joins – left outer join

The LEFT JOIN keyword returns all rows from the left table (r_1), with the matching rows in the right table (r_2).
The result is NULL in the right side when there is no match.

SELECT A_1, A_2, A_3, . . . FROM r_1 LEFT [OUTER] JOIN r_2 ON r_1. A_1 = r_2. A_1

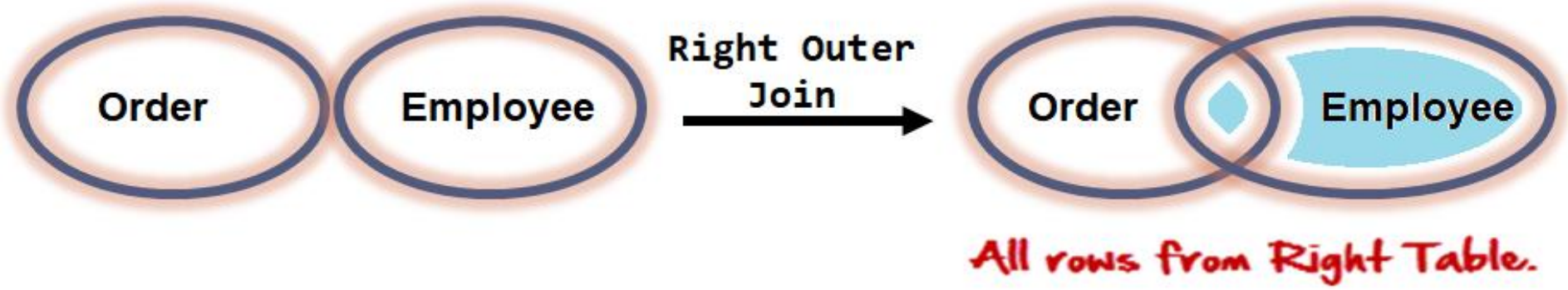
```
SELECT * FROM orders ord LEFT OUTER JOIN employee emp ON emp.id = ord.employeeid;
```

joins – right outer join

The RIGHT JOIN keyword returns all rows from the right table (r_2), with the matching rows in the left table (r_1).
The result is NULL in the left side table when there is no match.

SELECT A_1, A_2, A_3, . . . FROM r_1 RIGHT [OUTER] JOIN r_2 ON r_1. A_1 = r_2. A_1

```
SELECT * FROM orders ord RIGHT OUTER JOIN employee emp ON emp.id = ord.employeeid;
```



TODO

self joins

TODO

joins – self join

A SELF JOIN is a join in which a table is joined with itself (which is also called Unary relationships), especially when the table has a FOREIGN KEY which references its own PRIMARY KEY.

SELECT $r_x.A_1$, $r_x.A_2$, $r_y.A_1$, $r_y.A_2$, . . . FROM r_1 r_x, r_1 r_y WHERE $r_x.A_1 = r_y.A_1$

set operation in sql

Set operators are used to join the results of two (or more) SELECT statements.

Remember:

- The result set column names are taken from the column names of the first SELECT statement.
- SELECT statement should have the same data type. (Not in MySQL)