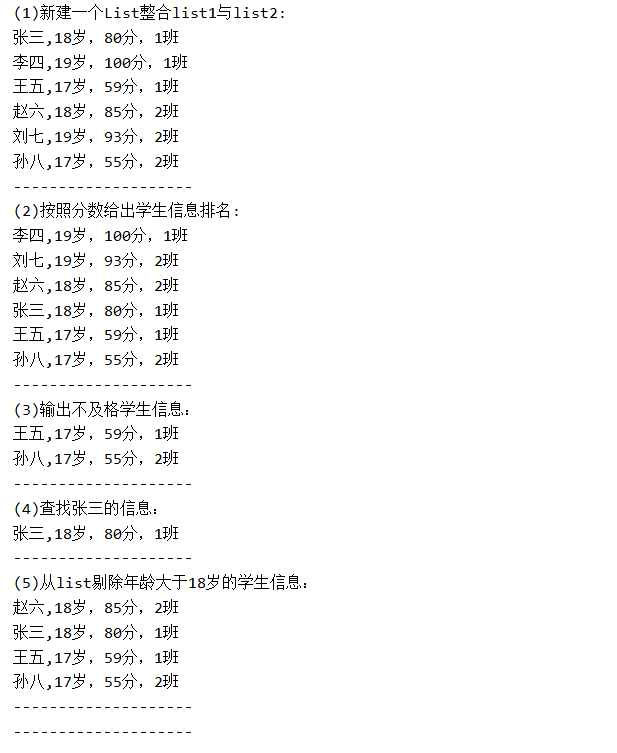
第一题代码：（list练习）

Student类：

**package** homework;  
  
**public class** Student {  
  
 */\*\*  
 \** ***@param Student类*** *\* lidantao  
 \* 2020.03.04 20:22  
 \*/  
  
 /\*\*  
 \* 姓名  
 \*/* **private** String **name**;  
 */\*\*  
 \* 年龄  
 \*/* **private int age**;  
 */\*\*  
 \* 成绩  
 \*/* **private int achievement**;  
  
 */\*\*  
 \* 班级  
 \*/* **private** String **room**;  
  
 **public** String getName() {  
 **return name**;  
 }  
  
 **public void** setName(String name) {  
 **this**.**name** = name;  
 }  
  
 **public int** getAge() {  
 **return age**;  
 }  
  
 **public void** setAge(**int** age) {  
 **this**.**age** = age;  
 }  
  
 **public int** getAchievement() {  
 **return achievement**;  
 }  
  
 **public void** setAchievement(**int** achievement) {  
 **this**.**achievement** = achievement;  
 }  
  
 **public** String getRoom() {  
 **return room**;  
 }  
  
 **public void** setRoom(String room) {  
 **this**.**room** = room;  
 }  
  
 @Override  
 **public** String toString() {  
 **return name** + **","** + **age** + **"岁，"** + **achievement** + **"分，"** + **room**;  
 }  
}

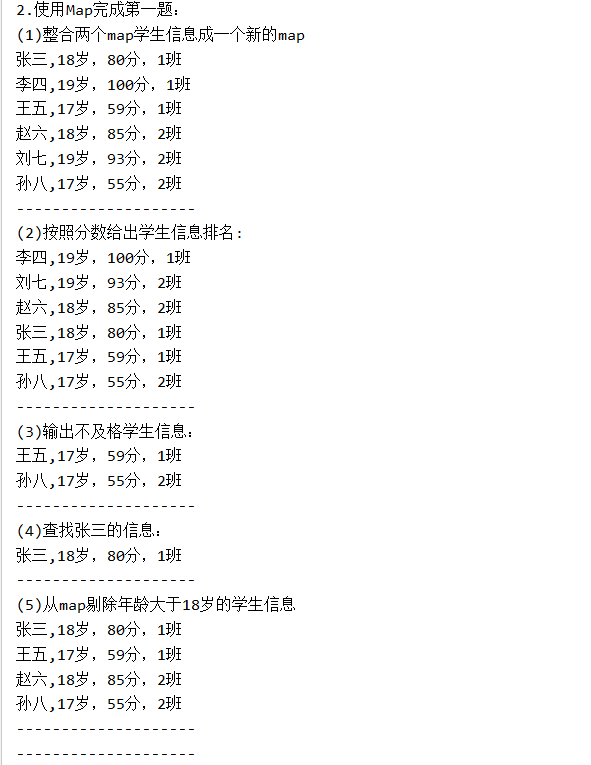
**package** homework;  
  
**import** javax.crypto.spec.PSource;  
**import** java.util.\*;  
  
**public class** MyTask {  
 **public static** <MAX\_ACCHIEVEMENT> **void** main(String[] args) {  
 */\*\*  
 \** ***@param list练习*** *\* lidantao  
 \* 2020.03.04 20:22  
 \*/  
  
 /\*\*  
 \* new 三个类，设置其属性  
 \*/* Student one = **new** Student();  
 one.setName(**"张三"**);  
 one.setAge(18);  
 one.setAchievement(80);  
 one.setRoom(**"1班"**);  
  
 Student two = **new** Student();  
 two.setName(**"李四"**);  
 two.setAge(19);  
 two.setAchievement(100);  
 two.setRoom(**"1班"**);  
  
 Student three = **new** Student();  
 three.setName(**"王五"**);  
 three.setAge(17);  
 three.setAchievement(59);  
 three.setRoom(**"1班"**);  
  
 */\*\*  
 \* 建立一个list1，包含以上三个类  
 \*/* List<Student> list1 = **new** ArrayList<>();  
 list1.add(one);  
 list1.add(two);  
 list1.add(three);  
  
 */\*\*  
 \* new 三个类，设置其属性  
 \*/* Student four = **new** Student();  
 four.setName(**"赵六"**);  
 four.setAge(18);  
 four.setAchievement(85);  
 four.setRoom(**"2班"**);  
  
 Student five = **new** Student();  
 five.setName(**"刘七"**);  
 five.setAge(19);  
 five.setAchievement(93);  
 five.setRoom(**"2班"**);  
  
 Student six = **new** Student();  
 six.setName(**"孙八"**);  
 six.setAge(17);  
 six.setAchievement(55);  
 six.setRoom(**"2班"**);  
  
 */\*\*  
 \* 建立一个list2，包含以上三个类  
 \*/* List<Student> list2 = **new** ArrayList<>();  
 list1.add(four);  
 list1.add(five);  
 list1.add(six);  
  
 */\*\*  
 \* (1)新建一个List整合list1与list2  
 \*/* List<Student> list = **new** ArrayList<>();  
 */\*\*  
 \* 使用增强for循环加入到list中去  
 \*/* **for**(Student s : list1) {  
 list.add(s);  
 }  
 **for**(Student s : list2){  
 list.add(s);  
 }  
 */\*\*  
 \* 整合成一个list后的情况输出  
 \*/* System.***out***.println(**"(1)新建一个List整合list1与list2:"**);  
 **for**(Student s : list){  
 System.***out***.println(s.toString());  
 }  
 System.***out***.println(**"--------------------"**);  
  
 */\*\*  
 \* (2)按照分数给出学生信息排名，使用Collections类的sort方法，以及lambda优化代码  
 \* 年龄从小到大排序  
 \*/* Collections.*sort*(list,(Student s1, Student s2)->{**return** s2.getAchievement() - s1.getAchievement();});  
 */\*\*  
 \* 使用增强for循环输出排序后内容  
 \*/* System.***out***.println(**"(2)按照分数给出学生信息排名:"**);  
 **for**(Student s : list){  
 System.***out***.println(s.toString());  
 }  
 System.***out***.println(**"--------------------"**);  
  
 */\*\*  
 \* 使用增强for循环输出不及格学生信息  
 \*/* System.***out***.println(**"(3)输出不及格学生信息："**);  
 **for**(Student s : list){  
 **if**(s.getAchievement() >= 60 ){  
 *// 如果分数大于60分就跳过循环* **continue**;  
 }  
 System.***out***.println(s.toString());  
 }  
 System.***out***.println(**"--------------------"**);  
  
 */\*\*  
 \* 查找张三的信息  
 \*/* System.***out***.println(**"(4)查找张三的信息："**);  
 **for**(Student s : list){  
 **if**(**"张三"**.equals(s.getName())){  
 *// 如果姓名等于张三就输出* System.***out***.println(s.toString());  
 }  
 }  
 System.***out***.println(**"--------------------"**);  
  
 */\*\*  
 \* 从list剔除年龄大于18岁的学生信息  
 \*/* System.***out***.println(**"(5)从list剔除年龄大于18岁的学生信息："**);  
 **for** (**int** i = 0; i < list.size(); i++) {  
 **if**(list.get(i).getAge() > 18){  
 *// 年龄大于十八就删除* list.remove(i);  
 i--; *// 移除后列表长度会 -1 ，同步避免跳过出错* }  
 }  
 **for** (**int** i = 0; i < list.size(); i++) {  
 System.***out***.println(list.get(i).toString());  
 }  
 System.***out***.println(**"--------------------"**);  
 System.***out***.println(**"--------------------"**);  
  
 }  
}

运行结果截图：

第二题代码（map练习）：

**package** homework;  
  
**import** java.util.ArrayList;  
**import** java.util.HashMap;  
**import** java.util.List;  
**import** java.util.Map;  
  
**public class** MapTask {  
 **public static void** main(String[] args) {  
 */\*\*  
 \** ***@param map*** *练习  
 \* lidantao  
 \* 2020.03.04 20:22  
 \*/  
  
 /\*\*  
 \* new 六个类，设置其属性  
 \*/* Student one = **new** Student();  
 one.setName(**"张三"**);  
 one.setAge(18);  
 one.setAchievement(80);  
 one.setRoom(**"1班"**);  
  
 Student two = **new** Student();  
 two.setName(**"李四"**);  
 two.setAge(19);  
 two.setAchievement(100);  
 two.setRoom(**"1班"**);  
  
 Student three = **new** Student();  
 three.setName(**"王五"**);  
 three.setAge(17);  
 three.setAchievement(59);  
 three.setRoom(**"1班"**);  
  
 Student four = **new** Student();  
 four.setName(**"赵六"**);  
 four.setAge(18);  
 four.setAchievement(85);  
 four.setRoom(**"2班"**);  
  
 Student five = **new** Student();  
 five.setName(**"刘七"**);  
 five.setAge(19);  
 five.setAchievement(93);  
 five.setRoom(**"2班"**);  
  
 Student six = **new** Student();  
 six.setName(**"孙八"**);  
 six.setAge(17);  
 six.setAchievement(55);  
 six.setRoom(**"2班"**);  
 System.***out***.println(**"2.使用Map完成第一题："**);  
 */\*\*  
 \* 新建一个map1,将1中三个类的属性加进去  
 \*/* Map<Integer,Student> map1 = **new** HashMap<>();  
 map1.put(1,one);  
 map1.put(2,two);  
 map1.put(3,three);  
 */\*\*  
 \* 新建一个map2,将1中三个类的属性加进去  
 \*/* Map<Integer,Student> map2 = **new** HashMap<>();  
 map2.put(4,four);  
 map2.put(5,five);  
 map2.put(6,six);  
 */\*\*  
 \* 新建一个map将map1与map2整合一起  
 \*/* Map<Integer,Student> map = **new** HashMap<>();  
 **for** (**int** entry : map1.keySet()){  
 map.put(entry, map1.get(entry));  
 }  
 **for** (**int** entry : map2.keySet()){  
 map.put(entry, map2.get(entry));  
 }  
 */\*\*  
 \* 整合后的结果  
 \*/* System.***out***.println(**"(1)整合两个map学生信息成一个新的map"**);  
 **for** (**int** entry : map.keySet()){  
 System.***out***.println(map.get(entry).toString());  
 }  
 System.***out***.println(**"--------------------"**);  
  
 */\*\*  
 \* 定义数组用来标志是否遍历过  
 \*/* **int**[] arr = **new int**[map.size()];  
 **for** (**int** i = 0; i < map.size(); i++) {  
 arr[i] = 0; *// 设置都为遍历过* }  
 */\*\*  
 \* 定义一个新的map用来收集排序后的map  
 \*/* Map<Integer,Student> new\_map = **new** HashMap<>();  
 */\*\*  
 \* 定义新的map下标key，初始化为0  
 \* 定义最大值索引，初始化为0  
 \* 定义最大值，初始化-1  
 \*/* **int** key = 0, index = 0, max = -1;  
 */\*\*  
 \* 定义新的map的value，初始化null  
 \*/* Student value = **null**;  
  
 *// 使用增强for循环查找最大值* **for** (**int** entry1 : map.keySet()){  
 *// 标志当前在哪个元素对比* **int** j = 0;  
 **for** (**int** entry2 : map.keySet()) {  
 *// 如果当前的分数大于max 且此类没被遍历过（已收集）就进入* **if**(map.get(entry2).getAchievement() >= max && arr[j] != 1){  
 *// 让最大值变为此元素* max = map.get(entry2).getAchievement();  
 *// 提取值* value = map.get(entry2);  
 *// 最大值索引为j* index = j;  
 }  
 *// 索引+1* j++;  
 }  
 *// 将最大值放进新的map* new\_map.put(key++,value);  
 *// 最大值的类标志为已遍历* arr[index] = 1;  
 *// 最大值变为-1* max = -1;  
 }  
 System.***out***.println(**"(2)按照分数给出学生信息排名:"**);  
 **for** (**int** entry1 : new\_map.keySet()) {  
 System.***out***.println(new\_map.get(entry1).toString());  
 }  
 System.***out***.println(**"--------------------"**);  
  
 */\*\*  
 \* 输出不及格学生信息  
 \*/* System.***out***.println(**"(3)输出不及格学生信息："**);  
 **for**(**int** entry : map.keySet()){  
 *// 成绩小于60分则进入* **if**(map.get(entry).getAchievement() < 60){  
 System.***out***.println(map.get(entry).toString());  
 }  
 }  
 System.***out***.println(**"--------------------"**);  
  
 */\*\*  
 \* 查找张三的信息  
 \*/* System.***out***.println(**"(4)查找张三的信息："**);  
 **for**(**int** entry : map.keySet()){  
 *// 成绩小于60分则进入* **if**(**"张三"**.equals(map.get(entry).getName())){  
 System.***out***.println(map.get(entry).toString());  
 }  
 }  
 System.***out***.println(**"--------------------"**);  
  
 */\*\*  
 \* 从map剔除年龄大于18岁的学生信息  
 \*/* System.***out***.println(**"(5)从map剔除年龄大于18岁的学生信息"**);  
 **for** (**int** i = 0; i < map.size(); i++){  
 **for** (**int** entry : map.keySet()){  
 **if** (map.get(entry).getAge() > 18){  
 map.remove(entry);  
 **break**;  
 }  
 }  
 }  
 **for** (**int** entry : map.keySet()){  
 System.***out***.println(map.get(entry).toString());  
 }  
 System.***out***.println(**"--------------------"**);  
 System.***out***.println(**"--------------------"**);  
 }  
}

运行截图：



第三题：

订单实体类

**package** homework;  
  
**public class** TaoBaoPaying {  
 */\*\*  
 \* lidantao  
 \* 2020.03.04 23:00  
 \*/  
  
 /\*\*  
 \* 类别名字（全部、待付款、代发货、待收货、待评价）  
 \*/* **private** String **catagoryName**;  
 */\*\*  
 \* 类别编码  
 \*/* **private** String **catagoryCode**;  
 */\*\*  
 \* 几级栏目  
 \*/* **private** String **rank**;  
 */\*\*  
 \* 父级栏目的编码  
 \*/* **private** String **parentCode**;  
 */\*\*  
 \* 此订单的卖家  
 \*/* **private** String **saleUser**;  
 */\*\*  
 \* 此卖家的编码  
 \*/* **private** String **saleUserCode**;  
 */\*\*  
 \* 此订单的评价  
 \*/* **private** String **comment**;  
  
 **public** String getSaleUser() {  
 **return saleUser**;  
 }  
  
 **public void** setSaleUser(String saleUser) {  
 **this**.**saleUser** = saleUser;  
 }  
  
 **public** String getSaleUserCode() {  
 **return saleUserCode**;  
 }  
  
 **public void** setSaleUserCode(String saleUserCode) {  
 **this**.**saleUserCode** = saleUserCode;  
 }  
  
 **public** String getComment() {  
 **return comment**;  
 }  
  
 **public void** setComment(String comment) {  
 **this**.**comment** = comment;  
 }  
  
 **public** String getTransport() {  
 **return transport**;  
 }  
  
 **public void** setTransport(String transport) {  
 **this**.**transport** = transport;  
 }  
  
 **public** String getStartTransport() {  
 **return startTransport**;  
 }  
  
 **public void** setStartTransport(String startTransport) {  
 **this**.**startTransport** = startTransport;  
 }  
  
 **public** String getReceive() {  
 **return receive**;  
 }  
  
 **public void** setReceive(String receive) {  
 **this**.**receive** = receive;  
 }  
  
 **public** String getPrice() {  
 **return price**;  
 }  
  
 **public void** setPrice(String price) {  
 **this**.**price** = price;  
 }  
  
 */\*\*  
 \* 此订单的物流  
 \*/* **private** String **transport**;  
 */\*\*  
 \* 此订单是否发货  
 \*/* **private** String **startTransport**;  
 */\*\*  
 \* 此订单是否签收  
 \*/* **private** String **receive**;  
 */\*\*  
 \* 此订单价格  
 \*/* **private** String **price**;  
 */\*\*  
 \* 生成此订单的时间  
 \*/* **private** String **createTime**;  
 */\*\*  
 \* 生成此订单的user  
 \*/* **private** String **createUser**;  
 */\*\*  
 \* 修改此订单的时间  
 \*/* **private** String **updateTime**;  
 */\*\*  
 \* 修改此订单的user  
 \*/* **private** String **updateUser**;  
  
 **public** String getCatagoryName() {  
 **return catagoryName**;  
 }  
  
 **public void** setCatagoryName(String catagoryName) {  
 **this**.**catagoryName** = catagoryName;  
 }  
  
 **public** String getCatagoryCode() {  
 **return catagoryCode**;  
 }  
  
 **public void** setCatagoryCode(String catagoryCode) {  
 **this**.**catagoryCode** = catagoryCode;  
 }  
  
 **public** String getRank() {  
 **return rank**;  
 }  
  
 **public void** setRank(String rank) {  
 **this**.**rank** = rank;  
 }  
  
 **public** String getParentCode() {  
 **return parentCode**;  
 }  
  
 **public void** setParentCode(String parentCode) {  
 **this**.**parentCode** = parentCode;  
 }  
  
 **public** String getCreateTime() {  
 **return createTime**;  
 }  
  
 **public void** setCreateTime(String createTime) {  
 **this**.**createTime** = createTime;  
 }  
  
 **public** String getCreateUser() {  
 **return createUser**;  
 }  
  
 **public void** setCreateUser(String createUser) {  
 **this**.**createUser** = createUser;  
 }  
  
 **public** String getUpdateTime() {  
 **return updateTime**;  
 }  
  
 **public void** setUpdateTime(String updateTime) {  
 **this**.**updateTime** = updateTime;  
 }  
  
 **public** String getUpdateUser() {  
 **return updateUser**;  
 }  
  
 **public void** setUpdateUser(String updateUser) {  
 **this**.**updateUser** = updateUser;  
 }  
}

商品实体类：

**package** homework;  
  
**public class** GoodsCategory {  
 */\*\*  
 \* lidantao  
 \* 2020.03.04 23:35  
 \*/  
  
 /\*\*  
 \* 类别名字（全部、待付款、代发货、待收货、待评价）  
 \*/* **private** String **catagoryName**;  
 */\*\*  
 \* 类别编码  
 \*/* **private** String **catagoryCode**;  
 */\*\*  
 \* 几级栏目  
 \*/* **private** String **rank**;  
 */\*\*  
 \* 父级栏目的编码  
 \*/* **private** String **parentCode**;  
 */\*\*  
 \* 超链接图片  
 \*/* **private** String **picture**;  
 */\*\*  
 \* 超链接图片编码  
 \*/* **private** String **pictureCode**;  
 */\*\*  
 \* 商品价格  
 \*/* **private** String **price**;  
 */\*\*  
 \* 商品的详情  
 \*/* **private** String **detail**;  
 */\*\*  
 \* 搜索框的信息  
 \*/* **private** String **queryMessage**;  
 */\*\*  
 \* 生成此订单的时间  
 \*/* **private** String **createTime**;  
 */\*\*  
 \* 生成此订单的user  
 \*/* **private** String **createUser**;  
 */\*\*  
 \* 修改此订单的时间  
 \*/* **private** String **updateTime**;  
 */\*\*  
 \* 修改此订单的user  
 \*/* **private** String **updateUser**;  
  
 **public** String getCatagoryName() {  
 **return catagoryName**;  
 }  
  
 **public void** setCatagoryName(String catagoryName) {  
 **this**.**catagoryName** = catagoryName;  
 }  
  
 **public** String getCatagoryCode() {  
 **return catagoryCode**;  
 }  
  
 **public void** setCatagoryCode(String catagoryCode) {  
 **this**.**catagoryCode** = catagoryCode;  
 }  
  
 **public** String getRank() {  
 **return rank**;  
 }  
  
 **public void** setRank(String rank) {  
 **this**.**rank** = rank;  
 }  
  
 **public** String getParentCode() {  
 **return parentCode**;  
 }  
  
 **public void** setParentCode(String parentCode) {  
 **this**.**parentCode** = parentCode;  
 }  
  
 **public** String getCreateTime() {  
 **return createTime**;  
 }  
  
 **public void** setCreateTime(String createTime) {  
 **this**.**createTime** = createTime;  
 }  
  
 **public** String getCreateUser() {  
 **return createUser**;  
 }  
  
 **public void** setCreateUser(String createUser) {  
 **this**.**createUser** = createUser;  
 }  
  
 **public** String getUpdateTime() {  
 **return updateTime**;  
 }  
  
 **public void** setUpdateTime(String updateTime) {  
 **this**.**updateTime** = updateTime;  
 }  
  
 **public** String getUpdateUser() {  
 **return updateUser**;  
 }  
  
 **public void** setUpdateUser(String updateUser) {  
 **this**.**updateUser** = updateUser;  
 }  
}