# 简介

sed英文全称是stream editor。由贝尔实验室开发，如今主流Unix/Linux操作系统上都集成了这个工具。sed由自由软件基金组织（FSF）开发和维护，并且随着GNU/Linux进行分发，通常它也称作 GNU sed。

sed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。

# 用法

## 语法

sed [OPTION]... {script-only-if-no-other-script} [input-file]...

## 选项描述

-n 不打印模式空间

-e 执行脚本、表达式来处理

-f 执行动作从文件读取执行

-i 修改原文件

-i.bak 备份文件并原处编辑

-r 使用扩展正则表达式

## 命令描述

s/regexp/replacement/ 替换字符串

p 打印当前模式空间

P 打印模式空间的第一行

d 删除模式空间，开始下一个循环

D 删除模式空间的第一行，开始下一个循环

= 打印当前行号

a \text 当前行追加文本

i \text 当前行上面插入文本

c \text 所选行替换新文本

q 立即退出 sed 脚本

r 追加文本来自文件

: label label 为 b 和 t 命令

b label 分支到脚本中带有标签的位置，如果分支不存在则分支到脚本的末尾

t label 如果 s///是一个成功的替换，才跳转到标签

h H 复制/追加模式空间到保持空间

g G 复制/追加保持空间到模式空间

x 交换模式空间和保持空间内容

l 打印模式空间的行，并显示控制字符$

n N 读取/追加下一行输入到模式空间

w filename 写入当前模式空间到文件

! 取反、否定

& 引用已匹配字符串

## 地址描述

first~step 步长，每 step 行，从第 first 开始

$ 匹配最后一行

/regexp/ 正则表达式匹配行

number 只匹配指定行

addr1,addr2 开始匹配 addr1 行开始，直接 addr2 行结束

addr1,+N 从 addr1 行开始，向后的 N 行

addr1,~N 从 addr1 行开始，到 N 行结束

# 样例

## 匹配打印（p）

### 打印匹配my开头的行

$ cat test.txt |sed -n '/^my/p'

my cat's name is betty

my dog's name is frank

my fish's name is george

my goat's name is adam

### 打印第n行

$ cat test.txt |sed -n '3p'

my dog's name is frank

### 打印第一行至第三行

$ cat test.txt |sed -n '1,3p'

my cat's name is betty

This is your dog

my dog's name is frank

### 打印奇数行

$ cat test.txt |sed -n '1~2p'

my cat's name is betty

my dog's name is frank

my fish's name is george

my goat's name is adam

### 打印匹配行及后一行

$ cat test.txt |sed -n '/^my/,+1p'

my cat's name is betty

This is your dog

my dog's name is frank

This is your fish

my fish's name is george

This is your goat

my goat's name is adam

### 打印最后一行

$ cat test.txt |sed -n '$p'

my goat's name is adam

### 不打印最后一行

$ cat test.txt |sed -n '$!p'

my cat's name is betty

This is your dog

my dog's name is frank

This is your fish

my fish's name is george

This is your goat

### 匹配范围

$ cat test.txt |sed -n '/fish/,/adam$/p'

This is your fish

my fish's name is george

This is your goat

my goat's name is adam

### 匹配开头行到最后一行

$ cat test.txt |sed -n '/fish/,$p'

### 引用系统变量，用引号

$ cat test.txt |sed -n '/'$a'/p'

my cat's name is betty

my dog's name is frank

my fish's name is george

my goat's name is adam

## 匹配删除（d）

### 删除匹配行

sed '/my/d'

### 删除指定行

sed '1d' 删除第一行

sed '1,3d' 删除1到2行

sed '1~2d' 删除奇数行

sed '/^#/d;/^$/d' 删除空行或开头#号的行

## 替换（s///）

### 替换my符串为 your

$ sed 's/my/your/' test.txt

your cat's name is betty

This is your dog

your dog's name is frank

This is your fish

your fish's name is george

This is your goat

your goat's name is adam

### 对1-4行的my进行替换

$ sed '1,4s/my/your/' test.txt

### 全局替换加 g

$ sed 's/my/your/g' test.txt

### 使用&命令引用匹配内容并替换

$ sed 's/my/&your/g' test.txt

等于

$ sed 's/my/myyour/g' test.txt

#### IP 加单引号

$ echo '10.10.10.1 10.10.10.2 10.10.10.3' |sed -r 's/[^ ]+/"&"/g'

"10.10.10.1" "10.10.10.2" "10.10.10.3"

### 对匹配进行替换

$ cat test.txt|sed -n '/my/s/my/test/p'

test cat's name is betty

test dog's name is frank

test fish's name is george

test goat's name is adam

### 二次匹配替换

$ cat test.txt|sed -n 's/my/test/;s/name/hello/p'

test cat's hello is betty

test dog's hello is frank

test fish's hello is george

test goat's hello is adam

### 分组使用，在每个字符串后面添加

#### 仅输出特定组

$ echo "ab"|sed -rn 's/(a.\*)(.\*b)/\1/p'

a

#### 顺序倒换

$ echo "ab"|sed -rn 's/(a.\*)(.\*b)/\2\1/p'

Ba

### 注释匹配行后的多少行

$ sed '3,+3s/^/#/' test.txt

my cat's name is betty

This is your dog

#my dog's name is frank

#This is your fish

#my fish's name is george

#This is your goat

my goat's name is adam

### 注释指定多行

$ sed -r 's/^my fish|^my goat/&#/' test.txt

my cat's name is betty

This is your dog

my dog's name is frank

This is your fish

my fish#'s name is george

This is your goat

my goat#'s name is adam

### 行首添加

$ sed 's/^/+++/' test.txt

+++my cat's name is betty

+++This is your dog

+++my dog's name is frank

+++This is your fish

+++my fish's name is george

+++This is your goat

+++my goat's name is adam

### 行末追加

$ sed 's/$/+++/' test.txt

my cat's name is betty+++

This is your dog+++

my dog's name is frank+++

This is your fish+++

my fish's name is george+++

This is your goat+++

my goat's name is adam+++

### 去除开头和结尾空格或制表符

$ echo " 1 2 3 " |sed 's/^[ \t]\*//;s/[ \t]\*$//'

1 2 3

### 备份+编辑

$ sed -i.bak 's/my/your/' a.txt

-rw-r--r-- 1 10431 197609 155 8月 18 17:52 a.txt

-rw-r--r-- 1 10431 197609 147 8月 18 17:52 a.txt.bak

## 多重编辑（-e）

先删除后替换

$ sed -e '1,2d' -e 's/my/your/' test.txt

your dog's name is frank

This is your fish

your fish's name is george

This is your goat

your goat's name is adam

也可以使用;

$ sed -e '1,2d;s/my/your/' test.txt

your dog's name is frank

This is your fish

your fish's name is george

This is your goat

your goat's name is adam

## 添加新内容（a、i 和 c）

### 在上一行添加 i

$ sed '/my goat/i \hello world' test.txt

my cat's name is betty

This is your dog

my dog's name is frank

This is your fish

my fish's name is george

This is your goat

hello world

my goat's name is adam

### 下一行添加 b

$ sed '/my goat/b \hello world' test.txt

### 替换新行 c

$ sed '/my goat/c hello world' test.txt

### 在指定行下一行添加一行

$ sed '2a \hello world' test.txt

my cat's name is betty

This is your dog

hello world

my dog's name is frank

This is your fish

my fish's name is george

This is your goat

my goat's name is adam

### 读取文件并追加到匹配行后（r）

$ echo ========|sed '/===/r test.txt'

========

my cat's name is betty

This is your dog

my dog's name is frank

This is your fish

my fish's name is george

This is your goat

my goat's name is adam

### 将匹配行写到文件（w）

cat test.txt|sed '/my/w test1.txt'

### 每3行替换一次

$ sed 'n;n;s/^/=/;s/$/=/' test.txt

my cat's name is betty

This is your dog

=my dog's name is frank=

This is your fish

my fish's name is george

=This is your goat=

my goat's name is adam

## 标签（: b 和 t）

标签可以控制流，实现分支判断。

: lable name 定义标签

b lable 跳转到指定标签，如果没有标签则到脚本末尾

t lable 跳转到指定标签，前提是 s///命令执行成功

### 将换行符替换成逗号

$ sed 'N;s/\n/,/' test.txt

my cat's name is betty,This is your dog

my dog's name is frank,This is your fish

my fish's name is george,This is your goat

my goat's name is adam

上面没有完全替换掉

sed ':a;N;s/\n/,/;b a'

### 每三个数字加个一个逗号

$ echo "123456789" |sed -r ':a;s/([0-9]+)([0-9]+{3})/\1,\2/;t a'

123,456,789

## 统计行数

$ sed -n '$=' test.txt

7