

shell脚本中一些特殊符号Linux认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/644/2021_2022_shell_E8_84_9A_E6_9C_c103_644914.htm 1、 {}大括号：用法一：通配符扩展 eg: ls my_{finger,toe}s 这条命令相当于如下命令的组合：ls

my_fingers my_toes eg: mkdir

{userA,userB,userC}-{home,bin,data}我们将得到 userA-home, userA-bin, userA-data, userB-home,

userB-bin,userB-data,userC-home, userC-bin, userC-data，这几个目录。用法二：可用于语句块的构造，语句之间用回车隔开。

如果你想在某些使用单个语句的地方（比如在AND或OR列表中）使用多条语句，你可以把它们括在花括号{}中来构造一个语句块。 eg: { grep -v "\$cdcatnum" \$strack_file gt. \$strack_file

echo cat -n file1 } (注：以上大括号中的四句命令够成了一个语句块) 用法三：参数扩展 \${name:-default}使用一个默认值（一般是空值）来代替那些空的或者没有赋值的变量name；

\${name:=default}使用指定值来代替空的或者没有赋值的变量name； \${name:?message}如果变量为空或者未赋值，那么就会显示出错误信息并中止脚本的执行同时返回退出码1。

\${#name}给出name的长度 \${name%word} 从name的尾部开始删除与word匹配的最小部分，然后返回剩余部分

\${name%%word} 从name的尾部开始删除与word匹配的最长部分，然后返回剩余部分

\${name#word} 从name的头部开始删除与word匹配的最小部分，然后返回剩余部分

\${name##word} 从name的头部开始删除与word匹配的最长部分，然后返回剩余部分

(注，name为变量名，word为要匹配的字符串) 用法

三在处理字符串和未知变量时，是很有用的。

2、[]中括号：
用法一：通配符扩展：允许匹配方括号中任何一个单个字符
eg: `ls /[eh][to][cm]*` 相当于执行 `ls /etc /home`（若有/eom目录，就相当于会执行`ls /etc /home /eom`）注：在`mkdir`命令下不能扩展
用法二：用于条件判断符号：[]符号可理解为指向`test`命令的一个软链接，所以其用法可完全参照`test`，将`test`位置替换为[]便可。 eg: `if ["$?" != 0]` 等价于 `if test "$?" != 0 then echo "Executes error"`

3、`command` 反引号：`command`
与`$(command)`的含义相同，都是返回当前执行命令的结果eg:
`#!/bin/sh for file in $(ls f*.sh).do lpr $file done exit 0` 该例实现了扩展`f*.sh`给出所有匹配模式的文件的名称。

100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com