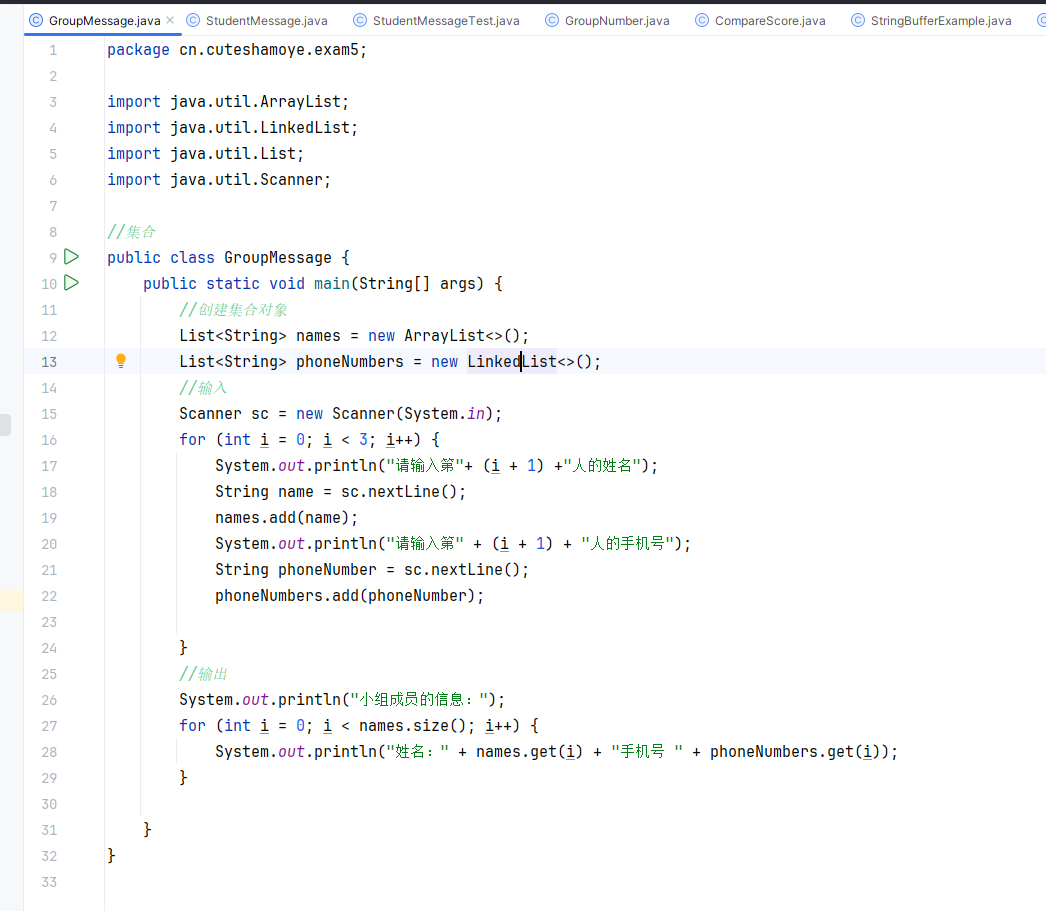
1. 使用ArrayList和LinkedList保存小组成员的信息（包括名字、手机号），让用户从控制台输入信息，之后依次遍历输出。

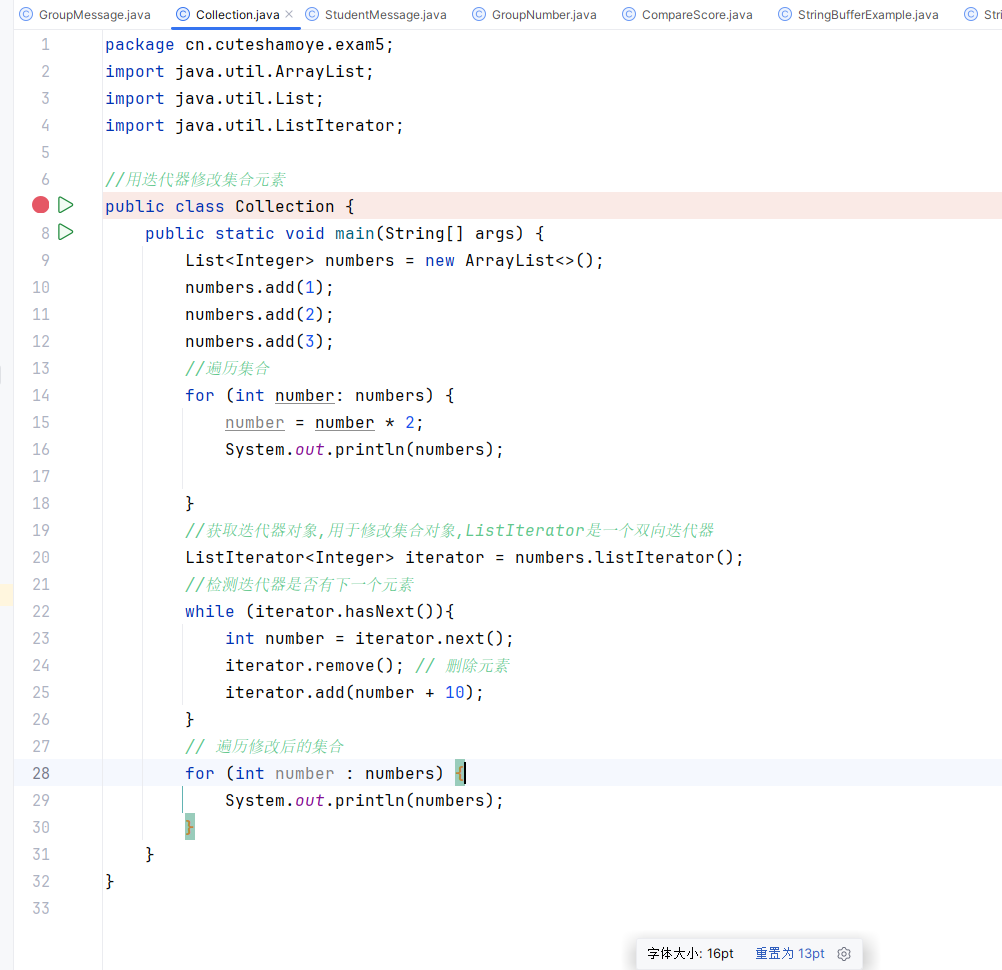
选择ArrayList或LinkedList来存储和操作数据。如果需要频繁地插入和删除元素，而对随机访问的性能要求不高，选择LinkedList。如果需要频繁地随机访问元素，选择ArrayList。

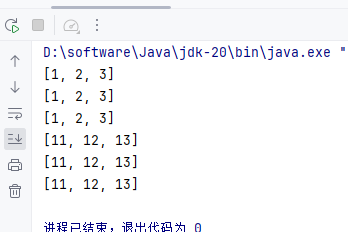




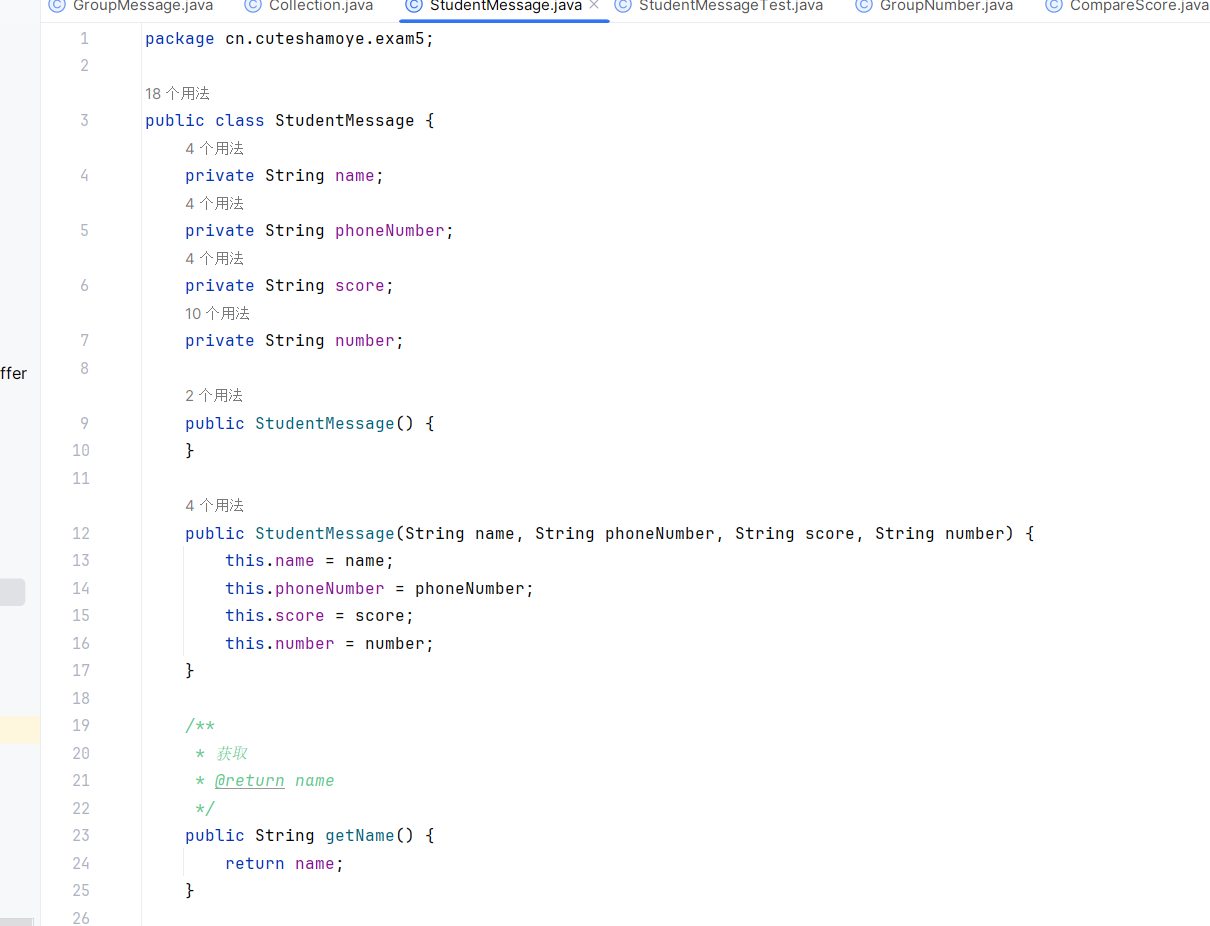
1. 思考书本“当使用foreach循环遍历集合和数组时，只能访问集合中的元素，不能对其中的元素进行修改”说法是否正确，举例说明。

正确，使用foreach循环遍历集合或数组时，只能访问集合或数组中的元素，不能对其中的元素进行修改。这是因为foreach循环是一种只读循环，用于遍历集合或数组的元素，而不是用于修改它们。但可以使用迭代器来修改其中的元素



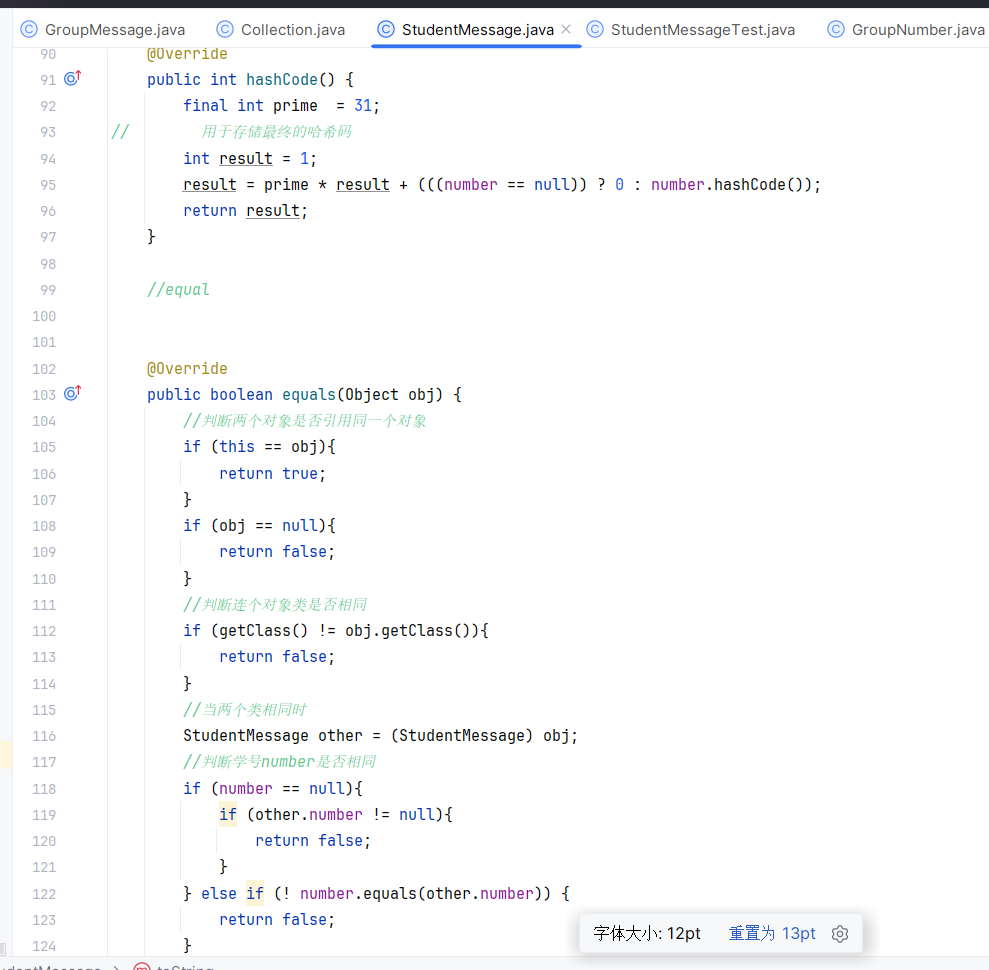


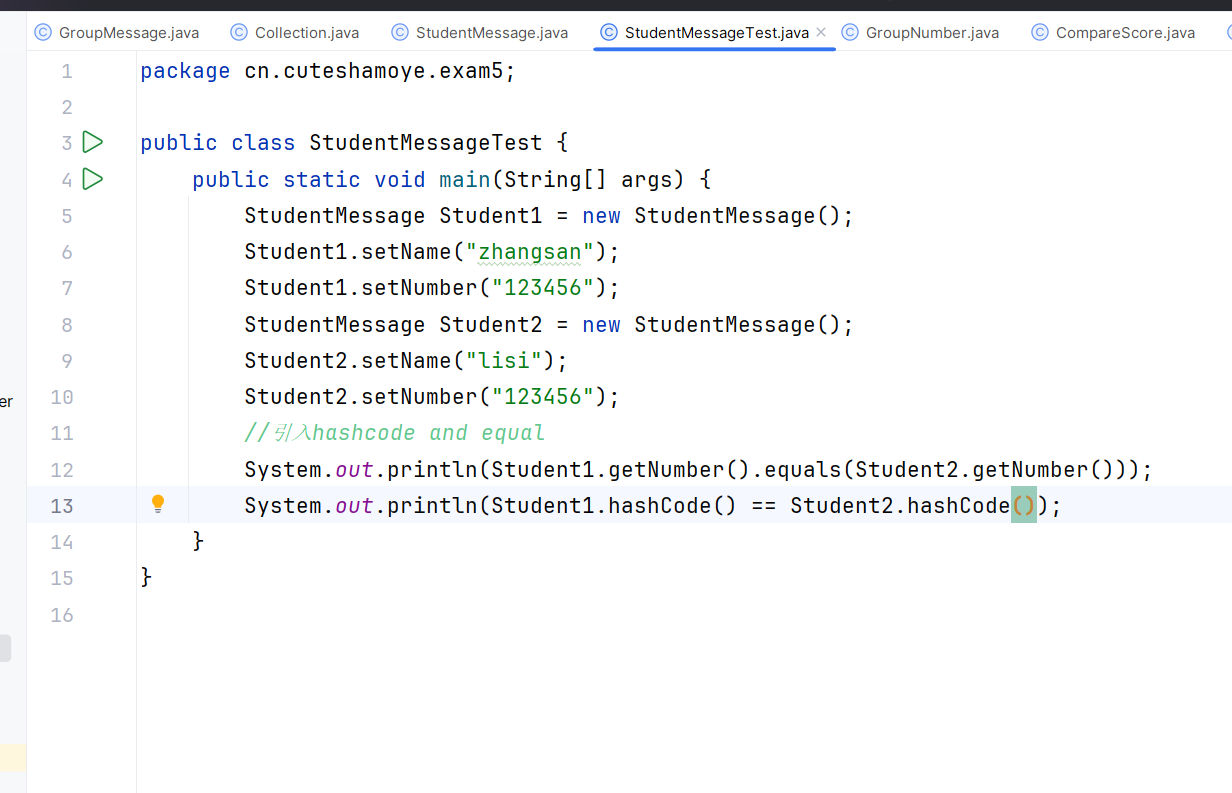
1. 1）第1题中的小组成员添加学号，成绩属性。

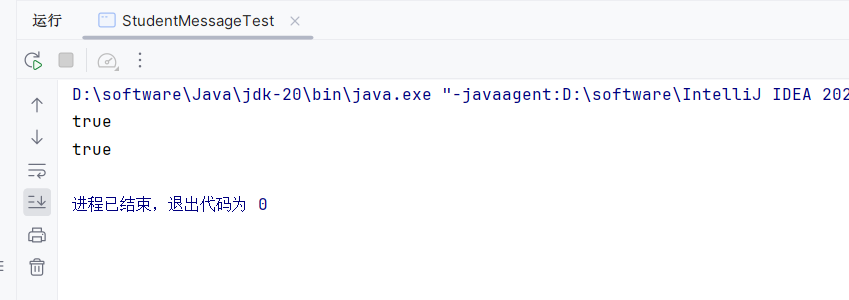




2）重写hashcode()和equal()方法，使用学号作为判断成员是否相等的条件，如果学号相等，则认为是相等的对象。

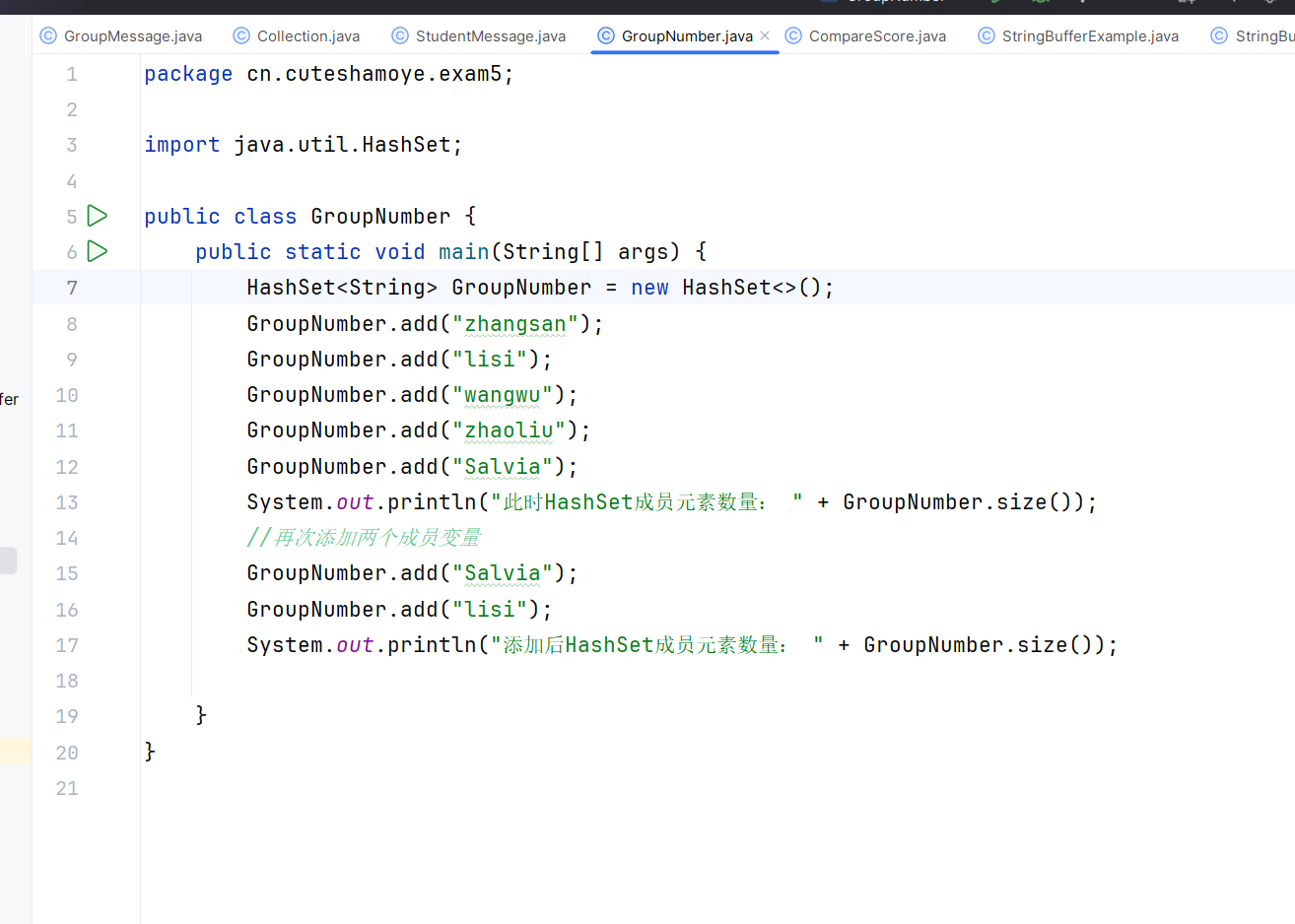


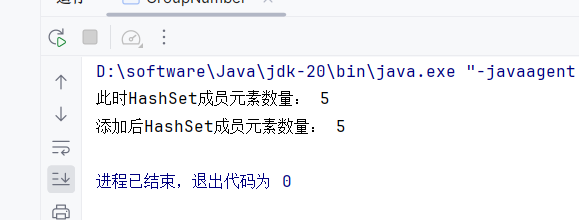




3）依次添加5个小组成员到一个HashSet对象里，之后再重复添加前面两个成员，查看这时候HashSet对象里有5个还是7个小组成员，为什么？

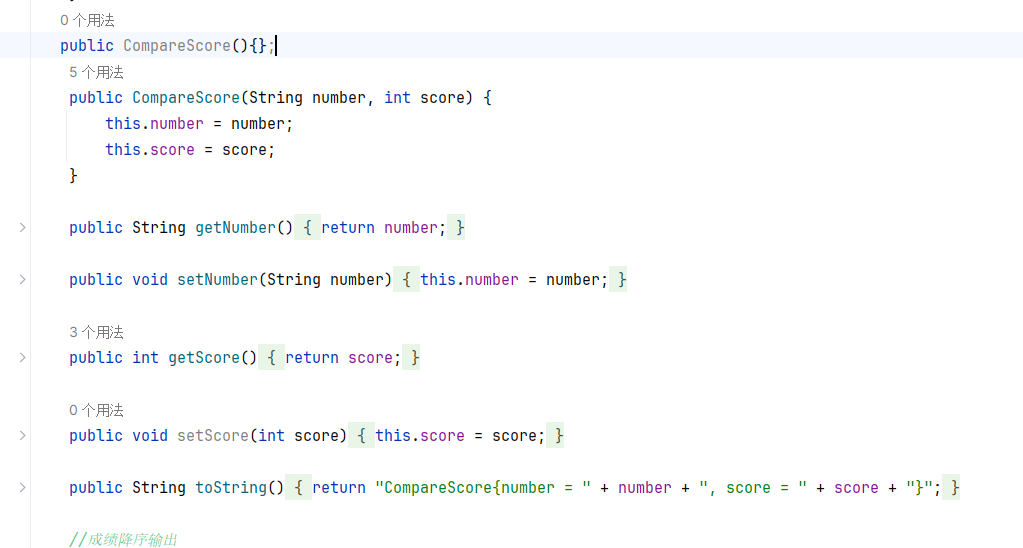
因为HashSet会根据元素的哈希码和equals()方法来判断元素是否重复。在添加元素"Salvia"和"lisi"时，HashSet会发现它们的哈希码与已有元素相同，并且equals()方法也返回true，因此HashSet不会将它们添加到集合中。



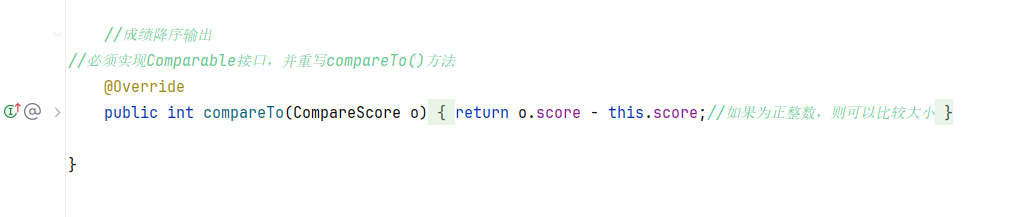


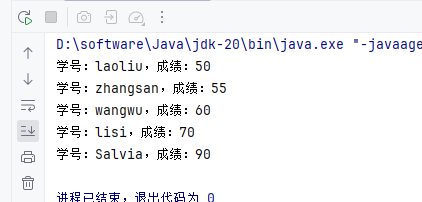
4） 依次添加5个小组成员到一个TreeSet对象里，然后遍历打印出每个成员的学号和成绩，要求以成绩作为排序条件，降序输出。



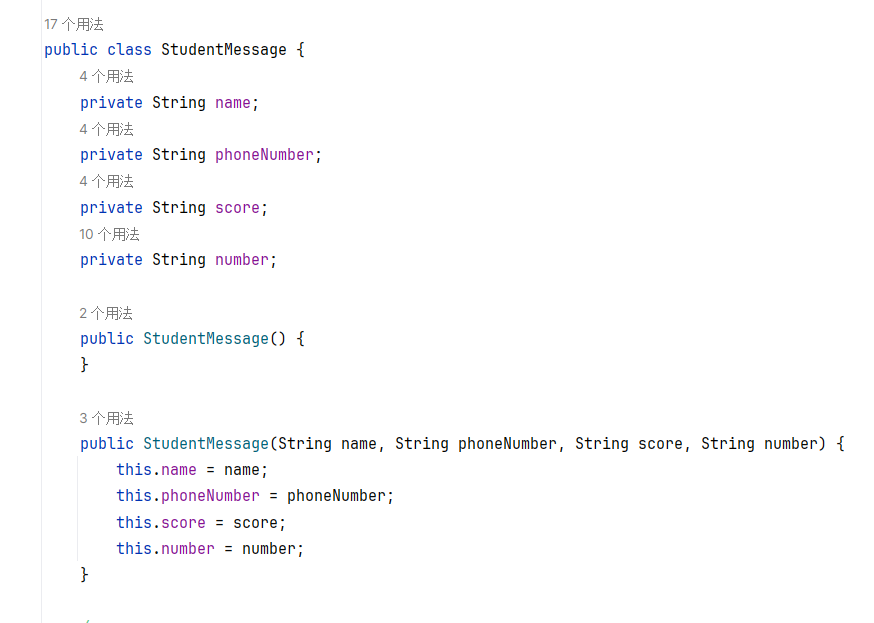




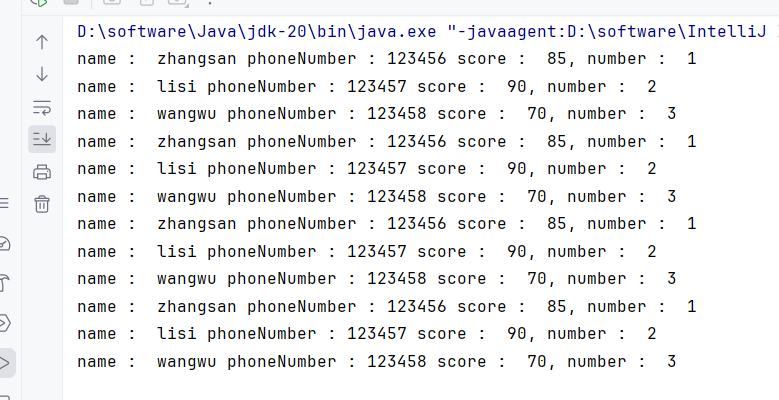




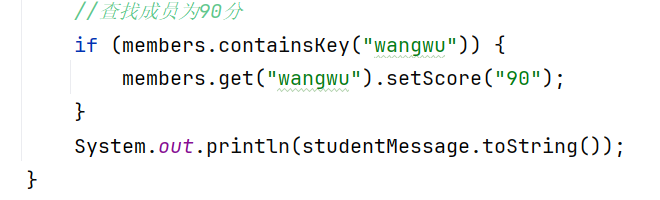
4、1）使用上面的小组成员类，以学号作为关键字存储小组成员信息到HashMap对象里，并使用三种遍历方法，把小组成员信息（包括学号、姓名，成绩）打印出来







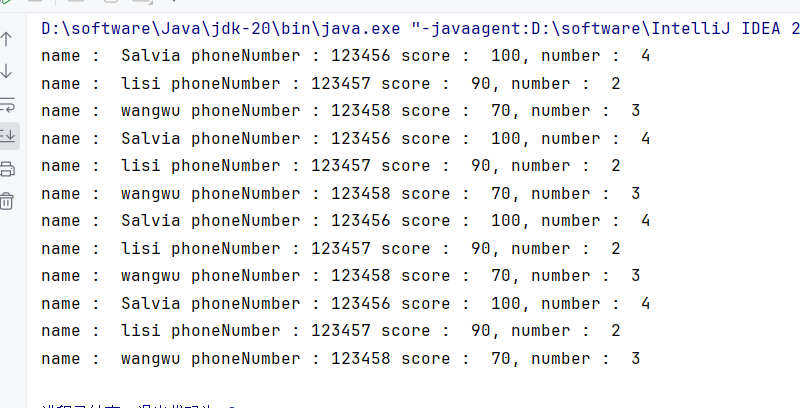
2）查找学号是“xxx”的成员，并修改成绩为90分。（假如存在xxx的成员）



3）把相同学号，不同姓名的成员添加进HashMap对象会怎么样？

HashMap的key是唯一的，如果你尝试添加相同的key，新的value将会覆盖旧的value。





4）把小组成员保存进TreeMap,并根据学号排序。

