

Create Author and Book Tables using DDL Commands

Score: 5 | Difficulty: easy

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**

SQL

```
1 create table authors(author_id int primary key not null, name varchar(50), country varchar(50));
2 create table books(book_id int primary key not null, title varchar(100), author_id int, foreign key(author_id) Reference
3 describe authors;
4 describe books;
```

Test & Results

Custom Input

Test Cases

Run Code

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 | |

144 ms

37m

Insert Sample Records into Author and Book Tables

Score: 5 | Difficulty: easy

1

2

3

Problem Statement

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 |

SQL

1 insert into authors values(1,'Ashish','India'),(2,'Smaran','USA'),(3,'Vaibhav','UK');

2 insert into books values(101,'Data Science Basics',1),(102,'AI in Education',2),(103,'SQL Simplified',1);

3 select * from authors;

4 select * from books;

Test & Results

Submit

Custom Input

Custom Input

Test Cases

Run Code

Output:

| 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 |

160 ms

37m

Retrieve Book Titles Along with Author Information Using INNER JOIN

Score: 5 | Difficulty: easy

Problem Statement

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:

SQL

```
1 select b.title, a.name, a.country from Authors a inner join Books b on a.author_id = b.author_id;
```

Test & Results

Custom Input

Test Cases

Run Code

Output:

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

186 ms