# 用类、用面向对象有什么用？

首先我们会很直观提出一个问题 —— class 这个东西有什么用呢？

在我刚学编程的时候，这个也是花了很长时间才厘清头绪，但是，我回过头来想，其实面向对象的方法就是blueprint 先于实体，我要造一栋楼，我先画一个图纸；我要造一辆车，我先定义这辆车的图纸。

还记得我们之前讲“don’t repeat yourself” 和函数的时候，讲到一个流程装备线么？

我们来假设一个场景

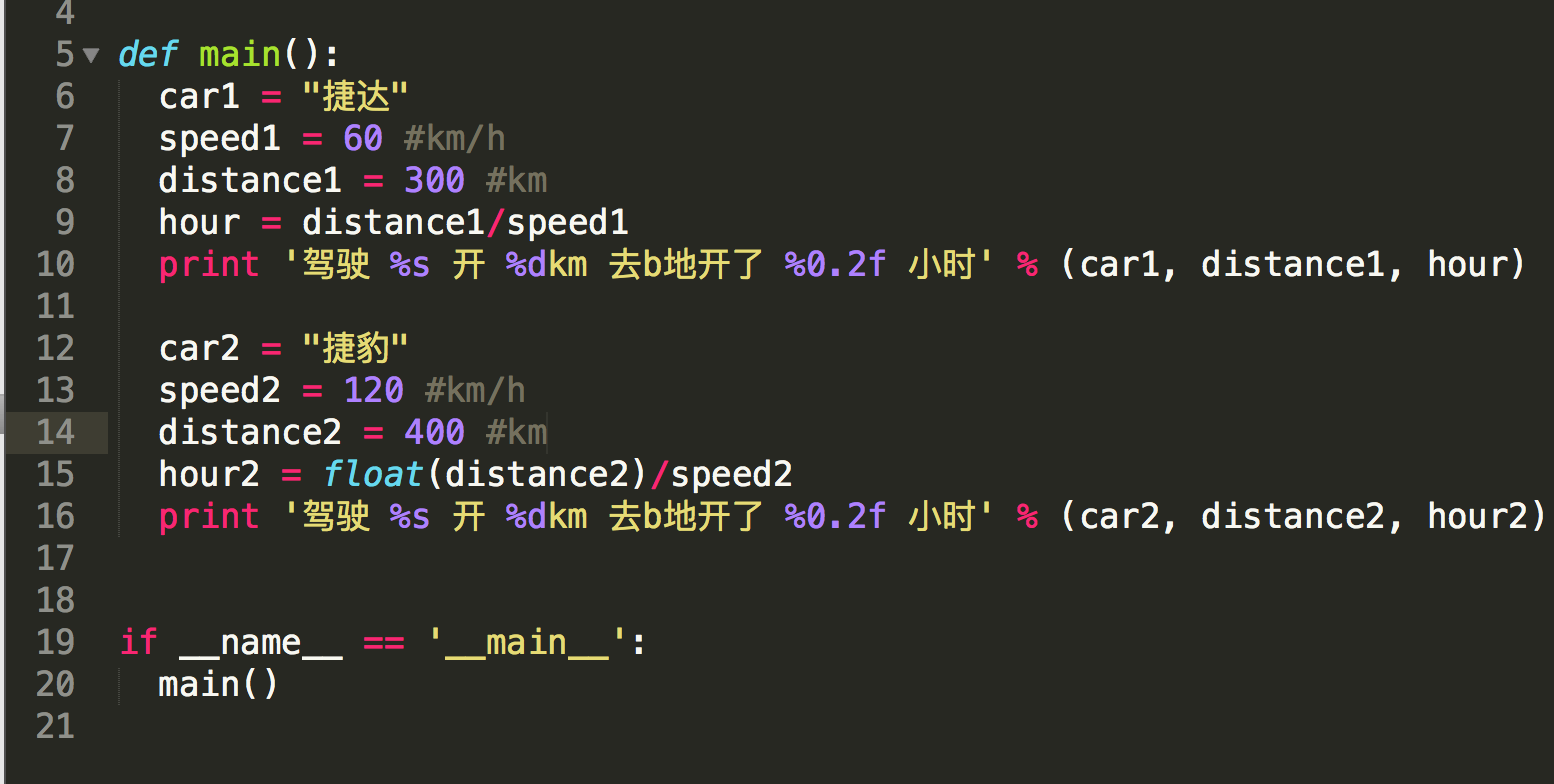
一辆车是捷达，开300 公里，时速可以到 60km，显示一共花了 几小时

一辆车是捷豹，开400 公里，时速可以到 120km，显示一共花了 几小时

我们写代码实现一下

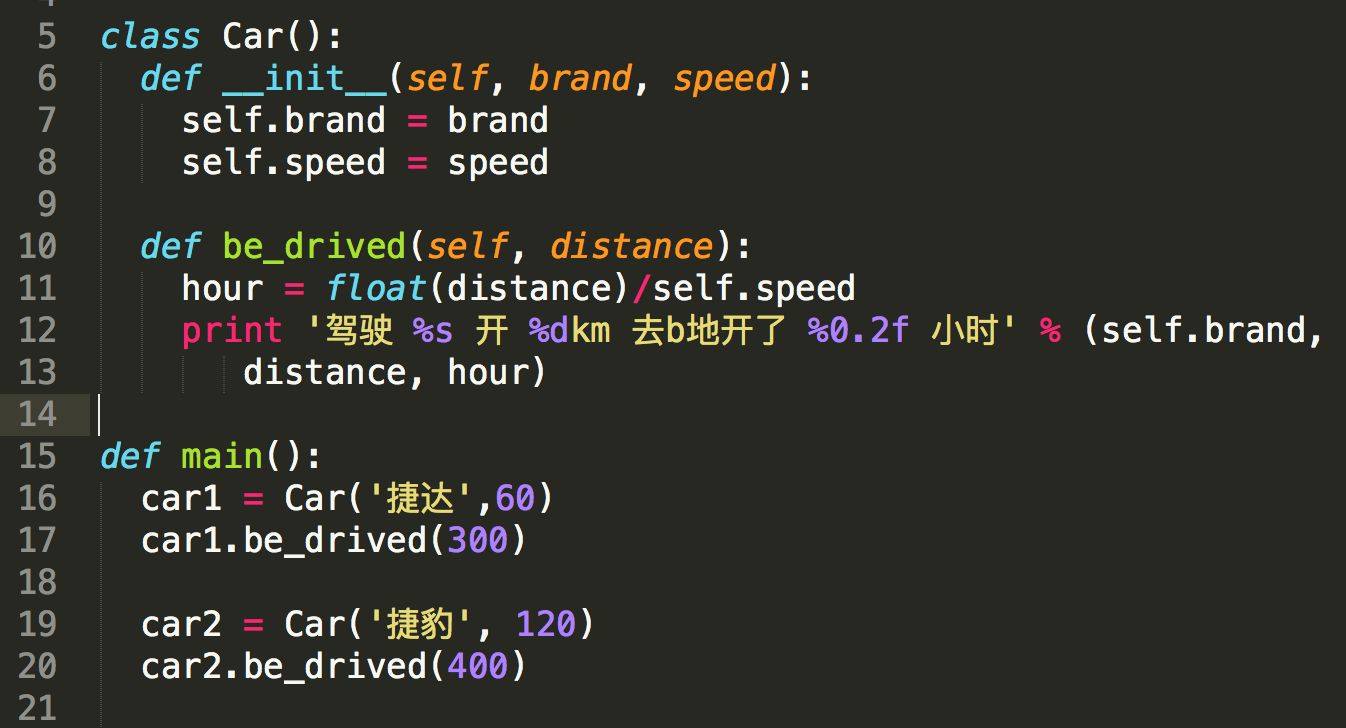
面向过程的写法：

面向过程的代码就有点面条代码了



面向对象的写法:

先定义一个车～然后定义它的属性，然后定义它的行为



如此以来，当这个行为被调用1 ～2 次的时候你感觉不出来，但是，如果没有抽象成一个对象，那么你的程序以后就没办法定义清楚行为。

想象一下，如果描述一个十字路口的交通，如果按照面向过程的写法，该是多么长的一段代码啊，然而代码之间还难以搞清楚关系，因为太多变量太多状态了。

而面向对象的代码，可以把数据、状态等信息都放在一个 class 的实例里面去管理，等于是给了一个笼子，把它们都关起来了。

如上面我们把 drive这个函数（其实应该是be\_drived) 这个行为封装了

就是一种抽象和封装

**所以，如果我来讲面向对象的概念的话，我会总结成两个点：**

1. 把代码的世界和真实的世界联系起来，并做到足够地抽象和封装
2. 对于真实世界的抽象能够形成一张蓝图，这张蓝图可以重复生产出实例出来
3. 重复这个过程，直到你描述足够简洁，可以复用

**作业**

1.把我给的一段面向过程的代码修改成面向对象的方式，**截图打卡**

2.思考一下面向对象的意义和价值，不用打卡