

Topic - SET OPERATORS(UNION, UNION ALL, MINUS, INTERSECT)

1. UNION

It returns a union of two select statements. It is returning unique (distinct) values of them.

union

```
SELECT * FROM table1
```

```
UNION
```

```
SELECT * FROM table2;
```

2. UNION ALL

Similar to UNION just that UNION ALL returns also the duplicated values.

```
SELECT * FROM table1
```

```
UNION
```

```
SELECT * FROM table2;
```

When using UNION and UNION ALL columns in SELECT statements need to match. This would return an error:

```
SELECT column1 FROM table1
```

```
UNION
```

```
SELECT * FROM table2;
```

3. MINUS

MINUS (also known as EXCEPT) returns the difference between the first and second SELECT statement. It is the one where we need to be careful which statement will be put first, cause we will get only those results that are in the first SELECT statement and not in the second.

minus

```
SELECT * FROM table1
```

```
MINUS
```

```
SELECT * FROM table2;
```

4. INTERSECT

Database CS 33 Unit III Topic - Set operation

INTERSECT is opposite from MINUS as it returns us the results that are both to be found in first and second SELECT statement.

intersect

```
SELECT * FROM table1
```

INTERSECT

```
SELECT * FROM table2;
```

EXAMPLE

There is an interesting example that we use SET OPERATORS to compare whether two tables have identical values, testing symmetric difference. If result of this entire query returns no rows, it will mean that they are identical.

```
(SELECT * FROM table1
```

MINUS

```
SELECT * FROM table2)
```

UNION

```
(SELECT * FROM table2
```

MINUS

```
SELECT * FROM table1);
```

COMPOUND QUERIES

Each query in a compound query will project its own list of selected columns. These lists must have the same number of elements, be nominated in the same sequence, and be of broadly similar data type. They do not have to have the same names (or column aliases), nor do they need to come from the same tables (or subqueries). If the column names (or aliases) are different, the result set of the compound query will have columns named as they were in the first query.

The corresponding columns in the queries that make up a compound query must be of the same data type group. The result set of the compound query will have columns with the higher level of precision.

UNION, MINUS, and INTERSECT will always combine the results sets of the input queries, then sort the results to remove duplicate rows. If the sort order (which is ascending, based on the order in which the columns happen to

Database CS 33 Unit III Topic - Set operation

appear in the select lists) is not the order you want, it is possible to put a single ORDER BY clause at the end of the compound query. It is not possible to use ORDER BY in any of the queries that make up the whole compound query, as this would disrupt the sorting that is necessary to remove duplicates. UNION ALL is the exception to the sorting-no-duplicates rule: the result sets of the two input queries will be concatenated to form the result of the compound query.

It is not possible to use ORDER BY in the individual queries that make a compound query. An ORDER BY clause can be appended to the end of a compound query.