

Ejercicios de normalización

Supongamos que, en una Base de Datos de una empresa de venta de computadoras, la información sobre la facturación se encuentra almacenada de la siguiente forma:

Nro. Factura	Fecha	Código cliente	Nombre cliente	Código de producto	Descripción	Cantidad	Precio unitario	Total factura
1	10/01/04	10	Juan Pérez	25	Mouse	5	10	
1	10/01/04	10	Juan Pérez	22	Teclado	10	15	
1	10/01/04	10	Juan Pérez	01	CPU	2	650	
1	10/01/04	10	Juan Pérez					1500
2	20/02/01	36	Julián Ail	12	Pad	10	2	
2	20/02/01	03	Julián Ail	99	Monitor	3	200	
2	20/02/01	03	Julián Ail					620

Normaliza a 3FN la siguiente relación.

Facturación (Nro_Fra, Fecha, Cod_Cliente, Nom_Cliente, Cod_Producto, Descripción, Cantidad, Precio_Unit, Total_Fra)

En primer lugar, debemos comprobar en qué nivel de normalización nos encontramos (si es que la tabla ya está normalizada). Para ello debemos comprobar desde el nivel más bajo de normalización hacia el nivel superior.

1FN: Una base de datos se encuentra en primera forma normal si, y sólo si, en todas sus tablas, cada atributo es atómico.

En la tabla vemos que los datos se repiten al poner un Nro_Fra, excepto con los campos de Cod_Producto, Descripción, Cantidad y Precio_Unit, que van contemplando datos diferentes referidos a diferentes productos que el cliente compró, por lo que estos cuatro atributos NO son atómicos, de modo que habrá que sacarlos de la relación a una tabla aparte.

Facturación (Nro_Fra, Fecha, Cod_Cliente, Nom_Cliente, ~~Cod_Producto, Descripción, Cantidad, Precio_Unit~~, Total_Fra)

Producto (Cod_Producto, Descripción, Cantidad, Precio_Unit)

De modo que la 1FN quedaría tal que así: (hay que unir ambas relaciones mediante una tabla nueva)

Facturación (Nro_Fra, Fecha, Cod_Cliente, Nom_Cliente, Total_Fra)

Producto (Cod_Producto, Descripción, Precio_Unit)

Fra-Producto (Nro_Fra, Cod_Producto, Cantidad) - la cantidad depende del Nro_Fra y Cod_Prod

Concluimos que la relación empleado cumple la 1FN.

1º CFGS - Bases de datos

2FN: Una base de datos se encuentra en 2FN si, solo si, está en 1FN y además todos los atributos que no forman parte de la clave principal tienen dependencia funcional completa de ella.

El atributo *Nom_Cliente* no depende del *Nro_Fra* solamente, sino también del *Cod_Cliente*.

$\text{Cod_Cliente} \rightarrow \text{Nom_Cliente}$ (MAL, ESTO ES TRANSITIVIDAD)

Facturación (*Nro_Fra*, *Fecha*, *Cod_Cliente*, ~~*Nom_Cliente*~~, *Total_Fra*)

Cliente (*Cod_Cliente*, *Nom_Cliente*)

Producto (*Cod_Producto*, *Descripción*, *Cantidad*, *Precio_Unit*)

Al final, a modo resumen, llegamos a concluir el 2FN así: MAL

Facturación (*Nro_Fra*, *Fecha*, *Cod_Cliente*, *Total_Fra*)

Cliente (*Cod_Cliente*, *Nom_Cliente*)

Producto (*Cod_Producto*, *Descripción*, *Cantidad*, *Precio_Unit*)

Por lo tanto, la relación ahora sí encuentra normalizada en 2FN.

A continuación, comprobamos si la base de datos está en 3FN (para ello todas sus relaciones tienen que estarlo)

3FN: Una base de datos se encuentra en 3FN si, solo si, está en 2FN y además no existen dependencias transitivas, es decir, todas las dependencias deben ser respecto de la clave principal.

Analizando las dependencias funcionales anteriores detectamos transitividad en las siguientes dependencias:

$\text{Nro_Fra} \rightarrow \text{Cod_Cliente}$

$\text{Cod_Cliente} \rightarrow \text{Nom_Cliente}$