PW SKILLS

Lesson Plan

SQL Joins

SQL Joins

A join is used to combine two or more tables. When you want to retrieve data from two tables, you have to use joins.
To perform joins, we first create tables and insert values into them.
We are creating three tables: department, manager, and Employee.

```
create database PWIOI
USE PWIOI
Create table department(dept_id int primary key,
dept_name varchar(20));

Create table manager(manager_id int primary key,
manager_name varchar(20),
dept_id int,
 FOREIGN KEY(dept_id) REFERENCES department(dept_id));

Create table Employee(
emp_id int primary key,
emp_name varchar(30),
 salary float,
 dept_id int,
 FOREIGN KEY(dept_id) REFERENCES department(dept_id),
 manager_id int,
 FOREIGN KEY(manager_id) REFERENCES manager(manager_id));
```

Check the structure of the tables-

desc manager;
desc department;
desc Employee;

Insert values into the tables-

```
insert into department(dept_id,dept_name)
values(10,"Finance"),(20,"Data Science"),(30,"Marketing");

insert into manager(manager_id,manager_name,dept_id)
values(101,"daksh",10),(102,"shiv",20),(103,"ganesh",30),(104,"var
un",10);

insert into Employee(emp_id,emp_name,salary,dept_id,manager_id)
values(1000,"poonam",35000,10,101),(2000,"shruti",45000,20,102),(3000,"rajani",50000,30,103),(4000,"ram",35000,10,104),
(5000,"vinay",45000,20,101),(6000,"sushma",35000,30,102);
```

Check the inserted values:

```
300 • select * from Employee;
```

Result Grid | Filter Rows: | Edit:

emp_id	emp_name	salary	dept_id	manager_id
1000	poonam	35000	10	101
2000	shruti	45000	20	102
3000	rajani	50000	30	103
4000	ram	35000	10	104
5000	vinay	45000	20	101
6000	sushma	35000	30	102

```
301 • select * from manager;
302
```

Result Grid | Filter Rows:

manager_id	manager_name	dept_id
101	daksh	10
102	shiv	20
103	ganesh	30
104	var un	10

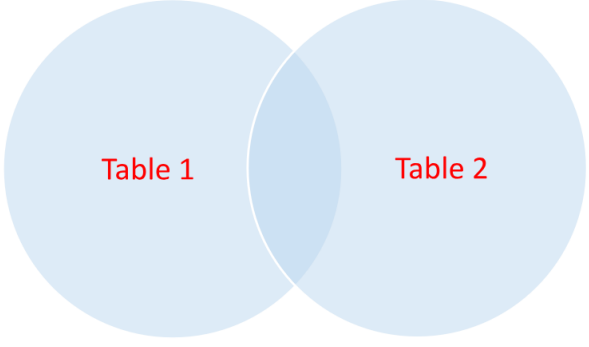
```
301 • select * from department;
302
```

Result Grid | Filter Rows:

dept_id	dept_name
10	Finance
20	Data Science
30	Marketing
NULL	NULL

INNER JOIN

An INNER join only returns those rows that have a match in both joined tables.
In the following diagram, the shaded region indicates the inner join.



Example:
Fetch the employee's name and the department name they belong to.

Query:

```
303 •  SELECT e.emp_name , d.dept_name
304    FROM Employee e
305    JOIN department d ON e.dept_id = d.dept_id;
306
```

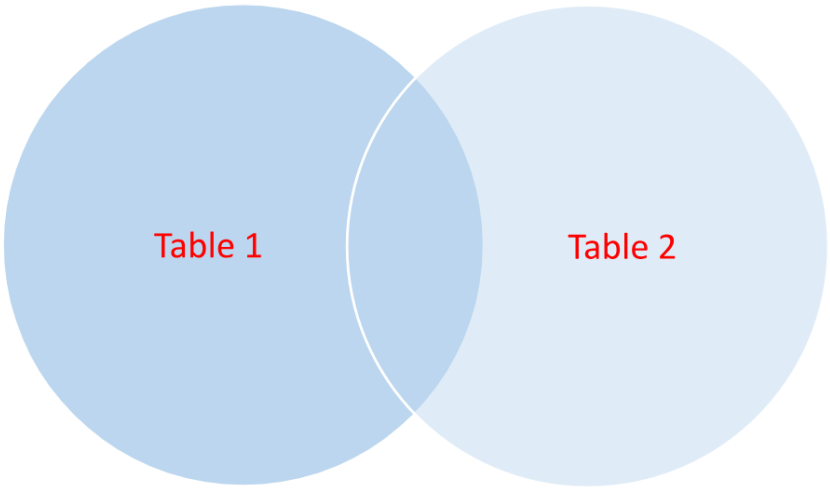
Result Grid | Filter Rows: | Export: | Wra

emp_name	dept_name
poonam	Finance
ram	Finance
shruti	Data Science
vinay	Data Science
rajani	Marketing
sushma	Marketing

You can use the keyword JOIN or INNER JOIN. It will give you the common records from the department and the Employee table.

LEFT JOIN

The left outer join takes all the tuples from the left relation that did not match with any tuple in the right relation, which pads the tuple with null values for all other attributes from the right relation, and adds them to the result of the natural join.



Example:
Fetch all the employees' names and the department names they belong to.
Here we have to fetch all the employee records, so we will apply for a leave here.

Query:

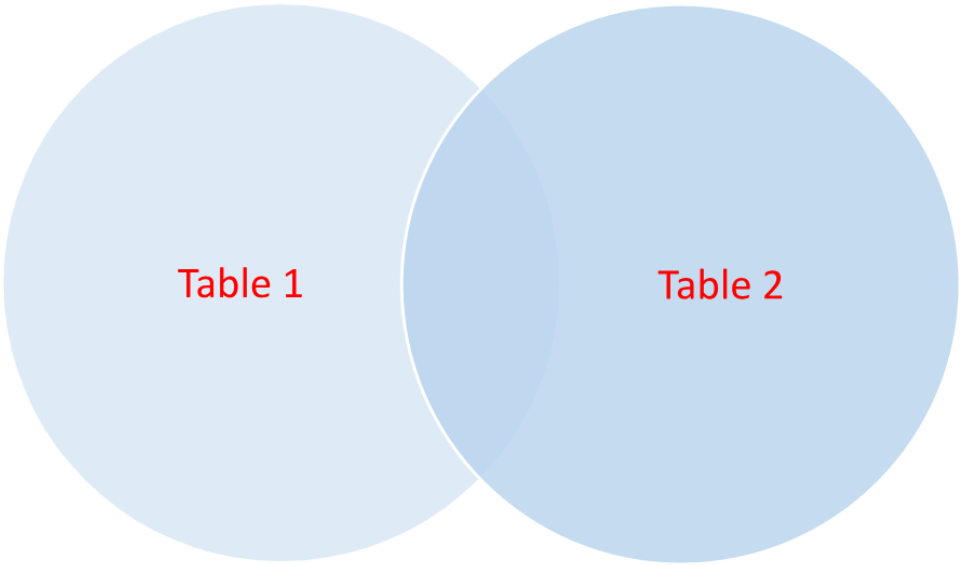
```
310 •  SELECT e.emp_name , d.dept_name
311    FROM Employee e
312    LEFT JOIN department d ON e.dept_id = d.dept_id;
313
314
315
316
```

Result Grid | Filter Rows: | Export: | Wrap Cell Cont

emp_name	dept_name
poonam	Finance
shruti	Data Science
rajani	Marketing
ram	Finance
vinay	Data Science
sushma	Marketing

RIGHT JOIN

In the right outer join It allows us to keep all the tuples in the right relation (table). If no matching tuple is found in the left relation, then the attributes from the left relation in the result are filled with null values.



Example:
Fetch employees' names and the all department names they belong to.

Query:

```
313
314 •  SELECT e.emp_name , d.dept_name
315    FROM Employee e
316    RIGHT JOIN department d ON e.dept_id = d.dept_id;
```

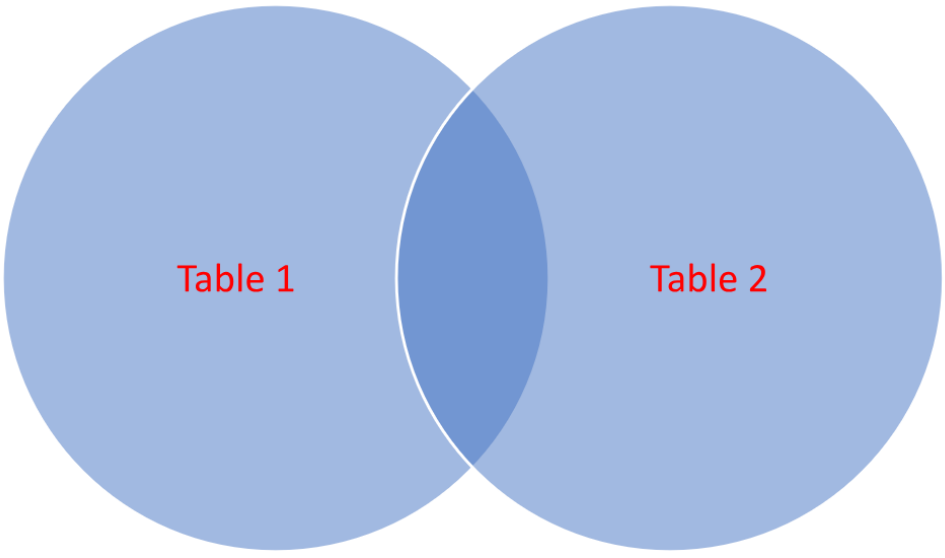
Result Grid | Filter Rows: | Export: | Wrap Cell Co

emp_name	dept_name
poonam	Finance
ram	Finance
shruti	Data Science
vinay	Data Science
rajani	Marketing
sushma	Marketing

Here, you will get all the common records from the Employee and the department table and all the remaining records from the department.

FULL OUTER JOIN

In the full outer join, all the tuples from both relations are added into the result.



SELF JOIN

Self-join it joins the table itself.

Query:

```
select a.manager_id , b.manager_name , a.dept_id
from manager a , manager b
where a.dept_id < b.dept_id;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

manager_id	manager_name	dept_id
101	shiv	10
101	ganesh	10
104	shiv	10
104	ganesh	10
102	ganesh	20

CROSS/CARTESIAN JOIN

When combining two tables without specifying a join condition, the database system merges each row from the first table with each record from the second.

Query:

```
SELECT e.emp_name , d.dept_name
FROM Employee e
CROSS JOIN department d;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Co

emp_name	dept_name
poonam	Data Science
poonam	Finance
shruti	Marketing
shruti	Data Science
shruti	Finance
rajani	Marketing
rajani	Data Science
rajani	Finance
ram	Marketing
ram	Data Science
ram	Finance
vinay	Marketing
vinay	Data Science
vinay	Finance
sushma	Marketing
sushma	Data Science
sushma	Finance

NATURAL JOIN

In SQL, a natural join merges data from two or more columns based on a shared column. In both tables, the common column must have the exact same name and data type. We do not need to explicitly provide the join condition because SQL connects the tables based on this shared field.

- The ON clause is not required for the join condition in a natural join.
- The result of a natural join will always contain columns that are unique.

Query:

```
331 •  SELECT * FROM Employee
332    NATURAL JOIN department;
333
334
335
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

dept_id	emp_id	emp_name	salary	manager_id	dept_name
10	1000	poonam	35000	101	Finance
20	2000	shruti	45000	102	Data Science
30	3000	rajani	50000	103	Marketing
10	4000	ram	35000	104	Finance
20	5000	vinay	45000	101	Data Science
30	6000	sushma	35000	102	Marketing