**Assignment 5:** Demonstrate the creation of an index on a table and discuss how it improves query performance. Use a DROP INDEX statement to remove the index and analyse the impact on query execution.

Step 1: What is an Index?

Think of an index like the index of a book.

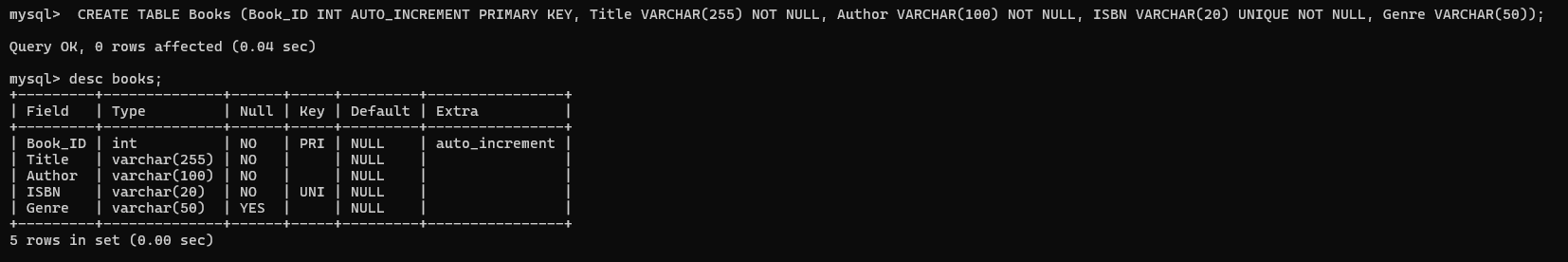
Without an index: You flip through every page to find something.

With an index: You quickly go to the right page.

In databases, an index makes searching for specific data in a table faster.

Step 2: Example Scenario:

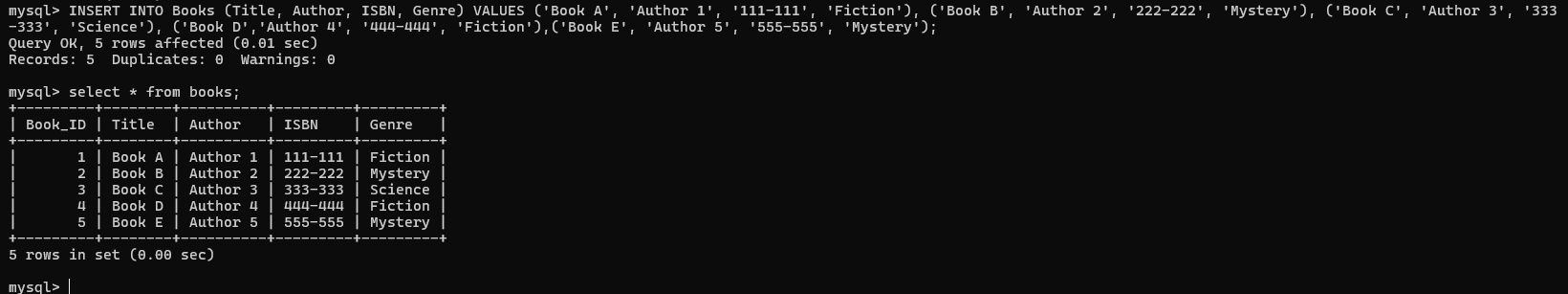
We will use the Books table from the Library\_system:



The table has a Genre column (e.g., Fiction, Mystery, Science).

Without an index, searching for books in a specific genre takes longer.

Step 3: Insert Sample Data:

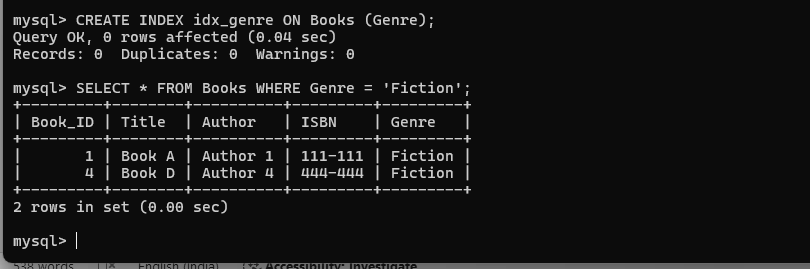


Step 4: Create an Index

To make searches faster, create an index on the Genre column:

Step 5: Test Query Performance

Before using the index: The database has to search all rows to find books in the "Fiction" genre:

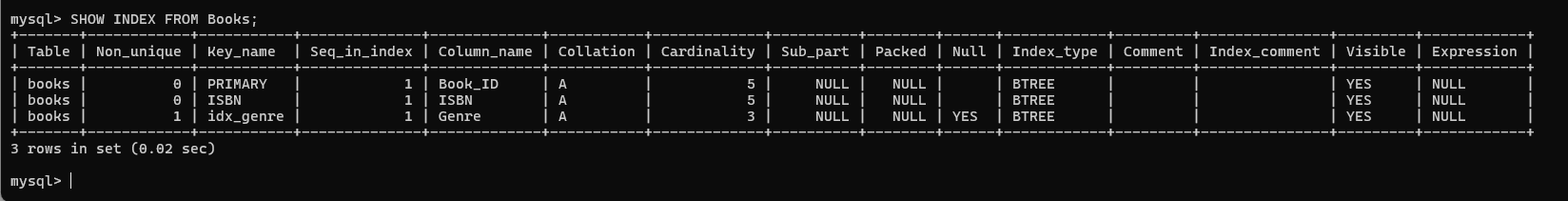


This is slow for large tables

After creating the index: The database uses the index to find rows faster.

Step 6: Check the Index

You can see the indexes on the Book table using:

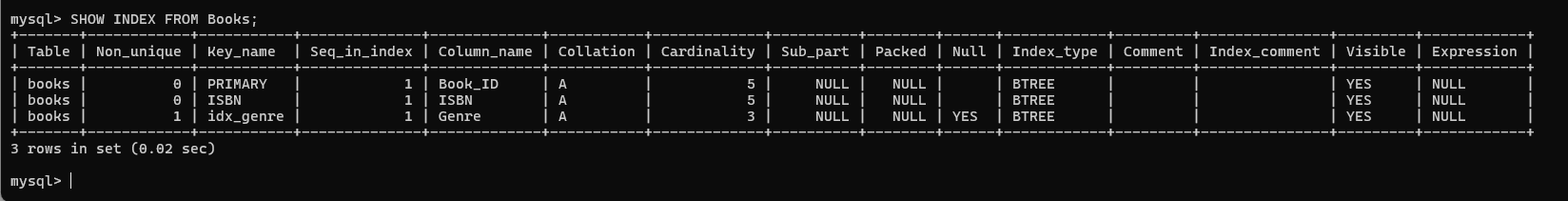


Step 7: Remove the Index

If you no longer need the index, you can remove it.

Step 8: Analyze the Impact

After removing the index, run the query again:



Notice that the query takes longer because the database has to scan all rows again.

**Summary of Benefits:**

**With Index:**

* Speeds up searches for specific values.
* Reduces query execution time for large datasets.

**Without Index:**

* Queries take longer as the table grows larger.

**Final Notes:**

Use indexes wisely. Too many indexes can slow down INSERT or UPDATE operations because the database has to update the indexes too.

Use EXPLAIN to analyze how indexes affect your query performance:

