

find命令复习

\$find 起始目录 查找操作 查找条件.

-name

-user -group

-perm

-(a/c/m) {min/day}

-amin

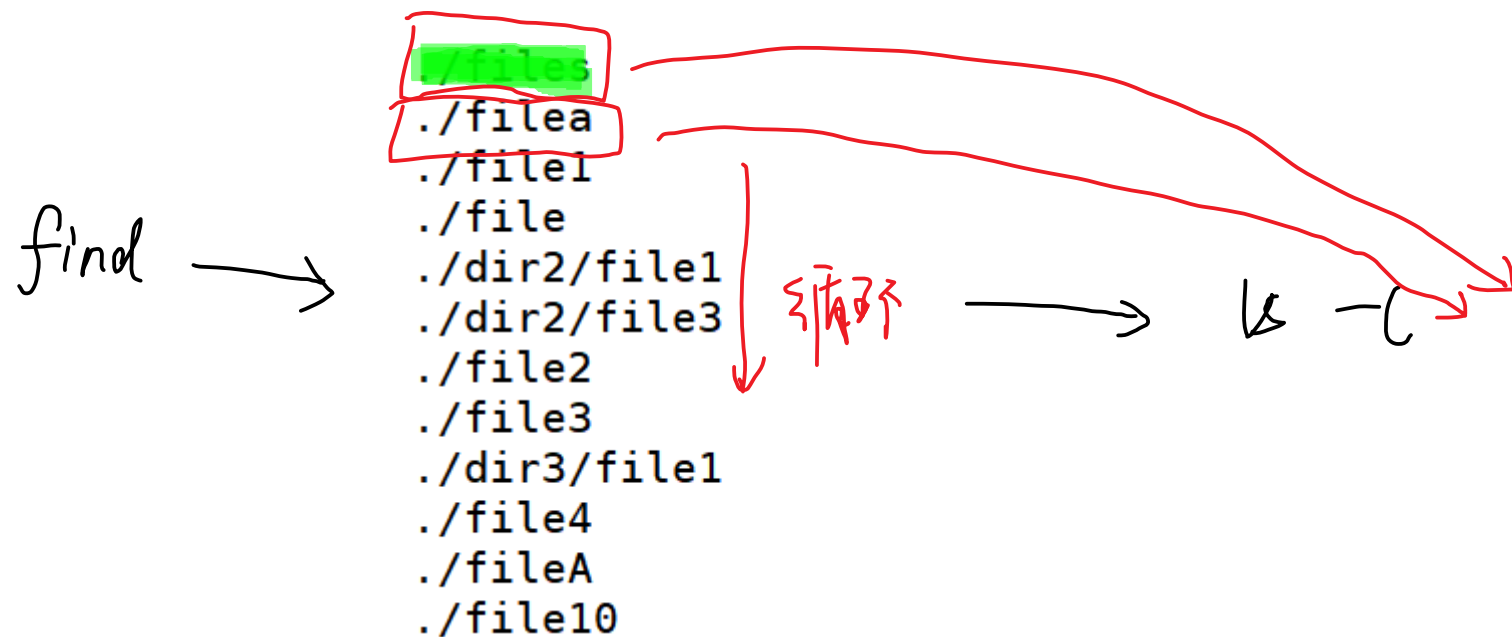
-cday ...

-size

-type

与 -a
或 -o
非 !

find找到的结果执行后续操作



find命令 -exec 命令1 \;

find命令的结果的每一行,作为参数传入命令1.

使用花括号,代表find命令的结果的一行.

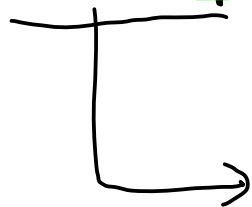
组合操作

```
[liao@ubuntu ~/test]$ find . -name "file*" -exec ls -l {} \;  
-rw-rw-r-- 1 liao liao 0 Apr  7 16:03 ./files  
-rw-rw-r-- 1 liao liao 0 Apr  7 16:03 ./filea  
-rwxrwxr-x 1 liao liao 4 Apr  6 23:39 ./file1  
-rw-rw-r-- 1 liao liao 0 Apr  7 16:00 ./file  
-r--r--r-- 1 liao liao 4 Apr  7 11:31 ./dir2/file1  
-rw-rw-r-- 1 liao liao 6 Apr  7 11:38 ./dir2/file3  
--w--w--w- 1 liao liao 4 Apr  6 23:40 ./file2  
-rw-rw-r-- 1 liao liao 6 Apr  7 11:35 ./file3  
-rw-rw-r-- 1 test1 test1 0 Apr  7 16:16 ./dir3/file1  
-rw-rw-rw- 1 liao liao 0 Apr  7 15:16 ./file4  
-rw-rw-r-- 1 liao liao 0 Apr  7 16:06 ./fileA  
-rw-rw-r-- 1 liao liao 0 Apr  7 16:01 ./file10
```

相当于先find. 再把结果的每一行作为ls-l的参数.

管道

`$ps -elf | grep sshd.`



ps 的输出

管道

grep 的输入



`$ find命令 | xargs ls -l.`

将管道的内容
拆成一行行的单词

```
./files
./filea
./file1
./file
./dir2/file1
./dir2/file3
./file2
./file3
./dir3/file1
./file4
./fileA
./file10
```



管道

统计当前目录有多少文件.

ls | wc -l

wc -l.

磁盘相关的命令

`df` 显示磁盘TS

`$du -h --max-depth=1` 目录.

文件查看和处理

`cat` 文件.

`-b` 对非空行编号

`-E` 显示换行

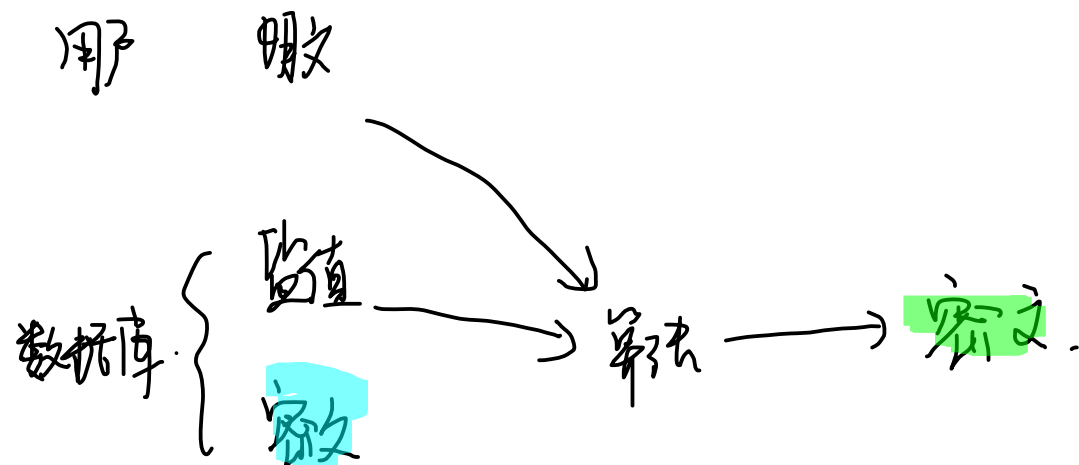
`-n` 对所有的编号.

`-S` 将连续空行合并显示.

加密



liao:\$6\$0em1ZRti\$QaU.fFCE55wxvBdyLNw0M1hTg67zjydR6Qea.y8XyFKlku0ccxJyYkZXJV9kd6WBZpWoN0PyGcECb2KlU7o..1:1:
sshd*:18208:0:99999:7:::
test:\$6\$lRcEofv5\$nmcmMjn4bTqXCxg7aiMts9MRbD5t7WDpkh4PsIPjMplA0ln0rdfjUsVgKpeWN0PD4TY0QWoNtl7tgFiVM.GNGw.:1:
test1:\$6\$h4WKdc0o\$9MDgvqt8XS/ISqo4WPW701rJWycLf8NDo/DPf3ieQV9Wkb4FRZS7s7PcPZed5Xfv8Cbq3UrWvDlcDaic95Ran.:1:



文件的类型

\$file. 比 ls -l 更详细.

标准输入输出

描述符	含义	设备
0	标准输入	键盘
1	标准输出	屏幕
2	错误输出	屏幕

重定向.

```
[liao@ubuntu ~/test]$ echo "how are you"
how are you
```

← 显示在屏幕.

```
[liao@ubuntu ~/test]$ echo "how are you"> file1
[liao@ubuntu ~/test]$ cat file1
how are you
```

↑
重定向原本显示在屏幕.
输入到文件.

ls ... ^{x1} ↓ → 屏幕
重定向 → 文件.

重定向

> 重定向标准输出

< 重定向标准输入

2> 重定向标准错误

>> 将结果追加尾部

<< cat 以什么结束

&> 将标准输出和错误同时输出

重定向

```
puts("this is stdout!");  
fputs("this is stderr!\n", stderr);  
return 0;
```

```
[liao@ubuntu ~/test]$ ./a.out  
this is stdout!  
this is stderr!
```

重定向 stdout

```
[liao@ubuntu ~/test]$ ./a.out > file1  
this is stderr!
```

重定向 stderr

```
[liao@ubuntu ~/test]$ ./a.out 2> file1  
this is stdout!  
[liao@ubuntu ~/test]$ cat file1  
this is stderr!
```

同时重定向

```
[liao@ubuntu ~/test]$ ./a.out &> file1  
[liao@ubuntu ~/test]$ cat file1  
this is stderr!  
this is stdout!
```

重定向

```
char str[20];  
scanf("%s",str);  
puts("content is:");  
puts(str);
```

```
[liao@ubuntu ~/test]$ ./a.out  
howareyou  
content is:  
howareyou  
[liao@ubuntu ~/test]$ ./a.out < file1  
content is:  
this
```

cat命令

\$cat

```
while( fgets() != NULL)
```

```
{
```

```
    puts(str)
```

```
}
```

输出 Ctrl+D (EOF)

\$cat >file1

直接按Ctrl+D

得到空文件.

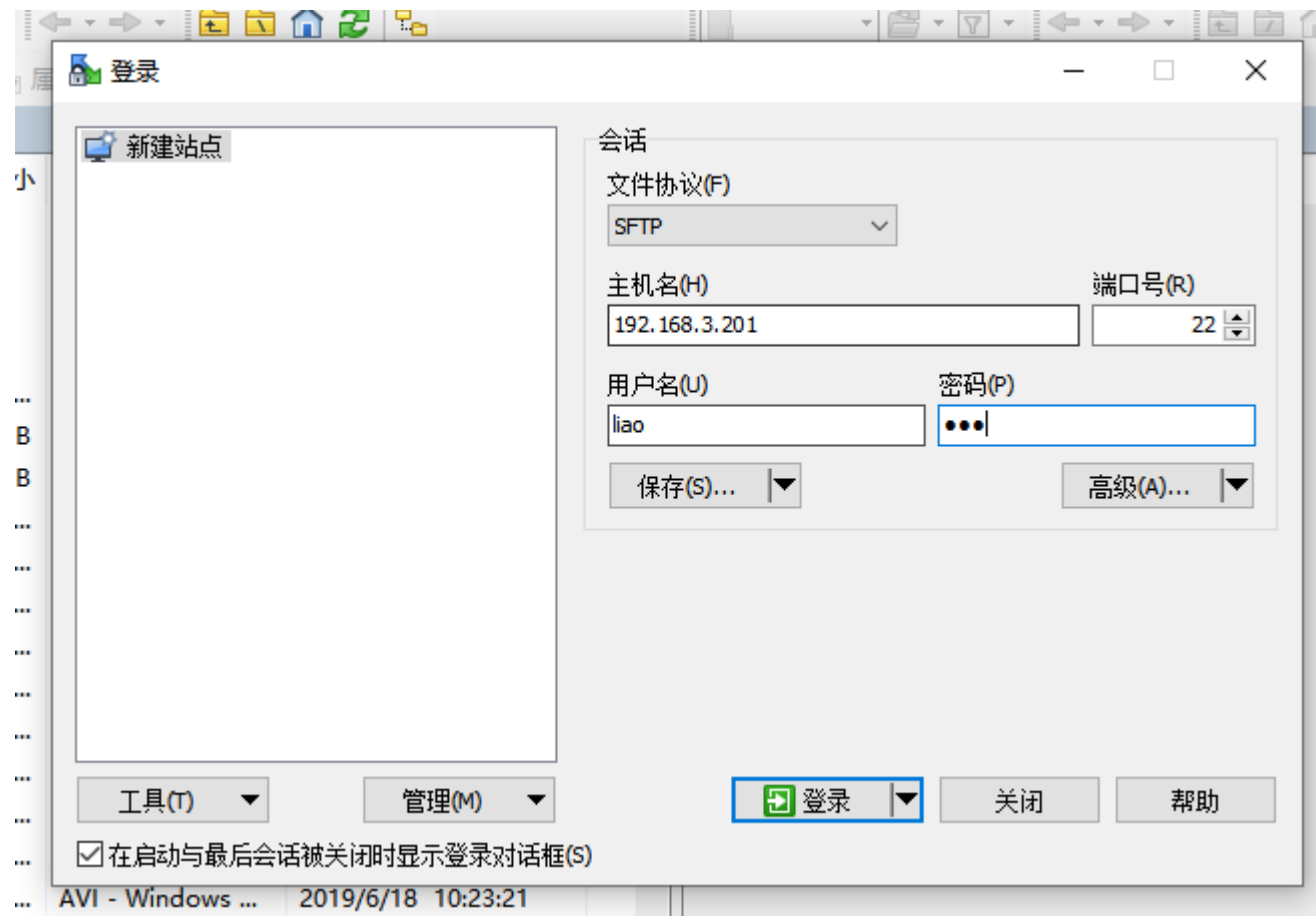
创建空文件

2. `$echo -n >file2.`

3. `$touch file3`

4. `$vim file4.` `ESC → :wq`

winscp



部分显示

\$head 显示前10行

-n x 显示x行

\$tail 显示后10行.

```
history> day01_liuwen
```

作业: 将操作记录的后200行放在文件里面.

单页显示

\$more 文件.

\$less 文件



{ b 上页
f 下页
g 退出
h 帮助

排序

\$sort

字典序

去重

\$uniq

连续重名.

文件字数统计

\$wc

行

单词

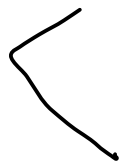
字节

```
[liao@ubuntu ~/test]$ wc -l The_Holy_Bible.txt  
32432 The_Holy_Bible.txt
```

```
[liao@ubuntu ~/test]$ wc -w The_Holy_Bible.txt  
824692 The_Holy_Bible.txt
```

```
[liao@ubuntu ~/test]$ wc -c The_Holy_Bible.txt  
4351658 The_Holy_Bible.txt
```

编码

中.  ^{2Byte}
GBK GB2313
utf-8
^{3Byte}

cc cc \Rightarrow 烫.

\$iconv -f 原 -t 目标 文件名.