

文章

[姚鑫](#) · 十二月 6, 2021 阅读大约需分钟

第章 SQL谓词 %FIND

第章 SQL谓词 %FIND

使用位图块迭代将一个值匹配到一组生成值。

大纲

```
scalar-expression %FIND valueset [SIZE ((nn))]
```

参数

- scalar-expression - 一个标量表达式(最常见的是表的RowId字段), 它的值正在与值集进行比较。
- valueset - 对用户定义对象的对象引用(oref), 该对象实现位图块迭代方法和ContainsItem()方法。该方法返回一组数据值, 并在与标量表达式中的值匹配时返回一个布尔值。
- SIZE ((nn)) - 可选-用于查询优化的数量级整数(10、100、1000等)。

描述

通过选择与值集中指定的值相匹配的数据值, 通过迭代位图块序列中的值, %FIND谓词允许筛选结果集。当标量表达式的值与valueset中的值匹配时, 此匹配将成功。如果值集值不匹配任何标量表达式值, %FIND返回空字符串。无论显示模式如何, 这个匹配总是在逻辑(内部存储)数据值上执行。

%FIND和其他比较条件一样, 用于SELECT语句的WHERE子句或HAVING子句中。

%FIND使用抽象的、通过编程指定的匹配值集来过滤字段值。具体来说, 它使用抽象的、编程指定的位图来过滤RowId字段值, 其中的值集行为类似于位图索引的标层。

用户定义类派生自抽象类%SQL.AbstractFind。这个抽象类定义了ContainsItem()布尔方法。ContainsItem()方法将标量表达式值与值集值匹配。

使用以下三种方法对位图块序列中的值进行迭代:

- GetChunk(c), 返回块编号为c的位图块。
- NextChunk(.c), 它返回第一个块编号为> c的位图块。
- PreviousChunk(.c), 返回第一个块号< c的位图块。

排序类型

%FIND使用与它匹配的列相同的排序规则类型。默认情况下, 字符串数据类型字段是用SQLUPPER排序规则定义的, 它不区分大小写。

SIZE子句

可选的%FIND SIZE子句提供整数nn，它指定valueset中值数量的数量级估计数。

IRIS使用这个数量级估计来确定最佳查询计划。

指定nn为以文字之一:10、100、1000、10000，等等。

因为nn必须在编译时作为常量值可用，所以在所有SQL代码中必须将其指定为文字。

注意，必须为所有SQL指定嵌套括号，嵌入式SQL除外。

%FIND和%INSET比较

- INSET是最简单和最通用的接口。
它支持ContainsItem()方法。
- %FIND支持使用位图索引对位图块进行迭代。
它模拟了ObjectScript \$ORDER函数的功能，支持NextChunk()、PreviousChunk()和GetChunk()迭代方法，
以及ContainsItem()方法。

[#SQL #Caché](#)

源 URL: <https://cn.community.intersystems.com/post/%E7%AC%AC%E4%B8%83%E7%AB%A0-sql%E8%B0%93%E8%AF%8D-find>