

1. Introducción

Este proyecto trata de la creación de una base de datos para una **Tienda de Venta de Productos** (simulada para efectos del proyecto). Para cumplir este objetivo se trabajará en la descripción de tablas, en la creación del diagrama E-R (entidad relación) y en el desarrollo de todos los scripts y queries necesarias para la creación de la base de datos y sus utilidades. La base de datos será creada utilizando el motor MySQL y su visualizador Workbench.

Toda la documentación del proyecto, paso a paso, imágenes de pantalla, scripts, desafíos y entregas intermedias de este Proyecto final están ordenadas en 14 carpetas ubicadas en el repositorio github:

https://github.com/cristiansu/MySQL_Coder.git

2. Objetivo

En el contexto de una Tienda de Venta de Productos que necesita manejar su información de forma ordenada, con trazabilidad e integridad de la información, el objetivo es crear una solución de base datos normalizada, procurando realizar las relaciones entre tablas necesarias para un correcto flujo de datos, la realización de consultas y otras utilidades.

3. Situación Problemática

Se requiere crear un manejo de datos para la Tienda de Venta de Productos que permita de forma ordenada almacenar la información de los clientes, sus compras, sus facturas, los productos y sus categorías, los vendedores y las consultas hacia la base datos que sean necesarias para la operación del negocio.

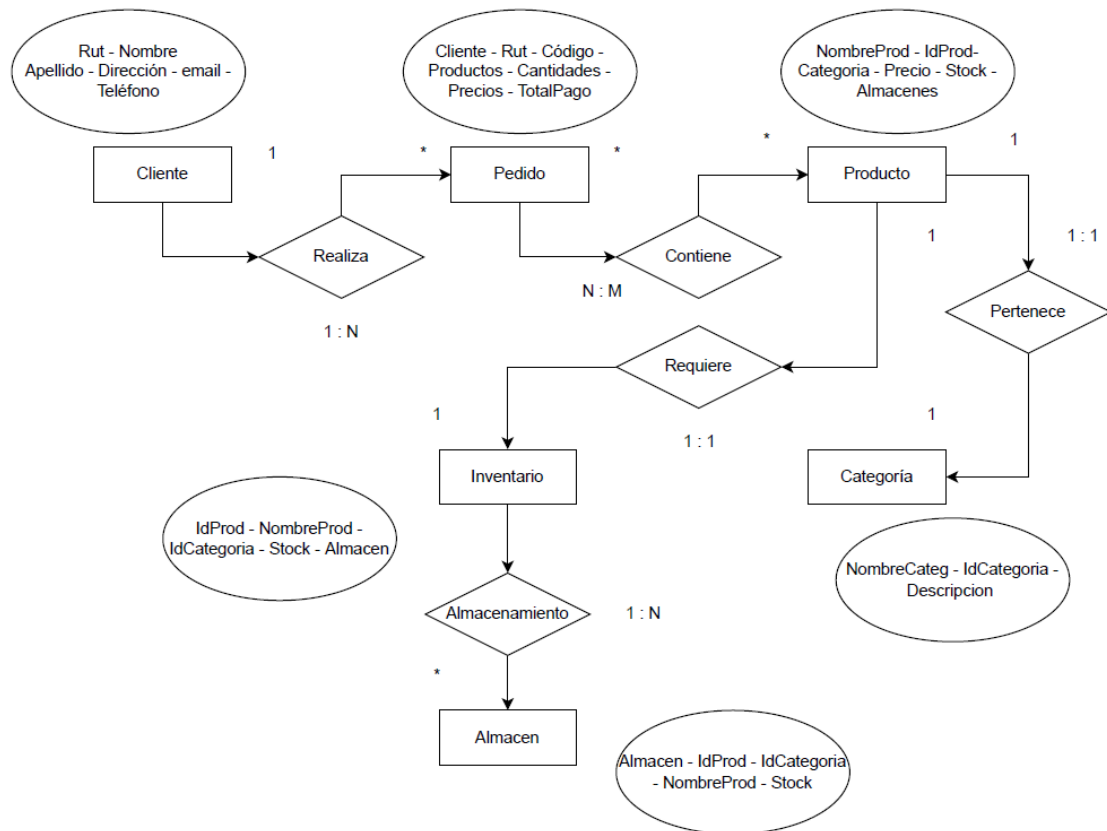
4. Modelo de Negocio

El modelo de negocio considera la venta de productos de diversas categorías, de una manera similar a un mi retail de venta de artículos por departamento, es decir: vestuario, electrónica, deportes, entre otros.

5. Diagrama E-R

ERD- Tienda Venta Productos

Tablas: Cliente + Pedido + Producto + Categoría + Inventario + Almacén



6. Listado de Tablas y Descripción

Tabla : Facturas				
Nombre Campo	Clave	Tipo Dato	Características	Comentario
IdFactura	PK	INT	NOT NULL AUTO_INCREMENT	Contiene datos de identificación de facturas
fecha		DATE		
idCliente	FK	INT	NOT NULL	
idVendedor	FK	INT	NOT NULL	

Tabla : Detalle Factura				
Nombre Campo	Clave	Tipo Dato	Características	Comentario
idDetalleFactura	PK	INT	NOT NULL AUTO_INCREMENT	Contiene detalle de facturas: cantidad, precio, total a pagar, más los datos de identificación de cada factura
idFactura	FK	INT	NOT NULL	
idProducto	FK	INT	NOT NULL	
cantidad		INT		
precio		INT		
totalApagar		INT		

Tabla : Clientes				
Nombre Campo	Clave	Tipo Dato	Características	Comentario
IdCliente	PK	INT	NOT NULL AUTO_INCREMENT	Tabla dimensional con datos de identificación de clientes
nombre		VARCHAR		
apellido		VARCHAR		
rut		VARCHAR		

Tabla : Vendedor				
Nombre Campo	Clave	Tipo Dato	Características	Comentario
IdVendedor	PK	INT	NOT NULL AUTO_INCREMENT	Tabla dimensional con datos de identificación de vendedores
nombre		VARCHAR		
apellido		VARCHAR		
rut		VARCHAR		

Tabla : Productos				
Nombre Campo	Clave	Tipo Dato	Características	Comentario
IdProducto	PK	INT	NOT NULL AUTO_INCREMENT	Contiene datos de productos, su nombre y descripción, más su conexión a la categoría de producto a la cual pertenecen.
idCategoria	FK	INT	NOT NULL	
nombreProducto		VARCHAR		
Descripcion		VARCHAR		

Tabla : Categoria				
Nombre Campo	Clave	Tipo Dato	Características	Comentario
idCategoria	PK	INT	NOT NULL AUTO_INCREMENT	Tabla dimensional con detalle de categorías de productos
nombreCategoria		VARCHAR		
Descripcion		VARCHAR		

7. Scripts de Creación Cada Objeto de la Base de Datos

```
create database db_tienda;

use db_tienda;

create table clientes (
  idCliente int not null auto_increment primary key,
  nombre varchar(45),
  apellido varchar(45),
  rut varchar(20)
);

create table vendedor (
  idVendedor int not null auto_increment primary key,
  nombre varchar(45),
  apellido varchar(45),
  rut varchar(20)
);

create table facturas (
  idFactura int not null auto_increment primary key,
  fecha date,
  id_Cliente int not null,
  id_Vendedor int not null,
  foreign key (id_Cliente) references clientes(idCliente),
  foreign key (id_Vendedor) references vendedor(idVendedor)
);

create table categoria (
  idCategoria int not null auto_increment primary key,
  nombreCategoria varchar(45),
  descripcion varchar(100)
);

create table productos (
  idProducto int not null auto_increment primary key,
  id_Categoria int not null,
  nombreProducto varchar(45),
  descripcion varchar (100),
  foreign key (id_Categoria) references categoria(idCategoria)
);

use db_tienda;
create table detalle_factura (
  idDetalleFactura int not null auto_increment primary key,
  id_Factura int not null,
  Id_Producto int not null,
  cantidad int,
  precio int,
  totalApagar int,
  foreign key (id_Factura) references facturas(idFactura),
  foreign key (id_Producto) references productos(idProducto)
);
```

8. Scripts de Inserción de Datos

```
use db_tienda;
```

```
insert into clientes(idCliente,nombre,apellido,rut)
values(null,'Cristian','Saavedra','123456789-1'),
(null,'Carlos','Urbina','723456789-1'),
(null,'Paola','Urbina','723476789-1'),
(null,'Hedy','Lamert','793456789-1'),
(null,'Arnold','Shwgr','323456789-1')
;
```

```
insert into vendedor(idVendedor,nombre,apellido,rut)
values(null,'Guiselle','Pozo','103456789-1'),
(null,'Jeanette','Cheukeman','721456789-1'),
(null,'Paola','Saavedra','793450789-1')
;
```

```
insert into facturas(idFactura,fecha,id_Cliente,id_Vendedor)
values(null,'2022-10-13',1,1),
(null,'2022-10-12',1,1),
(null,'2022-10-10',3,2),
(null,'2022-10-09',2,2),
(null,'2022-11-13',5,3),
(null,'2022-11-13',4,2),
(null,'2022-10-13',1,2)
;
```

```
insert into categoria(idCategoria,nombreCategoria,descripcion)
values(null,'Vestuario','Vestuario general M-F'),
```

```
(null,'Electronica','Articulos electronicos'),
(null,'Calzado','Calzado general M-F')
;
```

```
insert into productos(idProducto,id_Categoria,nombreProducto,descripcion)
values(null,1,'Polera H','Polera H tallas s-xl'),
(null,1,'Calza M','Calza M tallas s-xl'),
(null,2,'Audifono','Audifono bluetooth'),
(null,2,'Notebook Hp','Notebook 14p-ssd512gb-corei7'),
(null,3,'Zapatilla H','Zapatilla running numeros 39-47'),
(null,3,'Zapatilla M','Zapatilla M numeros 37-43');
```

```
insert
detalle_factura(idDetalleFactura,id_Factura,Id_Producto,cantidad,precio,totalApagar)
values(null,1,1,1,100,100),
(null,2,3,1,40,40),
(null,3,2,2,30,60),
(null,4,5,1,130,130),
(null,5,2,2,40,80),
(null,6,1,1,40,40),
(null,7,5,1,130,130);
```

```
select * from detalle_factura;
```

9. Informes Generados en Base a Tablas

```
use db_tienda;
```

```
/* vista 1 retorna el total a pagar por Id de factura, incluye id factura y total a pagar*/
```

```
create view vista_ingresos
```

```
as select id_Factura,totalApagar
```

```
from detalle_factura;
```

```
select * from vista_ingresos;
```

```
/* vista 2 retorna los productos que pertenecen a la categoria vestuario, incluye nombre  
producto y descripción */
```

```
create view vista_productos_vestuario
```

```
as select nombreProducto,descripcion
```

```
from productos
```

```
where id_Categoria=1;
```

```
select * from vista_productos_vestuario;
```

```
/* vista 3 retorna las facturas asociadas a la vendedora Paola, incluye id fatura e id cliente*/
```

```
create view vista_facturas_vendedor_paola
```

```
as select idFactura,id_Cliente
```

```
from facturas
```

```
where id_Vendedor=3;
```

```
select * from vista_facturas_vendedor_paola;
```

```
/* vista 4 retorna las ventas de cada producto, incluye nombre producto, detalle factura y total  
a pagar */
```

```
create view vista_ventasPorProducto
```

```
as select productos.nombreProducto,detalle_factura.id_Producto, detalle_factura.totalApagar
```

```
from productos, detalle_factura
```

```
where productos.idProducto = detalle_factura.id_Producto;
```

```
select * from vista_ventasPorProducto;
```

/* vista 5 retorna las compras realizadas por cada cliente, incluye id factura, nombre producto, nombre cliente y total a pagar*/

create view vista_compras_por_cliente

as select facturas.idFactura, productos.nombreProducto, clientes.nombre, detalle_factura.totalApagar

from clientes, facturas, productos, detalle_factura

where productos.idProducto = detalle_factura.id_Producto and
clientes.idCliente=facturas.id_cliente;

select * from vista_compras_por_cliente;

10. Herramientas y Tecnologías Utilizadas

La herramienta utilizada como motor de base de datos fue MySQL, el visor fue MySQL WorkBench.