算术运算：

整数：expr（运算符号前后要有空格）、$[]、**let（不回显）**

X++ x=x+1

X-- x=x-1

X+2 x=x+2

X\*=3 x=x\*3

X/=4 x=x/4

小数：（也可以整数计算）bc

Bc回车（交互方式）

Echo “scale=4(小数位保留几位)” | bc （非交互）

测试：

字符串：[ a == b ] 、[ a != b ] 、 **[ -z $abc (测试abc是否为空)]**

**Read -p “请输入用户：” abc**

**[ -z $abc ] && echo “你没有输入” && exit**

**Useradd $abc**

数字：

-eq -ne -lt -gt -le -ge

文件或目录：

-e exist(存在)

-f file（存在且为文件）

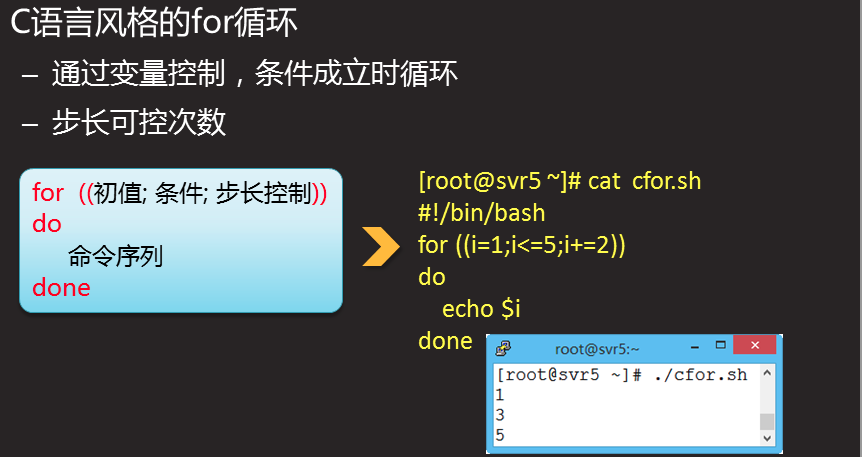
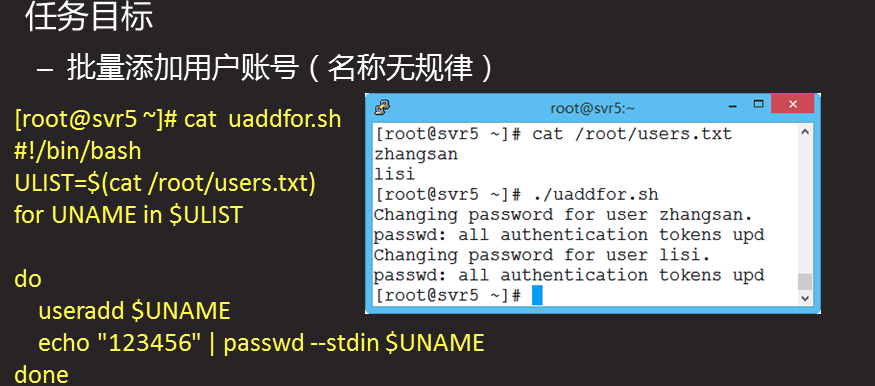
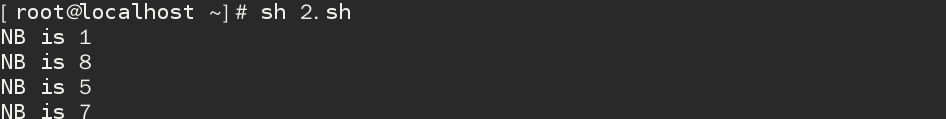
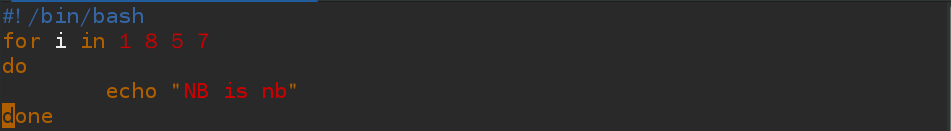
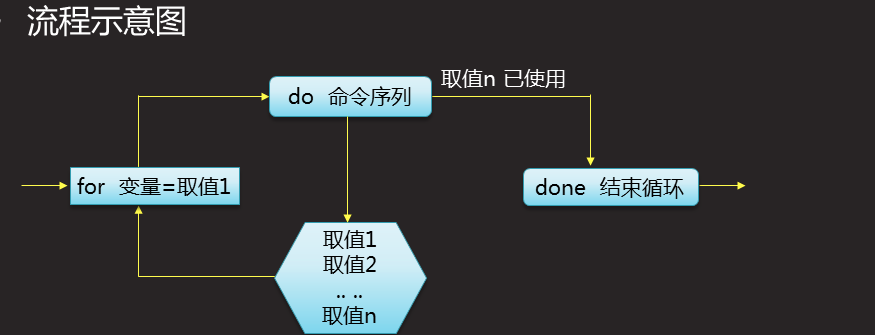
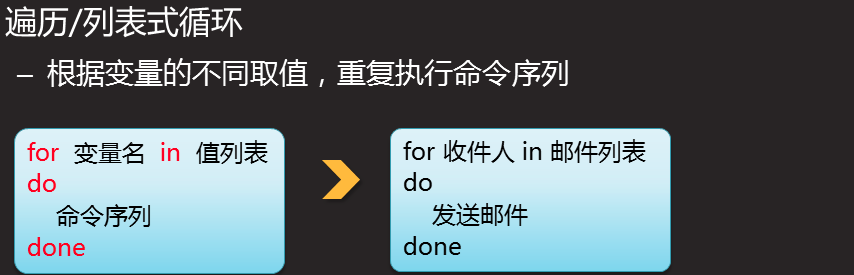
-d directory (存在且为目录)  **[ -d /abc ] || mkdir /abc**

-r （读取权限）

-w （写入权限）

-x （执行权限）

**For循环**



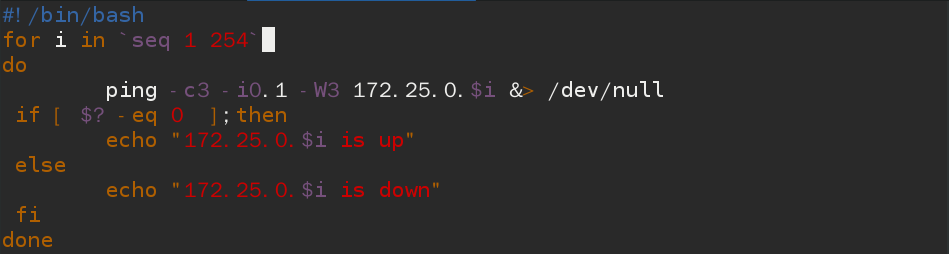
**# i控制次数，影响输出**

**# i控制次数 ，影响输出**

**# i控制次数**

**# i控制次数**

**批量检测多个主机的存活状态**



**# 造数--**

**{1..254} 中间不支持变量；**

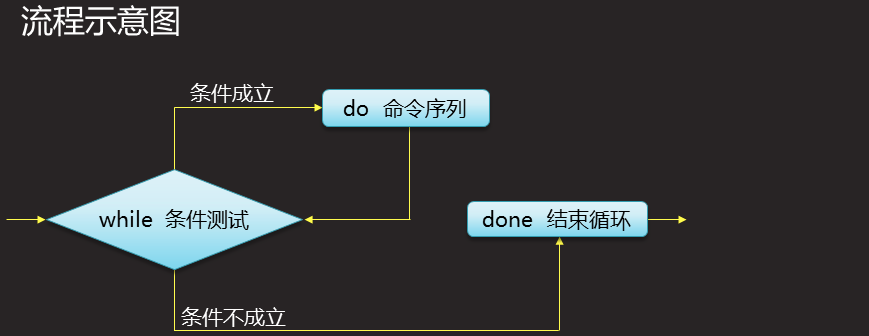
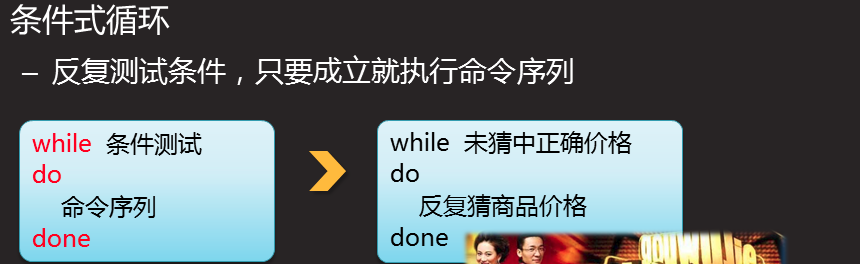
**不回显**

**`seq 1 254` 支持变量；回显**

**`seq 1 $i`**

**$(seq 1 $i)**

**While循环**



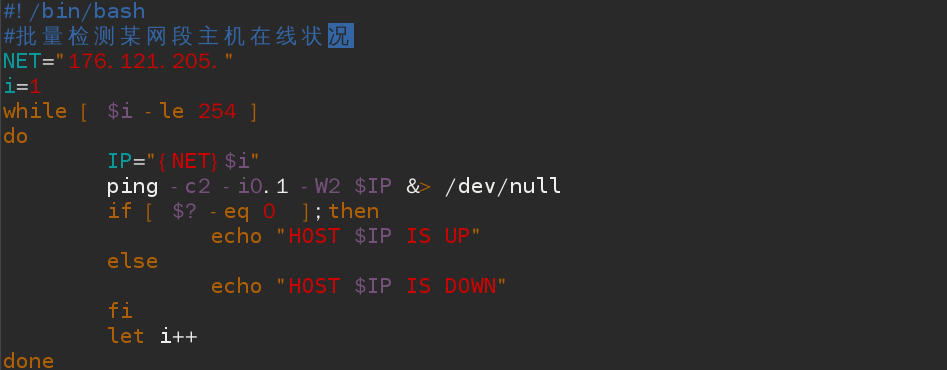
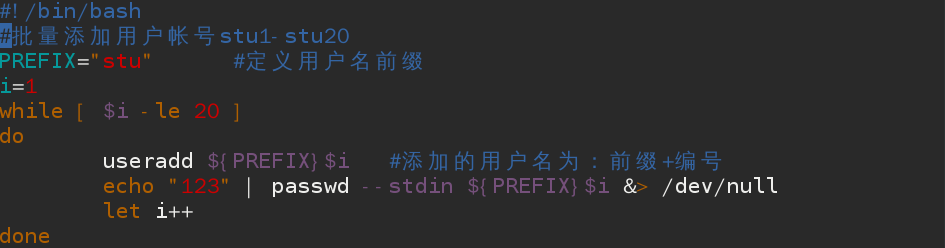
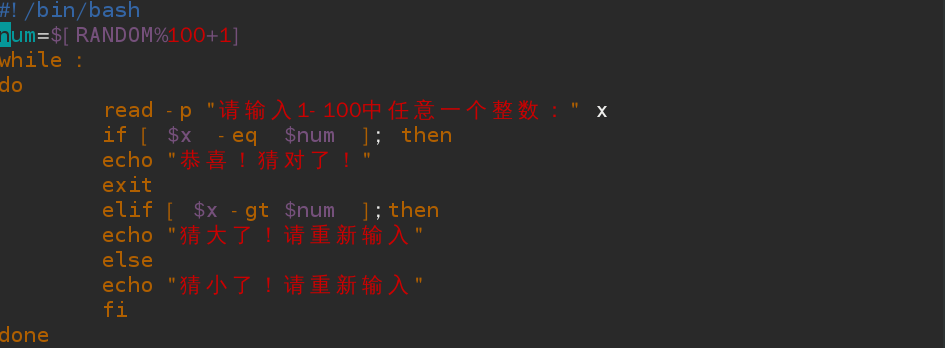
**for循环与while循环的区别:**

**for循环固定次数,取值循环；while循环不固定次数，条件判断**

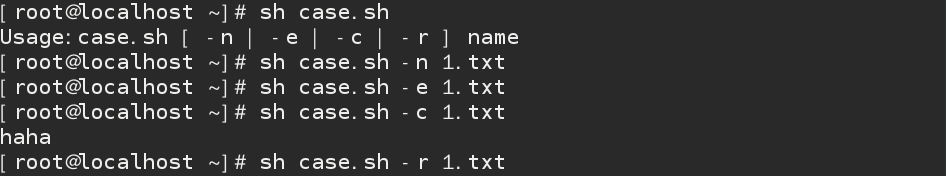
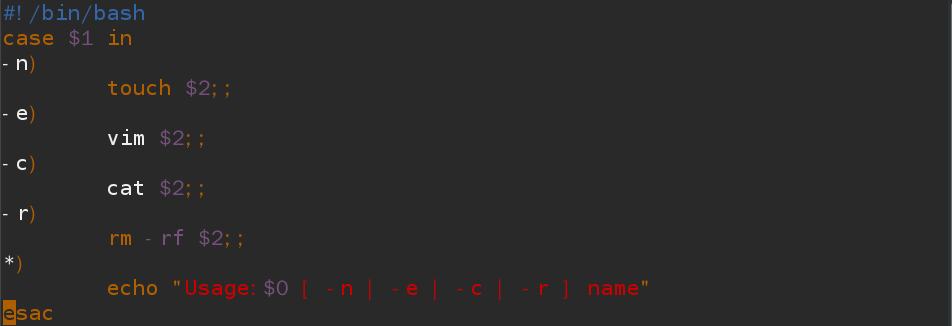
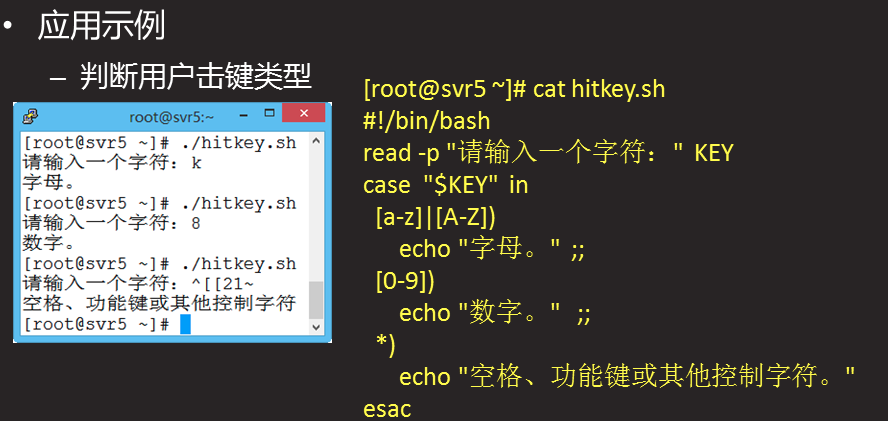
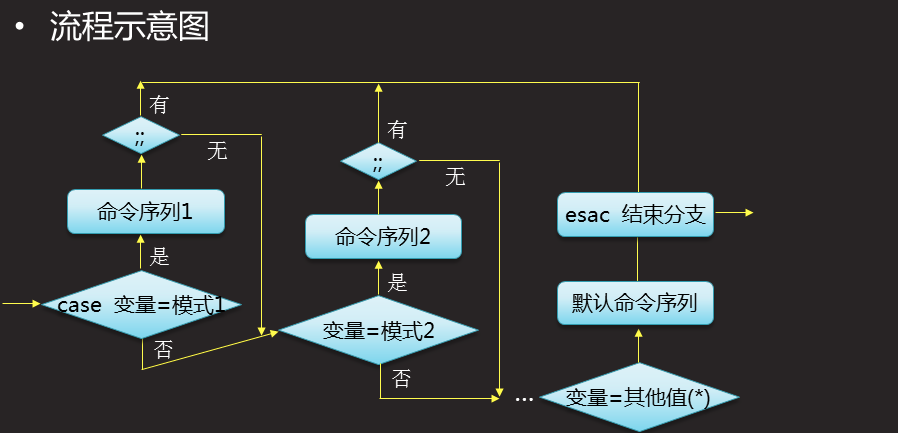
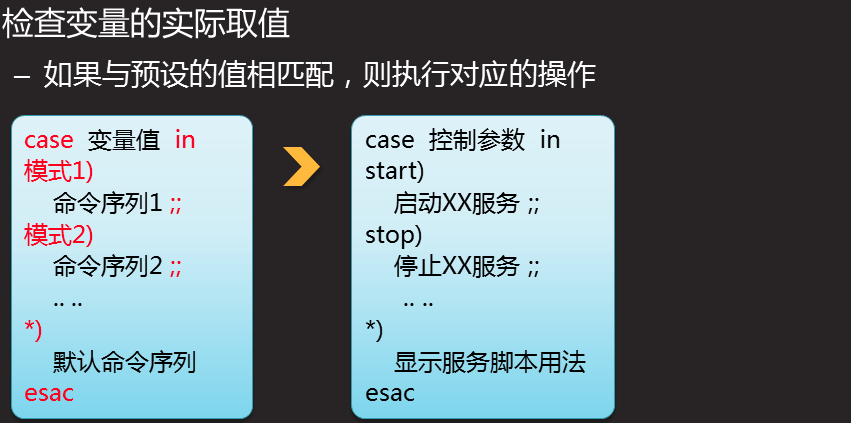


**#死循环**

**脚本实现猜数游戏**

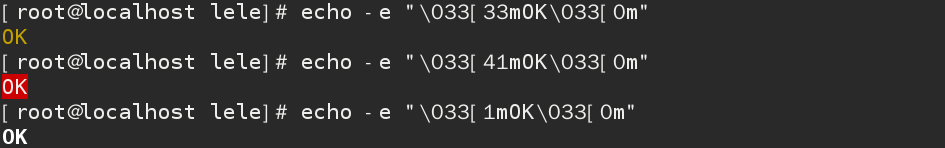
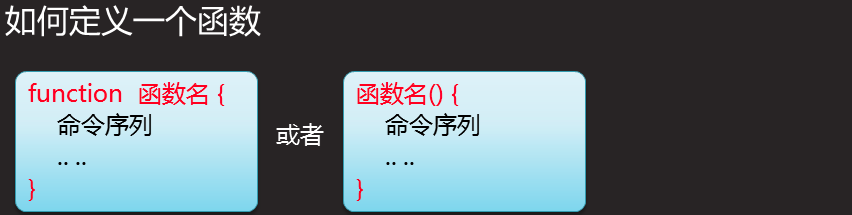


**Case语句（简化版的if循环，功能简单）**



**#[a-Z]**

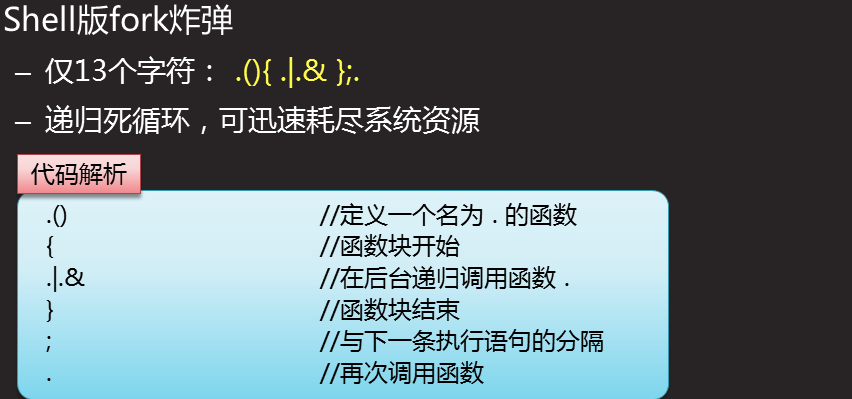
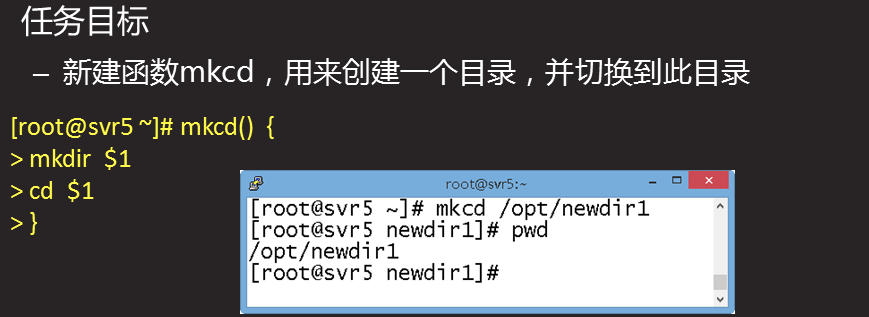
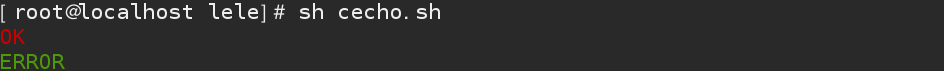
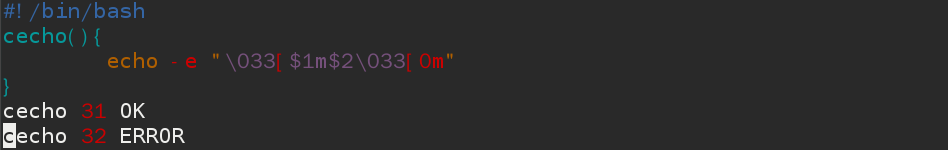
**SHELL函数（给一段代码取个别名）**



#3x字体色

4x背景色

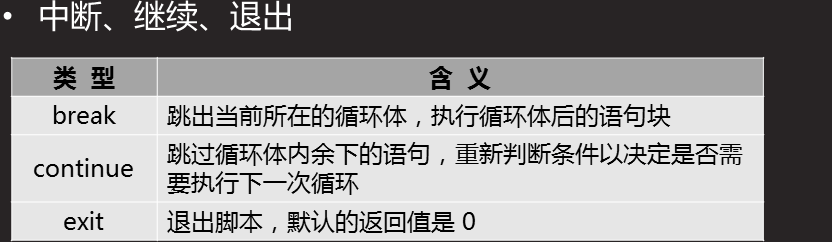
0x字体样式



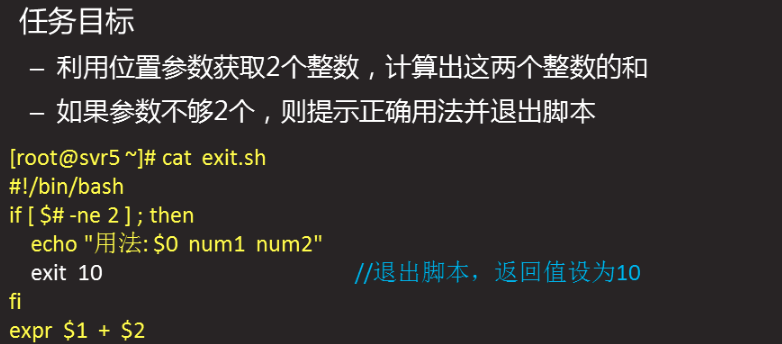
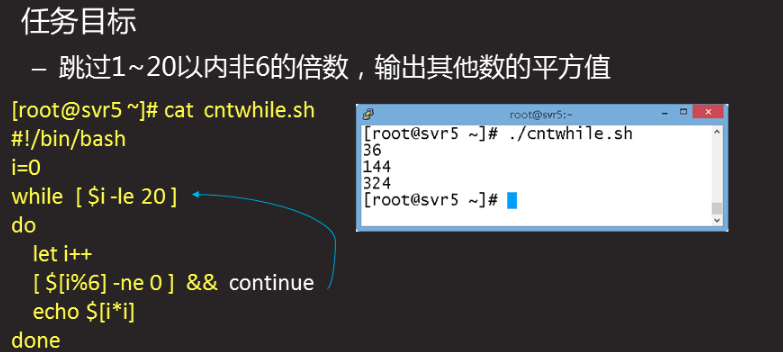
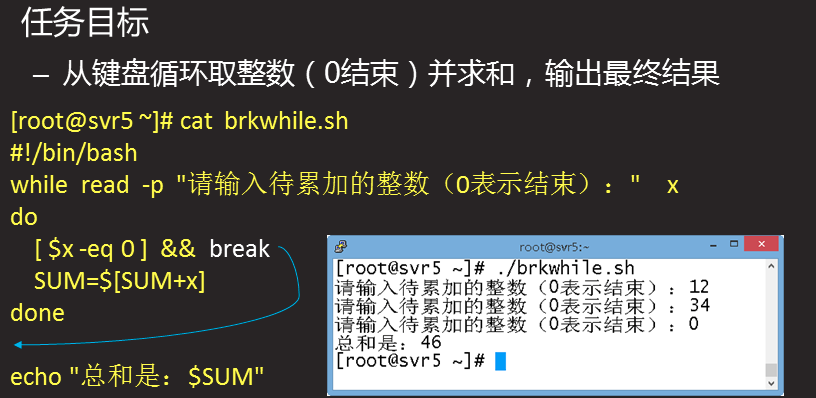
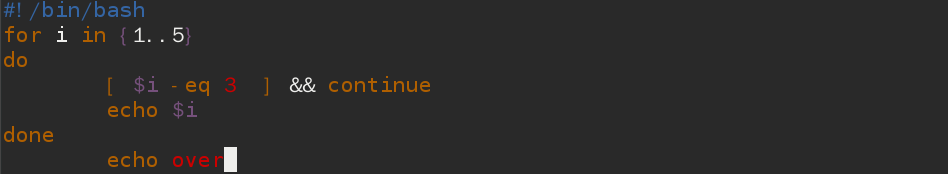
#脚本中可以定义函数让输出更加醒目

#脚本中可以定义函数让输出更加醒目

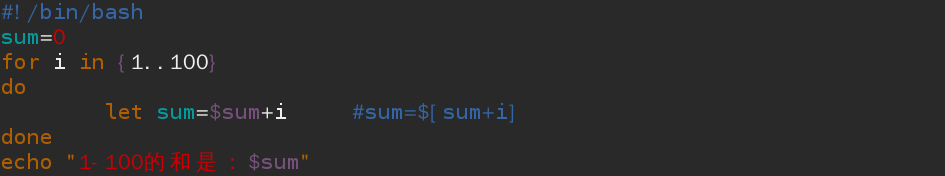
**脚本的终端及退出**



**#结束本次循环，进入下次循环**



1. **计算1+2+3+4+...+100的和**



1. **让用户输入数字，求和，直到输入over为止**

