Set Operators: Takeaways ⤴

Syntax

- Stacking tables without duplicates:

```
SELECT *
  FROM some_table
 UNION
SELECT *
  FROM another_table;
```

- Stacking tables including duplicates:

```
SELECT *
  FROM some_table
 UNION ALL
SELECT *
  FROM another_table;
```

- Finding rows common to both tables:

```
SELECT *
     FROM some_table
INTERSECT
   SELECT *
     FROM another_table;
```

- Filtering out the common rows from the top table:

```
SELECT *
  FROM some_table
EXCEPT
SELECT *
  FROM another_table;
```

- Selecting columns in a compound select statement:

```
SELECT a_column, another_column
       FROM some_table
SET_OPERATOR
     SELECT a_column, another_column
       FROM another_table;
```

Concepts

- Set operators combine results by stacking rows (vertically) and aligning columns.

- We use set operators to combine `SELECT` statement results.

- We call the combination of set operators with `SELECT` statements a compound select statement.

- Legal compound select statements must conform to some constraints:
  - Each SELECT statement must meet the following criteria:
    - Have the same number of columns
    - The columns should align on the datatype
  - The ORDER BY and LIMIT clauses can only occur after the last SELECT clause.
- The `UNION` operator doesn't keep duplicates. The `UNION ALL` operator does.
- We can use `UNION` and `UNION ALL` to rewrite queries using `OR`, thus boosting performance.

Resources

- [Sets basic operations](#)
- [Venn diagram](#)
- [Compound select statement](#)
- [Why is an `OR` statement slower than `UNION`?](#)