

SQL: Structured Query Language Aggregate Operations

Aggregate Operations

Aggregate database operations are operations that return individual values (*aggregates*) instead of tuples. These aggregates can be used in the SELECT clause of the SELECT statement.

There are five aggregate operations in SQL¹:

Aggregate Operation	Explanation
COUNT()	returns the number of rows
AVG()	returns the average value in the column
MIN()	returns the minimum value in the column
MAX()	returns the maximum value in the column
SUM()	returns the sum of the values in the column

NOTE: it does not make sense to have an aggregate operation and normal attribute in the same select-list. An aggregate operation reduces many tuples to a single value, it is undefined which tuple the attribute will chosen from.

AVG,SUM,MAX,MIN. For the AVG, SUM, MAX and MIN operations the syntax is:

```
AVG(<ColumnName>)  
SUM(<ColumnName>)  
MAX(<ColumnName>)  
MIN(<ColumnName>)
```

Here, <ColumnName> is expected be of a numeric type. E.g.:

```
SELECT AVG(Salary) FROM Employee;
```

returns the average of the Salary field in a relation Employee(ID, Name, Salary, Dept).

COUNT. COUNT has more interesting syntax. The two traditional forms of the COUNT() operator are

```
COUNT(*)  
COUNT([DISTINCT] <ColumnName>)
```

The COUNT(*) form counts the number of tuples returned by the rest of the query.

```
SELECT COUNT(*) FROM Employee;
```

¹ Most SQL servers will actually support more operations (like standard deviation or bitwise AND/OR). MySQL aggregate operations: <http://dev.mysql.com/doc/refman/5.0/en/group-by-functions.html>.

returns the number of employee records.

Similar result can be achieved by replacing * with an column name:

```
SELECT COUNT(Name) FROM Employee;
```

returns the same result as the previous query.

DISTINCT qualifier allows us to count only unique values found in the column:

```
SELECT COUNT(DISTINCT Salary) FROM Employee;
```

returns the number of different salary levels recorded in the Employee table.