**数学：**

|  |  |  |  |
| --- | --- | --- | --- |
| 函数 | 含义 | 参数 | 实例 |
| abs(num) | 绝对值 | - | - |
| bin(num) | 返回int类型的二进制，参数为负数，结果也有负号。且结果以0b开头。 | int或实现了\_\_index\_\_()的python对象 |  |
| bool(num) | 将数字转换为bool类型 | int/float：0=False，1=True  对象：为空=False，不为空=True |  |
| max(a,b,c...) | 取最大值 | 可以是多个数字、列表、字符串（找最大字符）、元组（按顺序比较）；不仅可以比较数字，还可以比较按字典顺序比较字符串。  另外可以配置key，比较元素key()的值。 |  |
| min(a,b,c...) | 取最小值 | 同上 |  |
| cmp(x,y) | 比较大小，相同返回0，x > y返回1，x<y返回-1 | 数值表达式 |  |
| complex(real[, imag]) | 返回一个复数，也可以将一个字符串转化为一个复数 | real：实部int/long/float/str  imag：虚部int/long/float |  |
| divmod(a, b) | 返回商和余数 | 两个数字int long float，float结果不准 |  |
|  |  |  |  |

**字符串：**

|  |  |  |  |
| --- | --- | --- | --- |
| 函数 | 含义 | 参数 | 实例 |
| chr(num) | 返回对应的字符，如果是[0-255]则为ascii码表 | int，支持十进制、十六进制 |  |
| ord(c) | chr的逆运算，返回字符的编码 | 字符串 |  |
| str(obj) | toString()，调用对象的\_\_str\_\_方法 | 任何对象 |  |

**集合&字典：**

|  |  |  |  |
| --- | --- | --- | --- |
| 函数 | 含义 | 参数 | 实例 |
| all(iterable) | 判断集合中是否所有元素都为true | Iterable |  |
| any(iterable) | 判断集合中是否包含一个为true的元素 | Iterable |  |
| dict(\*args) | 构建一个字典 | 多个关键字key:value  迭代对象[(k, v),(k, v)]  映射函数zip() |  |
| zip(object) | 将多个可迭代的对象对应的元素打包成元组，并返回由这些元组组成的列表。如果可迭代元素长度不同，返回最短。  可以理解为将可迭代元素压缩为元组，同时可以将压缩对象解压为原来的对象。在python3中压缩和解压的对象可以使用list(), dict()转化为列表和字典。解压缩时同理。 | 多个可迭代变量。字符串、列表、元组、字典 |  |
| enumerate(iterable) | 将可迭代元素 |  |  |
| sum(可迭代对象) | 可迭代对象求和 | 列表、元组、集合 | 跟list、set或列表切片组合使用，代码精简 |

**对象：**

|  |  |  |  |
| --- | --- | --- | --- |
| 函数 | 含义 | 参数 | 实例 |
| id(obj) | 获取对象内存地址 | object |  |
| delattr(obj, attr) | 删除对象属性 | object, 属性名 |  |
| getattr(obj, attr[, default]) | 获取对象属性，可设置默认值 | object, 属性名，默认值 |  |
| hasattr(obj, attr) | 对象是否包含属性 | object, 属性名 |  |
| setattr(obj, attr, value) | 设置对象属性值 | object, 属性名，属性值 |  |
| callable(obj) | 判断对象是否是可调用对象，即是否实现了\_\_call\_\_方法 | object |  |

其他：

|  |  |  |  |
| --- | --- | --- | --- |
| 函数 | 含义 | 参数 | 实例 |
| len(x) | 返回输入参数长度 | 字符串、元组、列表、range、字典、集合 |  |
| range(start, end[, step]) | 创建整数列表，开始，结束（不包含），步长 |  |  |
|  |  |  |  |

ascii()

breakpoint()

bytearray()

compile()