



## 1.2.4 反向选择

## 1.3 格式化输出

## 2. sed 处理一整行

## 2.1 常见用法

## 2.1.1 字符串替换

## 2.2 行的查改增删

## 2.2.1 查看行

## 2.2.2 增加行

```
1 # 文件字符串替换
2 sed -i 's/原字符串/替换字符串/g' filename
3 # 文件连续多行字符串替换
4 sed '2,5s/s1/s2/g' filename
5 # vim 文件字符串替换
6 :%s/s1/s2/g      ? ? ? ?      :%s/s1/s2/g
```

## 2.2 行的查改增删

## 2.2.1 查看行

```
1 # 查看指定行
2 sed -n '2,5p' aa.txt
```

## 2.2.2 增加行

```
1 # 在文档指定n行后增加一行, (hello world)
2 sed 'na hello world' aa.txt
3 # 在文档指定n行前增加一行 ($ 最后一行)
4 sed 'ni hello world' aa.txt
5 # 在以kk 开头的行后, 追加一行
6 sed -i '/kk /hello world' aa.txt
7 抽象出来就是: sed -i '/* /a*' <file>
```

## 2.2.3 删除行

```
1 # 删除文件中的第n行 ($d 删除最后一行)
2 sed -i 'nd' aa.txt
3 # 删除文件中的第2-5行
4 sed -i '2,5d' aa.txt
5
6 # 删除文件中以某个关键字(kk)开头的所有行
7 sed -i '/^kk/d' aa.txt
8 # 删除文件中包含某个关键字(kk)的所有行
9 sed -i '/kk/d' aa.txt
10 cat aa.txt | sed /kk/d
11
12 # 删除#注释行
13 sed -i 's/#.*$/g' aa.txt
14 # 删除空白行
15 sed -i '/^$/d' aa.txt
```

## 2.2.4 行替换

```
1 # 将 2-5行替换成"hello world"
2 sed '2,5c hello world' filename
```

## 2.2.5 行处理

```
1 # "inet addr:192.168.1.100 Bcast:192.168.1.255", 删除左右两边只保留中间ip
2 cat filename | sed 's/^.*addr://g' | sed 's/Bcast.*$/g'
3 sed -e 's/^.*addr://g' -e 's/Bcast.*$/g' filename
4 备注: 使用 -e 连接两个以上的动作
5
6 # 删除行首空白
7 sed 's/^[ \t]*/g' filename
8 备注: 中括号表示"或", 空格或tab中的任意一种; *表示一个或多个。
9 # 删除行尾空白
10 sed 's/[ \t]*$/g' filename
11 # 删除所有空白
12 sed 's/[[:space:]]/g' filename
```

## 一个实例: 去除字符串中的空格



```
中国 北京
美国 华盛顿
英国 伦敦
法国 巴黎
```

```
中国 北京
美国 华盛顿
英国 伦敦
法国 巴黎
```

```
中国 北京
美国 华盛顿
英国 伦敦
法国 巴黎
```

## 3. awk 处理字段

awk '条件类型1 {动作1} 条件类型2 {动作2}...' filename

awk 主要处理每一行字段内的数据, 默认字段分隔符尾空格或tab键。

awk 中变量可以直接使用, 不要加上\$

\$0	代表一整行数据
\$[i=1,2,3,...]	第i 个字段
NF	每一行字段总数
NR	目前所处理的是第几行
FS	目前的分隔符, 默认空格
\$NF	取最后一列: awk -F',' '{print \$NF}'

## 3.1 根据过滤条件输出指定列

指定分隔符为":" , 且第三个字段<10, 格式化输出第1、3字段。

```
awk 'BEGIN {FS=":"} $3<10 {printf $1 "\t" $3 "\n"}' data.txt
```

## 3.2 指定字段求和作为新的列

对p1、p2、p3字段求和, 结果存入Total。

```
1 awk 'NR==1{printf "%10s %10s %10s %10s\n", $1,$2,$3,$4,"Total"};
2 NR>2{total=$2+$3+$4; printf "%10s %10d %10d %10d %10.2f\n", $1,$2,$3,$4,total}' pay.txt
```

```
[java@18-135-35-2 util]$ cat pay.txt
name p1 p2 p3
qq 11 12 13
aa 22 23 24
cc 33 34 35
[java@18-135-35-2 util]$ awk 'NR==1{printf "%10s %10s %10s %10s\n", $1,$2,$3,$4,"Total"};
NR>=2{total=$2+$3+$4; printf "%10s %100 %100 %10.2f\n", $1,$2,$3,$4,total}' pay.txt
name      p1      p2      p3      Total
qq         11       12       13       36.00
aa         22       23       24       69.00
cc         33       34       35      102.00
```

3.3 统计字段次数并排序

统计phone\_list.txt中电话号码出现的次数，并按次数从高到低排序（实现类似group by的功能）

```
1 awk '{arr[$1]+=1}END{for(i in arr) print i,arr[i]}' phone_list.txt | sort -k 2 -r
2 # awk 中的数组类似字典（可以是任意字符或字符串）
3 # 使用sort 实现
4 cat phone_list.txt | sort | uniq -c | sort -k 1 -r
```

```
1 cat phone_list.txt
1761123456
1761123456
1761123666
1761123886
1761123666
1761123666
1761123666
1761123666 3
1761123456 2
1761123456 2
1761123886 1
2 cat phone_list.txt | sort | uniq -c | sort -k 1 -r
3 1761123666
2 1761123456
1 1761123886
```

3.4 统计指定条件的文件数量

统计aa目录下>100k文件的数量和总文件大小

```
1 cd aa
2 ls -l | awk '{if($5>100){count++;sum+=$5}}END{print "count: "count,"\\nsum: "sum}'
3 # awk 会轮询统计，显示整个过程，加入END后只显示最后结果
```

4. 通配符 wildcard

通配符与正则表达式完全不一样，不要搞混！！！！

*	代表0到多个任意字符
?	代表一定有一个任意字符
[]	代表一定有一个在[]内的字符，[abc]可能是a,b,c中任一个
[0-9]	[0-9] 代表0-9之间的所有数字
[^]	反向选择：[^abc] 非a, b, c的其他字符

5. Regular Expression

5.1 正则表达式RE

*	重复前一个0到无穷多次
.	一定有一个任意字符
^word	待查找的字符串（word）在行首
word\$	待查找的字符串（word）在行尾
\	转义符
[list]	[qaz] 一定是q,a,z中的任一个
[n1-n2]	n1,n2之间所有连续的字符
[^list]	反向选择
\{n,m\}	前一个字符重复n到m次；\{n\}重复n次；\{n,\}至少重复n次

5.2 扩展正则表达式 egrep

+	重复前一个RE字符1到无穷多次
?	前一个RE字符出现0或1次
	或
()	找出‘组’字符串
() +	多个重复组的判别

```
1 # 匹配dog,cat和apple
2 egrep 'dog|cat|apple' data.txt
3 # 匹配lad和good
4 egrep 'g(la|oo)d' data.txt
5 # 匹配以A开头以C结尾，中间有一个以上的“xyz”的字符串
6 egrep 'A(xyz)+C' data.txt
```

🔗 文章知识点与官方知识档案匹配，可进一步学习相关知识

CS入门技能树 > Linux入门 > 初识Linux 31995 人正在系统学习中

linux grep与正则表达式使用介绍01-19

grep（缩写来自Globally search a Regular Expression and Print）是一种强大的**文本**搜索工具，它能使用特定模式**匹配**（包括**正则表达...**

4.2\_1 Linux Sed命令详解(文本处理工具 - 匹配，替换，添加等)GroupiesM的博客 · 5459

Linux Sed命令详解(文本处理工具 - 匹配，替换，添加等)，优化投产步骤，使用命令批量修改数据库脚本代替逐个vim

Linux文本匹配命令grep&find&cat(zcat)(持续更新中)\_zcat grep\_普通网友...6-2

1、grep 关键字-截取**文本**搜集工具，结合**正则表达式**非常强大 1.1、基本格式如下: grep [参数] [要**匹配**的内容] [文件名] 1.2、主要...

linux 匹配命令、Linux文本匹配命令grep与fgrep使用完全讲解\_htrbs的...6-5

-L, --files-without-match **匹配**多个文件时,显示不**匹配**的文件名 -l, --files-with-matches **匹配**多个文件时,显示**匹配**的文件名 -c, --count 显示...

Linux——if条件结构 正则表达式 最新发布weixin\_61895606的博客 · 148

if条件结构 正则表达式

正则表达式 | 正则表达式学习篇weixin\_30480583的博客 · 147

概述 **正则表达式**是很多运维薄弱的一项技能。大家很多时候都会觉得**正则表达式**难记、难学、难用，但不可否认的是**正则表达式**是一项...

<b>Linux中使用正则表达式进行文本匹配</b> 陆离23333的博客	5-22
grep--color -E"[0-9](4)"Harry.txt[0-9];表示 <b>匹配</b> 数字0到9中的任意一个(4);表示前面所 <b>匹配</b> 的内容出现4次。此处如此搭配,可用来 <b>匹配</b> 年份...	
<b>linux匹配文本内容sed,Linux高级文本处理之sed(四)_扶摇ggr的博客-CSDN...</b>	5-11
<b>linux匹配文本内容sed,Linux高级文本处理之sed(四)</b> 模式空间是sed内部维护的一个缓存空间,它存放着读入的一行或者多行内容。但是模...	
<b>Linux正则表达式</b>	qq_42193790的博客 960
<b>linux正则</b>	
<b>shell脚本——正则表达式</b>	Canteveni的博客 3058
<b>正则表达式</b> 正则表达式的作用正则当中名词解释第一类 <b>正则表达式</b> 正则中普通常用的元字符案例 <b>正则表达式</b> 的作用 <b>正则表达式</b> 是一种字...	
<b>linux匹配文本内容sed,Linux“文本三剑客”之sed_weixin_39564386的博...</b>	5-26
<b>“文本三剑客”中,grep是文本过滤器,而sed是基于行的文本流编辑器。</b> sed是将文件中的 <b>文本</b> 逐行读取到内存中进行处理。sed工作原理 s...	
<b>linux 处理匹配文本的前后行_sed 匹配行的前n行_ton_tom的博客</b>	6-6
<b>linux 处理匹配文本</b> 的前后行 对于小文件,可以翻转后正序删除 tac file   sed '/tom/!,+2 d'   tac #即删除不区分大小写 <b>匹配</b> 到有tom的行,同时...	
<b>linux中正则匹配</b>	liwenqianye的博客 2778
![img](file:///C:/Users/liet/AppData/Roaming/Tencent/Users/2025838198/QQ/WinTemp/RichOle\$@/WY10QGAAZTCIN`OM@7B.png)s...	
<b>linux从文件中匹配查找内容</b>	shuchongqu的博客 1090
<b>linux从文件中匹配</b> 查找内容	
<b>linux匹配用法,Linux 匹配文字 grep 指令用法教学与範例_weixin_3973202...</b>	6-5
<b>Linux</b> 的 grep 是一个很好用的指令,可以從单流資料或檔案中,使用關鍵字或正規表示法(regular expression)篩選出想要尋找的資料,並且...	
<b>linux 正则表达式</b>	icanflying的博客 1787
<b>正则表达式</b> 的主要应用对象是 <b>文本</b> ,使用 <b>正则表达式</b> 可以指定想要 <b>匹配</b> 的字符串规则,然后通过这个规则来 <b>匹配</b> 、查找、替换或切割那...	
<b>Linux 正则表达式详解</b>	01-19
1.grep : 最早的 <b>文本匹配</b> 程序,使用POSIX定义的基本 <b>正则表达式</b> (BRE) 来 <b>匹配文本</b> 。2.egrep : 扩展式grep, 其使用扩展式正规表达...	
<b>详解Linux命令中的正则表达式</b>	01-10
<b>正则表达式</b> 是一套由多个元字符组成的模糊查找模式,使用 <b>正则表达式</b> 可以快速查找和定位 <b>文本</b> 中指定的内容。 1、单字符 <b>匹配符</b> 。正...	
<b>浅谈Linux grep与正则表达式</b>	12-13
grep 是一种强大的 <b>文本</b> 搜索工具,它可以使用 <b>正则表达式</b> 搜索 <b>文本</b> ,并把 <b>匹配</b> 的行打印出来。通常grep有三种版本grep、egrep (等同于...	
<b>linux grep正则表达式与grep用法详解</b>	01-21
需要大家牢记: <b>正则表达式</b> 与通配符不一样,它们表示的含义并不相同 <b>正则表达式</b> 只是字符串的一种描述,只有和支持 <b>正则表达式</b> 的工具...	
<b>linux 正则表达式grep实例分析</b>	01-19
通过特殊字符+普通字符来进行模式描述,从而达到 <b>文本匹配</b> 目的工具。 <b>正则表达式</b> 目前被集成到了各种 <b>文本编辑器</b> / <b>文本</b> 处理工具当中...	
<b>Python正则表达式匹配和提取IP地址</b>	12-25
下面是IPv4的IP <b>正则匹配</b> 实例: 简单的 <b>匹配</b> 给定的字符串是否是ip地址 import re if re.match("(^[?;0-9](1.3)\.)(3)[0-9](1.3)\$", "236.168.192...	
<b>linux下关于正则表达式grep的一点总结</b>	01-19
通过特殊字符+普通字符来进行模式描述,从而达到 <b>文本匹配</b> 目的工具。类似于生活中常见的寻人启示,通过描述一个人的特征来进行“...	
<b>Linux运维实例 shell脚本中运用正则表达式来精确匹配</b>	weixin_49172531的博客 2791
文章目录前言一、 <b>正则表达式</b> 的命令格式sort命令unip命令tr命令二、使用步骤1.引入库2.读入数据总结 前言 <b>正则表达式</b> 是通过一些特殊...	
<b>linux 文本编辑器和 正则表达式</b>	逆流而上Mr李 241
用 <b>文本编辑器</b> 来编辑生成文件 如下图: 1.mkdir/love 2 cd love 3 echo "I love you liyaozhou" &gt; love.txt创建一个 <b>文本</b> ,并在里面写内...	
<b>Linux：正则表达式</b>	qq_57377057的博客 5159
<b>正则表达式</b> 又称为正规表达式、常规表达式、在代码中常简写为regex、regex或RE。 <b>正则表达式</b> 是使用单个字符串来描述、 <b>匹配</b> 一系列...	
<b>linux正则表达式匹配</b>	03-16
<b>Linux正则表达式匹配</b> 是一种用于在Linux系统中进行 <b>文本</b> 搜索和替换的强大工具。它可以通过一系列的特殊字符和语法规则来 <b>匹配文本</b> 中...	

“相关推荐”对你有帮助么？

非常没帮助 没帮助 一般 有帮助 非常有帮助

关于我们 招贤纳士 商务合作 寻求报道 400-660-0108 kefu@csdn.net 在线客服 工作时间 8:30-22:00

公安备案号11010502030143 京ICP备19004658号 京网文〔2020〕1039-165号 经营性网站备案信息 北京互联网违法和不良信息举报中心 家长监护 网络110报警服务 中国互联网举报中心 Chrome商店下载 账号管理规范 版权与免费声明 版权申诉 出版物许可证 营业执照 ©1999-2023北京创新乐知网络技术有限公司