

1. 阶段六(字符流)

1.1. 知识点

- 字符流相关的读写操作,主要使用BufferedReader,BufferWriter

1.2. 目前程序状况

- 只能在内存中进行学生的增删改查
- 一旦重新运行,只能重新添加学生
- 不能够持久化

我们之前是这么做的,每次运行的时候都去创建几个学生对象,来作为数据源

```

    */
    public class StudentData {
        public static final Student[] STUDS = new Student[20];
        public static final String[] COLUMNS;
        static {
            // 初始化
            init();
            // 表格列数据
            COLUMNS = new String[]{"学号", "姓名", "性别", "学校", "专业", "年龄", "城市", "手机号", "电子邮箱"};
        }

        /**
         * 初始化数据源数据
         * @since 9:27
         * @author wuguidong@cskaoyan.onaliyun.com
         */
        private static void init() {
            // 插入学生对象
            STUDS[0] = new Student( stuld: "1", name: "李明", gender: "男", school: "北京大学", major: "软件工程", age: "18", city: "武汉", phone: "13817618
            STUDS[1] = new Student( stuld: "2", name: "马铭", gender: "男", school: "华中师范大学", major: "计算机科学与技术", age: "27", city: "武汉", phone:
            STUDS[2] = new Student( stuld: "3", name: "周杰伦", gender: "男", school: "华中科技大学", major: "计算机科学与技术", age: "16", city: "武汉", pho
            STUDS[3] = new Student( stuld: "4", name: "周杰", gender: "男", school: "华中科技大学", major: "软件工程", age: "33", city: "武汉", phone: "13847
            STUDS[4] = new Student( stuld: "5", name: "陈奕迅", gender: "男", school: "清华大学", major: "计算机科学与技术", age: "17", city: "武汉", phone: "
            STUDS[5] = new Student( stuld: "6", name: "蓝若惜", gender: "女", school: "清华大学", major: "计算机科学与技术", age: "16", city: "武汉", phone: "
            STUDS[6] = new Student( stuld: "7", name: "苏铭", gender: "男", school: "复旦大学", major: "软件工程", age: "43", city: "深圳", phone: "13523281
            STUDS[7] = new Student( stuld: "8", name: "罗峰", gender: "男", school: "武汉大学", major: "机械工程", age: "9", city: "武汉", phone: "137131376
            STUDS[8] = new Student( stuld: "9", name: "萧炎", gender: "男", school: "武汉大学", major: "土木工程", age: "22", city: "北京", phone: "13512381
            STUDS[9] = new Student( stuld: "10", name: "上官若", gender: "男", school: "清华大学", major: "计算机科学与技术", age: "31", city: "长沙", phone:
            STUDS[10] = new Student( stuld: "11", name: "李明", gender: "男", school: "武汉理工大学", major: "计算机科学与技术", age: "46", city: "长沙", pho
        }
    }

```

现在把学生信息都保存在student.txt文件中,以文件中的数据作为数据源

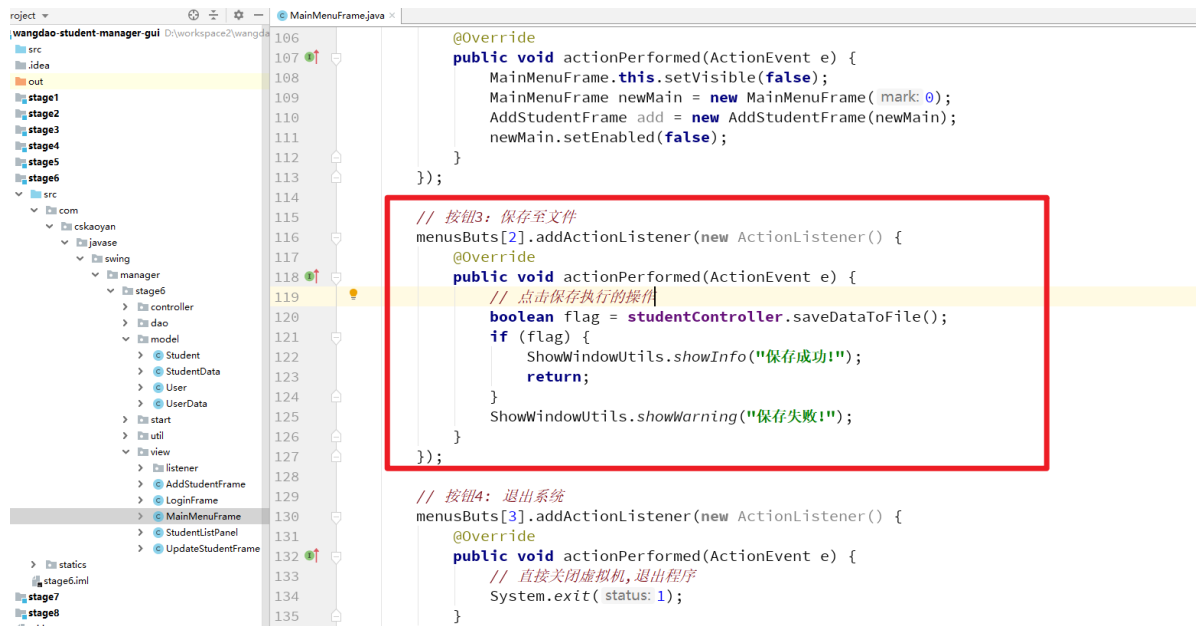
这样的话,就可以把内存中的学生数组,都给持久化的保存到文件当中,即使程序停止运行,数据还是保存在文件里面的,下次我们重新运行,从文件中读取数据,再把它还原成学生对象.

1.3. 改进方案(需求)

1.3.1. 整体思路

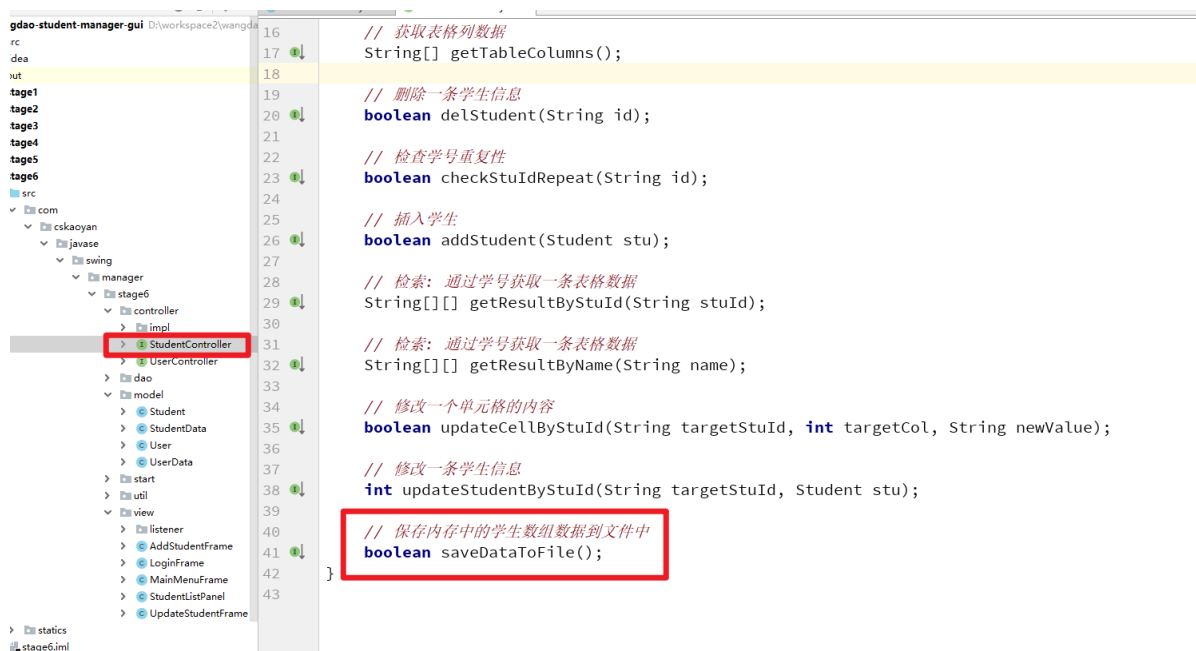
- 利用字符流(BufferedReader,BufferedWriter) 去处理文件数据
 - 利用BufferedReader中的readLine方法,一次读取一行数据,一行放置一个学生的信息
 - 利用String相关API去解析出来每个学生对象的各个数据,将其还原成一个个学生对象,放进数组中
 - 我们对数组进行学生对象的增删改查操作,操作完成后遍历数组,一个一个学生对象转化成字符串写入文件当中,一个对象一行

在这里,前置的一些条件已经帮大家设置好了,当点击这个按钮时,会去执行下方红色区域中的代码

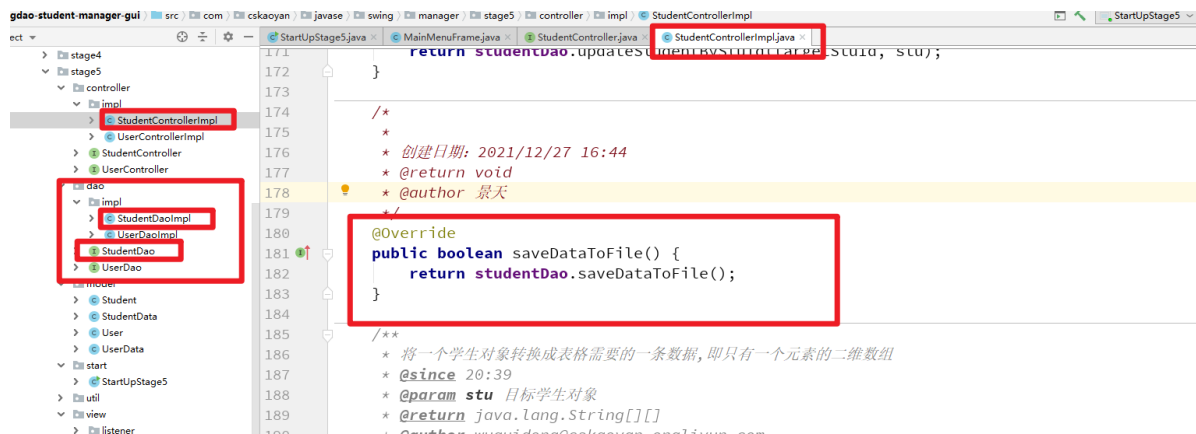


```
// 我们需要关心的是  
// 点击保存执行的操作,即去执行StudentController里面的saveDataToFile()这个方法,得到一个  
// boolean的返回值,给出成功或失败的提示信息,这一步已经做好了  
// 关键的是去看需要我们写代码的地方saveDataToFile();  
boolean flag = studentController.saveDataToFile();  
if (flag) {  
    ShowWindowUtils.showInfo("保存成功!");  
} else {  
    ShowWindowUtils.showWarning("保存失败!");  
}
```

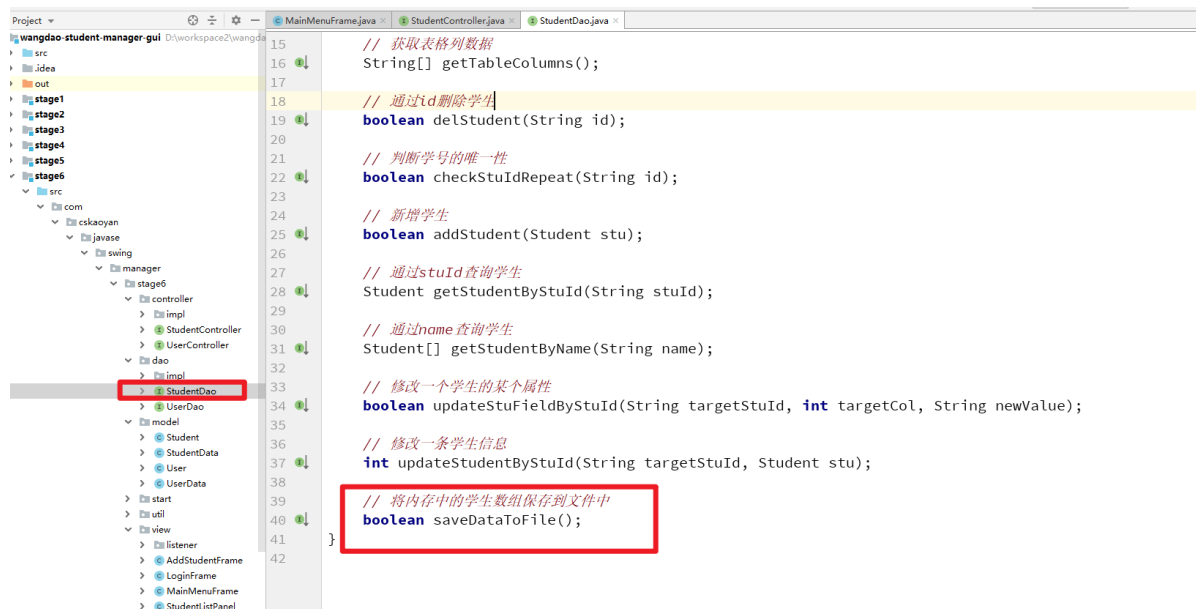
找到StudentController接口中的saveDataToFile()方法,我们要做的就是impl包中的StudentControllerImpl实现该方法



在StudentControllerImpl中该方法的实现是调用了dao的studentDao的saveDataToFile()方法



在StudentDao接口中定义的这个saveDataToFile方法,就是我们需要实现业务逻辑(保存数据到文件)的方法



在StudentDaoImpl中的saveToFile()方法就是我们需要写代码的地方

- 创建输出流对象
- 遍历StudentData中的STUDS数组
- 将每个student对象转化为字符串
- 依次写入student.txt文件中
- OK返回true,异常返回false
- 记得close

