

Cesar U. Guiritan
BSCPE 2B2

Laboratory Activity 4:

Result:

JOIN operations linking tables to retrieve combined data.

The screenshot shows the SQL Server Enterprise Manager interface. The Query window displays the following SQL query:

```
1 SELECT Books.Title, Members.FirstName, Members.LastName
2
3 FROM Transactions
4
5 INNER JOIN Books ON Transactions.BookID = Books.BookID
6
7 INNER JOIN Members ON Transactions.MemberID = Members.MemberID
```

The Results window shows the output of the query, displaying a table with columns: Title, FirstName, and LastName. The data is as follows:

Title	FirstName	LastName
The Great Gatsby	John	Doe
To Kill a Mockingbird	Jane	Smith
1984	Michael	Johnson
Pride and Prejudice	Emily	Davis
Moby Dick	David	Brown
War and Peace	Sarah	Williams
The Odyssey	James	Jones

The Output window shows the execution log, including the following messages:

- 17 11:39:01 SET FOREIGN_KEY_CHECKS=1 0 row(s) affected 0.000 sec
- 18 12:19:02 SELECT * FROM Books LIMIT 0, 1000 77 row(s) returned 0.000 sec / 0.000 sec
- 19 12:19:02 SELECT * FROM Books LIMIT 0, 1000 77 row(s) returned 0.016 sec / 0.000 sec
- 20 12:19:49 SELECT * FROM Books WHERE Genre = 'Fiction' LIMIT 0, 1000 2 row(s) returned 0.000 sec / 0.000 sec
- 21 12:20:29 SELECT * FROM Books ORDER BY Title ASC LIMIT 0, 1000 77 row(s) returned 0.000 sec / 0.000 sec
- 22 12:40:41 SELECT Books.Title, Members.FirstName, Members.LastName FROM Transaction... 45 row(s) returned 0.000 sec / 0.000 sec
- 23 12:41:21 SELECT Books.Title, Members.FirstName, Members.LastName FROM Transaction... 45 row(s) returned 0.016 sec / 0.000 sec

The screenshot shows the MySQL Workbench interface. The Query window displays the following SQL query:

```
1 SELECT Books.Title, Members.FirstName, Members.LastName
2
3 FROM Books
4
5 LEFT JOIN Transactions ON Books.BookID = Transactions.BookID
6
7 LEFT JOIN Members ON Transactions.MemberID = Members.MemberID;
```

The Results window shows the output of the query, displaying a table with columns: Title, FirstName, and LastName. The data is as follows:

Title	FirstName	LastName
The Great Gatsby	John	Doe
To Kill a Mockingbird	Jane	Smith
1984	Michael	Johnson
Pride and Prejudice	Emily	Davis
Moby Dick	David	Brown
War and Peace	Sarah	Williams
The Odyssey	James	Jones

The Output window shows the execution log, including the following messages:

- 18 12:19:02 SELECT * FROM Books LIMIT 0, 1000 77 row(s) returned 0.000 sec / 0.000 sec
- 19 12:19:02 SELECT * FROM Books LIMIT 0, 1000 77 row(s) returned 0.016 sec / 0.000 sec
- 20 12:19:49 SELECT * FROM Books WHERE Genre = 'Fiction' LIMIT 0, 1000 2 row(s) returned 0.000 sec / 0.000 sec
- 21 12:20:29 SELECT * FROM Books ORDER BY Title ASC LIMIT 0, 1000 77 row(s) returned 0.000 sec / 0.000 sec
- 22 12:40:41 SELECT Books.Title, Members.FirstName, Members.LastName FROM Transaction... 45 row(s) returned 0.000 sec / 0.000 sec
- 23 12:41:21 SELECT Books.Title, Members.FirstName, Members.LastName FROM Transaction... 45 row(s) returned 0.016 sec / 0.000 sec
- 24 12:46:53 SELECT Books.Title, Members.FirstName, Members.LastName FROM Books LEF... 77 row(s) returned 0.000 sec / 0.000 sec

Additional Questions/Discussions:

- **How does the LEFT JOIN differ from the INNER JOIN?**
 - The difference between **INNER JOIN** and **LEFT JOIN** is in how they handle unmatched records. **INNER JOIN** returns only the rows with matching values in both tables, excluding unmatched rows. **LEFT JOIN** returns all rows from the left table, including unmatched ones, filling the right table's columns with NULL values where there's no match. **INNER JOIN** is used when only matching records are needed, while **LEFT JOIN** includes all records from the left table, regardless of matches in the right table.

Conclusions:

In conclusion, mastering SQL JOIN operations is essential for combining data from multiple tables in relational databases. By understanding different types of joins, such as INNER JOIN, LEFT JOIN, and others, students gain the ability to retrieve and manipulate data efficiently. This knowledge is crucial for working with real-world databases where information is often distributed across various tables. The ability to write optimized JOIN queries allows students to access comprehensive and accurate data, which is key to solving complex data-related problems in database management.