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
6. 使用Map 完成练习1的习题。
7. 仿照手机淘宝，设计订单和商品的实体类。
8. student类

package Digital\_factory;

public class Student {

/\*\*

\* 学生名字

\*/

private String name;

/\*\*

\* 学生年龄

\*/

private int age;

/\*\*

\* 学生成绩

\*/

private int socre;

/\*\*

\* 学生班级

\*/

private int grade;

public Student(String name, int age, int socre, int grade) {

this.name = name;

this.age = age;

this.socre = socre;

this.grade = grade;

}

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 getSocre() {

return socre;

}

public void setSocre(int socre) {

this.socre = socre;

}

public int getGrade() {

return grade;

}

public void setGrade(int grade) {

this.grade = grade;

}

public Student() {

}

@Override

public String toString() {

return "Student{" +

"name='" + name + '\'' +

", age='" + age + '\'' +

", socre='" + socre + '\'' +

", grade='" + grade + '\'' +

'}';

}

public int compareTo(Student value) {

return value.getSocre() - this.getSocre();

}

}

1. list练习

package Digital\_factory;

import java.util.ArrayList;

import java.util.Collections;

import java.util.Comparator;

import java.util.List;

/\*\*

\* 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”

\* <p>

\* （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

\*

\* @author 72937

\* 2020年3月4日 20:05:31

\*/

public class ListDemo {

public static void main(String[] args) {

//建立list1

List<Student> list1 = new ArrayList<>();

list1.add(new Student("张三", 18, 80, 1));

list1.add(new Student("李四", 19, 100, 1));

list1.add(new Student("王五", 17, 59, 1));

//建立list2

List<Student> list2 = new ArrayList<>();

list2.add(new Student("赵六", 18, 85, 2));

list2.add(new Student("刘七", 19, 93, 2));

list2.add(new Student("孙八", 17, 55, 2));

//整合两个list

list1.addAll(list2);

for (Student student : list1) {

System.out.println(student);

}

System.out.println("-------------------2------------------");

//实现工具类的自定义排序方法

Collections.sort(list1, new Comparator<Student>() {

@Override

public int compare(Student o1, Student o2) {

int i = o2.getSocre() - o1.getSocre();

if (i == 0) {

return o2.getAge() - o1.getAge();

}

return i;

}

});

for (Student student : list1) {

System.out.println(student);

}

System.out.println("-------------------3------------------");

for (Student student : list1) {

if (student.getSocre() < 60) {

System.out.println(student);

}

}

System.out.println("-------------------4------------------");

for (Student student : list1) {

if ("张三".equals(student.getName())) {

System.out.println(student);

}

}

System.out.println("-------------------5------------------");

for (int i = list1.size() - 1; i >= 0; i--) {

Student student = list1.get(i);

if (student.getAge() > 18) {

list1.remove(student);

}

}

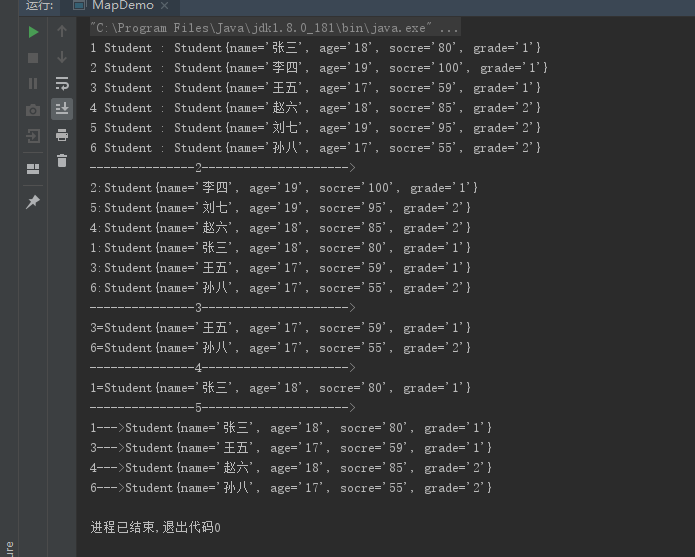
for (Student student : list1) {

System.out.println(student);

}

}

}



1. map练习

package Digital\_factory;

import java.util.\*;

/\*\*

\* 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map学生信息成一个新的map

\* （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map剔除年龄大于18岁的学生信息

\* 使用Map来完成

\*

\* @author 72937

\* 2020年3月4日 20:35:08

\*/

public class MapDemo {

public static void main(String[] args) {

//建立map1

Map<Integer, Student> map1 = new HashMap<>();

map1.put(1, new Student("张三", 18, 80, 1));

map1.put(2, new Student("李四", 19, 100, 1));

map1.put(3, new Student("王五", 17, 59, 1));

//建立map2

Map<Integer, Student> map2 = new HashMap<>();

map2.put(4, new Student("赵六", 18, 85, 2));

map2.put(5, new Student("刘七", 19, 95, 2));

map2.put(6, new Student("孙八", 17, 55, 2));

//合并两个map

map1.putAll(map2);

map1.forEach((t, v) -> System.out.println(t + " Student : " + v));

System.out.println("---------------2--------------------->");

//这里将map1.entrySet()转换成list

List<Map.Entry<Integer, Student>> list = new ArrayList<Map.Entry<Integer, Student>>(map1.entrySet());

//然后通过比较器来实现排序

Collections.sort(list, new Comparator<Map.Entry<Integer, Student>>() {

@Override

public int compare(Map.Entry<Integer, Student> o1, Map.Entry<Integer, Student> o2) {

return o1.getValue().compareTo(o2.getValue());

}

});

for (Map.Entry<Integer, Student> studentEntry : list) {

System.out.println(studentEntry.getKey() + ":" + studentEntry.getValue());

}

System.out.println("---------------3--------------------->");

//输出不及格的

for (Map.Entry<Integer, Student> studentEntry : list) {

if (studentEntry.getValue().getSocre() < 60) {

System.out.println(studentEntry);

}

}

System.out.println("---------------4--------------------->");

//输出张三

for (Map.Entry<Integer, Student> studentEntry : list) {

if ("张三".equals(studentEntry.getValue().getName())) {

System.out.println(studentEntry);

}

}

System.out.println("---------------5--------------------->");

//剔除大于18的学生

for (Map.Entry<Integer, Student> studentEntry : list) {

if (studentEntry.getValue().getAge() > 18) {

map1.remove(studentEntry.getKey());

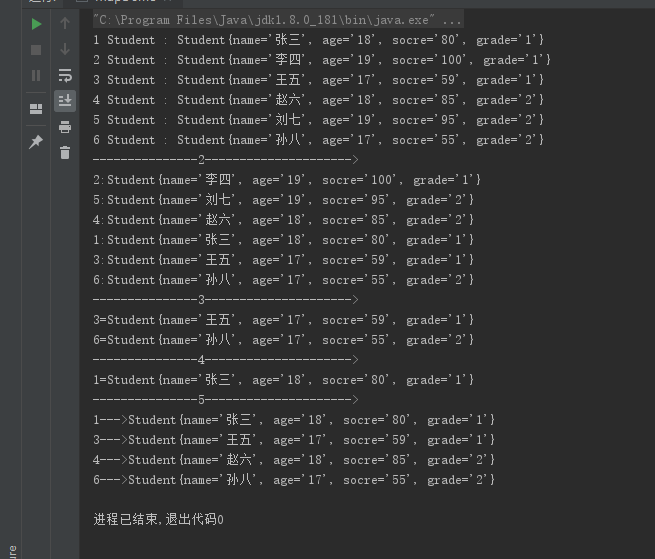
}

}

map1.forEach((t, s) -> System.out.println(t + "--->" + s));

}

}



1. 商品类建立：

package Digital\_factory;

public class goods {

/\*\*

\* 商品id

\*/

private int id;

/\*\*

\* 商品名称

\*/

private String name;

/\*\*

\* 商品售价

\*/

private int price;

/\*\*

\* 商品创建时间

\*/

private String goodsTime;

/\*\*

\* 套餐

\*/

private String setMeal;

/\*\*

\* 销量

\*/

private int salesVolume;

/\*\*

\* 分类

\*/

private String classIfication;

/\*\*

\* 数量

\*/

private int quantity;

/\*\*

\* 库存

\*/

private int reserve;

/\*\*

\* 商品修改时间

\*/

private String goodsUpdateTime;

/\*\*

\* 商品历史价格

\*/

private int historiPrice;

}

订单类建立：

package Digital\_factory;

public class Order {

/\*\*

\* 订单号

\*/

private int orderID;

/\*\*

\* 商品名

\*/

private String goodsName;

/\*\*

\* 商品id

\*/

private int goodsID;

/\*\*

\* 店主

\*/

private String shopkeeper;

/\*\*

\* 买家

\*/

private String Consignee;

/\*\*

\*

\* 收货人

\*/

private String seller;

/\*\*

\*

\* 收货地址

\*/

private String address;

/\*\*

\*

\* 金额

\*/

private float money;

/\*\*

\*

\* 交易号

\*/

private int transactionID;

/\*\*

\*

\* 创建时间

\*/

private String creationTime;

/\*\*

\*

\* 付款时间

\*/

private String paymentTime;

/\*\*

\*

\* 成交时间

\*/

private String closingTime;

}