|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **学校** | 吉林大学珠海学院 |  | **班级** | **软件7班** |  | **科目** | JAVA面向对象程序设计 |
| **学院** | 计算机学院 |  | **姓名** | 张子恒 |  | **日期** | 2023.\*.\* |
| **专业** | 软件工程 |  | **学号** | 04222119 |  | **实验名称** |  |

|  |
| --- |
| **实验内容**  第1题  代码（要有必要的注释说明程序思路）  package com.ex11;   import java.io.File;  import java.io.FileInputStream;  import java.io.FileOutputStream;  import java.io.IOException;  import java.io.ObjectInputStream;  import java.io.ObjectOutputStream;  import java.util.ArrayList;  import java.util.Collections;  import java.util.Scanner;   public class TestChengji {    private static final int *STUDENT\_NUMBER* = 3; //学生个数   private static final int *SCORE\_UP* = 100; //成绩上限   private static final int *SCORE\_DOWN* = 0; //成绩下限   public static Student creatStudent(File file) {   Scanner sc = new Scanner(System.*in*);   System.*out*.println("请输入姓名：");   String name = sc.next();   System.*out*.println("请输入年龄：");   int age = sc.nextInt();  //若年龄小于0，则重新读入   while (age < 0) {   System.*out*.println("错误，年龄不能小于零。请重新输入： ");  age = sc.nextInt();   }   System.*out*.println("请输入Java成绩和C语言成绩：");   int javaScore = sc.nextInt();   int cScore = sc.nextInt();    while (javaScore > *SCORE\_UP* || javaScore < *SCORE\_DOWN* || cScore > *SCORE\_UP* || cScore < *SCORE\_DOWN*) {   System.*out*.println("错误，成绩输入异常。请重新输入： ");  javaScore = sc.nextInt();   cScore = sc.nextInt();   }   return (new Student(name, age, javaScore, cScore));   }   public static void printStuentInformation(ArrayList<Student> s) {  double average = 0;   for (Student student : s) {   System.*out*.println(student);    average += student.getJavaScore() + student.getcScore();  }   System.*out*.println("学生的平均成绩为：" + average / (*STUDENT\_NUMBER* \* 2));  }    public static void writeStudentInformation(File file, ArrayList<Student> s) throws IOException {  //若该文件不存在， 则新建一个文件   if (!file.exists()) {   file.createNewFile();   }   ObjectOutputStream f = new ObjectOutputStream(new   FileOutputStream(file));   for (Student student : s) {   f.writeObject(student);   }   f.close();   }    public static ArrayList<Student> readStudentInformation(File file) throws ClassNotFoundException, IOException {   ArrayList<Student> s = new ArrayList<Student>();   if (!file.exists()) {   throw new IOException("无法打开学生数据文件！");   }   ObjectInputStream f = new ObjectInputStream(new   FileInputStream(file));   for (int i = 0; i < *STUDENT\_NUMBER*; ++i) {   s.add((Student) (f.readObject()));   }   f.close();   return s;   }    */\*\*  \* 对student进行按成绩大小从小到大排序  \*  \** ***@param*** *s 学生数组  \*/* public static void sortScore(ArrayList<Student> s) {  //按score大小进行排序   Collections.*sort*(s, (n1, n2) -> ((n1.getcScore() + n1.getJavaScore()) - (n2.getcScore() + n2.getJavaScore())));   }    public static void main(String[] args) {   ArrayList<Student> student = new ArrayList<Student>();  File file = new File("D:\\studentInformation.txt");  try {   for (int i = 0; i < *STUDENT\_NUMBER*; ++i) {   System.*out*.println("请输入第" + (i + 1) + "个学生的信息：");    student.add(*creatStudent*(file));   }   *writeStudentInformation*(file, student);   student = *readStudentInformation*(file);   *sortScore*(student);   *printStuentInformation*(student);   } catch (ClassNotFoundException e) {   e.printStackTrace();   } catch (IOException e) {   e.printStackTrace();   }   }  }  package com.ex11;  import java.io.\*; import java.util.ArrayList; import java.util.Collections;   public class Student implements Serializable {    private static final long *serialVersionUID* = 1L; //序列化ID    private String name; //姓名   private int age; //年龄   private int javaScore; //java成绩   private int cScore;//c语言成绩    public Student() {   }    public Student(String name, int age, int javaScore, int cScore) {   this.name = name;   this.age = age;   this.javaScore = javaScore;   this.cScore = cScore;   }   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 getJavaScore() {   return javaScore;   }    public void setJavaScore(int javaScore) {   this.javaScore = javaScore;   }    public int getcScore() {   return cScore;   }    public void setcScore(int cScore) {   this.cScore = cScore;   }   @Override   public String toString() {    return new String("学生" + name + "的年龄为" + age + "，Java成绩为" + javaScore + "，C语言成绩为" + cScore + "。");   }  }  运行结果截图    第2题  代码（要有必要的注释说明程序思路）  运行结果截图  第3题  代码  运行结果截图  第4题  代码  运行结果截图  第5题  代码  运行结果截图 |
| **小结** |