

# 分隔符、条件判断与任务实现

主讲教师：龙霄汉



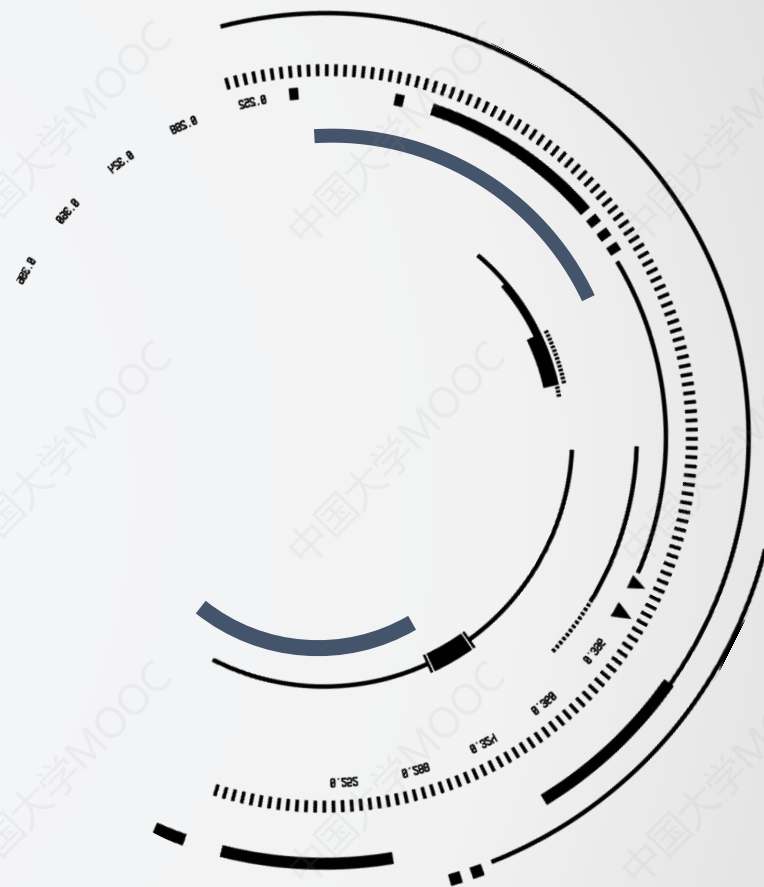
## ◆ 常见的分隔符:

◆ **;** : 完全分隔指令

◆ **&&** : 与操作, 前后两个指令都为真, 表达式才为真

◆ **||** : 或操作, 任意一个指令为真, 表达式就为真

◆ 使用格式: 指令1 **分隔符** 指令2



- ◆ 完全分隔
- ◆ 使用格式：指令1 ; 指令2

```
[root@ccit ~]# id root
uid=0(root) gid=0(root) groups=0(root)
[root@ccit ~]# ls
main.sh
[root@ccit ~]# id root; ls
uid=0(root) gid=0(root) groups=0(root)
main.sh
```



# && 分隔符

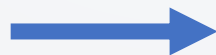


- ◆ 与操作
- ◆ 使用格式：指令1 **&&** 指令2

$T \&\& T = T$

- ◆ **T**表示指令执行成功：指令存在或指令本身有真假的输出
- ◆ 指令1执行**成功时继续**执行指令2
- ◆ 指令1执行**失败后不再**执行指令2

$F \&\& T = F$   
 $F \&\& F = F$



**无需执行指令2**



## || 分隔符



- ◆ 或操作
- ◆ 使用格式：指令1 || 指令2

T || T=T; T || F=T

- ◆ T表示指令执行成功：指令存在或指令本身有真假的输出
- ◆ 指令1执行成功时**不再**执行指令2

$$\begin{array}{l} T || T = T \\ T || F = T \end{array} \longrightarrow \text{不再执行指令2}$$



- ◆ 条件判断可以让程序具有“智能”。
- ◆ 一个条件表达式会有一个结果：
  - ◆ 条件表达式**成立**，称该表达式为**T**
  - ◆ 条件表达式**不成立**，称该表达式为**F**
- ◆ 例如：
  - ◆ 条件表达式：  $1 == 2$  不成立，表达式结果为**F**
  - ◆ 条件表达式：  $1 \leq 2$  成立，表达式结果为**T**



- ◆ 在bash中，变量 **\$?** 的值与上一条指令执行是否成功有关。
- ◆ 如果上一条指令执行成功，则 **\$?** 的值为0。
- ◆ 如果上一条指令执行失败，则 **\$?** 的值非0。

```
[root@ccit ~]# ls
main.sh
[root@ccit ~]# echo $?
0
[root@ccit ~]# ls test
ls: cannot access test: No such file or directory
[root@ccit ~]# echo $?
2
```





- ◆ 在bash中，可以使用 `[]` 或 `if` 进行条件判断。
- ◆ `[]`格式：`[ 表达式 ]`

比较操作符的含义	比较操作符的写法
等于	-eq
大于	-gt
大于等于	-ge
小于	-lt
小于等于	-le
不等于	-ne

- ◆ `[]`使用格式：`[ 数值1 操作符 数值2 ]`
- ◆ 例如：`[ 1 -ge 2 ] = 0`





◆ -f: 指令格式: **[ -f 文件名 ]**

功能: 判断文件是否为普通文件

◆ -d: 指令格式: **[ -d 目录 ]**

功能: 判断目录是否存在

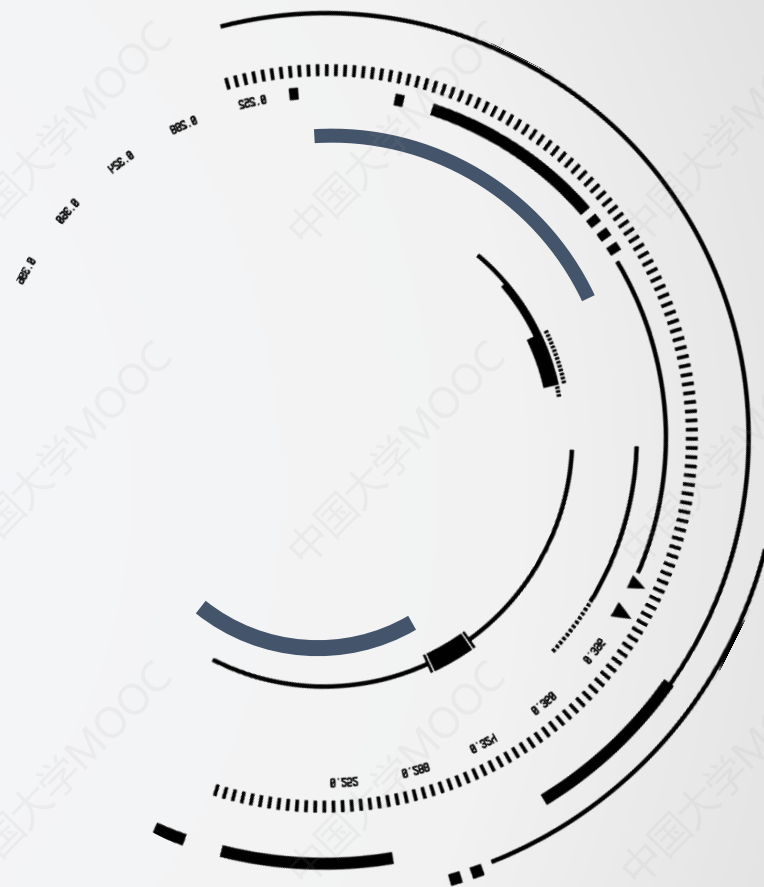


1. 设置回收站目录

2. 将要删除的复制到回收站

3. 删除文件

- ◆ 脚本文件命名: srm
- ◆ 使用方法: **srm** 要删除的文件或目录名



# 感谢您的观看!

