

shell 基础强化笔记

作者: [yuanhenglizhen](#)

原文链接: <https://ld246.com/article/1575020523006>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

1. 循环相关

1.1. 单条件循环

判断文件或目录是否存在，使用-e就够了，例如：

```
if [ -e /etc/hosts ];then echo "纯在"; fi
```

- -d 判断目录
- -e 判断文件或目录
- -f 判断文件
- -r 判断可读
- -w 判断可写
- -x 判断可执行

1.2. 整数判断

- -ne 判断不相等
- -eq 判断相等
- -gt 判断大于
- -lt 判断小于
- -ge 判断大于等于
- -le 判断小于等于

1.3. 判断程序服务是否正在运行

```
#!/bin/bash
if [ $# -ne 1 ];then
    echo "please input at least one param!"
    exit
fi
systemctl status "$1" &>/dev/null
rc=$?
if [ $rc -eq 0 ];then
    echo "$1 is running!"
elif [ $rc -eq 4 ];then
    echo "$1 is not install!"
else
    echo "$1 is not runing!"
fi
```

这里需要说明的一个地方就是，4表示未安装服务，127表示没有那个命令

1.4. 多整数判断比较

```
[ 1 -lt 2 -a 5 -gt 10 ];$?
```

```
>> 1
[ 1 -lt 2 -o 5 -gt 10 ];$?
>> 0
[[ 1 -lt 2 && 5 -gt 10 ]];$?
>> 1
[[ 1 -lt 2 || 5 -gt 10 ]];$?
>> 0
```

这边需要注意的一个问题就是，如果你用&&符号那么你一定要是用双中括号,反之亦然

1.5. 正则匹配判断

[["\$grade" =~ ^[0-9]+\$]] +号表示匹配多个

```
#!/bin/bash

# 获取学生输入成绩
read -p "输入你的考试成绩: " grade
[[ "$grade" =~ ^[0-9]+$ ]]
if [ $? -ne 0 ];then
    echo "请好好输入! "
    exit
fi
# 判断学生成绩在哪一个等级范围

if [ $grade -lt 60 ];then
    echo "滚去补考! "
elif [ $grade -ge 60 -a $grade -lt 80 ];then
    echo "你的考试成绩合格"
elif [ $grade -ge 80 -a $grade -le 100 ];then
    echo "你真棒:+1:"
else
    echo "瘪犊子, 好好输入你的成绩"
fi
```

1.6. 作业实战

1.6.1. 日志清理

使用root用户清理/var/log/messages,并且保留100行最新日志

提示: 1.先判断当前用户是否为root

- 2.判断文件是否存在
- 3.清空并保留最新100行

```
#!/bin/bash

logfile_path="/var/log/messages"
if [ $UID -eq 0 ];then
    if [ -f "$logfile_path" ];then
        tail -n 100 $logfile_path > $logfile_path.bak
```

```

    cat $logfile_path.bak > $logfile_path
    rm -rf $logfile_path.bak
    echo "删除成功并保留最近一百行"
    echo `cat $logfile_path`
else
    echo "日志文件不存在"
fi
else
    echo "请使用root用户操作!"
fi

```

1.6.2. 判断某进程运行

```
systemctl status docker |awk '/^.*Active:/{print $2}'
```

awk 后面接着是", 单引号里面是模式+空格+{print \$xxx}

我理解的模式就是正则匹配公式, 然后可以不写

1.6.3. 获取服务器所有用户

```
cat /etc/passwd |awk -F: '{print $1}'
```

1.6.4 备份所有的数据库

□ 每个库一个.sql文件, 记得排出没用的

1.拿到所有的数据库

2.一个一个的备份的数据库

```
mysql -u root -pxxx -h163.254.0.221 -e 'show databases'|sed 1d |grep -Ev "sys|mysql|*_schema"|sed -r 's#(.*)# mysqldump -u root -pxxx -h163.254.0.221 -B \1 > \1.sql #g'|bash
```

这个shell简直了, 当然正常的备份还是for循环去做比较好

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

```

cat <<eof
*****
**      你好      **
**      我好      **
**      大家好    **
*****
eof

```

2. case循环

2.1 rsync启停脚本

1.如何启动命令 rsync --daemon

2.如何停止pkill rsync

3.脚本名称rsync_daemon

```
#!/bin/bash
source /etc/init.d/functions
rs=$1
case $rs in
  start)
    if [ ! -f /var/run/rsync.pid ];then
      touch /var/run/rsync.pid
      rsync --daemon
      action "Rsync starting ....." /bin/true
    else
      action "Service Rsync Runing ....." /bin/false
    fi
    ;;
  stop)
    if [ -f /var/run/rsync.pid ];then
      rm -f /var/run/rsync.pid
      pkill rsync
      action "Rsync stoping ....." /bin/true
    else
      action "Service Rsync Stoping ....." /bin/false
    fi
    ;;
  restart)
    if [ -f /var/run/rsync.pid ];then
      rm -f /var/run/rsync.pid
      pkill rsync
      action "Rsync stoping ....." /bin/true
    else
      action "Service Rsync Stoping ....." /bin/false
    fi
    if [ ! -f /var/run/rsync.pid ];then
      touch /var/run/rsync.pid
      rsync --daemon
      action "Rsync starting ....." /bin/true
    else
      action "Service Rsync Runing ....." /bin/false
    fi
    ;;
  status)
    status=`ps -ef |grep rsync|grep -v pts |grep -v grep |awk '{print $2}`
    if [ ! -z "$status" ];then
      echo "Rsync is active [$status]"
    else
      echo "Rsync is inactive"
    fi
    ;;
  *)
    echo "USAGE: $0 {start|stop|restart|status}"
    exit
esac
```

rsync启停脚本

2.2 nginx启停脚本

```
#!/bin/bash
source /etc/init.d/functions
#加锁
if [ -f /tmp/nginx.lock ];then
    echo "此程序正在运行"
fi
touch /tmp/nginx.lock

rs=$1
case $rs in
    start)
        if [ ! -f /var/run/nginx.pid ];then
            touch /var/run/nginx.pid
            /usr/sbin/nginx
            action "nginx starting ....." /bin/true
        else
            action "Service nginx Runing ....." /bin/false
        fi
        ;;
    stop)
        if [ -f /var/run/nginx.pid ];then
            /usr/sbin/nginx -s stop
            action "nginx stoping ....." /bin/true
        else
            action "Service nginx Stopping ....." /bin/false
        fi
        ;;
    restart)
        /usr/sbin/nginx -s reload
        action "nginx restart ....." /bin/true
        ;;
    status)
        status=`ps -ef |grep nginx|grep -v pts |grep -v grep |awk '{print $2}`
        if [ ! -z "$status" ];then
            echo "nginx is active [$status]"
        else
            echo "nginx is inactive"
        fi
        ;;
    *)
        echo "USAGE: $0 {start|stop|restart|status}"
        exit
esac
#解锁
rm -f /tmp/nginx.lock
```

2.3 实现系统管理箱

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

```
cat << eof
=====
h. 显示系统帮助
f. 显示磁盘分区
d. 显示磁盘挂载
m. 查看内存使用
u. 查看系统负载
q. 退出此工具箱
=====
eof
while true
do
read -p "输入你需要的操作: " sys
case $sys in
  h)
    ;;
  f)
    df -h
    ;;
  d)
    mount -a||less
    ;;
  m)
    free -m
    ;;
  u)
    w
    ;;
  q)
    exit
    ;;
  *)
    echo "USAGE: $0 [ h | f | d | m | u | q]"
esac
done
```