

字符串内置方法分类

1. 字符串连接

1.1 join

2. 大小写转换

2.1 lower、upper

2.2 title、capitalize

2.3 swapcase

3. 子串搜索

3.1 count

3.2 endswith和startswith

3.3 find, rfind和index, rindex

4. 分割

4.1 split、rsplit和splitlines

4.2 partition和rpartition

5. 修剪

5.1 strip

5.2 lstrip

5.3 rstrip

6. 替换

6.1 replace

获取更多Python资源
加QQ群：913293727

6.2 expandtabs

6.3 translate和maketrans

获取更多Python资源
加QQ群：913293727

7.填充

7.1 center

7.2 ljust和rjust

7.3 zfill

8.isxxx判断

8.1 isalpha,isdecimal,isdigit,isnumeric,isalnum

8.2 islower,isupper,istitle

8.3 isspace,isprintable,isidentifier

方法及描述

方法	描述
<code>string.capitalize()</code>	把字符串的第一个字符大写
<code>string.center(width)</code>	返回一个原字符串居中,并使用空格填充至长度 width 的新字符串
<code>string.count(str, beg=0, end=len(string))</code>	返回 str 在 string 里面出现的次数，如果 beg 或者 end 指定则返回指定范围内 str 出现的次数
<code>string.decode(encoding='UTF-8', errors='strict')</code>	以 encoding 指定的编码格式解码 string，如果出错默认报一个 ValueError 的异常，除非 errors 指定的是 'ignore' 或者'replace'
<code>string.encode(encoding='UTF-8', errors='strict')</code>	以 encoding 指定的编码格式编码 string，如果出错默认报一个ValueError 的异常，除非 errors 指定的是'ignore'或者'replace'
<code>string.endswith(obj, beg=0, end=len(string))</code>	检查字符串是否以 obj 结束，如果beg 或者 end 指定则检查指定的范围内是否以 obj 结束，如果是，返回 True,否则返回 False.
<code>string.expandtabs(tabsize=8)</code>	把字符串 string 中的 tab 符号转为空格，tab 符号默认的空格数是 8。
<code>string.find(str, beg=0, end=len(string))</code>	检测 str 是否包含在 string 中，如果 beg 和 end 指定范围，则检查是否包含在指定范围内，如果是返回开始的索引值，否则返回-1
<code>string.format()</code>	格式化字符串
<code>string.index(str, beg=0, end=len(string))</code>	跟find()方法一样，只不过如果str不在 string中会报一个异常。
	如果 string 至少有一个字符并且所有字符都是字母或数字则返回 True,否则返回

获取更多Python资源
加QQ群：913293727

string.isalnum()	False
string.isalpha()	如果 string 至少有一个字符并且所有字符都是字母则返回 True,否则返回 False
string.isdecimal()	如果 string 只包含十进制数字则返回 True 否则返回 False.
string.isdigit()	如果 string 只包含数字则返回 True 否则返回 False.
string.islower()	如果 string 中包含至少一个区分大小写的字符，并且所有这些(区分大小写的)字符都是小写，则返回 True，否则返回 False
string.isnumeric()	如果 string 中只包含数字字符，则返回 True，否则返回 False
string.isspace()	如果 string 中只包含空格，则返回 True，否则返回 False.
string.istitle()	如果 string 是标题化的(见 title())则返回 True，否则返回 False
string.isupper()	如果 string 中包含至少一个区分大小写的字符，并且所有这些(区分大小写的)字符都是大写，则返回 True，否则返回 False
string.join(seq)	以 string 作为分隔符，将 seq 中所有的元素(的字符串表示)合并为一个新的字符串
string.ljust(width)	返回一个原字符串左对齐,并使用空格填充至长度 width 的新字符串
string.lower()	转换 string 中所有大写字符为小写.
string.lstrip()	截掉 string 左边的空格
string.maketrans(intab, outtab)	maketrans() 方法用于创建字符映射的转换表，对于接受两个参数的最简单的调用方式，第一个参数是字符串，表示需要转换的字符，第二个参数也是字符串表示转换的目标。
max(str)	返回字符串 str 中最大的字母。
min(str)	返回字符串 str 中最小的字母。
string.partition(str)	有点像 find()和 split()的结合体,从 str 出现的第一个位置起,把字符串 string 分成一个3元素的元组 (string_pre_str,str,string_post_str),如果 string 中不包含str 则 string_pre_str == string.
string.replace(str1, str2, num=string.count(str1))	把 string 中的 str1 替换成 str2,如果 num 指定，则替换不超过 num 次.
string.rfind(str, beg=0,end=len(string))	类似于 find()函数，不过是从右边开始查找.
string.rindex(str, beg=0,end=len(string))	类似于 index(), 不过是从右边开始.
string.rjust(width)	返回一个原字符串右对齐,并使用空格填充至长度 width 的新字符串
string.rpartition(str)	类似于 partition()函数,不过是从右边开始查找
string.rstrip()	删除 string 字符串末尾的空格.
string.split(str="", num=string.count(str))	以 str 为分隔符切片 string，如果 num 有指定值，则仅分隔 num+ 个子字符串
[string.splitlines(keepends)]	按照行('\r', '\r\n', \n')分隔，返回一个包含各行作为元素的列表，如果参数 keepends 为 False，不包含换行符，如果为 True，则保留换行符。
string.startswith(obj,	检查字符串是否是以 obj 开头，是则返回 True，否则返回 False。如果beg 和

<code>beg=0,end=len(string))</code>	end 指定值，则在指定范围内检查.
<code>[string.strip(obj)]</code>	在 string 上执行 lstrip()和 rstrip()
<code>string.swapcase()</code>	翻转 string 中的大小写
<code>string.title()</code>	返回"标题化"的 string,就是说所有单词都是以大写开始，其余字母均为小写(见 istitle())
<code>string.translate(str, del="")</code>	根据 str 给出的表(包含 256 个字符)转换 string 的字符,要过滤掉的字符放到 del 参数中
<code>string.upper()</code>	转换 string 中的小写字母为大写
<code>string.zfill(width)</code>	返回长度为 width 的字符串，原字符串 string 右对齐，前面填充0

获取更多Python资源
加QQ群：913293727