

Task1.对象和类

引入：我们都知道Java是一门面向对象的语言，而C是面向过程的语言，那么究竟什么是面向对象编程（OOP）呢？

面向对象编程（OOP）是一种编程范式，它将数据和操作数据的方法封装在对象中。主要特点包括：

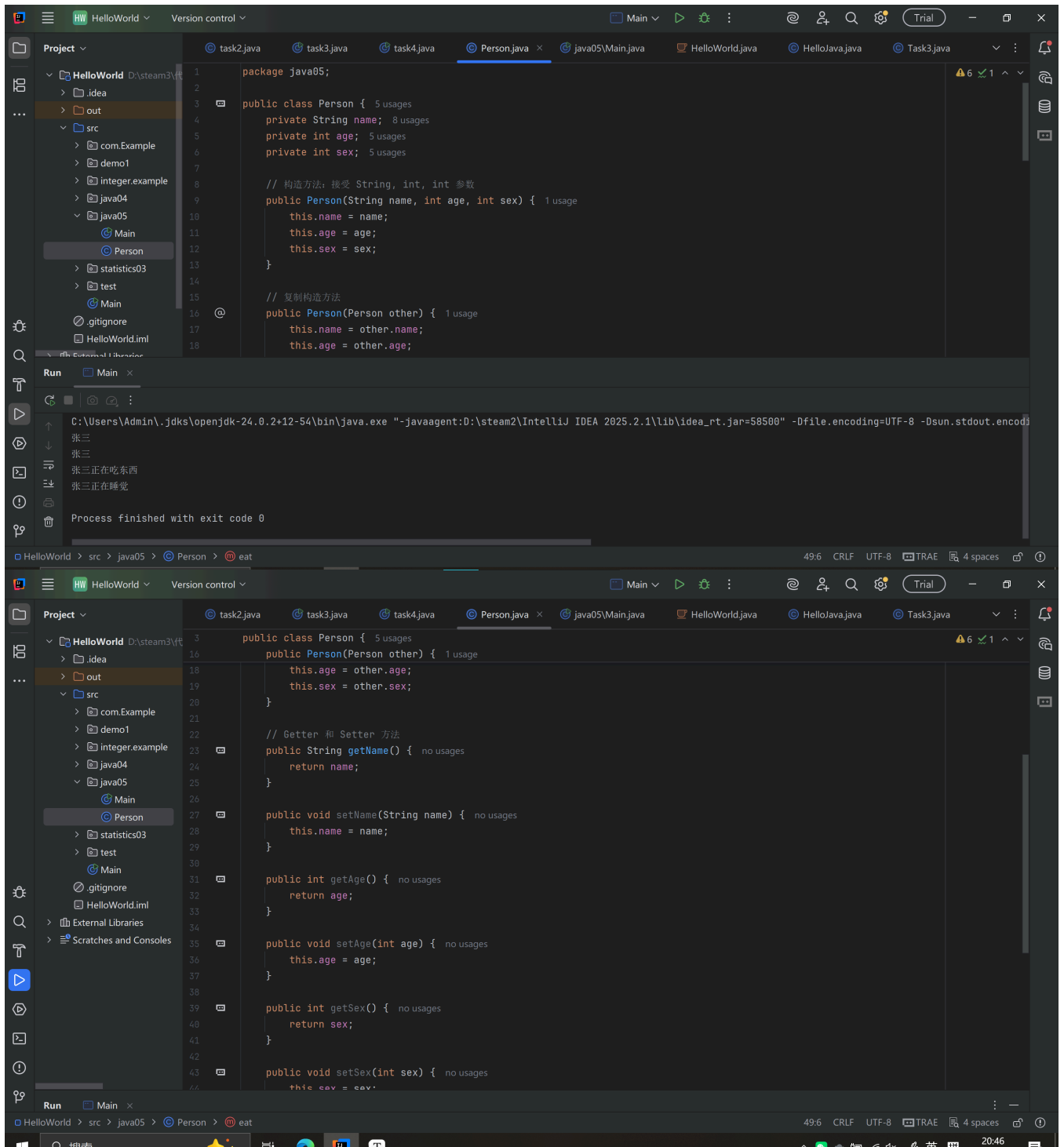
1. **封装：**将数据和方法绑定在一起，隐藏内部细节。
2. **继承：**允许创建新类时重用现有类的属性和方法。
3. **多态：**不同对象可以响应相同的消息，但行为各异。

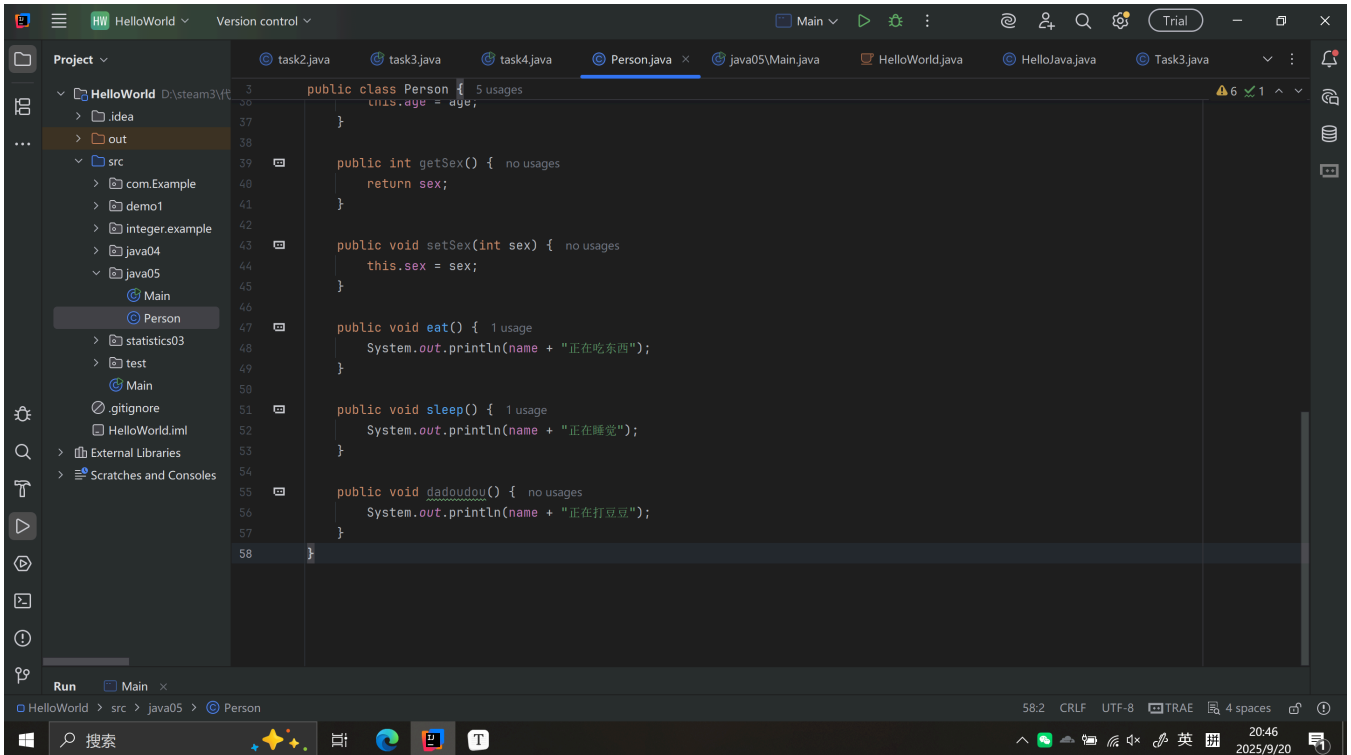
通过这些特性，OOP 提高了代码的可复用性、可维护性和扩展性。

为了完成下面的要求，请你先创建一个java项目，该项目有一个主类（名字任意，里面有一个main函数作为程序入口），一个Person类，Person类的内容在上面已经给出。

1、请你为这个Person类添加构造方法实现复制对象，并在题解附上你的Person类代码。你的构造方法用到this关键字了吗？请说说它的作用。

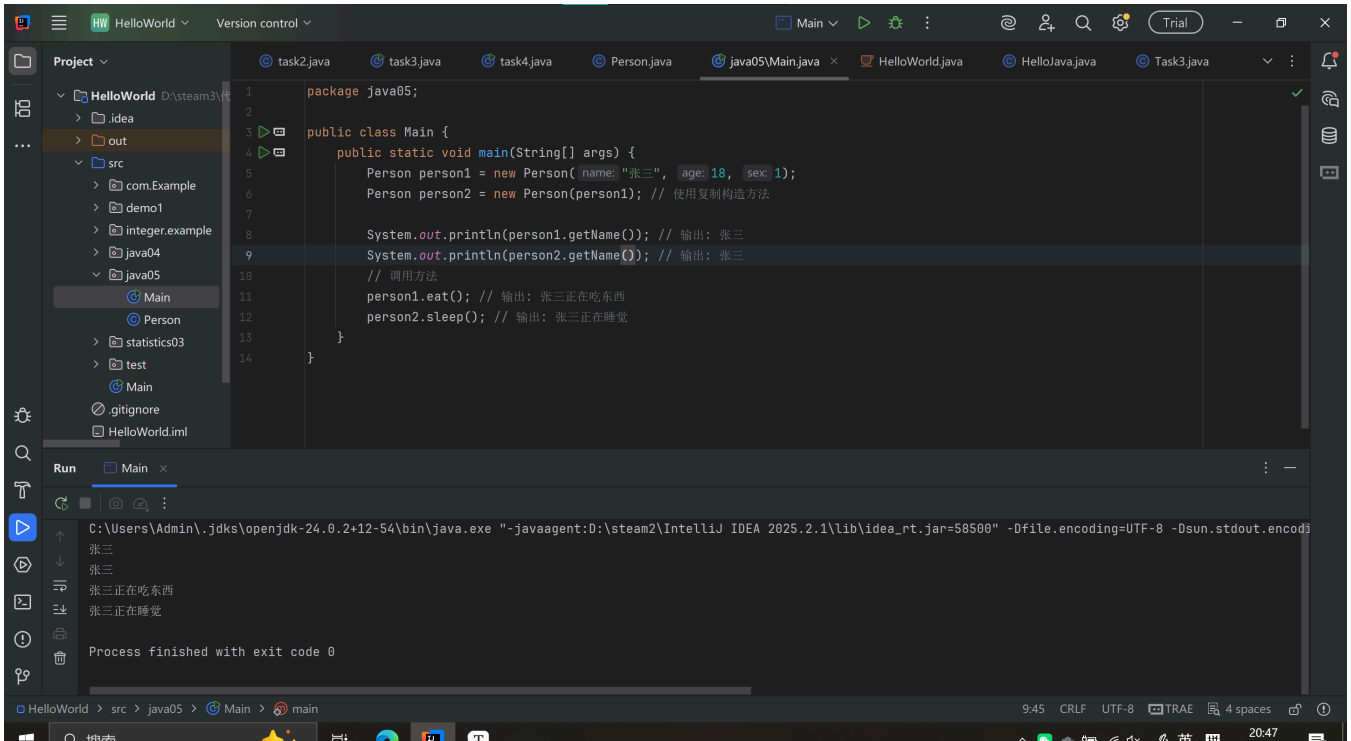
使用了this，作用：在构造方法中，`this` 关键字用于引用当前对象的属性，避免与参数名冲突。





2、在主类的main方法中创建Person类的一个对象，并给它的字段赋值（可以用构造函数，也可以用引用变量）。说说对象和类的关系。（就像真正的人有隐私不能让别人知道一样，对象也可以限制外界访问自己的属性和方法。）

对象和类的关系：类是对象的模板或蓝图，定义了对应的属性和行为；对象是类的实例，具有具体的属性值和行为。



3、学习访问修饰符，为你的Person类的字段和方法添加你认为合适的访问修饰符。尝试在不同的位置（当前类，相同包的其它类，包的外部等）访问这些字段和方法，并总结出各种访问修饰符的限制范围。

总结访问修饰符的限制范围

- **public**: 可以被任何类访问。
- **protected**: 可以被同一包内的类和子类访问。
- **default (无修饰符)**: 只能被同一包内的类访问。
- **private**: 只能被定义该字段或方法的类访问

Task2.类中的变量和方法

回想之前调用Person类的eat方法时，是不是必须先创建对象，再使用对象来调用方法？

类中的变量和方法分为成员（实例）和静态两种：

- **成员变量**: 每个对象独立拥有，互不干扰。例如，`Person` 类中不同对象的 `name` 可以是不同的值。
- **静态变量**: 所有对象共享同一份数据。例如，类变量。
- **实例方法**: 需要通过对象调用，如 `eat()` 方法。
- **静态方法**: 直接通过类名调用，无需创建对象。