

3.4. Normalización

El proceso de normalización en bases de datos relacionales consiste en aplicar un conjunto de técnicas para convertir un esquema relacional en otro que aunque representa la misma información es más eficiente, ya que contiene menos redundancia y evita anomalías en los procesos de alta, baja y actualización de datos.

Con esto además conseguiremos:

- Guardar la misma información con el menor espacio posible
- Eliminar datos repetidos
- Proteger la integridad de los datos
- Tener los datos ordenados

Existen 5 (incluso 6) formas de normalizar la base de datos, pero bastará con utilizar las 3 primeras para conseguir que la base de datos esté bien estructurada.

3.4.1 La primera forma normal (1FN)

Se dice que una tabla está en primera forma normal 1FN cuando:

- Cada celda de la tabla sólo tiene un valor, es decir, la tabla está formada por campos atómicos.
- No existen grupos repetidos

Por definición, una base de datos relacional siempre estará normalizada a la 1FN, porque los valores de los atributos son siempre atómicos.

Sin embargo, dejar la base de datos simplemente así puede llevarnos a una serie de problemas como redundancia y posibles anomalías de actualización. Las formas normales más altas fueron desarrolladas para corregir estos problemas.

Veamos la siguiente tabla, donde DNI es la clave principal:

| EMPLEADO | | | | |
|-----------|------------------|---------|--------|--------------------------|
| PK | | | | |
| DNI | Nombre | Cargo | Sueldo | Telefono |
| 1234560P | Sonia García | Técnico | 2500 | 914567895 |
| 1589801L | Alfredo Jaén | Técnico | 2450 | 9145678990 6894512000 |
| 89745844K | Adriana Martínez | Gerente | 3100 | 9878545001 6978452500 |
| 000125L | Raúl Pérez | Técnico | 2600 | 9745896600 6458545522 |

El campo teléfono no cumple la 1FN porque toma varios valores, para resolver este problema, podemos duplicar los registros con valores repetidos convirtiendo el atributo que impide que la tabla esté en 1FN en clave primaria de la tabla:

| EMPLEADO | | | | |
|-----------|------------------|---------|--------|------------|
| PK | | | | PK |
| DNI | Nombre | Cargo | Sueldo | Telefono |
| 1234560P | Sonia García | Técnico | 2500 | 914567895 |
| 1589801L | Alfredo Jaén | Técnico | 2450 | 9145678990 |
| 1589801L | Alfredo Jaén | Técnico | 2450 | 6894512000 |
| 89745844K | Adriana Martínez | Gerente | 3100 | 9878545001 |
| 89745844K | Adriana Martínez | Gerente | 3100 | 6978452500 |
| 000125L | Raúl Pérez | Técnico | 2600 | 9745896600 |
| 000125L | Raúl Pérez | Técnico | 2600 | 6458545522 |

Si nos fijamos en la columna **Nombre**, vemos que tiene datos que pueden separarse y cumplir así con la 1ª condición de la 1FN:

| EMPLEADO | | | | | |
|-----------|---------|----------|---------|--------|------------|
| PK | | | | | PK |
| DNI | Nombre | Apellido | Cargo | Sueldo | Telefono |
| 1234560P | Sonia | García | Técnico | 2500 | 914567895 |
| 1589801L | Alfredo | Jaén | Técnico | 2450 | 9145678990 |
| 1589801L | Alfredo | Jaén | Técnico | 2450 | 6894512000 |
| 89745844K | Adriana | Martínez | Gerente | 3100 | 9878545001 |
| 89745844K | Adriana | Martínez | Gerente | 3100 | 6978452500 |
| 000125L | Raúl | Pérez | Técnico | 2600 | 9745896600 |
| 000125L | Raúl | Pérez | Técnico | 2600 | 6458545522 |

Nuestra nueva tabla ya cumple la primera condición de la 1FN porque cada celda tiene un valor. Pero no cumple con la 2ª condición, porque tenemos grupos repetidos, así que debemos sacar a una tabla los datos de los empleados que se están repitiendo junto con la PK de la tabla original, así tendríamos:

| EMPLEADO |
|----------|
|----------|

| PK | | | | |
|-----------|---------|----------|---------|--------|
| DNI | Nombre | Apellido | Cargo | Sueldo |
| 1234560P | Sonia | García | Técnico | 2500 |
| 1589801L | Alfredo | Jaén | Técnico | 2450 |
| 89745844K | Adriana | Martínez | Gerente | 3100 |
| 000125L | Raúl | Pérez | Técnico | 2600 |

| TELEFONO | |
|-----------|------------|
| | PK |
| DNI | Telefono |
| 1234560P | 914567895 |
| 1589801L | 9145678990 |
| 1589801L | 6894512000 |
| 89745844K | 9878545001 |
| 89745844K | 6978452500 |
| 000125L | 9745896600 |
| 000125L | 6458545522 |
| FK | |

Ahora todas las columnas cumplen la 1FN, así que no hace falta hacer más modificaciones y podemos afirmar que nuestras tablas están en la 1FN.

3.4.2 La segunda forma normal (2FN)

Decimos que una tabla de una base de datos está en la segunda forma normal si:

está en la primera (1FN)

todos los atributos que no formen parte de la clave primaria dependen de toda la clave completa y no sólo de una parte de ella. Es decir, debemos tener una dependencia total y no parcial de la clave primaria. Si la clave primaria de una tabla está formada por un solo atributo y no tenemos claves secundarias, podemos afirmar que tenemos dependencia completa y no será necesario realizar ninguna transformación (este sería el caso del ejemplo anterior)

Tenemos la siguiente tabla:

| EMPLEADO |
|----------|
|----------|

| PK | PK | |
|----------|----------------|-------------------|
| Empleado | Puesto | Lugar_de_trabajo |
| E-105 | Comercial | Avda de la Paz,35 |
| E-105 | Vendedor | Avda de la Paz,35 |
| E-201 | Informático | Santa Tecla,16 |
| E-520 | Administrativo | Pintores, 14 |
| E-520 | Contable | Pintores, 14 |

El atributo **Lugar_de_trabajo** depende del **Empleado**, pero no del **Puesto**, por tanto lo sacamos a otra tabla con la PK de la que depende:

| EMPLEADO | |
|----------|----------------|
| PK | PK |
| Empleado | Puesto |
| E-105 | Comercial |
| E-105 | Vendedor |
| E-201 | Informático |
| E-520 | Administrativo |
| E-520 | Contable |

| LUGAR_EMPLEADO | |
|----------------|-------------------|
| PK | |
| Empleado | Lugar_de_trabajo |
| E-105 | Avda de la Paz,35 |
| E-201 | Santa Tecla,16 |
| E-520 | Pintores, 14 |
| FK | |

Otro ejemplo:

| DETALLE_FACTURA | | | | | | |
|-----------------|------------|-------------|----------------|----------|----------------|-------------|
| PK | | | | | | |
| Tienda | NumFactura | CodArticulo | NombreArticulo | Cantidad | PrecioUnitario | PrecioTotal |
| T1 | 00001 | 01 | Libro | 2 | 25 | 50 |
| T1 | 00001 | 10 | DVD | 1 | 42 | 42 |
| T1 | 00001 | 21 | Rotulador | 4 | 5 | 20 |
| T1 | 00002 | 15 | Zapatillas | 1 | 95 | 95 |
| T2 | 00001 | 16 | Pantalón | 1 | 34 | 34 |
| T2 | 00001 | 20 | Bufanda | 2 | 24 | 48 |

Claramente está en 1FN porque tiene valores atómicos y no hay grupos repetidos, pero no está en 2FN ya que la columna **NombreArticulo** depende del atributo **CodArticulo** que forma parte de la PK, pero no depende de **NumFactura**, por tanto no depende completamente de la PK sino de uno de los atributos que la forman. Lo mismo ocurre con **PrecioUnitario**, depende de **CodArticulo** pero no de la PK completa.

Para resolver esta situación y cumplir con la 2FN sacaremos los atributos que dependen de **CodArticulo** a otra tabla teniendo así 2 tablas:

| DETALLE_FACTURA | | | | | |
|-----------------|------------|-------------|----------|----------------|-------------|
| PK | | | | | |
| Tienda | NumFactura | CodArticulo | Cantidad | PrecioUnitario | PrecioTotal |
| T1 | 00001 | 01 | 2 | 25 | 50 |
| T1 | 00001 | 10 | 1 | 42 | 42 |
| T1 | 00001 | 21 | 4 | 5 | 20 |
| T1 | 00002 | 15 | 1 | 95 | 95 |
| T2 | 00001 | 16 | 1 | 34 | 34 |
| T2 | 00001 | 20 | 2 | 24 | 48 |

| | | |
|--|----|--|
| | FK | |
|--|----|--|

| ARTICULO | | |
|-------------|----------------|----------------|
| PK | | |
| CodArticulo | NombreArticulo | PrecioUnitario |
| 01 | Libro | 25 |
| 10 | DVD | 42 |
| 21 | Rotulador | 5 |
| 15 | Zapatillas | 95 |
| 16 | Pantalón | 34 |
| 20 | Bufanda | 24 |

Podemos ver que ahora **CodArticulo** ha pasado a ser clave foránea apuntando a la tabla **ARTICULO**.

También vemos que **PrecioUnitario** se mantiene en **DETALLE_FACTURA** porque realmente tenemos dos atributos distintos, el precio actual del artículo y el precio del artículo cuando se emitió la factura.

Conclusión: **La 2FN se utiliza para eliminar las dependencias parciales**. Cuando encontremos una violación a la 2FN debemos mover los atributos que dependen sólo de manera parcial a una nueva relación donde ese atributo dependa completamente de la clave primaria.

3.4.3. La 3ª forma normal (3FN)

Decimos que una tabla está en 3FN si:

está en 2FN

cualquier atributo que no sea clave depende de forma no transitiva de la clave primaria, dicho de otra manera, los atributos que no forman parte de la clave primaria, no pueden depender de otros atributos que no son clave.

Lo veremos mejor con un ejemplo:

| GANADOR_CAMPEONATO | | | |
|--------------------|------|---------|-----------------|
| PK | PK | | |
| Campeonato | Año | Ganador | FechaNacimiento |
| Indianápolis | 2010 | G-150 | 12/10/1990 |
| Montmeló | 2015 | G-198 | 19/10/1995 |

| | | | |
|--------|------|-------|------------|
| Cheste | 2017 | G-150 | 12/10/1990 |
| Jerez | 2021 | G-780 | 12/11/2001 |

En esta tabla podemos ver claramente que el atributo **FechaNacimiento** del ganador depende del **Ganador** (que es un atributo no clave) y no de la PK, por tanto tendremos que sacarlo a otra tabla junto con el atributo del que depende:

| PILOTO | |
|---------|-----------------|
| PK | |
| Ganador | FechaNacimiento |
| G-150 | 12/10/1990 |
| G-198 | 19/10/1995 |
| G-150 | 12/10/1990 |
| G-780 | 12/11/2001 |

Quedando así la tabla original:

| GANADOR_CAMPEONATO | | |
|--------------------|------|---------|
| PK | PK | |
| Campeonato | Año | Ganador |
| Indianápolis | 2010 | G-150 |
| Montmeló | 2015 | G-198 |
| Cheste | 2017 | G-150 |
| Jerez | 2021 | G-780 |
| | | FK |

Otro ejemplo:

| LIBRO | | | |
|----------|--------------------|----------------------------------|-------------|
| PK | | | |
| CodLibro | Autor | Libro | PaisAutor |
| L1 | J.K. Rowling | Harry Potter y la Cámara Secreta | Reino unido |
| L2 | George R.R. Martin | Juego de tronos | EEUU |

| | | | |
|----|--------------------|-----------------|------|
| L3 | George R.R. Martin | Choque de Reyes | EEUU |
|----|--------------------|-----------------|------|

En esta tabla podemos decir que el atributo **Libro** determina al atributo **Autor**, porque conociendo el Libro sabremos quién es su autor. Sin embargo, conociendo el **Autor** no tenemos porqué conocer el Libro, ya que un autor puede haber escrito muchos libros, esta dependencia la escribimos **Libro** → **Autor**

De la misma manera, conociendo al **Autor** sabremos cuál es su **PaisAutor**, pero no por conocer el Pais sabremos quién es el autor, por tanto **Autor** determina **PaisAutor**: **Autor** → **PaisAutor**

Tenemos entonces **Libro** → **Autor** → **PaisAutor**, por tanto en esta tabla hay una dependencia transitiva ya que a partir de **Libro** podríamos llegar a **PaisAutor**, pero a través del atributo **Autor**.

Podríamos decir también que **PaisAutor** depende de **Autor** que no es la clave primaria de la tabla.

Esta dependencia es la que tenemos que eliminar, para ello quitaremos la columna **PaisAutor** de la tabla y crearemos una nueva tabla para ella, con lo que tendremos dos tablas **AUTOR** y **LIBRO**

| AUTOR | | |
|----------|--------------------|-------------|
| PK | | |
| CodAutor | Autor | PaisAutor |
| A1 | J.K. Rowling | Reino unido |
| A2 | George R.R. Martin | EEUU |

| LIBRO | | |
|----------|----------------------------------|----------|
| PK | | |
| CodLibro | Libro | CodAutor |
| L1 | Harry Potter y la Cámara Secreta | A1 |
| L2 | Juego de tronos | A2 |
| L3 | Choque de Reyes | A2 |
| | | FK |

Veamos si con esto es suficiente, en la tabla **LIBRO**, sólo tenemos el atributo **Libro** que depende de **codLibro**, por otro lado **CodAutor** es la clave foránea que nos vincula con la tabla **AUTOR**, así que esta tabla cumple con la 3FN. Vamos ahora ver la tabla **AUTOR**:

Autor depende de **CodAutor** (**CodAutor** → **Autor**)

Con **Autor** podemos determinar el **PaisAutor** (**Autor** → **PaisAutor**),

Por tanto **CodAutor** → **Autor** → **PaisAutor**, así que de **CodAutor** podemos llegar a **PaisAutor** a través de **Autor**, tenemos de nuevo una dependencia transitiva que tendremos que eliminar creando una nueva tabla para **PAIS**

| PAIS | |
|---------|-------------|
| PK | |
| CodPais | Pais |
| 1 | Reino Unido |
| 2 | EEUU |

| AUTOR | | |
|----------|--------------------|---------|
| PK | | |
| CodAutor | Autor | CodPais |
| A1 | J.K. Rowling | 1 |
| A2 | George R.R. Martin | 2 |
| | | FK |

¿Por qué las dependencias transitivas son un mal diseño de la base de datos?

Consideremos nuestra primera tabla nuevamente y veamos los problemas que crea, por ejemplo:

Si eliminamos los dos libros "*Juego de Tronos*" y "*Choque de Reyes*", eliminaríamos por completo de la base de datos al autor "*George R.R. Martin*" y su nacionalidad.

Si queremos agregar un nuevo autor a la base de datos no podemos hacerlo a menos que también agreguemos un libro.

¿Qué pasa si el autor aún no ha publicado un libro o si no se conoce el nombre de un libro que ha escrito?

Si "*George R.R. Martin*" cambia su país de residencia, tendríamos que cambiarla en todos los registros en los que aparece.

Tener varios registros con el mismo autor puede generar datos inexactos: ¿qué sucede si la persona que introduce los datos no se da cuenta de que hay varios registros para él y cambia los datos en un solo registro?

Estas son solo algunas de las razones por las que al eliminar las dependencias transitivas, protegemos los datos y garantizamos la coherencia.

Ejercicio

Realiza las transformaciones necesarias para que la siguiente tabla cumpla con la 3FN

| FACTURA | | | | | | | | | | |
|-------------------|---------------|------------|-------------|----------------|-------------|-----------------|----------|----------------|-------------------|---------------|
| Tienda_NumFactura | Fecha Factura | Forma Pago | Cod Cliente | Nombre Cliente | CodArtículo | Nombre Artículo | Cantidad | PrecioUnitario | SubTotal Artículo | Total Factura |
| T1-1 | 12/10/2021 | Efectivo | 1020 | Marisa Pérez | 01 | Mesa | 1 | 150 | 150 | 550 |
| T1-1 | 12/10/2021 | Efectivo | 1020 | Marisa Pérez | 15 | Silla | 4 | 75 | 300 | 550 |
| T1-1 | 12/10/2021 | Efectivo | 1020 | Marisa Pérez | 25 | Lámpara | 2 | 50 | 100 | 550 |
| T1-2 | 16/11/2021 | Tarjeta | 2030 | Mariano García | 16 | Sofá | 1 | 610 | 610 | 610 |
| T2-1 | 05/08/2021 | Tarjeta | 1062 | Rosa Mosqueta | 101 | Cama | 2 | 300 | 600 | 720 |
| T2-1 | 05/08/2021 | Tarjeta | 1062 | Rosa Mosqueta | 123 | Almohada | 2 | 60 | 120 | 720 |

Lo primero que vemos es que no está en 1FN porque **Tienda_NumFactura** no tiene valores atómicos, tampoco los tiene la columna **NombreCliente**, así que debemos separarlos:

| FACTURA | | | | | | | | | | | | |
|---------|------------|---------------|------------|-------------|----------------|------------------|-------------|-----------------|----------|----------------|------------------|--------------|
| Tienda | NumFactura | Fecha Factura | Forma Pago | Cod Cliente | Nombre Cliente | Apellido Cliente | CodArtículo | Nombre Artículo | Cantidad | PrecioUnitario | SubTotalArtículo | TotalFactura |
| T1 | 1 | 12/10/2021 | Efectivo | 1020 | Marisa | Pérez | 01 | Mesa | 1 | 150 | 150 | 550 |
| T1 | 1 | 12/10/2021 | Efectivo | 1020 | Marisa | Pérez | 15 | Silla | 4 | 75 | 300 | 550 |

UT3. Los esquemas relacionales y su transformación

| | | | | | | | | | | | | |
|----|---|------------|----------|------|---------|----------|---------|----------|---|-----|-----|-----|
| T1 | 1 | 12/10/2021 | Efectivo | 1020 | Marisa | Pérez | 25 | Lámpara | 2 | 50 | 100 | 550 |
| T1 | 2 | 16/11/2021 | Tarjeta | 2030 | Mariano | García | 16 | Sofá | 1 | 610 | 610 | 610 |
| T2 | 1 | 05/08/2021 | Tarjeta | 1062 | Rosa | Mosqueta | 10 1 | Cama | 2 | 300 | 600 | 720 |
| T2 | 1 | 05/08/2021 | Tarjeta | 1062 | Rosa | Mosqueta | 12 3 | Almohada | 2 | 60 | 120 | 720 |

Ya tenemos todos los atributos con datos atómicos, pero vemos que tenemos grupos de datos repetitivos, para eliminarlos sin eliminar detalles de la factura, crearemos dos tablas:

| Tienda | Num Factura | Fecha Factura | Forma Pago | Cod Cliente | Nombre Cliente | Apellido Cliente | Total Factura |
|--------|-------------|---------------|------------|-------------|----------------|------------------|---------------|
| T1 | 1 | 12/10/2021 | Efectivo | 1020 | Marisa | Pérez | 550 |
| T1 | 1 | 12/10/2021 | Efectivo | 1020 | Marisa | Pérez | 550 |
| T1 | 1 | 12/10/2021 | Efectivo | 1020 | Marisa | Pérez | 550 |
| T1 | 2 | 16/11/2021 | Tarjeta | 2030 | Mariano | García | 610 |
| T2 | 1 | 05/08/2021 | Tarjeta | 1062 | Rosa | Mosqueta | 720 |
| T2 | 1 | 05/08/2021 | Tarjeta | 1062 | Rosa | Mosqueta | 720 |

| CodArticulo | Nombre Articulo | Cantidad | PrecioUnitario | SubTotalArticulo |
|-------------|-----------------|----------|----------------|------------------|
| 01 | Mesa | 1 | 150 | 150 |
| 15 | Silla | 4 | 75 | 300 |
| 25 | Lámpara | 2 | 50 | 100 |
| 16 | Sofá | 1 | 610 | 610 |
| 101 | Cama | 2 | 300 | 600 |
| 123 | Almohada | 2 | 60 | 120 |

Si en la primera tabla eliminamos los datos repetidos, nos queda:

| Tienda | Num Factura | Fecha Factura | Forma Pago | Cod Cliente | Nombre Cliente | Apellido Cliente | Total Factura |
|--------|-------------|---------------|------------|-------------|----------------|------------------|---------------|
| T1 | 1 | 12/10/2021 | Efectivo | 1020 | Marisa | Pérez | 550 |
| T1 | 2 | 16/11/2021 | Tarjeta | 2030 | Mariano | García | 610 |

| | | | | | | | |
|----|---|------------|---------|------|------|----------|-----|
| T2 | 1 | 05/08/2021 | Tarjeta | 1062 | Rosa | Mosqueta | 720 |
|----|---|------------|---------|------|------|----------|-----|

Ya estamos en condiciones de indicar la PK de la primera tabla:

| FACTURA | | | | | | | |
|---------|-------------|---------------|------------|-------------|----------------|------------------|---------------|
| PK | | | | | | | |
| Tienda | Num Factura | Fecha Factura | Forma Pago | Cod Cliente | Nombre Cliente | Apellido Cliente | Total Factura |
| T1 | 1 | 12/10/2021 | Efectivo | 1020 | Marisa | Pérez | 550 |
| T1 | 2 | 16/11/2021 | Tarjeta | 2030 | Mariano | García | 610 |
| T2 | 1 | 05/08/2021 | Tarjeta | 1062 | Rosa | Mosqueta | 720 |

Y con esto agregamos dos nuevas columnas a **DETALLE_FACTURA** que serán las FK para poder relacionarla con la **FACTURA**:

| DETALLE_FACTURA | | | | | | |
|-----------------|-------------|--------------|-----------------|----------|----------------|------------------|
| PK | | | | | | |
| Tienda | Num Factura | Cod Articulo | Nombre Articulo | Cantidad | PrecioUnitario | SubTotalArticulo |
| T1 | 1 | 01 | Mesa | 1 | 150 | 150 |
| T1 | 1 | 15 | Silla | 4 | 75 | 300 |
| T1 | 1 | 25 | Lámpara | 2 | 50 | 100 |
| T1 | 2 | 16 | Sofá | 1 | 610 | 610 |
| T2 | 1 | 101 | Cama | 2 | 300 | 600 |
| T2 | 1 | 123 | Almohada | 2 | 60 | 120 |
| FK | | | | | | |

Ya tenemos las tablas en 1FN, pero no están en 2FN porque **NombreArticulo** y **PrecioUnitario** dependen de **CodArticulo**, pero no de **NumFactura** o de **Tienda** que son los otros atributos que forman la PK, por tanto tenemos una dependencia parcial y no total, así que tendremos que sacarlos a otra tabla:

| ARTICULO | | |
|-------------|----------------|----------------|
| PK | | |
| CodArticulo | NombreArticulo | PrecioUnitario |
| 01 | Mesa | 150 |

| | | |
|-----|----------|-----|
| 15 | Silla | 75 |
| 25 | Lámpara | 50 |
| 16 | Sofá | 610 |
| 101 | Cama | 300 |
| 123 | Almohada | 60 |

Y la tabla **DETALLE_FACTURA** quedará:

| DETALLE_FACTURA | | | | | |
|-----------------|-------------|-------------|----------|----------------|------------------|
| PK | | | | | |
| Tienda | Num Factura | CodArticulo | Cantidad | PrecioUnitario | SubTotalArticulo |
| T1 | 1 | 01 | 1 | 150 | 150 |
| T1 | 1 | 15 | 4 | 75 | 300 |
| T1 | 1 | 25 | 2 | 50 | 100 |
| T1 | 2 | 16 | 1 | 610 | 610 |
| T2 | 1 | 101 | 2 | 300 | 600 |
| T2 | 1 | 123 | 2 | 60 | 120 |
| FK | | FK | | | |

Ya tenemos nuestras tablas en la 2FN, sólo nos queda ver si están en 3FN, para ello tendremos que comprobar que ningún atributo tenga una dependencia transitiva:

En la tabla FACTURA:

CodCliente determina **NombreCliente** y **ApellidoCliente**, por tanto **CodCliente** → **NombreCliente** y **CodCliente** → **ApellidoCliente**

La **PK (Tienda+NumFactura+CodArticulo)** determina el **CodCliente**:

Tienda+NumFactura+CodArticulo → **CodCliente**

Con esto obtenemos que **Tienda+NumFactura+CodArticulo** → **CodCliente** → **NombreCliente**

Tienda+NumFactura+CodArticulo → **CodCliente** → **ApellidoCliente**

Por tanto aquí tenemos dependencias transitivas que tendremos que eliminar sacando los datos del cliente a una nueva tabla:

| CLIENTE | | |
|-------------|----------------|------------------|
| PK | | |
| Cod Cliente | Nombre Cliente | Apellido Cliente |

UT3. Los esquemas relacionales y su transformación

| | | |
|------|---------|----------|
| 1020 | Marisa | Pérez |
| 2030 | Mariano | García |
| 1062 | Rosa | Mosqueta |

| FACTURA | | | | | |
|---------|-------------|---------------|------------|-------------|---------------|
| PK | | | | | |
| Tienda | Num Factura | Fecha Factura | Forma Pago | Cod Cliente | Total Factura |
| T1 | 1 | 12/10/2021 | Efectivo | 1020 | 550 |
| T1 | 2 | 16/11/2021 | Tarjeta | 2030 | 610 |
| T2 | 1 | 05/08/2021 | Tarjeta | 1062 | 720 |
| | | | | FK | |

Ya han desaparecido las dependencias transitivas, así que ya podemos decir que nuestras tablas están en 3FN