

Name: Farhan Siddiqui

Roll No: 87

Sr. No.	DATE	TITLE	SIGN
1.		Study of Data Definition Language Statement	
2.		Study of Data Manipulation Language Statement	
3.		Study of SELECT Statement.	
4.		Draw ER diagram for given scenario/project/case study	
5.		Study of various type of JOINS	
6.		Study of different functions	
7.		Study of various types of SET OPERATORS	
8.		Study of various types of views	
9.		Study of subqueries with all its clauses	
10.		Study of Transaction (Commit/ Rollback), Locks	
11.		Implementing deadlocks	

Name:Farhan Siddiqui**Roll No.87****PRACTICAL NO. 5**

1. **INNER JOIN:** The INNER JOIN keyword selects all rows from both the tables as long as the satisfies.

SYNTAX:

SELECT table1.column1,table1.column2,table2.column1,....


FROM table1 INNER JOIN table2

ON table1.matching_column=table2.matching_column; Example:

SELECT EMP.DEPTNO,ENAME, SAL,JOB,DNAME,LOC

FROM EMP INNER JOIN DEPT

ON EMP.DEPTNO=DEPT.DEPTNO;

 Run SQL Command Line

```
SQL> select EMP.DEPTNO,ENAME,SAL,JOB,DNAME,LOC
2 FROM EMP INNER JOIN DEPT
3 ON EMP.DEPTNO=DEPT.DEPTNO;
```

DEPTNO	ENAME	SAL	JOB	DNAME	LOC
10	KING	5000	PRESIDENT	ACCOUNTING	NEW YORK
30	BLAKE	2850	MANAGER	SALES	CHICAGO
10	CLARK	2450	MANAGER	ACCOUNTING	NEW YORK
20	JONES	2975	MANAGER	RESEARCH	DALLAS
20	SCOTT	3000	ANALYST	RESEARCH	DALLAS
20	FORD	3000	ANALYST	RESEARCH	DALLAS
20	SMITH	800	CLERK	RESEARCH	DALLAS
30	ALLEN	1600	SALESMAN	SALES	CHICAGO
30	WARD	1250	SALESMAN	SALES	CHICAGO
30	MARTIN	1250	SALESMAN	SALES	CHICAGO
30	TUENER	1500	SALESMAN	SALES	CHICAGO
20	ADAMS	1100	CLERK	RESEARCH	DALLAS
30	JAMES	950	CLERK	SALES	CHICAGO
10	MILLER	1300	CLERK	ACCOUNTING	NEW YORK

14 rows selected.

2. **NATURAL JOIN:** A Natural Join is a type equi join which occurs implicitly by comparing all the same names columns in both tables. The join result has only one column for each pair of equally named columns

SYNTAX:

FY-IT

select * From table1 natural join table2;

Example:

Select * from EMP natural join DEPT;

```
SQL> Select * from EMP natural join DEPT;
```

DEPTNO	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	AGE	DNAME	LOC
10	7839	KING	PRESIDENT		17-NOV-81	5000			ACCOUNTING	NEW YORK
30	7698	BLAKE	MANAGER	7839	01-MAY-81	2850			SALES	CHICAGO
10	7782	CLARK	MANAGER	7839	09-JUN-81	2450			ACCOUNTING	NEW YORK
20	7566	JONES	MANAGER	7839	02-APR-81	2975			RESEARCH	DALLAS
20	7788	SCOTT	ANALYST	7566	19-APR-87	3000			RESEARCH	DALLAS
20	7902	FORD	ANALYST	7566	03-DEC-81	3000			RESEARCH	DALLAS
20	7369	SMITH	CLERK	7902	17-DEC-80	800			RESEARCH	DALLAS
30	7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300		SALES	CHICAGO
30	7521	WARD	SALESMAN	7698	22-FEB-81	1250	500		SALES	CHICAGO
30	7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400		SALES	CHICAGO
30	7844	TUENER	SALESMAN	7698	08-SEP-81	1500	0		SALES	CHICAGO
20	7876	ADAMS	CLERK	7788	23-MAY-81	1100			RESEARCH	DALLAS
30	7900	JAMES	CLERK	7698	03-DEC-81	950			SALES	CHICAGO
10	7934	MILLER	CLERK	7782	23-JAN-82	1300			ACCOUNTING	NEW YORK

14 rows selected.

3. OUTER JOIN:

Theta Join, Equijoin, and Natural Join are called inner joins. An inner join includes only those tuples with matching attributes and the rest are discarded in the resulting relation. Therefore, we need to use outer joins to include all the tuples from the participating relations in the resulting relation. There are three kinds of outer joins – left outer join, right outer join, and full outer join.

A) RIGHT OUTER JOIN:

SYNTAX:

SELECT table1.column1, table2.column2....

FROM table1

RIGHT JOIN table2

ON table1.column_field = table2.column_field;

EXAMPLE:

SELECT EMPNO,ENAMESAL,EMP.DEPTNO,DEPT.DEPTNO,DNAME

FROM DEPT RIGHT OUTER JOIN EMP ON

EMP.DEPTNO=DEPT.DEPTNO;

Run SQL Command Line

```
SQL> select EMPNO,ENAME,SAL,EMP.DEPTNO,DEPT.DEPTNO,DNAME
2  FROM DEPT RIGHT OUTER JOIN EMP
3  ON EMP.DEPTNO=DEPT.DEPTNO;
```

EMPNO	ENAME	SAL	DEPTNO	DEPTNO	DNAME
7839	KING	5000	10	10	ACCOUNTING
7698	BLAKE	2850	30	30	SALES
7782	CLARK	2450	10	10	ACCOUNTING
7566	JONES	2975	20	20	RESEARCH
7788	SCOTT	3000	20	20	RESEARCH
7902	FORD	3000	20	20	RESEARCH
7369	SMITH	800	20	20	RESEARCH
7499	ALLEN	1600	30	30	SALES
7521	WARD	1250	30	30	SALES
7654	MARTIN	1250	30	30	SALES
7844	TUENER	1500	30	30	SALES
7876	ADAMS	1100	20	20	RESEARCH
7900	JAMES	950	30	30	SALES
7934	MILLER	1300	10	10	ACCOUNTING

14 rows selected.

B) LEFT OUTER JOIN :

SYNTAX:

SELECT table1.column1, table2.column2....

FROM table1

LEFT JOIN table2

ON table1.column_field = table2.column_field;

EXAMPLE:

SELECT EMPNO,ENAME,SAL,EMP.DEPTNO,DEPT.DEPTNO,DNAME

FROM DEPT LEFT OUTER JOIN EMP

ON EMP.DEPTNO=DEPT.DEPTNO;

FY-IT

```
Run SQL Command Line
SQL> SELECT EMPNO,ENAME,SAL,EMP.DEPTNO,DEPT.DEPTNO,DNAME
2 FROM DEPT LEFT OUTER JOIN EMP
3 ON EMP.DEPTNO=DEPT.DEPTNO;
```

EMPNO	ENAME	SAL	DEPTNO	DEPTNO	DNAME
7839	KING	5000	10	10	ACCOUNTING
7698	BLAKE	2850	30	30	SALES
7782	CLARK	2450	10	10	ACCOUNTING
7566	JONES	2975	20	20	RESEARCH
7788	SCOTT	3000	20	20	RESEARCH
7902	FORD	3000	20	20	RESEARCH
7369	SMITH	800	20	20	RESEARCH
7499	ALLEN	1600	30	30	SALES
7521	WARD	1250	30	30	SALES
7654	MARTIN	1250	30	30	SALES
7844	TUENER	1500	30	30	SALES
7876	ADAMS	1100	20	20	RESEARCH
7900	JAMES	950	30	30	SALES
7934	MILLER	1300	10	10	ACCOUNTING
				40	OPERATIONS

15 rows selected.

C) FULL OUTER JOIN:

SYNTAX:

SELECT table1.column1, table2.column2....

FROM table1

FULL JOIN table2

ON table1.column_field = table2.column_field;

EXAMPLE:

SELECT EMPNO,ENAME,SAL,EMP.DEPTNO,DEPT.DEPTNO,DNAME

FROM DEPT FULL OUTER JOIN EMP

ON EMP.DEPTNO=DEPT.DEPTNO;

```
SQL> SELECT EMPNO,ENAME,SAL,EMP.DEPTNO,DEPT.DEPTNO,DNAME
2 FROM DEPT FULL OUTER JOIN EMP
3 ON EMP.DEPTNO=DEPT.DEPTNO;
```

EMPNO	ENAME	SAL	DEPTNO	DEPTNO	DNAME
7839	KING	5000	10	10	ACCOUNTING
7698	BLAKE	2850	30	30	SALES
7782	CLARK	2450	10	10	ACCOUNTING
7566	JONES	2975	20	20	RESEARCH
7788	SCOTT	3000	20	20	RESEARCH
7902	FORD	3000	20	20	RESEARCH
7369	SMITH	800	20	20	RESEARCH
7499	ALLEN	1600	30	30	SALES
7521	WARD	1250	30	30	SALES
7654	MARTIN	1250	30	30	SALES
7844	TUENER	1500	30	30	SALES
7876	ADAMS	1100	20	20	RESEARCH
7900	JAMES	950	30	30	SALES
7934	MILLER	1300	10	10	ACCOUNTING
				40	OPERATIONS

15 rows selected.

4. **CROSS JOIN:** When each row of first table is combined with each row from the second table, known as Cartesian join or cross join.

SYNTAX:

FY-IT

SELECT * FROM TABLE1 CROSS JOIN TABLE2

OR

SELECT * FROM TABLE1 , TABLE2 EXAMPLE:

SELECT * FROM EMP CROSS JOIN DEPT;

OR

SELECT * FROM EMP, DEPT

SQL> SELECT * FROM EMP CROSS JOIN DEPT;

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO	AGE	DEPTNO	DNAME	LOC
7839	KING	PRESIDENT		17-NOV-81	5000		10		10	ACCOUNTING	NEW YORK
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		30		10	ACCOUNTING	NEW YORK
7782	CLARK	MANAGER	7839	09-JUN-81	2450		10		10	ACCOUNTING	NEW YORK
7566	JONES	MANAGER	7839	02-APR-81	2975		20		10	ACCOUNTING	NEW YORK
7788	SCOTT	ANALYST	7566	19-APR-87	3000		20		10	ACCOUNTING	NEW YORK
7902	FORD	ANALYST	7566	03-DEC-81	3000		20		10	ACCOUNTING	NEW YORK
7369	SMITH	CLERK	7902	17-DEC-80	800		20		10	ACCOUNTING	NEW YORK
7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	30		10	ACCOUNTING	NEW YORK
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30		10	ACCOUNTING	NEW YORK
7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30		10	ACCOUNTING	NEW YORK
7844	TUENER	SALESMAN	7698	08-SEP-81	1500	0	30		10	ACCOUNTING	NEW YORK
7876	ADAMS	CLERK	7788	23-MAY-81	1100		20		10	ACCOUNTING	NEW YORK
7900	JAMES	CLERK	7698	03-DEC-81	950		30		10	ACCOUNTING	NEW YORK
7934	MILLER	CLERK	7782	23-JAN-82	1300		10		10	ACCOUNTING	NEW YORK
7839	KING	PRESIDENT		17-NOV-81	5000		10		20	RESEARCH	DALLAS
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		30		20	RESEARCH	DALLAS
7782	CLARK	MANAGER	7839	09-JUN-81	2450		10		20	RESEARCH	DALLAS
7566	JONES	MANAGER	7839	02-APR-81	2975		20		20	RESEARCH	DALLAS
7788	SCOTT	ANALYST	7566	19-APR-87	3000		20		20	RESEARCH	DALLAS
7902	FORD	ANALYST	7566	03-DEC-81	3000		20		20	RESEARCH	DALLAS
7369	SMITH	CLERK	7902	17-DEC-80	800		20		20	RESEARCH	DALLAS
7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	30		20	RESEARCH	DALLAS
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30		20	RESEARCH	DALLAS
7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30		20	RESEARCH	DALLAS
7844	TUENER	SALESMAN	7698	08-SEP-81	1500	0	30		20	RESEARCH	DALLAS
7876	ADAMS	CLERK	7788	23-MAY-81	1100		20		20	RESEARCH	DALLAS
7900	JAMES	CLERK	7698	03-DEC-81	950		30		20	RESEARCH	DALLAS
7934	MILLER	CLERK	7782	23-JAN-82	1300		10		20	RESEARCH	DALLAS
7839	KING	PRESIDENT		17-NOV-81	5000		10		30	SALES	CHICAGO
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		30		30	SALES	CHICAGO
7782	CLARK	MANAGER	7839	09-JUN-81	2450		10		30	SALES	CHICAGO
7566	JONES	MANAGER	7839	02-APR-81	2975		20		30	SALES	CHICAGO
7788	SCOTT	ANALYST	7566	19-APR-87	3000		20		30	SALES	CHICAGO
7902	FORD	ANALYST	7566	03-DEC-81	3000		20		30	SALES	CHICAGO
7369	SMITH	CLERK	7902	17-DEC-80	800		20		30	SALES	CHICAGO
7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	30		30	SALES	CHICAGO
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30		30	SALES	CHICAGO
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30		30	SALES	CHICAGO

7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	30		30	SALES	CHICAGO
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30		30	SALES	CHICAGO
7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30		30	SALES	CHICAGO
7844	TUENER	SALESMAN	7698	08-SEP-81	1500	0	30		30	SALES	CHICAGO
7876	ADAMS	CLERK	7788	23-MAY-81	1100		20		30	SALES	CHICAGO
7900	JAMES	CLERK	7698	03-DEC-81	950		30		30	SALES	CHICAGO
7934	MILLER	CLERK	7782	23-JAN-82	1300		10		30	SALES	CHICAGO
7839	KING	PRESIDENT		17-NOV-81	5000		10		40	OPERATIONS	BOSTON
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		30		40	OPERATIONS	BOSTON
7782	CLARK	MANAGER	7839	09-JUN-81	2450		10		40	OPERATIONS	BOSTON
7566	JONES	MANAGER	7839	02-APR-81	2975		20		40	OPERATIONS	BOSTON
7788	SCOTT	ANALYST	7566	19-APR-87	3000		20		40	OPERATIONS	BOSTON
7902	FORD	ANALYST	7566	03-DEC-81	3000		20		40	OPERATIONS	BOSTON
7369	SMITH	CLERK	7902	17-DEC-80	800		20		40	OPERATIONS	BOSTON
7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	30		40	OPERATIONS	BOSTON
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30		40	OPERATIONS	BOSTON
7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30		40	OPERATIONS	BOSTON
7844	TUENER	SALESMAN	7698	08-SEP-81	1500	0	30		40	OPERATIONS	BOSTON
7876	ADAMS	CLERK	7788	23-MAY-81	1100		20		40	OPERATIONS	BOSTON
7900	JAMES	CLERK	7698	03-DEC-81	950		30		40	OPERATIONS	BOSTON
7934	MILLER	CLERK	7782	23-JAN-82	1300		10		40	OPERATIONS	BOSTON

56 rows selected.

5. **SELF JOIN:** The SQL **SELF JOIN** is used to join a table to itself as if the table were two tables; temporarily renaming at least one table in the SQL statement.

SYNTAX:

```
SELECT a.column_name, b.column_name
```

```
FROM table1 a, table1 b
```

```
WHERE a.common_field = b.common_field; EXAMPLE:
```

```
SELECT a.ENAME, b.ENAME
```

```
FROM EMP a, EMP BWHERE a.EMPNO = b.DEPTNO;
```

FY-IT

Harsh Davra

Roll-no:-09

```
SQL> SELECT b.ENAME EMPLOYEE, a.ENAME MANAGER
2  FROM EMP a, EMP b
3  WHERE a.EMPNO=b.MGR;
```

EMPLOYEE	MANAGER
JONES	KING
CLARK	KING
BLAKE	KING
JAMES	BLAKE
TUENER	BLAKE
MARTIN	BLAKE
WARD	BLAKE
ALLEN	BLAKE
MILLER	CLARK
FORD	JONES
SCOTT	JONES
ADAMS	SCOTT
SMITH	FORD

```
13 rows selected.
```



**MALAD KANDIVALI EDUCATION SOCIETY'S
NAGINDAS KHANDWALA COLLEGE OF COMMERCE,
ARTS & MANAGEMENT STUDIES & SHANTABEN NAGINDAS
KHANDWALA COLLEGE OF SCIENCE
MALAD [W], MUMBAI – 64
(AUTONOMOUS)**

**(Reaccredited 'A' Grade by NAAC)
(AFFILIATED TO UNIVERSITY OF MUMBAI)
(ISO 9001:2015)**

CERTIFICATE

Name: Mr. Farhan Siddiqui

Roll No: 87 Programme: BSc IT Semester: II

This is certified to be a bonafide record of practical works done by the above student in the college laboratory for the course **Database Management Systems I** (Course Code: **2023UISPR**) for the partial fulfillment of Second Semester of BSc IT/CS during the academic year 2020-2021.

The journal work is the original study work that has been duly approved in the year 2020-2021 by the undersigned.

External Examiner

**Subject-In-Charge
(Ms.Sweety Garg)**

Date of Examination: (College Stamp)