1. 从命令行得到5个整数，放入一整型数组，然后打印输出，要求：如果输入数据不为整数，要捕获Integer.parseInt()产生的异常，显示“请输入整数”，捕获输入参数不足5个的异常(数组越界)，显示“请输入至少5个整数”。

关键代码：

public class IntegerError {

    public static void main(String[] args) {

        int[] scan = new int[5];

        try {

        for(int i = 0; i < 5; i++) {

            scan[i] = Integer.parseInt(args[i]);

        }

        for(int i = 0; i < 5; i++) {

            System.out.println((i + 1) + "number is " + scan[i]);

        }

    }

        catch(NumberFormatException nfe) {

            System.out.println("pls enter integer!");

        }

        catch(ArrayIndexOutOfBoundsException aio) {

            System.out.println("pls enter at least five integers!");

        }

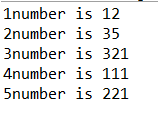
    }

}

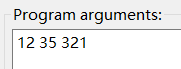
运行测试：

正确输入五个整数，并打印；



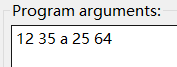


异常测试1.输入参数不足五个整数 throw ArrayIndexOutOfBoundsException





异常测试2.输入错误参数，throw NumberFormatException





2．自定义类Sanj，其中有成员 x,y,z,作为三边长，构造方法Sanj(a,b,c)分别给x,y,z赋值,方法求面积getArea和显示三角形信息(三个边长)showInfo，这2个方法中当三条边不能构成一个三角形时要抛出自定义异常NotSanjiaoException，否则显示正确信息。在另外一个类中的主方法中构造一个Sanj对象(三边为命令行输入的三个整数)，显示三角形信息和面积，要求捕获异常。

关键代码：

class NotSanjiaoException extends Exception{

    NotSanjiaoException(String s){

        super(s);

    }

    void showMess() {

        System.out.println("It's not a triangle!Error Code:1030;");

    }

}

/\*\*

 \* @author 96432

 \*

 \*/

class Sanj{

    private int x;

    private int y;

    private int z;

    Sanj(){}

    Sanj(int x, int y, int z){

        this.x = x;

        this.y = y;

        this.z = z;

    }

    double getArea()throws NotSanjiaoException {

        if((x <= 0) || (y <= 0) || (z <= 0)) {

            throw new NotSanjiaoException("Error!");

        }

        if(!(x + y > z && x + z > y && y + z > x)) {

            throw new NotSanjiaoException("Error!");

        }

        return 0.25 \* Math.sqrt((x + y + z) \* (x + y - z) \* (x + z - y) \* (y + z - x));

    }

    void showInfo() throws NotSanjiaoException{

        if((x <= 0) || (y <= 0) || (z <= 0)) {

            throw new NotSanjiaoException("Error!");

        }

        if(!(x + y > z && x + z > y && y + z > x)) {

            throw new NotSanjiaoException("Error!");

        }

        System.out.println("length of triangle:" + this.x);

        System.out.println("width of triangle:" + this.y);

        System.out.println("height of triangle:" + this.z);

    }

}

public class TriangleError {

    public static void main(String[] args) {

        Sanj sanj = new Sanj(Integer.parseInt(args[0]), Integer.parseInt(args[1]), Integer.parseInt(args[2]));

        double temp;

        try {

            sanj.showInfo();

            temp = sanj.getArea();

            System.out.print("the area is " + temp);

        }

        catch(NotSanjiaoException nsj) {

            nsj.showMess();

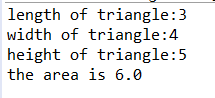
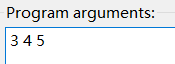
        }

    }

}

运行结果：

正确运行三角形面积计算输出；



捕获异常：





书上习题（1）：

编写贷款类Loan，若贷款总额，利率或年数小于等于0，抛出IllegalArgumentException异常；

关键代码：

class loan{

    private double all;

    private double r;

    private int year;

    loan(double a,double r, int y)throws IllegalArgumentException{

        if(a <= 0 || r <= 0 || y <= 0) {

            throw new IllegalArgumentException("Error Code:1030!");

        }

        this.all = a;

        this.r = r;

        this.year = y;

    }

    double getAll() {

        return this.all;

    }

    double getRate() {

        return this.r;

    }

    int getYear() {

        return this.year;

    }

}

public class LoanError {

    public static void main(String[] args) {

        try {

            loan lo = new loan(Double.parseDouble(args[0]), Double.parseDouble(args[1]), Integer.parseInt(args[2]));

            System.out.println("All of loan is " + lo.getAll());

            System.out.println("rate of loan is " + lo.getRate());

            System.out.println("year of loan is " + lo.getYear());

        }

        catch(IllegalArgumentException ill) {

            System.out.print(ill.getMessage());

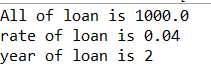
        }

    }

}

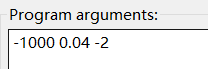
运行测试：

正确输入数据，正确输出数据



捕获异常测试：

由于有小于零的值所以抛出IllegalArgumentException异常



书上习题（2）：  
电气公司的电费异常问题；

关键代码：

import java.util.InputMismatchException;

import java.util.Scanner;

class electricErr extends Exception{

    electricErr(String s){

        super(s);

    }

    void showMess(int code) {

        switch(code) {

        case 1:{

            System.out.println("Error Code:1030!Calculate Error!");

            break;

        }

        case 2:{

            System.out.println("Error Code:1031!Negative Error!");

            break;

        }

            default:{

                System.out.println("Error Code:1032!Undefined Error!");

            }

        }

    }

}

class electric{

    private double ed;

    static private double price[] = {0.1, 0.3, 0.6};

    private double fee;

    electric(double ed){

        this.ed = ed;

        this.fee = 0;

    }

    electric(){this.ed = 0, this.fee = 0;}

    void calFee()throws electricErr {

        if(this.ed >= 0 && this.ed <= 200) {

            this.fee = electric.price[0] \* this.ed;

        }

        else if(200 < this.ed && this.ed <= 500) {

            this.fee = electric.price[1] \* this.ed;

        }

        else if(this.ed > 500){

            this.fee = electric.price[2] \* this.ed;

        }

        else {

            throw new electricErr("Negative Error!");

        }

    }

    double getFee()throws electricErr{

        this.calFee();

        return this.fee;

    }

    double getEd() {

        return this.ed;

    }

}

public class ElectricError {

    public static void main(String[] args) {

        Scanner in = new Scanner(System.in);

        boolean flag = true;

        electric elc;

        StringBuffer temp = new StringBuffer();

        do {

            System.out.println("Pls enter the electrical degrees number for month:");

            try {

                elc = new electric(in.nextDouble());

                System.out.println("the electrical degrees is " + elc.getEd());

                System.out.println("the electrical fee is " + elc.getFee());

                flag = false;

            }

            catch(InputMismatchException ime) {

                System.out.println("Input Error!Pls enter a positive number(0 ~ .....).");

                temp.append(in.nextLine());

            }

            catch(electricErr elec) {

                elec.showMess(1);

                elec.showMess(2);

            }

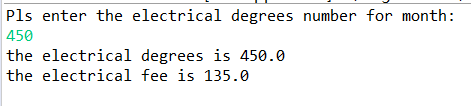
        }while(flag);

    }

}

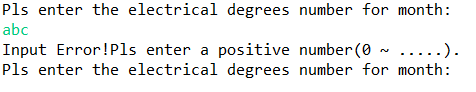
运行测试：

正确输入数据，正确计算出最终费用；



异常测试：

若输入为非数，则重复循环，直到输入符合预期数据要求为止；



若输入的数据是一个数，但这个数小于0那么这个数也是不合要求的，因为电费不可能为负，所以计算电费时抛出异常electricErr,输出错误信息，并要求重新输入，直到数据符合要求；

