## 登录tty的快捷建

Ctrl +Alt+F1:回到图形界面

Ctrl+Alt+F2 :terminal 1

Ctrl+Alt+F3:terminal 2

Ctrl+Alt+F4 :terminal 3

Ctrl+Alt+F5:terminal 4

Ctrl+Alt+F6 :terminal 5

## 打开图形模式终端的方式

调出系统搜索框，在搜索框中输入“terminal”

## 命令行提示符

[xiongzhuang@localhost ~]$

Xiongzhuang当前用户的名字（username） linux是多用户的操作系统

@前面是用户名，后面是所在的域

Localhost 电脑主机的名字

~当前所在目录的名称，会随着用户进入不同目录而改变

~表示当前用户的家目录（home directory）

$指示你所具有的权限 $表示普通用户 #表示root用户

## 切换至root用户

使用sudo su命令

（

若不能切换：使用vi /etc/sudoers

找到root ALL=（ALL） ALL

在下面插入 userName ALL=(ALL) ALL

）

退出root身份，可以用ctrl+d的组合键 或者用exit命令

## 查看用户名和主机名

whoami 查看用户名

Hostname 查看主机名

## 时间

date 获取当前时间

## 命令的参数

1参数是写在命令之后的一些补充选项。命令和参数之间有空格隔开

1. 格式：command parameters
2. command和parameter分别是英语“命令”和“参数”的意思
3. 参数里可以包含多个参数，由空格隔开
4. 参数也可以包含数字，字母等等
5. 参数没有固定的格式，但是一般来说还是遵循一定的规范

短参数

1 最长用的参数形式就是一个短横线后接一个字母：command -p

2 一次加好几个短参数，可以用空格隔开：command -p -a -T -c

3多个短参数也可以合并在一起 command -paTc

4字母的大小写代表的意义不同

长参数

1长参数没有短参数那么常用，但也是很有用的

2短参数是以一个短横线开始，而长参数是以两个短横线开始的

3格式：command --parameter

4多个长参数是不能像多个短参数那样合并写

5而是只能以空格隔开：command --parameter1 --parameter2

6可以组合使用短参数和长参数

7 同一个意义的参数有短参数和长参数的两种形式

参数的值

1有一些参数需要赋值。短参数和长参数的赋值方式不一样

2短参数复制通常是 command -p 10

3长参数复制通常是 command --parameter=10

## 找到命令

输入一个字母 按两次tab键可以补全命令，文件目录 文件名称

## 命令的历史记录

上下键来查找最近的命令

history 查找命令的历史 加！加命令的编号就会运行该命令

Ctrl+L用于清理终端的内容,就是清屏的作用，同clear命令

Ctrl+D给终端传递EOF（End Of File,文件结束符）

Shift+PgUp用于向上滚屏，与鼠标的滚轮向上滚屏是一个效果

Shift+PgDown 用于向下滚屏，与鼠标的滚动向下是一个效果

Ctrl+A光标跳到一行命令的开头，Home键有相同的效果

Ctrl+E光标跳到一行命令的结尾 End键有相同的效果

Ctrl+U删除光标左侧所有字符

Ctrl+K删除光标右侧所有字符

Ctrl+W删除光标左侧的一个单词

Ctrl+Y粘贴用Ctrl+U,Ctrl+K或者Ctrl+W删除的字符串

## Linux的根目录下的直属子目录

bin :是英语binary的缩写，表示“二进制文件”，bin目录包含了会被所有用户锁使用的可执行程序

boot : boot表示启动 boot目录包含于linux启动密切相关的文件

dev：device的缩写，表示设备，包含外设，里面的所有子目录都对应一个外设

etc : et cetera的缩写，翻译成英语就是and so on ,表示“..等等”，etc目录包含系统的配置文件,

home: 放置私人的文件 私人目录 /home/xiongzhuang

lib : library表示库，lib目录包含被程序锁调用的库文件，例如.so结尾的文件，windows下的库文件则是以.dll结尾

media : 表示媒体 USB ，SD卡，DVD,光盘插入电脑时，linux可以让我们通过media的子目录来访问这些外设中的内容

mnt : mount的缩写 表示挂载临时挂载，临时挂载一些装置

opt : optional application software package 的缩写 表示可选的应用软件包 安装第三方软件

root : 超级用户的目录

sbin : sytsem binary 表示系统二进制文件

srv: service表示服务，表示网络服务启动之后所需要取用的数据

tmp： temporary 表示临时的 普通用户和程序存放临时文件的地方

usr : Unix Software Resource的缩写 Unix操作系统软件资源（类似etc ）

var : variable的缩写表示动态的，可变的，包含log日志文件

## 显示当前目录的路径

pwd:打印出当前目录的路径，Print Working Directory的缩写

which :显示一个命令的对应的可执行程序的位置，每个命令对应的是一个可执行程序，而which则能使我们看到命令所对应的程序在linux中的位置，linux中的可执行程序没有后缀，而window则是以.exe结尾

Linux以点开头的文件都是隐藏文件,不包括. 和..两个文件 .表示当前目录 ..表示上级目录

## 目录相关的命令

Centos的终端默认是有颜色标注的一般来说，

蓝色->目录

绿色->可执行文件

红色->压缩文件

浅蓝色->链接文件

灰色->其他文件

ls 是list的缩写。List是列举列出的意思

ls命令用于列出当前目录下的文件和目录

ls --color=auto来开启颜色标注：ls --color=auto

ls --color=none来关闭颜色标注：ls --color=none

ls -a 列出当前目录下的所有文件（包括隐藏文件）

-A不列出.和..两个目录

ls -i 显示文件的inode

ls -ll 详细列表

ls -lh以Ko,Mo,Go的形式显示文件的大小

ls -ls按照最近的修改时间排序

ls中的参数使用效果是叠加的 比如ls lath

cd 是英文change directory的缩写 表示的是切换目录

cd .. 回到上级目录

回到家目录的三种方法

cd /home/xiongzhuang

cd ~

cd

du命令：disk usage的缩写 表示“磁盘占用” 显示目录包含的文件的大小

du 可以让我们知道文件和目录所占的空间的大小，相比于ls -l命令，du命令统计的才是真正的文件大小，du命令会深入遍历每个目录的子目录，统计所有文件的大小

du -h 以ko,mo,go的形式显示文件大小

du -a 显示目录和文件的大小

du -s 只显示当前目录的总计大小

## 操纵文件

Ctrl+c能够结束大部分终端的命令（类似alt+f4）

cat和less命令：显示文件内容

cat是concatenate的缩写，表示“连接/串联”，cat命令能够一次性在终端中显示文件的所有内容

less能够显示一个终端屏幕所能展示的文件内容，

在less中使用以下快捷键

空格键可以查询下一个终端屏幕的内容，

回车键能够查看下一行的内容，

y键能够后退一行（也可以通过上下键）

d键能够前进半页，

b键能够后退一页，与键盘的PageUp效果一样，

u键能够后退半页，

q键退出文件

=键显示你在文件的什么位置，会显示当前页面的内容是文件中的第几行到第几行，显示当前页占有整个文件的百分之多少以及字节百分比

h键显示帮助文档

/键可以搜索字符在文件中的匹配位置 比如/you 则会查找文件中包含you字符的位置，按n键可以跳向下一个搜索结果，N则会跳向上一个搜索结果

head用于显示文件的开头几行，默认显示前10行

head -n {num} 显示文件开头的前num行

tail命令用于显示文件结尾的几行，默认显示后10行

tail -n {num} 显示文件结尾的后num行

tail -f 追踪文件的更新每过一秒会检查一下文件是否有新内容也可以通过 -s来制定间隔检查的秒数 例如tail -f -s 4 xxxx

touch命令：创建一个空白的文件 “触摸，触碰”的意思

touch 可以创建一个或多个文件

mkdir 创建一个目录 make directory

mkdir也支持创建一个或多个文件

mkdir -p 可以递归的创建文件夹

（创建目录和文件的时候如果有空格可以用””来创建 但是在linux中为了避免不必要的麻烦尽量不要用特殊字符来命名文件或目录）

## 文件的复制和移动操作

cp : copy的缩写，拷贝的意思，可以拷贝单个或多个文件和目录

-r或者-R 拷贝目录 recursive递归

\* 通配符

cp \*.txt folder 把当前目录下的所有txt文件拷贝到folder目录中下

cp ha\* folder 把当前目录下的所有ha开头的文件都拷贝到folder目录下

mv move的缩写标识移动 mv有两个功能 移动文件（目录） 重命名文件（目录）

rm命令 删除文件和目录 可以删除一个或多个文件目录 甚至可以删除linux系统

-i 删除时终端会询问是否删除

-f 终端不会询问用户是否删除（强制删除）

-r 删除目录 并递归删除之下的目录

rmdir 只能删除空目录

rm -rf /\* 或者rm -rf/ 删除系统根目录 极度危险

ln命令：创建连接

Linux下有两种连接类型：

Physical link :物理连接或硬连接

Symbolic link:符号连接或软连接

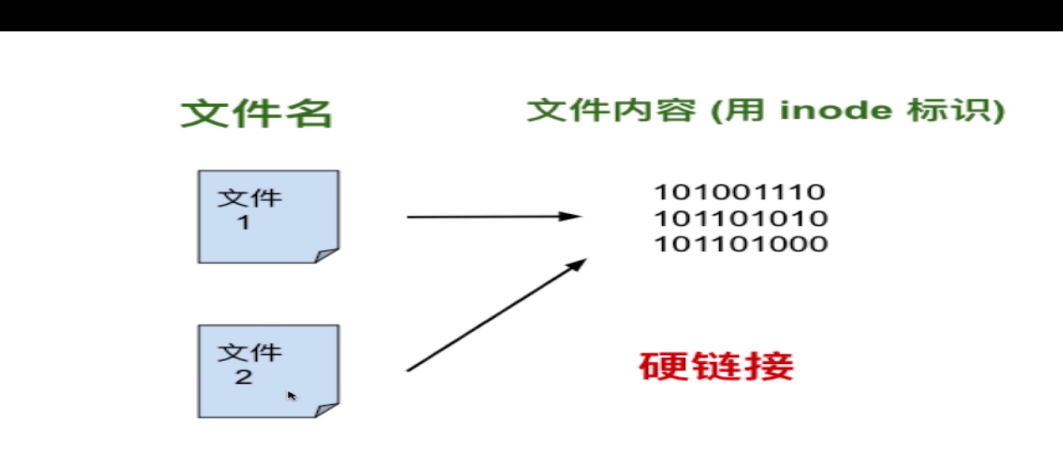
Linux文件存储：文件名和文件内容是分开存放的，方便linux管理

文件有三个部分：文件名，权限和文件内容

每个文件的文件内容被分配至一个表示号码，就是inode,因此每个文件名都绑定到它的文件内容（用inode标识）

ln 创建硬链接 硬链接绑定的是同一个文件 修改硬链接的文件同时也会修改源文件 只是用两个文件名修改同一个文件

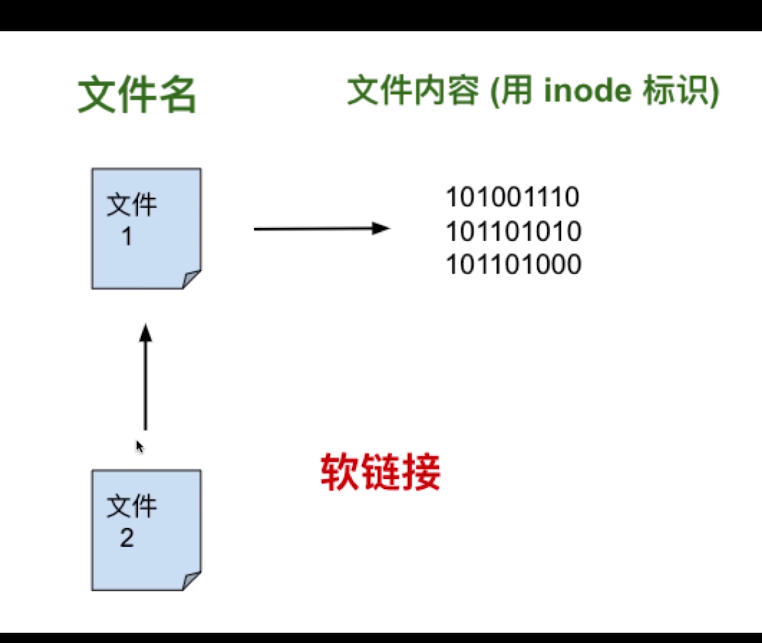
缺陷： 只能创建指向文件的硬链接，不能创建指向目录的，（通过一些参数的修改也可以但是比较复杂 所以创建指向目录的一般都是用软连接）



ln file1 file2 创建一个file1的硬链接名为file2

ln -s file1 file2 创建一个file2的软连接 名为file2

(file2是file1的一个快捷方式，他指向的是file1所以显示的是file1的内容，file2的inode和file1不一样，也就是文件内容不一样)



软连接的特点：

如果删除file1那么file2就会变成死链接，因为只想的文件不见了（硬链接删除其中一个另一个不会受到影响）

软连接可以只想目录，硬链接不行

## 用户与权限

sudo 暂时代替root用户执行root用户权限下的命令

sudo su 切换到root用户（只在当前终端，关闭终端后会重置当前用户）

exit 当在root用户下 执行该命令则会退回到当前用户

su 切换到root用户

su - 切换到root用户并打印登录时间去到root用户的家目录

添加用户

usaradd 添加用户（只能用root用户添加） useradd thomos

passwd 修改用户密码 passwd 密码

userdel 删除用户 userdel thomas

userdel -r或者--remove删除用户并删除家目录 userdel -r thomas

## 用户群组

groupadd 创建一个群组

groupdel 删除一个群组

usermod 修改用户账户

usermod -l 对用户重命名

usermod -g 修改用户群组 usermod -g friends thomas(将thomas放入friends群组)

usermod -G 将用户放入多个群组 usermod -G friends,happy,funny thomas

usermod -aG 在不退出当前群组的同时加入其它群组

groups 获取用户属于哪个群组

chown 改变文件或群组的所有者

chown thomas file.txt 修改file.txt文件的所有者为thomas （群组不变）

chgrp 修改文件群组

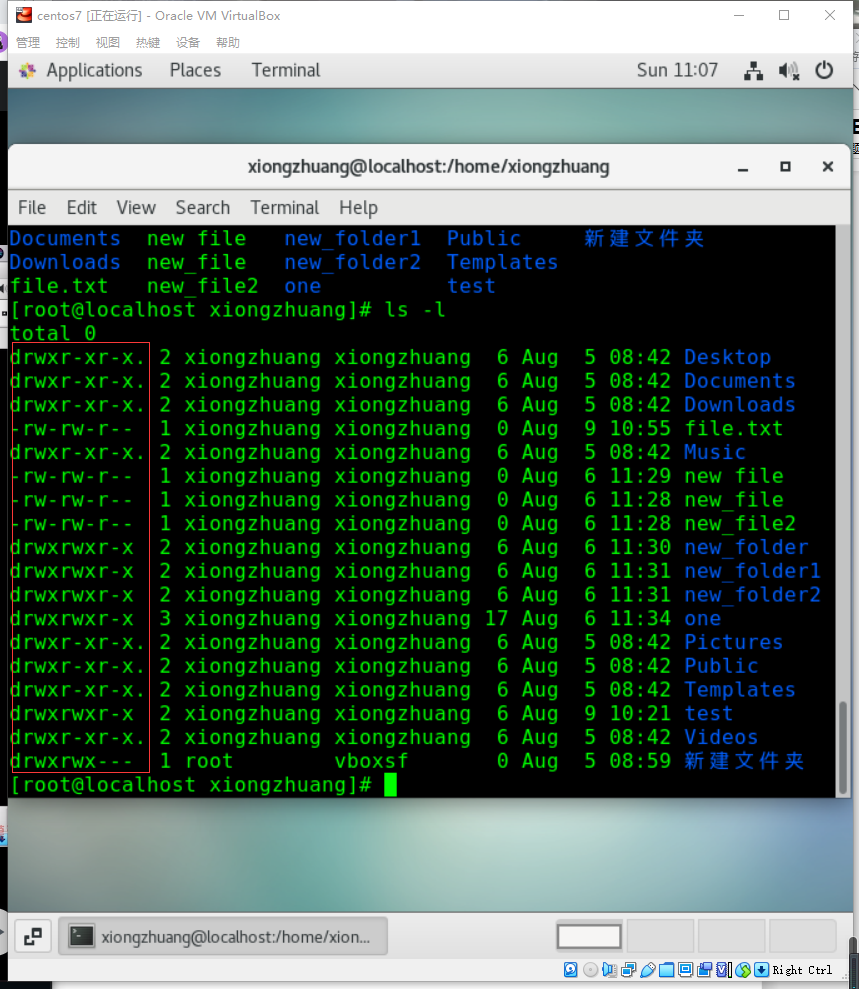
chgrp thomas file.txt 修改file.txt文件的群组为thomas

chown thomas:friends file.txt 修改文件的所有者为thomas，并修改群组为friends

-R 递归设置子目录和子文件

chown -R xiongzhuang:xiongzhuang /home/thomas 将thomas家目录的所有者和群组都改为xiongzhuang

## 文件权限管理



运行ls -l 第一列就是文件访问权限符

d 英语directory的缩写 表示目录 就是说这是一个目录

l 英语link的缩写 表示连接 就是说这是一个连接

r 英语read的缩写 表示读 就是说可以读这个文件

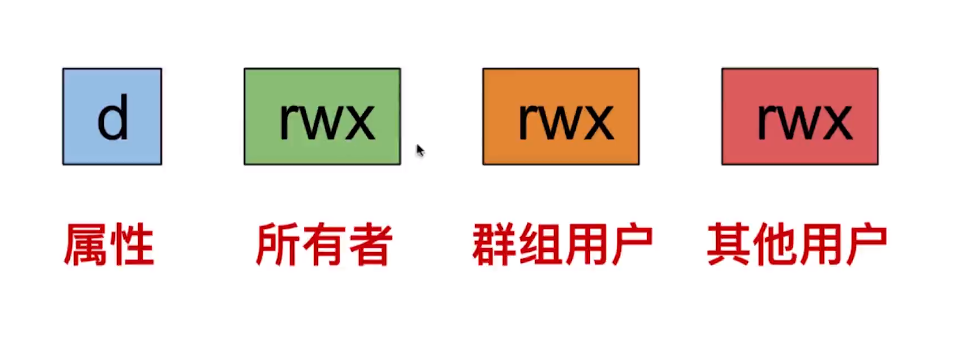
w 英语write的缩写 表示写 就是说可以修改这个文件

x 英语execute的缩写 表示执行 就是说可以运行这个文件

如果x权限在一个目录上的话表示这个目录可以被读，如果同时又r权限就是可以打开此目录来查看其子目录和子文件

如果相应位置有字母表示的是有相应权限 如果是短横- 表示没有 相应的权限

第十一位上的点 表示selinux的安全标签 是一种访问控制体系



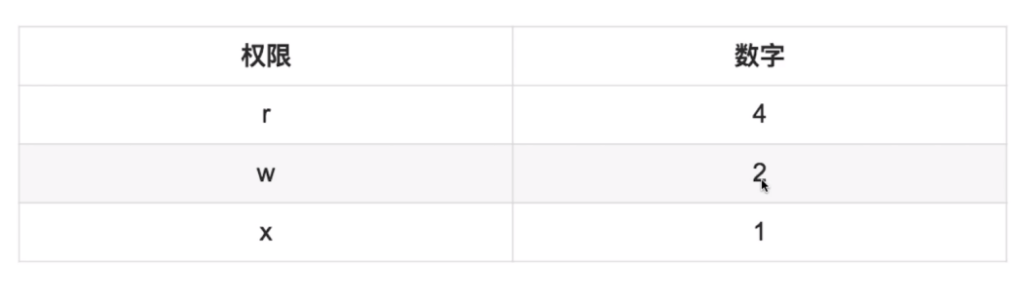
第一组rwx表示文件的所有者对于此文件的访问权限

第二组rwx表示文件所属群组的其他用户对于此文件的访问权限

第三组rwx表示文件除前两组之外的其他用户对于此文件的访问权限

chmod 修改文件的访问权限 change mode

权限所对应的数字



所有可能的权限数字组合



chomd 600 file.txt 设置file.txt文件对于所有者为可读写，对于所在群组的其他用户没有任何权限，对于其他用户没有任何权限

用字母来分配权限

u user的缩写，是英语用户的意思，表示所有者

g group的缩写，是英语群组的意思，表示群组用户

o other的缩写，是英语其他的意思，表示其他用户

a all的缩写，是所有的意思，表示所有用户

与之配合的还有

+ 加号，表示添加的权限

- 减号，表示去除权限

= 等号，表示分配权限

chmod u+rx file 文件file的所有者增加读和运行的权限

chmod g+r file 文件file的群组其他用户增加读的权限

chmod o-r file 文件file的其他用户移除读的权限

chmod g+r, o-r file 文件file的群组其他用户增加读的权限，其他用户移除读的权限

chmod go-r file 文件file的群组其他用户和其他用户均移除读的权限

chmod +x file 文件file的所有用户增加运行的权限

chmod u=rwx,g=r,o=- file 文件的所有者具有读写执行的权限，群组中其他用户具有读的权限，其他用户没有权限

chmod 配合-R 的参数可以递归的修改目录与文件访问权限

## Nano文本编辑器

Ctrl + x 退出文本编辑器

Ctrl + o 保存文件

ctrl + r 打开其他文件

ctrl + y 跳到上一个屏幕

ctrl + v 跳到下一个屏幕

ctrl + k 剪切当前行

ctrl + w 查找文本

ctrl + u 粘贴刚剪切的内容

ctrl + g 打开帮助

ctrl + / 替换

ctrl + F 向前移动一格光标

ctrl + B 向后移动一格光标

ctrl + p 向上移动一行

ctrl + n 向下移动一行

按下esc 再按下x键去除下边的帮助文档

nano file 用nano打开文件file

nano -m file 激活鼠标

nano -i 激活自动缩进的功能

nano -A 激活智能home 键

可以通过在用户家目录中创建.nanorc文件来配置nano

set mouse 激活鼠标

set autoindent 激活自动缩进的功能

set smarthome 激活智能home键

nanorc配置文件 全局nano配置的nanorc文件在etc下

bashrc配置文件 终端的配置文件在etc下名字是bashrc，可以通过bashrc文件来修改终端的展示形式

profile配置文件 配置通过F2到F6切换的需要登录非图形界面的终端的文件

profile文件会调用.bashrc，修改了.bashrc也就是间接的修改了profile文件，profile文件会用profile本身的配置加上.bashrc文件的配置，

修改配置文件时需要重启linux服务器才能生效，如果想要立即生效需要用到source命令重新加载配置文件

RC结尾的是配置文件，通常在系统启动的时候被调用

rc是runcomm的缩写，即运行命令

## 正则表达式和数据操作

grep 筛选数据

grep text file text标识要搜索的文本 file代表要搜索的文件（如果搜索文本有空格就需要用双引号）

grep -i 忽略大小写

grep -n 显示行号

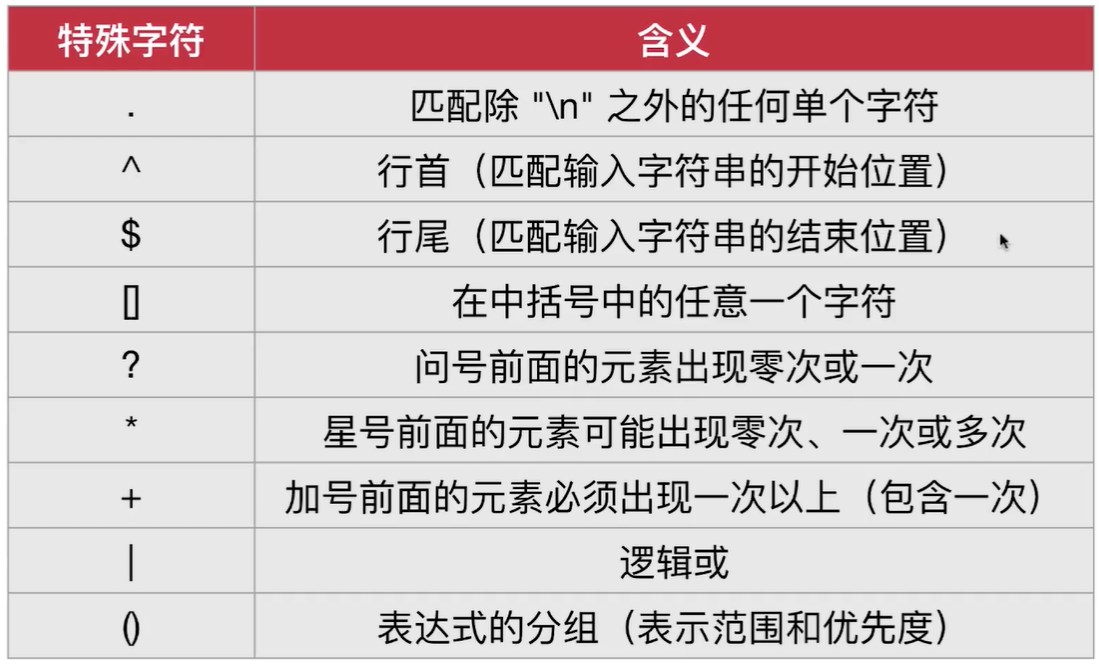
grep -v 只展示没有搜索内容的行

grep -r 在所有子目录与当前目录的文件中搜索

rgrep = grep -r

grep -E 使用正则表达式

egrep = grep -E



sort 为文件排序

sort file 为文件中的文字排序

sort -o target source 将源文件中的文字排序输入到另一个文件

sort -r 倒序排列

sort -R 随机排序

sort -n 对数字排序按照数值的大小



第一列表示行数

第二列表示单词数

第三列标识字节数

wc 文件的统计

wc -l 只对行数统计

wc -w 只对单词统计

wc -c 对字节统计

wc -m 对字符统计

uniq 删除文件的重复的内容 只能删除连续的重复行的内容

uniq source target 删除重复的内容并输出到另一个文件

uniq -c统计重复的行数

uniq -d 只显示重复行的值

cut -c 2-4 file 剪切文件的前2到前4个字符