# shell执行方式

1. 作为可执行程序

chmod +x ./test.sh

./test.sh

1. 作为解释器参数

/bin/bash test.sh

这种运行方式是，直接运行解释器，其参数就是shell脚本的文件名。

这种方式运行的脚本，不需要在第一行指定解释器信息，写了也没用。

# shell 变量

1. 数组操作与字符串的操作很重要。
2. shell的数据类型其实就两种，字符串与数字。

# inux后台运行和关闭、查看后台任务

1. **& 命令**

加在一个命令的最后，可以把这个命令放到后台执行。

如：**watch -n 10 sh test.sh & #每10s在后台执行一次test.sh脚本**

1. **ctrl + z 命令**

可以将一个正在前台执行的命令放到后台，并且处于暂停状态。

1. **jobs 命令**

查看当前有多少在后台运行的命令，jobs -l选项可显示所有任务的PID，jobs的状态可以是running, stopped, Terminated。但是如果任务被终止了（kill），shell 从当前的shell环境已知的列表中删除任务的进程标识。

1. **fg 命令**

将后台中的命令调至**前台**继续运行。如果后台中有多个命令，可以用fg %jobnumber（是命令编号，不是进程号）将选中的命令调出。

http://images.cnitblog.com/blog/408927/201409/182233000819052.png

1. **bg 命令**

将一个在后台暂停的命令，变成在**后台**继续执行。如果后台中有多个命令，可以用bg %jobnumber将选中的命令调出。

1. **kill 命令**

法子1：通过jobs命令查看job号（假设为num），然后执行kill %num

法子2：通过ps命令查看job的进程号（PID，假设为pid），然后执行kill pid前台进程的终止：Ctrl+c

1. **nohup 命令**

如果让程序始终在后台执行，即使关闭当前的终端也执行（之前的&做不到）。这时候需要nohup，该命令可以在你退出帐户/关闭终端之后继续运行相应的进程。关闭终端后，在另一个终端jobs已经无法看到后台跑得程序了，此时利用ps（进程查看命令）

**ps -aux | grep "test.sh" # a:显示所有程序 u:以用户为主的格式来显示 x:显示所有程序，不以终端机来区分**