



*Linux*系统管理

刘雪清



课程大纲

11.1 基础正则表达式

11.2 字符截取命令

11.3 字符处理命令

11.4 条件判断

11.5 流程控制



正则表达式与通配符

- ◆ 正则表达式用来在文件中匹配符合条件的字符串，正则包含**包含匹配**。grep、awk、sed等命令可以支持正则表达式。
- ◆ 通配符用来匹配符合条件的文件名，通配符是**完全匹配**。ls、find、cp。



1、通配符

通配符	作用
*	可以匹配任意数目的字符
?	匹配任意单个字符。
[...]	字符或者字符范围。代表其中包含的任何 ^{一个} 字符。-是连接范围的字符 例如：ls [amf]?? 例如：ls [a-c]*
[!...]	非。例如：ls [!abc]*
{}	{1,3,a} {1..5}



2、基础正则表达式

元字符	作用
*	前一个字符匹配0次或任意多次。
.	匹配除了换行符外任意一个字符。
^	匹配行首。例如：^hello会匹配以hello开头的行。
\$	匹配行尾。例如：hello\$会匹配以hello结尾的行。
[]	匹配中括号中指定的任意一个字符，只匹配一个字符。 例如：[aoeiu] 匹配任意一个元音字母，[0-9] 匹配任意一位数字，[a-z][0-9]匹配小写字和一位数字构成的两位字符。
[^]	匹配除中括号的字符以外的任意一个字符。例如：[^0-9] 匹配任意一位非数字字符，[^a-z] 表示任意一位非小写字母。
\	转义符。用于取消，将特殊符号的含义取消。
\{n\}	表示其前面的字符恰好出现n次。例如：[0-9]\{4\} 匹配4位数字，[1][3-8][0-9]\{9\} 匹配手机号码。
\{n, \}	表示其前面的字符出现不小于n次。例如：[0-9]\{2,\} 表示两位及以上的数字。
\{n, m\}	表示其前面的字符至少出现n次，最多出现m次。例如：[a-z]\{6,8\} 匹配6到8位的小写字母。

'*'前一个字符匹配0次，或任意多次

◆ `grep 'a*' a.txt`

#匹配所有内容，包括空白行

◆ `grep 'aa*' a.txt`

#匹配至少包含有一个a的行

◆ `grep 'aaa*' a.txt`

匹配最少包含两个连续a的字符串

◆ `grep 'aaaa*' a.txt`

#则会匹配最少包含四个个连续a的字符串



❶ '!' 匹配除了换行符外任意一个字符

◆ `grep 's..d' a.txt`

#'s..d' 会匹配在s和d这两个字母之间一定有两个字符的单词

◆ `grep 's.*d' a.txt`

#匹配在s和d字母之间有任意字符

◆ `grep '.*' a.txt`

#匹配所有内容



'^'匹配行首， '\$'匹配行尾

◆ `grep '^D' a.txt`

#匹配以大写'D'开头的行

◆ `grep 'u$' a.txt`

#匹配以小写'u'结尾的行

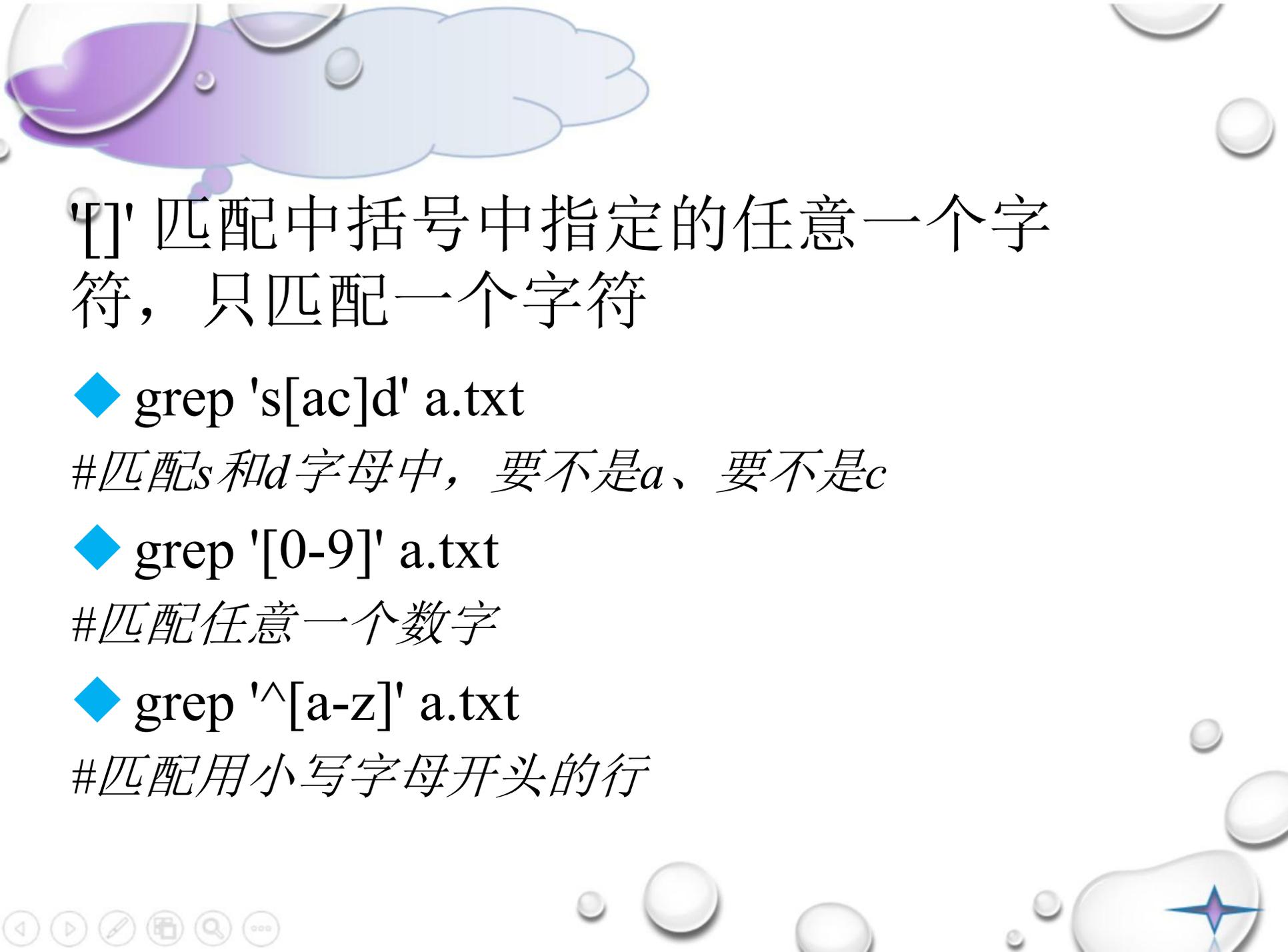
◆ `grep -n '^$' a.txt`

#会匹配空白行

◆ `grep -v '^$' a.txt`

#去除空白行





'[]' 匹配中括号中指定的任意一个字符，只匹配一个字符

◆ `grep 's[ac]d' a.txt`

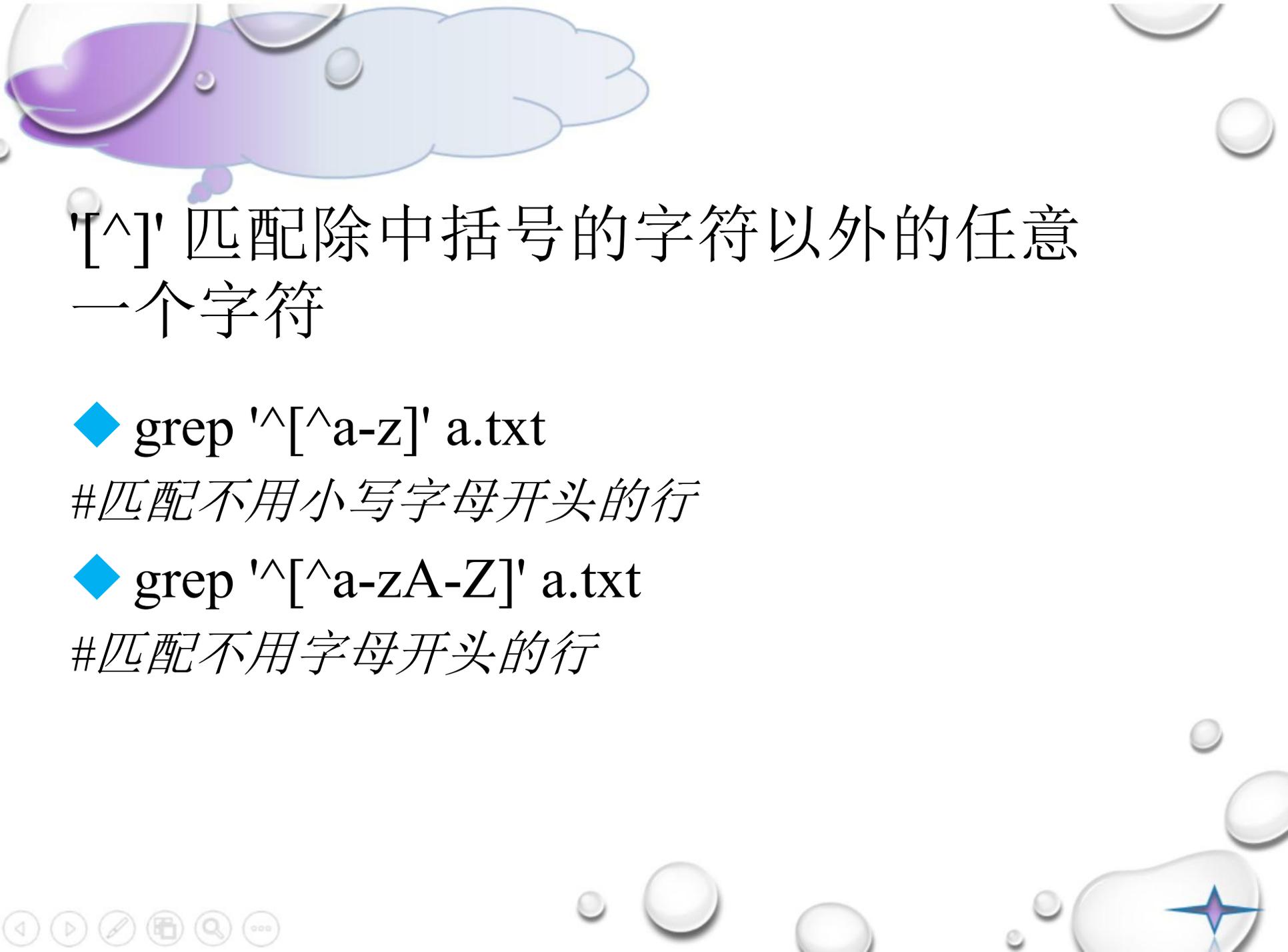
#匹配s和d字母中，要不是a、要不是c

◆ `grep '[0-9]' a.txt`

#匹配任意一个数字

◆ `grep '^[a-z]' a.txt`

#匹配用小写字母开头的行



'[^]' 匹配除中括号的字符以外的任意一个字符

◆ `grep '^[^a-z]' a.txt`

#匹配不用小写字母开头的行

◆ `grep '^[^a-zA-Z]' a.txt`

#匹配不用字母开头的行

'\' 转义符

◆ `grep '\.$' a.txt`

#匹配使用 '.' 结尾的行



`\{n\}`表示其前面的字符恰好出现n次

◆ `grep 'a\{3\}' a.txt`

#匹配a字母连续出现三次的字符串

◆ `grep '[0-9]\{3\}' a.txt`

#匹配包含连续的三个数字的字符串



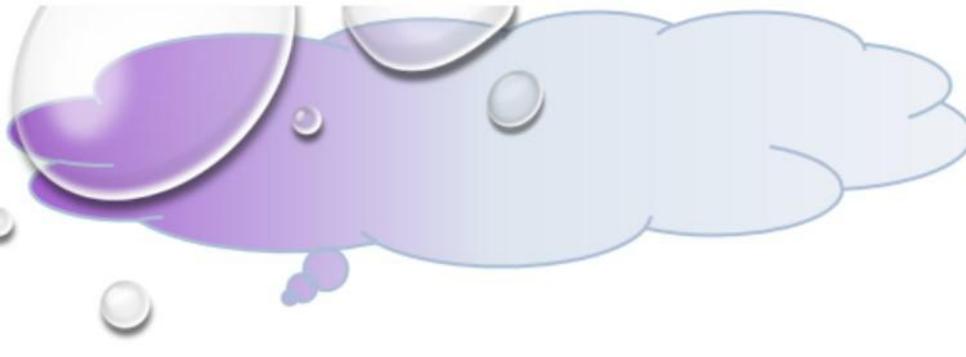


`\{n,\}`表示其前面的字符出现不小于n次

◆ `grep '^[0-9]\{3,\}[a-z]' a.txt`

#匹配最少用连续三个数字开头的行





`\{n,m\}`匹配其前面的字符至少出现n次，最多出现m次

◆ `grep 'sa\{1,3\}i' a.txt`

#匹配在字母s和字母i之间有最少一个a，最多三个a

