OOP编程题

1．编一个程序，程序中包含以下内容：

一个圆类(Circle)，包含：

属性：圆半径radius；常量：PI。

方法：构造方法；求面积方法area()；求周长方法：perimeter()。

主类(X4\_3\_1)，包含：

主方法main()，在主方法中创建圆类的对象c1和c2并初始化，c1的半径为100，c1的半径为200，然后分别显示两个圆的面积和周长。

【编程分析】按照要求创建Circle类，其中的半径可以定义为int类型，PI定义为final static double类型，求面积和周长的方法都应定义为double类型，在构造方法中对radius进行初始化。

【参考答案】

public class X4\_3\_1 {

public static void main(String[] args) {

Circle c1 = new Circle(100);

Circle c2 = new Circle(200);

System.out.println("c1.area() = " +c1.area()+"\tc1.perimenter() = "+c1.perimeter());

System.out.println("c2.area() = " +c2.area()+"\tc2.perimenter() = "+c2.perimeter());

}

}

class Circle{

int radius;

final static double PI=3.14;

Circle(int r){

radius = r;

}

double area(){

return PI\*radius\*radius;

}

double perimeter(){

return 2\*PI\*radius;

}

}

【运行结果】

c1.area() = 31400.0 c1.perimenter() = 628.0

c2.area() = 125600.0 c2.perimenter() = 1256.0

2．编一个程序，程序中包含以下内容：

一个学生类（Student），包含：

属性：学号s\_No，姓名s\_Name，性别s\_Sex，年龄s\_Age。

方法：构造方法，显示学号方法showNo()，显示姓名方法showName()，显示性别方法showSex()，显示年龄方法showAge()，修改年龄方法modifyAge()。

主类(X4\_3\_2)，包含：

主方法main()，在其中创建两个学生对象s1和s2并初始化，两个对象的属性自行确定，然后分别显示这两个学生的学号、姓名、性别、年龄，然后修改s1的年龄并显示修改后的结果。

【编程分析】按照要求首先编写Student类，其中的属性和方法根据实际情况选择相应的类型和权限修饰符，要通过方法来修改年龄，因此s\_Age属性应该为private类型，否则可以直接在主类中进行修改，就没有必要使用方法了。

【参考答案】

public class X4\_3\_2 {

public static void main(String[] args) {

Student s1 = new Student(101,"张三","男",18);

Student s2 = new Student(102,"李四","女",16);

System.out.println("第1个学生的信息为：");

s1.showNo();

s1.showName();

s1.showSex();

s1.showAge();

System.out.println("第2个学生的信息为：");

s2.showNo();

s2.showName();

s2.showSex();

s2.showAge();

System.out.println("修改第1个学生的年龄：");

s1.modifyAge(24);

System.out.println("第1个学生的年龄修改为：");

s1.showAge();

}

}

class Student{

private int s\_No;

private String s\_Name;

private String s\_Sex;

private int s\_Age;

Student(int no, String name, String sex, int age){

s\_No = no;

s\_Name = name;

s\_Sex = sex;

s\_Age = age;

}

void showNo(){

System.out.println("学号："+s\_No);

}

void showName(){

System.out.println("姓名：" + s\_Name);

}

void showSex(){

System.out.println("性别：" + s\_Sex);

}

void showAge(){

System.out.println("年龄：" + s\_Age);

}

void modifyAge(int newAge){

s\_Age = newAge;

}

}

【运行结果】

第1个学生的信息为：

学号：101

姓名：张三

性别：男

年龄：18

第2个学生的信息为：

学号：102

姓名：李四

性别：女

年龄：16

修改第1个学生的年龄：

第1个学生的年龄修改为：

年龄：24

3．编写一个实现方法重载的程序。

【编程分析】重载的含义就是在一个类中定义多个具有相同方法名，不同参数列表的方法。在下面程序中的类中定义了三个同名方法area，分别用于求圆、矩形和三角形的面积。三个方法具有不同的参数。

【参考程序】

public class X5\_3\_1{

public double area(double radius){

return Math.PI\*radius\*radius;

}

public double area(double width, double height){

return width\*height;

}

public double area(double a, double b, double c){

double s = (a+b+c)/2;

return Math.sqrt(s\*(s-a)\*(s-b)\*(s-c));

}

public static void main(String[] args){

X5\_3\_1 shape = new X5\_3\_1();

System.out.println("The area of circle is: "+shape.area(10));

System.out.println("The area of rectangle is: "+shape.area(10,20));

System.out.println("The area of triangle is: "+shape.area(10,15,20));

}

}

【运行结果】

The area of circle is: 314.1592653589793

The area of rectangle is: 200.0

The area of triangle is: 72.61843774138907

4．编写一个实现方法覆盖的程序。

【编程分析】方法覆盖是指在子类中重新定义了父类中的方法。本例中在Shape类、Circle类、Cylinder类中都定义了area方法，而且Cylinder类继承了Circle类、Circle类继承了Shape类，从而实现了area方法的覆盖。

【参考程序】

abstract class Shape {

abstract protected double area();

}

class Circle extends Shape{

float r;

public Circle(float a) {

r=a;

}

public double area(){

System.out.print("Calculate the area of circle: ");

return Math.PI\*r\*r;

}

}

class Cylinder extends Circle {

float h;

public Cylinder(float a,float b) {

super(a);

h=b;

}

public double area() {

System.out.print("Calculate the area of cylinder: ");

return 2\*Math.PI\*r\*r+2\*Math.PI\*r\*h;

}

}

public class EX5\_3\_2 {

public static void main(String args[]) {

Circle cl=new Circle(3);

Cylinder cd=new Cylinder(2,5);

System.out.println(cl.area());

System.out.println(cd.area());

}

}

【运行结果】

Calculate the area of circle: 28.274333882308138

Calculate the area of cylinder: 87.96459430051421

5．编写一个实现数据成员隐藏的程序。

【编程分析】数据成员的隐藏是指子类和父类都有一个相同的数据成员，当子类对象调用该数据成员时，默认调用子类的该成员，而不是父类的该成员，相当于隐藏了父类的该成员。本例中Hide类中定义了数据成员x，该类的子类X\_5\_3\_3中也定义了数据成员x，当子类对象调用x时，调用的是子类的x，而不是父类的x。

【参考程序】

public class X5\_3\_3 extends Hide {

int x = 200;

public static void main(String args[]) {

X5\_3\_3 obj = new X5\_3\_3 ();

System.out.println("x = "+obj.x); // 数据成员的隐藏

}

}

class Hide{

int x = 100;

}

【运行结果】

x = 200

6．编写一个使用this和super关键字的程序。

【编程分析】this关键字表示当前对象本身，而super关键字则是当前对象的直接父类对象。本例程序中定义了两个类X1和X2，其中都定义了成员变量x、y。此时就需要区分父类与子类的成员变量，采用super引用父类的成员变量和方法。

注意：父类和子类中都定义了show()方法，如果没有特别指定，则是调用当前类的方法。

【参考程序】

public class X5\_3\_4{

public static void main(String[] args){

X2 x = new X2(5);

x.setY();

x.show();

System.out.println("super.y="+x.getY()); // 输出父类数据成员y

System.out.println("this.y="+x.y); // 输出子类数据成员y

}

}

class X1{

int x,y;

X1(int i){

x=i;

y=x\*2;

}

int getY(){

return y;

}

void show(){

System.out.println("x="+x+" y="+y);

}

}

class X2 extends X1{

int x,y;

X2(int j){

super(3); // 利用super来调用父类的构造方法

this.x=j; // 利用this来调用本类的数据成员x

}

void show(){

System.out.println("super.x="+super.x+" this.x="+x); // 输出父类和子类的属性x

}

void setY(){

y=super.x+this.x; // 引用父类和子类中的数据成员x

}

}

【运行结果】

super.x=3 this.x=5

super.y=6

this.y=8

7．编写一个人类Person，其中包含姓名、性别和年龄的属性，包含构造方法以及显示姓名、性别和年龄的方法。再编写一个学生类Student，它继承Person类，其中包含学号属性，包含构造方法以及显示学号的方法。最后编写一个主类X5\_3\_5，包含main()方法，在main()方法中定义两个学生s1和s2并给他们赋值，最后显示他们的学号、姓名、性别以及年龄。

【编程分析】本题主要考察类的继承问题。

第一步：定义Person类。

第二步：定义Student类，该类继承Person类。

第三步：定义主类。

【参考程序】

public class X5\_3\_5 {

public static void main(String[] args) {

Student s1=new Student("Zhangsan","Male",20,"102A");

Student s2=new Student("Lisi","Female",18,"108S");

s1.show();

s1.showID();

s2.show();

s2.showID();

}

}

class Person{

String name;

String sex;

int age;

public Person(String n,String s,int a){

name = n;

sex = s;

age = a;

}

void show(){

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

System.out.println("sex: "+sex);

System.out.println("age: "+age);

}

}

class Student extends Person{

String sID;

public Student(String n,String s,int a,String sid){

super(n,s,a);

sID = sid;

}

void showID(){

System.out.println("sID: "+sID);

}

}

【运行结果】

name: Zhangsan

sex: Male

age: 20

sID: 102A

name: Lisi

sex: Female

age: 18

sID: 108S

8．编一个程序，包含以下文件。

（1）Shape.java文件，在该文件中定义接口Shape，该接口在shape包中。

属性：PI。

方法：求面积的方法area()。

（2）Circle.java文件，在该文件中定义圆类Circle，该类在circle包中，实现Shape接口。

属性：圆半径radius。

方法：构造方法；实现接口中求面积方法area()；求周长方法perimeter()。

（3）“Cylinder.java”文件，在该文件中定义圆柱体类Cylinder，该类口在cylinder包中，继承圆类。

属性：圆柱体高度height。

方法：构造方法；求表面积方法area()；求体积方法volume()。

（4）X5\_3\_6.java文件，在该文件中定义主类X5\_3\_6，该类在默认包中，其中包含主方法main()，在主方法中创建两个圆类对象cir1和cir2，具体尺寸自己确定，并显示圆的面积和周长；再创建两个圆柱体类的对象cy1和cy2，具体尺寸自己确定，然后分别显示圆柱体cy1和cy2的底圆的面积和周长以及它们各自的体积和表面积。

【编程分析】本题主要考察接口、包、继承、封装等问题。编程步骤如下：

第一步：首先创建p1包，在其中创建Shape接口

// Shape.java文件

package p1; // 创建p1包

public interface Shape{ // 定义Shape接口

…

}

第二步：创建Circle类和Cylinder类，它们都定义在p2包中。

// Circle.java文件

package p2; // 创建p2包

import p1.\*;

public class Circle implements Shape{ // 定义实现Shape接口的Circle类

…

}

// Cylinder.java文件

package p2;

public class Cylinder extends Circle{ // 创建继承Circle类的Cylinder类

…

}

第三步：创建主类，在其中的main()方法中创建对象，实现相应的功能。

// X5\_3\_6.java文件

package p3;

import p2.\*;

public class X5\_3\_6 { // 定义主类

public static void main(String[] args) {

…

}

}

【参考程序】

// X5\_3\_6.java文件

package p3;

import p2.\*;

public class X5\_3\_6 { // 定义主类

public static void main(String[] args) {

Circle cir1 = new Circle(120.5);

Circle cir2 = new Circle(183.8);

System.out.println("cir1.area: "+cir1.area());

System.out.println("cir1.perimeter: "+cir1.perimeter());

System.out.println("cir2.area: "+cir2.area());

System.out.println("cir2.perimeter: "+cir2.perimeter());

Cylinder cy1 = new Cylinder(27.3,32.7);

Cylinder cy2 = new Cylinder(133.5,155.8);

System.out.println("cy1.area: "+cy1.area());

System.out.println("cy1.volume: "+cy1.volume());

System.out.println("cy2.area: "+cy2.area());

System.out.println("cy2.volume: "+cy2.volume());

}

}

// Shape.java文件

package p1; // 创建p1包

public interface Shape{ // 定义Shape接口

double PI=Math.PI;

double area(); // 求面积方法

}

// Circle.java文件

package p2; // 创建p2包

import p1.\*;

public class Circle implements Shape{ // 定义实现Shape接口的Circle类

double radius; // 半径

public Circle(double r){

radius = r;

}

public double area(){ // 实现Shape接口中的方法（这是必须的）

return PI\*radius\*radius;

}

public double perimeter(){ // 定义求圆周长的方法

return 2\*PI\*radius;

}

}

// Cylinder.java文件

package p2;

public class Cylinder extends Circle{ // 创建继承Circle类的Cylinder类

double height;

public Cylinder(double r,double h){

super(r);

height = h;

}

public double area(){

return 2\*PI\*radius\*radius+2\*PI\*radius\*height;

}

public double volume(){

return PI\*radius\*radius\*height;

}

}

【运行结果】

cir1.area: 45616.710728287195

cir1.perimeter: 757.1238295151402

cir2.area: 106130.66532433797

cir2.perimeter: 1154.849459459608

cy1.area: 10291.857533160162

cy1.volume: 76563.70115356173

cy2.area: 242666.35550050176

cy2.volume: 8723280.89865466