construimos un esquema de la información utilizada independiente del Sistema de Gestión de Base de Datos (SGBD). Es aquí donde transformamos el esquema conceptual en un esquema lógico.

**Diseño Físico**, es aquí donde detallamos como se implementará físicamente el esquema lógico y para ello debemos tener definido el SGBD que se utilizará.

**Diseño de Transacciones**, entendiendo como transacciones las acciones realizadas por usuarios o aplicaciones que acceden y modifican el contenido de la base de datos.

Entonces, para definir la normalización de una base de datos diremos que es una técnica aplicada durante el Diseño Lógico con el objeto de optimizar la estructura de los datos de un sistema de información en el modelo relacional.

Esta técnica no es nueva, fue desarrollada por E.F. Codd en 1972 y es ampliamente utilizada hasta nuestros días.

### 2. ¿Por qué se normalizan las bases de datos?

Los principales motivos para normalizar las bases de datos son evitar la duplicidad de datos, garantizar la mínima redundancia y lograr la máxima estabilidad. Además, aplicando este proceso, generamos una estructura comprensible y similar a la situación que ellos representan.

En lo que a hardware se refiere, logramos una optimización del espacio físico de almacenamiento.

Con la normalización también tenemos ventajas en la futura operación de las bases de datos. Evitamos problemas de actualización de los datos en las tablas. Facilitamos el acceso e interpretación de los datos, reduciendo el tiempo y la complejidad de las revisiones de las

bases de datos. Protegemos la integridad de los datos previniendo borrados indeseados de datos.

### 3. Normalizar una base de datos

Antes de detallar la técnica de normalización, repasemos brevemente unos conceptos del modelo relacional:

- Una *base de datos* es un conjunto de tablas.
- Una *tabla* está compuesta por registros denominados también como filas o tuplas.
- Un *registro* está compuesto por campos a los que también se llaman columnas o atributos.
- Las *claves* nos permiten acceder a los registros de una tabla.
- Una *clave candidata* es un campo o una combinación de campos que identifican una fila de forma exclusiva.
- Una *clave primaria* es una clave candidata que ha sido designada para identificar de forma única los registros de una tabla.
- Una *clave foránea* o externa se compone de uno o varios atributos, que forman una clave primaria en otra tabla a la cual se desea relacionar.
- Las *bases de datos relacionales* son aquellas en las que cualquier tabla puede relacionarse con otra tabla a través de claves.

### 4. Pasos para normalizar una base de datos

El proceso de normalización se realiza en pasos consecutivos que se denominan *Formas Normales*.

Las formas normales deben ser aplicadas a todas las tablas de una base de datos. Por lo tanto, cuando afirmamos que una base de datos se encuentra en la *Forma Normal N* estamos asegurando que todas sus tablas lo están.

En general, las tres primeras formas normales son el mínimo que deben cubrir la mayoría de las bases de datos y, aunque son posibles otros niveles de normalización, es considerado como el máximo nivel necesario para la mayoría de las aplicaciones.

El creador de estas tres primeras formas normales fue Edgar F. Codd.

Cada uno de estos niveles cumple con ciertos requisitos que tienen el objetivo de simplificar la información sin que exista una pérdida de datos.

A continuación, veamos cada de ellas.

- **Primera Forma Normal (1FN)**

Una tabla está en la primera forma normal si, y solo si, cumple con estas reglas:

- Todos los datos son atómicos.
- Todos los atributos o columnas son del mismo tipo de datos.

- **Segunda Forma Normal (2FN)**

Una tabla está en la segunda forma normal si sigue estas reglas:

- Si la tabla está en la primera forma normal.
- Los atributos que no forman parte de ninguna clave han de depender funcionalmente de toda la clave primaria.

- **Tercera Forma Normal (3FN)**

Una tabla está en la tercera forma normal si cumple estas reglas:

- Si la tabla está en la segunda forma normal.
- Los atributos no-clave no pueden depender de forma transitiva de una clave candidata (cuando un atributo que no sea clave depende de una clave primaria a través de otro atributo que no sea clave).

- **Forma Normal de Boyce-Codd**

Una tabla está en la forma de Boyce-Codd si cumple las siguientes condiciones:

- Si está en la tercera forma normal.
- Si cada determinante es una clave candidata.
  - Un determinante es un atributo que determina el valor de otro atributo.
  - Una clave candidata es una clave o una clave alternativa (en otras palabras, el atributo puede ser una clave para dicha tabla).

- **Cuarta Forma Normal (4FN)**

Una tabla está en la cuarta forma normal si cumple los siguientes criterios:

- Si está en la forma normal de Boyce-Codd.
- Si no contiene más de una dependencia multi-valor.

- Quinta Forma Normal (5FN) y otras

Básicamente, una tabla está en la quinta forma normal, cuando no puede ser dividida en tablas más pequeñas con diferentes claves (la mayor parte de las tablas se pueden dividir en tablas más pequeñas con la misma clave).

Más allá de la quinta forma normal, entramos en el apasionante mundo de las formas normales de clave dominante, un tipo ideal teórico que no se utiliza en la práctica.



### 5. ¿Cómo normalizar una base de datos?

Demostremos la normalización con un ejemplo.

La siguiente tabla muestra los datos de facturas emitidas por una distribuidora de material de oficina.

El cliente Facundo Rodríguez ha realizado un pedido para su empresa de diez teclados inalámbricos, doce cables de red y una cámara web usb. La compra de Natalia Martínez comprende dos auriculares inalámbricos y dos mini adaptadores USB-C a USB 2.0, mientras que Carlos García ha solicitado veinticinco teclados inalámbricos.

**Datos de Facturas**

número factura	fecha	cliente	número cliente	dirección	ítem	artículo	código artículo	unidades	precio
335	01.08.2022	Facundo Rodríguez	21	Calle Rivero 41, 41004 Sevilla	1	Teclado inalámbrico	2- 0023-D	10	245 euros
335	01.08.2022	Facundo Rodríguez	21	Calle Rivero 41, 41004 Sevilla	2	Cable red UTP RJ45	4- 0023-D	12	90 céntimos
335	01.08.2022	Facundo Rodríguez	21	Calle Rivero 41, 41004 Sevilla	3	Cámara web USB	5- 0023-D	1	165 euros
336	02.08.2022	Natalia Martínez	22	Calle San Eloy 82, 41001 Sevilla	1	Auriculares inalámbricos	1- 0023-D	2	175 euros
336	02.08.2022	Natalia Martínez	22	Calle San Eloy 82, 41001 Sevilla	2	Mini Adaptador USB-C USB 2.0	3- 0023-D	2	75 céntimos
337	02.08.2022	Carlos García	23	Avenida de Italia 245, 21003, Huelva	1	Teclado inalámbrico	2- 0023-D	25	245 euros

En esta base de datos de una tienda, los datos de las facturas contienen los siguientes atributos: número de factura número factura, fecha de facturación fecha, nombre del cliente cliente, número de cliente número cliente, dirección del cliente dirección, ítem de la factura ítem factura, nombre del artículo artículo, código de artículo código artículo, cantidad de unidades unidades y precio unitario precio.

Tenemos aquí una tabla en la que encontramos redundancias y celdas que contienen datos compuestos por más de un valor.

### Primera forma normal (1FN)

Para iniciar el proceso de normalización de esta tabla y lograr la primera forma normal debemos:

- dividir todos los datos multi-valor en columnas separadas (datos atómicos) y
- comprobar que los valores en cada columna sean del mismo tipo.

Para cumplir con el estado atómico de los datos, subdividiremos el atributo cliente en los atributos más específicos: nombre y apellido, mientras que al atributo dirección lo desglosaremos en calle, número, código postal y municipio.

Además, en la columna precio tenemos algunos datos en euros y otros en céntimos.

Para generar campos coherentes unificamos este tipo de dato.

La nueva tabla quedaría así:

número factura	fecha	apellido	nombre	número cliente	calle	número	código postal	municipio	ítem factura	artículo	código artículo	unidades	precio
335	01.08.2022	Rodríguez	Facundo	21	Calle Rivero	41	41004	Sevilla	1	Teclado inalámbrico	2-0023-D	10	245
335	01.08.2022	Rodríguez	Facundo	21	Calle Rivero	41	41004	Sevilla	2	Cable red UTP RJ45	4-0023-D	12	0.90
335	01.08.2022	Rodríguez	Facundo	21	Calle Rivero	41	41004	Sevilla	3	Camara web USB	5-0023-D	1	165
336	02.08.2022	Martínez	Natalia	22	Calle San Eloy	82	41001	Sevilla	1	Auriculares inalámbricos	1-0023-D	2	175
336	02.08.2022	Martínez	Natalia	22	Calle San Eloy	82	41001	Sevilla	2	Mini Adaptador USB-C USB 2.0	3-0023-D	2	0.75
337	02.08.2022	García	Carlos	23	Avenida de Italia	245	21003	Huelva	1	Teclado inalámbrico	2-0023-D	25	245

Si bien esta tabla se encuentra en la primera forma normal, aún tenemos valores duplicados que impiden procesar los datos de forma eficiente.

Para reducir las redundancias continuamos normalizando.

### **Segunda forma normal (2FN)**

Para estar en la segunda forma normal, a las condiciones de la primera se añade la siguiente:

- Los atributos que no forman parte de ninguna clave han de depender funcionalmente de toda la clave primaria.

En nuestro ejemplo, los atributos número factura + número cliente + ítem factura podrían componer una posible clave:

número factura	número cliente	ítem factura	fecha	apellido	nombre	calle	número	código postal	municipio	artículo	código artículo	unidades	precio
335	21	1	01.08.2022	Rodríguez	Facundo	Calle Rivero	41	41004	Sevilla	Teclado inalámbrico	2-0023-D	10	245
335	21	2	01.08.2022	Rodríguez	Facundo	Calle Rivero	41	41004	Sevilla	Cable red UTP RJ45	4-0023-D	12	0.90
335	21	3	01.08.2022	Rodríguez	Facundo	Calle Rivero	41	41004	Sevilla	Camara web USB	5-0023-D	1	165
336	22	1	02.08.2022	Martínez	Natalia	Calle San Eloy	82	41001	Sevilla	Auriculares inalámbricos	1-0023-D	2	175
336	22	2	02.08.2022	Martínez	Natalia	Calle San Eloy	82	41001	Sevilla	Mini Adaptador USB-C USB 2.0	3-0023-D	2	0.75
337	23	1	02.08.2022	García	Carlos	Avenida de Italia	245	21003	Huelva	Teclado inalámbrico	2-0023-D	25	245

Entonces, una clave número factura + número cliente + ítem factura con los valores {335, 21, 1} permitiría identificar el siguiente registro de la compra que ha realizado Facundo Rodríguez:

número factura	número cliente	ítem factura	fecha	apellido	nombre	calle	número	código postal	municipio	artículo	código artículo	unidades	precio
335	21	1	01.08.2022	Rodríguez	Facundo	Calle Rivero	41	41004	Sevilla	Teclado inalámbrico	2-0023-D	10	245

Pero para esta identificación no es necesaria toda la información aportada por la clave porque una combinación de número cliente + ítem factura es suficiente para identificar cada registro.

número factura	ítem factura	fecha	apellido	nombre	número cliente	calle	número	código postal	municipio	artículo	código artículo	unidades	precio
335	1	01.08.2022	Rodríguez	Facundo	21	Calle Rivero	41	41004	Sevilla	Teclado inalámbrico	2-0023-D	10	245
335	2	01.08.2022	Rodríguez	Facundo	21	Calle Rivero	41	41004	Sevilla	Cable red UTP RJ45	4-0023-D	12	0.90
335	3	01.08.2022	Rodríguez	Facundo	21	Calle Rivero	41	41004	Sevilla	Camara web USB	5-0023-D	1	165
336	1	02.08.2022	Martínez	Natalia	22	Calle San Eloy	82	41001	Sevilla	Auriculares inalámbricos	1-0023-D	2	175
336	2	02.08.2022	Martínez	Natalia	22	Calle San Eloy	82	41001	Sevilla	Mini Adaptador USB-C USB 2.0	3-0023-D	2	0.75
337	1	02.08.2022	García	Carlos	23	Avenida de Italia	245	21003	Huelva	Teclado inalámbrico	2-0023-D	25	245

Estas claves, con la mínima cantidad de atributos, se conocen como claves candidatas.

Normalmente, determinamos una clave candidata por tabla y su valor ideal es una numeración correlativa. Esta clave se erige en clave primaria y señala el orden de los registros.

La clave primaria se puede componer de un solo valor o, como en nuestro ejemplo, de varios valores. Nuestra tabla utiliza una clave primaria compuesta; formada por los valores de *número factura + ítem factura*.

Pero para llevar a una tabla a la segunda forma normal, no solo es necesario conocer la clave primaria y todos los atributos que no lo son, sino también cómo se relacionan entre sí.

Para hacerlo seguimos estos pasos:

- Comprobamos que todos los atributos no-clave dependen por completo de la clave primaria. Esta dependencia existe si todos los atributos de la clave primaria son necesarios para identificar a los atributos no-clave.
- Relegamos a tablas diferentes los atributos no-clave que no dependen de la clave primaria.

Si observamos atentamente nuestra tabla, podremos ver que las condiciones para la segunda forma normal no se cumplen por los siguientes motivos:

- La columna fecha solo depende de número factura, pero no del ítem factura.

- Los datos de los clientes (apellido, nombre, calle, número, código postal, municipio) tampoco dependen del ítem factura.

Entonces, para que nuestra tabla esté en la 2FN enviamos a los atributos dependientes del número de factura a una tabla diferente llamada Facturas y al resto de datos a una tabla que llamaremos Items.

#### **Facturas**

número factura	fecha	apellido	nombre	número cliente	calle	número	código postal	municipio
335	01.08.2022	Rodríguez	Facundo	21	Calle Rivero	41	41004	Sevilla
336	02.08.2022	Martínez	Natalia	22	Calle San Eloy	82	41001	Sevilla
337	02.08.2022	García	Carlos	23	Avenida de Italia	245	21003	Huelva

#### **Items**

número factura	ítem factura	artículo	código artículo	unidades	precio
335	1	Teclado inalámbrico	2-0023-D	10	245
335	2	Cable red UTP RJ45	4-0023-D	12	0.90
335	3	Cámara web USB	5-0023-D	1	165
336	1	Auriculares inalámbricos	1-0023-D	2	175
336	2	Mini Adaptador USB-C USB 2.0	3-0023-D	2	0.75
337	1	Teclado inalámbrico	2-0023-D	25	245

Tras esta segunda fase de la normalización, la columna número factura se encuentra en ambas tablas y las conecta. Mientras que este atributo actúa como clave primaria en la tabla *Facturas*, en la tabla *Items* lo utilizamos como clave foránea, pero también forma parte de la clave primaria compuesta.

Nuestras tablas ahora están en la 2FN, pero aún no se han eliminado del todo las redundancias. Por eso, el objetivo de la normalización suele ser la tercera forma normal.

#### **Tercera forma normal (3FN)**

Recordemos que para que una tabla esté en la tercera forma normal debe cumplir las condiciones de las dos primeras y además:

- Los atributos no-clave no pueden depender de forma transitiva de una clave candidata (cuando un atributo que no sea clave depende de una clave primaria a través de otro atributo que no sea clave).

Nuestro esquema incumple las condiciones de la tercera forma normal en varios puntos:

En la tabla **Facturas**, los atributos nombre y apellido, así como calle, número, código postal y municipio no solo dependen de la clave primaria número factura sino que también dependen de la columna número cliente.

En la tabla **Items** los atributos artículo y precio dependen de la clave primaria compuesta por número factura + ítem factura, pero también de la columna código artículo. Esto infringe la condición específica de la tercera forma normal.

### **Facturas**

número factura	fecha	apellido	nombre	número cliente	calle	número	código postal	municipio
335	01.08.2022	Rodríguez	Facundo	21	Calle Rivero	41	41004	Sevilla
336	02.08.2022	Martínez	Natalia	22	Calle San Eloy	82	41001	Sevilla
337	02.08.2022	García	Carlos	23	Avenida de Italia	245	21003	Huelva

### **Items**

número factura	ítem factura	artículo	código artículo	unidades	precio
335	1	Teclado inalámbrico	2-0023-D	10	245
335	2	Cable red UTP RJ45	4-0023-D	12	0.90
335	3	Camara web USB	5-0023-D	1	165
336	1	Auriculares inalámbricos	1-0023-D	2	175
336	2	Mini Adaptador USB-C USB 2.0	3-0023-D	2	0.75
337	1	Teclado inalámbrico	2-0023-D	25	245

Para eliminar las dependencias entre atributos no-clave repartimos los datos en tablas separadas que se interconectan con claves foráneas.

De este modo, resultarán cuatro tablas normalizadas a las que denominaremos: **Facturas**, **Clientes**, **Items** y **Artículos**.

### **Facturas**

La clave primaria de la tabla **Facturas** es la columna número factura. Cada número de factura se clasifica con la fecha de la factura y el número de cliente:

número factura	fecha	número cliente
335	01.08.2022	21
336	02.08.2022	22
337	02.08.2022	23

### Clients

En la tabla Clientes depositamos los datos de los clientes, y ambas tablas, Facturas y Clientes, se conectan mediante la columna número cliente, que en la tabla Clientes hace de clave primaria y en la tabla Facturas de clave foránea:

número cliente	apellido	nombre	calle	número	código postal	municipio
21	Rodríguez	Facundo	Calle Rivero	41	41004	Sevilla
22	Martínez	Natalia	Calle San Eloy	82	41001	Sevilla
23	García	Carlos	Avenida de Italia	245	21003	Huelva

### Items

Una tabla importante en nuestra base de datos es la tabla *Items*, puesto que revela qué artículos se incluyen en cada factura y cuántas unidades se han pedido. La clave primaria de la tabla resulta de número factura + ítem factura. Los artículos están presentes en la tabla mediante la columna número artículo que actúa como clave foránea enlazando con la tabla *Artículos*.

número factura	ítem factura	código artículo	unidades
335	1	2-0023-D	10
335	2	4-0023-D	12
335	3	5-0023-D	1
336	1	1-0023-D	2
336	2	3-0023-D	2
337	1	2-0023-D	25

### Artículos

*La tabla Artículos solo contiene los detalles de cada artículo, como su denominación y el precio. Como clave primaria tenemos la columna código artículo.*

código artículo	artículo	precio
-----------------	----------	--------

2-0023-D	Teclado inalámbrico	245
4-0023-D	Cable red UTP RJ45	0.90
5-0023-D	Cámara web USB	165
1-0023-D	Auriculares inalámbricos	175
3-0023-D	Mini Adaptador USB-C USB 2.0	0.75
2-0023-D	Teclado inalámbrico	245

**Bibliografía:**

- <https://openwebinars.net/blog/como-realizar-la-normalizacion-de-bases-de-datos-y-por-que/>



