

9 while和until循环

while 和 until 循环

一、while 循环语法结构

`while` 循环语句主要用来重复执行一组命令或语句，常用于**守护进程**或**持续运行的程序**，其中循环的**次数既可以是固定的，也可以不固定**（用户自主控制）。

`while` 循环语句的基本语法如下。

```
1 while 条件测试
2 do
3     循环体
4 done
```

Shell | 复制代码

`while` 循环语句会对**条件测试**进行判断，如果条件测试**成立**时，则执行 `do` 和 `done` 之间的**循环体**，直到条件测试**不成立**时才**停止循环**。

案例：`while` 循环输出数字

```
1 [root@Shell]# vi while1.sh
2 #!/bin/bash
3 counter=1
4 while [[ $counter -le 5 ]]
5 do
6     echo $counter
7     let "counter+= 1"
8 done
9 [root@Shell]# . while1.sh
10 1
11 2
12 3
13 4
14 5
```

Shell | 复制代码

```
1 [root@Shell]# vi while2.sh
2 #!/bin/bash
3 counter=5
4 while [[ $counter -gt 0 ]]
5 do
6     echo $counter
7     let "counter-=1"
8 done
9 [root@Shell]# . while2.sh
10 5
11 4
12 3
13 2
14 1
```

注意：退出条件，往往配合自增或自减表达式。

二、until 循环语法结构

until 循环语句的基本语法格式如下。

```
1 until 条件测试
2 do
3     循环体
4 done
```

until 循环语句的用法与 while 循环语句的用法恰恰相反，until 循环语句是在条件表达式不成立时，执行循环体，条件表达式成立时，终止循环。

案例：until 循环输出数字

```

1 ▾ [root@Shell ~]# vi until1.sh
2   #!/bin/bash
3   counter=1
4   ▾ until [[ $counter -gt 5 ]]
5   do
6       echo $counter
7       let "counter+=1"
8   done
9   ▾ [root@Shell ~]# . until1.sh
10  1
11  2
12  3
13  4
14  5

```

三、while 和 until 循环实例

3.1 案例：while 循环语句实现批量用户创建

批量创建用户除了 for 循环语句可以实现外，while 循环语句也可以实现。

```

1 [root@Shell ~]# cat user1.txt
2 abcd 123
3 qwer 123
4 [root@Shell ~]# vi while3.sh
5 #!/usr/bin/bash
6 # 逐行读取user1.txt文件内容
7 while read line
8 do
9     if [ ${#line} -eq 0 ];then
10         echo "-----"
11         continue
12     fi
13     user=`echo $line|awk '{print $1}'`
14     pass=`echo $line|awk '{print $2}'`
15     id $user &>/dev/null
16     if [ $? -eq 0 ];then
17         echo "user $user already exists"
18     else
19         useradd $user
20         echo "$pass" |passwd --stdin $user &>/dev/null
21         if [ $? -eq 0 ];then
22             echo "$user is created."
23         fi
24     fi
25 done < user1.txt
26 [root@Shell ~]# . while3.sh
27 abcd is created.
28 qwer is created.

```

⚠️ 以下结构可避免设置字段分隔符 IFS 。

```

1 while read 临时变量
2 do
3     代码块
4 done < 文件

```

3.2 案例：while 循环语句和 until 循环语句测试远程主机连接

检测主机存活，ping 通时无限循环休眠，ping 不通时退出循环输出字符串。

```
1  # 条件为真无限循环
2  [root@Shell ~]# vi while_ping.sh
3  #!/bin/bash
4  ip=127.0.0.1
5  while ping -c1 -W1 $ip &>/dev/null
6  do
7      sleep 1
8  done
9  echo "$ip is down!"
10 [root@Shell ~]# . while_ping.sh
11 ^C
12 # 条件为假, 不执行循环
13 [root@Shell ~]# vi while_ping.sh
14 #!/bin/bash
15 ip=128.0.0.1
16 while ping -c1 -W1 $ip &>/dev/null
17 do
18     sleep 1
19 done
20 echo "$ip is down!"
21 [root@Shell ~]# . while_ping.sh
22 128.0.0.1 is down!
```

⚠ **while** 循环条件判断是否为真, 如果为**真**则**一直循环**, 否则停止循环。
而 **until** 循环刚好和 **while** 相反, 条件判断为**假**, 就**一直循环**。

until 一直 **ping**, **ping** 不通继续 **ping**, **ping** 通则退出循环。

```
1 [root@Shell ~]# vi until_ping.sh
2 #!/bin/bash
3 ip=127.0.0.1
4 until ping -c1 -W1 $ip &>/dev/null
5 do
6     sleep 1
7 done
8 echo "$ip is up!"
9 [root@Shell ~]# . until_ping.sh
10 127.0.0.1 is up!
11 [root@Shell ~]# vi until_ping.sh
12 #!/bin/bash
13 ip=128.0.0.1
14 until ping -c1 -W1 $ip &>/dev/null
15 do
16     sleep 1
17 done
18 echo "$ip is up!"
19 [root@Shell ~]# . until_ping.sh
20 ^C
```

3.3 案例：for、while、until 终极对决

接下来分别用 for、while 和 until 写 ping 通主机的脚本。

```
1 #!/bin/bash
2 for i in {1..5}
3 do
4     ip=192.168.149.${i}
5     # ping主机, 结果不在屏幕输出
6     ping -c1 -W1 $ip &>/dev/null
7     # 如果ping通输出ip
8     if [ $? -eq 0 ];then
9         echo "$ip"
10    fi
11 done
12 echo "finished..."
```

```
1  #!/bin/bash
2  i=1
3  while [ $i -le 5 ]
4  do
5      ip=192.168.149.${i}
6      # ping主机, 结果不在屏幕输出
7      ping -c1 -W1 $ip &>/dev/null
8      # 如果ping通输出ip
9      if [ $? -eq 0 ];then
10         echo "$ip"
11     fi
12     let i++
13 done
14 echo "finished..."
```

```
1  #!/bin/bash
2  i=1
3  until [ $i -gt 5 ]
4  do
5      ip=192.168.68.${i}
6      # ping主机, 结果不在屏幕输出
7      ping -c1 -W1 $ip &>/dev/null
8      # 如果ping通输出ip
9      if [ $? -eq 0 ];then
10         echo "$ip"
11     fi
12     let i++
13 done
14 echo "finished..."
```

通过以上三个脚本可以看出：

- `for` 循环用于**循环次数固定**的情形，常常用于正常的循环处理中。
- `while` 循环的特长是执行**守护进程**，以及实现**循环持续执行不退出**的应用。
- `until` 循环用于只要**条件测试语句为假**时，执行语句块；如果一开始条件测试语句就为真，则一次也不执行语句块。

小结

- `while` 循环：语法格式、运行逻辑

- `until` 循环：语法格式、运行逻辑

课程目标

- 知识目标：熟练掌握while和until循环语句的基本语法。
- 技能目标：能够根据实际需求利用while和until循环语句实现循环流程控制。

课外拓展

- 进一步了解while和until循环语句的应用场景。

参考资料

- `help while` `help until`
- 《Linux Shell核心编程指南》，丁明一，电子工业出版社