sed是一种流编辑器工具，它在文本的编辑处理中发挥着相当中重要的作用，能够完美的配合正则表达式来匹配文本内容并进行修改。在处理文本时，以行为单位进行处理，首先将当前要处理的行存储在临时缓冲区中，称为“模式空间”（pattern space）,然后用sed命令处理缓冲区中的内容，处理完成后把缓冲区中的内容输送到屏幕上，接着处理下一行，这样不断重复，直到文件末尾，源文件内容并没有改变，除非使用重定向存储输出，或者使用-i参数。主要用来自动编辑一个或多个文件，减少重复操作，简化对文件的处理。

命令格式

sed [options] 'command' input\_file(s)

sed [options] -f scriptfile input\_file(s)

其中参数（options）:

|  |  |
| --- | --- |
| -e<script>或--expression=<script> | 以选项中的指定的script来处理输入的文本文件； |
| -f<script文件>或--file=<script文件> | 以选项中指定的script文件来处理输入的文本文件； |
| -h或--help | 显示帮助； |
| -n或--quiet或——silent | 仅显示script处理后的结果； |
| -V或--version | 显示版本信息。 |
| -i[扩展名], --in-place[=扩展名] | 直接修改文件(如果指定扩展名就备份文件) |
| -b, --binary | 以二进制方式打开文件 (回车加换行不做特殊处理) |
| -l N, --line-length=N | 指定“l”命令的换行期望长度 |
| --posix | 关闭所有 GNU 扩展 |
| -r, --regexp-extended | 在脚本中使用扩展正则表达式 |
| -s, --separate | 将输入文件视为各个独立的文件而不是一个长的连续输入 |
| -u, --unbuffered | 从输入文件读取最少的数据，更频繁的刷新输出 |
| --help | 打印帮助并退出 |
| --version | 输出版本信息并退出 |

input\_files:要处理的文件列表

命令（command）：

|  |  |
| --- | --- |
| a\ | 在当前行下面插入文本。 |
| i\ | 在当前行上面插入文本。 |
| c\ | 把选定的行改为新的文本。 |
| d | 删除，删除选择的行。 |
| D | 删除模板块的第一行。 |
| s | 替换指定字符 |
| h | 拷贝模板块的内容到内存中的缓冲区。 |
| H | 追加模板块的内容到内存中的缓冲区。 |
| g | 获得内存缓冲区的内容，并替代当前模板块中的文本。 |
| G | 获得内存缓冲区的内容，并追加到当前模板块文本的后面。 |
| l | 列表不能打印字符的清单。 |
| n | 读取下一个输入行，用下一个命令处理新的行而不是用第一个命令。 |
| N | 追加下一个输入行到模板块后面并在二者间嵌入一个新行，改变当前行号码。 |
| p | 打印模板块的行。 |
| P | (大写) 打印模板块的第一行。 |
| q | 退出Sed。 |
| b lable | 分支到脚本中带有标记的地方，如果分支不存在则分支到脚本的末尾。 |
| r file | 从file中读行 |
| t label | if分支，从最后一行开始，条件一旦满足或者T，t命令，将导致分支到带有标号的命令处，或者到脚本的末尾 |
| T label | 错误分支，从最后一行开始，一旦发生错误或者T，t命令，将导致分支到带有标号的命令处，或者到脚本的末尾 |
| w file | 写并追加模板块到file末尾 |
| W file | 写并追加模板块的第一行到file末尾 |
| ! | 表示后面的命令对所有没有被选定的行发生作用 |
| = | 打印当前行号码 |
| # | 把注释扩展到下一个换行符以前 |

替换标记：用于替换操作

|  |  |
| --- | --- |
| g | 表示行内全面替换。 |
| p | 表示打印行。 |
| w | 表示把行写入一个文件。 |
| x | 表示互换模板块中的文本和缓冲区中的文本。 |
| y | 表示把一个字符翻译为另外的字符（但是不用于正则表达式） |
| \1 | 子串匹配标记 |
| & | 已匹配字符串标记 |

正则表达式：

|  |  |
| --- | --- |
| ^ | 匹配行开始，如：/^sed/匹配所有以sed开头的行。 |
| $ | 匹配行结束，如：/sed$/匹配所有以sed结尾的行。 |
| . | 匹配一个非换行符的任意字符，如：/s.d/匹配s后接一个任意字符，最后是d。 |
| \* | 匹配0个或多个字符，如：/\*sed/匹配所有模板是一个或多个空格后紧跟sed的行。 |
| [] | 匹配一个指定范围内的字符，如/[ss]ed/匹配sed和Sed。 |
| [^] | 匹配一个不在指定范围内的字符，如：/[^A-RT-Z]ed/匹配不包含A-R和T-Z的一个字母开头，紧跟ed的行 |
| \(..\) | 匹配子串，保存匹配的字符，如s/\(love\)able/\1rs，loveable被替换成lovers。 |
| & | 保存搜索字符用来替换其他字符，如s/love/\*\*&\*\*/，love这成\*\*love\*\* |
| \< | 匹配单词的开始，如:/\<love/匹配包含以love开头的单词的行。 |
| \> | 匹配单词的结束，如/love\>/匹配包含以love结尾的单词的行。 |
| x\{m\} | 重复字符x，m次，如：/0\{5\}/匹配包含5个0的行。 |
| x\{m,\} | 重复字符x，至少m次，如：/0\{5,\}/匹配至少有5个0的行。 |
| x\{m,n\} | 重复字符x，至少m次，不多于n次，如：/0\{5,10\}/匹配5~10个0的行。 |

举例：

修改替换操作：

|  |
| --- |
| # 查找到包含172.0.0.1的文件，并将172.0.0.1替换为127.0.0.1  # 其中-i表示要修改原文件，s表示替换操作，g用在替换操作中表示行内全面替换  sed –i “s/172.0.0.1/127.0.0.1/g” `grep –rl 172.0.0.1 ./` |