

Universidad Cenfotec

MBD101 – Programación Avanzada SQL

Normalización de Bases de Datos

¿Qué es normalización?

- Proceso mediante el cual los esquemas de relación insatisfactorios se descomponen repartiendo sus atributos entre esquemas de relación más pequeños que poseen propiedades deseables.
- En este proceso, se somete un esquema de relación a una serie de pruebas para “certificar” si cumple o no con las reglas definidas por una cierta forma normal.
- Las formas normales son conjuntos de condiciones que debe cumplir una relación para considerarse en un estado de normalización deseable, lo cual evita problemas de integridad y duplicidad de datos, entre otros.

Dependencia funcional

- Se dice que existe una dependencia funcional entre un conjunto de campos X y un conjunto de campos Y , si para cada conjunto de valores distintos para X sólo hay un conjunto de valores distintos para Y .
- Al conjunto X se le llama “determinante” y al conjunto Y “dependiente”.
- Una dependencia funcional es una relación que se da entre dos valores o conjuntos de valores, tal que al modificar el conjunto determinante, esto puede afectar los valores del conjunto dependiente.
- En matemáticas esto es análogo a cambiar una función $f(x)$ por otra, obviamente al cambiar la función (determinante) el resultado (dependiente puede probablemente variar).

Dependencia funcional

- La dependencia funcional se representa con una flecha que va desde el determinante hasta el dependiente.
- Cédula_Cliente → Nombre_Cliente (Al revisar una cédula diferente se podría encontrar el nombre de otro cliente):
- Ejs:
 - 1-2236-5655 (Cédula) → María Solano (Nombre_Cliente)
 - 2-5633-5655 (Cédula) → Pedro Solano (Nombre_Cliente)

Primera forma normal (1NF)

- La 1NF exige la existencia de una llave primaria, dado que una tabla en 1NF no permite filas duplicadas y no se puede garantizar esto al 100% sin la existencia de una llave primaria o clave.
- Esta forma normal prohíbe la existencia de atributos multivalorados, o sea de múltiples valores dentro de un solo atributo. (Ej. Un campo teléfonos, donde se escriben varios teléfonos separados con coma).

Primera forma normal (1NF)

- Asimismo en la forma más purista de las reglas de 1NF, la existencia múltiples atributos para almacenar un mismo dato tampoco es permitida, en algunas versiones más relajadas esto si es aceptable, aunque es poco recomendable. (Ej. Una tabla o relación con los siguientes atributos: Teléfono1, Teléfono2, Teléfono3).
- Aunque la definición de Codd en el modelo relacional para la 1NF permite que haya campos nulos, algunas definiciones de 1NF no lo consideran apropiado. Ya que si un campo puede ser nulo este es candidato a ser colocado en una relación o tabla aparte con una llave foránea solo a los registros de la tabla original que requieran el dato.

Ejemplo de normalización a 1NF

En este ejemplo nótese que nómina es como un número de empleado el cual identifica de manera única el registro, se muestra subrayado para mostrar que es la clave principal.

nómina	nombre	dirección	teléfono	clave_depto	departamento	sueldo
345232	Juan Pérez	Florida 23	55667788	MKT	Mercadotecnia	\$8,500
564738	Luisa López	B. Juárez 145	56252534 56132436	FIN	Finanzas	\$11,000

1. Esta es la tabla original en ella aparece dos teléfonos para Luisa López en el mismo campo.
NO esta en 1NF

2. Separar en dos registros los datos de Luisa elimina los atributos multivalorados (teléfonos) pero ahora violamos la regla de tener una clave primaria porque sale
NO esta en 1NF

nómina	nombre	dirección	teléfono	clave_depto	departamento	sueldo
345232	Juan Pérez	Florida 23	55667788	MKT	Mercadotecnia	\$8,500
564738	Luisa López	B. Juárez 145	56252534	FIN	Finanzas	\$11,000
564738	Luisa López	B. Juárez 145	56132436	FIN	Finanzas	\$11,000

EMPLEADO						
nómina	nombre	dirección	clave_depto	departamento	sueldo	
345232	Juan Pérez	Florida 23	MKT	Mercadotecnia	\$8,500	
564738	Luisa López	B. Juárez 145	FIN	Finanzas	\$11,000	

EMPLEADO-TELÉFONO	
nómina	teléfono
345232	55667788
564738	56252534
564738	56132436

3. Al separar los teléfonos en una tabla aparte cuya clave primaria es nómina y teléfono y sacar el teléfono de la tabla inicial, ahora tenemos dos tablas en 1NF.

Segunda forma normal (2NF)

- Una tabla está en segunda forma normal si esta en 1NF y además todos los atributos que NO forman parte de la clave primaria dependen funcionalmente de TODA la clave primaria.
- Tip: Una tabla en 1NF está automáticamente en 2NF si su clave primaria está formada por un solo atributo y los datos que no son clave son dependientes de la llave.
- A veces podría darse que haya una llave compuesta definida en una tabla. Si los valores que tenemos en la tabla son dependientes de sólo un subconjunto de los campos de la llave, la tabla no estaría en 2NF y se debería de pasar este valor con dependencia parcial a otra tabla.
- Algunas tablas tienen otras llaves o claves aparte de la primaria, si los datos que no son llave dependen de estas es posible que también deban sacarse de la tabla, ya que la tabla no estaría en 2NF

Ejemplo de normalización en 2NF

Recuerde que las columnas cuyos nombres están subrayados son los que conforman la llave primaria de la tabla mostrada en el ejemplo.

1. Aunque esta tabla esta en 1NF porque tiene una llave primaria y no tiene atributos multivalorados ni registros duplicados, no se encuentra en 2NF porque el único valor que depende funcionalmente de toda la llave es la Nota.

El Nombre y Apellido1, dependen funcionalmente sólo del DNI (que es una identificación de persona) pero no del Cod_Curso

ALUMNOS				
<u>DNI</u>	<u>Cod_Curso</u>	Nombre	Apellido1	Nota
12121219A	34	Pedro	Valiente	9
12121219A	25	Pedro	Valiente	8
3457775G	34	Ana	Fernández	6
5674378J	25	Sara	Crespo	7
5674378J	34	Sara	Crespo	6

Ejemplo de normalización en 2NF

2. Se hace la división de la tabla en dos:

En la primera se almacena la información de alumnos y donde van a quedar ligados sólo del DNI los datos de Nombre y Apellido1 (los cuales tenían una dependencia parcial en el cuadro anterior).

En la segunda se mantiene el atributo Nota, el cual depende del CodCurso y del DNI (este atributo depende de la toda la llave en vez de depender de un subconjunto de esta)

Por tanto ambas tablas quedarían en 2NF

ALUMNOS		
<u>DNI</u>	Nombre	Apellido1
12121219A	Pedro	Valiente
3457775G	Ana	Fernández
5674378J	Sara	Crespo

ASISTENCIA		
<u>DNI</u>	<u>Cod Curso</u>	Nota
12121219A	34	9
12121219A	25	8
3457775G	34	6
5674378J	25	7
5674378J	34	6

Tercera forma normal (3NF)

- Una tabla o relación está en tercera forma normal cuando está en 2NF y además no existen dependencias funcionales transitivas entre atributos no llave.
- Recuerde que: $A \rightarrow B$ significa que B depende funcionalmente de A.
- Dependencia transitiva (siendo A la llave primaria): Si $A \rightarrow B$ y $B \rightarrow C$, entonces $A \rightarrow C$
- Se forma una dependencia transitiva entre atributos no llave porque B y C no son llave y al depender uno del otro y uno de ellos depender de A se completa la transitividad.

Ejemplo de normalización en 3NF

nómina	nombre	dirección	teléfono	clave_depto	departamento	suelo
345232	Juan Pérez	Florida 23	55667788	MKT	Mercadotecnia	\$8,500
564738	Luisa López	B. Juárez 145	56252534 56132436	FIN	Finanzas	\$11,000

1. Esta es la tabla original en ella aparece dos teléfonos para Luisa López en el mismo campo.
NO esta en 1NF

2. Al separar los teléfonos en una tabla aparte cuya clave primaria es nómina y teléfono y sacar el teléfono de la tabla inicial, ahora tenemos dos tablas en 1NF.

EMPLEADO						
nómina	nombre	dirección	clave_depto	departamento	suelo	
345232	Juan Pérez	Florida 23	MKT	Mercadotecnia	\$8,500	
564738	Luisa López	B. Juárez 145	FIN	Finanzas	\$11,000	

EMPLEADO-TELÉFONO	
nómina	teléfono
345232	55667788
564738	56252534
564738	56132436

3. Como la tabla está en 1NF y tiene una llave primaria de un solo atributo, se considera que está automáticamente en 2NF.

Ejemplo de normalización en 3NF

4. Si vemos la tabla en 2NF vemos que el atributo clave_depto depende funcionalmente del código del empleado (o nómina), sin embargo la descripción del nombre del departamento (atributo Departamento) depende funcionalmente de clave_depto. Y ni clave_depto ni Departamento forman parte de la llave.

Así entonces, se completa la dependencia transitiva:
nómina. → clave_depto. → Departamento.

Para dejar las tablas en 3NF se requiere eliminar lo que genera esta dependencia transitiva, esto se logra haciendo una tabla aparte de departamentos.

EMPLEADO					
nómina	nombre	dirección	clave_depto	departamento	suelo
345232	Juan Pérez	Florida 23	MKT	Mercadotecnia	\$8,500
564738	Luisa López	B. Juárez 145	FIN	Finanzas	\$11,000

EMPLEADO-TELÉFONO	
nómina	teléfono
345232	55667788
564738	56252534
564738	56132436

DEPARTAMENTO	
clave_depto	Departamento
MKT	Mercadotecnia
FIN	Finanzas

EMPLEADO				
nómina	nombre	dirección	clave_depto	suelo
345232	Juan Pérez	Florida 23	MKT	\$8,500
564738	Luisa López	B. Juárez 145	FIN	\$11,000

EMPLEADO-TELÉFONO	
nómina	teléfono
345232	55667788
564738	56252534
564738	56132436

Resumen de formas normales

1NF a 3NF

- La normalización de la 1ra a 3ra forma normal consiste en que todos los atributos no-clave sean dependientes de "la clave, toda la clave, y nada más que la clave"

Forma normal de Boyce Codd (BCNF)

- Nótese que en 3NF se hizo énfasis a que las dependencias transitivas no se permitían entre atributos NO LLAVE.
- La BCNF es una versión más estricta de 3NF que incluye dependencias similares donde están involucrados atributos llave.
- Llaves Candidatas: Aunque una tabla tenga una llave primaria o clave principal electa, a veces hay ciertas tablas que tienen “llaves candidatas”. Estas son llaves que podrían fungir como llave primaria pero que no fueron electas como tal.

Ejemplo de normalización BCNF

COMPOSITE PRIMARY KEY

STUDENT

3NF

StudentID	Name	Course	Result
1	Tom	Java	60
2	Bob	Database	80
1	Tom	Database	100
3	Elvin	Database	140
2	Bob	Java	100

COMPOSITE PRIMARY KEY

STUDENT

StudentID	Name	Course	Result
1	Tom	Java	60
2	Bob	Database	80
1	Tom	Database	100
3	Elvin	Database	140
2	Bob	Java	100

En este ejemplo se ve como en la tabla Student, tanto la llave primaria electa formada por (StudentID y Course) como la llave candidata (Name y Course) pueden ser usadas como llaves primarias y como tal ambas generan cada una, una dependencia funcional con Result. La existencia de estas llaves candidatas no permite que la tabla se considere en BCNF.

(StudentID, Course) → Result

(Name, Course) → Result

Ejemplo de normalización BCNF

Para eliminar el problema y que las tablas queden en BCNF, se requiere dividir las tablas en dos. De manera que se crea una tabla para la información del estudiante y otra para la información de los cursos matriculados por el estudiante y sus notas (Result).

Form of 3NF

STUDENT			
StudentID	Name	Course	Result
1	Tom	Java	60
2	Bob	Database	80
1	Tom	Database	100
3	Elvin	Database	140
2	Bob	Java	100

Student_info	
StudentID	Name
1	Tom
2	Bob
3	Elvin

Student_course		
StudentID	Course	Result
1	Java	60
2	Database	80
1	Database	100
3	Database	140
2	Java	100

The Table is in Boyce-codd normal form.

Otras formas normales.

- Aunque en la gran mayoría de los casos, las formas normales que hemos visto hasta ahora son más que suficientes, existen algunas anomalías adicionales en bases de datos que pueden presentarse.
- Estas anomalías son tratadas por las formas normales de orden mayor, y aunque existen varias es importante conocer al menos las anomalías que estas atacan.
- Se deja al estudiante la tarea de revisar documentación adicional correspondiente a 4NF, 5NF, DKNF y 6NF a fin de expandir sus conocimientos.
- Un buen punto de inicio puede ser:
 - [https://es.wikipedia.org/wiki/Forma_normal_\(base_de_datos\)](https://es.wikipedia.org/wiki/Forma_normal_(base_de_datos))