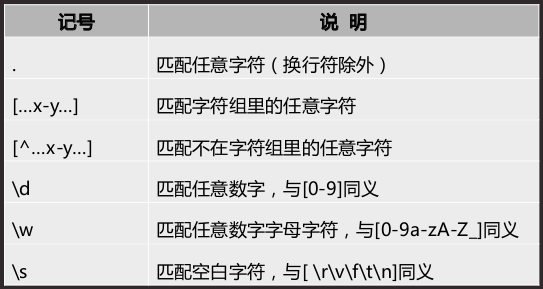
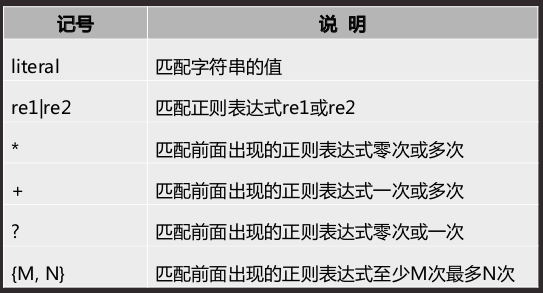
**Python2-4**

**匹配单个字符**

****

**匹配一组字符**

****

**其他元字符**

****

**贪婪匹配**

**• \*、+和?都是贪婪匹配操作符,在其后加上?可以取消其贪婪匹配行为**

**• 正则表达式匹配对象通过groups函数获取子组**

**match函数**

**• 尝试用正则表达式模式从字符串的开头匹配,如果匹配成功,则返回一个匹配对象;否则返回None**

**search函数**

**• 在字符串中查找正则表达式模式的第一次出现,如果匹配成功,则返回一个匹配对象;否则返回None**

**group方法**

**• 使用match或search匹配成功后,返回的匹配对象可以通过group方法获得匹配内容**

**findall函数**

**• 在字符串中查找正则表达式模式的所有(非重复)出现;返回一个匹配对象的列表**

**finditer函数**

**• 和findall()函数有相同的功能,但返回的不是列表而是迭代器;对于每个匹配,该迭代器返回一个匹配对象**

**ompile函数**

**• 对正则表达式模式进行编译,返回一个正则表达式对**

**• 不是必须要用这种方式,但是在大量匹配的情况下,可以提升效率**

**split方法**

**• 根据正则表达式中的分隔符把字符分割为一个列表,并返回成功匹配的列表**

**• 字符串也有类似的方法,但是正则表达式更加灵活**

**sub方法**

**• 把字符串中所有匹配正则表达式的地方替换成新的字符串**

**什么是C/S架构**

**• 服务器是一个软件或硬件,用于提供客户需要的“服务”**

**• 硬件上,客户端常见的就是平时所使用的PC机,服务器常见的有联想、DELL等厂商生产的各种系列服务器**

**• 软件上,服务器提供的服务主要是程序的运行,数据的发送与接收、合并、升级或其它的程序或数据的操作**

**• 套接字是一种具有“通讯端点”概念的计算机网络数据结构**

**• 套接字起源于20世纪70年代加利福尼亚大学伯克利分校版本的Unix**

**• 一种套接字是Unix套接字,其“家族名”为AF\_UNIX**

**• 另一种套接字是基于网络的,“家族名”为AF\_INET**

**• 如果把套接字比做电话的插口,那么主机与端口就像区号与电话号码的一对组合**

**• 无论你使用哪一种地址家族,套接字的类型只有两种。一种是面向连接的套接字,另一种是无连接的套接字**

**• 面向连接的主要协议就是传输控制协议TCP,套接字类型为SOCK\_STREAM**

**• 无连接的主要协议是用户数据报协议UDP,套接字类型为SOCK\_DGRAM**

**• python中使用socket模块中的socket函数实现套接字的创建**

**socket函数与方法**

**创建TCP服务器**

**• 创建TCP服务器的主要步骤如下:**

**1. 创建服务器套接字:s = socket.socket()**

**2. 绑定地址到套接字:s.bind()**

**3. 启动监听:s.listen()**

**4. 接受客户连接:s.accept()**

**5. 与客户端通信:recv()/send()**

**6. 关闭套接字:s.close()**