UNIVERSIDAD AUTÓNOMA DEL ESTADO DE HIDALGO

INSTITUTO DE CIENCIAS BASICAS E INGENIERÍA

LIC. EN CIENCIAS COMPUTACIONALES

BASES DE DATOS DISTRIBUIDAS

Consultas a la BD de Distribuidora

Catedrático: Dr. Eduardo Cornejo Velázquez

David Arael Vargas Alarcón

Semestre: 6

Grupo: 2

Diseñar las sentencias de álgebra relacional y sentencias SQL para las siguientes consultas:

- Reporte de compras del mes de enero que incluya el nombre del proveedor, el nombre de la herramienta, cantidad, precio unitario y precio total

```
SELECT
    s.name AS SupplierName,
    t.name AS ToolName,
    bt.amount AS Quantity,
    bt.unitPrice AS UnitPrice,
    (bt.amount * bt.unitPrice) AS TotalPrice
FROM
    buyTool bt
JOIN purchase p ON bt.idPurchase = p.idPurchase
JOIN supplier s ON p.idSupplier = s.idSupplier
JOIN inventory i ON bt.idInventory = i.idInventory
JOIN tool t ON i.idTool = t.idTool
WHERE
    MONTH(p.date) = 1;
```

- Reporte de inventario de la bodega de la calle 13 que incluya el nombre de la herramienta, cantidad y costo total (precio de compra * cantidad)

```
SELECT
    t.name AS ToolName,
    i.amount AS Quantity,
    (i.amount * i.purchasePrice) AS TotalCost
FROM
    inventory i
JOIN warehouse w ON i.idWarehouse = w.idWarehouse
JOIN tool t ON i.idTool = t.idTool
WHERE
    w.street = 'Calle 13';
```

- Reporte de compras de la herramienta "Tijera de jardinero" durante el año 2023 el nombre del proveedor, fecha de compra, cantidad, precio unitario y costo total

```
SELECT
    s.name AS SupplierName,
    p.date AS PurchaseDate,
    bt.amount AS Quantity,
    bt.unitPrice AS UnitPrice,
    (bt.amount * bt.unitPrice) AS TotalCost
FROM
    buyTool bt
JOIN purchase p ON bt.idPurchase = p.idPurchase
JOIN supplier s ON p.idSupplier = s.idSupplier
JOIN inventory i ON bt.idInventory = i.idInventory
JOIN tool t ON i.idTool = t.idTool
```

```
    WHERE
      t.name = 'Tijera de jardinero' AND YEAR(p.date) = 2023;
```

- Listado de responsables de las bodegas de la empresa con calle, número y teléfono

```
    SELECT

      w.manager AS Responsible,

      w.street AS Street,

      w.number AS Number,

      w.phone AS Phone

    FROM

      warehouse w;
```

- Listado de contactos con los proveedores con nombre de contacto, nombre de proveedor, teléfono y correo electrónico

```
    SELECT
      s.contact AS ContactName,
      s.name AS SupplierName,
      s.phone AS Phone,
      s.email AS Email
    FROM
      supplier s;
```

- Reporte de herramientas compradas cuyo precio unitario se menor o igual a $250.00 que incluya nombre de la herramienta, fecha de compra y cantidad comprada en orden cronológico descendiente.

```
    SELECT
      t.name AS ToolName,
      p.date AS PurchaseDate,
      bt.amount AS Quantity
    FROM
      buyTool bt
    JOIN purchase p ON bt.idPurchase = p.idPurchase
    JOIN inventory i ON bt.idInventory = i.idInventory
    JOIN tool t ON i.idTool = t.idTool
    WHERE
      bt.unitPrice <= 250
    ORDER BY
  p.date DESC;
```

- Reporte de herramientas en el inventario cuyo stock sea entre 5 y 20 piezas que incluya calle y número de la bodega, nombre de la herramienta, ubicación y cantidad en existencia.

```
SELECT
    w.street AS WarehouseStreet,
    w.number AS WarehouseNumber,
    t.name AS ToolName,
    i.location AS Location,
    i.amount AS Quantity
FROM
    inventory i
JOIN warehouse w ON i.idWarehouse = w.idWarehouse
JOIN tool t ON i.idTool = t.idTool
WHERE
    i.amount BETWEEN 5 AND 20;
```

- Reporte del stock de todas las bodegas que incluya calle, número, responsable, teléfono y total de herramientas almacenadas

```
SELECT
    w.street AS Street,
    w.number AS Number,
    w.manager AS Responsible,
    w.phone AS Phone,
    SUM(i.amount) AS TotalStock
FROM
    inventory i
JOIN warehouse w ON i.idWarehouse = w.idWarehouse
GROUP BY
    w.idWarehouse;
```

- Reporte de valor de inventario de todas las bodegas que incluya calle, estado y monto total de precio de venta de las herramientas almacenadas

```
SELECT
    w.street AS Street,
    w.state AS State,
    SUM(i.storeSalePrice * i.amount) AS TotalInventoryValue
FROM
    inventory i
JOIN warehouse w ON i.idWarehouse = w.idWarehouse
GROUP BY
    w.idWarehouse;
```

SENTENCIAS Y CÓDIGO SQL

```
-- Crear la base de datos
CREATE DATABASE IF NOT EXISTS HardwareStore;
USE HardwareStore;


-- Tabla Supplier
CREATE TABLE supplier (
    idSupplier INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    street VARCHAR(100),
    number VARCHAR(10),
    city VARCHAR(50),
    state VARCHAR(50),
    phone VARCHAR(20),
    email VARCHAR(100),
```

```
    contact VARCHAR(100)
);

-- Tabla Purchase
CREATE TABLE purchase (
    idPurchase INT AUTO_INCREMENT PRIMARY KEY,
    idSupplier INT,
    date DATE,
    time TIME,
    folio VARCHAR(50),
    payment VARCHAR(50),
    FOREIGN KEY (idSupplier) REFERENCES supplier(idSupplier)
    ON DELETE SET NULL ON UPDATE CASCADE
);

-- Tabla Tool
CREATE TABLE tool (
    idTool INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100),
    model VARCHAR(50),
    manufacturer VARCHAR(100),
    color VARCHAR(30),
    size VARCHAR(30),
    description TEXT
);

-- Tabla Warehouse
```

```
CREATE TABLE warehouse (
    idWarehouse INT AUTO_INCREMENT PRIMARY KEY,
    street VARCHAR(100),
    number VARCHAR(10),
    city VARCHAR(50),
    state VARCHAR(50),
    area DECIMAL(10, 2),
    phone VARCHAR(20),
    manager VARCHAR(100)
);

-- Tabla Inventory
CREATE TABLE inventory (
    idInventory INT AUTO_INCREMENT PRIMARY KEY,
    idWarehouse INT,
    idTool INT,
    zone VARCHAR(50),
    rack VARCHAR(50),
    level VARCHAR(50),
    location VARCHAR(50),
    amount INT UNSIGNED,
    purchasePrice DECIMAL(10, 2),
    storeSalePrice DECIMAL(10, 2),
    wholeSalePrice DECIMAL(10, 2),
    FOREIGN KEY (idWarehouse) REFERENCES warehouse(idWarehouse)
    ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (idTool) REFERENCES tool(idTool)
```

```
    ON DELETE CASCADE ON UPDATE CASCADE
);


-- Tabla BuyTool
CREATE TABLE buyTool (
    idBuyTool INT AUTO_INCREMENT PRIMARY KEY,
    idPurchase INT,
    idInventory INT,
    amount INT,
    unitPrice DECIMAL(10, 2),
    note TEXT,
    FOREIGN KEY (idPurchase) REFERENCES purchase(idPurchase)
    ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (idInventory) REFERENCES inventory(idInventory)
    ON DELETE CASCADE ON UPDATE CASCADE
);
```