Multiple Group Summary Statistics: Takeaways

Syntax

- Counting rows grouped by several columns:

```
SELECT group_column_1, group_column_2, COUNT(*) AS num_row
   FROM table
  GROUP BY group_column_1, group_column_2;
```

- Computing statistics grouped by several columns:

```
SELECT group_column_1, group_column_2, AVG(col) AS avg_col, MIN(col) AS min_col, MAX(col)
AS min_col
   FROM table
  GROUP BY group_column_1, group_column_2;
```

- Adding conditions on an aggregated column:

```
SELECT group_column_1, group_column_2, AVG(col) AS avg_col, MIN(col) AS min_col, MAX(col)
AS min_col
   FROM table
  GROUP BY group_column_1, group_column_2
 HAVING condition_on_aggregated_columns;
```

- Combining WHERE and HAVING clauses:

```
SELECT group_column_1, group_column_2, AVG(col) AS avg_col, MIN(col) AS min_col, MAX(col)
AS min_col
   FROM table
  WHERE conditions
  GROUP BY group_column_1, group_column_2
 HAVING condition_on_aggregated_columns;
```

Concepts

- Aggregate functions allow operations combining several rows over groups.
- We can aggregate rows grouped by several columns.
- With the new clause, the new SQL order of clauses is as follows:

```
SELECT > FROM > WHERE > GROUP BY > HAVING > ORDER BY > LIMIT
```

- With the new clause, the new SQL execution order is as follows:

```
FROM > WHERE > GROUP BY > HAVING > SELECT > ORDER BY > LIMIT
```

Resources

- SQL aggregate functions