## 使用for循环结构

.For 循环 重复执行命令

例子

For 变量名 in 值列表

Do

命令序列

done

for 变量名 in `ls /etc/\*.conf`

do

命令序列

done

for 变量名 in {1..5}

do

命令序列

done

for 变量名 in `seq 5`

do

命令序列

Done

For 循环基本用法

#！/bin/bash

For i in {1..10}

Do

Echo "hello world"

Done

通过循环批量显示10个hello world

使用 while 循环结构

反复测试 只要成立就执行

While 条件测试

Do

命令序列

Don

案例while循环基本用法

无心的死循环脚本：

· #!/bin/bash

· i=1

· while [ $i -le 5 ]

· do

·         echo "$i"

· done

有效循环脚本：

· #!/bin/bash

· i=1

· while [ $i -le 5 ]

· do

·         echo "$i"

·         let i++

· done

死循环的一般格式：

· #!/bin/bash

· while : 加入这个符号: 死循环

· do

·         echo "hello world"

· done

制作死循环。

#!/bin/bash

· num=$[RANDOM%100+1]

· i=0

· while :

· do

· read -p "随机数1-100,你猜:" guess

· let i++                                    //猜一次，计数器加1，统计猜的次数

· if [ $guess -eq $num ];then

· echo "恭喜，猜对了"

· echo "你猜了$i次"

· exit

· elif [ $guess -gt $num ];then

· echo "猜大了"

· else

· echo "猜小了"

· fi

列出不在线的主机地址

#!/bin/bash

· i=1

· while [ $i -le 254 ]

· do

· IP="192.168.4.$i"

· ping -c 3 -i 0.2 -W 1 $IP &> /dev/null

· if [ $? -eq 0 ] ; then

· echo "Host $IP is up."

· else

· echo "Host $IP is down."

· fi

· let i++

· done

## 基于case分支编写脚本

· case 变量值 in

· 模式1)

· 命令序列1 ;;

· 模式2)

· 命令序列2 ;;

· .. ..

· \*)

脚本编写参考如下：

· [root@svr5 ~]# vim test.sh

· #!/bin/bash

· case $1 in

· redhat)

·         echo "fedora";;

· fedora)

·         echo "redhat";;

·     \*)                                             //默认输出脚本用法

· echo "用法: $0 {redhat|fedora}"

· esac

## 使用Shell函数

## （设置别名的意思 可以使脚本代码更加简洁）

格式1：

1. function 函数名 {
2. 命令序列
3. .. ..
4. }

· 格式2：

1. 函数名() {
2. 命令序列
3. .. ..
4. }

注意：函数的定义语句必须出现在调用之前，否则无法执行。

参考 脚本

1. [root@svr5 ~]# mycd(){                        //定义函数
2. > mkdir /test
3. > cd /test
4. > }
5. [root@svr5 ~]# mycd                            //调用函数
6. [root@svr5 ~]# mycd(){                        //定义函数
7. > mkdir $1
8. > cd $1
9. > }
10. [root@svr5 ~]# mycd /abc                            //调用函数
11. [root@svr5 ~]# mycd /360                            //调用函数

· #!/bin/bash

· cecho() {

· echo –e "\033[$1m$2\033[0m"

· }

#!/bin/bash

i=0

while [ $i -le 20 ]

do

let i++

[ $[i%6] -ne 0 ] && continue

echo $[i\*i]

done

exit 2· cecho 32 OK

· cecho 33 OK

· cecho 34 OK

· cecho 35 OK

给cecho 赋值

## 中断及退出

· #!/bin/bash

· for i in {1..5}

· do

·      [ $i -eq 3 ]&& break //这里将break替换为continue，exit分别测试脚本执行效果     echo $i

· done

· echo "Game Over"

· #!/bin/bash

· while read -p "请输入待累加的整数（0表示结束）：" x

· do

· [ $x -eq 0 ] && break

· SUM=$[SUM+x]

· done

· echo "总和是：$SUM"