SCRIPT COMPLETO - PAPELERIA

```
CREATE DATABASE papeleria;
GO
 USE papeleria;
  GO

TABLE CLIENTES
 CREATE PROCEDURE
SP-Crear-tabla-Clientes
 AS
 BEGIN
 IF NOT EXISTS (SELECT*FROM)
 WHERE nome= Clientes
 AND
  BEGIN
CREATE TABLE Clientes(
 ClienteID INT PRIMARY KEY IDENTITY,
Nombre VARCHAR (50) NOT NULL,
Apellido VARCHAR (50) NOT
 NULL
Telefono VARCHAR (20),
Email VARCHAR (100),
 UNIQUE
);
 END
GO
 Exec sp_Crear_tabla_Clientes ;

 TABLA PROVEEDORES

CREATE PROCEDUCE
SP-Crear-tabla-proveedores
 AS
BEGIN
 IF NOT EXIST (SELECT*FROM SYSOBJECTS WHERE 1;
FROM SYSOBJECTS WHERE
 name=proveedores' AND)
BEGIN
 CREATE TABLE Proveedores
 AS
  BEGIN
IF NOT EXISTS (SELECT* FROM SYSOBJECTS WHERE
name='Proveedores' AND)
 BEGIN
CREATE TABLE Proveedores
 AS
 BEGIN
 IF NOT EXISTS (SELECT* FROM SYSOBJECTS WHERE
name='Proveedores' AND)
BEGIN
 CREATE TABLE proveedores(
 proveedorID INT PRIMARY
KEY IDENTITY,
 Nombre VARCHAR (100) NOT NULL,
 Telefono VARCHAR (20),
 Direccion VARCHAR (100),
```

```
Email VARCHAR (100)
); UNIQUE
  END;
GO
  EXEC sp-Crear-tabla-Proveedores;

TABLE CATEGORIAS
  CREATE PROCEDURE
SP-Crea-tabla-Categoria
  AS
BEGIN
  If NOT EXISTS (SELECT* FROM SYSOBJECTS WHERE
nome='Categoria' AND BEGIN

CREATE TABLE categorias(
  categoriaID INT PRIMARY
KEY IDENTITY
  Nombre VARCHAR (50) NOT NULL,
  Descripcion VARCHAR (100)
);
  END;
GO
  Exec sp-Crear-tabla-categorias;

TABLE PRODUCTOS
  SP-Crear-tabla-productos
AS
  BEGIN
If NOT EXISTS (SELECT* FROM) WHERE name=Producto
  BEGIN
CREATE TABLE Productos(
  ProductoID INT PRIMARY KEY IDENTITY,
Nombre VARCHAR (100) NOT NULL,
  Precio DECIMAL (10,2)
CategoriaID
  FOREIGN KEY REFERENCES
Proveedores (ProveedorID)
);
  END;
GO
  EXEC sp-Crear_tablaProductos

TABLE VENTAS
  CREATE PROCEDURE
sp-Crear_tabla_Ventas
  VentaID INT PRIMARY KE,
ClienteID INT PRIMARY FROM
FOREIGN KEY REFERENCES
  FOREI KEY DATEFAULT
GETDATE()
  Total DECIMAL (10,2)
CHECK (Total DECIMAL >0)
);
  END;
GO
  EXEC sp-Crear_tabla_Ventas;
TABLE DETALLE VENTAS
```

```
CREATE PROCEDURE
SP-Crear_tabla_detalleVentas
AS
BEGIN
IF NOT EXISTS (SELECT * FROM

INSERT INTO Clientes (Nombre, Apellido, Telefono, Email) VALUES
('Carlos', Gomez 3101234567', Carlos@gmail.com'),
('Maria, Lopez, 3112345678' maria@gmail.com'),
('Juan, Martinez, 3123456789, juan@gmail.com'),
('Ana, Rodriguez 3135678901, ana@gmail.com'),
('Pedro 'Sanchez, 3145678901, pedro@gmail.com'),
('Laura, Torres 3156789012' laura@gmail.com'),
('Andres Ramirez ', 3168912 0, andres@gmail.com'),
('Sofia', Morales , 3178901234, sofia@gmail.com')
('Diego, Castro 3189013 95, diego@gmail.com'),
('Camila, Hernandez 3190123415, Camila@gmail.com');

INSERT INTO proveedores (Nombre, Telefono, Direccion, Email
('Cafe de Colombia 320111111 Cra 10 # 20-30 prove1@gmai.com')
('Postres S.A', 3102222223 Calle 15 # 5H-12 prove2@gmail.com')
('Panaderia el trigo 32033333 Cra 45 # 12-50' prove4@gmail.com'),
('Chocolates Ltda 320444444 Cra 9 # 4-2 ' prove8@gmail.com'),
('Leche Alpina 320555555 Cra 25 # 7-18 prove9@gmail.com'
('Jugos Naturale, 3205558910 Calle 8 # 12-40, prove8@gmail.com');

INSERT INTO categorias (Nombre, Descripcion) VALUES
('Bebidas calientes, Cafe, te chocole')
('Bebidas frias, jugos, Malteadas, refrescos')
('Postres Tortas, galletas, flanes')
('Panaderia, Pan, Croissants, arepas')
('Snacks, Galletas, Papas, manyi')
('Lacteos, leche, yogurt, queso')
('Helados, variedad de helados');

INSERT INTO Productos (Nombre, precio, Stock, categoria ID) VALUES
('Cafe Americano 500, 100, 1, 1)
('Capoccino 7000, 80, 1, 1)
('Te verde 4000, 50, 1, 1)
('Jugo de Naranja, 6000, 70)
('Malteada de chocolate 8000, 60, 2, 9)
('Torta de chocolate 12000, 40, 3, 2)
('Croissant, 5000, 50, 4, 3');
('Sandwich de Pollo, 10000, 30, 9, 3)

INSERT INTO VENTAS (ClienteID, EmpleadosID) VALUES
(1,1 2025-09-1-19000)
(2,2 2025-09-2-22000)
(3,3 2025-09-3-25000)
(4,4 2025-09-4-25000)
(5,5 2025-09-5-12000)
(6,6 2025-09-6-3000);
(7,7 2025-09-7-9000);

INSERT INTO Detalleventas (VentaID, producto_ID,) VALUES
(7,12, 8000),
(2,22, 7000),
(3,32, 4000)
```

```
(4, 4, 3, 6000),
(5, 5, 1) 8000),
(6, 6, 2) 12000)
(7, 7, 3) 1500);

CREATE VIEW vw_total_ventas_Cliente AS)
SELECT C.Nombre, C.Apellido, SUM (v.Total AS TotalGastado
FROM Clientes C
JOIN Ventas v ON C.ClienteID= V.ClienteID
GROUP By.Nombre C.Apellido;

CREATE VIEW vw_productos_mas_vendidos AS
SELECT P.Nombre, SUM (d.Cantidad) AS CantidadVendida
FROM DetalleVentas d
JOIN productos p ON d.productoID =P.ProductoID
GROUP BY P.Nombre
ORDER BY CantidadVendida DESC;

CREATE VIEW vw_ventas_por_ejemplo AS
SELECT E.Nombre, e.Apellido, SUM (v.total) AS Total vendido
FROM Ventas v
JOIN Empleados e ON v.EmpleadoID = e.EmpleadoID
GROUP BY. e.Nombre, e.Apellido;

CREATE VIEW vw_stock_bajo AS
SELECT Nombre, Stock, Precio
FROM productos
WHERE stock <30;

CREATE VIEW vw_clientes_frecuentes AS
SELECT C.Nombre, C.Apellido, COUNT (v.ventaID) AS Compras
FROM ventas v
JOIN clientes c. ON v.ClienteID AS COMPRAS
FROM Ventas v
JOIN Clientes c. ON v.ClienteID=cClienteID
GROUP BY C.Nombre, C. Apellido
ORDER BY Compras DESC;
```

SCRIPT BD - CAFETERIA

CREACIÓN DE BASE DE DATOS CAFETERIA

```
CREATE DATABASE cafeteria;
GO
USE cafeteria;
GO
```

TABLA CLIENTES

```
CREATE PROCEDURE
sp_crear_tabla_clientes
AS
BEGIN
    IF NOT EXISTS (SELECT *FROM
  sysobjects WHERE name='clientes'
AND xtype='U')
 BEGIN
      CREATE TABLE clientes (
              clienteID INT PRIMARY KEY
IDENTITY,
          Nombre VARCHAR (50) NOT
NULL,
          Apellido VARCHAR (50) NOT
NULL,
          Telefono VARCHAR (20),
          Email VARCHAR (100)
UNIQUE
      );
    END
 END
 GO
 EXEC sp_crear_tabla_clientes;
```

TABLA PROVEEDORES

```
CREATE PROCEDURE
 sp_crear_tabla_proveedores
AS
BEGIN
    IF NOT EXISTS (SELECT
   *FROM sysobjects WHERE
   name='proveedores' AND xtype='U')
    BEGIN
      CREATE TABLE Proveedores (
      ProveedorID INT PRIMARY
KEY IDENTITY,
          Nombre VARCHAR(100)
NOT NULL,
          Telefono VARCHAR (20),
          Dirección VARCHAR(100),
          Email VARCHAR (100)
UNIQUE
      );
    END
  END;
```

```
GO
EXEC sp_crear_tabla_proveedores;

TABLA CATEGORIAS

CREATE PROCEDURE
sp_crear_tabla_categorias
AS
BEGIN
  IF NOT EXISTS (SELECT
  *FROM sysobjects WHERE
  name='categorias' AND xtype='U')
    BEGIN
      CREATE TABLE categorias (
        categoriaID INT PRIMARY
KEY IDENTITY,
        Nombre VARCHAR (50) NOT
NULL,
        Descripcion VARCHAR (100)
      );
    END
END;
GO
EXEC sp_crear_tabla_categorias;

TABLA PRODUCTOS

CREATE PROCEDURE
sp_crear_tabla_productos
AS
BEGIN
  IF NOT EXISTS (SELECT
  *FROM sysobjects WHERE
  name='productos' AND xtype='U')
  BEGIN
    CREATE TABLE productos (
        ProductoID INT PRIMARY
KEY IDENTITY,
        Nombre VARCHAR (100)
NOT NULL,
        precio DECIMAL (10,2)
CHECK (precio>0),
        stock INT CHECK (stock>=0),
        CategoriaID INT
FOREIGN KEY REFERENCES
Categorias (CategoriaID),
        ProveedorID INT
FOREIGN KEY REFERENCES
Proveedores (ProveedorID)
      );
    END
END;
GO
EXEC sp_crear_tabla_productos;
```

```
CREATE PROCEDURE
sp-crear_tabla_detalleVentas
AS
 BEGIN
 If NOT EXISTS (SELECT*FROM

INSERT INTO Clientes (Nombre, Apellido, Telefono, Email) VALUES
('Carlos, Gomez 3101234567 ', Carlos @gmail.com'),
(Maria , Lopez, 3112345678' maria@gmail.com'),
(Juan, Martinez, 3123456789, juan@gmail.com'),
(Ana, Rodriguez 3135678901, ana@gmail.com'),
(Pedro 'Sanchez, 3145678901, pedro@gmail.com),
('Laura, Torres 3156789012' laura@gmail.com),
('Andres Ramirez ', 3168912 3, andres@gmail.com'),
('Sofia', Morales , 3178901234, sofia@gmail.com')
('Diego , Castro 3189013 95, diego@gmail.com'),
('Camila, Hernandez 3190123415, Camila@gmail.com');

INSERT INTO proveedores (Nombre, Telefono, Direccion, Email
(Cafe de Colombia 320111111 Cra 10 # 20-30 prove1@gmai.com),
(Postres S.A', 3102222223 Calle 15 # 5H 12 prove2@gmail.com'),
(Panaderia el trigo 32033333 Cra 45 # 12-50' prove4@gmail.com),
(Chocolates Ltda 320444444 Cra 9 # 4-2 ' prove8@gmail.com),
(Leche Alpina 320555555 Cra 25 # 7-18 prove9@gmail.com'
(Jugos Naturale, 3205558910 Calle 8 # 12-40, prove8@gmail.com);

INSERT INTO categorias (Nombre, Descripcion) VALUES
(Bebidas calientes, Cafe, te chocole)
(Bebidas Frias jugos Malteadas, refrescos)
(Postres Tortas , Galletas , flanes )
(Panaderia, Pan, Croissants , arepas )
(Snacks , Galletas , Papas , maní )
(Lacteos , leche, yogurt , queso),
(Helados, variedad de helados);

INSERT INTO productos (Nombre, precio, Stock, categoria ID) values
(Cafe Americano 500, 100, 1, 1)
(Capoccino 7000, 80, 1, 1)
(Te verde 4000, 50, 1, 1)
(Jugo de naranja , 6000, 70)
(Malteada de chocolate 8000, 60, 2, 9)
(Torta de chocolate 12000, 40, 3, 2)
(Croissant , 5000, 50, 4, 3),
(Sandwich de Pollo, 10000, 30, 9, 3),

INSERT INTO ventas (ClienteID, EmpleadosID) VALUES
(1,1 2025 -09 -1 - 19000)
(2,2 2025 - 09 - 2 - 27000)
(3,3 2025- 09 - 3 - 25000)
(4,4 2025-09-4 - 25000)
(5,5 2025 -09-5 - 12000)
(6,6 2025 -09-6 - 3000);
(7,7 2025 - 09 7 - 9000);

INSERT INTO Detalleventas (VentaID, producto ID,) VALUES
(1,12, 5000),
(2,22 ,7000),
(3,32, 4000),
```

```
(4, 4, 3, 6000),
(5, 5, 1, 8000),
(6, 6, 2, 12000),
(7, 7, 3, 15000);

CREATE VIEW vw_total_ventas_cliente AS
SELECT c.Nombre, c.Apellido, SUM(v.Total) AS TotalGastado
FROM Clientes c
JOIN Ventas v ON c.ClienteID = v.ClienteID
GROUP BY c.Nombre, c.Apellido;

CREATE VIEW vw_productos_mas_vendidos AS
SELECT p.Nombre, SUM(d.Cantidad) AS CantidadVendida
FROM DetalleVentas d
JOIN Productos p ON d.ProductoID = p.ProductoID
GROUP BY p.Nombre
ORDER BY CantidadVendida DESC;

CREATE VIEW vw_ventas_por_ejemplo AS
SELECT e.Nombre, e.Apellido, SUM(v.total) AS TotalVendido
FROM Ventas v
JOIN Empleados e ON v.EmpleadoID = e.EmpleadoID
GROUP BY e.Nombre, e.Apellido;

CREATE VIEW vw_stock_bajo AS
SELECT Nombre, Stock, Precio
FROM productos
WHERE Stock < 30;

CREATE VIEW vw_clientes_frecuentes AS
SELECT c.Nombre, c.Apellido, COUNT(v.VentaID) AS Compras
FROM ventas v
JOIN clientes c. ON v.ClienteID AS COMPRAS
FROM Ventas v
JOIN Clientes c. ON v.ClienteID=c.ClienteID
GROUP BY c.Nombre, c.Apellido
ORDER BY Compras DESC;
```