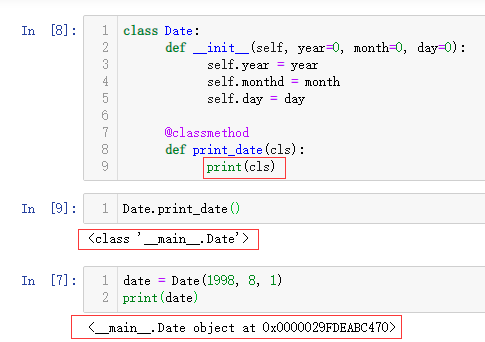
classmethod和staticmethod都是装饰器，不考虑类的继承的情况下，classmethod指类方法，staticmethod指静态方法。classmethod一般都可以取代staticmethod(实际使用也建议这么做，除非确定函数不需要类和类对象)，classmethod比staticmethod多了一个cls参数，它是类自身，在python中，类自身也是一个对象。例：

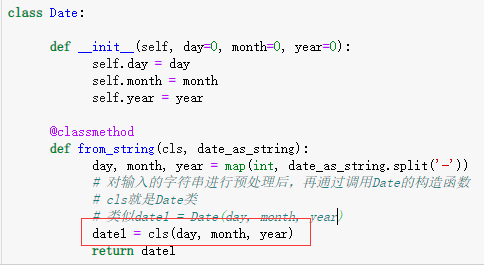


staticmethod更接近于C++中的静态方法，但它也可以被类的对象调用。

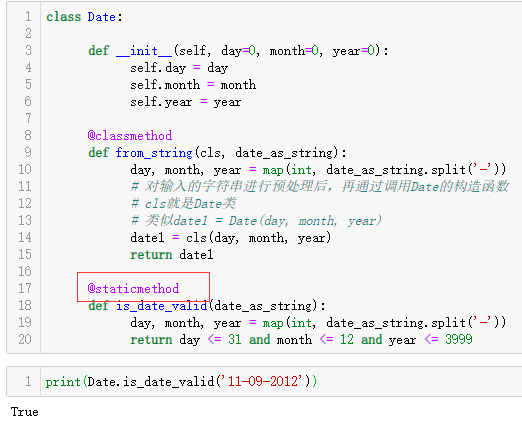
classmethod和staticmethod的一个区别是：

classmethod主要是和类进行交互，一般不与类的实例交互，cls也无法访问类的成员方法，因为一般成员方法的第一个参数self需要类的实例(对于某些类的成员方法虽然传入了self，但函数中没有调用，通过cls也能调用该方法，但这样的方法在pycharm中会提示你转成staticmethod，其实这样做没有意义)。

classmethod的一个重要的作用就是可以类实例化前，进行预处理，这样可以帮类实现类似C++中的多个构造函数。因为Python并不支持多个构造函数，所以可以用classmethod来实现。



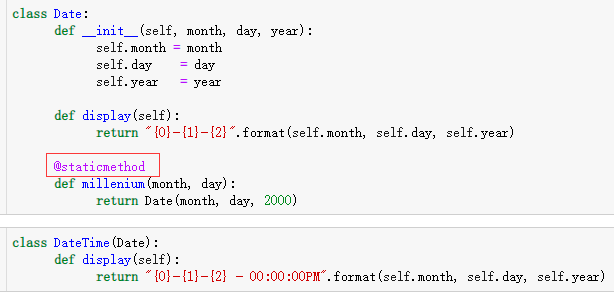
staticmethod主要是和类相关，但不要类和类的实例参与的方法。例：判断日期是否合法



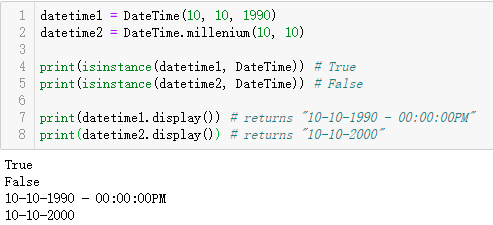
classmethod和staticmethod的另一区别是继承方面：

classmethod中的cls是调用类，这样可以通过它返回子类的对象，而staticmethod无法实现类似的功能。

staticmethod:

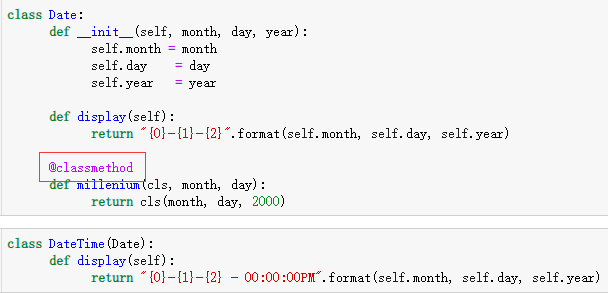


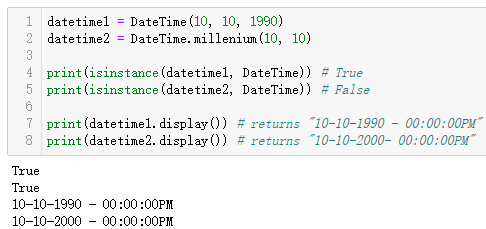
测试：



可以看出staticmethod返回的对象还是Date，而不是DateTime，因为millenium中使用的就是Date创建的对象

classmethod：





可以看出classmethod中的cls是子类DateTime，而不是Date，之后display()也调用的是子类的成员函数。