

UNIDAD 3

MODELO LÓGICO

BASES DE DATOS 22/23
CFGS DAW

BOLETÍN 3.D. MODELO (LÓGICO) RELACIONAL. NORMALIZACIÓN CON SOLUCIONES.

Revisado por:

Sergio Badal, Abelardo Martínez y Pau Miñana

Fecha: 05/11/22

Licencia Creative Commons



Reconocimiento - NoComercial - CompartirIgual (by-nc-sa): No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

ÍNDICE DE CONTENIDO

1. EJERCICIO 1 ÓRDENES.....	3
2. EJERCICIO 2 LIBROS.....	10

1. EJERCICIO 1 ÓRDENES

DADA ESTA TABLA Y ESTOS DATOS, OBTÉN LA VERSIÓN EN 3FN

ordenes (id_orden, fecha, id_cliente, nom_cliente, estado, num_art, nom_art, cant, precio)

ordenes

id_orden	fecha	id_cliente	nom_cliente	estado_cliente	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

Date cuenta que los cuatro primeros campos se repiten, por lo que realmente tenemos esta tabla:

ordenes (id_orden, fecha, id_cliente, nom_cliente, estado, (num_art, nom_art, cant, precio)n)

id_orden	fecha	id_cliente	nom_cliente	estado_cliente	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

SOLUCIÓN PROPUESTA EJERCICIO 1**PRIMERA FORMAL NORMAL (1FN)**

Comprobamos si todos los campos están en primera forma normal. Para ello, buscamos estos campos:

- Compuestos
- Derivados
- Multivaluados
- Combinaciones de los anteriores

En la tabla **ordenes**, vemos que hay un campo multivaluado que habla de artículos

Te recomendamos numerar las tablas para no olvidarte ninguna por el camino.

(1) **ordenes** (id_orden, fecha, id_cliente, nom_cliente, estado_cliente, (num_art, nom_art, cant, precio)n)

Como es multivaluado, hacemos esta transformación, extrayendo los datos a otra tabla y enlazándola con la original:

(1) ordenes (<ordenes>, (<artículos>)n)	(1) ordenes (<ordenes>)
	(2) articulos_ordenes (id_orden, <artículos>)

Quedando como resultado FINAL el EJERCICIO 1, en 1FN, de la siguiente manera:

(1) ordenes (id_orden, fecha, id_cliente, nom_cliente, estado_cliente)
(2) articulos_ordenes (id_orden, num_art, nom_art, cant, precio)

IMPORTANTE: Fíjate en el detalle de que la clave primaria (CP) de la nueva tabla es SIEMPRE compuesta y está formada por la CP de la otra tabla y uno del resto de campos (el que no se repita).

(1) ordenes

id_orden	fecha	id_cliente	nom_cliente	estado_cliente
2301	23/02/11	101	Martin	Caracas
2302	25/02/11	107	Herman	Coro
2303	27/02/11	110	Pedro	Maracay

(2) 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

El procedimiento es el mismo siempre con los campos multivaluados.

Por ejemplo, ¿qué harías con este caso concreto?:

(1) clientes (<u>id_cliente</u> , nombre, apellidos, dirección, (telefonos)n)

Como es un multivaluado, extraemos los datos a otra tabla y la enlazamos con la tabla original:

(1) clientes (<clientes>, (<telefonos>)n)	(1) clientes (<clientes>)
	(2) telefonos (<u>id_cliente</u> , <telefonos>)

Quedando como resultado FINAL, en 1FN, de la siguiente manera:

(1) clientes (<u>id_cliente</u> , nombre, apellidos, dirección)
(2) telefonos (<u>id_cliente</u> , telefono)

IMPORTANTE: La clave primaria (CP) de la nueva tabla es SIEMPRE compuesta y está formada por la CP de la otra tabla (ID_CLIENTE) y uno del resto de campos (el que no se repita), en este caso solo puede ser el número de teléfono.

Para el resto de campos conflictivos haríamos esto:

- Compuestos: Nos aseguramos de que cada parte es un campo individual.
 - Por ejemplo, **clientes**(id_cliente, (nombre_completo = nombre, apellidos), dirección)
 - Sería, **clientes**(id_cliente, nombre_completo.nombre, nombre_completo.apellidos, dirección)
- Derivados: Los eliminamos e incluimos un comentario en lenguaje natural
 - Por ejemplo, **clientes**(id_cliente, nombre, edad)
 - Sería, **clientes**(id_cliente, nombre); edad=(fechactual (año)-fechanacimiento(año))
- Multivaluados: Como acabamos de ver
- Combinaciones de los anteriores: Los convertimos en simples y actuamos con cada uno.

SEGUNDA FORMAL NORMAL (2FN)

Ahora tenemos que eliminar cualquier **columna no-clave que no dependa de la clave primaria (CP) de la tabla**.

Los pasos a seguir son:

- Determinar qué columnas no-clave no dependen de la CP de la tabla.
- Eliminar esas columnas de la tabla base.
- Crear una segunda tabla con esas columnas y la(s) columna(s) de la CP de la cual dependen.

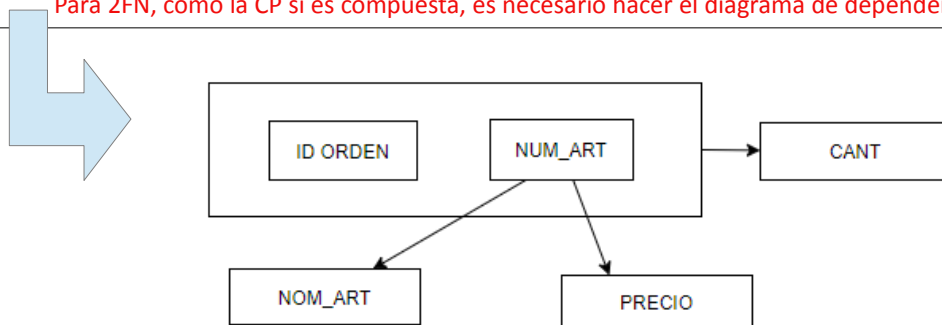
Veamos las tablas obtenidas en el paso anterior (1FN):

(1) **ordenes** (id_orden, fecha, id_cliente, nom_cliente, estado_cliente)

Como la CP no es compuesta, no es necesario hacer nada. La tabla está en 2FN.

(2) **articulos_ordenes** (id_orden, num_art, nom_art, cant, precio)

Para 2FN, como la CP sí es compuesta, es necesario hacer el diagrama de dependencias.

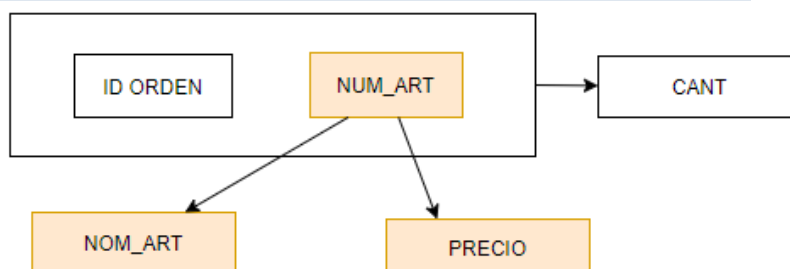


En este caso, la tabla se identifica (CP) por el id_orden y el número de artículo, de los que se deduce la cantidad.

- Ejemplo: en el tiquet número 123 de Mercadona (id_orden) tengo 5 unidades (cantidad) del artículo con código 33 (num_art) de nombre "Naranjas" (nom_art) que tiene un precio de 2 euros el kilo (precio).

Con esto podemos ver que, para saber el nombre (nom_art) y su precio (precio) me basta con el número de artículo (num_art). Es solamente para saber la cantidad (cant) para lo que necesito el número de ticket (id_orden) y el número de artículo (num_art).

El procedimiento para pasar a 2FN consiste en crear una nueva tabla con los campos que no dependan completamente de la CP compuesta, sacando de la tabla inicial **los campos que no dependen de toda la CP**:



Partimos de esta tabla:

(2) **articulos_ordenes** (id_orden, num_art, nom_art, cant, precio)

Que quedaría de la siguiente manera:

(2) **articulos_ordenes** (id_orden, num_art, cant)

(3) **articulos**(num_art, nom_art, precio)

IMPORTANTE: Fíjate que la CP de la nueva tabla está formada por el campo del que depende el resto de campos.

Quedando como resultado FINAL el EJERCICIO 1, en 2FN, de la siguiente manera:

(1) ordenes (<u>id_orden</u> , fecha, id_cliente, nom_cliente, estado_cliente)
(2) articulos_ordenes (<u>id_orden</u> , num_art, cant)
(3) articulos (<u>num_art</u> , nom_art, precio)

La representación de las tablas no es necesaria, pero sirve para aclarar el proceso.

(1) **ordenes**

id_orden	fecha	id_cliente	nom_cliente	estado_cliente
2301	23/02/11	101	Martin	Caracas
2301	23/02/11	101	Martin	Caracas
2301	23/02/11	101	Martin	Caracas
2302	25/02/11	107	Herman	Coro
2303	27/02/11	110	Pedro	Maracay
2303	27/02/11	110	Pedro	Maracay

(2) **articulos_ordenes**

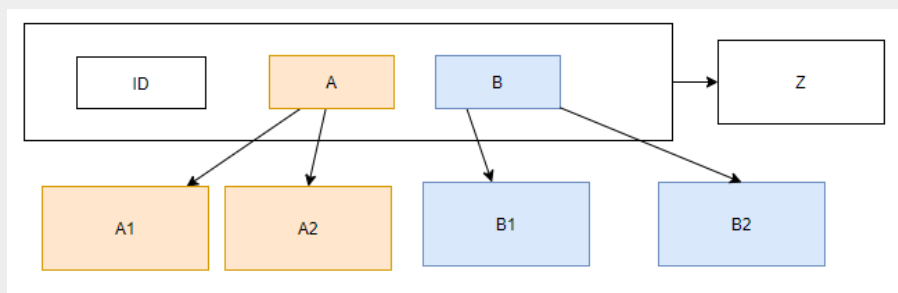
id_orden	num_art	cant
2301	3786	3
2301	4011	6
2301	9132	8
2302	5794	4
2303	4011	2
2303	3141	2

(3) **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

Por ejemplo, ¿qué harías con este caso genérico?:

(1) **mitabla** (ID, A, B, C, A1, A2, B1, B2, Z) con estas dependencias:



Quedando como resultado FINAL, en 2FN, de la siguiente manera:

(1) mitabla (<u>ID</u> , A, B, C, Z)
(2) mitabla_A (<u>A</u> , A1, A2)
(3) mitabla_B (<u>B</u> , B1, B2)

TERCERA FORMA 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:

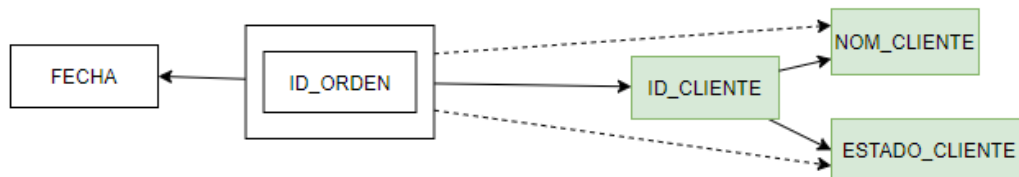
- Determinar las columnas que son dependientes de otra columna no-clave.
- Eliminar esas columnas de la tabla base.
- Crear una segunda tabla con esas columnas y con la columna no-clave de la cual son dependientes.

Veamos las tablas obtenidas en el paso anterior (2FN):

(1) ordenes (<u>id_orden</u> , fecha, id_cliente, nom_cliente, estado_cliente) Para 3FN, como los campos que son no-claves son más de uno es necesario el diagrama de dependencias.
(2) articulos_ordenes (<u>id_orden</u> , num_art, cant) Como solo hay un campo no-clave, ya está en 3FN.
(3) articulos (<u>num_art</u> , nom_art, precio) Para 3FN, como los campos que son no-claves son más de uno es necesario el diagrama de dependencias.

Necesitamos dos diagramas de dependencias. Uno para la tabla 1 y otro para la 3.

DIAGRAMA DE DEPENDENCIAS PARA (1) **ordenes** (id_orden, fecha, id_cliente, nom_cliente, estado_cliente)



Si analizamos cada campo, vemos que se puede obtener el nombre del cliente y el estado en el que vive el cliente a partir del id de cliente, sin necesidad de conocer el id de orden. En otras palabras, podemos decir que se pueden obtener estos dos campos a partir de la CP (id de orden), pero pasando por el id de cliente por lo que **tenemos una dependencia transitiva que debemos eliminar**.

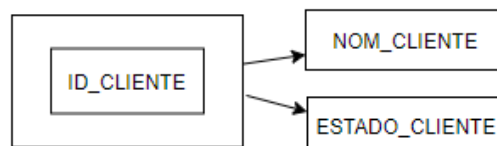
Para eliminar la dependencia transitiva extraemos esos dos campos a una tabla con el campo que hace de “puente” entre ellos y la CP como CP de la nueva tabla.

Que quedaría de la siguiente manera:

(1) ordenes (<u>id_orden</u> , fecha, <u>id_cliente</u>)
(4) clientes (<u>id_cliente</u> , nom_cliente, estado_cliente)

IMPORTANTE: Como hemos creado una nueva tabla, tenemos que ver si está en 1FN, 2FN y 3FN!!!!

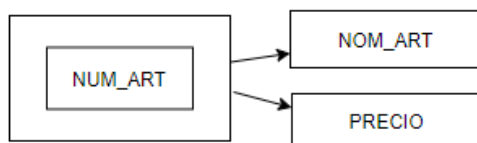
DIAGRAMA DE DEPENDENCIAS PARA (4) **clientes** (id_cliente, nom_cliente, estado_cliente)



En este caso, la CP es simple (está en 2FN) y no hay dependencias transitivas. **Por tanto, la tabla ya está en 3FN.**

Vamos a por el otro diagrama que teníamos pendiente:

DIAGRAMA DE DEPENDENCIAS PARA (3) **articulos** (num_art, nom_art, precio)



En este caso, no hay dependencia transitiva, ya que obtenemos los dos campos de la derecha a partir de la CP. Por tanto, la tabla ya está en 3FN.

Quedando como resultado FINAL el EJERCICIO 1, en 3FN, de la siguiente manera:

(1) ordenes (<u>id_orden</u> , fecha, id_cliente)
(2) articulos_ordenes (<u>id_orden</u> , num_art, cant)
(3) articulos (<u>num_art</u> , nom_art, precio)
(4) clientes (<u>id_cliente</u> , nom_cliente, estado_cliente)

La representación de las tablas no es necesaria, pero sirve para aclarar el proceso.

(1) **ordenes**

id_orden	fecha	id_cliente
2301	23/02/11	101
2301	23/02/11	101
2301	23/02/11	101
2302	25/02/11	107
2303	27/02/11	110
2303	27/02/11	110

(2) **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

(3) **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

(4) **clientes**

id_cliente	nom_cliente	estado_cliente
101	Martin	Caracas
107	Herman	Coro
110	Pedro	Maracay

2. EJERCICIO 2 LIBROS

DADA ESTA TABLA Y ESTOS DATOS, OBTÉN LA VERSIÓN EN 3FN

libros (codlibro, titulo, autor, editorial, nombreLector, fechaDev)

codLibro	titulo	autor	editorial	nombreLector	fechaDev
1001	Variable Compleja	Murray Spiegel	McGraw Hill	Pérez Gómez, Juan	15/04/2014
1004	Visual Basic	E. Petroustsos	Anaya	Ríos Terán, Ana	17/04/2014
1005	Estadística	Murray Spiegel	McGraw Hill	Roca, René	16/04/2014
1006	Oracle University	Nancy Greenberg y Priya Nathan	Oracle Corp.	García Roque, Luis	21/04/2014
1007	Clipper 5.01	Ramalho	McGraw Hill	Pérez Gómez, Juan	18/04/2014

SOLUCIÓN PROPUESTA EJERCICIO 2**PRIMERA FORMAL NORMAL (1FN)**

(1) **libros** (codlibro, titulo, autor, editorial, nombreLector, fechaDev)

Esta tabla no cumple el requisito de la Primera Forma Normal (1NF) de sólo tener campos atómicos, pues el **nombre del lector** es un campo que puede (y conviene) descomponerse en apellido1, apellido2 y nombre, tal como se muestra en la siguiente tabla.

codLibro	titulo	autor	editorial	apellido1 Lector	apellido2 Lector	nombreLector	fechaDev
1001	Variable compleja	Murray Spiegel	McGraw Hill	Pérez	Gómez	Juan	15/04/2014
1004	Visual Basic 5	E. Petroustos	Anaya	Ríos	Terán	Ana	17/04/2014
1005	Estadística	Murray Spiegel	McGraw Hill	Roca		René	16/04/2014
1006	Oracle University	NancyGreenberg	Oracle Corp.	García	Roque	Luis	20/04/2014
1007	Clipper 5.01	Ramalho	McGraw Hill	Pérez	Gómez	Juan	18/04/2014

(1) **libros** (codlibro, titulo, autor, editorial, apellido1Lector, apellido2Lector, nombreLector, fechaDev)

IMPORTANTE: Podrías pensar que el campo fechaDev también puede descomponerse en día, mes y año pero como existen funciones específicas para extraer estos datos a partir de una fecha **no se considera una fecha como un campo compuesto que haya que dividir**.

Otra apreciación bastante interesante, **aunque difícil de ver**, es que “la información del lector” (ya sea con un único campo o con tres campos) es una información multivaluada en esa tabla. Por tanto, tenemos realmente esta tabla:

(1) **libros** (codlibro, titulo, autor, editorial, (apellido1Lector, apellido2Lector, nombreLector)n, fechaDev)

¿Y qué hacemos cuando tenemos un atributo, o conjunto de atributos, multivaluado(s)?

En el ejercicio anterior vimos cómo extraíamos los datos de la orden del artículo a una tabla articulos_orden dado que para un artículo habían varias órdenes pero un articulo_orden solo tenía un artículo asociado (relación 1:N).

En este caso debemos ir más allá y ver la relación N:M entre libros y lectores, que nos pide una tabla intermedia donde debemos colocar la fecha de devolución y los ids de ambas tablas. Esto nos pide crear una clave codlibro.

(1) libros (<libros>, (<lectores>)n)	(1) libros (<u>codlibro</u> , titulo, autor, editorial)
	(2) lectores (<u>codlector</u> , apellido1Lector, apellido2Lector, nombreLector)
	(3) libro_lectores (<u>codlibro</u> , <u>codlector</u> , fechaDev)

Quedando como resultado FINAL el EJERCICIO 2, en 1FN, de la siguiente manera:

(1) libros (<u>codlibro</u> , titulo, autor, editorial)
(2) lectores (<u>codlector</u> , apellido1Lector, apellido2Lector, nombreLector)
(3) libro_lectores (<u>codlibro</u> , <u>codlector</u> , fechaDev)

SEGUNDA FORMAL NORMAL (2FN)

Ahora tenemos que eliminar cualquier **columna no-clave que no dependa de la clave primaria (CP) de la tabla**.

Los pasos a seguir son:

- Determinar qué columnas no-clave no dependen de la CP de la tabla.
- Eliminar esas columnas de la tabla base.
- Crear una segunda tabla con esas columnas y la(s) columna(s) de la CP de la cual dependen.

Veamos las tablas obtenidas en el paso anterior (1FN):

(1) **libros** (codlibro, titulo, autor, editorial)

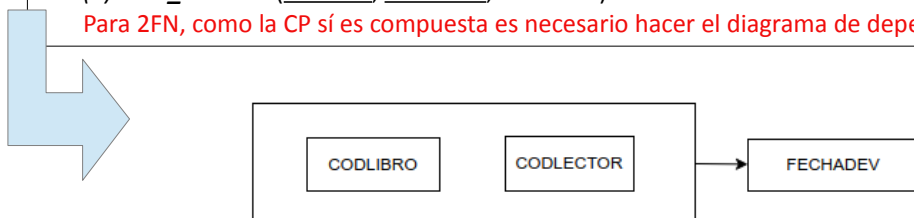
Como la CP no es compuesta, no es necesario hacer nada. La tabla está en 2FN.

(2) **lectores** (codlector, apellido1Lector, apellido2Lector, nombreLector)

Como la CP no es compuesta, no es necesario hacer nada. La tabla está en 2FN.

(3) **libro_lectores** (codlibro, codlector, fechaDev)

Para 2FN, como la CP sí es compuesta es necesario hacer el diagrama de dependencias.



La dependencia de la fechaDev es completa, por lo que la tabla sí que está en 2FN.

Quedando como resultado FINAL el EJERCICIO 2, en 2FN, de la siguiente manera:

(1) **libros** (codlibro, titulo, autor, editorial)

(2) **lectores** (codlector, apellido1Lector, apellido2Lector, nombreLector)

(3) **libro_lectores** (codlibro, codlector, fechaDev)

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:

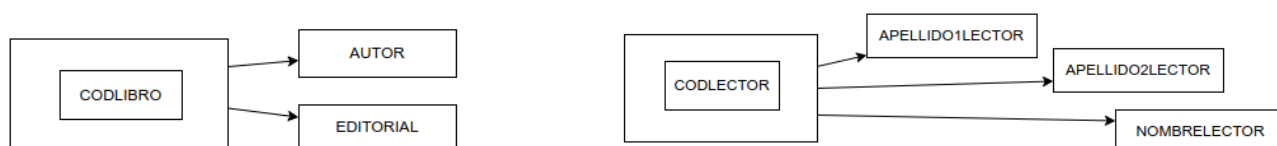
- Determinar las columnas que son dependientes de otra columna no-clave.
- Eliminar esas columnas de la tabla base.
- Crear una segunda tabla con esas columnas y con la columna no-clave de la cual son dependientes.

Veamos las tablas obtenidas en el paso anterior (2FN):

(1) libros (<u>codlibro</u> , titulo, autor, editorial) Para 3FN, como los campos que son no-claves son más de uno es necesario el diagrama de dependencias.
(2) lectores (<u>codlector</u> , apellido1Lector, apellido2Lector, nombreLector) Para 3FN, como los campos que son no-claves son más de uno es necesario el diagrama de dependencias.
(3) libro_lectores (<u>codlibro</u> , <u>codlector</u> , fechaDev) Como solo hay un campo no-clave, ya está en 3FN.

Necesitamos un diagrama de dependencias para la tabla 1 y la tabla 2.

DIAGRAMAS DE DEPENDENCIAS:



No hay ninguna dependencia transitiva en ninguno de los dos diagramas, por lo que ambas tablas están en 3FN.

Quedando como resultado FINAL el EJERCICIO 3, en 3FN, de la siguiente manera:

(1) libros (<u>codlibro</u> , titulo, autor, editorial)
(2) lectores (<u>codlector</u> , apellido1Lector, apellido2Lector, nombreLector)
(3) libro_lectores (<u>codlibro</u> , <u>codlector</u> , fechaDev)