

文章

姚鑫 · 四月 4, 2021 阅读大约需分钟

第十六章 导入SQL Code

第十六章 导入SQL Code

本章介绍如何将SQL代码从文本文件导入InterSystems SQL。导入SQL代码时，InterSystems Iris®数据平台使用动态SQL准备并执行每行SQL。如果遇到无法解析的代码行，则SQL导入跳过该行代码并继续准备和执行后续行，直到它到达文件末尾。所有SQL代码导入都导入到当前名称空间。

SQL导入主要用于导入数据定义语言(DDL)命令(例如Create Table)，并使用Insert, Update和Delete命令填充表。SQL导入确实准备并执行SELECT查询，但不创建结果。

SQL导入可用于导入InterSystems SQL代码。它也可以用于代码迁移，从其供应商导入SQL代码(FDBMS, Informix, Interbase, MSSQLServer, MySQL, Oracle, Sybase)。来自其他供应商的代码被转换为InterSystems SQL代码并执行。SQL导入无法将所有SQL命令导入InterSystems SQL。它导入与SQL标兼容的InterSystems Iris实现兼容的那些命令和条款。不兼容的功能通常被解析，但忽略了。

SQL导入可以准备一个SQL查询 - 在适当情况创建相应的缓存查询 - 但它不会执行查询。

通过从%system.sql.schema类中调用相应的方法，执行SQL代码导入。导入SQL代码时，这些方法可以创建其他两个文件：errors.log文件，它记录解析SQL命令中的错误，以及一个不支持的。其中包含该方法无法识别为SQL命令的文字文本。

导入InterSystems SQL.

可以使用%System.sql.schema方法从文本文件中导入InterSystems SQL代码：

- ImportDDL()是一个通用的SQL导入方法。此方法运行作为背景(非交互式)进程。要导入InterSystems SQL，请将"IRIS"指定为第一个参数。
- run()是一个InterSystems SQL导入方法。该方法从终端交互方式运行。它会提示指定导入文本文件的位置，创建Errors.log文件和Unsupported.log文件及其他信息。

注意：此导入和执行SQL DDL代码不应与门户SQL接口的Import语句混淆。该语句以XML格式导入SQL语句。

以下示例导入InterSystems Iris SQL代码文件Pathname MySQLCode.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("IRIS", $USERNAME, "c:\temp\mysqlcode.txt", , 1)
```

默认情况下，ImportDDL()创建错误日志文件。此示例在与SQL代码文件中创建名为mysqlcode_errors.log文件。第五个参数是一个布尔值，指定是否创建列出不受支持的SQL命令文件。默认值为0。在此示例中，第五个参数设置为1，在与SQL代码文件相同的目录中创建名为mysqlcode_unsupported.log文件。即使没有写入它们时，也会创建这些日志文件。

从终端执行ImportDDL()时,它列出输入文件,错误日志文件和不支持的日志文件。然后,对于每个SQL命令,它显示一个列表,例如以下内容:

```
SQL statement to process (number 1):
  CREATE TABLE Sample.MyStudents (StudentName VARCHAR(32),
  StudentDOB DATE)
  Preparing SQL statement...
  Executing SQL statement...
DONE
```

如果在任何SQL命令中发生错误,则终端显示错误,如以下示例所示:

```
SQL statement to process (number 3):
  INSERT INTO Sample.MyStudents (StudentName,StudentDOB) SELECT Name,
  DOB FROM Sample.Person WHERE Age <= '21'
  Preparing SQL statement...
ERROR #5540: SQLCODE: -30 Message: Table 'SAMPLE.PERSON' not found
  Pausing 5 seconds - read error message! (Type Q to Quit)
```

如果在5秒内没有退出,则导入DDL()继续执行一个SQL命令。错误记录在错误日志文件中,具有时间戳,用户名和命名空间名称。

导入文件格式

SQL文本文件必须是未格式化的文件,例如.txt文件。每个SQL命令必须在其自己的行中开始。SQL命令可能会被丢进该行,允许缩进。默认情况下,每个SQL命令必须在其自己的行上进行Go语句。

以下是一个有效的InterSystems SQL Import文件文本的示例:

```
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
```

```
DHC-APP>DO $SYSTEM.SQL.DDLImport("CACHE","glenn","e:\temp\yxtest.txt",,1)
```

```
Importing SQL Statements from file: e:\temp\yxtest.txt
```

```
Recording any errors to principal device and log file: e:\temp\yxtest_Errors.log
```

Recording unsupported statements to file: e:\temp\yxtest_Unsupported.log

```
SQL statement to process (number 1):
  CREATE TABLE Sample.MyStudents (StudentName VARCHAR(32),
  StudentDOB DATE)
  Preparing SQL statement...
  Executing SQL statement...
DONE

SQL statement to process (number 2):
  INSERT INTO Sample.MyStudents (StudentName,StudentDOB) SELECT Name,
  DOB FROM Sample.Person WHERE Age <= '21'
  Preparing SQL statement...
  Executing SQL statement...
DONE

SQL statement to process (number 3):
  INSERT INTO Sample.MyStudents (StudentName,StudentDOB)
  VALUES ('Jones,Mary',60123)
  Preparing SQL statement...
  Executing SQL statement...
DONE

SQL statement to process (number 4):
  INSERT OR UPDATE INTO Sample.MyStudents (StudentName,StudentDOB) VALUES ( '
  Smith-Jones,Mary',60123)
  Preparing SQL statement...
  Executing SQL statement...
DONE

SQL statement to process (number 5):
  DELETE FROM Sample.MyStudents WHERE StudentName %STARTSWITH 'A'
  Preparing SQL statement...
  Executing SQL statement...
DONE

SQL statement to process (number 6):
  SELECT TOP 5 * FROM Sample.MyStudents
  Preparing SQL statement...
  Executing SQL statement...
DONE
```

Elapsed time: 5.750462 seconds

通过设置ImportDDL("IRIS")DEOS第参数,此方法可以接受(但不需指定的语句末尾分隔符,通常是分号(;),在每个SQL命令的末尾。默认值不支持终止终止分隔符。始终支持SQL命令后行的"Go"语句,但如果deos指定语句结束分隔符,则不需

支持的SQL命令

并非所有有效的InterSystems都可以导入SQL命令。以下是支持的InterSystems SQL命令列表:

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

代码迁移: 导入非InterSystems SQL

可以导入以其他供应商使用的SQL格式的SQL代码。来自其他供应商的代码被转换为InterSystems SQL代码并执行。提供以下方法:

- ImportDDL()是一个通用的SQL导入方法。此方法运行作为背景(非交互式)进程。

要以特定格式导入SQL, 将该格式的名称指定为第一个参数: FDBMS, Informix, Interbase, MSSQLSERVER(或MSSQL), MySQL, Oracle或Sybase。

以下示例导入SQL代码文件SQLCode.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("MSSQL", $USERNAME, $l$("C:\temp\somesql.sql", "UTF8"), , 1)
```

请注意, 如果第一个参数是MSSQL, Sybase, Informix或MySQL, 则第三个参数可以是SQL代码文件路径名或具有第一个元素的两个元素%list, SQL代码文件路径名和第二个元素i / o用于使用的翻译表。

- 在%system.sql.schema中提供单个交互式方法, 用于导入以下类型的SQL: loadfdbms(), loadInformix(), loadInterbase(), loadmssqlserver(), loadoracle()和loadybase()。这些方法与终端交互式运行。它会提示指定导入文本文件的位置, 创建Errors.log文件和Unsupported.log文件以及其他信息。
- ImportDDLDIR()允许从目录中导入SQL代码。此方法运行作为背景(非交互式)进程。它支持Informix, MSSQLServer和Sybase。要导入的所有文件必须具有.sql扩展后缀。
- ImportDir()允许从目录中导入SQL代码。提供比ImportDDLDIR()更多选项。此方法运行作为背景(非交互式)进程。它支持MSSQLServer和Sybase。可以指定允许文件扩展后缀的列表。

[#SQL #Caché #InterSystems IRIS #InterSystems IRIS for Health](#)

源 URL: <https://cn.community.intersystems.com/post/%E7%AC%AC%E5%8D%81%E5%85%AD%E7%AB%A0-%E5%AF%BC%E5%85%A5sql-code>