# 一、选择题

|  |  |  |
| --- | --- | --- |
| 1. | 以下关于继承条件下构造方法执行过程的代码的执行结果是（）。（选择一项） | |
|  | class Person {  public Person() {  System.out.println("execute Person()");  }  }  class Student extends Person {  public Student() {  System.out.println("execute Student() ");  }  }  class PostGraduate extends Student {  public PostGraduate() {  System.out.println("execute PostGraduate()");  }  }  public class TestInherit {  public static void main(String[] args) {  new PostGraduate();  }  } | |
|  |  |  |
|  | A. | execute Person()  execute Student()  execute PostGraduate() |
|  | B. | execute PostGraduate() |
|  | C. | execute PostGraduate()  execute Student()  execute Person() |
|  | D. | 没有结果输出 |

|  |  |  |
| --- | --- | --- |
| 2. | 以下关于this和super关键字的说法错误的是（）。（选择二项） | |
|  |  |  |
|  | A. | this关键字指向当前对象自身，super关键字指向当前对象的直接父类 |
|  | B. | 在main方法中可以存在this或super关键字，但不能同时存在。  Static成员不使用非静态成员； |
|  | C. | this和super关键字都可以访问成员属性，成员方法和构造方法 |
|  | D. | 在一个类的构造方法中可以同时使用this和super来调用其他构造方法 |

|  |  |  |
| --- | --- | --- |
| 3. | 给出下面的代码段，在代码说明//assignment x=a,y=b处写入如下（）个代码是正确的。（选择一项） | |
|  | public class Base {  int w, x, y, z;  public Base(int a, int b) {  x = a;  y = b;  }  public Base(int a, int b, int c, int d) {  // assignment x=a,y=b  w = d;  z = c;  }  } | |
|  |  |  |
|  | A | Base(a,b); |
|  | B. | x=a,y=b; |
|  | C. | this(a),this(b); |
|  | D. | this(a,b) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4. | 以下Java程序编译运行后的输出结果是（ ）。（选择一项） | | | |
|  | class HelloA {  public HelloA( ) {  System.out.println("HelloA");  }  //一对大括号之间的代码 普通代码块{ }  {  System.out.println("I'm A class");  }  static {  System.out.println("static A");  }  }  public class HelloB extends HelloA {  public HelloB( ) {  super();  System.out.println("HelloB");  }  { System.out.println("I'm B class");  }  static {  System.out.println("static B");  }  public static void main(String[] args) {  new HelloB( );//匿名子类对象  }  } | | | |
|  |  |  | | |
|  | A | static A  static B  I'm A class  HelloA  I'm B class  HelloB | C. | static A  I'm A class  HelloA  static B  I'm B class  HelloB |
|  | B. | static A  static B  I'm A class  I'm B class  HelloA  HelloB | D | static A  static B  HelloA  HelloB  I'm A class  I'm B class |

|  |  |  |
| --- | --- | --- |
| 5. | 下列选项中关于Java中super关键字的说法正确的是（ ）。（选择二项） | |
|  |  |  |
|  | A | super关键字是在子类对象内部指代其父类对象的引用 |
|  | B. | super关键字不仅可以指代子类的直接父类，还可以指代父类的父类 |
|  | C. | 子类通过super关键字只能调用父类的方法，而不能调用父类的属性 |
|  | D. | 子类通过super关键字可以调用父类的构造方法 |

# 二、判断题

1、如果定义一个类时没有使用extends，则它的父类是java.lang.Object。（ T）

2、对子类实例化对象需要调用超类的构造函数进行初始化工作，完成对子类中的超类实例变量的初始化，则子类可以继承超类的构造函数。（ ）

3、在Java中任何一个子类的构造方法都必须调用其父类的构造方法（包括隐式调用），并且调用父类的构造方法的语句必须是子类构造方法的第一条语句。（ ）

4、Java中的类不允许多层/重继承，一个类只能有一个直接父类。（）

5、Object类有一个public方法是toString( )，一个对象通过调用该方法可以获得该对象的字符串表示。（ ）

6、父类Person中定义了一个private void show()的方法，那么子类要重写这个方法时，方法的访问修饰符可以是默认的，protected或public。（ ）

7、运算符“==”用于比较引用时，如果两个引用指向内存同一个对象，则返回true。（ ）

8、构造方法中，第一句话总是super。（ ）

# 三、简答题

1、继承的好处是什么? JAVA的继承使用哪个关键字实现? 定义某个类时，如果没有使用继承关键字，那么继承了哪个类?

减少代码的冗余，extends, Object类

2、方法重载和方法重写（覆盖）的区别。

3、Object类中的toString方法能否被子类重写?请做测试。

4、重写中，子类的返回值类型可不可以跟父类不完全一致?说出你的理由。

1. Object是所有类的根类吗?是所有类的直接父类吗?在哪里查看Object类的源代码?
2. java.lang.Object类的六个常用方法的声明并说明其作用。

toString equals ,hashCode;

1. 继承条件下子类构造方法的执行过程;

答：先构建父类的直接父类对象，然后再构建子类对象；

1. 假如父类有main方法，子类能不能继承?

9、super关键字的作用和使用

10、==和equals()的联系和区别

# 四、编码题

1、编写应用程序，创建类的对象，分别设置圆的半径、圆柱体的高，计算并分别显示圆半径、圆面积、圆周长，圆柱体的体积。（7分）

提示:

（1）编写一个圆类Circle，该类拥有：

一个成员变量，radius（私有，浮点型）;//存放圆的半径；

两个构造方法

Circle（） //将半径设为0

Circle（double r ） //创建Circle对象时将半径初始化为r

三个成员方法

double getArea（） //获取圆的面积

double getPerimeter（） //获取圆的周长

void show（） //将圆的关径、周长、面积输出到屏幕

（2） 编写一个圆柱体类Cylinder，它继承于上面的Circle类。还拥有：

一个成员变量，double hight （私有，浮点型）; //圆柱体的高；

构造方法

//创建Cylinder对象时将半径初始化为r,高度初始化为h

Cylinder（double r,double h）

成员方法

double getVolume（） //获取圆柱体的体积

void showVolume（） //将圆柱体的体积输出到屏幕

package homework;

public class Circle {

private double radius;

final double PI=3.1415;

public Circle(double radius) {

super();

this.radius = radius;

}

public Circle() {

}

public double getArea() {

return radius\*PI\*radius;

}

public double getPerimeter() {

return radius\*PI\*2;

}

public void show() {

System.*err*.println("半径："+radius+",周长"+getPerimeter()+",面积"+getArea());

}

}

class Cylinder extends Circle{

private double hight;

public Cylinder(double radius,double hight) {

super(radius);

this.hight=hight;

}

public Cylinder() {

}

public double getVolume(){

return getArea()\*hight;

}

public void showVolume() {

System.*err*.println("圆柱体的面积："+getVolume());

}

}

class Test{

public static void main(String[] args) {

Circle circle = new Circle(2);

circle.show();

Cylinder cylinder = new Cylinder(2,5);

cylinder.showVolume();

}

}

2、请使用面向对象的思想，设计自定义类，描述出租车和家用轿车的信息。

设定

出租车类:

属性包括：车型，车牌，所属出租公司；方法包括：启动，停止

家用轿车类：

属性包括：车型，车牌，车主姓名；方法包括：启动，停止

要求

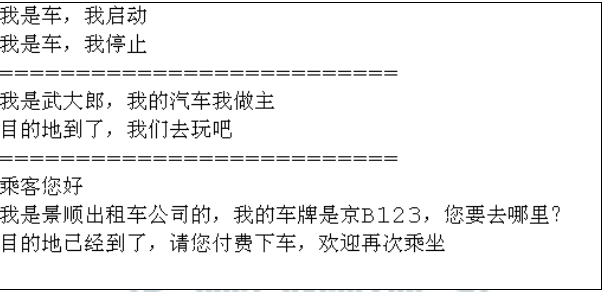
分析出租车和家用轿车的公共成员，提取出父类—汽车类

利用继承机制，实现出租车类和家用轿车类

编写测试类，分别测试汽车类，出租车类和家用轿车类对象的相关方法

定义名为car的包存放汽车类，出租车类，家用轿车类和测试类

运行效果



package homework;

public class Car {

String vehicleModel;

String plateNumber;

public Car(String vehicleModel, String plateNumber) {

super();

this.vehicleModel = vehicleModel;

this.plateNumber = plateNumber;

}

public Car() {

}

public void start() {

System.*out*.println("我是车，我启动");

}

public void stop() {

System.*out*.println("我是车，我停止");

}

}

class Taxi extends Car{

String taxiCompany;

public Taxi() {

}

public Taxi(String vehicleModel, String plateNumber, String taxiCompany) {

super(vehicleModel, plateNumber);

this.taxiCompany = taxiCompany;

}

public void start() {

System.*out*.println("乘客您好我是"+taxiCompany+"的，我的车牌是"+plateNumber+",您要去哪里");

}

public void stop() {

System.*out*.println("目的地已到，请您付费下车，欢迎再次乘坐");

}

}

class FamilCar extends Car{

String name;

public FamilCar() {

}

public FamilCar(String vehicleModel, String plateNumber,String name) {

super(vehicleModel, plateNumber);

this.name = name;

}

public void start() {

System.*out*.println("我是"+name+",我的汽车我做主");

}

public void stop() {

System.*out*.println("目的地了，我门去玩吧");

}

}

class CarTest{

public static void main(String[] args) {

Car car = new Car();

FamilCar familCar = new FamilCar("北京现代","12345","武大郎");

Taxi taxi = new Taxi("大众","京B123","景顺出租车公司");

car.start();

car.stop();

familCar.start();

familCar.stop();

taxi.start();

taxi.stop();

}

}

3、某公司要开发新游戏，请用面向对象的思想，设计游戏中的蛇怪和蜈蚣精

设定

蛇怪类:

属性包括：怪物名字，生命值，攻击力

方法包括：攻击，移动（曲线移动），补血（当生命值<10时，可以补加20生命值）

蜈蚣精类：

属性包括：怪物名字，生命值，攻击力

方法包括：攻击，移动（飞行移动）

要求

分析蛇怪和蜈蚣精的公共成员，提取出父类—怪物类

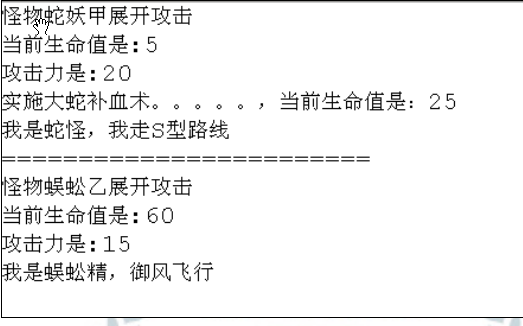
利用继承机制，实现蛇怪类和蜈蚣精类

攻击方法，描述攻击状态。内容包括怪物名字，生命值，攻击力

编写测试类，分别测试蛇怪和蜈蚣精的对象及相关方法

定义名为mon的包存怪物类，蛇怪类，蜈蚣精类和测试类

运行效果



# 可选题

请用面向对象的思想，设计自定义类描述演员和运动员的信息

设定

演员类:

属性包括：姓名，年龄，性别，毕业院校，代表作

方法包括：自我介绍

运动员类：

属性包括：姓名，年龄，性别，运动项目，历史最好成绩

方法包括：自我介始

要求

分析演员和运动员的公共成员，提取出父类—人类

利用继承机制，实现演员类和运动员类

编写测试类，分别测试人类，演员类和运动员类对象及相关方法

定义名为act的包存人类，演员类，运动员类和测试类

运行效果；

