PUNTO DE VENTA

---------------------------------------------BD PUNTO DE VENTA -------------------------------------------------------------------

-- Crear tabla para Clientes

CREATE TABLE Clientes (

ClienteID NUMBER PRIMARY KEY,

Nombre VARCHAR2(100) NOT NULL,

Direccion VARCHAR2(200),

Telefono VARCHAR2(20),

Correo VARCHAR2(100),

Sexo CHAR(1) CHECK (Sexo IN ('M', 'F', 'O')),

NIT VARCHAR2(20),

CUI VARCHAR2(20),

SeguroMedico VARCHAR2(50),

NumeroPoliza VARCHAR2(50)

);

-- Crear tabla para Categorías de Productos

CREATE TABLE Categorias (

CategoriaID NUMBER PRIMARY KEY,

Nombre VARCHAR2(50) NOT NULL,

Descripcion VARCHAR2(200)

);

-- Crear tabla para Productos

CREATE TABLE Productos (

ProductoID NUMBER PRIMARY KEY,

Codigo VARCHAR2(20) UNIQUE NOT NULL,

Descripcion VARCHAR2(200) NOT NULL,

PrecioUnitario NUMBER(10,2) NOT NULL,

ImpuestoAplicable NUMBER(5,2) DEFAULT 0,

CategoriaID NUMBER,

FOREIGN KEY (CategoriaID) REFERENCES Categorias(CategoriaID)

);

-- Crear tabla para Productos Específicos (con número de serie)

CREATE TABLE ProductosEspecificos (

ProductoEspecificoID NUMBER PRIMARY KEY,

ProductoID NUMBER,

NumeroSerie VARCHAR2(50) UNIQUE NOT NULL,

EstadoVenta VARCHAR2(20) DEFAULT 'Disponible',

FOREIGN KEY (ProductoID) REFERENCES Productos(ProductoID)

);

-- Crear tabla para Formas de Pago

CREATE TABLE FormasPago (

FormaPagoID NUMBER PRIMARY KEY,

Nombre VARCHAR2(50) UNIQUE NOT NULL

);

-- Crear tabla para Ventas

CREATE TABLE Ventas (

VentaID NUMBER PRIMARY KEY,

ClienteID NUMBER,

FechaHora TIMESTAMP DEFAULT SYSTIMESTAMP,

FormaPagoID NUMBER,

NumeroFactura VARCHAR2(20) UNIQUE NOT NULL,

DescuentoTotal NUMBER(10,2) DEFAULT 0,

TotalVenta NUMBER(10,2) NOT NULL,

EstadoCredito VARCHAR2(20) DEFAULT 'Pagado',

FOREIGN KEY (ClienteID) REFERENCES Clientes(ClienteID),

FOREIGN KEY (FormaPagoID) REFERENCES FormasPago(FormaPagoID)

);

-- Crear tabla para Detalle de Ventas

CREATE TABLE DetalleVentas (

DetalleVentaID NUMBER PRIMARY KEY,

VentaID NUMBER,

ProductoID NUMBER,

ProductoEspecificoID NUMBER NULL,

Cantidad NUMBER NOT NULL,

PrecioUnitario NUMBER(10,2) NOT NULL,

Subtotal NUMBER(10,2) NOT NULL,

FOREIGN KEY (VentaID) REFERENCES Ventas(VentaID),

FOREIGN KEY (ProductoID) REFERENCES Productos(ProductoID),

FOREIGN KEY (ProductoEspecificoID) REFERENCES ProductosEspecificos(ProductoEspecificoID)

);

-- Crear tabla para Devoluciones

CREATE TABLE Devoluciones (

DevolucionID NUMBER PRIMARY KEY,

VentaID NUMBER,

FechaDevolucion DATE DEFAULT SYSDATE,

Motivo VARCHAR2(200),

FOREIGN KEY (VentaID) REFERENCES Ventas(VentaID)

);

-- Crear tabla para Detalle de Devoluciones

CREATE TABLE DetalledDevoluciones (

DetalleDevolucionID NUMBER PRIMARY KEY,

DevolucionID NUMBER,

ProductoID NUMBER,

ProductoEspecificoID NUMBER NULL,

Cantidad NUMBER NOT NULL,

FOREIGN KEY (DevolucionID) REFERENCES Devoluciones(DevolucionID),

FOREIGN KEY (ProductoID) REFERENCES Productos(ProductoID),

FOREIGN KEY (ProductoEspecificoID) REFERENCES ProductosEspecificos(ProductoEspecificoID)

);

-- Crear tabla para Bancos

CREATE TABLE Bancos (

BancoID NUMBER PRIMARY KEY,

Nombre VARCHAR2(100) UNIQUE NOT NULL

);

-- Crear tabla para Pagos

CREATE TABLE Pagos (

PagoID NUMBER PRIMARY KEY,

VentaID NUMBER,

FechaPago DATE DEFAULT SYSDATE,

FormaPagoID NUMBER,

MontoPagado NUMBER(10,2) NOT NULL,

NumeroReferencia VARCHAR2(50),

BancoID NUMBER,

FOREIGN KEY (VentaID) REFERENCES Ventas(VentaID),

FOREIGN KEY (FormaPagoID) REFERENCES FormasPago(FormaPagoID),

FOREIGN KEY (BancoID) REFERENCES Bancos(BancoID)

);

-- Crear tabla para Inventario

CREATE TABLE Inventario (

InventarioID NUMBER PRIMARY KEY,

ProductoID NUMBER,

CantidadInicial NUMBER DEFAULT 0,

CantidadVendida NUMBER DEFAULT 0,

CantidadRecibida NUMBER DEFAULT 0,

CantidadExistencia NUMBER DEFAULT 0,

FOREIGN KEY (ProductoID) REFERENCES Productos(ProductoID)

);

-- Crear secuencias para los IDs auto-incrementales

CREATE SEQUENCE seq\_cliente START WITH 1 INCREMENT BY 1;

CREATE SEQUENCE seq\_categoria START WITH 1 INCREMENT BY 1;

CREATE SEQUENCE seq\_producto START WITH 1 INCREMENT BY 1;

CREATE SEQUENCE seq\_producto\_especifico START WITH 1 INCREMENT BY 1;

CREATE SEQUENCE seq\_forma\_pago START WITH 1 INCREMENT BY 1;

CREATE SEQUENCE seq\_venta START WITH 1 INCREMENT BY 1;

CREATE SEQUENCE seq\_detalle\_venta START WITH 1 INCREMENT BY 1;

CREATE SEQUENCE seq\_devolucion START WITH 1 INCREMENT BY 1;

CREATE SEQUENCE seq\_detalle\_devolucion START WITH 1 INCREMENT BY 1;

CREATE SEQUENCE seq\_banco START WITH 1 INCREMENT BY 1;

CREATE SEQUENCE seq\_pago START WITH 1 INCREMENT BY 1;

CREATE SEQUENCE seq\_inventario START WITH 1 INCREMENT BY 1;

-- Crear trigger para actualizar el inventario después de una venta

CREATE OR REPLACE TRIGGER trg\_actualizar\_inventario\_venta

AFTER INSERT ON DetalleVentas

FOR EACH ROW

BEGIN

UPDATE Inventario

SET CantidadVendida = CantidadVendida + :NEW.Cantidad,

CantidadExistencia = CantidadExistencia - :NEW.Cantidad

WHERE ProductoID = :NEW.ProductoID;

END;

/

-- Crear trigger para actualizar el inventario después de una devolución

CREATE OR REPLACE TRIGGER trg\_actualizar\_inventario\_devolucion

AFTER INSERT ON DetalledDevoluciones

FOR EACH ROW

BEGIN

UPDATE Inventario

SET CantidadVendida = CantidadVendida - :NEW.Cantidad,

CantidadExistencia = CantidadExistencia + :NEW.Cantidad

WHERE ProductoID = :NEW.ProductoID;

END;

/

-- Crear procedimiento almacenado para realizar una venta

CREATE OR REPLACE PROCEDURE realizar\_venta(

p\_cliente\_id IN NUMBER,

p\_forma\_pago\_id IN NUMBER,

p\_numero\_factura IN VARCHAR2,

p\_descuento\_total IN NUMBER,

p\_productos IN VARCHAR2,

p\_cantidades IN VARCHAR2,

p\_venta\_id OUT NUMBER

)

AS

v\_total\_venta NUMBER := 0;

v\_producto\_id NUMBER;

v\_cantidad NUMBER;

v\_precio\_unitario NUMBER;

v\_subtotal NUMBER;

v\_producto\_especifico\_id NUMBER;

BEGIN

-- Insertar la venta

INSERT INTO Ventas (VentaID, ClienteID, FormaPagoID, NumeroFactura, DescuentoTotal, TotalVenta)

VALUES (seq\_venta.NEXTVAL, p\_cliente\_id, p\_forma\_pago\_id, p\_numero\_factura, p\_descuento\_total, 0)

RETURNING VentaID INTO p\_venta\_id;

-- Procesar cada producto

FOR i IN 1..REGEXP\_COUNT(p\_productos, ',') + 1 LOOP

v\_producto\_id := TO\_NUMBER(REGEXP\_SUBSTR(p\_productos, '[^,]+', 1, i));

v\_cantidad := TO\_NUMBER(REGEXP\_SUBSTR(p\_cantidades, '[^,]+', 1, i));

-- Obtener el precio unitario del producto

SELECT PrecioUnitario INTO v\_precio\_unitario

FROM Productos

WHERE ProductoID = v\_producto\_id;

v\_subtotal := v\_precio\_unitario \* v\_cantidad;

v\_total\_venta := v\_total\_venta + v\_subtotal;

-- Verificar si es un producto específico

BEGIN

SELECT ProductoEspecificoID INTO v\_producto\_especifico\_id

FROM ProductosEspecificos

WHERE ProductoID = v\_producto\_id AND EstadoVenta = 'Disponible'

AND ROWNUM = 1;

-- Actualizar el estado del producto específico

UPDATE ProductosEspecificos

SET EstadoVenta = 'Vendido'

WHERE ProductoEspecificoID = v\_producto\_especifico\_id;

EXCEPTION

WHEN NO\_DATA\_FOUND THEN

v\_producto\_especifico\_id := NULL;

END;

-- Insertar el detalle de la venta

INSERT INTO DetalleVentas (DetalleVentaID, VentaID, ProductoID, ProductoEspecificoID, Cantidad, PrecioUnitario, Subtotal)

VALUES (seq\_detalle\_venta.NEXTVAL, p\_venta\_id, v\_producto\_id, v\_producto\_especifico\_id, v\_cantidad, v\_precio\_unitario, v\_subtotal);

END LOOP;

-- Actualizar el total de la venta

UPDATE Ventas

SET TotalVenta = v\_total\_venta - p\_descuento\_total

WHERE VentaID = p\_venta\_id;

COMMIT;

EXCEPTION

WHEN OTHERS THEN

ROLLBACK;

RAISE;

END;

/

-- Crear función para calcular el total de ventas por período

CREATE OR REPLACE FUNCTION calcular\_total\_ventas(

p\_fecha\_inicio IN DATE,

p\_fecha\_fin IN DATE

) RETURN NUMBER

AS

v\_total\_ventas NUMBER;

BEGIN

SELECT SUM(TotalVenta)

INTO v\_total\_ventas

FROM Ventas

WHERE FechaHora BETWEEN p\_fecha\_inicio AND p\_fecha\_fin;

RETURN NVL(v\_total\_ventas, 0);

END;

/

-- Crear job para realizar copias de seguridad diarias

BEGIN

DBMS\_SCHEDULER.CREATE\_JOB (

job\_name => 'BACKUP\_DIARIO',

job\_type => 'PLSQL\_BLOCK',

job\_action => 'BEGIN DBMS\_BACKUP\_RESTORE.BACKUP(''FULL''); END;',

start\_date => SYSTIMESTAMP,

repeat\_interval => 'FREQ=DAILY; BYHOUR=2',

enabled => TRUE

);

END;

/

-- Insertar datos en la tabla Clientes

INSERT INTO Clientes (ClienteID, Nombre, Direccion, Telefono, Correo, Sexo, NIT, CUI, SeguroMedico, NumeroPoliza)

VALUES (seq\_cliente.NEXTVAL, 'Juan Pérez', 'Calle 1-23 Zona 1', '5555-1234', 'juan@email.com', 'M', '1234567-8', '1234567890101', 'Seguro A', 'POL-001');

INSERT INTO Clientes (ClienteID, Nombre, Direccion, Telefono, Correo, Sexo, NIT, CUI, SeguroMedico, NumeroPoliza)

VALUES (seq\_cliente.NEXTVAL, 'María García', 'Avenida 2-34 Zona 2', '5555-5678', 'maria@email.com', 'F', '2345678-9', '2345678901012', 'Seguro B', 'POL-002');

-- Insertar datos en la tabla Categorias

INSERT INTO Categorias (CategoriaID, Nombre, Descripcion)

VALUES (seq\_categoria.NEXTVAL, 'Electrónicos', 'Productos electrónicos y gadgets');

INSERT INTO Categorias (CategoriaID, Nombre, Descripcion)

VALUES (seq\_categoria.NEXTVAL, 'Ropa', 'Prendas de vestir y accesorios');

-- Insertar datos en la tabla Productos

INSERT INTO Productos (ProductoID, Codigo, Descripcion, PrecioUnitario, ImpuestoAplicable, CategoriaID)

VALUES (seq\_producto.NEXTVAL, 'PROD001', 'Smartphone XYZ', 5000, 12, 1);

INSERT INTO Productos (ProductoID, Codigo, Descripcion, PrecioUnitario, ImpuestoAplicable, CategoriaID)

VALUES (seq\_producto.NEXTVAL, 'PROD002', 'Camiseta Algodón', 150, 12, 2);

-- Insertar datos en la tabla ProductosEspecificos

INSERT INTO ProductosEspecificos (ProductoEspecificoID, ProductoID, NumeroSerie, EstadoVenta)

VALUES (seq\_producto\_especifico.NEXTVAL, 1, 'SER001XYZ', 'Disponible');

INSERT INTO ProductosEspecificos (ProductoEspecificoID, ProductoID, NumeroSerie, EstadoVenta)

VALUES (seq\_producto\_especifico.NEXTVAL, 1, 'SER002XYZ', 'Disponible');

-- Insertar datos en la tabla FormasPago

INSERT INTO FormasPago (FormaPagoID, Nombre)

VALUES (seq\_forma\_pago.NEXTVAL, 'Efectivo');

INSERT INTO FormasPago (FormaPagoID, Nombre)

VALUES (seq\_forma\_pago.NEXTVAL, 'Tarjeta de Crédito');

-- Insertar datos en la tabla Bancos

INSERT INTO Bancos (BancoID, Nombre)

VALUES (seq\_banco.NEXTVAL, 'Banco Nacional');

INSERT INTO Bancos (BancoID, Nombre)

VALUES (seq\_banco.NEXTVAL, 'Banco Internacional');

-- Insertar datos en la tabla Inventario

INSERT INTO Inventario (InventarioID, ProductoID, CantidadInicial, CantidadVendida, CantidadRecibida, CantidadExistencia)

VALUES (seq\_inventario.NEXTVAL, 1, 100, 0, 100, 100);

INSERT INTO Inventario (InventarioID, ProductoID, CantidadInicial, CantidadVendida, CantidadRecibida, CantidadExistencia)

VALUES (seq\_inventario.NEXTVAL, 2, 200, 0, 200, 200);

-- Realizar una venta

DECLARE

v\_venta\_id NUMBER;

BEGIN

realizar\_venta(

p\_cliente\_id => 1,

p\_forma\_pago\_id => 1,

p\_numero\_factura => 'FAC001',

p\_descuento\_total => 0,

p\_productos => '1,2',

p\_cantidades => '1,2',

p\_venta\_id => v\_venta\_id

);

END;

/

-- Insertar una devolución

DECLARE

v\_venta\_id NUMBER;

v\_devolucion\_id NUMBER;

BEGIN

-- Obtener el ID de la última venta

SELECT MAX(VentaID) INTO v\_venta\_id FROM Ventas;

-- Insertar la devolución

INSERT INTO Devoluciones (DevolucionID, VentaID, FechaDevolucion, Motivo)

VALUES (seq\_devolucion.NEXTVAL, v\_venta\_id, SYSDATE, 'Producto defectuoso')

RETURNING DevolucionID INTO v\_devolucion\_id;

-- Insertar el detalle de la devolución

INSERT INTO DetalledDevoluciones (DetalleDevolucionID, DevolucionID, ProductoID, ProductoEspecificoID, Cantidad)

VALUES (seq\_detalle\_devolucion.NEXTVAL, v\_devolucion\_id, 1, NULL, 1);

-- Actualizar el inventario

UPDATE Inventario

SET CantidadVendida = CantidadVendida - 1,

CantidadExistencia = CantidadExistencia + 1

WHERE ProductoID = 1;

COMMIT;

END;

/

-- Insertar un pago

INSERT INTO Pagos (PagoID, VentaID, FechaPago, FormaPagoID, MontoPagado, NumeroReferencia, BancoID)

VALUES (seq\_pago.NEXTVAL, 1, SYSDATE, 2, 5300, 'REF001', 1);

COMMIT;

-- 1. Producto más vendido (por cantidad)

SELECT

p.ProductoID,

p.Codigo,

p.Descripcion,

SUM(dv.Cantidad) AS TotalVendido

FROM

Productos p

JOIN DetalleVentas dv ON p.ProductoID = dv.ProductoID

GROUP BY

p.ProductoID, p.Codigo, p.Descripcion

ORDER BY

TotalVendido DESC

FETCH FIRST 1 ROW ONLY;

-- 2. Producto más vendido (por ingresos)

SELECT

p.ProductoID,

p.Codigo,

p.Descripcion,

SUM(dv.Subtotal) AS TotalIngresos

FROM

Productos p

JOIN DetalleVentas dv ON p.ProductoID = dv.ProductoID

GROUP BY

p.ProductoID, p.Codigo, p.Descripcion

ORDER BY

TotalIngresos DESC

FETCH FIRST 1 ROW ONLY;

-- 3. Día con más ventas

SELECT

TRUNC(v.FechaHora) AS Fecha,

COUNT(\*) AS NumeroVentas,

SUM(v.TotalVenta) AS TotalVentas

FROM

Ventas v

GROUP BY

TRUNC(v.FechaHora)

ORDER BY

TotalVentas DESC

FETCH FIRST 1 ROW ONLY;

-- 4. Cliente más frecuente

SELECT

c.ClienteID,

c.Nombre,

COUNT(\*) AS NumeroCompras,

SUM(v.TotalVenta) AS TotalGastado

FROM

Clientes c

JOIN Ventas v ON c.ClienteID = v.ClienteID

GROUP BY

c.ClienteID, c.Nombre

ORDER BY

NumeroCompras DESC

FETCH FIRST 1 ROW ONLY;

-- 5. Ventas por categoría de producto

SELECT

c.Nombre AS Categoria,

COUNT(\*) AS NumeroVentas,

SUM(dv.Subtotal) AS TotalVentas

FROM

Categorias c

JOIN Productos p ON c.CategoriaID = p.CategoriaID

JOIN DetalleVentas dv ON p.ProductoID = dv.ProductoID

GROUP BY

c.Nombre

ORDER BY

TotalVentas DESC;

-- 6. Tendencia de ventas mensuales

SELECT

TO\_CHAR(v.FechaHora, 'YYYY-MM') AS Mes,

COUNT(\*) AS NumeroVentas,

SUM(v.TotalVenta) AS TotalVentas

FROM

Ventas v

GROUP BY

TO\_CHAR(v.FechaHora, 'YYYY-MM')

ORDER BY

Mes;

-- 7. Productos con bajo stock (menos de 10 unidades)

SELECT

p.ProductoID,

p.Codigo,

p.Descripcion,

i.CantidadExistencia

FROM

Productos p

JOIN Inventario i ON p.ProductoID = i.ProductoID

WHERE

i.CantidadExistencia < 10

ORDER BY

i.CantidadExistencia;

-- 8. Top 5 productos más rentables

SELECT

p.ProductoID,

p.Codigo,

p.Descripcion,

SUM(dv.Subtotal - (dv.Cantidad \* p.PrecioUnitario)) AS Utilidad

FROM

Productos p

JOIN DetalleVentas dv ON p.ProductoID = dv.ProductoID

GROUP BY

p.ProductoID, p.Codigo, p.Descripcion

ORDER BY

Utilidad DESC

FETCH FIRST 5 ROWS ONLY;

-- 9. Efectividad de descuentos

SELECT

CASE

WHEN v.DescuentoTotal > 0 THEN 'Con Descuento'

ELSE 'Sin Descuento'

END AS TipoVenta,

COUNT(\*) AS NumeroVentas,

AVG(v.TotalVenta) AS PromedioVenta

FROM

Ventas v

GROUP BY

CASE

WHEN v.DescuentoTotal > 0 THEN 'Con Descuento'

ELSE 'Sin Descuento'

END;

-- 10. Análisis de devoluciones

SELECT

p.ProductoID,

p.Codigo,

p.Descripcion,

COUNT(\*) AS NumeroDevoluciones,

SUM(dd.Cantidad) AS CantidadDevuelta

FROM

Productos p

JOIN DetalledDevoluciones dd ON p.ProductoID = dd.ProductoID

JOIN Devoluciones d ON dd.DevolucionID = d.DevolucionID

GROUP BY

p.ProductoID, p.Codigo, p.Descripcion

ORDER BY

NumeroDevoluciones DESC;

-- 11. Ventas por forma de pago

SELECT

fp.Nombre AS FormaPago,

COUNT(\*) AS NumeroVentas,

SUM(v.TotalVenta) AS TotalVentas

FROM

FormasPago fp

JOIN Ventas v ON fp.FormaPagoID = v.FormaPagoID

GROUP BY

fp.Nombre

ORDER BY

TotalVentas DESC;

-- 12. Clientes que no han comprado en los últimos 3 meses

SELECT

c.ClienteID,

c.Nombre,

MAX(v.FechaHora) AS UltimaCompra

FROM

Clientes c

LEFT JOIN Ventas v ON c.ClienteID = v.ClienteID

GROUP BY

c.ClienteID, c.Nombre

HAVING

MAX(v.FechaHora) < ADD\_MONTHS(SYSDATE, -3) OR MAX(v.FechaHora) IS NULL

ORDER BY

UltimaCompra;

CONTROL DE CONTABILIDAD

-----------------------------------------BD CONTABILIDAD---------------------------------------

-- Tabla C\_Clientes

CREATE TABLE C\_Clientes (

id\_cliente INT PRIMARY KEY,

nombre VARCHAR(100),

direccion VARCHAR(200),

rfc VARCHAR(20),

correo\_electronico VARCHAR(100),

telefono VARCHAR(20)

);

INSERT INTO C\_Clientes (id\_cliente, nombre, direccion, rfc, correo\_electronico, telefono) VALUES

(1, 'Juan Pérez', 'Calle 123, Ciudad A', 'PERJ800101ABC', 'juan@email.com', '5551234567');

INSERT INTO C\_Clientes (id\_cliente, nombre, direccion, rfc, correo\_electronico, telefono) VALUES

(2, 'María García', 'Av. Principal 456, Ciudad B', 'GARM790305DEF', 'maria@email.com', '5559876543');

INSERT INTO C\_Clientes (id\_cliente, nombre, direccion, rfc, correo\_electronico, telefono) VALUES

(3, 'Carlos López', 'Plaza Central 789, Ciudad C', 'LOPC881210GHI', 'carlos@email.com', '5552468135');

INSERT INTO C\_Clientes (id\_cliente, nombre, direccion, rfc, correo\_electronico, telefono) VALUES

(4, 'Ana Martínez', 'Calle Secundaria 321, Ciudad D', 'MARA850715JKL', 'ana@email.com', '5553698741');

INSERT INTO C\_Clientes (id\_cliente, nombre, direccion, rfc, correo\_electronico, telefono) VALUES

(5, 'Roberto Sánchez', 'Av. Norte 654, Ciudad E', 'SANR900220MNO', 'roberto@email.com', '5557412589');

INSERT INTO C\_Clientes (id\_cliente, nombre, direccion, rfc, correo\_electronico, telefono) VALUES

(6, 'Laura Rodríguez', 'Calle Sur 987, Ciudad F', 'RODL870630PQR', 'laura@email.com', '5558529637');

INSERT INTO C\_Clientes (id\_cliente, nombre, direccion, rfc, correo\_electronico, telefono) VALUES

(7, 'Pedro Gómez', 'Av. Este 147, Ciudad G', 'GOMP910825STU', 'pedro@email.com', '5554569872');

INSERT INTO C\_Clientes (id\_cliente, nombre, direccion, rfc, correo\_electronico, telefono) VALUES

(8, 'Sofía Fernández', 'Calle Oeste 258, Ciudad H', 'FERS930112VWX', 'sofia@email.com', '5556987412');

INSERT INTO C\_Clientes (id\_cliente, nombre, direccion, rfc, correo\_electronico, telefono) VALUES

(9, 'Miguel Torres', 'Plaza Sur 369, Ciudad I', 'TOMM880707YZA', 'miguel@email.com', '5551597536');

INSERT INTO C\_Clientes (id\_cliente, nombre, direccion, rfc, correo\_electronico, telefono) VALUES

(10, 'Isabel Ramírez', 'Av. Central 753, Ciudad J', 'RAMI900418BCD', 'isabel@email.com', '5558524169');

-- Tabla C\_Productos

CREATE TABLE C\_Productos (

id\_producto INT PRIMARY KEY,

codigo\_producto VARCHAR(20),

descripcion VARCHAR(200),

precio\_compra DECIMAL(10, 2),

precio\_venta DECIMAL(10, 2),

impuesto DECIMAL(5, 2),

categoria VARCHAR(50),

proveedor VARCHAR(100),

stock\_minimo INT,

stock\_maximo INT

);

INSERT INTO C\_Productos (id\_producto, codigo\_producto, descripcion, precio\_compra, precio\_venta, impuesto, categoria, proveedor, stock\_minimo, stock\_maximo) VALUES

(1, 'PROD001', 'Laptop HP 15"', 8000.00, 12000.00, 16.00, 'Electrónicos', 'HP México', 5, 50);

INSERT INTO C\_Productos (id\_producto, codigo\_producto, descripcion, precio\_compra, precio\_venta, impuesto, categoria, proveedor, stock\_minimo, stock\_maximo) VALUES

(2, 'PROD002', 'Smartphone Samsung Galaxy', 5000.00, 8000.00, 16.00, 'Electrónicos', 'Samsung Electronics', 10, 100);

INSERT INTO C\_Productos (id\_producto, codigo\_producto, descripcion, precio\_compra, precio\_venta, impuesto, categoria, proveedor, stock\_minimo, stock\_maximo) VALUES

(3, 'PROD003', 'Impresora Epson L3150', 3000.00, 4500.00, 16.00, 'Electrónicos', 'Epson México', 3, 30);

INSERT INTO C\_Productos (id\_producto, codigo\_producto, descripcion, precio\_compra, precio\_venta, impuesto, categoria, proveedor, stock\_minimo, stock\_maximo) VALUES

(4, 'PROD004', 'Monitor LG 24"', 2000.00, 3500.00, 16.00, 'Electrónicos', 'LG Electronics', 5, 40);

INSERT INTO C\_Productos (id\_producto, codigo\_producto, descripcion, precio\_compra, precio\_venta, impuesto, categoria, proveedor, stock\_minimo, stock\_maximo) VALUES

(5, 'PROD005', 'Teclado mecánico Logitech', 800.00, 1500.00, 16.00, 'Accesorios', 'Logitech', 15, 100);

INSERT INTO C\_Productos (id\_producto, codigo\_producto, descripcion, precio\_compra, precio\_venta, impuesto, categoria, proveedor, stock\_minimo, stock\_maximo) VALUES

(6, 'PROD006', 'Mouse inalámbrico Microsoft', 300.00, 600.00, 16.00, 'Accesorios', 'Microsoft', 20, 150);

INSERT INTO C\_Productos (id\_producto, codigo\_producto, descripcion, precio\_compra, precio\_venta, impuesto, categoria, proveedor, stock\_minimo, stock\_maximo) VALUES

(7, 'PROD007', 'Disco duro externo 1TB', 1000.00, 1800.00, 16.00, 'Almacenamiento', 'Western Digital', 8, 60);

INSERT INTO C\_Productos (id\_producto, codigo\_producto, descripcion, precio\_compra, precio\_venta, impuesto, categoria, proveedor, stock\_minimo, stock\_maximo) VALUES

(8, 'PROD008', 'Memoria USB 64GB', 150.00, 300.00, 16.00, 'Almacenamiento', 'Kingston', 30, 200);

INSERT INTO C\_Productos (id\_producto, codigo\_producto, descripcion, precio\_compra, precio\_venta, impuesto, categoria, proveedor, stock\_minimo, stock\_maximo) VALUES

(9, 'PROD009', 'Audífonos Bluetooth Sony', 1200.00, 2000.00, 16.00, 'Audio', 'Sony', 10, 80);

INSERT INTO C\_Productos (id\_producto, codigo\_producto, descripcion, precio\_compra, precio\_venta, impuesto, categoria, proveedor, stock\_minimo, stock\_maximo) VALUES

(10, 'PROD010', 'Cámara web Logitech C920', 1500.00, 2500.00, 16.00, 'Accesorios', 'Logitech', 5, 40);

-- Tabla C\_Ventas

CREATE TABLE C\_Ventas (

id\_venta NUMBER(10) PRIMARY KEY,

fecha\_hora DATE,

numero\_factura VARCHAR2(20),

forma\_pago VARCHAR2(50),

descuento NUMBER(10, 2),

impuestos NUMBER(10, 2),

total NUMBER(10, 2),

id\_cliente NUMBER(10),

FOREIGN KEY (id\_cliente) REFERENCES C\_Clientes(id\_cliente)

);

INSERT INTO C\_Ventas (id\_venta, fecha\_hora, numero\_factura, forma\_pago, descuento, impuestos, total, id\_cliente) VALUES

(1, TO\_DATE('2024-09-13 10:30:00', 'YYYY-MM-DD HH24:MI:SS'), 'FAC001', 'Efectivo', 0.00, 1920.00, 13920.00, 1);

INSERT INTO C\_Ventas (id\_venta, fecha\_hora, numero\_factura, forma\_pago, descuento, impuestos, total, id\_cliente) VALUES

(2, TO\_DATE('2024-09-13 11:45:00', 'YYYY-MM-DD HH24:MI:SS'), 'FAC002', 'Tarjeta de crédito', 500.00, 1200.00, 8700.00, 2);

INSERT INTO C\_Ventas (id\_venta, fecha\_hora, numero\_factura, forma\_pago, descuento, impuestos, total, id\_cliente) VALUES

(3, TO\_DATE('2024-09-13 14:20:00', 'YYYY-MM-DD HH24:MI:SS'), 'FAC003', 'Transferencia', 0.00, 720.00, 5220.00, 3);

INSERT INTO C\_Ventas (id\_venta, fecha\_hora, numero\_factura, forma\_pago, descuento, impuestos, total, id\_cliente) VALUES

(4, TO\_DATE('2024-09-13 16:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'FAC004', 'Efectivo', 200.00, 528.00, 3828.00, 4);

INSERT INTO C\_Ventas (id\_venta, fecha\_hora, numero\_factura, forma\_pago, descuento, impuestos, total, id\_cliente) VALUES

(5, TO\_DATE('2024-09-14 09:15:00', 'YYYY-MM-DD HH24:MI:SS'), 'FAC005', 'Tarjeta de débito', 0.00, 240.00, 1740.00, 5);

INSERT INTO C\_Ventas (id\_venta, fecha\_hora, numero\_factura, forma\_pago, descuento, impuestos, total, id\_cliente) VALUES

(6, TO\_DATE('2024-09-14 12:30:00', 'YYYY-MM-DD HH24:MI:SS'), 'FAC006', 'PayPal', 100.00, 384.00, 2784.00, 6);

INSERT INTO C\_Ventas (id\_venta, fecha\_hora, numero\_factura, forma\_pago, descuento, impuestos, total, id\_cliente) VALUES

(7, TO\_DATE('2024-09-14 15:45:00', 'YYYY-MM-DD HH24:MI:SS'), 'FAC007', 'Efectivo', 0.00, 288.00, 2088.00, 7);

INSERT INTO C\_Ventas (id\_venta, fecha\_hora, numero\_factura, forma\_pago, descuento, impuestos, total, id\_cliente) VALUES

(8, TO\_DATE('2024-09-15 10:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'FAC008', 'Tarjeta de crédito', 300.00, 352.00, 2552.00, 8);

INSERT INTO C\_Ventas (id\_venta, fecha\_hora, numero\_factura, forma\_pago, descuento, impuestos, total, id\_cliente) VALUES

(9, TO\_DATE('2024-09-15 13:20:00', 'YYYY-MM-DD HH24:MI:SS'), 'FAC009', 'Transferencia', 0.00, 320.00, 2320.00, 9);

INSERT INTO C\_Ventas (id\_venta, fecha\_hora, numero\_factura, forma\_pago, descuento, impuestos, total, id\_cliente) VALUES

(10, TO\_DATE('2024-09-15 17:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'FAC010', 'Efectivo', 150.00, 376.00, 2726.00, 10);

-- Tabla C\_Compras

CREATE TABLE C\_Compras (

id\_compra INT PRIMARY KEY,

fecha\_compra DATE,

proveedor VARCHAR(100),

numero\_factura VARCHAR(20),

total DECIMAL(10, 2),

forma\_pago VARCHAR(50)

);

INSERT INTO C\_Compras (id\_compra, fecha\_compra, proveedor, numero\_factura, total, forma\_pago) VALUES

(1, TO\_DATE('2024-09-10', 'YYYY-MM-DD'), 'HP México', 'FACHP001', 80000.00, 'Transferencia');

INSERT INTO C\_Compras (id\_compra, fecha\_compra, proveedor, numero\_factura, total, forma\_pago) VALUES

(2, TO\_DATE('2024-09-10', 'YYYY-MM-DD'), 'Samsung Electronics', 'FACSAM001', 50000.00, 'Cheque');

INSERT INTO C\_Compras (id\_compra, fecha\_compra, proveedor, numero\_factura, total, forma\_pago) VALUES

(3, TO\_DATE('2024-09-11', 'YYYY-MM-DD'), 'Epson México', 'FACEP001', 30000.00, 'Tarjeta de crédito');

INSERT INTO C\_Compras (id\_compra, fecha\_compra, proveedor, numero\_factura, total, forma\_pago) VALUES

(4, TO\_DATE('2024-09-11', 'YYYY-MM-DD'), 'LG Electronics', 'FACLG001', 20000.00, 'Transferencia');

INSERT INTO C\_Compras (id\_compra, fecha\_compra, proveedor, numero\_factura, total, forma\_pago) VALUES

(5, TO\_DATE('2024-09-12', 'YYYY-MM-DD'), 'Logitech', 'FACLOG001', 12000.00, 'PayPal');

INSERT INTO C\_Compras (id\_compra, fecha\_compra, proveedor, numero\_factura, total, forma\_pago) VALUES

(6, TO\_DATE('2024-09-12', 'YYYY-MM-DD'), 'Microsoft', 'FACMSC001', 6000.00, 'Tarjeta de débito');

INSERT INTO C\_Compras (id\_compra, fecha\_compra, proveedor, numero\_factura, total, forma\_pago) VALUES

(7, TO\_DATE('2024-09-13', 'YYYY-MM-DD'), 'Western Digital', 'FACWD001', 10000.00, 'Transferencia');

INSERT INTO C\_Compras (id\_compra, fecha\_compra, proveedor, numero\_factura, total, forma\_pago) VALUES

(8, TO\_DATE('2024-09-13', 'YYYY-MM-DD'), 'Kingston', 'FACKNG001', 4500.00, 'Efectivo');

INSERT INTO C\_Compras (id\_compra, fecha\_compra, proveedor, numero\_factura, total, forma\_pago) VALUES

(9, TO\_DATE('2024-09-14', 'YYYY-MM-DD'), 'Sony', 'FACSNY001', 12000.00, 'Cheque');

INSERT INTO C\_Compras (id\_compra, fecha\_compra, proveedor, numero\_factura, total, forma\_pago) VALUES

(10, TO\_DATE('2024-09-14', 'YYYY-MM-DD'), 'Logitech', 'FACLOG002', 7500.00, 'Tarjeta de crédito');

-- Tabla C\_Gastos

CREATE TABLE C\_Gastos (

id\_gasto INT PRIMARY KEY,

fecha DATE,

concepto VARCHAR(100),

monto DECIMAL(10, 2),

categoria VARCHAR(50)

);

INSERT INTO C\_Gastos (id\_gasto, fecha, concepto, monto, categoria) VALUES

(1, TO\_DATE('2024-09-01', 'YYYY-MM-DD'), 'Renta de local', 15000.00, 'Alquiler');

INSERT INTO C\_Gastos (id\_gasto, fecha, concepto, monto, categoria) VALUES

(2, TO\_DATE('2024-09-05', 'YYYY-MM-DD'), 'Servicio de electricidad', 2500.00, 'Servicios');

INSERT INTO C\_Gastos (id\_gasto, fecha, concepto, monto, categoria) VALUES

(3, TO\_DATE('2024-09-05', 'YYYY-MM-DD'), 'Servicio de agua', 500.00, 'Servicios');

INSERT INTO C\_Gastos (id\_gasto, fecha, concepto, monto, categoria) VALUES

(4, TO\_DATE('2024-09-05', 'YYYY-MM-DD'), 'Internet y teléfono', 1200.00, 'Servicios');

INSERT INTO C\_Gastos (id\_gasto, fecha, concepto, monto, categoria) VALUES

(5, TO\_DATE('2024-09-10', 'YYYY-MM-DD'), 'Sueldos', 50000.00, 'Nómina');

INSERT INTO C\_Gastos (id\_gasto, fecha, concepto, monto, categoria) VALUES

(6, TO\_DATE('2024-09-15', 'YYYY-MM-DD'), 'Mantenimiento de equipos', 3000.00, 'Mantenimiento');

INSERT INTO C\_Gastos (id\_gasto, fecha, concepto, monto, categoria) VALUES

(7, TO\_DATE('2024-09-20', 'YYYY-MM-DD'), 'Material de oficina', 1500.00, 'Suministros');

INSERT INTO C\_Gastos (id\_gasto, fecha, concepto, monto, categoria) VALUES

(8, TO\_DATE('2024-09-25', 'YYYY-MM-DD'), 'Seguro del local', 2000.00, 'Seguros');

INSERT INTO C\_Gastos (id\_gasto, fecha, concepto, monto, categoria) VALUES

(9, TO\_DATE('2024-09-28', 'YYYY-MM-DD'), 'Publicidad en redes sociales', 1000.00, 'Marketing');

INSERT INTO C\_Gastos (id\_gasto, fecha, concepto, monto, categoria) VALUES

(10, TO\_DATE('2024-09-30', 'YYYY-MM-DD'), 'Gastos de limpieza', 800.00, 'Mantenimiento');

-- Tabla C\_Bancos

CREATE TABLE C\_Bancos (

id\_movimiento INT PRIMARY KEY,

fecha DATE,

tipo\_movimiento VARCHAR(20),

monto DECIMAL(10, 2),

concepto VARCHAR(100),

numero\_cuenta VARCHAR(20)

);

INSERT INTO C\_Bancos (id\_movimiento, fecha, tipo\_movimiento, monto, concepto, numero\_cuenta) VALUES

(1, TO\_DATE('2024-09-01', 'YYYY-MM-DD'), 'Depósito', 50000.00, 'Venta del día', '1234567890');

INSERT INTO C\_Bancos (id\_movimiento, fecha, tipo\_movimiento, monto, concepto, numero\_cuenta) VALUES

(2, TO\_DATE('2024-09-05', 'YYYY-MM-DD'), 'Retiro', 15000.00, 'Pago de renta', '1234567890');

INSERT INTO C\_Bancos (id\_movimiento, fecha, tipo\_movimiento, monto, concepto, numero\_cuenta) VALUES

(3, TO\_DATE('2024-09-10', 'YYYY-MM-DD'), 'Depósito', 30000.00, 'Venta del día', '1234567890');

INSERT INTO C\_Bancos (id\_movimiento, fecha, tipo\_movimiento, monto, concepto, numero\_cuenta) VALUES

(4, TO\_DATE('2024-09-15', 'YYYY-MM-DD'), 'Retiro', 50000.00, 'Pago de nómina', '1234567890');

INSERT INTO C\_Bancos (id\_movimiento, fecha, tipo\_movimiento, monto, concepto, numero\_cuenta) VALUES

(5, TO\_DATE('2024-09-20', 'YYYY-MM-DD'), 'Depósito', 40000.00, 'Venta del día', '1234567890');

INSERT INTO C\_Bancos (id\_movimiento, fecha, tipo\_movimiento, monto, concepto, numero\_cuenta) VALUES

(6, TO\_DATE('2024-09-25', 'YYYY-MM-DD'), 'Retiro', 5000.00, 'Pago de servicios', '1234567890');

INSERT INTO C\_Bancos (id\_movimiento, fecha, tipo\_movimiento, monto, concepto, numero\_cuenta) VALUES

(7, TO\_DATE('2024-09-26', 'YYYY-MM-DD'), 'Depósito', 20000.00, 'Venta del día', '1234567890');

INSERT INTO C\_Bancos (id\_movimiento, fecha, tipo\_movimiento, monto, concepto, numero\_cuenta) VALUES

(8, TO\_DATE('2024-09-27', 'YYYY-MM-DD'), 'Retiro', 3000.00, 'Compra de suministros', '1234567890');

INSERT INTO C\_Bancos (id\_movimiento, fecha, tipo\_movimiento, monto, concepto, numero\_cuenta) VALUES

(9, TO\_DATE('2024-09-28', 'YYYY-MM-DD'), 'Depósito', 35000.00, 'Venta del día', '1234567890');

INSERT INTO C\_Bancos (id\_movimiento, fecha, tipo\_movimiento, monto, concepto, numero\_cuenta) VALUES

(10, TO\_DATE('2024-09-30', 'YYYY-MM-DD'), 'Retiro', 2000.00, 'Pago de seguro', '1234567890');

-- Tabla C\_Inventarios

CREATE TABLE C\_Inventarios (

id\_inventario INT PRIMARY KEY,

id\_producto INT,

cantidad\_inicial INT,

cantidad\_vendida INT,

cantidad\_comprada INT,

cantidad\_existencia INT,

valor\_inventario DECIMAL(10, 2),

FOREIGN KEY (id\_producto) REFERENCES C\_Productos(id\_producto)

);

INSERT INTO C\_Inventarios (id\_inventario, id\_producto, cantidad\_inicial, cantidad\_vendida, cantidad\_comprada, cantidad\_existencia, valor\_inventario) VALUES

(1, 1, 20, 5, 10, 25, 200000.00);

INSERT INTO C\_Inventarios (id\_inventario, id\_producto, cantidad\_inicial, cantidad\_vendida, cantidad\_comprada, cantidad\_existencia, valor\_inventario) VALUES

(2, 2, 50, 15, 20, 55, 275000.00);

INSERT INTO C\_Inventarios (id\_inventario, id\_producto, cantidad\_inicial, cantidad\_vendida, cantidad\_comprada, cantidad\_existencia, valor\_inventario) VALUES

(3, 3, 15, 8, 10, 17, 51000.00);

INSERT INTO C\_Inventarios (id\_inventario, id\_producto, cantidad\_inicial, cantidad\_vendida, cantidad\_comprada, cantidad\_existencia, valor\_inventario) VALUES

(4, 4, 30, 12, 15, 33, 66000.00);

INSERT INTO C\_Inventarios (id\_inventario, id\_producto, cantidad\_inicial, cantidad\_vendida, cantidad\_comprada, cantidad\_existencia, valor\_inventario) VALUES

(5, 5, 100, 30, 50, 120, 96000.00);

INSERT INTO C\_Inventarios (id\_inventario, id\_producto, cantidad\_inicial, cantidad\_vendida, cantidad\_comprada, cantidad\_existencia, valor\_inventario) VALUES

(6, 6, 150, 40, 50, 160, 48000.00);

INSERT INTO C\_Inventarios (id\_inventario, id\_producto, cantidad\_inicial, cantidad\_vendida, cantidad\_comprada, cantidad\_existencia, valor\_inventario) VALUES

(7, 7, 40, 20, 30, 50, 50000.00);

INSERT INTO C\_Inventarios (id\_inventario, id\_producto, cantidad\_inicial, cantidad\_vendida, cantidad\_comprada, cantidad\_existencia, valor\_inventario) VALUES

(8, 8, 200, 80, 100, 220, 33000.00);

INSERT INTO C\_Inventarios (id\_inventario, id\_producto, cantidad\_inicial, cantidad\_vendida, cantidad\_comprada, cantidad\_existencia, valor\_inventario) VALUES

(9, 9, 60, 25, 30, 65, 78000.00);

INSERT INTO C\_Inventarios (id\_inventario, id\_producto, cantidad\_inicial, cantidad\_vendida, cantidad\_comprada, cantidad\_existencia, valor\_inventario) VALUES

(10, 10, 25, 10, 15, 30, 45000.00);

-- Tabla C\_CuentasPorCobrar

CREATE TABLE C\_CuentasPorCobrar (

id\_cuenta\_cobrar INT PRIMARY KEY,

id\_cliente INT,

id\_venta INT,

monto\_pendiente DECIMAL(10, 2),

fecha\_vencimiento DATE,

estado VARCHAR(20),

FOREIGN KEY (id\_cliente) REFERENCES C\_Clientes(id\_cliente),

FOREIGN KEY (id\_venta) REFERENCES C\_Ventas(id\_venta)

);

INSERT INTO C\_CuentasPorCobrar (id\_cuenta\_cobrar, id\_cliente, id\_venta, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(1, 1, 1, 5000.00, TO\_DATE('2024-10-13', 'YYYY-MM-DD'), 'Pendiente');

INSERT INTO C\_CuentasPorCobrar (id\_cuenta\_cobrar, id\_cliente, id\_venta, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(2, 2, 2, 3000.00, TO\_DATE('2024-10-13', 'YYYY-MM-DD'), 'Pendiente');

INSERT INTO C\_CuentasPorCobrar (id\_cuenta\_cobrar, id\_cliente, id\_venta, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(3, 3, 3, 2000.00, TO\_DATE('2024-10-13', 'YYYY-MM-DD'), 'Pendiente');

INSERT INTO C\_CuentasPorCobrar (id\_cuenta\_cobrar, id\_cliente, id\_venta, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(4, 4, 4, 1500.00, TO\_DATE('2024-10-13', 'YYYY-MM-DD'), 'Pendiente');

INSERT INTO C\_CuentasPorCobrar (id\_cuenta\_cobrar, id\_cliente, id\_venta, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(5, 5, 5, 1000.00, TO\_DATE('2024-10-14', 'YYYY-MM-DD'), 'Pendiente');

INSERT INTO C\_CuentasPorCobrar (id\_cuenta\_cobrar, id\_cliente, id\_venta, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(6, 6, 6, 1200.00, TO\_DATE('2024-10-14', 'YYYY-MM-DD'), 'Pendiente');

INSERT INTO C\_CuentasPorCobrar (id\_cuenta\_cobrar, id\_cliente, id\_venta, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(7, 7, 7, 800.00, TO\_DATE('2024-10-14', 'YYYY-MM-DD'), 'Pendiente');

INSERT INTO C\_CuentasPorCobrar (id\_cuenta\_cobrar, id\_cliente, id\_venta, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(8, 8, 8, 1300.00, TO\_DATE('2024-10-15', 'YYYY-MM-DD'), 'Pendiente');

INSERT INTO C\_CuentasPorCobrar (id\_cuenta\_cobrar, id\_cliente, id\_venta, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(9, 9, 9, 1100.00, TO\_DATE('2024-10-15', 'YYYY-MM-DD'), 'Pendiente');

INSERT INTO C\_CuentasPorCobrar (id\_cuenta\_cobrar, id\_cliente, id\_venta, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(10, 10, 10, 900.00, TO\_DATE('2024-10-15', 'YYYY-MM-DD'), 'Pendiente');

-- Tabla C\_CuentasPorPagar

CREATE TABLE C\_CuentasPorPagar (

id\_cuenta\_pagar INT PRIMARY KEY,

id\_compra INT,

proveedor VARCHAR(100),

monto\_pendiente DECIMAL(10, 2),

fecha\_vencimiento DATE,

estado VARCHAR(20),

FOREIGN KEY (id\_compra) REFERENCES C\_Compras(id\_compra)

);

INSERT INTO C\_CuentasPorPagar (id\_cuenta\_pagar, id\_compra, proveedor, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(1, 1, 'HP México', 40000.00, TO\_DATE('2024-10-10', 'YYYY-MM-DD'), 'Pendiente');

INSERT INTO C\_CuentasPorPagar (id\_cuenta\_pagar, id\_compra, proveedor, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(2, 2, 'Samsung Electronics', 25000.00, TO\_DATE('2024-10-10', 'YYYY-MM-DD'), 'Pendiente');

INSERT INTO C\_CuentasPorPagar (id\_cuenta\_pagar, id\_compra, proveedor, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(3, 3, 'Epson México', 15000.00, TO\_DATE('2024-10-11', 'YYYY-MM-DD'), 'Pendiente');

INSERT INTO C\_CuentasPorPagar (id\_cuenta\_pagar, id\_compra, proveedor, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(4, 4, 'LG Electronics', 10000.00, TO\_DATE('2024-10-11', 'YYYY-MM-DD'), 'Pendiente');

INSERT INTO C\_CuentasPorPagar (id\_cuenta\_pagar, id\_compra, proveedor, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(5, 5, 'Logitech', 6000.00, TO\_DATE('2024-10-12', 'YYYY-MM-DD'), 'Pendiente');

INSERT INTO C\_CuentasPorPagar (id\_cuenta\_pagar, id\_compra, proveedor, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(6, 6, 'Microsoft', 3000.00, TO\_DATE('2024-10-12', 'YYYY-MM-DD'), 'Pendiente');

INSERT INTO C\_CuentasPorPagar (id\_cuenta\_pagar, id\_compra, proveedor, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(7, 7, 'Western Digital', 5000.00, TO\_DATE('2024-10-13', 'YYYY-MM-DD'), 'Pendiente');

INSERT INTO C\_CuentasPorPagar (id\_cuenta\_pagar, id\_compra, proveedor, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(8, 8, 'Kingston', 2250.00, TO\_DATE('2024-10-13', 'YYYY-MM-DD'), 'Pendiente');

INSERT INTO C\_CuentasPorPagar (id\_cuenta\_pagar, id\_compra, proveedor, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(9, 9, 'Sony', 6000.00, TO\_DATE('2024-10-14', 'YYYY-MM-DD'), 'Pendiente');

INSERT INTO C\_CuentasPorPagar (id\_cuenta\_pagar, id\_compra, proveedor, monto\_pendiente, fecha\_vencimiento, estado) VALUES

(10, 10, 'Logitech', 3750.00, TO\_DATE('2024-10-14', 'YYYY-MM-DD'), 'Pendiente');