

运算符原理：

python运算符实际上是通过调用特殊方法实现

常见特殊方法：

方法	作用
<code>__init__</code>	对象创建
<code>__del__</code>	对象回收
<code>__repr__</code> , <code>__str__</code>	转换，打印
<code>__call__</code>	调用
<code>__getattr__</code>	点号运算
<code>__setattr__</code>	属性赋值
<code>__getitem__</code>	索引运算
<code>__setitem__</code>	索引赋值
<code>__len__</code>	长度

常见运算符的方法

可以重写这些方法实现运算符重载

运算符	特殊方法
<code>+</code>	<code>__add__</code>
<code>-</code>	<code>__sub__</code>
<code>&lt;</code> , <code>&lt;=</code> , <code>==</code>	<code>__lt__</code> , <code>__le__</code> , <code>__eq__</code>
<code>&gt;</code> , <code>&gt;=</code> , <code>!=</code>	<code>__gt__</code> , <code>__ge__</code> , <code>__ne__</code>
<code> </code> , <code>^</code> , <code>&amp;</code>	<code>__or__</code> , <code>__xor__</code> , <code>__and__</code>
<code>&lt;&lt;</code> , <code>&gt;&gt;</code>	<code>__lshift__</code> , <code>__rshift__</code>
<code>*</code> / <code>%</code> / <code>//</code>	<code>__mul__</code> , <code>__truediv__</code> , <code>__mod__</code> , <code>__floordiv__</code>
<code>**</code>	<code>__pow__</code>

【附加】类型字符串提取

```
Sentence = 1  
type_str = str(type(Sentence))  
type_str = type_str.replace("<", "")  
type_str = type_str.replace(">", "")  
type_str = type_str.split(" ")  
type_str = type_str[-1]  
print(type_str)
```