

《Linux基础》

第十三讲 shell脚本基础-逻辑判断与循环



test

- test是shell内建命令，可以处理脚本里的各类工作，产生的不是一般形式的输出，而是可用的退出状态。使用`help test`查看帮助文档。
- test命令有其他形式：`[.....]`，`[[.....]]`。当在`[]`中使用`&& ||`会出错，这时候要使用`[[]]`。
- test返回true或false，但是test返回的true是0，false是1，这和通常的编程语言定义的true是1（或非0值），false是0有所区别。（Linux/Unix上程序退出状态为0表示0错误正确执行，而非0值表示有错。）
- 例：`test "abc"="abc" ; test -f ~/tmp/a.sh ; [-f ~/tmp/a.sh]`

if,else,elif

- if , else, elif的语法结构:
- 写在一行要使用分号分隔: if [COMMAND] ; then [COMMAND] ; fi

```
if [COMMAND]
then
    [COMMAND]
fi
```

```
if [COMMAND]; then
    [COMMAND]
else
    [COMMAND]
fi
```

```
if [COMMAND]; then
    [COMMAND]
elif [COMMAND] ;
then
    [COMMAND]
else
    [COMMAND]
fi
```

if,else,elif示例

- if ,else,elif的用法:

```
file=~sh/a.sh
dbin=~bin
if test -f "$file" ; then
    cat "$file"
elif [ -d "$dbin" ] ; then
    ls "$dbin"
else
    echo "file not found"
fi
```

- if [[-f "sh/a.sh" && -f "sh/h.sh"]] ; then cat sh/a.sh
sh/h.sh ; fi

case

- 多个判断值可以使用if, elif, else组合。更简洁的形式是使用case语句实现, 就像普通编程语言的switch。语法结构:

```
case WORD in
    VALUE1)
        [COMMANDS]
        ;;
    VALUE2)
        [COMMANDS]
        ;;
    *)
        [COMMANDS]
        ;; //esac之前的;;可以省略
esac
```

-)是必须要加的, 每个逻辑块执行到;;结束。*)是默认情况, 并非必须。

case示例

- 创建shell脚本casetest.sh, 写入一下代码并运行:

```
case $1 in
    "hello")
        echo "hey!"
        ;;
    "time")
        date
        ;;
    *)
        echo "nothing to do"
esac
```

- 选项可以写成不带引号的形式: time hello

for

- for循环用于重复整个列表对象，基本用法：
 - `for NAME in WORDS; do COMMANDS; done`
 - `for NAME in WORDS`
 `do`
 `COMMANDS`
 `done`

- 示例：

循环列表

```
for i in a b c
do
    echo $i
done
```

遍历目录下所有文件

```
for i in ./* ; do
    echo $i
done
```

计数循环，这种结构bash支持，
sh不支持

```
for (( i=0;i<100;i++ ))
do
    echo $i
done
```

while与until

- while与until循环的结构一致，不同的是对待条件退出的状态，while是成功则执行，until是不成功则执行。结构使用如下：

```
while CONDITION ; do  #do写在单独一行，；可以不写
    COMMANDS
done
```

- 示例：

```
catfile=~ /sh/test.sh
while [ -f "$catfile" ] ; do
    cat "$catfile"
    sleep 5
done
```

```
catfile=~ /sh/null.sh
until [ -f "$catfile" ] ; do
    echo "$catfile not found"
done
```


本节课任务

- 在当前用户主目录创建bin目录（如果bin目录不存在）。
- 在bin/创建脚本文件pse。
- 要求脚本运行时检测参数个数是否 >0 ，大于0则使用第一个参数作为筛选条件。否则直接运行`ps -e -o user,pid,ppid,tty,comm,args`