Lesson 05 Demo 07

Performing Joins on Tables

Objective: To showcase various types of join operations in MySQL, illustrating how to effectively combine data from multiple tables

Tools required: MySQL

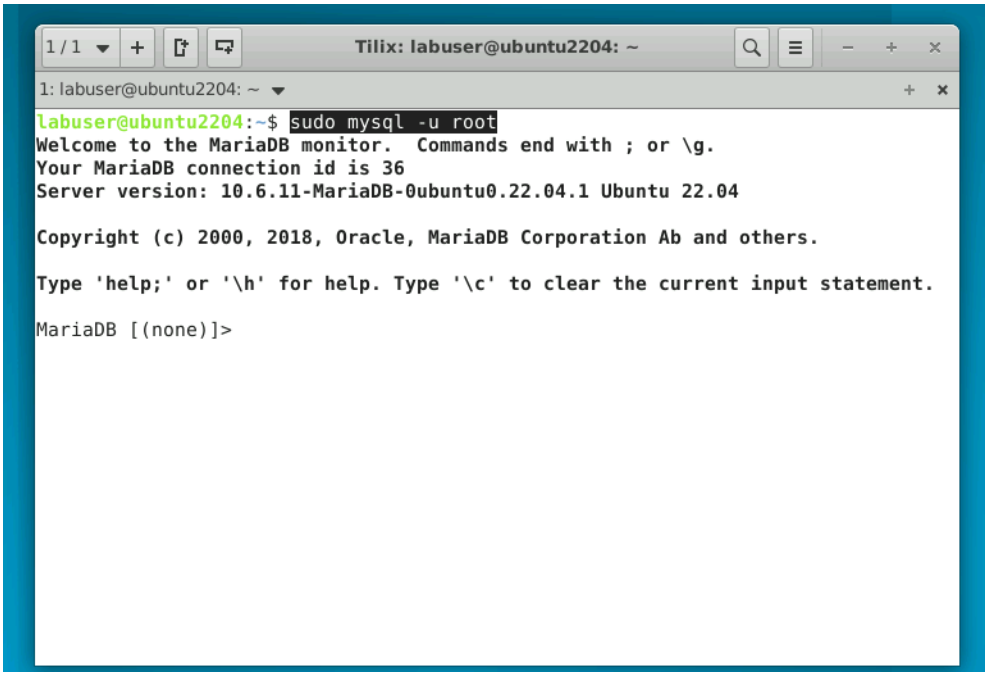
Prerequisites: None

Steps to be followed:

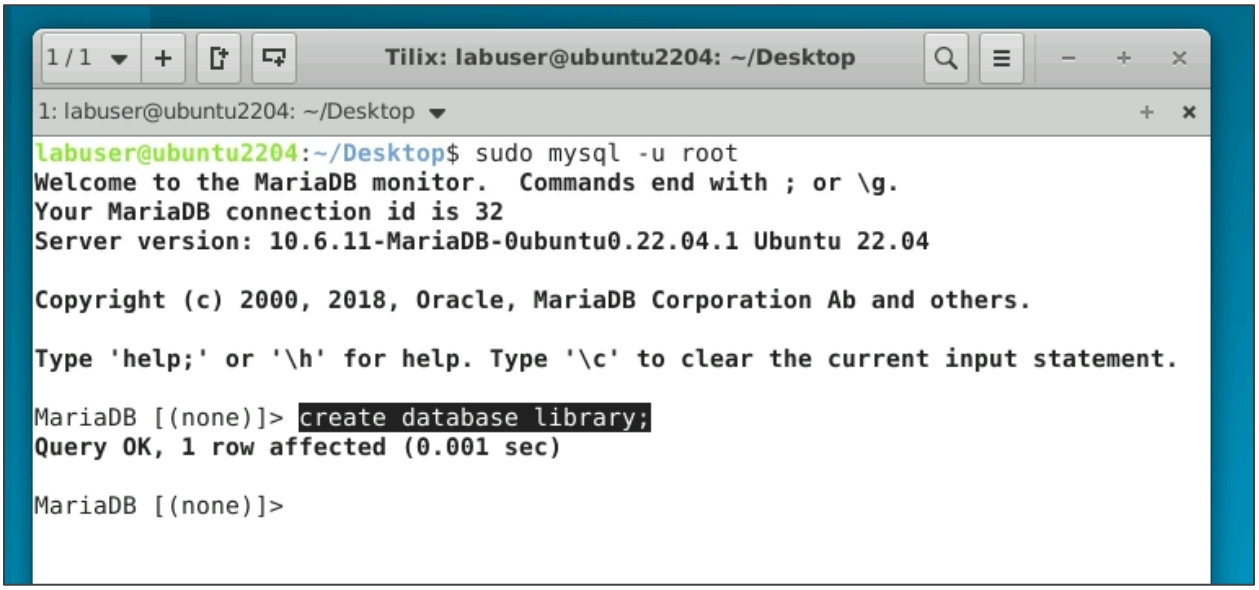
1. Set up a database and table
2. Perform different types of joins

Step 1: Set up a database and table

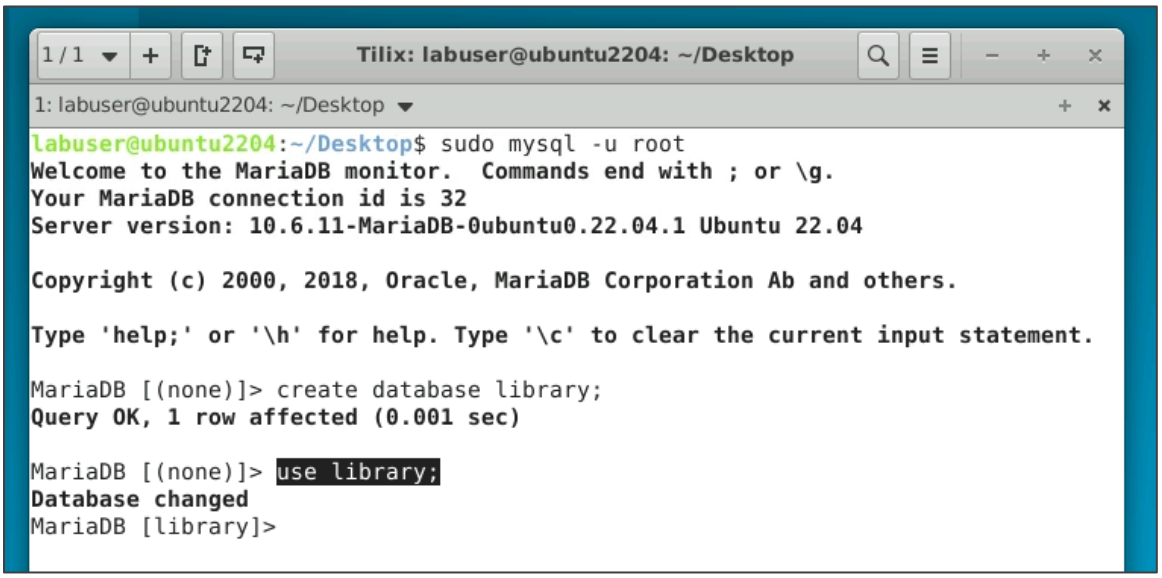
1.1 Open a terminal window and access MySQL as root user:
sudo mysql -u root



1.2 Create a new database named **library**:
create database library;

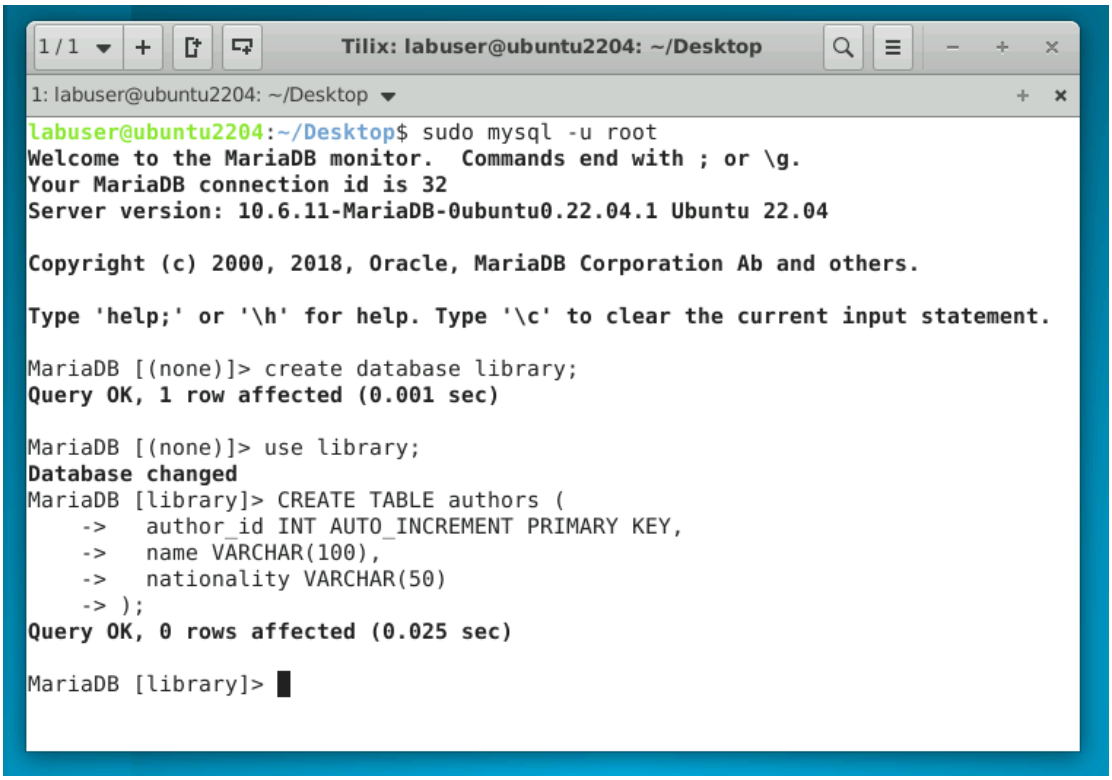


1.3 Select the **library** database:
use library;



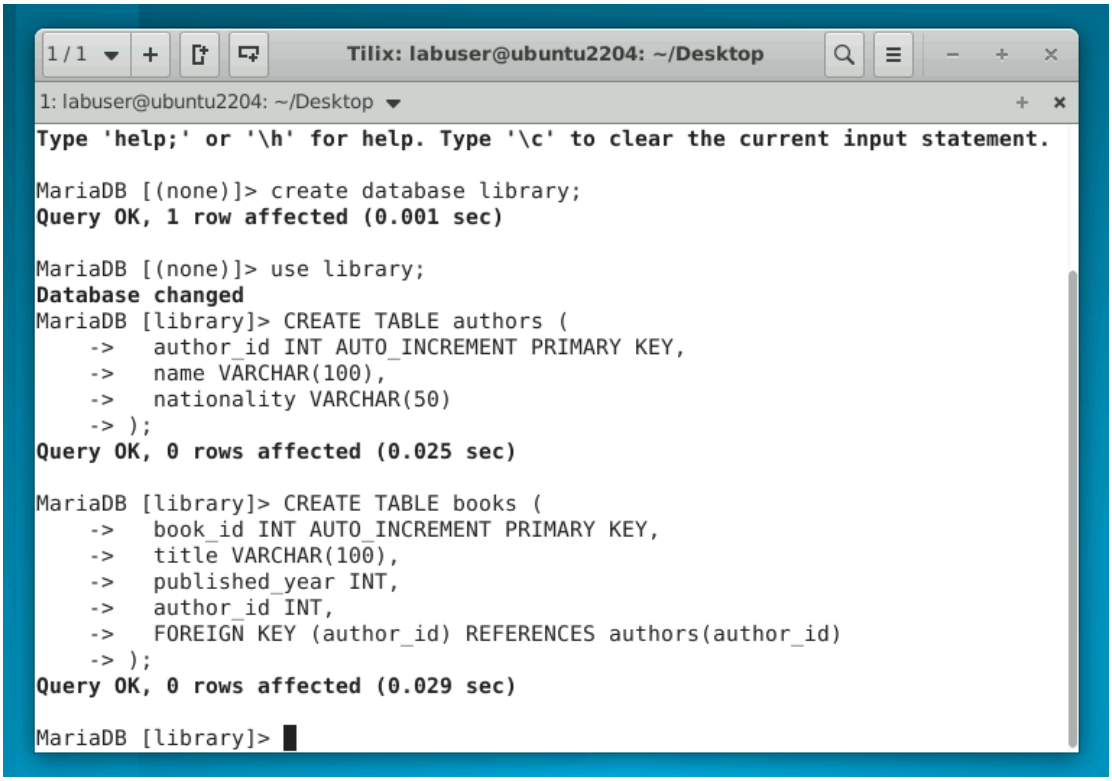
1.4 Create an **authors** table with relevant fields:

```
CREATE TABLE authors (
  author_id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(100),
  nationality VARCHAR(50)
);
```



1.5 Create a **books** table with relevant fields:

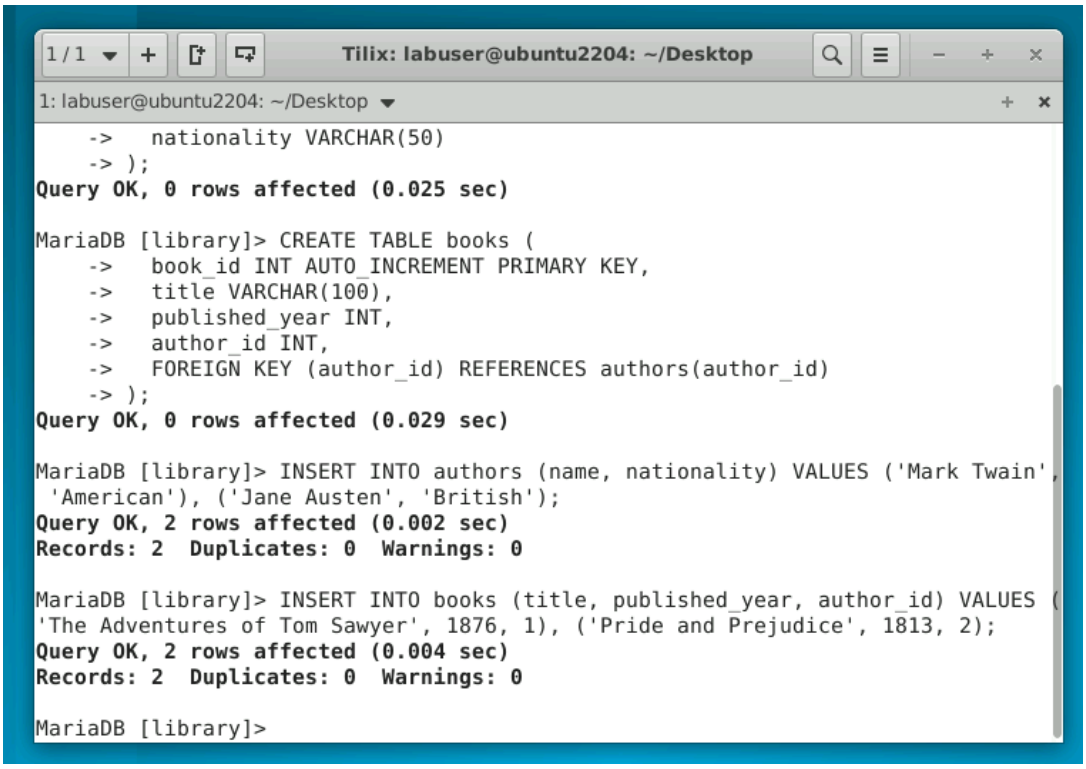
```
CREATE TABLE books (
  book_id INT AUTO_INCREMENT PRIMARY KEY,
  title VARCHAR(100),
  published_year INT,
  author_id INT,
  FOREIGN KEY (author_id) REFERENCES authors(author_id)
);
```



1.6 Insert data into **authors** and **books** tables:

INSERT INTO authors (name, nationality) VALUES ('Mark Twain', 'American'), ('Jane Austen', 'British');

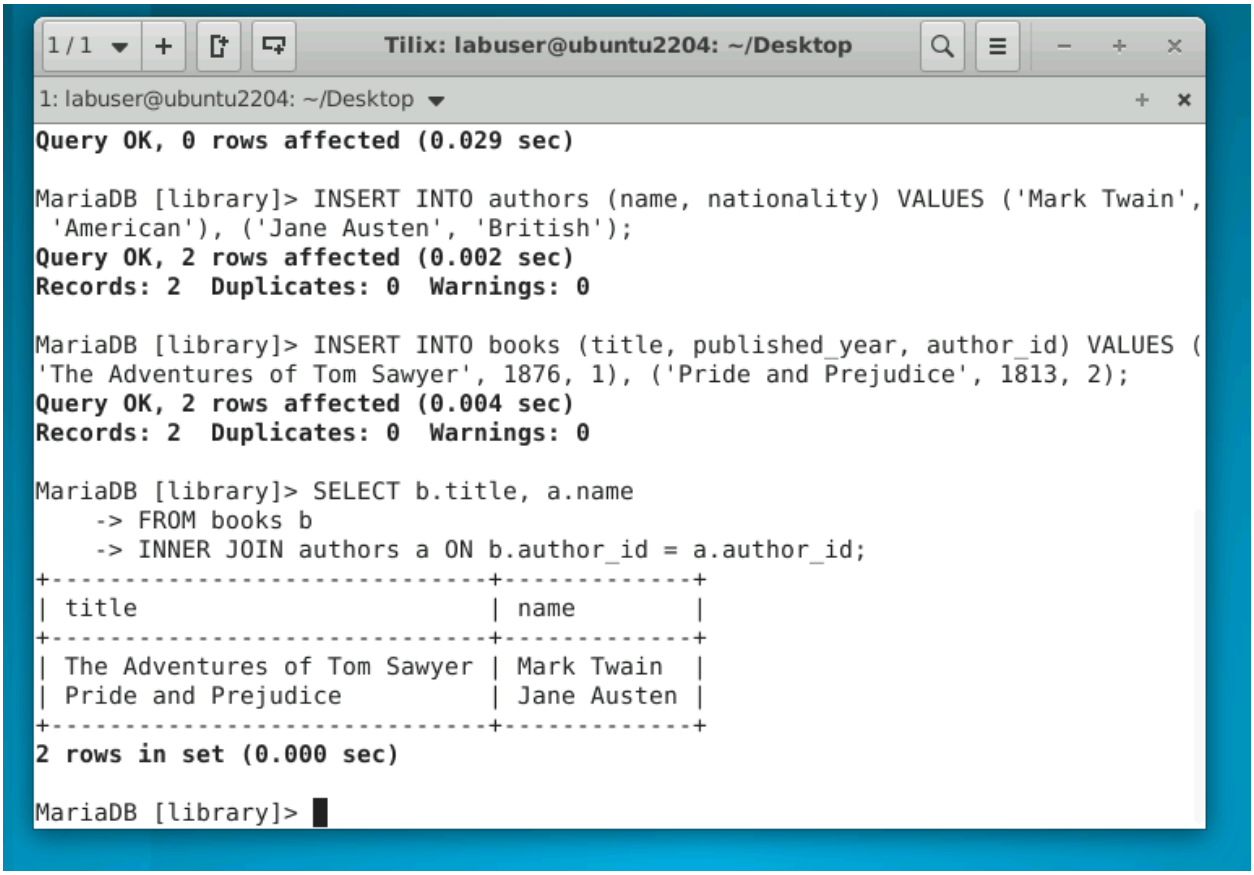
INSERT INTO books (title, published_year, author_id) VALUES ('The Adventures of Tom Sawyer', 1876, 1), ('Pride and Prejudice', 1813, 2);



Step 2: Perform different types of joins

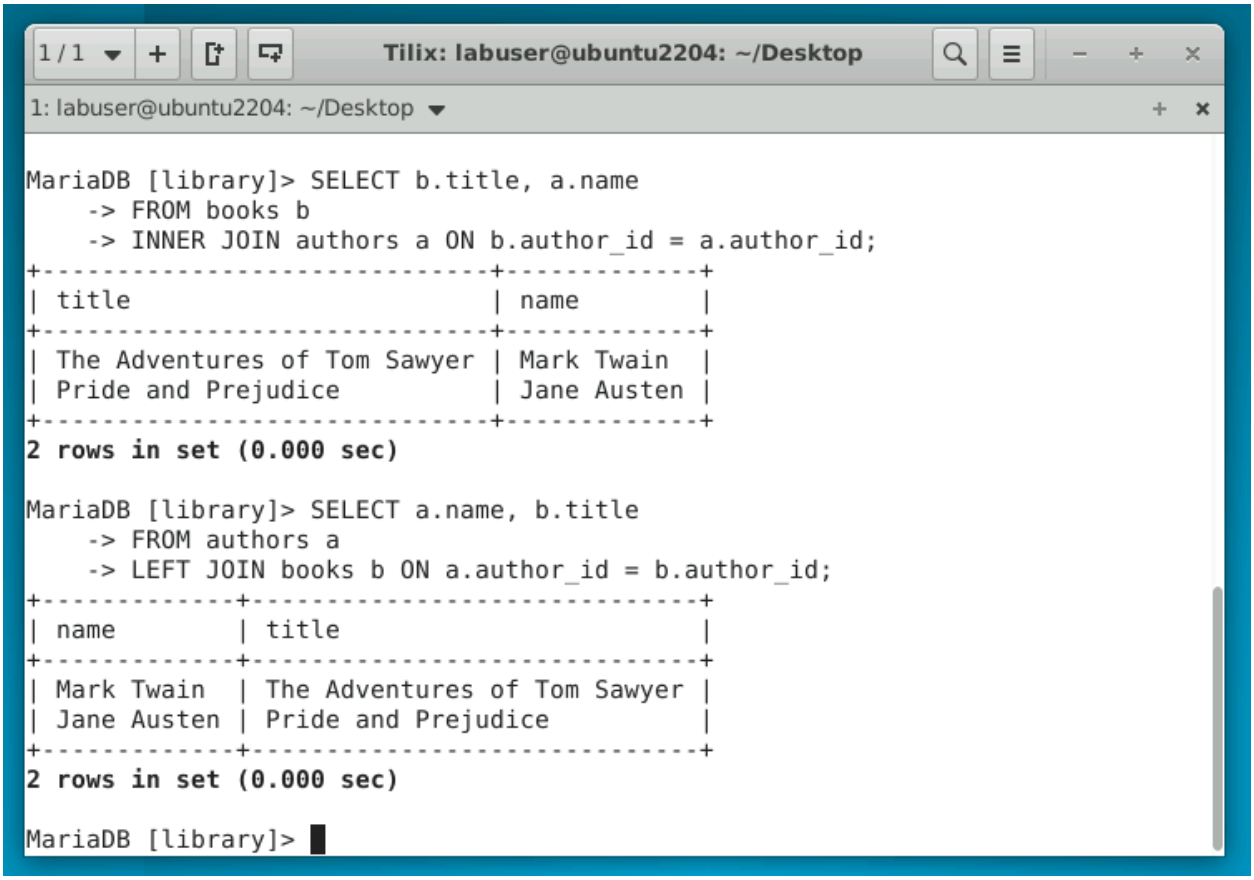
2.1 Join **books** and **authors** to display book titles with their authors:

```
SELECT b.title, a.name
FROM books b
INNER JOIN authors a ON b.author_id = a.author_id;
```



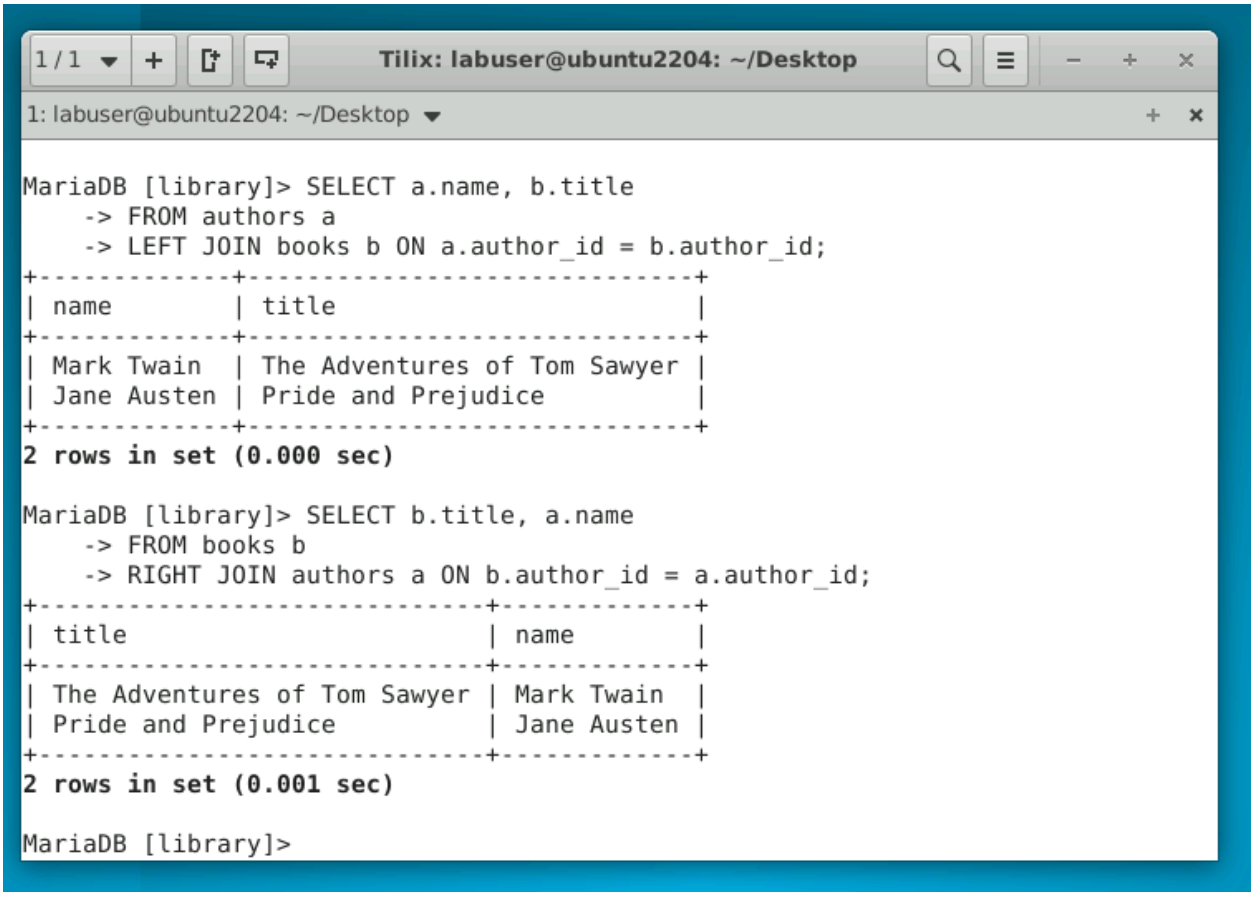
2.2 Perform a left join to display all the authors and their books (including authors with no books):

SELECT a.name, b.title
FROM authors a
LEFT JOIN books b ON a.author_id = b.author_id;



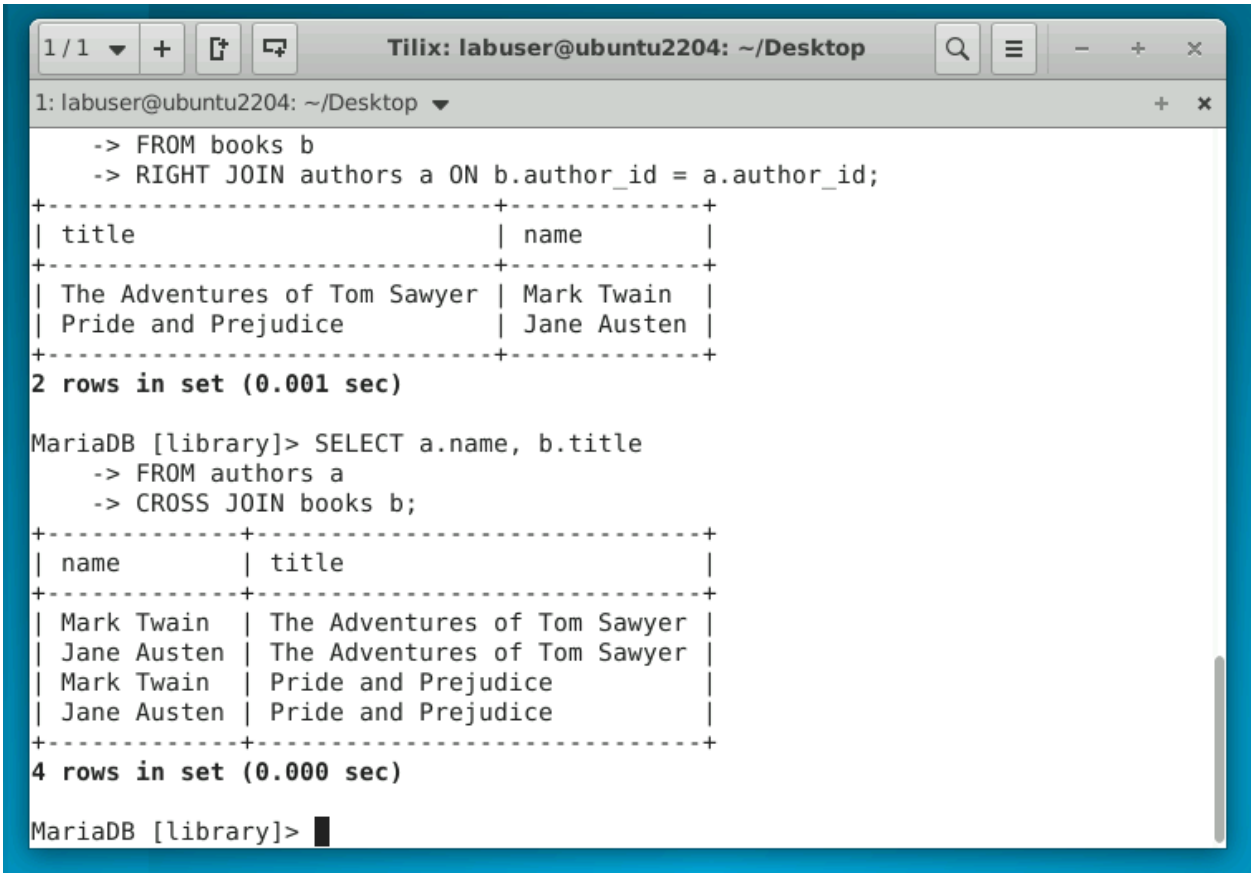
2.3 Perform a right join to display all the books and their authors (including books with no listed authors):

SELECT b.title, a.name
FROM books b
RIGHT JOIN authors a ON b.author_id = a.author_id;



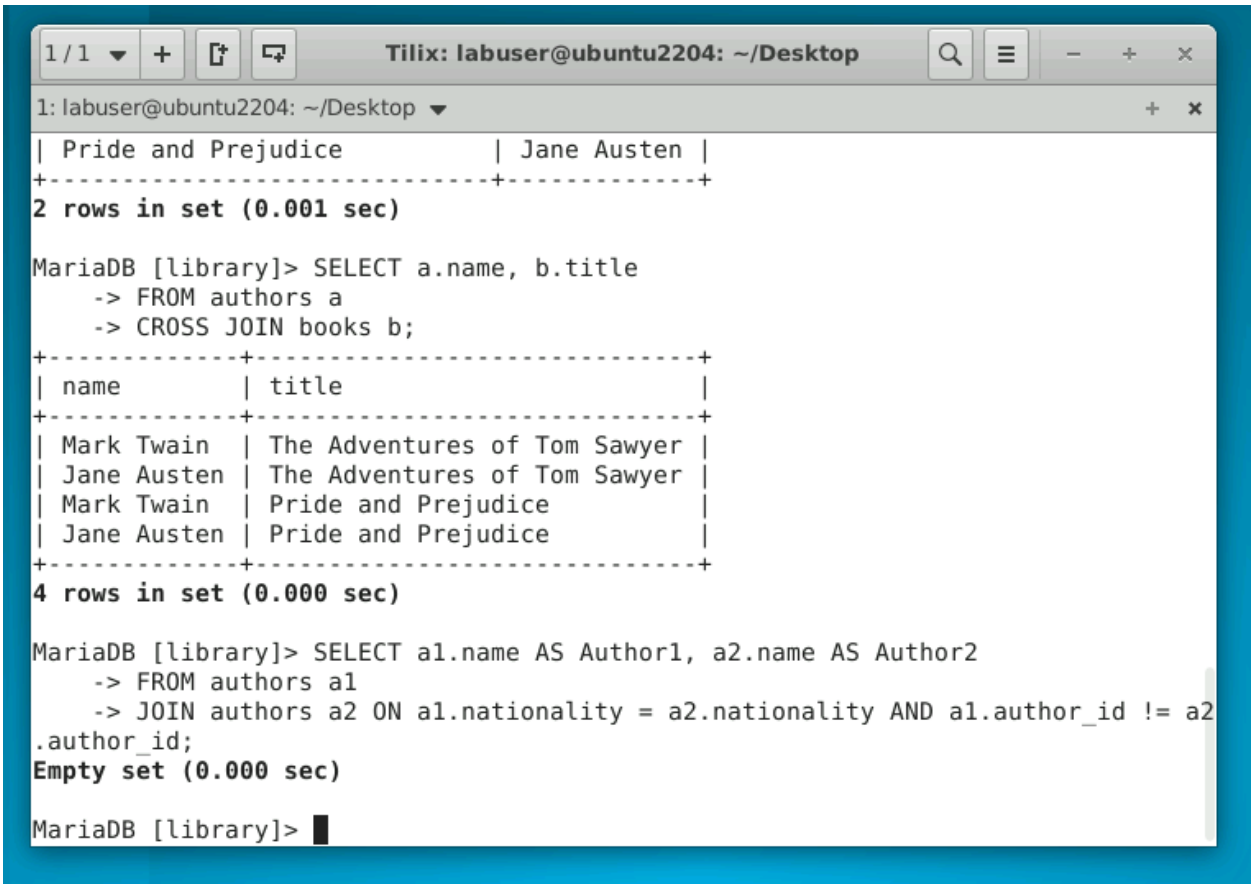
2.4 Perform a cross-join to combine all the authors with all books (cartesian product):

SELECT a.name, b.title
FROM authors a
CROSS JOIN books b;



2.5 Perform a self-join on **authors** to find the authors from the same country: (Assuming additional data)

SELECT a1.name AS Author1, a2.name AS Author2
FROM authors a1
JOIN authors a2 ON a1.nationality = a2.nationality AND a1.author_id != a2.author_id;



By following these steps, you have successfully performed different types of joins in MySQL, demonstrating the ability to link and analyze data across multiple tables.