

---

## 4.SQL JOIN

The SQL Join help in retrieving data from two or more database tables. The tables are mutually related using primary keys and foreign keys.

### Type of Join

#### INNER JOIN

The **INNER JOIN** is used to print rows from both tables that satisfy the given condition. For example, the user wants to get a list of users who have rented movies together with titles of movies rented by them. Users can use an INNER JOIN for that, which returns rows from both tables that satisfy with given conditions.

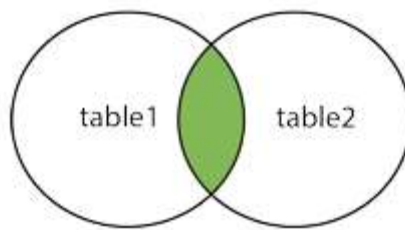


Fig. INNER JOIN

The INNER JOIN keyword selects records that have matching values in both the tables.

#### INNER JOIN Syntax

```
SELECT column_name(s)
FROM table1
INNER JOIN table2
ON table1.column_name = table2.column_name;
```

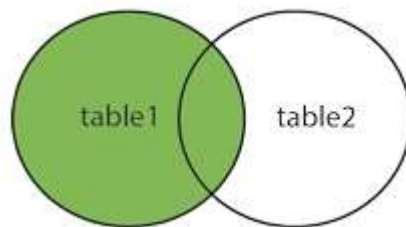
```
SELECT city.city_id, country.country, city.last_update, country.last_update FROM city
INNER JOIN country ON city.country_id = country.country_id
```

```
mysql> SELECT city.city_id, country.country, city.last_update, country.last_update
-> FROM city
-> INNER JOIN country ON city.country_id=country.country_id;
```

city_id	country	last_update	last_update
251	Afghanistan	2006-02-15 04:45:25	2006-02-15 04:44:00
59	Algeria	2006-02-15 04:45:25	2006-02-15 04:44:00
63	Algeria	2006-02-15 04:45:25	2006-02-15 04:44:00
483	Algeria	2006-02-15 04:45:25	2006-02-15 04:44:00
516	American Samoa	2006-02-15 04:45:25	2006-02-15 04:44:00
67	Angola	2006-02-15 04:45:25	2006-02-15 04:44:00
360	Angola	2006-02-15 04:45:25	2006-02-15 04:44:00
493	Anguilla	2006-02-15 04:45:25	2006-02-15 04:44:00
20	Argentina	2006-02-15 04:45:25	2006-02-15 04:44:00
43	Argentina	2006-02-15 04:45:25	2006-02-15 04:44:00
45	Argentina	2006-02-15 04:45:25	2006-02-15 04:44:00
128	Argentina	2006-02-15 04:45:25	2006-02-15 04:44:00

## LEFT JOIN

The **LEFT JOIN** returns all the records from the table1 (left table) and the matched records from the table2 (right table). The output is NULL from the right side if there is no match.



Left Join

### LEFT JOIN Syntax

```
SELECT column_name(s)
```

```
FROM table1
```

```
LEFT JOIN table2
```

```
ON table1.column_name = table2.column_name;
```

```
SELECT city.city_id, country.country, city.last_update, country.last_update FROM city
```

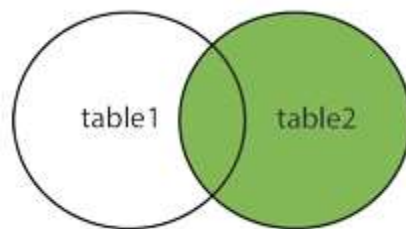
```
LEFT JOIN country ON city.country_id = country.country_id
```

```
mysql> SELECT city.city_id, country.country, city.last_update
-> FROM country
-> LEFT JOIN city ON city.country_id=country.country_id;
```

city_id	country	last_update
251	Afghanistan	2006-02-15 04:45:25
59	Algeria	2006-02-15 04:45:25
63	Algeria	2006-02-15 04:45:25
483	Algeria	2006-02-15 04:45:25
516	American Samoa	2006-02-15 04:45:25
67	Angola	2006-02-15 04:45:25
360	Angola	2006-02-15 04:45:25
493	Anguilla	2006-02-15 04:45:25

## RIGHT JOIN

The RIGHT JOIN is the opposite of LEFT JOIN. The RIGHT JOIN prints all the columns from the table2(right table) even if there no matching rows have been found in the table1 (left table). If there no matches have been found in the table (left table), NULL is returned.



RIGHT JOIN

### RIGHT JOIN Syntax

```
SELECT column_name(s)

FROM table1

RIGHT JOIN table2 ON table1.column_name = table2.column_name;

SELECT city.city_id, country.country, city.last_update, country.last_update FROM city

RIGHT JOIN country ON city.country_id = country.country_id
```

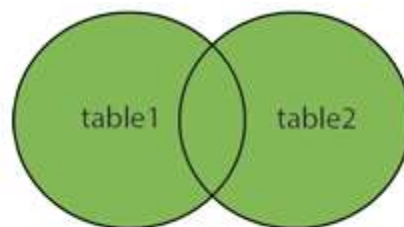
```
mysql> SELECT city.city_id, country.country, city.last_update
-> FROM city
-> RIGHT JOIN country ON city.country_id=country.country_id;
```

city_id	country	last_update
251	Afghanistan	2006-02-15 04:45:25
59	Algeria	2006-02-15 04:45:25
63	Algeria	2006-02-15 04:45:25
483	Algeria	2006-02-15 04:45:25
516	American Samoa	2006-02-15 04:45:25

## Full OUTER JOIN

The FULL OUTER JOIN keyword returns all records when there are a match in left (table1) or right (table2) table records.

**Note:** FULL OUTER JOIN can potentially return very large result-sets!



Full Join

Tip: FULL OUTER JOIN and FULL JOIN are the same.

### FULL OUTER JOIN Syntax

```
SELECT column_name(s)
FROM table1
FULL OUTER JOIN table2
ON table1.column_name = table2.column_name WHERE condition;
```

**Note:** MySQL does not support the Full Join, so we can perform left join and right join separately then take the union of them.

---

```
SELECT * FROM t1  
  
LEFT JOIN t2 ON t1.id = t2.id  
  
UNION  
  
SELECT * FROM t1  
  
RIGHT JOIN t2 ON t1.id = t2.id
```

## **SELF-JOIN**

A self-JOIN is a regular join, but the table is joined with itself.

### **Self -JOIN Syntax**

```
SELECT column_name(s)  
  
FROM table1 T1, table1 T2  
  
WHERE condition;
```