

shell
by:晏

tips

shell脚本首行需要写上: #! /bin/bash
用来表示使用功能那种shell解释器来执行该脚本

参数

第x个参数 (从1开始) : \$x

所有参数: \$*

有多少入参: \$#

当前脚本名: \$0

接收参数

shell的工作方式

交互式

用户每输入一条命令就立即执行

批处理

先编写好shell脚本, 在统一执行

定时执行任务

一次性

增加使用at

可以交互式定时任务, 也可以使用管道符一行处理例如:
echo "systemctl restart httpd" | at 23:30

倒计时执行任务: at now +x MINUTE

删除使用atrm 任务序号

周期性crond

表 4-7

crontab 命令中的参数及其作用

参数	作用
-e	编辑计划任务
-u	指定用户名称
-l	列出任务列表
-r	删除计划任务

需要使用绝对路径来写, 可以使用whereis来查绝对路径

表 4-6

at 命令中的参数及其作用

参数	作用
-f	指定包含命令的任务文件
-q	指定新任务名称
-l	显示待执行任务的列表
-d	删除指定的待执行任务
-m	任务执行后向用户发邮件

测试语句(判断语句)
格式: [表达式]
注意表达式左右有空格

比较

文件测试参数

逻辑测试

与或非&&, ||, !

表 4-4

可用的整数比较运算符

运算符	作用
-eq	是否等于
-ne	是否不等于
-gt	是否大于
-lt	是否小于
-le	是否等于或小于
-ge	是否大于或等于

表 4-5

常见的字符串比较运算符

运算符	作用
=	比较字符串内容是否相同
!=	比较字符串内容是否不同
-z	判断字符串内容是否为空

条件语句

```
if 条件测试操作 1
then 命令序列 1
elif 条件测试操作 2
then 命令序列 2
else
命令序列 3
fi
```

```
case 变量值 in
模式 1)
命令序列 1
;;
模式 2)
命令序列 2
;;
.....
*)
默认命令序列
esac
```

```
for 变量名 in 取值列表
do
命令序列
done
```

循环语句

```
while 条件测试操作
do
命令序列
done
```

表 4-3

文件测试所用的参数

运算符	作用
-d	测试文件是否为目录类型
-e	测试文件是否存在
-f	判断是否为一般文件
-r	测试当前用户是否有权限读取
-w	测试当前用户是否有权限写入
-x	测试当前用户是否有权限执行