

# GNU/Linux Shell 25 问



# GNU/Linux

## Shell 25 问 -- 源自互联网

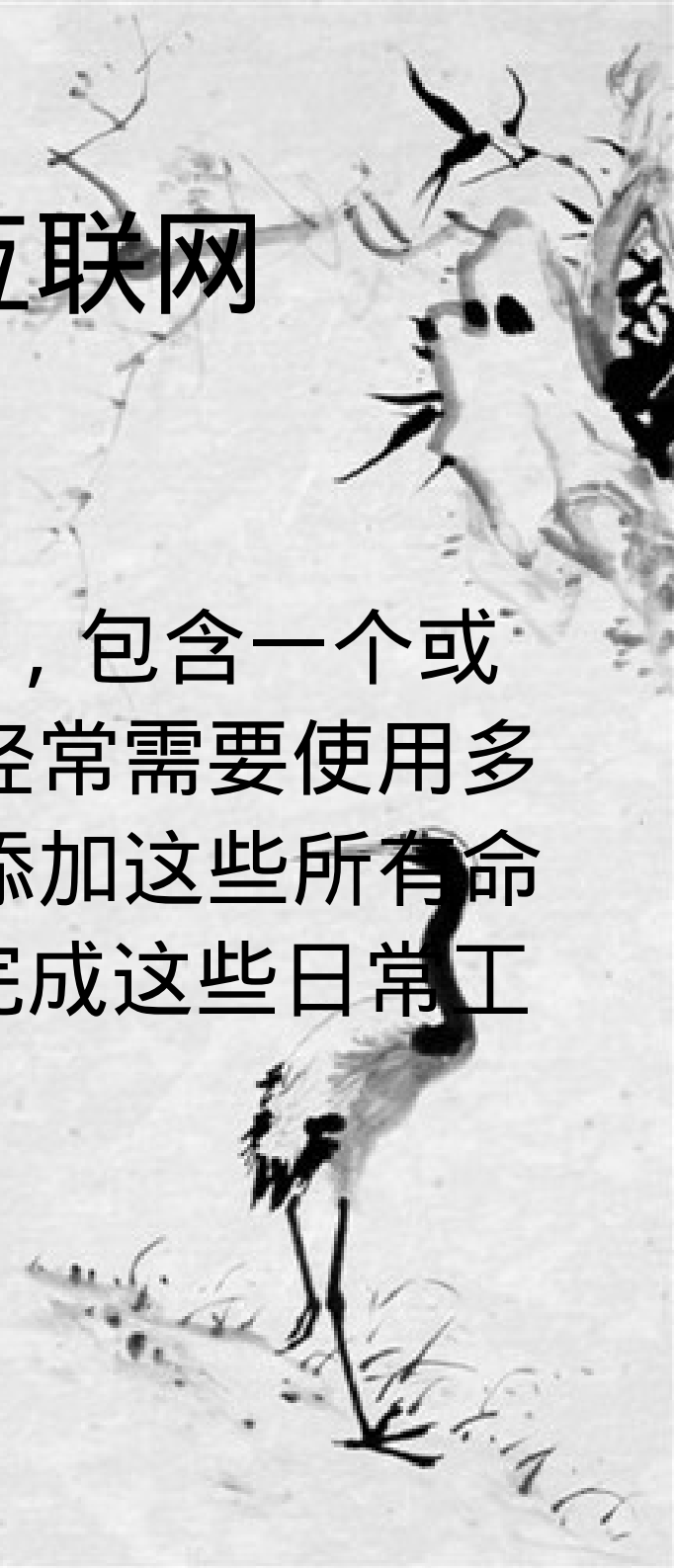
Q:1 Shell 脚本是什么、它是必需的吗？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：一个 Shell 脚本是一个文本文件，包含一个或多个命令。作为系统管理员，我们经常需要使用多个命令来完成一项任务，我们可以添加这些所有命令在一个文本文件 (Shell 脚本) 来完成这些日常工作任务。



# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:2 什么是默认登录 shell ，如何改变指定用户的登录 shell



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：在 Linux 操作系统，“/bin/bash”是默认登录 shell，是在创建用户时分配的。使用 chsh 命令可以改变默认的 shell。示例如下所示：

```
# chsh <用户名>-s <新 shell>  
# chsh linuxtechi -s /bin/sh
```



# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:3 可以在 shell 脚本中使用哪些类型的变量？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：在 shell 脚本，我们可以使用两种类型的变量：

系统定义变量  
用户定义变量

系统变量是由系统系统自己创建的。这些变量通常由大写字母组成，可以通过“set”命令查看。

用户变量由系统用户来生成和定义；变量的值可以通过命令“echo \$< 变量名 >”查看。

# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:4 如何将标准输出和错误输出同时重定向到同一位置？





# GNU/Linux

## Shell 25 问 -- 源自互联网

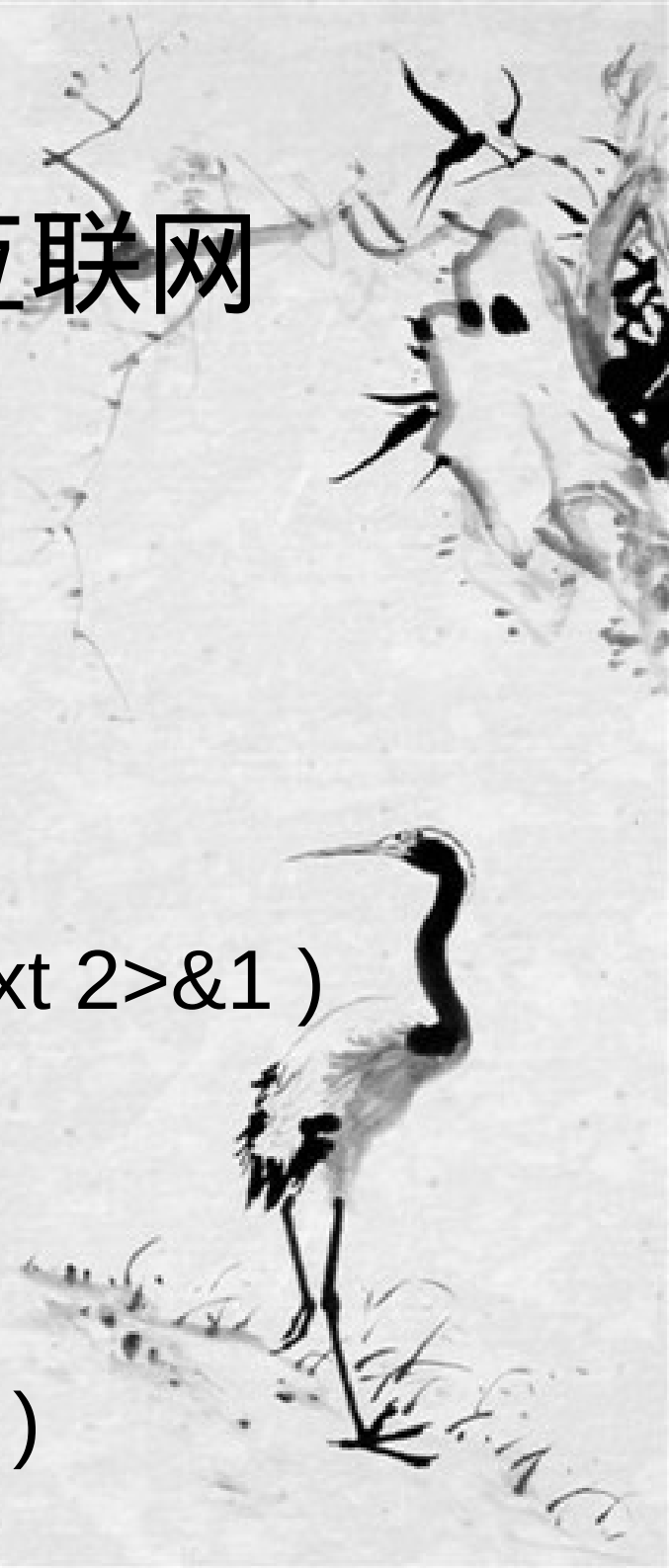
答：这里有两个方法来实现：

方法一：

2>&1 ( 如 # ls /usr/share/doc >out.txt 2>&1 )

方法二：

&>( 如 # ls /usr/share/doc &>out.txt )



# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:5 shell 脚本中“ if” 语法如何嵌套？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：基础语法如下：

```
if [ 条件 ]
```

```
then
```

```
命令 1
```

```
命令 2
```

```
.....
```

```
else
```



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：基础语法如下：

```
if [ 条件 ]
```

```
then
```

```
命令 1
```

```
命令 2
```

```
....
```

```
else
```

```
命令 1
```

```
命令 2
```

```
.....
```

```
fi
```

```
fi
```



# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:6 shell 脚本中“\$?” 标记的用途是什么？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：在写一个 shell 脚本时，如果你想要检查前一命令是否执行成功，在 if 条件中使用“\$?” 可以用来检查前一命令的结束状态。简单的例子如下：

```
root@localhost:~# ls /usr/bin/shar
/usr/bin/shar
root@localhost:~# echo $?
0
```

如果结束状态是 0，说明前一个命令执行成功。

# GNU/Linux

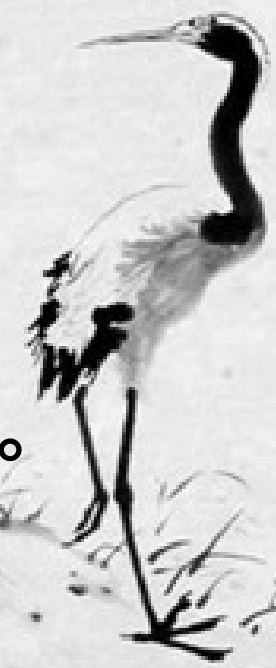
## Shell 25 问 -- 源自互联网

```
root@localhost:~# ls /usr/bin/share
```

```
ls: cannot access /usr/bin/share: No such file or directory
```

```
root@localhost:~# echo $?  
2
```

如果结束状态不是 0 ，说明命令执行失败。



# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:7 在 shell 脚本中如何比较两个数字 ？





# GNU/Linux

## Shell 25 问 -- 源自互联网

答：在 if-then 中使用测试命令（ -gt 等）来比较两个数字，例子如下：

```
#!/bin/bash
```

```
x=10
```

```
y=20
```

```
if [ $x -gt $y ]
```

```
then
```

```
echo "x is greater than y"
```

```
else
```

```
echo "y is greater than x"
```

```
fi
```



# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:8 shell 脚本中 break 命令的作用 ？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：break 命令一个简单的用途是退出执行中的循环。我们可以在 while 和 until 循环中使用 break 命令跳出循环。

# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:9 shell 脚本中 continue 命令的作用 ？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：continue 命令不同于 break 命令，它只跳出当前循环的迭代，而不是整个循环。continue 命令很多时候是很有用的，例如错误发生，但我们依然希望继续执行大循环的时候。

# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:10 告诉我 shell 脚本中 Case 语句的语法 ？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：基础语法如下：

case 变量 in

值 1)

命令 1

命令 2

.....

最后命令

::



# GNU/Linux

## Shell 25 问 -- 源自互联网

值 2)

命令 1

命令 2

.....

最后命令

::

\*)

命令 1

.....

最后命令

esac





# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:11 shell 脚本中 while 循环语法 ？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：如同 for 循环，while 循环只要条件成立就重复它的命令块。不同于 for 循环，while 循环会不断迭代，直到它的条件不为真。基础语法：

```
while [ 条件 ]  
do  
命令...  
done
```



# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:12 如何使脚本可执行 ？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：使用 `chmod` 命令来使脚本可执行。例子如下：

```
# chmod a+x myscript.sh
```



# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:13 “#!/bin/bash” 的作用 ？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：#!/bin/bash 是 shell 脚本的第一行，称为释伴（shebang）行。这里 # 符号叫做 hash，而 ! 叫做 bang。它的意思是命令通过 /bin/bash 来执行。

# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:14 shell 脚本中 for 循环语法 ？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：for 循环的基础语法：

```
for 变量 in 循环列表
do
命令 1
命令 2
....
最后命令
done
```





# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:15 如何调试 shell 脚本 ？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：使用 '-x' 参数 ( `sh -x myscript.sh` ) 可以调试 shell 脚本。另一个种方法是使用 '-nv' 参数 ( `sh -nv myscript.sh` )。

# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:16 shell 脚本如何比较字符串？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：test 命令可以用来比较字符串。测试命令会通过比较字符串中的每一个字符来比较。



# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:17 Bourne shell(bash) 中有哪些特殊的变量 ？



# GNU/Linux

## Shell 25 问 -- 源自互联网

### 内建变量

\$0

\$1

\$2

.....

\$9

\$#

\$\*

\$@

### 解释

命令行中的脚本名字

第一个命令行参数

第二个命令行参数

.....

第九个命令行参数

命令行参数的数量

所有命令行参数，以空格隔开

命令行的参数逐一调用

# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:18 在 shell 脚本中，如何测试文件？



# GNU/Linux

## Shell 25 问 -- 源自互联网

- d 文件名: 如果文件存在并且是目录, 返回 true
- e 文件名: 如果文件存在, 返回 true
- f 文件名: 如果文件存在并且是普通文件, 返回 true
- r 文件名: 如果文件存在并可读, 返回 true
- s 文件名: 如果文件存在并且不为空, 返回 true
- w 文件名: 如果文件存在并可写, 返回 true
- x 文件名: 如果文件存在并可执行, 返回 true



# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:19 在 shell 脚本中，如何写入注释？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：注释可以用来描述一个脚本可以做什么和它是如何工作的。每一行注释以 # 开头。例子如下：

```
#!/bin/bash  
# This is a command  
echo "I am logged in as $USER"
```



# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:20 如何让 shell 就脚本得到来自终端的输入？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：read 命令可以读取来自终端（使用键盘）的数据。read 命令得到用户的输入并置于你给出的变量中。例子如下：

```
#cat /tmp/test.sh
#!/bin/bash
echo 'Please enter your name'
read name
echo "My Name is $name"
# ./test.sh
Please enter your name
LinuxTechi
My Name is LinuxTechi
```



# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:21 如何取消变量或取消变量赋值 ？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：“unset”命令用于取消变量或取消变量赋值。  
语法如下所示：

```
# unset < 变量名 >
```



# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:22 如何执行算术运算 ？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：有两种方法来执行算术运算：

1. 使用 `expr` 命令 ( `# expr 5 + 2` )

2. 用一个美元符号和方括号 ( `$( 表达式 )` ) 例如：  
`test=$(16 + 4);test=$(16 + 4)`



# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:23 do-while 语句的基本格式 ？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：do-while 语句类似于 while 语句，但检查条件语句之前先执行命令（ LCTT 译注：意即至少执行一次。）。下面是用 do-while 语句的语法

```
do  
{  
命令  
} while ( 条件 )
```



# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:24 在 shell 脚本如何定义函数呢 ？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：函数是拥有名字的代码块。当我们定义代码块，我们就可以在我们的脚本调用函数名字，该块就会被执行。示例如下所示：

```
$ diskusage () { df -h ;}
```



# GNU/Linux

## Shell 25 问 -- 源自互联网

函数声明

```
[ function ] 函数名 [()]  
{
```

```
    命令 ;  
    [return int;]
```

```
}
```



# GNU/Linux

## Shell 25 问 -- 源自互联网

Q:25 如何在 shell 脚本中使用 BC ( bash 计算器 ) ？



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：使用下列格式，在 shell 脚本中使用 bc：

```
variable=`echo "options;expression"| bc`
```



# GNU/Linux

## Shell 25 问 -- 源自互联网

答：在 Linux 操作系统，“/bin/bash”是默认登录 shell，是在创建用户时分配的。使用 chsh 命令可以改变默认的 shell。示例如下所示：

```
# chsh <用户名>-s <新 shell>  
# chsh linuxtechi -s /bin/sh
```





# GNU/Linux

## Shell 25 问 -- 源自互联网

答：在 Linux 操作系统，“/bin/bash”是默认登录 shell，是在创建用户时分配的。使用 chsh 命令可以改变默认的 shell。示例如下所示：

```
# chsh <用户名>-s <新 shell>  
# chsh linuxtechi -s /bin/sh
```

