java基础知识 第11天

**【学习目标】理解、了解、应用、记忆**

通过今天的学习，参训学员能够：（解释的时候说出二级目标的掌握程度）

1. **【应用】综合练习**
2. 【应用】能够独立完成把集合中的学生信息写入文本文件案例
3. 【应用】能够独立完成文件中的数据读到集合并遍历集合案例
4. 【应用】能够独立完成学生管理系统IO版案例

# 综合练习

## 综合练习一

A:键盘录入3个学生信息(学号,姓名,年龄,居住地)存入集合,要求学生信息的学号不能重复

B:遍历集合把每一个学生信息存入文本文件

C:每一个学生信息为一行数据,每行学生的学号,姓名,年龄和居住地在文件中均以逗号分隔

### 案例代码一

|  |
| --- |
| **package** com.igeek\_01;  /\*\*  \* **@ClassName**: Student  \* **@Description**: 学生类  \* **@date** 2017年10月14日 上午10:35:59  \* Company www.igeekhome.com  \*  \*/  **public** **class** Student {  /\*\*  \* **@Fields** id : 学号  \*/  **private** String id;  /\*\*  \* **@Fields** name : 姓名  \*/  **private** String name;  /\*\*  \* **@Fields** age : 年龄  \*/  **private** String age;  /\*\*  \* **@Fields** address : 居住地  \*/  **private** String address;  /\*\*  \* **@Title**: Student  \*/  **public** Student() {  }  /\*\*  \* **@Title**: Student  \* **@param** id  \* **@param** name  \* **@param** age  \* **@param** address  \*/  **public** Student(String id, String name, String age, String address) {  **this**.id = id;  **this**.name = name;  **this**.age = age;  **this**.address = address;  }  /\*\*  \* **@return** the id  \*/  **public** String getId() {  **return** id;  }  /\*\*  \* **@param** id the id to set  \*/  **public** **void** setId(String id) {  **this**.id = id;  }  /\*\*  \* **@return** the name  \*/  **public** String getName() {  **return** name;  }  /\*\*  \* **@param** name the name to set  \*/  **public** **void** setName(String name) {  **this**.name = name;  }  /\*\*  \* **@return** the age  \*/  **public** String getAge() {  **return** age;  }  /\*\*  \* **@param** age the age to set  \*/  **public** **void** setAge(String age) {  **this**.age = age;  }  /\*\*  \* **@return** the address  \*/  **public** String getAddress() {  **return** address;  }  /\*\*  \* **@param** address the address to set  \*/  **public** **void** setAddress(String address) {  **this**.address = address;  }  } |

|  |
| --- |
| **package** com.igeek\_01;  **import** java.io.BufferedWriter;  **import** java.io.FileWriter;  **import** java.io.IOException;  **import** java.util.ArrayList;  **import** java.util.Scanner;  /\*\*  \* **@ClassName**: ArrayListToFileTest  \* **@Description**: 学生信息添加到文件中  \* **@date** 2017年10月14日 下午2:57:18  \* Company www.igeekhome.com  \*  \* 键盘录入3个学生信息(学号,姓名,年龄,居住地)存入集合。然后遍历集合把每一个学生信息存入文本文件(每一个学生信息为一行数据，自己定义分割标记)  \*  \* 分析：  \* A:定义学生类  \* B:创建集合对象  \* C:写方法实现键盘录入学生信息，并把学生对象作为元素添加到集合  \* D:创建输出缓冲流对象  \* E:遍历集合,得到每一个学生信息，并把学生信息按照一定的格式写入文本文件  \* 举例：1001,jim,18,无锡  \* F:释放资源  \*/  **public** **class** ArrayListToFileTest {  **public** **static** **void** main(String[] args) **throws** IOException {  //创建集合对象  ArrayList<Student> array = **new** ArrayList<Student>();    //写方法实现键盘录入学生信息，并把学生对象作为元素添加到集合  *addStudent*(array);  *addStudent*(array);  *addStudent*(array);    //创建输出缓冲流对象  BufferedWriter bw = **new** BufferedWriter(**new** FileWriter("array.txt"));    //遍历集合,得到每一个学生信息，并把学生信息按照一定的格式写入文本文件  **for**(**int** x=0; x<array.size(); x++) {  Student s = array.get(x);  //1001,jim,18,无锡  StringBuilder sb = **new** StringBuilder();  sb.append(s.getId()).append(",").append(s.getName()).append(",").append(s.getAge()).append(",").append(s.getAddress());    bw.write(sb.toString());  bw.newLine();  bw.flush();  }    //释放资源  bw.close();  }    /\*\*  \* **@Title**: addStudent  \* **@Description**: 添加学生  \* **@param** array  \*/  **public** **static** **void** addStudent(ArrayList<Student> array) {  //创建键盘录入对象  Scanner sc = **new** Scanner(System.***in***);    //为了让id能够被访问到，我们就把id定义在了循环的外面  String id;    //为了让代码能够回到这里，用循环  **while**(**true**) {  System.***out***.println("请输入学生学号：");  //String id = sc.nextLine();  id = sc.nextLine();    //判断学号有没有被人占用  //定义标记  **boolean** flag = **false**;  //遍历集合，得到每一个学生  **for**(**int** x=0; x<array.size(); x++) {  Student s = array.get(x);  //获取该学生的学号，和键盘录入的学号进行比较  **if**(s.getId().equals(id)) {  flag = **true**; //说明学号被占用了  **break**;  }  }    **if**(flag) {  System.***out***.println("你输入的学号已经被占用,请重新输入");  }**else** {  **break**; //结束循环  }  }      System.***out***.println("请输入学生姓名：");  String name = sc.nextLine();  System.***out***.println("请输入学生年龄：");  String age = sc.nextLine();  System.***out***.println("请输入学生居住地：");  String address = sc.nextLine();    //创建学生对象  Student s = **new** Student();  s.setId(id);  s.setName(name);  s.setAge(age);  s.setAddress(address);    //把学生对象作为元素添加到集合  array.add(s);    //给出提示  System.***out***.println("添加学生成功");  }  } |

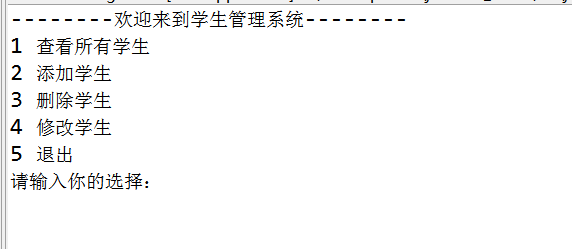
## 综合练习二

把上一案例的array.txt文本文件中的学生信息读取出来存储到集合中，然后遍历集合，在控制台输出

### 案例代码二

|  |
| --- |
| **package** com.igeek\_02;  **import** java.io.BufferedReader;  **import** java.io.FileReader;  **import** java.io.IOException;  **import** java.util.ArrayList;  /\*\*  \* **@ClassName**: FileToArrayListTest  \* **@Description**: 把文本文件中的学生信息读取出来存储到集合中  \* **@date** 2017年10月14日 下午3:04:38  \* Company www.igeekhome.com  \*  \* 把上一题的文本文件中的学生信息读取出来存储到集合中，然后遍历集合，在控制台输出  \*  \* 分析：  \* A:定义学生类  \* B:创建输入缓冲流对象  \* C:创建集合对象  \* D:读取文件数据，并把数据按照一定的格式进行分割赋值给学生对象，然后把学生对象作为元素存储到集合  \* 1001,jim,18,无锡  \* E:释放资源  \* F:遍历集合  \*/  **public** **class** FileToArrayListTest {  **public** **static** **void** main(String[] args) **throws** IOException {  //创建输入缓冲流对象  BufferedReader br = **new** BufferedReader(**new** FileReader("array.txt"));    //创建集合对象  ArrayList<Student> array = **new** ArrayList<Student>();    //读取文件数据，并把数据按照一定的格式进行分割赋值给学生对象，然后把学生对象作为元素存储到集合  String line;  **while**((line=br.readLine())!=**null**) {  //分割字符串  String[] strArray = line.split(",");  //创建学生对象并赋值  Student s = **new** Student();  s.setId(strArray[0]);  s.setName(strArray[1]);  s.setAge(strArray[2]);  s.setAddress(strArray[3]);  //把学生对象作为元素存储到集合  array.add(s);  }    //释放资源  br.close();    //遍历集合  System.***out***.println("学号\t姓名\t年龄\t居住地");  **for**(**int** x=0; x<array.size(); x++) {  Student s = array.get(x);  System.***out***.println(s.getId()+"\t"+s.getName()+"\t"+s.getAge()+"\t"+s.getAddress());  }  }  } |

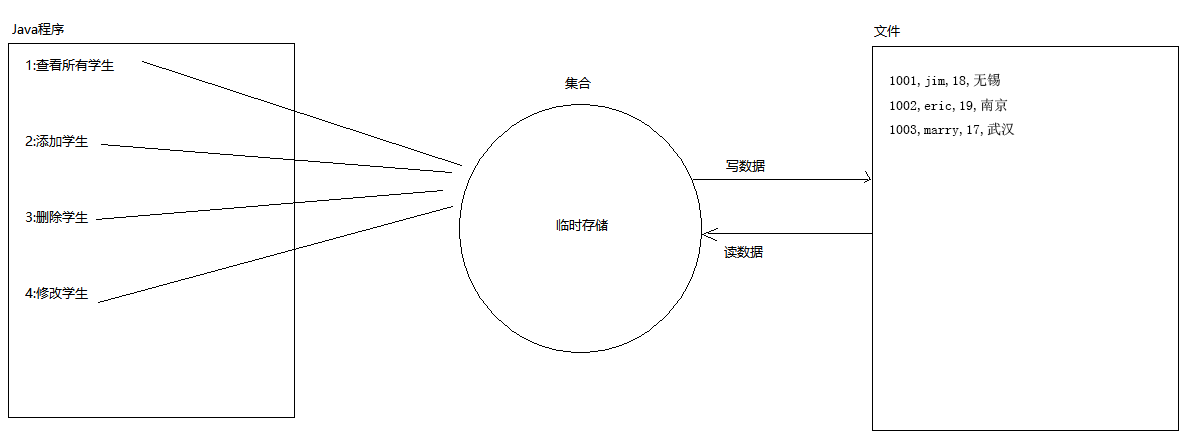
## 综合案例三



学生管理系统IO版:

A:在第五天上午学生管理系统案例的基础上,添加新需求

B:查看所有的学生的数据需要从students.txt(提供好的)中读取出来

 C:增删改都需要先从students.txt中读出学生的数据然后再把改后的学生数据重新写回students.txt

### 原有的集合版学生管理系统

#### 案例代码三

|  |
| --- |
| **package** com.igeek\_03;  **import** java.util.ArrayList;  **import** java.util.Scanner;  /\*\*  \* **@ClassName**: StudentManagerTest  \* **@Description**: 学生管理系统的主类  \* **@date** 2017年10月14日 上午10:50:19  \* Company www.igeekhome.com  \*  \* 这是我的学生管理系统的主类  \*  \* 步骤如下：  \* A:定义学生类  \* B:学生管理系统的主界面的代码编写  \* C:学生管理系统的查看所有学生的代码编写  \* D:学生管理系统的添加学生的代码编写  \* E:学生管理系统的删除学生的代码编写  \* F:学生管理系统的修改学生的代码编写  \*/  **public** **class** StudentManagerTest {  **public** **static** **void** main(String[] args) {  // 创建集合对象，用于存储学生数据  ArrayList<Student> array = **new** ArrayList<Student>();  // 为了让程序能够回到这里来，我们使用循环  **while** (**true**) {  // 这是学生管理系统的主界面  System.***out***.println("--------欢迎来到学生管理系统--------");  System.***out***.println("1 查看所有学生");  System.***out***.println("2 添加学生");  System.***out***.println("3 删除学生");  System.***out***.println("4 修改学生");  System.***out***.println("5 退出");  System.***out***.println("请输入你的选择：");  // 创建键盘录入对象  Scanner sc = **new** Scanner(System.***in***);  String choiceString = sc.nextLine();  // 用switch语句实现选择  **switch** (choiceString) {  **case** "1":  // 查看所有学生  *findAllStudent*(array);  **break**;  **case** "2":  // 添加学生  *addStudent*(array);  **break**;  **case** "3":  // 删除学生  *deleteStudent*(array);  **break**;  **case** "4":  // 修改学生  *updateStudent*(array);  **break**;  **case** "5":  // 退出  // System.out.println("谢谢你的使用");  // break;  **default**:  System.***out***.println("谢谢你的使用");  System.*exit*(0); // JVM退出  **break**;  }  }  }    /\*\*  \* **@Title**: findAllStudent  \* **@Description**: 查看所有学生  \* **@param** array  \*/  **public** **static** **void** findAllStudent(ArrayList<Student> array) {  //首先来判断集合中是否有数据，如果没有数据，就给出提示，并让该方法不继续往下执行  **if**(array.size() == 0) {  System.***out***.println("不好意思,目前没有学生信息可供查询,请回去重新选择你的操作");  **return**;  }    //\t 其实就是一个tab键的位置  System.***out***.println("学号\t姓名\t年龄\t居住地");  **for**(**int** x=0; x<array.size(); x++) {  Student s = array.get(x);  System.***out***.println(s.getId()+"\t"+s.getName()+"\t"+s.getAge()+"\t"+s.getAddress());  }  }    /\*\*  \* **@Title**: addStudent  \* **@Description**: 添加学生  \* **@param** array  \*/  **public** **static** **void** addStudent(ArrayList<Student> array) {  //创建键盘录入对象  Scanner sc = **new** Scanner(System.***in***);    //为了让id能够被访问到，我们就把id定义在了循环的外面  String id;    //为了让代码能够回到这里，用循环  **while**(**true**) {  System.***out***.println("请输入学生学号：");  //String id = sc.nextLine();  id = sc.nextLine();    //判断学号有没有被人占用  //定义标记  **boolean** flag = **false**;  //遍历集合，得到每一个学生  **for**(**int** x=0; x<array.size(); x++) {  Student s = array.get(x);  //获取该学生的学号，和键盘录入的学号进行比较  **if**(s.getId().equals(id)) {  flag = **true**; //说明学号被占用了  **break**;  }  }    **if**(flag) {  System.***out***.println("你输入的学号已经被占用,请重新输入");  }**else** {  **break**; //结束循环  }  }    System.***out***.println("请输入学生姓名：");  String name = sc.nextLine();  System.***out***.println("请输入学生年龄：");  String age = sc.nextLine();  System.***out***.println("请输入学生居住地：");  String address = sc.nextLine();    //创建学生对象  Student s = **new** Student();  s.setId(id);  s.setName(name);  s.setAge(age);  s.setAddress(address);    //把学生对象作为元素添加到集合  array.add(s);    //给出提示  System.***out***.println("添加学生成功");  }    /\*\*  \* **@Title**: deleteStudent  \* **@Description**: 删除学生  \* **@param** array  \*/  **public** **static** **void** deleteStudent(ArrayList<Student> array) {  //删除学生的思路：键盘录入一个学号，到集合中去查找，看是否有学生使用的是该学号，如果有就删除该学生  //创建键盘录入对象  Scanner sc = **new** Scanner(System.***in***);  System.***out***.println("请输入你要删除的学生的学号：");  String id = sc.nextLine();    //我们必须给出学号不存在的时候的提示    //定义一个索引  **int** index = -1;    //遍历集合  **for**(**int** x=0; x<array.size(); x++) {  //获取到每一个学生对象  Student s = array.get(x);  //拿这个学生对象的学号和键盘录入的学号进行比较  **if**(s.getId().equals(id)) {  index = x;  **break**;  }  }    **if**(index == -1) {  System.***out***.println("不好意思,你要删除的学号对应的学生信息不存在,请回去重新你的选择");  }**else** {  array.remove(index);  System.***out***.println("删除学生成功");  }    }    /\*\*  \* **@Title**: updateStudent  \* **@Description**: 修改学生  \* **@param** array  \*/  **public** **static** **void** updateStudent(ArrayList<Student> array) {  //修改学生的思路：键盘录入一个学号，到集合中去查找，看是否有学生使用的是该学号，如果有就修改该学生  //创建键盘录入对象  Scanner sc = **new** Scanner(System.***in***);  System.***out***.println("请输入你要修改的学生的学号：");  String id = sc.nextLine();    //定义一个索引  **int** index = -1;    //遍历集合  **for**(**int** x=0; x<array.size(); x++) {  //获取每一个学生对象  Student s = array.get(x);  //拿学生对象的学号和键盘录入的学号进行比较  **if**(s.getId().equals(id)) {  index = x;  **break**;  }  }    **if**(index == -1) {  System.***out***.println("不好意思,你要修改的学号对应的学生信息不存在,请回去重新你的选择");  }**else** {  System.***out***.println("请输入学生新姓名：");  String name = sc.nextLine();  System.***out***.println("请输入学生新年龄：");  String age = sc.nextLine();  System.***out***.println("请输入学生新居住地：");  String address = sc.nextLine();    //创建学生对象  Student s = **new** Student();  s.setId(id);  s.setName(name);  s.setAge(age);  s.setAddress(address);    //修改集合中的学生对象  array.set(index, s);    //给出提示  System.***out***.println("修改学生成功");  }  }  } |

### 对文本文件进行读写操作

#### 案例代码四

public static void readData(String fileName,ArrayList<Student> array):从fileName的文件中读取学生的数据,并把学生的数据封装到array集合中

public static void writeData(String fileName,ArrayList<Student> array):

将array中的每个元素中包含的数据写入到fileName的文件中

|  |
| --- |
| /\*\*  \* **@Title**: readData  \* **@Description**: 从文件中读数据到集合  \* **@param** fileName  \* **@param** array  \* **@throws** IOException  \*/  **public** **static** **void** readData(String fileName, ArrayList<Student> array)  **throws** IOException {  // 创建输入缓冲流对象  BufferedReader br = **new** BufferedReader(**new** FileReader(fileName));  String line;  **while** ((line = br.readLine()) != **null**) {  String[] datas = line.split(",");  Student s = **new** Student();  s.setId(datas[0]);  s.setName(datas[1]);  s.setAge(datas[2]);  s.setAddress(datas[3]);  array.add(s);  }  br.close();  }  /\*\*  \* **@Title**: writeData  \* **@Description**: 把集合中的数据写入文件  \* **@param** fileName  \* **@param** array  \* **@throws** IOException  \*/  **public** **static** **void** writeData(String fileName, ArrayList<Student> array)  **throws** IOException {  // 创建输出缓冲流对象  BufferedWriter bw = **new** BufferedWriter(**new** FileWriter(fileName));  **for** (**int** x = 0; x < array.size(); x++) {  Student s = array.get(x);  StringBuilder sb = **new** StringBuilder();  sb.append(s.getId()).append(",").append(s.getName()).append(",")  .append(s.getAge()).append(",").append(s.getAddress());  bw.write(sb.toString());  bw.newLine();  bw.flush();  }  bw.close();  } |

### 增删改查功能调用读写文件方法

#### 案例代码五

|  |
| --- |
| **package** com.igeek\_03\_03;  **import** java.io.BufferedReader;  **import** java.io.BufferedWriter;  **import** java.io.FileReader;  **import** java.io.FileWriter;  **import** java.io.IOException;  **import** java.util.ArrayList;  **import** java.util.Scanner;  /\*\*  \* **@ClassName**: StudentManagerTest  \* **@Description**: 学生管理系统的主类  \* **@date** 2017年10月14日 上午10:50:19  \* Company www.igeekhome.com  \*  \* 这是我的学生管理系统的主类  \*  \* 步骤如下：  \* A:定义学生类  \* B:学生管理系统的主界面的代码编写  \* C:学生管理系统的查看所有学生的代码编写  \* D:学生管理系统的添加学生的代码编写  \* E:学生管理系统的删除学生的代码编写  \* F:学生管理系统的修改学生的代码编写  \*/  **public** **class** StudentManagerTest {  **public** **static** **void** main(String[] args) **throws** IOException {  //定义文件路径  String fileName = "students.txt";    //为了让程序能够回到这里来，我们使用循环  **while**(**true**) {  //这是学生管理系统的主界面  System.***out***.println("--------欢迎来到学生管理系统--------");  System.***out***.println("1 查看所有学生");  System.***out***.println("2 添加学生");  System.***out***.println("3 删除学生");  System.***out***.println("4 修改学生");  System.***out***.println("5 退出");  System.***out***.println("请输入你的选择：");  //创建键盘录入对象  Scanner sc = **new** Scanner(System.***in***);  String choiceString = sc.nextLine();  //用switch语句实现选择  **switch**(choiceString) {  **case** "1":  //查看所有学生  *findAllStudent*(fileName);  **break**;  **case** "2":  //添加学生  *addStudent*(fileName);  **break**;  **case** "3":  //删除学生  *deleteStudent*(fileName);  **break**;  **case** "4":  //修改学生  *updateStudent*(fileName);  **break**;  **case** "5":  **default**:  System.***out***.println("谢谢你的使用");  System.*exit*(0); //JVM退出  **break**;  }  }  }    /\*\*  \* **@Title**: readData  \* **@Description**: 从文件中读数据到集合  \* **@param** fileName  \* **@param** array  \* **@throws** IOException  \*/  **public** **static** **void** readData(String fileName, ArrayList<Student> array)  **throws** IOException {  // 创建输入缓冲流对象  BufferedReader br = **new** BufferedReader(**new** FileReader(fileName));  String line;  **while** ((line = br.readLine()) != **null**) {  String[] datas = line.split(",");  Student s = **new** Student();  s.setId(datas[0]);  s.setName(datas[1]);  s.setAge(datas[2]);  s.setAddress(datas[3]);  array.add(s);  }  br.close();  }  /\*\*  \* **@Title**: writeData  \* **@Description**: 把集合中的数据写入文件  \* **@param** fileName  \* **@param** array  \* **@throws** IOException  \*/  **public** **static** **void** writeData(String fileName, ArrayList<Student> array)  **throws** IOException {  // 创建输出缓冲流对象  BufferedWriter bw = **new** BufferedWriter(**new** FileWriter(fileName));  **for** (**int** x = 0; x < array.size(); x++) {  Student s = array.get(x);  StringBuilder sb = **new** StringBuilder();  sb.append(s.getId()).append(",").append(s.getName()).append(",")  .append(s.getAge()).append(",").append(s.getAddress());  bw.write(sb.toString());  bw.newLine();  bw.flush();  }  bw.close();  }    /\*\*  \* **@Title**: updateStudent  \* **@Description**: 修改学生  \* **@param** fileName  \* **@throws** IOException  \*/  **public** **static** **void** updateStudent(String fileName) **throws** IOException {  //创建集合对象  ArrayList<Student> array = **new** ArrayList<Student>();  //从文件中把数据读取到集合中  *readData*(fileName, array);    //修改学生的思路：键盘录入一个学号，到集合中去查找，看是否有学生使用的是该学号，如果有就修改该学生  //创建键盘录入对象  Scanner sc = **new** Scanner(System.***in***);  System.***out***.println("请输入你要修改的学生的学号：");  String id = sc.nextLine();    //定义一个索引  **int** index = -1;    //遍历集合  **for**(**int** x=0; x<array.size(); x++) {  //获取每一个学生对象  Student s = array.get(x);  //拿学生对象的学号和键盘录入的学号进行比较  **if**(s.getId().equals(id)) {  index = x;  **break**;  }  }    **if**(index == -1) {  System.***out***.println("不好意思,你要修改的学号对应的学生信息不存在,请回去重新你的选择");  }**else** {  System.***out***.println("请输入学生新姓名：");  String name = sc.nextLine();  System.***out***.println("请输入学生新年龄：");  String age = sc.nextLine();  System.***out***.println("请输入学生新居住地：");  String address = sc.nextLine();    //创建学生对象  Student s = **new** Student();  s.setId(id);  s.setName(name);  s.setAge(age);  s.setAddress(address);    //修改集合中的学生对象  array.set(index, s);  //把集合中的数据重新写回到文件  *writeData*(fileName, array);  //给出提示  System.***out***.println("修改学生成功");  }  }    /\*\*  \* **@Title**: deleteStudent  \* **@Description**: 删除学生  \* **@param** fileName  \* **@throws** IOException  \*/  **public** **static** **void** deleteStudent(String fileName) **throws** IOException {  //创建集合对象  ArrayList<Student> array = **new** ArrayList<Student>();  //从文件中把数据读取到集合中  *readData*(fileName, array);    //删除学生的思路：键盘录入一个学号，到集合中去查找，看是否有学生使用的是该学号，如果有就删除该学生  //创建键盘录入对象  Scanner sc = **new** Scanner(System.***in***);  System.***out***.println("请输入你要删除的学生的学号：");  String id = sc.nextLine();    //我们必须给出学号不存在的时候的提示    //定义一个索引  **int** index = -1;    //遍历集合  **for**(**int** x=0; x<array.size(); x++) {  //获取到每一个学生对象  Student s = array.get(x);  //拿这个学生对象的学号和键盘录入的学号进行比较  **if**(s.getId().equals(id)) {  index = x;  **break**;  }  }    **if**(index == -1) {  System.***out***.println("不好意思,你要删除的学号对应的学生信息不存在,请回去重新你的选择");  }**else** {  array.remove(index);  //把集合中的数据重新写回到文件  *writeData*(fileName, array);  System.***out***.println("删除学生成功");  }    }    /\*\*  \* **@Title**: addStudent  \* **@Description**: 添加学生  \* **@param** fileName  \* **@throws** IOException  \*/  **public** **static** **void** addStudent(String fileName) **throws** IOException {  //创建集合对象  ArrayList<Student> array = **new** ArrayList<Student>();  //从文件中把数据读取到集合中  *readData*(fileName, array);    //创建键盘录入对象  Scanner sc = **new** Scanner(System.***in***);    //为了让id能够被访问到，我们就把id定义在了循环的外面  String id;    //为了让代码能够回到这里，用循环  **while**(**true**) {  System.***out***.println("请输入学生学号：");  //String id = sc.nextLine();  id = sc.nextLine();    //判断学号有没有被人占用  //定义标记  **boolean** flag = **false**;  //遍历集合，得到每一个学生  **for**(**int** x=0; x<array.size(); x++) {  Student s = array.get(x);  //获取该学生的学号，和键盘录入的学号进行比较  **if**(s.getId().equals(id)) {  flag = **true**; //说明学号被占用了  **break**;  }  }    **if**(flag) {  System.***out***.println("你输入的学号已经被占用,请重新输入");  }**else** {  **break**; //结束循环  }  }    System.***out***.println("请输入学生姓名：");  String name = sc.nextLine();  System.***out***.println("请输入学生年龄：");  String age = sc.nextLine();  System.***out***.println("请输入学生居住地：");  String address = sc.nextLine();    //创建学生对象  Student s = **new** Student();  s.setId(id);  s.setName(name);  s.setAge(age);  s.setAddress(address);    //把学生对象作为元素添加到集合  array.add(s);  //把集合中的数据重新写回到文件  *writeData*(fileName, array);    //给出提示  System.***out***.println("添加学生成功");  }    /\*\*  \* **@Title**: findAllStudent  \* **@Description**: 查看所有学生  \* **@param** fileName  \* **@throws** IOException  \*/  **public** **static** **void** findAllStudent(String fileName) **throws** IOException {  //创建集合对象  ArrayList<Student> array = **new** ArrayList<Student>();  //从文件中把数据读取到集合中  *readData*(fileName, array);    //首先来判断集合中是否有数据，如果没有数据，就给出提示，并让该方法不继续往下执行  **if**(array.size() == 0) {  System.***out***.println("不好意思,目前没有学生信息可供查询,请回去重新选择你的操作");  **return**;  }    //\t 其实就是一个tab键的位置  System.***out***.println("学号\t姓名\t年龄\t居住地");  **for**(**int** x=0; x<array.size(); x++) {  Student s = array.get(x);  System.***out***.println(s.getId()+"\t"+s.getName()+"\t"+s.getAge()+"\t"+s.getAddress());  }  }  } |