

文章

[姚鑫](#) · 十二月 31, 2021 阅读大约需 4 分钟

第十一章 SQL聚合函数 SUM

第十一章 SQL聚合函数 SUM

返回指定列值之和的聚合函数。

大纲

`SUM([ALL | DISTINCT [BY(col-list)]] expression [%FOREACH(col-list)] [%AFTERHAVING])`

参数

- ALL - 可选-指定SUM返回表达式中所有值的和。
如果没有指定关键字，这是默认值。
- DISTINCT - 可选-一个DISTINCT子句，指定SUM返回表达式中不同(唯一)值的和。
DISTINCT可以指定BY(col-list)子句，其中col-list可以是单个字段，也可以是用逗号分隔的字段列表。
- expression - 任何有效的表达式。
通常是包含要求和的数据值的列的名称。
- %FOREACH(col-list) - 可选-列名或以逗号分隔的列名列表。
- %AFTERHAVING - 可选-应用在HAVING子句中的条件。

SUM返回与expression相同的数据类型，但有如下异常:TINYINT、SMALLINT和INTEGER返回的数据类型都是INTEGER。

注意:SUM可以指定为聚合函数，也可以指定为窗口函数。

本参考页面描述了SUM作为聚合函数的使用。

SUM作为一个窗口函数在窗口函数概述中描述。

描述

SUM聚合函数返回表达式值的和。

通常，表达式是查询返回的多行中字段的名称(或包含一个或多个字段名称的表达式)。

SUM可以在引用表或视图的SELECT查询或子查询中使用。

SUM可以在SELECT列表或HAVING子句中与普通字段值一起出现。

SUM不能在WHERE子句中使用。

SUM不能在JOIN的ON子句中使用，除非SELECT是子查询。

和所有聚合函数一样，SUM可以接受一个可选的DISTINCT子句。

SUM(DISTINCT col1)只汇总那些不同(唯一)的col1字段值。

SUM(DISTINCT BY(col2) col1)只汇总col2值不同(唯一)的记录中的col1字段值。

但是请注意，不同的col2值可能包含一个单独的NULL值。

数据值

对于数据类型为INT、SMALLINT或TINYINT的表达式，SUM返回的数据类型为INTEGER。
对于数据类型为BIGINT的表达式，SUM返回数据类型BIGINT。
对于数据类型为DOUBLE的表达式，SUM返回数据类型为DOUBLE的表达式。
对于所有其他数字数据类型，SUM返回数据类型numeric。

SUM返回精度为18的值。

返回值的尺度与表达式的尺度相同，但有以下例外。

如果expression是一个数据类型为VARCHAR或VARBINARY的数值，则返回值的尺度为8。

默认情况下，聚合函数使用逻辑(内部)数据值，而不是显示值。

SUM通常应用于具有数值的字段或表达式。

因为只执行最小的类型检查，所以有可能(尽管很少有意义)对非数字字段调用它。

SUM计算包括空字符串("")在内的非数值值为0(0)。如果expression是数据类型VARCHAR，则返回到ODBC或JDBC的值是数据类型DOUBLE。

在派生SUM聚合函数值时，数据字段中的NULL值将被忽略。

如果查询没有返回任何行，或者返回的所有行的数据字段值为NULL, SUM返回NULL。

优化

SUM计算的SQL优化可以使用一个位片索引，如果这个索引是为字段定义的。

当前事务期间所做的更改

与所有聚合函数一样，SUM总是返回数据的当前状态，包括未提交的更改，而不考虑当前事务的隔离级别。

示例

在下面的例子中，美元符号(\$)连接到薪金数额。

下面的查询返回示例中所有员工的工资之和。

雇员数据库:

```
SELECT '$' || SUM(Salary) AS Total_Payroll
FROM Sample.Employee
```

下面的查询使用

%AFTERHAVING必须返回所有工资的总和和每个州超过\$80,000的工资的总和，其中至少有一个人的工资为>\$80,000:

```
SELECT Home_State,
       '$' || SUM(Salary) AS Total_Payroll,
       '$' || SUM(Salary %AFTERHAVING) AS Exec_Payroll
FROM Sample.Employee
GROUP BY Home_State
HAVING Salary > 80000
ORDER BY Home_State
```

第十一章 SQL聚合函数 SUM

Published on InterSystems Developer Community (<https://community.intersystems.com>)

下面的查询返回示例中每个职位工资的总和和平均值。
雇员数据库:

```
SELECT Title,  
       '$' || SUM(Salary) AS Total,  
       '$' || AVG(Salary) AS Average  
FROM Sample.Employee  
GROUP BY Title  
ORDER BY Average
```

下面的查询显示了与算术表达式一起使用的SUM。
对于示例中的每个职位名称。
员工数据库, 返回当前工资和工资增加10%的工资之和:

```
SELECT Title,  
       '$' || SUM(Salary) AS BeforeRaises,  
       '$' || SUM(Salary * 1.1) AS AfterRaises  
FROM Sample.Employee  
GROUP BY Title  
ORDER BY Title
```

下面的查询显示了使用CASE语句与逻辑表达式一起使用的SUM。
它计算所有受薪员工, 并使用SUM计算所有年薪为9万美元或以上的受薪员工。

```
SELECT COUNT(Salary) As AllPaid,  
       SUM(CASE WHEN (Salary >= 90000)  
            THEN 1 ELSE 0 END) As TopPaid  
FROM Sample.Employee
```

[#SQL #Caché](#)

源

URL:

<https://cn.community.intersystems.com/post/%E7%AC%AC%E5%8D%81%E4%B8%80%E7%AB%A0-sql%E8%81%9A%E5%90%88%E5%87%BD%E6%95%B0-sum>