

文章

姚鑫 · 二月 27, 2021



阅读大约需分钟

第四十九章 Caché 变量大全 ^\$ROUTINE 变量

第四十九章 Caché 变量大全 ^\$ROUTINE 变量

提供例程信息。

大纲

^\$ | nspace | ROUTINE (routine_name)

^\$ | nspace | R (routine_name)

参数

- [nspace]或[nspace] 可选-扩展SSVN引用,可以是显式名称空间名称,也可以是隐含名称空间。必须计算为带引号的字符串,该字符串括在方括号(["nspace"])或竖线("|nspace"|)中。命名空间名称不区分大小写;它们以大写字母存储和显示。
- routine_name 计算结果为包含例程名称的字符串的表达式。

描述

可以将^\$ROUTINE结构化系统变量用作\$DATA、\$ORDER和\$QUERY函数的参数,以从当前命名空间(默认)或指定命名空间返回例程信息。^\$ROUTINE返回有关例程的OBJ代码版本的例程信息。

在InterSystems ObjectScript中,一个例程有三个代码版本:MAC(用户编写的代码,可能包括宏预处理器语句)、INT(编译的MAC代码,用于执行宏预处理)和OBJ(可执行目标代码)。可以使用^\$ROUTINE global返回关于int代码版本的信息。可以使用^\$ROUTINE返回有关OBJ代码版本的信息。

参数

nspace

此可选参数允许使用扩展SSVN引用在另一个命名空间中指定全局。可以显式地将命名空间名称指定为带引号的字符串文字或变量,也可以通过指定隐含的命名空间来指定。命名空间名称不区分大小写。可以使用方括号语法["user"]或环境语法["user"]。Nspace分隔符前后不允许有空格。

```
WRITE ##class(%SYS.Namespace).Exists("USER"),! ; an existing namespace
WRITE ##class(%SYS.Namespace).Exists("LOSER") ; a non-existent namespace
```

可以使用\$NAMESPACE特殊变量来确定当前名称空间。更改当前名称空间的首选方式是新建\$NAMESPACE,然后设置\$NAMESPACE="nspace ename"。

routine_name

计算结果为包含现有例程名称的字符串的表达式。例程名称在前255个字符内必须是唯一的，应避免超过220个字符。

示例

以下示例使用^\$例程作为\$DATA、\$ORDER和\$QUERY函数的参数。

作为\$DATA的参数

`$DATA(^$|nspace|ROUTINE(routine_name))`

^\$ROUTINE作为\$DATA的参数将返回一个整数值，该整数值指定例程名OBJ代码版本是否作为^\$ROUTINE中的节点存在。下表显示了\$DATA可以返回的整数值。

Value

0

10

下面的Terminal示例测试myrou例程的OBJ代码版本是否存在。此示例假定在USER名称空间中有一个名为myrou的已编译MAC例程：

```
USER>WRITE ^ROUTINE("myrou",0,"GENERATED") // INT code version exists
1
USER>WRITE $DATA(^$ROUTINE("myrou")) // OBJ code version exists
1
USER>KILL ^rOBJ("myrou") // Kills the OBJ code version

USER>DO ^myrou

DO ^myrou
^
<NOROUTINE> *myrou
USER>WRITE ^ROUTINE("myrou",0,"GENERATED") // INT code version exists
1
USER>WRITE $DATA(^$ROUTINE("myrou")) // OBJ code version does not exist
0
USER>
```

作为\$ORDER的参数

`$ORDER(^$|nspace|ROUTINE(routine_name),direction)`

`^$ROUTINE`作为`$ORDER`的参数,按整理顺序返回指定的例程名称的一个或上一个例程名称。如果在`^$ROUTINE`中没有这样的例程名称作为节点存在,则`$ORDER`返回空字符串。

`direction`参数指定是否返回一个或上一个例程名称:1 = 一个, -1 = 上一个。如果不提供方向参数,则InterSystems IRIS将按整理顺序将第一个例程名称返回到指定的名称。

例子例程搜索USER名称空间,并将例程名称存储在名为ROUTINE的本地数组中。

```

/// d ##class(PHA.TEST.SpecialVariables).ROUTINE()
ClassMethod ROUTINE()
{
    SET rname=""
    FOR I=1:1 {
        SET rname=$ORDER(^$|"USER"|ROUTINE(rname))
        QUIT:rname=""
        SET ROUTINE(I)=rname
        WRITE !,"Routine name: ",rname
    }
    WRITE !,"All routines stored"
    QUIT
}

```

```

Routine name: INFORMATION.SCHEMA.TABLECONSTRAINTS.1
Routine name: INFORMATION.SCHEMA.TABLES.0
Routine name: INFORMATION.SCHEMA.TABLES.1
Routine name: INFORMATION.SCHEMA.TRIGGERS.0
Routine name: INFORMATION.SCHEMA.TRIGGERS.1
Routine name: INFORMATION.SCHEMA.VIEWCOLUMNUSAGE.0
Routine name: INFORMATION.SCHEMA.VIEWCOLUMNUSAGE.1
Routine name: INFORMATION.SCHEMA.VIEWS.0
Routine name: INFORMATION.SCHEMA.VIEWS.1
Routine name: INFORMATION.SCHEMA.VIEWTABLEUSAGE.0
Routine name: INFORMATION.SCHEMA.VIEWTABLEUSAGE.1
All routines stored

```

作为\$QUERY的参数

```
$QUERY(^$|nspace|ROUTINE(routine_name))
```

`^$ROUTINE`作为`$QUERY`的参数,按整理顺序将第一个例程名称返回到指定的例程名称。指定的例程名称不必存在。如果以后在排序序列中没有例程名称,则`$QUERY(^$ROUTINE)`返回一个空字符串。

在面的示例中,两个`$QUERY`函数在USER名称空间中指定例程名称之后返回一个例程。

```

/// d ##class(PHA.TEST.SpecialVariables).ROUTINE1()
ClassMethod ROUTINE1()
{
    SET rname=""
    WRITE !,"1st routine: ",$QUERY(^$|"USER"|ROUTINE(rname))
    SET rname="%m"
    WRITE !,"1st ",rname, " routine: ",$QUERY(^$|"USER"|ROUTINE(rname))
    QUIT
}

```

```
DHC-APP>d ##class(PHA.TEST.SpecialVariables).ROUTINE1()
```

```
lst routine: ^$|"USER"|ROUTINE("%APILIB")
```

```
lst %m routine: ^$|"USER"|ROUTINE("%mgw1")
```

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

源 URL: <https://cn.community.intersystems.com/post/%E7%AC%AC%E5%9B%9B%E5%8D%81%E4%B9%9D%E7%AB%A0-cach%C3%A9-%E5%8F%98%E9%87%8F%E5%A4%A7%E5%85%A8-routine-%E5%8F%98%E9%87%8F>