

A grayscale photograph of a person with dark hair wearing large headphones, sitting at a desk and looking intently at a laptop screen. Their hands are clasped together near their chin. The background is blurred, showing what appears to be a modern office or studio environment with other people and equipment.

# UNIDAD 4

Normalización

# Normalización

La normalización es un término que deriva de la metodología que se utiliza para evitar la redundancia de datos y el fácil acceso y actualización de estos. Esta metodología fue enunciada por Codd, y consistía en definir un conjunto de normas a las que llamó **formas normales**. Cada forma normal (**FN**) fue numerada desde 1 hasta 5 y aparte existe una forma normal llamada de Boyce-Codd.

Lo importante de esta metodología es que para que una relación esté en 2FN debe estar también en 1FN, y para estar en 3FN debe estar antes en 2FN y por transitividad estará en 1FN.



# Normalización

A partir de ese modelo explicaremos cómo se aplican las diferentes formas normales sobre un modelo relacional. Explicaremos en detalle hasta la 3FN, ya que en la práctica se llega hasta esta FN como máximo

| Venta                   |
|-------------------------|
| nroFactura              |
| codigoProducto          |
| descripcionProducto     |
| precioUnitarioProducto  |
| cantidadVendidaProducto |
| codigoCliente           |
| noombreCliente          |
| fechaVenta              |
| domicilioCliente        |







# Clave Primaria

Un atributo será **clave primaria** si el resto de los atributos de la relación que no son clave **dependen funcional y únicamente** de dicho atributo y no dependen funcionalmente de ningún otro.

# 1era Forma Normal

Es una regla de normalización utilizada en el diseño de bases de datos relacionales. Esta regla establece que una tabla en una base de datos relacional debe tener un valor único en cada celda de la tabla y que no debe haber una repetición de grupos de valores dentro de una sola fila.

Además, cada celda de la tabla debe contener un solo valor.

En resumen, la primera forma normal es importante para garantizar que los datos se almacenen de manera clara y precisa en una base de datos relacional, lo que a su vez ayuda a garantizar la integridad de los datos y la eficiencia del acceso a la información.



# Aplicamos la 1FN

| Venta      |                |                     |                        |                         |               |               |            |                  |  |
|------------|----------------|---------------------|------------------------|-------------------------|---------------|---------------|------------|------------------|--|
| nroFactura | codigoProducto | descripcionProducto | precioUnitarioProducto | cantidadVendidaProducto | codigoCliente | nombreCliente | fechaVenta | domicilioCliente |  |
| 3          | 1              | Martillo            | 20,4                   | 15                      | 15            | Dibu Martinez | 20/4/2023  | Cuenca 255       |  |
| 3          | 5              | Clavo               | 0,8                    | 300                     | 15            | Dibu Martinez | 20/4/2023  | Cuenca 255       |  |
| 3          | 19             | Pinza               | 120                    | 1                       | 15            | Dibu Martinez | 20/4/2023  | Cuenca 255       |  |
| 3          | 15             | Destornillador      | 110                    | 8                       | 15            | Dibu Martinez | 20/4/2023  | Cuenca 255       |  |

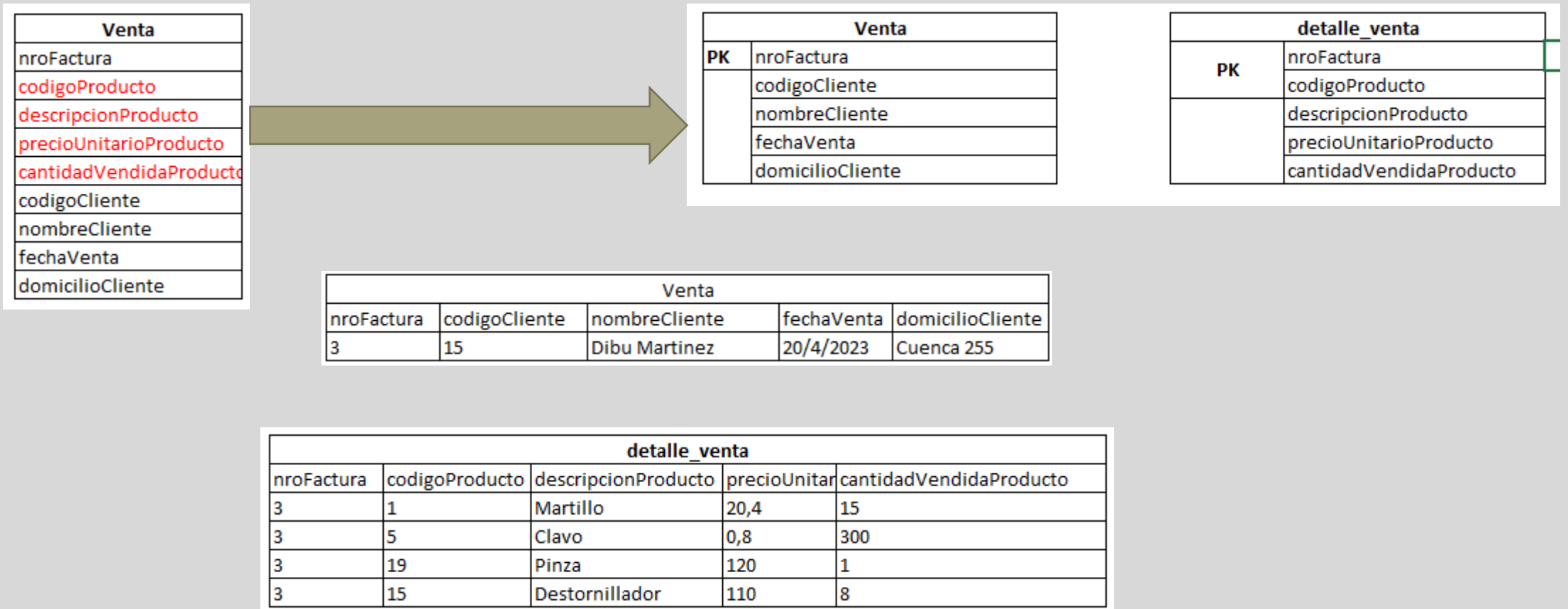
nroFactura, codigoCliente, nombreCliente, fechaVenta, domicilioCliente  
se repiten, esto es redundancia de datos.

Tampoco está definida la **clave primaria**

# Pasos a seguir para que quede en 1FN

1. Eliminamos los atributos
  1. codigoProducto
  2. cantidadVendidaProducto
  3. precioUnitarioProducto
  4. descripcionProducto
2. Indicamos como clave primaria nroFactura
3. Creamos una nueva tabla (relación) llamada **detalle\_venta** con los atributos antes mencionados.
4. Agregamos a detalle\_venta el atributo **nroFactura** que formará parte de la clave primaria compuesta

# El modelo nos queda así





# 2da Forma Normal

Condiciones para cumplir con la 2FN:

1. Debe estar en 1FN
2. Todos los atributos no clave dependen funcionalmente de la clave primaria

La relación **Venta** ya está en 1FN y también se encuentra en 2FN, pero la relación **detalle\_venta** que está en 1FN no se encuentra en la 2FN.

El atributo **descripcionProducto** depende funcionalmente del atributo **codigoProducto** que forma parte de la clave primaria, pero no depende funcionalmente del atributo **nroFactura** que también forma la clave primaria. En cambio los atributos **cantidadVendidaProducto** y **precioUnitarioProducto** dependen funcionalmente de ambos atributos que forman la clave primaria.



# Pasos a seguir para que quede en 2FN

Para que **detalle\_venta** quede en 2FN hacemos lo siguiente:

1. Eliminamos los atributos descripcionProducto
2. Definimos una nueva relación/tabla llamada **producto** con el atributo: descripcionProducto
3. Agregaremos a la nueva relación el atributo codigoProducto.
4. Definiremos el atributo codigoProducto como clave primaria.

El atributo **precioUnitarioProducto** dependiendo de las reglas de negocio podría tener dependencia funcional de la clave primaria completa o no. En nuestro caso asumimos que varía en el tiempo, si fuera fijo debería estar en la tabla Producto.

# El modelo nos queda así

1FN

| Venta |                  |
|-------|------------------|
| PK    | nroFactura       |
|       | codigoCliente    |
|       | nombreCliente    |
|       | fechaVenta       |
|       | domicilioCliente |

| detalle_venta |                         |
|---------------|-------------------------|
| PK            | nroFactura              |
|               | codigoProducto          |
|               | descripcionProducto     |
|               | precioUnitarioProducto  |
|               | cantidadVendidaProducto |

2FN

| Venta |                  |
|-------|------------------|
| PK    | nroFactura       |
|       | codigoCliente    |
|       | nombreCliente    |
|       | fechaVenta       |
|       | domicilioCliente |

| detalle_venta |                         |
|---------------|-------------------------|
| PK            | nroFactura              |
|               | codigoProducto          |
|               | precioUnitarioProducto  |
|               | cantidadVendidaProducto |

| producto |                     |
|----------|---------------------|
| PK       | codigoProducto      |
|          | descripcionProducto |

# 3ra Forma Normal

Condiciones para cumplir con la 3FN:

1. Debe estar en 2FN
2. Todos los atributos no clave dependen de manera no transitiva de la clave primaria

La relación **Venta** ya está en 1FN y también se encuentra en 2FN, pero la relación **detalle\_venta** que está en 1FN no se encuentra en la 2FN.

Las relaciones **detalle\_venta** y **producto** satisfacen ambas condiciones, pero no sucede lo mismo con **venta**, ya que **nombreCliente** y **domicilioCliente** dependen funcionalmente del atributo **codigoCliente** que no es un atributo clave y no dependen funcionalmente de **nroFactura** que es clave primaria





# Pasos a seguir para que quede en 3FN

Para que **venta** quede en 3FN hacemos lo siguiente:

1. Eliminamos los atributos nombreCliente y domicilioCliente
2. Definimos una nueva relación/tabla llamada **cliente** con los atributos: nombreCliente y domicilioCliente
3. Agregaremos a la nueva relación el atributo codigoCliente.
4. Definiremos el atributo codigoCliente como clave primaria.

# El modelo nos queda así

2FN

| Venta |                  |
|-------|------------------|
| PK    | nroFactura       |
|       | codigoCliente    |
|       | nombreCliente    |
|       | fechaVenta       |
|       | domicilioCliente |

| detalle_venta |                         |
|---------------|-------------------------|
| PK            | nroFactura              |
|               | codigoProducto          |
|               | precioUnitarioProducto  |
|               | cantidadVendidaProducto |

| producto |                     |
|----------|---------------------|
| PK       | codigoProducto      |
|          | descripcionProducto |

3FN

| Venta |               |
|-------|---------------|
| PK    | nroFactura    |
|       | codigoCliente |
|       | fechaVenta    |

| detalle_venta |                         |
|---------------|-------------------------|
| PK            | nroFactura              |
|               | codigoProducto          |
|               | precioUnitarioProducto  |
|               | cantidadVendidaProducto |

| Cliente |                  |
|---------|------------------|
| PK      | codigoCliente    |
|         | nombreCliente    |
|         | domicilioCliente |

| producto |                     |
|----------|---------------------|
| PK       | codigoProducto      |
|          | descripcionProducto |



# Reversibilidad

El proceso de normalización es reversible. Esto quiere decir que estando en cualquiera de las formas normales (excepto la primera) se puede volver a la forma normal antecesora sin pérdida de información.

**“Si poseen un conjunto de entidades en 3FN, desean regresar al modelo de dichas entidades y pierden datos de las relaciones en el intento, se normalizó de manera incorrecta”.**





# Integridad

Un modelo de datos posee integridad bajo estas condiciones:

1. Las claves primarias no pueden contener valor nulo, ya que la clave primaria no puede ser indefinida.
2. No pueden existir en el modelo valores de clave foránea, sin coincidencia con los valores de las claves primarias asociadas a ella.

Volviendo al modelo de las ventas, no puedo tener una venta para un cliente inexistente; primero debo tener registrado al cliente y luego registrar la venta (Integridad referencial).



# Desnormalización

Hay ocasiones que en función del volumen de datos almacenado en la BDD o la complejidad de las consultas a ejecutar en función de la lógica de negocio a aplicar,  
**se gana en perfomance desnormalizando la BDD.**

A black and white photograph of a person clapping their hands. The person is wearing a plaid shirt. In the foreground, there is a wooden desk with an open notebook and a smartphone resting on it. A tablet is also visible in the background. A dark grey rectangular box with a thin white border is overlaid on the left side of the image, containing the text '¡GRACIAS!'.

¡GRACIAS!