



MINI PROJECT

E-Commerce Management System

A report submitted to the
Department of Electrical and Information Engineering
Faculty of Engineering
University of Ruhuna
Sri Lanka

On 13th of September 2025
In completing an assignment for the module
EE4203/EC4203 Database Systems

By Group No: 51
EG/2022/4905: AASIF A.K.M.
EG/2022/5226: NUSAIR M.N.M.
EG/2022/5261: RAJAAEI M.N.I.
EG/2022/5343: SHARAZ R.A.W.
EG/2022/5367: THARUKSHAN G.

TABLE OF CONTENTS

PART A – RELATIONAL DATABASES

CHAPTER 1 - REQUIREMENT ANALYSIS	6
1.1 Introduction.....	6
1.2 Functional Requirements	6
1.3 Data Requirements.....	8
CHAPTER 2 - CONCEPTUAL DESIGN (ER DIAGRAM).....	10
2.1 ER Diagram.....	10
2.2 Relations in the ER Diagram.....	11
2.2.1 Strong Entities	11
CHAPTER 3 – LOGICAL DESIGN AND NORMALIZATION.....	14
3.2 Functional Dependencies	18
3.3 Normalization to Second Normal Form (2NF)	18
CHAPTER 4 – IMPLEMENTATION.....	19
4.1 Schema Creation	19
4.2 Table Creation and Definition.....	19
4.3 Data Insertion (INSERT operations).....	25
4.4 Data Modification (UPDATE Operations)	35
4.5 Data Deletion (DELETE Operations)	36
CHAPTER 5 – TRANSACTIONS.....	38
5.1 Simple Queries	38
5.1.1 Select Operation	38
5.1.2 Project Operation.....	38
5.1.3 Cartesian Product.....	39
5.1.4 Create View	39
5.1.5 Rename Operation	40
5.1.6 Aggregate Function	40
5.1.7 LIKE Keyword	41
5.2 Complex Queries.....	42
5.2.1 Set Operations (Union, Intersection, Set Difference, Division)	42
5.2.2 Join Operations (Inner, Natural, Left Outer, Right Outer, Full Outer).....	44
5.2.3 Nested Queries.....	46
CHAPTER 6 - DATABASE TUNING	48
6.1 Query Tuning Methodology.....	48

6.2 Analysis of Original Queries.....	48
6.3 Tuned Queries and Index Implementation	48
6.4 Performance Comparison.....	53
PART B - NOSQL DATABASES	
Chapter 1: Aggregation model.....	54
1.1 Bucket Creation and Data Insertion	54
1.2 Query Operations on the Aggregate.....	55
Chapter 2: Graph data Model.....	56
2.1 Graph Database Creation	56
2.2 Querying the Social Network.....	59

LIST OF FIGURES

Figure 1 : ER Diagram for E-Commerce Management System	10
Figure 2 Creating Database.....	19
Figure 3 Creating Customer Table	19
Figure 4 Creating Category Table	20
Figure 5 Creating Product Table	20
Figure 6 Creating ProductImage Table	21
Figure 7 Creating Address Table.....	21
Figure 8 Creating ShoppingCart Table.....	22
Figure 9 Creating CartItem Table.....	22
Figure 10 Creating Order Table	23
Figure 11 Creating OrderItem Table.....	23
Figure 12 Creating Payment Table	24
Figure 13 Inserting Data to the Customer Table	25
Figure 14 Customer Table with Inserted Data.....	25
Figure 15 Inserting Data to the Category Table.....	26
Figure 16 Category Table with Inserted Data.....	26
Figure 17 Inserting Data to the Product Table.....	27
Figure 18 Product Table with Inserted Data	27
Figure 19 Inserting Data to the ProductImage Table.....	28
Figure 20 ProductImage Table with Inserted Data.....	28
Figure 21 Inserting Data to the Address Table	29
Figure 22 Address Table with Inserted Data	29
Figure 23 Inserting Data to the ShoppingCart Table	30
Figure 24 ShoppingCart Table with Inserted Data	30
Figure 25 Inserting Data to the CartItem Table	31
Figure 26 CartItem Table with Inserted Data	31
Figure 27 Inserting Data to the Order Table	32
Figure 28 Order Table with Inserted Data	32
Figure 29 Inserting Data to the OrderItem Table.....	33
Figure 30 OrderItem Table with Inserted Data.....	33
Figure 31 Inserting Data to the Payment Table	34
Figure 32 Payment Table with Inserted Data	34
Figure 33 Update a Product's Price	35
Figure 34 Update an Order's Status	35
Figure 35 Delete from CartItem Table	36
Figure 36 Verify the first Update operation	36
Figure 37 Verify the second Update operation	37
Figure 38 Verify the Delete operation.....	37
Figure 39 Select all columns from Customer Table	38
Figure 40 Select specific columns from Customer Table	38
Figure 41 Cartesian product between Category and Product Tables.....	39
Figure 42 View for active products with their categories.....	39
Figure 43 Rename Operation in Customer Table.....	40
Figure 44 Aggregate Functions in Product Table.....	40
Figure 45 Use of LIKE in Product Table.....	41
Figure 46 Union Operation	42

Figure 47 Intersection Operation	42
Figure 48 Set Difference Operation.....	43
Figure 49 Division Operation.....	43
Figure 50 Inner Join Operation.....	44
Figure 51 Left Outer Join Operation	44
Figure 52 Right Outer Join Operation	45
Figure 53 Full Outer Join Operation.....	45
Figure 54 Outer Union Operation	46
Figure 55 Nested Query with IN.....	46
Figure 56 Nested Query with ANY	47
Figure 57 Correlated Nested Query.....	47
Figure 58 Before tuned and after tuned for query 1	48
Figure 59 Before tuned and after tuned for query 2	49
Figure 60 Before tuned and after tuned for query 3	49
Figure 61 Before tuned and after tuned for query 4	50
Figure 62 Before tuned and after tuned for query 5	50
Figure 63 Before tuned and after tuned for query 6	51
Figure 64 Before tuned and after tuned for query 7	51
Figure 65 Before tuned and after tuned for query 8	52
Figure 66 Before tuned and after tuned for query 9	52
Figure 67 Before tuned and after tuned for query 10.....	53
Figure 68 Creation and insertion into table.....	54
Figure 69 First Query.....	55
Figure 70 Second Query.....	55
Figure 71 Created Database	56
Figure 72 Creating Nodes	57
Figure 73 Created 10 nodes	57
Figure 74 Creating 15 Edges (Relationships)	58
Figure 75 Created 15 Edges (Relationship)	58
Figure 76 Graph	59
Figure 77 Query to retrieve people who are friends with Alice	59
Figure 78 Output of 1st Query.....	60
Figure 79 Query to retrieve path from Alice to Julia.....	60
Figure 80 Output of second Query	61

PART A – RELATIONAL DATABASES

CHAPTER 1 - REQUIREMENT ANALYSIS

1.1 Introduction

A Database Management System, or DBMS, is a fundamental computer software program for designing, managing, and accessing databases efficiently. A DBMS provides a structured interface to define a database schema, insert and update data, and extract key information useful in making business decisions. E-commerce websites are the finest examples of systems that are completely reliant on high-performance DBMS in today's digital world to handle enormous inventories, customer details, and massive volumes of transactions. It would be unpractical and error-prone to manage this complexity without a DBMS.

This project entails the design and implementation of a database for an E-Commerce Management System. This system will serve as the backbone for an online store, maintaining data about products, customers, orders, and payments. It will enable secure registration of users, efficient product cataloguing, seamless order processing, and guaranteed payment tracking. Therefore, this database is an essential resource for business owners, administrators, and customers to access a single source of truth regarding all e-commerce activity and to make retrieving data-driven insights effortless.

1.2 Functional Requirements

The E-Commerce Management System shall incorporate the following characteristics

- The system shall facilitate new customers to sign up and create a personal account.
- The system shall facilitate registered customers to log in securely to their accounts.
- The system shall facilitate customers to view, edit, and maintain their personal profile and shipping addresses.
- The system shall facilitate administrators to add, edit, and remove product listings from the catalogue.
- The system will allow customers to browse products by product categories and search for products by product name or product description.
- The system will allow customers to place products into an online shopping cart, update item quantities, and remove items.
- The system will allow customers to check out and have the shopping cart contents transferred into an official order.
- The system will debit payments against orders and change the status of the order after successful payment.

- The system will manage product stock, decreasing automatically when a successful order is completed.
- The system will allow customers to view their ordering history and the status of all orders.
- The system will allow user views to generate user-specific reports of sales levels and popularity of products.

1.3 Data Requirements

The system must store and manage the following core data entities and their attributes to support the functional requirements

Entity: Customer - Attributes

- Customer ID
- Email
- Password Hash
- First Name
- Last Name
- Phone
- Registration Date

Entity: Address - Attributes

- Address ID
- Customer ID
- Address Line 1
- Address Line 2
- City
- State
- Zip Code
- Address Type

Entity: Category - Attributes

- Category ID
- Category Name
- Description
- Parent Category ID

Entity: Product - Attributes

- Product ID
- Product Name
- Description
- Price
- Stock Quantity
- Weight
- Date Added
- Status
- Category ID

Entity: ProductImage - Attributes

- Product ID
- Image URL

Entity: ShoppingCart - Attributes

- Cart ID
- Creation Date
- Customer ID

Entity: CartItem - Attributes

- Cart ID
- Product ID
- Quantity

Entity: Order - Attributes

- Order ID
- Order Date
- Total Amount
- Status
- Customer ID

Entity: OrderItem - Attributes

- Order ID
- Product ID
- Quantity
- Unit Price

Entity: Payment - Attributes

- Payment ID
- Payment Method
- Amount
- Transaction Date
- Status
- Order ID

CHAPTER 2 - CONCEPTUAL DESIGN (ER DIAGRAM)

2.1 ER Diagram

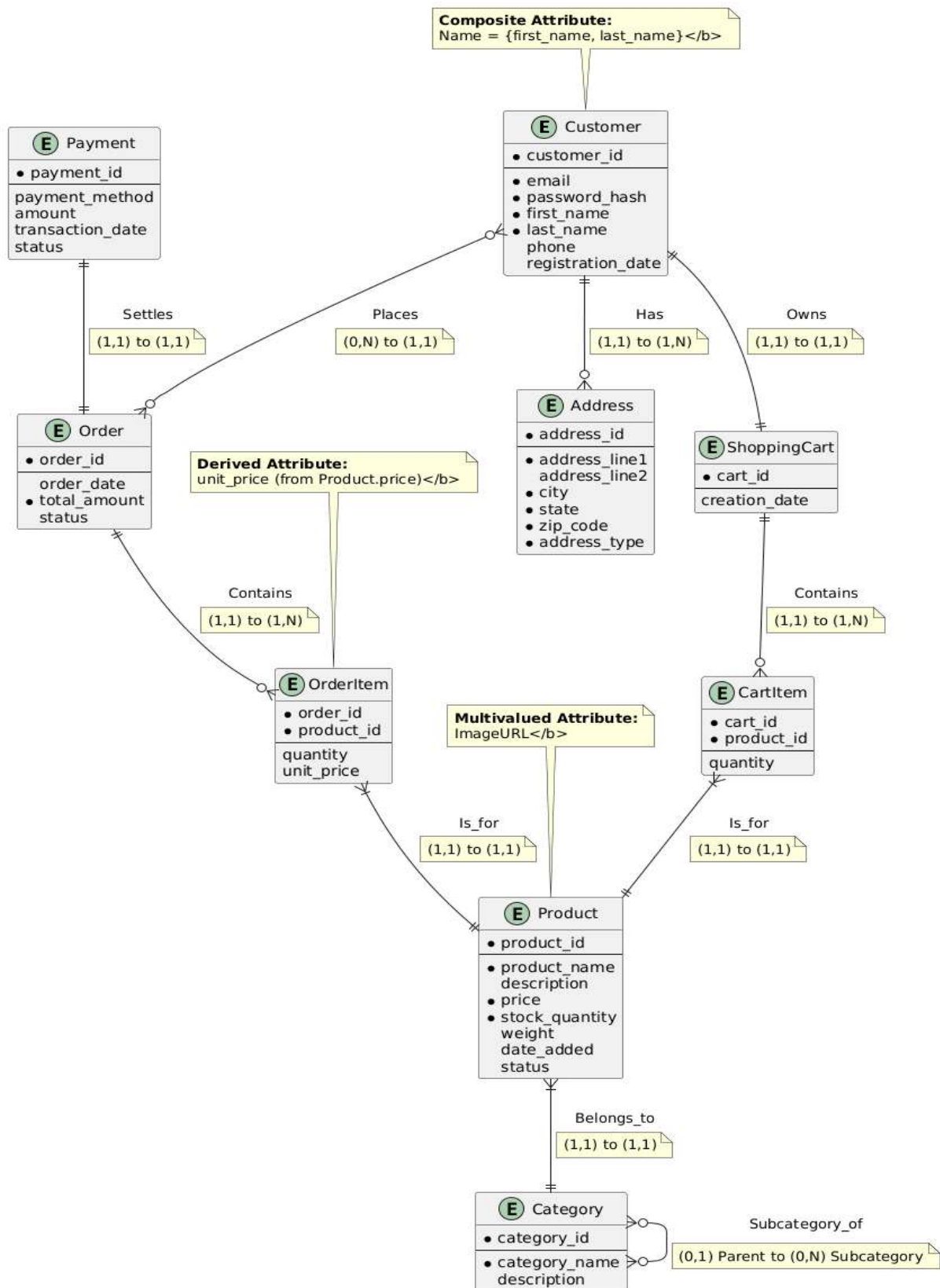


Figure 1 : ER Diagram for E-Commerce Management System

2.2 Relations in the ER Diagram

The Entity-Relationship (ER) diagram created provides a complete conceptual view of the database for the E-Commerce Management System. The key elements and design decisions included in the diagram are explained in this section.

2.2.1 Strong Entities

Strong entities are those that exist independently and do not rely on any other entity for their identification. Their primary key is uniquely defined by their own attributes.

- Customer: Represents a registered user of the platform.
- Category: Represents a classification for products (e.g., Electronics, Clothing).
- Product: Represents an item available for sale.
- ShoppingCart: Represents a virtual cart associated with a single customer.
- Order: Represents a completed purchase transaction.
- Payment: Represents a financial transaction that settles an order.

2.2.2 Weak Entities

Weak entities are existence-dependent on a strong entity. Their primary key is partially or fully derived from the primary key of the strong entity they relate to.

- Address: A weak entity that depends on the Customer entity. An address cannot exist without a customer.
- CartItem: A weak entity that depends on the ShoppingCart entity. An item cannot be in a cart without the cart itself.
- ProductImage: A weak entity that depends on the Product entity. An image cannot exist without a product.

2.2.3 Relationship Cardinalities

One-to-One (1 : 1) Relationships

- Customer OWNS ShoppingCart: (1, 1) to (1, 1)
One Customer must own exactly one ShoppingCart. One ShoppingCart is owned by exactly one Customer.
- Payment SETTLES Order: (1, 1) to (1, 1)
One Payment must settle exactly one Order. One Order must be settled by exactly one Payment.

One-to-Many (1 : N) Relationships

- Customer PLACES Order: (0, N) to (1, 1)
One Customer can place zero or many Orders. One Order must be placed by exactly one Customer.
- Customer HAS Address: (1, 1) to (1, N)
One Customer must have one or more Addresses. One Address must belong to exactly one Customer.
- ShoppingCart CONTAINS CartItem: (1, 1) to (1, N)
One ShoppingCart must contain one or more CartItems. One CartItem must be contained in exactly one ShoppingCart.
- Order CONTAINS OrderItem: (1, 1) to (1, N)
One Order must contain one or more OrderItems. One OrderItem must be part of exactly one Order.
- Product BELONGS_TO Category: (1, 1) to (1, 1)
One Product must belong to exactly one Category. One Category can have one or many Products.
- Product HAS ProductImage: (1, 1) to (1, N)
One Product must have one or more ProductImages. One ProductImage must belong to exactly one Product.

Many-to-Many (M : N) Relationships

- The relationship between Product and ShoppingCart is resolved by the weak entity CartItem. One ShoppingCart can contain many Products, and one Product can be in many ShoppingCarts.
- The relationship between Product and Order is resolved by the weak entity OrderItem. One Order can contain many Products, and one Product can be in many Orders.

Recursive Relationship

- Category IS_SUBCATEGORY_OF Category: (0, 1) to (0, N)
This is a one-to-many relationship within the same entity. One Category (as a parent) can have zero or many subcategories. One Category (as a subcategory) can have zero or one parent Categories.

2.2.4 Special Constructs

Composite Attributes

- Customer Name: Composed of the atomic attributes First Name and Last Name.
- Customer Address: (In the Address entity) Composed of Address Line 1, Address Line 2, City, State, and Zip Code.

Multivalued Attributes

- **Product Image:** A product can have multiple image URLs. This is modeled as the separate weak entity ProductImage.

Derived Attribute

- **OrderItem.UnitPrice:** This attribute is derived from the Price attribute in the Product entity at the time the order is placed. It is stored historically in the OrderItem entity to preserve the sale price, making it a derived and stored attribute.

CHAPTER 3 – LOGICAL DESIGN AND NORMALIZATION

3.1 Mapping to the Relational Model (1NF)

The following tables represent the initial mapping of the ER diagram to a relational schema in First Normal Form (1NF). Each table has a primary key, and all attributes are atomic.

1. Customer Table

Attribute	Data Type	Constraints
customer_id	INT	PRIMARY KEY
email	VARCHAR(255)	NOT NULL, UNIQUE
password_hash	VARCHAR(255)	NOT NULL
first_name	VARCHAR(100)	NOT NULL
last_name	VARCHAR(100)	NOT NULL
phone	VARCHAR(20)	
registration_date	DATE	NOT NULL

2. Address Table

Attribute	Data Type	Constraints
address_id	INT	PRIMARY KEY
customer_id	INT	NOT NULL, FOREIGN KEY
address_line1	VARCHAR(255)	NOT NULL
address_line2	VARCHAR(255)	
city	VARCHAR(100)	NOT NULL
state	VARCHAR(100)	NOT NULL
zip_code	VARCHAR(20)	NOT NULL

Attribute	Data Type	Constraints
address_type	ENUM('S','B')	NOT NULL

3. Category Table

Attribute	Data Type	Constraints
category_id	INT	PRIMARY KEY
category_name	VARCHAR(100)	NOT NULL
description	TEXT	
parent_category_id	INT	FOREIGN KEY (self)

4. Product Table

Attribute	Data Type	Constraints
product_id	INT	PRIMARY KEY
product_name	VARCHAR(255)	NOT NULL
description	TEXT	
price	DECIMAL(10,2)	NOT NULL
stock_quantity	INT	NOT NULL
weight	DECIMAL(10,2)	
date_added	DATE	NOT NULL
status	ENUM('A','I')	NOT NULL
category_id	INT	NOT NULL, FOREIGN KEY

5. ProductImage Table

Attribute	Data Type	Constraints
product_id	INT	FOREIGN KEY (PK)
image_url	VARCHAR(500)	NOT NULL (PK)
PRIMARY KEY (product_id, image_url)		

6. ShoppingCart Table

Attribute	Data Type	Constraints
cart_id	INT	PRIMARY KEY
creation_date	DATE	NOT NULL
customer_id	INT	NOT NULL, UNIQUE, FK

7. CartItem Table

Attribute	Data Type	Constraints
cart_id	INT	FOREIGN KEY (PK)
product_id	INT	FOREIGN KEY (PK)
quantity	INT	NOT NULL, CHECK > 0
PRIMARY KEY (cart_id, product_id)		

8. Order Table

Attribute	Data Type	Constraints
order_id	INT	PRIMARY KEY
order_date	DATETIME	NOT NULL
total_amount	DECIMAL(10,2)	NOT NULL
status	ENUM('P','D','C')	NOT NULL

Attribute	Data Type	Constraints
customer_id	INT	NOT NULL, FOREIGN KEY

9. OrderItem Table

Attribute	Data Type	Constraints
order_id	INT	FOREIGN KEY (PK)
product_id	INT	FOREIGN KEY (PK)
quantity	INT	NOT NULL, CHECK > 0
unit_price	DECIMAL(10,2)	NOT NULL
PRIMARY KEY (order_id, product_id)		

10. Payment Table

Attribute	Data Type	Constraints
payment_id	INT	PRIMARY KEY
payment_method	VARCHAR(50)	NOT NULL
amount	DECIMAL(10,2)	NOT NULL
transaction_date	DATETIME	NOT NULL
status	ENUM('S','F','R')	NOT NULL
order_id	INT	NOT NULL, UNIQUE, FK

3.2 Functional Dependencies

For each table, the primary key functionally determines all other attributes.

- **Customer:** $\text{customer_id} \rightarrow \{\text{email}, \text{password_hash}, \text{first_name}, \text{last_name}, \text{phone}, \text{registration_date}\}$
- **Address:** $\text{address_id} \rightarrow \{\text{customer_id}, \text{address_line1}, \text{address_line2}, \text{city}, \text{state}, \text{zip_code}, \text{address_type}\}$
- **Category:** $\text{category_id} \rightarrow \{\text{category_name}, \text{description}, \text{parent_category_id}\}$
- **Product:** $\text{product_id} \rightarrow \{\text{product_name}, \text{description}, \text{price}, \text{stock_quantity}, \text{weight}, \text{date_added}, \text{status}, \text{category_id}\}$
- **ProductImage:** $\{\text{product_id}, \text{image_url}\} \rightarrow \{\}$ (No other attributes)
- **ShoppingCart:** $\text{cart_id} \rightarrow \{\text{creation_date}, \text{customer_id}\}$
- **CartItem:** $\{\text{cart_id}, \text{product_id}\} \rightarrow \text{quantity}$
- **Order:** $\text{order_id} \rightarrow \{\text{order_date}, \text{total_amount}, \text{status}, \text{customer_id}\}$
- **OrderItem:** $\{\text{order_id}, \text{product_id}\} \rightarrow \{\text{quantity}, \text{unit_price}\}$
- **Payment:** $\text{payment_id} \rightarrow \{\text{payment_method}, \text{amount}, \text{transaction_date}, \text{status}, \text{order_id}\}$

3.3 Normalization to Second Normal Form (2NF)

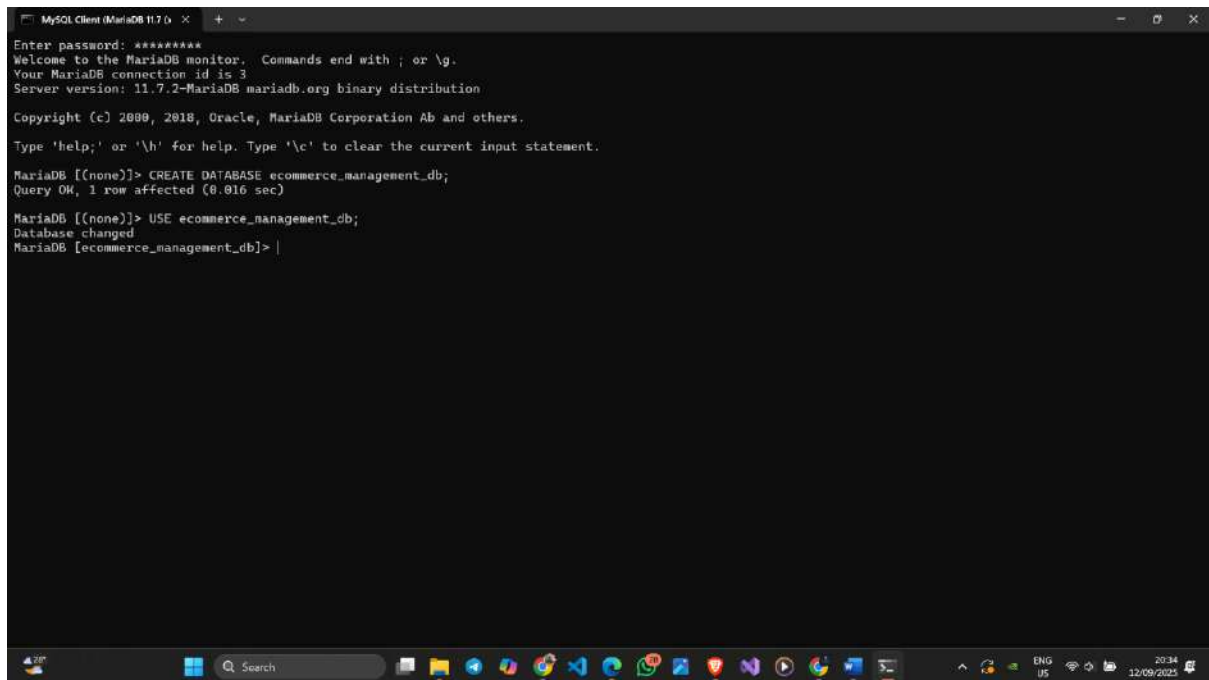
All tables are already in 2NF because:

1. They are all in 1NF (atomic values, no repeating groups).
2. They have no partial dependencies. In all tables with composite primary keys (ProductImage, CartItem, OrderItem), the non-key attributes depend on the entire composite key, not just a part of it. For example, in OrderItem, the quantity and unit_price depend on both the order_id and the product_id together.

No further normalization is required to achieve 2NF. The schema is now optimized to avoid update anomalies and redundancy for this normal form.

CHAPTER 4 – IMPLEMENTATION

4.1 Schema Creation



The screenshot shows a MySQL Client window with the following text:

```
MySQL Client (MariaDB 11.7.0) x + -
Enter password: *****
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 3
Server version: 11.7.2-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

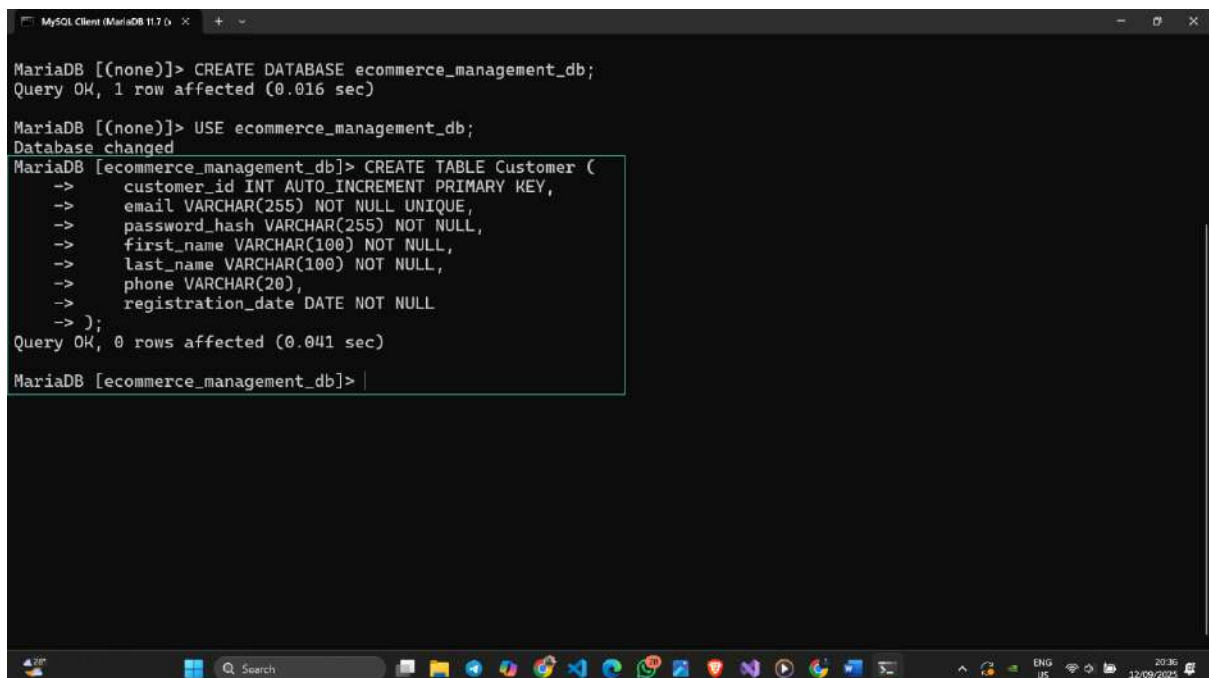
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CREATE DATABASE ecommerce_management_db;
Query OK, 1 row affected (0.016 sec)

MariaDB [(none)]> USE ecommerce_management_db;
Database changed
MariaDB [ecommerce_management_db]> |
```

Figure 2 Creating Database

4.2 Table Creation and Definition



The screenshot shows a MySQL Client window with the following text:

```
MySQL Client (MariaDB 11.7.0) x + -

MariaDB [(none)]> CREATE DATABASE ecommerce_management_db;
Query OK, 1 row affected (0.016 sec)

MariaDB [(none)]> USE ecommerce_management_db;
Database changed
MariaDB [ecommerce_management_db]> CREATE TABLE Customer (
-> customer_id INT AUTO_INCREMENT PRIMARY KEY,
-> email VARCHAR(255) NOT NULL UNIQUE,
-> password_hash VARCHAR(255) NOT NULL,
-> first_name VARCHAR(100) NOT NULL,
-> last_name VARCHAR(100) NOT NULL,
-> phone VARCHAR(20),
-> registration_date DATE NOT NULL
-> );
Query OK, 0 rows affected (0.041 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 3 Creating Customer Table

```
MySQL Client (MariaDB 11.7.0) x + -
MariaDB [(none)]> CREATE DATABASE ecommerce_management_db;
Query OK, 1 row affected (0.016 sec)

MariaDB [(none)]> USE ecommerce_management_db;
Database changed
MariaDB [ecommerce_management_db]> CREATE TABLE Customer (
->     customer_id INT AUTO_INCREMENT PRIMARY KEY,
->     email VARCHAR(255) NOT NULL UNIQUE,
->     password_hash VARCHAR(255) NOT NULL,
->     first_name VARCHAR(100) NOT NULL,
->     last_name VARCHAR(100) NOT NULL,
->     phone VARCHAR(20),
->     registration_date DATE NOT NULL
-> );
Query OK, 0 rows affected (0.041 sec)

MariaDB [ecommerce_management_db]> CREATE TABLE Category (
->     category_id INT AUTO_INCREMENT PRIMARY KEY,
->     category_name VARCHAR(100) NOT NULL,
->     description TEXT,
->     parent_category_id INT NULL,
->     FOREIGN KEY (parent_category_id) REFERENCES Category(category_id)
-> );
Query OK, 0 rows affected (0.007 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 4 Creating Category Table

```
MySQL Client (MariaDB 11.7.0) x + -
->     first_name VARCHAR(100) NOT NULL,
->     last_name VARCHAR(100) NOT NULL,
->     phone VARCHAR(20),
->     registration_date DATE NOT NULL
-> );
Query OK, 0 rows affected (0.041 sec)

MariaDB [ecommerce_management_db]> CREATE TABLE Category (
->     category_id INT AUTO_INCREMENT PRIMARY KEY,
->     category_name VARCHAR(100) NOT NULL,
->     description TEXT,
->     parent_category_id INT NULL,
->     FOREIGN KEY (parent_category_id) REFERENCES Category(category_id)
-> );
Query OK, 0 rows affected (0.007 sec)

MariaDB [ecommerce_management_db]> CREATE TABLE Product (
->     product_id INT AUTO_INCREMENT PRIMARY KEY,
->     product_name VARCHAR(255) NOT NULL,
->     description TEXT,
->     price DECIMAL(10, 2) NOT NULL,
->     stock_quantity INT NOT NULL,
->     weight DECIMAL(10, 2),
->     date_added DATE NOT NULL,
->     status ENUM('Active', 'Inactive') NOT NULL DEFAULT 'Active',
->     category_id INT NOT NULL,
->     FOREIGN KEY (category_id) REFERENCES Category(category_id)
-> );
Query OK, 0 rows affected (0.014 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 5 Creating Product Table

```
MySQL Client (MariaDB 11.7.0)
-> category_id INT AUTO_INCREMENT PRIMARY KEY,
-> category_name VARCHAR(100) NOT NULL,
-> description TEXT,
-> parent_category_id INT NULL,
-> FOREIGN KEY (parent_category_id) REFERENCES Category(category_id)
-> );
Query OK, 0 rows affected (0.007 sec)

MariaDB [ecommerce_management_db]> CREATE TABLE Product (
-> product_id INT AUTO_INCREMENT PRIMARY KEY,
-> product_name VARCHAR(255) NOT NULL,
-> description TEXT,
-> price DECIMAL(10, 2) NOT NULL,
-> stock_quantity INT NOT NULL,
-> weight DECIMAL(10, 2),
-> date_added DATE NOT NULL,
-> status ENUM('Active', 'Inactive') NOT NULL DEFAULT 'Active',
-> category_id INT NOT NULL,
-> FOREIGN KEY (category_id) REFERENCES Category(category_id)
-> );
Query OK, 0 rows affected (0.014 sec)

MariaDB [ecommerce_management_db]> CREATE TABLE ProductImage (
-> product_id INT,
-> image_url VARCHAR(500) NOT NULL,
-> PRIMARY KEY (product_id, image_url),
-> FOREIGN KEY (product_id) REFERENCES Product(product_id) ON DELETE CASCADE
-> );
Query OK, 0 rows affected (0.006 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 6 Creating ProductImage Table

```
MySQL Client (MariaDB 11.7.0)
-> stock_quantity INT NOT NULL,
-> weight DECIMAL(10, 2),
-> date_added DATE NOT NULL,
-> status ENUM('Active', 'Inactive') NOT NULL DEFAULT 'Active',
-> category_id INT NOT NULL,
-> FOREIGN KEY (category_id) REFERENCES Category(category_id)
-> );
Query OK, 0 rows affected (0.014 sec)

MariaDB [ecommerce_management_db]> CREATE TABLE ProductImage (
-> product_id INT,
-> image_url VARCHAR(500) NOT NULL,
-> PRIMARY KEY (product_id, image_url),
-> FOREIGN KEY (product_id) REFERENCES Product(product_id) ON DELETE CASCADE
-> );
Query OK, 0 rows affected (0.006 sec)

MariaDB [ecommerce_management_db]> CREATE TABLE Address (
-> address_id INT AUTO_INCREMENT PRIMARY KEY,
-> customer_id INT NOT NULL,
-> address_line1 VARCHAR(255) NOT NULL,
-> address_line2 VARCHAR(255),
-> city VARCHAR(100) NOT NULL,
-> state VARCHAR(100) NOT NULL,
-> zip_code VARCHAR(20) NOT NULL,
-> address_type ENUM('Shipping', 'Billing') NOT NULL,
-> FOREIGN KEY (customer_id) REFERENCES Customer(customer_id) ON DELETE CASCADE
-> );
Query OK, 0 rows affected (0.008 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 7 Creating Address Table

```
MySQL Client (MariaDB 11.7.0) x + -
MariaDB [ecommerce_management_db]> CREATE TABLE ProductImage (
->   product_id INT,
->   image_url VARCHAR(500) NOT NULL,
->   PRIMARY KEY (product_id, image_url),
->   FOREIGN KEY (product_id) REFERENCES Product(product_id) ON DELETE CASCADE
-> );
Query OK, 0 rows affected (0.006 sec)

MariaDB [ecommerce_management_db]> CREATE TABLE Address (
->   address_id INT AUTO_INCREMENT PRIMARY KEY,
->   customer_id INT NOT NULL,
->   address_line1 VARCHAR(255) NOT NULL,
->   address_line2 VARCHAR(255),
->   city VARCHAR(100) NOT NULL,
->   state VARCHAR(100) NOT NULL,
->   zip_code VARCHAR(20) NOT NULL,
->   address_type ENUM('Shipping', 'Billing') NOT NULL,
->   FOREIGN KEY (customer_id) REFERENCES Customer(customer_id) ON DELETE CASCADE
-> );
Query OK, 0 rows affected (0.008 sec)

MariaDB [ecommerce_management_db]> CREATE TABLE ShoppingCart (
->   cart_id INT AUTO_INCREMENT PRIMARY KEY,
->   creation_date DATE NOT NULL,
->   customer_id INT NOT NULL UNIQUE,
->   FOREIGN KEY (customer_id) REFERENCES Customer(customer_id)
-> );
Query OK, 0 rows affected (0.010 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 8 Creating ShoppingCart Table

```
MySQL Client (MariaDB 11.7.0) x + -
->   address_id INT AUTO_INCREMENT PRIMARY KEY,
->   customer_id INT NOT NULL,
->   address_line1 VARCHAR(255) NOT NULL,
->   address_line2 VARCHAR(255),
->   city VARCHAR(100) NOT NULL,
->   state VARCHAR(100) NOT NULL,
->   zip_code VARCHAR(20) NOT NULL,
->   address_type ENUM('Shipping', 'Billing') NOT NULL,
->   FOREIGN KEY (customer_id) REFERENCES Customer(customer_id) ON DELETE CASCADE
-> );
Query OK, 0 rows affected (0.008 sec)

MariaDB [ecommerce_management_db]> CREATE TABLE ShoppingCart (
->   cart_id INT AUTO_INCREMENT PRIMARY KEY,
->   creation_date DATE NOT NULL,
->   customer_id INT NOT NULL UNIQUE,
->   FOREIGN KEY (customer_id) REFERENCES Customer(customer_id)
-> );
Query OK, 0 rows affected (0.010 sec)

MariaDB [ecommerce_management_db]> CREATE TABLE CartItem (
->   cart_id INT,
->   product_id INT,
->   quantity INT NOT NULL CHECK (quantity > 0),
->   PRIMARY KEY (cart_id, product_id),
->   FOREIGN KEY (cart_id) REFERENCES ShoppingCart(cart_id) ON DELETE CASCADE,
->   FOREIGN KEY (product_id) REFERENCES Product(product_id)
-> );
Query OK, 0 rows affected (0.013 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 9 Creating CartItem Table


```
MySQL Client (MariaDB 11.7.0) x + -
Query OK, 0 rows affected (0.008 sec)

MariaDB [ecommerce_management_db]> CREATE TABLE ShoppingCart (
->   cart_id INT AUTO_INCREMENT PRIMARY KEY,
->   creation_date DATE NOT NULL,
->   customer_id INT NOT NULL UNIQUE,
->   FOREIGN KEY (customer_id) REFERENCES Customer(customer_id)
-> );
Query OK, 0 rows affected (0.010 sec)

MariaDB [ecommerce_management_db]> CREATE TABLE CartItem (
->   cart_id INT,
->   product_id INT,
->   quantity INT NOT NULL CHECK (quantity > 0),
->   PRIMARY KEY (cart_id, product_id),
->   FOREIGN KEY (cart_id) REFERENCES ShoppingCart(cart_id) ON DELETE CASCADE,
->   FOREIGN KEY (product_id) REFERENCES Product(product_id)
-> );
Query OK, 0 rows affected (0.013 sec)

MariaDB [ecommerce_management_db]> CREATE TABLE 'Order' (
->   order_id INT AUTO_INCREMENT PRIMARY KEY,
->   order_date DATETIME NOT NULL,
->   total_amount DECIMAL(10, 2) NOT NULL,
->   status ENUM('Pending', 'Paid', 'Shipped', 'Delivered', 'Cancelled') NOT NULL DEFAULT 'Pending',
->   customer_id INT NOT NULL,
->   FOREIGN KEY (customer_id) REFERENCES Customer(customer_id)
-> );
Query OK, 0 rows affected (0.009 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 10 Creating Order Table

```
MySQL Client (MariaDB 11.7.0) x + -
->   cart_id INT,
->   product_id INT,
->   quantity INT NOT NULL CHECK (quantity > 0),
->   PRIMARY KEY (cart_id, product_id),
->   FOREIGN KEY (cart_id) REFERENCES ShoppingCart(cart_id) ON DELETE CASCADE,
->   FOREIGN KEY (product_id) REFERENCES Product(product_id)
-> );
Query OK, 0 rows affected (0.013 sec)

MariaDB [ecommerce_management_db]> CREATE TABLE 'Order' (
->   order_id INT AUTO_INCREMENT PRIMARY KEY,
->   order_date DATETIME NOT NULL,
->   total_amount DECIMAL(10, 2) NOT NULL,
->   status ENUM('Pending', 'Paid', 'Shipped', 'Delivered', 'Cancelled') NOT NULL DEFAULT 'Pending',
->   customer_id INT NOT NULL,
->   FOREIGN KEY (customer_id) REFERENCES Customer(customer_id)
-> );
Query OK, 0 rows affected (0.009 sec)

MariaDB [ecommerce_management_db]> CREATE TABLE OrderItem (
->   order_id INT,
->   product_id INT,
->   quantity INT NOT NULL CHECK (quantity > 0),
->   unit_price DECIMAL(10, 2) NOT NULL,
->   PRIMARY KEY (order_id, product_id),
->   FOREIGN KEY (order_id) REFERENCES 'Order'(order_id),
->   FOREIGN KEY (product_id) REFERENCES Product(product_id)
-> );
Query OK, 0 rows affected (0.010 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 11 Creating OrderItem Table

```
MySQL Client (MariaDB 11.7.0)
-> order_date DATETIME NOT NULL,
-> total_amount DECIMAL(10, 2) NOT NULL,
-> status ENUM('Pending', 'Paid', 'Shipped', 'Delivered', 'Cancelled') NOT NULL DEFAULT 'Pending',
-> customer_id INT NOT NULL,
-> FOREIGN KEY (customer_id) REFERENCES Customer(customer_id)
-> );
Query OK, 0 rows affected (0.009 sec)

MariaDB [ecommerce_management_db]> CREATE TABLE OrderItem (
-> order_id INT,
-> product_id INT,
-> quantity INT NOT NULL CHECK (quantity > 0),
-> unit_price DECIMAL(10, 2) NOT NULL,
-> PRIMARY KEY (order_id, product_id),
-> FOREIGN KEY (order_id) REFERENCES `Order`(order_id),
-> FOREIGN KEY (product_id) REFERENCES Product(product_id)
-> );
Query OK, 0 rows affected (0.010 sec)

MariaDB [ecommerce_management_db]> CREATE TABLE Payment (
-> payment_id INT AUTO_INCREMENT PRIMARY KEY,
-> payment_method VARCHAR(50) NOT NULL,
-> amount DECIMAL(10, 2) NOT NULL,
-> transaction_date DATETIME NOT NULL,
-> status ENUM('Success', 'Failed', 'Refunded') NOT NULL,
-> order_id INT NOT NULL UNIQUE,
-> FOREIGN KEY (order_id) REFERENCES `Order`(order_id)
-> );
Query OK, 0 rows affected (0.010 sec)

MariaDB [ecommerce_management_db]>
```

Figure 12 Creating Payment Table

4.3 Data Insertion (INSERT operations)

```
MySQL Client (MariaDB 11.7.0) x + -
MariaDB [ecommerce_management_db]> CREATE TABLE OrderItem (
->   order_id INT,
->   product_id INT,
->   quantity INT NOT NULL CHECK (quantity > 0),
->   unit_price DECIMAL(10, 2) NOT NULL,
->   PRIMARY KEY (order_id, product_id),
->   FOREIGN KEY (order_id) REFERENCES `Order` (order_id),
->   FOREIGN KEY (product_id) REFERENCES Product (product_id)
-> );
Query OK, 0 rows affected (0.010 sec)

MariaDB [ecommerce_management_db]> CREATE TABLE Payment (
->   payment_id INT AUTO_INCREMENT PRIMARY KEY,
->   payment_method VARCHAR(50) NOT NULL,
->   amount DECIMAL(10, 2) NOT NULL,
->   transaction_date DATETIME NOT NULL,
->   status ENUM('Success', 'Failed', 'Refunded') NOT NULL,
->   order_id INT NOT NULL UNIQUE,
->   FOREIGN KEY (order_id) REFERENCES `Order` (order_id)
-> );
Query OK, 0 rows affected (0.010 sec)

MariaDB [ecommerce_management_db]> INSERT INTO Customer (email, password_hash, first_name, last_name, phone, registration_date) VALUES
-> ('alice.johnson@email.com', 'hash1', 'Alice', 'Johnson', '123-456-7890', '2024-01-15'),
-> ('bob.smith@email.com', 'hash2', 'Bob', 'Smith', '123-456-7891', '2024-02-20'),
-> ('charlie.brown@email.com', 'hash3', 'Charlie', 'Brown', NULL, '2024-03-05'),
-> ('diana.prince@email.com', 'hash4', 'Diana', 'Prince', '123-456-7893', '2024-03-10'),
-> ('evan.lee@email.com', 'hash5', 'Evan', 'Lee', '123-456-7894', '2024-04-22'),
-> ('fiona.campbell@email.com', 'hash6', 'Fiona', 'Campbell', '123-456-7895', '2024-05-30');
Query OK, 6 rows affected (0.125 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> |
```

Figure 13 Inserting Data to the Customer Table

```
MySQL Client (MariaDB 11.7.0) x + -
MariaDB [ecommerce_management_db]> CREATE TABLE Payment (
->   payment_id INT AUTO_INCREMENT PRIMARY KEY,
->   payment_method VARCHAR(50) NOT NULL,
->   amount DECIMAL(10, 2) NOT NULL,
->   transaction_date DATETIME NOT NULL,
->   status ENUM('Success', 'Failed', 'Refunded') NOT NULL,
->   order_id INT NOT NULL UNIQUE,
->   FOREIGN KEY (order_id) REFERENCES `Order` (order_id)
-> );
Query OK, 0 rows affected (0.010 sec)

MariaDB [ecommerce_management_db]> INSERT INTO Customer (email, password_hash, first_name, last_name, phone, registration_date) VALUES
-> ('alice.johnson@email.com', 'hash1', 'Alice', 'Johnson', '123-456-7890', '2024-01-15'),
-> ('bob.smith@email.com', 'hash2', 'Bob', 'Smith', '123-456-7891', '2024-02-20'),
-> ('charlie.brown@email.com', 'hash3', 'Charlie', 'Brown', NULL, '2024-03-05'),
-> ('diana.prince@email.com', 'hash4', 'Diana', 'Prince', '123-456-7893', '2024-03-10'),
-> ('evan.lee@email.com', 'hash5', 'Evan', 'Lee', '123-456-7894', '2024-04-22'),
-> ('fiona.campbell@email.com', 'hash6', 'Fiona', 'Campbell', '123-456-7895', '2024-05-30');
Query OK, 6 rows affected (0.125 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from Customer;
+-----+-----+-----+-----+-----+-----+
| customer_id | email | password_hash | first_name | last_name | phone | registration_date |
+-----+-----+-----+-----+-----+-----+
| 1 | alice.johnson@email.com | hash1 | Alice | Johnson | 123-456-7890 | 2024-01-15 |
| 2 | bob.smith@email.com | hash2 | Bob | Smith | 123-456-7891 | 2024-02-20 |
| 3 | charlie.brown@email.com | hash3 | Charlie | Brown | NULL | 2024-03-05 |
| 4 | diana.prince@email.com | hash4 | Diana | Prince | 123-456-7893 | 2024-03-10 |
| 5 | evan.lee@email.com | hash5 | Evan | Lee | 123-456-7894 | 2024-04-22 |
| 6 | fiona.campbell@email.com | hash6 | Fiona | Campbell | 123-456-7895 | 2024-05-30 |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.006 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 14 Customer Table with Inserted Data

```
MySQL Client (MariaDB 11.7.0) x + -
MariaDB [ecommerce_management_db]> INSERT INTO Customer (email, password_hash, first_name, last_name, phone, registration_date) VALUES
-> ('alice.johnson@email.com', 'hash1', 'Alice', 'Johnson', '123-456-7890', '2024-01-15'),
-> ('bob.smith@email.com', 'hash2', 'Bob', 'Smith', '123-456-7891', '2024-02-20'),
-> ('charlie.brown@email.com', 'hash3', 'Charlie', 'Brown', NULL, '2024-03-05'),
-> ('diana.prince@email.com', 'hash4', 'Diana', 'Prince', '123-456-7893', '2024-03-10'),
-> ('evan.lee@email.com', 'hash5', 'Evan', 'Lee', '123-456-7894', '2024-04-22'),
-> ('fiona.campbell@email.com', 'hash6', 'Fiona', 'Campbell', '123-456-7895', '2024-05-30');
Query OK, 6 rows affected (0.125 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from Customer;
+-----+-----+-----+-----+-----+-----+-----+
| customer_id | email | password_hash | first_name | last_name | phone | registration_date |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | alice.johnson@email.com | hash1 | Alice | Johnson | 123-456-7890 | 2024-01-15 |
| 2 | bob.smith@email.com | hash2 | Bob | Smith | 123-456-7891 | 2024-02-20 |
| 3 | charlie.brown@email.com | hash3 | Charlie | Brown | NULL | 2024-03-05 |
| 4 | diana.prince@email.com | hash4 | Diana | Prince | 123-456-7893 | 2024-03-10 |
| 5 | evan.lee@email.com | hash5 | Evan | Lee | 123-456-7894 | 2024-04-22 |
| 6 | fiona.campbell@email.com | hash6 | Fiona | Campbell | 123-456-7895 | 2024-05-30 |
+-----+-----+-----+-----+-----+-----+-----+
6 rows in set (0.006 sec)

MariaDB [ecommerce_management_db]> INSERT INTO Category (category_name, description, parent_category_id) VALUES
-> ('Electronics', 'Gadgets and devices', NULL),
-> ('Computers', 'Desktop and laptop computers', 1),
-> ('Laptops', 'Portable computers', 2),
-> ('Smartphones', 'Mobile phones', 1),
-> ('Fashion', 'Clothing and accessories', NULL),
-> ('Men's Clothing', 'Clothing for men', 5);
Query OK, 6 rows affected (0.006 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> |
```

Figure 15 Inserting Data to the Category Table

```
MySQL Client (MariaDB 11.7.0) x + -
+-----+-----+-----+-----+-----+-----+-----+
| customer_id | email | password_hash | first_name | last_name | phone | registration_date |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | alice.johnson@email.com | hash1 | Alice | Johnson | 123-456-7890 | 2024-01-15 |
| 2 | bob.smith@email.com | hash2 | Bob | Smith | 123-456-7891 | 2024-02-20 |
| 3 | charlie.brown@email.com | hash3 | Charlie | Brown | NULL | 2024-03-05 |
| 4 | diana.prince@email.com | hash4 | Diana | Prince | 123-456-7893 | 2024-03-10 |
| 5 | evan.lee@email.com | hash5 | Evan | Lee | 123-456-7894 | 2024-04-22 |
| 6 | fiona.campbell@email.com | hash6 | Fiona | Campbell | 123-456-7895 | 2024-05-30 |
+-----+-----+-----+-----+-----+-----+-----+
6 rows in set (0.006 sec)

MariaDB [ecommerce_management_db]> INSERT INTO Category (category_name, description, parent_category_id) VALUES
-> ('Electronics', 'Gadgets and devices', NULL),
-> ('Computers', 'Desktop and laptop computers', 1),
-> ('Laptops', 'Portable computers', 2),
-> ('Smartphones', 'Mobile phones', 1),
-> ('Fashion', 'Clothing and accessories', NULL),
-> ('Men's Clothing', 'Clothing for men', 5);
Query OK, 6 rows affected (0.006 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from Category;
+-----+-----+-----+-----+
| category_id | category_name | description | parent_category_id |
+-----+-----+-----+-----+
| 1 | Electronics | Gadgets and devices | NULL |
| 2 | Computers | Desktop and laptop computers | 1 |
| 3 | Laptops | Portable computers | 2 |
| 4 | Smartphones | Mobile phones | 1 |
| 5 | Fashion | Clothing and accessories | NULL |
| 6 | Men's Clothing | Clothing for men | 5 |
+-----+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 16 Category Table with Inserted Data

```
MySQL Client (MariaDB 11.7.0) x + -
3 | charlie.brown@email.com | hash3 | Charlie | Brown | NULL | 2024-03-05 |
4 | diana.prince@email.com | hash4 | Diana | Prince | 123-456-7893 | 2024-03-10 |
5 | evan.lee@email.com | hash5 | Evan | Lee | 123-456-7894 | 2024-04-22 |
6 | fiona.campbell@email.com | hash6 | Fiona | Campbell | 123-456-7895 | 2024-05-30 |
6 rows in set (0.006 sec)

MariaDB [ecommerce_management_db]> INSERT INTO Category (category_name, description, parent_category_id) VALUES
-> ('Electronics', 'Gadgets and devices', NULL),
-> ('Computers', 'Desktop and laptop computers', 1),
-> ('Laptops', 'Portable computers', 2),
-> ('Smartphones', 'Mobile phones', 1),
-> ('Fashion', 'Clothing and accessories', NULL),
-> ('Men's Clothing', 'Clothing for men', 5);
Query OK, 6 rows affected (0.006 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from Category;
+-----+-----+-----+-----+
| category_id | category_name | description | parent_category_id |
+-----+-----+-----+-----+
| 1 | Electronics | Gadgets and devices | NULL |
| 2 | Computers | Desktop and laptop computers | 1 |
| 3 | Laptops | Portable computers | 2 |
| 4 | Smartphones | Mobile phones | 1 |
| 5 | Fashion | Clothing and accessories | NULL |
| 6 | Men's Clothing | Clothing for men | 5 |
+-----+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> INSERT INTO Product (product_name, description, price, stock_quantity, weight, date_added, status, category_id) VALUES
-> ('Smartphone X', 'Latest model with high-resolution camera', 749.99, 50, 0.2, '2024-05-01', 'Active', 4),
-> ('Gaming Laptop Pro', 'Powerful laptop for gaming and productivity', 1599.99, 15, 2.5, '2024-04-15', 'Active', 3),
-> ('Wireless Earbuds', 'Noise-cancelling wireless earbuds', 129.99, 100, 0.05, '2024-05-10', 'Active', 1),
-> ('Cotton T-Shirt', '100% cotton, comfortable fit', 24.99, 200, 0.3, '2024-03-20', 'Active', 6),
-> ('Men's Running Shoes', 'Lightweight and breathable for running', 89.99, 75, 0.8, '2024-02-25', 'Active', 6),
-> ('Bluetooth Speaker', 'Portable speaker with 10-hour battery', 59.99, 30, 0.7, '2024-05-05', 'Active', 1);
Query OK, 6 rows affected (0.009 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> |
```

Figure 17 Inserting Data to the Product Table

```
MySQL Client (MariaDB 11.7.0) x + -
-> ('Men's Clothing', 'Clothing for men', 5);
Query OK, 6 rows affected (0.006 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from Category;
+-----+-----+-----+-----+
| category_id | category_name | description | parent_category_id |
+-----+-----+-----+-----+
| 1 | Electronics | Gadgets and devices | NULL |
| 2 | Computers | Desktop and laptop computers | 1 |
| 3 | Laptops | Portable computers | 2 |
| 4 | Smartphones | Mobile phones | 1 |
| 5 | Fashion | Clothing and accessories | NULL |
| 6 | Men's Clothing | Clothing for men | 5 |
+-----+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> INSERT INTO Product (product_name, description, price, stock_quantity, weight, date_added, status, category_id) VALUES
-> ('Smartphone X', 'Latest model with high-resolution camera', 749.99, 50, 0.2, '2024-05-01', 'Active', 4),
-> ('Gaming Laptop Pro', 'Powerful laptop for gaming and productivity', 1599.99, 15, 2.5, '2024-04-15', 'Active', 3),
-> ('Wireless Earbuds', 'Noise-cancelling wireless earbuds', 129.99, 100, 0.05, '2024-05-10', 'Active', 1),
-> ('Cotton T-Shirt', '100% cotton, comfortable fit', 24.99, 200, 0.3, '2024-03-20', 'Active', 6),
-> ('Men's Running Shoes', 'Lightweight and breathable for running', 89.99, 75, 0.8, '2024-02-25', 'Active', 6),
-> ('Bluetooth Speaker', 'Portable speaker with 10-hour battery', 59.99, 30, 0.7, '2024-05-05', 'Active', 1);
Query OK, 6 rows affected (0.009 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from Product;
+-----+-----+-----+-----+-----+-----+-----+-----+
| product_id | product_name | description | price | stock_quantity | weight | date_added | status | category_id |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | Smartphone X | Latest model with high-resolution camera | 749.99 | 50 | 0.20 | 2024-05-01 | Active | 4 |
| 2 | Gaming Laptop Pro | Powerful laptop for gaming and productivity | 1599.99 | 15 | 2.50 | 2024-04-15 | Active | 3 |
| 3 | Wireless Earbuds | Noise-cancelling wireless earbuds | 129.99 | 100 | 0.05 | 2024-05-10 | Active | 1 |
| 4 | Cotton T-Shirt | 100% cotton, comfortable fit | 24.99 | 200 | 0.30 | 2024-03-20 | Active | 6 |
| 5 | Men's Running Shoes | Lightweight and breathable for running | 89.99 | 75 | 0.80 | 2024-02-25 | Active | 6 |
| 6 | Bluetooth Speaker | Portable speaker with 10-hour battery | 59.99 | 30 | 0.70 | 2024-05-05 | Active | 1 |
+-----+-----+-----+-----+-----+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 18 Product Table with Inserted Data


```
MySQL Client (MariaDB 11.7.0) x + -
3 | Laptops | Portable computers | 2
4 | Smartphones | Mobile phones | 1
5 | Fashion | Clothing and accessories | NULL
6 | Men's Clothing | Clothing for men | 5
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> INSERT INTO Product (product_name, description, price, stock_quantity, weight, date_added, status, category_id) VALUES
-> ('Smartphone X', 'Latest model with high-resolution camera', 749.99, 50, 0.2, '2024-05-01', 'Active', 4),
-> ('Gaming Laptop Pro', 'Powerful laptop for gaming and productivity', 1599.99, 15, 2.5, '2024-04-15', 'Active', 3),
-> ('Wireless Earbuds', 'Noise-cancelling wireless earbuds', 129.99, 100, 0.05, '2024-05-10', 'Active', 1),
-> ('Cotton T-Shirt', '100% cotton, comfortable fit', 24.99, 200, 0.3, '2024-03-20', 'Active', 6),
-> ('Men's Running Shoes', 'Lightweight and breathable for running', 89.99, 75, 0.8, '2024-02-25', 'Active', 6),
-> ('Bluetooth Speaker', 'Portable speaker with 10-hour battery', 59.99, 30, 0.7, '2024-05-05', 'Active', 1);
Query OK, 6 rows affected (0.009 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from Product;
+-----+-----+-----+-----+-----+-----+-----+-----+
| product_id | product_name | description | price | stock_quantity | weight | date_added | status | category_id |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | Smartphone X | Latest model with high-resolution camera | 749.99 | 50 | 0.20 | 2024-05-01 | Active | 4 |
| 2 | Gaming Laptop Pro | Powerful laptop for gaming and productivity | 1599.99 | 15 | 2.50 | 2024-04-15 | Active | 3 |
| 3 | Wireless Earbuds | Noise-cancelling wireless earbuds | 129.99 | 100 | 0.05 | 2024-05-10 | Active | 1 |
| 4 | Cotton T-Shirt | 100% cotton, comfortable fit | 24.99 | 200 | 0.30 | 2024-03-20 | Active | 6 |
| 5 | Men's Running Shoes | Lightweight and breathable for running | 89.99 | 75 | 0.80 | 2024-02-25 | Active | 6 |
| 6 | Bluetooth Speaker | Portable speaker with 10-hour battery | 59.99 | 30 | 0.70 | 2024-05-05 | Active | 1 |
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> INSERT INTO ProductImage (product_id, image_url) VALUES
-> (1, 'https://example.com/images/smartphone_x_1.jpg'),
-> (1, 'https://example.com/images/smartphone_x_2.jpg'), -- Product 1 has 2 images
-> (2, 'https://example.com/images/laptop_pro_1.jpg'),
-> (3, 'https://example.com/images/earbuds_1.jpg'),
-> (4, 'https://example.com/images/tshirt_1.jpg'),
-> (5, 'https://example.com/images/runningshoes_1.jpg');
Query OK, 6 rows affected (0.004 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> |
```

Figure 19 Inserting Data to the ProductImage Table

```
MySQL Client (MariaDB 11.7.0) x + -
-> ('Bluetooth Speaker', 'Portable speaker with 10-hour battery', 59.99, 30, 0.7, '2024-05-05', 'Active', 1);
Query OK, 6 rows affected (0.009 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from Product;
+-----+-----+-----+-----+-----+-----+-----+-----+
| product_id | product_name | description | price | stock_quantity | weight | date_added | status | category_id |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | Smartphone X | Latest model with high-resolution camera | 749.99 | 50 | 0.20 | 2024-05-01 | Active | 4 |
| 2 | Gaming Laptop Pro | Powerful laptop for gaming and productivity | 1599.99 | 15 | 2.50 | 2024-04-15 | Active | 3 |
| 3 | Wireless Earbuds | Noise-cancelling wireless earbuds | 129.99 | 100 | 0.05 | 2024-05-10 | Active | 1 |
| 4 | Cotton T-Shirt | 100% cotton, comfortable fit | 24.99 | 200 | 0.30 | 2024-03-20 | Active | 6 |
| 5 | Men's Running Shoes | Lightweight and breathable for running | 89.99 | 75 | 0.80 | 2024-02-25 | Active | 6 |
| 6 | Bluetooth Speaker | Portable speaker with 10-hour battery | 59.99 | 30 | 0.70 | 2024-05-05 | Active | 1 |
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> INSERT INTO ProductImage (product_id, image_url) VALUES
-> (1, 'https://example.com/images/smartphone_x_1.jpg'),
-> (1, 'https://example.com/images/smartphone_x_2.jpg'), -- Product 1 has 2 images
-> (2, 'https://example.com/images/laptop_pro_1.jpg'),
-> (3, 'https://example.com/images/earbuds_1.jpg'),
-> (4, 'https://example.com/images/tshirt_1.jpg'),
-> (5, 'https://example.com/images/runningshoes_1.jpg');
Query OK, 6 rows affected (0.004 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from ProductImage;
+-----+-----+
| product_id | image_url |
+-----+-----+
| 1 | https://example.com/images/smartphone_x_1.jpg |
| 1 | https://example.com/images/smartphone_x_2.jpg |
| 2 | https://example.com/images/laptop_pro_1.jpg |
| 3 | https://example.com/images/earbuds_1.jpg |
| 4 | https://example.com/images/tshirt_1.jpg |
| 5 | https://example.com/images/runningshoes_1.jpg |
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 20 ProductImage Table with Inserted Data

```
MySQL Client (MariaDB 11.7.0) x + -
3 | Wireless Earbuds | Noise-cancelling wireless earbuds | 129.99 | 100 | 0.05 | 2024-05-10 | Active | 1 |
4 | Cotton T-Shirt | 100% cotton, comfortable fit | 24.99 | 200 | 0.30 | 2024-03-20 | Active | 6 |
5 | Men's Running Shoes | Lightweight and breathable for running | 89.99 | 75 | 0.80 | 2024-02-25 | Active | 6 |
6 | Bluetooth Speaker | Portable speaker with 10-hour battery | 59.99 | 30 | 0.70 | 2024-05-05 | Active | 1 |
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> INSERT INTO ProductImage (product_id, image_url) VALUES
-> (1, 'https://example.com/images/smartphone_x_1.jpg'),
-> (1, 'https://example.com/images/smartphone_x_2.jpg'), -- Product 1 has 2 images
-> (2, 'https://example.com/images/laptop_pro_1.jpg'),
-> (3, 'https://example.com/images/earbuds_1.jpg'),
-> (4, 'https://example.com/images/tshirt_1.jpg'),
-> (5, 'https://example.com/images/runningshoes_1.jpg');
Query OK, 6 rows affected (0.004 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from ProductImage;
+-----+-----+
| product_id | image_url |
+-----+-----+
| 1 | https://example.com/images/smartphone_x_1.jpg |
| 1 | https://example.com/images/smartphone_x_2.jpg |
| 2 | https://example.com/images/laptop_pro_1.jpg |
| 3 | https://example.com/images/earbuds_1.jpg |
| 4 | https://example.com/images/tshirt_1.jpg |
| 5 | https://example.com/images/runningshoes_1.jpg |
+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> INSERT INTO Address (customer_id, address_line1, address_line2, city, state, zip_code, address_type) VALUES
-> (1, '123 Main St', 'Apt 4B', 'Colombo', 'Western', '10000', 'Shipping'),
-> (1, '456 Central Rd', NULL, 'Galle', 'Southern', '20000', 'Billing'), -- Customer 1 has 2 addresses
-> (2, '789 Park Ave', NULL, 'Kandy', 'Central', '30000', 'Shipping'),
-> (3, '321 Oak Lane', 'Floor 5', 'Jaffna', 'Northern', '40000', 'Billing'),
-> (4, '555 Beach Rd', NULL, 'Matara', 'Southern', '50000', 'Shipping'),
-> (5, '999 Hill St', 'Suite 100', 'Negombo', 'Western', '60000', 'Shipping');
Query OK, 6 rows affected (0.005 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> |
```

Figure 21 Inserting Data to the Address Table

```
MySQL Client (MariaDB 11.7.0) x + -
-> (5, 'https://example.com/images/runningshoes_1.jpg');
Query OK, 6 rows affected (0.004 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from ProductImage;
+-----+-----+
| product_id | image_url |
+-----+-----+
| 1 | https://example.com/images/smartphone_x_1.jpg |
| 1 | https://example.com/images/smartphone_x_2.jpg |
| 2 | https://example.com/images/laptop_pro_1.jpg |
| 3 | https://example.com/images/earbuds_1.jpg |
| 4 | https://example.com/images/tshirt_1.jpg |
| 5 | https://example.com/images/runningshoes_1.jpg |
+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> INSERT INTO Address (customer_id, address_line1, address_line2, city, state, zip_code, address_type) VALUES
-> (1, '123 Main St', 'Apt 4B', 'Colombo', 'Western', '10000', 'Shipping'),
-> (1, '456 Central Rd', NULL, 'Galle', 'Southern', '20000', 'Billing'), -- Customer 1 has 2 addresses
-> (2, '789 Park Ave', NULL, 'Kandy', 'Central', '30000', 'Shipping'),
-> (3, '321 Oak Lane', 'Floor 5', 'Jaffna', 'Northern', '40000', 'Billing'),
-> (4, '555 Beach Rd', NULL, 'Matara', 'Southern', '50000', 'Shipping'),
-> (5, '999 Hill St', 'Suite 100', 'Negombo', 'Western', '60000', 'Shipping');
Query OK, 6 rows affected (0.005 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from Address;
+-----+-----+-----+-----+-----+-----+-----+-----+
| address_id | customer_id | address_line1 | address_line2 | city | state | zip_code | address_type |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | 1 | 123 Main St | Apt 4B | Colombo | Western | 10000 | Shipping |
| 2 | 1 | 456 Central Rd | NULL | Galle | Southern | 20000 | Billing |
| 3 | 2 | 789 Park Ave | NULL | Kandy | Central | 30000 | Shipping |
| 4 | 3 | 321 Oak Lane | Floor 5 | Jaffna | Northern | 40000 | Billing |
| 5 | 4 | 555 Beach Rd | NULL | Matara | Southern | 50000 | Shipping |
| 6 | 5 | 999 Hill St | Suite 100 | Negombo | Western | 60000 | Shipping |
+-----+-----+-----+-----+-----+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 22 Address Table with Inserted Data

```
MySQL Client (MariaDB 11.7.0) x + -
| 5 | https://example.com/images/runningshoes.1.jpg |
+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> INSERT INTO Address (customer_id, address_line1, address_line2, city, state, zip_code, address_type) VALUES
-> (1, '123 Main St', 'Apt 4B', 'Colombo', 'Western', '10000', 'Shipping'),
-> (1, '456 Central Rd', NULL, 'Galle', 'Southern', '20000', 'Billing'), -- Customer 1 has 2 addresses
-> (2, '789 Park Ave', NULL, 'Kandy', 'Central', '30000', 'Shipping'),
-> (3, '321 Oak Lane', 'Floor 5', 'Jaffna', 'Northern', '40000', 'Billing'),
-> (4, '555 Beach Rd', NULL, 'Matara', 'Southern', '50000', 'Shipping'),
-> (5, '999 Hill St', 'Suite 100', 'Negombo', 'Western', '60000', 'Shipping');
Query OK, 6 rows affected (0.005 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from Address;
+-----+-----+-----+-----+-----+-----+-----+
| address_id | customer_id | address_line1 | address_line2 | city | state | zip_code | address_type |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | 1 | 123 Main St | Apt 4B | Colombo | Western | 10000 | Shipping |
| 2 | 1 | 456 Central Rd | NULL | Galle | Southern | 20000 | Billing |
| 3 | 2 | 789 Park Ave | NULL | Kandy | Central | 30000 | Shipping |
| 4 | 3 | 321 Oak Lane | Floor 5 | Jaffna | Northern | 40000 | Billing |
| 5 | 4 | 555 Beach Rd | NULL | Matara | Southern | 50000 | Shipping |
| 6 | 5 | 999 Hill St | Suite 100 | Negombo | Western | 60000 | Shipping |
+-----+-----+-----+-----+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> INSERT INTO ShoppingCart (creation_date, customer_id) VALUES
-> ('2024-06-01', 1),
-> ('2024-06-02', 2),
-> ('2024-06-03', 3),
-> ('2024-06-04', 4),
-> ('2024-06-05', 5),
-> ('2024-06-06', 6);
Query OK, 6 rows affected (0.003 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> |
```

Figure 23 Inserting Data to the ShoppingCart Table

```
MySQL Client (MariaDB 11.7.0) x + -

MariaDB [ecommerce_management_db]> select * from Address;
+-----+-----+-----+-----+-----+-----+-----+
| address_id | customer_id | address_line1 | address_line2 | city | state | zip_code | address_type |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | 1 | 123 Main St | Apt 4B | Colombo | Western | 10000 | Shipping |
| 2 | 1 | 456 Central Rd | NULL | Galle | Southern | 20000 | Billing |
| 3 | 2 | 789 Park Ave | NULL | Kandy | Central | 30000 | Shipping |
| 4 | 3 | 321 Oak Lane | Floor 5 | Jaffna | Northern | 40000 | Billing |
| 5 | 4 | 555 Beach Rd | NULL | Matara | Southern | 50000 | Shipping |
| 6 | 5 | 999 Hill St | Suite 100 | Negombo | Western | 60000 | Shipping |
+-----+-----+-----+-----+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> INSERT INTO ShoppingCart (creation_date, customer_id) VALUES
-> ('2024-06-01', 1),
-> ('2024-06-02', 2),
-> ('2024-06-03', 3),
-> ('2024-06-04', 4),
-> ('2024-06-05', 5),
-> ('2024-06-06', 6);
Query OK, 6 rows affected (0.003 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from ShoppingCart;
+-----+-----+-----+
| cart_id | creation_date | customer_id |
+-----+-----+-----+
| 1 | 2024-06-01 | 1 |
| 2 | 2024-06-02 | 2 |
| 3 | 2024-06-03 | 3 |
| 4 | 2024-06-04 | 4 |
| 5 | 2024-06-05 | 5 |
| 6 | 2024-06-06 | 6 |
+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 24 ShoppingCart Table with Inserted Data


```
MySQL Client (MariaDB 11.7.0) x + -
| 6 | 5 | 999 Hill St | Suite 100 | Negombo | Western | 68000 | Shipping |
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> INSERT INTO ShoppingCart (creation_date, customer_id) VALUES
-> ('2024-06-01', 1),
-> ('2024-06-02', 2),
-> ('2024-06-03', 3),
-> ('2024-06-04', 4),
-> ('2024-06-05', 5),
-> ('2024-06-06', 6);
Query OK, 6 rows affected (0.003 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from ShoppingCart;
+-----+-----+-----+
| cart_id | creation_date | customer_id |
+-----+-----+-----+
| 1 | 2024-06-01 | 1 |
| 2 | 2024-06-02 | 2 |
| 3 | 2024-06-03 | 3 |
| 4 | 2024-06-04 | 4 |
| 5 | 2024-06-05 | 5 |
| 6 | 2024-06-06 | 6 |
+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> INSERT INTO CartItem (cart_id, product_id, quantity) VALUES
-> (1, 1, 1), -- Cart 1 has Product 1 (Smartphone X)
-> (1, 3, 2), -- Cart 1 also has Product 3 (Wireless Earbuds), quantity 2
-> (2, 4, 5), -- Cart 2 has Product 4 (Cotton T-Shirt), quantity 5
-> (3, 2, 1), -- Cart 3 has Product 2 (Gaming Laptop Pro)
-> (4, 5, 1), -- Cart 4 has Product 5 (Running Shoes)
-> (5, 6, 2); -- Cart 5 has Product 6 (Bluetooth Speaker), quantity 2
Query OK, 6 rows affected (0.007 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> |
```

Figure 25 Inserting Data to the CartItem Table

```
MySQL Client (MariaDB 11.7.0) x + -

MariaDB [ecommerce_management_db]> select * from ShoppingCart;
+-----+-----+-----+
| cart_id | creation_date | customer_id |
+-----+-----+-----+
| 1 | 2024-06-01 | 1 |
| 2 | 2024-06-02 | 2 |
| 3 | 2024-06-03 | 3 |
| 4 | 2024-06-04 | 4 |
| 5 | 2024-06-05 | 5 |
| 6 | 2024-06-06 | 6 |
+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> INSERT INTO CartItem (cart_id, product_id, quantity) VALUES
-> (1, 1, 1), -- Cart 1 has Product 1 (Smartphone X)
-> (1, 3, 2), -- Cart 1 also has Product 3 (Wireless Earbuds), quantity 2
-> (2, 4, 5), -- Cart 2 has Product 4 (Cotton T-Shirt), quantity 5
-> (3, 2, 1), -- Cart 3 has Product 2 (Gaming Laptop Pro)
-> (4, 5, 1), -- Cart 4 has Product 5 (Running Shoes)
-> (5, 6, 2); -- Cart 5 has Product 6 (Bluetooth Speaker), quantity 2
Query OK, 6 rows affected (0.007 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from CartItem;
+-----+-----+-----+
| cart_id | product_id | quantity |
+-----+-----+-----+
| 1 | 1 | 1 |
| 1 | 3 | 2 |
| 2 | 4 | 5 |
| 3 | 2 | 1 |
| 4 | 5 | 1 |
| 5 | 6 | 2 |
+-----+-----+-----+
6 rows in set (0.000 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 26 CartItem Table with Inserted Data

```

MySQL Client (MariaDB 11.7.0) x + -
6 | 2024-06-06 | 6 |
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> INSERT INTO CartItem (cart_id, product_id, quantity) VALUES
--> (1, 1, 1), -- Cart 1 has Product 1 (Smartphone X)
--> (1, 3, 2), -- Cart 1 also has Product 3 (Wireless Earbuds), quantity 2
--> (2, 4, 5), -- Cart 2 has Product 4 (Cotton T-Shirt), quantity 5
--> (3, 2, 1), -- Cart 3 has Product 2 (Gaming Laptop Pro)
--> (4, 5, 1), -- Cart 4 has Product 5 (Running Shoes)
--> (5, 6, 2); -- Cart 5 has Product 6 (Bluetooth Speaker), quantity 2
Query OK, 6 rows affected (0.007 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from CartItem;
+-----+-----+-----+
| cart_id | product_id | quantity |
+-----+-----+-----+
| 1 | 1 | 1 |
| 1 | 3 | 2 |
| 2 | 4 | 5 |
| 3 | 2 | 1 |
| 4 | 5 | 1 |
| 5 | 6 | 2 |
+-----+-----+-----+
6 rows in set (0.000 sec)

MariaDB [ecommerce_management_db]> INSERT INTO 'Order' (order_date, total_amount, status, customer_id) VALUES
--> ('2024-06-10 09:30:00', 749.99, 'Paid', 1), -- Order for Customer 1
--> ('2024-06-11 14:22:15', 124.95, 'Shipped', 2), -- Order for Customer 2 (5 * 24.99 = 124.95)
--> ('2024-06-12 10:05:43', 1599.99, 'Delivered', 3), -- Order for Customer 3
--> ('2024-06-13 16:45:30', 89.99, 'Pending', 4), -- Order for Customer 4
--> ('2024-06-14 11:11:11', 119.98, 'Paid', 5), -- Order for Customer 5 (2 * 59.99 = 119.98)
--> ('2024-06-15 12:00:00', 0.00, 'Pending', 6); -- An empty "Pending" order for Customer 6
Query OK, 6 rows affected (0.006 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]>

```

Figure 27 Inserting Data to the Order Table

```

MySQL Client (MariaDB 11.7.0) x + -
+-----+-----+-----+
| cart_id | product_id | quantity |
+-----+-----+-----+
| 1 | 1 | 1 |
| 1 | 3 | 2 |
| 2 | 4 | 5 |
| 3 | 2 | 1 |
| 4 | 5 | 1 |
| 5 | 6 | 2 |
+-----+-----+-----+
6 rows in set (0.000 sec)

MariaDB [ecommerce_management_db]> INSERT INTO 'Order' (order_date, total_amount, status, customer_id) VALUES
--> ('2024-06-10 09:30:00', 749.99, 'Paid', 1), -- Order for Customer 1
--> ('2024-06-11 14:22:15', 124.95, 'Shipped', 2), -- Order for Customer 2 (5 * 24.99 = 124.95)
--> ('2024-06-12 10:05:43', 1599.99, 'Delivered', 3), -- Order for Customer 3
--> ('2024-06-13 16:45:30', 89.99, 'Pending', 4), -- Order for Customer 4
--> ('2024-06-14 11:11:11', 119.98, 'Paid', 5), -- Order for Customer 5 (2 * 59.99 = 119.98)
--> ('2024-06-15 12:00:00', 0.00, 'Pending', 6); -- An empty "Pending" order for Customer 6
Query OK, 6 rows affected (0.006 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from Order;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax
to use near 'Order' at line 1
MariaDB [ecommerce_management_db]> select * from 'Order' ;
+-----+-----+-----+-----+-----+
| order_id | order_date | total_amount | status | customer_id |
+-----+-----+-----+-----+-----+
| 1 | 2024-06-10 09:30:00 | 749.99 | Paid | 1 |
| 2 | 2024-06-11 14:22:15 | 124.95 | Shipped | 2 |
| 3 | 2024-06-12 10:05:43 | 1599.99 | Delivered | 3 |
| 4 | 2024-06-13 16:45:30 | 89.99 | Pending | 4 |
| 5 | 2024-06-14 11:11:11 | 119.98 | Paid | 5 |
| 6 | 2024-06-15 12:00:00 | 0.00 | Pending | 6 |
+-----+-----+-----+-----+-----+
6 rows in set (0.002 sec)

MariaDB [ecommerce_management_db]>

```

Figure 28 Order Table with Inserted Data


```

MySQL Client (MariaDB 11.7.0) x + -
MariaDB [ecommerce_management_db]> INSERT INTO `Order` (order_date, total_amount, status, customer_id) VALUES
-> ('2024-06-10 09:30:00', 749.99, 'Paid', 1), -- Order for Customer 1
-> ('2024-06-11 14:22:15', 124.95, 'Shipped', 2), -- Order for Customer 2 (5 * 24.99 = 124.95)
-> ('2024-06-12 10:05:43', 1599.99, 'Delivered', 3), -- Order for Customer 3
-> ('2024-06-13 16:45:30', 89.99, 'Pending', 4), -- Order for Customer 4
-> ('2024-06-14 11:11:11', 119.98, 'Paid', 5), -- Order for Customer 5 (2 * 59.99 = 119.98)
-> ('2024-06-15 12:00:00', 0.00, 'Pending', 6); -- An empty "Pending" order for Customer 6
Query OK, 6 rows affected (0.006 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from Order;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax
to use near 'Order' at line 1
MariaDB [ecommerce_management_db]> select * from `Order`;
+-----+-----+-----+-----+-----+
| order_id | order_date | total_amount | status | customer_id |
+-----+-----+-----+-----+-----+
| 1 | 2024-06-10 09:30:00 | 749.99 | Paid | 1 |
| 2 | 2024-06-11 14:22:15 | 124.95 | Shipped | 2 |
| 3 | 2024-06-12 10:05:43 | 1599.99 | Delivered | 3 |
| 4 | 2024-06-13 16:45:30 | 89.99 | Pending | 4 |
| 5 | 2024-06-14 11:11:11 | 119.98 | Paid | 5 |
| 6 | 2024-06-15 12:00:00 | 0.00 | Pending | 6 |
+-----+-----+-----+-----+-----+
6 rows in set (0.002 sec)

MariaDB [ecommerce_management_db]> INSERT INTO OrderItem (order_id, product_id, quantity, unit_price) VALUES
-> (1, 1, 1, 749.99), -- Order 1 contains 1x Smartphone X @ 749.99
-> (2, 4, 5, 24.99), -- Order 2 contains 5x Cotton T-Shirt @ 24.99
-> (3, 2, 1, 1599.99), -- Order 3 contains 1x Gaming Laptop Pro @ 1599.99
-> (4, 5, 1, 89.99), -- Order 4 contains 1x Running Shoes @ 89.99
-> (5, 6, 2, 59.99), -- Order 5 contains 2x Bluetooth Speaker @ 59.99
-> (1, 3, 1, 129.99); -- Order 1 also contains 1x Wireless Earbuds @ 129.99
Query OK, 6 rows affected (0.004 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]>

```

Figure 29 Inserting Data to the OrderItem Table

```

MySQL Client (MariaDB 11.7.0) x + -
to use near 'Order' at line 1
MariaDB [ecommerce_management_db]> select * from `Order`;
+-----+-----+-----+-----+-----+
| order_id | order_date | total_amount | status | customer_id |
+-----+-----+-----+-----+-----+
| 1 | 2024-06-10 09:30:00 | 749.99 | Paid | 1 |
| 2 | 2024-06-11 14:22:15 | 124.95 | Shipped | 2 |
| 3 | 2024-06-12 10:05:43 | 1599.99 | Delivered | 3 |
| 4 | 2024-06-13 16:45:30 | 89.99 | Pending | 4 |
| 5 | 2024-06-14 11:11:11 | 119.98 | Paid | 5 |
| 6 | 2024-06-15 12:00:00 | 0.00 | Pending | 6 |
+-----+-----+-----+-----+-----+
6 rows in set (0.002 sec)

MariaDB [ecommerce_management_db]> INSERT INTO OrderItem (order_id, product_id, quantity, unit_price) VALUES
-> (1, 1, 1, 749.99), -- Order 1 contains 1x Smartphone X @ 749.99
-> (2, 4, 5, 24.99), -- Order 2 contains 5x Cotton T-Shirt @ 24.99
-> (3, 2, 1, 1599.99), -- Order 3 contains 1x Gaming Laptop Pro @ 1599.99
-> (4, 5, 1, 89.99), -- Order 4 contains 1x Running Shoes @ 89.99
-> (5, 6, 2, 59.99), -- Order 5 contains 2x Bluetooth Speaker @ 59.99
-> (1, 3, 1, 129.99); -- Order 1 also contains 1x Wireless Earbuds @ 129.99
Query OK, 6 rows affected (0.004 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from OrderItem;
+-----+-----+-----+-----+
| order_id | product_id | quantity | unit_price |
+-----+-----+-----+-----+
| 1 | 1 | 1 | 749.99 |
| 1 | 3 | 1 | 129.99 |
| 2 | 4 | 5 | 24.99 |
| 3 | 2 | 1 | 1599.99 |
| 4 | 5 | 1 | 89.99 |
| 5 | 6 | 2 | 59.99 |
+-----+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]>

```

Figure 30 OrderItem Table with Inserted Data

```
MySQL Client (MariaDB 11.7.0)
-> (2, 4, 5, 24.99), -- Order 2 contains 5x Cotton T-Shirt @ 24.99
-> (3, 2, 1, 1599.99), -- Order 3 contains 1x Gaming Laptop Pro @ 1599.99
-> (4, 5, 1, 89.99), -- Order 4 contains 1x Running Shoes @ 89.99
-> (5, 6, 2, 59.99), -- Order 5 contains 2x Bluetooth Speaker @ 59.99
-> (1, 3, 1, 129.99); -- Order 1 also contains 1x Wireless Earbuds @ 129.99
Query OK, 6 rows affected (0.004 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from OrderItem;
+-----+-----+-----+-----+
| order_id | product_id | quantity | unit_price |
+-----+-----+-----+-----+
| 1 | 3 | 1 | 129.99 |
| 1 | 3 | 1 | 129.99 |
| 2 | 4 | 5 | 24.99 |
| 3 | 2 | 1 | 1599.99 |
| 4 | 5 | 1 | 89.99 |
| 5 | 6 | 2 | 59.99 |
+-----+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> UPDATE `Order` SET total_amount = 879.98 WHERE order_id = 1;
Query OK, 1 row affected (0.008 sec)
Rows matched: 1 Changed: 1 Warnings: 0

MariaDB [ecommerce_management_db]> INSERT INTO Payment (payment_method, amount, transaction_date, status, order_id) VALUES
-> ('Credit Card', 879.98, '2024-06-10 09:31:05', 'Success', 1),
-> ('PayPal', 124.95, '2024-06-11 14:25:00', 'Success', 2),
-> ('Bank Transfer', 1599.99, '2024-06-12 10:06:20', 'Success', 3),
-> ('Credit Card', 89.99, '2024-06-13 16:46:00', 'Failed', 4), -- This payment failed
-> ('Debit Card', 119.98, '2024-06-14 11:12:00', 'Success', 5),
-> -- Order 6 is still pending, so it has no payment record yet.
-> -- We'll insert a 6th payment for a different order to meet the requirement.
-> ('Cash on Delivery', 0.00, '2024-06-15 12:00:00', 'Success', 6);
Query OK, 6 rows affected (0.005 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]>
```

Figure 31 Inserting Data to the Payment Table

```
MySQL Client (MariaDB 11.7.0)
+-----+-----+-----+-----+
| 1 | 3 | 1 | 129.99 |
| 2 | 4 | 5 | 24.99 |
| 3 | 2 | 1 | 1599.99 |
| 4 | 5 | 1 | 89.99 |
| 5 | 6 | 2 | 59.99 |
+-----+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> UPDATE `Order` SET total_amount = 879.98 WHERE order_id = 1;
Query OK, 1 row affected (0.008 sec)
Rows matched: 1 Changed: 1 Warnings: 0

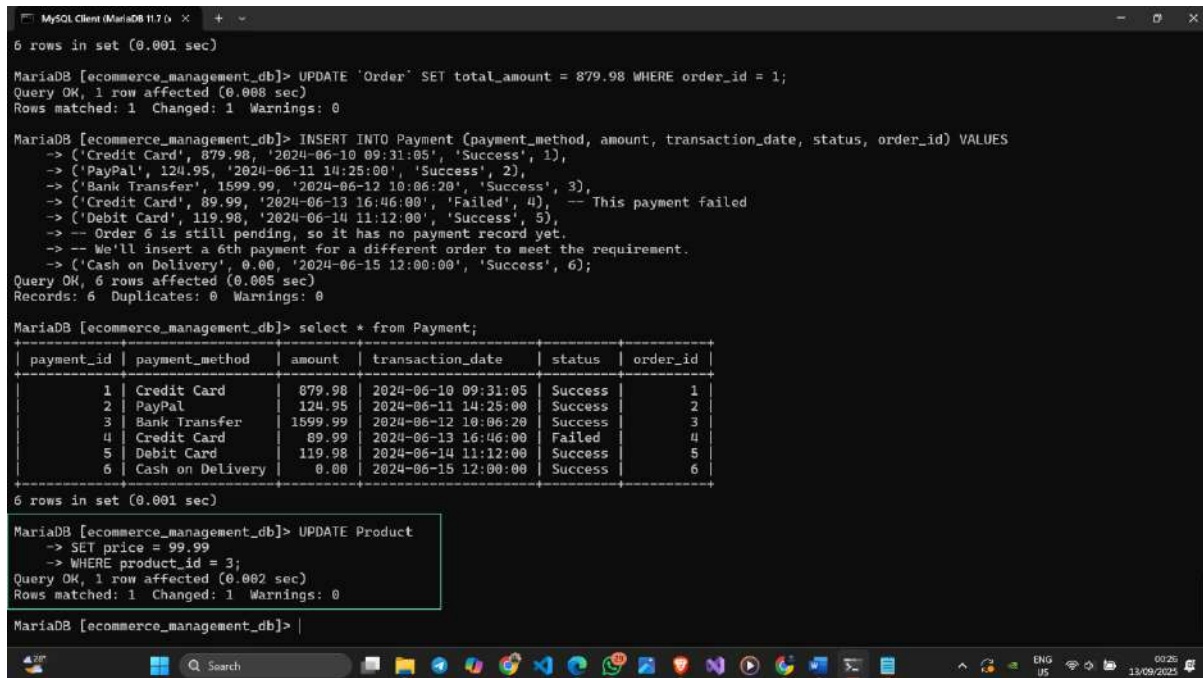
MariaDB [ecommerce_management_db]> INSERT INTO Payment (payment_method, amount, transaction_date, status, order_id) VALUES
-> ('Credit Card', 879.98, '2024-06-10 09:31:05', 'Success', 1),
-> ('PayPal', 124.95, '2024-06-11 14:25:00', 'Success', 2),
-> ('Bank Transfer', 1599.99, '2024-06-12 10:06:20', 'Success', 3),
-> ('Credit Card', 89.99, '2024-06-13 16:46:00', 'Failed', 4), -- This payment failed
-> ('Debit Card', 119.98, '2024-06-14 11:12:00', 'Success', 5),
-> -- Order 6 is still pending, so it has no payment record yet.
-> -- We'll insert a 6th payment for a different order to meet the requirement.
-> ('Cash on Delivery', 0.00, '2024-06-15 12:00:00', 'Success', 6);
Query OK, 6 rows affected (0.005 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from Payment;
+-----+-----+-----+-----+-----+-----+
| payment_id | payment_method | amount | transaction_date | status | order_id |
+-----+-----+-----+-----+-----+-----+
| 1 | Credit Card | 879.98 | 2024-06-10 09:31:05 | Success | 1 |
| 2 | PayPal | 124.95 | 2024-06-11 14:25:00 | Success | 2 |
| 3 | Bank Transfer | 1599.99 | 2024-06-12 10:06:20 | Success | 3 |
| 4 | Credit Card | 89.99 | 2024-06-13 16:46:00 | Failed | 4 |
| 5 | Debit Card | 119.98 | 2024-06-14 11:12:00 | Success | 5 |
| 6 | Cash on Delivery | 0.00 | 2024-06-15 12:00:00 | Success | 6 |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]>
```

Figure 32 Payment Table with Inserted Data

4.4 Data Modification (UPDATE Operations)



MySQL Client (MariaDB 11.7.0)

```
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> UPDATE `Order` SET total_amount = 879.98 WHERE order_id = 1;
Query OK, 1 row affected (0.008 sec)
Rows matched: 1 Changed: 1 Warnings: 0

MariaDB [ecommerce_management_db]> INSERT INTO Payment (payment_method, amount, transaction_date, status, order_id) VALUES
-> ('Credit Card', 879.98, '2024-06-10 09:31:05', 'Success', 1),
-> ('PayPal', 124.95, '2024-06-11 14:25:00', 'Success', 2),
-> ('Bank Transfer', 1599.99, '2024-06-12 10:06:20', 'Success', 3),
-> ('Credit Card', 89.99, '2024-06-13 16:46:00', 'Failed', 4), -- This payment failed
-> ('Debit Card', 119.98, '2024-06-14 11:12:00', 'Success', 5),
-> -- Order 6 is still pending, so it has no payment record yet.
-> -- We'll insert a 6th payment for a different order to meet the requirement.
-> ('Cash on Delivery', 0.00, '2024-06-15 12:00:00', 'Success', 6);
Query OK, 6 rows affected (0.005 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from Payment;
```

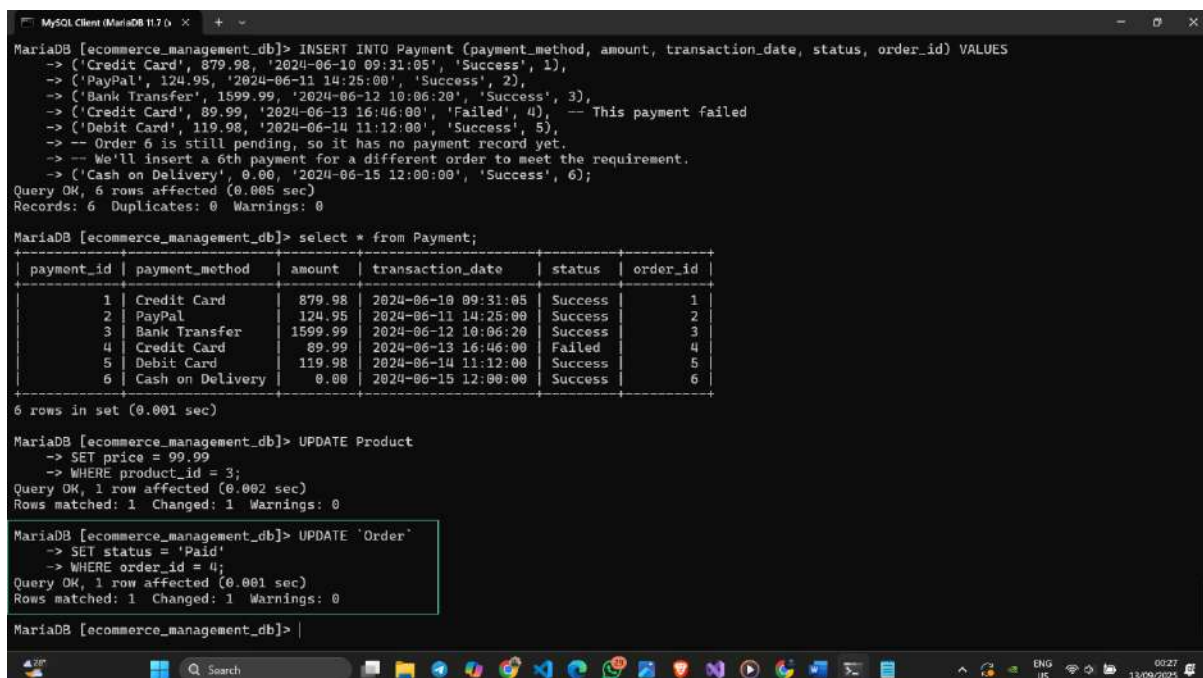
payment_id	payment_method	amount	transaction_date	status	order_id
1	Credit Card	879.98	2024-06-10 09:31:05	Success	1
2	PayPal	124.95	2024-06-11 14:25:00	Success	2
3	Bank Transfer	1599.99	2024-06-12 10:06:20	Success	3
4	Credit Card	89.99	2024-06-13 16:46:00	Failed	4
5	Debit Card	119.98	2024-06-14 11:12:00	Success	5
6	Cash on Delivery	0.00	2024-06-15 12:00:00	Success	6

```
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> UPDATE Product
-> SET price = 99.99
-> WHERE product_id = 3;
Query OK, 1 row affected (0.002 sec)
Rows matched: 1 Changed: 1 Warnings: 0

MariaDB [ecommerce_management_db]>
```

Figure 33 Update a Product's Price



MySQL Client (MariaDB 11.7.0)

```
MariaDB [ecommerce_management_db]> INSERT INTO Payment (payment_method, amount, transaction_date, status, order_id) VALUES
-> ('Credit Card', 879.98, '2024-06-10 09:31:05', 'Success', 1),
-> ('PayPal', 124.95, '2024-06-11 14:25:00', 'Success', 2),
-> ('Bank Transfer', 1599.99, '2024-06-12 10:06:20', 'Success', 3),
-> ('Credit Card', 89.99, '2024-06-13 16:46:00', 'Failed', 4), -- This payment failed
-> ('Debit Card', 119.98, '2024-06-14 11:12:00', 'Success', 5),
-> -- Order 6 is still pending, so it has no payment record yet.
-> -- We'll insert a 6th payment for a different order to meet the requirement.
-> ('Cash on Delivery', 0.00, '2024-06-15 12:00:00', 'Success', 6);
Query OK, 6 rows affected (0.005 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from Payment;
```

payment_id	payment_method	amount	transaction_date	status	order_id
1	Credit Card	879.98	2024-06-10 09:31:05	Success	1
2	PayPal	124.95	2024-06-11 14:25:00	Success	2
3	Bank Transfer	1599.99	2024-06-12 10:06:20	Success	3
4	Credit Card	89.99	2024-06-13 16:46:00	Failed	4
5	Debit Card	119.98	2024-06-14 11:12:00	Success	5
6	Cash on Delivery	0.00	2024-06-15 12:00:00	Success	6

```
6 rows in set (0.001 sec)

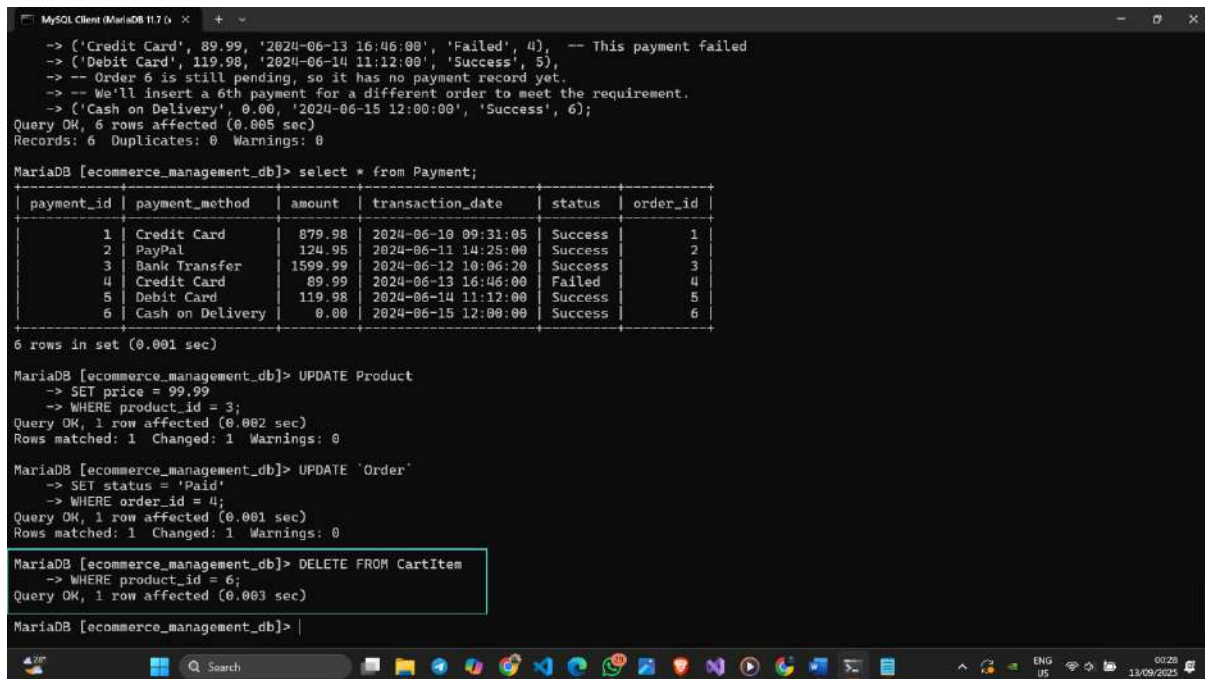
MariaDB [ecommerce_management_db]> UPDATE Product
-> SET price = 99.99
-> WHERE product_id = 3;
Query OK, 1 row affected (0.002 sec)
Rows matched: 1 Changed: 1 Warnings: 0

MariaDB [ecommerce_management_db]> UPDATE `Order`
-> SET status = 'Paid'
-> WHERE order_id = 4;
Query OK, 1 row affected (0.001 sec)
Rows matched: 1 Changed: 1 Warnings: 0

MariaDB [ecommerce_management_db]>
```

Figure 34 Update an Order's Status

4.5 Data Deletion (DELETE Operations)



The screenshot shows a MySQL Client window with the following content:

```
-- ('Credit Card', 89.99, '2024-06-13 16:46:00', 'Failed', 4), -- This payment failed
-- ('Debit Card', 119.98, '2024-06-14 11:12:00', 'Success', 5),
-- -- Order 6 is still pending, so it has no payment record yet.
-- -- We'll insert a 6th payment for a different order to meet the requirement.
-- ('Cash on Delivery', 0.00, '2024-06-15 12:00:00', 'Success', 6);
Query OK, 6 rows affected (0.005 sec)
Records: 6 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> select * from Payment;
+-----+-----+-----+-----+-----+-----+
| payment_id | payment_method | amount | transaction_date | status | order_id |
+-----+-----+-----+-----+-----+-----+
| 1 | Credit Card | 879.98 | 2024-06-10 09:31:05 | Success | 1 |
| 2 | PayPal | 124.95 | 2024-06-11 14:25:00 | Success | 2 |
| 3 | Bank Transfer | 1599.99 | 2024-06-12 10:06:20 | Success | 3 |
| 4 | Credit Card | 89.99 | 2024-06-13 16:46:00 | Failed | 4 |
| 5 | Debit Card | 119.98 | 2024-06-14 11:12:00 | Success | 5 |
| 6 | Cash on Delivery | 0.00 | 2024-06-15 12:00:00 | Success | 6 |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.001 sec)

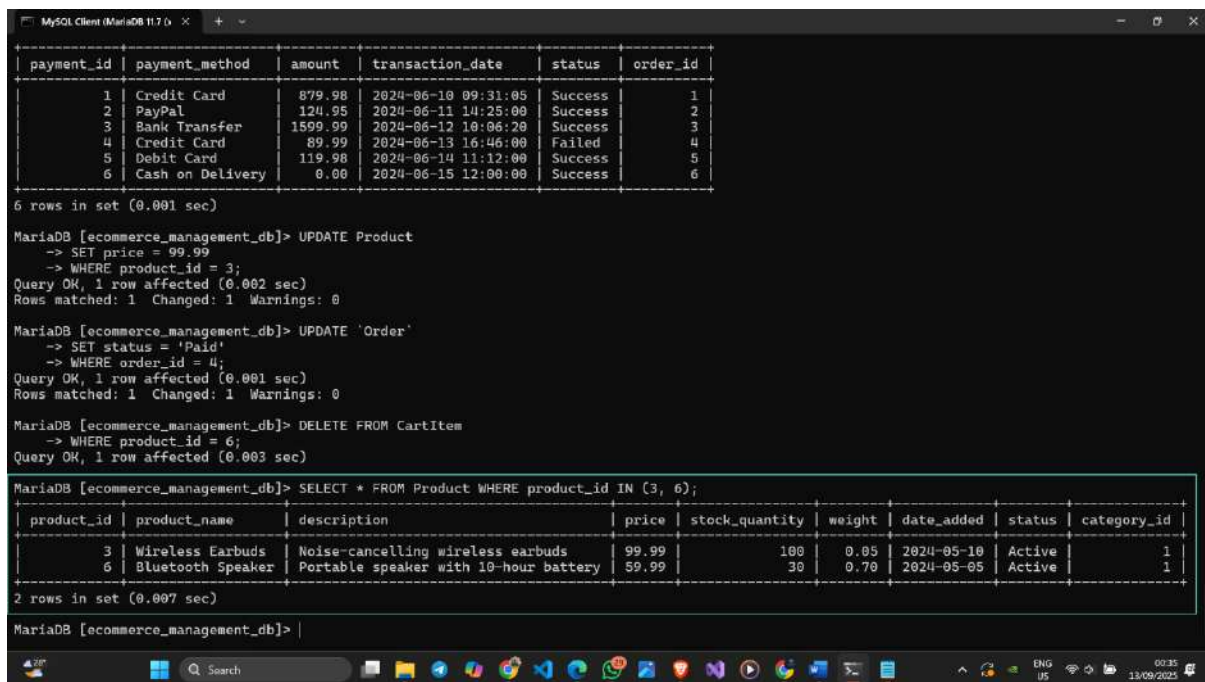
MariaDB [ecommerce_management_db]> UPDATE Product
--> SET price = 99.99
--> WHERE product_id = 3;
Query OK, 1 row affected (0.002 sec)
Rows matched: 1 Changed: 1 Warnings: 0

MariaDB [ecommerce_management_db]> UPDATE 'Order'
--> SET status = 'Paid'
--> WHERE order_id = 4;
Query OK, 1 row affected (0.001 sec)
Rows matched: 1 Changed: 1 Warnings: 0

MariaDB [ecommerce_management_db]> DELETE FROM CartItem
--> WHERE product_id = 6;
Query OK, 1 row affected (0.003 sec)

MariaDB [ecommerce_management_db]>
```

Figure 35 Delete from CartItem Table



The screenshot shows a MySQL Client window with the following content:

```
payment_id | payment_method | amount | transaction_date | status | order_id |
+-----+-----+-----+-----+-----+-----+
| 1 | Credit Card | 879.98 | 2024-06-10 09:31:05 | Success | 1 |
| 2 | PayPal | 124.95 | 2024-06-11 14:25:00 | Success | 2 |
| 3 | Bank Transfer | 1599.99 | 2024-06-12 10:06:20 | Success | 3 |
| 4 | Credit Card | 89.99 | 2024-06-13 16:46:00 | Failed | 4 |
| 5 | Debit Card | 119.98 | 2024-06-14 11:12:00 | Success | 5 |
| 6 | Cash on Delivery | 0.00 | 2024-06-15 12:00:00 | Success | 6 |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> UPDATE Product
--> SET price = 99.99
--> WHERE product_id = 3;
Query OK, 1 row affected (0.002 sec)
Rows matched: 1 Changed: 1 Warnings: 0

MariaDB [ecommerce_management_db]> UPDATE 'Order'
--> SET status = 'Paid'
--> WHERE order_id = 4;
Query OK, 1 row affected (0.001 sec)
Rows matched: 1 Changed: 1 Warnings: 0

MariaDB [ecommerce_management_db]> DELETE FROM CartItem
--> WHERE product_id = 6;
Query OK, 1 row affected (0.003 sec)

MariaDB [ecommerce_management_db]> SELECT * FROM Product WHERE product_id IN (3, 6);
+-----+-----+-----+-----+-----+-----+-----+-----+
| product_id | product_name | description | price | stock_quantity | weight | date_added | status | category_id |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 3 | Wireless Earbuds | Noise-cancelling wireless earbuds | 99.99 | 100 | 0.05 | 2024-05-10 | Active | 1 |
| 6 | Bluetooth Speaker | Portable speaker with 10-hour battery | 59.99 | 30 | 0.70 | 2024-05-05 | Active | 1 |
+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.007 sec)

MariaDB [ecommerce_management_db]>
```

Figure 36 Verify the first Update operation

```
MySQL Client (MariaDB 11.7.0) x + -
| 6 | Cash on Delivery | 0.00 | 2024-06-15 12:00:00 | Success | 6 |
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> UPDATE Product
-> SET price = 99.99
-> WHERE product_id = 3;
Query OK, 1 row affected (0.002 sec)
Rows matched: 1 Changed: 1 Warnings: 0

MariaDB [ecommerce_management_db]> UPDATE `Order`
-> SET status = 'Paid'
-> WHERE order_id = 4;
Query OK, 1 row affected (0.001 sec)
Rows matched: 1 Changed: 1 Warnings: 0

MariaDB [ecommerce_management_db]> DELETE FROM CartItem
-> WHERE product_id = 6;
Query OK, 1 row affected (0.003 sec)

MariaDB [ecommerce_management_db]> SELECT * FROM Product WHERE product_id IN (3, 6);
+-----+-----+-----+-----+-----+-----+-----+-----+
| product_id | product_name | description | price | stock_quantity | weight | date_added | status | category_id |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 3 | Wireless Earbuds | Noise-cancelling wireless earbuds | 99.99 | 100 | 0.05 | 2024-05-10 | Active | 1 |
| 6 | Bluetooth Speaker | Portable speaker with 10-hour battery | 59.99 | 30 | 0.70 | 2024-05-05 | Active | 1 |
+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.007 sec)

MariaDB [ecommerce_management_db]> SELECT * FROM `Order` WHERE order_id = 4;
+-----+-----+-----+-----+-----+
| order_id | order_date | total_amount | status | customer_id |
+-----+-----+-----+-----+-----+
| 4 | 2024-06-13 16:45:30 | 89.99 | Paid | 4 |
+-----+-----+-----+-----+-----+
1 row in set (0.003 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 37 Verify the second Update operation

```
MySQL Client (MariaDB 11.7.0) x + -

MariaDB [ecommerce_management_db]> UPDATE Product
-> SET price = 99.99
-> WHERE product_id = 3;
Query OK, 1 row affected (0.002 sec)
Rows matched: 1 Changed: 1 Warnings: 0

MariaDB [ecommerce_management_db]> UPDATE `Order`
-> SET status = 'Paid'
-> WHERE order_id = 4;
Query OK, 1 row affected (0.001 sec)
Rows matched: 1 Changed: 1 Warnings: 0

MariaDB [ecommerce_management_db]> DELETE FROM CartItem
-> WHERE product_id = 6;
Query OK, 1 row affected (0.003 sec)

MariaDB [ecommerce_management_db]> SELECT * FROM Product WHERE product_id IN (3, 6);
+-----+-----+-----+-----+-----+-----+-----+-----+
| product_id | product_name | description | price | stock_quantity | weight | date_added | status | category_id |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 3 | Wireless Earbuds | Noise-cancelling wireless earbuds | 99.99 | 100 | 0.05 | 2024-05-10 | Active | 1 |
| 6 | Bluetooth Speaker | Portable speaker with 10-hour battery | 59.99 | 30 | 0.70 | 2024-05-05 | Active | 1 |
+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.007 sec)

MariaDB [ecommerce_management_db]> SELECT * FROM `Order` WHERE order_id = 4;
+-----+-----+-----+-----+-----+
| order_id | order_date | total_amount | status | customer_id |
+-----+-----+-----+-----+-----+
| 4 | 2024-06-13 16:45:30 | 89.99 | Paid | 4 |
+-----+-----+-----+-----+-----+
1 row in set (0.003 sec)

MariaDB [ecommerce_management_db]> SELECT * FROM CartItem WHERE product_id = 6;
Empty set (0.003 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 38 Verify the Delete operation

CHAPTER 5 – TRANSACTIONS

5.1 Simple Queries

5.1.1 Select Operation

```
MySQL Client (MariaDB 11.7.0) x + -
MariaDB [ecommerce_management_db]> DELETE FROM CartItem
-> WHERE product_id = 6;
Query OK, 1 row affected (0.003 sec)

MariaDB [ecommerce_management_db]> SELECT * FROM Product WHERE product_id IN (3, 6);
+-----+-----+-----+-----+-----+-----+-----+-----+
| product_id | product_name | description | price | stock_quantity | weight | date_added | status | category_id |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 3 | Wireless Earbuds | Noise-cancelling wireless earbuds | 99.99 | 100 | 0.05 | 2024-05-10 | Active | 1 |
| 6 | Bluetooth Speaker | Portable speaker with 10-hour battery | 59.99 | 30 | 0.70 | 2024-05-05 | Active | 1 |
+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.007 sec)

MariaDB [ecommerce_management_db]> SELECT * FROM `Order` WHERE order_id = 4;
+-----+-----+-----+-----+-----+
| order_id | order_date | total_amount | status | customer_id |
+-----+-----+-----+-----+-----+
| 4 | 2024-06-13 16:45:30 | 89.99 | Paid | 4 |
+-----+-----+-----+-----+-----+
1 row in set (0.003 sec)

MariaDB [ecommerce_management_db]> SELECT * FROM CartItem WHERE product_id = 6;
Empty set (0.003 sec)

MariaDB [ecommerce_management_db]> SELECT * FROM Customer;
+-----+-----+-----+-----+-----+-----+-----+
| customer_id | email | password_hash | first_name | last_name | phone | registration_date |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | alice.johnson@email.com | hash1 | Alice | Johnson | 123-456-7890 | 2024-01-15 |
| 2 | bob.smith@email.com | hash2 | Bob | Smith | 123-456-7891 | 2024-02-20 |
| 3 | charlie.brown@email.com | hash3 | Charlie | Brown | NULL | 2024-03-05 |
| 4 | diana.prince@email.com | hash4 | Diana | Prince | 123-456-7893 | 2024-03-10 |
| 5 | evan.lee@email.com | hash5 | Evan | Lee | 123-456-7894 | 2024-04-22 |
| 6 | fiona.campbell@email.com | hash6 | Fiona | Campbell | 123-456-7895 | 2024-05-30 |
+-----+-----+-----+-----+-----+-----+-----+
6 rows in set (0.004 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 39 Select all columns from Customer Table

5.1.2 Project Operation

```
MySQL Client (MariaDB 11.7.0) x + -
MariaDB [ecommerce_management_db]> SELECT * FROM `Order` WHERE order_id = 4;
+-----+-----+-----+-----+-----+
| order_id | order_date | total_amount | status | customer_id |
+-----+-----+-----+-----+-----+
| 4 | 2024-06-13 16:45:30 | 89.99 | Paid | 4 |
+-----+-----+-----+-----+-----+
1 row in set (0.003 sec)

MariaDB [ecommerce_management_db]> SELECT * FROM CartItem WHERE product_id = 6;
Empty set (0.003 sec)

MariaDB [ecommerce_management_db]> SELECT * FROM Customer;
+-----+-----+-----+-----+-----+-----+-----+
| customer_id | email | password_hash | first_name | last_name | phone | registration_date |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | alice.johnson@email.com | hash1 | Alice | Johnson | 123-456-7890 | 2024-01-15 |
| 2 | bob.smith@email.com | hash2 | Bob | Smith | 123-456-7891 | 2024-02-20 |
| 3 | charlie.brown@email.com | hash3 | Charlie | Brown | NULL | 2024-03-05 |
| 4 | diana.prince@email.com | hash4 | Diana | Prince | 123-456-7893 | 2024-03-10 |
| 5 | evan.lee@email.com | hash5 | Evan | Lee | 123-456-7894 | 2024-04-22 |
| 6 | fiona.campbell@email.com | hash6 | Fiona | Campbell | 123-456-7895 | 2024-05-30 |
+-----+-----+-----+-----+-----+-----+-----+
6 rows in set (0.004 sec)

MariaDB [ecommerce_management_db]> SELECT first_name, last_name, email FROM Customer;
+-----+-----+-----+
| first_name | last_name | email |
+-----+-----+-----+
| Alice | Johnson | alice.johnson@email.com |
| Bob | Smith | bob.smith@email.com |
| Charlie | Brown | charlie.brown@email.com |
| Diana | Prince | diana.prince@email.com |
| Evan | Lee | evan.lee@email.com |
| Fiona | Campbell | fiona.campbell@email.com |
+-----+-----+-----+
6 rows in set (0.003 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 40 Select specific columns from Customer Table

5.1.3 Cartesian Product

The screenshot shows the MySQL Client interface with the following queries and results:

```
0 rows in set (0.004 sec)

MariaDB [e-commerce_management_db]> SELECT first_name, last_name, email FROM Customer;
```

first_name	last_name	email
Alice	Johansen	alice.johansen@gmail.com
Bob	Smith	bob.smith@gmail.com
Charlie	Brown	charlie.brown@gmail.com
Diana	Lee	diana.lee@gmail.com
Eve	Kim	eve.kim@gmail.com
Fiona	Campbell	fiona.campbell@gmail.com

```
6 rows in set (0.002 sec)

MariaDB [e-commerce_management_db]> SELECT * FROM category CROSS JOIN product;
```

category_id	category_name	description	parent_category_id	product_id	product_name	description	price	stock_quantity	weight	date_added	status	category_id
1	Electronics	Gadgets and devices	MALL	1	Smartphone X	Latest model with high-resolution camera	799.99	50	0.15	2024-01-15	Active	1
2	Computers	Desktop and laptop computers	MALL	2	Smartphone X	Latest model with high-resolution camera	799.99	50	0.15	2024-01-15	Active	2
3	Laptops	Portable computers	MALL	3	Smartphone X	Latest model with high-resolution camera	799.99	50	0.15	2024-01-15	Active	3
4	Smartphones	Mobile phones	MALL	4	Smartphone X	Latest model with high-resolution camera	799.99	50	0.15	2024-01-15	Active	4
5	Fashion	Clothing and accessories	MALL	5	Smartphone X	Latest model with high-resolution camera	799.99	50	0.15	2024-01-15	Active	5
6	Men's Clothing	Clothing for men	MALL	6	Smartphone X	Latest model with high-resolution camera	799.99	50	0.15	2024-01-15	Active	6
7	Electronics	Gadgets and devices	MALL	7	Gaming Laptop Pro	Powerful laptop for gaming and productivity	1599.99	15	2.50	2024-01-15	Active	7
8	Computers	Desktop and laptop computers	MALL	8	Gaming Laptop Pro	Powerful laptop for gaming and productivity	1599.99	15	2.50	2024-01-15	Active	8
9	Laptops	Portable computers	MALL	9	Gaming Laptop Pro	Powerful laptop for gaming and productivity	1599.99	15	2.50	2024-01-15	Active	9
10	Smartphones	Mobile phones	MALL	10	Gaming Laptop Pro	Powerful laptop for gaming and productivity	1599.99	15	2.50	2024-01-15	Active	10
11	Fashion	Clothing and accessories	MALL	11	Gaming Laptop Pro	Powerful laptop for gaming and productivity	1599.99	15	2.50	2024-01-15	Active	11
12	Men's Clothing	Clothing for men	MALL	12	Gaming Laptop Pro	Powerful laptop for gaming and productivity	1599.99	15	2.50	2024-01-15	Active	12
13	Electronics	Gadgets and devices	MALL	13	Wireless Earbuds	Water-resistant wireless earbuds	99.99	100	0.05	2024-01-15	Active	13
14	Computers	Desktop and laptop computers	MALL	14	Wireless Earbuds	Water-resistant wireless earbuds	99.99	100	0.05	2024-01-15	Active	14
15	Laptops	Portable computers	MALL	15	Wireless Earbuds	Water-resistant wireless earbuds	99.99	100	0.05	2024-01-15	Active	15
16	Smartphones	Mobile phones	MALL	16	Wireless Earbuds	Water-resistant wireless earbuds	99.99	100	0.05	2024-01-15	Active	16
17	Fashion	Clothing and accessories	MALL	17	Wireless Earbuds	Water-resistant wireless earbuds	99.99	100	0.05	2024-01-15	Active	17
18	Men's Clothing	Clothing for men	MALL	18	Wireless Earbuds	Water-resistant wireless earbuds	99.99	100	0.05	2024-01-15	Active	18
19	Electronics	Gadgets and devices	MALL	19	Cotton T-Shirt	100% cotton, comfortable fit	24.99	200	0.10	2024-01-15	Active	19
20	Computers	Desktop and laptop computers	MALL	20	Cotton T-Shirt	100% cotton, comfortable fit	24.99	200	0.10	2024-01-15	Active	20
21	Laptops	Portable computers	MALL	21	Cotton T-Shirt	100% cotton, comfortable fit	24.99	200	0.10	2024-01-15	Active	21
22	Smartphones	Mobile phones	MALL	22	Cotton T-Shirt	100% cotton, comfortable fit	24.99	200	0.10	2024-01-15	Active	22
23	Fashion	Clothing and accessories	MALL	23	Cotton T-Shirt	100% cotton, comfortable fit	24.99	200	0.10	2024-01-15	Active	23
24	Men's Clothing	Clothing for men	MALL	24	Cotton T-Shirt	100% cotton, comfortable fit	24.99	200	0.10	2024-01-15	Active	24
25	Electronics	Gadgets and devices	MALL	25	Men's Running Shoes	Lightweight and breathable for running	89.99	75	0.80	2024-01-15	Active	25
26	Computers	Desktop and laptop computers	MALL	26	Men's Running Shoes	Lightweight and breathable for running	89.99	75	0.80	2024-01-15	Active	26
27	Laptops	Portable computers	MALL	27	Men's Running Shoes	Lightweight and breathable for running	89.99	75	0.80	2024-01-15	Active	27
28	Smartphones	Mobile phones	MALL	28	Men's Running Shoes	Lightweight and breathable for running	89.99	75	0.80	2024-01-15	Active	28
29	Fashion	Clothing and accessories	MALL	29	Men's Running Shoes	Lightweight and breathable for running	89.99	75	0.80	2024-01-15	Active	29
30	Men's Clothing	Clothing for men	MALL	30	Men's Running Shoes	Lightweight and breathable for running	89.99	75	0.80	2024-01-15	Active	30
31	Electronics	Gadgets and devices	MALL	31	Bluetooth Speaker	Portable speaker with 10-hour battery	59.99	100	0.10	2024-01-15	Active	31
32	Computers	Desktop and laptop computers	MALL	32	Bluetooth Speaker	Portable speaker with 10-hour battery	59.99	100	0.10	2024-01-15	Active	32
33	Laptops	Portable computers	MALL	33	Bluetooth Speaker	Portable speaker with 10-hour battery	59.99	100	0.10	2024-01-15	Active	33
34	Smartphones	Mobile phones	MALL	34	Bluetooth Speaker	Portable speaker with 10-hour battery	59.99	100	0.10	2024-01-15	Active	34
35	Fashion	Clothing and accessories	MALL	35	Bluetooth Speaker	Portable speaker with 10-hour battery	59.99	100	0.10	2024-01-15	Active	35
36	Men's Clothing	Clothing for men	MALL	36	Bluetooth Speaker	Portable speaker with 10-hour battery	59.99	100	0.10	2024-01-15	Active	36

```
36 rows in set (0.086 sec)

MariaDB [e-commerce_management_db]>
```

Figure 41 Cartesian product between Category and Product Tables

5.1.4 Create View

The screenshot shows the MySQL Client interface with the following queries and results:

```
36 rows in set (0.086 sec)

MariaDB [e-commerce_management_db]> CREATE VIEW ActiveProducts AS
-> SELECT p.product_id, p.product_name, p.price, c.category_name
-> FROM Product p
-> JOIN Category c ON p.category_id = c.category_id
-> WHERE p.status = 'Active';
Query OK, 0 rows affected (0.021 sec)

MariaDB [e-commerce_management_db]> SELECT * FROM ActiveProducts;
```

product_id	product_name	price	category_name
1	Smartphone X	799.99	Smartphones
2	Gaming Laptop Pro	1599.99	Laptops
3	Wireless Earbuds	99.99	Electronics
4	Cotton T-Shirt	24.99	Men's Clothing
5	Men's Running Shoes	89.99	Men's Clothing
6	Bluetooth Speaker	59.99	Electronics

```
6 rows in set (0.021 sec)

MariaDB [e-commerce_management_db]>
```

Figure 42 View for active products with their categories

5.1.5 Rename Operation

The screenshot shows a MySQL Client window with the following content:

```
MySQL Client (MariaDB 11.7.0) x + -
```

product_id	product_name	price	category_name
1	Smartphone X	749.99	Smartphones
2	Gaming Laptop Pro	1599.99	Laptops
3	Wireless Earbuds	99.99	Electronics
4	Cotton T-Shirt	24.99	Men's Clothing
5	Men's Running Shoes	89.99	Men's Clothing
6	Bluetooth Speaker	59.99	Electronics

36 rows in set (0.006 sec)

```
MariaDB [ecommerce_management_db]> CREATE VIEW ActiveProducts AS
-> SELECT p.product_id, p.product_name, p.price, c.category_name
-> FROM Product p
-> JOIN Category c ON p.category_id = c.category_id
-> WHERE p.status = 'Active';
Query OK, 0 rows affected (0.021 sec)
```

```
MariaDB [ecommerce_management_db]> SELECT * FROM ActiveProducts;
```

product_id	product_name	price	category_name
1	Smartphone X	749.99	Smartphones
2	Gaming Laptop Pro	1599.99	Laptops
3	Wireless Earbuds	99.99	Electronics
4	Cotton T-Shirt	24.99	Men's Clothing
5	Men's Running Shoes	89.99	Men's Clothing
6	Bluetooth Speaker	59.99	Electronics

6 rows in set (0.021 sec)

```
MariaDB [ecommerce_management_db]> SELECT
-> customer_id AS "ID",
-> first_name AS "First Name",
-> last_name AS "Last Name",
-> email AS "Email Address"
-> FROM Customer;
```

ID	First Name	Last Name	Email Address
1	Alice	Johnson	alice.johnson@email.com
2	Bob	Smith	bob.smith@email.com
3	Charlie	Brown	charlie.brown@email.com
4	Diana	Prince	diana.prince@email.com
5	Evan	Lee	evan.lee@email.com
6	Fiona	Campbell	fiona.campbell@email.com

6 rows in set (0.001 sec)

```
MariaDB [ecommerce_management_db]>
```

Figure 43 Rename Operation in Customer Table

5.1.6 Aggregate Function

The screenshot shows a MySQL Client window with the following content:

```
MySQL Client (MariaDB 11.7.0) x + -
```

product_id	product_name	price	category_name
1	Smartphone X	749.99	Smartphones
2	Gaming Laptop Pro	1599.99	Laptops
3	Wireless Earbuds	99.99	Electronics
4	Cotton T-Shirt	24.99	Men's Clothing
5	Men's Running Shoes	89.99	Men's Clothing
6	Bluetooth Speaker	59.99	Electronics

6 rows in set (0.021 sec)

```
MariaDB [ecommerce_management_db]> SELECT
-> customer_id AS "ID",
-> first_name AS "First Name",
-> last_name AS "Last Name",
-> email AS "Email Address"
-> FROM Customer;
```

ID	First Name	Last Name	Email Address
1	Alice	Johnson	alice.johnson@email.com
2	Bob	Smith	bob.smith@email.com
3	Charlie	Brown	charlie.brown@email.com
4	Diana	Prince	diana.prince@email.com
5	Evan	Lee	evan.lee@email.com
6	Fiona	Campbell	fiona.campbell@email.com

6 rows in set (0.001 sec)

```
MariaDB [ecommerce_management_db]> SELECT
-> COUNT(*) AS TotalProducts,
-> AVG(price) AS AveragePrice,
-> MAX(price) AS HighestPrice,
-> MIN(price) AS LowestPrice,
-> SUM(stock_quantity) AS TotalInventory
-> FROM Product;
```

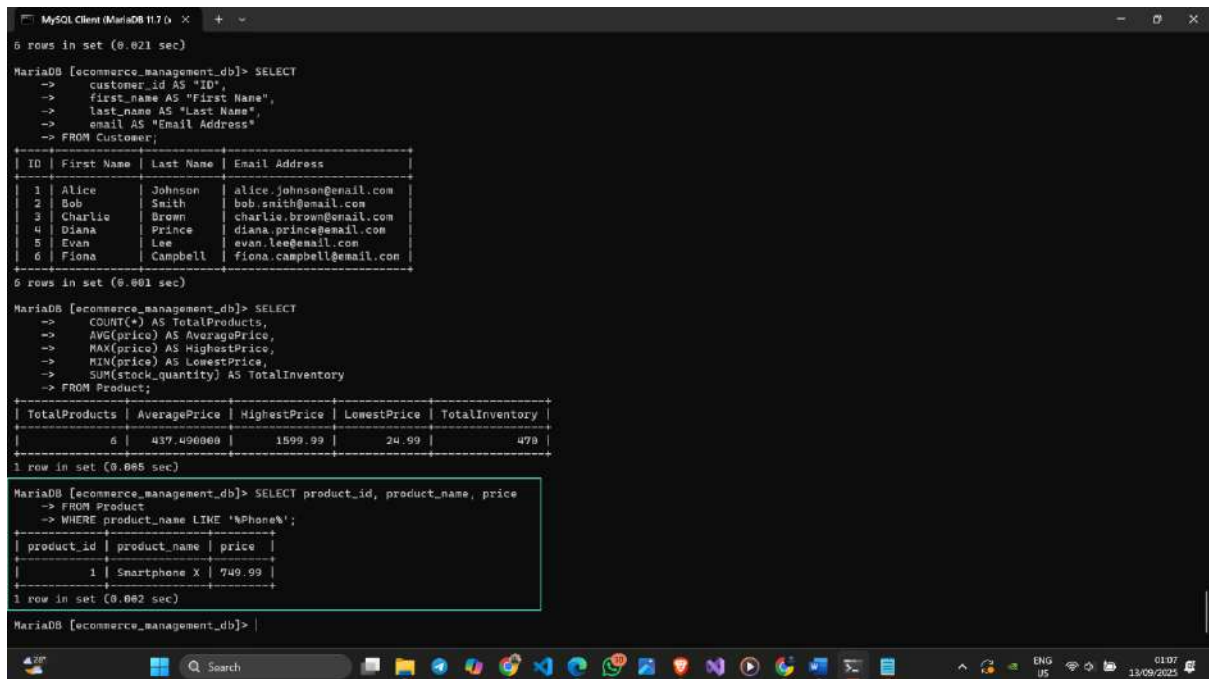
TotalProducts	AveragePrice	HighestPrice	LowestPrice	TotalInventory
6	439.490000	1599.99	24.99	470

1 row in set (0.005 sec)

```
MariaDB [ecommerce_management_db]>
```

Figure 44 Aggregate Functions in Product Table

5.1.7 LIKE Keyword



The screenshot shows a MySQL Client window with three SQL queries and their results. The first query selects all columns from the Customer table. The second query is an aggregate query on the Product table. The third query, highlighted with a green box, uses the LIKE keyword to filter products by name.

```
MySQL Client (MariaDB 11.7.0) x + -
6 rows in set (0.021 sec)

MariaDB [ecommerce_management_db]> SELECT
-> customer_id AS "ID",
-> first_name AS "First Name",
-> last_name AS "Last Name",
-> email AS "Email Address"
-> FROM Customer;
+----+-----+-----+-----+
| ID | First Name | Last Name | Email Address |
+----+-----+-----+-----+
| 1  | Alice     | Johnson  | alice.johnson@email.com |
| 2  | Bob       | Smith    | bob.smith@email.com     |
| 3  | Charlie   | Brown    | charlie.brown@email.com  |
| 4  | Diana     | Prince   | diana.prince@email.com   |
| 5  | Evan      | Lee      | evan.lee@email.com       |
| 6  | Fiona     | Campbell | fiona.campbell@email.com |
+----+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> SELECT
-> COUNT(*) AS TotalProducts,
-> AVG(price) AS AveragePrice,
-> MAX(price) AS HighestPrice,
-> MIN(price) AS LowestPrice,
-> SUM(stock_quantity) AS TotalInventory
-> FROM Product;
+-----+-----+-----+-----+-----+
| TotalProducts | AveragePrice | HighestPrice | LowestPrice | TotalInventory |
+-----+-----+-----+-----+-----+
| 6             | 437.4000000  | 1599.99      | 24.99       | 478            |
+-----+-----+-----+-----+-----+
1 row in set (0.005 sec)

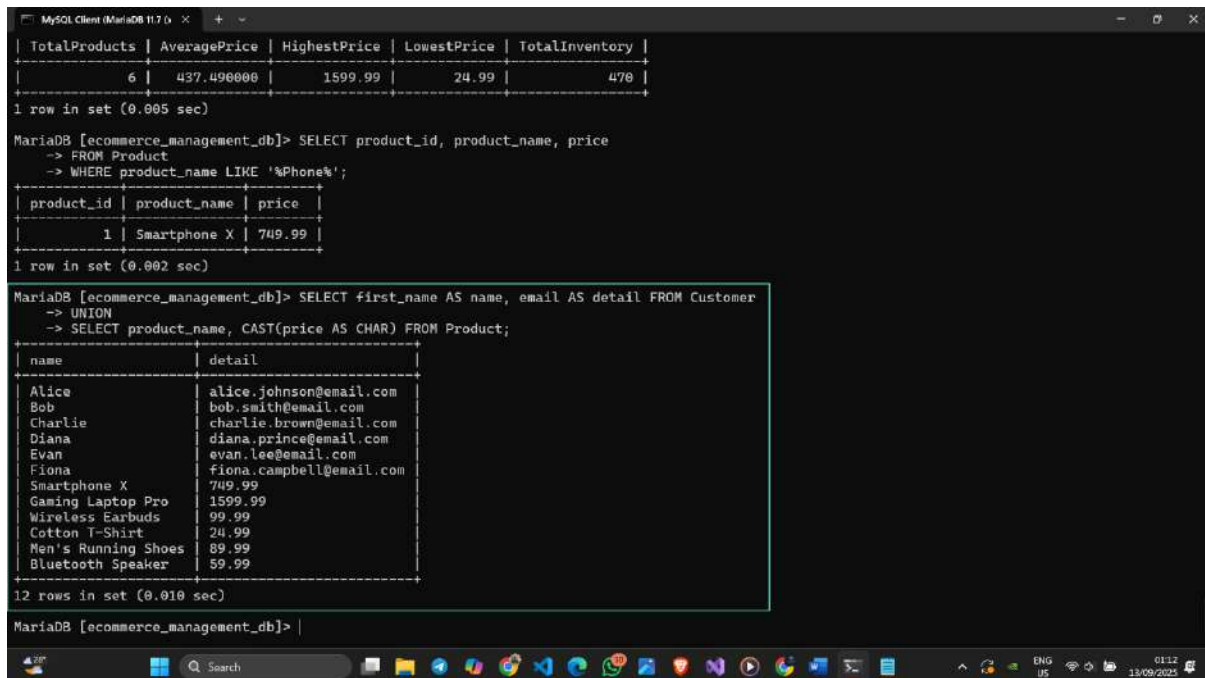
MariaDB [ecommerce_management_db]> SELECT product_id, product_name, price
-> FROM Product
-> WHERE product_name LIKE 'iPhone%';
+-----+-----+-----+
| product_id | product_name | price |
+-----+-----+-----+
| 1           | Smartphone X | 749.99 |
+-----+-----+-----+
1 row in set (0.002 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 45 Use of LIKE in Product Table

5.2 Complex Queries

5.2.1 Set Operations (Union, Intersection, Set Difference, Division)



The screenshot shows the MySQL Client interface with a query window titled 'MySQL Client (MariaDB 11.7.0)'. The query executed is:

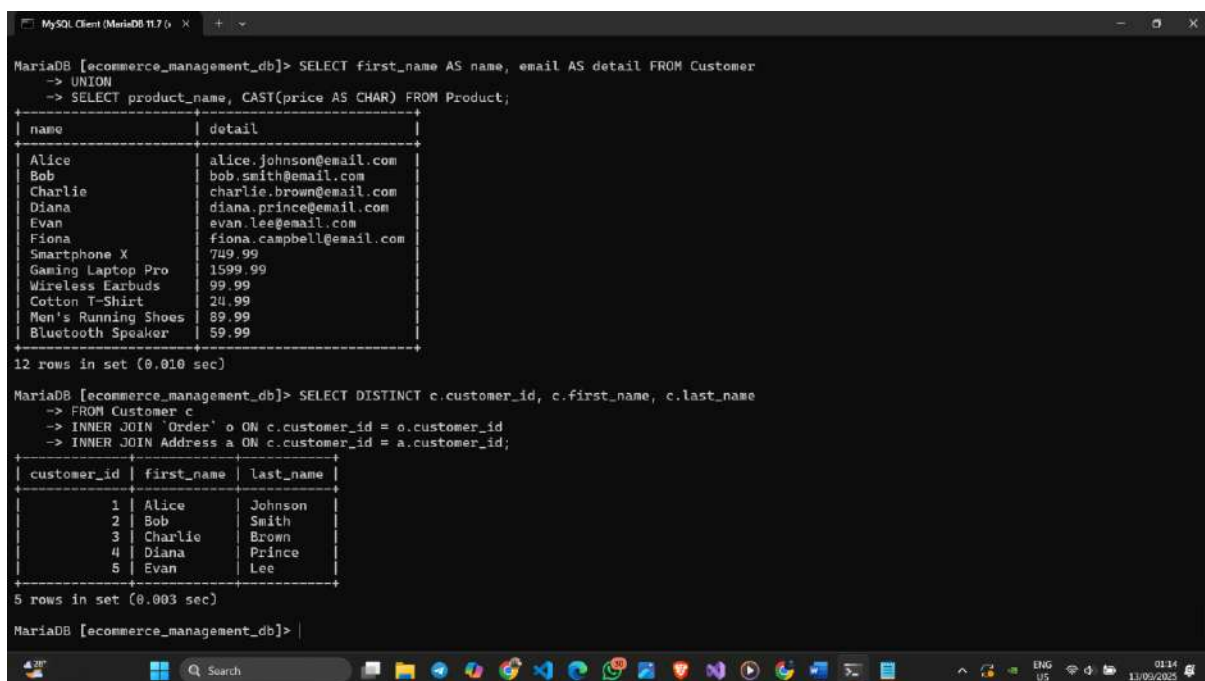
```
MariaDB [ecommerce_management_db]> SELECT first_name AS name, email AS detail FROM Customer
-> UNION
-> SELECT product_name, CAST(price AS CHAR) FROM Product;
```

The result is displayed as a table with two columns: 'name' and 'detail'. It contains 12 rows, which are the union of the 'Customer' table and the 'Product' table.

name	detail
Alice	alice.johnson@email.com
Bob	bob.smith@email.com
Charlie	charlie.brown@email.com
Diana	diana.prince@email.com
Evan	evan.lee@email.com
Fiona	fiona.campbell@email.com
Smartphone X	749.99
Gaming Laptop Pro	1599.99
Wireless Earbuds	99.99
Cotton T-Shirt	24.99
Men's Running Shoes	89.99
Bluetooth Speaker	59.99

The status bar at the bottom indicates '12 rows in set (0.010 sec)'.

Figure 46 Union Operation



The screenshot shows the MySQL Client interface with a query window titled 'MySQL Client (MariaDB 11.7.0)'. The query executed is:

```
MariaDB [ecommerce_management_db]> SELECT DISTINCT c.customer_id, c.first_name, c.last_name
-> FROM Customer c
-> INNER JOIN 'Order' o ON c.customer_id = o.customer_id
-> INNER JOIN Address a ON c.customer_id = a.customer_id;
```

The result is displayed as a table with three columns: 'customer_id', 'first_name', and 'last_name'. It contains 5 rows, representing the intersection of customers who have placed orders and have an address.

customer_id	first_name	last_name
1	Alice	Johnson
2	Bob	Smith
3	Charlie	Brown
4	Diana	Prince
5	Evan	Lee

The status bar at the bottom indicates '5 rows in set (0.003 sec)'.

Figure 47 Intersection Operation

```
MySQL Client (MariaDB 11.7.0) x + -
Diana      diana.prince@email.com
Evan       evan.lee@email.com
Fiona      fiona.campbell@email.com
Smartphone X 749.99
Gaming Laptop Pro 1599.99
Wireless Earbuds 99.99
Cotton T-Shirt 24.99
Men's Running Shoes 89.99
Bluetooth Speaker 59.99
-----
12 rows in set (0.010 sec)

MariaDB [ecommerce_management_db]> SELECT DISTINCT c.customer_id, c.first_name, c.last_name
--> FROM Customer c
--> INNER JOIN 'Order' o ON c.customer_id = o.customer_id
--> INNER JOIN Address a ON c.customer_id = a.customer_id;
-----
customer_id | first_name | last_name |
-----
1 | Alice | Johnson |
2 | Bob | Smith |
3 | Charlie | Brown |
4 | Diana | Prince |
5 | Evan | Lee |
-----
5 rows in set (0.003 sec)

MariaDB [ecommerce_management_db]> SELECT p.product_id, p.product_name
--> FROM Product p
--> WHERE p.product_id NOT IN (SELECT product_id FROM CartItem);
-----
product_id | product_name |
-----
6 | Bluetooth Speaker |
-----
1 row in set (0.005 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 48 Set Difference Operation

```
MySQL Client (MariaDB 11.7.0) x + -
MariaDB [ecommerce_management_db]> SELECT DISTINCT c.customer_id, c.first_name, c.last_name
--> FROM Customer c
--> INNER JOIN 'Order' o ON c.customer_id = o.customer_id
--> INNER JOIN Address a ON c.customer_id = a.customer_id;
-----
customer_id | first_name | last_name |
-----
1 | Alice | Johnson |
2 | Bob | Smith |
3 | Charlie | Brown |
4 | Diana | Prince |
5 | Evan | Lee |
-----
5 rows in set (0.003 sec)

MariaDB [ecommerce_management_db]> SELECT p.product_id, p.product_name
--> FROM Product p
--> WHERE p.product_id NOT IN (SELECT product_id FROM CartItem);
-----
product_id | product_name |
-----
6 | Bluetooth Speaker |
-----
1 row in set (0.005 sec)

MariaDB [ecommerce_management_db]> SELECT c.customer_id, c.first_name, c.last_name, o.total_amount
--> FROM Customer c
--> JOIN 'Order' o ON c.customer_id = o.customer_id
--> WHERE o.total_amount > (SELECT AVG(total_amount) FROM 'Order');
-----
customer_id | first_name | last_name | total_amount |
-----
1 | Alice | Johnson | 879.98 |
3 | Charlie | Brown | 1599.99 |
-----
2 rows in set (0.004 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 49 Division Operation

5.2.2 Join Operations (Inner, Natural, Left Outer, Right Outer, Full Outer)

```
MySQL Client (MariaDB 11.7.0) x + -
+-----+
| 6 | Bluetooth Speaker |
+-----+
1 row in set (0.005 sec)

MariaDB [ecommerce_management_db]> SELECT c.customer_id, c.first_name, c.last_name, o.total_amount
-> FROM Customer c
-> JOIN 'Order' o ON c.customer_id = o.customer_id
-> WHERE o.total_amount > (SELECT AVG(total_amount) FROM 'Order');
+-----+
| customer_id | first_name | last_name | total_amount |
+-----+
| 1 | Alice | Johnson | 879.98 |
| 3 | Charlie | Brown | 1599.99 |
+-----+
2 rows in set (0.004 sec)

MariaDB [ecommerce_management_db]> CREATE VIEW OrderDetails AS
-> SELECT o.order_id, o.order_date, o.total_amount, o.status,
-> c.customer_id, c.first_name, c.last_name
-> FROM 'Order' o
-> JOIN Customer c ON o.customer_id = c.customer_id;
Query OK, 0 rows affected (0.007 sec)

MariaDB [ecommerce_management_db]> SELECT * FROM OrderDetails;
+-----+
| order_id | order_date | total_amount | status | customer_id | first_name | last_name |
+-----+
| 1 | 2024-06-10 09:30:00 | 879.98 | Paid | 1 | Alice | Johnson |
| 2 | 2024-06-11 14:22:15 | 124.95 | Shipped | 2 | Bob | Smith |
| 3 | 2024-06-12 10:05:43 | 1599.99 | Delivered | 3 | Charlie | Brown |
| 4 | 2024-06-13 16:45:30 | 89.99 | Paid | 4 | Diana | Prince |
| 5 | 2024-06-14 11:11:11 | 119.98 | Paid | 5 | Evan | Lee |
| 6 | 2024-06-15 12:00:00 | 0.00 | Pending | 6 | Fiona | Campbell |
+-----+
6 rows in set (0.005 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 50 Inner Join Operation

```
MySQL Client (MariaDB 11.7.0) x + -
-> FROM 'Order' o
-> JOIN Customer c ON o.customer_id = c.customer_id;
Query OK, 0 rows affected (0.007 sec)

MariaDB [ecommerce_management_db]> SELECT * FROM OrderDetails;
+-----+
| order_id | order_date | total_amount | status | customer_id | first_name | last_name |
+-----+
| 1 | 2024-06-10 09:30:00 | 879.98 | Paid | 1 | Alice | Johnson |
| 2 | 2024-06-11 14:22:15 | 124.95 | Shipped | 2 | Bob | Smith |
| 3 | 2024-06-12 10:05:43 | 1599.99 | Delivered | 3 | Charlie | Brown |
| 4 | 2024-06-13 16:45:30 | 89.99 | Paid | 4 | Diana | Prince |
| 5 | 2024-06-14 11:11:11 | 119.98 | Paid | 5 | Evan | Lee |
| 6 | 2024-06-15 12:00:00 | 0.00 | Pending | 6 | Fiona | Campbell |
+-----+
6 rows in set (0.005 sec)

MariaDB [ecommerce_management_db]> SELECT p.product_name, c.category_name
-> FROM Product p
-> NATURAL JOIN Category c;
Empty set (0.003 sec)

MariaDB [ecommerce_management_db]> SELECT c.customer_id, c.first_name, c.last_name, o.order_id, o.order_date
-> FROM Customer c
-> LEFT JOIN 'Order' o ON c.customer_id = o.customer_id;
+-----+
| customer_id | first_name | last_name | order_id | order_date |
+-----+
| 1 | Alice | Johnson | 1 | 2024-06-10 09:30:00 |
| 2 | Bob | Smith | 2 | 2024-06-11 14:22:15 |
| 3 | Charlie | Brown | 3 | 2024-06-12 10:05:43 |
| 4 | Diana | Prince | 4 | 2024-06-13 16:45:30 |
| 5 | Evan | Lee | 5 | 2024-06-14 11:11:11 |
| 6 | Fiona | Campbell | 6 | 2024-06-15 12:00:00 |
+-----+
6 rows in set (0.002 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 51 Left Outer Join Operation

```

MySQL Client (MariaDB 11.7.0)
6 rows in set (0.005 sec)

MariaDB [ecommerce_management_db]> SELECT p.product_name, c.category_name
-> FROM Product p
-> NATURAL JOIN Category c;
Empty set (0.003 sec)

MariaDB [ecommerce_management_db]> SELECT c.customer_id, c.first_name, c.last_name, o.order_id, o.order_date
-> FROM Customer c
-> LEFT JOIN 'Order' o ON c.customer_id = o.customer_id;

+-----+-----+-----+-----+-----+
| customer_id | first_name | last_name | order_id | order_date |
+-----+-----+-----+-----+-----+
| 1 | Alice | Johnson | 1 | 2024-06-10 09:30:00 |
| 2 | Bob | Smith | 2 | 2024-06-11 14:22:15 |
| 3 | Charlie | Brown | 3 | 2024-06-12 10:05:43 |
| 4 | Diana | Prince | 4 | 2024-06-13 16:45:30 |
| 5 | Evan | Lee | 5 | 2024-06-14 11:11:11 |
| 6 | Fiona | Campbell | 6 | 2024-06-15 12:00:00 |
+-----+-----+-----+-----+-----+
6 rows in set (0.002 sec)

MariaDB [ecommerce_management_db]> SELECT o.order_id, o.order_date, c.customer_id, c.first_name, c.last_name
-> FROM Customer c
-> RIGHT JOIN 'Order' o ON c.customer_id = o.customer_id;

+-----+-----+-----+-----+-----+
| order_id | order_date | customer_id | first_name | last_name |
+-----+-----+-----+-----+-----+
| 1 | 2024-06-10 09:30:00 | 1 | Alice | Johnson |
| 2 | 2024-06-11 14:22:15 | 2 | Bob | Smith |
| 3 | 2024-06-12 10:05:43 | 3 | Charlie | Brown |
| 4 | 2024-06-13 16:45:30 | 4 | Diana | Prince |
| 5 | 2024-06-14 11:11:11 | 5 | Evan | Lee |
| 6 | 2024-06-15 12:00:00 | 6 | Fiona | Campbell |
+-----+-----+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]>

```

Figure 52 Right Outer Join Operation

```

MySQL Client (MariaDB 11.7.0)
6 rows in set (0.002 sec)

MariaDB [ecommerce_management_db]> SELECT o.order_id, o.order_date, c.customer_id, c.first_name, c.last_name
-> FROM Customer c
-> RIGHT JOIN 'Order' o ON c.customer_id = o.customer_id;

+-----+-----+-----+-----+-----+
| order_id | order_date | customer_id | first_name | last_name |
+-----+-----+-----+-----+-----+
| 1 | 2024-06-10 09:30:00 | 1 | Alice | Johnson |
| 2 | 2024-06-11 14:22:15 | 2 | Bob | Smith |
| 3 | 2024-06-12 10:05:43 | 3 | Charlie | Brown |
| 4 | 2024-06-13 16:45:30 | 4 | Diana | Prince |
| 5 | 2024-06-14 11:11:11 | 5 | Evan | Lee |
| 6 | 2024-06-15 12:00:00 | 6 | Fiona | Campbell |
+-----+-----+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> SELECT c.customer_id, c.first_name, c.last_name, o.order_id, o.order_date
-> FROM Customer c
-> LEFT JOIN 'Order' o ON c.customer_id = o.customer_id
-> UNION
-> SELECT c.customer_id, c.first_name, c.last_name, o.order_id, o.order_date
-> FROM Customer c
-> RIGHT JOIN 'Order' o ON c.customer_id = o.customer_id;

+-----+-----+-----+-----+-----+
| customer_id | first_name | last_name | order_id | order_date |
+-----+-----+-----+-----+-----+
| 1 | Alice | Johnson | 1 | 2024-06-10 09:30:00 |
| 2 | Bob | Smith | 2 | 2024-06-11 14:22:15 |
| 3 | Charlie | Brown | 3 | 2024-06-12 10:05:43 |
| 4 | Diana | Prince | 4 | 2024-06-13 16:45:30 |
| 5 | Evan | Lee | 5 | 2024-06-14 11:11:11 |
| 6 | Fiona | Campbell | 6 | 2024-06-15 12:00:00 |
+-----+-----+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]>

```

Figure 53 Full Outer Join Operation


```
MySQL Client (MariaDB 11.7.0) x + -
1 | Alice | Johnson | 1 | 2024-06-10 09:30:00 |
2 | Bob | Smith | 2 | 2024-06-11 14:22:15 |
3 | Charlie | Brown | 3 | 2024-06-12 10:05:43 |
4 | Diana | Prince | 4 | 2024-06-13 16:45:30 |
5 | Evan | Lee | 5 | 2024-06-14 11:11:11 |
6 | Fiona | Campbell | 6 | 2024-06-15 12:00:00 |
6 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> SELECT
-> 'Customer' AS source,
-> CONCAT(first_name, ' ', last_name) AS name,
-> NULL AS product_name
-> FROM Customer
-> UNION ALL
-> SELECT
-> 'Product' AS source,
-> NULL AS name,
-> product_name
-> FROM Product
-> LIMIT 10;

+-----+-----+-----+
| source | name | product_name |
+-----+-----+-----+
| Customer | Alice Johnson | NULL |
| Customer | Bob Smith | NULL |
| Customer | Charlie Brown | NULL |
| Customer | Diana Prince | NULL |
| Customer | Evan Lee | NULL |
| Customer | Fiona Campbell | NULL |
| Product | NULL | Smartphone X |
| Product | NULL | Gaming Laptop Pro |
| Product | NULL | Wireless Earbuds |
| Product | NULL | Cotton T-Shirt |
+-----+-----+-----+
10 rows in set (0.004 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 54 Outer Union Operation

5.2.3 Nested Queries

```
MySQL Client (MariaDB 11.7.0) x + -
-> SELECT
-> 'Product' AS source,
-> NULL AS name,
-> product_name
-> FROM Product
-> LIMIT 10;

+-----+-----+-----+
| source | name | product_name |
+-----+-----+-----+
| Customer | Alice Johnson | NULL |
| Customer | Bob Smith | NULL |
| Customer | Charlie Brown | NULL |
| Customer | Diana Prince | NULL |
| Customer | Evan Lee | NULL |
| Customer | Fiona Campbell | NULL |
| Product | NULL | Smartphone X |
| Product | NULL | Gaming Laptop Pro |
| Product | NULL | Wireless Earbuds |
| Product | NULL | Cotton T-Shirt |
+-----+-----+-----+
10 rows in set (0.004 sec)

MariaDB [ecommerce_management_db]> SELECT product_id, product_name, price
-> FROM Product
-> WHERE category_id IN (
-> SELECT category_id FROM Category
-> WHERE category_name = 'Electronics' OR parent_category_id = 1
-> );

+-----+-----+-----+
| product_id | product_name | price |
+-----+-----+-----+
| 1 | Smartphone X | 749.99 |
| 3 | Wireless Earbuds | 99.99 |
| 6 | Bluetooth Speaker | 59.99 |
+-----+-----+-----+
3 rows in set (0.003 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 55 Nested Query with IN

```

MySQL Client (MariaDB 11.7.0)
+-----+
| Customer | Fiona Campbell | NULL |
| Product  | NULL           | NULL |
| Product  | NULL           | NULL |
| Product  | NULL           | NULL |
| Product  | NULL           | NULL |
+-----+
10 rows in set (0.004 sec)

MariaDB [ecommerce_management_db]> SELECT product_id, product_name, price
  -> FROM Product
  -> WHERE category_id IN (
  ->   SELECT category_id FROM Category
  ->   WHERE category_name = 'Electronics' OR parent_category_id = 1
  -> );
+-----+
| product_id | product_name | price |
+-----+
| 1           | Smartphone X | 749.99 |
| 3           | Wireless Earbuds | 99.99 |
| 6           | Bluetooth Speaker | 59.99 |
+-----+
3 rows in set (0.003 sec)

MariaDB [ecommerce_management_db]> SELECT DISTINCT c.customer_id, c.first_name, c.last_name
  -> FROM Customer c
  -> JOIN 'Order' o ON c.customer_id = o.customer_id
  -> WHERE o.total_amount > ANY (
  ->   SELECT total_amount FROM 'Order' WHERE customer_id = 2
  -> );
+-----+
| customer_id | first_name | last_name |
+-----+
| 1           | Alice     | Johnson   |
| 3           | Charlie   | Brown     |
+-----+
2 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]>

```

Figure 56 Nested Query with ANY

```

MySQL Client (MariaDB 11.7.0)
+-----+
| 1 | Smartphone X | 749.99 |
| 3 | Wireless Earbuds | 99.99 |
| 6 | Bluetooth Speaker | 59.99 |
+-----+
3 rows in set (0.003 sec)

MariaDB [ecommerce_management_db]> SELECT DISTINCT c.customer_id, c.first_name, c.last_name
  -> FROM Customer c
  -> JOIN 'Order' o ON c.customer_id = o.customer_id
  -> WHERE o.total_amount > ANY (
  ->   SELECT total_amount FROM 'Order' WHERE customer_id = 2
  -> );
+-----+
| customer_id | first_name | last_name |
+-----+
| 1           | Alice     | Johnson   |
| 3           | Charlie   | Brown     |
+-----+
2 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> SELECT p1.product_id, p1.product_name, p1.price, c.category_name
  -> FROM Product p1
  -> JOIN Category c ON p1.category_id = c.category_id
  -> WHERE p1.price > (
  ->   SELECT AVG(p2.price)
  ->   FROM Product p2
  ->   WHERE p2.category_id = p1.category_id
  -> );
+-----+
| product_id | product_name | price | category_name |
+-----+
| 3           | Wireless Earbuds | 99.99 | Electronics |
| 5           | Men's Running Shoes | 89.99 | Men's Clothing |
+-----+
2 rows in set (0.003 sec)

MariaDB [ecommerce_management_db]>

```

Figure 57 Correlated Nested Query

CHAPTER 6 - DATABASE TUNING

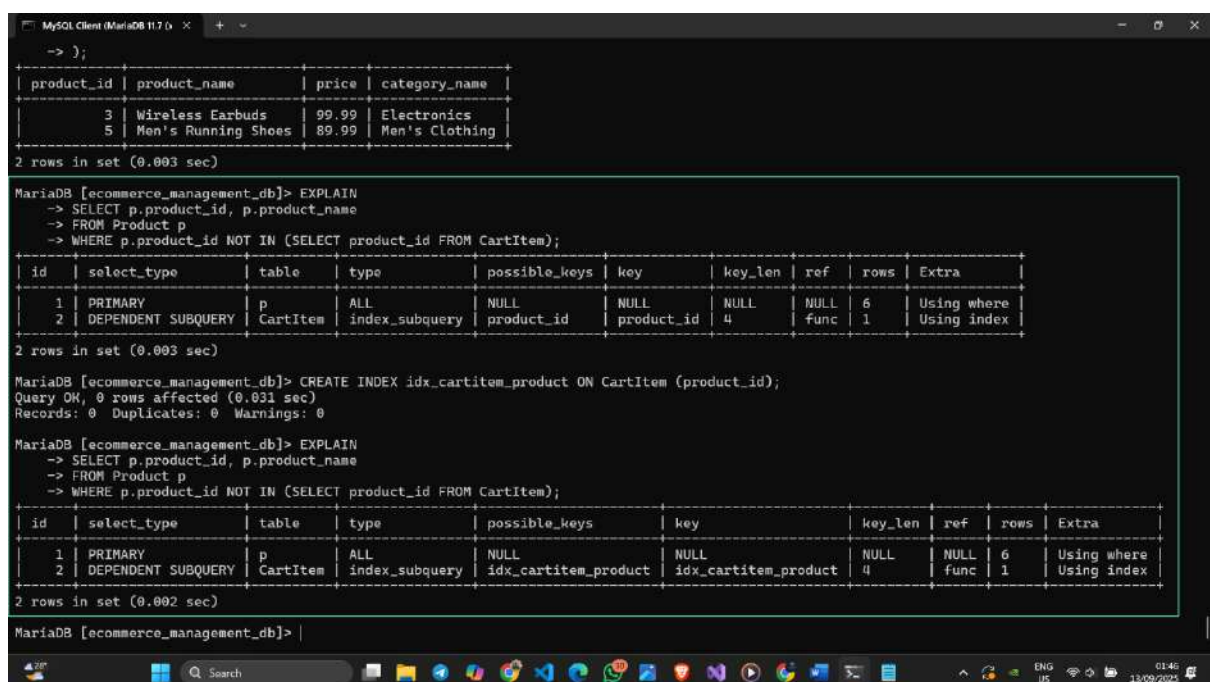
6.1 Query Tuning Methodology

We will select 10 complex queries from Chapter 5 and attempt to tune them. The primary method will be adding **indexes** on columns used in WHERE, JOIN, and ORDER BY clauses. We will use the EXPLAIN command to analyze the query execution plan before and after tuning to prove the improvement. We will compare the number of rows scanned ("rows" in the EXPLAIN output).

6.2 Analysis of Original Queries

First, we need to see how MariaDB currently executes the queries. We do this by putting EXPLAIN before the query.

6.3 Tuned Queries and Index Implementation



```
MySQL Client (MariaDB 11.7.0) x + -
-> );

+-----+
| product_id | product_name | price | category_name |
+-----+
| 3 | Wireless Earbuds | 99.99 | Electronics |
| 5 | Men's Running Shoes | 89.99 | Men's Clothing |
+-----+
2 rows in set (0.003 sec)

MariaDB [ecommerce_management_db]> EXPLAIN
-> SELECT p.product_id, p.product_name
-> FROM Product p
-> WHERE p.product_id NOT IN (SELECT product_id FROM CartItem);

+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+-----+
| 1 | PRIMARY | p | ALL | NULL | NULL | NULL | NULL | 6 | Using where |
| 2 | DEPENDENT SUBQUERY | CartItem | index_subquery | product_id | product_id | 4 | func | 1 | Using index |
+-----+
2 rows in set (0.003 sec)

MariaDB [ecommerce_management_db]> CREATE INDEX idx_cartitem_product ON CartItem (product_id);
Query OK, 0 rows affected (0.031 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> EXPLAIN
-> SELECT p.product_id, p.product_name
-> FROM Product p
-> WHERE p.product_id NOT IN (SELECT product_id FROM CartItem);

+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+-----+
| 1 | PRIMARY | p | ALL | NULL | NULL | NULL | NULL | 6 | Using where |
| 2 | DEPENDENT SUBQUERY | CartItem | index_subquery | idx_cartitem_product | idx_cartitem_product | 4 | func | 1 | Using index |
+-----+
2 rows in set (0.002 sec)

MariaDB [ecommerce_management_db]> |
```

Figure 58 Before tuned and after tuned for query 1


```

MySQL Client (MariaDB 11.7.0)
MariaDB [ecommerce_management_db]> CREATE INDEX idx_cartitem_product ON CartItem (product_id);
Query OK, 8 rows affected (0.031 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> EXPLAIN
-> SELECT p.product_id, p.product_name
-> FROM Product p
-> WHERE p.product_id NOT IN (SELECT product_id FROM CartItem);

```

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	PRIMARY	p	ALL	idx_cartitem_product	NULL	NULL	NULL	6	Using where
2	DEPENDENT SUBQUERY				idx_cartitem_product	4	func	1	Using index

```

2 rows in set (0.002 sec)

MariaDB [ecommerce_management_db]> EXPLAIN
-> SELECT c.customer_id, c.first_name, c.last_name, o.total_amount
-> FROM Customer c
-> JOIN Order o ON c.customer_id = o.customer_id
-> WHERE o.total_amount > (SELECT AVG(total_amount) FROM Order);

```

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	PRIMARY	c	ALL	customer_id	NULL	NULL	NULL	6	Using where
1	PRIMARY	c	eq_ref	PRIMARY	PRIMARY	4	ecommerce_management_db.o.customer_id	1	
2	SUBQUERY	Order	ALL	NULL	NULL	NULL	NULL	6	Using index

```

3 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> CREATE INDEX idx_order_total_amount ON Order (total_amount);
Query OK, 8 rows affected (0.019 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> EXPLAIN
-> SELECT c.customer_id, c.first_name, c.last_name, o.total_amount
-> FROM Customer c
-> JOIN Order o ON c.customer_id = o.customer_id
-> WHERE o.total_amount > (SELECT AVG(total_amount) FROM Order);

```

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	PRIMARY	c	range	customer_id, idx_order_total_amount	idx_order_total_amount	5	NULL	2	Using where
1	PRIMARY	c	eq_ref	PRIMARY	PRIMARY	4	ecommerce_management_db.o.customer_id	1	
2	SUBQUERY	Order	index	NULL	idx_order_total_amount	5	NULL	6	Using index

```

3 rows in set (0.004 sec)

MariaDB [ecommerce_management_db]>

```

Figure 59 Before tuned and after tuned for query 2

```

MySQL Client (MariaDB 11.7.0)
-> JOIN Order o ON c.customer_id = o.customer_id
-> WHERE o.total_amount > (SELECT AVG(total_amount) FROM Order);

```

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	PRIMARY	c	range	customer_id, idx_order_total_amount	idx_order_total_amount	5	NULL	2	Using where
1	PRIMARY	c	eq_ref	PRIMARY	PRIMARY	4	ecommerce_management_db.o.customer_id	1	
2	SUBQUERY	Order	index	NULL	idx_order_total_amount	5	NULL	6	Using index

```

3 rows in set (0.004 sec)

MariaDB [ecommerce_management_db]> EXPLAIN
-> SELECT product_id, product_name, price
-> FROM Product
-> WHERE category_id IN (
-> SELECT category_id FROM Category
-> WHERE category_name = 'Electronics' OR parent_category_id = 1
-> );

```

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	PRIMARY	Product	ALL	category_id	NULL	NULL	NULL	6	Using where
1	PRIMARY	Category	eq_ref	PRIMARY, parent_category_id	PRIMARY	4	ecommerce_management_db.Product.category_id	1	

```

2 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> CREATE INDEX idx_product_category_id ON Product (category_id);
Query OK, 8 rows affected (0.012 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> CREATE INDEX idx_category_name_parent ON Category (category_name, parent_category_id);
Query OK, 8 rows affected (0.012 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> EXPLAIN
-> SELECT product_id, product_name, price
-> FROM Product
-> WHERE category_id IN (
-> SELECT category_id FROM Category
-> WHERE category_name = 'Electronics' OR parent_category_id = 1
-> );

```

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	PRIMARY	Product	ALL	idx_product_category_id	NULL	NULL	NULL	6	Using where
1	PRIMARY	Category	eq_ref	PRIMARY, parent_category_id, idx_category_name_parent	PRIMARY	4	ecommerce_management_db.Product.category_id	1	

```

2 rows in set (0.005 sec)

MariaDB [ecommerce_management_db]>

```

Figure 60 Before tuned and after tuned for query 3

```

MySQL Client (MariaDB 11.7.0)
MariaDB [ecommerce_management_db]> CREATE INDEX idx_category_name_parent ON Category (category_name, parent_category_id);
Query OK, 0 rows affected (0.012 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> EXPLAIN
--> SELECT product_id, product_name, price
--> FROM Product
--> WHERE category_id IN (
-->   SELECT category_id FROM Category
-->   WHERE category_name = 'Electronics' OR parent_category_id = 1
--> );

```

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	PRIMARY	Product	ALL	idx_product_category_id	NULL	NULL	NULL	6	
1	PRIMARY	Category	eq_ref	PRIMARY, parent_category_id, idx_category_name_parent	PRIMARY	4	ecommerce_management_db.Product.category_id	1	Using where

```

2 rows in set (0.005 sec)

MariaDB [ecommerce_management_db]> EXPLAIN
--> SELECT DISTINCT c.customer_id, c.first_name, c.last_name
--> FROM Customer c
--> JOIN `Order` o ON c.customer_id = o.customer_id
--> WHERE o.total_amount > ANY (
-->   SELECT total_amount FROM `Order` WHERE customer_id = 2
--> );

```

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	PRIMARY	o	ALL	customer_id	NULL	NULL	NULL	6	Using where; Using temporary
1	PRIMARY	c	eq_ref	PRIMARY	PRIMARY	4	ecommerce_management_db.o.customer_id	1	
2	SUBQUERY	Order	ref	customer_id	customer_id	4	const	1	

```

3 rows in set (0.003 sec)

MariaDB [ecommerce_management_db]> CREATE INDEX idx_order_customer_id_amount ON `Order` (customer_id, total_amount);
Query OK, 0 rows affected (0.013 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> EXPLAIN
--> SELECT DISTINCT c.customer_id, c.first_name, c.last_name
--> FROM Customer c
--> JOIN `Order` o ON c.customer_id = o.customer_id
--> WHERE o.total_amount > ANY (
-->   SELECT total_amount FROM `Order` WHERE customer_id = 2
--> );

```

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	PRIMARY	o	index	idx_order_customer_id_amount	idx_order_customer_id_amount	9	NULL	6	Using where; Using index; Using temporary
1	PRIMARY	c	eq_ref	PRIMARY	PRIMARY	4	ecommerce_management_db.o.customer_id	1	
2	SUBQUERY	NULL	NULL	NULL	NULL	NULL	NULL	1	Select tables optimized away

```

3 rows in set (0.002 sec)

MariaDB [ecommerce_management_db]>

```

Figure 61 Before tuned and after tuned for query 4

```

MySQL Client (MariaDB 11.7.0)
--> SELECT DISTINCT c.customer_id, c.first_name, c.last_name
--> FROM Customer c
--> JOIN `Order` o ON c.customer_id = o.customer_id
--> WHERE o.total_amount > ANY (
-->   SELECT total_amount FROM `Order` WHERE customer_id = 2
--> );

```

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	PRIMARY	o	index	idx_order_customer_id_amount	idx_order_customer_id_amount	9	NULL	6	Using where; Using index; Using temporary
1	PRIMARY	c	eq_ref	PRIMARY	PRIMARY	4	ecommerce_management_db.o.customer_id	1	
2	SUBQUERY	NULL	NULL	NULL	NULL	NULL	NULL	1	Select tables optimized away

```

3 rows in set (0.002 sec)

MariaDB [ecommerce_management_db]> EXPLAIN
--> SELECT p1.product_id, p1.product_name, p1.price, c.category_name
--> FROM Product p1
--> JOIN Category c ON p1.category_id = c.category_id
--> WHERE p1.price > (
-->   SELECT AVG(p2.price)
-->   FROM Product p2
-->   WHERE p2.category_id = p1.category_id
--> );

```

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	PRIMARY	p1	ALL	idx_product_category_id	NULL	NULL	NULL	6	
1	PRIMARY	c	eq_ref	PRIMARY	PRIMARY	4	ecommerce_management_db.p1.category_id	1	Using where
2	DEPENDENT SUBQUERY	p2	ref	idx_product_category_id	idx_product_category_id	4	ecommerce_management_db.p1.category_id	1	

```

3 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> CREATE INDEX idx_product_category_price ON Product (category_id, price);
Query OK, 0 rows affected (0.005 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> EXPLAIN
--> SELECT p1.product_id, p1.product_name, p1.price, c.category_name
--> FROM Product p1
--> JOIN Category c ON p1.category_id = c.category_id
--> WHERE p1.price > (
-->   SELECT AVG(p2.price)
-->   FROM Product p2
-->   WHERE p2.category_id = p1.category_id
--> );

```

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	PRIMARY	p1	ALL	idx_product_category_id, idx_product_category_price	NULL	PRIMARY	NULL	6	Using where
1	PRIMARY	c	eq_ref	PRIMARY	PRIMARY	4	ecommerce_management_db.p1.category_id	1	
2	DEPENDENT SUBQUERY	p2	ref	idx_product_category_id, idx_product_category_price	idx_product_category_price	4	ecommerce_management_db.p1.category_id	1	Using index

```

3 rows in set (0.002 sec)

MariaDB [ecommerce_management_db]>

```

Figure 62 Before tuned and after tuned for query 5

```

MySQL Client (MariaDB 11.7.0)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> EXPLAIN
-> SELECT p1.product_id, p1.product_name, p1.price, c.category_name
-> FROM Product p1
-> JOIN Category c ON p1.category_id = c.category_id
-> WHERE p1.price > (
->   SELECT AVG(p2.price)
->   FROM Product p2
->   WHERE p2.category_id = p1.category_id
-> );

+----+ select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 | SIMPLE | p1 | ALL | idx_product_category_id,idx_product_category_price | NULL | NULL | NULL | 6 | Using where |
1 | SIMPLE | c | eq_ref | PRIMARY | PRIMARY | 4 | ecommerce_management_db.p1.category_id | 1 | |
2 | DEPENDENT SUBQUERY | p2 | ref | idx_product_category_id,idx_product_category_price | idx_product_category_price | 4 | ecommerce_management_db.p1.category_id | 1 | Using index |
3 rows in set (0.002 sec)

MariaDB [ecommerce_management_db]> EXPLAIN SELECT * FROM OrderDetails;

+----+ select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
1 | SIMPLE | o | ALL | idx_order_customer_id_amount | NULL | NULL | NULL | 6 | |
1 | SIMPLE | c | eq_ref | PRIMARY | PRIMARY | 4 | ecommerce_management_db.o.customer_id | 1 | |
2 rows in set (0.002 sec)

MariaDB [ecommerce_management_db]> CREATE INDEX idx_order_customer_id ON 'Order' (customer_id);
Query OK, 0 rows affected (0.011 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> EXPLAIN SELECT * FROM OrderDetails;

+----+ select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
1 | SIMPLE | o | ALL | idx_order_customer_id_amount,idx_order_customer_id | NULL | NULL | NULL | 6 | |
1 | SIMPLE | c | eq_ref | PRIMARY | PRIMARY | 4 | ecommerce_management_db.o.customer_id | 1 | |
2 rows in set (0.002 sec)

MariaDB [ecommerce_management_db]>

```

Figure 63 Before tuned and after tuned for query 6

```

MySQL Client (MariaDB 11.7.0)

+----+ select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
1 | SIMPLE | o | ALL | idx_order_customer_id_amount,idx_order_customer_id | NULL | NULL | NULL | 6 | |
1 | SIMPLE | c | eq_ref | PRIMARY | PRIMARY | 4 | ecommerce_management_db.o.customer_id | 1 | |
2 rows in set (0.002 sec)

MariaDB [ecommerce_management_db]> EXPLAIN
-> SELECT customer_id, first_name, last_name, email
-> FROM Customer
-> WHERE last_name = 'Smith';

+----+ select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
1 | SIMPLE | Customer | ALL | NULL | NULL | NULL | NULL | 6 | Using where |
1 row in set (0.001 sec)

MariaDB [ecommerce_management_db]> CREATE INDEX idx_customer_last_name ON Customer (last_name);
Query OK, 0 rows affected (0.011 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> EXPLAIN
-> SELECT customer_id, first_name, last_name, email
-> FROM Customer
-> WHERE last_name = 'Smith';

+----+ select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
1 | SIMPLE | Customer | ref | idx_customer_last_name | idx_customer_last_name | 402 | const | 1 | Using index condition |
1 row in set (0.002 sec)

MariaDB [ecommerce_management_db]>

```

Figure 64 Before tuned and after tuned for query 7


```

MySQL Client (MariaDB 11.7.0)
+-----+
| 1 | SIMPLE | Customer | ref | idx_customer_last_name | idx_customer_last_name | 402 | const | 1 | Using index condition |
+-----+
1 row in set (0.002 sec)

MariaDB [ecommerce_management_db]> EXPLAIN
-> SELECT product_id, product_name, price
-> FROM Product
-> WHERE product_name LIKE '%Phone%';
+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+-----+
| 1 | SIMPLE | Product | ALL | NULL | NULL | NULL | NULL | 6 | Using where |
+-----+
1 row in set (0.001 sec)

MariaDB [ecommerce_management_db]> CREATE FULLTEXT INDEX idx_product_name_ft ON Product (product_name);
Query OK, 0 rows affected (0.047 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> EXPLAIN
-> SELECT product_id, product_name, price
-> FROM Product
-> WHERE MATCH(product_name) AGAINST('Phone' IN BOOLEAN MODE);
+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+-----+
| 1 | SIMPLE | Product | fulltext | idx_product_name_ft | idx_product_name_ft | 0 | | 1 | Using where |
+-----+
1 row in set (0.004 sec)

MariaDB [ecommerce_management_db]>

```

Figure 65 Before tuned and after tuned for query 8

```

MySQL Client (MariaDB 11.7.0)
-> SELECT product_id, product_name, price
-> FROM Product
-> WHERE MATCH(product_name) AGAINST('Phone' IN BOOLEAN MODE);
+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+-----+
| 1 | SIMPLE | Product | fulltext | idx_product_name_ft | idx_product_name_ft | 0 | | 1 | Using where |
+-----+
1 row in set (0.004 sec)

MariaDB [ecommerce_management_db]> EXPLAIN
-> SELECT order_id, order_date, total_amount, status
-> FROM 'Order'
-> WHERE order_date BETWEEN '2024-06-10' AND '2024-06-15';
+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+-----+
| 1 | SIMPLE | Order | ALL | NULL | NULL | NULL | NULL | 6 | Using where |
+-----+
1 row in set (0.003 sec)

MariaDB [ecommerce_management_db]> CREATE INDEX idx_order_date ON 'Order' (order_date);
Query OK, 0 rows affected (0.009 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> EXPLAIN
-> SELECT order_id, order_date, total_amount, status
-> FROM 'Order'
-> WHERE order_date BETWEEN '2024-06-10' AND '2024-06-15';
+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+-----+
| 1 | SIMPLE | Order | range | idx_order_date | idx_order_date | 5 | NULL | 5 | Using index condition |
+-----+
1 row in set (0.005 sec)

MariaDB [ecommerce_management_db]>

```

Figure 66 Before tuned and after tuned for query 9

```
MySQL Client (MariaDB 11.7.0) x + -
1 row in set (0.003 sec)

MariaDB [ecommerce_management_db]> CREATE INDEX idx_order_date ON `Order` (order_date);
Query OK, 0 rows affected (0.009 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [ecommerce_management_db]> EXPLAIN
-> SELECT order_id, order_date, total_amount, status
-> FROM `Order`
-> WHERE order_date BETWEEN '2024-06-19' AND '2024-06-15';
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | Order | range | idx_order_date | idx_order_date | 5 | NULL | 5 | Using index condition |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.005 sec)

MariaDB [ecommerce_management_db]> EXPLAIN
-> SELECT p.product_name, c.category_name
-> FROM Product p
-> INNER JOIN Category c ON p.category_id = c.category_id;
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | p | ALL | idx_product_category_id,idx_product_category_price | NULL | NULL | NULL | 6 | |
| 1 | SIMPLE | c | eq_ref | PRIMARY | PRIMARY | 4 | ecommerce_management_db.p.category_id | 1 | |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]> CREATE INDEX idx_product_category_id ON Product (category_id);
ERROR 1061 (42000): Duplicate key name 'idx_product_category_id'
MariaDB [ecommerce_management_db]> EXPLAIN
-> SELECT p.product_name, c.category_name
-> FROM Product p
-> INNER JOIN Category c ON p.category_id = c.category_id;
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref | rows | Extra |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | p | ALL | idx_product_category_id,idx_product_category_price | NULL | NULL | NULL | 6 | |
| 1 | SIMPLE | c | eq_ref | PRIMARY | PRIMARY | 4 | ecommerce_management_db.p.category_id | 1 | |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.001 sec)

MariaDB [ecommerce_management_db]>
```

Figure 67 Before tuned and after tuned for query 10

6.4 Performance Comparison

Example: This query finds products never added to a cart. Initially, MariaDB had to perform a full table scan on the CartItem table, examining all 6 rows. After creating an index on CartItem(product_id), the database could use this index to quickly locate the required values, reducing the number of examined rows to 1. This significantly improves the query's performance.

Like this etc...

PART B – NoSQL DATABASES

Chapter 1: Aggregation model

1.1 Bucket Creation and Data Insertion

The aggregation model for this project is implemented as a key-value store using PostgreSQL's JSONB data type. This approach treats a unique integer primary key as the main key and a flexible JSONB document as the value, allowing for a variable number of attributes per record. This schema-less design is well-suited for a NoSQL approach within a relational database context.

The following operations were performed to demonstrate this model:

- Creation of the Customer_EE00 bucket/collection: A table was created to store customer data.
- Data Insertion: Ten rows with diverse customer information were inserted.
- Data Retrieval: Two queries were executed to retrieve specific data aggregates, proving the model's functionality.

The subsequent screenshots in your report will visually document these steps, from table creation and data population to the final query results.

```
postgres=# CREATE DATABASE e_commerce_db;
CREATE DATABASE
postgres=# \c e_commerce_db
You are now connected to database "e_commerce_db" as user "postgres".
e_commerce_db=# CREATE TABLE Customer_EE00 (
e_commerce_db=#     customer_id INT PRIMARY KEY,
e_commerce_db=#     customer_data JSONB
e_commerce_db=# );
PS C:\Program Files\PostgreSQL\17\bin> .\psql.exe -U postgres -h localhost
psql: error: could not translate host name "localhost" to address: Name or service not known
PS C:\Program Files\PostgreSQL\17\bin> .\psql.exe -U postgres -h localhost -d postgres
Password for user postgres:

psql (17.6)
WARNING: Console code page (850) differs from Windows code page (1252)
        8-bit characters might not work correctly. See psql reference
        page "Notes for Windows users" for details.
Type "help" for help.

postgres=# \c e_commerce_db
You are now connected to database "e_commerce_db" as user "postgres".
e_commerce_db=# CREATE TABLE Customer_EE00 (customer_id INT PRIMARY KEY, customer_data JSONB);
CREATE TABLE
e_commerce_db=# INSERT INTO Customer_EE00 (customer_id, customer_data) VALUES
e_commerce_db=# (1, '{"firstName": "Alice", "lastName": "Johnson", "email": "alice.j@example.com", "registrationDate": "2024-01-15", "phone": "123-456-7890"}');
e_commerce_db=# (2, '{"firstName": "Bob", "lastName": "Smith", "email": "bob.smith@example.com", "registrationDate": "2024-02-20", "phone": "123-456-7891"}');
e_commerce_db=# (3, '{"firstName": "Charlie", "lastName": "Brown", "email": "charlie.b@example.com", "phone": "123-456-7892"}');
e_commerce_db=# (4, '{"firstName": "Diana", "lastName": "Prince", "email": "diana.p@example.com", "address": "123 Main St, Anytown"}');
e_commerce_db=# (5, '{"firstName": "Evan", "lastName": "Lee", "email": "evan.lee@example.com", "registrationDate": "2024-04-22", "isVIP": true}');
e_commerce_db=# (6, '{"firstName": "Fiona", "lastName": "Campbell", "email": "fiona.c@example.com", "orders": 5}');
e_commerce_db=# (7, '{"firstName": "George", "lastName": "Harrison", "email": "george.h@example.com", "phone": "123-456-7893"}');
e_commerce_db=# (8, '{"firstName": "Hannah", "lastName": "Montana", "email": "hannah.m@example.com", "registrationDate": "2024-06-01", "isVIP": false}');
e_commerce_db=# (9, '{"firstName": "Isaac", "lastName": "Newton", "email": "isaac.n@example.com", "orders": 2, "lastOrderDate": "2024-05-10"}');
e_commerce_db=# (10, '{"firstName": "Jack", "lastName": "Sparrow", "email": "jack.s@example.com", "registrationDate": "2024-06-15", "isVIP": true}');
INSERT 0 10
```

Figure 68 Creation and insertion into table

1.2 Query Operations on the Aggregate

```
e_commerce_db=# SELECT customer_id, customer_data FROM Customer_EE00
e_commerce_db=# WHERE customer_data ->> 'isVIP' = 'true';
customer_id | customer_data
-----
5 | {"email": "evan.lee@example.com", "isVIP": true, "lastName": "Lee", "firstName": "Evan", "registrationDate": "2024-04-22"}
10 | {"email": "jack.s@example.com", "isVIP": true, "lastName": "Sparrow", "firstName": "Jack", "registrationDate": "2024-06-15"}
(2 rows)
```

Figure 69 First Query

```
e_commerce_db=# SELECT customer_id, customer_data ->> 'firstName' AS first_name, customer_data ->> 'lastName' AS last_name
e_commerce_db=# FROM Customer_EE00
e_commerce_db=# WHERE (customer_data ->> 'orders')::int >= 5;
customer_id | first_name | last_name
-----
6 | Fiona | Campbell
(1 row)

e_commerce_db=# SELECT * FROM Customer_EE00
e_commerce_db=#
e_commerce_db=# SELECT * FROM Customer_EE00;
ERROR: syntax error at or near "SELECT"
LINE 2: SELECT * FROM Customer_EE00;
          ^

e_commerce_db=# SELECT * FROM Customer_EE00;
customer_id | customer_data
-----
1 | {"email": "alice.j@example.com", "phone": "123-456-7890", "lastName": "Johnson", "firstName": "Alice", "registrationDate": "2024-01-15"}
2 | {"email": "bob.smith@example.com", "phone": "123-456-7891", "lastName": "Smith", "firstName": "Bob", "registrationDate": "2024-02-20"}
3 | {"email": "charlie.b@example.com", "phone": "123-456-7892", "lastName": "Brown", "firstName": "Charlie"}
4 | {"email": "diana.p@example.com", "address": "123 Main St, Anytown", "lastName": "Prince", "firstName": "Diana"}
5 | {"email": "evan.lee@example.com", "isVIP": true, "lastName": "Lee", "firstName": "Evan", "registrationDate": "2024-04-22"}
6 | {"email": "fiona.c@example.com", "orders": 5, "lastName": "Campbell", "firstName": "Fiona"}
7 | {"email": "george.h@example.com", "phone": "123-456-7893", "lastName": "Harrison", "firstName": "George"}
8 | {"email": "hannah.m@example.com", "isVIP": false, "lastName": "Montana", "firstName": "Hannah", "registrationDate": "2024-06-01"}
9 | {"email": "isaac.n@example.com", "orders": 2, "lastName": "Newton", "firstName": "Isaac", "lastOrderDate": "2024-05-10"}
10 | {"email": "jack.s@example.com", "isVIP": true, "lastName": "Sparrow", "firstName": "Jack", "registrationDate": "2024-06-15"}
(10 rows)
```

Figure 70 Second Query

Chapter 2: Graph data Model

2.1 Graph Database Creation

In this project, a graph database was created using ArangoDB to represent relationships between a group of students. The graph consists of 10 student nodes and 15 edges connecting them, representing friendships and interactions.

Each student node in the Person collection contains the following properties:

- `_key`: a unique identifier for the student
- `name`: the student's name

Each edge in the FriendssWith collection represents a friendship and contains the following properties:

- `_from` and `_to`: references to the connected students
- `since`: the date when the friendship started
- `chat_count`: the total number of chats exchanged between the two students
- `common_interests`: a list of interests shared by both students

This graph model allows us to analyze social connections, common interests, and communication patterns between students. Screenshots were captured during the implementation process and are included below in the order they were created.

1. Created Collections for nodes and edges inside database `_system`

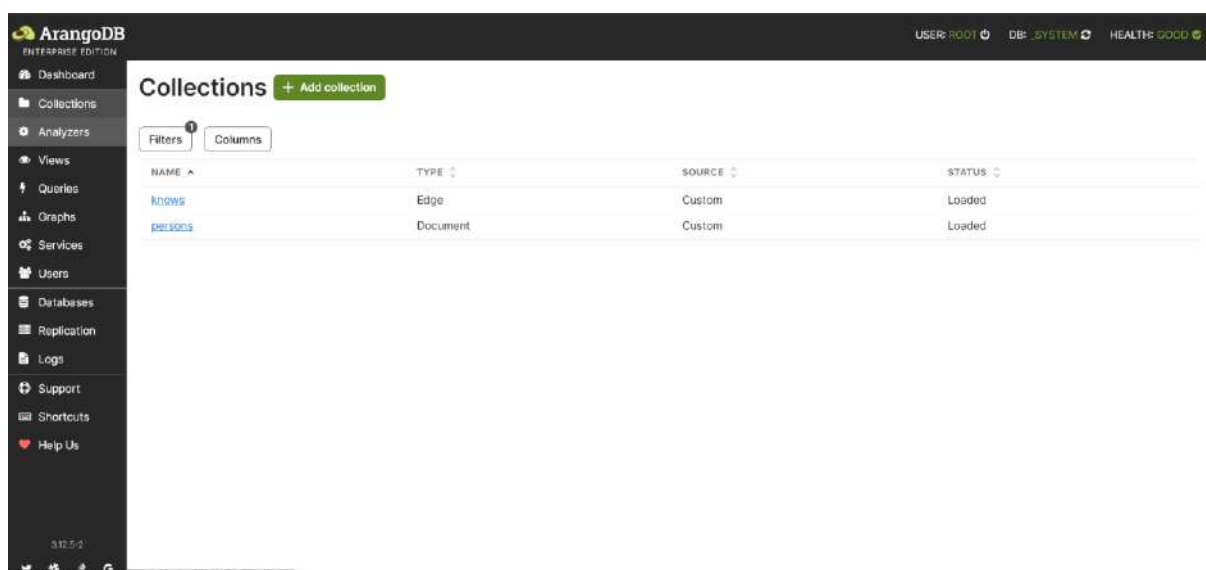


Figure 71 Created Database

2. Creating 10 Nodes

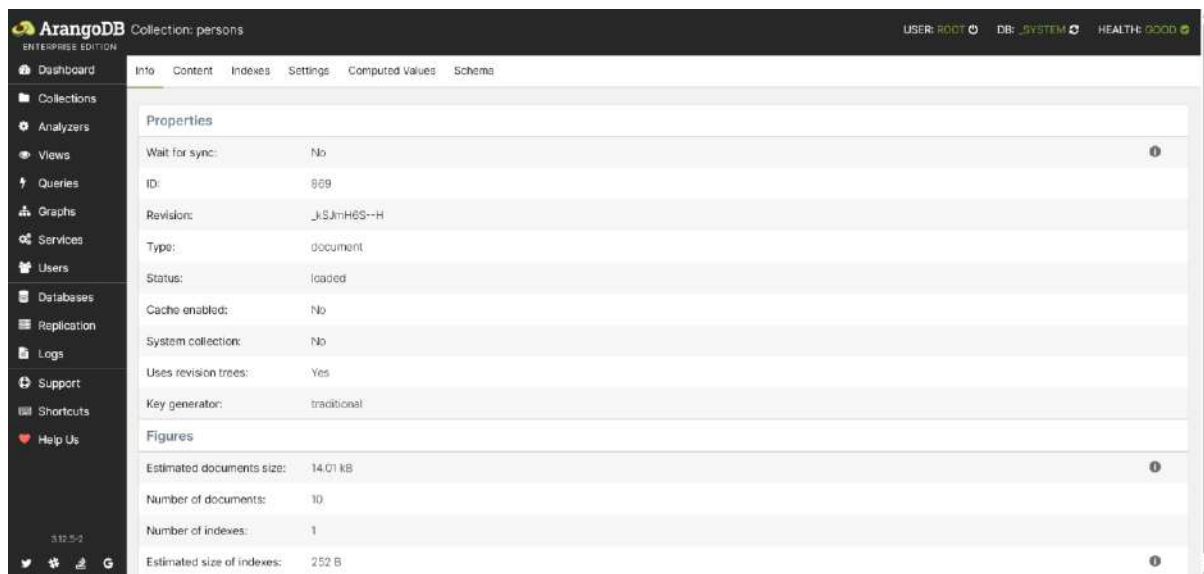


Figure 72 Creating Nodes

3. Created 10 Nodes

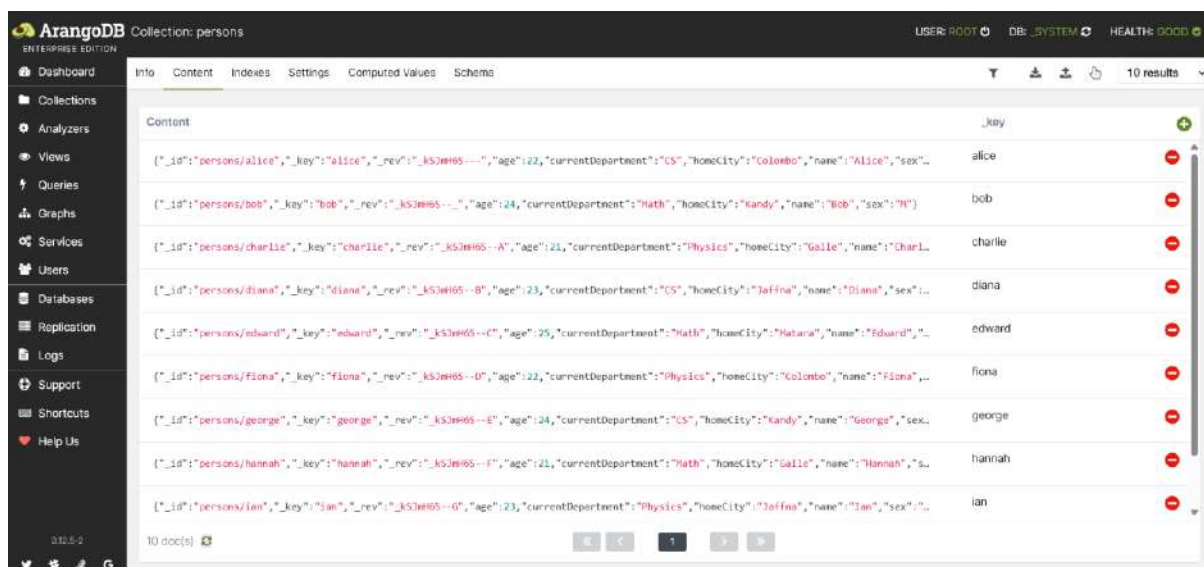


Figure 73 Created 10 nodes

4. Creating 15 edges or relationships

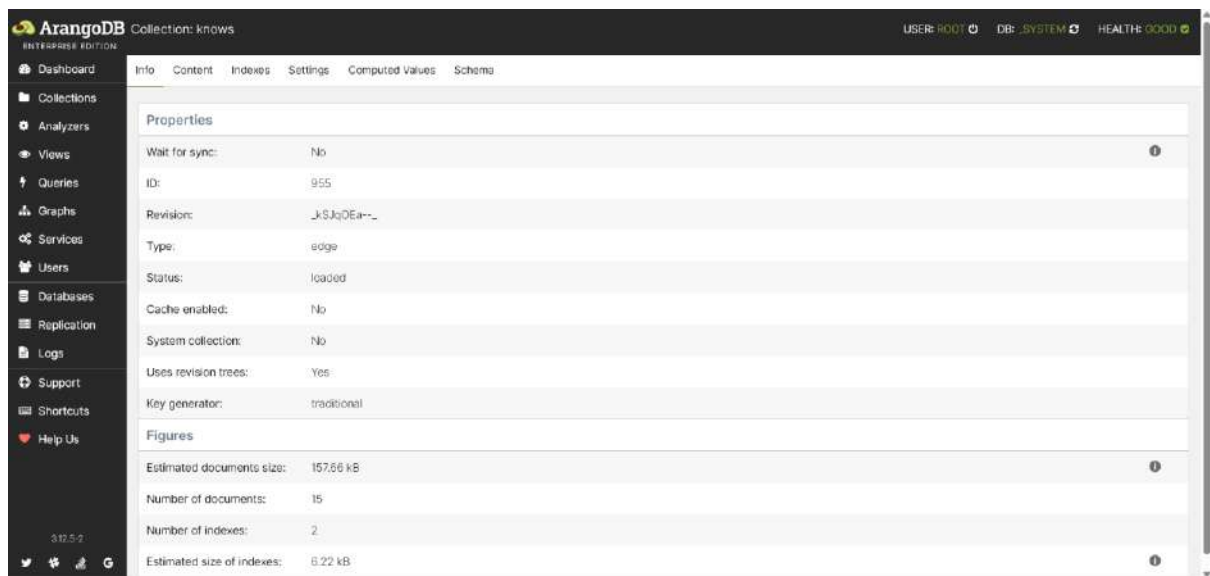


Figure 74 Creating 15 Edges (Relationships)

5. Created 15 edges or relationships

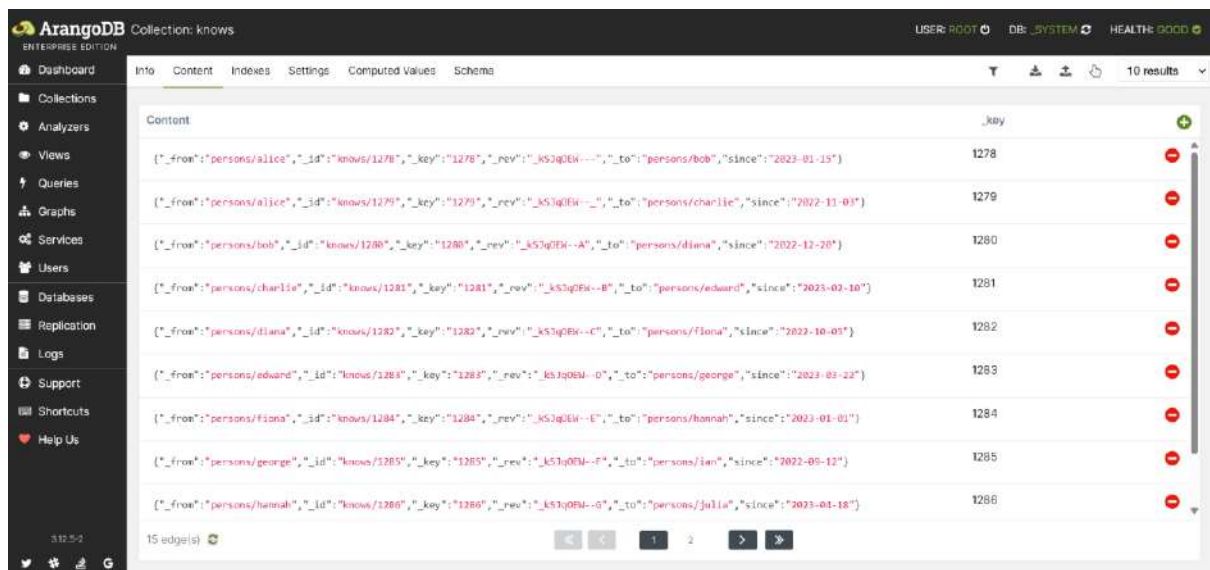


Figure 75 Created 15 Edges (Relationship)

6. Created and Implemented Graph model

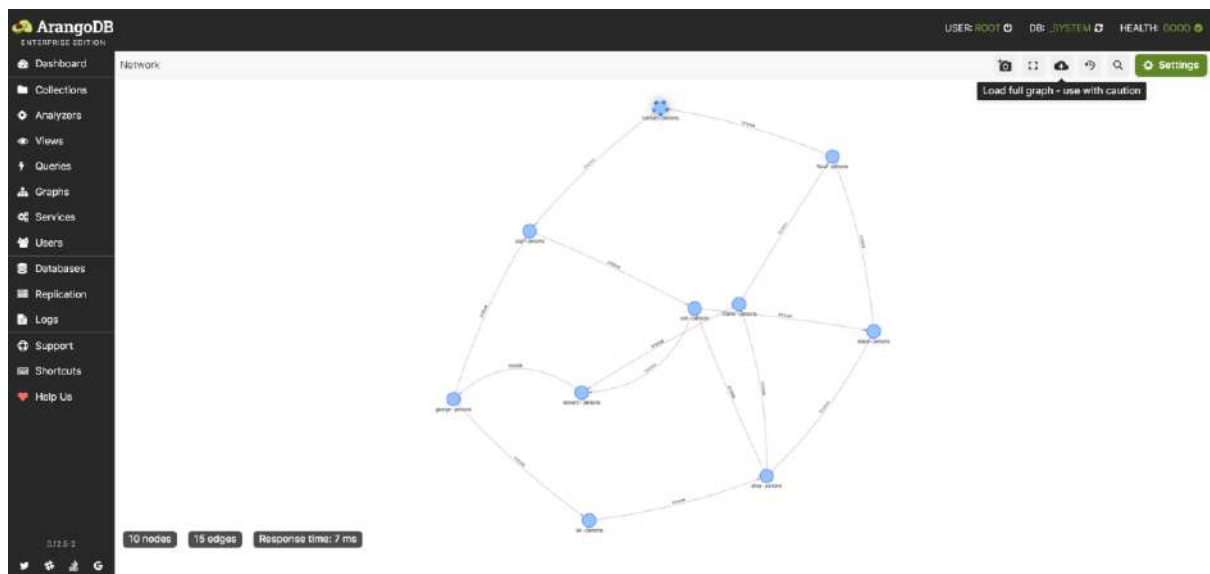


Figure 76 Graph

2.2 Querying the Social Network

7. Performed Query for data retrievals People who are Friends with Alice

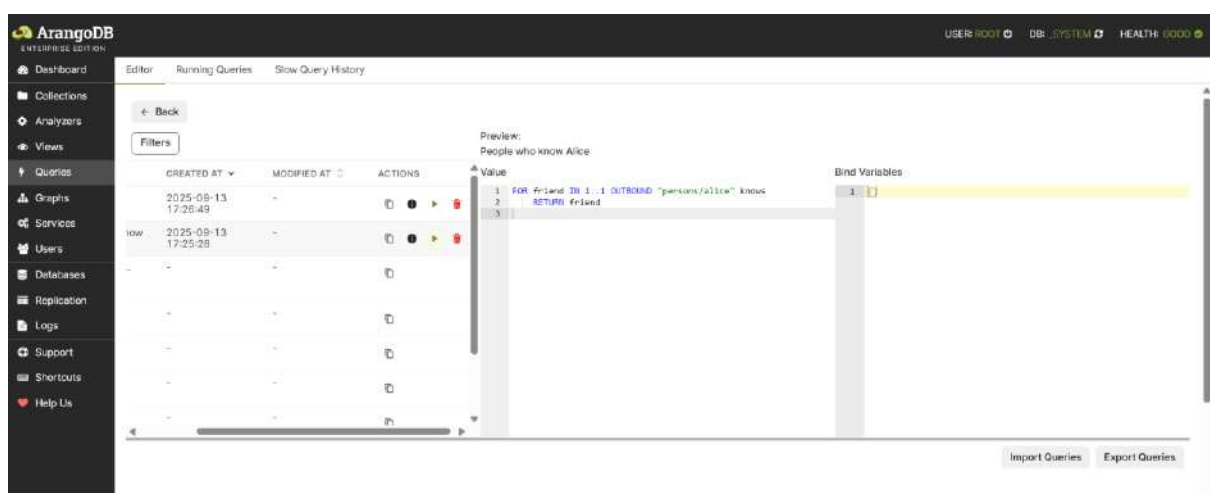


Figure 77 Query to retrieve people who are friends with Alice

ArangoDB
ENTERPRISE EDITION

USER: ROOT DB: SYSTEM HEALTH: OK

Dashboard Editor Running Queries Slow Query History

Import Queries Export Queries

Remove all results

Query 3 elements 1.589 ms

_key	_id	_rev	name	sex	age	homeCity	currentDepartment
diana	persons/diana	_kS.jmH65--B	Diana	F	23	Jaffna	CS
charlie	persons/charlie	_kS.jmH65--A	Charlie	M	21	Galle	Physics
bob	persons/bob	_kS.jmH65--C	Bob	M	24	Kandy	Math

Download CSV Download JSON Copy query to editor

Figure 78 Output of 1st Query

8. Performed Query for retrieve path from Alice to Julia

ArangoDB
ENTERPRISE EDITION

USER: ROOT DB: SYSTEM HEALTH: OK

Dashboard Editor Running Queries Slow Query History

Back Filters

Preview: alice to julia

CREATED AT	MODIFIED AT	ACTIONS
2025-09-13 17:26:49	-	[Icons]
2025-09-13 17:25:28	-	[Icons]

Value

```
1 FOR v, e, p IN 1..5 OUTBOUND "persons/alice" knows
2 FILTER v._key == "julia"
3 RETURN p
```

Bind Variables

1 []

Import Queries Export Queries

Figure 79 Query to retrieve path from Alice to Julia

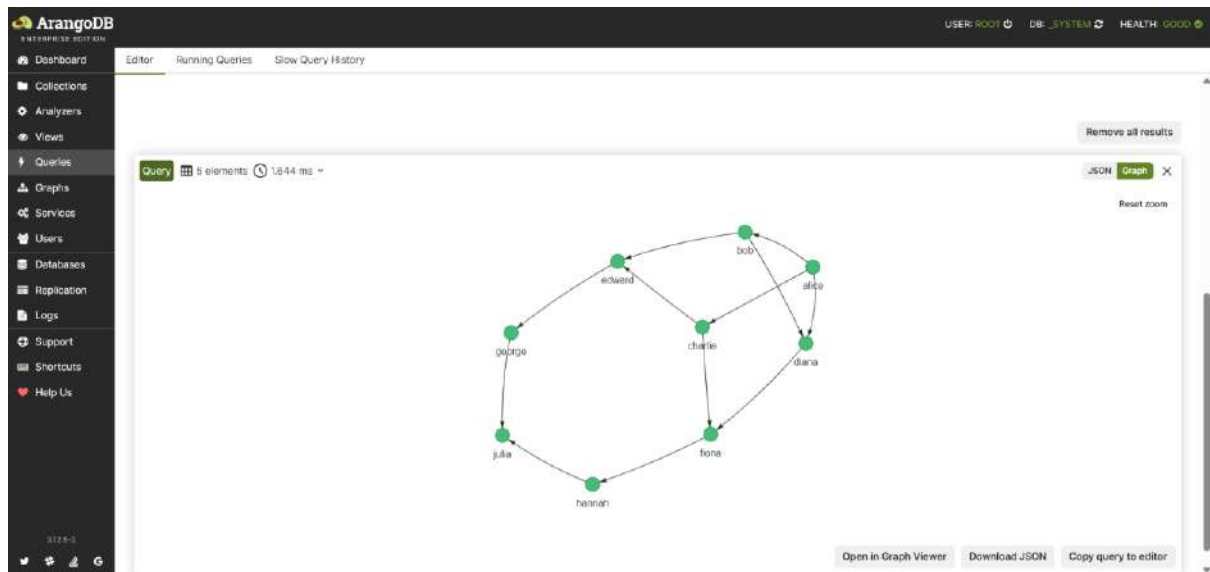


Figure 80 Output of second Query