

内置方法

<code>abs(num)</code>	求绝对值
<code>all(iterable)</code>	判断可迭代对象中所有元素是否都为True
<code>any(iterable)</code>	判断可迭代对象中任意元素是否为True
<code>ascii(object)</code>	将对象中的非ascii字符转换为ascii字符
<code>bin(num)</code>	将数字转换为二进制数的字符串
<code>bool([value])</code>	将value转换为bool值
<code>bytearray(source)</code>	将source转换为bytearray类型
<code>bytes(source)</code>	将source转换为byte类型
<code>callable(object)</code>	判断一个对象是否是可调用的
<code>chr(i)</code>	将ascii码转换为字符
<code>classmethod(function)</code>	将一个方法转换为类方法
<code>compile(source, filename, mode)</code>	将source编译为一个可执行的code对象
<code>complex([real[, imag]])</code>	通过传入一个实数和一个虚数创建一个复数
<code>delattr(object, name)</code>	从对象object中移除名为name的属性，不存在时抛出异常
<code>dict()</code>	创建一个字典
<code>dir([object])</code>	返回对象的属性列表
<code>divmod(x, y)</code>	x除以y，返回商和余数
<code>enumerate(iterable, start=0)</code>	将一个可迭代对象组合成为一个索引序列，包含下标和数据
<code>eval(expression)</code>	执行单个字符串表达式，并且返回结果
<code>exec(object, globals, locals)</code>	执行code类型或字符串类型的复杂代码，不返回结果
<code>filter(function, iterable)</code>	过滤可迭代对象，保留function中返回True的对象

内置方法 (cont)

<code>float([x])</code>	将数字类型或字符串类型转换为浮点类型
<code>format(value[, format_spec])</code>	将对象格式化为字符串
<code>frozenset([iterable])</code>	将可迭代对象转换为不可变集合类型
<code>getattr(object, name[, default])</code>	根据名称获取对象的指定属性，如果属性名不存在则返回默认值
<code>globals()</code>	以字典形式返回所有的全局变量
<code>hasattr(object, name)</code>	判断对象是否含有指定名称的属性
<code>hash(object)</code>	获取对象的hash值
<code>help(object)</code>	获得某个对象的帮助文档
<code>hex(x)</code>	将一个整数转换为十六进制字符串
<code>id(object)</code>	返回对象的唯一标识符
<code>input([prompt])</code>	读取用户输入的一行内容，并返回
<code>int(x=0, base=10)</code>	将数字类型或字符串转换为一个整数
<code>isinstance(object, classinfo)</code>	判断一个对象是否是一个类或者其父类的实例
<code>issubclass(object, classinfo)</code>	判断一个类是否是另一个类 (或其父类) 的子类
<code>iter(object[, sentinel])</code>	根据一个对象生成一个迭代器，如果sentinel为True则object必须是可调用的
<code>len(s)</code>	获取一个对象的长度 (s必须是序列或集合类型)
<code>list([iterable])</code>	将一个可迭代对象转换为一个列表



By simpleapples

cheatography.com/simpleapples/

Published 30th May, 2018.

Last updated 30th May, 2018.

Page 1 of 5.

Sponsored by **ApolloPad.com**

Everyone has a novel in them. Finish Yours!

<https://apollopad.com>

内置方法 (cont)

locals()	以字典形式返回所有的局部变量
map(function, iterable, ...)	将function应用到可迭代对象的每一个元素上，并返回执行后的结果列表
max(arg1, arg2, *args[, key])	获取可迭代对象或传入参数中的最大值
memoryview(obj)	将一个对象包装成一个内存查看对象
min(arg1, arg2, *args[, key])	获取可迭代对象或传入参数中的最大值
next(iterator, default)	获取可迭代对象的下一个元素，如果不存在下一个元素返回默认值
object()	返回一个对象
oct(x)	将一个整数转换成八进制字符串
open(file)	打开一个文件，并返回一个文件类型的对象
ord(c)	将一个unicode字符转换成整数类型的ascii码或-unicode数值
pow(x, y[, z])	计算x的y次方，除以z的余数
print(*objects)	将对象输出到标准输出或文件流中
property()	根据输入的getter、getter、deleter生成属性
range(start, stop[, step])	根据开始、结束、步长来返回一个数字类型的不可变序列
repr(obj)	返回一个对象的可打印形式（适合解释器打印的形式）
reversed(seq)	将一个序列翻转并返回一个迭代器
round(number[, ndigits])	根据输入的小数位，将一个浮点型数字四舍五入

内置方法 (cont)

set([iterable])	讲一个可迭代对象转换成一个集合类型
setattr(object, name, value)	根据属性名查找并设置对象的属性
slice(start, stop, step)	根据开始、结束、步长来返回一个切片对象
sorted(iterable[, key][, reverse])	对一个可迭代对象进行排序，并返回一个新的列表
staticmethod(function)	将一个方法转换为静态方法
str(object="")	讲一个对象转为字符串类型（适合人类阅读的形式）
sum(iterable, start)	求可迭代对象中所有元素的和
super(type[, object-or-type])	获取父类
tuple(iterable)	将可迭代对象转换为一个元组
type(object)	返回一个对象的类型
type(name, bases, dict)	创建一个新的type对象
vars(object)	以字典形式返回对象的属性和属性值
zip(*iterables)	将多个可迭代对象中的对应元素打包成一个元组，并返回元组的列表
__import__(name)	动态导入模块

列表处理方法

append(item)	将一个元素添加到列表尾部
extend(list)	将list中的所有元素添加到列表中
insert(index, element)	在列表指定位置插入元素
remove(element)	从左侧查找并移除第一个找到的元素，找不到时-抛出异常



By simpleapples

cheatography.com/simpleapples/

Published 30th May, 2018.

Last updated 30th May, 2018.

Page 2 of 5.

Sponsored by **ApolloPad.com**

Everyone has a novel in them. Finish Yours!

<https://apollopadd.com>

列表处理方法 (cont)

index(element)	从左侧查找并返回第一个找到元素的位置，找不到时抛出异常
count(element)	返回指定元素的个数
pop(index)	移除并返回指定位置元素
reverse()	反转列表
sort(key=..., reverse=...)	对列表进行排序
copy()	浅拷贝列表
clear()	清除列表中所有元素

字典处理方法

clear()	清除所有元素
copy()	浅拷贝
fromkeys(sequence[, value])	以sequence元素为键，value为值创建一个新字典
get(key[, value])	返回字典中key对应的值，如果不存在则返回value
items()	返回字典中的所有键值对（键值对以tuple形式返回）
keys()	返回字典中所有的键
popitem()	移除并返回任意的（不是随机）元素
setdefault(key[, default_value])	如果key不存在，则插入key，值为default_value，返回key对应的值
pop(key[, default])	移除并返回key对应的值，如果key不存在则返回default
values()	返回字典中所有的值
update([other])	使用other更新字典，other可以是一个字典或一个字典组成的可迭代对象

集合处理方法

remove(element)	移除指定元素，元素不存在则抛出异常
add(elem)	添加元素，元素已经存在则什么都不做
copy()	浅拷贝集合
clear()	移除集合中的所有元素
a.difference(b)	返回在集合a中存在，在集合b中不存在的元素的集合
a.difference_update(b)	返回集合a移除了集合a和集合b交集元素的集合
discard(x)	移除指定元素
a.intersection(*other_sets)	返回集合a和其他集合交集的元素集合
a.intersection_update(*other_sets)	返回集合a和其他集合交集与集合a的并集的集合
a.isdisjoint(b)	判断两个集合是否有交集
a.issubset(b)	判断集合a是否是集合b的子集
pop()	移除并返回集合中的任意（不是随机）元素
a.symmetric_difference(b)	返回集合a和集合b各自独有元素的集合
a.symmetric_difference_update(b)	返回集合a和集合b各自独有元素与集合a的并集的集合
a.union(*other_sets)	返回集合a和其他集合的并集
a.update(b)	使用集合b更新集合a

元组处理方法

index(element)	从左侧查找元素并返回元素位置，找不到时抛出异常
count(element)	计算指定元素出现次数



By simpleapples

cheatography.com/simpleapples/

Published 30th May, 2018.

Last updated 30th May, 2018.

Page 3 of 5.

Sponsored by [ApolloPad.com](https://apollopadd.com)

Everyone has a novel in them. Finish Yours!

<https://apollopadd.com>

索引和切片

a[x]	获取第x + 1个元素
a[-x]	获取从结尾开始第x个元素
a[x:]	获取第x + 1到最后一个元素
a[:x]	获取第一个元素到第x + 1个元素
a[:-x]	获取第一个元素到从结尾开始的第x个元素
a[x:y]	获取第x + 1个元素到第y + 1个元素
a[:]	浅拷贝a

字符串处理方法

capitalize()	将首字符转换为大写
center(width[, fillchar])	居中并填充字符到指定宽度
casefold()	讲字符串转换成小写
count(substring[, start[, end]])	计算子字符串出现的次数
endswith(suffix[, start[, end]])	判断是否以指定的字符串结尾
expandtabs(tabsize)	将字符串中的\t转换为指定数量的空格
encode(encoding='UTF-8', errors='strict')	将unicode字符串转换为任何Python支持的编码类型
find(sub[, start[, end]])	从左侧查找指定字符串位置 (不存在返回-1)
format(p0, p1, ..., k0=v0, k1=v1, ...)	格式化字符串
index(sub[, start[, end]])	从左侧查找指定字符串位置 (不存在抛出异常)
isalnum()	检查字符串是否仅由字母和数字组成
isalpha()	检查字符串是否仅由字母组成
isdecimal()	检查字符串是否只由十进制字符组成
isdigit()	检测字符串是否只由数字组成
isidentifier()	判断字符串是否是有效的标识符

字符串处理方法 (cont)

islower()	检测字符串是否由小写字母组成
isnumeric()	检测字符串是否只由数字组成 (只支持unicode字符串)
isprintable()	检测字符串是否可打印 (空字符串也可以打印)
isspace()	检测字符串是否只由空白字符组成 (包括空格、换行符、制表符等)
istitle()	判断字符串首字母是否大写
isupper()	判断字符串是否只由大写字母组成
join(iterable)	用指定字符连接字符串可迭代对象
ljust(width[, fillchar])	居左返回指定长度的字符串, 不足部分用fillchar填充
rjust(width[, fillchar])	居右返回指定长度的字符串, 不足部分用fillchar填充
lower()	将所有字符转换为小写
upper()	将所有字符转换为大写
swapcase()	将大写字母转换为小写, 小写字母转换为大写
lstrip([chars])	从左侧截取指定的字符 (可以是多个)
rstrip([chars])	从右侧截取指定的字符 (可以是多个)
strip([chars])	从两侧截取指定的字符 (可以是多个)



By simpleapples

Published 30th May, 2018.
Last updated 30th May, 2018.
Page 4 of 5.

Sponsored by **ApolloPad.com**
Everyone has a novel in them. Finish Yours!
<https://apollopad.com>

cheatography.com/simpleapples/

字符串处理方法 (cont)

<code>partition(separator)</code>	从左侧开始查找separator，找到后将字符串分割为-separator左边部分、separator、separator右边部分
<code>maketrans(x[, y[, z]])</code>	创建将指定字符映射为映射的字符的映射表
<code>rpartition(separator)</code>	从右侧开始查找separator，找到后将字符串分割为-separator左边部分、separator、separator右边部分
<code>translate(table)</code>	将指定字符映射为映射的字符
<code>replace(old, new [, count])</code>	将字符串中的old字符串中替换成new字符串，最多替换count次
<code>rfind(sub[, start[, end]])</code>	从右侧查找指定字符串位置（不存在返回-1）
<code>rindex(sub[, start[, end]])</code>	从右侧查找指定字符串位置（不存在抛出异常）
<code>split([separator [, maxsplit]])</code>	从左侧以separator为分隔符切片，最多maxsplit次
<code>rsplit([separator [, maxsplit]])</code>	从右侧以separator为分隔符切片，最多maxsplit次
<code>splitlines([keepends])</code>	按行切分字符串，如果keepends=True则显示换行符
<code>title()</code>	将字符串转换为标题形式（所有单词首字母大写）
<code>zfill(width)</code>	将字符串左侧填充0到指定长度
<code>format_map(mapping)</code>	使用字典格式化字符串



By **simpleapples**

cheatography.com/simpleapples/

Published 30th May, 2018.

Last updated 30th May, 2018.

Page 5 of 5.

Sponsored by **ApolloPad.com**

Everyone has a novel in them. Finish Yours!

<https://apollopad.com>