《面向对象程序设计与实践》实验报告

实验名称： 《面向对象程序设计与实践》实验七

学 期： 2022春季学期 任课教师： 陆歌皓 学院 软件学院

专业： 计算机 学号： 20211120138 姓名： 薛凡豪 成绩：

# 一、实验源代码及运行结果：

## 内部购物券

### （1）问题分析：

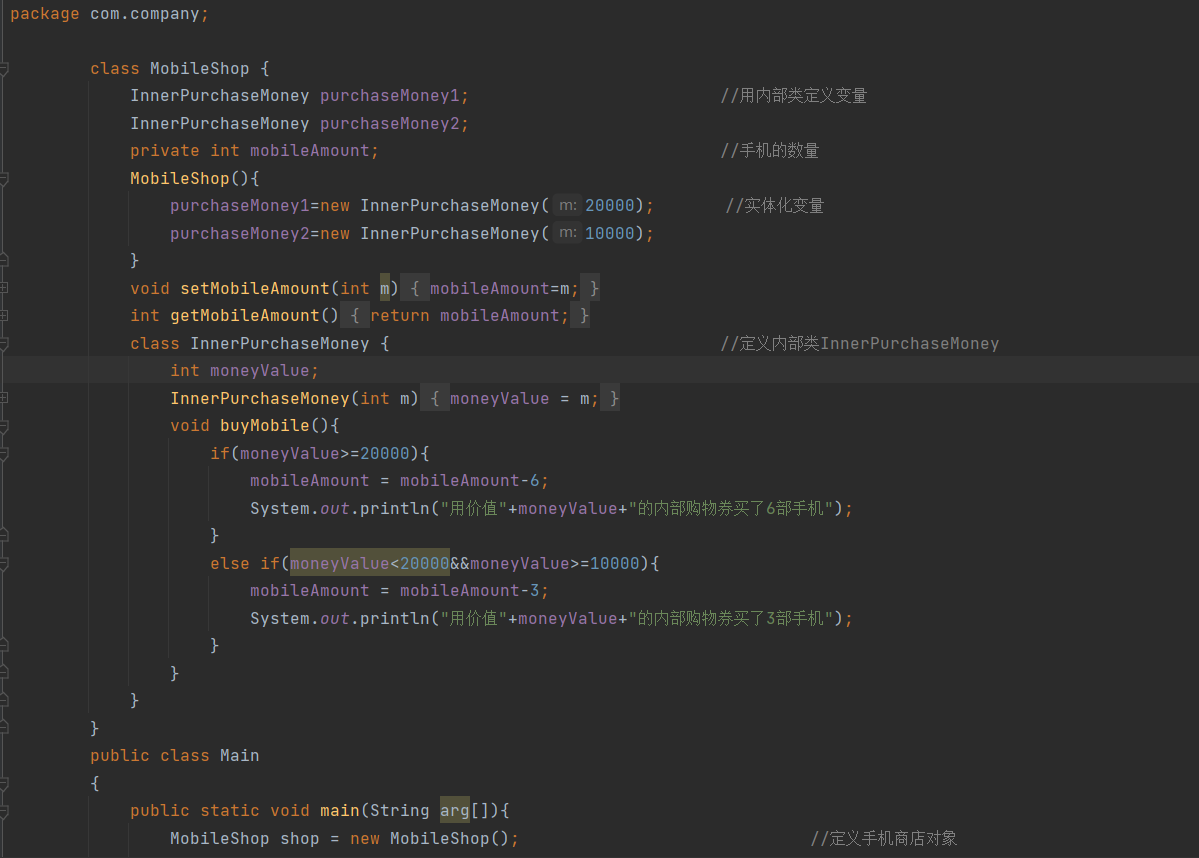
1.本实验主要需要运用的是内部类（即在一个类中再声明一个类），需要注意的是：外嵌类的成员变量在内部类中仍然有效，内部类中的方法也可以调用外嵌类的方法。内部类只能被外嵌类使用，不能被其他类使用。

2.第一个类要写出一个手机店类，来模拟手机专卖店。

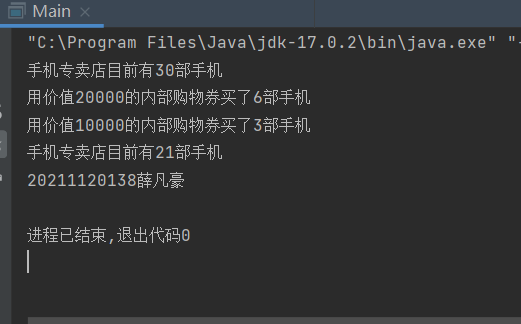
3.第二个类即写手机店类的内部类，即内部购物券类。

4.最后一个运行类，求出买了多少手机和剩余多少手机。

### （2）程序的运行和结果

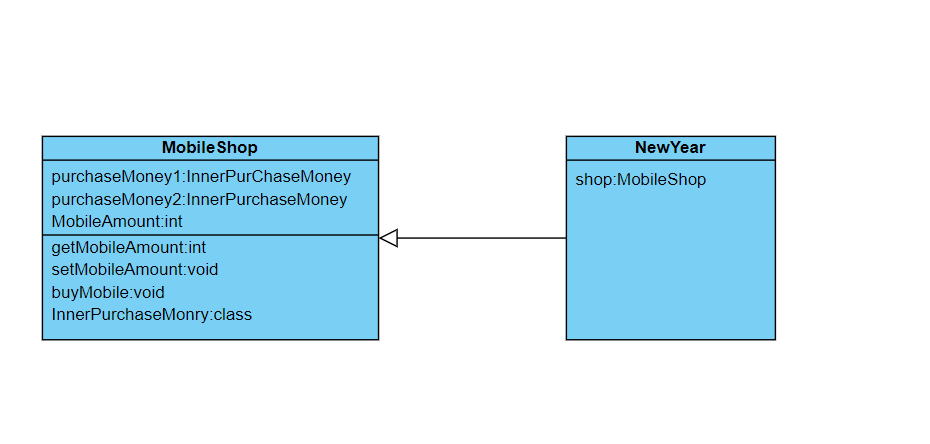




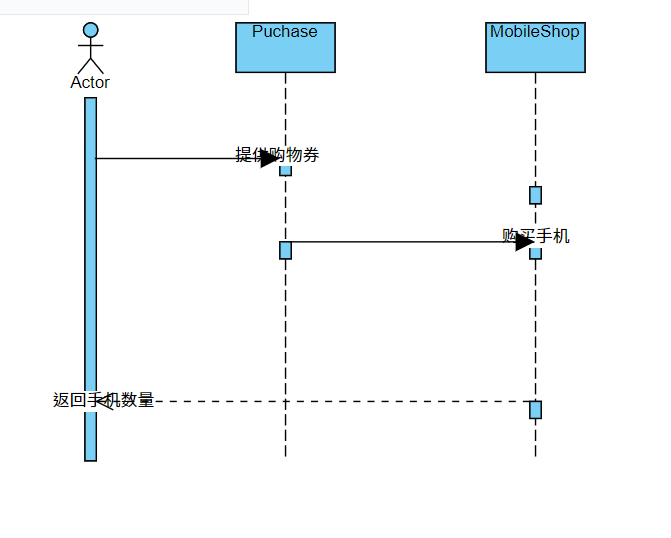


### （3）UML图

类图：



时序图：



### （4）源代码：

package com.company;

class MobileShop {

InnerPurchaseMoney purchaseMoney1; //用内部类定义变量

InnerPurchaseMoney purchaseMoney2;

private int mobileAmount; //手机的数量

MobileShop(){

purchaseMoney1=new InnerPurchaseMoney(20000); //实体化变量

purchaseMoney2=new InnerPurchaseMoney(10000);

}

void setMobileAmount(int m){

mobileAmount=m;

}

int getMobileAmount(){

return mobileAmount;

}

class InnerPurchaseMoney { //定义内部类InnerPurchaseMoney

int moneyValue;

InnerPurchaseMoney(int m){

moneyValue = m;

}

void buyMobile(){

if(moneyValue>=20000){

mobileAmount = mobileAmount-6;

System.out.println("用价值"+moneyValue+"的内部购物券买了6部手机");

}

else if(moneyValue<20000&&moneyValue>=10000){

mobileAmount = mobileAmount-3;

System.out.println("用价值"+moneyValue+"的内部购物券买了3部手机");

}

}

}

}

public class Main

{

public static void main(String arg[]){

MobileShop shop = new MobileShop(); //定义手机商店对象

shop.setMobileAmount(30);

System.out.println("手机专卖店目前有"+shop.getMobileAmount()+"部手机");

shop.purchaseMoney1.buyMobile(); //外嵌类对象调用内部类对象，再调用方法

shop.purchaseMoney2.buyMobile();

System.out.println("手机专卖店目前有"+shop.getMobileAmount()+"部手机");

System.out.println("20211120138薛凡豪");

}

## 2.检查危险品

### （1）问题分析：

1.本次实验主要练习异常类（即try-catch语句处理异常问题）

2.本次实验另外还用到了子类的继承问题。

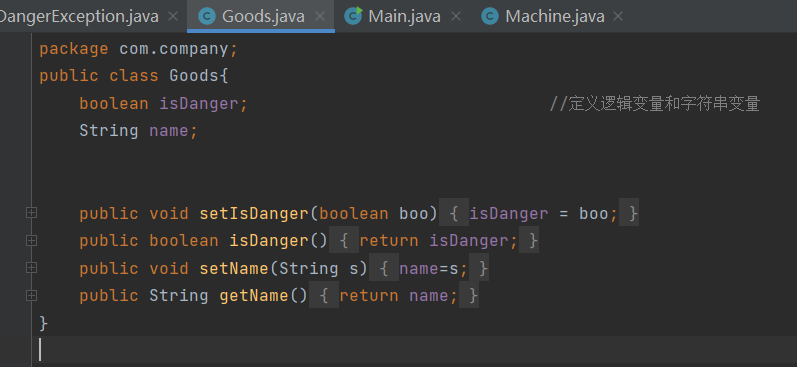
3.第一个是DangerException类，属于Exception的子类，可以创建异常对象并通过toShow方法输出。

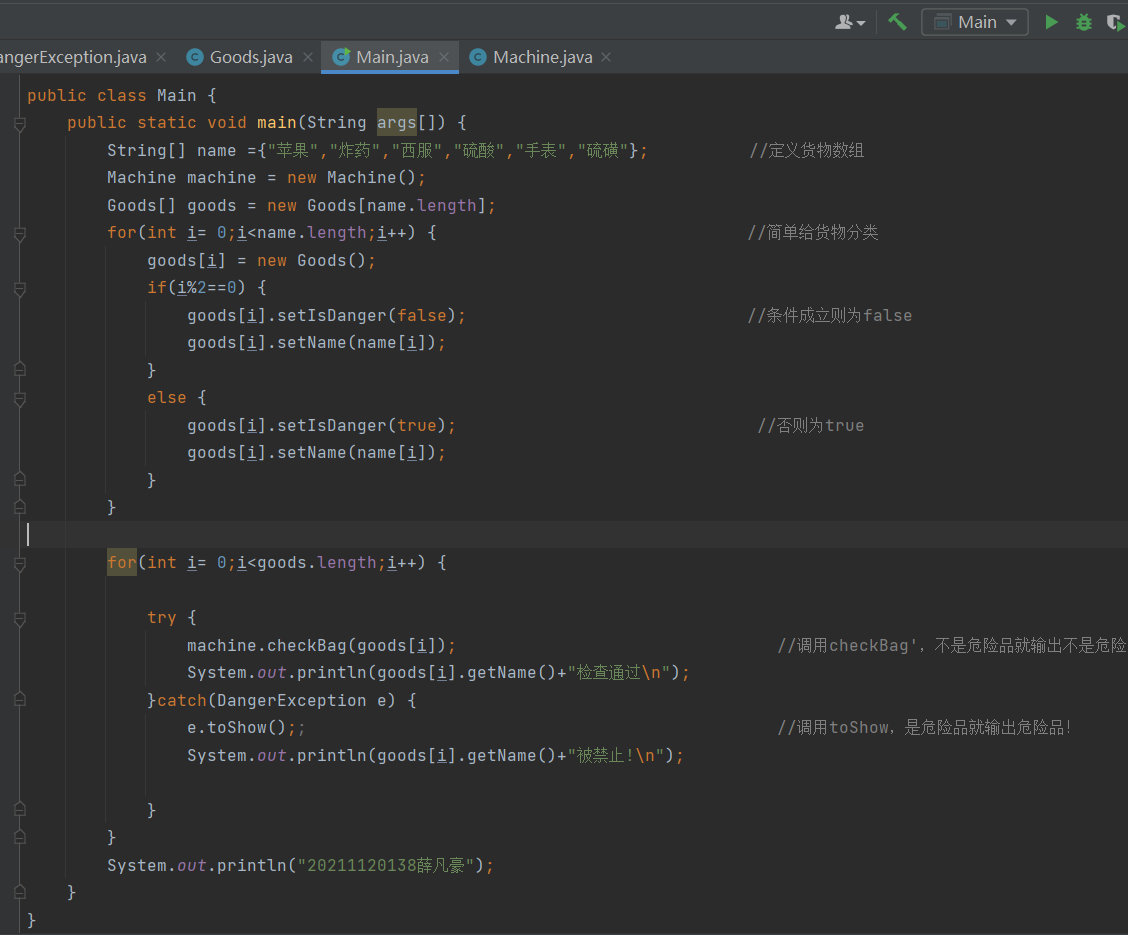
4.第二个类Machine，有checkBag方法，用于发现goods是不是危险品，如果发现goods是危险品则抛出异常。

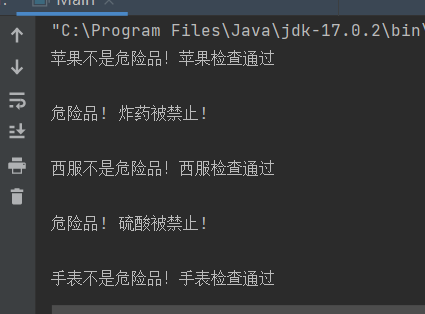
5.运行类则利用try-catch语句的try部分让Machine类的实例调用checkBag方法，如果发现危险品则用catch部分来处理。

### （2）程序的运行和结果



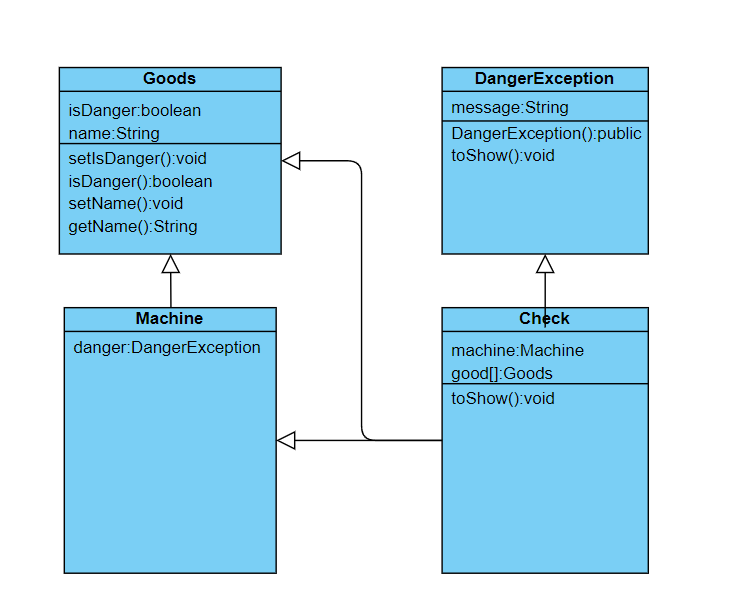


（3

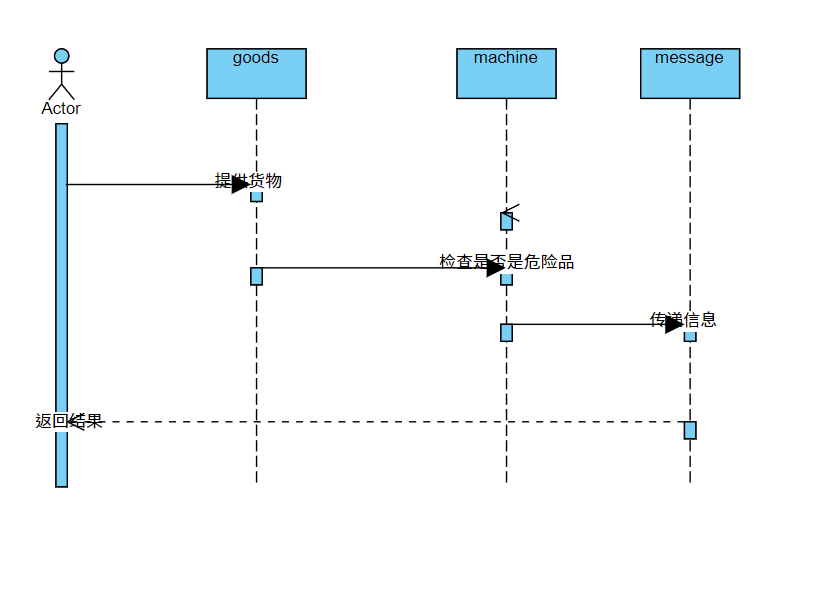


（3）UML图

类图：



时序图：



### （4）源代码

#### 1.DangerException.java

package com.company;

public class DangerException extends Exception{

String message;

public DangerException(){

message = "危险品!";

}

public void toShow(){

System.out.print(message+" "); //输出危险品！

}

}

2.Goods.java  
package com.company;

public class Goods{

boolean isDanger; //定义逻辑变量和字符串变量

String name;

public void setIsDanger(boolean boo){ //设置逻辑变量

isDanger = boo;

}

public boolean isDanger(){ //获取逻辑变量

return isDanger;

}

public void setName(String s){ //设置字符串变量

name=s;

}

public String getName() { //获取字符串变量

return name;

}

}

#### 3.Machine.java

package com.company;

public class Machine{

public void checkBag(Goods goods) throws DangerException { //参数为Goods类型

if(goods.isDanger()) { //判断货物的逻辑类型

DangerException danger=new DangerException();

throw(danger); //危险品就抛出

}

else{

System.out.print(goods.getName()+"不是危险品！"); //不是危险品就输出

}

}

}

#### 4.Main.java

package com.company;

public class Main {

public static void main(String args[]) {

String[] name ={"苹果","炸药","西服","硫酸","手表","硫磺"}; //定义货物数组

Machine machine = new Machine();

Goods[] goods = new Goods[name.length];

for(int i= 0;i<name.length;i++) { //简单给货物分类

goods[i] = new Goods();

if(i%2==0) {

goods[i].setIsDanger(false); //条件成立则为false

goods[i].setName(name[i]);

}

else {

goods[i].setIsDanger(true); //否则为true

goods[i].setName(name[i]);

}

}

for(int i= 0;i<goods.length;i++) {

try {

machine.checkBag(goods[i]); //调用checkBag'，不是危险品就输出不是危险品

System.out.println(goods[i].getName()+"检查通过\n");

}catch(DangerException e) {

e.toShow();; //调用toShow，是危险品就输出危险品！

System.out.println(goods[i].getName()+"被禁止!\n");

}

}

System.out.println("20211120138薛凡豪");

}

}

# 二、总结

1. 本次实验主要学习运用了内部类和异常类来解决问题
2. 主要使用了子类继承以及参数（数组）的传递等方法。

1. 利用java的多态性解决问题，会更方便更简洁。