25 CASE STUDY - Library

Problem Statement

- A Library wants to maintain the record of books, members, book issue, book returns and fines collected for late returns in a database
- The database can be loaded with the book information
- Students can register with the library to be a member
- Books can be issued to students with valid library membership
- A student can keep an issued book with him/her for a maximum period of 2 weeks from the date of issue, beyond which a fine will be charged
- Fine is calculated on the delay in days of return for 1-7 days 2Rs , 8-30 days 10Rs and above 30 days 5Rs will be charged per day

Database Schema

Entity	Attribute	Primary Key	Foreign Key
BOOK	Book_id, Title, Language_id, MRP, Publisher_id, Published_date, Volume, Status	Book_id	Language_id, Pblisher_id
AUTHOR	Author_id, Name, Email, Phone, Status	Author_id	
BOOK_AUTHOR	Book_id, Author_id	Book_id, Author_id	Book_id, Author_id
PUBLISHER	Publisher_id, Name, Address	Publisher_id	
MEMBER	Member_id, Name, Branch_code, Roll_no, Phone, Email, Date_of_join, Status	Member_id	
BOOK_ISSUE	Issue_id, Date_of_issue, Book_id, Member_id, Expected_date_of_return, Status	Issue_id	Book_id, Member_id
BOOK_RETURN	Issue_id, Actual_date_of_return, Late_days, Late_fee	Issue_id	Issue_id
LANGUAGE	Language_id, Name	Language_id	
LATE_FEE_RULE	FromDays, ToDays, Amount	composite	key

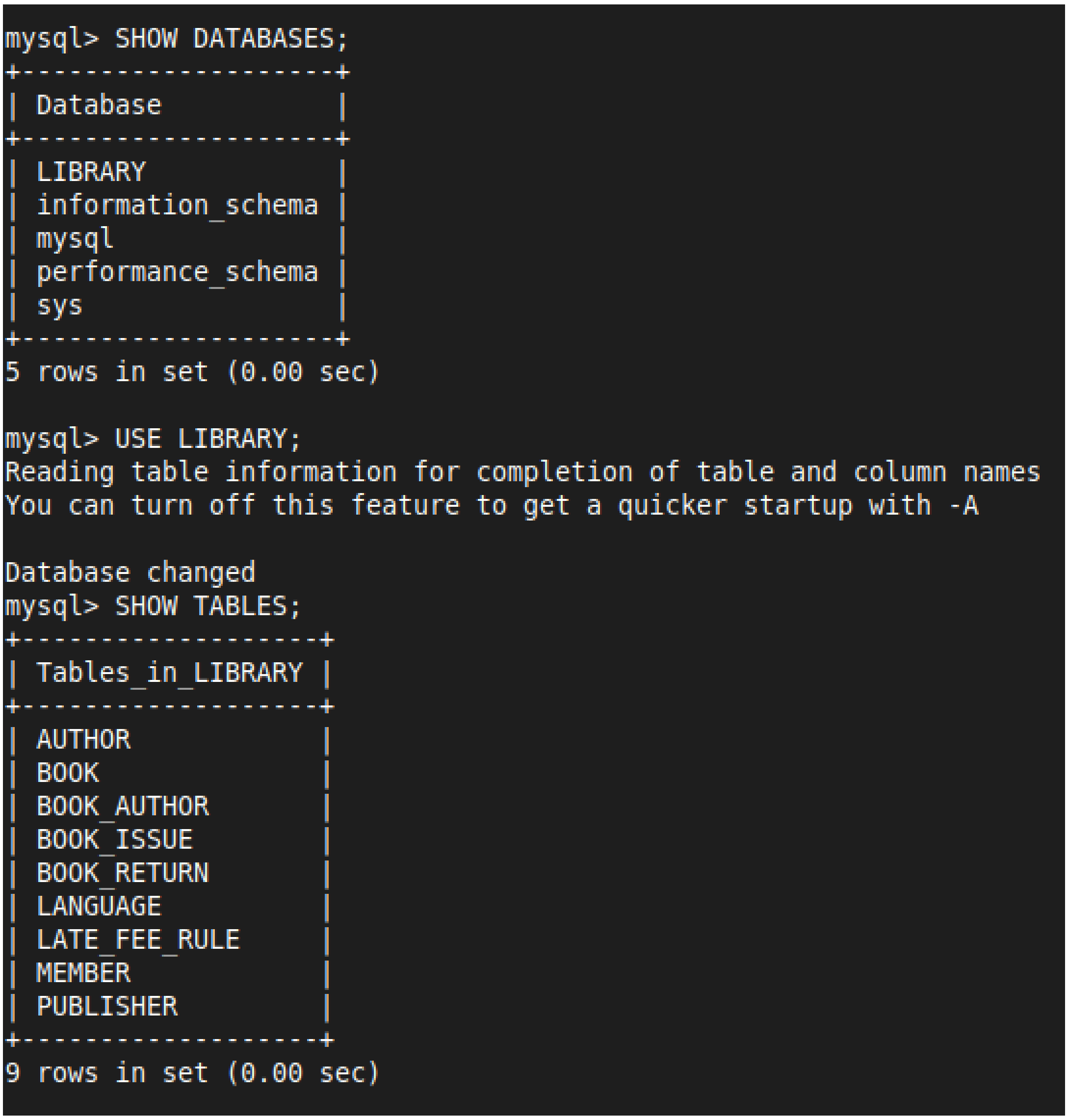
SQL - DDL

1. Create and Use Database

```
CREATE DATABASE LIBRARY;
```

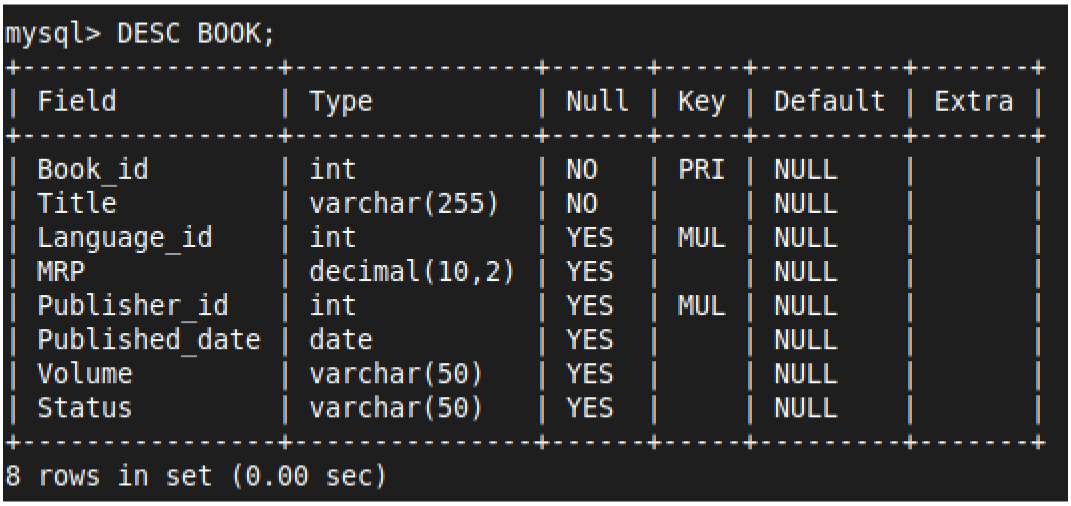
```
USE LIBRARY;
```

2. Creating Tables



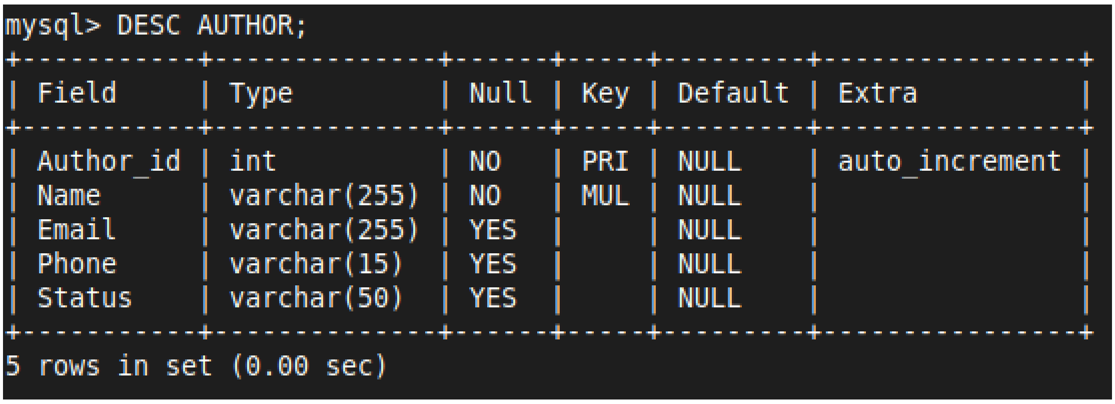
```
mysql> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| LIBRARY            |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.00 sec)

mysql> USE LIBRARY;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> SHOW TABLES;
+-------------------+
| Tables_in_LIBRARY |
+-------------------+
| AUTHOR            |
| BOOK              |
| BOOK_AUTHOR       |
| BOOK_ISSUE        |
| BOOK_RETURN       |
| LANGUAGE          |
| LATE_FEE_RULE     |
| MEMBER            |
| PUBLISHER         |
+-------------------+
9 rows in set (0.00 sec)
```

```
CREATE TABLE BOOK (
    Book_id INT PRIMARY KEY,
    Title VARCHAR(255) NOT NULL,
    Language_id INT,
    MRP DECIMAL(10, 2),
    Publisher_id INT,
    Published_date DATE,
    Volume VARCHAR(50),
    Status VARCHAR(50),
);
```



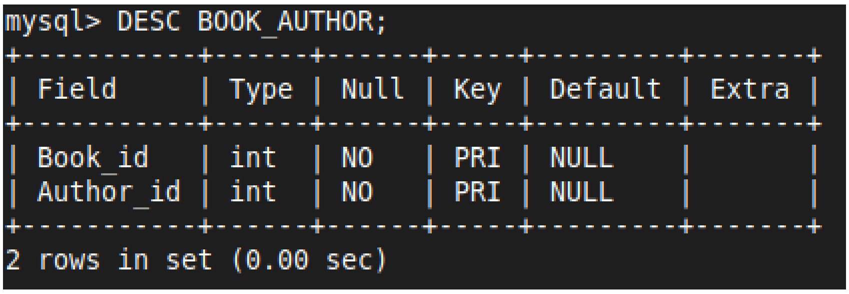
```
mysql> DESC BOOK;
+----------------+---------------+------+-----+---------+-------+
| Field          | Type          | Null | Key | Default | Extra |
+----------------+---------------+------+-----+---------+-------+
| Book_id        | int           | NO   | PRI | NULL    |       |
| Title          | varchar(255)  | NO   |     | NULL    |       |
| Language_id    | int           | YES  | MUL | NULL    |       |
| MRP            | decimal(10,2) | YES  |     | NULL    |       |
| Publisher_id   | int           | YES  | MUL | NULL    |       |
| Published_date | date          | YES  |     | NULL    |       |
| Volume         | varchar(50)   | YES  |     | NULL    |       |
| Status         | varchar(50)   | YES  |     | NULL    |       |
+----------------+---------------+------+-----+---------+-------+
8 rows in set (0.00 sec)
```

```
CREATE TABLE AUTHOR (
    Author_id INT PRIMARY KEY AUTO_INCREMENT,
    Name VARCHAR(255) NOT NULL,
    Email VARCHAR(255),
    Phone VARCHAR(15),
    Status VARCHAR(50)
);
```



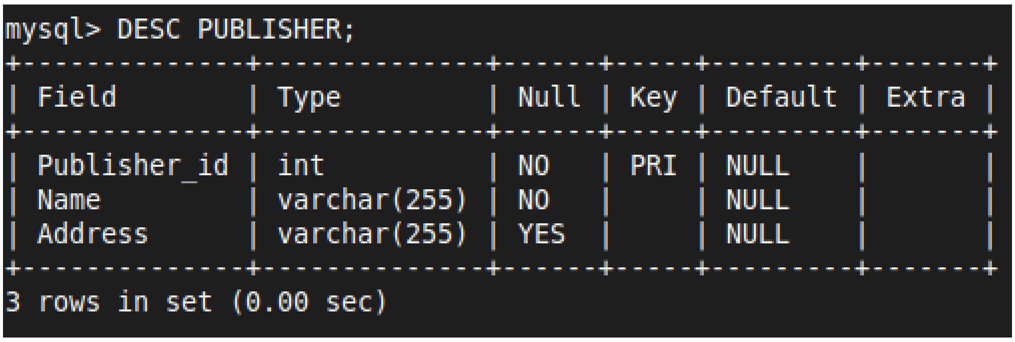
```
mysql> DESC AUTHOR;
+-----------+--------------+------+-----+---------+----------------+
| Field     | Type         | Null | Key | Default | Extra          |
+-----------+--------------+------+-----+---------+----------------+
| Author_id | int          | NO   | PRI | NULL    | auto_increment |
| Name      | varchar(255) | NO   | MUL | NULL    |                |
| Email     | varchar(255) | YES  |     | NULL    |                |
| Phone     | varchar(15)  | YES  |     | NULL    |                |
| Status    | varchar(50)  | YES  |     | NULL    |                |
+-----------+--------------+------+-----+---------+----------------+
5 rows in set (0.00 sec)
```

```
CREATE TABLE BOOK_AUTHOR (
    Book_id INT,
    Author_id INT,
    PRIMARY KEY (Book_id, Author_id),
    FOREIGN KEY (Book_id) REFERENCES BOOK(Book_id),
    FOREIGN KEY (Author_id) REFERENCES AUTHOR(Author_id)
);
```



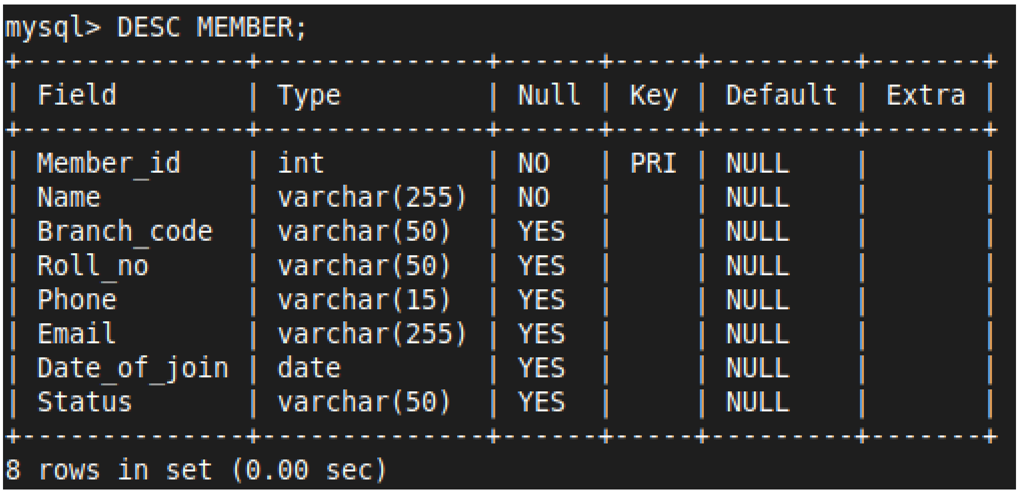
```
mysql> DESC BOOK_AUTHOR;
+-----------+------+------+-----+---------+-------+
| Field     | Type | Null | Key | Default | Extra |
+-----------+------+------+-----+---------+-------+
| Book_id   | int  | NO   | PRI | NULL    |       |
| Author_id | int  | NO   | PRI | NULL    |       |
+-----------+------+------+-----+---------+-------+
2 rows in set (0.00 sec)
```

```sql
CREATE TABLE PUBLISHER (
    Publisher_id INT PRIMARY KEY,
    Name VARCHAR(255) NOT NULL,
    Address VARCHAR(255)
);
```



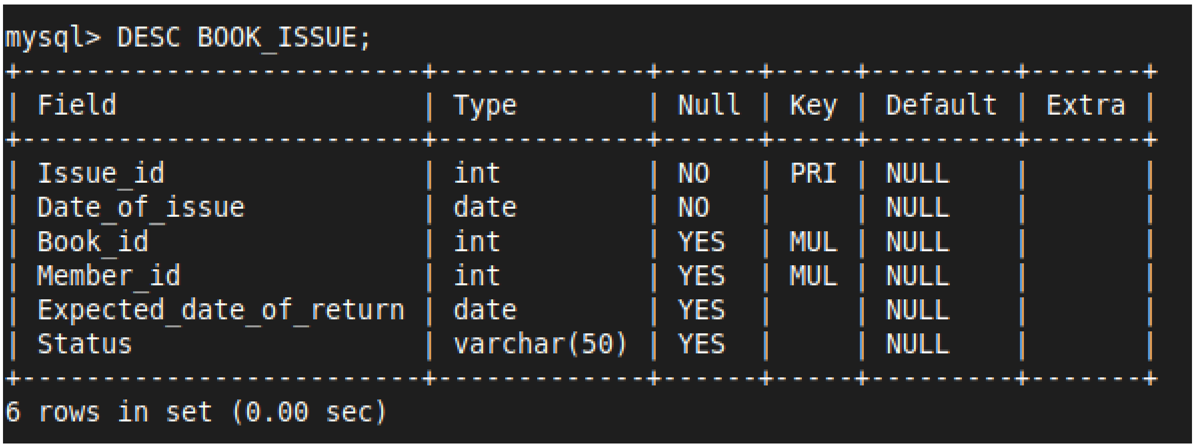
```
mysql> DESC PUBLISHER;
+--------------+--------------+------+-----+---------+-------+
| Field        | Type         | Null | Key | Default | Extra |
+--------------+--------------+------+-----+---------+-------+
| Publisher_id | int          | NO   | PRI | NULL    |       |
| Name         | varchar(255) | NO   |     | NULL    |       |
| Address      | varchar(255) | YES  |     | NULL    |       |
+--------------+--------------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```

```sql
CREATE TABLE MEMBER (
    Member_id INT PRIMARY KEY,
    Name VARCHAR(255) NOT NULL,
    Branch_code VARCHAR(50),
    Roll_no VARCHAR(50),
    Phone VARCHAR(15),
    Email VARCHAR(255),
    Date_of_join DATE,
    Status VARCHAR(50)
);
```



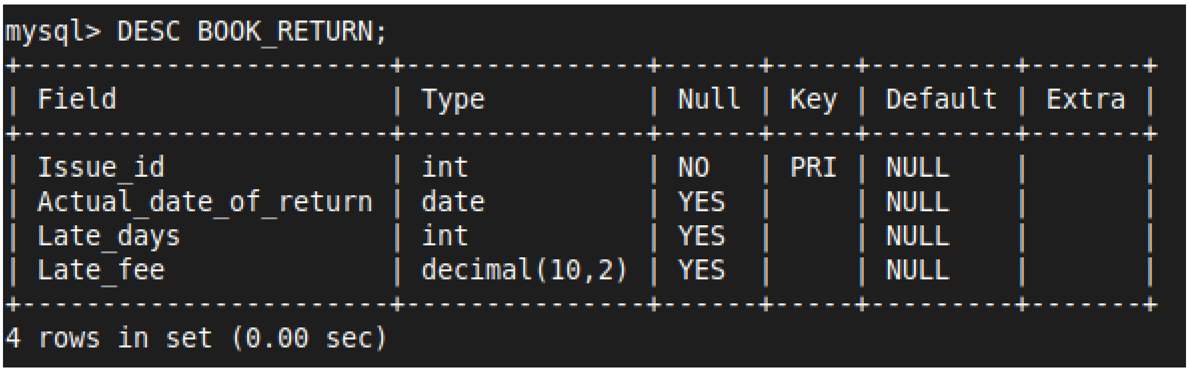
```
mysql> DESC MEMBER;
+--------------+--------------+------+-----+---------+-------+
| Field        | Type         | Null | Key | Default | Extra |
+--------------+--------------+------+-----+---------+-------+
| Member_id    | int          | NO   | PRI | NULL    |       |
| Name         | varchar(255) | NO   |     | NULL    |       |
| Branch_code  | varchar(50)  | YES  |     | NULL    |       |
| Roll_no      | varchar(50)  | YES  |     | NULL    |       |
| Phone        | varchar(15)  | YES  |     | NULL    |       |
| Email        | varchar(255) | YES  |     | NULL    |       |
| Date_of_join | date         | YES  |     | NULL    |       |
| Status       | varchar(50)  | YES  |     | NULL    |       |
+--------------+--------------+------+-----+---------+-------+
8 rows in set (0.00 sec)
```

```
CREATE TABLE BOOK_ISSUE (
    Issue_id INT PRIMARY KEY,
    Date_of_issue DATE NOT NULL,
    Book_id INT,
    Member_id INT,
    Expected_date_of_return DATE,
    Status VARCHAR(50),
    FOREIGN KEY (Book_id) REFERENCES BOOK(Book_id) ON DELETE CASCADE ON
UPDATE       CASCADE,
    FOREIGN KEY (Member_id) REFERENCES MEMBER(Member_id) ON DELETE CASCADE
ON        UPDATE CASCADE
);
```



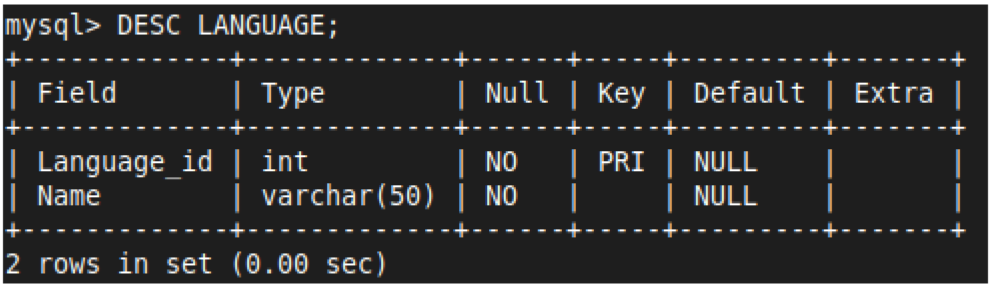
```
mysql> DESC BOOK_ISSUE;
+-------------------------+-------------+------+-----+---------+-------+
| Field                   | Type        | Null | Key | Default | Extra |
+-------------------------+-------------+------+-----+---------+-------+
| Issue_id                | int         | NO   | PRI | NULL    |       |
| Date_of_issue           | date        | NO   |     | NULL    |       |
| Book_id                 | int         | YES  | MUL | NULL    |       |
| Member_id               | int         | YES  | MUL | NULL    |       |
| Expected_date_of_return | date        | YES  |     | NULL    |       |
| Status                  | varchar(50) | YES  |     | NULL    |       |
+-------------------------+-------------+------+-----+---------+-------+
6 rows in set (0.00 sec)
```

```
CREATE TABLE BOOK_RETURN (
    Issue_id INT PRIMARY KEY,
    Actual_date_of_return DATE,
    Late_days INT,
    Late_fee DECIMAL(10, 2),
    FOREIGN KEY (Issue_id) REFERENCES BOOK_ISSUE(Issue_id) ON DELETE
CASCADE ON      UPDATE CASCADE
);
```



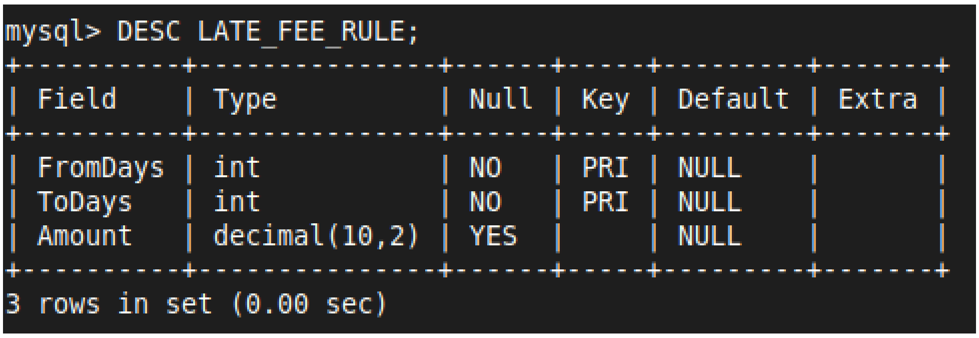
```
mysql> DESC BOOK_RETURN;
+-----------------------+---------------+------+-----+---------+-------+
| Field                 | Type          | Null | Key | Default | Extra |
+-----------------------+---------------+------+-----+---------+-------+
| Issue_id              | int           | NO   | PRI | NULL    |       |
| Actual_date_of_return | date          | YES  |     | NULL    |       |
| Late_days             | int           | YES  |     | NULL    |       |
| Late_fee              | decimal(10,2) | YES  |     | NULL    |       |
+-----------------------+---------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

```
CREATE TABLE LANGUAGE (
    Language_id INT PRIMARY KEY,
    Name VARCHAR(50) NOT NULL
);
```



```
mysql> DESC LANGUAGE;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| Language_id | int         | NO   | PRI | NULL    |       |
| Name        | varchar(50) | NO   |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
2 rows in set (0.00 sec)
```

```
CREATE TABLE LATE_FEE_RULE (
    FromDays INT,
    ToDays INT,
    Amount DECIMAL(10, 2),
    PRIMARY KEY (FromDays, ToDays)
);
```



```
mysql> DESC LATE_FEE_RULE;
+----------+---------------+------+-----+---------+-------+
| Field    | Type          | Null | Key | Default | Extra |
+----------+---------------+------+-----+---------+-------+
| FromDays | int           | NO   | PRI | NULL    |       |
| ToDays   | int           | NO   | PRI | NULL    |       |
| Amount   | decimal(10,2) | YES  |     | NULL    |       |
+----------+---------------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```

3. Adding Foreign Key After Creating the Table

```
ALTER TABLE BOOK
ADD CONSTRAINT fk_language FOREIGN KEY (Language_id) REFERENCES
LANGUAGE(Language_id);
```

```
ALTER TABLE BOOK
ADD CONSTRAINT fk_publisher FOREIGN KEY (Publisher_id) REFERENCES
PUBLISHER(Publisher_id);
```

4. Changing Data Type After Creating the Table

```
-- Change the data type of the Status column in the BOOK table
ALTER TABLE BOOK
MODIFY Status TINYINT(1);
```

5. To Add ON DELETE CASCADE & ON UPDATE CASCADE After Creating the Table

- **ON DELETE CASCADE**: When a row in the parent table (e.g., `BOOK`) is deleted, all corresponding rows in the child table (`BOOK_ISSUE`) will also be deleted automatically
- **ON UPDATE CASCADE**: When the primary key in the parent table is updated, the foreign key in the child table will automatically be updated to match

1. **Identify the Existing Foreign Key Constraints**
    - Before dropping a foreign key, you need to know its name. You can retrieve the foreign key constraints using

```
SHOW CREATE TABLE table_name;
```

2. **Drop Existing Foreign Key Constraints**
    - If you need to remove existing foreign key constraints, use the names identified from the previous step

```
ALTER TABLE BOOK
DROP FOREIGN KEY fk_language;
```

```
ALTER TABLE BOOK
DROP FOREIGN KEY fk_publisher;
```

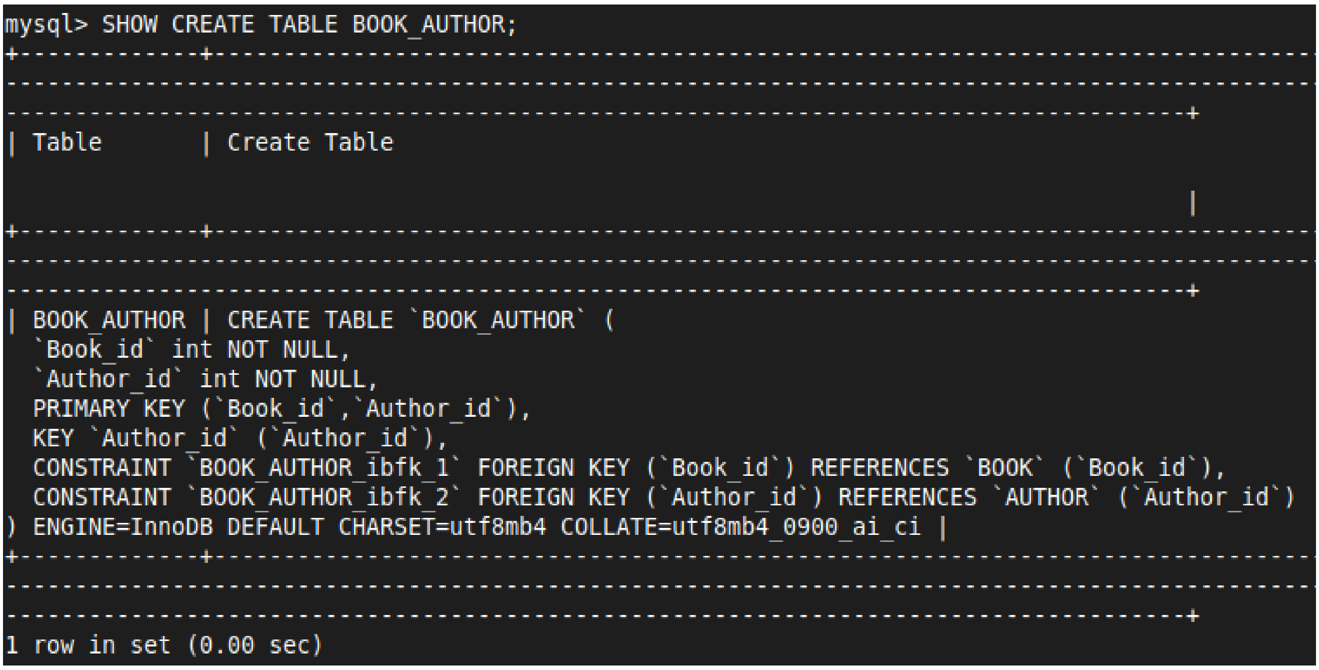
3. **Add New Foreign Key Constraints with Cascading Options**
    - After removing the existing constraints, you can add new foreign key constraints with `ON DELETE CASCADE` and `ON UPDATE CASCADE`

```
-- Add foreign key constraint for Language_id with ON DELETE CASCADE and
ON UPDATE CASCADE
ALTER TABLE BOOK
ADD CONSTRAINT fk_language
FOREIGN KEY (Language_id)
REFERENCES LANGUAGE(Language_id)
ON DELETE CASCADE
ON UPDATE CASCADE;
```

```
-- Add foreign key constraint for Publisher_id with ON DELETE CASCADE and
ON UPDATE CASCADE
ALTER TABLE BOOK
ADD CONSTRAINT fk_publisher
FOREIGN KEY (Publisher_id)
REFERENCES PUBLISHER(Publisher_id)
```

```
ON DELETE CASCADE
ON UPDATE CASCADE;
```

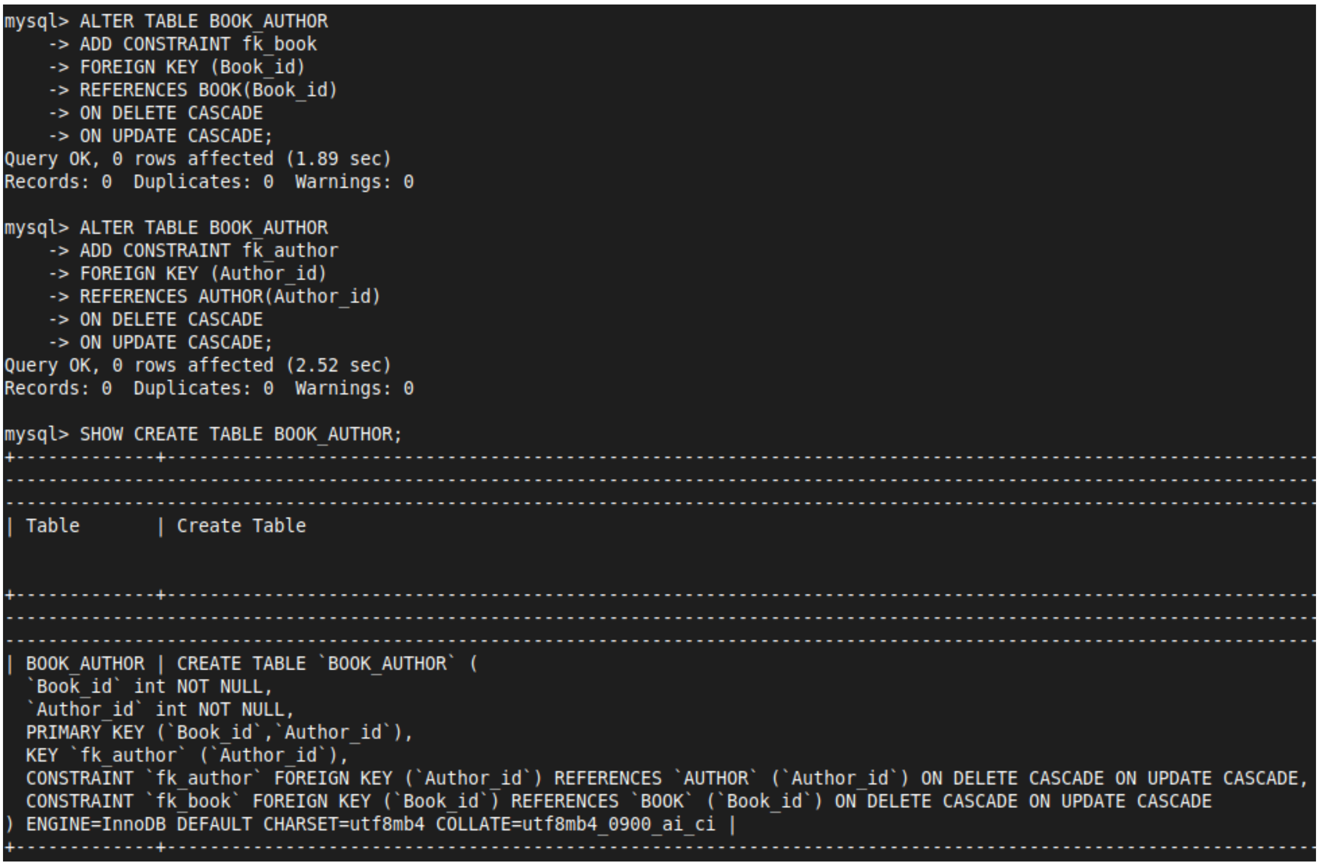
4. **If foreign key was declared while creating the table (not case2)**



```
mysql> SHOW CREATE TABLE BOOK_AUTHOR;
+-------------+--------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------+
| Table       | Create Table

                                                                                     |
+-------------+--------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------+
| BOOK_AUTHOR | CREATE TABLE `BOOK_AUTHOR` (
  `Book_id` int NOT NULL,
  `Author_id` int NOT NULL,
  PRIMARY KEY (`Book_id`,`Author_id`),
  KEY `Author_id` (`Author_id`),
  CONSTRAINT `BOOK_AUTHOR_ibfk_1` FOREIGN KEY (`Book_id`) REFERENCES `BOOK` (`Book_id`),
  CONSTRAINT `BOOK_AUTHOR_ibfk_2` FOREIGN KEY (`Author_id`) REFERENCES `AUTHOR` (`Author_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci |
+-------------+--------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------+
1 row in set (0.00 sec)
```

```
ALTER TABLE BOOK_AUTHOR
DROP FOREIGN KEY BOOK_AUTHOR_ibfk_1;
```

```
ALTER TABLE BOOK_AUTHOR
DROP FOREIGN KEY BOOK_AUTHOR_ibfk_2;
```

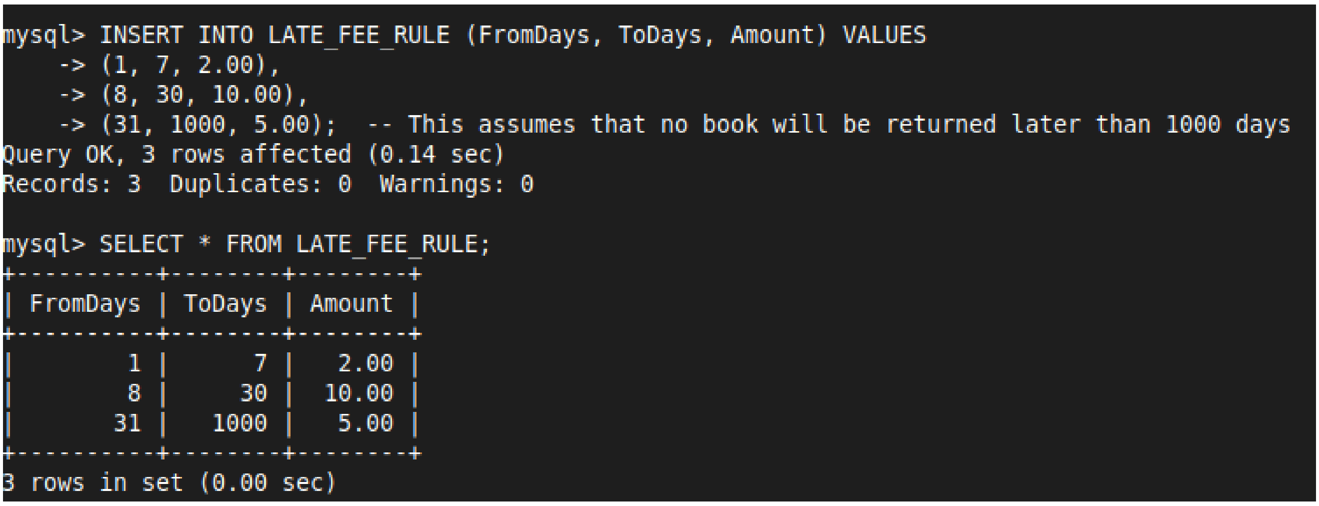


```
mysql> ALTER TABLE BOOK_AUTHOR
    -> ADD CONSTRAINT fk_book
    -> FOREIGN KEY (Book_id)
    -> REFERENCES BOOK(Book_id)
    -> ON DELETE CASCADE
    -> ON UPDATE CASCADE;
Query OK, 0 rows affected (1.89 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> ALTER TABLE BOOK_AUTHOR
    -> ADD CONSTRAINT fk_author
    -> FOREIGN KEY (Author_id)
    -> REFERENCES AUTHOR(Author_id)
    -> ON DELETE CASCADE
    -> ON UPDATE CASCADE;
Query OK, 0 rows affected (2.52 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> SHOW CREATE TABLE BOOK_AUTHOR;
+-------------+------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------
| Table       | Create Table


+-------------+------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------
| BOOK_AUTHOR | CREATE TABLE `BOOK_AUTHOR` (
  `Book_id` int NOT NULL,
  `Author_id` int NOT NULL,
  PRIMARY KEY (`Book_id`,`Author_id`),
  KEY `fk_author` (`Author_id`),
  CONSTRAINT `fk_author` FOREIGN KEY (`Author_id`) REFERENCES `AUTHOR` (`Author_id`) ON DELETE CASCADE ON UPDATE CASCADE,
  CONSTRAINT `fk_book` FOREIGN KEY (`Book_id`) REFERENCES `BOOK` (`Book_id`) ON DELETE CASCADE ON UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci |
+-------------+------------------------------------------------------------------------------------------------------------------------
```

6. Inserting Values

- `INSERT` is not a DDL it's DML

```
INSERT INTO table_name (columns) VALUES (value)
```

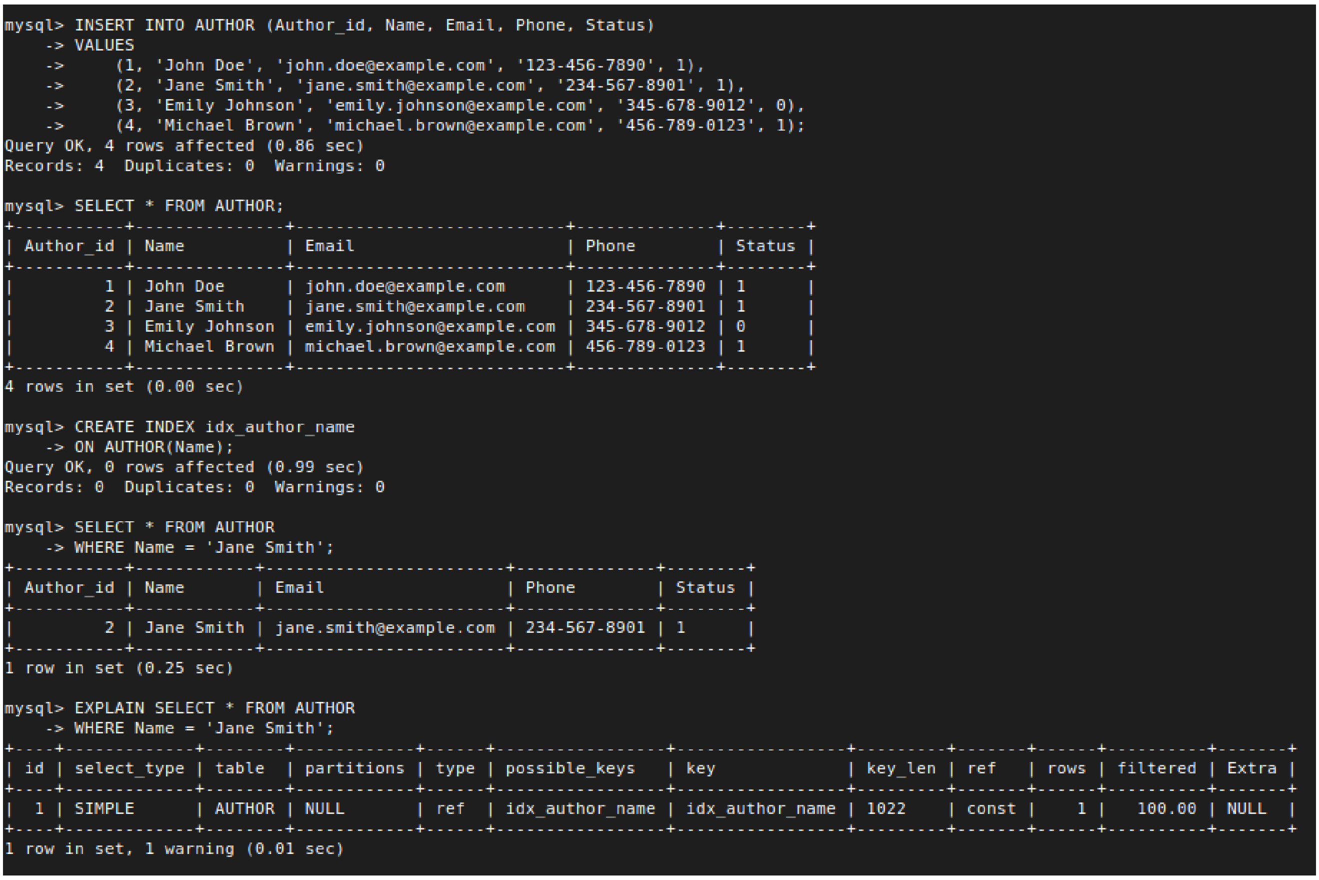


```
mysql> INSERT INTO LATE_FEE_RULE (FromDays, ToDays, Amount) VALUES
    -> (1, 7, 2.00),
    -> (8, 30, 10.00),
    -> (31, 1000, 5.00);  -- This assumes that no book will be returned later than 1000 days
Query OK, 3 rows affected (0.14 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM LATE_FEE_RULE;
+----------+--------+--------+
| FromDays | ToDays | Amount |
+----------+--------+--------+
|        1 |      7 |   2.00 |
|        8 |     30 |  10.00 |
|       31 |   1000 |   5.00 |
+----------+--------+--------+
3 rows in set (0.00 sec)
```

7. Indexing

- To confirm that the index is being used, you can use the `EXPLAIN` command before your query

- This command provides information about how MySQL executes your query, including whether it uses indexes

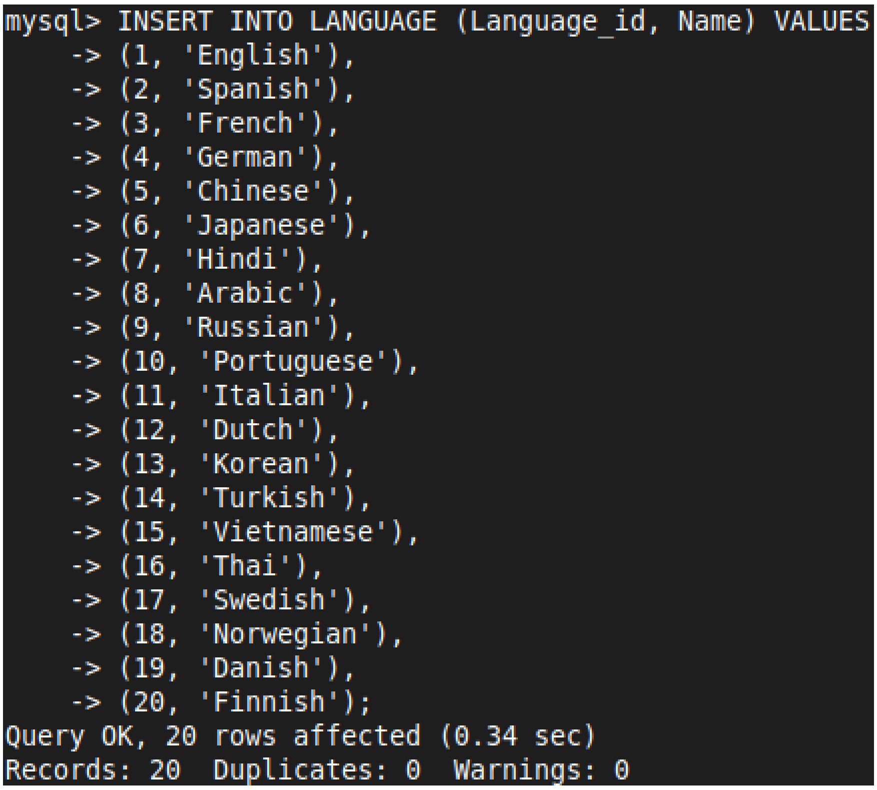


```
mysql> INSERT INTO AUTHOR (Author_id, Name, Email, Phone, Status)
    -> VALUES
    ->     (1, 'John Doe', 'john.doe@example.com', '123-456-7890', 1),
    ->     (2, 'Jane Smith', 'jane.smith@example.com', '234-567-8901', 1),
    ->     (3, 'Emily Johnson', 'emily.johnson@example.com', '345-678-9012', 0),
    ->     (4, 'Michael Brown', 'michael.brown@example.com', '456-789-0123', 1);
Query OK, 4 rows affected (0.86 sec)
Records: 4  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM AUTHOR;
+-----------+---------------+---------------------------+--------------+--------+
| Author_id | Name          | Email                     | Phone        | Status |
+-----------+---------------+---------------------------+--------------+--------+
|         1 | John Doe      | john.doe@example.com      | 123-456-7890 | 1      |
|         2 | Jane Smith    | jane.smith@example.com    | 234-567-8901 | 1      |
|         3 | Emily Johnson | emily.johnson@example.com | 345-678-9012 | 0      |
|         4 | Michael Brown | michael.brown@example.com | 456-789-0123 | 1      |
+-----------+---------------+---------------------------+--------------+--------+
4 rows in set (0.00 sec)

mysql> CREATE INDEX idx_author_name
    -> ON AUTHOR(Name);
Query OK, 0 rows affected (0.99 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM AUTHOR
    -> WHERE Name = 'Jane Smith';
+-----------+------------+------------------------+--------------+--------+
| Author_id | Name       | Email                  | Phone        | Status |
+-----------+------------+------------------------+--------------+--------+
|         2 | Jane Smith | jane.smith@example.com | 234-567-8901 | 1      |
+-----------+------------+------------------------+--------------+--------+
1 row in set (0.25 sec)

mysql> EXPLAIN SELECT * FROM AUTHOR
    -> WHERE Name = 'Jane Smith';
+----+-------------+--------+------------+------+-----------------+-----------------+---------+-------+------+----------+-------+
| id | select_type | table  | partitions | type | possible_keys   | key             | key_len | ref   | rows | filtered | Extra |
+----+-------------+--------+------------+------+-----------------+-----------------+---------+-------+------+----------+-------+
|  1 | SIMPLE      | AUTHOR | NULL       | ref  | idx_author_name | idx_author_name | 1022    | const |    1 |   100.00 | NULL  |
+----+-------------+--------+------------+------+-----------------+-----------------+---------+-------+------+----------+-------+
1 row in set, 1 warning (0.01 sec)
```

SQL - DML & DQL

1. INSERT

- Insert Values to the LANGUAGE Table

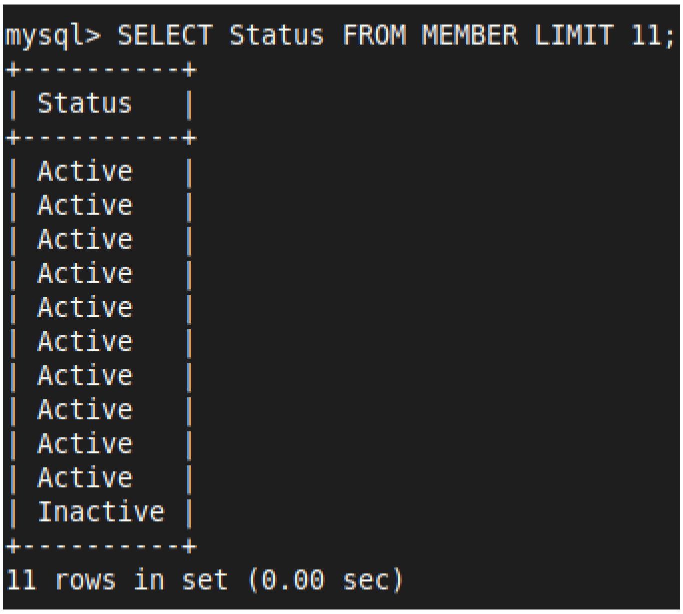


```
mysql> INSERT INTO LANGUAGE (Language_id, Name) VALUES
    -> (1, 'English'),
    -> (2, 'Spanish'),
    -> (3, 'French'),
    -> (4, 'German'),
    -> (5, 'Chinese'),
    -> (6, 'Japanese'),
    -> (7, 'Hindi'),
    -> (8, 'Arabic'),
    -> (9, 'Russian'),
    -> (10, 'Portuguese'),
    -> (11, 'Italian'),
    -> (12, 'Dutch'),
    -> (13, 'Korean'),
    -> (14, 'Turkish'),
    -> (15, 'Vietnamese'),
    -> (16, 'Thai'),
    -> (17, 'Swedish'),
    -> (18, 'Norwegian'),
    -> (19, 'Danish'),
    -> (20, 'Finnish');
Query OK, 20 rows affected (0.34 sec)
Records: 20  Duplicates: 0  Warnings: 0
```

2. UPDATE

- **Modify status from 'Active' to 'Inactive' on MEMBER where id=11**

```
UPDATE MEMBER SET Status='Inactive' WHERE Member_id=11;
```

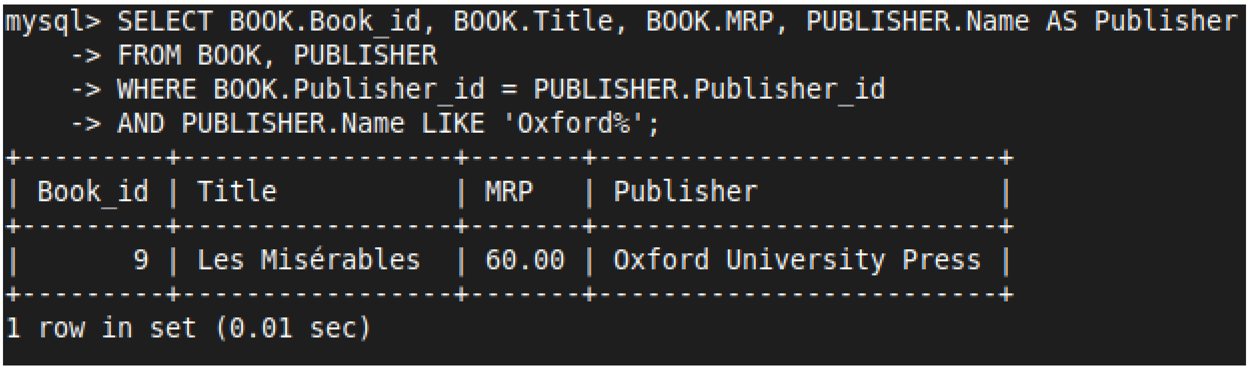


```
mysql> SELECT Status FROM MEMBER LIMIT 11;
+----------+
| Status   |
+----------+
| Active   |
| Active   |
| Active   |
| Active   |
| Active   |
| Active   |
| Active   |
| Active   |
| Active   |
| Active   |
| Inactive |
+----------+
11 rows in set (0.00 sec)
```

- **Modify the MRP of all books published by 'Oxford University Press', provide 10% discount on MRP**

```
UPDATE BOOK, PUBLISHER SET BOOK.MRP = 0.1*BOOK.MRP
WHERE BOOK.Publisher_id = PUBLISHER.Publisher_id AND PUBLISHER.Name LIKE
```

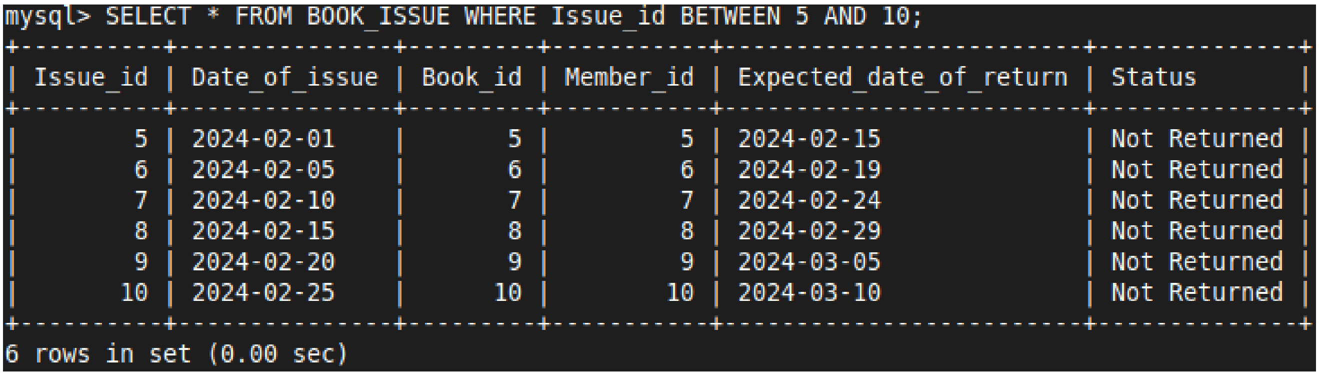
```
'Oxford%';
```



```
mysql> SELECT BOOK.Book_id, BOOK.Title, BOOK.MRP, PUBLISHER.Name AS Publisher
    -> FROM BOOK, PUBLISHER
    -> WHERE BOOK.Publisher_id = PUBLISHER.Publisher_id
    -> AND PUBLISHER.Name LIKE 'Oxford%';
+---------+-----------------+-------+--------------------------+
| Book_id | Title           | MRP   | Publisher                |
+---------+-----------------+-------+--------------------------+
|       9 | Les Misérables  | 60.00 | Oxford University Press |
+---------+-----------------+-------+--------------------------+
1 row in set (0.01 sec)
```

- **Modify the Status from 'Issued' to 'Not Returned' on id 5 to 10 in BOOK_ISSUE**

```
UPDATE BOOK_ISSUE SET Status='Not Returned' WHERE Issue_id BETWEEN 5 AND
10;
```



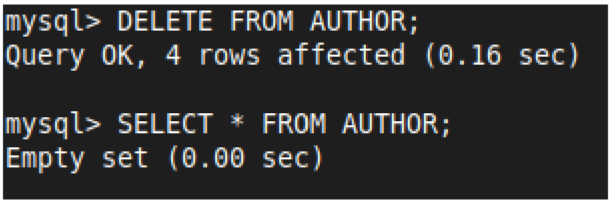
```
mysql> SELECT * FROM BOOK_ISSUE WHERE Issue_id BETWEEN 5 AND 10;
+----------+---------------+---------+-----------+------------------------+--------------+
| Issue_id | Date_of_issue | Book_id | Member_id | Expected_date_of_return | Status       |
+----------+---------------+---------+-----------+------------------------+--------------+
|        5 | 2024-02-01    |       5 |         5 | 2024-02-15             | Not Returned |
|        6 | 2024-02-05    |       6 |         6 | 2024-02-19             | Not Returned |
|        7 | 2024-02-10    |       7 |         7 | 2024-02-24             | Not Returned |
|        8 | 2024-02-15    |       8 |         8 | 2024-02-29             | Not Returned |
|        9 | 2024-02-20    |       9 |         9 | 2024-03-05             | Not Returned |
|       10 | 2024-02-25    |      10 |        10 | 2024-03-10             | Not Returned |
+----------+---------------+---------+-----------+------------------------+--------------+
6 rows in set (0.00 sec)
```

3. Delete

- Delete existing records from a table

```
DELETE FROM table_name;
```

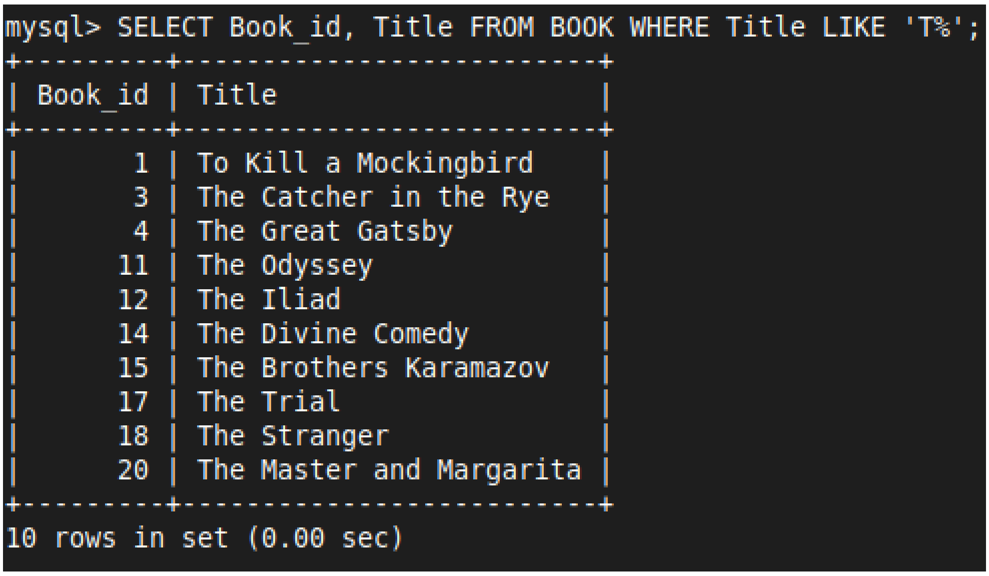
```
DELETE FROM table_name WHERE condition;
```



```
mysql> DELETE FROM AUTHOR;
Query OK, 4 rows affected (0.16 sec)

mysql> SELECT * FROM AUTHOR;
Empty set (0.00 sec)
```

4. SELECT

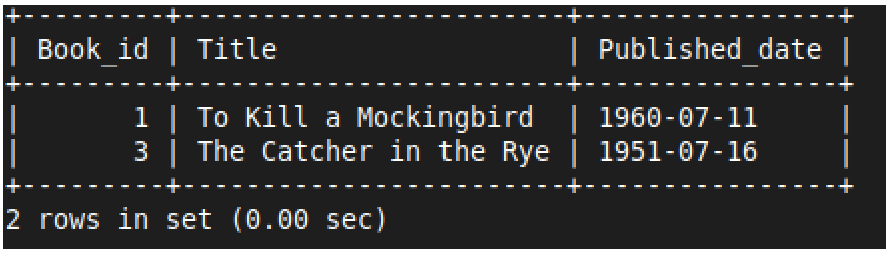
- Find all books where title starts with 'T'



```
mysql> SELECT Book_id, Title FROM BOOK WHERE Title LIKE 'T%';
+---------+--------------------------+
| Book_id | Title                    |
+---------+--------------------------+
|       1 | To Kill a Mockingbird    |
|       3 | The Catcher in the Rye   |
|       4 | The Great Gatsby         |
|      11 | The Odyssey              |
|      12 | The Iliad                |
|      14 | The Divine Comedy        |
|      15 | The Brothers Karamazov   |
|      17 | The Trial                |
|      18 | The Stranger             |
|      20 | The Master and Margarita |
+---------+--------------------------+
10 rows in set (0.00 sec)
```

- Find all books published between Jan 1st 1950 and Dec 31st 1960

```
SELECT Book_id, Title, Published_date
FROM BOOK
WHERE Published_date BETWEEN '1950-01-01' AND '1960-12-31'
```

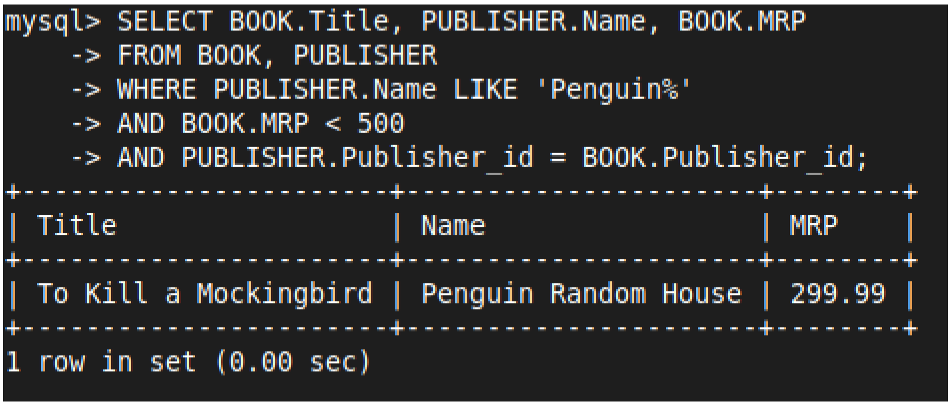


```
+---------+------------------------+----------------+
| Book_id | Title                  | Published_date |
+---------+------------------------+----------------+
|       1 | To Kill a Mockingbird  | 1960-07-11     |
|       3 | The Catcher in the Rye | 1951-07-16     |
+---------+------------------------+----------------+
2 rows in set (0.00 sec)
```

- Find all books published by 'Penguin Random House' having MRP less than 500

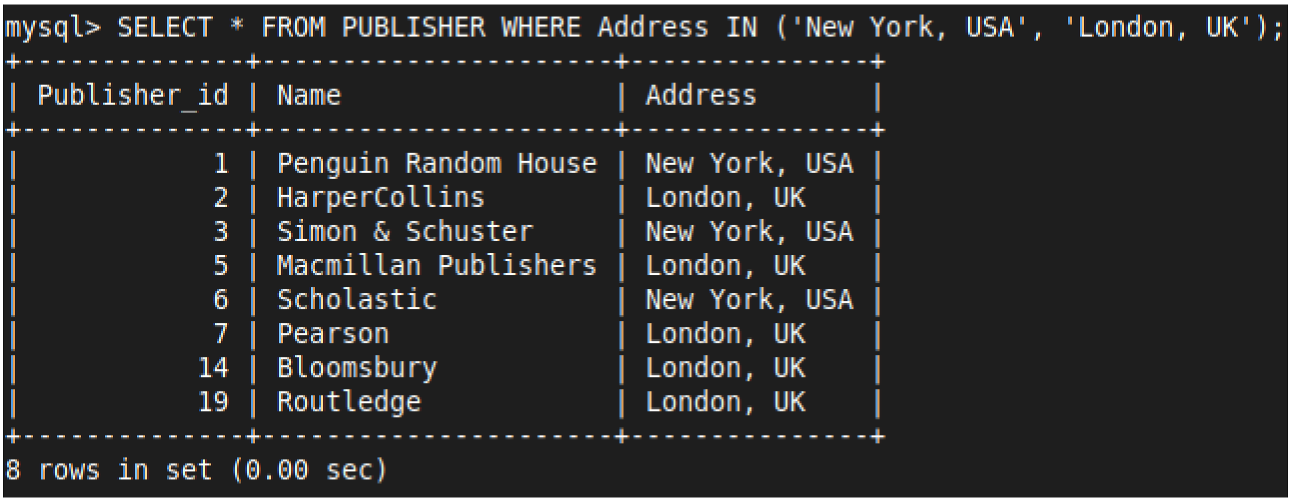
```
SELECT BOOK.Title, PUBLISHER.Name, BOOK.MRP
FROM BOOK JOIN PUBLISHER
ON PUBLISHER.Publisher_id = BOOK.Publisher_id
WHERE PUBLISHER.Name LIKE 'Penguin%' AND BOOK.MRP < 500;
```

```
SELECT BOOK.Title, PUBLISHER.Name, BOOK.MRP
FROM BOOK, PUBLISHER
WHERE PUBLISHER.Name LIKE 'Penguin%'
AND BOOK.MRP < 500
AND PUBLISHER.Publisher_id = BOOK.Publisher_id;
```



```
mysql> SELECT BOOK.Title, PUBLISHER.Name, BOOK.MRP
    -> FROM BOOK, PUBLISHER
    -> WHERE PUBLISHER.Name LIKE 'Penguin%'
    -> AND BOOK.MRP < 500
    -> AND PUBLISHER.Publisher_id = BOOK.Publisher_id;
+-----------------------+----------------------+--------+
| Title                 | Name                 | MRP    |
+-----------------------+----------------------+--------+
| To Kill a Mockingbird | Penguin Random House | 299.99 |
+-----------------------+----------------------+--------+
1 row in set (0.00 sec)
```

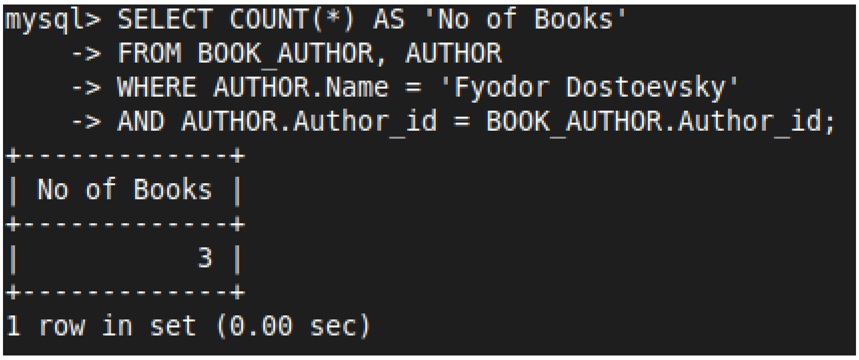
- Find publisher who are from 'New York, USA' and 'London, UK'



```
mysql> SELECT * FROM PUBLISHER WHERE Address IN ('New York, USA', 'London, UK');
+--------------+----------------------+---------------+
| Publisher_id | Name                 | Address       |
+--------------+----------------------+---------------+
|            1 | Penguin Random House | New York, USA |
|            2 | HarperCollins        | London, UK    |
|            3 | Simon & Schuster     | New York, USA |
|            5 | Macmillan Publishers | London, UK    |
|            6 | Scholastic           | New York, USA |
|            7 | Pearson              | London, UK    |
|           14 | Bloomsbury           | London, UK    |
|           19 | Routledge            | London, UK    |
+--------------+----------------------+---------------+
8 rows in set (0.00 sec)
```

- Get the no of books written by author named 'Fyodor Dostoevsky'

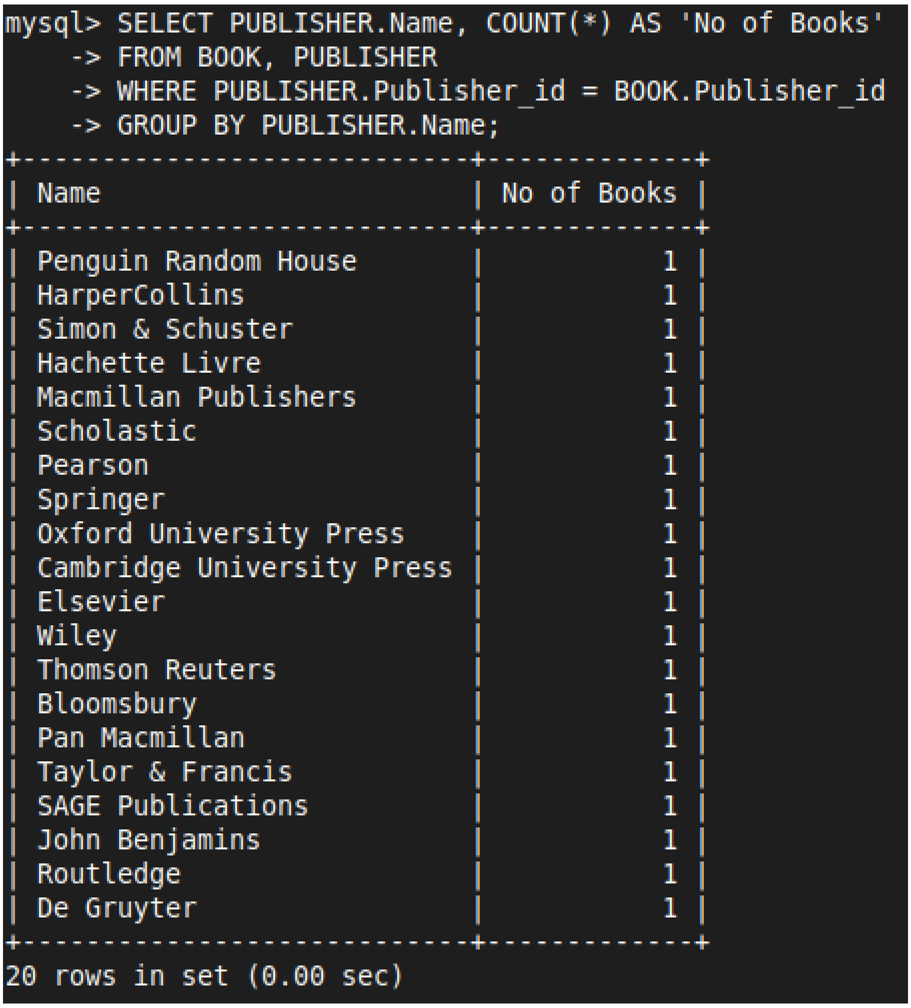
```sql
SELECT COUNT(*) AS 'No of Books'
FROM BOOK_AUTHOR, AUTHOR
WHERE AUTHOR.Name = 'Fyodor Dostoevsky'
AND AUTHOR.Author_id = BOOK_AUTHOR.Author_id;
```



```
mysql> SELECT COUNT(*) AS 'No of Books'
    -> FROM BOOK_AUTHOR, AUTHOR
    -> WHERE AUTHOR.Name = 'Fyodor Dostoevsky'
    -> AND AUTHOR.Author_id = BOOK_AUTHOR.Author_id;
+-------------+
| No of Books |
+-------------+
|           3 |
+-------------+
1 row in set (0.00 sec)
```

- Get the list of publishers and the no of books published by each publisher

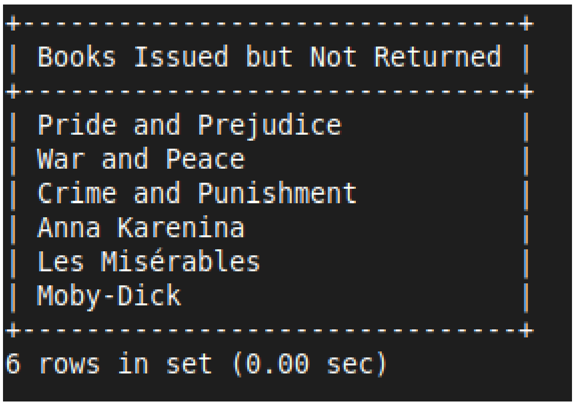
```sql
SELECT PUBLISHER.Name, COUNT(*) AS 'No of Books'
FROM BOOK, PUBLISHER
WHERE PUBLISHER.Publisher_id = BOOK.Publisher_id
GROUP BY PUBLISHER.Name;
```



```
mysql> SELECT PUBLISHER.Name, COUNT(*) AS 'No of Books'
    -> FROM BOOK, PUBLISHER
    -> WHERE PUBLISHER.Publisher_id = BOOK.Publisher_id
    -> GROUP BY PUBLISHER.Name;
+----------------------------+-------------+
| Name                       | No of Books |
+----------------------------+-------------+
| Penguin Random House       |           1 |
| HarperCollins              |           1 |
| Simon & Schuster           |           1 |
| Hachette Livre             |           1 |
| Macmillan Publishers       |           1 |
| Scholastic                 |           1 |
| Pearson                    |           1 |
| Springer                   |           1 |
| Oxford University Press    |           1 |
| Cambridge University Press |           1 |
| Elsevier                   |           1 |
| Wiley                      |           1 |
| Thomson Reuters            |           1 |
| Bloomsbury                 |           1 |
| Pan Macmillan              |           1 |
| Taylor & Francis           |           1 |
| SAGE Publications          |           1 |
| John Benjamins             |           1 |
| Routledge                  |           1 |
| De Gruyter                 |           1 |
+----------------------------+-------------+
20 rows in set (0.00 sec)
```

- Get the list of books that are not returned

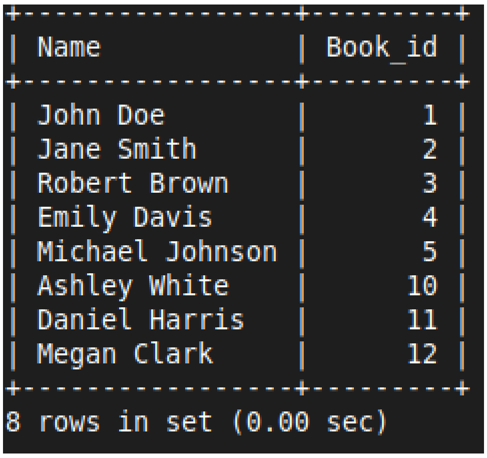
```
SELECT BOOK.Title AS 'Books Issued but Not Returned' FROM BOOK, BOOK_ISSUE
WHERE BOOK.Book_id = BOOK_ISSUE.Book_id AND BOOK_ISSUE.Status = 'Not
Returned';
```



```
+-------------------------------+
| Books Issued but Not Returned |
+-------------------------------+
| Pride and Prejudice           |
| War and Peace                 |
| Crime and Punishment          |
| Anna Karenina                 |
| Les Misérables                |
| Moby-Dick                     |
+-------------------------------+
6 rows in set (0.00 sec)
```

- Get the list of the students who reads only 'English' books

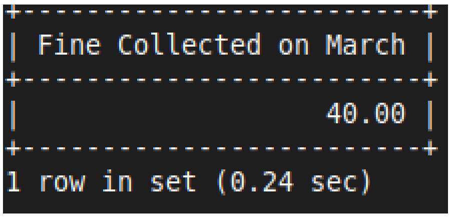
```
SELECT A.Name, B.Book_id
FROM MEMBER A, BOOK B, BOOK_ISSUE C, LANGUAGE D
WHERE D.Name = 'English'
AND D.Language_id = B.Language_id
AND B.Book_id = C.Book_id
AND A.Member_id = C.Member_id;
```



```
+------------------+---------+
| Name             | Book_id |
+------------------+---------+
| John Doe         |       1 |
| Jane Smith       |       2 |
| Robert Brown     |       3 |
| Emily Davis      |       4 |
| Michael Johnson  |       5 |
| Ashley White     |      10 |
| Daniel Harris    |      11 |
| Megan Clark      |      12 |
+------------------+---------+
8 rows in set (0.00 sec)
```

- Get the total fine collected on 'March'

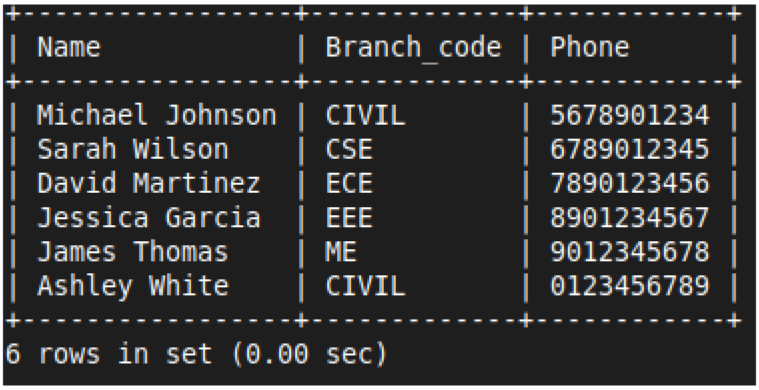
```
SELECT SUM(Late_fee) AS 'Fine Collected on March'
FROM BOOK_RETURN WHERE Actual_date_of_return > '2024-02-29'
AND Actual_date_of_return < '2024-04-01';
```



```
+-------------------------+
| Fine Collected on March |
+-------------------------+
|                   40.00 |
+-------------------------+
1 row in set (0.24 sec)
```

- Get the list of students who have overdue (not returned the book even after due date)

```
SELECT Name, Branch_code, Phone FROM MEMBER, BOOK_ISSUE
WHERE MEMBER.Member_id = BOOK_ISSUE.Member_id
AND BOOK_ISSUE.Status = 'Not Returned';
```



```
+------------------+-------------+------------+
| Name             | Branch_code | Phone      |
+------------------+-------------+------------+
| Michael Johnson  | CIVIL       | 5678901234 |
| Sarah Wilson     | CSE         | 6789012345 |
| David Martinez   | ECE         | 7890123456 |
| Jessica Garcia   | EEE         | 8901234567 |
| James Thomas     | ME          | 9012345678 |
| Ashley White     | CIVIL       | 0123456789 |
+------------------+-------------+------------+
6 rows in set (0.00 sec)
```