**字符串**

**字符串方法**

**string.strip()**

按括号中的给定字符进行删除，默认删除字符串首尾的空格和换行符

注意：并不是对字符串对象本身进行修改，而是返回另一个修改后的字符串对象

s = '\n\n\n 123 \n \n '

s.strip() >>>'123'

**string.split()**

按括号内给定的字符分割字符串，返回字符串列表，默认按空格进行分割

**string.strip().split()**

连用两个方法

french= lines.read().strip().split('\n')

**string.join(list)**

join()方法用于把列表list中所有成员放入一个字符串，成员间通过指定的分隔字符串符string进行分隔

**str = string.lower()**

lower()方法将字符串中所有大写字母变为小写

**转义字符**

所有的ASCII码都可以用]"\"加数字(一般是八进制数字)表示

C语言中定义了一些字母前面加"\"来表示常见却不能显示的ASCII字符，如\0，\n，\t等，就称为转义字符，因为后面的字符都不是它原来的ASCII字符的意思了

|  |  |
| --- | --- |
| 转义字符 | 意义 |
| "\ddd" | 1到3位八进制数所代表的任意字符 |
| "\xhh" | 十六进制所代表的任意数量的字符 |

如'\x41'，'\101'都表示字符'A'

['\x41'=='\101','\101'=='A'] >>>[True, True]

**取消转义**

**r'a\nb'**

字符串前加r，是告诉解释器所有字符按原本的样子进行解释

print(r'123\n') >>>123\n

**'123\\n'**

多加一个杠取消转义

print('123\\n') >>>123\n

**chr()函数和ord()函数**

chr()函数和ord()函数是Python的内置函数

**chr(int)**

用一个10进制或16进制整数int作为参数(int实际取值范围很大

)，返回对应的字符

chr(65) >>>'A'

**ord(char)**

用一个字符作为参数，返回该字符对应的ASCII码值

chr(ord('A')+1) >>>'B'

**组合字符**

Unicode中有一类字符称为组合字符，它可以附加在前一个非组合字符上，从而使整体看上去像一个字符。最常见的组合字符为重音符号

详见unicodedata模块

**陌生字符**

有的字符虽然无法显示，但却可以作为整体被识别

s = '123\u202f'

s[3] >>>'\u202f'