**实 验 报 告**

|  |  |
| --- | --- |
| **课程名称：** | **Java语言程序设计** |
| **学生姓名：** | **葛子午** |
| **学生学号：** | **202330550381** |
| **学生专业：** | **软件工程** |
| **开课学期：** | **2024-2025学年第一学期** |

**软件学院**

**2024年10月**

**实验二 面向对象的编程技术**

**一、实验目的**

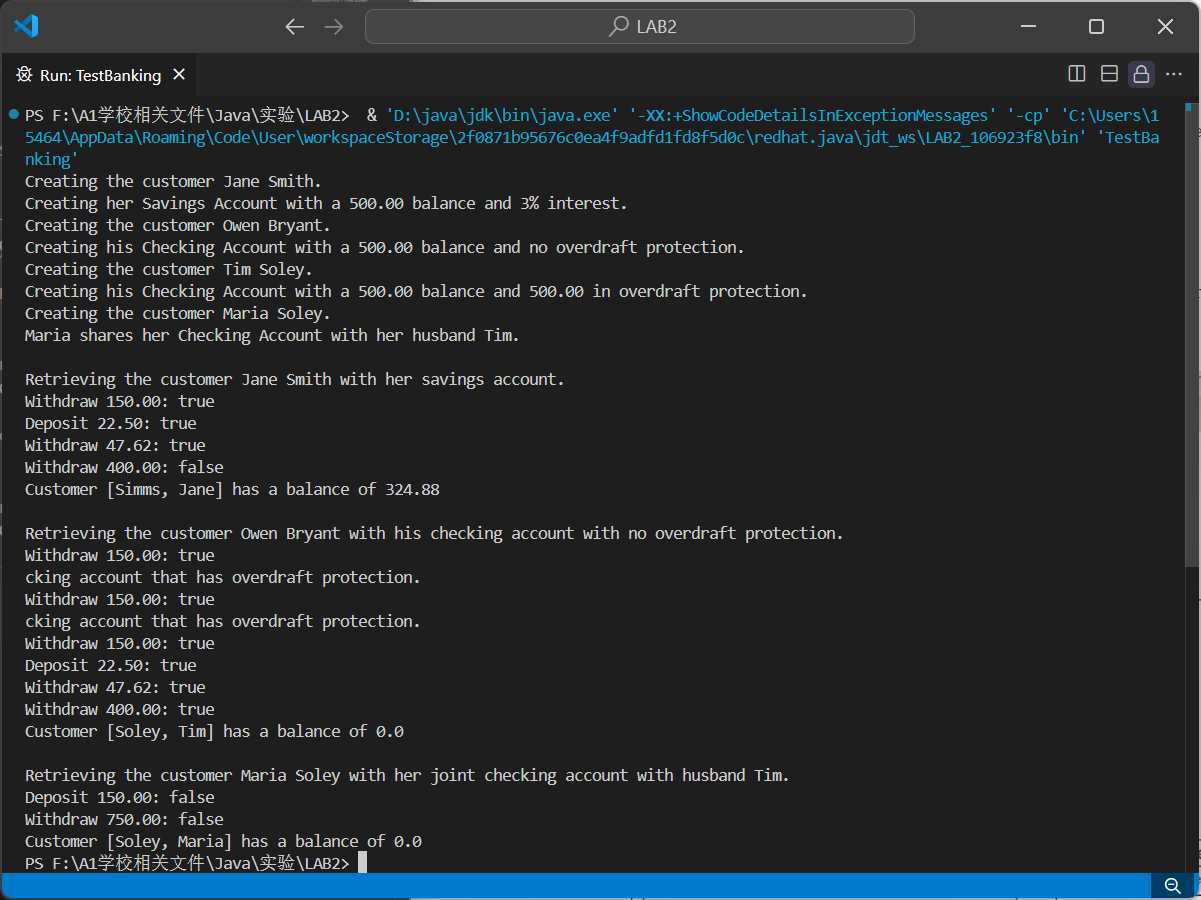
1. 掌握面向对象的继承与多态以及类型转换；
2. 掌握ArrayList类创建动态数组；
3. 掌握Singleton模式的使用
4. **实验内容及结果**

**STEP5:**

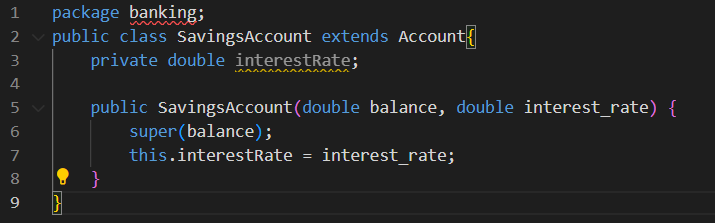
实验内容：

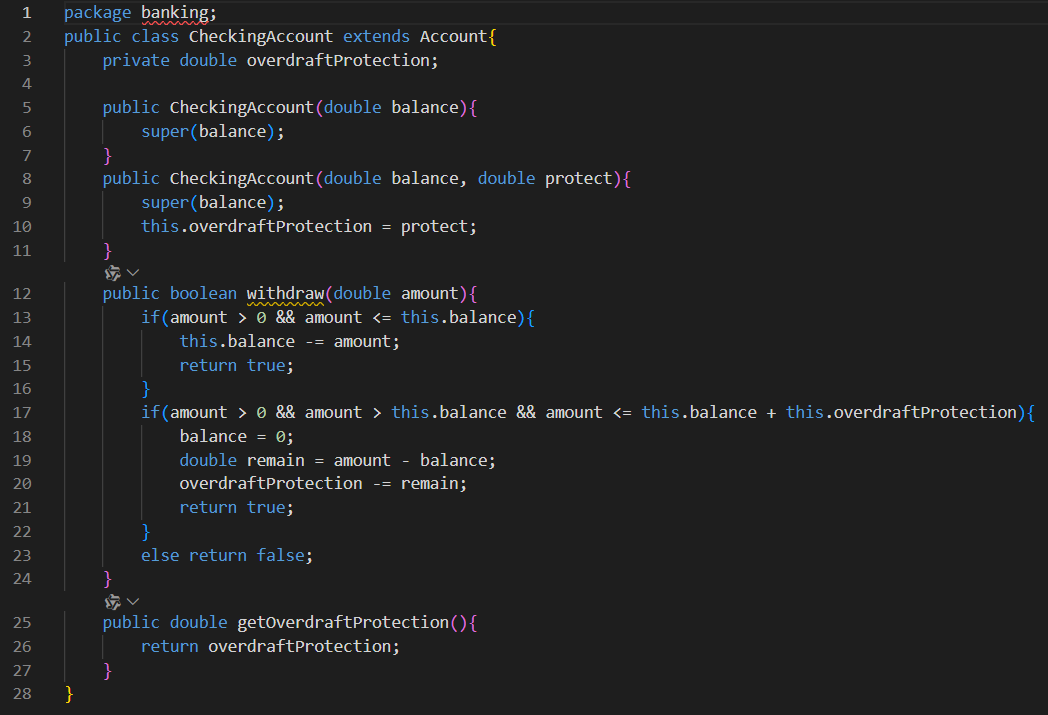
创建Account类的两个子类SavingsAccount 和CheckingAccount类，掌握方法的覆盖，以及使用super()调用父类的构造函数。

实验结果：



代码





**STEP6\_pre1**

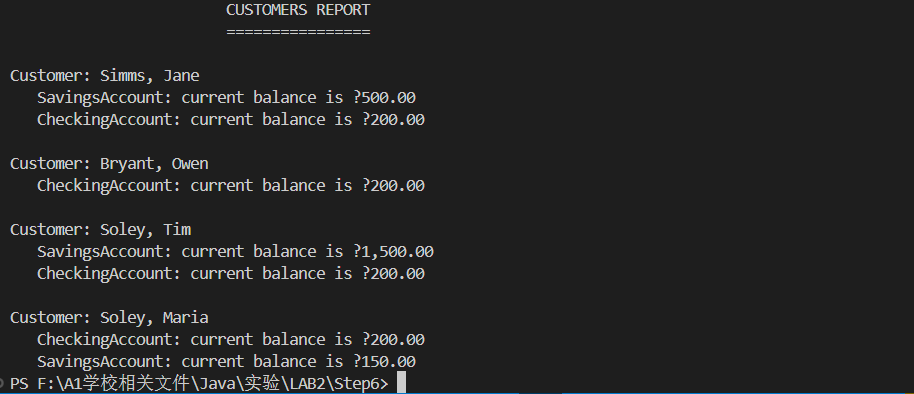
实验内容：

修改Customer类，支持一个客户拥有多个账户，使用java.util.ArrayList类创建客户拥有的不同类型的账户集合。

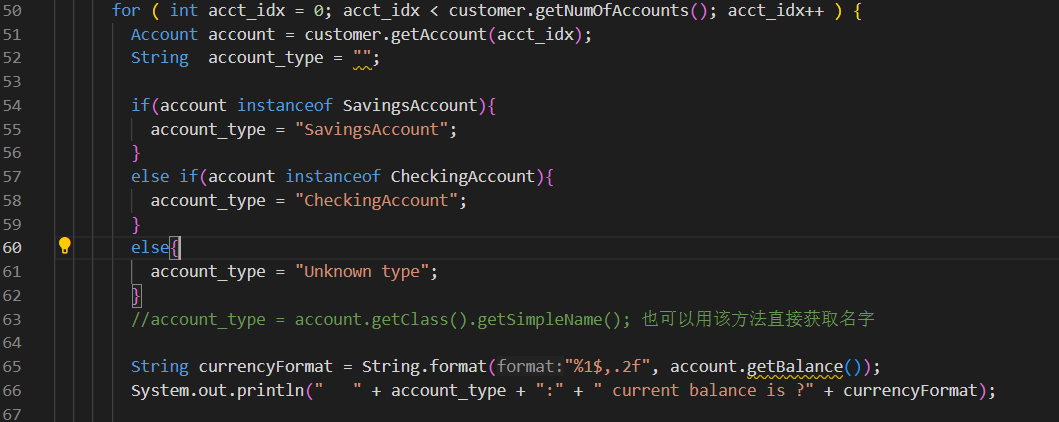
当直接使用ArrayList类时编译出现警告：“Customer类使用了未经检查或不安全的操作”，此为JDK1.5以后版本的泛型警告（泛型类似C++的模板），可忽略不计，或是使用ArrayList的泛型定义ArrayList<Account>。

java.text.NumberFormat表示数值的格式化类，可以按照本地风格习惯进行数值的显示。

实验结果：



代码：



**STEP6\_pre2：**

实验内容：

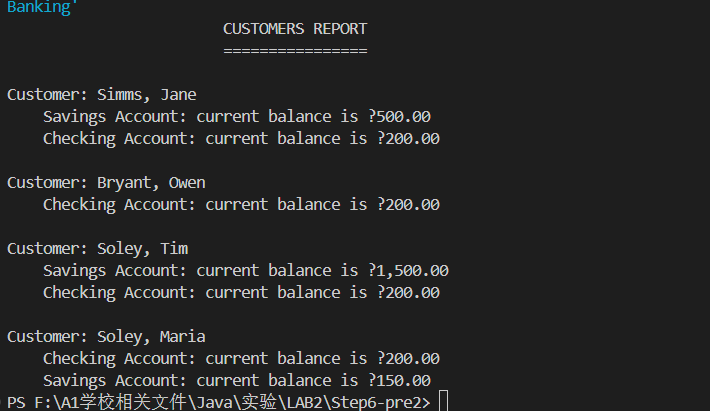
修改Bank类为单例类，实现Singleton单例设计模式。

1）设计模式用于复用好的设计方法，模式的定义包括模式名称、模式应用的问题背景及解决方案，常用的设计模式是23种GOF模式，如Singleton模式、Adaptor模式、Abstract Factory模式等。

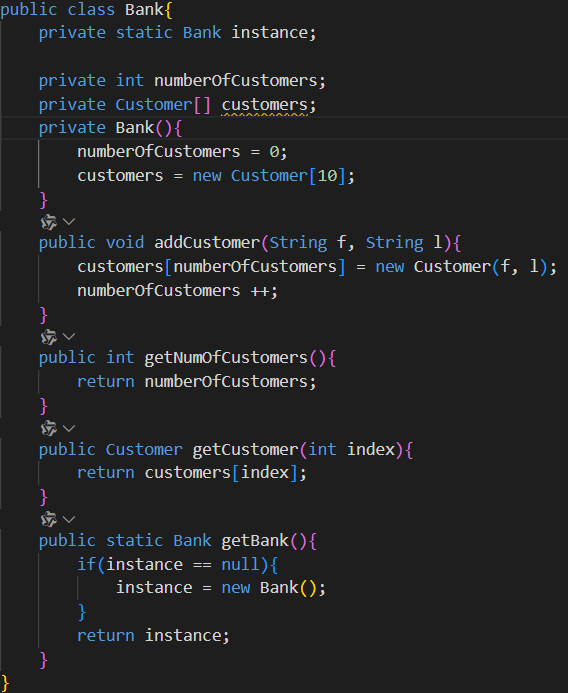
2）单例模式：用于确保一个类仅有一个实例，并提供一个访问它的全局访问点。

3）单例类（ServicesFctory类）结构示例：

实验结果：



代码：





1. **思考题**
2. 思考如何对ArrayList定义的Account集合排序，按照账户的余额由小到大排列？请描述实现思想或关键语句。

答：可以直接使用ArrayList类的sort函数，参数可以使用一个lambda表达式

accounts.sort((a, b) -> Double.compare(a.getBalance(), b.getBalance()));

其中accounts是Account类的ArrayList，比较balance参数大小即可

1. 归纳单例模式的设计原理，思考为什么属性和方法为static ？构造函数为private？

答：单例模式是确保一个类只有一个实例，并提供一个全局访问点来获取这个实例的方法，比如在该例子中利用了Bank类的getBank函数对单例操作，其中Bank类有一个static类型的bank实例。

单例模式的关键特点包括：

私有构造函数：单例类的构造函数被声明为私有（private），这样其他类就不能通过new关键字来创建该类的实例，从而确保了只有一个实例可以被创建。

静态属性：单例类中包含一个静态的实例属性，这个属性持有单例类的唯一实例。因为静态属性属于类，而不是对象，所以这个实例在所有单例类的实例之间是共享的。

静态方法：提供一个公共的静态方法来获取静态属性中保存的唯一实例。这个方法通常被称为 getInstance 方法，它负责创建实例（如果尚未创建）或返回已创建的实例。

为什么属性和方法为static，构造函数为private

①属性为static：因为单例类需要一个全局可访问的实例，所以这个实例必须是静态的。这样，它就可以不依赖于类的任何特定实例而被访问。

②方法为static：获取单例实例的方法也必须是静态的，因为调用这个方法不应该依赖于类的任何特定实例。它应该可以直接通过类名来调用。

③构造函数为private：这是为了防止外部通过new关键字来创建类的实例，这是确保只创建一个实例的关键。通过使构造函数为私有，类的实例化只能在类内部进行，通常是在静态方法中进行。

总结来说，单例模式通过控制实例化过程来确保全局只有一个实例，并提供一个全局访问点。这种模式在管理共享资源时非常有用，例如配置管理器、线程池、缓存等。