SQL Joins

INNER JOIN
LEFT JOIN
RIGHT JOIN
FULL OUTER JOIN

SUBMITTED BY:SHALINI T

1. Introduction

The objective of this task is to practice and understand different types of SQL joins using MySQL Workbench. We created two related tables: Customers and Orders, and applied INNER JOIN, LEFT JOIN, RIGHT JOIN, and FULL OUTER JOIN to analyze how data is combined from multiple tables.

2. Database Setup

I have created a database named 'practice_db' and defined two tables: Customers and Orders.

SQL Code:

```
CREATE DATABASE practice_db;
USE practice_db;

-- Customers table
CREATE TABLE Customers (
   CustomerID INT PRIMARY KEY,
   CustomerName VARCHAR(50),
   City VARCHAR(50)
);

-- Orders table
CREATE TABLE Orders (
   OrderID INT PRIMARY KEY,
   CustomerID INT,
   Product VARCHAR(50),
   FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)
);
```

3. Data Insertion

Sample data was inserted into both tables.

```
INSERT INTO Customers (CustomerID, CustomerName, City)
VALUES
(1, 'John', 'New York'),
(2, 'Alice', 'London'),
(3, 'David', 'Paris'),
```

(4, 'Sophia', 'Berlin');

INSERT INTO Orders (OrderID, CustomerID, Product)
VALUES
(101, 1, 'Laptop'),
(102, 2, 'Phone'),
(103, 1, 'Tablet'),
(104, 5, 'Camera');

4. SQL Joins and Results

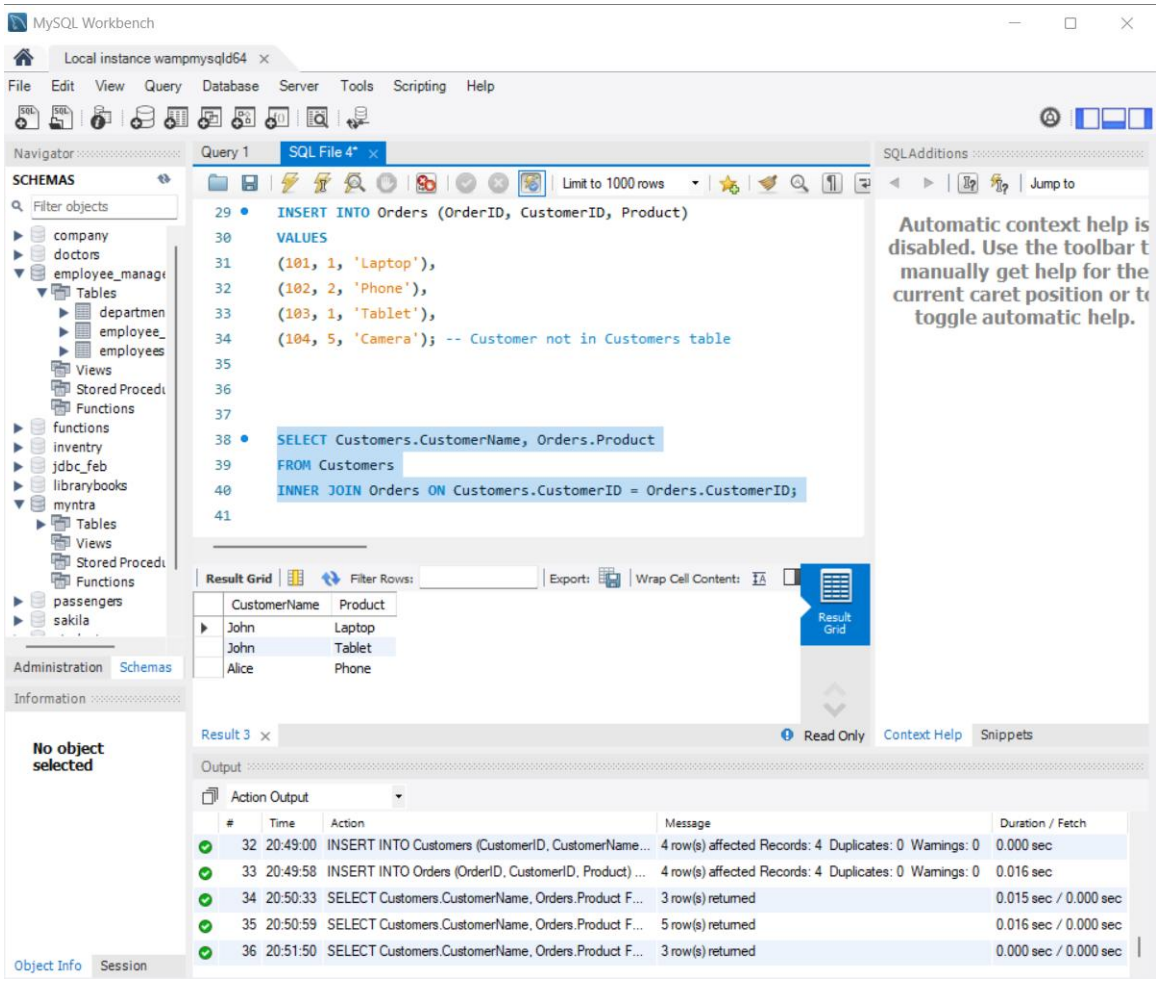
4.1 INNER JOIN

Returns only the rows that have matching values in both tables.
It is used when you want to see the common records between related tables.

Query:

```
SELECT Customers.CustomerName, Orders.Product
FROM Customers
INNER JOIN Orders ON Customers.CustomerID = Orders.CustomerID;
```

➡ This shows only customers who have placed orders.



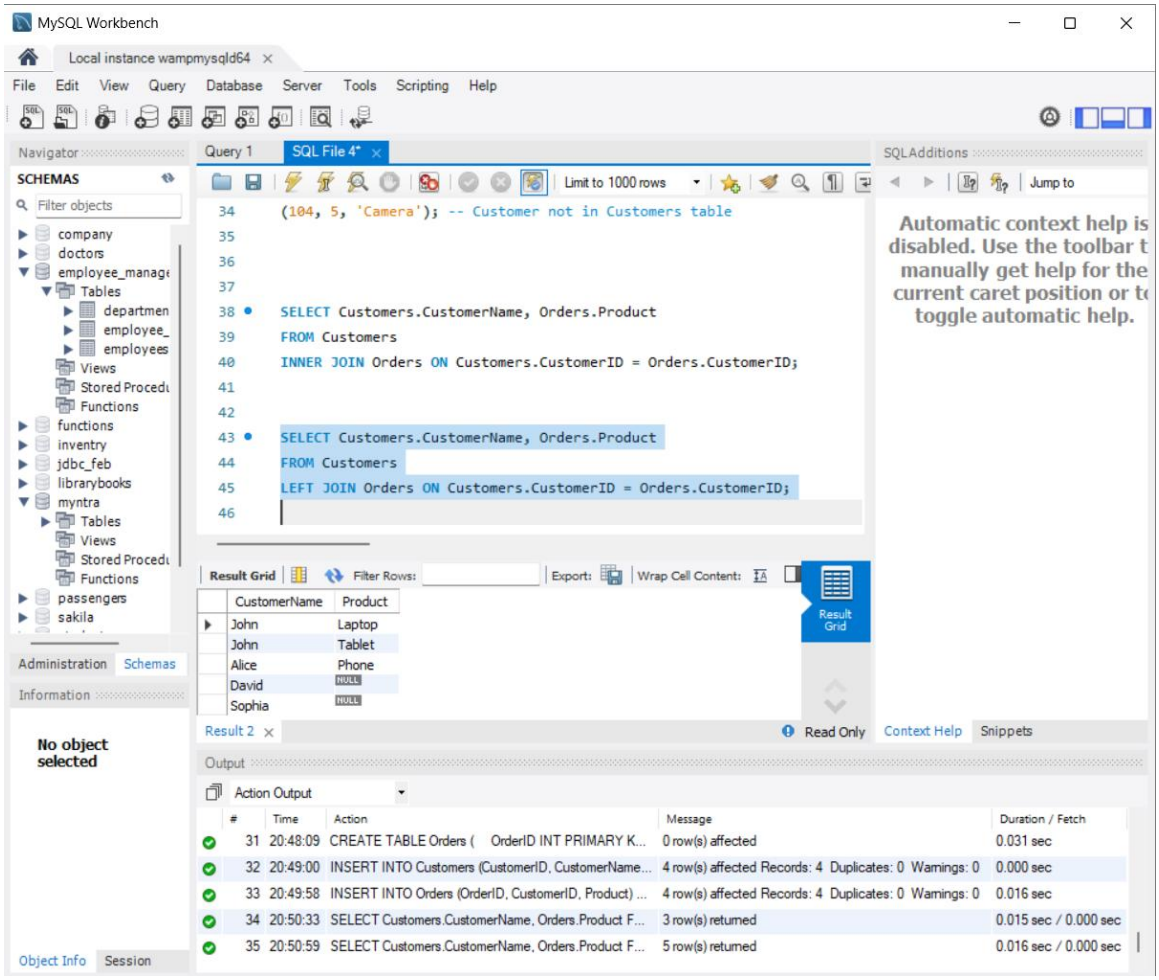
4.2 LEFT JOIN

Returns all rows from the left table, even if there are no matches in the right table.
If no match is found, NULL values are returned for the right table's columns.

Query:

SELECT Customers.CustomerName, Orders.Product
FROM Customers
LEFT JOIN Orders ON Customers.CustomerID = Orders.CustomerID;

➡ This shows all customers, including those without orders (NULL in Product).



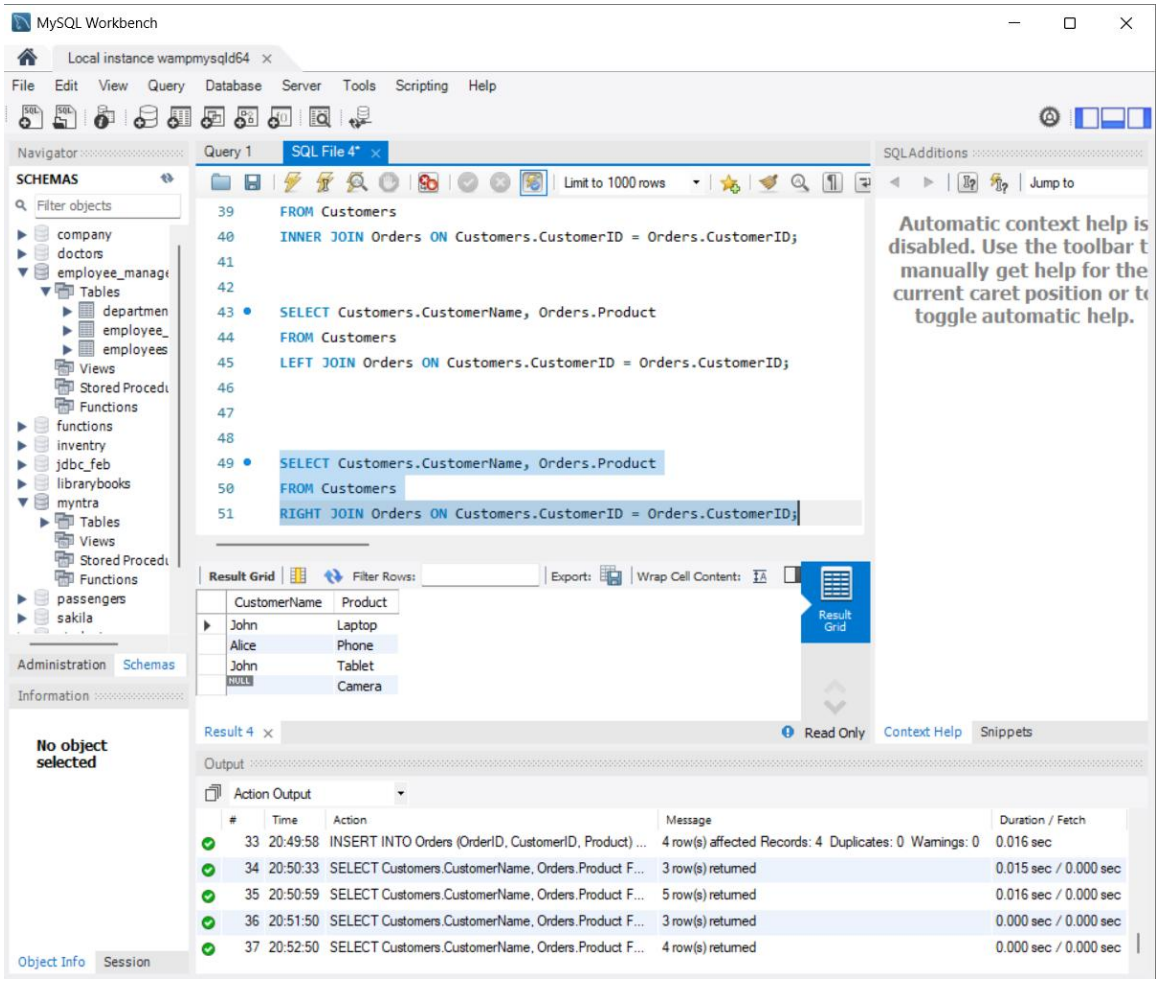
4.3 RIGHT JOIN

Returns all rows from the right table, even if there are no matches in the left table.
If no match is found, NULL values are returned for the left table's columns.

Query:

SELECT Customers.CustomerName, Orders.Product
FROM Customers
RIGHT JOIN Orders ON Customers.CustomerID = Orders.CustomerID;

➡ This shows all orders, including those without matching customers (NULL in CustomerName).



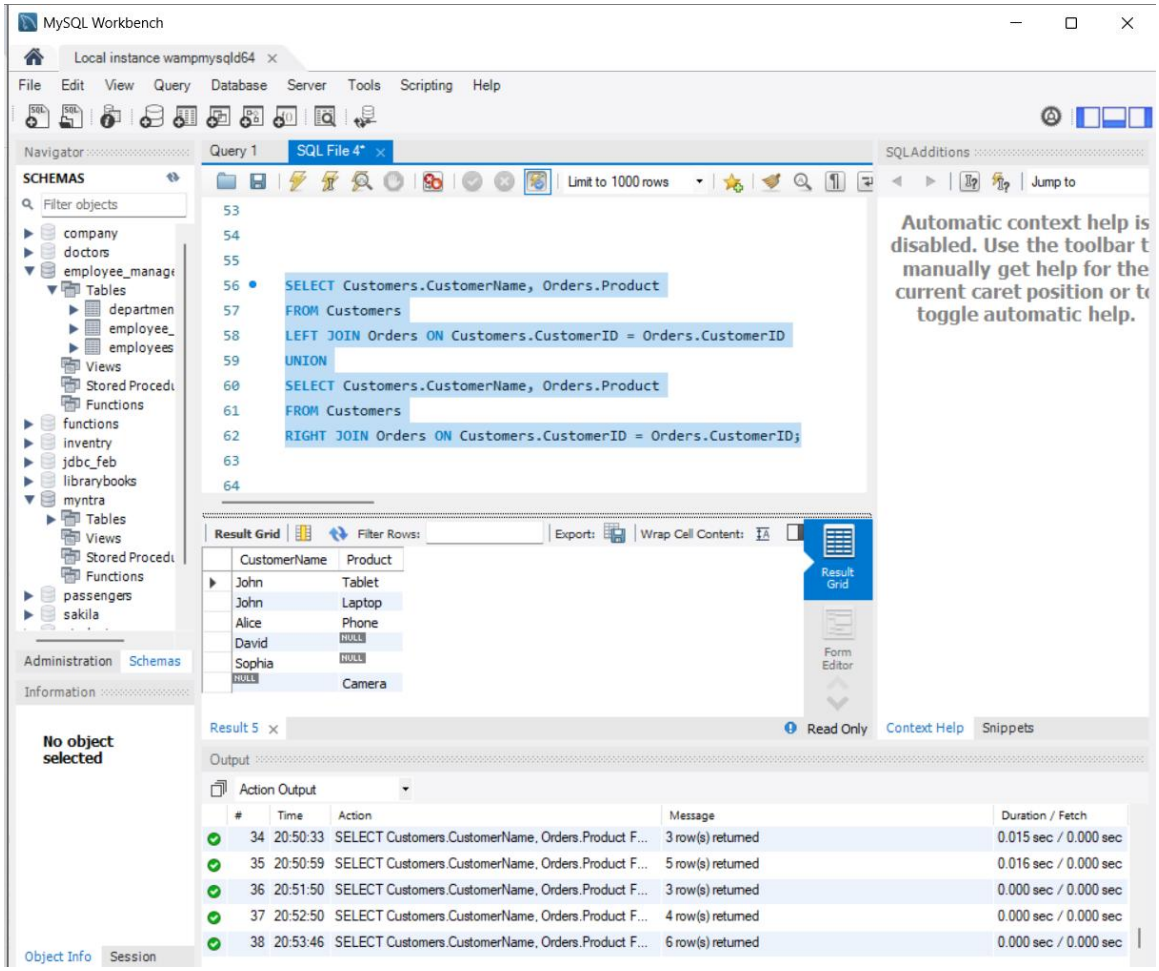
4.4 FULL OUTER JOIN (using UNION)

Returns all rows from both tables, with matches where possible.
Rows without a match in either table will show NULL values.

Query:

```
SELECT Customers.CustomerName, Orders.Product
FROM Customers
LEFT JOIN Orders ON Customers.CustomerID = Orders.CustomerID
UNION
SELECT Customers.CustomerName, Orders.Product
FROM Customers
RIGHT JOIN Orders ON Customers.CustomerID = Orders.CustomerID;
```

➡ This combines results of LEFT and RIGHT JOIN, showing all customers and all orders.



5. Conclusion

Through this exercise, we understood how INNER, LEFT, RIGHT, and FULL OUTER JOIN work in MySQL. These joins are essential for merging and analyzing data across multiple related tables.