文章
Jingwei Wang · 七月 21, 2022 阅读大约需 4 分钟

InterSystems SQL 的使用 - 第五部分 - 导入SQL Code

本章介绍了如何将SQL code从文本文件导入InterSystems SQL。当你导入SQL code时，InterSystems IRIS数据平台使用动态SQL准备并执行每一行的SQL。如果遇到无法解析的SQL code行，SQL导入会跳过该行code，继续准备和执行后续的code行，直到到达文件的末端。所有的SQL code导入操作都会导入到当前的命名空间。

SQL导入主要用于导入数据定义语言（DDL）命令，如CREATE TABLE，并使用INSERT、UPDATE和DELETE命令来填充表。SQL导入可以准备和执行SELECT查询，但不创建结果集。

SQL导入可以用来导入InterSystems的SQL code。它也可以用于code迁移，从其他供应商（FDBMS、Informix、InterBase、MSSQLServer、MySQL、Oracle、Sybase）导入SQL。来自其他供应商的code被转换为InterSystems的SQL并执行。SQL导入不能将所有的SQL命令导入到InterSystems SQL中。它导入的是那些与InterSystems IRIS实现的SQL标准兼容的命令和条款。不兼容的特征通常会被解析，但会被忽略。

SQL导入可以成功地准备一个SQL查询--在适当的时候创建一个相应的缓存查询--但它不会执行查询。

你通过调用%SYSTEM.SQL.Schema类中的适当方法来执行SQL code导入。当导入SQL code时，这些方法可以创建另外两个文件：一个是Errors.log文件，它记录了解析SQL命令的错误；另一个是Unsupported.log文件，它包含了该方法不能识别为SQL命令的行的字面文本。

导入 InterSystems SQL Code

你可以使用**%SYSTEM.SQL.Schema**方法从一个文本文件中导入InterSystems的SQL code。

ImportDDL()
是一个通用的SQL导入方法。该方法作为一个后台（非交互式）进程运行。如下面的例子中所示，示例中第一个参数mysqlcode.txt为SQL命令文件，第二个参数是默认的错误日志文件mysqlcodeErrors.log，第三个参数指定为 "IRIS"。

```
 DO $SYSTEM.SQL.Schema.ImportDDL("c:\InterSystems\mysqlcode.txt",,"IRIS")
```

注意：这种SQL DDL代码的导入和执行不应该与管理门户SQL界面的导入语句操作相混淆。管理门户SQL界面的导入语句是以XML格式导入SQL语句。

这个mysqlcode.txt文本文件必须是一个没有格式化的文件，如.txt文件。每条SQL命令必须在自己的行中开始。一个SQL命令可以分成多行，允许缩进。每条SQL命令后面必须有一个GO语句，并在自己的行上，下面是一个有效的InterSystems SQL导入文件文本的例子：

```
CREATE TABLE Sample.MyStudents (StudentName VARCHAR(32),StudentDOB DATE)
GO
 CREATE INDEX NameIdx ON TABLE Sample.MyStudents (StudentName)
GO
 INSERT INTO Sample.MyStudents (StudentName,StudentDOB) SELECT Name,
DOB FROM Sample.Person WHERE Age <= '21'
GO
 INSERT INTO Sample.MyStudents (StudentName,StudentDOB)
        VALUES ('Jones,Mary',60123)
```

```
GO
 UPDATE Sample.MyStudents SET StudentName='Smith-
Jones,Mary' WHERE StudentName='Jones,Mary'
GO
 DELETE FROM Sample.MyStudents WHERE StudentName %STARTSWITH 'A'
GO
```

Run() 是一个InterSystems SQL导入方法。这个方法在**终端**以交互方式运行。它提示你指定导入文本文件的位置，创建Errors.log文件和Unsupported.log文件的位置，以及其他信息。

支持的SQL命令

不是所有有效的InterSystems SQL命令都能被导入。以下是支持的InterSystems SQL命令的列表。

```
CREATE TABLE, ALTER TABLE, DROP TABLE
?
CREATE VIEW, ALTER VIEW, DROP VIEW
?
CREATE INDEX all index types, except bitslice
?
CREATE USER, DROP USER
?
CREATE ROLE
?
GRANT, REVOKE
?
INSERT, UPDATE, INSERT OR UPDATE, DELETE
?
SET OPTION
?
SELECT for optimizer plan mode only
```

导入非 InterSystems SQL Code

InterSystems SQL 支持导入其他供应商使用的SQL。来自其他供应商的code被转换为InterSystems的SQL code并执行，我们提供了以下方法。

ImportDDL() 也可以用来导入非InterSystems SQL。

要导入特定格式的SQL，你需要指定该格式的名称作为一个参数，例如 FDBMS, Informix, InterBase, MSSQLServer (或MSSQL), MySQL, Oracle, 或Sybase。下面的例子导入了MSSQL代码文件mssqlcode.txt，在当前命名空间中执行该文件中列出的SQL命令。

```
DO $SYSTEM.SQL.Schema.ImportDDL($lb("C:\temp\somesql.sql","UTF8"),,"MSSQL")
```

注意，如果第三个参数是MSSQL、Sybase、Informix或MySQL，第一个参数可以是一个SQL code文件路径名或一个双元素%List，第一个元素是SQL code文件路径名，第二个元素是要使用的I/O转换表。

在**%SYSTEM.SQL.Schema**中提供了单独的交互式方法来导入以下类型的SQL。**LoadFDBMS(), LoadInformix(),**

LoadInterBase(), LoadMSSQLServer(), LoadOracle(), and LoadSybase()
。这些方法从终端交互式地运行。它提示你指定导入文本文件的位置，创建Errors.log文件和Unsupported.log文件的位置，以及其他信息。

ImportDDLDir()允许你从一个目录中的多个文件导入SQL code。该方法作为一个后台进程运行。它支持**Informix、MSSQLServer和Sybase。所有要导入的文件必须有一个**.sql**的后缀。

ImportDir()允许你从一个目录中的多个文件导入SQL code。比ImportDDLDir()提供更多的选项。该方法作为一个后台进程运行。它支持MSSQLServer，和Sybase。你可以指定一个允许的文件扩展名后缀列表。

#SQL #InterSystems IRIS for Health

源URL:https://cn.community.intersystems.com/post/intersystems-sql-%E7%9A%84%E4%BD%BF%E7%94%A8-%E7%AC%AC%E4%BA%94%E9%83%A8%E5%88%86-%E5%AF%BC%E5%85%A5sql-code