

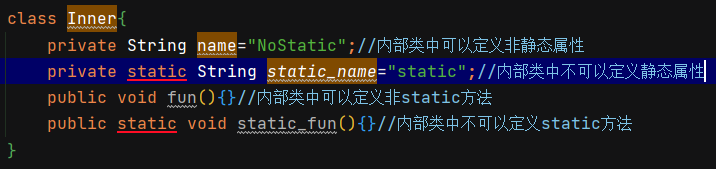
1，成员内部类，顾名思义，作为外部类的一个成员存在，与外部类的属性，方法并列

内部类依附于外部类，只有先创建了外部类，才能创建内部类

2，成员内部类可以自由的访问外部类**所有的属性，方法(**private也行，static也行)

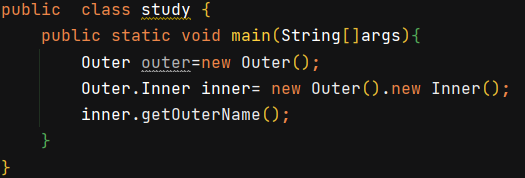
但是外部类想要访问内部类的属性，方法必须通过内部类实例才行，不能直接访问，并且，即使实例化了，内部类的private成员也不能直接访问

3，成员内部类中不可以定义static属性，static方法



4，如果内部类的声明不是private，那么其他类可以通过

外部类.内部类 引用=new 外部类().new 内部类() 来创建内部类



**但如果内部类的声明是private，那么除了外部类，其他类就无法访问**

5，注意，内部类是一个编译时的概念，一旦编译成功，就会成为完全不同的两类，

这里会形成 Outer.class文件 和 Outer.Inner.class文件