编写脚本的规范：

1，声明解释器

2，注释，可以说明此脚本的功能，变量的含义

3，执行指令（在无特殊需求时，需要非交互式指令

脚本的执行方式

我i脚本赋予X权限

调用解释器执行脚本 会产生子进程

bash 脚本

source 使用当前的解释器 执行脚本

变量：自定义变量，

环境变量

预定义变量与位置变量

自定义变量，变量名称可以是数字字母下划线，不能以数字开头，不能使用特殊符号

变量名称=变量的值 等号两边不能有空格

环境变量，系统提前定义好可以直接使用

USER 用户名，UID 用户id号，HOME 用户的家目录 ， SHELL 用户使用的解释器 ，PWD 当前所在路径 ， PATH 命令存储路径 ， PS1 一级提示符 ， PS2 二级提示符 ， HOSTNAME 主机名 RANDOM随机数

预定义变量与位置变量

预定义变量有：$0 $# $\* $$ $?

位置变量有： $1 $2 $3 ....

echo $1 执行脚本后的第1个位置参数

echo $2 执行脚本后的第2个位置参数

echo $3 执行脚本后的第3个位置参数

echo $0 执行脚本的名称

echo $# 脚本后跟随的参数的个数

echo $\* 所有的位置参数

echo $$ 随机的进程号

echo $? 获取上一条任务的结果，0是成功，非0是失败

变量的扩展应用：

引号

" " 双引号 界定范围

' ' 单引号 界定范围 可以屏蔽特殊符号

` `或$( ) 反撇号 调用命令的执行结果

#!/bin/bash

stty -echo 屏蔽回显

stty echo 恢复回显

read指令，可以将脚本变为交互式，向用户询问变量内容

export 全局变量

让=变量可以在子进程中执行

export a 发布全局变量

export b=20 创建并发布全局变量

export -n a 恢复局部效果

shell中的运算：

\ 反斜杠也叫转义符，可以屏蔽之后一个特殊符号

加 减 乘 除 求模（取余数

expr

expr 1 + 1

expr 2 - 1

expr 2 \\* 2

expr 4 / 2

expr 5 % 3

[root@red opt]# a=10

[root@red opt]# b=20

[root@red opt]# expr $a + $b

$[ ]

echo $[1+1]

echo $[2-1]

echo $[2\*2]

echo $[2/2]

echo $[2%2]

[root@red opt]# echo $[a+b]

let 运算结果不显示，通常用于创建变量或者变量计算

let c=a+b 通过计算创建变量c

echo $c 查看

a=11

b=22

let c=a+b

echo $c

对变量进行自增减的操作

主流写法

let c=c-3 let c-=3

let c=c-1 let c--

let c=c+1 let c++

let c=c+3 let c+=3