

## Bases de Datos I.

### Práctica 4. Dependencias funcionales, formas normales y normalización.

#### Conceptos que aborda la práctica:

- Reconocimiento de restricciones de integridad
- Identificación de dependencias funcionales
- Tipos de dependencia funcionales
- Reconocimiento de superclaves y claves candidatas
- Atributos primos y no primos
- Formas normales: 1FN, 2FN, 3FN y FNBC

#### Objetivos de los Ejercicios:

- Apropiarse de los conceptos asociados a las relaciones (esquema, extensión, tuplas, dependencias funcionales, dominios).
- Identificar las dependencias funcionales para eliminar anomalías en el modelo relacional.
- Reconocer los elementos que se asocian a las formas normales. (superclaves, claves candidatas, dependencia funcional plena, dependencias funcional transitiva, etc.).
- Eliminación de anomalías mediante normalización.

#### Ejercicio Introductorio:

Determinar si se puede realizar la inserción de tuplas sin violar la restricción de integridad:

Dada la relación **R(AT, DF)**, con:

AT = {A, B, C}

DF = {A → B, B → C}

Suponiendo que la relación contiene las siguientes tuplas:

A	B	C
a <sub>1</sub>	b <sub>2</sub>	c <sub>1</sub>
a <sub>2</sub>	b <sub>1</sub>	c <sub>2</sub>
a <sub>4</sub>	b <sub>3</sub>	c <sub>4</sub>
a <sub>3</sub>	b <sub>4</sub>	c <sub>5</sub>

Determinar si se puede realizar la inserción de las siguientes tuplas sin violar las restricciones de integridad:

- a)  $(a_1, b_2, c_2)$
- b)  $(a_5, b_2, c_3)$
- c)  $(a_5, b_5, c_3)$

Solución de los ítems:

- a)  $(a_1, b_2, c_2)$

No se puede insertar la tupla.

Como el atributo A con valor  $a_1$  ya se encuentra en la extensión de la relación R, y considerando la dependencia funcional  $A \rightarrow B$ , es que si tengo un valor  $a_1$  para el atributo A, siempre va a implicar un valor  $b_2$  para el atributo B, y en esta tupla se cumple.

El valor  $b_2$  en el atributo B también se encuentra en la extensión de R, pero asociado a un valor  $c_1$  en el atributo C. Sin embargo la tupla a insertar tiene un valor  $c_2$  para el atributo C, por lo tanto viola la dependencia funcional  $B \rightarrow C$  donde siempre para un valor  $b_2 \rightarrow c_1$ .

- b)  $(a_5, b_2, c_3)$

No se puede insertar la tupla.

Si bien el valor  $a_5$  no está en la extensión de la relación y cumple con la dependencia funcional  $A \rightarrow B$ , no pasa así con  $B \rightarrow C$ , porque  $b_2$  ya se encuentra en la extensión de la relación asociado a un valor  $c_1$ .

- c)  $(a_5, b_5, c_3)$

Sí se puede insertar la tupla.

Los valores  $a_5$  y  $b_5$  no están en la extensión de la relación, por lo que su inserción no viola la restricción de integridad.

## Ejercicios:

**Ejercicio 1:** Dada la relación **R(AT, DF)**, con:

AT = {A, B, C, D}

DF = { $A \rightarrow B$ ,  $D \rightarrow C$ }

Suponiendo que la relación contiene las siguientes tuplas:

A	B	C	D
$a_1$	$b_2$	$c_1$	$d_6$
$a_2$	$b_1$	$c_2$	$d_2$
$a_4$	$b_3$	$c_4$	$d_4$
$a_5$	$b_4$	$c_5$	$d_1$

Determinar si se puede realizar la inserción de las siguientes tuplas sin violar las restricciones de integridad:

a)  $(a_3, b_5, c_3, d_3)$

**Se puede insertar la tupla,  $a_3$  no está en la extensión de la relación y tampoco lo está  $d_3$ , y las restricciones están dadas sobre los valores del atributo A respecto de B y D respecto e C, respectivamente.**

b)  $(a_5, b_4, c_3, d_1)$

**No se puede insertar la tupla, si bien cumple que para el valor  $a_5 \rightarrow b_4$ , no cumple  $d_1 \rightarrow c_3$**

c)  $(a_1, b_1, c_1, d_6)$

**No se puede insertar la tupla, no cumple  $a_1 \rightarrow b_1$  (en la extensión teníamos que para un valor  $a_1 \rightarrow b_2$ )**

**Ejercicio 2:** Dada la relación **FACULTAD(AT, DF)**

AT = {Legajo, Nombre, Materia, FechaInscripcion, FechaExamen, Nota}

DF = {Legajo  $\rightarrow$  Nombre;

Legajo, Materia  $\rightarrow$  FechaInscripcion

Legajo, Materia, FechaExamen  $\rightarrow$  Nota }

Suponiendo que la relación contiene las siguientes tuplas:

FACULTAD					
Legajo	Nombre	Materia	Fecha Inscripción	Fecha Examen	Nota
3092	Santiago Carranza	IS001	23/02/2022	04/06/2022	4
5816	Simón Cuerva	AL008	21/02/2022	28/06/2022	8
3591	Melissa Salgueiro	AL008	03/03/2022	28/06/2022	6
8924	Santino Oancea	ES012	01/03/2022	02/07/2022	6
6751	Antonela Lores	SO006	18/02/2022	20/06/2022	4
7481	Lautaro Gallegos	ES012	27/02/22	02/07/2022	7
5816	Simón Cuerva	IS001	21/02/2022	04/06/2022	5

Determinar si se puede realizar la inserción de las siguientes tuplas sin violar las restricciones de integridad:

a) (3092, Santiago Carranza, IS001, 12/08/17, 10/09/17, 3)

**No se puede insertar la tupla, porque si bien el legajo 3092 implica el nombre Santiago Carranza, el legajo 3092 junto con la materia IS001 ya implicaban en la extensión de la relación la fecha 23/02/22.**

b) (6223, Romina Freiria, IS001, 22/02/18, null, null)

**No se puede insertar la tupla, FechaExamen no puede tener valores nulos.**

c) (4533, Eduardo Beraza, ES012, 28/02/18, 02/07/18, 5)

**Puede insertarse la tupla, no existe en la extensión el legajo 45333**

**Ejercicio 3:** Dada la relación **LIBRO(AT, DF)**

AT = {Autor, Nacionalidad, ISBN, Título, Editorial}

DF = {Autor → Nacionalidad;

Autor, ISBN → Título, Editorial }

Suponiendo que la relación contiene las siguientes tuplas:

LIBRO				
Autor	Nacionalidad	ISBN	Título	Editorial
Hermann Hesse	Alemán	9788408163442	El lobo estepario	Alianza
Oscar Wilde	Irlandés	9788416968213	El retrato de Dorian Grey	Imaginador
Eduardo Galeano	Uruguayo	9789876295116	Las venas abiertas de América Latina	Siglo XXI editores
Julio Cortázar	Argentino	9789934886112	Intercontextualidades de Castillo, Borges y Cortázar	Alfaguara
Mario Vargas Llosa	Peruano	9878665489921	La ciudad y los perros	Alfaguara
Carlos Ruiz Zafón	Español	9788439731603	El cementerio de los libros olvidados	Alianza

Determinar si se pueden realizar la inserción de las siguientes tuplas sin violar las restricciones de integridad:

a) (Jorge Luís Borges, Argentino, 9789934886112, Intercontextualidades de Castillo, Borges y Cortázar, Alfaguara)

**Puede insertarse la tupla, en la extensión no existe el autor Jorge Luís Borges.**

b) (Mario Vargas Llosa, Español, 9878665489921, La ciudad y los perros, Alfaguara)

**No puede insertarse la tupla, el autor Mario Vargas Llosa está en la extensión con nacionalidad Peruano y la tupla a insertar tiene como nacionalidad Español**

- c) (Carlos Ruíz Zafón, Español, 9788666448813, La sombra del Viento, Alianza)

**Puede insertarse la tupla, porque Autor, ISBN implican el título, como el ISBN a insertar es diferente, podemos insertar el título La sombra del Viento y la editorial Alianza.**

**Ejercicio 4:** A partir de la siguiente narrativa, extraer los atributos y las dependencias funcionales correspondientes, simbolizar:

Supongamos un laboratorio farmacológico que desea corroborar la eficacia de los medicamentos (M) producidos, así como los tratamientos (T) que se realizan sobre pacientes voluntarios. Los supuestos son:

$R = (M, C, N, D, A, T)$

Y las restricciones a considerar son las siguientes:

- a) Los pacientes voluntarios se identifican por un código alfabético (C) o por su número de documento (N).

**$C \rightarrow N$  y  $N \rightarrow C$**

- b) Todos los médicos que trabajan en el laboratorio son identificados por su documento (D), pero asimismo también tienen apellidos únicos (A).

**$D \rightarrow A$  y  $A \rightarrow D$**

- c) Un tratamiento sólo puede ser aplicado a un único paciente, y por un único médico.

**$T, C \rightarrow D$**

- d) Un médico no puede ser asociado a más de un medicamento por paciente.

**$D, C \rightarrow M$**

- e) Un médico puede aplicar sólo un tratamiento a cada paciente.

**$D, C \rightarrow T$**

- f) En un determinado tratamiento, a cada paciente sólo se le puede aplicar un único medicamento.

**$T, C \rightarrow M$**

**Ejercicio 5:** A partir de la siguiente narrativa, extraer los atributos y las dependencias funcionales correspondientes, simbolizar:

Se desea construir una base de datos para una universidad, considerando los siguientes supuestos:

a) Un profesor se identifica por un código de profesor (CP), a su vez todos los profesores tienen nombres diferentes (NP).

**CP → NP**

**como también indica que los profesores tienen nombres diferentes, el nombre va a determinar funcionalmente al código.**

**NP → CP**

b) Cada materia (M) es impartida por un único profesor a cargo, pero un mismo profesor puede ser responsable de más de una materia. Las materias se dividen en una o más comisiones (C). Cada alumno (A) pertenece únicamente a una determinada comisión para cada materia.

**M → CP**

**A, M → C**

c) Las materias se dividen en uno o más grupos (comisiones), un grupo sólo puede pertenecer a una materia, por lo que para cada valor de grupo existe un único valor de materia. Además todos los alumnos en una materia pertenecen a un único grupo.

**G → M**

d) Cada profesor depende únicamente de un Instituto (I), y una asignatura pertenece únicamente a un Instituto, el que se corresponde con el departamento al que pertenece el docente que la imparte.

**CP → I**

**M, CP → I**

## Conceptos previos:

**Dependencia funcional plena o completa:** Un atributo o conjunto de atributos Y, tiene una dependencia funcional plena sobre otro conjunto de atributos X, siempre que Y dependa funcionalmente de X y además no pueda obtenerse de X ningún subconjunto del que dependa Y.

DNI, APELLIDO → NOMBRE no es plena porque DNI → NOMBRE

**Dependencia funcional elemental:** Una dependencia funcional elemental debe ser plena y además el implicado está compuesto por sólo un atributo.

**Dependencia funcional transitiva:** Para analizar la transitividad se necesitan mínimamente tres conjuntos de atributos, X, Y, Z. Una dependencia es transitiva cuando el implicado de una dependencia actúa en otra como implicante

**Formas normales:** La teoría de normalización permite obtener estructuras de datos eficientes y evitar las anomalías de actualización. Es un método formal que permite realizar un diseño flexible de relaciones, que permita extenderse al agregar nuevos atributos a las relaciones existentes y nuevas relaciones.

**Atributos primos:** Son aquellos que pertenecen a alguna clave candidata

**Atributos no primos:** Son aquellos no que pertenecen a ninguna clave candidata

**Primera forma normal (1NF):** La primera forma normal conforma en parte la definición del modelo relacional, donde a cada valor de un atributo le correspondía un valor atómico y simple de su dominio.

**Ejercicio 6:** Dadas las siguientes relaciones, normalizar a 1FN. ¿Cómo quedaría cada relación luego de normalizar?

- a) Sea **PERSONA(AT)**  
AT = {dni, apellido, nombre, dirección, teléfonos\*}

Dni	Apellido	Nombre	Dirección	Teléfonos
11.222.333	García	Vicenta	Av. San Martín N° 143	(2901)15514897 (2901)422631
22.333.444	Ponce	Ángel	Los castores N° 345	(2901)15615729

Dni	Apellido	Nombre	Dirección	Teléfonos
				(2901)15404947 (2901)433085
33.444.555	Aliaga	Mariana	Marcos Zar Sur N° 345	(2901)15476924

**Para llevar a 1FN cada atributo tiene que tener un único valor de dominio, para normalizar a 1FN deberíamos o duplicamos las tuplas para cada uno de los valores múltiples en el atributo teléfono, o separamos en dos relaciones una que tenga el atributo teléfono y la clave de la relación de la que se desprende.**

Dni	Apellido	Nombre	Dirección
11.222.333	García	Vicenta	Av. San Martin N° 143
22.333.444	Ponce	Ángel	Los castores N° 345
33.444.555	Aliaga	Mariana	Marcos Zar Sur N° 345

Dni	Teléfonos
11.222.333	(2901)15514897
11.222.333	(2901)422631
22.333.444	(2901)15615729
22.333.444	(2901)15404947
22.333.444	(2901)433085
33.444.555	(2901)15476924

- b) Sea **CUENTA(AT)**  
AT = {banco, sucursal, número, tipo, titulares\*}

Banco	Sucursal	Número	Tipo	Titulares
Río	Ushuaia	3002	Cuenta Corriente	Pedro Perez
Río	Rio Grande	2000	Caja de Ahorros	Maria Gomez, Pedro Perez
HSBC	Ushuaia	1001	Cuenta Corriente	Ester Gonzalez, Maria Gomez, Julian Bucefi

**El atributo que viola la 1FN es titulares, en este caso la clave de la relación son todos los atributos excepto titulares, por lo que para que esté en 1FN deberemos repetir esos valores para cada valor de atributo Titulares.**



<u>Banco</u>	<u>Sucursal</u>	<u>Número</u>	<u>Tipo</u>	<u>Titulares</u>
Río	Ushuaia	3002	Cuenta Corriente	Pedro Perez
Río	Rio Grande	2000	Caja de Ahorros	Maria Gomez,
Río	Rio Grande	2000	Caja de Ahorros	Pedro Perez
HSBC	Ushuaia	1001	Cuenta Corriente	Ester Gonzalez
HSBC	Ushuaia	1001	Cuenta Corriente	Maria Gomez
HSBC	Ushuaia	1001	Cuenta Corriente	Julian Bucefi

- c) Sea **LIBRO(AT, DF)**  
 AT = {isbn, título, autores\*}  
 DF = {isbn → título}  
 CLAVE = isbn

***El atributo que viola la 1FN es autores, lo extraemos a una nueva relación junto con la clave de la relación de la que se desprende.***

LIBRO(AT, DF)	LIBRO-AUTOR(AT)
AT = {isbn, título}	AT = {isbn, autor}
DF = {isbn → título}	CLAVE: isbn, autor
CLAVE: isbn	

- d) Sea **EMPLEADO(AT, DF)**  
 AT = {documento, nombre, puesto, salario, correos\*}  
 DF = {documento → nombre, puesto; puesto → salario}  
 CLAVE = documento

***El atributo que viola la 1FN es correos, lo extraemos a una nueva relación junto con la clave de la relación de la que se desprende.***

EMPLEADO(AT, DF)	EMPLEADO-CORREO(AT)
AT = {AT = {documento, nombre, puesto, salario}}	AT = {documento, correo}
DF = { documento → nombre, puesto; puesto → salario}	CLAVE: documento, correo
CLAVE: documento	

**Ejercicio 7:** Dadas las siguientes relaciones, normalizar a 2FN. ¿Cómo quedaría cada relación luego de normalizar?

**Segunda forma normal (2NF):** La segunda forma normal tiene sus bases en el concepto de dependencia funcional plena y las interrelaciones que se establecen entre los atributos primos y no primos de la relación.

Para que una relación esté en 2NF, ésta deberá estar en 1NF y todo atributo no primo debe depender funcionalmente de cada una de las claves. Esto quiere decir que cuando en una relación se tienen atributos no primos que dependen funcionalmente de algún subconjunto de alguna clave, ésta no se encuentra en 2NF.

Al proyectar en nuevas relaciones, la relación original mantiene su clave compuesta y los atributos que de ella dependen, inclusive si no hubieren atributos que dependan completamente de la clave de la relación, ésta deberá mantenerse para poder realizar la reconstrucción de la relación original.

a) Sea **FACTURA(AT, DF)**

AT = {número, renglón, fecha, descripción, cantidad, precio}

DF = {número → fecha; número, renglón → descripción, cantidad, precio}

CLAVE = número, renglón

- **Esta relación ya está en 1FN.**
- **En esta relación la clave a su vez es única clave candidata.**
- **Encontremos los atributos primos y no primos:**
  - At. primos: número, renglón
  - At. no primos: fecha, descripción, cantidad, precio
- **Identificamos dependencias donde un atributo no primo dependa parcialmente de la clave:**
  - número → fecha
- **Proyectamos la dependencia que viola la 2FN, quedando dos relaciones:**

R1(AT, DF)

AT = {número, renglón, descripción, cantidad, precio}

DF = {número, renglón → descripción, cantidad, precio}

CLAVE: número, renglón

R2(AT, DF)

AT = {número, fecha}

DF = {número → fecha}

CLAVE: número

b) Sea **EXPEDIENTE(AT, DF)**

AT = {año, letra, número, área, fecha, fojas}

DF = {año, letra, número → fecha, fojas; letra → área}

CLAVE = año, letra, número

- ***Esta relación ya está en 1FN.***
- ***En esta relación la clave a su vez es única clave candidata.***
- ***Encontremos los atributos primos y no primos:***
  - At. primos: año, letra, número*
  - At. no primos: área, fecha, fojas*
- ***Identificamos dependencias donde un atributo no primo dependa parcialmente de la clave:***
  - letra → área*
- ***Proyectamos la dependencia que viola la 2FN, quedando dos relaciones:***

R1(AT, DF)

AT = {letra, área}

DF = {letra → área}

CLAVE: letra

R2(AT, DF)

AT = {año, letra, número, fecha, fojas}

DF = {año, letra, número → fecha, fojas}

CLAVE: año, letra, número

c) Sea **VEHICULO(AT, DF)**

AT = {patente, marca, modelo, documento, apellido, nombre}

DF = {patente → marca, modelo, documento; documento → apellido, nombre}

CLAVE = patente

- ***Esta relación ya está en 1FN.***
- ***En esta relación la clave a su vez es única clave candidata.***
- ***Encontremos los atributos primos y no primos:***
  - At. primos: patente*
  - At. no primos: marca, modelo, documento, apellido*

***Esta relación está en 2FN, cada atributo no primo depende funcionalmente de la clave (la clave no es compuesta y es única clave candidata, por lo que no puede haber una dependencia parcial), y podemos observar la DF documento → apellido, pero***

**documento es una atributo no primo del que depende funcionalmente apellido (otro atributo no primo), por lo que no viola la 2FN.**

d) Sea **AREA(AT, DF)**

AT = {código, área, legajo, apellido, nombre, desde, ascensos\*}

DF = {código → área; legajo → apellido, nombre; código, legajo → desde}

CLAVE = código, legajo

- **Esta relación NO está en 1FN, el atributo ascensos tiene múltiples valores, lo extraemos a una nueva relación junto con la clave de la relación de la que se desprende.**

AREA(AT, DF)

AT = {código, área, legajo, apellido, nombre, desde}

DF = { código → área; legajo → apellido, nombre; código, legajo → desde}

CLAVE: código, legajo

AREA-ASCENSO(AT)

AT = {código, legajo, ascenso}

CLAVE: código, legajo, ascenso

- **Ahora la relación AREA se encuentra en 1FN, pero no está en 2FN.**
- **En esta relación la clave a su vez es única clave candidata.**
- **Encontremos los atributos primos y no primos:**

At. primos: código, legajo

At. no primos: área, apellido, nombre, desde

- **Identificamos dependencias donde un atributo no primo dependa parcialmente de la clave:**

código → área

legajo → apellido, nombre

- **Proyectamos las dependencias que violan la 2FN, quedando tres relaciones:**

R1(AT, DF)

AT = {código, área}

DF = {código → área}

CLAVE: código

R2(AT, DF)

AT = {legajo, apellido, nombre}

DF = {legajo → apellido, nombre}

CLAVE: legajo

R3(AT, DF)

AT = {código, legajo, desde}

DF = {código, legajo → desde}

CLAVE: código, legajo

**Ejercicio 8:** Dadas las siguientes relaciones, normalizar a 3FN. ¿Cómo quedaría cada relación luego de normalizar?

**Tercera forma normal (3FN):** Una relación se encuentra en tercera forma normal siempre que esté en 2FN y todos los atributos no primos no dependan transitivamente de alguna clave. Esta forma normal está basada en el concepto de dependencia transitiva.

Para normalizar una relación a 3FN deberán proyectarse las relaciones teniendo en cuenta la eliminación de las dependencias transitivas.

a) Sea **EMPLEADO(AT, DF)**

AT = {documento, nombre, puesto, salario, correos\*}

DF = {documento → nombre, documento → puesto, puesto → salario}

CLAVE = documento

- ***Esta relación NO está en 1FN, el atributo correos tiene múltiples valores, lo extraemos a una nueva relación junto con la clave de la relación de la que se desprende.***

EMPLEADO(AT, DF)

AT = {documento, nombre, puesto, salario }

DF = { documento → nombre; documento → puesto; puesto → salario }

CLAVE: documento

EMPLEADO-CORREO(AT)

AT = {documento, correo}

CLAVE: documento, correo

- ***Ahora la relación EMPLEADO se encuentra en 1FN, asimismo también se encuentra en 2FN, tenemos una clave simple (única clave candidata), pero no se encuentra en 3FN.***
- ***En esta relación la clave a su vez es única clave candidata.***
- ***Encontremos los atributos primos y no primos:***
  - At. primos: documento*
  - At. no primos: nombre, puesto, salario*
- ***Identificamos dependencias donde un atributo no primo dependa transitivamente de la clave:***

*puesto → salario*

- **Proyectamos la dependencia que viola la 3FN, quedando dos relaciones (además de la existente cuando se normalizó a 1FN):**

R1(AT, DF)

AT = {puesto, salario}

DF = {puesto → salario}

CLAVE: puesto

R2(AT, DF)

AT = {documento, nombre, puesto}

DF = {documento → nombre, puesto}

CLAVE: documento

b) Sea **FACTURA(AT, DF)**

AT = {número, fecha, tipo, documento, nombre}

DF = {número → fecha, tipo, documento; documento → nombre}

CLAVE = número

- **Esta relación ya está en 1FN, asimismo también se encuentra en 2FN, pero no se encuentra en 3FN.**
- **En esta relación la clave a su vez es única clave candidata.**
- **Encontremos los atributos primos y no primos:**

*At. primos: número*

*At. no primos: fecha, tipo, documento, nombre*

- **Identificamos dependencias donde un atributo no primo dependa transitivamente de la clave:**

*documento → nombre*

- **Proyectamos la dependencia que viola la 3FN, quedando dos relaciones:**

R1(AT, DF)

AT = {documento, nombre}

DF = {documento → nombre}

CLAVE: documento

R2(AT, DF)

AT = {número, fecha, tipo, documento}

DF = {número → fecha, tipo, documento}

CLAVE: número

c) Sea **CUENTA(AT, DF)**

AT = {banco, sucursal, número, tipo, ciudad, documento, apellido, nombre}

DF = {banco, sucursal, número → tipo, documento; documento → apellido, nombre; sucursal → ciudad}

CLAVE = banco, sucursal, número

- ***Esta relación ya está en 1FN, pero no se encuentra en 2FN.***
- ***En esta relación la clave a su vez es única clave candidata.***
- ***Encontremos los atributos primos y no primos:***  
     *At. primos: banco, sucursal, número*  
     *At. no primos: tipo, documento, apellido, nombre, ciudad*
- ***Identificamos dependencias donde un atributo no primo dependa parcialmente de la clave:***  
     *sucursal → ciudad*
- ***Proyectamos las dependencias que violan la 2FN, quedando dos relaciones:***

R1(AT, DF)

AT = {sucursal, ciudad}

DF = {sucursal → ciudad}

CLAVE: sucursal

R2(AT, DF)

AT = {banco, número, sucursal, documento, tipo, nombre, apellido}

DF = {banco, número, sucursal → documento, tipo; documento → apellido, nombre}

CLAVE: banco, número, sucursal

- ***Pero la relación R2 no se encuentra en 3FN, volvemos a identificar atributos primos y no primos:***  
     *At. primos: banco, número, sucursal*  
     *At. no primos: documento, tipo, apellido, nombre*
- ***Identificamos dependencias donde un atributo no primo dependa transitivamente de la clave:***  
     *banco, número, sucursal → documento y documento → apellido, nombre*
- ***Proyectamos la dependencia que viola la 3FN, quedando dos relaciones (además de la primera que se proyectó para llegar a 2FN):***

R1(AT, DF)

AT = {sucursal, ciudad}

DF = {sucursal → ciudad}

CLAVE: sucursal

R2(AT, DF)

AT = {documento, apellido, nombre}

DF = {documento → apellido, nombre}

CLAVE: documento

R3(AT)

AT = {banco, número, sucursal, documento, tipo}

DF = {banco, número, sucursal → documento, tipo}

CLAVE: banco, número, sucursal

d) Sea **NOTA(AT, DF)**

AT = {año, número, fecha, tema, legajo, apellido, nombre, firmantes\*}

DF = {año, número → fecha, tema; año → legajo; legajo → apellido, nombre}

CLAVE = año, número

- ***Esta relación NO está en 1FN, el atributo firmantes tiene múltiples valores, lo extraemos a una nueva relación junto con la clave de la relación de la que se desprende.***

NOTA(AT, DF)

AT = { año, número, fecha, tema, legajo, apellido, nombre}

DF = { año, número → fecha, tema; año → legajo; legajo → apellido, nombre}

CLAVE: año, número

NOTA-FIRMANTES(AT)

AT = {año, numero, firmante}

CLAVE: año, número, firmante

- ***Ahora la relación NOTA se encuentra en 1FN, pero no se encuentra en 2FN, tenemos una clave compuesta (año, número) y una dependencia parcial de la clave de un atributo no primo.***
- ***En esta relación la clave a su vez es única clave candidata.***
- ***Encontremos los atributos primos y no primos:***

*At. primos: año número*

*At. no primos: fecha, tema, legajo, apellido, nombre*

- ***Identificamos dependencias donde un atributo no primo dependa parcialmente de la clave:***

*año → legajo*

***Proyectamos la dependencia que viola la 2FN, quedando dos relaciones (además de la primera cuando convertimos a 1FN):***

R1(AT, DF)

AT = {año, número, fecha, tema}

DF = {año, número → fecha, tema}

CLAVE: año, número

R2(AT, DF)

AT = {año, legajo, apellido, nombre}

DF = {año → legajo; legajo → apellido, nombre}

CLAVE: año



- ***Pero la relación R2 no se encuentra en 3FN, volvemos a identificar atributos primos y no primos:***
  - At. primos: año*
  - At. no primos: legajo, apellido, nombre*
- ***Identificamos dependencias donde un atributo no primo dependa transitivamente de la clave:***
  - año → legajo   legajo → apellido, nombre*
- ***Proyectamos la dependencia que viola la 3FN, quedando dos relaciones (además de la primera que se proyectó para llegar a 1FN y la segunda proyección para llevarla a 2FN):***

NOTA-FIRMANTES(AT)

AT = {año, numero, firmante}

CLAVE: año, número, firmante

R1(AT, DF)

AT = {año, número, fecha, tema}

DF = {año, número → fecha, tema}

CLAVE: año, número

R2(AT, DF)

AT = {año, legajo}

DF = {año → legajo}

CLAVE: año

R3(AT, DF)

AT = {legajo, apellido, nombre}

DF = {legajo → apellido, nombre}

CLAVE: legajo

**Ejercicio 9:** Dadas las siguientes relaciones, normalizar a FNBC. ¿Cómo quedaría cada relación luego de normalizar?

**Forma normal de Boyce-Codd:** Surge como una solución a las anomalías de actualización que se producen en una relación en 3NF cuando se cumple:

- la relación tiene varias claves candidatas
- las claves candidatas son compuestas
- las claves candidatas se solapan o tienen atributos en común

La BCNF se utiliza para subsanar las anomalías de actualización en relaciones que tuvieran varias claves candidatas compuestas que se solapan.

Una relación está en BCNF si todo determinante es una clave candidata.

a) Sea **ENTRENAMIENTO(AT, DF)**

AT = {perro, entrenamiento, entrenador}

DF = {perro, entrenamiento → entrenador; entrenador → entrenamiento}

CLAVE = perro, entrenamiento y perro, entrenador

- ***Esta relación está en 1FN (no tenemos atributos con múltiples valores), está en 2FN (no tenemos atributos no primos que dependen parcialmente de la clave, aunque tenemos entrenador → entrenamiento ambos son atributos primos), está en 3FN (no hay atributos no primos que dependan transitivamente de la clave), pero tenemos la DF entrenados → entrenamiento donde su implicante no es CC, por lo que no está en FNBC.***
- ***Proyectamos la DF que viola la FNBC:***

R1(AT, DF)

AT = {entrenador, entrenamiento}

DF = {entrenador → entrenamiento}

CLAVE: entrenador

R3(AT)

AT = {perro, entrenamiento}

CLAVE: perro, entrenamiento

b) Sea **PELICULA(AT, DF)**

AT = {título, año, duración, género, director, actor}

DF = {título, año → duración, género, director}

CLAVE = título, año, actor

- ***Esta relación ya está en 1FN, pero no se encuentra en 2FN.***
- ***En esta relación la clave a su vez es única clave candidata.***
- ***Encontremos los atributos primos y no primos:***  
*At. primos: título, año, actor*  
*At. no primos: duración, género, director*
- ***Identificamos dependencias donde un atributo no primo dependa parcialmente de la clave:***  
*título, año → duración, género, director*
- ***Proyectamos las dependencias que violan la 2FN, quedando dos relaciones:***

R1(AT, DF)

AT = {título, año, duración, género,  
director, actor}

DF = {título, año → duración, género,  
director}

CLAVE: título, año

R2(AT)

AT = {título, año, actor}

CLAVE: título, año, actor

- ***Ambas relaciones ya se encuentran en FNBC.***

c) Sea **R(AT, DF)**

AT = {A, B, C, D, E, F}

DF = {A, B → C, D, E; D → B; E → F}

CLAVE = A, B

- ***Esta relación ya está en 1FN, asimismo también se encuentra en 2FN, pero no se encuentra en 3FN.***
- ***En esta relación la clave NO es única clave candidata, AD también los es.***

- ***Encontremos los atributos primos y no primos:***

At. primos: A, B, D

At. no primos: C, E, F

- ***Identificamos dependencias donde un atributo no primo dependa transitivamente de la clave:***

A, B → E   E → F

- ***Proyectamos la dependencia que viola la 3FN, quedando dos relaciones:***

R1(AT, DF)

AT = {E, F}

DF = {E → F}

CLAVE: E

R2(AT, DF)

AT = {A, B, C, D, E}

DF = {A, B → E, C, D; D → B}

CLAVE: A, B y A, D

- ***La relación no se encuentra en FNBC, el implicante D no es CC de la relación. Proyectamos la DF en una nueva relación:***

R1(AT, DF)

AT = {E, F}

DF = {E → F}

CLAVE: E

R2(AT, DF)

AT = {B, D}

DF = {D → B}

CLAVE: D

$R_3(AT, DF)$   
 $AT = \{A, C, D, E\}$   
 $DF = \{A, D \rightarrow E, C\}$   
CLAVE: A, D

d) Sea  $R(AT, DF)$

$AT = \{A, B, C, D, E, F, G\}$   
 $DF = \{A, B \rightarrow C, E; E \rightarrow F, G; B \rightarrow D\}$   
CLAVE = A, B

- ***Esta relación ya está en 1FN, pero no se encuentra en 2FN.***
- ***En esta relación la clave a su vez es única clave candidata.***
- ***Encontremos los atributos primos y no primos:***  
At. primos: A, B  
At. no primos: C, D, E, F, G
- ***Identificamos dependencias donde un atributo no primo dependa parcialmente de la clave:***  
 $B \rightarrow D$
- ***Proyectamos las dependencias que violan la 2FN, quedando dos relaciones:***

$R_1(AT, DF)$   
 $AT = \{B, D\}$   
 $DF = \{B \rightarrow D\}$   
CLAVE: B

$R_2(AT, DF)$   
 $AT = \{A, B, C, E, F, G\}$   
 $DF = \{A, B \rightarrow C, E; E \rightarrow F, G\}$   
CLAVE: A, B

- ***La relación  $R_2$  no se encuentra en 3FN.***
- ***Encontremos los atributos primos y no primos:***  
At. primos: A, B  
At. no primos: C, E, F, G
- ***Identificamos dependencias donde un atributo no primo dependa transitivamente de la clave:***  
 $A, B \rightarrow E \quad E \rightarrow F, G$

- **Proyectamos la dependencia que viola la 3FN, quedando dos relaciones:**

R1(AT, DF)  
AT = {B, D}  
DF = {B → D}  
CLAVE: B

R2(AT, DF)  
AT = {E, F, G }  
DF = {E → F, G}  
CLAVE: E

R2(AT, DF)  
AT = {A, B, C, E }  
DF = {A, B → C, E}  
CLAVE: A, B

- **Estas relaciones ya se encuentran en FNBC.**



Base de Datos I. Práctica 4. Dependencias Funcionales, formas normales y normalización by Lic. Mariana Adó; Lic. Nadia Ramos is licensed under a Creative Commons Reconocimiento-NoComercial – CompartirIgual 4.0 Internacional License.