**1.ls cd pwd**

Ls

① -a 所有文件包括隐藏文件

-l 除了文件名外，列出许可权所有者文件大小等

-d将目录象文件一样显示，而不是显示其下的文件。

-h 显示可读列表的文件大小

-t 按文件修改时间排序

-l d\*列出当前目录中所有以“d”开头的文件目录的内容。

Cd

①cd / 从当前目录进入系统根目录

Cd .. 进入父目录

Cd ~ 进入用户在该系统的home目录

Cd - 从当前目录到上一个目录

Pwd

①-L 目录连接链接时，输出连接路径

-P 输出物理路径

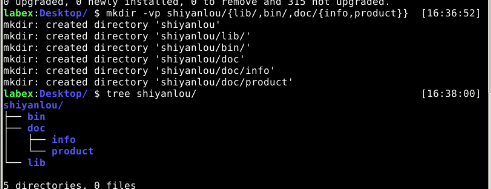
**2.mkdir rm mv cp cat nl**

mkdir

①-m 设定权限

-p 它可以是路径名。如果路径中的某些目录尚不存在，则此选项将自动创建尚不存在的目录，即可以一次创建多个目录。

-v 每次创建新目录时，都会显示信息



Rm

①-f 忽略不存在的文件，并且删除从不提示

-i 进行交互式删除

rm \*.log 或 rm -i \*.log后者有交互

-r 递归删除参数中列出的所有目录和子目录

-v 详细显示执行的步骤

Mv 选项 源文件 目标文件

①-b如果需要覆盖文件，则需要先备份它，然后再覆盖

-f如果目标文件已经存在，则无需询问即可直接覆盖

-i如果目标文件已经存在，它将询问是否覆盖

-u若目标文件已经存在，且 source 比较新，才会更新(update)

-t此选项适用于将多个源文件移动到目录中。这时，首先是目标目录，然后是源文件。

mv -t /opt/soft/test/test4/ log1.txt log2.txt log3.txt



Cp 选项 源文件 目录

①-t 指定目标目录

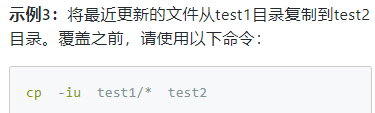
-f 强制复制

-i 覆盖前询问

-n 不覆盖现有文件

-u仅当源文件的修改时间大于目标文件的时间或相应的目标文件不存在时，才会复制文件。

-s 创建到源文件的符号链接，而不是复制文件



Cat cat命令的功能是将文件或标准输入组合输出到标准输出。此命令通常用于显示文件的内容，或将多个文件链接在一起，或从标准输入中读取和显示内容。它通常与重定向符号一起使用

①-a 显示全部，等效于-vET

-b非空输出线数

-e 等效于-vE

-E 在每行末尾显示

-n对于所有输出行号，对于所有输出行号，从1开始

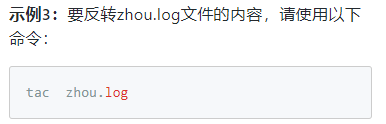
-s 如果有两个以上的空行，请用空行替换

-t 等效于-vT

-T 将制表符显示为^ I

-u 被忽略

-v Use ^ and M-references, except LFD and TAB



nl nl命令用于计算文件中的行号。Nl可以自动将行号添加到输出文件内容中。缺省结果与cat -n略有不同。nl可以显示比显示设计更多的行号，包括位数以及是否自动填充0等。

-b a指示该行号是否也为空行也被列出（类似于cat -n）

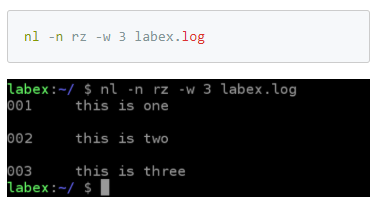
-b t如果有空行，请不要在空行中列出行号（默认）

-n ln行号显示在屏幕的最左侧

-n rn行号显示在其自己字段的最右边，而不加0

-n rz行号显示在其自己字段的最右边，外加0

-w行号字段占用的位数



**3.more,less,head,tail**

More more命令是在屏幕上从上到下显示整个文件的内容。more命令逐页显示，允许用户逐页阅读。最基本的命令是按空格键显示在下一页，然后按b按钮返回到下一页。还有一个搜索字符串功能。more命令从前到后读取文件，因此在启动时会加载整个文件。

①+n从n行开始

-n将屏幕尺寸定义为n行

+/patten在显示每个文件之前搜索模式，然后从patten的前两行开始

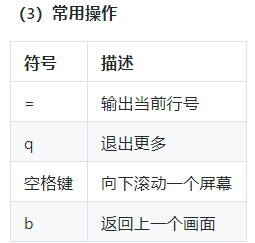
-c从顶部清除屏幕，然后显示

-d Prompt "Press space to continue, 'q' to quiet", disable the ring function

-p通过清除窗口而不是滚动来写文件，类似于-c选项

-s将连续的空白行显示为一行

-u从文件内容中删除下一行





Less less工具也是用于分页文件或其他输出的工具。应该说linux是用于查看文件内容的正统工具，并且功能非常强大。

①-e显示文件时，它会自动离开

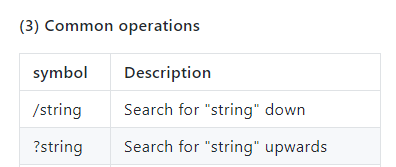
-f强制打开特殊文件，例如外设代码号，目录和二进制文件。显示文件时，它会自动离开

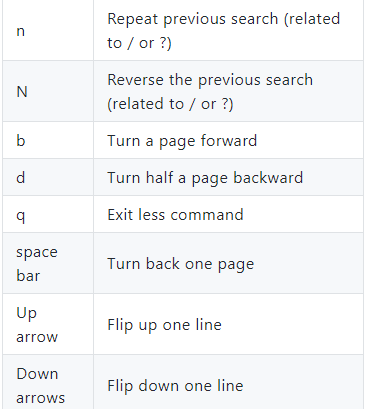
-i搜索时忽略大小写

-m像more命令一样显示百分比

-N显示每行的行号

-s显示每行的行号





Cat命令功能：用于显示整个文件的内容，单独使用而无需翻页功能。因此，它经常与more命令一起使用。cat命令还具有将多个文件合并为一个文件的功能。

more命令功能：显示整页时允许屏幕暂停。按空格键继续显示下一个屏幕，或按q按钮停止显示。

less命令功能：less命令的使用类似于more命令。它还可以用于浏览一页以上的文件。区别在于less命令除了按空格键向下显示文件外，还可以使用向上和向下键滚动浏览文件。要结束浏览时，只需在less命令的“：”提示下按q。

head head命令和它的名字一样容易理解。它主要用于将文件的开头显示为标准输出。默认的head命令打印其对应文件的前10行。

①-q隐藏文件名

-v显示文件名

-c显示字节数

-n显示的行数



tail tail命令主要用于显示指定文件的内容。常用于查看日志文件。

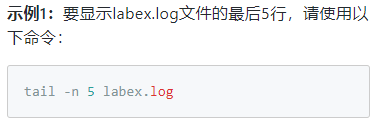
①-f循环阅读

-q不显示处理信息

-v显示详细处理信息

-c显示字节数

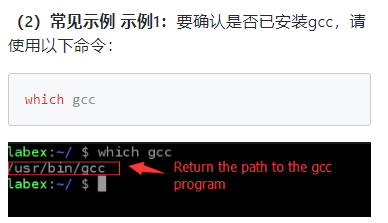
-n显示的行数



**4.which,whereis,locate**

Which which命令的作用是在PATH变量指定的路径中搜索可执行文件的位置。通常用于确认系统中是否安装了指定的软件。

①



Whereis whereis命令主要用于在文件系统中找到可执行文件，源代码文件和帮助文件。whereis命令还具有搜索源代码，指定备用搜索路径以及搜索异常项的功能。

whereis命令查找非常快，因为它不仅在磁盘上随机查找，而且在数据库（/ var / lib / mlocate /）中查找。该数据库由Linux系统自动创建，包含所有本地文件的信息，并通过自动执行updatedb命令每天更新一次。正是由于该数据库每天更新一次，这会使whereis命令的搜索结果有时不准确，例如可能找不到刚添加的文件。

①-b定位可执行文件

-m找到帮助文件

-s查找源代码文件

-u在默认路径中搜索可执行文件，源文件和帮助文件以外的文件

-B指定搜索可执行文件的路径

-M指定搜索帮助文件的路径

-S指定搜索源文件的路径

Locate ocate命令类似于whereis命令，并且它们使用相同的数据库。但是，whereis命令只能搜索可执行文件，联机帮助文件和源代码文件。如果要获得更全面的搜索结果，可以使用locate命令。

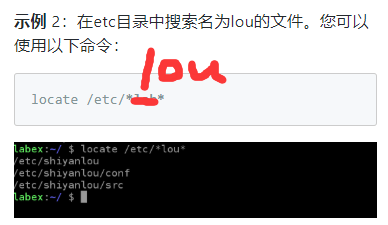
查找命令使用非常复杂的匹配语法，该语法使用特殊字符（例如“ \*”和“？”）来指定要查找的样本

①-q安静模式，不显示任何错误信息

-n显示最多n个输出

-r使用正则表达式进行搜索

-V显示版本信息

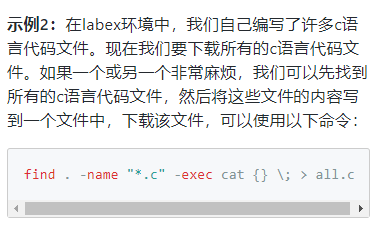


**5.find,xargs**

Find find命令的主要功能是遍历文件层次结构，匹配满足条件的文件并执行相应的操作。Linux下的find命令提供了很多搜索条件，功能非常强大，相应的学习也比较困难。

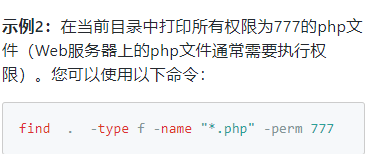
①-print查找命令将匹配文件输出到标准输出

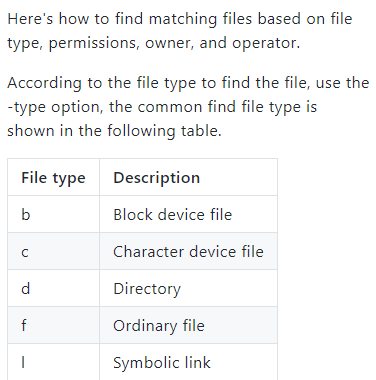
-exec find命令执行为匹配文件提供的shell命令



-name按文件名查找文件

-type查找文件类型（使用-perm选项基于文件权限查找文件。）



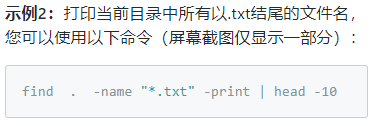


-prune 使用此选项可以使find命令不在当前指定的目录中查找。如果使用-depth选项，则find命令将忽略-prune。

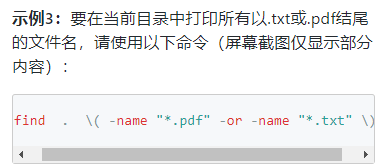
-user按文件所有者查找文件

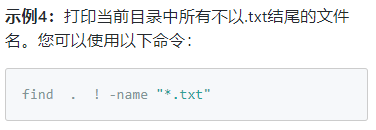
-group按文件所属组查找文件

-mtime –n +n根据文件的更改时间查找文件，-n表示文件更改时间小于n天，+ n表示文件更改时间现在大于n天，find命令还具有-atime和-ctime选项



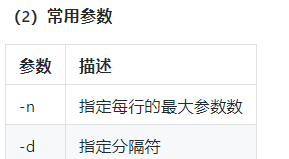






Xargs 我们可以使用管道将一个命令的stdout（标准输出）重定向到另一命令的stdin（标准输入）。但是，某些命令只能以命令行参数的形式接收数据，而不能通过stdin接收数据流。在这种情况下，无法通过管道将数据重定向到这些命令。

在这种情况下，xargs可以发挥作用。xargs命令可以接收来自标准输入的输入，并将输入转换为特定的参数列表。







**6.wc,grep,Regular express**

Wc wc命令是一种统计工具，用于显示文件中包含的行数，字数和字节数。

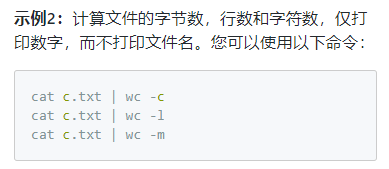
①-c统计字节

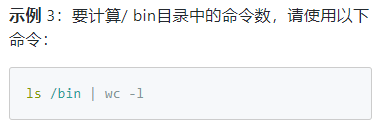
-l统计行

-m计算的字符数。该标志不能与-c标志一起使用

-w统计单词的数量，一个单词定义为由空格，制表符或换行符分隔的字符串

-L打印最长行的长度





Grep grep是一个非常强大的命令，用于在文件中查找匹配的文本，接受正则表达式和通配符，并使用多个grep命令选项来生成各种格式的输出。Grep可用于Shell脚本，因为grep通过返回状态值来声明搜索状态，如果模板搜索成功，则返回0，如果搜索失败，则返回1，如果搜索的文件不存在，则返回2。我们可以使用这些返回值来执行一些自动化的文本处理。

grep [选项]模式[文件]

①-c计算找到的“搜索字符串”（即模式）的数量

-i忽略大小写差异，因此大小写相同

-n输出行号

-v反向选择，打印不匹配的行

-r递归搜索

--color=anto在找到的关键字部分添加彩色显示

Regular expression 正则表达式

^以开始 $以结尾

.匹配任何字符

[ ]匹配[]中包含的任意字符

[-]匹配指定范围内的任何字符

[^]匹配除[^字符]外的任何字符

{n}匹配上个项目n次；[0-9] {2}匹配任何两位数字，等同于[0-9] [0-9]

{n,}上一项至少要匹配n次

{n，m}指定上一个项目需要匹配的最小和最大次数

? 匹配上一项1或0次; lab？ex与labex，laex匹配

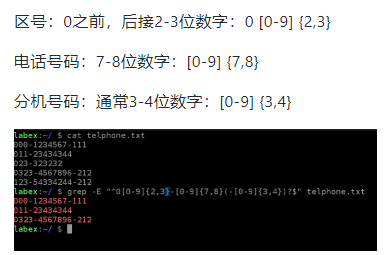
\*匹配上一项0次或多次lab \* ex匹配laex,labex，labbex

+匹配上一项或多次lab + ex匹配labex，labbex

( ) 创建一个用于匹配的子字符串; Ma(in)? 匹配 ma ，main

| 匹配| 双方中的任何一方; Dec（1st | 2nd）匹配Dec 1st或Dec 2nd

\ 转义上方的特殊字符 ；a\+b matches a+b



**7.cut,paste,tr**

**8.sort,uniq,join**

**9.comm,diff,patch**

**10.df,du,time**