Java基础练习3

1. 建立一个实体类Student类，属性：姓名，年龄，成绩，班级

建立一个list1，包含

“张三，18岁，80分，1班”，

“李四，19岁，100分，1班”，

“王五，17岁，59分，1班”。

建立一个list2，

包含

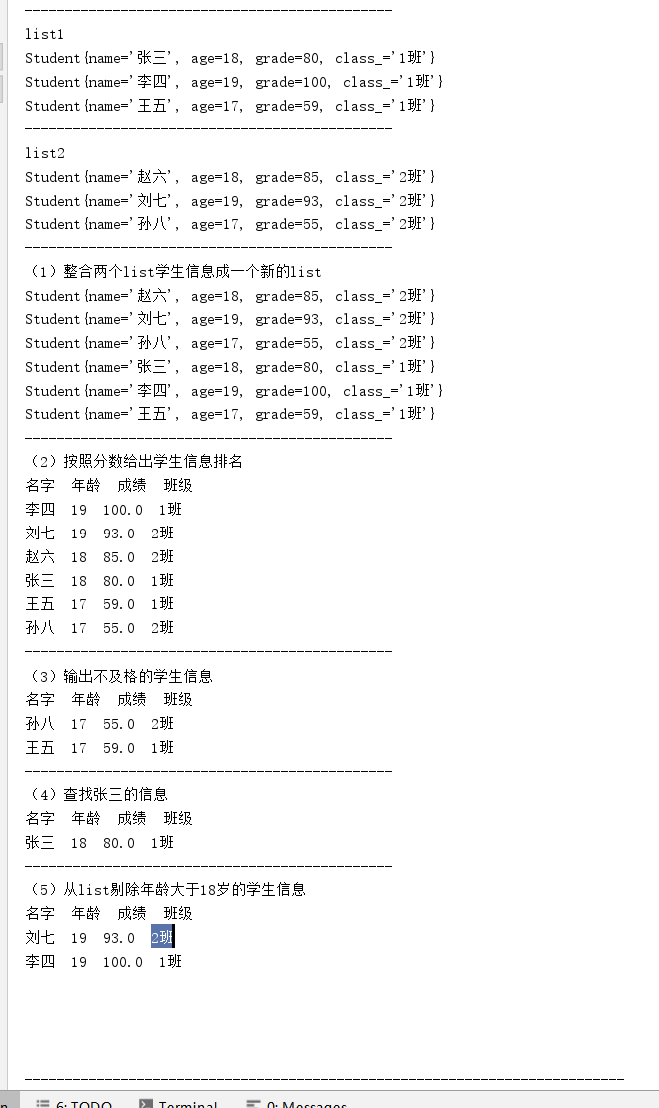
“赵六，18岁，85分，2班”，

“刘七，19岁，93分，2班”，

“孙八，17岁，55分，2班”。

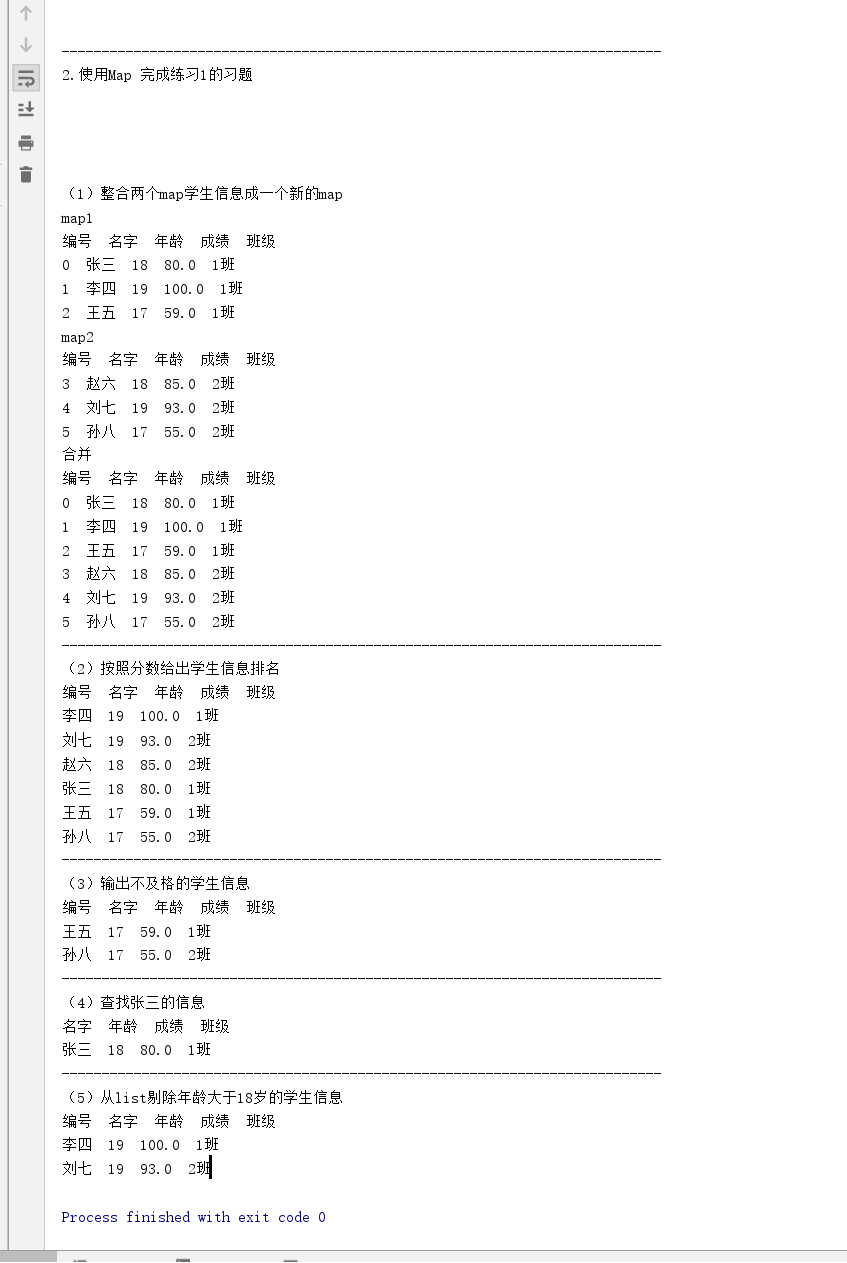
1. 整合两个list学生信息成一个新的list
2. 按照分数给出学生信息排名
3. 输出不及格的学生信息
4. 查找张三的信息
5. 从list剔除年龄大于18岁的学生信息

运行截图:



1. 使用Map 完成练习1的习题。

运行截图:



代码：

**import** java.util.ArrayList;  
**import** java.util.HashMap;  
**import** java.util.List;  
**import** java.util.Map;  
  
  
*/\*\*  
 \* 2020年3月4日作业  
 \*zhonghecheng  
 \* 2020年3月4日16:59:53  
 \*/***public class** HomeWork3\_4 {  
 **public static void** main(String[] args){  
 *problemOne*();  
 *problemTwo*();  
 }  
 */\*\*  
 \* 第一道题 1.建立一个实体类Student类，属性：姓名，年龄，成绩，班级  
 \* （1）整合两个list学生信息成一个新的list  
 \* （2）按照分数给出学生信息排名  
 \* （3）输出不及格的学生信息  
 \* （4）查找张三的信息  
 \* （5）从list剔除年龄大于18岁的学生信息  
 \*/* **public static void** problemOne(){  
 *//数据准备* String []name = **new** String[]{**"张三"**,**"李四"**,**"王五"**,**"赵六"**,**"刘七"**,**"孙八"**}; *//名字* **int** []age = **new int**[]{18,19,17,18,19,17}; *//年龄* **double** []grade = **new double** []{80,100,59,85,93,55}; *//成绩* String []class\_ = **new** String []{**"1班"**,**"1班"**,**"1班"**,**"2班"**,**"2班"**,**"2班"**}; *//成班级  
 //创建List 1 和 List2* List<Student> list1 = **new** ArrayList<>();  
 List<Student> list2 = **new** ArrayList<>();  
 *//增加数据* **for**(**int** i = 0;i<name.**length**;i++){  
 Student temp = **new** Student();  
 temp.setAge(age[i]);  
 temp.setClass\_(class\_[i]);  
 temp.setName(name[i]);  
 temp.setGrade(grade[i]);  
 **if**(i <name.**length**/2){  
 list1.add(temp);  
 }  
 **else**{  
 list2.add(temp);  
 }  
 *//打印测试  
// System.out.println(temp.toString());* }  
 *//（1）整合两个list学生信息成一个新的list\_new* List<Student> list\_new = **new** ArrayList<>();  
 list\_new.addAll(list2);  
 list\_new.addAll(list1);  
 System.***out***.println(**"----------------------------------------------"**);  
 System.***out***.println(**"list1"**);  
 *printList*(list1);  
 System.***out***.println(**"----------------------------------------------"**);  
 System.***out***.println(**"list2"**);  
 *printList*(list2);  
 System.***out***.println(**"----------------------------------------------"**);  
 System.***out***.println(**"（1）整合两个list学生信息成一个新的list"**);  
 *printList*(list\_new);  
 *//（2）按照分数给出学生信息排名* System.***out***.println(**"----------------------------------------------"**);  
 System.***out***.println(**"（2）按照分数给出学生信息排名"**);  
 *sortByGrade*(list\_new);  
 *//（3）输出不及格的学生信息* System.***out***.println(**"----------------------------------------------"**);  
 System.***out***.println(**"（3）输出不及格的学生信息"**);  
 *printNotAccept*(list\_new);  
 System.***out***.println(**"----------------------------------------------"**);  
 System.***out***.println(**"（4）查找张三的信息"**);  
 *findStudentData*(list\_new,**"张三"**);  
 System.***out***.println(**"----------------------------------------------"**);  
 System.***out***.println(**"（5）从list剔除年龄大于18岁的学生信息"**);  
 *deleteDataFromListByAge*(list\_new,18);  
 }  
  
 */\*\*  
 \* 2.使用Map 完成练习1的习题。  
 \*/* **public static void** problemTwo(){  
 *//数据准备* String []name = **new** String[]{**"张三"**,**"李四"**,**"王五"**,**"赵六"**,**"刘七"**,**"孙八"**}; *//名字* **int** []age = **new int**[]{18,19,17,18,19,17}; *//年龄* **double** []grade = **new double** []{80,100,59,85,93,55}; *//成绩* String []class\_ = **new** String []{**"1班"**,**"1班"**,**"1班"**,**"2班"**,**"2班"**,**"2班"**}; *//成班级  
  
 //创建Map* Map<Integer,Student>map1 = **new** HashMap<Integer,Student>();  
 Map<Integer,Student>map2 = **new** HashMap<Integer,Student>();  
 **for**(**int** i = 0;i<name.**length**;i++){  
 Student temp = **new** Student();  
 temp.setAge(age[i]);  
 temp.setClass\_(class\_[i]);  
 temp.setName(name[i]);  
 temp.setGrade(grade[i]);  
 **if**(i <name.**length**/2){  
 map1.put(i,temp);  
 }  
 **else**{  
 map2.put(i,temp);  
 }  
 *//打印测试  
// System.out.println(temp.toString());* }  
 System.***out***.println(**"\n\n\n"**);  
 System.***out***.println(**"---------------------------------------------------------------------------"**);  
 System.***out***.println(**"2.使用Map 完成练习1的习题"**);  
 System.***out***.println(**"\n\n\n"**);  
 System.***out***.println(**"（1）整合两个map学生信息成一个新的map"**);  
 System.***out***.println(**"map1"**);  
 *printMap*(map1);  
 System.***out***.println(**"map2"**);  
 *printMap*(map2);  
 Map<Integer,Student>map\_combine = **new** HashMap<>();  
 map\_combine.putAll(map1);  
 map\_combine.putAll(map2);  
 System.***out***.println(**"合并"**);  
 *printMap*(map\_combine);  
 System.***out***.println(**"---------------------------------------------------------------------------"**);  
 System.***out***.println(**"（2）按照分数给出学生信息排名"**);  
 *sortByGrade*(map\_combine);  
 System.***out***.println(**"---------------------------------------------------------------------------"**);  
 System.***out***.println(**"（3）输出不及格的学生信息"**);  
 *printNotAccept*(map\_combine);  
 System.***out***.println(**"---------------------------------------------------------------------------"**);  
 System.***out***.println(**"（4）查找张三的信息"**);  
 *findStudentData*(map\_combine,**"张三"**);  
 System.***out***.println(**"---------------------------------------------------------------------------"**);  
 System.***out***.println(**"（5）从list剔除年龄大于18岁的学生信息"**);  
 *deleteDataFromListByAge*(map\_combine,18);  
  
  
 }  
 */\*\*  
 \* 打印List  
 \*/* **public static void** printList(List<Student> list){  
 **for**(**int** i = 0;i<list.size();i++){  
 System.***out***.println(list.get(i).toString());  
 }  
 }  
 */\*\*  
 \* 根据分数给出排名  
 \*/* **public static void** sortByGrade(List<Student>list){  
 **int** temp = 0;  
 **double** min = -1;  
 System.***out***.println(**"名字 年龄 成绩 班级 "**);  
 List<Integer>count = **new** ArrayList<>();  
 **boolean** control = **false**;  
 **for**(**int** i = 0;i < list.size();i++){  
 temp = 0;  
  
 **for**(**int** j = 0;j < list.size();j++){  
 control = **false**;  
 *//先判断是否已经处理过* **for**(**int** k = 0;k < count.size();k++){  
 **if**(j == count.get(k)){  
 control = **true**;  
 **break**;  
 }  
 }  
 **if**(list.get(j).getGrade() > min && !control){  
 temp = j;  
 min = list.get(j).getGrade();  
 }  
 }  
 System.***out***.println(list.get(temp).getName()+**" "**+list.get(temp).getAge()+**" "**+list.get(temp).getGrade()+**" "**+list.get(temp).getClass\_());  
 *//处理标记* count.add(temp);  
 min = -1;  
 }  
 }  
 */\*\*  
 \* 输出不及格的同学  
 \*/* **public static void** printNotAccept(List<Student>list){  
 System.***out***.println(**"名字 年龄 成绩 班级 "**);  
 **for**(**int** i = 0;i< list.size();i++){  
 **if**(list.get(i).getGrade()<60){  
 System.***out***.println(list.get(i).getName()+**" "**+list.get(i).getAge()+**" "**+list.get(i).getGrade()+**" "**+list.get(i).getClass\_());  
 }  
 }  
  
 }  
 */\*\*  
 \* 查找同学信息  
 \** ***@param list*** *\** ***@param name*** *\* 2020年3月4日17:42:18  
 \*/* **public static void** findStudentData(List<Student>list,String name){  
 System.***out***.println(**"名字 年龄 成绩 班级 "**);  
 **for**(**int** i = 0;i< list.size();i++){  
 **if**(list.get(i).getName().equals(name)){  
 System.***out***.println(list.get(i).getName()+**" "**+list.get(i).getAge()+**" "**+list.get(i).getGrade()+**" "**+list.get(i).getClass\_());  
 }  
 }  
 }  
 */\*\*  
 \* 删除满足年龄信息  
 \** ***@param list*** *\** ***@param age*** *\* 2020年3月4日17:42:22  
 \*/* **public static void** deleteDataFromListByAge(List<Student>list,**int** age){  
 System.***out***.println(**"名字 年龄 成绩 班级 "**);  
 **for**(**int** i = 0;i< list.size();i++){  
 **if**(list.get(i).getAge()>age){  
 System.***out***.println(list.get(i).getName()+**" "**+list.get(i).getAge()+**" "**+list.get(i).getGrade()+**" "**+list.get(i).getClass\_());  
 }  
 }  
  
  
 }  
  
 */\*\*  
 \* 打印 Map  
 \*/* **public static void** printMap(Map<Integer,Student>map){  
 System.***out***.println(**"编号 名字 年龄 成绩 班级"**);  
 **for**(Map.Entry<Integer,Student>entry :map.entrySet()){  
 System.***out***.print(entry.getKey()+**" "**);  
 System.***out***.println(entry.getValue().getName()+**" "**+entry.getValue().getAge()+**" "**+entry.getValue().getGrade()+**" "**+entry.getValue().getClass\_());  
 }  
  
 }  
  
  
 */\*\*  
 \* map排序  
 \** ***@param map*** *\*/* **public static void** sortByGrade(Map<Integer,Student>map){  
 **int** temp = 0;  
 **double** min = -1;  
 System.***out***.println(**"编号 名字 年龄 成绩 班级 "**);  
 List<Integer>count = **new** ArrayList<>();  
 **boolean** control = **false**;  
 **for**(**int** i = 0;i < map.size();i++){  
 temp = 0;  
  
 **for**(**int** j = 0;j < map.size();j++){  
 control = **false**;  
 *//先判断是否已经处理过* **for**(**int** k = 0;k < count.size();k++){  
 **if**(j == count.get(k)){  
 control = **true**;  
 **break**;  
 }  
 }  
 **if**(map.get(j).getGrade() > min && !control){  
 temp = j;  
 min = map.get(j).getGrade();  
 }  
 }  
 System.***out***.println(map.get(temp).getName()+**" "**+map.get(temp).getAge()+**" "**+map.get(temp).getGrade()+**" "**+map.get(temp).getClass\_());  
 *//处理标记* count.add(temp);  
 min = -1;  
 }  
 }  
  
 */\*\*  
 \* 输出不及格的同学  
 \** ***@param map*** *\*/* **public static void** printNotAccept(Map<Integer,Student>map){  
 System.***out***.println(**"编号 名字 年龄 成绩 班级 "**);  
 **for**(**int** i = 0;i< map.size();i++){  
 **if**(map.get(i).getGrade()<60){  
 System.***out***.println(map.get(i).getName()+**" "**+map.get(i).getAge()+**" "**+map.get(i).getGrade()+**" "**+map.get(i).getClass\_());  
 }  
 }  
  
 }  
  
 */\*\*  
 \* 查找张三信息  
 \** ***@param list*** *\** ***@param name*** *\*/* **public static void** findStudentData(Map<Integer,Student>list,String name){  
 System.***out***.println(**"名字 年龄 成绩 班级 "**);  
 **for**(**int** i = 0;i< list.size();i++){  
 **if**(list.get(i).getName().equals(name)){  
 System.***out***.println(list.get(i).getName()+**" "**+list.get(i).getAge()+**" "**+list.get(i).getGrade()+**" "**+list.get(i).getClass\_());  
 }  
 }  
 }  
  
 */\*\*  
 \* 剔除18岁的学生  
 \** ***@param list*** *\** ***@param age*** *\*/* **public static void** deleteDataFromListByAge(Map<Integer,Student>list,**int** age){  
 System.***out***.println(**"编号 名字 年龄 成绩 班级 "**);  
 **for**(**int** i = 0;i< list.size();i++){  
 **if**(list.get(i).getAge()>age){  
 System.***out***.println(list.get(i).getName()+**" "**+list.get(i).getAge()+**" "**+list.get(i).getGrade()+**" "**+list.get(i).getClass\_());  
 }  
 }  
  
  
 }  
}

3.仿照手机淘宝，设计订单和商品的实体类。

截图：



代码：

**import** java.util.Arrays;  
  
*/\*\*  
 \* 需求：  
 \* 仿照手机淘宝，设计订单和商品的实体类。  
 \*  
 \*  
 \* 以下是 购物订单  
 \*  
 \* 分析:  
 \*  
 \* 订单 有如下属性:  
 \* 订单编号 创建订单的用户ID 创建订单的用户昵称 订单总价格 订单包含的每个商品名称、分别对应商品的数量、商品图片、商品描述、每种商品ID编号 订单中每件商品的价格 订单创建时间  
 \*  
 \*  
 \* zhonghecheng  
 \* 2020年3月4日18:11:46  
 \*  
 \*  
 \*  
 \*/***public class** HomeWork3\_4\_2 {  
 **private** String **orderNumber**;*//订单编号 一般式md5码* **private** String **userID**; *// 创建订单的用户ID* **private** String **createOrderNickname** ;*//创建订单的用户昵称* **private double allPrice**; *//总价* **private** String []**eachGoodName**; *// 订单包含的每个商品名称* **private int** []**eachGoodNum**; *// 、分别对应商品的数量* **private** String []**eachGoodImageURL**;*//、商品图片* **private** String []**eachGoodScription**;*//、商品描述、* **private** String []**eachGoodOrderNumber**;*//每种商品ID编号* **private double**[]**eachGoodPrice**;*// 订单中每件商品的价格* **private** String **createTime**;*// 订单创建时间* @Override  
 **public** String toString() {  
 **return "HomeWork3\_4\_2{"** +  
 **"orderNumber='"** + **orderNumber** + **'\''** +  
 **", userID='"** + **userID** + **'\''** +  
 **", createOrderNickname='"** + **createOrderNickname** + **'\''** +  
 **", allPrice="** + **allPrice** +  
 **", eachGoodName="** + Arrays.*toString*(**eachGoodName**) +  
 **", eachGoodNum="** + Arrays.*toString*(**eachGoodNum**) +  
 **", eachGoodImageURL="** + Arrays.*toString*(**eachGoodImageURL**) +  
 **", eachGoodScription="** + Arrays.*toString*(**eachGoodScription**) +  
 **", eachGoodOrderNumber="** + Arrays.*toString*(**eachGoodOrderNumber**) +  
 **", eachGoodPrice="** + Arrays.*toString*(**eachGoodPrice**) +  
 **", createTime='"** + **createTime** + **'\''** +  
 **'}'**;  
 }  
  
 **public** String getOrderNumber() {  
 **return orderNumber**;  
 }  
  
 **public void** setOrderNumber(String orderNumber) {  
 **this**.**orderNumber** = orderNumber;  
 }  
  
 **public** String getUserID() {  
 **return userID**;  
 }  
  
 **public void** setUserID(String userID) {  
 **this**.**userID** = userID;  
 }  
  
 **public** String getCreateOrderNickname() {  
 **return createOrderNickname**;  
 }  
  
 **public void** setCreateOrderNickname(String createOrderNickname) {  
 **this**.**createOrderNickname** = createOrderNickname;  
 }  
  
 **public double** getAllPrice() {  
 **return allPrice**;  
 }  
  
 **public void** setAllPrice(**double** allPrice) {  
 **this**.**allPrice** = allPrice;  
 }  
  
 **public** String[] getEachGoodName() {  
 **return eachGoodName**;  
 }  
  
 **public void** setEachGoodName(String[] eachGoodName) {  
 **this**.**eachGoodName** = eachGoodName;  
 }  
  
 **public int**[] getEachGoodNum() {  
 **return eachGoodNum**;  
 }  
  
 **public void** setEachGoodNum(**int**[] eachGoodNum) {  
 **this**.**eachGoodNum** = eachGoodNum;  
 }  
  
 **public** String[] getEachGoodImageURL() {  
 **return eachGoodImageURL**;  
 }  
  
 **public void** setEachGoodImageURL(String[] eachGoodImageURL) {  
 **this**.**eachGoodImageURL** = eachGoodImageURL;  
 }  
  
 **public** String[] getEachGoodScription() {  
 **return eachGoodScription**;  
 }  
  
 **public void** setEachGoodScription(String[] eachGoodScription) {  
 **this**.**eachGoodScription** = eachGoodScription;  
 }  
  
 **public** String[] getEachGoodOrderNumber() {  
 **return eachGoodOrderNumber**;  
 }  
  
 **public void** setEachGoodOrderNumber(String[] eachGoodOrderNumber) {  
 **this**.**eachGoodOrderNumber** = eachGoodOrderNumber;  
 }  
  
 **public double**[] getEachGoodPrice() {  
 **return eachGoodPrice**;  
 }  
  
 **public void** setEachGoodPrice(**double**[] eachGoodPrice) {  
 **this**.**eachGoodPrice** = eachGoodPrice;  
 }  
  
 **public** String getCreateTime() {  
 **return createTime**;  
 }  
  
 **public void** setCreateTime(String createTime) {  
 **this**.**createTime** = createTime;  
 }  
}

商品实体类:



代码:

**import** java.util.Arrays;  
*/\*\*  
 \* 需求：  
 \* 仿照手机淘宝，设计订单和商品的实体类。  
 \* 以下是 商品  
 \* 分析:  
 \* 商品 有如下属性:[其他属性在其他表,比如商品有哪些颜色分类什么的]  
 \* 商品编号 商品名称 商品图片 商品描述 商品价格 付款人数 商品优惠 开卖时间 月销 商品库存 是否限制买的数量 创建作者 创建时间  
 \* zhonghecheng  
 \* 2020年3月4日18:11:46  
 \*  
 \*  
 \*  
 \*/***public class** HomeWork3\_4\_2 {  
 **private** String **goodNumber**;*//订单编号 一般式md5码* **private** String **createOrderNickname** ;*//创建订单的用户昵称* **private** String []**goodImageURL**;*//、商品图片* **private** String **goodScription**;*//、商品描述、* **private int buyPeopleNum**;*//付款人数* **private int display**;*//打折* **private** String **saleTime**;*//开卖时间* **private int saleNum**;*//卖出去的数量* **private double price**;*//价格* **private int save**;*//库存* **private int limitNum**;*//限制买几件* **private** String **author**;*//创建作者* **private** String **createTime**;*// 创建时间* @Override  
 **public** String toString() {  
 **return "HomeWork3\_4\_2{"** +  
 **"goodNumber='"** + **goodNumber** + **'\''** +  
 **", createOrderNickname='"** + **createOrderNickname** + **'\''** +  
 **", goodImageURL="** + Arrays.*toString*(**goodImageURL**) +  
 **", goodScription='"** + **goodScription** + **'\''** +  
 **", buyPeopleNum="** + **buyPeopleNum** +  
 **", display="** + **display** +  
 **", saleTime='"** + **saleTime** + **'\''** +  
 **", saleNum="** + **saleNum** +  
 **", price="** + **price** +  
 **", save="** + **save** +  
 **", limitNum="** + **limitNum** +  
 **", author='"** + **author** + **'\''** +  
 **", createTime='"** + **createTime** + **'\''** +  
 **'}'**;  
 }  
  
 **public** String getGoodNumber() {  
 **return goodNumber**;  
 }  
  
 **public void** setGoodNumber(String goodNumber) {  
 **this**.**goodNumber** = goodNumber;  
 }  
  
 **public** String getCreateOrderNickname() {  
 **return createOrderNickname**;  
 }  
  
 **public void** setCreateOrderNickname(String createOrderNickname) {  
 **this**.**createOrderNickname** = createOrderNickname;  
 }  
  
 **public** String[] getGoodImageURL() {  
 **return goodImageURL**;  
 }  
  
 **public void** setGoodImageURL(String[] goodImageURL) {  
 **this**.**goodImageURL** = goodImageURL;  
 }  
  
 **public** String getGoodScription() {  
 **return goodScription**;  
 }  
  
 **public void** setGoodScription(String goodScription) {  
 **this**.**goodScription** = goodScription;  
 }  
  
 **public int** getBuyPeopleNum() {  
 **return buyPeopleNum**;  
 }  
  
 **public void** setBuyPeopleNum(**int** buyPeopleNum) {  
 **this**.**buyPeopleNum** = buyPeopleNum;  
 }  
  
 **public int** getDisplay() {  
 **return display**;  
 }  
  
 **public void** setDisplay(**int** display) {  
 **this**.**display** = display;  
 }  
  
 **public** String getSaleTime() {  
 **return saleTime**;  
 }  
  
 **public void** setSaleTime(String saleTime) {  
 **this**.**saleTime** = saleTime;  
 }  
  
 **public int** getSaleNum() {  
 **return saleNum**;  
 }  
  
 **public void** setSaleNum(**int** saleNum) {  
 **this**.**saleNum** = saleNum;  
 }  
  
 **public double** getPrice() {  
 **return price**;  
 }  
  
 **public void** setPrice(**double** price) {  
 **this**.**price** = price;  
 }  
  
 **public int** getSave() {  
 **return save**;  
 }  
  
 **public void** setSave(**int** save) {  
 **this**.**save** = save;  
 }  
  
 **public int** getLimitNum() {  
 **return limitNum**;  
 }  
  
 **public void** setLimitNum(**int** limitNum) {  
 **this**.**limitNum** = limitNum;  
 }  
  
 **public** String getAuthor() {  
 **return author**;  
 }  
  
 **public void** setAuthor(String author) {  
 **this**.**author** = author;  
 }  
  
 **public** String getCreateTime() {  
 **return createTime**;  
 }  
  
 **public void** setCreateTime(String createTime) {  
 **this**.**createTime** = createTime;  
 }  
}