## 一、

1、使用链表和映射存放多个图书信息，遍历并输出。其中商品属性：编号，名称，单价，出版社；使用商品编号作为映射中的key。

package work8;
  
  
import java.util.\*;
  
  
public class BookInfo {
  
 public static void main(String[] args) {
  
 Map<String, ArrayList<String>> bookInfo = new HashMap<>();
  
 bookInfo.put("ID001", new ArrayList<>(Arrays.asList("ID001", "Java程序设计", "99.00", "机械工业出版社")));
  
 bookInfo.put("ID002", new ArrayList<>(Arrays.asList("ID002", "算法竞赛入门经典", "49.80", "清华大学出版社")));
  
 bookInfo.put("ID003", new ArrayList<>(Arrays.asList("ID003", "牛津高阶英汉词典", "138.00", "牛津大学出版社")));
  
 bookInfo.put("ID004", new ArrayList<>(Arrays.asList("ID004", "C++ Primer", "128.0", "电子工业出版社")));
  
 for (String keys : bookInfo.keySet()) {
  
 for (String val : bookInfo.get(keys)) {
  
 System.out.print(val + " ");
  
 }
  
 System.out.println();
  
 }
  
  
 }
  
}

## 二、

2、由控制台按照固定格式输入学生信息，包括学号，姓名，年龄信息，当输入的内容为exit退出；将输入的学生信息分别封装到一个Student对象中，再将每个Student对象加入到一个集合中，要求集合中的元素按照年龄大小正序排序；最后遍历集合，将集合中学生信息写入到记事本，每个学生数据占单独一行。

package work8;
  
  
import java.io.File;
  
import java.io.FileWriter;
  
import java.io.IOException;
  
import java.io.PrintWriter;
  
import java.util.Scanner;
  
import java.util.Set;
  
import java.util.TreeSet;
  
  
public class InputStudent {
  
 public static void main(String[] args) throws IOException {
  
 Scanner in = new Scanner(System.in);
  
 File file = new File("src/work8/Student.txt");
  
 if (!file.exists()) {
  
 file.createNewFile();
  
 }
  
 PrintWriter pr = new PrintWriter(new FileWriter(file));
  
 Set<Student> students = new TreeSet<>();
  
 while (true) {
  
 String id = in.next();
  
 if (id.equals("exit"))
  
 break;
  
 String name = in.next();
  
 int age = in.nextInt();
  
 students.add(new Student(id, name, age));
  
 }
  
 for (Student student : students) {
  
 pr.println(student.getId() + " " + student.getName() + " " + student.getAge());
  
 }
  
 pr.close();
  
 in.close();
  
  
 }
  
}

package work8;
  
  
public class Student implements Comparable {
  
 private String id;
  
 private String name;
  
 private int age;
  
  
 public Student(String \_id, String \_name, int \_age) {
  
 this.id = \_id;
  
 this.name = \_name;
  
 this.age = \_age;
  
 }
  
  
 public String getId() {
  
 return id;
  
 }
  
  
 public String getName() {
  
 return name;
  
 }
  
  
 public int getAge() {
  
 return age;
  
 }
  
  
 @Override
  
 public int compareTo(Object o) {
  
 if (!(o instanceof Student)) {
  
 throw new RuntimeException("不是Student对象");
  
 }
  
 Student p = (Student) o;
  
 if (this.age > p.age)
  
 return 1;
  
 if (this.age == p.age) {
  
 return this.name.compareTo(p.name);
  
 }
  
 return -1;
  
 }
  
}