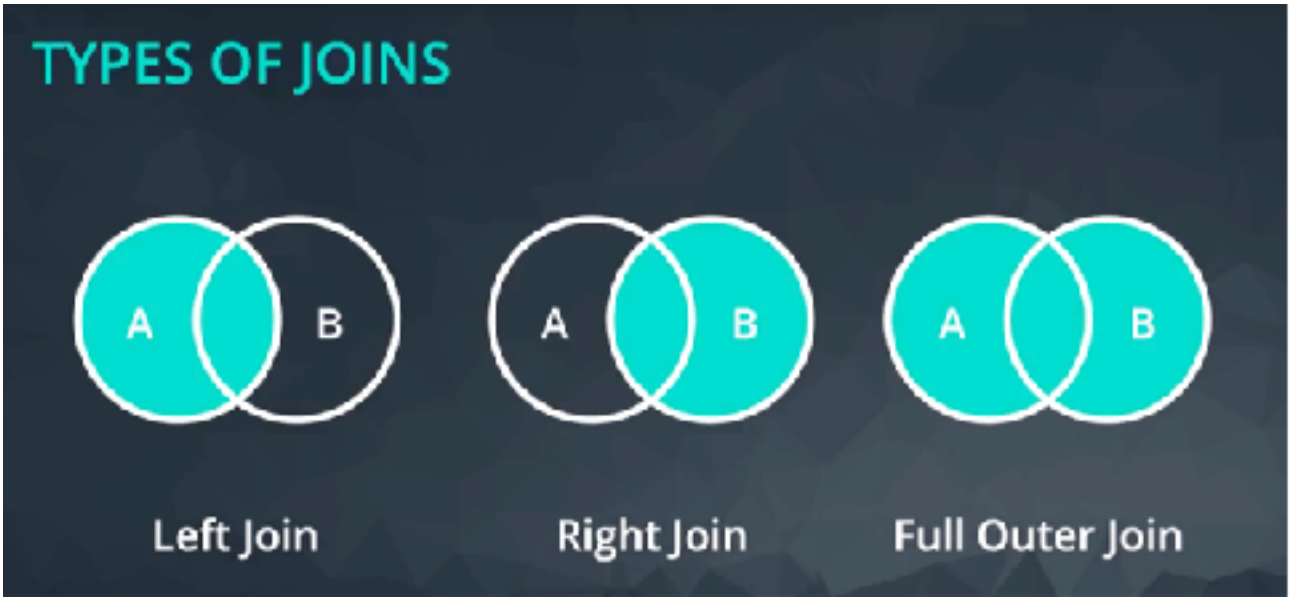
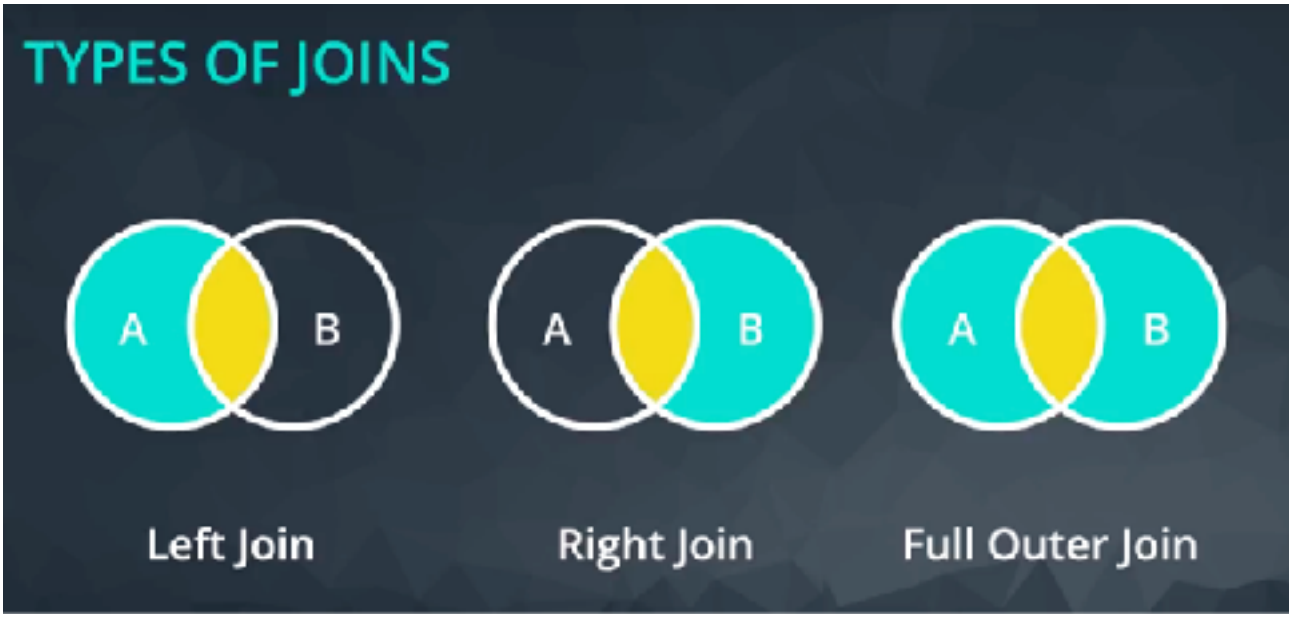
OUTER JOINS

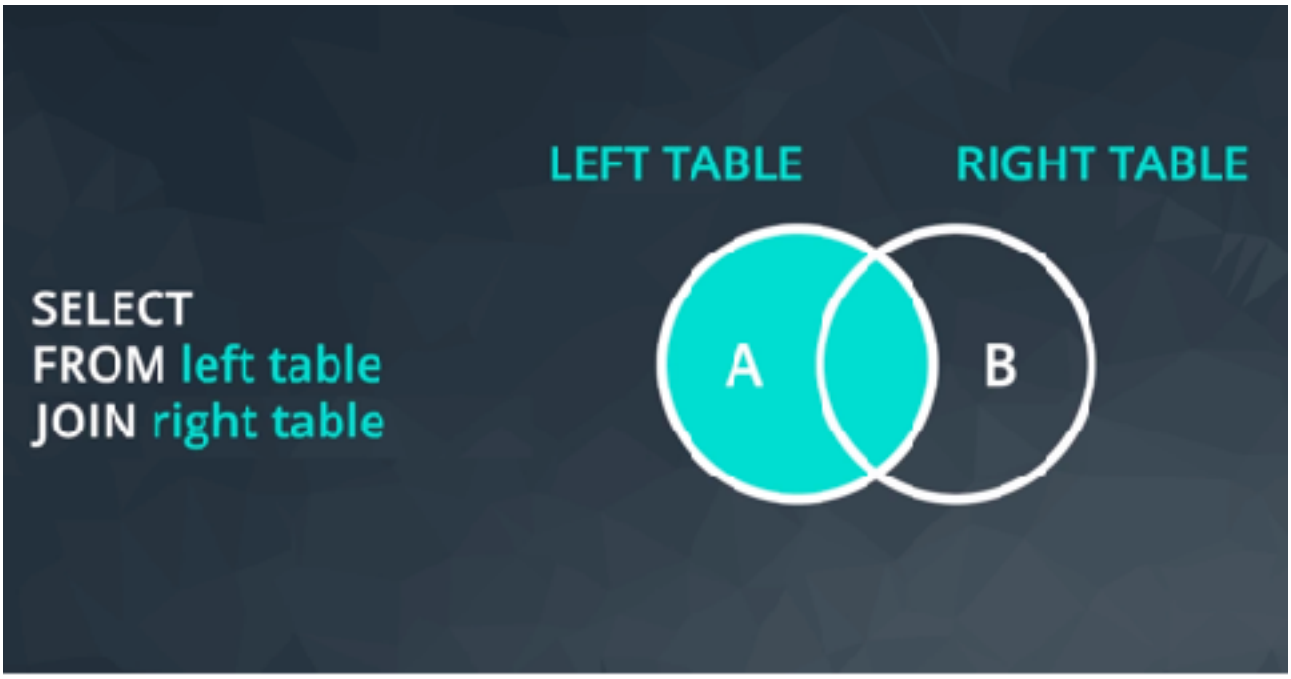


Outer Joins contain all the rows from the Inner Join (with the same query) and some additional rows.

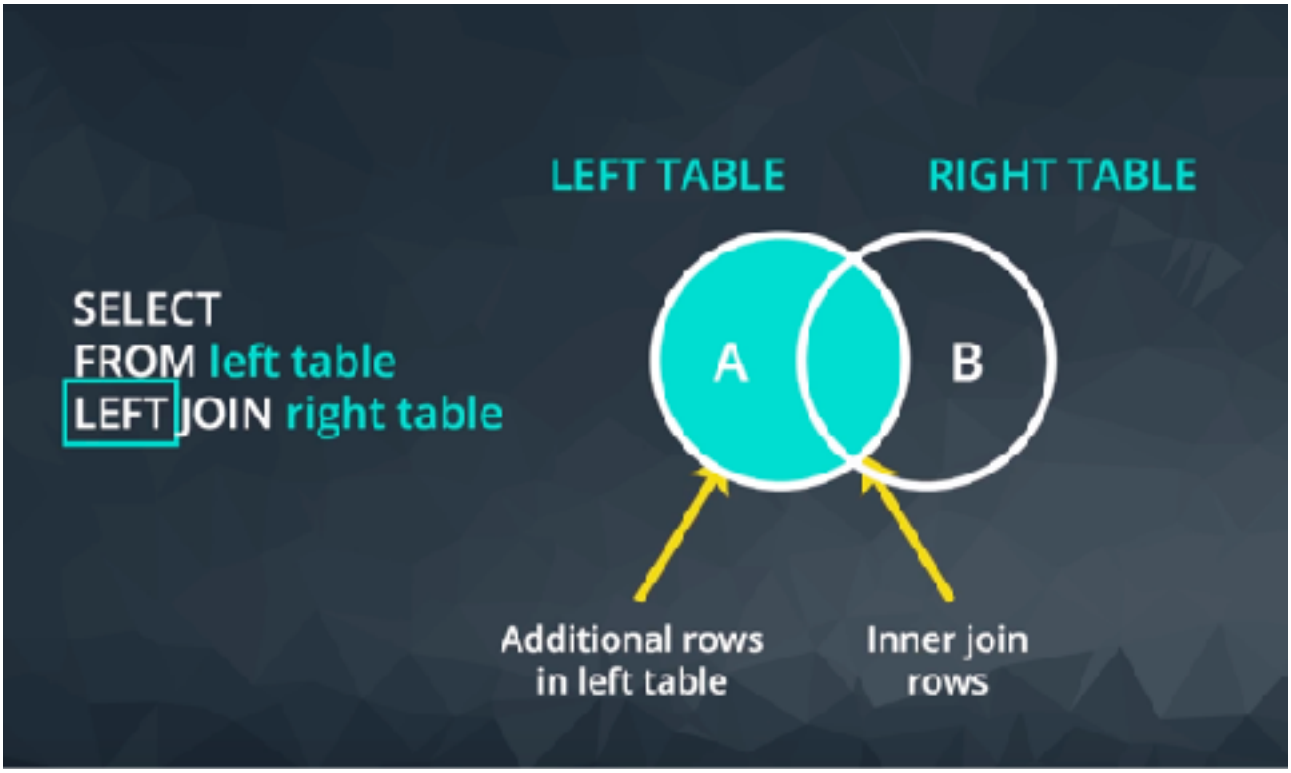


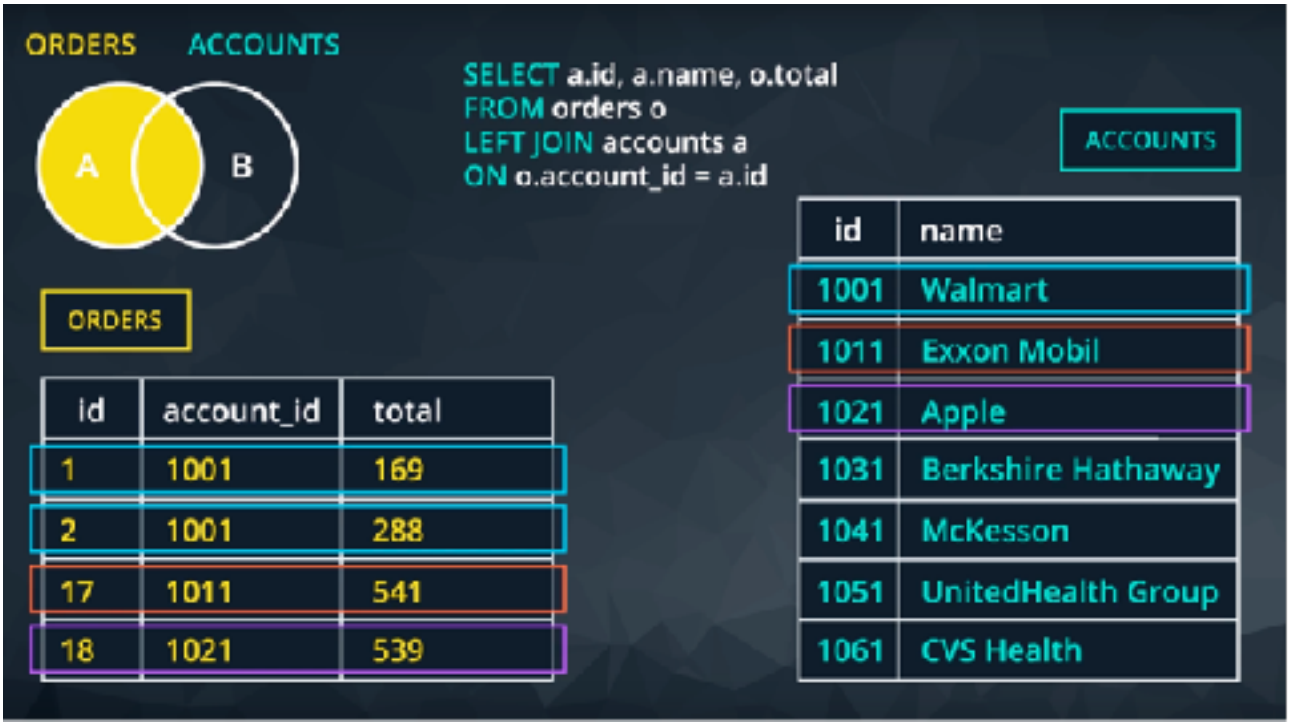
Basic SQL query syntax for an outer join. The table in the FROM clause is considered the left table and the one in JOIN clause is considered as the right table, no matter which outer join you use (left, right or full outer).

NOTE - In inner join, you can interchange the tables in the FROM and JOIN clause.

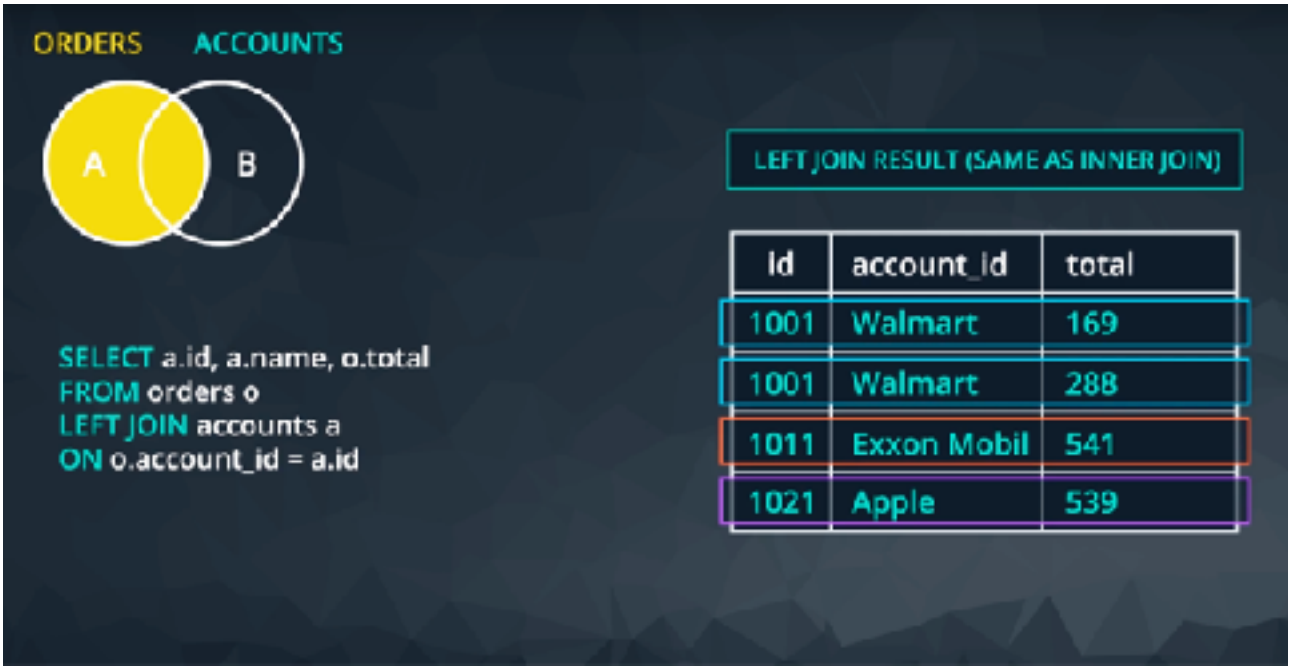


LEFT JOIN

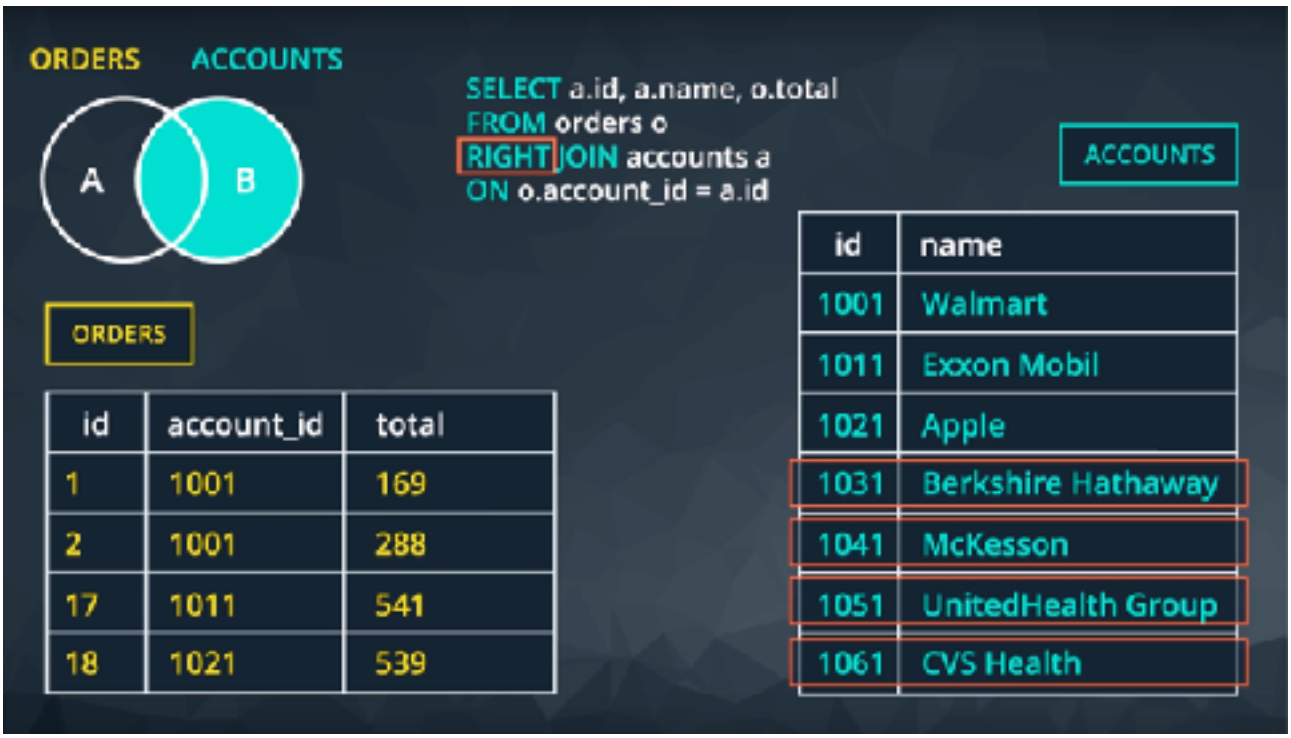




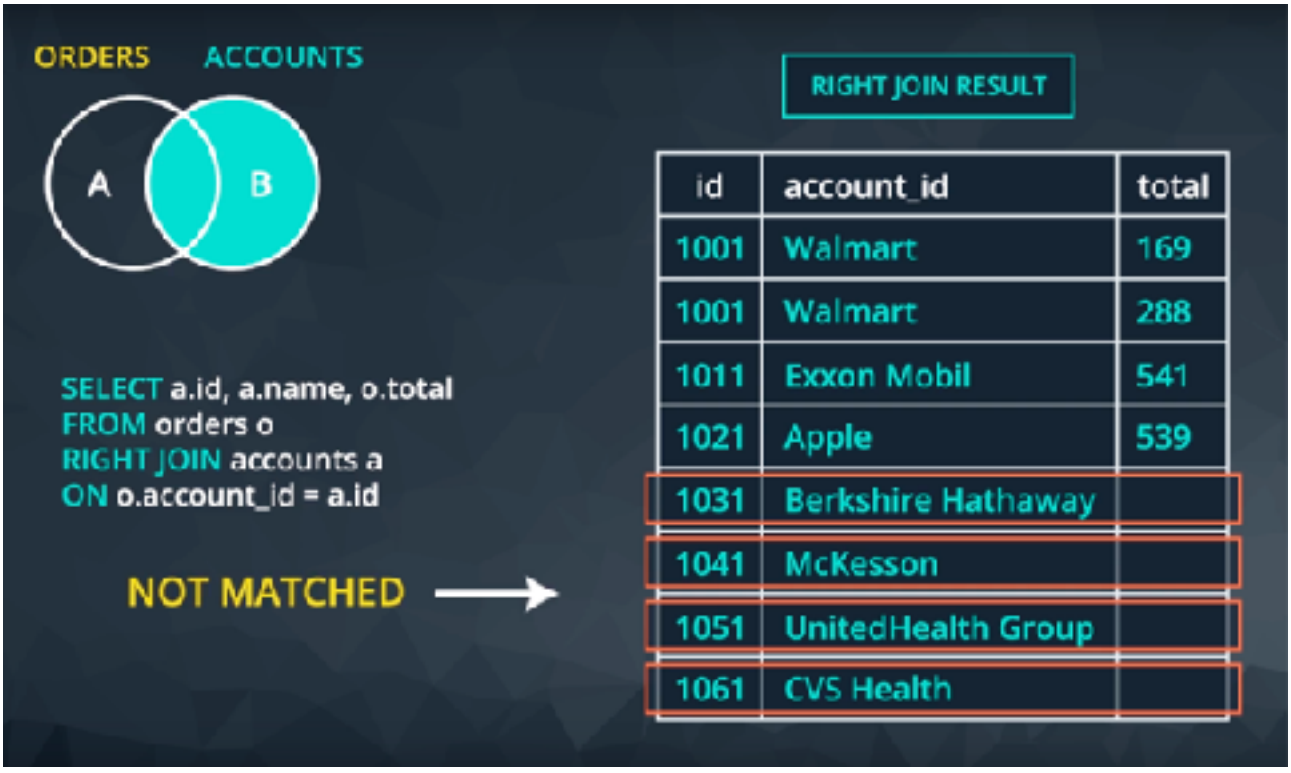
Since there is every row in the left table has a matching row from the right table, this behaves the same as Inner Join.



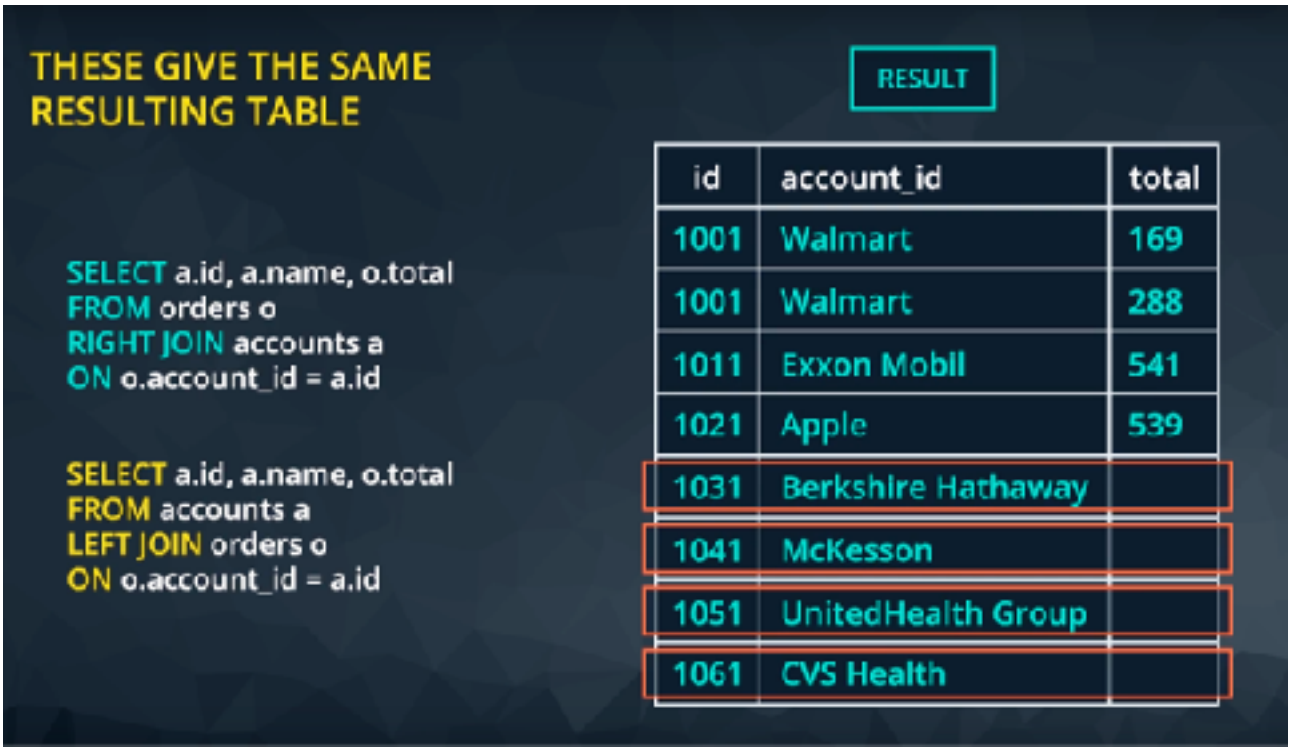
RIGHT JOIN



The rows in the right table that do not have a matching row in the right table are NOT DROPPED. Those are included at the end of the resulting table and the empty cells contain NULL data.



Right Join and Left Join behave the same if tables in FROM clause and JOIN clause are interchanged.



Quick Note

You might see the SQL syntax of

```
LEFT OUTER JOIN
```

OR

```
RIGHT OUTER JOIN
```

These are the exact same commands as the **LEFT JOIN** and **RIGHT JOIN** we learned about in the previous video.

- A **LEFT JOIN** and **RIGHT JOIN** do the same thing if we change the tables that are in the **FROM** and **JOIN** statements.
- A **LEFT JOIN** will **at least** return all the rows that are in an **INNER JOIN**.
- **JOIN** and **INNER JOIN** are the same.
- A **LEFT OUTER JOIN** is the same as **LEFT JOIN**.

OUTER JOINS

The last type of join is a full outer join. This will return the inner join result set, as well as any unmatched rows from either of the two tables being joined.

Again this returns rows that **do not match** one another from the two tables. The use cases for a full outer join are **very rare**.

Similar to the above, you might see the language **FULL OUTER JOIN**, which is the same as **OUTER JOIN**.