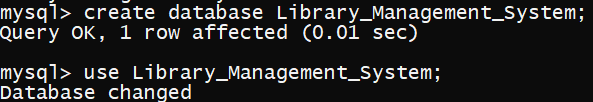
**CSC 785 - INFORMATION STORAGE AND RETRIEVAL**

**Project**

Lalitha Priya Bijja – 101168225  
Venkata Sai Bhargav Veeramsetty – 101149826

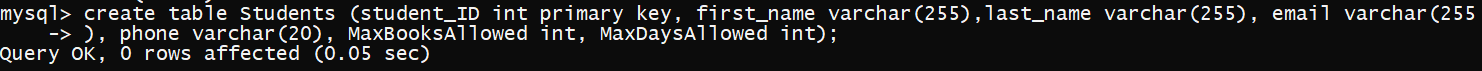
**Library Management System**

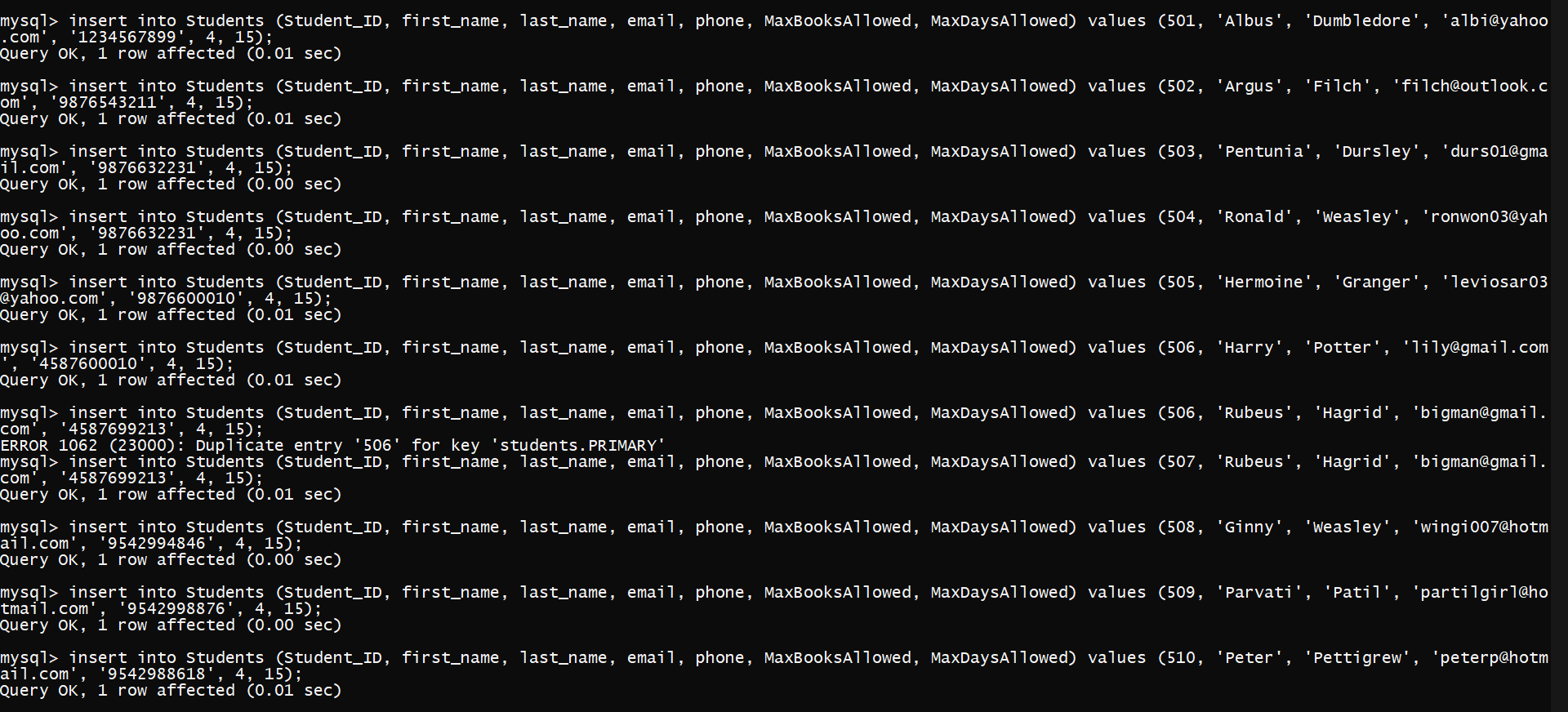
Creating a Database named Library\_Management\_System

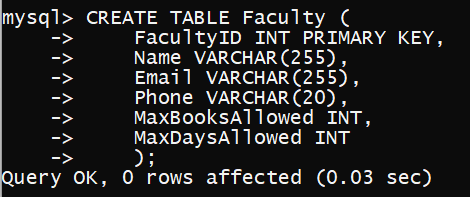


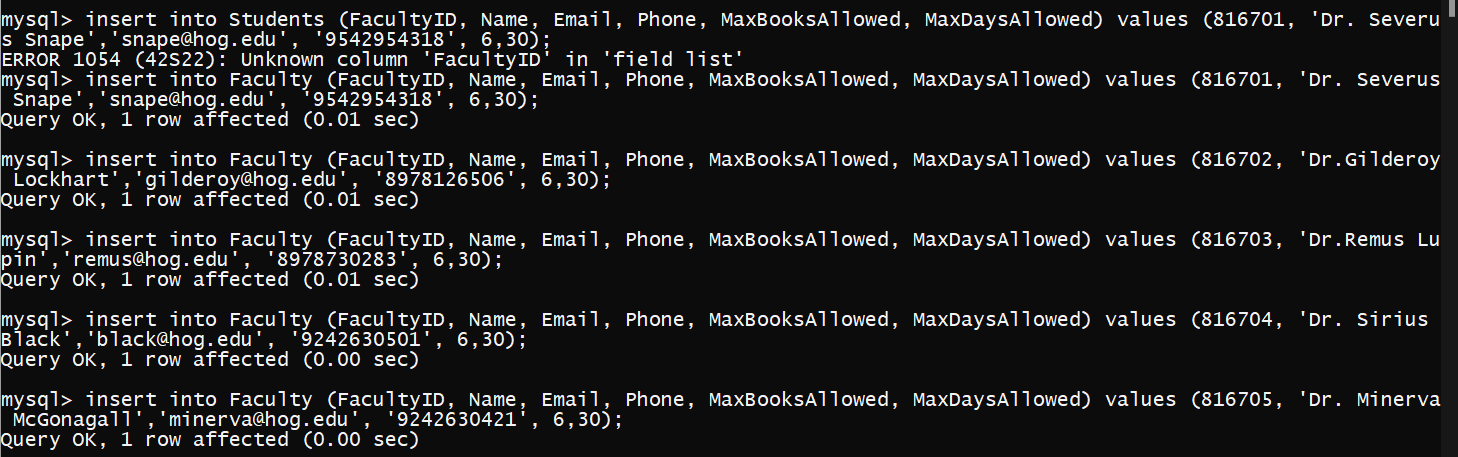
**Created tables of:**

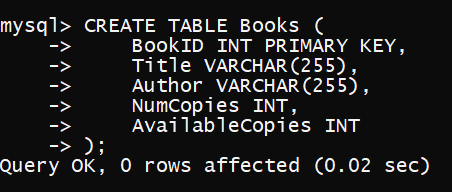
* Students
* Faculty
* Books
* BooksIssued

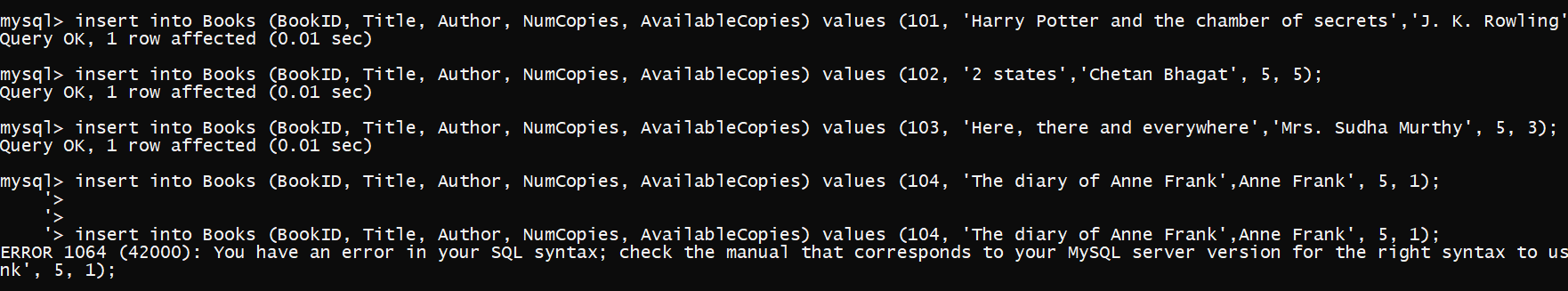


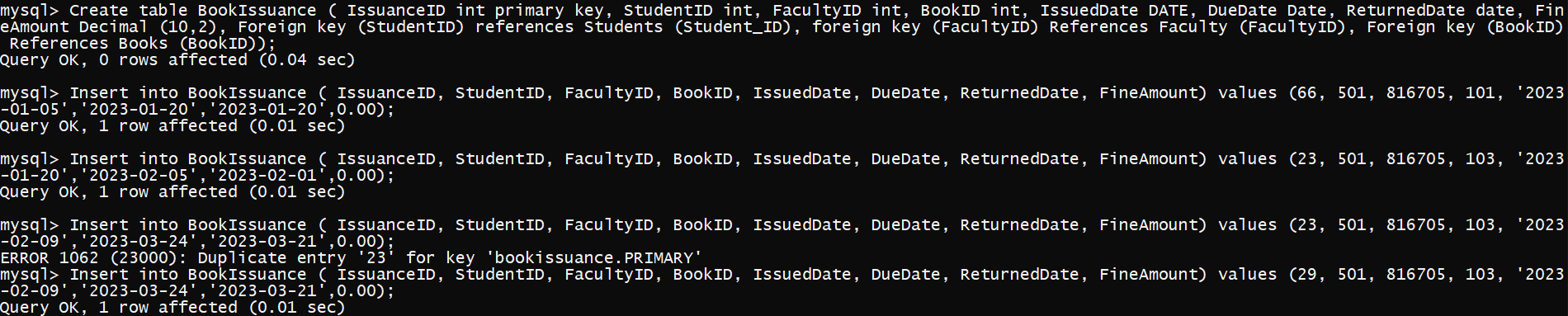


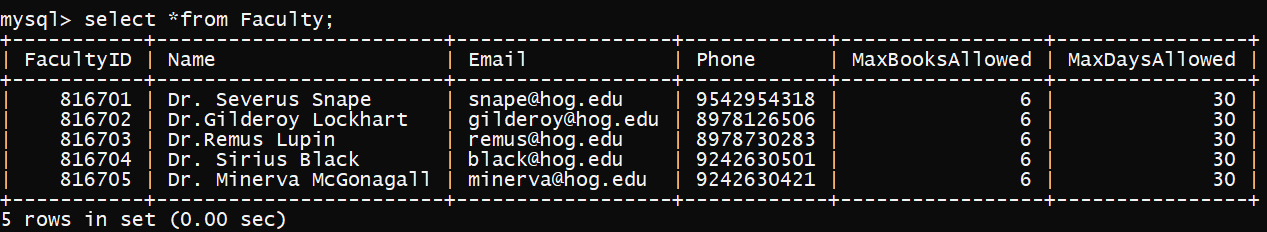
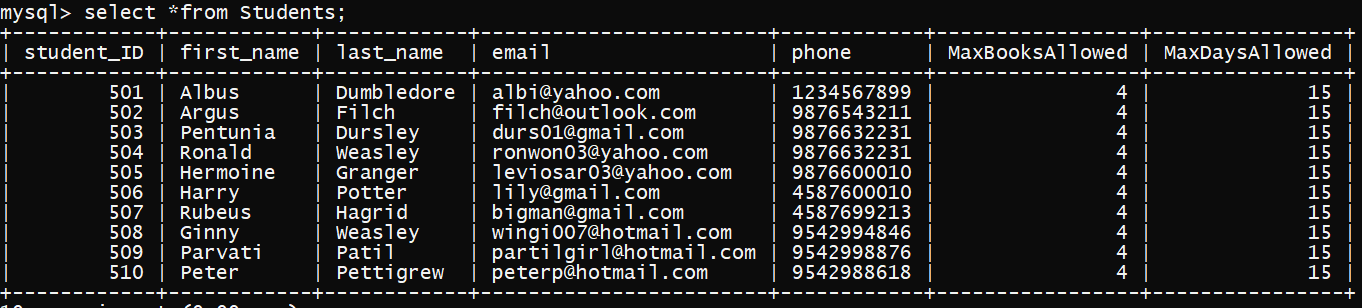


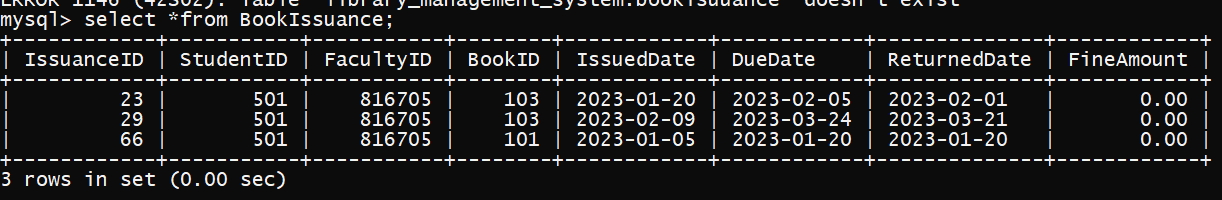


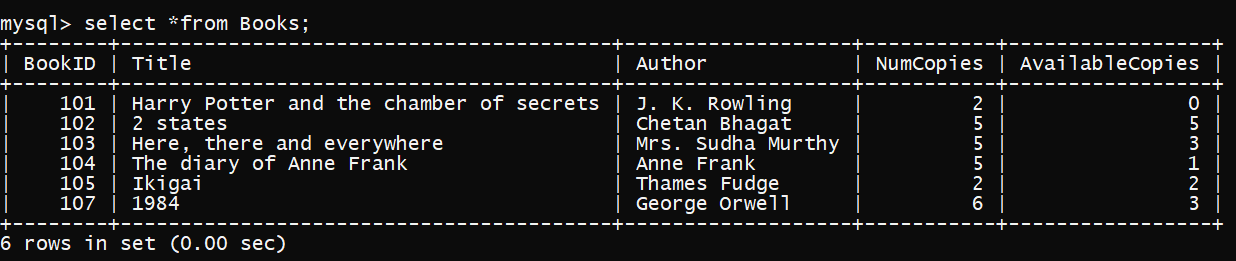












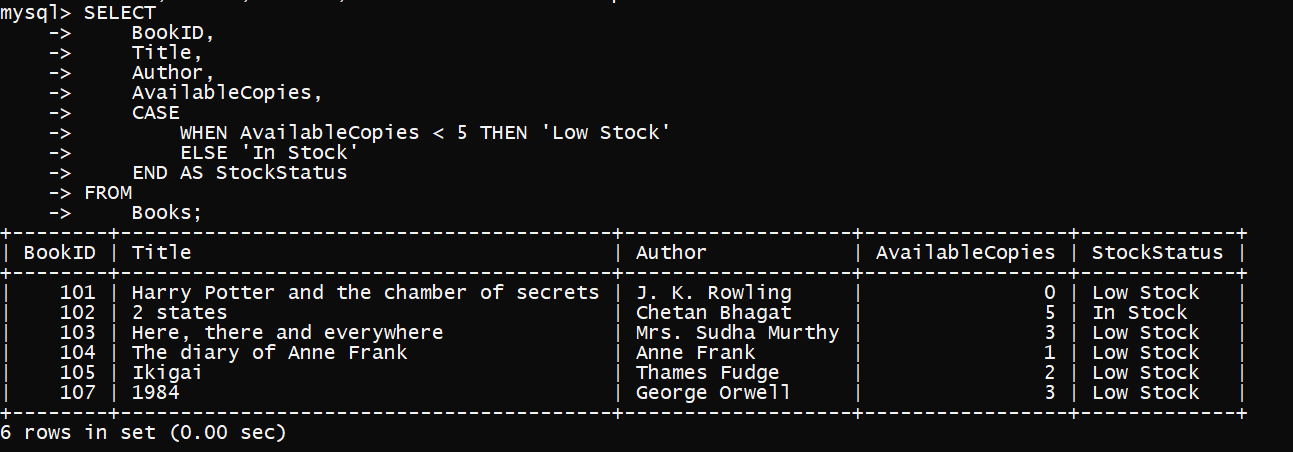
**Properties of the relations:**

* The Students table stores information about students, including their ID, first name, last name, contact details, and the maximum number of books and days allowed for borrowing.
* The Faculty table stores similar information for faculty members, including their ID, name, contact details, and borrowing limits.
* The Books table contains information about each book, including a unique Book ID, title, author, and the number of copies available. Each copy of a book has its unique ID within this table.
* The Book Issuance table tracks book loans. It includes an Issuance ID as a unique identifier, the IDs of the student or faculty member borrowing the book, the Book ID

they're borrowing, the date of issuance, the due date for return, the date the book was returned (if it has been returned), and any associated fine for late return.

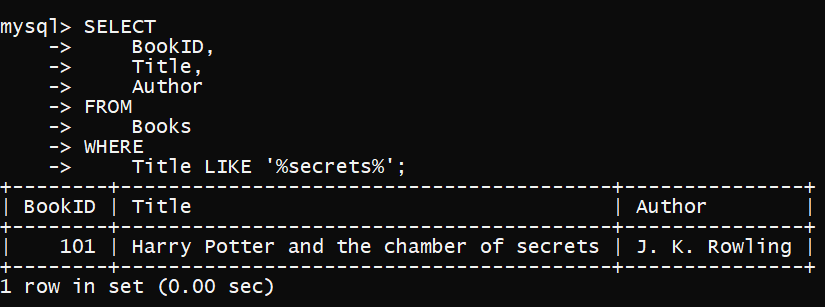
**Applying search conditions with calculated fields**We calculate the StockStatus for each book using the CASE statement based on the value of AvailableCopies. If AvailableCopies is less than 5, it's considered "Low Stock"; otherwise, it's "In Stock."

The query selects all books from the Books table and includes their BookID, Title, Author, AvailableCopies, and the calculated StockStatus.



**Used pattern search**We use the LIKE operator with % wildcard characters before and after the word "Database" ('%Database%'). This pattern search will match any book title that contains the word "Database" anywhere within it.

The query selects the BookID, Title, and Author columns from the Books table for books that match the pattern.

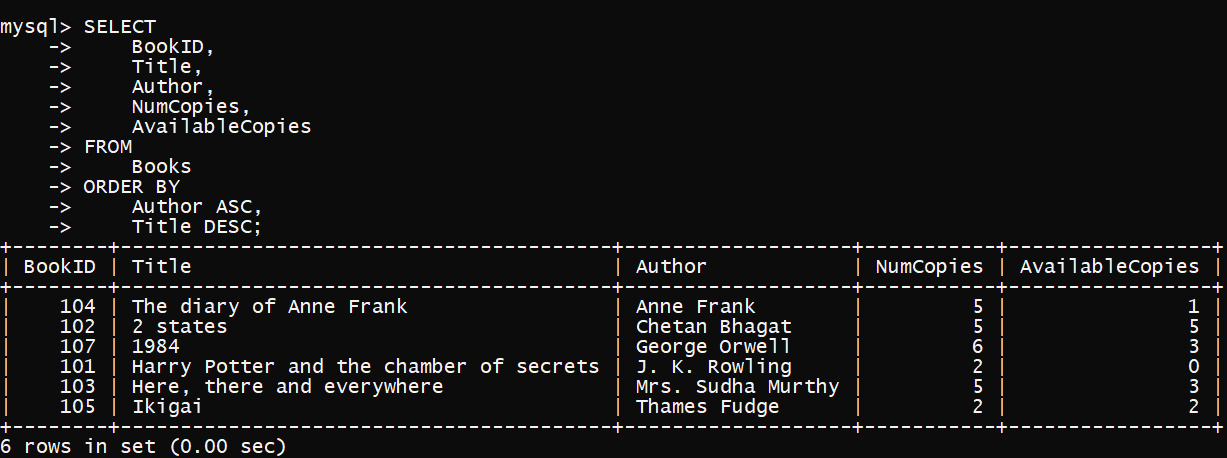


**select tuples based on ordering (but multiple columns);**

We specify the columns we want to retrieve (BookID, Title, Author, NumCopies, AvailableCopies) from the Books table.

We use the ORDER BY clause to define the ordering criteria. In this case, we're ordering the results first by the Author column in ascending order (ASC) and then by the Title column in descending order (DESC).

The result of this query will be a list of books ordered by Author in ascending order, and within each author group, the titles will be in descending order. You can adjust the ordering columns and directions as needed to match your specific requirements.

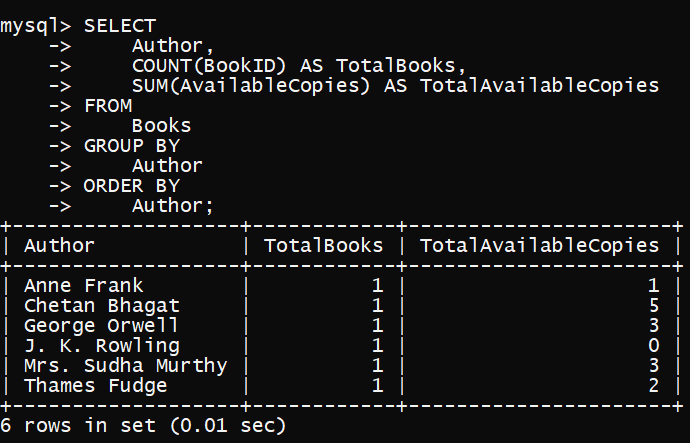


**Use Aggregated function**

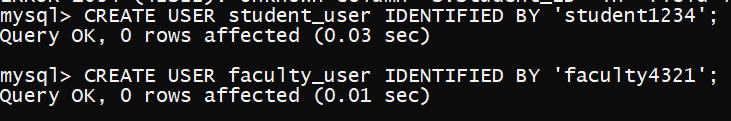
We specify the columns we want to retrieve (BookID, Title, Author, NumCopies, AvailableCopies) from the Books table.

We use the ORDER BY clause to define the ordering criteria. In this case, we're ordering the results first by the Author column in ascending order (ASC) and then by the Title column in descending order (DESC).

The result of this query will be a list of books ordered by Author in ascending order, and within each author group, the titles will be in descending order. You can adjust the ordering columns and directions as needed to match your specific requirements.



**Created users and granted privileges**



**Make multiple relations in a query**

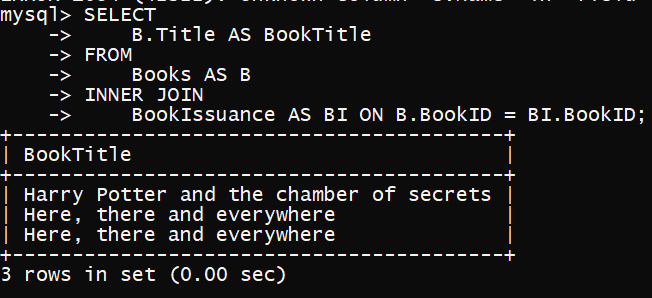
We select the StudentID from the Students table.

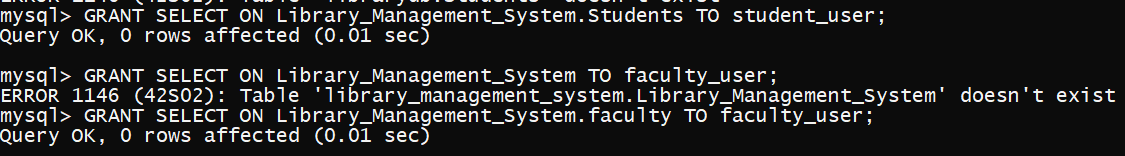
We use a LEFT JOIN operation to combine the Students and BookIssuance tables based on the StudentID.

We use the COUNT function to count the number of books issued by each student and alias it as TotalBooksIssued.

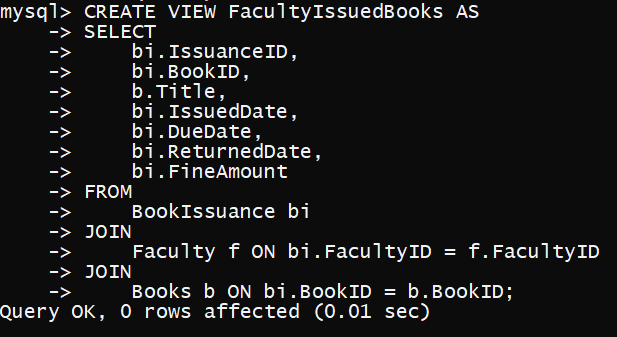
We group the results by StudentID to get the count of books issued per student.

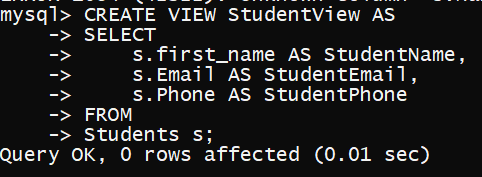
Finally, we order the results by StudentID.



****

**Created Views for both Faculty and Students**

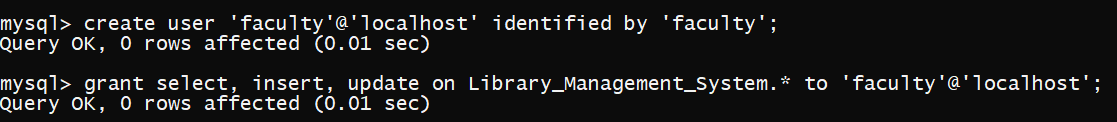




**Granted privileges to the views to both students and faculty**







**Created Backup**

mysqldump -u root -p Library\_Management\_System > mybachup.sql  
Enter password : \*\*\*\*\*\*\*  
  
Backup created