**面对对象 类和对象**

【实验习题1】   
为“无名的粉”写一个类：class WuMingFen 要求：

1. 有三个属性：  
   面码:String theMa

粉的分量(两)：int quantity

是否带汤：boolean likeSoup

2）写一个构造方法，以便于简化初始化过程，如：  WuMingFen f1 = new WuMingFen("牛肉",3,true);

3）重载构造方法，使得初始化过程可以多样化：  WuMingFen f2 = new WuMingFen("牛肉",2);

4）如何使得下列语句构造出来的粉对象是 酸辣面码、2两、带汤的？  WuMingFen f3 = new WuMingFen();

5）写一个普通方法：check()，用于查看粉是否符合要求。即：将对象的三个属性打印在控制台上。

**class** WuMingFen{

**private** String thema;

**private** **int** quantity;

**private** **boolean** likesoup;

**public** WuMingFen(String thema, **int** quantity, **boolean** likesoup){

**this**.thema=thema;

**this**.quantity=quantity;

**this**.likesoup=likesoup;

}

**public** WuMingFen(String thema,**int** quantity){

**this**.thema=thema;

**this**.quantity=quantity;

}

**public** WuMingFen(){

**this**.thema="酸辣粉";

**this**.quantity=2;

**this**.likesoup=**true**;

}

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

System.*out*.println("面码"+(String) **this**.thema+"\n粉的分量为"+**this**.quantity+"两"+(**this**.likesoup==**true**?"\n是":"\n否"));

}

}

**public** **class** Text1 {

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

WuMingFen f1=**new** WuMingFen("牛肉",3,**true**);

f1.check();

System.*out*.println("\n");

WuMingFen f2=**new** WuMingFen("牛肉",2);

f2.check();

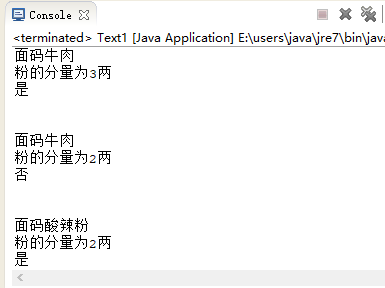
System.*out*.println("\n");

WuMingFen f3=**new** WuMingFen();

f3.check();

}

}



【实验习题2】  
 定义类Human，具有若干个属性和功能：定义其子类Man、Woman;在主类Test中分别创建子类、父类和上转型对象，并测试其特性

**class** Human{

String name;

String sex;

**int** age;

String cardnumber;//身份证号

**public** **void** getname(){

System.*out*.println(name);

}

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

**this**.name=name;

}

**public** **void** getsex(){

System.*out*.println(sex);

}

**public** **void** showsex(String sex){

**this**.sex=sex;

}

**public** **void** getage(){

System.*out*.println(age);

}

**public** **void** showage(**int** age){

**this**.age=age;

}

**public** **void** getcardnumber(){

System.*out*.println(cardnumber);

}

**public** **void** showcardnumber(String cardnumber){

**this**.cardnumber=cardnumber;

}

**public** **void** read(){

System.*out*.println("Human类可以进行读操作");

}

**public** **void** write(){

System.*out*.println("Human类可以进行写操作");

}

}

**class** Man **extends** Human{

**public** **void** m1(){

System.*out*.println("Man/n");

}

}

**class** Women **extends** Human{

**public** **void** w1(){

System.*out*.println("women/n");

}

}

**public** **class** Text2 {

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

Human hu=**new** Man();

Man man=(Man)hu;

man.m1();

Human people=**new** Women();

Women women=(Women)people;

women.w1();

}

}

【实验习题3】

自己实现一个时间类(myTime)

定义三个私有整型成员：时（hour），分（minute），秒（second）。

为MyTime类定义构造方法，以方便创建对象时初始化成员变量。//this

再定义diaplay方法，用于将时间信息打印出来。

为MyTime类添加以下方法：

addSecond(int sec) subSecond(int sec)

addMinute(int min) subMinute(int min)

addHour(int hou) subHour(int hou)

分别对时、分、秒进行加减运算。

**class** myTime{

**private** **int** hour;

**private** **int** minute;

**private** **int** second;

**public** **void** gethour(){

System.*out*.println(hour);

}

**public** **void** showhour(**int** hour){

**this**.hour=hour;

}

**public** **void** getminute(){

System.*out*.println(minute);

}

**public** **void** showminute(**int** minute){

**this**.minute=minute;

}

**public** **void** getsecond(){

System.*out*.println(second);

}

**public** **void** showsecond(**int** second){

**this**.second=second;

}

**public** **void** display(){

System.*out*.println("当前时间为"+hour+":"+minute+":"+second);

}

**public** **void** addSecond(**int** sec){

second=second+sec;

}

**public** **void** addMinute(**int** min){

minute=minute+min;

}

**public** **void** addHour(**int** hou){

hour=hour+hou;

}

**public** **void** subSecond(**int** sec){

second=second-sec;

}

**public** **void** subMinute(**int** min){

minute=minute-min;

}

**public** **void** subHour(**int** hou){

hour=hour-hou;

}

}

**public** **class** Text3 {

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

myTime time=**new** myTime();

time.showhour(11);

time.showminute(59);

time.showsecond(58);

time.addHour(1);

time.addMinute(1);

time.addSecond(1);

time.subHour(1);

time.subMinute(1);

time.subSecond(1);

time.display();

}

}

【实验习题4】

写一个账户类(Account)

属性:   
 id:账户号码 long

password:账户密码 String

name:真实姓名 String

personId:身份证号码 String

email:客户的电子邮箱 String

balance:账户余额 double

方法:

deposit: 存款方法,参数是double型的金额

withdraw:取款方法,参数是double型的金额

构造方法:

有参和无参,有参构造方法用于设置必要的属性

**class** Account{

**long** id;//账户号码

String password;//密码

String name;

String personld;

String email;

**double** balance;//账户余额

**public** **void** deposit(**double** money){

balance=money+balance;

System.*out*.println("现有余额为"+balance);

}

**public** **void** deposit(){

**double** money=1;

balance=money+balance;

System.*out*.println("现有余额为"+balance);

}

**public** **void** withdraw(**double** money){

**if**(balance<0||balance<money){

System.*out*.println("银行卡里剩余金额不足，无法取出");

}**else**{

balance=balance-money;

System.*out*.println("现有余额为"+balance);

}

}

**public** **void** withdraw(){

**double** money=200;

**if**(balance<0||balance<money){

System.*out*.println("银行卡里剩余金额不足，无法取出");

}**else**{

balance=balance-money;

System.*out*.println("现有余额为"+balance);

}

}

}

**public** **class** Text4 {

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

Account act=**new** Account();

act.id=123456789;

act.password="123456";

act.name="张三";

act.personld="360302100802014512";

act.balance=200;

act.deposit(100);

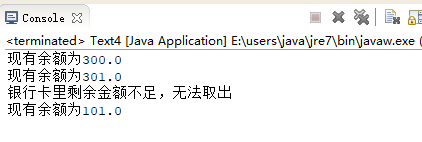
act.deposit();

act.withdraw(400);

act.withdraw();

}

}



1. 【实验习题5】  
   定义一个父类的"Role"类，有姓名，年龄，性别等成员变量

1）要求尽可能隐藏所有变量(能够私有就私有,能够保护就不要公有)，再通 过GetXXX()和SetXXX()方法对各变量进行读写。具有一个play()方法，该方法不返回任何值，同时至少定义两个构造方法。Role类中要体现出this的几种用法。

2） 从Role类派生出一个"Employee"类，该类具有Role类的所有成员（构 造方法除外），并扩展 salary成员变量，同时增加一个静态成员变量“职工编号（ID）”。同样要有至少两个构造方法，要体现出this和super的几种用法，还要求覆盖play()方法，并提供一个final sing()方法。

3）"Manager"类继承"Employee"类，有一个final成员变量"vehicle"

4） 在main()方法中制造Manager和Employee对象,并测试这些对象的方法

**class** Role{

**protected** String name;

**protected** String sex;

**protected** **int** age;

**public** **void** getname(){

System.*out*.println(name);

}

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

**this**.name=name;

}

**public** **void** getsex(){

System.*out*.println(sex);

}

**public** **void** showsex(String sex){

**this**.sex=sex;

}

**public** **void** getnage(){

System.*out*.println(age);

}

**public** **void** showname(**int** age){

**this**.age=age;

}

**public** **void** play(){

}

**public** **void** get(){

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

}

**public** **void** show(){

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

}

}

**class** Employee **extends** Role{

**int** salary;

**static** String *id*;

**public** Employee(String name,String sex,**int** age,String id,**int** salary){

**this**.name=name;

**this**.sex=sex;

**this**.age=age;

**this**.salary=salary;

**this**.*id*=id;

}

Employee(){

**super**();

}

**public** **void** play(){

System.*out*.println("职工信息");

}

**public** **void** finalsing(){

System.*out*.println("姓名"+name+"\n性别"+sex+"\n年龄"+age+"\n工资"+salary+"\n职工编号"+*id*);

}

}

**class** Manager **extends** Employee{

**final** String vehicle="三轮";

Manager(){

**super**();

}

}

**public** **class** Text5 {

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

Employee em=**new** Employee();

em.play();

em.finalsing();

}

}