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

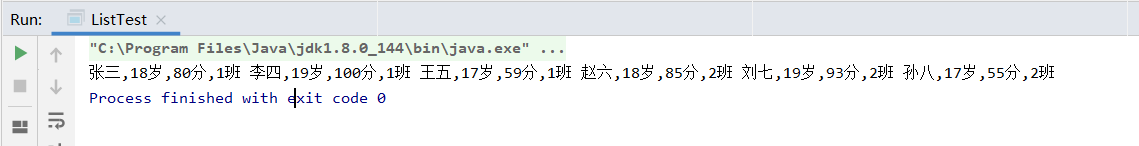
建立一个list1，包含 “张三，18岁，80分，1班”，“李四，19岁，100分，1班”，“王五，17岁，59分，1班”。

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

*//定义list1和list2集合*List <Student> list1 = **new** ArrayList<>();  
List <Student> list2 = **new** ArrayList<>();  
*//添加数据*list1.add(**new** Student(**"张三"**,18,80,**"1班"**));  
list1.add(**new** Student(**"李四"**,19,100,**"1班"**));  
list1.add(**new** Student(**"王五"**,17,59,**"1班"**));  
list2.add(**new** Student(**"赵六"**,18,85,**"2班"**));  
list2.add(**new** Student(**"刘七"**,19,93,**"2班"**));  
list2.add(**new** Student(**"孙八"**,17,55,**"2班"**));  
*//定义listend总集合*List <Student> list = **new** ArrayList<>();  
list1.addAll(list2);  
list = list1;  
*//打印所有学生的信息***for**(**int** x = 0;x < list.size(); x++){  
 System.***out***.println(list.get(x).getName()+**","**+list.get(x).getAge()+**"岁,"**+list.get(x).getAchievement()+**"分,"**+list.get(x).getClasses()+**" "**);  
}

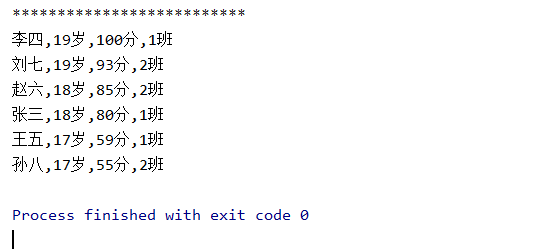
输出效果如下：



1. 按照分数给出学生信息排名

*//按分数进行排序并输出学生信息  
//使用冒泡排序进行排序***for**(**int** x = 0;x<list.size(); x++){  
 **for** (**int** y = 0;y < list.size()-1; y++){  
 **if**(list.get(y).getAchievement()<list.get(y+1).getAchievement()){  
 Student center = list.get(y);  
 list.set(y,list.get(y+1));  
 list.set(y+1,center);  
 }  
 }  
}  
*//输出***for**(**int** x = 0;x < list.size(); x++){  
 System.***out***.println(list.get(x).getName()+**","**+list.get(x).getAge()+**"岁,"**+list.get(x).getAchievement()+**"分,"**+list.get(x).getClasses()+**" "**);  
}

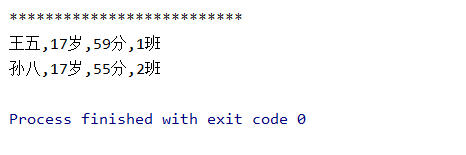
输出结果如下：



1. 输出不及格的学生信息

*//输出不及格的同学信息***for**(**int** x = 0;x < list.size(); x++){  
 **if**(list.get(x).getAchievement()<60){  
 System.***out***.println(list.get(x).getName()+**","**+list.get(x).getAge()+**"岁,"**+list.get(x).getAchievement()+**"分,"**+list.get(x).getClasses()+**" "**);  
 }  
}

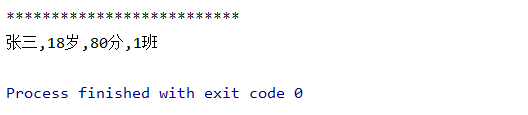
输出结果如下：



1. 查找张三的信息

*//查找张三的信息***for**(**int** x = 0;x < list.size(); x++){  
 **if**(list.get(x).getName()==**"张三"**){  
 System.***out***.println(list.get(x).getName()+**","**+list.get(x).getAge()+**"岁,"**+list.get(x).getAchievement()+**"分,"**+list.get(x).getClasses()+**" "**);  
 }  
}

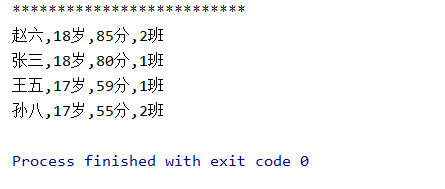
输出结果如下：



1. 从list剔除年龄大于18岁的学生信息

*//删除年龄大于18岁的学生***for**(**int** x = 0;x < list.size(); x++){  
 **if**(list.get(x).getAge()>18){  
 list.remove(list.get(x));  
 }**else** {  
 System.***out***.println(list.get(x).getName()+**","**+list.get(x).getAge()+**"岁,"**+list.get(x).getAchievement()+**"分,"**+list.get(x).getClasses()+**" "**);  
 }  
}

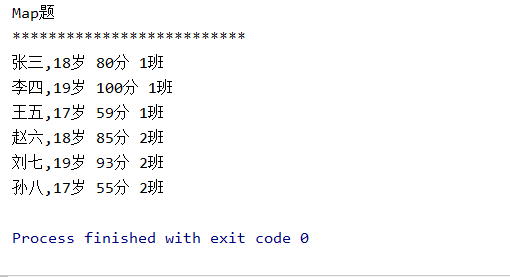
输出结果如下：



1. 使用Map 完成练习1的习题。
2. 整合两个list学生信息成一个新的list

*//整合两个list学生信息成一个新的list  
//定义map1和map2*Map <Integer,Student> map1 = **new** HashMap<Integer, Student>();  
Map <Integer,Student> map2 = **new** HashMap<Integer, Student>();  
map1.put(1,**new** Student(**"张三"**,18,80,**"1班"**));  
map1.put(2,**new** Student(**"李四"**,19,100,**"1班"**));  
map1.put(3,**new** Student(**"王五"**,17,59,**"1班"**));  
map2.put(4,**new** Student(**"赵六"**,18,85,**"2班"**));  
map2.put(5,**new** Student(**"刘七"**,19,93,**"2班"**));  
map2.put(6,**new** Student(**"孙八"**,17,55,**"2班"**));  
*//定义总的map*Map <Integer,Student> map = **new** HashMap<Integer, Student>();  
map1.putAll(map2);  
*//输出所有信息***for**(Map.Entry<Integer,Student>entry:map1.entrySet()){  
 System.***out***.println(entry.getValue().getName()+**","**+entry.getValue().getAge()+**"岁 "**+entry.getValue().getAchievement()+**"分 "**+entry.getValue().getClasses());  
}

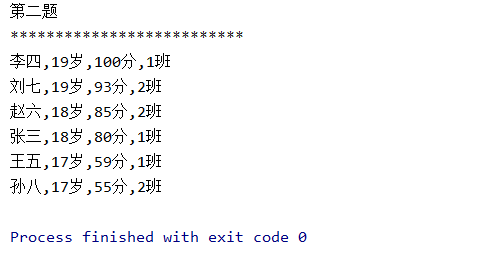
输出结果如下：



1. 按照分数给出学生信息排名

*//根据分数对学生进行排序  
//定义一个list,将map转化为list*List <Student> result = **new** ArrayList<>(map.values());  
*//使用冒泡排序进行排序***for**(**int** x = 0;x<result.size(); x++){  
 **for** (**int** y = 0;y < result.size()-1; y++){  
 **if**(result.get(y).getAchievement()<result.get(y+1).getAchievement()){  
 Student centertwo = result.get(y);  
 result.set(y,result.get(y+1));  
 result.set(y+1,centertwo);  
 }  
 }  
}  
**for**(**int** x = 0;x < result.size(); x++){  
 System.***out***.println(result.get(x).getName()+**","**+result.get(x).getAge()+**"岁,"**+result.get(x).getAchievement()+**"分,"**+result.get(x).getClasses()+**" "**);  
}

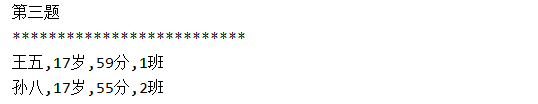
输出结果如下：



1. 输出不及格的学生信息

*//输出不及格信息  
//使用上面转化所得的result集合进行计算并输出***for**(**int** x = 0;x < result.size(); x++){  
 **if**(result.get(x).getAchievement()<60){  
 System.***out***.println(result.get(x).getName()+**","**+result.get(x).getAge()+**"岁,"**+result.get(x).getAchievement()+**"分,"**+result.get(x).getClasses()+**" "**);  
 }  
}

输出结果如下：



1. 查找张三的信息

*//查找张三的信息  
//使用上面转化所得的result集合筛选出张三信息***for**(**int** x = 0;x < result.size(); x++){  
 **if**(result.get(x).getName()==**"张三"**){  
 System.***out***.println(result.get(x).getName()+**","**+result.get(x).getAge()+**"岁,"**+result.get(x).getAchievement()+**"分,"**+result.get(x).getClasses()+**" "**);  
 }  
}

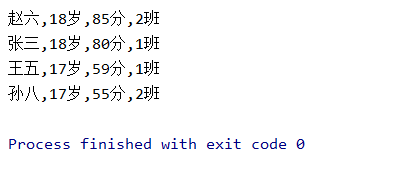
输出结果如下：



1. 从list剔除年龄大于18岁的学生信息

*//删除年龄大于18岁的学生  
//使用上面转化所得的result集合进行筛选大于18岁的学生信息并输出***for**(**int** x = 0;x < result.size(); x++){  
 **if**(result.get(x).getAge()>18){  
 result.remove(result.get(x));  
 }**else** {  
 System.***out***.println(result.get(x).getName()+**","**+result.get(x).getAge()+**"岁,"**+result.get(x).getAchievement()+**"分,"**+result.get(x).getClasses()+**" "**);  
 }  
}

输出结果如下：



1. 仿照手机淘宝，设计订单和商品的实体类。

**class** Order{  
 **private** String **orderId**; *//订单编号* **private** String **tradeId**; *//商品编号* **private** String **tradeName**; *//商品名称* **private** String **business**; *//卖家* **private** String **buyer**; *//买家* **private int buyNumber**; *//购买数量* **private int totalSum**; *//总金额* **private** String **buyTime**; *//购买时间* **private** String **logistics**; *//物流信息* **public** Order(){}  
  
 **public void** setOrderId(String orderId){  
 **this**.**orderId** = orderId;  
 }  
 **public void** setTradeId(String tradeId){  
 **this**.**tradeId** = tradeId;  
 }  
 **public void** setTradeName(String tradeName){  
 **this**.**tradeName** = tradeName;  
 }  
 **public void** setBusiness(String business){  
 **this**.**business** = business;  
 }  
 **public void** setBuyer(String buyer){  
 **this**.**buyer** = buyer;  
 }  
 **private int** getBuyNumber(){  
 **return buyNumber**;  
 }  
 **private int** getTotalSum(){  
 **return totalSum**;  
 }  
 **public void** setBuyNumber(**int** buyNumber){  
 **this**.**buyNumber** = buyNumber;  
 }  
 **public void** setTotalSum(**int** totalSum){  
 **this**.**totalSum** = totalSum;  
 }  
 **public void** setBuyTime(String buyTime){  
 **this**.**buyTime** = buyTime;  
 }  
 **public void** setLogistics(String logistics){  
 **this**.**logistics** = logistics;  
 }  
 **private** String getOrderId(){  
 **return orderId**;  
 }  
 **private** String getTradeId(){  
 **return tradeId**;  
 }  
 **private** String getTradeName(){  
 **return tradeName**;  
 }  
 **private** String getBusiness(){  
 **return business**;  
 }  
 **private** String getBuyer(){  
 **return buyer**;  
 }  
 **private** String getBuyTime(){  
 **return buyTime**;  
 }  
 **private** String getLogistics(){  
 **return logistics**;  
 }  
  
}  
**class** Commodity{  
 **private** String **tradeId**; *//商品编号* **private** String **tradeName**; *//商品名称* **private** String **tradeDescribe**; *//商品详情* **private int tradePrice**; *//商品价格* **private** String **business**; *//卖家* **private int stock**; *//库存* **private** String **evaluate**; *//评价* **private** String **relevantTrade**; *//相关商品* **public** Commodity(){}  
  
 **private void** setTradeId(String tradeId){  
 **this**.**tradeId** = tradeId;  
 }  
 **private void** setTradeName(String tradeName){  
 **this**.**tradeName** = tradeName;  
 }  
 **private void** setTradeDescribe(String tradeDescribe){  
 **this**.**tradeDescribe** = tradeDescribe;  
 }  
 **private void** setTradePrice(**int** tradePrice){  
 **this**.**tradePrice** = tradePrice;  
 }  
 **private void** setBusiness(String business){  
 **this**.**business** = business;  
 }  
 **private void** setStock(**int** stock){  
 **this**.**stock** = stock;  
 }  
 **private void** setEvaluate(String evaluate){  
 **this**.**evaluate** = evaluate;  
 }  
 **private void** setRelevantTrade(String relevantTrade){  
 **this**.**relevantTrade** = relevantTrade;  
 }  
 **private** String getTradeId(){  
 **return tradeId**;  
 }  
 **private** String getTradeName(){  
 **return tradeName**;  
 }  
 **private** String getTradeDescribe(){  
 **return tradeDescribe**;  
 }  
 **private int** getTradePrice(){  
 **return tradePrice**;  
 }  
 **private** String getBusiness(){  
 **return business**;  
 }  
 **private** String getEvaluate(){  
 **return evaluate**;  
 }  
 **private** String getRelevantTrade(){  
 **return relevantTrade**;  
 }  
}