

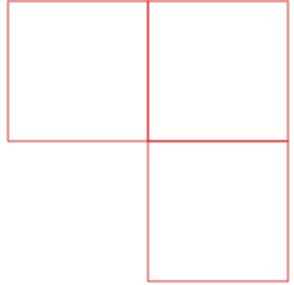
UF3: Modelo Relacional

3.4 Normalización - ejemplo

Ve más allá



Índice



1. Teoría Referencial
2. Ejercicio Resuelto
3. Ejercicio planteado



Normalización

Resumen

La normalización de bases de datos relacionales toma un esquema relacional y le aplica un conjunto de **técnicas** para producir un nuevo esquema, que representa la **misma información**, pero:

- contiene menos redundancias
- evita posibles anomalías en las inserciones, actualizaciones y borrados.



Normalización

Profundizando: Teoría de conjuntos

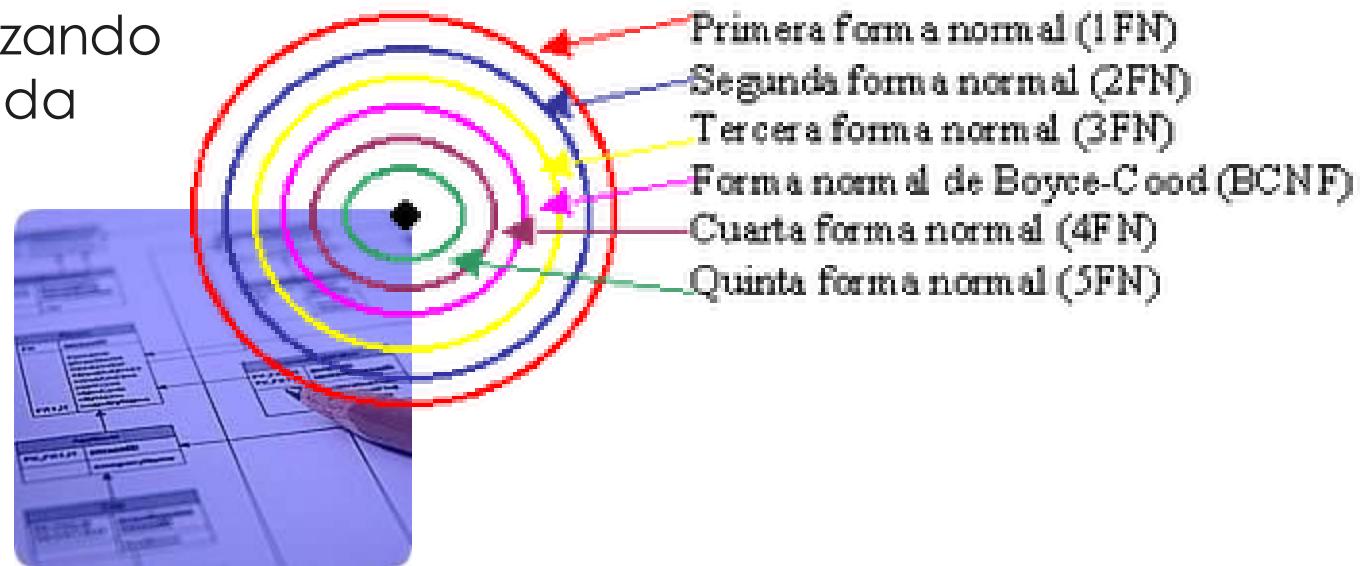
Una base de datos relacional puede considerarse como un conjunto de relaciones o tablas de la forma $R(A_1, \dots, A_n)$, donde R es el nombre de la relación, que se define por una serie de atributos A_i . Sobre las tablas relacionales se pueden definir diferentes **restricciones**.

- **Integridad de entidad:** restricción que nos indica que cada entidad representada por una tupla tiene que ser diferente de las demás en su relación, es decir, debe haber algunos atributos cuyos valores identifiquen únicamente las tuplas.
- **Integridad referencial:** una clave ajena solo debe contener valores que, o bien sean nulos, o bien existan en la relación referenciada por la clave ajena.

Normalización

Proceso

El proceso de normalización consiste en comprobar en secuencia si el esquema original está en 1FN, 2FN y 3FN, analizando las dependencias funcionales en cada paso.



Ejemplo

Enunciado

Tenemos una empresa pública donde los puestos de trabajo están regulados por el Estado, de modo que las condiciones salariales están determinadas por el puesto.

Se ha creado el siguiente **esquema relacional**:

EMPLEADOS (nss, nombre, puesto, salario, emails)

PK: nss;

nss	nombre	puesto	salario	emails
111	Juan Pérez	Jefe de Área	3000	juanp@ecn.es; jefe2@ecn.es
222	José Sánchez	Administrativo	1500	jsanchez@ecn.es
333	Ana Díaz	Administrativo	1500	adiaz@ecn.es; ana32@gmail.com
...

Ejemplo

1FN

Una tabla está en 1FN si sus atributos contienen **valores atómicos**.

En el ejemplo, podemos ver que el atributo **emails** puede contener más de un valor, por lo que viola 1FN.

En general, tenemos una relación R con clave primaria K. Si un atributo M viola la condición de 1FN, tenemos dos opciones:

- **Solución 1:** duplicar los registros con valores repetidos.
- **Solución 2:** separar el atributo que viola 1FN en una tabla.



Ejemplo

1FN - Solución 1: duplicar registros

En general, sustituimos R por una nueva relación modificada R', en la cual:

- El atributo M que violaba 1FN se elimina.
- Se incluye un nuevo atributo M' que solo puede contener **valores simples**, de modo que para una tupla con **n valores duplicados en M**, en la nueva relación habrá **n tuplas**, que sólo varían en que cada una de ellas guarda uno de los valores que había en M.

La clave primaria de R' es (K, M'), dado que podrá haber valores de K repetidos, para los valores multivaluados en M.

Ejemplo

1FN - Solución 1: duplicar registros

Siguiendo el ejemplo, tendríamos el siguiente esquema para la nueva tabla **EMPLEADOS'(a)** con clave primaria (nss, email):

nss	nombre	puesto	salario	email
111	Juan Pérez	Jefe de Área	3000	juanp@ecn.es
111	Juan Pérez	Jefe de Área	3000	jefe2@ecn.es
222	José Sánchez	Administrativo	1500	jsanchez@ecn.es
333	Ana Díaz	Administrativo	1500	adiaz@ecn.es
333	Ana Díaz	Administrativo	1500	ana32@gmail.com
...



Ejemplo

1FN - Solución 2: nueva tabla

En general, esta solución pasa por:

- Sustituir R por una nueva relación modificada R' que no contiene el atributo M.
- Crear una nueva relación $N(K, M')$, es decir, una relación con una clave ajena K referenciando R' , junto al atributo M' , que es la variante monovaluada del atributo M.

La nueva relación N tiene como clave (K, M') .



Ejemplo

1FN - Solución 2: nueva tabla

Siguiendo el ejemplo, tendríamos el siguiente esquema para la nueva tabla **EMPLEADOS'(b)**, y además tendríamos una nueva tabla **EMAILS** con clave primaria (nss, email):

nss	nombre	puesto	salario
111	Juan Pérez	Jefe de Área	3000
222	José Sánchez	Administrativo	1500
333	Ana Díaz	Administrativo	1500
...

nss	email
111	juanp@ecn.es
111	jefe2@ecn.es
222	jsanchez@ecn.es
333	adiaz@ecn.es
333	ana32@gmail.com
...	...



Ejemplo

2FN

Un esquema está en 2FN si:

- Está en 1FN.
- Todos sus atributos, que no son de la clave principal, tienen dependencia funcional completa respecto de todas las claves existentes en el esquema. En otras palabras, para determinar cada atributo no clave se necesita la clave primaria completa.

La 2FN se aplica a las relaciones que tienen claves primarias compuestas por dos o más atributos. Si una relación está en 1FN y su clave primaria es simple (tiene un solo atributo), entonces también está en 2FN.



Ejemplo

2FN

Por tanto, de las soluciones anteriores, la tabla EMPLEADOS'(b) está en 1FN (y la tabla EMAILS no tiene atributos no clave), por lo que el esquema está en 2FN.

Sin embargo, tenemos que examinar las dependencias funcionales de los atributos no clave de EMPLEADOS'(a).

Las dependencias funcionales que tenemos son las siguientes:

$nss \rightarrow nombre, salario, email$

$puesto \rightarrow salario$

Como la clave es $(nss, email)$, las dependencias de nombre, salario y email son incompletas, por lo que la relación **no está en 2FN**.



Ejemplo

2FN

En general, tendremos que observar los atributos no clave que dependan de parte de la clave.

Para solucionar este problema, tenemos que hacer lo siguiente para los grupos de atributos con dependencia incompleta M:

- Eliminar de R el atributo M.
- Crear una nueva relación N con el atributo M y la parte de la clave primaria K de la que depende, que llamaremos K'.
- La clave primaria de la nueva relación será K'.



Ejemplo

2FN

Siguiendo el ejemplo anterior, crearíamos una nueva relación con los atributos que tienen dependencia incompleta (nombre, puesto y salario):

nss	nombre	puesto	salario
111	Juan Pérez	Jefe de Área	3000
222	José Sánchez	Administrativo	1500
333	Ana Díaz	Administrativo	1500
...

Y al eliminar de la tabla original estos atributos nos quedaríamos con lo siguiente:

nss	email
111	juanp@ecn.es
111	jefe2@ecn.es
222	jsanchez@ecn.es
333	adiaz@ecn.es
333	ana32@gmail.com
...	...

Que es lo mismo que en la otra opción de solución para el problema de 1FN.

Ejemplo

3FN

Una relación está en tercera forma normal si, y sólo si:

- Está en 2FN
- Cada atributo que no está incluido en la clave primaria no depende transitivamente de la clave primaria.

Por lo tanto, a partir de un esquema en 2FN, tenemos que buscar dependencias funcionales entre atributos que no estén en la clave.

- En general, tenemos que buscar **dependencias transitivas de la clave**, es decir, secuencias como : $K \rightarrow A$ y $A \rightarrow B$, donde A y B no pertenecen a la clave.
- La solución está en separar en una tabla adicional N el/los atributos B, y poner como clave primaria de N el atributo que define la transitividad A.

Ejemplo

3FN

Siguiendo el ejemplo anterior, podemos detectar la siguiente transitividad:

nss->puesto

puesto->salario

Por lo tanto, tendríamos que crear una nueva tabla PUESTOS, la clave sería el puesto, que también queda como clave ajena referenciando la tabla EMPLEADOS. La otra tabla (EMAILS) queda como estaba.

<u>nss</u>	<u>email</u>
111	juanp@ecn.es
111	jefe2@ecn.es
222	jsanchez@ecn.es
333	adiaz@ecn.es
333	ana32@gmail.com
...	...

<u>puesto</u>	<u>salario</u>
Jefe de Área	3000
Administrativo	1500
...	...

<u>nss</u>	<u>nombre</u>	<u>puesto</u>
111	Juan Pérez	Jefe de Área
222	José Sánchez	Administrativo
333	Ana Díaz	Administrativo
...

Ejercicio propuesto

Dada la siguiente tabla de empleados:

DNI	NOMBRE	APELLIDOS	DIRECCIÓN	C_POST	POBLACIÓN	PROVINCIA
413245-B	JUAN	RAMOS	C/Las cañas 59 C/Pilón 12	19005 45589	GUADALAJARA CALERUELA	GUADALAJARA TOLEDO
23456-J	PEDRO	PÉREZ	C/Vitoria 3 C/El altozano	28804 10392	ALCALÁ DE HENARES BERROCALEJO	MADRID CÁCERES
34561-B	MARÍA	RODRÍGUEZ	C/Sanz Vázquez 2	19004	GUADALAJARA	GUADALAJARA
222346-J	JUAN	CABELLO	C/El ensanche 3 C/Los abedules 10	28802 10300	ALCALÁ DE HENARES NAVALMORAL DE LA MATA	MADRID CÁCERES

Realiza un documento formal, en el que debes normalizar hasta la 3FN explicando cada uno de los pasos que has dado.

Debes tener en cuenta que las dependencias son:

DNI -> NOMBRE, APELLIDOS

C_POST -> POBLACIÓN, PROVINCIA

Y su Clave primaria es DNI, DIRECCIÓN, C_POST