

SQL Aggregate Functions

Summary: in this tutorial, you will learn about the SQL aggregate functions including `AVG()`, `COUNT()`, `MIN()`, `MAX()`, and `SUM()`.

An SQL aggregate function calculates on a set of values and returns a single value. For example, the average function (`AVG`) takes a list of values and returns the average.

Because an aggregate function operates on a set of values, it is often used with the `GROUP BY` clause of the `SELECT` statement. The `GROUP BY` clause divides the result set into groups of values and the aggregate function returns a single value for each group.

The following illustrates how the aggregate function is used with the `GROUP BY` clause:

```
SELECT c1, aggregate_function(c2)
FROM table
GROUP BY c1;
```

Code language: SQL (Structured Query Language) (sql)

The following are the commonly used SQL aggregate functions:

- `AVG()` – returns the average of a set.
- `COUNT()` – returns the number of items in a set.
- `MAX()` – returns the maximum value in a set.
- `MIN()` – returns the minimum value in a set
- `SUM()` – returns the sum of all or distinct values in a set

Except for the `COUNT()` function, SQL aggregate functions ignore null.

You can use aggregate functions as expressions only in the following:

- The select list of a `SELECT` statement, either a subquery or an outer query.
- A `HAVING` clause

`AVG`

The `AVG()` function returns the average values in a set. The following illustrates the syntax of the `AVG()` function:

```
AVG( ALL | DISTINCT)
```

Code language: SQL (Structured Query Language) (sql)

The ALL keyword instructs the AVG() function to calculate the average of all values while the DISTINCT keyword forces the function to operate on distinct values only. By default, the ALL option is used.

The following example shows how to use the AVG() function to calculate the average salary of each department:

```
SELECT
  department_name, ROUND(AVG(salary), 0) avg_salary
FROM
  employees
  INNER JOIN
  departments USING (department_id)
GROUP BY department_name
ORDER BY department_name;
```

Code language: SQL (Structured Query Language) (sql)

	department_name	avg_salary
▶	Accounting	10150
	Administration	4400
	Executive	19333
	Finance	8600
	Human Resources	6500
	IT	5760
	Marketing	9500
	Public Relations	10000
	Purchasing	4150
	Sales	9617
	Shipping	5886

MIN

The MIN() function returns the minimum value of a set. The following illustrates the syntax of the MIN() function:

```
MIN(column | expression)
```

Code language: SQL (Structured Query Language) (sql)

For example, the following statement returns the minimum salary of the employees in each department:

```
SELECT
  department_name, MIN(salary) min_salary
FROM
```

```
employees
INNER JOIN
departments USING (department_id)
GROUP BY department_name
ORDER BY department_name;
```

Code language: SQL (Structured Query Language) (sql)

	department_name	min_salary
▶	Accounting	8300.00
	Administration	4400.00
	Executive	17000.00
	Finance	6900.00
	Human Resources	6500.00
	IT	4200.00
	Marketing	6000.00
	Public Relations	10000.00
	Purchasing	2500.00
	Sales	6200.00
	Shipping	2700.00

MAX

The `MAX()` function returns the maximum value of a set. The `MAX()` function has the following syntax:

```
MAX(column | expression)
```

Code language: SQL (Structured Query Language) (sql)

For example, the following statement returns the highest salary of employees in each department:

```
SELECT
department_name, MAX(salary) highest_salary
FROM
employees
INNER JOIN
departments USING (department_id)
GROUP BY department_name
ORDER BY department_name;
```

Code language: SQL (Structured Query Language) (sql)

	department_name	highest_salary
▶	Accounting	12000.00
	Administration	4400.00
	Executive	24000.00
	Finance	12000.00
	Human Resources	6500.00
	IT	9000.00
	Marketing	13000.00
	Public Relations	10000.00
	Purchasing	11000.00
	Sales	14000.00
	Shipping	8200.00

COUNT

The `COUNT()` function returns the number of items in a set. The following shows the syntax of the `COUNT()` function:

```
COUNT ( [ALL | DISTINCT] column | expression | *)
```

Code language: SQL (Structured Query Language) (sql)

For example, the following example uses the `COUNT(*)` function to return the headcount of each department:

```
SELECT
  department_name, COUNT(*) headcount
FROM
  employees
  INNER JOIN
  departments USING (department_id)
GROUP BY department_name
ORDER BY department_name;
```

Code language: SQL (Structured Query Language) (sql)

	department_name	headcount
▶	Accounting	2
	Administration	1
	Executive	3
	Finance	6
	Human Resources	1
	IT	5
	Marketing	2
	Public Relations	1
	Purchasing	6
	Sales	6
	Shipping	7

SUM

The `SUM()` function returns the sum of all values. The following illustrates the syntax of the `SUM()` function:

SUM(ALL | DISTINCT column)

Code language: SQL (Structured Query Language) (sql)

For example, the following statement returns the total salary of all employees in each department:

SELECT

department_id, **SUM(salary)**

FROM

employees

GROUP BY department_id;

Code language: SQL (Structured Query Language) (sql)

	department_name	SUM(salary)
▶	Accounting	20300.00
	Administration	4400.00
	Executive	58000.00
	Finance	51600.00
	Human Resources	6500.00
	IT	28800.00
	Marketing	19000.00
	Public Relations	10000.00
	Purchasing	24900.00
	Sales	57700.00
	Shipping	41200.00

In this tutorial, you have learned the most commonly used SQL aggregate functions including `AVG()`, `COUNT()`, `MIN()`, `MAX()`, and `SUM()` functions