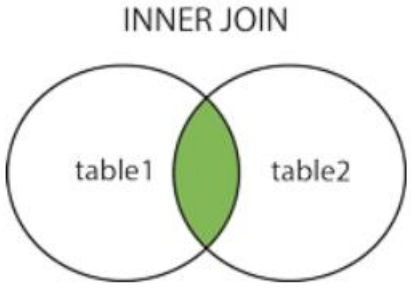
Bhartiya Vidya Bhavan's
Sardar Patel Institute of Technology
Bhavan's Campus, Munshi Nagar, Andheri (West), Mumbai-400058-India
(Autonomous College Affiliated to University of Mumbai)

NAME:	VAISHNAVI BORKAR, DEEPANSHU AGGARWAL, UTSAV AVAIYA
UID:	2021300016,2021300002,2021300005
SUBJECT	DBMS
EXPERIMENT NO :	4
DATE OF PERFORMANCE	4/11/22
DATE OF SUBMISSION	11/11/22
AIM:	**To study the JOIN operations**
SQL QUERIES/ COMMANDS/ THEORY:	**1.INNER JOIN:** The INNER JOIN keyword selects records that have matching values in both tables.  **QUERY: select * from student inner join courses where student.course_code = courses.course_code;** **2.NATURAL JOIN:**

Natural join is an SQL join operation that creates join on the base of the common columns in the tables. To perform natural join there must be one common attribute(Column) between two tables.

QUERY: select s_name,s_id,course_code from student natural join courses;

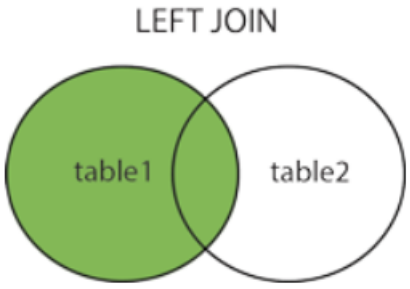
3.SELF JOIN:

A self join is a regular join, but the table is joined with itself. *T1* and *T2* are different table aliases for the same table.

QUERY: select t1.s_id from enrollment as t1,enrollment as t2 where t1.s_id = t2.s_id and t1.course_code<>t2.course_code;

4.LEFT OUTER JOIN:

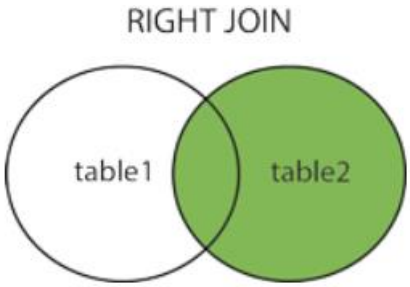
Returns all records from the left table, and the matched records from the right table.



QUERY: select s_name,s_id,course_name from student left outer join courses on (student.course_code = courses.course_code);

5.RIGHT OUTER JOIN:

Returns all records from the right table, and the matched records from the left table.



QUERY: select s_name,s_id,course_name from student right outer join courses on (student.course_code = courses.course_code);

RESULT:

1. INNER JOIN:

```
mysql> select * from student inner join courses where student.course_code = courses.course_code;
+---------+------+------------+------------+----------+-------------+-------------+-------------+--------------+-----------+--------+
| S_NAME  | S_ID | ADDRESS    | CONTACT_NO | field_NO | course_code | Course_Code | Total_Seats | Course_Name  | StartDate | Fee    |
+---------+------+------------+------------+----------+-------------+-------------+-------------+--------------+-----------+--------+
| DEEP    | 2001 | JAMMU      | 700606**** |        1 |        1011 |        1011 |          60 | JEE          | 1042      | 150000 |
| DEEPAK  | 2002 | LUDHIANA   | 990628**** |        1 |        1011 |        1011 |          60 | JEE          | 1042      | 150000 |
| TARAK   | 2003 | GUJRAT     | 98761****  |        1 |        1012 |        1012 |          60 | JEE ADVANCED | 1242      | 180000 |
| DHARA   | 2004 | WESTBENGAL | 700606**** |        1 |        1012 |        1012 |          60 | JEE ADVANCED | 1242      | 180000 |
| ANKITA  | 2005 | AMRITSAR   | 785426**** |        2 |        1013 |        1013 |          70 | NEET         | 842       | 130000 |
| VASUDHA | 2006 | RAIGARH    | 659823**** |        2 |        1013 |        1013 |          70 | NEET         | 842       | 130000 |
| JUSTIN  | 2009 | AMSTERDAM  | 663467**** |        1 |        1014 |        1014 |          70 | NEET PG      | 1042      | 170000 |
| TOMPER  | 2007 | MUMBAI     | 983467**** |        1 |        1015 |        1015 |         120 | GATE         | 472       | 100000 |
| VINEET  | 2008 | KOLABA     | 981167**** |        1 |        1015 |        1015 |         120 | GATE         | 472       | 100000 |
+---------+------+------------+------------+----------+-------------+-------------+-------------+--------------+-----------+--------+
9 rows in set (0.00 sec)
```

2.NATURAL JOIN:

```
mysql> select s_name,s_id,course_code from student natural join courses;
+---------+------+-------------+
| s_name  | s_id | course_code |
+---------+------+-------------+
| DEEP    | 2001 |        1011 |
| DEEPAK  | 2002 |        1011 |
| TARAK   | 2003 |        1012 |
| DHARA   | 2004 |        1012 |
| ANKITA  | 2005 |        1013 |
| VASUDHA | 2006 |        1013 |
| JUSTIN  | 2009 |        1014 |
| TOMPER  | 2007 |        1015 |
| VINEET  | 2008 |        1015 |
+---------+------+-------------+
9 rows in set (0.01 sec)
```

3.SELF JOIN:

```
mysql> select * from enrollment;
+------+-------------+
| S_id | course_code |
+------+-------------+
| 2002 |        1015 |
| 2003 |        1013 |
| 2004 |        1011 |
| 2005 |        1012 |
| 2006 |        1014 |
| 2007 |        1013 |
| 2008 |        1015 |
| 2009 |        1011 |
| 2001 |        1011 |
| 2001 |        1012 |
+------+-------------+
10 rows in set (0.00 sec)

mysql> select t1.s_id from enrollment as t1,enrollment as t2 where t1.s_id=t2.s_id and t1.course_code<>t2.course_code;
+------+
| s_id |
+------+
| 2001 |
| 2001 |
+------+
2 rows in set (0.00 sec)
```

4.LEFT OUTER JOIN:

```
mysql> select s_name,s_id,course_name from student left outer join courses on (student.course_code=courses.course_code);
+---------+------+--------------+
| s_name  | s_id | course_name  |
+---------+------+--------------+
| DEEP    | 2001 | JEE          |
| DEEPAK  | 2002 | JEE          |
| TARAK   | 2003 | JEE ADVANCED |
| DHARA   | 2004 | JEE ADVANCED |
| ANKITA  | 2005 | NEET         |
| VASUDHA | 2006 | NEET         |
| TOMPER  | 2007 | GATE         |
| VINEET  | 2008 | GATE         |
| JUSTIN  | 2009 | NEET PG      |
+---------+------+--------------+
9 rows in set (0.00 sec)

mysql>
```

5.RIGHT OUTER JOIN:

```
mysql> select s_name,s_id,course_name from student right outer join courses on (student.course_code=courses.course_code);
+---------+------+--------------+
| s_name  | s_id | course_name  |
+---------+------+--------------+
| DEEP    | 2001 | JEE          |
| DEEPAK  | 2002 | JEE          |
| TARAK   | 2003 | JEE ADVANCED |
| DHARA   | 2004 | JEE ADVANCED |
| ANKITA  | 2005 | NEET         |
| VASUDHA | 2006 | NEET         |
| JUSTIN  | 2009 | NEET PG      |
| TOMPER  | 2007 | GATE         |
| VINEET  | 2008 | GATE         |
+---------+------+--------------+
9 rows in set (0.00 sec)
```

CONCLUSION:	With the help of this experiment, we were able combine two or more records from a database by using values which are common to each.