

## Advanced SQL topics

---

### Join

JOIN operations, which are used to combine rows from two or more tables based on related columns. When you perform a 'JOIN' the database engine 1. id's the col to match: it looks at the columns specified in the "ON" clause to determine how tables should be combined. 2. matches rows: for each row in the left table, it looks for matching rows in the right table based on the condition. 3. Combines Rows: if a match is found, combines all rows into a single row in the result set. If no match is found the behavior is based on the type of 'JOIN' operation: - 'inner join': the row is excluded from the result - 'left join': the left row is included with a "NULL" for the right tables - 'right join': the row is included with a "NULL" for the left tables - 'full join': all rows from both tables are included, with "NULL" where there is no match

Different 'JOIN' types exist to allow flexibility in how data is combined:

- 'inner join': focuses only on data that exists in both tables. Used when you need results that are relevant to both sides of the relationship

### Inner Join

- combines rows from two tables where join condition is met
- only returns rows where match in both tables

Ex.

---

```
select customer_name, first_name from customer inner join rep on customer.rep_num = rep.rep_num;
```

### Left Outer Join

- return all rows from the left table
- if no match result is 'NULL' on the side of the right table

Ex.

```
SELECT customer_name, first_name
FROM customer
LEFT OUTER JOIN rep ON customer.rep_num = rep.rep_num;
```

### Right Outer Join

- return all rows from the right table matched to the rows from left table
- if no match result is 'NULL' on the side of the right table

**Ex.**

```
SELECT customer_name, first_name
FROM customer
RIGHT OUTER JOIN rep ON customer.rep_num = rep.rep_num;
```

### Full Outer Join

- returns all rows when there is a match in one of the tables
- if no match result is 'NULL' on the side of the right table

**Ex.**

```
SELECT customer_name, first_name
FROM customer
FULL OUTER JOIN rep ON customer.rep_num = rep.rep_num;
```

### Self Join

- regular join but the table is joined with itself.

**Ex.**

```
SELECT t1.customer_name AS t1_cus, t2.customer_name AS t2_cus, t1.city
FROM customer t1, customer t2
WHERE t1.customer_num < t2.customer_num AND t1.city = t2.city
ORDER BY t1.city;
```

---

### Join vs Group