高级语言程序设计I实验报告

实验三 类和对象

班 级： 22软件工程

姓 名： 王玉升

学 号： 22206091012

成 绩：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 实验名称 | | **实验三 类和对象** | | | |
| 实验日期 | | | 2023.10.20 | 实验课时 | 4 |
| 实  验  目  的  及  要  求 | （给出本次实验内容及要求的具体描述）   1. 实验目的：   1．掌握类的定义和对象的创建，理解类来封装对象的属性和行为  2．掌握对象的引用及参数传递  3．掌握类变量与实例变量，以及类方法与实验实例方法的区别  4．掌握方法的重载  5.掌握包的定义和类的导入  6．掌握各种访问权限区别及正确使用   1. 实验要求： 2. 安JDK及Ｅclipse开发环境，并使用Ｅclipse开发平台进行java语言程序的编辑、编译和运行。 3. 在D盘或E盘以自己的学号末尾四位数字为名创建文件夹，并在其下面为每个实验项目创建子文件夹，程序相关文件存储在相应的文件夹中（如学号末尾是”1023”，实验三则应创建目录结构“E:\1023\sy03”，实验三程序文件放在“…\sy03”）。 4. 实验报告中附上程序清单和程序运行结果截图（控制台中输出结果后用自己的真实姓名注释）。 5. 每题程序代码中至少有两处用自己的真实姓名进行注释   5．实验结束后进行总结。 | | | | |
| 实  验  环  境 | （列出本次实验所使用的平台和相关软件）  地点：南6207  开发环境：JDK1.8、Eclipse | | | | |
| 实  验  过  程  及  实  验  结  果 | 1. **编写一个java应用程序，该程序中有两个类：Tank和Fight，Tank类中包括一个double变量speed，刻画速度，一个int变量bulletAmount刻画坦克炮弹数量。有四个没有返回值的方法，分别表示加速、减速，计数坦克炮弹数及一个开炮行为。在主类Fight的main方法中用Tank类创建坦克。程序代码如下，请补充完善（注释不能删除）：**   **/\*本题重点是掌握类的定义、对象创建、方法调用、参数传递\*/**   * 1. 程序清单：   **Tank.java**  public class Tank {  double speed = 0; //【代码1】声明double型变量speed,刻画速度  int bulletAmount = 0; //【代码2】声明int型变量bulletAmount,刻画炮弹数量  void speedUp(int s) { //加速  speed += s; //【代码3】将速度speed增加s  }  void speedDown(int d) { //减速  if(speed-d>=0)  speed -= d;//【代码4】将速度speed减小d  else  speed = 0;  }  void setBulletAmount(int m) { //设置炮弹数量  bulletAmount = m;  }  int getBulletAmount() { //获取炮弹数  return bulletAmount;  }  double getSpeed() { //获取速度  return speed;  }  void fire() { //开火  if(bulletAmount>=1){  bulletAmount -= 1;//【代码5】炮弹数减少一个  System.out.println("打出一发炮弹");  }  else {  System.out.println("没有炮弹了,无法开火");  }  }  }**Fight.java**  public class Fight {  public static void main(String args[]) {  Tank tank1,tank2;  tank1 = new Tank();//创建对象tank1  tank2 = new Tank();//【代码6】创建对象tank2  tank1.setBulletAmount(10);  tank2.setBulletAmount(10);  System.out.println("tank1的炮弹数量："+tank1.getBulletAmount());  System.out.println("tank2的炮弹数量："+tank2.getBulletAmount());  tank1.speedUp(80);  tank2.speedUp(90);  System.out.println("tank1目前的速度："+tank1.getSpeed());  System.out.println("tank2目前的速度："+tank2.getSpeed());  tank1.speedDown(12);//【代码7】 tank1的速度减小15  tank2.speedDown(30);// 【代码8】tank2的速度减小30  System.out.println("tank1目前的速度："+tank1.getSpeed());  System.out.println("tank2目前的速度："+tank2.getSpeed());  System.out.println("王玉升驾驶tank1开火：");  tank1.fire();  System.out.println("王玉升驾驶tank2开火：");  tank2.fire();  tank2.fire();  System.out.println("tank1的炮弹数量：" + tank1.getBulletAmount());// 【代码9】获取tank1的炮弹数量  System.out.println("tank2的炮弹数量："+ tank2.getBulletAmount()); //【代码10】获取tank2的炮弹数量  }  }   * 1. 运行程序和运行结果（截图）：      1. **编写一个java应用程序，该程序中有两个类：Village和Land，Village类中包括一个int变量waterAmount，刻画水井的水量，水井为多个村庄共有，一个int变量peopleNumber，刻画村庄的人数，一个String变量name，刻画村庄的名字，有两个构造方法；主类是Land，创建了两个村庄，在同一口井中饮水。将下面程序代码补充完善（注释不能删除）：**   **/\*本题重点是掌握构造方法的定义、实例成员与类成员的区别与使用\*/**   * 1. 程序清单：   **Village.java:**  public class Village {  static int waterAmount = 0;//【代码1】定义int 型变量waterAmount模拟水井的水量，水井为多个村庄共享  int peopleNumber; //村庄的人数  String name; //村庄的名字  public Village() {  //【代码2】添加无参没有语句的构造方法  }  Village(String name) {  setName(name); //【代码3】参数name作为村庄的名字  }  void setName (String name) { //设置村庄的名字  this.name=name;  }  static void setWaterAmount(int m) { //设置水井的水量  if(m>=0)  waterAmount = m;  }  void drinkWater(int n){ //从水井饮水  if( waterAmount-n>=0) {  waterAmount -= n;//【代码4】重新计算水井的水量  System.out.println(name + "喝了" + n + "升水");  }  else  waterAmount = 0;  }  static int lookWaterAmount() { //查看水井的水量  return waterAmount;  }  void setPeopleNumber(int n) { //【代码5】设置村庄的人数  this.peopleNumber = n;  }  int getPeopleNumber() { //【代码6】获取村庄的人数  return this.peopleNumber;  }  }**Land.java**  public class Land {  public static void main(String args[]) {  Village.setWaterAmount(200); //【代码7】用类名调用setWaterAmount(int m),并向参数传值200  int leftWater =Village.waterAmount; //【代码8】用Village类的类名查看水量  System.out.println("水井中有 "+leftWater+" 升水");  Village zhaoZhuang,maJiaHeZhi;  zhaoZhuang = new Village();  zhaoZhuang. setName("赵庄");  maJiaHeZhi = new Village("马家河子");  zhaoZhuang.setPeopleNumber(80);  maJiaHeZhi.setPeopleNumber(120);  zhaoZhuang.drinkWater(50);//【代码9】zhaoZhuang调用drinkWater(int n),并向参数传值50  leftWater = maJiaHeZhi.lookWaterAmount();//【代码10】maJiaHeZhi调用lookWaterAmount()方法，能否表示为leftWater =Village.waterAmount  String name=maJiaHeZhi.name;  System.out.println(name+"发现水井中有 "+leftWater+" 升水");  maJiaHeZhi.drinkWater(100);  leftWater = zhaoZhuang.lookWaterAmount();//【代码11】zhaoZhuang调用lookWaterAmount()方法  name=zhaoZhuang.name;  System.out.println(name+"发现水井中有 "+leftWater+" 升水");  int peopleNumber = zhaoZhuang.getPeopleNumber(); //能否表示为int peopleNumber =Village.peopleNumber  System.out.println("赵庄的人口:"+peopleNumber);  peopleNumber = maJiaHeZhi.getPeopleNumber();  System.out.println("马家河子的人口:"+peopleNumber);  }  }   * 1. 运行程序和运行结果（截图）：      1. **编程求形状的面积，有以下几个类：**   **Triangle ：三角形类，有side1、side2、side3三个成员变量，表示三角形三边，有构造方法（有三个参数，初始三个成员变量），有getArea()方法，求面积。**  **Lader: 等腰梯形类，有above、bottom、height三个成员变量，表示梯形上底、下底和高，有构造方法（有三个参数，初始三个成员变量），有getArea()方法，求面积。**  **Circle ：圆类，有r成员变量，表示圆的半径，有构造方法（有一个参数，初始化半径），有getArea()方法，求面积。**  **Student：学生类，有三个重载方法computeArea（有一个参数，为上面三个类的对象），分别求相应形状的面积。**  **Sy3\_3：主类，分别创建Triangle、Lader和Circle 三个对象，并通过Student类的对象调用computeArea方法分别计算并显示三角形、等腰梯形和圆的面积。**  **/\*本题重点是掌握重载方法的使用\*/**   |  | | --- | | Student | |  | | computeArea ( Triangle t):double  computeArea ( Lader t):double  computeArea ( Circle  t):double |  |  | | --- | | Triangle | | side1:double  side2:double  side3:double | | Triangle (double s1,double s2,double s3):void  getArea():double |  |  | | --- | | Lader | | above:double  bottom :double  height:double | | Lader (double a,double b,double h):void  getArea():double |  |  | | --- | | Circle | | r:double | | Circle (double r):void  getArea():double |   图3-1 类的UML图   * 1. 程序清单：   public class Circle {  double r;        Circle(){  r = 3.0;  }    Scanner sc = new Scanner(System.in);        double getArea() {  double Area = 3.14 \* r \* r;  return Area;  }  }  public class Lader {  double above;  double bottom;  double height;      Lader(){  above = 3.0;  bottom = 4.0;  height = 5.0;  }      double getArea(){  double Area = (above + bottom) \* height / 2;  return Area;    }  }  public class Triangle {  double side1;  double side2;  double side3;  Triangle(){  side1 = 3.0;  side2 = 4.0;  side3 = 5.0;    }    double getArea(){ //面积  double sinC = ( side1\*side1 + side2\*side2 - side3\*side3) / ( 2 \* side1 \* side2);  double Area = side1 \* side2 \* sinC / 2;  return Area;    }  }  public class Student {  public void computeArea(Triangle t) { //三角形  double a = t.getArea();  System.out.println(a);    }  public void computeArea(Lader t) { //梯形  double a = t.getArea();  System.out.println(a);    }  public void computeArea(Circle t) { //圆  double a = t.getArea();  System.out.println(a);    }              public static void main(String args[]) {  Triangle triangle = new Triangle();  Lader lader = new Lader();  Circle circle = new Circle();    Student Wang = new Student();    Wang.computeArea(triangle);  Wang.computeArea(lader);  Wang.computeArea(circle);      }  }   * 1. 运行程序和运行结果（截图）：      1. **编写程序模拟一个家族成员的姓名，姓名由两部分构成：姓氏和名字。编写一个FamilyPerson类，该类有一个静态的String型成员变量surname，用于存储姓氏、一个实例的String型成员变量name，用于存储名字。在主类Members的main方法中首先用类名访问surname，并为surname赋值，然后FamilyPerson创建3个对象：father,sonOne和sonTwo,并分别给他们的name属性赋值，然后输出（注释不能删除）：**   **/\*本题重点是掌握构实例成员与类成员的区别与使用\*/**  （1） 程序清单：  **FamilyPerson.java**  public class FamilyPerson {  static String Surname;  String name;  public static void setSurname(String Surname){  FamilyPerson.Surname = Surname; //【代码1】将参数surname设置成姓氏  }  public void setName(String name) {  this.name = name; //【代码2】将参数name设置成名字  }  }**Members.java**  public class Members {  public static void main(String args[]) {  FamilyPerson.setSurname("李");//【代码3】将类FamilyPerson中surname变量赋值:"李"  FamilyPerson father,sonOne,sonTwo;  father = new FamilyPerson();//【代码4】创建father对象  sonOne = new FamilyPerson();//【代码5】创建sonOne对象  sonTwo = new FamilyPerson();//【代码6】创建sonTwo对象  father.setName("向阳");//【代码7】将father名字设为"向阳"  sonOne.setName("抗日");//【代码8】将sonOne名字设为"抗日"  sonTwo.setName("抗日");//【代码9】将sonTwo名字设为"抗战"  System.out.println("父亲:"+ FamilyPerson.Surname + father.name); //【代码10】输出father姓名  System.out.println("大儿子:"+ FamilyPerson.Surname + sonOne.name); //【代码11】输出sonOne姓名  System.out.println("二儿子:"+ FamilyPerson.Surname + sonTwo.name); //【代码12】输出sonTwo姓名  FamilyPerson.setSurname("张");// 【代码13】将father姓氏改为"张"  System.out.println("父亲:"+ FamilyPerson.Surname + father.name); //【代码14】输出father姓名  System.out.println("大儿子:"+ FamilyPerson.Surname + sonOne.name); //【代码15】输出sonOne姓名  System.out.println("二儿子:"+ FamilyPerson.Surname + sonTwo.name); //【代码16】输出sonTwo姓名  }  }  （2） 运行程序和运行结果（截图）：     1. **编写一个java应用程序，该程序中有两个类：SquareEquation和SunRise，SquareEquation类的包名为tom.jiafei，表示一元二次方程，有一个构造方法、一个求根方法getRoots、一个设置方法系数setCoefficient；主类是SunRise，无包名。将下面程序代码补充完善（注释不能删除）：**   **/\*本题重点是掌握构包的定义、类的引用与访问权限正确使用\*/**   * 1. 程序清单：   **SquareEquation.java:**  【代码1】 //【代码1】声明包名  public class SquareEquation {  double a,b,c; //表示方程三个系数  double root1,root2; //表示方程的两个实根  boolean boo; //表示是否为一元二次方程  SquareEquation(double a, double b, double c){//【代码2】有三个参数的构造方法方法头  this.a=a;  this.b=b;  this.c=c;  if(a!=0)  boo=true;  else  boo=false;  }  public void getRoots(){ //【代码3】名为getRoots方法的方法头,功能求方程两实根  if(boo) {  System.out.println("是一元2次方程");  double disk=b\*b-4\*a\*c;  if(disk>=0) {  root1=(-b+Math.sqrt(disk))/(2\*a);  root2=(-b-Math.sqrt(disk))/(2\*a);  System.out.printf("wys 说方程的根:%f,%f\n",root1,root2);  }  else {  System.out.printf("wys 说方程没有实根\n");  }  }  else {  System.out.println("不是一元2次方程");  }  }  public void setCoefficient(double a, double b, double c){//【代码4】名为setCoefficient方法的方法头，功能设置方程三系数  this.a=a;  this.b=b;  this.c=c;  if(a!=0)  boo=true;  else  boo=false;  }  }**SunRise.java:**  【代码5】 //【代码5】导入SquareEquation类  public class SunRise {  public static void main(String args[]) {  SquareEquation equation=new SquareEquation(4,5,1);//创建系数是4、5、1的一元二次方程  equation.getRoots();//【代码6】求方程的根  equation.setCoefficient(-3,4,5);//重新设置方程的系数  equation.getRoots();  }   * 1. }运行程序和运行结果（截图）：      1. **用类描述计算机中CPU的速度和硬盘的容量，程序由四个类构成：**   **CPU：有int型speed成员变量，表示CPU速度，用setSpeed (int m)方法设置CPU速度speed，getSpeed ()方法返回CPU速度speed。**  **HardDisk：有int型amount成员变量，表示硬盘容量，用setAmount (int m)方法设置硬盘容量amount，getAmount ()方法返回硬盘容量amount。**  **PC：该类与CPU类和HardDisk关联，有CPU型cpu成员变量和HardDisk型HD成员变量，用setCPU (CPU c)方法设置cpu值（即c赋值给cpu），setHardDisk (HardDisk h)方法设置HD值（即h赋值给HD）,show()方法显示CPU的速度和硬盘的容量。**  **Sy3\_6：主类，main方法完成以下功能：**  **（1）创建CPU对象cpu，并将cpu的速度设为2200。**  **（2）创建HardDisk对象disk，并将disk的容量设为500。**  **（3）创建PC对象pc，并将cpu和disk设置为pc的相应变量的值。**  **（4）显示pc中CPU的速度和硬盘的容量。**   |  | | --- | | PC | | Cpu: CPU  HD: HardDisk | | setCPU (CPU c):void  setHardDisk (HardDisk h) :void  show():void |  |  | | --- | | HardDisk | | amount: int | | setAmount (int m):void  getAmount ():int |  |  | | --- | | CPU | | peed: int | | setSpeed (int m):void  getSpeed () :int |   图3-2 类的UML图   * 1. 程序清单：   public class CPU {  int speed; //CPU速度  public void setSpeed(int speed) {  this.speed = speed; //设置CPU速度  }  public int getSpeed() {  return speed; //获取CPU速度  }  }  public class HardDisk {  int amount; //储存容量  public void setAmount(int amount) {  this.amount = amount; //设置硬盘容量  }  public int getAmount() {  return amount; //获取硬盘容量  }  }  public class PC {  CPU Cpu;  HardDisk HD;  public void setCpu(CPU cpu) {  Cpu = cpu;  }  public void setHD(HardDisk HD) {  this.HD = HD;  }  public void show(){  System.out.println("wys说，CPU的速度是" + Cpu.speed + "硬盘的容量是" + HD.amount);  }  }  public class main {  public static void main(String[] args) {  CPU cpu = new CPU();  cpu.setSpeed(2200);  HardDisk disk = new HardDisk();  disk.setAmount(500);  PC pc = new PC();  pc.Cpu = cpu;  pc.HD = disk;  pc.show();  }  }   * 1. 运行程序和运行结果（截图）：      1. **设计一个长方体Cbox类， 其成员变量有：length,width,height； 提供有参和无参两个构造方法（参数初始化成员变量）、 设置成员变量值方法setData(double,double,double)、计算体积方法getVolume（），要求对象只能通过方法操作成员变量。**   **Sy3\_7：主类，main方法完成以下功能：**  **（1）利用无参构造方法创建对象box1，设置对象box1的length，width,height值，计算box1体积并输出。**  **（2）利用有参构造方法创建对象box2，计算box2体积并输出。**  **/\*本题需注意访问权限的正确使用\*/**   * 1. 程序清单：   public class Cbox {  double length;  double width;  double height;  Cbox(){  this.length = 0;  this.width = 0;  this.height = 0;  }  Cbox(double length, double width, double height){  this.length = length;  this.width = width;  this.height = height;  }  public void setData(double length, double width, double height){  this.length = length;  this.width = width;  this.height = height;  }  public double getVolume(){  return multiply(multiply(this.height, this.length), this.width);  }  public double multiply(double a, double b){  return a \* b;  }  }  public class main7 {  public static void main(String[] args) {  Cbox box1 = new Cbox();  box1.setData(20.0, 30.0, 40.0);  System.out.println("wys 计算得出 box1 的体积是" + box1.getVolume());  Cbox box2 = new Cbox(20.0, 30.0, 40.0);  System.out.println("wys 计算得出 box2 的体积是" + box2.getVolume());  }  }   * 1. 运行程序和运行结果（截图）： | | | | |
| 总  结 | 实验心得体会： | | | | |