

# Python数据类型常用方法参考手册

## 一、内建函数

函数	描述
<code>abs()</code>	返回数的绝对值
<code>all()</code>	如果可迭代对象中的所有项均为 true，则返回 True。
<code>any()</code>	如果可迭代对象中的任何项为 true，则返回 True。
<code>ascii()</code>	返回对象的可读版本。用转义字符替换 non-ascii 字符。
<code>bin()</code>	返回数的二进制版本。
<code>bool()</code>	返回指定对象的布尔值。
<code>bytearray()</code>	返回字节数组。
<code>bytes()</code>	返回字节对象。
<code>callable()</code>	如果指定的对象是可调用的，则返回 True，否则返回 False。
<code>chr()</code>	返回指定 Unicode 代码中的字符。
<code>classmethod()</code>	把方法转换为类方法。
<code>compile()</code>	把指定的源作为对象返回，准备执行。
<code>complex()</code>	返回复数。
<code>delattr()</code>	从指定的对象中删除指定的属性（属性或方法）。
<code>dict()</code>	返回字典（数组）。
<code>dir()</code>	返回指定对象的属性和方法的列表。
<code>divmod()</code>	当参数1除以参数2时，返回商和余数。

函数	描述
<code>enumerate()</code>	获取集合（例如元组）并将其作为枚举对象返回。
<code>eval()</code>	评估并执行表达式。
<code>exec()</code>	执行指定的代码（或对象）。
<code>filter()</code>	使用过滤器函数排除可迭代对象中的项目。
<code>float()</code>	返回浮点数。
<code>format()</code>	格式化指定值。
<code>frozenset()</code>	返回 frozenset 对象。
<code>getattr()</code>	返回指定属性的值（属性或方法）。
<code>globals()</code>	以字典返回当前全局符号表。
<code>hasattr()</code>	如果指定的对象拥有指定的属性（属性/方法），则返回 True。
<code>hash()</code>	返回指定对象的哈希值。
<code>help()</code>	执行内建的帮助系统。
<code>hex()</code>	把数字转换为十六进制值。
<code>id()</code>	返回对象的 id。
<code>input()</code>	允许用户输入。
<code>int()</code>	返回整数。
<code>isinstance()</code>	如果指定的对象是指定对象的实例，则返回 True。
<code>issubclass()</code>	如果指定的类是指定对象的子类，则返回 True。
<code>iter()</code>	返回迭代器对象。

函数	描述
<code>len()</code>	返回对象的长度。
<code>list()</code>	返回列表。
<code>locals()</code>	返回当前本地符号表的更新字典。
<code>map()</code>	返回指定的迭代器，其中指定的函数应用于每个项目。
<code>max()</code>	返回可迭代对象中的最大项目。
<code>memoryview()</code>	返回内存视图（memory view）对象。
<code>min()</code>	返回可迭代对象中的最小项目。
<code>next()</code>	返回可迭代对象中的下一项。
<code>object()</code>	返回新对象。
<code>oct()</code>	把数转换为八进制。
<code>open()</code>	打开文件并返回文件对象。
<code>ord()</code>	转换表示指定字符的 Unicode 的整数。
<code>pow()</code>	返回 x 的 y 次幂的值。
<code>print()</code>	打印标准输出设备。
<code>property()</code>	获取、设置、删除属性。
<code>range()</code>	返回数字序列，从 0 开始且以 1 为增量（默认地）。
<code>repr()</code>	返回对象的可读版本。
<code>reversed()</code>	返回反转的迭代器。
<code>round()</code>	对数进行舍入。
<code>set()</code>	返回新的集合对象。

函数	描述
<code>setattr()</code>	设置对象的属性（属性/方法）。
<code>slice()</code>	返回 slice 对象。
<code>sorted()</code>	返回排序列表。
<code>@staticmethod()</code>	把方法转换为静态方法。
<code>str()</code>	返回字符串对象。
<code>sum()</code>	对迭代器的项目进行求和。
<code>super()</code>	返回表示父类的对象。
<code>tuple()</code>	返回元组。
<code>type()</code>	返回对象的类型。
<code>vars()</code>	返回对象的 <code>dict</code> 属性。
<code>zip()</code>	从两个或多个迭代器返回一个迭代器。

## 二、字符串内置方法

方法	描述
<code>capitalize()</code>	把首字符转换为大写。
<code>casefold()</code>	把字符串转换为小写。
<code>center()</code>	返回居中的字符串。
<code>count()</code>	返回指定值在字符串中出现的次数。
<code>encode()</code>	返回字符串的编码版本。
<code>endswith()</code>	如果字符串以指定值结尾，则返回 <code>true</code> 。
<code>expandtabs()</code>	设置字符串的 <code>tab</code> 尺寸。
<code>find()</code>	在字符串中搜索指定的值并返回它被找到的位置。
<code>format()</code>	格式化字符串中的指定值。
<code>format_map()</code>	格式化字符串中的指定值。
<code>index()</code>	在字符串中搜索指定的值并返回它被找到的位置。
<code>isalnum()</code>	如果字符串中的所有字符都是字母数字，则返回 <code>True</code> 。
<code>isalpha()</code>	如果字符串中的所有字符都在字母表中，则返回 <code>True</code> 。
<code>isdecimal()</code>	如果字符串中的所有字符都是小数，则返回 <code>True</code> 。
<code>isdigit()</code>	如果字符串中的所有字符都是数字，则返回 <code>True</code> 。
<code>isidentifier()</code>	如果字符串是标识符，则返回 <code>True</code> 。
<code>islower()</code>	如果字符串中的所有字符都是小写，则返回 <code>True</code> 。

方法	描述
<code>isnumeric()</code>	如果字符串中的所有字符都是数，则返回 True。
<code>isprintable()</code>	如果字符串中的所有字符都是可打印的，则返回 True。
<code>isspace()</code>	如果字符串中的所有字符都是空白字符，则返回 True。
<code>istitle()</code>	如果字符串遵循标题规则，则返回 True。
<code>isupper()</code>	如果字符串中的所有字符都是大写，则返回 True。
<code>join()</code>	把可迭代对象的元素连接到字符串的末尾。
<code>ljust()</code>	返回字符串的左对齐版本。
<code>lower()</code>	把字符串转换为小写。
<code>lstrip()</code>	返回字符串的左修剪版本。
<code>maketrans()</code>	返回在转换中使用的转换表。
<code>partition()</code>	返回元组，其中的字符串被分为三部分。
<code>replace()</code>	返回字符串，其中指定的值被替换为指定的值。
<code>rfind()</code>	在字符串中搜索指定的值，并返回它被找到的最后位置。
<code>rindex()</code>	在字符串中搜索指定的值，并返回它被找到的最后位置。
<code>rjust()</code>	返回字符串的右对齐版本。
<code>rpartition()</code>	返回元组，其中字符串分为三部分。
<code>rsplit()</code>	在指定的分隔符处拆分字符串，并返回列表。

方法	描述
<code>rstrip()</code>	返回字符串的右边修剪版本。
<code>split()</code>	在指定的分隔符处拆分字符串，并返回列表。
<code>splitlines()</code>	在换行符处拆分字符串并返回列表。
<code>startswith()</code>	如果以指定值开头的字符串，则返回 true。
<code>strip()</code>	返回字符串的剪裁版本。
<code>swapcase()</code>	切换大小写，小写成为大写，反之亦然。
<code>title()</code>	把每个单词的首字符转换为大写。
<code>translate()</code>	返回被转换的字符串。
<code>upper()</code>	把字符串转换为大写。
<code>zfill()</code>	在字符串的开头填充指定数量的 0 值。

**注释：**所有字符串方法都返回新值。它们不会更改原始字符串。

## 三、列表内置方法



方法	描述
<code>append()</code>	在列表的末尾添加一个元素
<code>clear()</code>	删除列表中的所有元素
<code>copy()</code>	返回列表的副本
<code>count()</code>	返回具有指定值的元素数量。
<code>extend()</code>	将列表元素（或任何可迭代的元素）添加到当前列表的末尾
<code>index()</code>	返回具有指定值的第一个元素的索引
<code>insert()</code>	在指定位置添加元素
<code>pop()</code>	删除指定位置的元素
<code>remove()</code>	删除具有指定值的项目
<code>reverse()</code>	颠倒列表的顺序
<code>sort()</code>	对列表进行排序

**注释：** Python 没有内置对数组的支持，但可以使用 Python 列表。

## 四、字典内置方法

方法	描述
<code>clear()</code>	删除字典中的所有元素
<code>copy()</code>	返回字典的副本
<code>fromkeys()</code>	返回拥有指定键和值的字典
<code>get()</code>	返回指定键的值
<code>items()</code>	返回包含每个键值对的元组的列表
<code>keys()</code>	返回包含字典键的列表
<code>pop()</code>	删除拥有指定键的元素
<code>popitem()</code>	删除最后插入的键值对
<code>setdefault()</code>	返回指定键的值。如果该键不存在，则插入具有指定值的键。
<code>update()</code>	使用指定的键值对字典进行更新
<code>values()</code>	返回字典中所有值的列表

## 五、元组内置方法

方法	描述
<code>count()</code>	返回元组中指定值出现的次数。
<code>index()</code>	在元组中搜索指定的值并返回它被找到的位置。

## 六、集合内置方法

方法	描述
<code>add()</code>	向集合添加元素。
<code>clear()</code>	删除集合中的所有元素。
<code>copy()</code>	返回集合的副本。
<code>difference()</code>	返回包含两个或更多集合之间差异的集合。
<code>difference_update()</code>	删除此集合中也包含在另一个指定集合中的项目。
<code>discard()</code>	删除指定项目。
<code>intersection()</code>	返回为两个其他集合的交集的集合。
<code>intersection_update()</code>	删除此集合中不存在于其他指定集合中的项目。
<code>isdisjoint()</code>	返回两个集合是否有交集。
<code>issubset()</code>	返回另一个集合是否包含此集合。
<code>issuperset()</code>	返回此集合是否包含另一个集合。
<code>pop()</code>	从集合中删除一个元素。
<code>remove()</code>	删除指定元素。
<code>symmetric_difference()</code>	返回具有两组集合的对称差集的集合。
<code>symmetric_difference_update()</code>	插入此集合和另一个集合的对称差集。
<code>union()</code>	返回包含集合并集的集合。

方法	描述
<code>update()</code>	用此集合和其他集合的并集来更新集合。

## 七、文件操作内置方法

方法	描述
<code>close()</code>	关闭文件。
<code>detach()</code>	从缓冲区返回分离的原始流（raw stream）。
<code>fileno()</code>	从操作系统的角度返回表示流的数字。
<code>flush()</code>	刷新内部缓冲区。
<code>isatty()</code>	返回文件流是否是交互式的。
<code>read()</code>	返回文件内容。
<code>readable()</code>	返回是否能够读取文件流。
<code>readline()</code>	返回文件中的一行。
<code>readlines()</code>	返回文件中的行列表。
<code>seek()</code>	更改文件位置。
<code>seekable()</code>	返回文件是否允许我们更改文件位置。
<code>tell()</code>	返回当前的文件位置。
<code>truncate()</code>	把文件调整为指定的大小。
<code>writable()</code>	返回是否能够写入文件。
<code>write()</code>	把指定的字符串写入文件。
<code>writelines()</code>	把字符串列表写入文件。

## 八、Python关键字

关键字	描述
and	逻辑运算符。
as	创建别名。
assert	用于调试。
break	跳出循环。
class	定义类。
continue	继续循环的下一个迭代。
def	定义函数。
del	删除对象。
elif	在条件语句中使用，等同于 else if。
else	用于条件语句。
except	处理异常，发生异常时如何执行。
False	布尔值，比较运算的结果。
finally	处理异常，无论是否存在异常，都将执行一段代码。
for	创建 for 循环。
from	导入模块的特定部分。
global	声明全局变量。
if	写一个条件语句。
import	导入模块。
in	检查列表、元组等集合中是否存在某个值。
is	测试两个变量是否相等。
lambda	创建匿名函数。

关键字	描述
None	表示 null 值。
nonlocal	声明非局部变量。
not	逻辑运算符。
or	逻辑运算符。
pass	null 语句，一条什么都不做的语句。
raise	产生异常。
return	退出函数并返回值。
True	布尔值，比较运算的结果。
try	编写 try...except 语句。
while	创建 while 循环。
with	用于简化异常处理。
yield	结束函数，返回生成器。