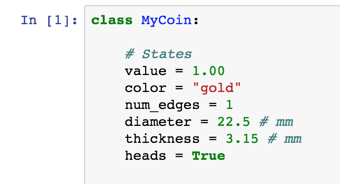
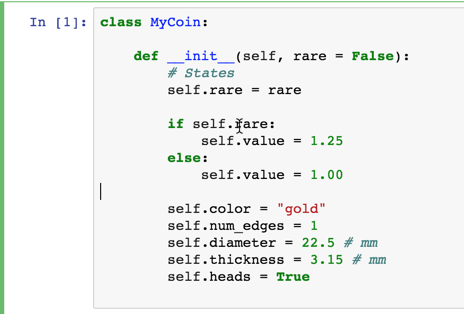
继续我们的学习。

在这一课的内容中，我们将重点学习类方法，并完成MyCoin项目。

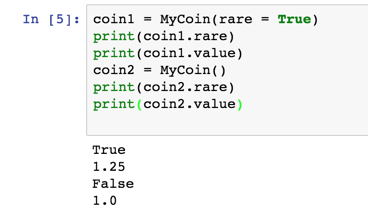
首先打开Jupyter Notebook，并打开MyCoin文件。

到目前为止，我们的类定义中只有简单的States，而没有方法：

首先我们要添加一个类的初始化方法(constructor)：

在以上代码中，我们使用def \_\_init\_\_定义了一个类的初始化方法，并添加了两个参数，一个是self,指代类的对象，另一个是rare。

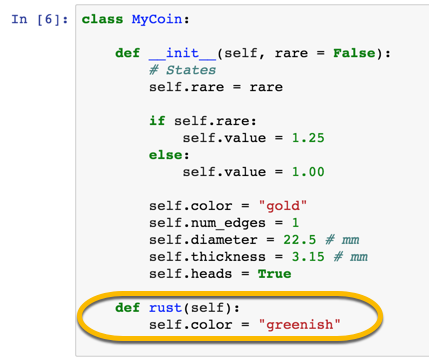
接下来我们把所有的states都放到这个初始化方法里面。

好了，实际运行一下看看效果：

在以上代码中，我们分别创建了coin1和coin2两个对象。其中coin1指定了rare的初始值为True，而coin2则采用了默认的值。

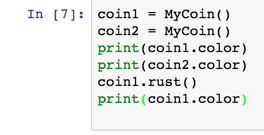
需要注意的是，类的初始化方法中是不存在return返回值的。

接下来添加一个新的类方法，让硬币可以生锈变色。



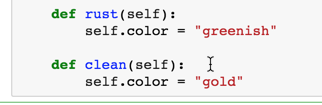
在以上的代码中，我们添加了一个新的类方法rust，它的作用就是让硬币生锈，从而变换颜色。

好了，实际来看看怎么使用这个方法：



在以上代码中，coin1.rust()这行代码就调用了rust这个方法，从而让coin1对象的硬币生锈。

接着我们再添加另一个新的类方法clean，让生锈的硬币可以恢复原样。和rust方法类似，我们在类的定义中使用如下方式添加一个clean方法。



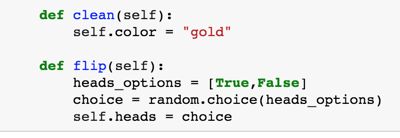
接着来实验一下：

在以上代码中，我们创建了coin1对象，并输出其颜色。然后调用rust方法，再次输出其颜色。最后调用clean方法，并输出其颜色。可以看到，硬币的色彩先是初始的金色，然后变成生锈的绿色，最后又变回金色~

接下来我们再添加一个有意思的方法，可以让硬币随机翻转：

首先在类的定义之前添加一行代码：

import random

这样我们就可以在后面的代码中调用随机数生成器了~

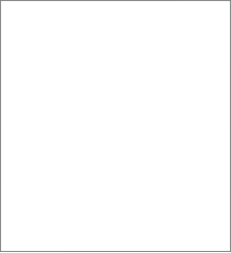
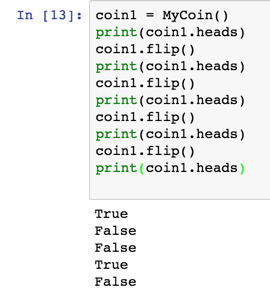
在以上代码中，我们创建了一个新的flip方法（紧接着clean方法之后）。

在这个方法中，我们定义了一个heads\_options列表，里面只包含了两个元素，分别是True,False。

然后定义了一个choice变量，让它从heads\_options列表中随机获取一个元素。

最后将choice的值赋予硬币对象的heads属性。

好了，让我们来试着运行下看看：



在以上代码中，我们创建了coin1这个对象，然后输出它的朝向。接着调用flip方法，每次调用之后都会输出它的朝向。

可以看到，硬币的朝向会是随机的，正好就是我们所希望的~

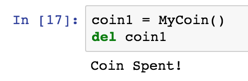
最后我们希望添加一个spend方法，可以让玩家把这个硬币花掉，实际上就是销毁这枚硬币。所以我们需要一个destructor方法，python会在程序结束的时候自动销毁这个对象。



在以上代码中我们添加了一个desctructor方法，注意需要把它放在类定义的代码之中。

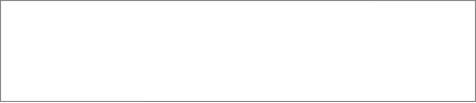
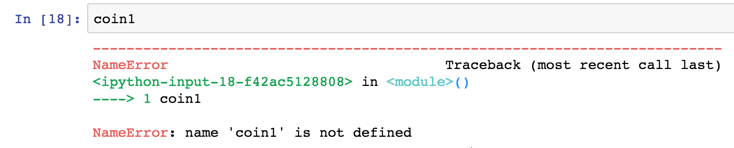
这里我们只是让程序输出一行文字，告诉玩家硬币已经被花掉了。

好了，来实际运行看看：



在以上代码中，我们首先创建了一个硬币对象，然后使用del coin1这行代码，就可以销毁对象。

此时如果再次查看coin1的状态，就会报错，因为对象已经销毁了~



到此为止，我们已经使用类和对象的概念开发了一个简单的MyCoin项目，发行了自己的货币。

好了，本课的内容就到此结束了。

我们下一课再见。