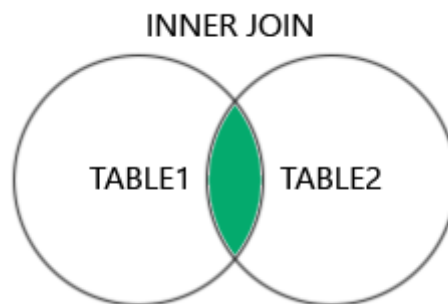


## JOIN

It is used to combine rows from two or more tables based on a related column between them.

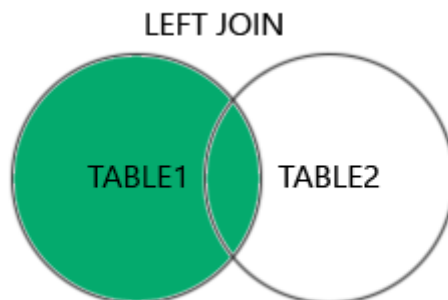
### Types of JOINS

**(INNER) JOIN:** Returns records that have matching values in both tables.



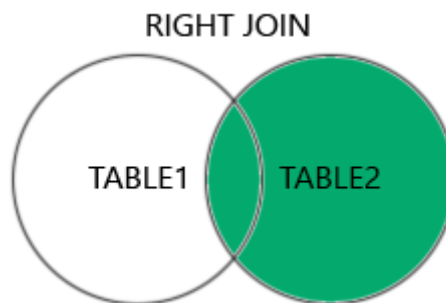
```
SELECT * FROM customer INNER JOIN orders ON orders.cust_id=customer.cust_id;
```

**LEFT (OUTER) JOIN:** Returns all records from the left table, and the matched records from the right table.



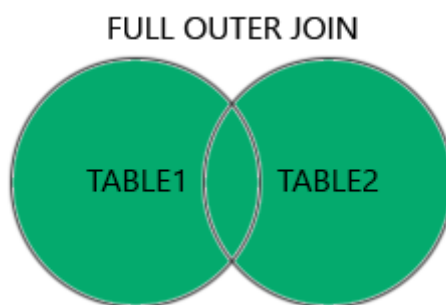
```
INSERT INTO customer(name, email) VALUES ('Peach', 'peach@example.com');  
SELECT * FROM customer LEFT JOIN orders ON orders.cust_id=customer.cust_id;  
SELECT name, SUM(amount) FROM customer LEFT JOIN orders ON  
orders.cust_id=customer.cust_id GROUP BY name;  
SELECT name, IFNULL(SUM(amount),0) AS Total FROM customer LEFT JOIN orders ON  
orders.cust_id=customer.cust_id GROUP BY name;
```

**RIGHT (OUTER) JOIN:** Returns all records from the right table, and the matched records from the left table.



```
SELECT * FROM orders RIGHT JOIN customer ON orders.cust_id=customer.cust_id;
```

**FULL (OUTER) JOIN:** Returns all records when there is a match in either left or right table.



```
SELECT * FROM customer, orders;
```

### INNER JOIN with Group By

```
SELECT name FROM customer INNER JOIN orders ON orders.cust_id=customer.cust_id  
GROUP BY name;
```

```
SELECT name, SUM(amount) FROM customer INNER JOIN orders ON  
orders.cust_id=customer.cust_id GROUP BY name;
```

## ON DELETE CASCADE

It is used in MySQL to delete the rows from the child table automatically, when the rows from the parent table are deleted. For example when a student registers in an online learning platform, then all the details of the student are recorded with their unique number/id. All the courses in these online learning platforms had their own code, title, and name. Students can enroll in any course according to their wishes.

There is no rule that all students must enroll in all courses, or they have to join the course on the same date. A student can enroll in one or more courses. Suppose you delete a row from the "Student" table, now you will also want to delete all rows in the "Enroll" table that references the row in the "Student" table.

```
DROP TABLE orders;
CREATE TABLE orders(
    order_id INT AUTO_INCREMENT PRIMARY KEY,
    date DATE,
    amount DECIMAL(10,2),
    cust_id INT,
    FOREIGN KEY (cust_id) REFERENCES customer(cust_id) ON DELETE CASCADE
);
INSERT INTO orders (date, amount, cust_id) VALUES ('2022-05-22', 500.50, 1), ('Peach',
'peach@example.com'), ('2022-06-18', 600.50, 2), ('2022-05-22', 300.70, 3), ('2022-05-22', 900.70,
3), ('2022-05-22', 400.70, 2);
```