6-SQL Aggregation Functions

Author: Vincent Lau

Learning Objectives

- Understand the basic syntax and result of using Aggregation Functions.

Introduction

Aggregation operations in SQL involve performing calculations on groups of rows to produce summary results. These operations are commonly used to obtain insights from data, such as calculating totals, averages, counts, and more.

Aggregation Functions

COUNT

The `COUNT` function is used to count the number of rows that match a specific condition.

```sql
SELECT COUNT(*) AS total_orders
FROM orders;
```

SUM

The `SUM` function calculates the sum of a numeric column's values.

```sql
SELECT SUM(total_amount) AS total_sales
FROM orders;
```

AVG

The `AVG` function calculates the average value of a numeric column.

```sql
SELECT AVG(salary) AS average_salary
FROM employees;
```

MIN & MAX

The `MIN` function retrieves the minimum value from a column, while the `MAX` function retrieves the maximum value.

```sql
SELECT MIN(unit_price) AS min_price,
MAX(unit_price) AS max_price
FROM products;
```

Main usage of Aggregation Functions

Let's have a quick check before going through the details in the next chapter.

GROUP BY

The `GROUP BY` clause is used to group rows based on one or more columns and perform aggregate calculations on each group.

```sql
SELECT department_id,
AVG(salary) AS average_salary
FROM employees
```

```
4 GROUP BY department_id;
```

HAVING

The `HAVING` clause is used with the `GROUP BY` clause to filter groups based on aggregate calculations.

```
1 SELECT department_id,
2 AVG(salary) AS average_salary
3 FROM employees
4 GROUP BY department_id
5 HAVING AVG(salary) > 50000;
```

Combining Aggregation Functions

You can combine multiple aggregation functions in a single query

```
1 SELECT department_id,
2 COUNT(*) AS total_employees,
3 AVG(salary) AS average_salary
4 FROM employees
5 GROUP BY department_id;
```

Aggregation with Joins

Aggregation can be used with join operations to retrieve summarized information from related tables.

For details, refer to JOIN chapter.

```
1 SELECT c.country_name,
2 COUNT(o.order_id) AS total_orders
3 FROM countries c
4 INNER JOIN customers cust ON c.country_id = cust.country_id
5 INNER JOIN orders o ON cust.customer_id = o.customer_id
6 GROUP BY c.country_name;
```

Aggregation operations are essential for summarizing and analyzing data. By using these functions along with the `GROUP BY` and `HAVING` clauses, you can obtain valuable insights and make informed decisions based on summarized data.