1.输出当前的日期，时间，用户名和当前工作目录

echo "Current date is `date`"

echo "User is `whoami`"

echo "Current directory `pwd`"

2.数字的运算

expr :

echo `expr $v1 + $v2`

echo `expr $v1 - $v2`

echo `expr $v1 \\* $v2`

echo `expr $v1 / $v2`

echo `expr $v1 % $v2`

bc:

echo “$v1+$v2” | bc

echo “scale=2;$v1+$v2” | bc #保留两位小数

echo “obase=2;ibase=10;$v1” | bc #十进制转二进制

echo “obase=10;ibase=2;$v1” | bc #二进制转十进制

echo “10^2” | bc #乘方

echo “sqrt(100)” | bc #平方根

3.数字或字符串逆序输出

#从最右端依次截取

v1=$1

len=${#v1}

while [ $len -ge 1 ]

do

rev="$rev""${v1:len-1:1}" #拼接字符串

let "len--"

done

echo $rev

#数字还可以采用除留余数法

rev=0

while [ $v1 -gt 0 ]

do

b=`expr $v1 % 10`

v1=`expr $v1 / 10`

rev=`expr $rev \\* 10 + $b`

done

echo $rev

4.获取随机字符串或数字

随机获取8位字符

echo $RANDOM | md5sum | cut -c 1-8

echo /dev/urandom | tr -dc [:alnum:] | cut -c 1-8

随机获取8位数字

echo $RANDOM | cksum | cut -c 1-8

echo /dev/urandom | tr -dc "1-8" | head -c 8 #8个数字

date +%N | cut –c 1-8

5. 批量创建用户

#!/bin/bash

DATE=$(date +%F\_%T)

USER\_FILE=user.txt

if [ -s $USER\_FILE ]

then

mv $USER\_FILE ${USER\_FILE}-${DATE}.bak

echo "$USER\_FILE exist, rename ${USER\_FILE}-${DATE}.bak"

fi

echo -e "User\tPassword" >> $USER\_FILE

echo "-----------------" >> $USER\_FILE

for USER in user{1..10}

do

if ! id $USER &> /dev/null

then

PASS=$(echo $RANDOM | md5sum | cut -c 1-8)

useradd $USER

echo $PASS | passwd --stdin $USER &> /dev/null

echo -e "$USER\tPASS" >> $USER\_FILE

echo "$USER User creat successful."

else

echo "$USER user already exists!"

fi

done

6. 检查软件包是否安装

#!/bin/bash

if rpm -q sysstat &> /dev/null

then

echo “sysstat is already installed.”

else

echo “sysstat is not installed.”

fi

7. 检查主机存活状态

#!/bin/bash

ping\_status() {

if ping -c 1 $1 >/dev/null

then

echo "$1 ping is successful."

continue

fi

}

IP\_List="10.13.4.18 10.13.4.201 10.13.4.110"

for ip in $IP\_List

do

ping\_status ip

ping\_status ip

ping\_status ip

echo "$ip ping is failure!"

done

8. 监控CPU、内存和硬盘利用率

CPU:借助vmstat工具

#!/bin/bash

DATE=$(date +%F” “%H:%M)

if ! which vmstat &>/dev/null

then

echo “vmstat command no found, please install”

exit 1

fi

US=$(vmstat | awk 'NR==3{print $13}')

SY=$(vmstat | awk 'NR==3{print $14}')

USE=`expr $US + $SY`

if [ $USE -ge 0 ]

then

echo "

date: $DATE

problem: CPU usage $USE"

fi

内存:

DATE=$(date +%F" "%H:%M)

USE=$(free -m | awk '/Mem/{print $3-$6-$7}')

TOTAL=$(free -m | awk '/Mem/{print $2}')

FREE=`expr $TOTAL - $USE`

if [ $FREE -le 4096 ]

then

echo "

date: $DATE

problem: total=$TOTAL, use=$USE, free=$FREE."

fi

9. 检查网站可用性

n/bash

check\_url1() {

HTTP=$(curl -o /dev/null --connect-timeout 3 -s -w "%{http\_code}" $1)

if [ $HTTP -ne 200 ]

then

echo "warning : $1 access failure!"

fi

}

check\_url2() {

if ! wget -T 10 -tries=1 -spider $1 >/dev/null 2>&1

then

echo "warning : $1 access failure!"

fi

}

check\_url1 $1

check\_url2 $1

10.清理本机除当前登录用户以外的所有用户

**11.**使用for循环在/oldboy目录下通过随机小写10个字母加固定字符串oldboy批量创建10个html文件

#!/bin/bash

WORK\_DIR=/oldboy/

create(){

i=1

while [ $i -lt 11 ]

do

cd $WORK\_DIR && touch `tr -dc "a-z"</dev/urandom | head -c 10`\_oldboy.html

i=$(($i+1))

done

}

check(){

if [ -d $WORK\_DIR ];

then

create

else

mkdir $WORK\_DIR

create

fi

}

check

**12.请用至少两种方法实现:** 将以上文件名中的oldboy全部改成oldgirl (用for循环实现),并且html改成大写。

#!/bin/bash

change\_name(){

DIR=/oldboy

FILE=`ls /oldboy`

GIRL=\_oldgirl.HTML

for i in $FILE

do

c=`echo $i | cut -c 1-10` #方法1

c=`echo $i | awk -F '\_' '{print $1}'` #方法2

mv $DIR/$c\* $DIR/$c$GIRL

done

}

change\_name

13.批量创建10个系统帐号oldboy01-oldboy10并设置密码（密码为随机8位字符串）。

#!/bin/bash

USER=oldboy

NUM=” 01 02 03 04 05 06 07 08 09 10”

create\_user(){

for i in $NUM

do

RPASSWD=$(tr -dc [a-zA-Z] < /dev/urandom | head -c 8)

useradd $USER$i

echo $RPASSWD|passwd $USER$i --stdin

echo $USER$i----$RPASSWD >> /oldboy/userpasswd.txt

done

}

del\_user(){

for j in $NUM

do

userdel $USER$j

rm -rf /home/$USER$j

done

}

case $1 in

create)

create\_user

;;

del)

del\_user

;;

\*)

echo "Usage:Please use $0 create or $0 del."

echo "####################################"

echo "create: create user $USER 01-10 and give it random passwd."

echo "del: delete $USER 01-10 and /home/$USER 01-10 directory. "

;;

esac

14.for循环打印下面这句话中字母数不大于6的单词。  
I am oldboy teacher welcome to oldboy training class.

#!/bin/bash

echo '###Usage1#####'

a='I am oldboy teacher welcome to oldboy training class.'

for i in $a

do

NUM=`echo $i |wc -L`

if [ $NUM -le 6 ];then

echo $i

fi

done

echo '###Usage2######'

for i in $a

do

if [ ${#i} -le 6 ];then

echo $i

fi

done

15.打印选择菜单，一键安装Web服务:

[root@oldboyscripts]# sh menu.sh

    1.[install lamp]

    2.[install lnmp]

    3.[exit]

    pls input the num you want:

要求：

1、当用户输入1时，输出“startinstalling lamp.”然后执行/server/scripts/lamp.sh，脚本内容输出"lampis installed"后退出脚本；

2、当用户输入2时，输出“startinstalling lnmp.”然后执行/server/scripts/lnmp.sh输出"lnmpis installed"后退出脚本;

3、当输入3时，退出当前菜单及脚本；

4、当输入任何其它字符，给出提示“Input error”后退出脚本。

5、要对执行的脚本进行相关条件判断，例如：脚本是否存在，是否可执行等。

#!/bin/bash

echo "1.[install lamp]

2.[install lnmp]"

3.[exit]"

read -t 30 -p "Please input tut the num you want:" a

case $a in

1)

if [ -f lamp.sh ];then

./lamp.sh

sleep 3

echo "lamp is installed!"

else

echo "no lamp.sh!"

fi

;;

2)

if [ -f lnmp.sh ];then

./lnmp.sh

sleep 3

echo "lnmp is installed!"

else

echo "no lnmp.sh!"

fi

;;

3)

exit 0

;;

\*)

echo "Input error"

exit 0

esac

16.按行读取文件内容

#!/bin/bash

#while循环, 效率最高

readbyline() {

while read line

do

echo $line

done < $1

}

readbyline $1

#管道法

readbyline2() {

cat $1 | while read line

do

echo $line

done

}

readbyline2 $1

#for循环,除了会按行输出,每行中会按空格分隔输出

readbyline3() {

for line in `cat $1`

do

echo $line

done

}

readbyline3 $1