

19. 正则三剑客 - sed

笔记本： 优秀笔记

创建时间： 2018/12/28 8:57

更新时间： 2018/12/28 9:19

作者： 306798658@qq.com

sed工具介绍（替换、及替换指定的字符）

sed介绍：我们使用sed工具主要是它的查找和替换功能，当然，它还有其他更丰富的功能。

sed命令格式：sed -n 'n' p filename，单引号内的n是一个数字，表示第几行。

-n选项的作用就是只显示我们要打印的行，无关紧要的内容不显示。

创建一个sed目录，把passwd文件拷贝进来，方便后面学习。

```
[root@localhost ~]# mkdir sed #创建一个目录
[root@localhost ~]# cd sed/ #进入到目录sed
[root@localhost sed]# cp /etc/passwd test.txt
#拷贝passwd文件到本目录下，并且修改名字
[root@localhost sed]# ls #查看当前目录下都有哪些文件
test.txt
[root@localhost sed]# pwd #当前所在的位置
/root/sed
```

只显示我们需要查找的内容显示出来，示例如下：-n

```
[root@localhost sed]# sed -n '/root/'p test.txt root:x:0:0:root:/root:/bin/bash
operator:x:11:0:operator:/root:/sbin/nologin #n选项是指显示我们要打印的行。
```

在grep中使用的特殊字符如（^ \$ *等）都可以在sed中使用。

指定一个区间打印，比如指定一个区域2-3行，示例如下：-n

```
[root@localhost sed]# sed -n '2,3'p test.txt bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
```

sed命令加上-e选项可以实现多个行为，示例如下：-e

```
[root@localhost sed]# sed -e '1'p -e '/bus/'p -n test.txt root:x:0:0:root:/root:/bin/bash dbus:x:81:81:System
message bus:/:/sbin/nologin #sed -e 实现多个行为，如我查找第一行，并且把文档中含有bus的显示出来
```

如果需求说查找一个文件，把bus不管大小写都列出来：I（大写字母i）

```
[root@localhost sed]# sed -n '/bus/'lp test.txt #这里一定要加一个大写I字母。 dbus:x:81:81:System
message bus:/:/sbin/nologin
BUS
```

删除某些行（并非删除），把剩下的行数都显示出来，示例如下：d

```
[root@localhost sed]# sed '1,18'd test.txt chrony:x:998:996::/var/lib/chrony:/sbin/nologin
BUS
#这里后面跟选项d,会把1到18行的内容删除，把后面的内容显示出来。
```

这里参数d表示删除的东西，它不仅可以删除指定的单行以及多行，而且可以删除匹配某个字符的行，还可以删除从某一行开始到文档最后一行的所有行，这个命令仅仅是在屏幕上并不显示这些行。

sed真正删除选中的行数，示例如下：-i

```
首先先拷贝一份文件过来：
[root@localhost sed]# cp test.txt test.txt.bak #拷贝过来个备份
[root@localhost sed]# ls #查看当前目录下的文件
test.txt test.txt.bak
[root@localhost sed]# wc -l test.txt #wc查看一个文件有多少行 20 test.txt
[root@localhost sed]# sed -i '1,18'd test.txt #i删除1到18行。
[root@localhost sed]# wc -l test.txt #再来看这个文件有多少行。 2 test.txt
```

-i还可以指定删除某个字符的行，示例如下：//

```
[root@localhost sed]# sed '/root/'d test.txt chrony:x:998:996::/var/lib/chrony:/sbin/nologin
BUS
```

全局指定替换1，10行。s

```
[root@localhost sed]# sed '1,10s/root/toor/g' test.txt
#1, 10是范围，后面跟s是替换，/root/替换成toor/, g表示全局替换。 toor:x:0:0:toor:toor:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin daemon:x:2:2:daemon:/sbin:/sbin/nologin adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
```

上述中的参数s就表示替换的动作，参数g表示本行全局替换，如果不加g只会替换本行出现的第一个。

调整两个字符串的位置

扩展正则选项（类似grep的-E）：-r

第一段和最后一段替换位置

```
# head -5 /etc/passwd |sed -r 's/([^:]+):(.*):/\3:\2:\1/'
/bin/bash:x:0:0:root:/root:root
/sbin/nologin:x:1:1:bin:/bin:bin
/sbin/nologin:x:2:2:daemon:/sbin:daemon
/sbin/nologin:x:3:4:adm:/var/adm:adm
/sbin/nologin:x:4:7:lp:/var/spool/lpd:lp
如果不加-r,这个( )就需要脱义了。
# head -5 /etc/passwd |sed 's/\([^:]+\):(\.*):/\3:\2:\1/'
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
```

\1、\2或\3表示第一部分（）里面的字符、第二部分（）里面的字符、第三部分（）里面的字符，-r选项让这个表达式更加清晰了。

```
[root@localhost sed]# head -5 test.txt | sed 's/[a-zA-Z]//g' #这里括号里的是范围 ::0:0:/:/:/
::1:1:/:/:/ ::2:2:/:/:/ ::3:4:/:/:/ ::4:7:/:/:/ 说明：这样相当于删除了所有的英文字母
```

如果给文档前面统一添加个字符串，比如aaa，示例如下：