目录

[基础 1](#_Toc536025738)

[使用 1](#_Toc536025739)

[版本 2](#_Toc536025740)

[read 2](#_Toc536025741)

[echo 2](#_Toc536025742)

[> 3](#_Toc536025743)

[printf 3](#_Toc536025744)

[流程控制 3](#_Toc536025745)

[调用方式对比 3](#_Toc536025746)

[变量 4](#_Toc536025747)

[传入参数 4](#_Toc536025748)

[系统变量 4](#_Toc536025749)

[字符串 4](#_Toc536025750)

# 基础

## 使用

普通目录下没有执行权限。需要chmod +x ./test.sh

一般以sh后缀，但实际上和后缀没关系。哪怕不用后缀。

-n

### 版本

### read

read name: 读取一行输入赋值给name

一个一个词组地接收输入的参数，每个词组需要使用空格进行分隔；如果输入的词组个数大于需要的参数个数，则多出的词组将被作为整体为最后一个参数接收。

read -p "请输入一段文字:" -n 6 -t 5 -s password

echo "\npassword is $password"

参数说明：

-p 输入提示文字

-n 输入字符长度限制(达到6位，自动结束)

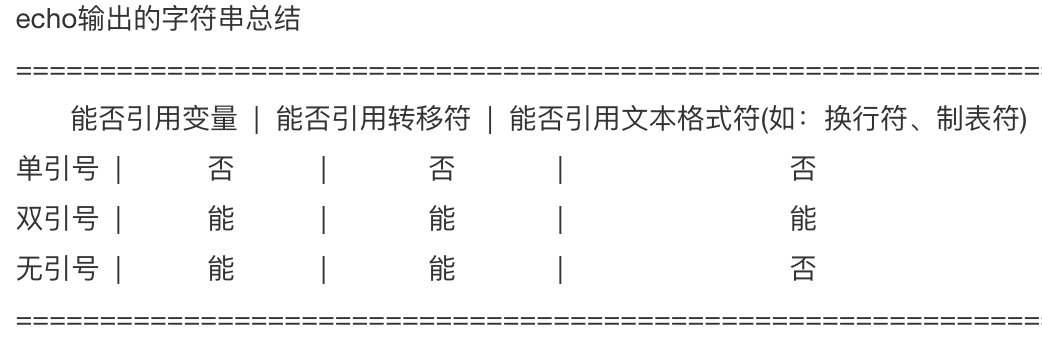
-t 输入限时

-s 隐藏输入内容

### echo

默认会添加换行符

echo -e "OK! \n" # -e 开启转义



### >

### printf

默认 printf 不会像 echo 自动添加换行符，我们可以手动添加 \n。

### 流程控制

sh的流程控制不可为空，如果else分支没有语句执行，就不要写这个else。

### 调用方式对比

1. source命令用法：

source FileName

作用:在当前 bash 环境下读取并执行 FileName 中的命令。该 filename 文件可以无 "执行权限"。

注：该命令通常用命令 . 来替代。

2. sh、bash的命令用法：

sh FileName或bash FileName

作用:打开一个子 shell 来读取并执行 FileName 中命令。该 filename 文件可以无 "执行权限"。

注：运行一个shell脚本时会启动另一个命令解释器。

3、./的命令用法：

./FileName

作用: 打开一个子 shell 来读取并执行 FileName 中命令，该 filename 文件需要 "执行权限"。

注：运行一个 shell 脚本时会启动另一个命令解释器。

## 变量

使用一个定义过的变量，只要在变量名前面加美元符号即可

### 传入参数

$0:文件名

$1,$2。。。变量

### 系统变量

date

echo `date`

### 字符串

拼接

双引号支持${}和"",单引号只支持''

greeting="hello, "$your\_name" !"

greeting\_1="hello, ${your\_name} !"

greeting\_2='hello, '$your\_name' !'

greeting\_3='hello, ${your\_name} !' // 这种不行