LU04d - SQL-DQL: Selects with Aggregate Functions

SQL (Structured Query Language) provides powerful tools to perform calculations and data analysis on database records. Among these tools, aggregate functions play a crucial role in summarizing data by performing operations on multiple rows and returning a single result. This document focuses on five commonly used SQL aggregate functions: **MIN, MAX, COUNT, SUM, and AVG**, along with practical examples to illustrate their usage.

1. MIN Function

The **MIN** function returns the smallest value in a specified column. This is particularly useful when you need to find the lowest value in a dataset, such as the minimum price of a product or the earliest date in a list of events.

Example:

```
SELECT MIN(price) AS LowestPrice
FROM products;
```

In this example, the query returns the lowest price from the „products“ table.

2. MAX Function

The **MAX** function works oppositely to MIN, returning the largest value in a specified column. It’s commonly used to find the highest values, such as the maximum salary in a company or the latest date in a schedule.

Example:

```
SELECT MAX(salary) AS HighestSalary
FROM employees;
```

Here, the query retrieves the highest salary from the „employees“ table.

3. COUNT Function

The **COUNT** function counts the number of rows that match a specified condition or simply counts all rows in a column. This is useful for determining the size of datasets, such as counting the number of orders placed by a customer or the total number of employees in a department.

Example:

```
SELECT COUNT(*) AS NumberOfOrders
```

```
FROM orders
WHERE customer_id = 123;
```

This query returns the total number of orders placed by a customer with customer_id 123.

4. SUM Function

The **SUM** function adds up all the values in a specified numeric column. It's often used for calculating totals, such as the total sales revenue or the sum of all expenses in a financial report.

Example:

```
SELECT SUM(quantity * price) AS TotalRevenue
FROM sales;
```

In this example, the query calculates the total revenue by multiplying the quantity of items sold by their price and summing the results.

5. AVG (Average) Function

The **AVG** function calculates the average value of a numeric column. This is useful for finding the mean value in a dataset, such as the average test score of students or the average price of products in a store.

Example:

```
SELECT AVG(price) AS AveragePrice
FROM products;
```

Here, the query returns the average price of all products in the „products“ table.

6. Combining Aggregate Functions with GROUP BY

While each of these functions is powerful on its own, their true potential is unlocked when combined with the GROUP BY clause. This clause allows you to apply aggregate functions to specific groups of data, such as calculating the total sales per region or the average salary per department.

Example:

```
SELECT department_id, AVG(salary) AS AverageSalary
FROM employees
GROUP BY department_id;
```

In this example, the query calculates the average salary for each department in the „employees“ table.

Vocabulary

English	German
oppositely	gegenüberliegend
to perform	ausführen, durchführen
to aggregate	vermengen, auswerten
commonly	allgemein
dataset	Datensatz
expense	Kosten, Ausgaben
revenue	Ertrag, Einnahmen
mean value	Durschnittswert
to apply	anwenden
