

shell中的特殊字符

特殊字符

\$

取值

\$(cal)：相当于执行cal命令，把结果作为字符串原位替换

\${x}：相当于\$x，取x的值

\*

filepath通配符

echo \*：把当前目录所有文件名输出

转义

echo \\$：打印\$

echo \\：打印\

\

换行

在行末加上\，可以把换行符转化为空格

mv \ /path/to/foo \ /path/to/bar  
相当于  
mv /path/to/foo /path/to/bar

`

执行反引号里的命令，用字符串原位替换  
(可以替换为\$0)

`echo "echo 123"`  
打印  
123

单引号

单引号里的特殊字符全部变为普通字符

双引号

双引号里大部分的特殊字符变为普通字符

例外：\$ ` \

echo "\$SHELL"  
打印  
/bin/bash

echo "`date`"  
打印  
Mon Jan 27 13:33:18 CST 2020

子主题

换行符在双引号中就是普通的换行

echo "hello  
world"  
打印  
hello  
world

可以保存原始命令的输出格式

echo \$(cal)  
不加双引号，打印

November 2021 Su Mo Tu We Th Fr Sa 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30

echo "\$(cal)"  
加双引号，打印

November 2021  
Su Mo Tu We Th Fr Sa  
1 2 3 4 5 6  
7 8 9 10 11 12 13  
14 15 16 17 18 19 20  
21 22 23 24 25 26 27  
28 29 30

使用\$获取变量值是否要加双引号?

当我们需要保持变量本身值的完整，不想被空格扩展为多个参数，那么就需要用双引号括起来

例子：

x.sh:

echo \$#

echo \$1

echo \$2

zz="1 2 3"

sh x.sh \$zz

输出

3

1

2

sh x.sh "\$zz"

输出

1

1 2 3