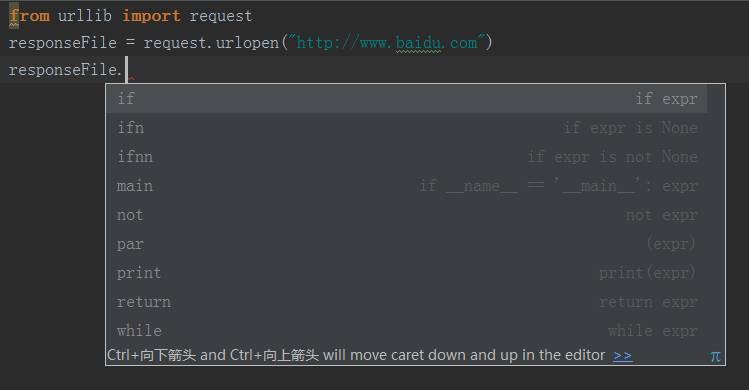
python和Java不同，是一门动态语言：在创建对象的时候，我们并不需要声明对象的类型。

这在简化书写方面带来了极大的便利，但是对于IDE来说，智能地辨别这些类型，并给出相应的代码提示就十分困难。

比如爬虫的代码：



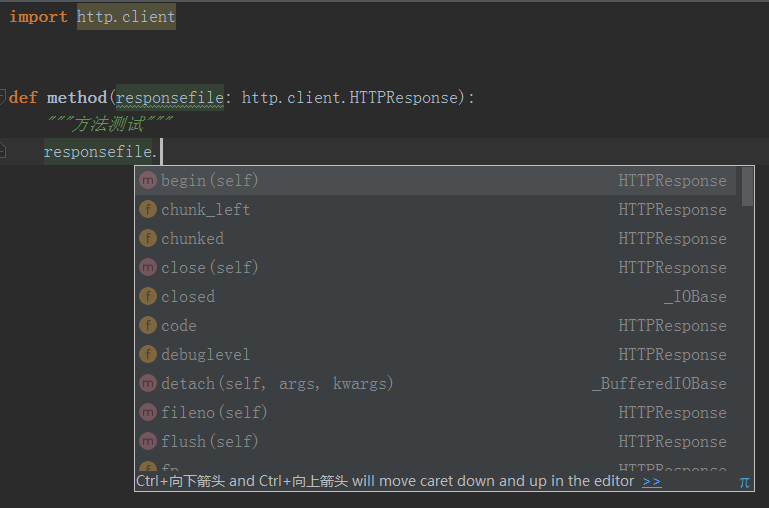
IDE**通常无法有效地识别类型**，然后给出相应的提示。

实际上pycharm提供了称为Type Hint的方法，来给IDE进行类型提示，以更便捷地获取代码提示。参考网站：https://www.jetbrains.com/help/pycharm/type-hinting-in-pycharm.html

下面是几种获取代码提示的方法：

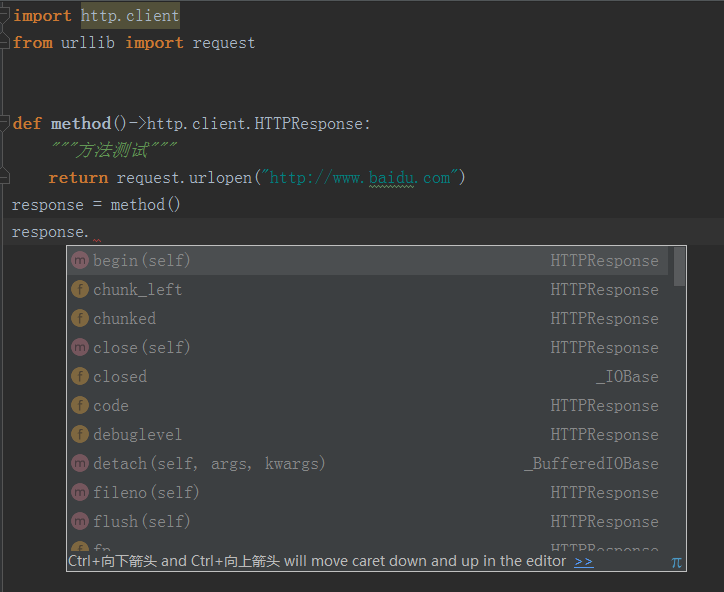
1、自己写的方法中添加类型提示

在定义方法的时候，可以为参数设置类型，如果设置了类型，就可以在接下来的代码里获取相应的提示，比如：



这样就可以顺利地获取提示。

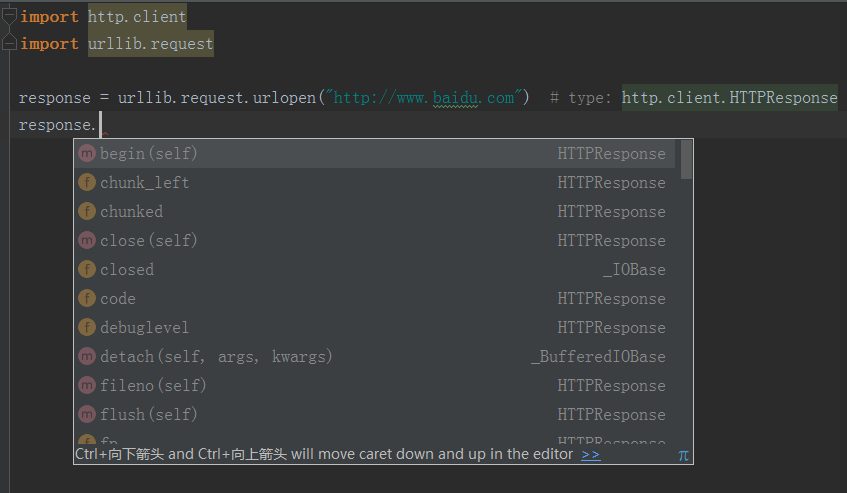
同样如果方法有返回值的话，可以为方法的返回设置类型，这样在使用方法获取对象时，就可以在接下来的代码里获取相应的提示，比如：



这样也可以顺利地获取提示。

2、使用注解注释本地变量

如果遇到使用第三方包的情况，而且提供包的人并没有给完整的type hint，再去重写他们的方法显然不现实，这时可以通过在对象后添加注解来提示IDE，以获取提示。比如：

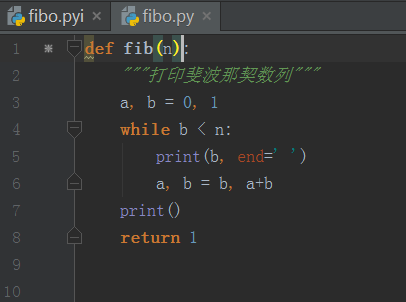


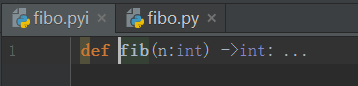
格式为：

# type: 对应的类型

3、使用 [Python stub files](https://www.python.org/dev/peps/pep-0484/#stub-files)来储存类型提示

python stub文件实际上就是将代码和注解分离，单独将注解存在同名的.pyi文件中。例如：





这样就可以将代码从繁琐的类型提示中解放出来。注意到方法左侧有一个\*，这代表存在这个方法的stub。

4、在doc string中声明类型

python语法中，提供以三个双引号包围的文档字符串，用于方法的注释。

这里不进行详述。