Shell脚本编程入门

Shell编程语法：

1.脚本基本格式

#!/bin/bash

# 注释内容

Shell编程语句

2.脚本执行方式

(1)chmod a+x xx.sh

(2) bash xxx.sh

(3)source xxx.sh

(4) .xxx.sh

3.shell程序结构

(1)顺序结构

(2)选择结构

(3)循环结构

任何一种复杂的程序都是由这三个基本结构叠加组合而来

程序=算法+数据结构：数据结构用于描述一个对象，算法教对象如何做事。

学习编程语言，最重要的是要学习它的语言特性，它适用于什么场景。

1. 选择结构注意事项

if 和fi成对出现

if后面紧跟then

elif 后面紧跟then

else 后面不要跟then

if后面的[]中的条件和[]之间要有空格，比如if [ xxxx ]

case语句中每一个模式（匹配）的结束 标志“;;”

case语句中默认匹配模式 \*)，case语句最后采用esac作为结束标志

5.循环结构注意事项

for语句中采用do…done包含循环体

for语句别忘记(())，双括号

for语句可以“离散”控制执行也可以“连续”控制执行

while语句还可以用于“读取一行，处理一行”

until语句中，条件满足，程序停止循环

循环语句中，注意continue和break区别，continue结束当前循环，进入下一次循环；break语句直接跳出循环体，并结束循环。

1. 函数的使用

模块化设计思想。参见P157菜单实例。

编写shell脚本，可以结合先前所学内容：基本命令、重定向、管道、序列号、AWK/SED、位置变量使用，循环/选择/顺序结构。

敲代码，千万不要从第一行敲到最后一行，一定要先敲框架，然后填充细节。

Shell函数传递参数（通过位置变量进行）。

可以参考源码（书本《运维之道》第2版）：

菜单实例(在P154案例基础上，结合函数思想) P157

位置变量实例 P111

Shell脚本编程实例（一）：

#!/bin/bash

#Date:2018-03-25

#Version:0.01

#Author:ZhangPeng

echo "this is the first shell script"

echo "generate files"

for name in {1..100}.txt

do

touch $name

done

echo "analyze documentation information"

file\_num=`ls -la | grep "^-"|wc -l`

echo "the num of file is $file\_num"

folder\_num=`ls -la | grep "^d"|wc -l`

echo "the num of folder is $folder\_num"

docu\_size=`du -sh . | awk '{print $1}'`

echo "the size of documentation is $docu\_size"

echo "backup files start..."

backup\_date=`date | awk '{print $6$2$3}'`

backup\_path=$backup\_date$RANDOM

backup\_name=$backup\_path+tar.gz

mkdir /$backup\_path

cp -r . /$backup\_path

cd /

tar -czf $backup\_name /$backup\_path

echo "backip files end"

exit 0

Shell脚本编程实例（二）：

#!/bin/bash

#Date:2018-03-25

#Version:0.01

#Author:ZhangPeng

# this program is used to manage linux system

echo "this is the second shell script"

echo "generate some user"

#need root

if [ "$(id -u)" == "0" ]

then

echo "under root"

else

echo "u need root,please input root passwd"

su

fi

echo "add some user,passwd see user2passwd"

for USER\_NAME in Ali Bob Carl Dave Eden Frank

do

useradd $USER\_NAME

PASSWD=$RANDOM

echo $USER\_NAME >>user2passwd

echo $PASSWD >>user2passwd

echo $PASSWD | passwd --stdin $USER\_NAME

done

echo "software management"

rpm -qai > software\_info

yum list > yum\_list

## 后续可以对上述两个文件进行相关信息提取

exit 0

Shell脚本编程实例（三）：

#!/bin/bash

#modify configuration files

#Date:2018-04-10

#Author:zp

FILE\_CONF='./hehe.conf'

# 单引号和双引号不一样，单引号屏蔽一切特殊符号

TMP\_FILE="./tmp.$$"

# make anonymous\_enable=yes

sed 's/^.\*anonymous\_enable=.\*/anonymous\_enable=YES/' $FILE\_CONF > $TMP\_FILE

echo $TMP\_FILE

mv -f $TMP\_FILE $FILE\_CONF

Shell脚本编程实例（四）

#!/bin/bash

#编写一个菜单脚本，里面融合函数定义、case案例，参数传递

HINT(){

read -p "Press Enter to continue:"

}

CPU\_INFO(){

echo "run cpu\_info"

echo "system working time"

#uptime -p | awk 'BEGIN{OFS="-"}{print $2,$3}'

uptime -p | awk '{print $2,$3,$4,$5}'| awk -F, '{print $1,$2}'| awk 'BEGIN{OFS="-"}{print $1,$2,$3,$4}'

echo "system load average:"

#uptime | awk -F, '{print $3,$4,$5}' | awk -F: '{print $2}'

uptime | awk -F: '{print $5}'

echo "user numbers:"

uptime | awk -F, '{print $2}'

}

MEM\_INFO(){

echo "run mem\_info"

USED\_MEM=`free -m | awk '/Mem/{print $3}' `

echo "used-mem : $USED\_MEM M"

TOTAL\_MEM=`free -m | awk '/Mem/{print $2}'`

echo "total-mem : $TOTAL\_MEM M"

REMAIN\_MEM=$((TOTAL\_MEM-USED\_MEM))

echo "remain-mem : $REMAIN\_MEM M"

}

PROGRESS\_INFO(){

echo "run progress\_info"

echo "the num of progress:"

NUM=$(ps -aux | wc -l)

PROGRESS\_NUM=$((NUM - 1))

echo "$PROGRESS\_NUM"

echo "the num of runing progress:"

ps -aux | awk '{print $8}' | grep R | wc -l

}

while true

do

clear

echo "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"

echo "1.Display CPU info:"

echo "2.Display Mem info:"

echo "3.Display Progress info:"

echo "4.Exit Script:"

echo "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"

read -p "please select an iterm(1-4):" U\_SELECT

case $U\_SELECT in

1)

CPU\_INFO

HINT

;;

2)

MEM\_INFO

HINT

;;

3)

PROGRESS\_INFO

HINT

;;

4)

exit

;;

\*)

read -p "please select 1-4,press enter to continue:"

esac

done

Shell脚本编程实例（五）

#!/bin/bash

#这是一个测试脚本，用于测试CPU 内存 磁盘空间占用

#Date:2018.04.12

#Author:zp

HINT(){

read -p "Press Enter to continue:"

}

TEST\_CPU(){

echo "run test\_cpu"

for((i=0;i<50000;i++))

do

cd /tmp

echo "generate directory"

PATH1=$RANDOM

mkdir ./test2test$PATH1

cd ./test2test$PATH1

echo "generate file"

echo "heheda$RANDOM" >nihao$PATH1

echo "delete directory and file"

rm -rf /tmp/test2test\*

done

}

TEST\_MEM(){

echo "run test\_mem"

}

TEST\_DISK(){

echo "run test\_disk"

for ((i=0;i<2000;i++))

do

echo "generate $i 300M"

dd if=/dev/zero of=tmp bs=1M count=300

rm -f ./tmp

echo "delete $i 300M"

done

}

while true

do

clear

echo "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"

echo "1.TEST CPU:"

echo "2.TEST MEM:"

echo "3.TEST DISK:"

echo "4.Exit Script:"

echo "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"

read -p "please select an iterm(1-4):" U\_SELECT

case $U\_SELECT in

1)

TEST\_CPU

HINT

;;

2)

TEST\_MEM

HINT

;;

3)

TEST\_DISK

HINT

;;

4)

exit

;;

\*)

read -p "please select 1-4,press enter to continue:"

esac

done

Shell编程相关命令汇总：

<http://man.linuxde.net/shell-script>

[终端打印、算术运算、常用变量](http://man.linuxde.net/shell-script/shell-1" \t "http://man.linuxde.net/_blank)

从键盘或文件中获取标准输入：[read命令](http://man.linuxde.net/read" \t "http://man.linuxde.net/_blank)

[文件的描述符和重定向](http://man.linuxde.net/shell-script/shell-2" \t "http://man.linuxde.net/_blank)

[数组、关联数组和别名的使用](http://man.linuxde.net/shell-script/shell-3" \t "http://man.linuxde.net/_blank)

[函数的定义、执行、传参和递归函数](http://man.linuxde.net/shell-script/shell-4" \t "http://man.linuxde.net/_blank)

[条件测试操作与流程控制语句](http://man.linuxde.net/shell-script/shell-5" \t "http://man.linuxde.net/_blank)

获取时间日期格式和延时：[date命令](http://man.linuxde.net/date" \t "http://man.linuxde.net/_blank)、[sleep命令](http://man.linuxde.net/sleep" \t "http://man.linuxde.net/_blank)

[内部字段分隔符IFS和脚本的调试DEBUG](http://man.linuxde.net/shell-script/shell-6" \t "http://man.linuxde.net/_blank)

显示、读取或拼接文件内容：[cat命令](http://man.linuxde.net/cat" \t "http://man.linuxde.net/_blank)

文件查找与打印文件列表：[find命令](http://man.linuxde.net/find" \t "http://man.linuxde.net/_blank)

命令传参过滤器、命令组合工具：[xargs命令](http://man.linuxde.net/xargs" \t "http://man.linuxde.net/_blank)

字符转换、删除及压缩工具：[tr命令](http://man.linuxde.net/tr" \t "http://man.linuxde.net/_blank)

对文本进行排序、单一和重复操作：[sort命令](http://man.linuxde.net/sort" \t "http://man.linuxde.net/_blank)、[uniq命令](http://man.linuxde.net/uniq" \t "http://man.linuxde.net/_blank)

[切分文件名提取文件扩展名或提取文件名：%、%% 和 #、##](http://man.linuxde.net/shell-script/shell-7" \t "http://man.linuxde.net/_blank)

[Shell的正则表达式](http://man.linuxde.net/docs/shell_regex.html" \t "http://man.linuxde.net/_blank)

在文件中搜索文本工具：[grep命令](http://man.linuxde.net/grep" \t "http://man.linuxde.net/_blank)

按列切分文件字段工具：[cut命令](http://man.linuxde.net/cut" \t "http://man.linuxde.net/_blank)

文本处理流编辑器：[sed命令](http://man.linuxde.net/sed" \t "http://man.linuxde.net/_blank)

对文本和数据进行处理：[awk编程](http://man.linuxde.net/awk" \t "http://man.linuxde.net/_blank)

临时文件的命名方法与随机数：[tempfile命令](http://man.linuxde.net/tempfile" \t "http://man.linuxde.net/_blank)

创建任意大小的文件和分割任意大小的文件：[dd命令](http://man.linuxde.net/dd" \t "http://man.linuxde.net/_blank)、[split命令](http://man.linuxde.net/split" \t "http://man.linuxde.net/_blank)、[csplit命令](http://man.linuxde.net/csplit" \t "http://man.linuxde.net/_blank)

Linux文件比较，文本文件的交集、差集与求差：[comm命令](http://man.linuxde.net/comm" \t "http://man.linuxde.net/_blank)

使用命令下载网站文件或网页：[wget命令](http://man.linuxde.net/wget" \t "http://man.linuxde.net/_blank)、[lynx命令](http://man.linuxde.net/lynx" \t "http://man.linuxde.net/_blank)

命令行下的高级网络工具：[curl命令](http://man.linuxde.net/curl" \t "http://man.linuxde.net/_blank)

监视文件及目录访问信息并记录：[inotifywait命令](http://man.linuxde.net/inotifywait" \t "http://man.linuxde.net/_blank)

Linux系统日志的相关命令、文件和管理工具：[logrotate命令](http://man.linuxde.net/logrotate" \t "http://man.linuxde.net/_blank)、[logger命令](http://man.linuxde.net/syslog" \t "http://man.linuxde.net/_blank)

当前登录用户、启动日志及启动故障相关信息：[who命令](http://man.linuxde.net/who" \t "http://man.linuxde.net/_blank)、[w命令](http://man.linuxde.net/w" \t "http://man.linuxde.net/_blank)、[users命令](http://man.linuxde.net/users" \t "http://man.linuxde.net/_blank)、[uptime命令](http://man.linuxde.net/uptime" \t "http://man.linuxde.net/_blank)、[last命令](http://man.linuxde.net/last" \t "http://man.linuxde.net/_blank)、[lastb命令](http://man.linuxde.net/lastb" \t "http://man.linuxde.net/_blank)

计算命令执行花费的时间信息：[time命令](http://man.linuxde.net/time" \t "http://man.linuxde.net/_blank)

数据归档和解压缩：[tar命令](http://man.linuxde.net/tar" \t "http://man.linuxde.net/_blank)、[cpio命令](http://man.linuxde.net/cpio" \t "http://man.linuxde.net/_blank)、[gzip命令](http://man.linuxde.net/gzip" \t "http://man.linuxde.net/_blank)、[bzip2命令](http://man.linuxde.net/bzip2" \t "http://man.linuxde.net/_blank)、[zip命令](http://man.linuxde.net/zip" \t "http://man.linuxde.net/_blank)