

Joins Assignment 1

a) What are joins in SQL? State its types.

A JOIN in SQL is used to combine rows from two or more tables based on a related column (usually a primary key–foreign key relationship).

It allows you to retrieve data that is spread across multiple tables.

General Syntax:

SELECT columns

FROM table1

JOIN table2

ON table1.column = table2.column;

- ◆ **Types of Joins in SQL**

a) INNER JOIN

Returns only the rows that have matching values in both tables.

SELECT E.name, D.dept_name

FROM Employee E

INNER JOIN Department D

ON E.dept_id = D.dept_id;

b) LEFT JOIN (or LEFT OUTER JOIN)

Returns all rows from the left table, and matching rows from the right table.

If no match, NULL is returned from the right side.

SELECT E.name, D.dept_name

FROM Employee E

LEFT JOIN Department D

ON E.dept_id = D.dept_id;

c) RIGHT JOIN (or RIGHT OUTER JOIN)

Returns all rows from the right table, and matching rows from the left table.

SELECT E.name, D.dept_name

FROM Employee E

RIGHT JOIN Department D

```
ON E.dept_id = D.dept_id;
```

d) FULL JOIN (or FULL OUTER JOIN)

Returns all rows from both tables, with NULLs where no match exists.

```
SELECT E.name, D.dept_name
```

```
FROM Employee E
```

```
FULL OUTER JOIN Department D
```

```
ON E.dept_id = D.dept_id;
```

e) CROSS JOIN

Returns the Cartesian product of two tables (every row from the first table is combined with every row from the second).

If Table A has m rows and Table B has n rows → result will have $m \times n$ rows.

```
SELECT E.name, D.dept_name
```

```
FROM Employee E
```

```
CROSS JOIN Department D;
```

f) SELF JOIN

A table joins with itself (commonly used for hierarchical data).

```
SELECT E1.name AS Employee, E2.name AS Manager
```

```
FROM Employee E1
```

```
INNER JOIN Employee E2
```

```
ON E1.manager_id = E2.emp_id;
```

2) Define Cross join.

A CROSS JOIN is a type of join that returns the Cartesian product of two tables, meaning it pairs each row from the first table with all rows of the second table.

- No ON condition is needed.
- Result size = (Number of rows in Table A) \times (Number of rows in Table B).

Example:

If Employee has 3 rows and Department has 2 rows, then:

```
SELECT *
```

```
FROM Employee
```

```
CROSS JOIN Department;
```