

文章

[姚鑫](#) · 十二月 24, 2021 阅读大约需分钟

## 第四章 SQL聚合函数 COUNT(二)

## 第四章 SQL聚合函数 COUNT(二)

### 权限

要使用COUNT(\*), 必须对指定的表具有表级别的SELECT权限。

要使用COUNT(column-

name), 必须对指定的列具有列级别的SELECT权限, 或者对指定的表具有表级的SELECT权限。

可以通过调用%CHECKPRIV命令来确定当前用户是否具有SELECT权限。

可以通过调用\$SYSTEM.SQL.Security.CheckPrivilege()方法来确定指定的用户是否具有表级SELECT权限。

### 性能

为了获得最优的COUNT性能, 你应该按照以下方式定义索引:

- 对于COUNT(\*), 如果需定义位图扩展索引。  
这个索引可能是在创建表时自动定义的。
- 对于COUNT(fieldname), 为指定字段定义一个位片索引。  
查询计划优化器COUNT(fieldname)自动应用默认的排序规则到fieldname。

### 未提交事务所做的更改

与所有聚合函数一样, COUNT总是返回数据的当前状态, 包括未提交的更改, 而不管当前事务的隔离级别如何, 如所示:

- COUNT插入和更新的记录的计数, 即使这些更改尚未提交并可能回滚。
- COUNT不计算已删除的记录, 即使这些删除还没有提交并且可以回滚。

### 示例

下面的示例返回Sample.Person中的总行数:

```
SELECT COUNT(*) AS TotalPersons
FROM Sample.Person
```

下面的示例返回Sample.Person中的姓名、配偶和最喜欢的颜色的计数。这些计数不同, 因为有些配偶和FavoriteColors字段为NULL; Count不计算NULL:

```
SELECT COUNT(Name) AS People,
```

```
COUNT(Spouse) AS PeopleWithSpouses,  
COUNT(FavoriteColors) AS PeopleWithColorPref  
FROM Sample.Person
```

下面的示例返回三个值：总行数、FavoriteColors字段中的非空值总数和FavoriteColors字段中不同的非空值的总数：

```
SELECT COUNT(*) As TotalPersons,  
COUNT(FavoriteColors) AS WithColorPref,  
COUNT(DISTINCT FavoriteColors) AS ColorPrefs  
FROM Sample.Person
```

下面的示例使用Count DISTINCT返回Sample.Person中不同FavoriteColors值的计数。(FavoriteColors包含3个数据值和1个Null。)。此示例还使用DISTINCT子句为每个不同的FavoriteColors值返回一行。行计数比COUNT(DISTINCT FavoriteColors)计数大1，因为DISTINCT返回单个NULL的行作为DISTINCT值，但COUNT DISTINCT不计算NULL。COUNT(DISTINCT BY(FavoriteColors)%ID)值与行计数相同，因为BY子句将单个NULL计数为DISTINCT值：

```
SELECT DISTINCT FavoriteColors,  
COUNT(DISTINCT FavoriteColors) AS DistColors,  
COUNT(DISTINCT BY(FavoriteColors) %ID) AS DistColorPeople  
FROM Sample.Person
```

下面的例子使用GROUP BY为每个FavoriteColors值返回一行，包括一行NULL。与每行关联的是两个计数。第一个用FavoriteColors选项计算数字或记录；不计算空记录。第二个计算与每个favoritecolor选择关联的名称数量；由于Name不包含空值，因此可以使用空值来计算有利颜色的数量：

```
SELECT FavoriteColors,  
COUNT(FavoriteColors) AS ColorPreference,  
COUNT(Name) AS People  
FROM Sample.Person  
GROUP BY FavoriteColors
```

以下示例返回Sample.Person中每个Home\_State值的Person记录计数：

```
SELECT Home_State, COUNT(*) AS AllPersons  
FROM Sample.Person  
GROUP BY Home_State
```

以下示例使用%AFTERHAVING返回至少有一个人超过65岁的每个州的个人记录计数和超过65岁的人数计数：

```
SELECT Home_State, COUNT(Name) AS AllPersons,  
COUNT(Name %AFTERHAVING) AS Seniors  
FROM Sample.Person  
GROUP BY Home_State  
HAVING Age > 65  
ORDER BY Home_State
```

以下示例同时使用%FOREACH和%AFTERHAVING关键字。它为姓名以“A”、“M”或“W”(HAVING子句和GROUP

BY子句)开头的人员所在的州返回一行。每个状态行包含列值:

- Count(Name): 数据库中所有人员的计数。(此数字对于所有行都是相同的。)
- COUNT(Name %FOREACH(Home\_State)): 该州所有人的计数。
- COUNT(Name %AFTERHAVING): 数据库中符合HAVING子句条件的所有人员的计数。(此数字对于所有行都是相同的。)
- COUNT(Name %FOREACH(Home\_State) %AFTERHAVING): 该州符合HAVING子句条件的所有人员的计数。

```
SELECT Home_State,
       COUNT(Name) AS NameCount,
       COUNT(Name %FOREACH(Home_State)) AS StateNameCount,
       COUNT(Name %AFTERHAVING) AS NameCountHaving,
       COUNT(Name %FOREACH(Home_State) %AFTERHAVING) AS StateNameCountHaving
FROM Sample.Person
GROUP BY Home_State
HAVING Name LIKE 'A%' OR Name LIKE 'M%' OR Name LIKE 'W%'
ORDER BY Home_State
```

下面的示例显示具有串联表达式的Count, 它使用连接运算符(||)返回FavoriteColors字段中非空值的总数, 以及FavoriteColors中与其两个字段连接的非空值的总数:

```
SELECT COUNT(FavoriteColors) AS Color,
       COUNT(FavoriteColors||Home_State) AS ColorState,
       COUNT(FavoriteColors||Spouse) AS ColorSpouse
FROM Sample.Person
```

当两个字段连接在一起时, COUNT只计算其中两个字段都没有空值的那些行。因为Sample.Person中的每一行都有一个非空的Home\_State值, 所以串联FavoriteColors||Home\_State返回与FavoriteColors相同的计数。由于Sample.Person中的某些行的配偶值为NULL, 因此串联FavoriteColors||SPOUSE将返回FavoriteColors和配偶的值均为NULL值的行数。

#### [#SQL #Caché](#)

源 URL: <https://cn.community.intersystems.com/post/%E7%AC%AC%E5%9B%9B%E7%AB%A0-sql%E8%81%9A%E5%90%88%E5%87%BD%E6%95%B0-count%E5%BC%88%E4%BA%8C%E5%BC%89>