第二十一章：shell编程规范及变量

**一、编写脚本的基本流程；**

**二、脚本的基础知识；**

**三、脚本中变量的应用；**

**一、编写脚本的基本流程；**

1.shell脚本语言：帮助管理员实现批量化完成任务，结合计划任务实现自动化运维；

2.编写脚本的流程：

a.创建目录，集中存放脚本文件；

b.编写脚本内容；

vi 脚本名.sh ##编辑脚本文件

#!/bin/bash ##声明解释器

#注释内容 ##可以注释些信息，方便后续查看

命令及控制语句 ##脚本的核心内容

:wq ##保存文件

c.赋予脚本文件执行权限：chmod +x 文件

d.运行脚本；

3.脚本的声明：声明使用哪种解释器，cat /etc/shells文件中的所有解释器都可以使用，解释器可安装；

4.执行脚本的方式：

a.路径执行（必须有执行权限）；

相对路径

绝对路径

b.source 脚本的绝对路径或相对路径（不需要执行权限）；

c.sh -x 脚本的路径（不需要执行权限），并且会显示命令和输出结果；

**二、脚本的基础知识；**

1. 重定向：改变了标准的输入输出状态；

>：重定向输出到某个位置，替换原有文件的所有内容；

>>：重定向追加输出到某个位置，在原有文件的末尾添加内容；

<：重定向输入某个位置的文件；

2>：重定向错误输出；

2>>：重定向错误追加输出到文件的末尾；

&>：混合重定向输出，正确错误的输出结果都会输出；

1,2>：同上；

2.管道符：将|前的命令的输出结果作为管道|后的命令的输入；

3.echo命令：返回结果并显示；

语法：echo [选项] "需要返回的值"

选项： 选项-e 若出现特殊字符则不将其转义，当普通字符对待；

在返回值中\n 代表强制换行、\t代表制表符；

内容中的选项："内容\t内容" 插入tab

"内容\n内容" 强制换行

**三、脚本中的变量的应用；**

1.概述：变量是一个筐，筐里装的是变量的值，只需要将筐带走，那么值也就随之带走，值为可变；

2.优点：简化输入，使脚本内容更加清晰，提高脚本的适用性；

3.分类：自定义变量、环境变量、位置变量、预定义变量；

4.自定义变量的定义：

a.定义变量：变量名=变量值

b.查看变量：echo $变量名

c.使用变量：在命令中直接使用 $变量名

d.取消变量：unset 变量名

e.变量的作用范围：

局部变量：变量名=变量值，此种变量只能在当前bash中有效；

全局变量：export 变量名=变量值，此种变量在当前bash和当前子bash中有效；

注：可用 bash命令开启当前进程的子进程bash，exit退回到父bash进程；

f.变量值中使用的括号：

""：弱引用，会将双引号内的特殊符号与命令一起执行，例：A="ls -ld"

''：强引用，会将单引号内的$、"、\等特殊符号失效，例：B="$A /etc/ "，B='$A /etc/'；

``：在定义变量时，引用命令，在值中引用变量等同于$()，''不可嵌套，$()可以嵌套；例：A=`du -sh /etc/`

$()：在定义变量时，引用命令，可以嵌套；例：AC=$(rpm -qf $(which rz))

注：当变量值中包含'单引号时，应使用\'将单引号进行转义；

g.变量名不能使用特殊符号，一般用大写字母和下划线；

h.交互定义变量：

read -p "提示文字信息" 变量的名称

提示文字信息：123

echo $变量名

123

5.环境变量的定义：

a.环境变量设置用户工作环境，名称不能修改，值可以修改；

b.环境变量的常用：

PATH=设置外部命令的存放的路径，用冒号隔开

TMOUT=设置登录超时，单位为秒

HISTSIZE=设置历史命令记录的条数

LANG=设置语言环境（vi /etc/sysconfig/i18n 进行永久修改字符集）

注：设置临时环境变量时（注销就失效了）：export PATH=

c.环境变量的查看：env命令查看当前工作环境的所有环境变量；

/etc/profile文件为用户每次登陆执行的的配置文件；

每个用户家目录中的.bash\_profile是每个用户独立的环境变量配置文件；

6.位置变量和预定义变量：

a.位置变量用于将脚本参数传递给脚本，有$1-$9，除命令或脚本本身外的所有；

b.预定义变量：用户只能使用，不能创建，不能直接赋值；

$0：为脚本本身或命令本身

$\*：为脚本后的所有参数

$#：脚本参数数量统计

$?：上一条命令是否执行成功，返回值为0则成功，否则失败

注：在shell脚本中，所有的路径必须为绝对路径；