链滴

python:访问限制

作者：jaz

原文链接：https://ld246.com/article/1456984851452

来源网站：链滴

在Class内部，可以有属性和方法，而外部代码可以通过直接调用实例变量的方法来操作数据，样，就隐藏了内部的复杂逻辑。
但是，从Student类的定义来看，外部代码还是可以自由地修改一个实例的<code>name</cod>、<code>score</code>属性：
<pre><code class="python">>>> bart = Student(<pan class="string">'Bart Simpson', 98)
>>> bart.score
98
>>> bart.score = 59</spa>
>>> bart.score
59
</code></pre>
如果要让内部属性不被外部访问，可以把属性的名称前加上两个下划线<code>__</code>，在Python中，实例的变量名如果以<code>__</code>开头，就变成了一个私有变量（private），只有部可以访问，外部不能访问，所以，我们把Student类改一改：
<pre><code class="python">class <san class="title">Student(object):

def __init__</san>(self, name, score):
    self.__name = name
    self.__score = score

def print_scor(self):
    print('%s: %s' % (self.__name, self.__score))

</code></pre>

改完后，对于外部代码来说，没什么变动，但是已经无法从外部访问<code>实例变量.__name<code>和<code>实例变量.__score</code>了：
<pre><code class="xml">>>> bart = Student('Bart Simpson', 98)
>>> bart.__name
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
AttributeError: 'Student' object has no attribute '__name'
</code></pre>
这样就确保了外部代码不能随意修改对象内部的状态，这样通过访问限制的保护，代码更加健壮
但是如果外部代码要获取name和score怎么办？可以给Student类增加<code>get_name</cod>和<code>get_score</code>这样的方法：
<pre><code class="python">class <san class="title">Student(object):
    ...

def get_name(self):
    return self.__name

def get_score(self):
    return self.__score

</code></pre>

如果又要允许外部代码修改score怎么办？可以再给Student类增加<code>set_score</code>法：
<pre><code class="python">class <san class="title">Student(object):
    ...

def set_score(self, score):
    self.__score = score

</code></pre>

你也许会问，原先那种直接通过<code>bart.score = 59</code>也可以修改啊，为什么要定一个方法大费周折？因为在方法中，可以对参数做检查，避免传入无效的参数：
<pre><code class="python">class <san class="title">Student(object):
    ...

def set_score(self, score):
    if 0 <= score <= <san class="number">100:
        self.__score = score
    else:
        raise ValueError('bad score'</spn>)

</code></pre>

需要注意的是，在Python中，变量名类似<code>__xxx__</code>的，也就是以双下划线开头并且以双下划线结尾的，是特殊变量，特殊变量是可以直接访问的，不是private变量，所以，不能用code>__name__</code>、<code>__score__</code>这样的变量名。
有些时候，你会看到以一个下划线开头的实例变量名，比如<code>_name</code>，这样的实变量外部是可以访问的，但是，按照约定俗成的规定，当你看到这样的变量时，意思就是，“然我可以被访问，但是，请把我视为私有变量，不要随意访问”。
双下划线开头的实例变量是不是一定不能从外部访问呢？其实也不是。不能直接访问<code>__nme</code>是因为Python解释器对外把<code>__name</code>变量改成了<code>_Student__nme</code>，所以，仍然可以通过<code>_Student__name</code>来访问<code>__name</coe>变量：
<pre><code class="python">>>> bart._Student__nme
'Bart Simpson'
</code></pre>
但是强烈建议你不要这么干，因为不同版本的Python解释器可能会把<code>__name</code>成不同的变量名。
总的来说就是，Python本身没有任何机制阻止你干坏事，一切全靠自觉。