

文章

姚鑫 · 九月 2, 2021 阅读大约需 4 分钟

第四章 SQL命令 ALTER VIEW

第四章 SQL命令 ALTER VIEW

修改视图。

大纲

```
ALTER VIEW view-name [(column-commalist)] AS query [WITH READ ONLY]

ALTER VIEW view-name [(column-commalist)] AS query [WITH [level] CHECK OPTION]
```

参数

参数	描述
view-name	被修改的视图，它具有与表名相同的命名规则。视图名可以是限定的(schema.viewname)，也可以是非限定的(view name)。非限定视图名接受默认模式名。
column-commalist	可选-组成视图的列名。如果这里没有指定，可以在查询中指定列名，如下所示。
query	作为视图基础的结果集(来自查询)。
WITH READ ONLY	可选-指定不能通过此视图对视图所基于的表执行插入、更新或删除操作。默认情况下，允许通过视图进行这些操作，约束如下所示。
WITH level CHECK OPTION	可选-指定如何通过此视图对视图所基于的表执行插入、更新或删除操作。级别可以是关键字LOCAL或CASCADED。如果没有指定级别，则WITH CHECK OPTION默认为级联。

描述

ALTER VIEW命令用于修改视图。
视图基于由SELECT语句或由两个或多个SELECT语句组成的UNION组成的查询的结果集。

要确定当前名称空间中是否存在指定的视图，请使用\$SYSTEM.SQL.Schema.ViewExists()方法。

可选的column-commalist指定视图中包含的列的名称。
它们必须在数量和顺序上与SELECT语句中指定的表列相对应。
还可以在SELECT语句中指定这些视图列名作为列名别名。
如果两者都不指定，则使用表列名作为视图列名。

以下示例显示了指定视图列名的两种方法:

```
ALTER VIEW MyView (MyViewCol1,MyViewCol2,MyViewCol3) AS
SELECT TableCol1, TableCol2, TableCol3 FROM MyTable
```

等于:

```
ALTER VIEW MyView AS SELECT TableCol1 AS ViewCol1,  
    TableCol2 AS ViewCol2,  
    TableCol3 AS ViewCol3  
FROM MyTable
```

列规范将替换为视图指定的任何先前的列。

视图查询不能包含主机变量或包含INTO关键字。

如果试图在查询中引用主机变量，系统将生成SQLCODE -148错误。

权限

ALTER VIEW命令是一个特权操作。

用户必须具有%ALTERVIEW管理权限才能执行ALTER VIEW。

如果不这样做，将导致与的SQLCODE -99错误%msg User 'name' does not have %ALTERVIEW privileges.

用户必须在指定的视图上拥有%ALTER权限。

如果用户是视图的Owner(创建者)，则自动授予该用户该视图的%ALTER权限。

否则，用户必须被授予视图%ALTER权限。

如果不这样做，将导致带有%msg的SQLCODE -99错误User 'name' does not have privilege to ALTER the view 'Schema.ViewName'.

可以使用GRANT命令分配%ALTERVIEW和%ALTER权限，如果拥有适当的授予权限。

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

在嵌入式SQL中，可以使用\$SYSTEM.Security.Login()方法以具有适当权限的用户登录：

```
DO $SYSTEM.Security.Login("_SYSTEM", "SYS")  
&sql(      )
```

必须具有%ServiceLogin:Use权限才能调用 \$SYSTEM.Security.Login ,

不能在基于从部署的持久类投射的表的视图上使用ALTER VIEW。

此操作失败，并出现一个带有%msg的SQLCODE -400错误Unable to execute DDL that modifies a deployed class: 'classname'.

示例

下面的示例创建一个视图，然后更改该视图。

提供了查询视图和删除视图的程序。

注意，改变视图将列列表替换为一个新的列列表；

它不保留前面的列列表。

```
/// w ##class(PHA.TEST.SQLCommand).AlterView()  
ClassMethod AlterView()  
{  
    if $SYSTEM.SQL.Schema.ViewExists("MassFolks") {
```

```

        w "?????????????"
        q
    }
    &sql(
        CREATE VIEW MassFolks (vFullName) AS
        SELECT Name FROM Sample.Person WHERE Home_State='MA'
    )
    if SQLCODE=0 {
        w !,"????",!
    } else {
        w "??????SQLCODE=",SQLCODE
    }
}

```

SELECT * FROM MassFolks



```

ClassMethod AlterView1()
{
    if 0 = $SYSTEM.SQL.Schema.ViewExists("MassFolks") {
        w "?????"
        q
    }
    &sql(
        ALTER VIEW MassFolks (vMassAbbrev,vCity) AS
        SELECT Home_State,Home_City FROM Sample.Person WHERE Home_State='MA'
    )
    if SQLCODE = 0 {
        WRITE !,"????",!
    } ELSE {
        WRITE "??????SQLCODE=",SQLCODE
    }
}

```

DROP VIEW MassFolks

下面的嵌入式SQL示例使用WITH CHECK OPTION查询更改视图:

```

ClassMethod AlterView2()
{
    DO $SYSTEM.Security.Login("_SYSTEM","SYS")
    &sql(
        ALTER VIEW Sample.MyTestView AS

```

```
        SELECT FIRSTWORD FROM Sample.MyTest WITH CHECK OPTION
    )
    if SQLCODE = 0 {
        WRITE !, "????", !
    } ELSE {
        WRITE "??????SQLCODE=", SQLCODE
    }
}
```

[#SQL](#) [#Caché](#)

源

URL:

<https://cn.community.intersystems.com/post/%E7%AC%AC%E5%9B%9B%E7%AB%A0-sql%E5%91%BD%E4%B%B%A4-alter-view>