**Java上机作业4**

1131190111-唐川淇

1. **简答**
2. 什么是封装性？用什么权限访问修饰符做到封装？为什么能做到？

答：:以类为载体,将类的属性、行为(可以理解为Java中的方法)封装起来,对使用者隐藏其具体实现细节,只对外公开简单的接口。Private、缺省、protected、public。在不同包下的子父类中： public修饰的能被访问，对于protected修饰的，子类的实例可以访问从父类继承而来的protected方法。即在子类中的某个方法fun1()直接访问父类的protected方法，再去实例化子类，通过子类的实例调用fun1()，从而间接实现访问父类的protected方法，但不能通过父类的实例直接去访问其protected方法。

1. 基本数据类型的赋值原理是？引用类型的赋值原理是？

答：赋值：直接存储在栈中作为临时变量使用。引用类型的赋值：创建对象指向来进行赋值操作。

1. C++和Java中增强for循环分别是怎样的？

答：for(元素类型 变量 ： 数组/Iterable对象){

//TODO

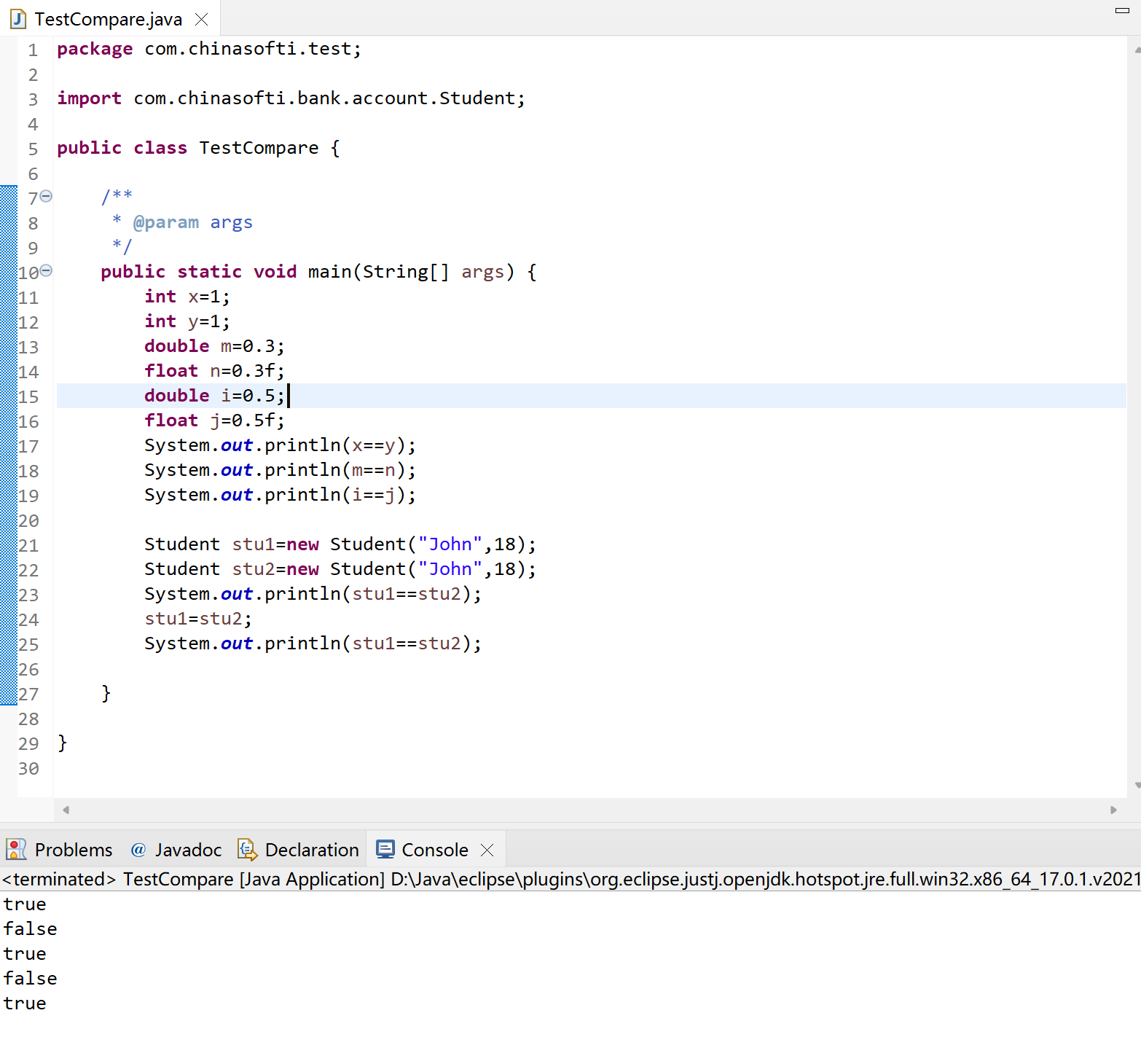
}

1. **代码补全并运行**

使用Eclipse或者IDEA加载群文件《上机4代码》中的代码，将运行结果截图，并解释为什么会输出这样的结果：

1. TestCompare.java

答：



Int为基本数据类型，==判断是否值相等，所以输出true。

0.3和0.3f二进制不同，输出false。

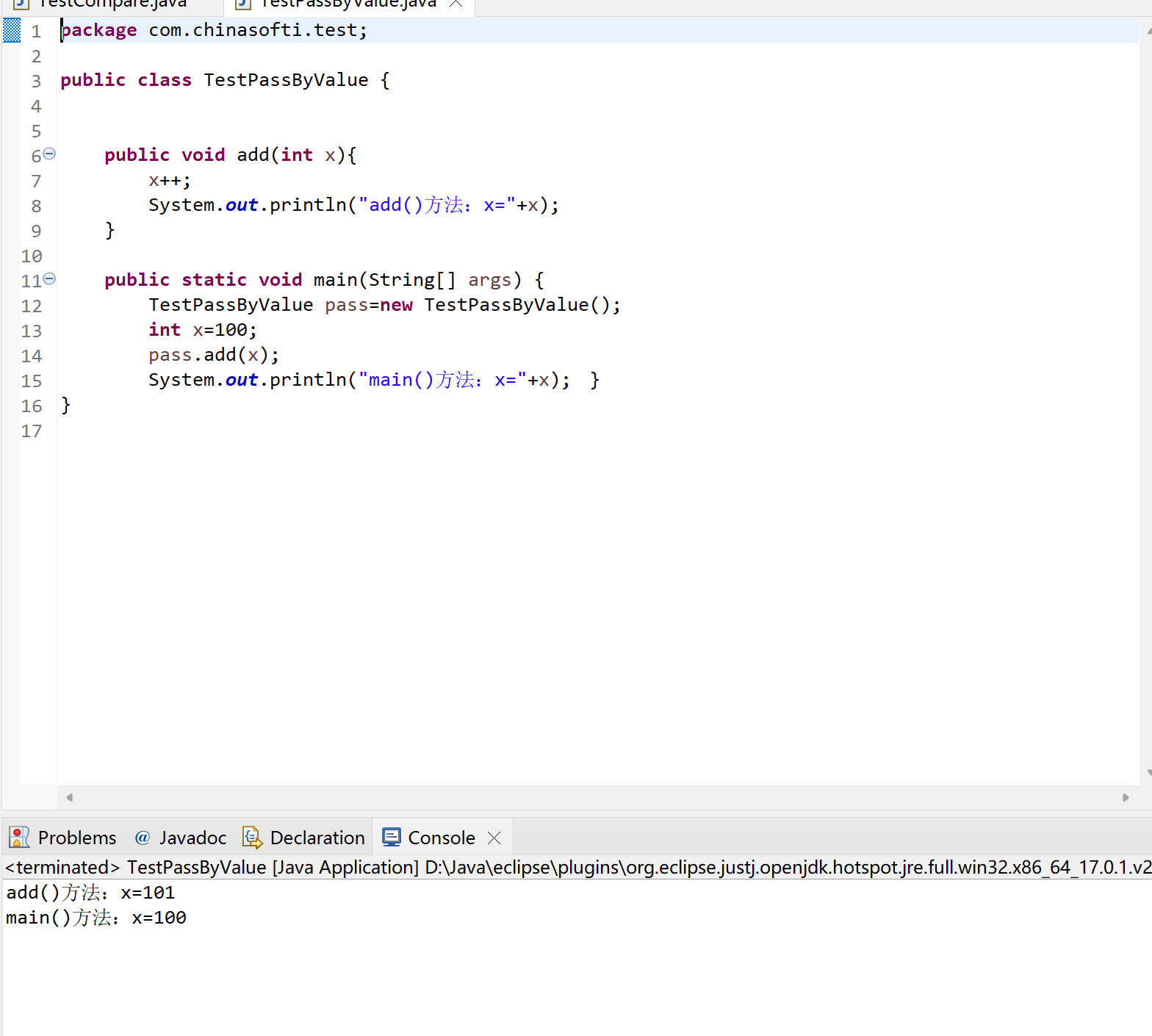
0.5和0.5f二进制相同，输出true。

引用类型比较内存是否相同，false。

两字符串共用一片内存，true。

1. TestPassByValue.java

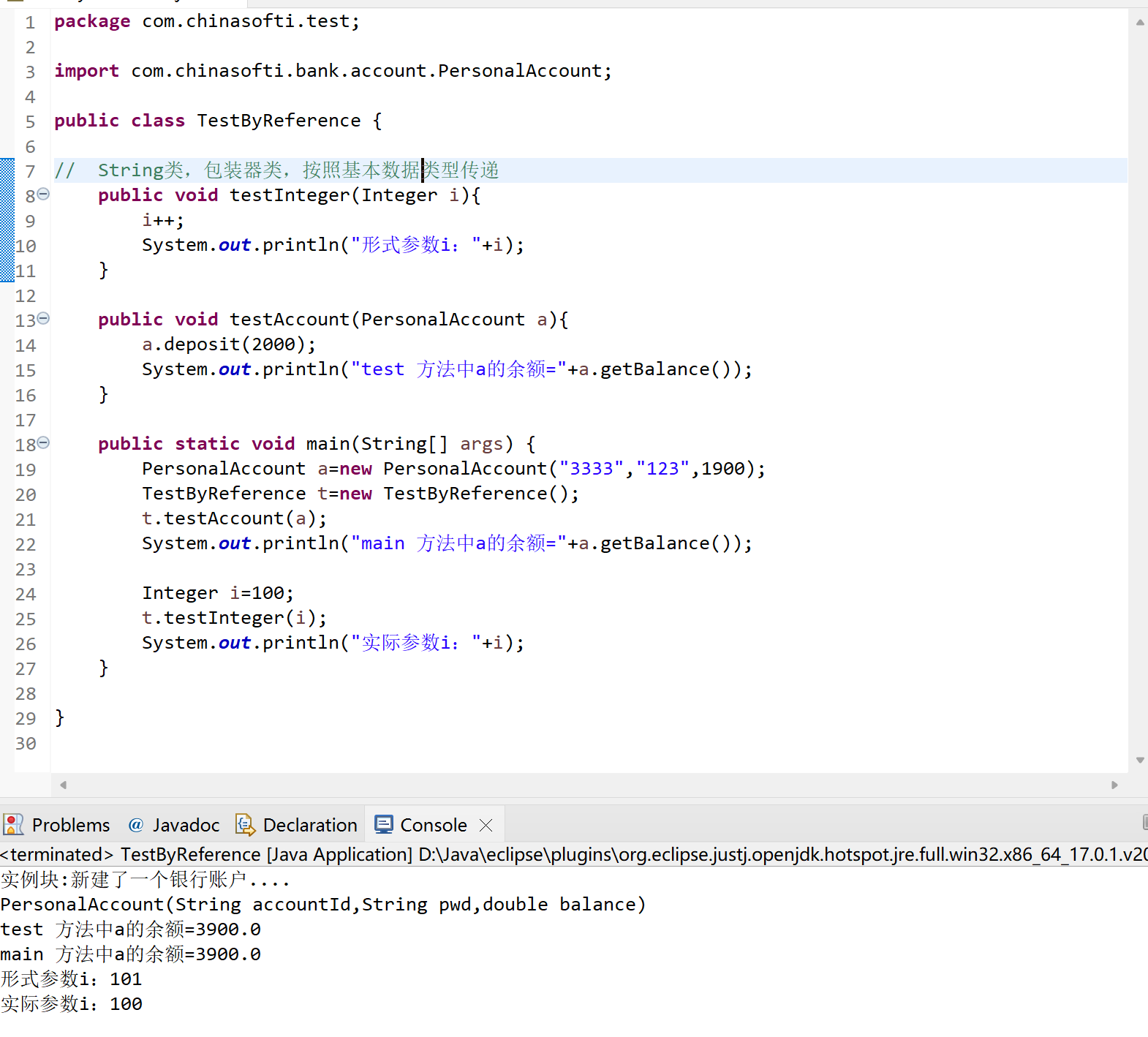
答：



X本身存储在全局变量区，值为100，当传入函数后产生了新的x存储在栈区，作用域为add函数内，值从100变为101，运行结束之后释放栈区x，此时全局变量区x仍为100。

1. TestByReference.java

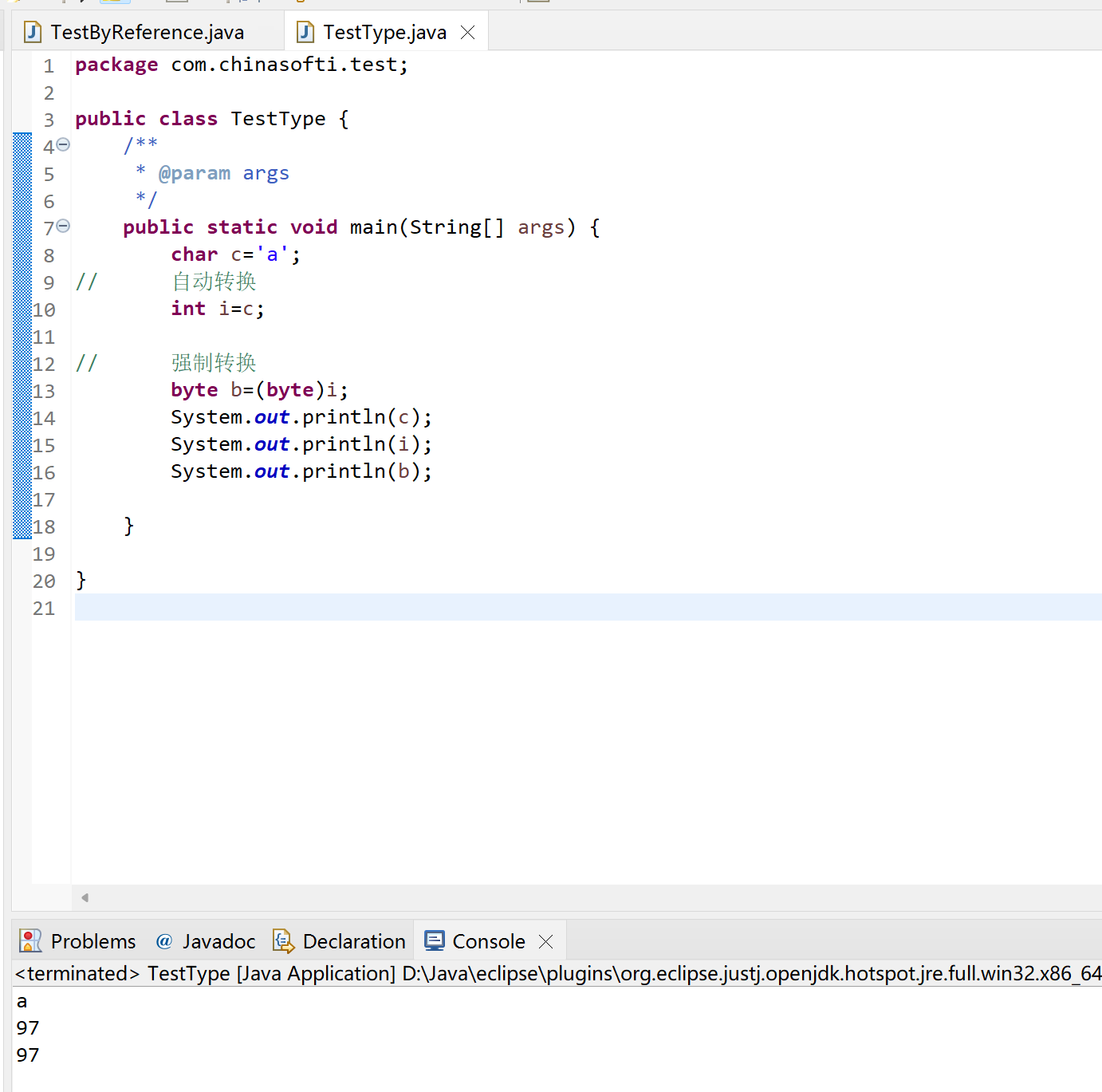
答：



引用参数和原变量占用同一片内存，可以对原值修改。

1. TestType.java

答：



C为字符，i转换为ASSIC码，

1. TestWrapper.java

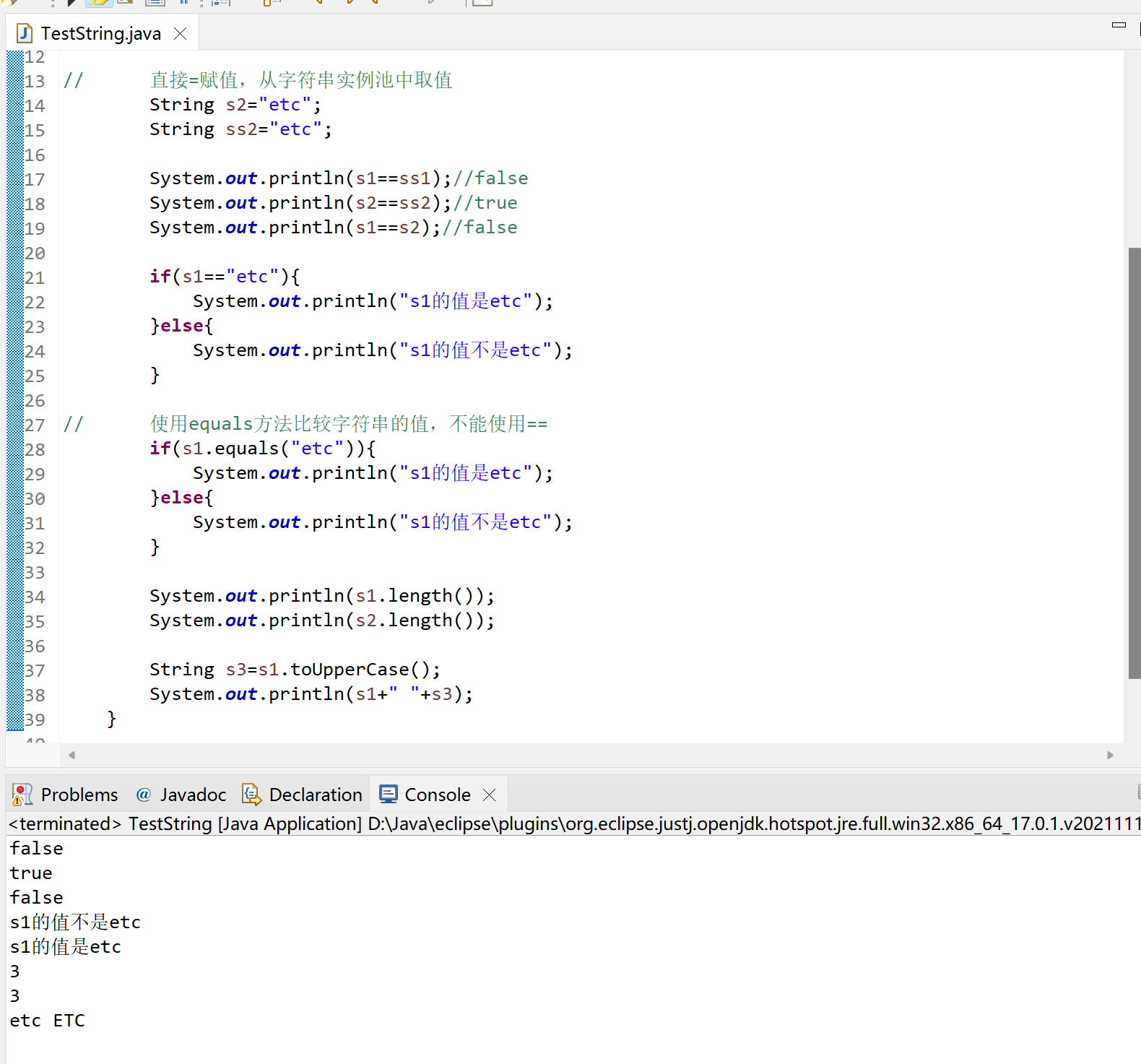
答：



在封装中有一种特殊的类，能够把基本的数据类型进行转换来方便实际的使用。

1. TestString.java

答：



String中用“==”比较的是地址，用equals比较的是内容。

1. TestStringBuffer.java（举个例子）

答：运行结果

解释