文章
姚 鑫 · 十月 7, 2021 阅读大约需 6 分钟

第三十八章 SQL命令 DROP TABLE

第三十八章 SQL命令 DROP TABLE

删除表及其数据(可选)。

大纲

```
DROP TABLE table
     [RESTRICT | CASCADE] [%DELDATA | %NODELDATA]
```

参数

- table - 要删除的表的名称。
  表名可以是限定的(schema.table)，也可以是非限定的(table)。
  非限定表名接受默认模式名。
  没有使用架构搜索路径值。
- RESTRICT ，CASCADE - 可选-限制仅允许删除没有依赖视图或完整性约束的表。如果未指定关键字，则默认设置为RESTRITION。CASCADE允许删除具有依赖视图或完整性约束的表；作为表删除的一部分，任何引用视图或完整性约束也将被删除。外键约束不支持CASCADE关键字选项。
- %DELDATA，%NODELDATA - 可选-
  这些关键字指定在删除表时是否删除与表关联的数据。默认情况下，删除表格数据。

描述

DROP TABLE命令删除一个表及其对应的持久化类定义。如果该表是其架构中的最后一项，则删除该表也会删除该架构及其相应的持久化类包。

默认情况下，DROP TABLE同时删除表定义和表数据(如果存在)。%NODELDATA关键字允许指定删除表定义，但不能指定删除表的数据。

DROP TABLE删除与该表关联的所有索引和触发器。

要删除表格，必须满足以下条件：

- 该表必须存在于当前命名空间中。尝试删除不存在的表会生成SQLCODE-30错误。
- 表定义必须是可修改的。如果投影表的类没有定义[DdlAllowed]，则尝试删除该表会生成SQLCODE-300错误。
- 该表不能被另一个并发进程锁定。如果表被锁定，DROP
  TABLE将无限期等待锁被释放。如果可能出现锁争用，那么在发出DROP
  TABLE之前以独占模式锁定表是很重要的。
- 该表必须没有关联的视图，或者DROP TABLE必须指定CASCADE关键字。尝试在不级联的情况下删除具有关联视图的表会生成SQLCODE-321错误。
- 必须具有必要的权限才能删除该表。尝试在没有必要权限的情况下删除表会生成SQLCODE-99错误。
- 即使相应的类被定义为已部署的类，也可以删除表。
- 如果投影表的持久类具有派生类(子类)，则不能删除该表。尝试删除会使子类成为孤立的超类时，会生成SQL

> CODE-300错误，并显示以下消息：Class 'MySuperClass' has derived classes and therefore cannot be dropped via DDL.

可以使用$SYSTEM.SQL.Schema.DropTable()方法删除当前名称空间中的表。可以指定SQL表名。与DROP TABLE不同，此方法可以删除未使用[DdlAllowed]定义的表。第二个参数指定是否也应该删除表数据；默认情况下，不删除数据。

```
DO $SYSTEM.SQL.Schema.DropTable("Sample.MyTable",1,.SQLCODE,.%msg)
IF SQLCODE '= 0 {WRITE "SQLCODE ",SQLCODE," error: ",%msg}
```

可以使用$SYSTEM.OBJ.Delete()方法删除当前名称空间中的一个或多个表。必须指定投影表的永久类名(而不是SQL表名)。可以使用通配符指定多个类名。第二个参数指定是否也应该删除表数据；默认情况下，不删除数据。

权限

DROP TABLE命令是特权操作。用户必须具有%DROPTABLE管理权限才能执行DROP TABLE。否则将导致SQLCODE-99错误，因为%msg用户没有%DROPTABLE权限。如果拥有适当的授予权限，则可以使用GRANT命令分配%DROPTABLE权限。

即使DROP TABLE操作同时删除了表和表数据，用户也不必具有指定表的DELETE OBJECT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身份登录：

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

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

DROP TABLE不能用于通过定义持久类创建的表，除非表类定义包括[DdlAllowed]。否则，操作将失败，并出现SQLCODE-300错误，同时未为类 ‘ Schema.tablename ’ 启用%msg DDL。

Existing Object Privileges

删除表不会删除该表的对象权限。例如，授予用户在该表上插入、更新或删除数据的权限。
- 如果删除一个表，然后创建另一个同名的表，则用户和角色对新表的权限将与对旧表的权限相同。
- 一旦表被删除，就不可能撤销该表的对象权限。

由于这些原因，通常建议在删除表之前使用REVOKE命令撤消表中的对象权限。

包含数据的表

默认情况下，DROP TABLE删除表定义和表数据。此表数据删除是原子操作；如果DROP TABLE遇到无法删除的数据(例如，具有引用约束的行)，则任何已执行的数据删除都会自动回滚，结果是不会删除表数据。

可以使用$SYSTEM.SQL.Util.SetOption()方法DDLDropTabDelData选项设置表数据删除的系统范围默认值。要确定当前设置，请调用$SYSTEM.SQL.CurrentSettings()，显示Does DDL DROP TABLE delete the table's data? setting.

默认值为1(“ 是 ”)。这是此选项的推荐设置。如果希望DROP TABLE在删除表定义时不删除表数据，请将此选项设置为0(“ 否 ”)。

可以在每个表的基础上覆盖数据删除。删除表时，可以使用%NODELDATA选项指定DROP TABLE，以防止自动删

除表数据。如果系统范围的默认值设置为不删除表数据，则可以通过使用%DELDATA选项指定DROP TABLE来逐个表删除数据。

在大多数情况下，DROP TABLE使用高效的终止范围操作自动删除表的数据。以下情况阻止使用KILL EXTEND：表有引用它的外键；投影表的类是持久类的子类；类不使用默认存储；有ForEach = "row/object"触发器；有引用非默认流字段全局位置的流字段。如果其中任何一个适用，DROP TABLE将使用效率较低的DELETE RECORD操作删除表的数据。

可以使用TRUNCATE TABLE命令删除表的数据，而不删除表定义。

锁应用

DROP TABLE语句获取表的独占表级锁。这可以防止其他进程在表删除过程中修改表定义或表数据。这个表级锁足以删除表定义和表数据；DROP TABLE不会获得表数据的每一行的锁。此锁在DROP TABLE操作结束时自动释放。

外键约束

默认情况下，如果在引用尝试删除的表的另一个表上定义了任何外键约束，则不能删除该表。在删除它们引用的表之前，必须删除所有引用的外键约束。在尝试DROP TABLE操作之前未删除这些外键约束会导致SQLCODE-320错误。

此默认行为与限制关键字选项一致。外键约束不支持CASCADE关键字选项。

要更改此默认外键约束行为，请参考SET OPTION命令的COMPILEMODE=NOCHECK选项。

相关查询

删除表会自动清除所有相关的高速缓存查询，并清除%SYS.PTools.StatsSQL生成的查询信息。删除表会自动清除任何相关查询的所有SQL运行时统计信息(SQL Stats)信息。

不存在的表

要确定当前命名空间中是否存在指定表，请使用$SYSTEM.SQL.Schema.TableExists()方法。

默认情况下，如果尝试删除不存在的表，DROP TABLE会发出SQLCODE-30错误。这是推荐的设置。要确定当前设置，请调用$SYSTEM.SQL.CurrentSettings()，它显示允许DDL删除不存在的表或视图设置。默认值为0(“否”)。如果此选项设置为1(“是”)，则不存在的表的DROP TABLE不执行任何操作，也不会发出错误消息。

在管理门户、系统管理、配置、SQL和对象设置中，通过选中忽略冗余DDL语句复选框，可以在系统范围内设置此选项(以及其他类似的创建、更改和删除选项)。

示例

下面的嵌入式SQL示例创建名为SQLUser.MyEmployees的表，然后将其删除。此示例指定在删除该表时不删除与该表关联的任何数据：

```
ClassMethod DropTable()
{
    &sql(CREATE TABLE SQLUser.MyEmployees
        (
            NAMELAST     CHAR (30) NOT NULL,
            NAMEFIRST    CHAR (30) NOT NULL,
```

```
            STARTDATE    TIMESTAMP,
            SALARY       MONEY
        )
    )
    w !,"???"
    /*
    &sql(SQL code populating SQLUser.MyEmployees table)
    &sql(SQL code using SQLUser.MyEmployees table)
    */
    n SQLCODE,%msg
    &sql(DROP TABLE SQLUser.MyEmployees %NODELDATA)
    if SQLCODE = 0 {
        w !,"????"
    } else {
        w !,"SQLCODE=",SQLCODE,": ",%msg
    }
}
```

#SQL #Caché

源
URL:
https://cn.community.intersystems.com/post/%E7%AC%AC%E4%B8%89%E5%8D%81%E5%85%AB%E7%AB%A0-sql%E5%91%BD%E4%BB%A4-drop-table