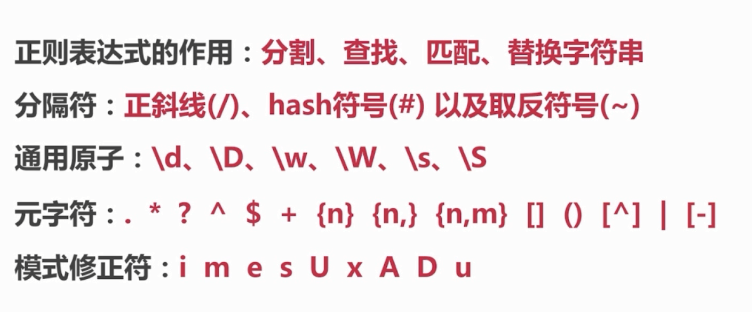
正则：



i 表示在和模式进行匹配进不区分大小写  
m 将模式视为多行，使用^和$表示任何一行都可以以正则表达式开始或结束  
s 如果没有使用这个模式修正符号，元字符中的"."默认不能表示换行符号,将字符串视为单行  
x 表示模式中的空白忽略不计  
e 正则表达式必须使用在preg\_replace替换字符串的函数中时才可以使用(讲这个函数时再说)  
A 以模式字符串开头，相当于元字符^  
Z 以模式字符串结尾，相当于元字符$

U 正则表达式的特点：就是比较“贪婪”，使用该模式修正符可以取消贪婪模式

\d 匹配一个数字字符。等价于 [0-9]。   
• \D 匹配一个非数字字符。等价于 [^0-9]。   
• \f 匹配一个换页符。等价于 \x0c 和 \cL。   
• \n 匹配一个换行符。等价于 \x0a 和 \cJ。   
• \r 匹配一个回车符。等价于 \x0d 和 \cM。   
• \s 匹配任何空白字符，包括空格、制表符、换页符等等。等价于 [ \f\n\r\t\v]。   
• \S 匹配任何非空白字符。等价于 [^ \f\n\r\t\v]。   
• \t 匹配一个制表符。等价于 \x09 和 \cI。   
• \v 匹配一个垂直制表符。等价于 \x0b 和 \cK。   
• \w 匹配包括下划线的任何单词字符。等价于’[A-Za-z0-9\_]’。   
• \W 匹配任何非单词字符。等价于 ‘[^A-Za-z0-9\_]’。

(pattern) 匹配pattern 并获取这一匹配。

(?:pattern) 匹配pattern 但不获取匹配结果，也就是说这是一个非获取匹配，不进行存储供以后使用。

(?=pattern) 正向预查，在任何匹配 pattern 的字符串开始处匹配查找字符串。这是一个非获取匹配，也就是说，该匹配不需要获取供以后使用。

(?!pattern) 负向预查，与(?=pattern)作用相反

