**封装和多态作业**

1. **选择题**

|  |  |  |
| --- | --- | --- |
| **1.** | **使用权限修饰符（ B ）修饰的类的成员变量和成员方法，可以被当前包中所有类访问，也可以被它的子类（同一个包以及不同包中的子类）访问。（选择一项）** | |
|  |  |  |
|  | **A** | public |
|  | **B.** | protected |
|  | **C.** | 默认 |
|  | **D.** | private |

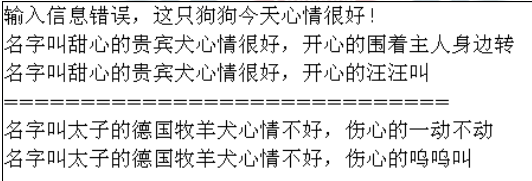
|  |  |  |
| --- | --- | --- |
| **2.** | **给出如下代码，如何使成员变量m被方法fun()直接访问（ C ）。（选择一项）** | |
|  | **class** Test {  **private** **int** m;  **public** **static** **void** fun() {  }  } | |
|  |  |  |
|  | **A** | 将private int m 改为protected int m |
|  | **B.** | 将private int m 改为public int m |
|  | **C.** | 将private int m 改为static int m |
|  | **D.** | 将private int m 改为int m |

1. **判断题**
2. 使用public修饰的成员属性和方法可以被当前项目中所有包的所有类访问。( **〇** )
3. 类的方法通常设为public，而类的实例变量一般也设为public。（ **×** ）
4. 与未加访问控制符的缺省情况相比，public和protected修饰符扩大了类及其属性和方法的被访问范围，private修饰符则缩小了这种范围。（ **〇** ）
5. 访问权限是private的变量，只能在本类和与本类同一个包中的其他类使用。（ **×** ）
6. **简答题**
   1. private、默认、protected、public四个权限修饰符的作用
7. **编码题**
   1. 使用面向对象的思想，编写自定义描述狗的信息。设定属性包括：品种，年龄，心情，名字；方法包括：叫，跑。

要求：

1. 设置属性的私有访问权限，通过公有的get,set方法实现对属性的访问
2. 限定心情只能有“心情好”和“心情不好”两种情况，如果无效输入进行提示，默认设置“心情好”。
3. 设置构造函数实现对属性赋值
4. 叫和跑的方法，需要根据心情好坏，描述不同的行为方式。
5. 编写测试类，测试狗类的对象及相关方法（测试数据信息自定义）

运行效果图:



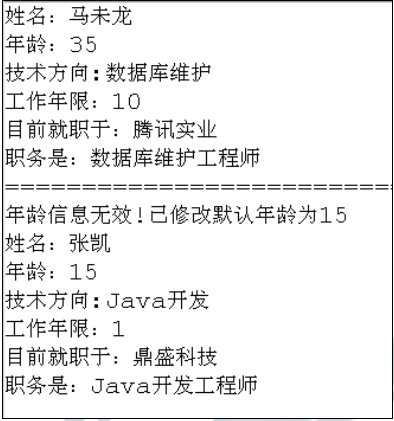
**public class** Dog {  
 **private** String **variety**;  
 **private int age**;  
 **private** String **mood**;  
 **private** String **name**;  
  
 **public** Dog(){  
  
 }  
  
 **public** Dog(String variety,**int** age,String mood,String name){  
 **this**.**variety** = variety;  
 **this**.**age** = age;  
 setMood(mood);  
 **this**.**name** = name;  
 }  
  
  
 **public** String getMood() {  
 **return mood**;  
 }  
  
 **public void** setMood(String mood){  
 **if**(**"心情好"**.equals(mood)||**"心情不好"**.equals(mood)){  
 **this**.**mood** = mood;  
 }**else**{  
 System.***out***.println(**"输入信息错误，这只狗狗今天心情很好！"**);  
 **this**.**mood** = **"心情好"**;  
 }  
 }  
  
 **public** String getVariety() {  
 **return variety**;  
 }  
  
 **public void** setVariety(String variety) {  
 **this**.**variety** = variety;  
 }  
  
 **public int** getAge() {  
 **return age**;  
 }  
  
 **public void** setAge(**int** age) {  
 **this**.**age** = age;  
 }  
  
 **public** String getName() {  
 **return name**;  
 }  
  
 **public void** setName(String name) {  
 **this**.**name** = name;  
 }  
  
  
  
 **public void** run() {  
 **if** (**"心情好"**.equals(**mood**)) {  
 System.***out***.println(**"名字叫"** + **name** + **"的"** + **variety** + **mood** + **",开心的围着主任身边转"**);  
 } **else** {  
 System.***out***.println(**"名字叫"** + **name** + **"的"** + **variety** + **mood** + **",伤心的一动不动"**);  
 }  
 }  
 **public void** bark (){  
 **if** (**"心情好"**.equals(**mood**)) {  
 System.***out***.println(**"名字叫"** + **name** + **"的"** + **variety** + **mood** + **",开心的汪汪叫"**);  
 } **else** {  
 System.***out***.println(**"名字叫"** + **name** + **"的"** + **variety** + **mood** + **",伤心的呜呜叫"**);  
 }  
 }  
  
 **public static void** main (String[]args) {  
 Dog dog1 = **new** Dog(**"杜宾犬"**, 2, **"心情不好"**,**"甜心"**);  
 dog1.run();  
 dog1.bark();  
 Dog dog2 = **new** Dog(**"京巴"**, 2, **"心情好"**,**"花花"**);  
 dog2.run();  
 dog2.bark();  
 Dog dog3 = **new** Dog(**"腊肠"**, 2, **"心情还行"**,**"虎子"**);  
 dog3.run();  
 dog3.bark();  
 }  
}

* 1. 以面向对象的思想，编写自定义类描述IT从业者。设定属性包括：姓名，年龄，技术方向，工作年限, 工作单位和职务；方法包括：工作

要求：

1. 设置属性的私有访问权限，通过公有的get,set方法实现对属性的访问
2. 限定IT从业人员必须年满15岁，无效信息需提示，并设置默认年龄为15。
3. 限定“技术方向”是只读属性
4. 工作方法通过输入参数，接收工作单位和职务，输出个人工作信息
5. 编写测试类，测试IT从业者类的对象及相关方法（测试数据信息自定义）

运行效果图:



**public class** ITWork {  
 **private** String **name**;  
 **private int age**;  
 **private** String **tend**;  
 **private int year**;  
  
  
 **public** ITWork(){  
  
 }  
  
 **public** ITWork(String name,**int** age,String tend,**int** year){  
 **this**.**name** = name;  
 **this**.setAge(age);  
 **this**.**tend** = tend;  
 **this**.**year** = year;  
 }  
  
 **public** String getName() {  
 **return name**;  
 }  
  
 **public void** setName(String name) {  
 **this**.**name** = name;  
 }  
  
 **public int** getAge() {  
 **return age**;  
 }  
  
 **public void** setAge(**int** age) {  
 **if**(age>=15){  
 **this**.**age** = age;  
 }**else**{  
 **this**.**age** = 15;  
 System.***out***.println(**"年龄信息无效!已修改默认年龄为15"**);  
 }  
 }  
  
 **public** String getTend() {  
 **return tend**;  
 }  
  
  
  
 **public int** getYear() {  
 **return year**;  
 }  
  
 **public void** setYear(**int** year) {  
 **this**.**year** = year;  
 }  
  
 **public void** work(String company,String position){  
 System.***out***.println(**"姓名："**+**name**);  
 System.***out***.println(**"年龄："**+**age**);  
 System.***out***.println(**"技术方向："**+**tend**);  
 System.***out***.println(**"工作年限："**+**year**);  
 System.***out***.println(**"目前就职于："**+company);  
 System.***out***.println(**"职务是："**+position);  
 }  
}

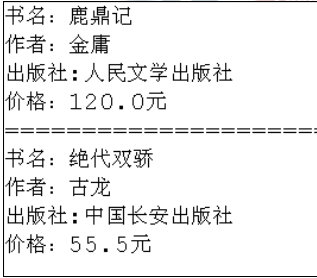
**public class** ITWorkTest {  
 **public static void** main(String[] args) {  
  
 ITWork it1 = **new** ITWork(**"ling"**,13,**"软件工程师"**,15);  
 it1.work(**"大华有限公司"**,**"副经理"**);  
 ITWork it2 = **new** ITWork(**"gen"**,28,**"运维工程师"**,8);  
 it2.work(**"青鸟公司"**,**"经理"**);  
  
 }  
}

1. **可选题**
   1. 以面向对象的思想，编写自定义类描述图书信息。设定属性包括：书名，作者，出版社名，价格；方法包括：信息介绍

要求：

1. 设置属性的私有访问权限，通过公有的get,set方法实现对属性的访问
2. 限定介格必须大于10，如果无效进行提示
3. 限定作者，书名为只读属性
4. 设计构造方法实现对属性赋值
5. 信息介绍方法描述图书所有信息
6. 编写测试类，测试图书类的对象及相关方法（测试数据信息自定）

运行效果图:



**public class** Book {  
 **private** String **bookName**;  
 **private** String **author**;  
 **private** String **publisher**;  
 **private double price**;  
  
 **public** Book(){  
 }  
  
 **public** Book(String bookName,String author,String publisher,**double** price){  
 **this**.**bookName** = bookName;  
 **this**.**author** = author;  
 **this**.**publisher** = publisher;  
 **this**.setPrice(price);  
 }  
  
 **public** String getBookName() {  
 **return bookName**;  
 }  
  
 **public** String getAuthor() {  
 **return author**;  
 }  
  
 **public** String getPublisher() {  
 **return publisher**;  
 }  
  
 **public void** setPublisher(String publisher) {  
 **this**.**publisher** = publisher;  
 }  
  
 **public double** getPrice() {  
 **return price**;  
 }  
  
 **public void** setPrice(**double** price) {  
 **if**(price>10){  
 **this**.**price** = price;  
 }**else**{  
 System.***out***.println(**"价格必须大于10"**);  
 **this**.**price** = price;  
 }  
 }  
  
 **public void** infoIntroduce(){  
 System.***out***.println(**"书名："**+**bookName**);  
 System.***out***.println(**"作者："**+**author**);  
 System.***out***.println(**"出版社："**+**publisher**);  
 System.***out***.println(**"价格:"**+**price**);  
 System.***out***.println(**"======================================="**);  
 }  
}

**public class** BookTest {  
 **public static void** main(String[] args) {  
 Book book1 = **new** Book(**"鹿鼎记"**,**"金庸"**,**"人民文学出版社"**,120);  
 book1.infoIntroduce();  
 Book book2 = **new** Book(**"绝代双骄"**,**"古龙"**,**"中国长安出版社"**,5);  
 book2.infoIntroduce();  
 }  
}

* 1. 某公司要开发名为”我爱购物狂”的购物网站，请使用面向对象的思想设计描述商品信息

要求：

1. 分析商品类别和商品详细信息属性和方法，设计商品类别类和商品详细信息类
2. 在商品详细信息类中通过属性描述该商品所属类别
3. 设置属性的私有访问权限，通过公有的get,set方法实现对属性的访问
4. 编写测试类，测试商品类别类和商品详细信息类的对象及相关方法（测试数据信息自定）
5. 创建包info—存放商品类别类和商品详细信息类，创建包test—存放测试类

参考分析思路:

商品类别类：

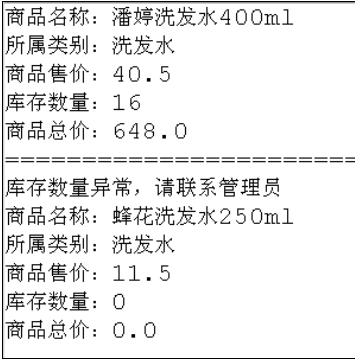
属性：类别编号，类别名称

商品详细信息类：

属性：商品编号，商品名称，所属类别，商品数量（大于0），商品价格（大于0），

方法：盘点的方法，描述商品信息。内容包括商品名称，商品数量，商品价格，现在商品总价以及所属类别信息

运行效果图:



**package** info;

**public** **class** ProductCategory {

**private** String cid;

**private** String name;

**public** ProductCategory() {

**super**();

}

**public** ProductCategory(String cid, String name) {

**super**();

**this**.cid = cid;

**this**.name = name;

}

**public** String getCid() {

**return** cid;

}

**public** **void** setCid(String cid) {

**this**.cid = cid;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

}

**package** info;

**public** **class** Product {

**private** String pid;

**private** String name;

**private** **int** amount;

**private** **double** price;

**private** ProductCategory category;

**public** Product() {

**super**();

}

**public** Product(String pid, String name, **int** amount, **double** price) {

**super**();

**this**.pid = pid;

**this**.name = name;

**this**.amount = amount;

**this**.price = price;

}

**public** Product(String pid, String name, **int** amount, **double** price,

ProductCategory category) {

**super**();

**this**.pid = pid;

**this**.name = name;

**this**.setAmount(amount);

**this**.price = price;

**this**.category = category;

}

**public** String getPid() {

**return** pid;

}

**public** **void** setPid(String pid) {

**this**.pid = pid;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** **int** getAmount() {

**return** amount;

}

**public** **void** setAmount(**int** amount) {

**if**(amount<0){

System.***out***.println("库存数量异常，请联系管理员");

**this**.amount = 0;

}**else**{

**this**.amount = amount;

}

}

**public** **double** getPrice() {

**return** price;

}

**public** **void** setPrice(**double** price) {

**this**.price = price;

}

**public** ProductCategory getCategory() {

**return** category;

}

**public** **void** setCategory(ProductCategory category) {

**this**.category = category;

}

**public** **void** check(){

System.***out***.println("商品名称："+**this**.name);

System.***out***.println("所属类别："+**this**.category.getName());

System.***out***.println("库存数量："+**this**.price);

System.***out***.println("商品售价："+**this**.amount);

System.***out***.println("商品总价："+**this**.price\***this**.amount);

}

}

**public** **class** TestProduct {

**public** **static** **void** main(String[] args) {

//指定商品信息并盘点

ProductCategory category1 = **new** ProductCategory("11", "洗发水");

Product p1 = **new** Product("111", "潘婷洗发水400ml",

16, 40.5, category1);

p1.check();

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

//指定商品信息并盘点

Product p2 = **new** Product();

p2.setPid("222");

p2.setName("蜂花洗发水250ml");

p2.setPrice(11.5);

p2.setAmount(-5);

p2.setCategory(category1);

p2.check();

}

}