PARIJAT PAL | 23BCC70037 | 23BCC1-A |ADBMS

EXPERIMENT 1.1

Title: Create Author and Book Tables using DDL Commands

Description:

Problem Statement

You are tasked with designing a basic book management system. Create two tables — **Authors** and **Books** — to represent a one-to-many relationship (one author can write multiple books). Use proper **primary and foreign key constraints** while designing the schema.

Input Format:

Table **Authors** with columns:

- **author_id**(INT, Primary Key)
- **name** (VARCHAR(50))
- **country** (VARCHAR(50))

Table **Books** with columns:

- **book_id** (INT, Primary Key)
- **title** (VARCHAR(100))
- **author_id** (INT, Foreign Key referencing Authors)

Output Format:

- Authors and Books tables created. Print description of the table.

Constraints:

- The **author_id** in **Books** must exist in the **Authors** table.
- Use appropriate data types and constraints.
- **name** and **country** should allow up to 50 characters.

Sample Input:
Write query to create tables for Authors and Books

Sample Output:

`authors`
Table:

Field	Type	Null	Key	Default	Extra
author_id	int	NO	PRI	NULL	
name	varchar(50)	YES		NULL	
country	varchar(50)	YES		NULL	

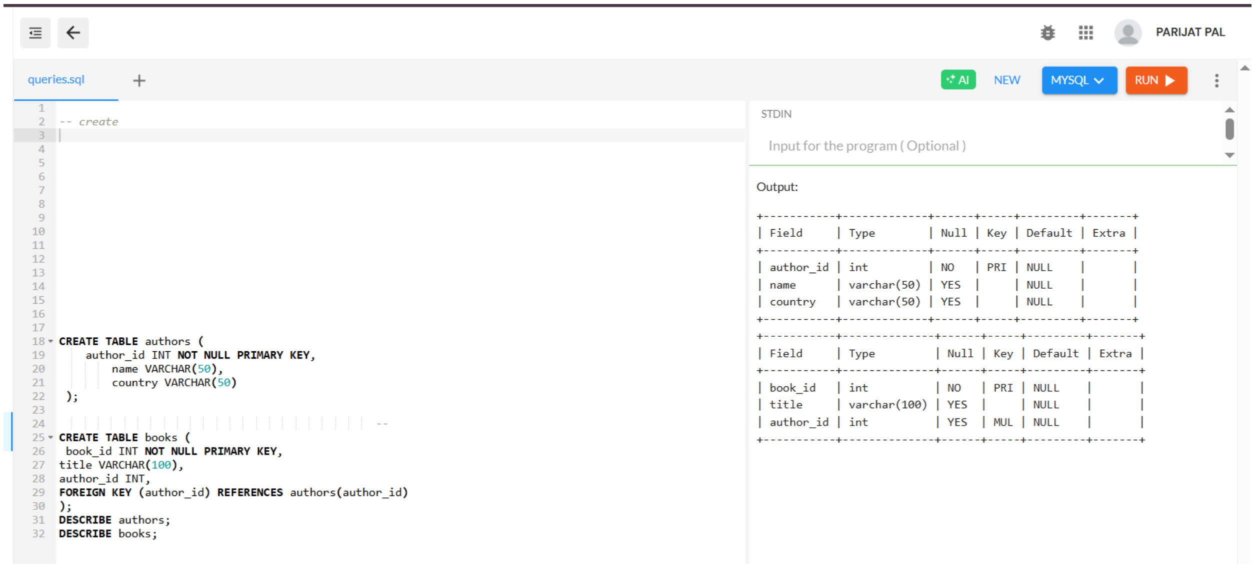
`books`
Table:

Field	Type	Null	Key	Default	Extra
book_id	int	NO	PRI	NULL	
title	varchar(100)	YES		NULL	
author_id	int	YES	MUL	NULL	

Query:

```
CREATE TABLE authors (
    author_id INT NOT NULL PRIMARY KEY,
        name VARCHAR(50),
        country VARCHAR(50)
 );


                                               --
CREATE TABLE books (
 book_id INT NOT NULL PRIMARY KEY,
title VARCHAR(100),
author_id INT,
FOREIGN KEY (author_id) REFERENCES authors(author_id)
);
DESCRIBE authors;
DESCRIBE books;
```

Output:



```sql

-- create
















CREATE TABLE authors (
    author_id INT NOT NULL PRIMARY KEY,
        name VARCHAR(50),
        country VARCHAR(50)
 );

                                                   --
CREATE TABLE books (
 book_id INT NOT NULL PRIMARY KEY,
title VARCHAR(100),
author_id INT,
FOREIGN KEY (author_id) REFERENCES authors(author_id)
);
DESCRIBE authors;
DESCRIBE books;
```

Output:

```
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| author_id | int         | NO   | PRI | NULL    |       |
| name      | varchar(50) | YES  |     | NULL    |       |
| country   | varchar(50) | YES  |     | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
+-----------+--------------+------+-----+---------+-------+
| Field     | Type         | Null | Key | Default | Extra |
+-----------+--------------+------+-----+---------+-------+
| book_id   | int          | NO   | PRI | NULL    |       |
| title     | varchar(100) | YES  |     | NULL    |       |
| author_id | int          | YES  | MUL | NULL    |       |
+-----------+--------------+------+-----+---------+-------+
```

Learning Outcome:

Learned how to create a tables.

Learned how to insert data into the tables.

PARIJAT PAL | 23BCC70037 | 23BCC1-A |ADBMS

EXPERIMENT 1.2

Title: Insert Sample Records into Author and Book Tables

Description:

After creating the Authors and Books tables, your next task is to insert sample records. Insert **at least 3 authors and 3 books,** ensuring books reference valid authors using the foreign key.

Input Format:

- Pre-existing Authors and Books table structures from Problem 1.

Output Format:

`Authors`
Table:

author_id	name	country
1	Ashish	India
2	Smaran	USA
3	Vaibhav	UK

`Books`
Table:

book_id	title	author_id
101	Data Science Basics	1
102	AI in Education	2
103	SQL Simplified	1

Constraints:

- Insert meaningful names and countries (e.g., Ashish, Smaran, Vaibhav).
- Insert book titles that are easy to associate with those authors.
- Use valid foreign keys.

Sample Input:
Input tables Authors and Books

Sample Output:

Authors Table:

author_id	name	country
1	Ashish	India
2	Smaran	USA
3	Vaibhav	UK

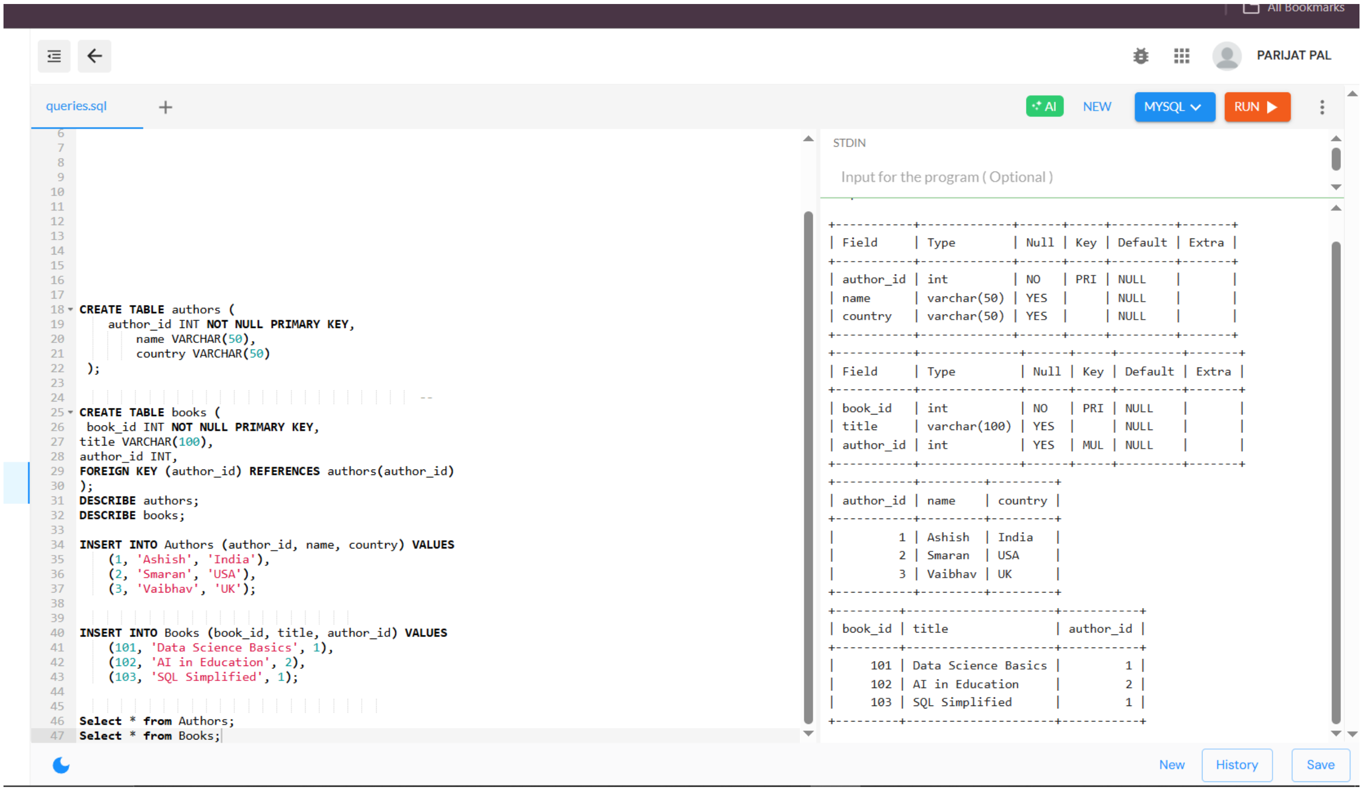
Books Table

book_id	title	author_id
101	Data Science Basics	1
102	AI in Education	2
103	SQL Simplified	1

Query:

```
INSERT INTO Authors (author_id, name, country) VALUES
   (1, 'Ashish', 'India'),
   (2, 'Smaran', 'USA'),
   (3, 'Vaibhav', 'UK');



INSERT INTO Books (book_id, title, author_id) VALUES
   (101, 'Data Science Basics', 1),
   (102, 'AI in Education', 2),
   (103, 'SQL Simplified', 1);



Select * from Authors;
Select * from Books;
```

Output:



```sql
CREATE TABLE authors (
    author_id INT NOT NULL PRIMARY KEY,
        name VARCHAR(50),
        country VARCHAR(50)
 );

                                                --
CREATE TABLE books (
 book_id INT NOT NULL PRIMARY KEY,
title VARCHAR(100),
author_id INT,
FOREIGN KEY (author_id) REFERENCES authors(author_id)
);
DESCRIBE authors;
DESCRIBE books;

INSERT INTO Authors (author_id, name, country) VALUES
    (1, 'Ashish', 'India'),
    (2, 'Smaran', 'USA'),
    (3, 'Vaibhav', 'UK');


INSERT INTO Books (book_id, title, author_id) VALUES
    (101, 'Data Science Basics', 1),
    (102, 'AI in Education', 2),
    (103, 'SQL Simplified', 1);


Select * from Authors;
Select * from Books;
```

```
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| author_id | int         | NO   | PRI | NULL    |       |
| name      | varchar(50) | YES  |     | NULL    |       |
| country   | varchar(50) | YES  |     | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
+-----------+--------------+------+-----+---------+-------+
| Field     | Type         | Null | Key | Default | Extra |
+-----------+--------------+------+-----+---------+-------+
| book_id   | int          | NO   | PRI | NULL    |       |
| title     | varchar(100) | YES  |     | NULL    |       |
| author_id | int          | YES  | MUL | NULL    |       |
+-----------+--------------+------+-----+---------+-------+
+-----------+---------+---------+
| author_id | name    | country |
+-----------+---------+---------+
|         1 | Ashish  | India   |
|         2 | Smaran  | USA     |
|         3 | Vaibhav | UK      |
+-----------+---------+---------+
+---------+---------------------+-----------+
| book_id | title               | author_id |
+---------+---------------------+-----------+
|     101 | Data Science Basics |         1 |
|     102 | AI in Education     |         2 |
|     103 | SQL Simplified      |         1 |
+---------+---------------------+-----------+
```

Learning Outcome:

Learned how to create a tables.

Learned how to insert data into the records of the table.

PARIJAT PAL | 23BCC70037 | 23BCC1-A |ADBMS

EXPERIMENT 1.3

Title: Retrieve Book Titles Along with Author Information Using INNER JOIN

Description:

Given two tables, Authors and Books, retrieve the titles of all books along with their **author's name and country.** This involves creating tables, inserting data, and using an INNER JOIN to combine records based on author_id.

Input Format:

- Pre-existing Authors and Books table structures from Problem 1.

Table **Authors** with columns:

- **author_id** (INT, Primary Key)
- **name** (VARCHAR(50))
- **country** (VARCHAR(50))

Table **Books** with columns:

- **book_id** (INT, Primary Key)
- **title** (VARCHAR(100))
- **author_id** (INT, Foreign Key referencing Authors)

Output Format:

- A list of books with their **title, name** of the author, and **country** of the author.

Constraints:

- Each book must be linked to one valid author.
- Each author can be linked to one or more books.
- No NULLs are allowed in the author_id field of the Books table.
- Use the same data as shown in the sample table.

Sample Input:

`Authors`
Table

author_id	name	country
1	Ashish	India

author_id	name	country
2	Smaran	USA
3	Vaibhav	UK

`Books` Table

book_id	title	author_id
101	Data Science Basics	1
102	AI in Education	2
103	SQL Simplified	1

Sample Output

title	name	country
Data Science Basics	Ashish	India
AI in Education	Smaran	USA
SQL Simplified	Ashish	India

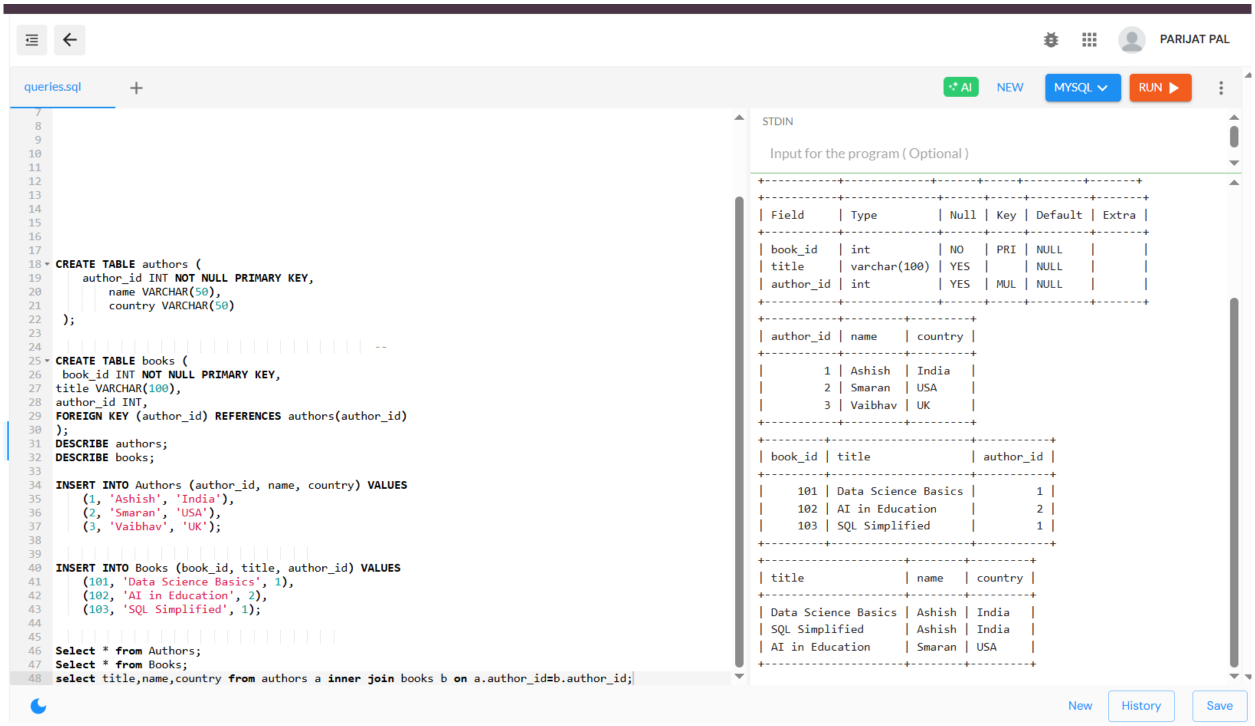
Explanation:

- The INNER JOIN links each book's author_id to the Authors table.
- The result shows the book title along with the author’s name and country.
- For example, "SQL Simplified" is written by Ashish from India.

Query:

select title,name,country from authors a inner join books b on a.author_id=b.author_id;

Output:



```sql
CREATE TABLE authors (
    author_id INT NOT NULL PRIMARY KEY,
        name VARCHAR(50),
        country VARCHAR(50)
 );

                                                  --
CREATE TABLE books (
 book_id INT NOT NULL PRIMARY KEY,
title VARCHAR(100),
author_id INT,
FOREIGN KEY (author_id) REFERENCES authors(author_id)
);
DESCRIBE authors;
DESCRIBE books;

INSERT INTO Authors (author_id, name, country) VALUES
    (1, 'Ashish', 'India'),
    (2, 'Smaran', 'USA'),
    (3, 'Vaibhav', 'UK');


INSERT INTO Books (book_id, title, author_id) VALUES
    (101, 'Data Science Basics', 1),
    (102, 'AI in Education', 2),
    (103, 'SQL Simplified', 1);


Select * from Authors;
Select * from Books;
select title,name,country from authors a inner join books b on a.author_id=b.author_id;
```

```
+-----------+-------------+------+-----+---------+-------+
+-----------+--------------+------+-----+---------+-------+
| Field     | Type         | Null | Key | Default | Extra |
+-----------+--------------+------+-----+---------+-------+
| book_id   | int          | NO   | PRI | NULL    |       |
| title     | varchar(100) | YES  |     | NULL    |       |
| author_id | int          | YES  | MUL | NULL    |       |
+-----------+--------------+------+-----+---------+-------+
+-----------+---------+---------+
| author_id | name    | country |
+-----------+---------+---------+
|         1 | Ashish  | India   |
|         2 | Smaran  | USA     |
|         3 | Vaibhav | UK      |
+-----------+---------+---------+
+---------+---------------------+-----------+
| book_id | title               | author_id |
+---------+---------------------+-----------+
|     101 | Data Science Basics |         1 |
|     102 | AI in Education     |         2 |
|     103 | SQL Simplified      |         1 |
+---------+---------------------+-----------+
+---------------------+--------+---------+
| title               | name   | country |
+---------------------+--------+---------+
| Data Science Basics | Ashish | India   |
| SQL Simplified      | Ashish | India   |
| AI in Education     | Smaran | USA     |
+---------------------+--------+---------+
```

Learning Outcome:

Learned about Joins.

Learned about how to implement inner joins.