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:	

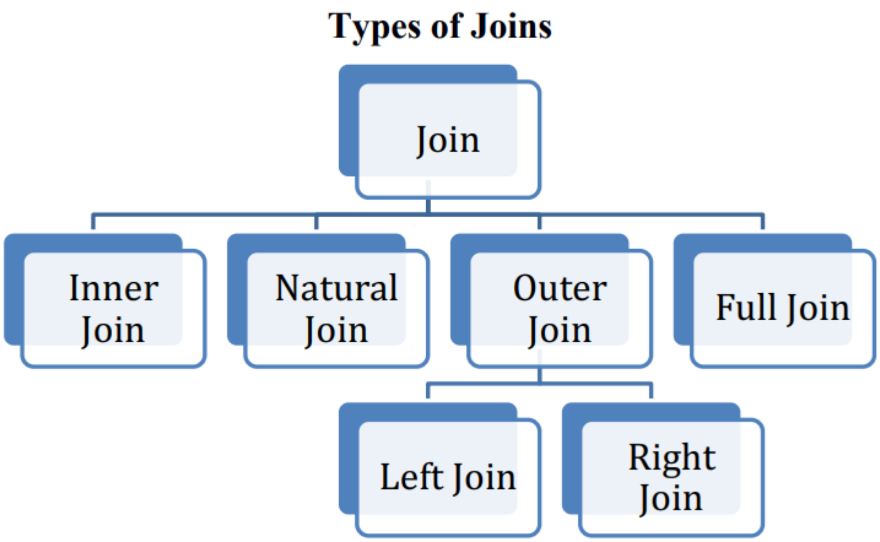
What is a Join?

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;
  ```

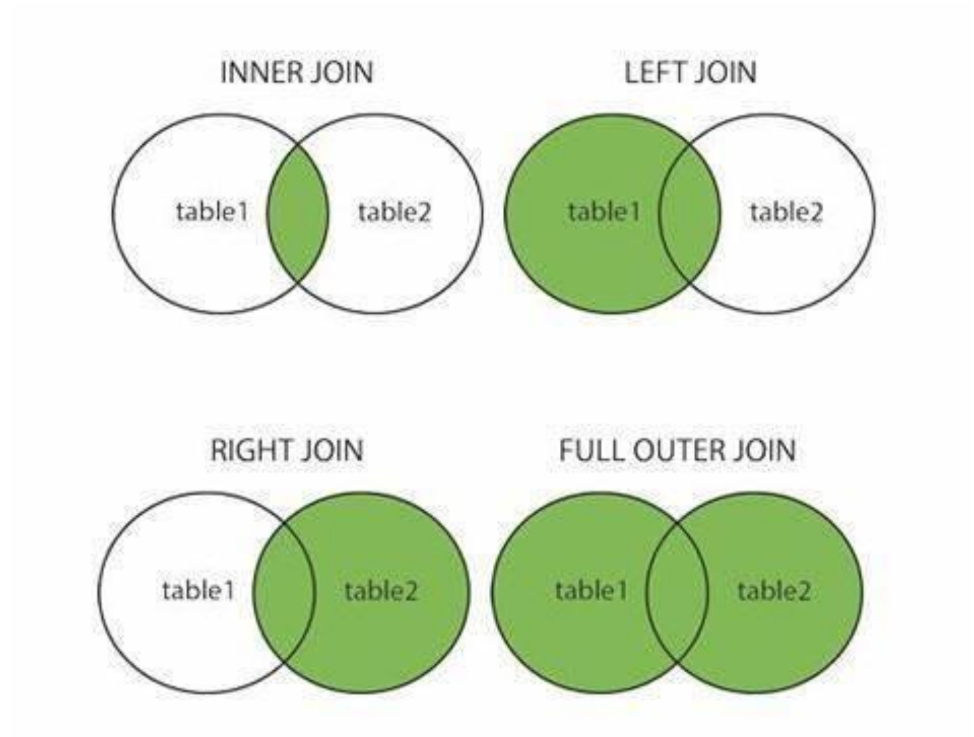
- The FULL JOIN keyword returns all records from both tables.

  Its syntax is as follows:

  ```
  SELECT column_name(s)
  ```

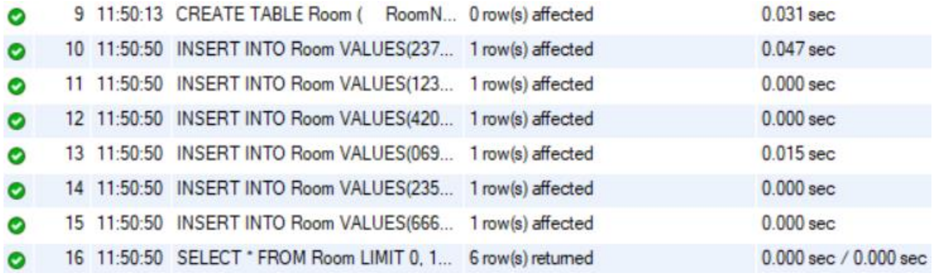
```
FROM table1
LEFT JOIN table2
ON table1.column_name = table2.column_name;
UNION
SELECT column_name(s)
FROM table1
RIGHT JOIN table2
ON table1.column_name = table2.column_name;
```

Visual representation of the joins



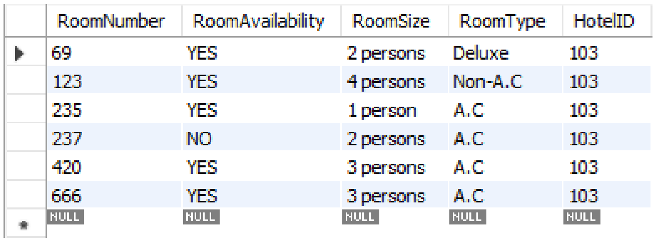
QUERIES:

Using Create, Insert Into, Select Commands:



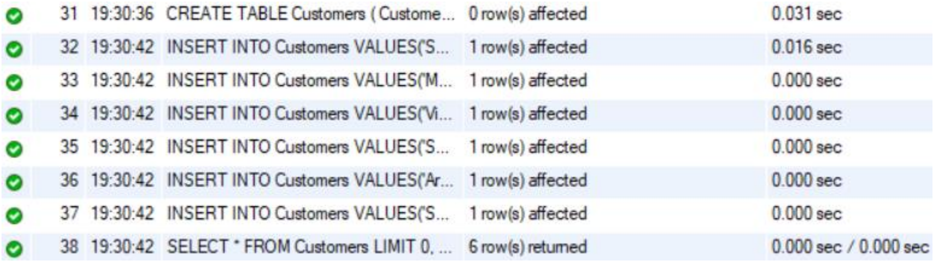
```
 9  11:50:13  CREATE TABLE Room (    RoomN...  0 row(s) affected   0.031 sec
10  11:50:50  INSERT INTO Room VALUES(237...   1 row(s) affected   0.047 sec
11  11:50:50  INSERT INTO Room VALUES(123...   1 row(s) affected   0.000 sec
12  11:50:50  INSERT INTO Room VALUES(420...   1 row(s) affected   0.000 sec
13  11:50:50  INSERT INTO Room VALUES(069...   1 row(s) affected   0.015 sec
14  11:50:50  INSERT INTO Room VALUES(235...   1 row(s) affected   0.000 sec
15  11:50:50  INSERT INTO Room VALUES(666...   1 row(s) affected   0.000 sec
16  11:50:50  SELECT * FROM Room LIMIT 0, 1...  6 row(s) returned  0.000 sec / 0.000 sec
```

Table Room



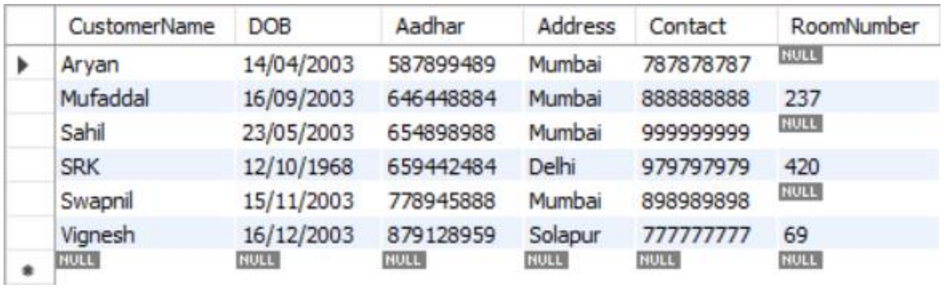
	RoomNumber	RoomAvailability	RoomSize	RoomType	HotelID
▶	69	YES	2 persons	Deluxe	103
	123	YES	4 persons	Non-A.C	103
	235	YES	1 person	A.C	103
	237	NO	2 persons	A.C	103
	420	YES	3 persons	A.C	103
	666	YES	3 persons	A.C	103
*	NULL	NULL	NULL	NULL	NULL

Using Create, Insert Into, Select Commands:



```
✓ 31 19:30:36 CREATE TABLE Customers ( Custome...  0 row(s) affected   0.031 sec
✓ 32 19:30:42 INSERT INTO Customers VALUES('S...   1 row(s) affected   0.016 sec
✓ 33 19:30:42 INSERT INTO Customers VALUES('M...   1 row(s) affected   0.000 sec
✓ 34 19:30:42 INSERT INTO Customers VALUES('Vi...  1 row(s) affected   0.000 sec
✓ 35 19:30:42 INSERT INTO Customers VALUES('S...   1 row(s) affected   0.000 sec
✓ 36 19:30:42 INSERT INTO Customers VALUES('Ar...  1 row(s) affected   0.000 sec
✓ 37 19:30:42 INSERT INTO Customers VALUES('S...   1 row(s) affected   0.000 sec
✓ 38 19:30:42 SELECT * FROM Customers LIMIT 0, ... 6 row(s) returned   0.000 sec / 0.000 sec
```

Table Customers

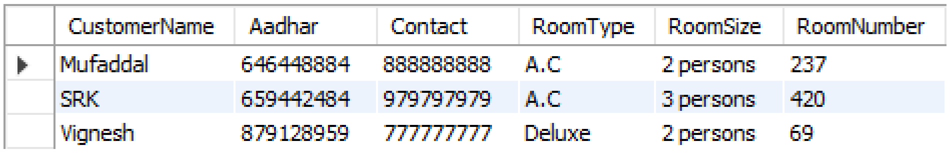


	CustomerName	DOB	Aadhar	Address	Contact	RoomNumber
▶	Aryan	14/04/2003	587899489	Mumbai	787878787	NULL
	Mufaddal	16/09/2003	646448884	Mumbai	888888888	237
	Sahil	23/05/2003	654898988	Mumbai	999999999	NULL
	SRK	12/10/1968	659442484	Delhi	979797979	420
	Swapnil	15/11/2003	778945888	Mumbai	898989898	NULL
	Vignesh	16/12/2003	879128959	Solapur	777777777	69
*	NULL	NULL	NULL	NULL	NULL	NULL

- **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:



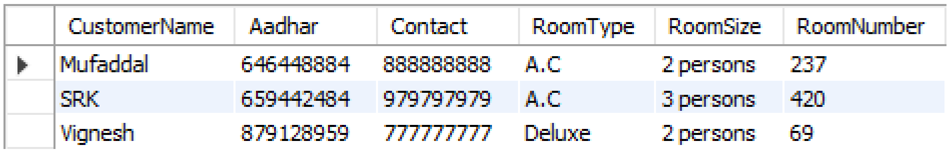
	CustomerName	Aadhar	Contact	RoomType	RoomSize	RoomNumber
▶	Mufaddal	646448884	888888888	A.C	2 persons	237
	SRK	659442484	979797979	A.C	3 persons	420
	Vignesh	879128959	777777777	Deluxe	2 persons	69

- **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:

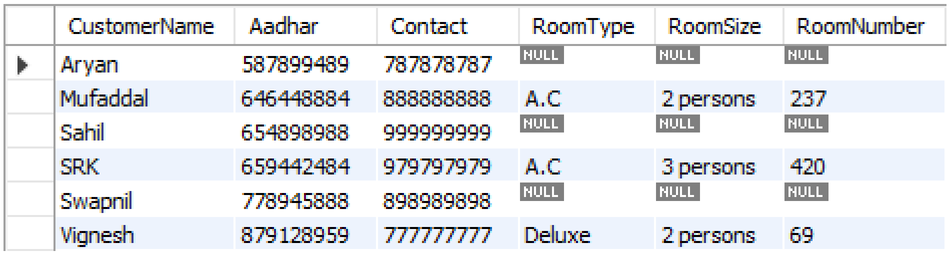


CustomerName	Aadhar	Contact	RoomType	RoomSize	RoomNumber
Mufaddal	646448884	888888888	A.C	2 persons	237
SRK	659442484	979797979	A.C	3 persons	420
Vignesh	879128959	777777777	Deluxe	2 persons	69

- **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:

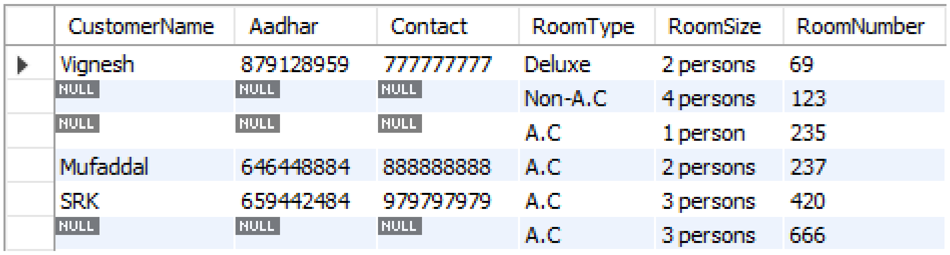


CustomerName	Aadhar	Contact	RoomType	RoomSize	RoomNumber
Aryan	587899489	787878787	NULL	NULL	NULL
Mufaddal	646448884	888888888	A.C	2 persons	237
Sahil	654898988	999999999	NULL	NULL	NULL
SRK	659442484	979797979	A.C	3 persons	420
Swapnil	778945888	898989898	NULL	NULL	NULL
Vignesh	879128959	777777777	Deluxe	2 persons	69

- **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:

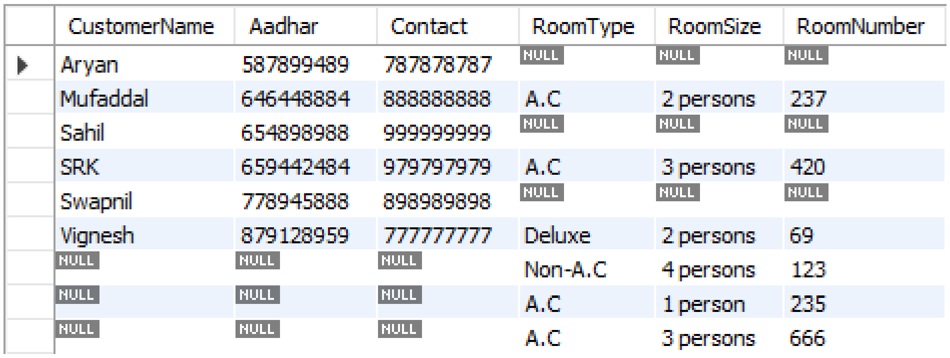


CustomerName	Aadhar	Contact	RoomType	RoomSize	RoomNumber
Vignesh	879128959	777777777	Deluxe	2 persons	69
NULL	NULL	NULL	Non-A.C	4 persons	123
NULL	NULL	NULL	A.C	1 person	235
Mufaddal	646448884	888888888	A.C	2 persons	237
SRK	659442484	979797979	A.C	3 persons	420
NULL	NULL	NULL	A.C	3 persons	666

- **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:



CustomerName	Aadhar	Contact	RoomType	RoomSize	RoomNumber
Aryan	587899489	787878787	NULL	NULL	NULL
Mufaddal	646448884	888888888	A.C	2 persons	237
Sahil	654898988	999999999	NULL	NULL	NULL
SRK	659442484	979797979	A.C	3 persons	420
Swapnil	778945888	898989898	NULL	NULL	NULL
Vignesh	879128959	777777777	Deluxe	2 persons	69
NULL	NULL	NULL	Non-A.C	4 persons	123
NULL	NULL	NULL	A.C	1 person	235
NULL	NULL	NULL	A.C	3 persons	666

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.