

Proyecto de diseño de una tienda online

Enunciado

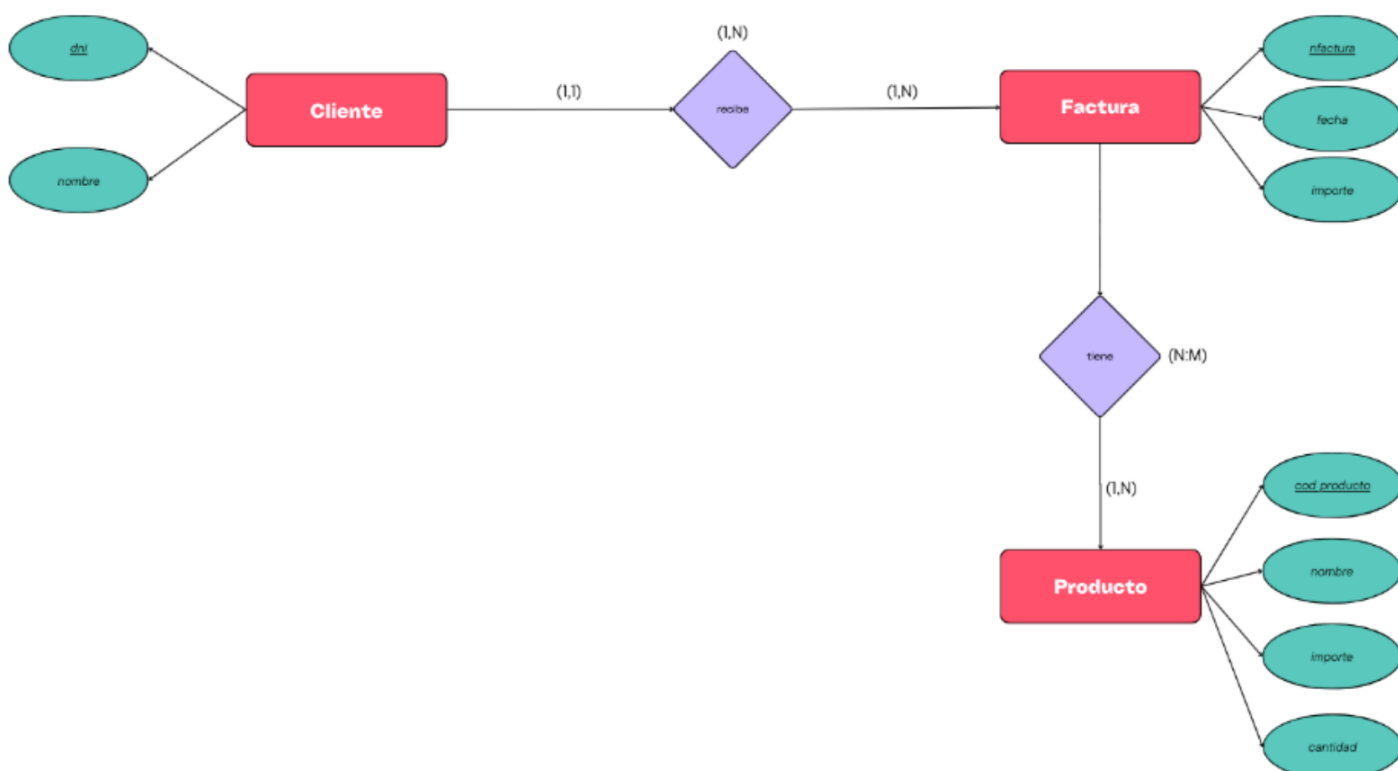
Tenemos una tienda donde se venden una serie de productos y necesitamos almacenar datos de las facturas que se emiten. El funcionamiento principal del modelo es el siguiente.

- En cada venta se pueden vender varios productos y se emite una factura.
- De las facturas necesitamos el DNI y nombre del cliente, la fecha, el importe y el número de factura que identificará a la factura de forma única.
- Relacionado con la factura que emitimos en nuestra tienda es necesario almacenar los productos que se han comprado y se han incluido en esa factura, o sea, las líneas de factura.
- Necesitamos saber el código de cada producto, el nombre del producto, el importe y la cantidad de artículos de ese producto.

Se pide:

- Dibujar el modelo E/ R resultante.
- Dibujar el diccionario de tres tablas reflejando: el tipo de datos, si es simple o compuesto, si es multivaluado, si es clave y qué tipo de clave es, si es índice o no.
- Modelo E/R transformado, normalizado las relaciones la cardinalidad de las mismas.
- Modelo E/R normalizado hasta la 3 FN explicando paso a paso el resultado de la normalización y el porqué de tu decisión.

DIAGRAMA ENTIDAD-RELACIÓN



DICCIONARIO DE DATOS

Cliente	Tipo	Clave	NULL	Restricción	Índice	Multievaluado	Compuesto
dni	Numérico	PK	NO	NO	SI	NO	NO
nombre	Cadena	NO	NO	NO	SI	NO	NO

Consideramos índices ambos atributos como índices puesto que nos interesa filtrarlos.

Factura	Tipo	Clave	NULL	Restricción	Índice	Multievaluado	Compuesto
nfactura	Cadena	PK	NO	NO	SI	NO	NO
fecha	Date time	NO	NO	NO	SI	NO	SI
importe	Real	NO	NO	NO	NO	NO	NO

Consideramos índices los atributos nfactura y fecha como índices puesto que nos interesa filtrarlos.

Además, consideramos compuesto el atributo fecha ya que ésta puede venir tanto con números como con cadena de texto (para el mes).

Producto	Tipo	Clave	NULL	Restricción	Índice	Multievaluado	Compuesto
cod_producto	Cadena	PK	NO	NO	SI	NO	SI
nombre	Cadena	NO	NO	NO	SI	NO	NO
importe	Real	NO	NO	NO	NO	NO	NO
cantidad	Entero	NO	NO	NO	NO	NO	NO

Consideramos índices los atributos cod_producto y nombre como índices puesto que nos interesa filtrarlos.

Además, consideramos compuesto el atributo cod_producto ya que ésta puede venir tanto con números como con cadena de texto.

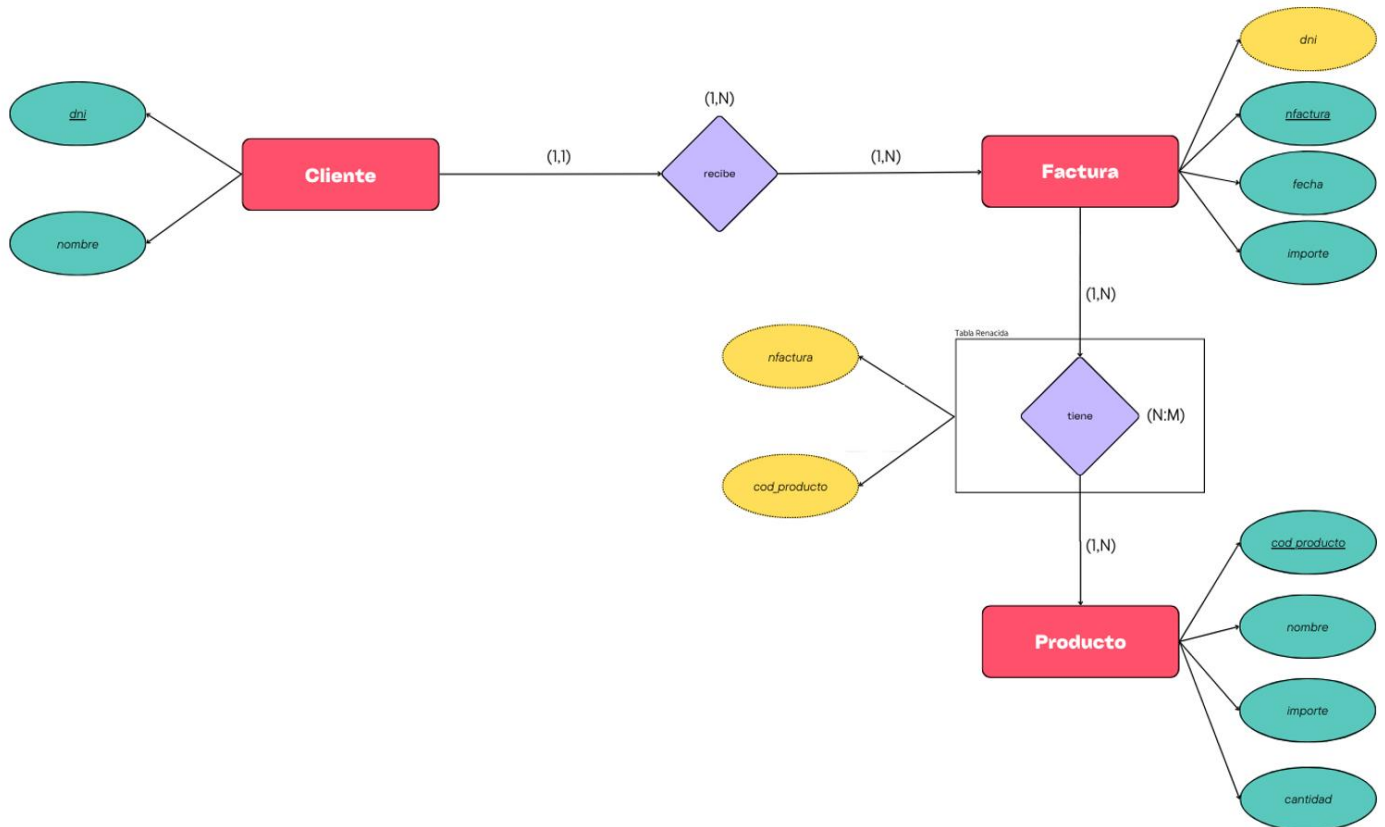
ER SIN TRANSFORMAR → 3 tablas

Cliente (dni, nombre)

Factura (nfactura, fecha, importe)

Producto (cod_producto, nombre, importe, cantidad)

ER TRANSFORMADO → 4 tablas



Cliente (dni, nombre)

Factura (nfactura, dni, fecha, importe)

Factura_tiene_producto (nfactura, cod_producto) (TABLA RENACIDA)

Producto (cod_producto, nombre, importe, cantidad)

En este modelo transformado, en la tabla Factura aparece el dni como FK, heredado de la relación “recibe” con la tabla Cliente que al ser (1:N), la clave PK de Cliente pasa a ser clave FK de Factura.

Nos aparece una tabla renacida de la relación “tiene” con las tablas Factura y Producto. En esta nueva tabla las PK de ambas tablas aparecen como FK en la nueva tabla Factura_tiene_producto.

PRIMERA FORMA NORMAL (1FN) → 4 tablas

Cliente (dni, nombre) (Tabla 1)

Factura (nfactura, dni, fecha, importe) (Tabla 2)

Factura_tiene_producto (nfactura, cod_producto) (RENACIDA) (Tabla 3)

Producto (cod_producto, nombre, importe, cantidad) (Tabla 4)

Se mantiene igual puesto que no tenemos datos multievaluados.

Todas las tablas cumplen la 1FN y además al ser una BBDD nuevo no permite datos repetidos con la Integridad Referencial de los índices; ni datos separados con comas en los atributos con la Integridad de Dominio.

SEGUNDA FORMA NORMAL (2FN) → 6 tablas

- Cliente (dni, nombre) (Tabla 1) →

Cumple la 2FN ya que hay una asociación burocrática entre el DNI y el nombre (durante la vida del cliente).

- Factura (nfactura, dni, fecha, importe) →

No cumple la 2FN. Se transforma en dos tablas.
Factura y Factura_detalle

- Factura (nfactura, dni, importe) (Tabla 2)
- Factura_detalle (nfactura, fecha) (Tabla 3)

- Factura_tiene_producto (nfactura, cod_producto) (RENACIDA) (Tabla 4) →

Cumple la 2FN ya que no hay PK

- Producto (cod_producto, nombre, importe, cantidad) →

No cumple la 2FN- Se transforma en dos tablas.
Producto y Producto_detalle

- Producto (cod_producto, nombre) (Tabla 5)
- Producto_detalle (cod_producto, importe, cantidad) (Tabla 6)

TERCERA FORMA NORMAL (3FN)

- Cliente (dni, nombre) (Tabla 1) →

Cumple la 3FN. Solo hay una no clave con lo que no puede depender de ninguna otra no clave.

- Factura (nfactura, dni, importe) (Tabla 2) →

Cumple la 3FN. Solo hay una no clave con lo que no puede depender de ninguna otra no clave.

- Factura_detalle (nfactura, fecha) (Tabla 3) →

Cumple la 3FN. Solo hay una no clave con lo que no puede depender de ninguna otra no clave.

- Factura_tiene_producto (nfactura, cod_producto) (RENACIDA) (Tabla 4) →

Cumple la 3FN. No hay una no claves.

- Producto (cod_producto, nombre) (Tabla 5) →

Cumple la 3FN. Solo hay una no clave con lo que no puede depender de ninguna otra no clave.

- Producto (cod_producto, importe, cantidad) (Tabla 6)→

Cumple la 3FN. Importe no depende de la cantidad. El precio del producto seguirá siendo mismo independientemente de la cantidad que se compre. (Salvo que hubiese alguna restricción definida en la que nos indicasen que a mayor cantidad de producto disminuiría el precio unitario, por ejemplo un descuento por comprar X cantidad).