

REPORTE DE PRÁCTICA NO. 3.1

NOMBRE DE LA PRÁCTICA

ALUMNO:

KEVIN AXEL CHAVEZ QUIROZ



1. Introducción

En el ámbito de la informática y la gestión de datos, los conceptos relacionados con las bases de datos y su manejo son fundamentales para el desarrollo de sistemas eficientes y seguros. Este reporte de práctica se centra en la creación y manipulación de una base de datos utilizando MySQL, abarcando no solo la implementación técnica, sino también los conceptos teóricos que sustentan dichas operaciones.

El análisis de las relaciones entre entidades, la identificación de claves, y las técnicas de fragmentación, entre otros, son aspectos cruciales que se deben comprender para diseñar y mantener bases de datos robustas. A través de esta práctica, se buscará no solo aplicar estos conceptos, sino también documentar el proceso de diseño y ejecución de las consultas necesarias para satisfacer requerimientos específicos.

La introducción de índices y vistas en el contexto de la base de datos permitirá optimizar el acceso y la gestión de la información, mejorando tanto el rendimiento como la facilidad de uso del sistema. A lo largo de este documento, se proporcionarán ejemplos prácticos y explicaciones detalladas para asegurar una comprensión completa de los procedimientos y técnicas empleadas.

2. Marco teórico

Grado de la Relación: El grado de una relación es el número de entidades que participan en una relación. Puede ser binario (dos entidades), ternario (tres entidades) o n-ario (n entidades).

Clave Candidata: Una clave candidata es un conjunto de atributos que puede ser utilizado como una clave primaria debido a que es única y no nula. Cada tabla puede tener múltiples claves candidatas.

Clave Primaria: La clave primaria es una clave candidata seleccionada para identificar de manera única las tuplas en una tabla. No puede contener valores nulos y debe ser única.

Superclave: Una superclave es cualquier combinación de atributos que puede ser utilizada para identificar de manera única una tupla en una tabla. Puede contener atributos adicionales que no sean necesarios para la unicidad.

Clave Foránea: Una clave foránea es un conjunto de uno o más atributos en una tabla que hace referencia a la clave primaria de otra tabla. Se utiliza para mantener la integridad referencial entre tablas.

Cardinalidad: La cardinalidad define la cantidad de entidades que pueden estar asociadas con otra entidad en una relación. Puede ser uno a uno, uno a muchos, o muchos a muchos.

Fragmentación Horizontal: La fragmentación horizontal divide una tabla en subconjuntos de tuplas, donde cada subconjunto contiene un número específico de filas que cumplen ciertas condiciones.

Fragmentación Vertical: La fragmentación vertical divide una tabla en subconjuntos de columnas, donde cada subconjunto contiene un número específico de columnas de la tabla original.

Índice (Index): Un índice es una estructura de datos que mejora la velocidad de recuperación de datos en una tabla a expensas de espacio de almacenamiento adicional y mayor tiempo en operaciones de escritura.

Create View: CREATE VIEW es una instrucción SQL que se utiliza para crear una vista. Una vista es una tabla virtual basada en el resultado de una consulta SQL que permite simplificar consultas complejas.

3. Sentencias SQL para Crear Fragmentos

Fragmentación Horizontal:

Crear fragmento horizontal de inventory donde la zona es 'Zone A':

```
CREATE TABLE inventory-zone-a AS SELECT * FROM inventory WHERE zone = 'Zone A';
```

Fragmentación Vertical:

Crear fragmento vertical de inventory con columnas específicas:

```
CREATE TABLE inventory_price_info AS SELECT idInventory, purchasePrice, storeSalePrice, wholeSalePrice FROM inventory;
```

Índice:

Crear un índice en la columna **name** de la tabla **tool**:

```
CREATE INDEX idx_tool_name ON tool(name);
```

Vista:

Crear una vista que muestre el nombre de la herramienta y el nombre del proveedor:

```
CREATE VIEW tool_supplier_view AS SELECT t.name AS tool_name, s.name AS supplier_name FROM
tool t JOIN inventory i ON t.idTool = i.idTool JOIN buyTool bt ON i.idInventory = bt.idInventory JOIN
purchase p ON bt.idPurchase = p.idPurchase JOIN supplier s ON p.idSupplier = s.idSupplier;
```

3. Código de la base de datos

```
CREATE DATABASE Distribuidora; USE Distribuidora;
```

```
CREATE TABLE supplier
```

```
( idSupplier INT PRIMARY KEY, name VARCHAR(100), street VARCHAR(100), number INT, city VARCHAR(50), state VARCHAR(50), phone VARCHAR(15), email VARCHAR(100), contact VARCHAR(100) );
```

```
CREATE TABLE purchase
```

```
( idPurchase INT PRIMARY KEY, idSupplier INT, date DATE, time TIME, folio VARCHAR(50), payment VARCHAR(50), FOREIGN KEY (idSupplier) REFERENCES supplier(idSupplier) );
```

```
CREATE TABLE tool
```

```
( idTool INT PRIMARY KEY, name VARCHAR(100), model VARCHAR(50), manufacturer VARCHAR(50), color VARCHAR(20), size VARCHAR(20), description TEXT );
```

```
CREATE TABLE warehouse
```

```
( idWarehouse INT PRIMARY KEY, street VARCHAR(100), number INT, city VARCHAR(50), state VARCHAR(50), area FLOAT, phone VARCHAR(15), manager VARCHAR(100) );
```

```
CREATE TABLE inventory
```

```
( idInventory INT PRIMARY KEY, idWarehouse INT, idTool INT, zone VARCHAR(50), rack VARCHAR(50), level INT, location VARCHAR(100), amount INT, purchasePrice DECIMAL(10, 2), storeSalePrice DECIMAL(10, 2), wholeSalePrice DECIMAL(10, 2), FOREIGN KEY (idWarehouse) REFERENCES warehouse(idWarehouse), FOREIGN KEY (idTool) REFERENCES tool(idTool) );
```

CREATE TABLE buyTool

(idBuyTool INT PRIMARY KEY, idPurchase INT, idInventory INT, amount INT, unitPrice DECIMAL(10, 2), note TEXT,
FOREIGN KEY (idPurchase) REFERENCES purchase(idPurchase),
FOREIGN KEY (idInventory) REFERENCES inventory(idInventory));

Insertar 5 registros en la tabla supplier

INSERT INTO supplier VALUES

(1, 'Supplier A', 'Street A', 10, 'City A', 'State A', '1234567890', 'a@example.com', 'Contact A'),
(2, 'Supplier B', 'Street B', 20, 'City B', 'State B', '0987654321', 'b@example.com', 'Contact B'),
(3, 'Supplier C', 'Street C', 30, 'City C', 'State C', '1122334455', 'c@example.com', 'Contact C'),
(4, 'Supplier D', 'Street D', 40, 'City D', 'State D', '5566778899', 'd@example.com', 'Contact D'),
(5, 'Supplier E', 'Street E', 50, 'City E', 'State E', '6677889900', 'e@example.com', 'Contact E');

Insertar 5 registros en la tabla tool

INSERT INTO tool VALUES

(1, 'Hammer', 'Model A', 'Manufacturer A', 'Red', 'Large', 'A strong hammer.'),
(2, 'Screwdriver', 'Model B', 'Manufacturer B', 'Blue', 'Medium', 'A useful screwdriver.'),
(3, 'Wrench', 'Model C', 'Manufacturer C', 'Yellow', 'Small', 'An adjustable wrench.'),
(4, 'Pliers', 'Model D', 'Manufacturer D', 'Green', 'Large', 'Durable pliers.'),
(5, 'Drill', 'Model E', 'Manufacturer E', 'Black', 'Medium', 'A powerful drill.');

Insertar 5 registros en la tabla warehouse

INSERT INTO warehouse VALUES

(1, 'Warehouse Street A', 100, 'Warehouse City A', 'Warehouse State A', 500.0, '1234567890', 'Manager A'),
(2, 'Warehouse Street B', 200, 'Warehouse City B', 'Warehouse State B', 600.0, '0987654321', 'Manager B'),
(3, 'Warehouse Street C', 300, 'Warehouse City C', 'Warehouse State C', 700.0, '1122334455', 'Manager C'),
(4, 'Warehouse Street D', 400, 'Warehouse City D', 'Warehouse State D', 800.0, '5566778899', 'Manager D'),
(5, 'Warehouse Street E', 500, 'Warehouse City E', 'Warehouse State E', 900.0, '6677889900', 'Manager E');

Insertar 10 registros en la tabla purchase

INSERT INTO purchase VALUES

(1, 1, '2024-01-01', '10:00:00', 'FOLIO001', 'Credit'),
(2, 2, '2024-02-01', '11:00:00', 'FOLIO002', 'Cash'),
(3, 3, '2024-03-01', '12:00:00', 'FOLIO003', 'Credit'),
(4, 4, '2024-04-01', '13:00:00', 'FOLIO004', 'Cash'),
(5, 5, '2024-05-01', '14:00:00', 'FOLIO005', 'Credit'),
(6, 1, '2024-06-01', '15:00:00', 'FOLIO006', 'Cash'),
(7, 2, '2024-07-01', '16:00:00', 'FOLIO007', 'Credit'),
(8, 3, '2024-08-01', '17:00:00', 'FOLIO008', 'Cash'),
(9, 4, '2024-09-01', '18:00:00', 'FOLIO009', 'Credit'),
(10, 5, '2024-10-01', '19:00:00', 'FOLIO010', 'Cash');

Insertar 25 registros en la tabla inventory

```
INSERT INTO inventory (idInventory, idWarehouse, idTool, zone, rack, level, location, amount, purchasePrice, storeSalePrice, wholeSalePrice) VALUES
```

```
(1, 1, 1, 'Zone A', 'Rack 1', 1, 'Location 1', 10, 100.00, 150.00, 140.00),
(2, 2, 2, 'Zone B', 'Rack 2', 2, 'Location 2', 20, 200.00, 250.00, 240.00),
(3, 3, 3, 'Zone C', 'Rack 3', 3, 'Location 3', 30, 300.00, 350.00, 340.00),
(4, 4, 4, 'Zone D', 'Rack 4', 4, 'Location 4', 40, 400.00, 450.00, 440.00),
(5, 5, 5, 'Zone E', 'Rack 5', 5, 'Location 5', 50, 500.00, 550.00, 540.00),
(6, 1, 2, 'Zone F', 'Rack 6', 1, 'Location 6', 15, 110.00, 160.00, 150.00),
(7, 2, 3, 'Zone G', 'Rack 7', 2, 'Location 7', 25, 210.00, 260.00, 250.00),
(8, 3, 4, 'Zone H', 'Rack 8', 3, 'Location 8', 35, 310.00, 360.00, 350.00),
(9, 4, 5, 'Zone I', 'Rack 9', 4, 'Location 9', 45, 410.00, 460.00, 450.00),
(10, 5, 1, 'Zone J', 'Rack 10', 5, 'Location 10', 55, 510.00, 560.00, 550.00),
(11, 1, 3, 'Zone K', 'Rack 11', 1, 'Location 11', 12, 105.00, 155.00, 145.00),
(12, 2, 4, 'Zone L', 'Rack 12', 2, 'Location 12', 22, 205.00, 255.00, 245.00),
(13, 3, 5, 'Zone M', 'Rack 13', 3, 'Location 13', 32, 305.00, 355.00, 345.00),
(14, 4, 1, 'Zone N', 'Rack 14', 4, 'Location 14', 42, 405.00, 455.00, 445.00),
(15, 5, 2, 'Zone O', 'Rack 15', 5, 'Location 15', 52, 505.00, 555.00, 545.00),
(16, 1, 4, 'Zone P', 'Rack 16', 1, 'Location 16', 17, 115.00, 165.00, 155.00),
(17, 2, 5, 'Zone Q', 'Rack 17', 2, 'Location 17', 27, 215.00, 265.00, 255.00),
(18, 3, 1, 'Zone R', 'Rack 18', 3, 'Location 18', 37, 315.00, 365.00, 355.00),
(19, 4, 2, 'Zone S', 'Rack 19', 4, 'Location 19', 47, 415.00, 465.00, 455.00),
(20, 5, 3, 'Zone T', 'Rack 20', 5, 'Location 20', 57, 515.00, 565.00, 555.00),
(21, 1, 5, 'Zone U', 'Rack 21', 1, 'Location 21', 19, 120.00, 170.00, 160.00),
(22, 2, 1, 'Zone V', 'Rack 22', 2, 'Location 22', 29, 220.00, 270.00, 260.00),
(23, 3, 2, 'Zone W', 'Rack 23', 3, 'Location 23', 39, 320.00, 370.00, 360.00),
(24, 4, 3, 'Zone X', 'Rack 24', 4, 'Location 24', 49, 420.00, 470.00, 460.00),
(25, 5, 4, 'Zone Y', 'Rack 25', 5, 'Location 25', 59, 520.00, 570.00, 560.00);
```

Insertar 50 registros en la tabla buyTool

```
INSERT INTO buyTool (idBuyTool, idPurchase, idInventory, amount, unitPrice, note) VALUES
```

```
(1, 1, 1, 10, 10.00, 'First note'),
(2, 1, 2, 20, 20.00, 'Second note'),
(3, 2, 3, 15, 15.00, 'Third note'),
(4, 2, 4, 25, 25.00, 'Fourth note'),
(5, 3, 5, 30, 30.00, 'Fifth note'),
(6, 3, 6, 10, 10.00, 'Sixth note'),
(7, 4, 7, 20, 20.00, 'Seventh note'),
(8, 4, 8, 15, 15.00, 'Eighth note'),
(9, 5, 9, 25, 25.00, 'Ninth note'),
(10, 5, 10, 30, 30.00, 'Tenth note'),
(11, 6, 11, 10, 10.00, 'Eleventh note'),
(12, 6, 12, 20, 20.00, 'Twelfth note'),
(13, 7, 13, 15, 15.00, 'Thirteenth note'),
(14, 7, 14, 25, 25.00, 'Fourteenth note'),
(15, 8, 15, 30, 30.00, 'Fifteenth note'),
(16, 8, 16, 10, 10.00, 'Sixteenth note'),
(17, 9, 17, 20, 20.00, 'Seventeenth note'),
(18, 9, 18, 15, 15.00, 'Eighteenth note'),
(19, 10, 19, 25, 25.00, 'Nineteenth note'),
```

(20, 10, 20, 30, 30.00, 'Twentieth note'),
(21, 1, 21, 10, 10.00, 'Twenty-first note'),
(22, 2, 22, 20, 20.00, 'Twenty-second note'),
(23, 3, 23, 15, 15.00, 'Twenty-third note'),
(24, 4, 24, 25, 25.00, 'Twenty-fourth note'),
(25, 5, 25, 30, 30.00, 'Twenty-fifth note'),
(26, 6, 1, 10, 10.00, 'Twenty-sixth note'),
(27, 7, 2, 20, 20.00, 'Twenty-seventh note'),
(28, 8, 3, 15, 15.00, 'Twenty-eighth note'),
(29, 9, 4, 25, 25.00, 'Twenty-ninth note'),
(30, 10, 5, 30, 30.00, 'Thirtieth note'),
(31, 1, 6, 10, 10.00, 'Thirty-first note'),
(32, 2, 7, 20, 20.00, 'Thirty-second note'),
(33, 3, 8, 15, 15.00, 'Thirty-third note'),
(34, 4, 9, 25, 25.00, 'Thirty-fourth note'),
(35, 5, 10, 30, 30.00, 'Thirty-fifth note'),
(36, 6, 11, 10, 10.00, 'Thirty-sixth note'),
(37, 7, 12, 20, 20.00, 'Thirty-seventh note'),
(38, 8, 13, 15, 15.00, 'Thirty-eighth note'),
(39, 9, 14, 25, 25.00, 'Thirty-ninth note'),
(40, 10, 15, 30, 30.00, 'Fortieth note'),
(41, 1, 16, 10, 10.00, 'Forty-first note'),
(42, 2, 17, 20, 20.00, 'Forty-second note'),
(43, 3, 18, 15, 15.00, 'Forty-third note'),
(44, 4, 19, 25, 25.00, 'Forty-fourth note'),
(45, 5, 20, 30, 30.00, 'Forty-fifth note'),
(46, 6, 21, 10, 10.00, 'Forty-sixth note'),
(47, 7, 22, 20, 20.00, 'Forty-seventh note'),
(48, 8, 23, 15, 15.00, 'Forty-eighth note'),
(49, 9, 24, 25, 25.00, 'Forty-ninth note'),
(50, 10, 25, 30, 30.00, 'Fiftieth note');

5. Conclusiones

A lo largo de este ejercicio, hemos explorado diversos conceptos fundamentales en el diseño y manejo de bases de datos, aplicando tanto conocimientos teóricos como prácticos. Desde la creación y población de tablas en MySQL, hasta la implementación de consultas en álgebra relacional y SQL, cada paso ha sido esencial para comprender cómo estructurar y optimizar una base de datos eficazmente.

La práctica de fragmentación de datos, tanto horizontal como vertical, así como la creación de índices y vistas, nos ha permitido mejorar la eficiencia y el rendimiento de nuestras consultas. Además, el diseño de reportes específicos para diferentes necesidades empresariales destaca la importancia de saber extraer y presentar datos de manera clara y precisa.

Esta experiencia no solo ha reforzado la comprensión de los conceptos clave como la cardinalidad, las claves primaria y foránea, y la relación entre entidades, sino que también ha subrayado la importancia de la integridad referencial y la optimización de consultas para el manejo de grandes volúmenes de datos.

En resumen, esta práctica proporciona una base sólida para el desarrollo y mantenimiento de bases de datos robustas y eficientes, preparándonos para enfrentar desafíos reales en el ámbito de la tecnología y los sistemas de información.

Referencias Bibliográficas

Grado de la relación, clave candidata, clave primaria, superclave, clave foránea, cardinalidad: García-Molina, H., Ullman, J. D., & Widom, J. (2008). *Database Systems: The Complete Book* (2nd ed.). Pearson Education.

Fragmentación horizontal, fragmentación vertical: Özsu, M. T., & Valduriez, P. (2011). *Principles of Distributed Database Systems* (3rd ed.). Springer.

Index: Elmasri, R., Navathe, S. B. (2016). *Fundamentals of Database Systems* (7th ed.). Pearson Education.

Create View: Silberschatz, A., Korth, H. F., & Sudarshan, S. (2019). *Database System Concepts* (7th ed.). McGraw-Hill Education. Consultas SQL y álgebra relacional: Date, C. J. (2019). *An Introduction to Database Systems* (8th ed.). Addison-Wesley.