* If：注意第三种 有elif 一定要有else否则出错

结构：



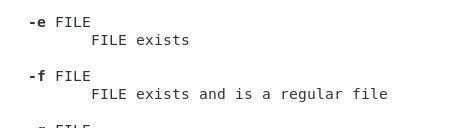
注意then后面最好有缩进进行区分，if后面的命令执行成功才会进入then



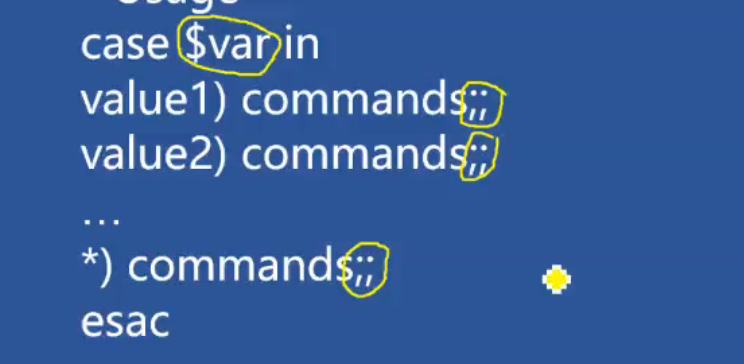
-n -z一定要加双引号 否则空了就不对

Inode number就是之前说的 ls -i

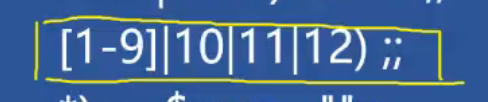
Test -e标准判断文件是否存在

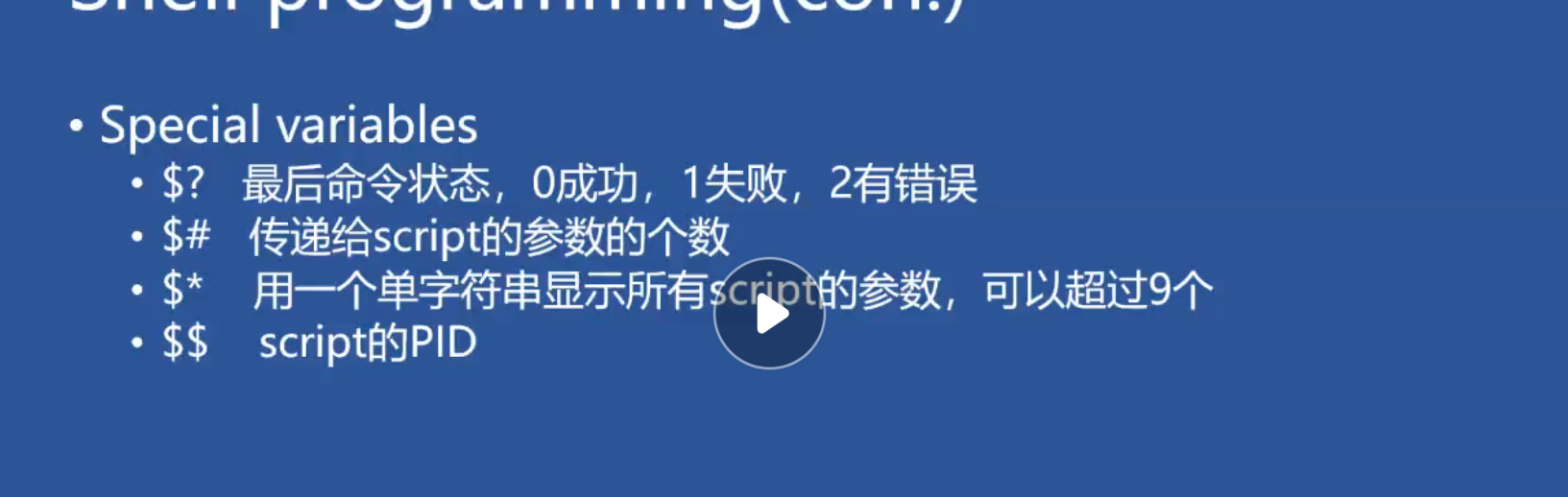


* Case :注意一定要有变量取值（因为是要根据变量值进行判断），双分号，最后是\*)，也就是default



* $#表示给出参数的个数
* 列表：中只能处理单个数字，如果放进去12 他是分开处理，处理成1和2

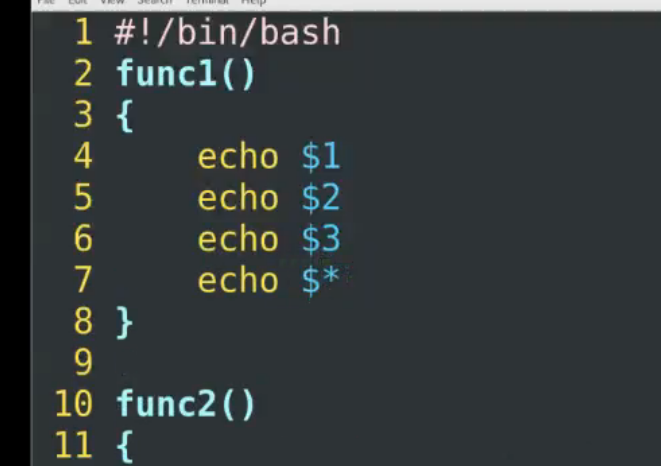


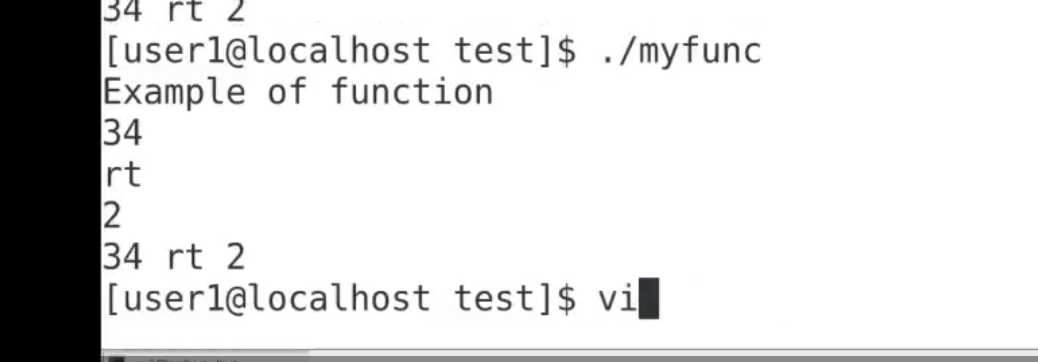


还有一个$@ 作用类似于 $\*，

但是$\*单字符串显示所有参数，中间用空格隔开

$@ 也是显示所有参数，中间用引号隔开

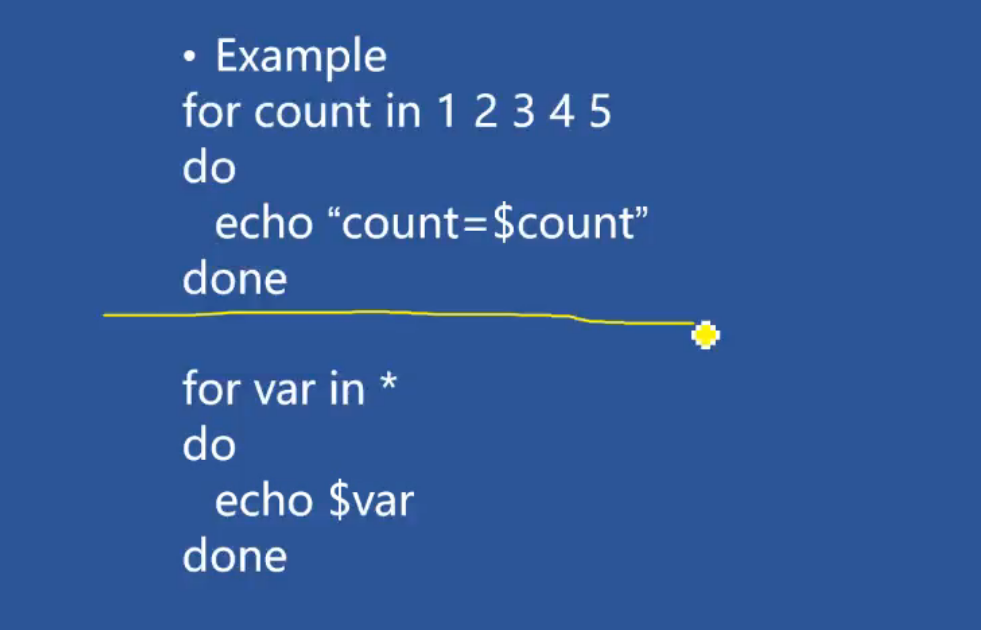




空格隔开

* Ls -l最上面有一行total 注意
* For：本质是后面var-list的值不断赋值给var，每次赋值一个，所以循环次数就是list的长度

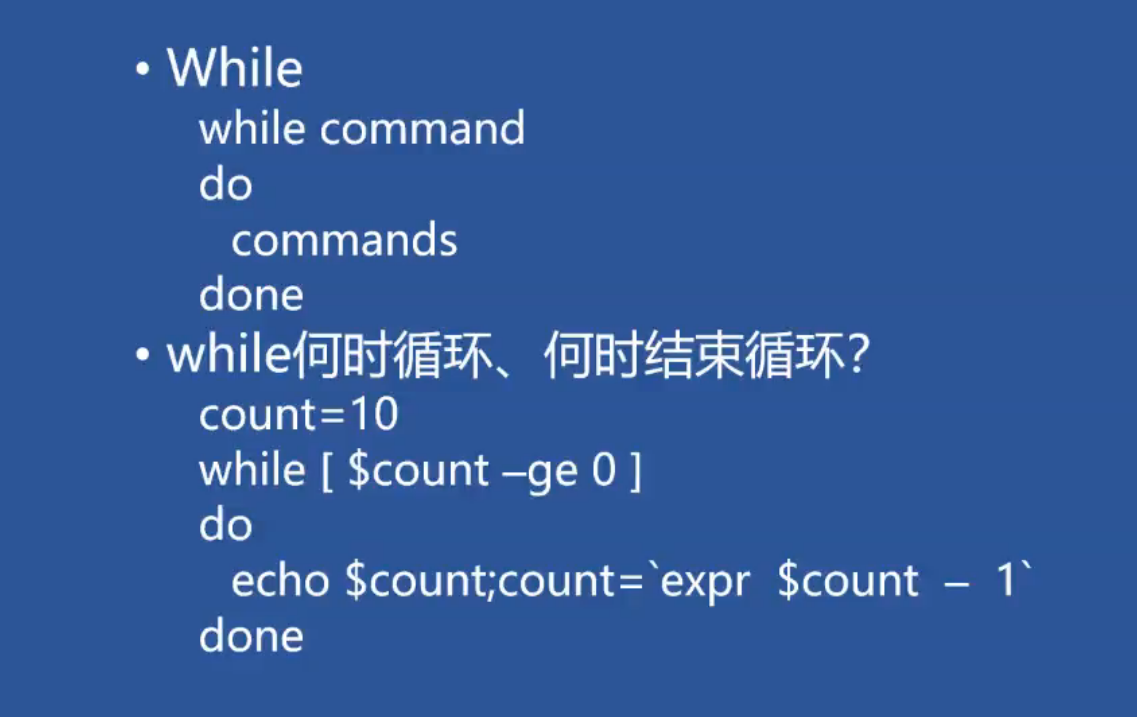




第一个就是打印1 2 3 4 5

\*表示当前目录下所有的entry的列表

* While

  
test的另一种写法就是[]，但是要注意前后都得有空格。

成立的时候进入循环。

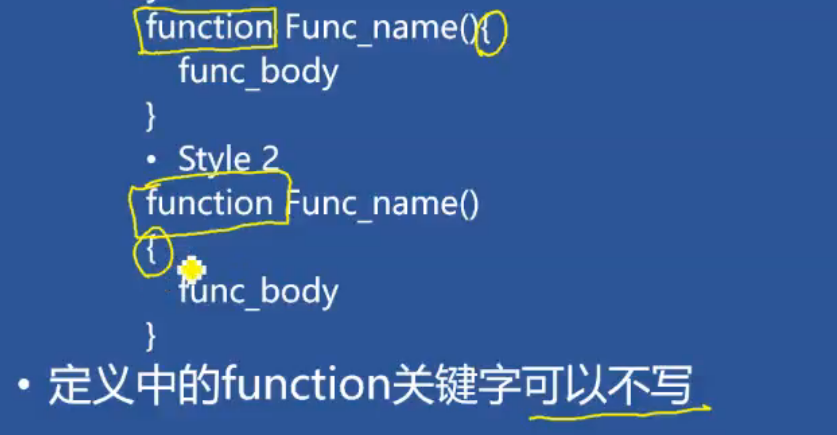
* Until command 条件不成立的时候进入

Do

Commands

Done

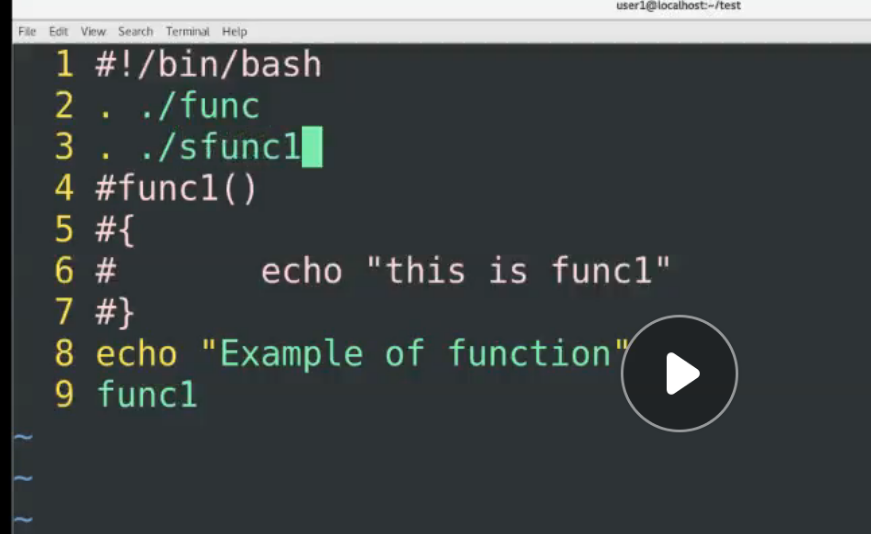
* 函数：



两种格式：

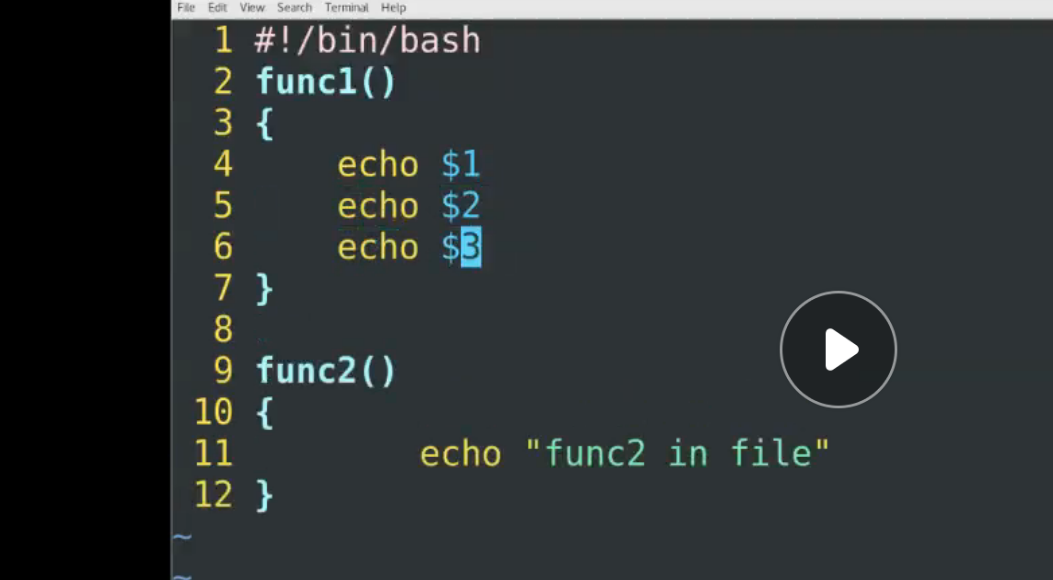
注意

* function可以不写
* （）只是样式（不可以加参数）
* 大括号只有这两种格式
* 在脚本中写了一个函数，且要调用他，那么定义一定要放在前面
* 调用的时候不能加()，直接函数名进行调用
* 函数放在脚本外，脚本进行调用，需要用. 命令进行函数文件（注意一定要文件名要路径）的导入

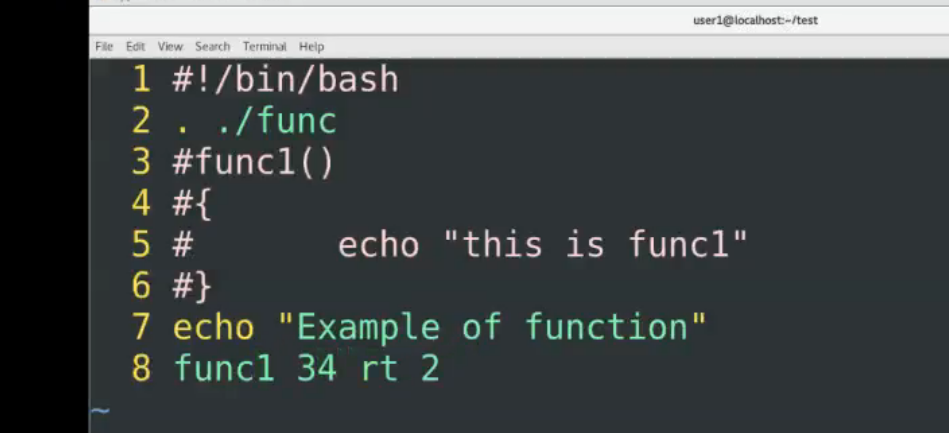


上面的是载入脚本，下面的是点命令执行脚本（样子一样可以有两个意思）

* 传参数，位置变量

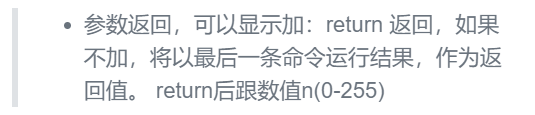


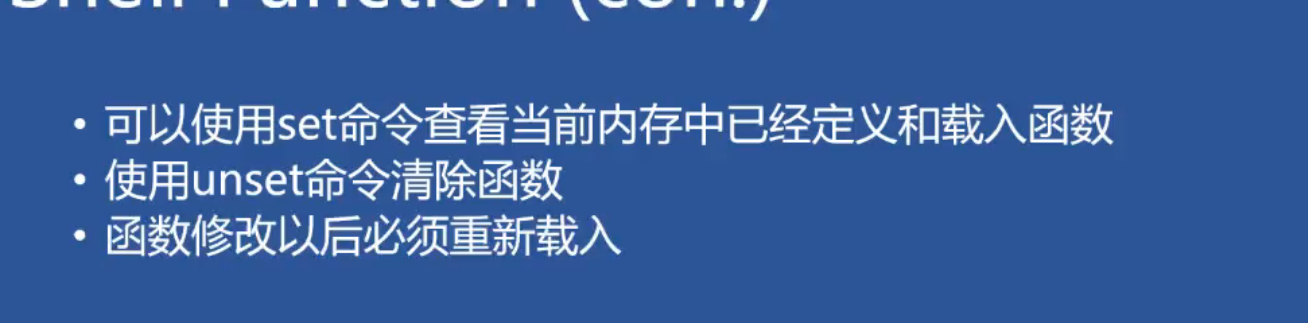
需要传入三个参数，



直接在函数名后面传三个数

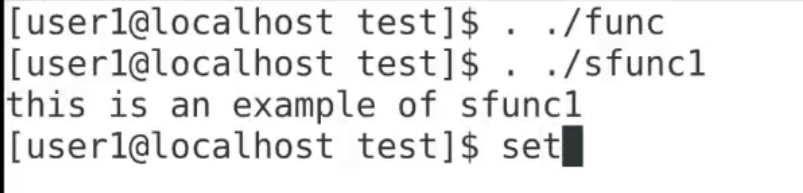
* 函数参数返回



* 

函数修改一定要重新载入，进行刷新，否则内存中的函数没有变话的

例如：



载入./func 文件 set可以查看