

文本处理三剑客之SED

一.Sed介绍

- sed是linux下出名的行编辑器(Stream Editor)

简介: sed是一种流编辑器,它一次处理一行内容。处理时,把当前处理的行存储在临时缓冲区中,称为“模式空间”(pattern space),接着用sed命令处理缓冲区中的内容,处理完成后,把缓冲区的内容送往屏幕。然后读入下行,执行下一个循环。如果没有使用诸如‘D’的特殊命令,那会在两个循环之间清空模式空间,但不会清空保留空间。这样不断重复,直到文件末尾。文件内容并没有改变,除非你使用重定向存储输出。
sed主要用来自动编辑一个或多个文件,简化对文件的反复操作,编写转换程序等

[sed详细参考手册](#)

二.Sed用法

1.基本选项功能和用法

```
sed [option]... 'script' inputfile...  
-n 不输出模式空间内容到屏幕,即不自动打印  
-e 多点编辑  
-f /PATH/SCRIPT_FILE 从指定文件中读取编辑脚本  
-r 支持使用扩展正则表达式  
-i.bak 备份文件并原处编辑  
script: '地址命令'  
注意: sed的脚本、地址定界和命令都一般写在单引号中间
```

2.地址定界和编辑命令

- 地址定界

- (1) 不给地址: 对全文进行处理
- (2) 单地址:
 - #: 指定的行
 - #: 最后一行
 - /pattern/: 被此处模式所能够匹配到的每一行
- (3) 地址范围:
 - #, #
 - #, + #
 - /pat1/, /pat2/ #模式1到模式2之间的行
 - #, /pat1/
- (4) ~: 步进

1~2 奇数行
2~2 偶数行

- 编辑命令

d 删除模式空间匹配的行，并立即启用下一轮循环
p 打印当前模式空间内容，追加到默认输出之后
a [\]text 在指定行后面追加文本，支持使用\n实现多行追加
i [\]text 在行前面插入文本
c [\]text 替换行为单行或多行文本
w /path/file 保存模式匹配的行至指定文件
r /path/file 读取指定文件的文本至模式空间中匹配到的行后
= 为模式空间中的行打印行号
! 模式空间中匹配行取反处理

3.搜索替换

s///
查找替换,支持使用其它分隔符, s@@@, s####
替换标记:
g 行内全局替换
p 显示替换成功的行
w /PATH/FILE 将替换成功的行保存至文件中

4.sed示例

```
sed '2p' /etc/passwd
sed -n '2p' /etc/passwd
sed -n '1,4p' /etc/passwd
sed -n '/root/p' /etc/passwd
sed -n '2,/root/p' /etc/passwd 从2行开始
sed -n '/^$/=' file 显示空行行号
sed -n -e '/^$/p' -e '/^$/=' file
Sed '/root/a\superman' /etc/passwd行后
sed '/root/i\superman' /etc/passwd 行前
sed '/root/c\superman' /etc/passwd 代替行

sed '1,10d' file
nl /etc/passwd | sed '2,5d'
nl /etc/passwd | sed '2a tea'
sed 's/test/mytest/g' example
sed -n 's/root/&superman/p' /etc/passwd 单词后
sed -n 's/root/superman&/p' /etc/passwd 单词前
sed -e 's/dog/cat/' -e 's/hi/lo/' pets
sed -i.bak 's/dog/cat/g' pets
```

三.Sed高级用法

- 高级编辑命令

P: 打印模式空间开端至\n内容，并追加到默认输出之前

h: 把模式空间中的内容覆盖至保持空间中

H: 把模式空间中的内容追加至保持空间中

g: 从保持空间取出数据覆盖至模式空间

G: 从保持空间取出内容追加至模式空间

x: 把模式空间中的内容与保持空间中的内容进行互换

n: 读取匹配到的行的下一行覆盖至模式空间

N: 读取匹配到的行的下一行追加至模式空间

d: 删除模式空间中的行

D: 如果模式空间包含换行符，则删除直到第一个换行符的模式空间中的文本，并不会读取新的输入行，而使用合成的模式空间重新启动循环。如果模式空间不包含换行符，则会像发出d命令那样启动正常的新循环

高级用法sed示例

```
sed -n 'n;p' FILE
sed '1!G;h;$!d' FILE
sed 'N;D' FILE
sed '$!N;$!D' FILE
sed '$!d' FILE
sed 'G' FILE
sed 'g' FILE
sed '/^$/d;G' FILE
sed 'n;d' FILE
sed -n '1!G;h;$p' FILE
```

- 练习

1、删除centos7系统/etc/grub2.cfg文件中所有以空白开头的行行首的空白字符

```
sed -inr 's#^+(.*)#\1#p' /etc/grub2.cfg
```

2、删除/etc/fstab文件中所有以#开头，后面至少跟一个空白字符的行的行首的#和空白字符

```
sed -inr 's#^#+(.*)#\1#p' /etc/grub2.cfg
```

3、在centos6系统/root/install.log每一行行首增加#号

```
sed -inr 's#.*#\&#p' /etc/grub2.cfg
```

4、在/etc/fstab文件中不以#开头的行的行首增加#号

```
sed -inr 's#^[^#](.*)#\&\1#p' /etc/grub2.cfg
```

5、处理/etc/fstab路径,使用sed命令取出其目录名和基名

```
echo /etc/fstab/ | sed -nr 's#(.*/)([^/]+)/?#\1#p'
echo /etc/fstab/ | sed -nr 's#(.*/)([^/]+)/?#\2#p'
```

6、利用sed 取出ifconfig命令中本机的IPv4地址

```
ifconfig eth0|sed -nr '2s@[^0-9]+([.0-9]+).*\1@p'
```

7、统计centos安装光盘中Package目录下的所有rpm文件的以.分隔倒数第二个字段的重复次数

```
ls /misc/cd/{BaseOS,AppStream}/Packages/ | sed -nr 's#.*\.(.*)\.rpm#\1#p' |  
sort |uniq -c | sort -nr  
ls /misc/cd/{BaseOS,AppStream}/Packages/ | rev | cut -d. -f2 | sort | uniq -c  
| uniq
```

8、统计/etc/init.d/functions文件中每个单词的出现次数，并排序（用grep和sed两种方法分别实现）

```
grep -Eow "[^[:punct:]]+([[:alpha:]]0-9+)" /etc/init.d/functions | grep -Eow "  
[[:alpha:]]+|[0-9]+" | wc -l  
sed -r "s/^[[:alpha:]]/\n/g" /etc/init.d/functions | sort | uniq -c | sort -nr
```

9、将文本文件的n和n+1行合并为一行，n为奇数行

```
seq 10 | sed -nr '1~2N;s/\n/ /p'
```