

test

执行条件表达式。

概要

```
test [expr]
```

主要用途

- 执行条件表达式。

参数

文件操作符：

-a FILE	如果文件存在，则为true
-b FILE	如果文件是块特殊的，则为true
-c FILE	如果文件是特殊字符，则为true
-d FILE	如果文件是目录，则为true
-e FILE	如果文件存在，则为true
-f FILE	如果文件存在并且是常规文件，则为true
-g FILE	如果文件是set-group-id则为true
-h FILE	如果文件是符号链接，则为true
-L FILE	如果文件是符号链接，则为true
-k FILE	如果文件的粘滞位sticky设置了，则为true
-p FILE	如果文件是命名管道，则为true
-r FILE	如果您可以读取文件，则为true
-s FILE	如果文件存在且不为空，则为true
-S FILE	如果文件是套接字，则为true
-t FD	如果在终端上打开FD则为True
-u FILE	如果文件是set-user-id则为true
-w FILE	如果文件可写，则为true
-x FILE	如果您可以执行文件，则为true
-O FILE	如果文件有效地归您所有，则为true
-G FILE	如果文件有效地归您的组所有，则为true
-N FILE	如果文件自上次读取以来已被修改，则为true

FILE1 -nt FILE2	根据修改日期，如果 file1 比 file2 新，则为true
FILE1 -ot FILE2	根据修改日期，如果 file1 比 file2 旧，则为true
FILE1 -ef FILE2	如果 file1 为 file2 的硬链接，则为true

字符串运算符：

-z STRING	如果字符串为空，则为true
-n STRING	如果字符串不为空，则为true
STRING	如果字符串不为空，则为true
STRING1 = STRING2	如果字符串相等，则为true
STRING1 != STRING2	如果字符串不相等，则为true
STRING1 < STRING2	如果 STRING1 的字典排序在 STRING2 之前，则为true

STRING1 > STRING2 如果 STRING1 在字典排序在 STRING2 之后，则为true

其他运算符：

-o OPTION	如果启用了shell选项OPTION则为true
-v VAR	如果设置了shell变量VAR则为true
-R VAR	如果设置了shell变量VAR并且是变量引用，则为true
[]EXPR	如果expr为假，则为true
EXPR1 -a EXPR2	如果expr1和expr2都为true则为true
EXPR1 -o EXPR2	如果expr1或expr2为true则为true
arg1 OP arg2	算术表达式测试 OP是 -eq -ne -lt -le -gt -ge 中的一个；算术表达式为真时返回true

返回值

如果表达式执行结果为成功时返回0，当表达式执行结果为失败或给出非法参数时返回1。

例子

```
# 执行条件表达式并显示返回值。
[root@pc root]$ test ! "abc" == 123; echo $?
0

# 等价形式，注意：方括号 [ 后面的空格以及方括号 ] 前面的空格。
[root@pc root]$ [ ! "abc" == 123 ]; echo $?
0

[root@pc root]$ [[ ! "abc" == 123 ]]; echo $?
0
```

注意

1. 该命令等价于 []
2. 编写 bash 条件表达式可用内建命令 test []，组合命令 [[]; > - 关于条件表达式可以查看[这里](#)；> - 关于内建命令的索引可以查看[这里](#)；> - 关于组合命令的索引可以查看[这里](#)
3. 该命令是bash内建命令，相关的帮助信息请查看help命令。

From:
<https://rd.irust.top/> - 学习笔记

Permanent link:
<https://rd.irust.top/doku.php?id=command:test>

Last update: **2021/10/15 14:58**

