

java搭建学生管理系统

创建对象

```
package ruanjian;

public class Student {
    private String id;
    private String name;
    private int age;
    private String address;
    public Student(){

    }
    public Student(String id, String name, int age, String address) {
        this.id = id;
        this.name = name;
        this.age = age;
        this.address = address;
    }
    public String getId() {
        return id;
    }
    public void setId(String id) {
        this.id = id;
    }
    public String getName() {

        return name;
    }

    public int getAge() {
        return age;
    }

    public void setAge(int age) {
        this.age = age;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getAddress() {
        return address;
    }

    public void setAddress(String address) {
        this.address = address;
    }
}
```

搭建选择界面

```
package ruanjian;

import java.util.Scanner;

public class StudentSystem {
    public static void main(String[] args) {
        loop: while (true) {
            System.out.println("-----欢迎来到学生管理系统-----");
            System.out.println("1:添加学生");
            System.out.println("2:删除学生");
            System.out.println("3:修改学生");
            System.out.println("4:查询学生");
            System.out.println("5:退出系统");
            Scanner sc = new Scanner(System.in);
            String choice = sc.next();
            switch (choice) {
                case "1":System.out.println("添加学生");
                    break;
                case "2":System.out.println("删除学生");
                    break;
                case "3": System.out.println("修改学生");
                    break;
                case "4": System.out.println("查询学生");
                    break;
                case "5":    System.out.println("退出系统");
                    //break loop;//第一种退出方法
                    System.exit(0);//退出整个虚拟机 第二种退出方法

                default:
                    System.out.println("没有这个选项");
            }/**/
        }

    }
}
```

(1)创建四个方法

```
public static void addStudent(){
    System.out.println("添加学生");
}
public static void deletStudent(){
    System.out.println("删除学生");
}
public static void updateStudent(){
    System.out.println("修改学生");
}
public static void queryStudent(){
    System.out.println("查询学生");
}
}
```

修改整个代码

```
package ruanjian;

import java.util.Scanner;

public class StudentsSystem {
    public static void main(String[] args) {
        loop: while (true) {
            System.out.println("-----欢迎来到学生管理系统-----");
            System.out.println("1:添加学生");
            System.out.println("2:删除学生");
            System.out.println("3:修改学生");
            System.out.println("4:查询学生");
            System.out.println("5:退出系统");
            Scanner sc = new Scanner(System.in);
            String choice = sc.next();
            switch (choice) {
                case "1":addStudent();
                break;
                case "2":deletStudent();
                break;
                case "3": updateStudent();
                break;
                case "4": queryStudent();
                break;
                case "5":    System.out.println("退出系统");
                //break loop;//第一种退出方法
                System.exit(0);//退出整个虚拟机 第二种退出方法

                default:
                    System.out.println("没有这个选项");
            }/**/
        }
    }
    //添加学生
    public static void addStudent(){
        System.out.println("添加学生");
    }
}
```

```

    }
    public static void deletStudent(){
        System.out.println("删除学生");
    }
    public static void updateStudent(){
        System.out.println("修改学生");
    }
    public static void queryStudent(){
        System.out.println("查询学生");
    }
}

```

查找和添加

(1)创建集合

```
ArrayList<Student> students = new ArrayList<>();
```

(2)查询功能的实现

需求

查询功能：

打印所有的学生信息，需要满足以下要求

- 如果没有学生信息，提示：当前无学生信息，请添加后再查询
- 如果有学生信息，需要按照以下格式输出。（不用过于纠结对齐的问题）

id	姓名	年龄	家庭住址
heima001	张三	23	南京
heima002	李四	24	北京
heima003	王五	25	广州
heima004	赵六	26	深圳

```

public static void queryStudent( ArrayList<Student> students ){
    if(students.size() == 0){
        System.out.println("当前表头无信息,请添加学生信息");
        return;
    }//如果无表头信息则退出方法
    System.out.println("id\t姓名\t年龄\t家庭住址");
    for (int i = 0; i < students.size(); i++) {
        Student stu=students.get(i);//创建对象Student 名为stu

        System.out.println(stu.getId()+"\t"+stu.getName()+"\t"+stu.getAge()+"\t"+stu.getAddress());
    }
}

```

(3)添加功能的实现

添加功能:

键盘录入每一个学生信息并添加，需要满足以下要求：

1. id唯一

```
//加入到创建的集合中
ArrayList<Student> students = new ArrayList<>(); //创建的集合
students.add(s);
```

id值不唯一的情况

```
public static void addStudent( ArrayList<Student> students ){
    Scanner sc = new Scanner(System.in);
    Student s = new Student();
    System.out.println("请输入学生id");
    String id=sc.next();
    s.setId(id);
    //学生姓名
    System.out.println("请输入学生姓名");
    String name=sc.next();
    s.setName(name);
    //学生年龄
    System.out.println("请输入学生年龄");
    int age=sc.nextInt();
    s.setAge(age);
    //学生地址
    System.out.println("请输入学生地址");
    String address=sc.next();
    s.setAddress(address);
    //加入到创建的集合中
    students.add(s);
    System.out.println("学生信息添加成功");
}
```

查询id是否重复

```
public static boolean contains(ArrayList<Student> students , String id){
    for (int i = 0; i < students.size(); i++) {
        Student stu=students.get(i);
        if(stu.getId().equals(id)){//字符串的比较boolean equals
            return true;//不重复返回ture
        }
    }
    return false;//重复返回false
}
```

带入添加中

```

String id=null;
while (true) {
    System.out.println("请输入学生id");
    id=sc.next();
    boolean flag = contains(students,id);
    if (flag) {
        System.out.println("id存在,请重新输入id");
    } else {
        s.setId(id);
        break;
    }
}

```

整体完成

```

public static void addStudent( ArrayList<Student> students ){
    Scanner sc = new Scanner(System.in);
    Student s = new Student();
    String id=null;//因为需要进入循环判断所以先赋Null值
    while (true) {
        System.out.println("请输入学生id");
        id=sc.next();
        boolean flag = contains(students,id);
        if (flag) {
            System.out.println("id存在,请重新输入id");
        } else {
            s.setId(id);
            break;
        }
    }
    //学生姓名
    System.out.println("请输入学生姓名");
    String name=sc.next();
    s.setName(name);
    //学生年龄
    System.out.println("请输入学生年龄");
    int age=sc.nextInt();
    s.setAge(age);
    //学生地址
    System.out.println("请输入学生地址");
    String address=sc.next();
    s.setAddress(address);
    //加入到创建的集合中
    students.add(s);
    System.out.println("学生信息添加成功");
}

```

删除和修改

(1) 删除

利用id获取索引

```
public static int getIndex(ArrayList<Student> students , String id) {
    for (int i = 0; i < students.size(); i++) {
        Student stu = students.get(i); //得到每一个学生对象
        if (stu.getId().equals(id)) {
            return i; //如果id存在则返回索引
        }
    }
    return -1; //若未找到则不存在则返回值为-1
}
```

改进 contains方法

```
public static boolean contains(ArrayList<Student> students , String id){
    /* for (int i = 0; i < students.size(); i++) {
        Student stu=students.get(i);
        if(stu.getId().equals(id)){//字符串的比较
            return true; //不重复返回ture
        }
    }
    return false; //重复返回false*/
    return getIndex(students,id)>=0;
}
```

删除方法的总体

```
public static void deletStudent( ArrayList<Student> students ){
    Scanner sc = new Scanner(System.in);
    System.out.println("请输入要删除的学生id");
    Student s = new Student();
    String id= sc.next();
    int index=getIndex(students,id);
    if(index>=0){
        students.remove(index);
        System.out.println("id为"+id+"已经被删除");
    }else{
        System.out.println("该学生不存在");
    }
}
```

(2)修改

修改的总体方法

```
public static void updateStudent( ArrayList<Student> students ){
    Scanner sc = new Scanner(System.in);
```

```

        System.out.println("请输入学生的id");
        String id= sc.next();
        int index=getIndex(students,id);
        if(index==-1){
            System.out.println("id为"+id+"的学生不存在");
            return ;
        }
        Student s=students.get(index);    //修改名字

        System.out.println("请输入新的名字");
        String name=sc.next();
        s.setName(name);
        System.out.println("请输入新的年龄");
        int age=sc.nextInt();
        s.setAge(age);
        System.out.println("请输入新的地址");
        String address=sc.next();
        s.setAddress(address);
        System.out.println("学生信息修改成功");
    }
}

```

总体

```

package ruanjian;

import java.util.ArrayList;
import java.util.Scanner;

public class StudentsSystem {
    public static void main(String[] args) {
        ArrayList<Student> students = new ArrayList<>();
        loop: while (true) {
            System.out.println("-----欢迎来到学生管理系统-----");
            System.out.println("1:添加学生");
            System.out.println("2:删除学生");
            System.out.println("3:修改学生");
            System.out.println("4:查询学生");
            System.out.println("5:退出系统");
            Scanner sc = new Scanner(System.in);
            String choice = sc.next();
            switch (choice) {
                case "1":addStudent(students);
                    break;
                case "2":deletStudent(students);
                    break;
                case "3": updateStudent(students);
                    break;
                case "4": queryStudent(students);
                    break;
                case "5":    System.out.println("退出系统");
                    //break loop;//第一种退出方法
                    System.exit(0);//退出整个虚拟机 第二种退出方法

                default:

```



```

        System.out.println("没有这个选项");
    }/**/
}
}
//添加学生
public static void addStudent( ArrayList<Student> students ){
    Scanner sc = new Scanner(System.in);
    Student s = new Student();
    String id=null;
    while (true) {
        System.out.println("请输入学生id");
        id=sc.next();
        boolean flag = contains(students,id);
        if (flag) {
            System.out.println("id存在,请重新输入id");
        } else {
            s.setId(id);
            break;
        }
    }
    //学生姓名
    System.out.println("请输入学生姓名");
    String name=sc.next();
    s.setName(name);
    //学生年龄
    System.out.println("请输入学生年龄");
    int age=sc.nextInt();
    s.setAge(age);
    //学生地址
    System.out.println("请输入学生地址");
    String address=sc.next();
    s.setAddress(address);
    //加入到创建的集合中
    students.add(s);
    System.out.println("学生信息添加成功");
}

public static void deletStudent( ArrayList<Student> students ){
    Scanner sc = new Scanner(System.in);
    System.out.println("请输入要删除的学生id");
    Student s = new Student();
    String id= sc.next();
    int index=getIndex(students,id);
    if(index>=0){
        students.remove(index);
        System.out.println("id为"+id+"已经被删除");
    }else{
        System.out.println("该学生不存在");
    }
}

public static void updateStudent( ArrayList<Student> students ){
    Scanner sc = new Scanner(System.in);
    System.out.println("请输入学生的id");
    String id= sc.next();
    int index=getIndex(students,id);
    if(index== -1){
        System.out.println("id为"+id+"的学生不存在");
    }
}

```

```

        return ;
    }
    Student s=students.get(index);    //修改名字

    System.out.println("请输入新的名字");
    String name=sc.next();
    s.setName(name);
    System.out.println("请输入新的年龄");
    int age=sc.nextInt();
    s.setAge(age);
    System.out.println("请输入新的地址");
    String address=sc.next();
    s.setAddress(address);
    System.out.println("学生信息修改成功");
}

public static void queryStudent( ArrayList<Student> students ){
    if(students.size() == 0){
        System.out.println("当前表头无信息,请添加学生信息");
        return;
    }//若无表头信息则退出方法
    System.out.println("id\t姓名\t年龄\t家庭住址");
    for (int i = 0; i < students.size(); i++) {
        Student stu=students.get(i);//创建对象Student 名为stu

        System.out.println(stu.getId()+"\t"+stu.getName()+"\t"+stu.getAge()+"\t"+stu.getAddress());
    }
}

//查询id是否重复
public static boolean contains(ArrayList<Student> students , String id){
    /* for (int i = 0; i < students.size(); i++) {
        Student stu=students.get(i);
        if(stu.getId().equals(id)){//字符串的比较
            return true;//不重复返回ture
        }
    }
    return false;//重复返回false*/
    return getIndex(students,id)>=0;
}

public static int getIndex(ArrayList<Student> students , String id) {
    for (int i = 0; i < students.size(); i++) {
        Student stu = students.get(i);//得到每一个学生对象
        if (stu.getId().equals(id)) {
            return i;//如果id存在则返回索引
        }
    }
    return -1;//若未找到则不存在则返回值为-1
}
}

```

