|  |  |
| --- | --- |
| **Name** | Sahil Shah |
| **UID No.** | 2021300115 |
| **Experiment No.** | 4 |

|  |  |
| --- | --- |
| **AIM:** | To perform join operations on the database. |
| **Program 1** | |
| **PROBLEM STATEMENT:** | Write queries on the tables in the database using joins on MySQL. |
| **THEORY:** | **What is a Join?**  A JOIN clause is used to combine rows from two or more tables, based on a related column between them.  **What are the different types of joins?**  Types of Joins:   * Natural Join * Inner or Simple Join * Left Outer Join * Right Outer Join * Full join      * The INNER JOIN keyword selects records that have matching values in both tables.   **Its syntax is as follows:**  SELECT column\_name(s)  FROM table1  INNER JOIN table2  ON table1.column\_name = table2.column\_name;   * The NATURAL JOIN keyword selects records that have matching values in both tables.   **Its syntax is as follows:**  SELECT column\_name(s)  FROM table1  INNER JOIN table2   * The LEFT JOIN keyword returns all records from the left table (table1), and the matching records (if any) from the right table (table2).   **Its syntax is as follows:**  SELECT column\_name(s)  FROM table1  LEFT JOIN table2  ON table1.column\_name = table2.column\_name;   * The RIGHT JOIN keyword returns all records from the right table (table2), and the matching records (if any) from the left table.   **Its syntax is as follows:**  SELECT column\_name(s)  FROM table1  RIGHT JOIN table2  ON table1.column\_name = table2.column\_name;   * The FULL JOIN keyword returns all records from both tables.   **Its syntax is as follows:**  SELECT column\_name(s)  FROM table1  LEFT JOIN table2  ON table1.column\_name = table2.column\_name;  UNION  SELECT column\_name(s)  FROM table1  RIGHT JOIN table2  ON table1.column\_name = table2.column\_name;  **Visual representation of the joins** |
| **QUERIES:**  **Using Create, Insert Into, Select Commands:**    **Table Room**    **Using Create, Insert Into, Select Commands:**    **Table Customers**     * **Using the Inner Join command:**   -- Inner Join  SELECT Customers.CustomerName, Customers.Aadhar, Customers.Contact, Room.RoomType, Room.RoomSize, Room.RoomNumber  FROM Customers  INNER JOIN Room  ON Customers.RoomNumber = Room.RoomNumber;  **Output:**     * **Using the Natural Join command:**   -- Natural Join  SELECT Customers.CustomerName, Customers.Aadhar, Customers.Contact, Room.RoomType, Room.RoomSize, Room.RoomNumber  FROM Customers  NATURAL JOIN Room;  **Output:**     * **Using the Left Join command:**   -- Left Join  SELECT Customers.CustomerName, Customers.Aadhar, Customers.Contact, Room.RoomType, Room.RoomSize, Customers.RoomNumber  FROM Customers  LEFT JOIN Room  ON Customers.RoomNumber = Room.RoomNumber;  **Output:**     * **Using the Right Join command:**   -- Right Join  SELECT Customers.CustomerName, Customers.Aadhar, Customers.Contact, Room.RoomType, Room.RoomSize, Room.RoomNumber  FROM Customers  RIGHT JOIN Room  ON Customers.RoomNumber = Room.RoomNumber;  **Output:**     * **Using the Full Join command:**   -- Full Join  SELECT Customers.CustomerName, Customers.Aadhar, Customers.Contact, Room.RoomType, Room.RoomSize, Room.RoomNumber  FROM Customers  LEFT JOIN Room  ON Customers.RoomNumber = Room.RoomNumber  UNION  SELECT Customers.CustomerName, Customers.Aadhar, Customers.Contact, Room.RoomType, Room.RoomSize, Room.RoomNumber  FROM Customers  RIGHT JOIN Room  ON Customers.RoomNumber = Room.RoomNumber;  **Output:** | |
| **CONCLUSION:**  In this experiment, I learned about the various joins in MySQL, and using that knowledge, I implemented all the joins on the two tables of the database – Room and Customers. There was no pre-defined syntax for the FULL JOIN in MySQL, therefore I’ve used the UNION operator to perform this and have mentioned its syntax with the UNION operator. I came to learn that MySQL offers CROSS JOIN which is simply the cartesian product of the two tables irrespective of the condition and it’s not to be confused with the FULL JOIN. | |