

## Sprint Final Módulo 3

### Integrantes:

- Ana Andrade
- Matias Henriquez Torres
- Jorge Lira Montalba
- Bárbara Ramos
- Lorena Suárez

### SPRINT DE ENTREGA:

Se solicita como entregable de este Sprint la implementación final de todos los conceptos vistos durante el Módulo 2: Bases de datos. Por tanto, se debe poner foco en lo siguiente:

Comentar debidamente el código para que sea comprensible por un tercero.

El script SQL debe utilizarse para crear la estructura de la base de datos, realizar operaciones en la base de datos (p. ej. rellenarla con datos) y cambiar o eliminar la estructura de la base de datos.

-- Creando base de datos

```
CREATE DATABASE `telovendo-m3`;
```

**Deben crear un usuario con privilegios para crear, eliminar y modificar tablas, insertar registros.**

-- Creando usuario y otorgándole los permisos para crear, eliminar, modificar tablas, e insertar registros.

```
CREATE USER 'exploradorsprint3'@'localhost' IDENTIFIED BY '1234';
```

```
GRANT SELECT, CREATE, DROP, ALTER, INSERT, UPDATE ON `telovendo-m3`. * TO 'exploradorsprint3'@'localhost';
```

```
FLUSH PRIVILEGES;
```

-- Seleccionando la base de datos recién creada para su uso

```
USE `telovendo-m3`;
```

-- Crear tabla Proveedor

```
CREATE TABLE Proveedor (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nombre_representante VARCHAR(100),  
    nombre_corporativo VARCHAR(100),  
    telefono1 VARCHAR(15),  
    contacto1 VARCHAR(100),  
    telefono2 VARCHAR(15),  
    contacto2 VARCHAR(100),  
    categoria_proveedor VARCHAR(50),  
    correo_electronico VARCHAR(100)
```

```
);
```

```
-- Crear tabla Cliente
```

```
CREATE TABLE Cliente (id INT AUTO_INCREMENT PRIMARY KEY,  
    nombre VARCHAR(100),  
    apellido VARCHAR(100),  
    direccion VARCHAR(255)
```

```
);
```

```
-- Crear tabla Producto
```

```
CREATE TABLE Producto (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nombre VARCHAR(100),  
    precio DECIMAL(10, 2),  
    categoria_producto VARCHAR(50),  
    color VARCHAR(50),  
    stock INT
```

```
);
```

```
-- Crear tabla Proveedor_Producto (relación muchos a muchos)
```

```
CREATE TABLE Proveedor_Producto (  
    id_proveedor INT, id_producto INT,  
    FOREIGN KEY (id_proveedor) REFERENCES Proveedor(id),  
    FOREIGN KEY (id_producto) REFERENCES Producto(id),  
    PRIMARY KEY (id_proveedor, id_producto)
```

```
);
```

```
-- Insertar registros en Proveedor
```

```
INSERT INTO Proveedor (nombre_representante, nombre_corporativo, telefono1,  
contacto1, telefono2, contacto2,  
categoria_proveedor, correo_electronico) VALUES  
(  
'Juan Pérez', 'ElectroMundo', '123456789', 'Ana López', '987654321', 'Carlos Méndez',  
'Electrónica', 'juan@electromundo.com'),  
(  
'María García', 'TecnoHouse', '234567890', 'Luis Fernández', '876543210', 'Sara Martínez',  
'Electrónica', 'maria@tecnohouse.com'),  
(  
'Carlos Ruiz', 'CompuStore', '345678901', 'Pedro González', '765432109', 'Laura Gutiérrez',  
'Computación', 'carlos@compustore.com'),  
(  
'Ana Torres', 'GadgetShop', '456789012', 'Diego Rojas', '654321098', 'Elena Castillo',  
'Electrónica', 'ana@gadgetshop.com'),  
(  
'Luis Martínez', 'Innovatech', '567890123', 'Jorge Pérez', '543210987', 'Lucía Sánchez',  
'Tecnología', 'luis@innovatech.com');
```

```
-- Insertar registros en Cliente
```

```
INSERT INTO Cliente (nombre, apellido, direccion) VALUES  
(  
'Pedro', 'Gómez', 'Calle Falsa 123'),
```

('Laura', 'Hernández', 'Avenida Siempre Viva 456'),  
('Jorge', 'Pérez', 'Boulevard de los Sueños Rotos 789'),  
('Sara', 'López', 'Calle del Olvido 101'),  
('Lucía', 'Martínez', 'Calle de la Amargura 202');

-- Insertar registros en Producto

```
INSERT INTO Producto (nombre, precio, categoria_producto, color, stock) VALUES  
('Repisa', 59990.00, 'Muebles', 'Café', 50),  
('Smartphone', 219990.00, 'Electrónica y Computación', 'Gris Oscuro', 30),  
('Camiseta', 19990.00, 'Vestuario', 'Rojo', 100),  
('Jogger', 49990.00, 'Vestuario', 'Negro', 15),  
('Juguera', 39990.00, 'Línea Blanca', 'Negro', 100),  
('Escritorio', 150000.00, 'Muebles', 'Marrón', 20),  
('Silla de Oficina', 80000.00, 'Muebles', 'Negro', 15),  
('Impresora', 140780.00, 'Electrónica y Computación', 'Blanco', 10),  
('Chaqueta', 31990.00, 'Vestuario', 'Negro', 25),  
('Buzo Deportivo', 34990.00, 'Vestuario', 'Gris', 40),  
('Microondas', 68980.00, 'Línea Blanca', 'Plateado', 5);
```

-- Relacionar Proveedores con Productos

```
INSERT INTO Proveedor_Producto (id_proveedor, id_producto) VALUES  
(1, 1), (1, 2), (1, 3), (2, 4), (2, 5), (3, 6), (3, 7), (4, 8), (4, 9), (5, 10);
```

**Nuestra tienda virtual ha crecido mucho estas últimas semanas. No solo aumentó significativamente el número de colaboradores y usuarios, sino que también los productos disponibles.**

**En general, los proveedores venden muchos productos.**

**En este sentido, nos pidieron que diseñemos una base de datos capaz de satisfacer la creciente demanda de información y datos.**

**TeLoVendo recibe productos de diferentes proveedores para comercializarlos.**

**Cada proveedor debe informarnos el nombre del representante legal, su nombre corporativo, al menos dos teléfonos de contacto (y el nombre de quien recibe las llamadas), la categoría de sus productos (solo nos pueden indicar una categoría) y un correo electrónico para enviar la factura. Sabemos que la mayoría de los proveedores son de productos electrónicos.**

**Agregue 5 proveedores a la base de datos. En**

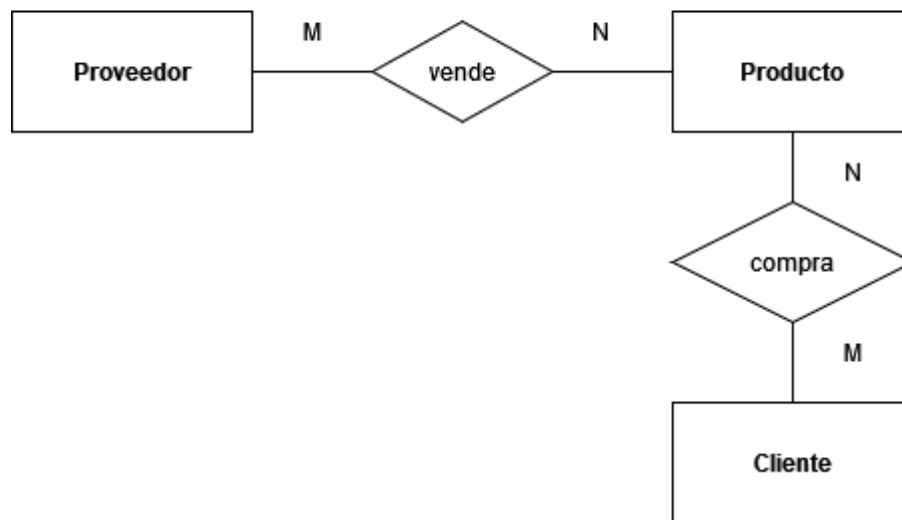
**TeLoVendo tiene actualmente muchos clientes, pero nos piden que ingresemos solo 5 para probar la nueva base de datos. Cada cliente tiene un nombre, apellido, dirección (solo pueden ingresar una).**

**TeLoVendo tiene diferentes productos. Ingrese 10 productos y su respectivo stock.**

Cada producto tiene información sobre su precio, su categoría, proveedor y color.  
Los productos pueden tener muchos proveedores.

Como un entregable, nos piden que diseñemos un diagrama entidad relación sólo con la información que tenemos.

## Te lo Vendo



**NOTA:** Como grupo propusimos crear una nueva tabla para anexar clientes y productos, de manera que sirva para constatar la compra, ya sea como transacción o boleto, de manera que sea más realista en la práctica.

En caso de tener nuevas ideas respecto a futura información requerida y nuevas entidades, solo nos piden que la indiquemos en un archivo .docx.

A partir del diagrama, debemos construir un script que cree tablas de acuerdo a las entidades e ingrese datos.

Luego debemos realizar consultas SQL que indiquen:

--Cuál es la categoría de productos que más se repite.

```
SELECT categoria_producto, COUNT(*) as cantidad
FROM Producto
GROUP BY categoria_producto
ORDER BY cantidad DESC
LIMIT 1;
```

- Cuáles son los productos con mayor stock

```
SELECT id, nombre, stock
FROM producto
ORDER BY stock DESC;
```

**-- 3 Qué color de producto es más común en nuestra tienda.**

```
SELECT color, COUNT(*) as cantidad
FROM Producto
GROUP BY color
ORDER BY cantidad DESC
LIMIT 1;
```

**-- Cual o cuales son los proveedores con menor stock de productos.**

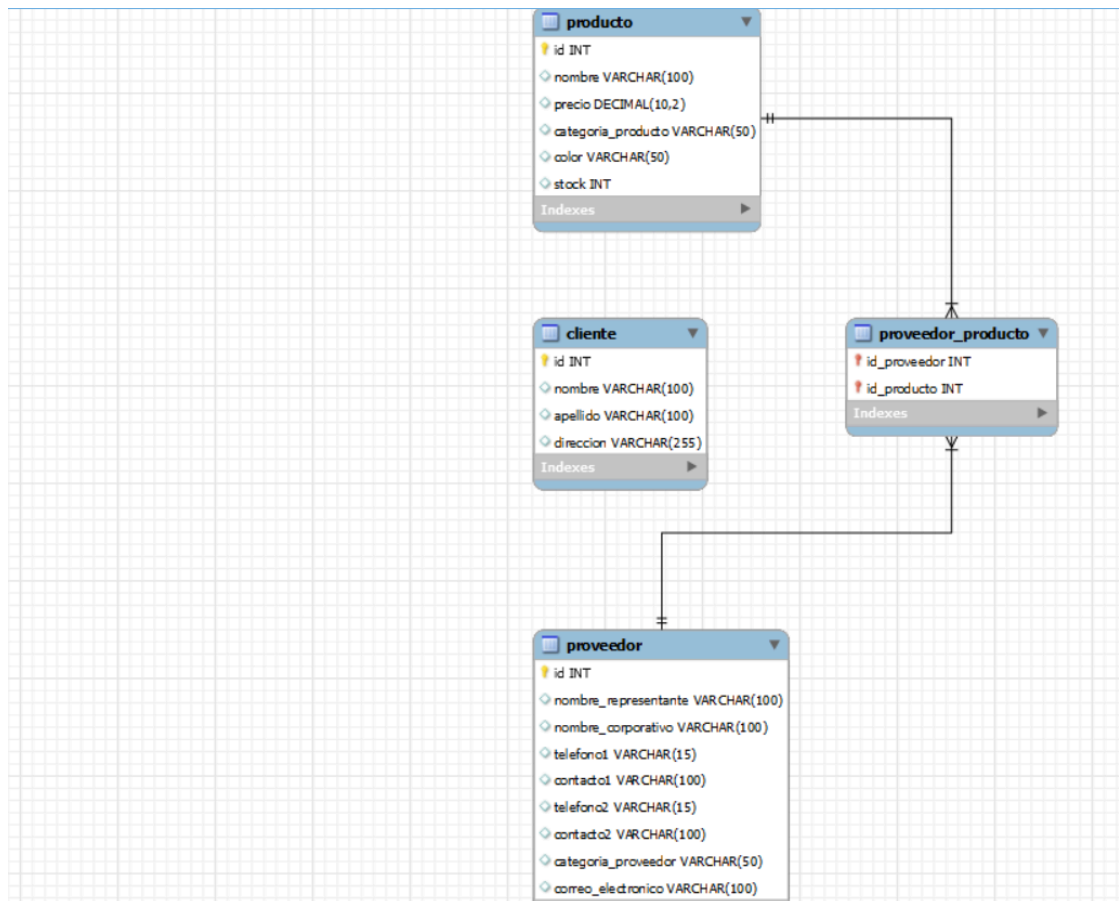
**-- 4. Proveedores con menor stock de productos**

```
SELECT p.nombre_corporativo, pr.nombre, pr.stock
FROM Proveedor p
JOIN Proveedor_Producto pp ON p.id = pp.id_proveedor
JOIN Producto pr ON pp.id_producto = pr.id
ORDER BY pr.stock ASC
LIMIT 1;
```

**-- Por último:**

**-- 5. Cambiar la categoría de productos más popular por 'Electrónica y computación'.**

```
UPDATE Producto
SET categoria_producto = 'Electrónica y Computación'
WHERE categoria_producto = (
    SELECT categoria_producto
    FROM (
        SELECT categoria_producto
        FROM Producto
        GROUP BY categoria_producto
        ORDER BY COUNT(*) DESC
        LIMIT 1
    ) AS categoria_max
);
```



(Modelo Entidad-Relacional actual)

**Deben subir el trabajo a un repositorio en Git-Hub.**

**A modo de entrega, se debe disponer un documento Word o PDF en el que se indique:**

**- Los nombres de los integrantes del equipo**

**# Proyecto TeLoVendo-Sprint**

**## Integrantes**

Ana Andrade

Matias Henriquez Torres

Jorge Lira

Bárbara Ramos

Lorena Suárez

**- Ruta del repositorio en GitHub**

**Consideraciones adicionales**

**- El código debe estar debidamente comentado**