

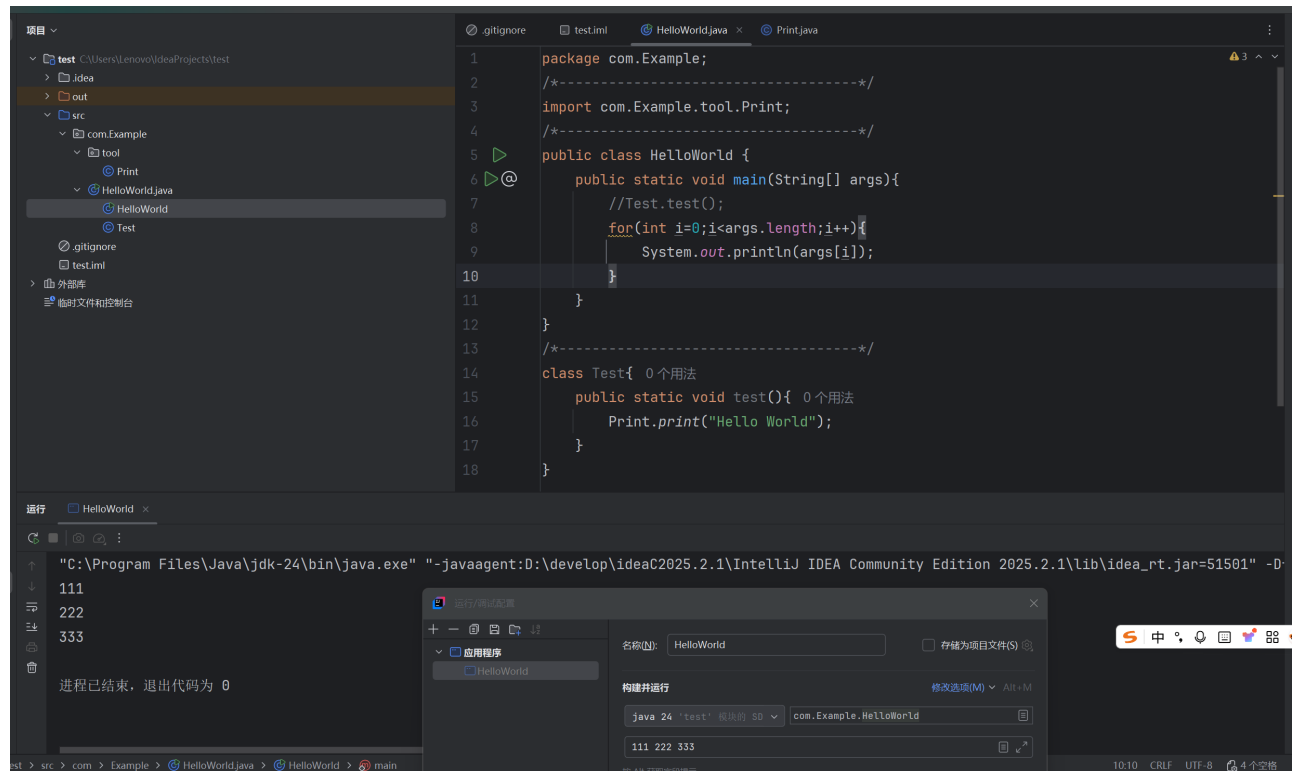
Java02

Task1

1. **第一部分**声明了一个包，它定义了当前java文件所在的包，即HelloWorld.java位于"（项目名称）/src/com/Example"。**第二部分**是导入声明，通过它，可以直接使用位于另一个包（com.Example.tool）中的Print类，而无需写全名（即com.Example.tool.Print）。**第三部分**第三部分声明了一个HelloWorld类，里面包含了程序的起点——main函数。**第四部分**定义了另一个类Test，它包含了一个静态方法test()，该方法调用了从其他包导入的Print类的print方法。
 2.
 1. 包是Java中用于组织和管理类（Class）和接口（Interface）的一种命名空间机制。它本质上对应着文件系统上的目录结构。
 2. 包的作用有：**1、避免命名冲突**：这是包最重要的作用。不同的开发者或不同的项目可以使用相同的类名，只要它们属于不同的包即可。**2、更好的组织类**：将功能相近或相关的类组织在同一个包中，使代码结构更清晰，易于管理和查找。**3、提供访问控制**：Java的访问控制修饰符（如protected和默认包权限）的作用域是基于包的，这有助于实现封装和信息隐藏。
 3. main函数是Java应用程序的入口点。当使用java命令运行一个编译后的Java程序时，Java虚拟机（JVM）会去寻找一个具有特定签名的方法来开始执行程序，这个方法就是main方法。它的签名必须严格定义为：`public static void main(String[] args)`
 4. 一个单文件Java程序的基本结构为：**1.包声明（0或1句）**：例如`package com.example.mypackage;`必须是文件的第一句有效代码（注释除外）。如果省略，类将位于无名默认包中。**2.导入声明（0句或多句）**：例如`import java.util.Scanner;`用于引入其他包中的类，方便在当前类中使用。位于包声明之后，类定义之前。**3.类定义（1个或多个）**：**公共主类定义（1个）**：文件名必须与此公共类的类名完全一致（如HelloWorld.java）。这个类中包含程序的入口点main方法。`public class HelloWorld { ... }`**其他辅助类定义（0个或多个）**：可以在同一个文件中定义其他非公共类，作为主类的辅助类。例如`class Test { ... }`（注：虽然一个java文件可以包含多个类，但最多只能有一个被声明为public的类，并且该公共类的类名必须与文件名相同。）
2. 修改如下：

```
public class HelloWorld {
    public static void main(String[] args){
        //Test.test();
        for(int i=0;i<args.length;i++){
            System.out.println(args[i]);
        }
    }
}
```

运行结果截图：



Task2

Print.java内容如下：

```
package com.Example.tool;

public class Print {
    public static void print(String x ) {
        System.out.println(x);
    }
}
```

运行结果截图：

