**河南工业大学 Java程序设计实验 实验报告**

专业班级： 计科2101 学号：211040100127 姓名：翟向阳 指导老师： 赵晨阳 评分：

1. **实验题目**：实验（5）：接口、内部类、匿名类和异常类
2. **实验目的**：1. 熟悉Java 中的接口；2.理解内部类；3. 理解匿名类；4.理解异常类。
3. **实验要求：1. 每个题目，写出实验思路；2. 代码实现；3. 运行结果截图。**
4. **实验内容**：

1.编写应用程序，从命令行输入两个数据，求它们的商。要求程序能够捕获NumberFormatException异常和ArithmeticException异常。

2.设计一个训练场，这个训练场专门训练海豚，使它们能够跳舞dance（）和发出“呼呼”的声音cry（），要求用内部类实现该训练场。

3.创建一个名称为Vehicle的接口，在接口中添加两个方法start()和stop()；创建Bike和Bus两个类，它们实现Vehicle接口；最后，创建一个名称为InterfaceDemo的测试类，在该类的main()方法中创建Bike和Bus对象，并调用start()和stop()方法，要求运用接口回调。

4.假设Computable是一个接口，该接口含有一个compute()方法，参数自定义。然后，编写一个用于计算的测试类，该程序有一个方法ToCompute（Computable c）方法，可以计算加法和乘法，要求测试类中运用匿名类实现功能。

5. 模拟火车站安检，编写java程序，要求如下：

(1)编写一个异常类goodsException，该类包括toPrint（）方法，输出“该物品是违禁品”；

(2)编写一个检测工具类Tool，该类中有checkGoods（Goods g）方法，当参数g是违禁品时，将抛出goodsException对象；

(3)编写一个Goods类，该类中包含如下方法：设置物品是否是违禁品的方法setIsDanger（boolean b）（默认否），判定物品是否是违禁品的方法isDanger（），如果是违禁品，返回ture，否则false；设置物品的名字的方法setName（String n）；返回物品名字的方法getName（）；

(4)编写测试类，创建一个Goods类对象“explosive”，并设置该物品是违禁品，然后模拟检测到该物品是违禁品。

1. **实验执行：**

1. 编写应用程序，从命令行输入两个数据，求它们的商。要求程序能够捕获NumberFormatException异常和ArithmeticException异常。

（1）思路

先输入两个String字符，在try块中将其转化为Int类型的，并计算其结果，如果发生异常，则在catch中抛出NumberFormatException这个异常（数字格式错误），如果算术异常，则抛出ArithmeticException异常。

（2）代码实现

Scanner scanner = new Scanner(System.in);

String a = scanner.next();

String b = scanner.next();

int x,y,z;

try{

x = Integer.parseInt(a);

y = Integer.parseInt(b);

z = x/y;

System.out.println(z);

} catch (NumberFormatException e) {

System.out.println("数字格式异常");

throw new RuntimeException(e);

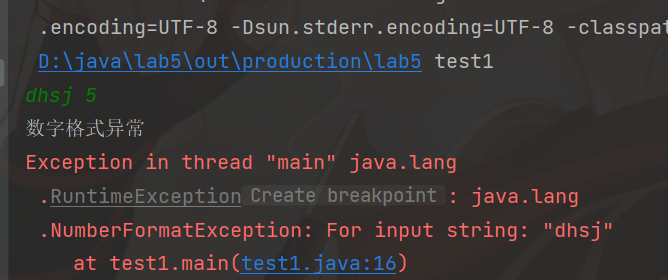
}

catch(ArithmeticException e){

System.out.println("算数异常(除数为零)");

}

（3）运行结果截图



2.设计一个训练场，这个训练场专门训练海豚，使它们能够跳舞dance（）和发出“呼呼”的声音cry（），要求用内部类实现该训练场。

（1）先声明一个haitun类，再在类内声明一个xunlian内部类，在内部类中实现dance（）和cry（）方法。在创建一个对象时，需要haitun.xunlian a = new haitun("花花",10).new xunlian() 来new一个对象。

（2）

public class test2 {

public static void main(String[] args) {

haitun.xunlian a = new haitun("花花",10).new xunlian();

a.dance();

a.cry();

}

}

class haitun{

String name;

int age;

haitun(String name,int age){

this.name=name;

this.age = age;

}

haitun(){}

class xunlian{

public void dance(){

System.out.println("我的名字是："+name);

System.out.println("我在高兴地跳舞");

}

public void cry(){

System.out.println("我的年龄是："+age);

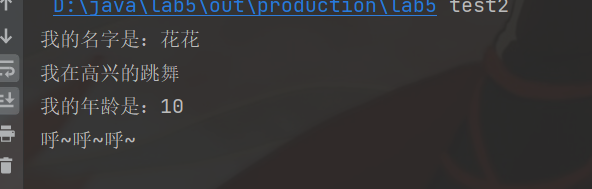
System.out.println("呼~呼~呼~");

}

}

}

（3）



3.创建一个名称为Vehicle的接口，在接口中添加两个方法start()和stop()；创建Bike和Bus两个类，它们实现Vehicle接口；最后，创建一个名称为InterfaceDemo的测试类，在该类的main()方法中创建Bike和Bus对象，并调用start()和stop()方法，要求运用接口回调。

（1）

声明Vehicle接口，在接口中声明两个方法start()和stop()，因为是abstract不用具体实现，在其子类中实现具体的方法就好，在对象创建的时候Vehicle bike = new Bike()， 利用上转型的方法来实现接口回调

（2）

public class test3 {

public static void main(String[] args) {

Vehicle bike = new Bike();

Vehicle bus = new Bus();

bike.start();

bike.stop();

bus.start();

bus.stop();

}

}

interface Vehicle{

void start();

void stop();

}

class Bike implements Vehicle{

@Override

public void start() {

System.out.println("自行车开始启动了");

}

@Override

public void stop() {

System.out.println("自行车停止了");

}

}

class Bus implements Vehicle{

@Override

public void start() {

System.out.println("公共汽车开始启动了");

}

@Override

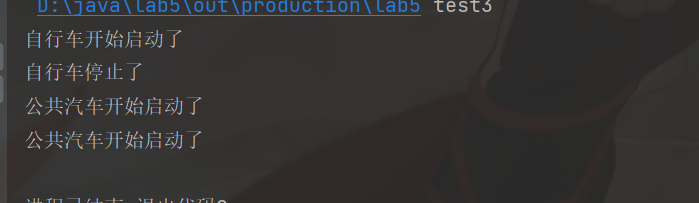
public void stop() {

System.out.println("公共汽车开始启动了");

}

}

（3）



4.假设Computable是一个接口，该接口含有一个compute()方法，参数自定义。然后，编写一个用于计算的测试类，该程序有一个方法ToCompute（Computable c）方法，可以计算加法和乘法，要求测试类中运用匿名类实现功能。

（1）

利用匿名类时，不直接创建对象，而是在传形参的时候利用（new 类名（）{}），通过实现该类中调用的接口函数来实现具体的方法，从而实现乘法和加法。

（2）

import java.util.Scanner;

public class test4 {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

Computabletest computabletest = new Computabletest();

computabletest.ToCompute(new Computable() {

@Override

public void compute() {

System.out.println("计算加法输入a,b的值:");

int a = scanner.nextInt();

int b = scanner.nextInt();

System.out.println(a+b);

}

});

computabletest.ToCompute(new Computable() {

@Override

public void compute() {

System.out.println("计算乘法输入a,b的值:");

int a = scanner.nextInt();

int b = scanner.nextInt();

System.out.println(a\*b);

}

});

}

}

interface Computable{

void compute();

}

class Computabletest {

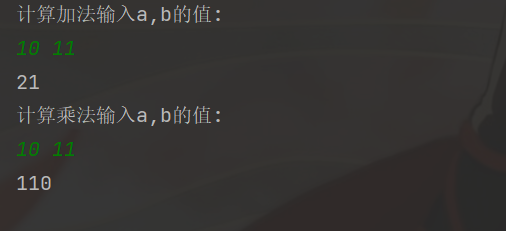
public void ToCompute(Computable c) {

c.compute();

}

}

（3）



5. 模拟火车站安检，编写java程序，要求如下：

(1)编写一个异常类goodsException，该类包括toPrint（）方法，输出“该物品是违禁品”；

(2)编写一个检测工具类Tool，该类中有checkGoods（Goods g）方法，当参数g是违禁品时，将抛出goodsException对象；

(3)编写一个Goods类，该类中包含如下方法：设置物品是否是违禁品的方法setIsDanger（boolean b）（默认否），判定物品是否是违禁品的方法isDanger（），如果是违禁品，返回ture，否则false；设置物品的名字的方法setName（String n）；返回物品名字的方法getName（）；

(4)编写测试类，创建一个Goods类对象“explosive”，并设置该物品是违禁品，然后模拟检测到该物品是违禁品。

（1）

该题刚开始按着题目要求写，但是到中间设置异常类的时候确实没有思路了，不知道该怎么来传递这个违禁品的消息，第一次直接把违禁品的消息在异常类中的构造函数中输出，在tool中的checkGoods方法中检测到违禁品，于是抛出异常，输出其实违禁品的信息，在mian函数中try catch中的catch又输出一次该异常信息，导致一共输出两次，最后参考课本中的方法，设置一个变量message，在构造函数中设置该message信息，并不输出。在main中的catch中将该信息通过toPrint输出。

（2）

public class test5 {

public static void main(String[] args) throws goodsException {

try {

Goods goods1 = new Goods();

goods1.setIsDanger(false);

goods1.setName("苹果");

Goods goods2 = new Goods();

goods2.setName("炸药");

goods2.setIsDanger(true);

Tool tool = new Tool();

System.out.println("判断第一个物品：");

tool.checkGoods(goods1);

System.out.println("判断第二个物品：");

tool.checkGoods(goods2);

} catch (goodsException e) {

e.toPrint();

throw new goodsException();

}

}

}

class goodsException extends Throwable {

private String message;

goodsException(){

message = "该物品为违禁品";

}

public void toPrint(){

System.out.println(message);

}

}

class Tool{

public void checkGoods(Goods g) throws goodsException {

if(g.IsDanger()){

throw new goodsException();//抛出方法异常

}

System.out.println("name:"+g.getName());

System.out.println("是否为违禁物:"+g.IsDanger());

}

}

class Goods{

private boolean danger = false;

private String name;

public void setIsDanger(boolean b){

this.danger = b;

}

public boolean IsDanger(){

return danger;

}

public void setName(String name) {

this.name = name;

}

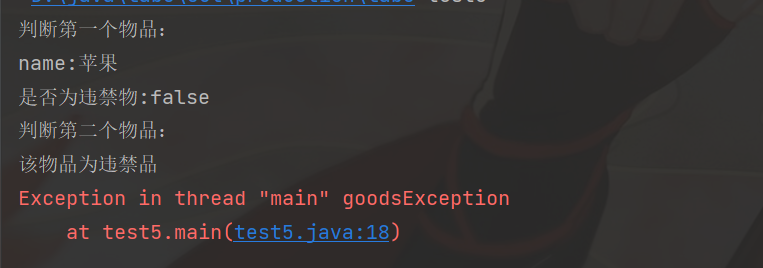
public String getName() {

return name;

}

}

（3）



1. **实验测试中遇到的问题和自己的解决方案**

在内部类的声明的时候，忘记怎么声明才是正确的了，看课本才想起来

类名.内部类名 a =new 类名（）.内部类名（）

最后一个实验中，中间设置异常类的时候确实没有思路了，不知道该怎么来传递这个违禁品的消息，第一次直接把违禁品的消息在异常类中的构造函数中输出，在tool中的checkGoods方法中检测到违禁品，于是抛出异常，输出其实违禁品的信息，在mian函数中try catch中的catch又输出一次该异常信息，导致一共输出两次，最后参考课本中的方法，设置一个变量message，在构造函数中设置该message信息，并不输出。在main中的catch中将该信息通过toPrint输出，问题得到解决，将异常抛出并且输出其是违禁品的信息。