MySQL JOINS

MySQL JOINS are used with SELECT statement. It is used to retrieve data from multiple tables. It is performed whenever you need to fetch records from two or more tables.

There are three types of [MySQL](https://www.javatpoint.com/mysql-tutorial) joins:

* MySQL INNER JOIN (or sometimes called simple join)
* MySQL LEFT OUTER JOIN (or sometimes called LEFT JOIN)
* MySQL RIGHT OUTER JOIN (or sometimes called RIGHT JOIN)

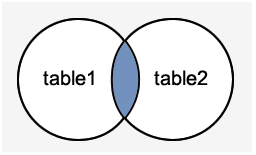
## **MySQL Inner JOIN (Simple Join)**

The [MySQL INNER JOIN](https://www.javatpoint.com/mysql-inner-join) is used to return all rows from multiple tables where the join condition is satisfied. It is the most common type of join.

**Syntax:**

1. **SELECT** columns
2. **FROM** table1
3. **INNER** JOIN table2
4. **ON** table1.**column** = table2.**column**;

**Image representation:**



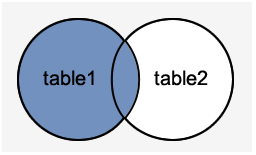
## **MySQL Left Outer Join**

The LEFT OUTER JOIN returns all rows from the left hand table specified in the ON condition and only those rows from the other table where the join condition is fulfilled.

**Syntax:**

1. **SELECT** columns
2. **FROM** table1
3. LEFT [OUTER] JOIN table2
4. **ON** table1.**column** = table2.**column**;

**Image representation:**



## **MySQL Right Outer Join**

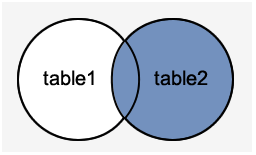
The MySQL Right Outer Join returns all rows from the RIGHT-hand table specified in the ON condition and only those rows from the other table where he join condition is fulfilled.

AD

**Syntax:**

1. **SELECT** columns
2. **FROM** table1
3. RIGHT [OUTER] JOIN table2
4. **ON** table1.**column** = table2.**column**;

**Image representation:**



**Table 1:**

CREATE TABLE employee (

emp\_id VARCHAR(10) PRIMARY KEY,

emp\_name VARCHAR(50),

salary INT,

dept\_id VARCHAR(10),

manager\_id VARCHAR(10));

INSERT INTO employee (emp\_id, emp\_name, salary, dept\_id, manager\_id)

VALUES

('E1', 'Rahul', 15000, 'D1', 'M1'),

('E2', 'Manoj', 15000, 'D1', 'M1'),

('E3', 'James', 55000, 'D2', 'M2'),

('E4', 'Michael', 25000, 'D2', 'M2'),

('E5', 'Ali', 20000, 'D10', 'M3'),

('E6', 'Robin', 35000, 'D10', 'M3');

**Table 2:**

CREATE TABLE projects (

project\_id VARCHAR(10) NOT NULL,

project\_name VARCHAR(50),

team\_member\_id VARCHAR(10),

PRIMARY KEY (project\_id, team\_member\_id));

INSERT INTO projects (project\_id, project\_name, team\_member\_id)

VALUES

('P1', 'Data Migrations', 'E1'),

('P1', 'Data Migrations', 'E2'),

('P1', 'Data Migrations', 'M3'),

('P2', 'ETL Tools', 'E1'),

('P2', 'ETL Tools', 'M4');

**Table 3:**

CREATE TABLE Manager (

manager\_id VARCHAR(10) PRIMARY KEY,

manager\_name VARCHAR(50),

dept\_id VARCHAR(10));

INSERT INTO Manager (manager\_id, manager\_name, dept\_id)

VALUES

('M1', 'Prem', 'D3'),

('M2', 'Shripadh', 'D4'),

('M3', 'Nick', 'D1'),

('M4', 'Cory', 'D1');

**Table 4:**

CREATE TABLE department (

dept\_id VARCHAR(10) PRIMARY KEY,

dept\_name VARCHAR(50)

);

INSERT INTO department (dept\_id, dept\_name)

VALUES

('D1', 'IT'),

('D2', 'HR'),

('D3', 'Finance'),

('D4', 'Admin');

* **Fetch the employee name and the department name they belong to.**
* **Inner Join / Join (Fetch Matching Records only)**

Select e.emp\_name, d.dept\_name

From employee e

Join department d on e.dept\_id = d.dept\_id;

* **Fetch all the employee name and their department name they belong to.**
* **Left Join = inner join + any additional information in the left table.**

Select e.emp\_name, d.dept\_name

From employee e

Left Join department d on e.dept\_id = d.dept\_id;

* **Fetch all the department names and their employee name they belong to.**
* **Right Join = inner join + any additional information in the right table.**

Select e.emp\_name, d.dept\_name

From employee e

right Join department d on e.dept\_id = d.dept\_id;

* Fetch details od All Employee, their manager, their department and the project they work on.

**Step 1:**

Select e.emp\_name, d.dept\_name

From employee e

Left Join department d on e.dept\_id = d.dept\_id;

**Step 2:**

Select e.emp\_name, d.dept\_name, m.manager\_name

From employee e

Left Join department d on e.dept\_id = d.dept\_id

Inner Join manager m on m.manager\_id = e.manager\_id;

**Step 3:**

Select e.emp\_name, d.dept\_name, m.manager\_name, p.project\_name

From employee e

Left Join department d on e.dept\_id = d.dept\_id

Inner Join manager m on m.manager\_id = e.manager\_id

Left Join projects p on p.team\_member\_id = e.emp\_id;

* Full Join/ Full outer Join = Inner Join

+ all the remaining records from left table

+ all remaining records from right table

select e.emp\_name, d.dept\_name

from employee e

full join department d on d.dept\_id = e.dept\_id;

Query using in MySQL DB:

SELECT e.emp\_name, d.dept\_name

FROM employee e

LEFT JOIN department d ON d.dept\_id = e.dept\_id

UNION

SELECT e.emp\_name, d.dept\_name

FROM employee e

RIGHT JOIN department d ON d.dept\_id = e.dept\_id;

* Cross Join – Cartesian Product

SELECT e.emp\_name, d.dept\_name

FROM employee e

CROSS JOIN department d;

---write a query to fetch the employee name and their corresponding department names

---Also make sure to display the company name and the company location corresponding to each employee.

As we can’t join w.r.t certain columns in the table, we can use cross join

SELECT e.emp\_name, d.dept\_name, c. company\_name, c.location

FROM employee e

RIGHT JOIN department d ON d.dept\_id = e.dept\_id

Cross join company c;