

PEDIDOSCAJAS (IdCliente, NombreCli, ApellidosCli, CalleCli, CiudadCli, ProvinciaCli, CPostalCli, TeléfonoCli, IdPedido, FechaPedido, IdCaja, CantidadCajasPedidas)

CAJASBOMBONES (IdCaja, NombreCaja, DescripciónCaja, PrecioCaja, CantidadBombónenCaja, IdBombón, NombreBombón, DescripciónBombón, TipoChocolate, CosteBombón, Envoltorio)

- El nombre de ciudad es único.
- Cada pedido se identifica unívocamente por el IdPedido. Un pedido lo realiza un cliente.
- En el pedido se detallan las cajas y la cantidad de cajas pedidas de cada tipo.
- En cada caja puede haber bombones de varios tipos.
- No hay dos cajas de bombones con el mismo nombre y,
- No hay dos bombones con el mismo nombre.

Problemas

Inserción: no puedes crear bombones nuevos sin tener que añadirlos a una caja o crear una caja sin un pedido .

Actualización: cuando repitamos caja bombones o estés mismos tendremos que modificarlos en cada t  pla .

Borrado: cuando se de el caso de solo tener una caja de bombones y borremos el pedido esta misma se perder   de los datos .

Dependencias funcionales:

IdBomb  n → NombreBomb  n, Descripci  nBomb  n, TipoChocolate, CosteBomb  n, Envoltorio)

IdCliente → nombreCli, ApellidosCli , CalleCli , CiudadCli , ProvinciaCli, CpostalCli ,
Tel  fonoCli.

IdPedido → FechaPedido , CantidadCajasPedidas

IdPedido + IdCaja → CantidadCajasPedidas

IdCaja → nombreCaja , Descripci  nCaja , precioCaja , cantidadBombonEnCaja

IdCaja + IdBomb  n → cantidadBomb  nEnCaja

nombreCiudad ↔

nombreCaja ↔ IdCaja

nombreBomb  n ↔ IdBomb  n

Claves candidatas - CP

PEDIDOSCAJAS: IdPedido + IdCaja

CAJASBOMBONES: IdCaja + IdBombom

Claves candidatas – CA

PEDIDOSCAJAS: IdPedido + nombreCaja

CAJASBOMBONES: nombreCaja + nombreBombom

Atributos primos y no primos

PEDIDOSCAJAS (IdCliente, NombreCli, ApellidosCli, CalleCli, CiudadCli, ProvinciaCli, CPostalCli, TeléfonoCli, IdPedido, FechaPedido, IdCaja, CantidadCajasPedidas)

CAJASBOMBONES (IdCaja, NombreCaja, DescripciónCaja, PrecioCaja, CantidadBombónenCaja, IdBombón, NombreBombón, DescripciónBombón, TipoChocolate, CosteBombón, Envoltorio)

Regla 1FN

Todos los atributos son átomicos.

Regla 2FN

Razonamiento 1

"**PEDIDOSCAJAS**" no cumple con esta regla debido a que sus atributos no primos deben depender completamente de la clave primaria . En este caso los datos del cliente y fecha pedido no depende de 'IdCaja'.

Desglose:

PEDIDO(IdCliente, NombreCli, ApellidosCli, CalleCli, CiudadCli, ProvinciaCli, CPostalCli, TeléfonoCli, IdPedido, FechaPedido)

PEDIDOCAJAS(IdPedido, IdCaja, CantidadCajasPedidas)

Razonamiento 2:

"CAJABOMBONES" no cumple con esta regla por que todo atributo no primo debe depender completamente de su clave primaria , en este caso los datos de bombom no dependen completamente de 'IdCaja'

Desglose:

Caja(IdCaja, NombreCaja, DescripciónCaja, PrecioCaja)

Caja-Bombom(IdCaja ,IdBombom , cantidadBombomCaja)

Bombom(IdBombón, NombreBombón, DescripciónBombón, TipoChocolate, CosteBombón, Envoltorio)

TOTAL:

PEDIDO(IdCliente, NombreCli, ApellidosCli, CalleCli, CiudadCli, ProvinciaCli, CPostalCli, TeléfonoCli, IdPedido, FechaPedido)

PEDIDOCAJAS(IdPedido, IdCaja, CantidadCajasPedidas)

Caja(IdCaja, NombreCaja, DescripciónCaja, PrecioCaja)

Caja-Bombom(IdCaja ,IdBombom , cantidadBombomCaja)

Bombom(IdBombón, NombreBombón, DescripciónBombón, TipoChocolate, CosteBombón, Envoltorio)

Regla 3 FN

Para cumplir esta regla no debe de haber dependencias transitivas en cada desglose .

BOMBOM : sí cumple esta regla.

CAJA-BOMBOM : sí cumple esta regla

Caja: sí cumple esta regla

PEDIDOCAJAS: sí cumple esta regla

PEDIDO : no cumple esta regla debido a que nombreCliente depende de IdCliente que a su vez depende de IdPedido

Resultado :

CLIENTE(IdCliente, NombreCli, ApellidosCli, CalleCli, CiudadCli, ProvinciaCli, CPostalCli, TeléfonoCli,)

PEDIDO(IdPedido, IdCliente, FechaPedido)

PEDIDOCAJAS(IdPedido, IdCaja, CantidadCajasPedidas)

Caja(IdCaja, NombreCaja, DescripciónCaja, PrecioCaja)

Caja-Bombom(IdCaja ,IdBombom , cantidadBombomCaja)

Bombom(IdBombón, NombreBombón, DescripciónBombón, TipoChocolate, CosteBombón, Envoltorio)

Regla de FNBC:

“La forma normal de Boyce-Codd requiere que no existan dependencias funcionales no triviales de los atributos que no sean un conjunto de la clave candidata”

Se cumple esta regla cuando hay claves alternativas

TABLA BOMBOM

IdBombom ↔ nombreBombom

TABLA CAJA

IdCaja ↔ nombreCaja

TABLA CLIENTE

nombreCiudad ↔ 0

Resultado :

BOMBOM2(IdBombom, nombreBombom)

CAJA2(IdCaja, nombreCaja)