

sed 快速参考

流编辑、替换、地址、保持空间与原地变换

基础

运行 sed

```
sed 's/old/new/' file.txt      # substitute first match per line
sed 's/old/new/g' file.txt    # substitute all matches per line
sed -n '5p' file.txt          # print only line 5
sed '3d' file.txt             # delete line 3
echo "hello" | sed 's/hello/hi/' # pipe input
```

命令行标志

-n	抑制自动打印; 仅在使用 p 时打印
-e 'cmd'	执行 sed 命令 (可用多个 -e 链式调用)
-f script.sed	从文件读取命令
-i[suffix]	原地编辑文件 (可选备份后缀)
-E / -r	使用扩展正则表达式

替换

替换语法

```
sed 's/foo/bar/' f           # first occurrence per line
sed 's/foo/bar/g' f         # all occurrences
sed 's/foo/bar/3' f         # 3rd occurrence only
sed 's/foo/bar/gi' f        # all, case-insensitive
sed 's|usr/bin|opt/bin|g' f # alternate delimiter
```

替换标志

g	替换该行所有匹配
N (number)	仅替换第 N 个匹配
p	替换成功时打印该行
w file	将替换后的行写入文件
i / I	大小写不敏感匹配 (GNU)

地址

地址示例

```
sed '3s/a/b/' f           # only on line 3
sed '2,5s/a/b/' f        # lines 2 through 5
sed '/^#/d' f            # delete lines starting with #
sed '/start/,/end/d' f   # delete range between patterns
sed '1~2d' f             # delete odd-numbered lines (GNU)
```

地址类型

N	第 N 行
\$	最后一行
N,M	第 N 行到第 M 行
/regex/	匹配正则的行
/regex1/,/regex2/	从第一个匹配到第二个匹配的范围
N~step	从第 N 行开始, 每隔 step 行 (GNU)
addr!	取反——应用于不匹配的行

删除与打印

删除与打印命令

```
sed '5d' f                # delete line 5
sed '/^$/d' f            # delete blank lines
sed -n '10,20p' f        # print lines 10-20
sed -n '/error/p' f      # print lines matching pattern
sed '/debug!d' f         # keep only matching lines
```

命令参考

d	删除模式空间, 开始下一个循环
D	删除模式空间中第一个换行前的内容
p	打印模式空间
P	打印模式空间中第一个换行前的内容
q	打印当前模式空间后退出
Q	不打印直接退出 (GNU)

插入与追加

插入、追加与替换

```
sed '3i\inserted line' f    # insert before line 3
sed '3a\appended line' f   # append after line 3
sed '3c\replaced line' f   # replace line 3
sed '/marker/a\new line' f  # append after pattern match
```

命令

i\text	在当前行前插入文本
a\text	在当前行后追加文本
c\text	用文本替换当前行
r file	读取文件内容并追加
R file	从文件读取一行并追加 (GNU)
w file	将模式空间写入文件

保持空间

保持空间命令

h	将模式空间复制到保持空间
H	将模式空间追加到保持空间
g	将保持空间复制到模式空间
G	将保持空间追加到模式空间
x	交换模式空间和保持空间

保持空间示例

```
sed -n '1!G;h;$p' f        # reverse lines (tac)
sed '/^$/ { x; s/\n//; x; }' f # collapse hold on blank
sed -n 'H;${x;s/\n/ /g;p;}' f # join all lines with space
```

多命令

链式命令

```
sed -e 's/foo/bar/g' -e 's/baz/qux/g' f
sed 's/foo/bar/g; s/baz/qux/g' f
sed '/header/{ s/old/new/; s/foo/bar/; }' f
sed -f commands.sed input.txt
```

分组与分支

{ cmd1; cmd2; }	对同一地址执行多个命令
:label	定义分支标签
b label	跳转到标签
t label	若上一次 s/// 成功则跳转
T label	若上一次 s/// 失败则跳转 (GNU)

原地编辑

原地编辑示例

```
sed -i 's/old/new/g' file.txt      # edit in place (GNU)
sed -i.bak 's/old/new/g' file.txt  # backup as file.txt.bak
sed -i '' 's/old/new/g' file.txt   # macOS in-place (no backup)
sed -i '/^#/d' config.txt          # remove comments in place
```

平台注意事项

GNU sed -i	后缀可选; 单独 -i 不创建备份
BSD/macOS sed -i	需要后缀参数; 用 -i '' 表示不备份
-i.bak	创建 .bak 扩展名的备份文件
多文件	sed -i 's/a/b/g' *.txt 编辑所有匹配文件

正则表达式

sed 中的正则

.	任意单个字符
*	0 个或多个前置字符
\+	一个或多个 (BRE) —— ERE 中为 +
\?	0 个或 1 个 (BRE) —— ERE 中为 ?
^	行首
\$	行尾
[abc]	字符类
(\)	捕获组 (BRE) —— ERE 中为 ()
\1, \2	反向引用捕获组
&	整个匹配字符串 (用于替换部分)

正则示例

```
sed 's/[0-9]\+/NUM/g' f      # replace numbers
sed -E 's/(foo)(bar)\2\1/g' f # swap groups (ERE)
sed 's.//(&)/' f             # wrap line in parens
sed 's/[ \t]*$//' f          # strip trailing whitespace
```

常用模式

单行命令

```
sed -n '1p' f                # first line (head -1)
sed '$!d' f                  # last line (tail -1)
sed '/^$/d' f                 # remove blank lines
sed 's/^[ \t]*$//' f          # strip leading whitespace
sed '= f | sed 'N;s/\n\t/' f  # number lines
```

实用技巧

Double-space	sed G —— 每行后追加空行
Remove HTML tags	sed 's/<[>]*>/g'
Extract emails	sed -nE 's/.*([a-z]+@[a-z.]+).*/\1/p'
Comment lines	sed 's/^/# /' —— 每行前加 #
Trim blank lines at EOF	sed -e 'a -e '/^\$/{s/.\$//;N;ba -e '}'
Replace nth line	sed 'Nc\new text' —— 替换第 N 行