

文章

[姚鑫](#) · 三月 18, 2022 阅读大约需 5 分钟

第八十五章 SQL函数 \$LISTGET

第八十五章 SQL函数 \$LISTGET

返回列表中的元素或指定默认值的列表函数。

大纲

`$LISTGET(list[,position[,default]])`

参数

- list - 计算结果为有效列表的表达式。列表是包含一个或多个元素的编码字符串。可以使用 SQL 或 ObjectScript 的 \$LISTBUILD 或 \$LISTFROMSTRING 函数创建列表。可以使用 SQL 或 ObjectScript \$LIST 函数从现有列表中提取列表。
- position - 可选 — 解释为指定列表中的位置的表达式。
- default - 可选 — 如果列表元素具有未定义的值，则该表达式提供要返回的值。

描述

\$LISTGET 将指定列表中的请求元素作为标准字符串返回。如果位置参数的值引用不存在的成员或标识具有未定义值的元素，则返回指定的默认值。

\$LISTGET 函数与 \$LIST 函数的单参数和双参数形式相同，只是在导致 \$LIST 返回空字符串的条件下，\$LISTGET 返回默认值。

此函数返回 VARCHAR 类型的数据。

可以使用 \$LISTGET 从串行容器字段中检索字段值。在以下示例中，Home 是一个串行容器字段，其第三个元素是 HomeState：

```
SELECT Name,$LISTGET(Home,3) AS HomeState
FROM Sample.Person
```

参数

list

包含一个或多个元素的编码字符串。可以使用 SQL \$LISTBUILD 函数或 ObjectScript \$LISTBUILD 函数创建列表。可以使用 SQL \$LISTFROMSTRING 函数或 ObjectScript \$LISTFROMSTRING 函数将分隔字符串转换为列表。可以使用 SQL \$LIST 函数或 ObjectScript \$LIST 函数从现有列表中提取列表。

position

位置参数必须计算为整数。如果省略，默认情况下，该函数检查列表的第一个元素。如果 position 参数的值为 -1，则相当于指定列表的最后一个元素。

default

字符串。如果省略默认参数，则默认值假定为零长度字符串。

示例

以下嵌入式 SQL 示例中的 \$LISTGET 函数都返回 “ Red ”，即列表中的第一个元素：

```
/// d ##class(PHA.TEST.SQLFunction).ListGet()  
ClassMethod ListGet()  
{  
    s a = $lb("Red", "Blue", "Green")  
    &sql(  
        SELECT $LISTGET(:a), $LISTGET(:a, 1)  
        INTO :b, :c)  
    if SQLCODE '= 0 {  
        w !, "Error code:", SQLCODE  
    } else {  
        w !, "The one-arg element returned is ", b  
        w !, "The two-arg element returned is ", c  
    }  
}
```

```
DHC-APP>d ##class(PHA.TEST.SQLFunction).ListGet()
```

The one-arg element returned is Red

The two-arg element returned is Red

以下嵌入式 SQL 示例中的 \$LISTGET 函数都返回 “ Green ”，即列表中的第三个和最后一个元素：

```
/// d ##class(PHA.TEST.SQLFunction).ListGet1()  
ClassMethod ListGet1()  
{  
    s a = $lb("Red", "Blue", "Green")  
    &sql(  
        SELECT $LISTGET(:a, 3), $LISTGET(:a, -1)  
        INTO :b, :c)  
    if SQLCODE '= 0 {  
        w !, "Error code:", SQLCODE  
    } else {  
        w !, "The third element is ", b  
        w !, "The last element is ", c  
    }  
}
```

```
DHC-APP>d ##class(PHA.TEST.SQLFunction).ListGet1()
```

The third element is Green

The last element is Green

以下嵌入式 SQL 示例中的 \$LISTGET 函数都在遇到列表中未定义的第二个元素时返回一个值。第一个返回问号 (?), 用户将其定义为默认值。第二个返回空字符串, 因为未指定默认值:

```
/// d ##class(PHA.TEST.SQLFunction).ListGet2()
ClassMethod ListGet2()
{
    s a = $lb("Red", , "Green")
    &sql(
        SELECT $LISTGET(:a, 2, '?'), $LISTGET(:a, 2)
            INTO :b, :c)
    if SQLCODE '= 0 {
        w !, "Error code:", SQLCODE
    } else {
        w !, "The default value is ", b
        w !, "The no-default value is ", c
    }
}
```

```
DHC-APP>d ##class(PHA.TEST.SQLFunction).ListGet2()
```

The default value is ?

The no-default value is

以下嵌入式 SQL 示例中的 \$LISTGET 函数都指定了一个大于三元素列表中最后一个元素的位置。第一个返回空字符串, 因为未指定默认值。第二个返回用户指定的默认值 “ERR”:

```
/// d ##class(PHA.TEST.SQLFunction).ListGet3()
ClassMethod ListGet3()
{
    s a = $lb("Red", "Blue", "Green")
    &sql(
        SELECT $LISTGET(:a, 4), $LISTGET(:a, 4, 'ERR')
            INTO :b, :c)
    if SQLCODE '= 0 {
        w !, "Error code:", SQLCODE
    } else {
        w !, "The no-default 4th element is ", b
        w !, "The default for 4th element is ", c
    }
}
```

```
DHC-APP>d ##class(PHA.TEST.SQLFunction).ListGet3()
```

The no-default 4th element is

The default for 4th element is ERR

以下嵌入式 SQL 示例中的 \$LISTGET 函数均返回空字符串:

```

/// d ##class(PHA.TEST.SQLFunction).ListGet4()
ClassMethod ListGet4()
{
    s a = $lb("Red", "Blue", "Green")
    &sql(
        SELECT $LISTGET(:a, 0), $LISTGET(NULL)
            INTO :b,:c
    )
    if SQLCODE '= 0 {
        w !,"Error code;",SQLCODE
    } else {
        w !,"The zero element is ",b
        w !,"The NULL element is ",c
    }
}

```

```
DHC-APP>d ##class(PHA.TEST.SQLFunction).ListGet4()
```

```

The zero element is
The NULL element is

```

注意

无效的参数值

如果列表参数中的表达式未计算为有效列表，则会发生 SQLCODE -400 致命错误，因为 \$LISTGET 返回变量仍未定义。即使提供了默认值也会发生这种情况，如以下嵌入式 SQL 示例所示：

```

/// d ##class(PHA.TEST.SQLFunction).ListGet5()
ClassMethod ListGet5()
{
    &sql(
        SELECT $LISTGET('fred', 1, 'failsafe') INTO :b
    )
    if SQLCODE '= 0 {
        w !,"Error code;",SQLCODE
    } else {
        w !,"The non-list element is ",b ; Variable not set
    }
}

```

```
DHC-APP>d ##class(PHA.TEST.SQLFunction).ListGet5()
```

```
Error code;-400
```

如果 position 参数的值小于 -1，则会发生 SQLCODE -400 致命错误，因为 \$LISTGET 返回变量仍未定义。即使提供了默认值也会发生这种情况，如以下嵌入式 SQL 示例所示：

```

/// d ##class(PHA.TEST.SQLFunction).ListGet6()
ClassMethod ListGet6()
{
    s a = $lb("Red", "Blue", "Green")

```

```
&sql(
    SELECT $LISTGET(:a ,-3, 'failsafe') INTO :c
)
if SQLCODE '= 0 {
    w !,"Error code;",SQLCODE
} else {
    w !,"A neg-num position returns ",c ; Variable not set
}
}
```

```
DHC-APP>d ##class(PHA.TEST.SQLFunction).ListGet6()
```

```
Error code;-400
```

当 position 是一个非数字值时，这不会发生：

```
/// d ##class(PHA.TEST.SQLFunction).ListGet7()
ClassMethod ListGet7()
{
    s a = $lb("Red", "Blue", "Green")
    &sql(
        SELECT $LISTGET(:a,'g','failsafe') INTO :c
    )
    if SQLCODE '= 0 {
        w !,"Error code;",SQLCODE
    } else {
        w !,"A nonnumeric position returns ",c
    }
}
}
```

```
DHC-APP>d ##class(PHA.TEST.SQLFunction).ListGet7()
```

```
A nonnumeric position returns failsafe
```

[#SQL #Caché](#)

源

URL:

<https://cn.community.intersystems.com/post/%E7%AC%AC%E5%85%AB%E5%8D%81%E4%BA%94%E7%AB%A0-sql%E5%87%BD%E6%95%B0-listget>