

Realizado por:

- Jesús David Bonelo Cuellar
- Devon José Alvarez Osorio

Ejercicio asignado: (Ejercicio C)

# Tienda Virtual Supermercado Don Pepe

## Tabla de contenido:

Tabla de contenido:	1
Entidades identificadas y sus atributos:	1
Reglas de negocio supuestas:	2
<b>Modelo Entidad-Relación</b>	<b>3</b>
<b>Modelo Relacional:</b>	<b>3</b>
<b>Normalización</b>	<b>4</b>
<b>Definición de la base de datos en SQL</b>	<b>5</b>
<b>Insertión de registros en las tablas</b>	<b>7</b>
<b>Creación de vistas</b>	<b>15</b>
<b>Consultas a la base de datos</b>	<b>18</b>
<b>Definición de procedimientos almacenados</b>	<b>21</b>
<b>Definición de triggers</b>	<b>25</b>
<b>Conclusión</b>	<b>27</b>

## Entidades identificadas y sus atributos:

- Cliente
  - ID
  - Cédula
  - Nombres
  - Apellidos
  - Dirección
  - Código postal
  - Telefono
  - Email
  - Contraseña
- Producto
  - ID (Llave)
  - Nombre
  - Marca

- Origen
- Volumen
- Peso
- Fotografía
- Precio
- Stock
- Categoría
  - Nombre (Llave)
  - Condiciones de almacenamiento
  - Observaciones
- Pedido
  - ID
  - Estado
  - Dirección
  - Número de tarjeta
  - Vencimiento de la tarjeta
  - Fecha solicitado
  - Fecha servido
  - Fecha entregado
- Repartidor
  - ID (Llave)
  - Nombre
  - Matricula furgoneta
  - Código postal (Multivaluado)
- Almacenista
  - ID (Llave)
  - Nombre
  - Area (Multivaluado)

## Reglas de negocio supuestas:

Los productos pueden ser de origen nacional o internacional.

El cliente posee una sola ubicación y al momento de realizar el pedido puede utilizar su dirección o ingresar una diferente.

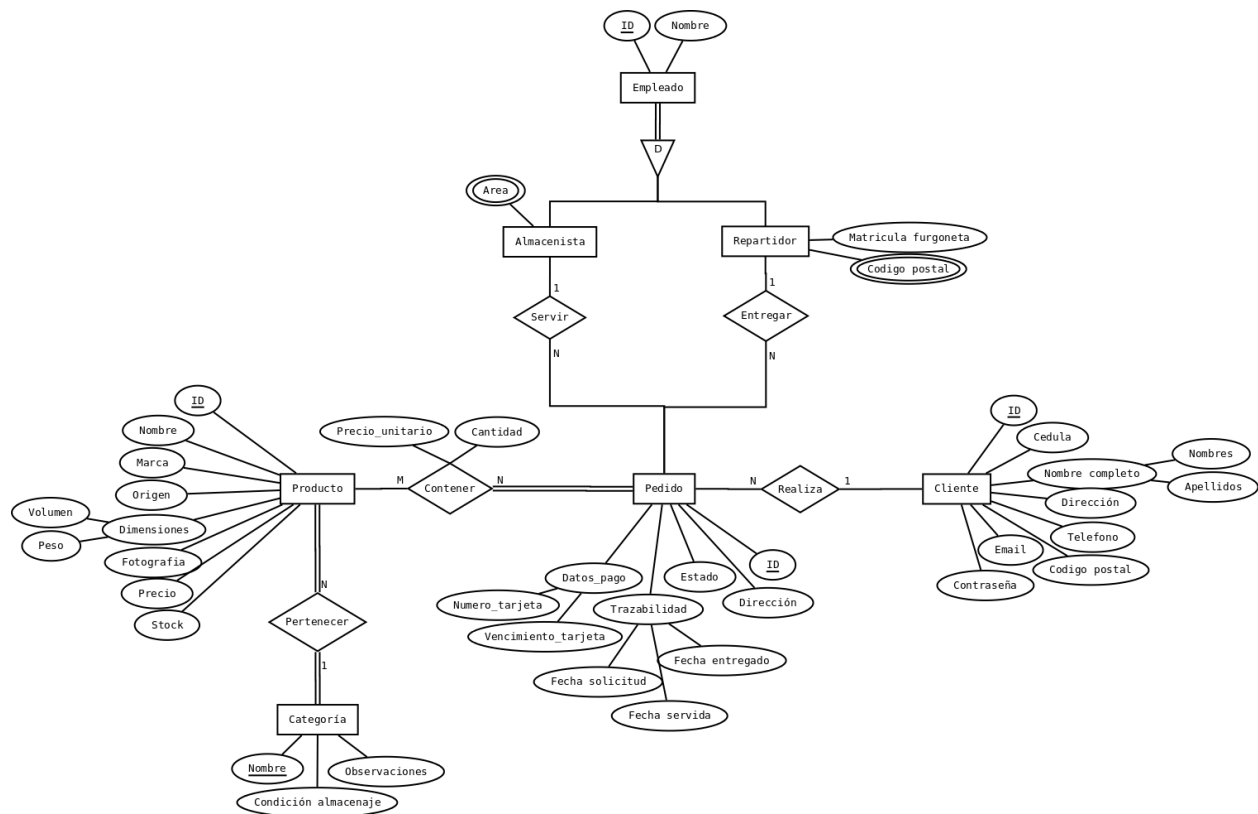
La zona de repartición está compuesta por uno o varios códigos postales, por lo tanto la representamos como un atributo multivaluado de la entidad `Repartidor`.

El orden de los estados de un pedido son **En preparación** -> **Pendiente entrega** -> **Entregado**.

Los clientes solo pueden pagar con tarjeta.

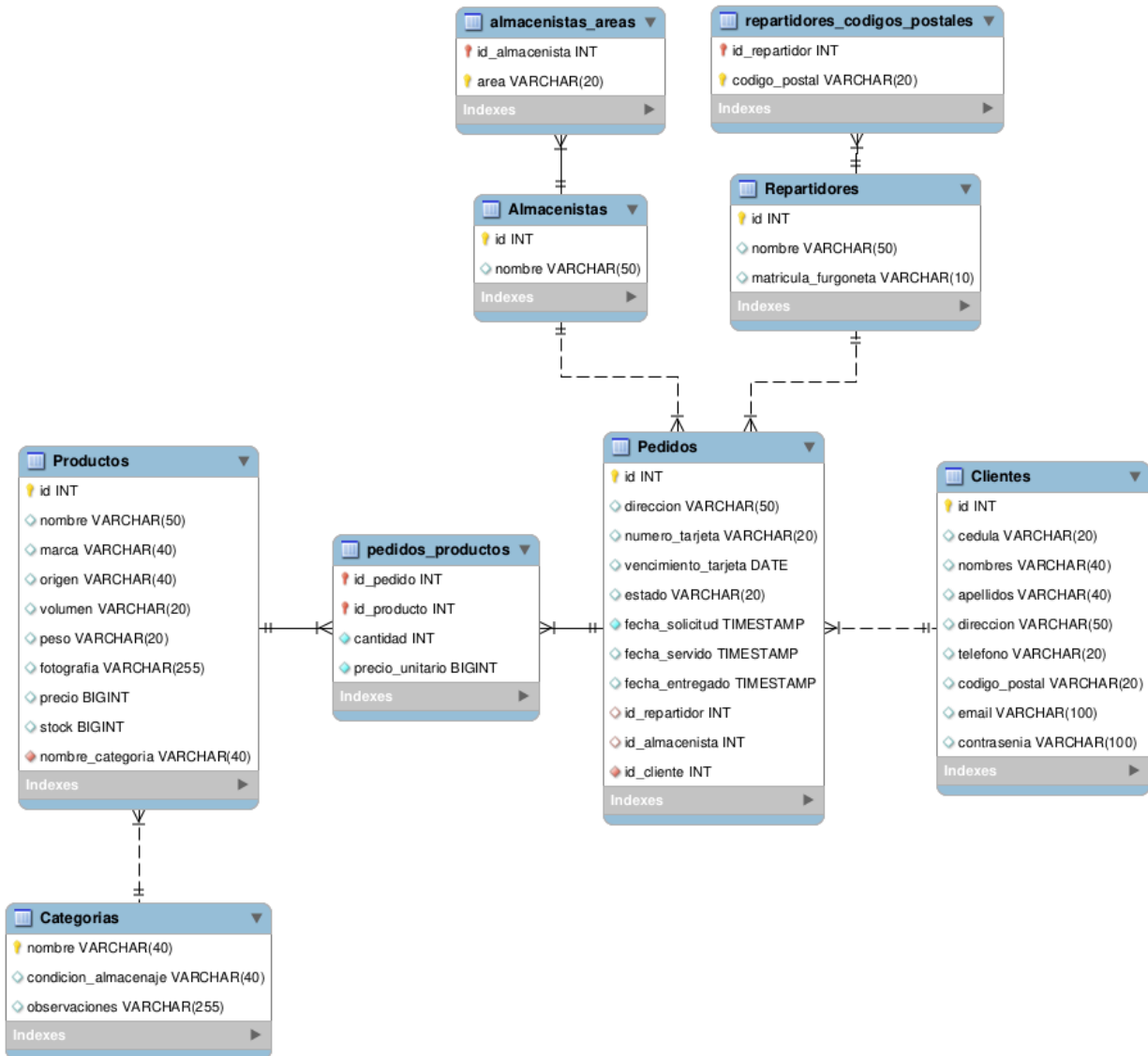
Los pedidos guardan la cantidad y el precio unitario de los productos al momento de la compra, el importe total se calcula con base en esta información.

## Modelo Entidad-Relación



## Modelo Relacional:

- Se crearon las tablas de las entidades independientes:
  - Clientes
  - Categorías
- Como la disyunción de empleado es total, se crearon las tablas **Repartidores** y **Almacenistas** con todos los atributos heredados.
- Se crearon las tablas para los atributos multivaluados de código postal de los repartidores y áreas de los almacenistas.
- Se crearon las tablas con llaves foraneas **Productos** y **Pedidos**
- Se creó la tabla intermedia de la relación muchos a muchos entre pedidos y productos.



## Normalización

Al analizar el modelo relacional observamos que ya se cumplen con las condiciones de normalización

### 1NF:

- ☒ ~~Todos los atributos tienen valores atómicos.~~
- ☒ ~~No hay atributos multivaluados.~~
- ☒ ~~No deben existir registros duplicados.~~

- ☒ ~~Las columnas repetidas deben eliminarse y colocarse agrupadas en tablas separadas bajo un contexto.~~
- ☒ ~~Definir clave principal.~~

## 2NF:

- ☒ ~~Estar en 1FN~~
- ☒ ~~Todos los valores de las columnas deben depender únicamente de la llave primaria de la tabla.~~
- ☒ ~~Las tablas deben tener una única llave primaria que identifique a la tabla y que sus atributos dependen de ella.~~

## 3NF:

- ☒ ~~Estar en 2FN.~~
- ☒ ~~Cada atributo que no está incluido en la clave primaria no depende transitivamente de la clave primaria~~

# Definición de la base de datos en SQL

```
CREATE DATABASE IF NOT EXISTS supermercado;
```

```
USE supermercado;
```

```
-- Crear las tablas independientes
```

```
CREATE TABLE IF NOT EXISTS Clientes (  
    id INT NOT NULL PRIMARY KEY,  
    cedula VARCHAR(20),  
    nombres VARCHAR(40),  
    apellidos VARCHAR(40),  
    direccion VARCHAR(50),  
    telefono VARCHAR(20),  
    codigo_postal VARCHAR(20),  
    email VARCHAR(100),  
    contrasenia VARCHAR(100)  
) ENGINE=INNODB;
```

```
CREATE TABLE IF NOT EXISTS Categorías (  
    id INT NOT NULL PRIMARY KEY,  
    nombre VARCHAR(40),  
    descripcion VARCHAR(100),  
    precio DECIMAL(10,2),  
    imagen VARCHAR(100)
```

```

    nombre VARCHAR(40) PRIMARY KEY,
    condicion_almacenaje VARCHAR(40),
    observaciones VARCHAR(255)
) ENGINE=INNODB;

CREATE TABLE IF NOT EXISTS Almacenistas (
    id INT NOT NULL PRIMARY KEY,
    nombre VARCHAR(50),
    area VARCHAR(20)
) ENGINE=INNODB;

CREATE TABLE IF NOT EXISTS Repartidores (
    id INT NOT NULL PRIMARY KEY,
    nombre VARCHAR(50),
    matricula_furgoneta VARCHAR(10)
) ENGINE=INNODB;

-- Crear tabla de atributo multivaluado
CREATE TABLE IF NOT EXISTS repartidores_codigos_postales (
    id_repartidor INT NOT NULL,
    codigo_postal VARCHAR(20),
    FOREIGN KEY (id_repartidor)
        REFERENCES Repartidores (id),
    PRIMARY KEY (id_repartidor , codigo_postal)
) ENGINE=INNODB;

-- Crear tabla con llaves foraneas
CREATE TABLE IF NOT EXISTS Productos (
    id INT NOT NULL PRIMARY KEY,
    nombre VARCHAR(50),
    marca VARCHAR(40),
    origen VARCHAR(40),
    volumen VARCHAR(20),
    peso VARCHAR(20),
    fotografia VARCHAR(255),
    precio BIGINT,
    stock BIGINT,
    id_categoria VARCHAR(40) NOT NULL,
    FOREIGN KEY (id_categoria)
        REFERENCES Categorias (nombre)
) ENGINE=INNODB;

```

```

CREATE TABLE IF NOT EXISTS Pedidos (
    id INT NOT NULL PRIMARY KEY,
    estado VARCHAR(20),
    fecha_solicitud TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,
    fecha_servido TIMESTAMP,
    fecha_entregado TIMESTAMP,
    id_repartidor INT,
    id_almacenista INT,
    id_cliente INT NOT NULL,
    FOREIGN KEY (id_repartidor)
        REFERENCES Repartidores (id),
    FOREIGN KEY (id_almacenista)
        REFERENCES Almacenistas (id),
    FOREIGN KEY (id_cliente)
        REFERENCES Clientes (id)
) ENGINE=INNODB;

-- Crear tablas intermedias de relaciones muchos a muchos
CREATE TABLE IF NOT EXISTS pedidos_productos (
    id_pedido INT NOT NULL,
    id_producto INT NOT NULL,
    cantidad INT NOT NULL,
    FOREIGN KEY (id_pedido)
        REFERENCES Pedidos (id),
    FOREIGN KEY (id_producto)
        REFERENCES Productos (id),
    PRIMARY KEY (id_pedido , id_producto)
) ENGINE=INNODB;

```

## Inserción de registros en las tablas

```

-- Clientes

INSERT INTO Clientes (id, cedula, nombres, apellidos, direccion, telefono,

```

```
codigo_postal, email, contrasenia) VALUES
(1, '1003920497', 'Juan', 'Hernandez', 'Carrera 10 #25-30', '3119049820',
'110111', 'juan.hernandez@gmail.com', 'contraseña1'),
(2, '1067038518', 'María', 'Consa', 'Calle 20 #15-45', '3223432978',
'220222', 'maria.consa@gmail.com', 'contraseña2'),
(3, '1003059648', 'Carlos', 'Vertel', 'Avenida 5 #8-12', '3136983097',
'330333', 'carlos.vertel@gmail.com', 'contraseña3'),
(4, '90549392', 'Dina', 'Martínez', 'Calle 15 #30-18', '3142984509',
'440444', 'dina.martinez@gmail.com', 'contraseña4'),
(5, '56079368', 'Luis', 'Velazques', 'Carrera 25 #12-10', '3164389870',
'550555', 'luis.velazques@gmail.com', 'contraseña5'),
(6, '1003048327', 'Laura', 'Perdomo', 'Calle 30 #5-25', '3172370976',
'660666', 'laura.perdomo@gmail.com', 'contraseña6'),
(7, '96069378', 'Daniel', 'Hernández', 'Avenida 10 #40-30', '3175489038',
'770777', 'daniel.hernandez@gmail.com', 'contraseña7'),
(8, '89308194', 'Natalia', 'Buitrago', 'Carrera 20 #18-22', '3194098327',
'880888', 'Natalia.buitrago@gmail.com', 'contraseña8'),
(9, '902085890', 'Jesus', 'Laverde', 'Calle 12 #6-14', '3229295955',
'990999', 'jesus.laverde@gmail.com', 'contraseña9'),
(10, '1003049382', 'Marta', 'Torres', 'Avenida 35 #25-40', '3001268947',
'001010', 'marta.torres@gmail.com', 'contraseña10'),
(11, '1067039278', 'Andrés', 'Chaparro', 'Carrera 30 #10-5', '3110794738',
'110121', 'andres.chaparro@gmail.com', 'contraseña11'),
(12, '29378569', 'Milena', 'Uribe', 'Calle 22 #15-28', '3222907895',
'220232', 'milena.uribe@gmail.com', 'contraseña12'),
(13, '34509873', 'maria', 'Portillo', 'Avenida 15 #20-35', '3235249807',
'330343', 'maria.portillo@gmail.com', 'contraseña13'),
(14, '45098072', 'Valentina', 'Arias', 'Calle 8 #12-30', '3240872937',
'440454', 'valentina.arias@gmail.com', 'contraseña14'),
(15, '50987073', 'Miguel angel', 'Paes', 'Carrera 12 #25-15', '3190347856',
'550565', 'miguel.paes@gmail.com', 'contraseña15'),
(16, '1003049780', 'Camilo', 'Sandobal', 'Avenida 25 #30-20', '3164906738',
'660676', 'camila.sandobal@gmail.com', 'contraseña16'),
(17, '982376590', 'Diego', 'Ortiz', 'Calle 18 #8-40', '3170478327',
'770787', 'diego.ortiz@gmail.com', 'contraseña17');
```



	id	cedula	nombres	apellidos	direccion	telefono	codigo_postal	email	contrasenia
▶	1	1003920497	Juan	Hernandez	Carrera 10 #25-30	3119049820	110111	juan.hernandez@gmail.com	contraseña1
	2	1067038518	María	Consa	Calle 20 #15-45	3223432978	220222	maria.consa@gmail.com	contraseña2
	3	1003059648	Carlos	Vertel	Avenida 5 #8-12	3136983097	330333	carlos.vertel@gmail.com	contraseña3
	4	90549392	Dina	Martínez	Calle 15 #30-18	3142984509	440444	dina.martinez@gmail.com	contraseña4
	5	56079368	Luis	Velazques	Carrera 25 #12-10	3164389870	550555	luis.velazques@gmail.com	contraseña4
	6	1003048327	Laura	Perdomo	Calle 30 #5-25	3172370976	660666	laura.perdomo@gmail.com	contraseña6
	7	96069378	Daniel	Hernández	Avenida 10 #40-30	3175489038	770777	daniel.hernandez@gmail.com	contraseña7
	8	89308194	Natalia	Buitrago	Carrera 20 #18-22	3194098327	880888	Natalia.buitrago@gmail.com	contraseña8
	9	902085890	Jesus	Laverde	Calle 12 #6-14	3229295955	990999	jesus.laverde@gmail.com	contraseña9
	10	1003049382	Marta	Torres	Avenida 35 #25-40	3001268947	001010	marta.torres@gmail.com	contraseña10
	11	1067039278	Andrés	Chaparro	Carrera 30 #10-5	3110794738	110121	andres.chaparro@gmail.com	contraseña11
	12	29378569	Milena	Uribe	Calle 22 #15-28	3222907895	220232	milena.uribe@gmail.com	contraseña12
	13	34509873	maria	Portillo	Avenida 15 #20-35	3235249807	330343	maria.portillo@gmail.com	contraseña13
	14	45098072	Valentina	Arias	Calle 8 #12-30	3240872937	440454	valentina.arias@gmail.com	contraseña14
	15	50987073	Miguel ...	Paes	Carrera 12 #25-15	3190347856	550565	miguel.paes@gmail.com	contraseña15
	16	1003049780	Camilo	Sandohal	Avenida 25 #30-20	3164906738	660676	camila.sandohal@gmail.com	contraseña16

-- Almacenistas

INSERT INTO Almacenistas (id, nombre) VALUES

```
(1, 'Juan Pérez'),
(2, 'María García'),
(3, 'Luis Martínez'),
(4, 'Ana Rodríguez'),
(5, 'Pedro López'),
(6, 'Laura González'),
(7, 'Carlos Sánchez'),
(8, 'Sofía Hernández'),
(9, 'Pablo Díaz'),
(10, 'Elena Ruiz');
```

Result Grid		Filter Rows:	Edit:	Export/Import:
	id	nombre		
▶	1	Juan Pérez		
	2	María García		
	3	Luis Martínez		
	4	Ana Rodríguez		
	5	Pedro López		
	6	Laura González		
	7	Carlos Sánchez		
	8	Sofía Hernández		
	9	Pablo Díaz		
	10	Elena Ruiz		
	NULL	NULL		

almacenistas 1 x

-- Repartidores

```
INSERT INTO Repartidores (id, nombre, matricula_furgoneta) VALUES
(1, 'Marlon Buendia', '123ABC'),
(2, 'Ismael Arsuaga', '456DEF'),
(3, 'Leonel Perez', '789GHI'),
(4, 'Jesus daniel Lozano', '101JKL'),
(5, 'Ahi Landero', '112MNO'),
(6, 'Rosiris Puente', '131PQR'),
(7, 'Rodrigo Marangoni', '415STU'),
(8, 'Hernan Sifuentes', '161VWX'),
(9, 'Diogenes Vertel', '718YZA'),
(10, 'Pablo Ruiz', '192BCD');
```

	id	nombre	matricula_furgoneta
▶	1	Marlon Buendia	123ABC
	2	Ismael Arsuaga	456DEF
	3	Leonel Perez	789GHI
	4	Jesus daniel Lozano	101JKL
	5	Ahi Landero	112MNO
	6	Rosiris Puente	131PQR
	7	Rodrigo Marangoni	415STU
	8	Hernan Sifuentes	161VWX
	9	Diogenes Vertel	718YZA
	10	Pablo Ruiz	192BCD
▲	NULL	NULL	NULL

repartidores 1 x

-- Categorías

```
INSERT INTO Categorías (nombre, condicion_almacenaje, observaciones) VALUES
('Alimentos', 'Seco', 'Categoría principal para alimentos no perecederos'),
('Lácteos', 'Frio', 'Productos lácteos que necesitan refrigeración'),
('Congelados', 'Congelado', 'Productos congelados que requieren
almacenamiento a temperaturas muy bajas'),
('Frutas y Verduras', 'Frio', 'Frutas y verduras frescas que necesitan
refrigeración'),
('Carne', 'Frio', 'Productos cárnicos que necesitan refrigeración'),
('Bebidas', 'Seco', 'Bebidas no alcohólicas y refrescos'),
('Limpieza', 'Seco', 'Productos de limpieza para el hogar'),
('Higiene', 'Seco', 'Productos de higiene personal'),
```

```
( 'Cuidado del Hogar', 'Seco', 'Productos para el cuidado del hogar'),
( 'Electrodomésticos', 'Seco', 'Electrodomésticos y artículos para el
hogar');
```

nombre	condicion_almacenaje	observaciones
Alimentos	Seco	Categoría principal para alimentos no perecederos
Bebidas	Seco	Bebidas no alcohólicas y refrescos
Carnes	Frio	Productos cárnicos que necesitan refrigeración
Congelados	Congelado	Productos congelados que requieren almace...
Cuidado del Hogar	Seco	Productos para el cuidado del hogar
Electrodomésticos	Seco	Electrodomésticos y artículos para el hogar
Frutas y Verduras	Frio	Frutas y verduras frescas que necesitan refrige...
Higiene	Seco	Productos de higiene personal
Lácteos	Frio	Productos lácteos que necesitan refrigeración
Limpieza	Seco	Productos de limpieza para el hogar
NULL	NULL	NULL

```
-- Almacenista_areas
```

```
INSERT INTO almacenistas_areas (id_almacenista, area) VALUES
(1, 'recepcion y control'),
(1, 'Almacenaje'),
(2, 'Preparacion de pedido'),
(3, 'Carga y descarga'),
(4, 'salida y verificacion de pedido'),
(5, 'Preparacion de pedido'),
(5, 'Carga y descarga'),
(6, 'salida y verificacion de pedido'),
(7, 'Almacenaje'),
(7, 'Preparacion de pedido');
```

id_almacenistaFK	area
1	recepcion y control
1	Almacenaje
2	Preparacion de pedido
3	Carga y descarga
4	salida y verificacion de pedido
5	Preparacion de pedido
5	Carga y descarga
6	salida y verificacion de pedido
7	Almacenaje
7	Preparacion de pedido

```
-- Pedidos
```

```

INSERT INTO Pedidos (id, direccion, numero_tarjeta, vencimiento_tarjeta,
estado, fecha_solicitud, fecha_servido, fecha_entregado, id_repartidor,
id_almacenista, id_cliente) VALUES
(1, 'Carrera 10 #25-30', '1234567812345678', '2025-01-01', 'En proceso',
'2024-04-02 10:00:00', NULL, NULL, 1, 1, 1),
(2, 'Avenida 456 # 15-67', '2345678923456789', '2025-02-01', 'En tránsito',
'2024-04-02 11:30:00', NULL, NULL, 2, 2, 2),
(3, 'Avenida 5 #8-12', '3456789034567890', '2025-03-01', 'Entregado',
'2024-04-02 13:45:00', '2024-04-02 15:30:00', '2024-04-02 16:15:00', 3, 3,
3),
(4, 'Calle 15 #30-18', '4567890145678901', '2025-04-01', 'En proceso',
'2024-04-02 09:15:00', NULL, NULL, 4, 4, 4),
(5, 'calle 16 # 24-19', '5678901256789012', '2025-05-01', 'Entregado',
'2024-04-02 10:45:00', '2024-04-02 12:30:00', '2024-04-02 13:00:00', 5, 5,
5),
(6, 'Calle 456 # 29-56', '6789012367890123', '2025-06-01', 'En proceso',
'2024-04-02 14:20:00', NULL, NULL, 4, 6, 6),
(7, 'Plaza del Sol calle 76', '7890123478901234', '2025-07-01', 'En
tránsito', '2024-04-02 16:00:00', NULL, NULL, 7, 7, 7),
(8, 'Avenida 78 calle 34', '8901234589012345', '2025-08-01', 'Entregado',
'2024-04-02 17:10:00', '2024-04-02 19:00:00', '2024-04-02 19:30:00', 8, 8,
8),
(9, 'Manzana 45 lote 18', '9012345690123456', '2025-09-01', 'En proceso',
'2024-04-02 08:00:00', NULL, NULL, 9, 9, 9),
(10, 'Diagonal 19 transversal 6', '0123456701234567', '2025-10-01',
'Entregado', '2024-04-02 12:20:00', '2024-04-02 13:45:00', '2024-04-02
14:15:00', 10, 10, 10);

```

Result Grid									
Filter Rows:									
Edit:									
Export/Import:									
Wrap Cell Content:									
	id	direccion	numero_tarjeta	vencimiento_tarjeta	estado	fecha_solicitud	fecha_servido	fecha_entregado	id_repartidor
1	1	Carrera 10 #25-30	1234567812345678	2025-01-01	En proceso	2024-04-02 10:00:00			1
2	2	Avenida 456 # 15-67	2345678923456789	2025-02-01	En tránsito	2024-04-02 11:30:00			2
3	3	Avenida 5 #8-12	3456789034567890	2025-03-01	Entregado	2024-04-02 13:45:00	2024-04-02 15:30:00	2024-04-02 16:15:00	3
4	4	Calle 15 #30-18	4567890145678901	2025-04-01	En proceso	2024-04-02 09:15:00			4
5	5	calle 16 # 24-19	5678901256789012	2025-05-01	Entregado	2024-04-02 10:45:00	2024-04-02 12:30:00	2024-04-02 13:00:00	5
6	6	Calle 456 # 29-56	6789012367890123	2025-06-01	En proceso	2024-04-02 14:20:00			4
7	7	Plaza del Sol calle 76	7890123478901234	2025-07-01	En tránsito	2024-04-02 16:00:00			7
8	8	Avenida 78 calle 34	8901234589012345	2025-08-01	Entregado	2024-04-02 17:10:00	2024-04-02 19:00:00	2024-04-02 19:30:00	8
9	9	Manzana 45 lote 18	9012345690123456	2025-09-01	En proceso	2024-04-02 08:00:00			9
10	10	Diagonal 19 transversal 6	0123456701234567	2025-10-01	Entregado	2024-04-02 12:20:00	2024-04-02 13:45:00	2024-04-02 14:15:00	10

id_almacenista	id_cliente
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10

-- Productos

```
INSERT INTO Productos (id, nombre, marca, origen, volumen, peso,
fotografia, precio, stock, nombre_categoria) VALUES
(1, 'Arroz', 'Diana', 'Colombia', '1 kg', '1 kg', 'arroz.jpg', 5000, 100,
'Alimentos'),
(2, 'Leche', 'Alquería', 'Colombia', '1 litro', '1 litro', 'leche.jpg',
3000, 200, 'Alimentos'),
(3, 'Pan', 'Bimbo', 'Colombia', '500 g', '500 g', 'pan.jpg', 2000, 150,
'Alimentos'),
(4, 'Azúcar', 'Incauca', 'Colombia', '1 kg', '1 kg', 'azucar.jpg', 4000,
80, 'Alimentos'),
(5, 'Aceite', 'Doñarepa', 'Colombia', '500 ml', '500 ml', 'aceite.jpg',
6000, 120, 'Alimentos'),
(6, 'Jabón', 'Protex', 'Colombia', '100 g', '100 g', 'jabon.jpg', 1500,
300, 'Higiene'),
(7, 'Shampoo', 'Sedal', 'Colombia', '400 ml', '400 ml', 'shampoo.jpg',
4000, 100, 'Higiene'),
(8, 'Cepillo de dientes', 'Colgate', 'Colombia', '1 unidad', '50 g',
'cepillo_dientes.jpg', 2500, 200, 'Higiene'),
(9, 'Papel higiénico', 'Scott', 'Colombia', '4 rollos', '100 g',
'papel_higienico.jpg', 6000, 150, 'Higiene'),
(10, 'Desodorante', 'Axe', 'Colombia', '150 ml', '150 ml',
'desodorante.jpg', 4500, 80, 'Higiene');
```

Result Grid										
Filter Rows:										
Edit:										
Export/Import:										
Wrap Cell Content:										
	id	nombre	marca	origen	volumen	peso	fotografia	precio	stock	nombre_categoria
▶	1	Arroz	Diana	Colombia	1 kg	1 kg	arroz.jpg	5000	100	Alimentos
	2	Leche	Alquería	Colombia	1 litro	1 litro	leche.jpg	3000	200	Alimentos
	3	Pan	Bimbo	Colombia	500 g	500 g	pan.jpg	2000	150	Alimentos
	4	Azúcar	Incauca	Colombia	1 kg	1 kg	azucar.jpg	4000	80	Alimentos
	5	Aceite	Doñarepa	Colombia	500 ml	500 ml	aceite.jpg	6000	120	Alimentos
	6	Jabón	Protex	Colombia	100 g	100 g	jabon.jpg	1500	300	Higiene
	7	Shampoo	Sedal	Colombia	400 ml	400 ml	shampoo.jpg	4000	100	Higiene
	8	Cepillo de dientes	Colgate	Colombia	1 unidad	50 g	cepillo_dientes.jpg	2500	200	Higiene
	9	Papel higiénico	Scott	Colombia	4 rollos	100 g	papel_higienico.jpg	6000	150	Higiene
	10	Desodorante	Axe	Colombia	150 ml	150 ml	desodorante.jpg	4500	80	Higiene
	11	cuchilla de hoja	Dorco	Colombia	2g	2g	Cuchilla.jpg	500	60	Higiene
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

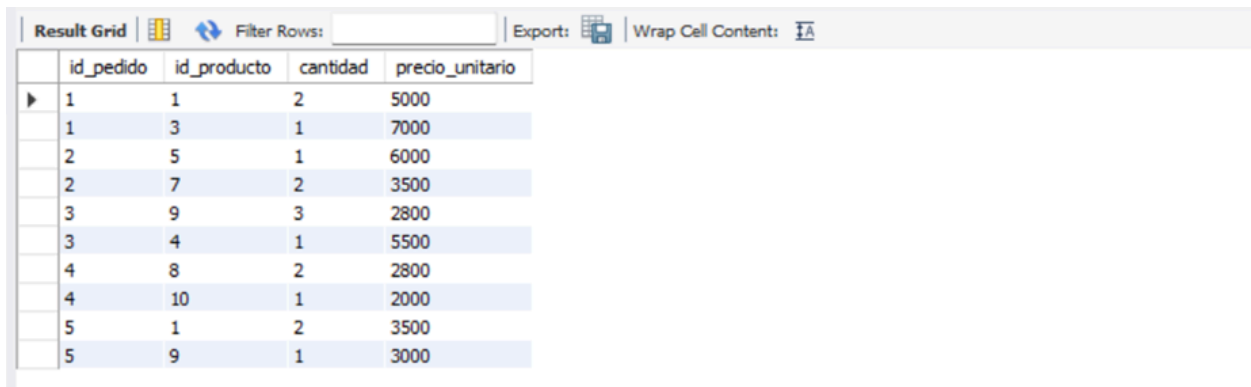
```
-- Repartidores_codigos_postales
```

```
INSERT INTO repartidores_codigos_postales (id_repartidor, codigo_postal)
VALUES
(1, '110111'),
(1, '220222'),
(2, '330333'),
(2, '440444'),
(3, '550555'),
(3, '660666'),
(4, '770777'),
(4, '001010'),
(5, '110121'),
(5, '220232');
```

Result Grid		
Filter Rows:		
Edit:		
Export/Import:		
	id_repartidor	codigo_postal
▶	1	110111
	1	220222
	2	330333
	2	440444
	3	550555
	3	660666
	4	001010
	4	770777
	5	110121
	5	220232
*	NULL	NULL

```
-- Pedidos_productos
```

```
INSERT INTO pedidos_productos (id_pedido, id_producto, cantidad,
precio_unitario) VALUES
(1, 1, 2, 5000),
(1, 3, 1, 7000),
(2, 5, 1, 6000),
(2, 7, 2, 3500),
(3, 9, 3, 2800),
(3, 4, 1, 5500),
(4, 8, 2, 2800),
(4, 10, 1, 2000),
(5, 1, 2, 3500),
(5, 9, 1, 3000);
```



	id_pedido	id_producto	cantidad	precio_unitario
▶	1	1	2	5000
	1	3	1	7000
	2	5	1	6000
	2	7	2	3500
	3	9	3	2800
	3	4	1	5500
	4	8	2	2800
	4	10	1	2000
	5	1	2	3500
	5	9	1	3000

## Creación de vistas

```
-- Listado de pedidos con sus productos
```

```
CREATE VIEW pedidos_con_productos AS
SELECT
    pe.*, pp.id_producto, p.nombre AS `producto`, p.marca,
    pp.precio_unitario, pp.cantidad
FROM
    Pedidos pe
    LEFT JOIN
        pedidos_productos pp ON pp.id_pedido = pe.id
    LEFT JOIN
        Productos p ON pp.id_producto = p.id
SELECT * FROM pedidos_con_productos;
```

	id	direccion	numero_tarjeta	vencimiento_tarjeta	estado	fecha_solicitud	fecha_servido	fecha_entregado	id_repartidor
▶	1	Carrera 10 #25-30	1234567812345678	2025-01-01	En proceso	2024-04-02 10:00:00	NULL	NULL	1
	1	Carrera 10 #25-30	1234567812345678	2025-01-01	En proceso	2024-04-02 10:00:00	NULL	NULL	1
	2	Avenida 456 # 15-67	2345678923456789	2025-02-01	En tránsito	2024-04-02 11:30:00	NULL	NULL	2
	2	Avenida 456 # 15-67	2345678923456789	2025-02-01	En tránsito	2024-04-02 11:30:00	NULL	NULL	2
	3	Avenida 5 #8-12	3456789034567890	2025-03-01	Entregado	2024-04-02 13:45:00	2024-04-02 15:30:00	2024-04-02 16:15:00	3
	3	Avenida 5 #8-12	3456789034567890	2025-03-01	Entregado	2024-04-02 13:45:00	2024-04-02 15:30:00	2024-04-02 16:15:00	3

	id_almacenista	id_cliente	id_producto	producto	marca	precio_unitario	cantidad
	1	1	1	Arroz	Diana	5000	2
	1	1	3	Pan	Bimbo	7000	1
	2	2	5	Aceite	Doñarepa	6000	1
	2	2	7	Shampoo	Sedal	3500	2
	3	3	9	Papel higiénico	Scott	2800	3
	3	3	4	Azúcar	Incauca	5500	1

-- Listado de pedidos con su repartidor

```
CREATE VIEW pedidos_con_repartidor AS
SELECT
  pe.*, r.nombre AS `repartidor`
FROM
  Pedidos pe
  LEFT JOIN
  Repartidores r ON pe.id_repartidor = r.id;
SELECT * FROM pedidos_con_repartidor;
```

	id	direccion	numero_tarjeta	vencimiento_tarjeta	estado	fecha_solicitud	fecha_servido	fecha_entregado	id_repartidor
▶	1	Carrera 10 #25-30	1234567812345678	2025-01-01	En proceso	2024-04-02 10:00:00	NULL	NULL	1
	2	Avenida 456 # 15-67	2345678923456789	2025-02-01	En tránsito	2024-04-02 11:30:00	NULL	NULL	2
	3	Avenida 5 #8-12	3456789034567890	2025-03-01	Entregado	2024-04-02 13:45:00	2024-04-02 15:30:00	2024-04-02 16:15:00	3
	4	Calle 15 #30-18	4567890145678901	2025-04-01	En proceso	2024-04-02 09:15:00	NULL	NULL	4
	5	calle 16 # 24-19	5678901256789012	2025-05-01	Entregado	2024-04-02 10:45:00	2024-04-02 12:30:00	2024-04-02 13:00:00	5
	6	Calle 456 # 29-56	6789012367890123	2025-06-01	En proceso	2024-04-02 14:20:00	NULL	NULL	4

	id_almacenista	id_cliente	repartidor
	1	1	Marlon Buendia
	2	2	Ismael Arsuaga
	3	3	Leonel Perez
	4	4	Jesus daniel Lozano
	5	5	Ahi Landero
	6	6	Jesus daniel Lozano

-- Listado de pedidos con su almacenista

```
CREATE VIEW pedidos_con_almacenista AS
```



```

SELECT
pe.*, a.nombre AS `almacenista`
FROM
Pedidos pe
LEFT JOIN
Repartidores r ON pe.id_almacenista = a.id
SELECT * FROM pedidos_con_almacenista;

```

	id	direccion	numero_tarjeta	vencimiento_tarjeta	estado	fecha_solicitud	fecha_servido	fecha_entregado	id_repartidor
▶	1	Carrera 10 #25-30	1234567812345678	2025-01-01	En proceso	2024-04-02 10:00:00	NULL	NULL	1
	2	Avenida 456 # 15-67	2345678923456789	2025-02-01	En tránsito	2024-04-02 11:30:00	NULL	NULL	2
	3	Avenida 5 #8-12	3456789034567890	2025-03-01	Entregado	2024-04-02 13:45:00	2024-04-02 15:30:00	2024-04-02 16:15:00	3
	4	Calle 15 #30-18	4567890145678901	2025-04-01	En proceso	2024-04-02 09:15:00	NULL	NULL	4
	5	calle 16 # 24-19	5678901256789012	2025-05-01	Entregado	2024-04-02 10:45:00	2024-04-02 12:30:00	2024-04-02 13:00:00	5
	6	Calle 456 # 29-56	6789012367890123	2025-06-01	En proceso	2024-04-02 14:20:00	NULL	NULL	4

id_almacenista	id_cliente	almacenista
1	1	Juan Pérez
2	2	María García
3	3	Luis Martínez
4	4	Ana Rodríguez
5	5	Pedro López
6	6	Laura González

-- productos más vendidos

```

CREATE VIEW productos_mas_vendidos AS
SELECT
    p.id AS id_producto,
    p.nombre AS nombre_producto,
    p.marca AS marca_producto,
    SUM(pp.cantidad) AS total_vendido
FROM
    Productos p
INNER JOIN
    pedidos_productos pp ON p.id = pp.id_producto
GROUP BY p.id
ORDER BY total_vendido DESC;

```

	id_producto	nombre_producto	marca_producto	total_vendido
▶	1	Arroz	Diana	4
	9	Papel higiénico	Scott	4
	7	Shampoo	Sedal	2
	8	Cepillo de dientes	Colgate	2
	3	Pan	Bimbo	1
	5	Aceite	Doñarepa	1
	4	Azúcar	Incauca	1

nas\_vendidos 1 x

## Consultas a la base de datos

```
-- Listado de productos con categoría
SELECT
    p.*, c.condicion_almacenaje, c.observaciones
FROM
    Productos p
    LEFT JOIN
    Categorías c ON p.nombre_categoria = c.nombre;
```

Result Grid				Filter Rows:		Export:		Wrap Cell Content:				
	id	nombre	marca	origen	volumen	peso	fotografia	precio	stock	nombre_categoria	condicion_almacenaje	observaciones
▶	1	Arroz	Diana	Colombia	1 kg	1 kg	arroz.jpg	5000	100	Alimentos	Seco	Categoría principal para alimentos no perecederos
	2	Leche	Alquería	Colombia	1 litro	1 litro	leche.jpg	3000	200	Alimentos	Seco	Categoría principal para alimentos no perecederos
	3	Pan	Bimbo	Colombia	500 g	500 g	pan.jpg	2000	150	Alimentos	Seco	Categoría principal para alimentos no perecederos
	4	Azúcar	Incauca	Colombia	1 kg	1 kg	azucar.jpg	4000	80	Alimentos	Seco	Categoría principal para alimentos no perecederos
	5	Aceite	Doñarepa	Colombia	500 ml	500 ml	aceite.jpg	6000	120	Alimentos	Seco	Categoría principal para alimentos no perecederos
	6	Jabón	Protex	Colombia	100 g	100 g	jabon.jpg	1500	300	Higiene	Seco	Productos de higiene personal
	7	Shampoo	Sedal	Colombia	400 ml	400 ml	shampoo.jpg	4000	100	Higiene	Seco	Productos de higiene personal
	8	Cepillo de dientes	Colgate	Colombia	1 unidad	50 g	cepillo_dientes.jpg	2500	200	Higiene	Seco	Productos de higiene personal
	9	Papel higiénico	Scott	Colombia	4 rollos	100 g	papel_higienico.jpg	6000	150	Higiene	Seco	Productos de higiene personal
	10	Desodorante	Axe	Colombia	150 ml	150 ml	desodorante.jpg	4500	80	Higiene	Seco	Productos de higiene personal
	11	cuchilla de hoja	Dorco	Colombia	2g	2g	Cuchilla.jpg	500	60	Higiene	Seco	Productos de higiene personal

```
-- Listado ordenado de repartidores que más pedidos entregan
SELECT r.*, count(p.id) as `cantidad_pedidos` FROM Repartidores r LEFT JOIN
Pedidos p on p.id_repartidor = r.id;
```

Result Grid				
	id	nombre	matricula_furgoneta	cantidad_pedidos
▶	1	Marlon Buendia	123ABC	10

-- Listado ordenado de almacenistas que más pedidos preparan

```
SELECT a.*, count(p.id) as `cantidad_pedidos` FROM Almacenistas a LEFT JOIN
Pedidos p on p.id_almacenista = a.id;
```

Result Grid			
	id	nombre	cantidad_pedidos
▶	1	Juan Pérez	10

-- Listado de pedidos pendientes por entregar

```
SELECT * FROM pedidos_con_repartidor pr WHERE pr.estado = "Pendiente
entrega";
```

Result Grid												
	id	direccion	numero_tarjeta	vencimiento_tarjeta	estado	fecha_solicitud	fecha_servido	fecha_entregado	id_repartidor	id_almacenista	id_cliente	repartidor
▶	2	Avenida 456 # 15-67	2345678923456789	2025-02-01	Pendiente entrega	2024-04-02 11:30:00	NULL	NULL	2	2	2	Ismael A
	7	Plaza del Sol calle 76	7890123478901234	2025-07-01	Pendiente entrega	2024-04-02 16:00:00	NULL	NULL	7	7	7	Rodrigo

-- Listado de pedidos entregados con su repartidor

```
SELECT * FROM pedidos_con_repartidor pr WHERE pr.estado = 'Entregado';
```

id	direccion	numero_tarjeta	vencimiento_tarjeta	estado	fecha_solicitud	fecha_servido	fecha_entregado
▶ 3	Avenida 5 #8-12	3456789034567890	2025-03-01	Entregado	2024-04-02 13:45:00	2024-04-02 15:30:00	2024-04-02 16:15:00
5	calle 16 # 24-19	5678901256789012	2025-05-01	Entregado	2024-04-02 10:45:00	2024-04-02 12:30:00	2024-04-02 13:00:00
8	Avenida 78 calle 34	8901234589012345	2025-08-01	Entregado	2024-04-02 17:10:00	2024-04-02 19:00:00	2024-04-02 19:30:00
10	Diagonal 19 transversal 6	0123456701234567	2025-10-01	Entregado	2024-04-02 12:20:00	2024-04-02 13:45:00	2024-04-02 14:15:00

id_repartidor	id_almacenista	id_cliente	repartidor
3	3	3	Leonel Perez
5	5	5	Ahi Landero
8	8	8	Hernan Sifuentes
10	10	10	Pablo Ruiz

-- Listado de pedidos pendientes por preparar

```
SELECT * FROM pedidos_con_almacenista pr WHERE pr.estado = "En
preparación";
```

id	direccion	numero_tarjeta	vencimiento_tarjeta	estado	fecha_solicitud	fecha_servido	fecha_entregado	id_repartidor	id_almacenista	id_cliente	almacenista
1	Carrera 10 #25-30	1234567812345678	2025-01-01	En preparación	2024-04-02 10:00:00	NULL	NULL	1	1	1	Juan Pérez
4	Calle 15 #30-18	4567890145678901	2025-04-01	En preparación	2024-04-02 09:15:00	NULL	NULL	4	4	4	Ana Rodríguez
6	Calle 456 # 29-56	6789012367890123	2025-06-01	En preparación	2024-04-02 14:20:00	NULL	NULL	4	6	6	Laura González
9	Manzana 45 lote 18	9012345690123456	2025-09-01	En preparación	2024-04-02 08:00:00	NULL	NULL	9	9	9	Pablo Díaz



```
-- Listado de pedidos ya preparados con su almacenista
```

```
SELECT * FROM pedidos_con_almacenista pr WHERE pr.estado = "Pendiente
entrega";
```

id	direccion	numero_tarjeta	vencimiento_tarjeta	estado	fecha_solicitud	fecha_servido	fecha_entregado	id_repartidor	id_almacenista	id_cliente	almacenista
2	Avenida 456 # 15-67	2345678923456789	2025-02-01	Pendiente entrega	2024-04-02 11:30:00	NULL	NULL	2	2	2	María García
7	Plaza del Sol calle 76	7890123478901234	2025-07-01	Pendiente entrega	2024-04-02 16:00:00	NULL	NULL	7	7	7	Carlos Sánchez



```
-- lista de clientes ordenados de mayor a menor valor total de compra
```

```
SELECT
    c.id,
    CONCAT(c.nombres, ' ', c.apellidos) AS nombre_cliente,
    COUNT(p.id) AS total_pedidos,
    SUM(pp.precio_unitario * pp.cantidad) AS valor_total_compra
FROM
    Clientes c
JOIN
    Pedidos p ON c.id = p.id_cliente
JOIN
    pedidos_productos pp ON p.id = pp.id_pedido
GROUP BY
    c.id
ORDER BY
    valor_total_compra DESC;
```

Result Grid				
		Filter Rows:		Export:  Wrap Cell Content: 
	id	nombre_cliente	total_pedidos	valor_total_compra
▶	1	Juan Hernandez	2	17000
	3	Carlos Vertel	2	13900
	2	María Consa	2	13000
	5	Luis Velazques	2	10000
	4	Dina Martínez	2	7600

```
-- Listado ordenado de productos más vendidos
```

```
SELECT
    pcp.id_producto,
    pcp.producto,
    SUM(pcp.cantidad) AS `cantidad_vendidos`
FROM
    pedidos_con_productos pcp
GROUP BY pcp.id_producto
ORDER BY cantidad_vendidos DESC;
```

Result Grid			
		Filter Rows:	
		Export:	 Wrap Cell Content: 
	id_producto	producto	cantidad_vendidos
▶	1	Arroz	4
	9	Papel higiénico	4
	7	Shampoo	2
	8	Cepillo de dientes	2
	3	Pan	1
	5	Aceite	1
	4	Azúcar	1
	10	Desodorante	1
	NULL	NULL	NULL

## Definición de procedimientos almacenados

```
DELIMITER $$
```

```
-- Listado de productos por categoría
```

```
CREATE PROCEDURE productos_en_categoria(IN nombre_categoria VARCHAR(40))
```

```
BEGIN
```

```
    SELECT * FROM Productos p WHERE p.nombre_categoria =
```

```

nombre_categoria;
END $$
DELIMITER ;
-- Ejemplo:
CALL productos_en_categoria("alimentos");

```

Result Grid										
		Filter Rows:		Export:		Wrap Cell Content:				
	id	nombre	marca	origen	volumen	peso	fotografia	precio	stock	nombre_categoria
▶	1	Arroz	Diana	Colombia	1 kg	1 kg	arroz.jpg	5000	100	Alimentos
	2	Leche	Alquería	Colombia	1 litro	1 litro	leche.jpg	3000	200	Alimentos
	3	Pan	Bimbo	Colombia	500 g	500 g	pan.jpg	2000	150	Alimentos
	4	Azúcar	Incauca	Colombia	1 kg	1 kg	azucar.jpg	4000	80	Alimentos
	5	Aceite	Doñarepa	Colombia	500 ml	500 ml	aceite.jpg	6000	120	Alimentos

```

CALL productos_en_categoria("Higiene");

```

Result Grid										
		Filter Rows:		Export:		Wrap Cell Content:				
	id	nombre	marca	origen	volumen	peso	fotografia	precio	stock	nombre_categoria
▶	6	Jabón	Protex	Colombia	100 g	100 g	jabon.jpg	1500	300	Higiene
	7	Shampoo	Sedal	Colombia	400 ml	400 ml	shampoo.jpg	4000	100	Higiene
	8	Cepillo de dientes	Colgate	Colombia	1 unidad	50 g	cepillo_dientes.jpg	2500	200	Higiene
	9	Papel higiénico	Scott	Colombia	4 rollos	100 g	papel_higienico.jpg	6000	150	Higiene
	10	Desodorante	Axe	Colombia	150 ml	150 ml	desodorante.jpg	4500	80	Higiene
	11	cuchilla de hoja	Dorco	Colombia	2g	2g	Cuchilla.jpg	500	60	Higiene

```

DELIMITER $$

```

```

-- Detalles de pedido con importe total

```

```

CREATE PROCEDURE pedido_con_valor(IN id_pedido INT)
BEGIN
    SELECT
        p.*,
        CONCAT(c.nombres, ' ', c.apellidos) AS `cliente`,
        SUM(pp.precio_unitario * pp.cantidad) AS `importe_total`
    FROM
        (SELECT * FROM Pedidos p WHERE p.id = id_pedido) AS p
    LEFT JOIN
        pedidos_productos pp ON pp.id_pedido = p.id
    LEFT JOIN
        Clientes c ON p.id_cliente = c.id
    GROUP BY p.id;

```

```

END $$
DELIMITER ;

CALL pedido_con_valor(1);

```

Result Grid								
		Filter Rows:			Export:	Wrap Cell Content:		
	id	direccion	numero_tarjeta	vencimiento_tarjeta	estado	fecha_solicitud	fecha_servido	fecha_entregado
▶	1	Carrera 10 #25-30	1234567812345678	2025-01-01	En preparación	2024-04-02 10:00:00	NULL	NULL

id_repartidor	id_almacenista	id_cliente	cliente	importe_total
1	1	1	Juan Hernandez	17000

```

DELIMITER $$
-----
-- Insertar un nuevo producto
-----

CREATE PROCEDURE insertar_producto(
    IN id INT,
    IN nombre_producto VARCHAR(50),
    IN marca_producto VARCHAR(40),
    IN origen_producto VARCHAR(40),
    IN volumen_producto VARCHAR(20),
    IN peso_producto VARCHAR(20),
    IN fotografia_producto VARCHAR(255),
    IN precio_producto BIGINT,
    IN stock_producto BIGINT,
    IN categoria_producto VARCHAR(40)
)
BEGIN
    INSERT INTO Productos (id,nombre, marca, origen, volumen, peso,
fotografia, precio, stock, nombre_categoria)
    VALUES (id,nombre_producto, marca_producto, origen_producto,
volumen_producto, peso_producto, fotografia_producto, precio_producto,
stock_producto, categoria_producto);
END $$
DELIMITER ;
-- ejemplo
CALL insertar_producto('12','botella de
Agua','Brisa','Colombia','20ml','200g','botella_de_agua.jpg','1500','30','B
ebidas');

```

Result Grid										
Filter Rows:										
Edit:										
Export/Import:										
Wrap Cell Content:										
	id	nombre	marca	origen	volumen	peso	fotografia	precio	stock	nombre_categoria
▶	1	Arroz	Diana	Colombia	1 kg	1 kg	arroz.jpg	5000	100	Alimentos
	2	Leche	Alquería	Colombia	1 litro	1 litro	leche.jpg	3000	200	Alimentos
	3	Pan	Bimbo	Colombia	500 g	500 g	pan.jpg	2000	150	Alimentos
	4	Azúcar	Incauca	Colombia	1 kg	1 kg	azucar.jpg	4000	80	Alimentos
	5	Aceite	Doña repa	Colombia	500 ml	500 ml	aceite.jpg	6000	120	Alimentos
	6	Jabón	Protex	Colombia	100 g	100 g	jabon.jpg	1500	300	Higiene
	7	Shampoo	Sedal	Colombia	400 ml	400 ml	shampoo.jpg	4000	100	Higiene
	8	Cepillo de dientes	Colgate	Colombia	1 unidad	50 g	cepillo_dientes.jpg	2500	200	Higiene
	9	Papel higiénico	Scott	Colombia	4 rollos	100 g	papel_higienico.jpg	6000	150	Higiene
	10	Desodorante	Axe	Colombia	150 ml	150 ml	desodorante.jpg	4500	80	Higiene
	11	cuchilla de hoja	Dorco	Colombia	2g	2g	Cuchilla.jpg	500	60	Higiene
	12	botella de Agua	Brisa	Colombia	20ml	200g	botella_de_agua.jpg	1500	30	Bebidas
✱	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

```
-- eliminar producto de la tabla productos
```

```
DELIMITER $$
CREATE PROCEDURE eliminar_producto(
    IN id_producto INT
)
BEGIN
    DELETE FROM Productos
    WHERE id = id_producto;
END $$
DELIMITER ;
```

```
CALL eliminar_producto(12);
```

✓	196	21:27:47	CALL eliminar_producto(12)	1 row(s) affected
---	-----	----------	----------------------------	-------------------



	id	nombre	marca	origen	volumen	peso	fotografia	precio	stock	nombre_categoria
▶	1	Arroz	Diana	Colombia	1 kg		Insert new row	5000	100	Alimentos
	2	Leche	Alquería	Colombia	1 litro	1 litro	leche.jpg	3000	200	Alimentos
	3	Pan	Bimbo	Colombia	500 g	500 g	pan.jpg	2000	150	Alimentos
	4	Azúcar	Incauca	Colombia	1 kg	1 kg	azucar.jpg	4000	80	Alimentos
	5	Aceite	Doñarepa	Colombia	500 ml	500 ml	aceite.jpg	6000	120	Alimentos
	6	Jabón	Protex	Colombia	100 g	100 g	jabon.jpg	1500	300	Higiene
	7	Shampoo	Sedal	Colombia	400 ml	400 ml	shampoo.jpg	4000	100	Higiene
	8	Cepillo de dientes	Colgate	Colombia	1 unidad	50 g	cepillo_dientes.jpg	2500	200	Higiene
	9	Papel higiénico	Scott	Colombia	4 rollos	100 g	papel_higienico.jpg	6000	150	Higiene
	10	Desodorante	Axe	Colombia	150 ml	150 ml	desodorante.jpg	4500	80	Higiene
	11	cuchilla de hoja	Dorco	Colombia	2g	2g	Cuchilla.jpg	500	60	Higiene
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

## Definición de triggers

- Se crea la tabla historial\_clientes

```
CREATE TABLE IF NOT EXISTS historial_clientes (
    id INT AUTO_INCREMENT PRIMARY KEY,
    id_cliente INT,
    fecha TIMESTAMP,
    accion VARCHAR(100)
) ENGINE=INNODB;
```

-- -- Registrar la inserción del nuevo cliente en el historial de clientes

DELIMITER \$\$

CREATE TRIGGER after\_insert\_cliente

AFTER INSERT ON Clientes

FOR EACH ROW

BEGIN

INSERT INTO historial\_clientes (id\_cliente, fecha, accion)

VALUES (NEW.id, NOW(), 'Nuevo cliente registrado');

END \$\$

-- Registrar la eliminación de un cliente en el historial de clientes

CREATE TRIGGER after\_delete\_cliente

AFTER DELETE ON Clientes

FOR EACH ROW

BEGIN

-- Registrar la eliminación del cliente en el historial de clientes

```

INSERT INTO historial_clientes (id_cliente, fecha, accion)
VALUES (OLD.id, NOW(), 'Cliente eliminado');
END $$
DELIMITER ;

```

id	id_cliente	fecha	accion
1	24	2024-04-02 22:20:04	Nuevo cliente registrado
2	24	2024-04-02 22:23:04	Cliente eliminado
NULL	NULL	NULL	NULL

```

CREATE TABLE IF NOT EXISTS historial_precios (
    id_producto INT,
    precio_anterior BIGINT,
    precio_nuevo BIGINT,
    fecha TIMESTAMP,
    actualizado_por VARCHAR(50)
) ENGINE = INNODB;

```

```

CREATE
    TRIGGER precios_registro
    AFTER UPDATE ON Productos FOR EACH ROW
    INSERT INTO historial_precios VALUES (NEW.id , OLD.precio , NEW.precio
    , NOW() , CURRENT_USER());

```

– precio producto actualizado

id	nombre	marca	origen	volumen	peso	fotografia	precio	stock	nombre_categoria
7	Shampoo	Sedal	Colombia	400 ml	400 ml	shampoo.jpg	4000	100	Higiene
8	Cepillo de dientes	Colgate	Colombia	1 unidad	50 g	cepillo_dientes.jpg	2500	200	Higiene
9	Papel higiénico	Scott	Colombia	4 rollos	100 g	papel_higienico.jpg	6500	150	Higiene
10	Desodorante	Axe	Colombia	150 ml	150 ml	desodorante.jpg	6000	80	Higiene

– se agrega la actualización del precio del producto a la tabla historial precio

id_producto	precio_anterior	precio_nuevo	fecha	actualizado_por
9	5900	6500	2024-04-03 11:01:17	root@localhost

-- Guardar productos borrados en una tabla de respaldo

```
CREATE TABLE IF NOT EXISTS productos_archivo LIKE Productos;
```

```
CREATE
```

```
    TRIGGER delete_logger
```

```
    AFTER DELETE ON Productos FOR EACH ROW
```

```
    INSERT INTO productos_archivo VALUES
```

```
        (OLD.id , OLD.nombre , OLD.marca , OLD.origen , OLD.volumen ,  
        OLD.peso , OLD.fotografia , OLD.precio , OLD.stock , OLD.nombre_categoria);
```

– se elimina el registro de la tabla producto

```
48 11:21:43 DELETE FROM productos WHERE id = 12 1 row(s) affected
```

– se agrega el producto borrado a la tabla productos archivo

Result Grid										
Filter Rows:										
Edit: Export/Import: Wrap Cell Content:										
	id	nombre	marca	origen	volumen	peso	fotografia	precio	stock	nombre_categoria
▶	12	Avena	Alpina	Colombia	100 ml	100 ml		6800	100	Alimentos
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

## Conclusión

- ¿ Está conforme con el resultado obtenido según el contexto o cree que hubiera obtenido un mejor resultado con una base de datos no relacional?

Estamos satisfechos con la base de datos que hemos diseñado, definido e implementado utilizando MySQL. En el contexto del supermercado local, consideramos que las bases de datos relacionales son una elección adecuada. Esto se debe a que el modelo de negocio se basa en procesos conocidos, no prevé cambios fundamentales y trabaja con datos estructurados, como el control de inventario, las transacciones y la alta interconexión de la información.

Por otro lado Si la aplicación maneja grandes volúmenes de datos, necesita escalabilidad o mayor flexibilidad en el esquema de datos no estructurados, entonces una base de datos no relacional podría ser más adecuada.