《面向对象程序设计与训练》实验报告

信息 学院 计算机科学与技术 专业 2019 级

实验时间 2020 年 10 月 20 日

姓名 李志伟 学号 20191060063 分工 第2、3题

姓名 郑广敏 学号 20191130038 分工 第5题

姓名 程奕飞 学号 20191060220 分工 第4题

实验名称 类的构成和对象的使用

实验成绩

一、实验目的

1. 理解Java语言是如何体现面向对象编程基本思想，
2. 了解类的封装方法，以及如何创建类和对象，
3. 了解成员变量和成员方法的特性。
4. 了解类的继承性和多态性的作用。

二、实验仪器设备及软件

Windows2000 或XP，JDK1.6与Jcreator4.0

三、实验方案

2.

创建Triangle类，定义边变量、状态变量status和构造方法以及判断和计算方法。

创建YZT类，定义半径和高变量、状态变量，构造方法，判断及计算方法。

创建callClass类导入name包并使用Triangle类和YZT类。

其中判断错误，赋给status属性一个错误值

3.

使用Math库的静态方法计算。

创建Point类，将Point实例化后存入其中并输出结果。

4.

创建Person类，定义姓名、性别、年龄变量以及状态变量status，并声明构造方法。

创建Student类继承Person类，加入学号变量并声明构造方法。

创建Undergraduate类继承Student类，加入专业变量并声明构造方法。

创建callClass4类进行实例化和使用。

其中判断错误，赋给status一个错误值

5.

1. 用for语句逐个输出第5到第10个字符即可
2. 初始化一个可显字符的哈希表，初值为0，遍历一趟字符串中的每个字符（忽略英文大小写），每个字符出现就在对应的哈希表中加1，哈希函数为char-32；
3. 根据哈希表找到最大值的位置，并将哈希值还原为字符
4. 使用replace成员函数进行替换
5. 使用for语句逆序逐个输出字符

四、实验步骤

2.

package Lizhiwei;

public class YZT {

private float radius,height;

public int status=1;

public float getRadius() {

return radius;

}

public void setRadius(float radius) {

this.radius = radius;

}

public float getHeight() {

return height;

}

public void setHeight(float height) {

this.height = height;

}

public YZT() {

super();

this.radius = 1.0f;

this.height = 1.0f;

}

public YZT(float radius, float height) {

super();

if(radius>0.0f && height>0.0f) {

this.radius = radius;

this.height = height;

}

else {

this.status=-1;

}

}

public float calTJ() {

return (float)(Math.***PI***\*this.radius\*this.radius\*this.height);

}

public float calBMJ() {

return (float)(Math.***PI***\*this.radius\*this.radius\*2+Math.***PI***\*2\*this.radius\*this.height);

}

}

package Lizhiwei;

public class Triangle {

private float sideA,sideB,sideC;

public int status=1;

public Triangle() {

super();

this.sideA = 1.0f;

this.sideB = 1.0f;

this.sideC = 1.0f;

}

public Triangle(float sideA, float sideB, float sideC) {

super();

if(sideA>0.0f && sideB>0.0f && sideC>0.0f) {

if(sideA+sideB>sideC && sideC+sideB>sideA && sideA+sideC>sideB) {

this.sideA = sideA;

this.sideB = sideB;

this.sideC = sideC;

return;

}

}

this.status=-1;

return;

}

public float getArea() {

return 1.0f/4.0f\*(float)Math.*sqrt*((this.sideA+this.sideB+this.sideC)\*(this.sideB+this.sideC-this.sideA)\*(this.sideA+this.sideB-this.sideC)\*(this.sideA+this.sideC-this.sideB));

}

public float getLength() {

return this.sideA+this.sideB+this.sideC;

}

}

import Lizhiwei.Triangle;

import Lizhiwei.YZT;

public class callClass {

public static void main(String[] args) {

//默认三角形

Triangle a1=new Triangle();

System.out.printf("面积：%.2f 周长：%.2f\n",a1.getArea(),a1.getLength());

//错误

Triangle a2=new Triangle(1.0f,2.0f,5.0f);

if(a2.status==-1) System.out.println("error!");

//自定义三角形

Triangle a3=new Triangle(25,20,37.5f);

System.out.printf("面积：%.2f 周长：%.2f\n",a3.getArea(),a3.getLength());

//默认圆柱体

YZT b1=new YZT();

System.out.printf("体积：%.2f 表面积：%.2f\n",b1.calTJ(),b1.calBMJ());

//错误

YZT b2=new YZT(2.0f,-10.5f);

if(b2.status==-1) System.out.println("error!");

//自定义圆柱体

YZT b3=new YZT(12.5f,11.0f);

System.out.printf("体积：%.2f 表面积：%.2f\n",b3.calTJ(),b3.calBMJ());

}

}

3.

public class work3 {

public static void main(String[] args) {

System.***out***.println(Math.*sin*(30)+Math.*tan*(50)+Math.*sqrt*(10));

}

}

public class Point {

private double x,y;

static int *cnt*=0;

public double getX() {

return x;

}

public void setX(double x) {

this.x = x;

}

public double getY() {

return y;

}

public void setY(double y) {

this.y = y;

}

public Point() {

super();

// **TODO** Auto-generated constructor stub

*cnt*++;

this.x=*cnt*;

this.y=*cnt*\**cnt*;

}

}

public class work3\_2 {

public static void main(String[] args) {

Point[] point=new Point[11];

for(int i=1;i<=10;i++) {

point[i]=new Point();

System.***out***.printf("第%d个点的坐标(%.2f,%.2f),当前已有%d个点\n",i,point[i].getX(),point[i].getY(),point[i].*cnt*);

}

}

}

4.

package Chengyifei;

public class Person {

private String name;

protected String sex;

int age;

int status=1;

public Person() {

super();

this.name =new String();;

this.sex =new String();

this.age = 0;

}

public Person(String name, String sex, int age) {

super();

this.name =new String(name);

if(sex!="男" && sex!="女") {

this.status=-1;

return;

}

this.sex = new String(sex);

if(age<=0 || age>=130) {

this.status=-2;

return;

}

this.age = age;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getSex() {

return sex;

}

public void setSex(String sex) {

this.sex = sex;

}

public int getAge() {

return age;

}

public void setAge(int age) {

this.age = age;

}

public void outError() {

if(this.status==-1) {

System.***out***.println("性别输入有误");

}else if(this.status==-2) {

System.***out***.println("年龄输入有误");

}

}

public String toString() {

return "姓名："+this.name+"\n性别："+this.sex+"\n年龄："+this.age+"\n";

}

}

package Chengyifei;

public class Student extends Person {

String num;

public Student() {

super();

// **TODO** Auto-generated constructor stub

this.num="";

}

public Student(String name,String sex,int age,String num) {

super(name,sex,age);

this.num = num;

}

public String getNum() {

return num;

}

public void setNum(String num) {

this.num = num;

}

public String toString() {

return "姓名："+this.getName()+"\n性别："+this.sex+"\n年龄："+this.age+"\n学号："+this.num+"\n";

}

}

package Chengyifei;

public class Undergraduate extends Student{

String major;

public Undergraduate() {

super();

// **TODO** Auto-generated constructor stub

this.major="";

}

public Undergraduate(String name, String sex, int age, String num,String major) {

super(name, sex, age, num);

// **TODO** Auto-generated constructor stub

this.major=major;

}

public String getMajor() {

return major;

}

public void setMajor(String major) {

this.major = major;

}

public String toString() {

return "姓名："+this.getName()+"\n性别："+this.sex+"\n年龄："+this.age+"\n学号："+this.num+"\n专业："+this.major+"\n";

}

}

package Chengyifei;

public class Graduate extends Undergraduate{

String teacher;

public Graduate() {

super();

// **TODO** Auto-generated constructor stub

this.teacher="";

}

public Graduate(String name, String sex, int age, String num, String major,String teacher) {

super(name, sex, age, num, major);

// **TODO** Auto-generated constructor stub

this.teacher=teacher;

}

public String getTeacher() {

return teacher;

}

public void setTeacher(String teacher) {

this.teacher = teacher;

}

public String toString() {

return "姓名："+this.getName()+"\n性别："+this.sex+"\n年龄："+this.age+"\n学号："+this.num+"\n专业："+this.major+"\n导师："+this.teacher+"\n";

}

}

import Chengyifei.\*;

public class callClass4 {

public static void main(String[] args) {

Person a=new Person("郭靖","男",20);

System.***out***.println(a.toString());

Student b=new Student("黄蓉","女",17,"1263001");

System.***out***.println(b.toString());

Undergraduate c=new Undergraduate("韦小宝","男",300,"1892001","中国武术学");

c.outError();

}

}

5.

package zgm;

public class work5 {

public static void main(String[] args) {

String s="AbC123sdfg7fFCM987D0ddFFPAZvvvFFFggg43";

String a=s.toLowerCase();//无大写

int biao[]=new int[95];

//初始化biao

for(int i=0;i<95;i++) biao[i]=0;

int max=0,min=100;

char max\_c=0,min\_c=0;

//输出第5到第10个字符

for(int i=5;i<=10;i++) {

System.***out***.print(s.charAt(i));

}

System.***out***.println();

//计算每个字符出现个数

int len=s.length();

for(int i=0;i<len;i++) {

int j=(int)(a.charAt(i)-32);

biao[j]++;

//计算出现次数最多的字符及出现次数

if(max<biao[j]) {

max=biao[j];

max\_c=a.charAt(i);

}

if(min>biao[j]) {

min=biao[j];

min\_c=a.charAt(i);

}

}

//输出出现次数最多最少的字符及他们的出现次数

System.***out***.println("出现最多的字符："+max\_c+" 出现了"+max+"次");

System.***out***.println("出现最少的字符："+min\_c+" 出现了"+min+"次");

//替换字符

s=s.replace(max\_c, '#');

s=s.replace((char)(max\_c-32), '#');

System.***out***.println(s);

for(int i=len-1;i>=0;i--) {

System.***out***.print(s.charAt(i));

}

}

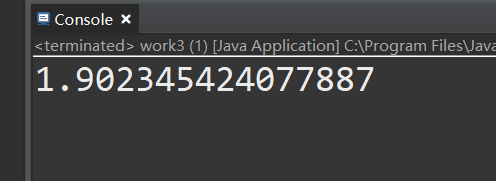
}

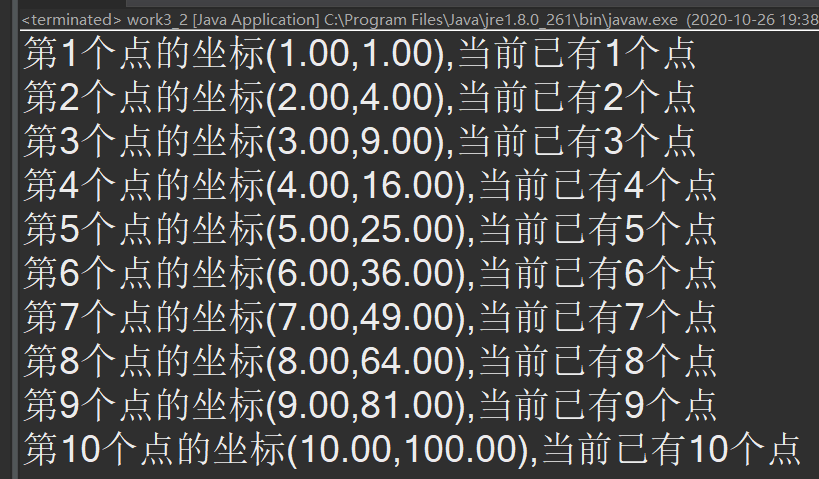
五、实验结果及分析

2.

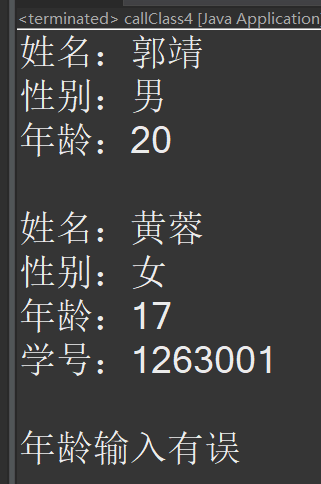


3.

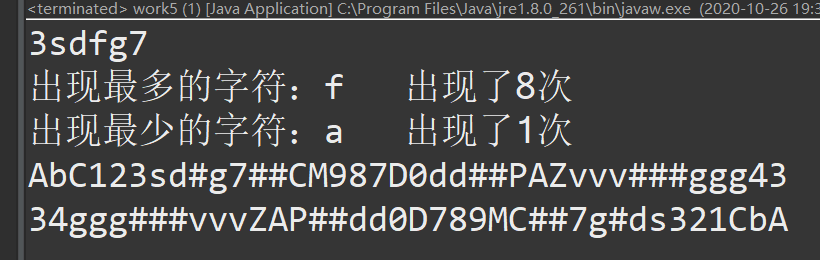




4.



5.



六、实验总结及体会

1.为维护代码通用性，对于不合法对象的创建，不输出错误信息，而是在类中设置一个status属性，不同的值代表成功或者不同的失败状态。

2.第五题不区分英文字母的大小写，应先将原字符串全部换成小写或大写存在一个新的字符串中，查询判断等操作使用新字符串，而修改输出等操作使用原字符串。

3.通过第四题体会到了不同访问权限的区别

（1）public：public表明该数据成员、成员函数是对所有用户开放的，所有用户都可以直接进行调用  
 （2）private：private表示私有，私有的意思就是除了class自己之外，任何人都不可以直接使用，私有财产神圣不可侵犯嘛，即便是子女，朋友，都不可以使用。  
 （3）protected：protected对于子女、朋友来说，就是public的，可以自由使用，没有任何限制，而对于其他的外部class，protected就变成private。

七、教师评语