第09天 java集合

今日内容介绍

* 对象数组
* 集合类之ArrayList
* 学生管理系统案例

# 对象数组

## 对象数组概述

A:基本类型的数组:存储的元素为基本类型

int[] arr={1,2,3,4}

B:对象数组:存储的元素为引用类型

Student[] stus=new Student[3];

Student代表一个自定义类

Stus数组中stus[0],stus[1],stus[2]的元素数据类型为Student,

都可以指向一个Student对象

## 对象数组案例:

创建一个学生数组，存储三个学生对象并遍历

### 案例代码一:

**package** com.itheima;

/\*

\* 自动生成构造方法：

\* 代码区域右键 -- Source -- Generate Constructors from Superclass... 无参构造方法

\* 代码区域右键 -- Source -- Generate Constructor using Fields... 带参构造方法

\* 自动生成getXxx()/setXxx():

\* 代码区域右键 -- Source -- Generate Getters and Setters...

\*/

**public** **class** Student {

**private** String name;

**private** **int** age;

**public** Student() {

}

**public** Student(String name, **int** age) {

**this**.name = name;

**this**.age = age;

}

**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;

}

}

**package** com.itheima;

/\*

\* 创建一个学生数组，存储三个学生对象并遍历

\*

\* 分析：

\* A:定义学生类

\* B:创建学生数组

\* C:创建学生对象

\* D:把学生对象作为元素赋值给学生数组

\* E:遍历学生数组

\*/

**public** **class** StudentDemo {

**public** **static** **void** main(String[] args) {

//创建学生数组

Student[] students = **new** Student[3];

//创建学生对象

Student s1 = **new** Student("曹操",40);

Student s2 = **new** Student("刘备",35);

Student s3 = **new** Student("孙权",30);

//把学生对象作为元素赋值给学生数组

students[0] = s1;

students[1] = s2;

students[2] = s3;

//遍历学生数组

**for**(**int** x=0; x<students.length; x++) {

Student s = students[x];

//System.out.println(s);

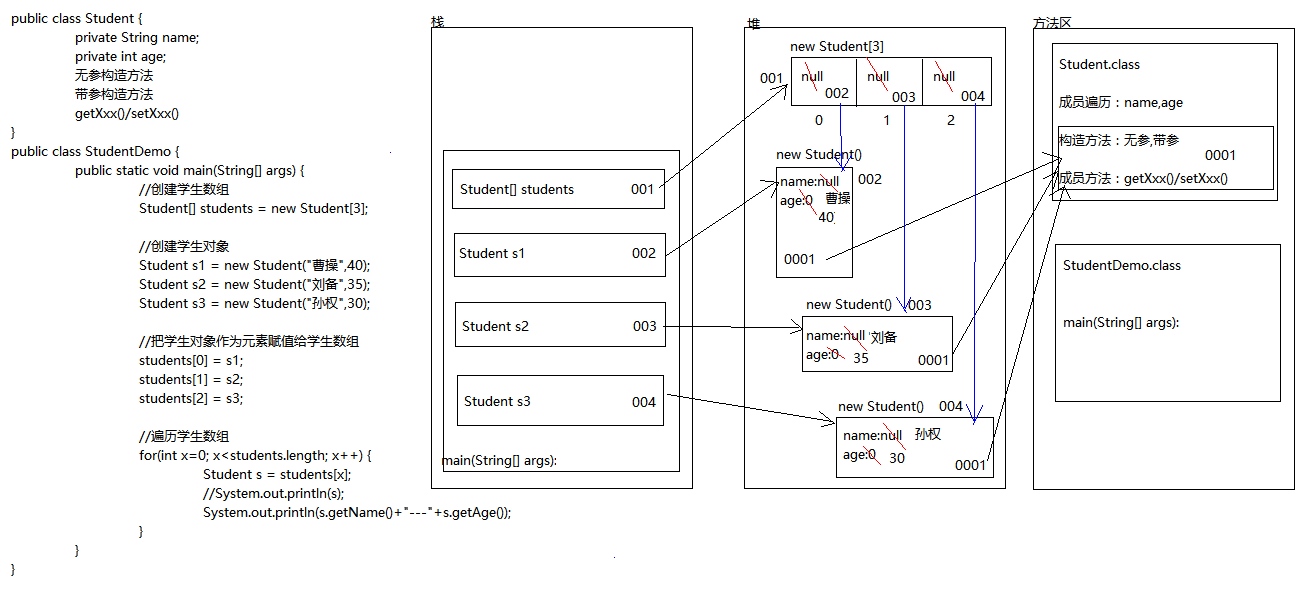
System.*out*.println(s.getName()+"---"+s.getAge());

}

}

}

## 对象数组的内存图



# 集合类之ArrayList

## 集合概述

A:我们学习的是面向对象编程语言，而面向对象编程语言对事物的描述都是通过对象来体现的。

为了方便对多个对象进行操作，我们就必须对这多个对象进行存储，而要想对多个对象进行存储， 就不能是一个基本的变量，而应该是一个容器类型的变量。

B:到目前为止，我们学习过了哪些容器类型的数据呢？

StringBuilder,数组。

StringBuilder的结果只能是一个字符串类型，不一定满足我们的需求。

所以，我们目前只能选择数组了，也就是我们前面学习过的对象数组。

但是，数组的长度是固定的， 如果有时候元素的个数不确定的,我们无法定义出数组的长度,这个时候，java就提供了集合类供我们使用。

## ArrayList集合

### ArrayList添加新元素

#### 案例代码二:

**package** com.itheima\_01;

**import** java.util.ArrayList;

/\*

\* 为什么会出现集合类：

\* 我们学习的是面向对象编程语言，而面向对象编程语言对事物的描述都是通过对象来体现的。

\* 为了方便对多个对象进行操作，我们就必须对这多个对象进行存储，而要想对多个对象进行存储，

\* 就不能是一个基本的变量，而应该是一个容器类型的变量。

\* 到目前为止，我们学习过了哪些容器类型的数据呢？StringBuilder,数组。

\* StringBuilder的结果只能是一个字符串类型，不一定满足我们的需求。

\* 所以，我们目前只能选择数组了，也就是我们前面学习过的对象数组。

\* 但是，数组的长度是固定的，适应不了变化的需求，那么，我们该如何选择呢?

\* 这个时候，java就提供了集合类供我们使用。

\*

\* 集合类的特点：

\* 长度可变。

\*

\* ArrayList<E>:

\* 大小可变数组的实现

\*

\* <E>:是一种特殊的数据类型，泛型。

\* 怎么用呢?

\* 在出现E的地方我们使用引用数据类型替换即可

\* 举例：ArrayList<String>,ArrayList<Student>

\*

\* 构造方法：

\* ArrayList()

\*

\* 添加元素：

\* public boolean add(E e):添加元素

\* public void add(int index,E element):在指定的索引处添加一个元素

\*/

**public** **class** ArrayListDemo {

**public** **static** **void** main(String[] args) {

//创建集合对象

ArrayList<String> array = **new** ArrayList<String>();

//add(E e):添加元素

array.add("hello");

array.add("world");

array.add("java");

//add(int index,E element):在指定的索引处添加一个元素

//array.add(1, "android");

System.*out*.println("array:"+array);

}

}

### ArrayList删改查方法

A:获取元素

public E get(int index):返回指定索引处的元素

B:集合长度

public int size():返回集合中的元素的个数

C:删除元素

public boolean remove(Object o):删除指定的元素，返回删除是否成功

public E remove(int index):删除指定索引处的元素，返回被删除的元素

D:修改元素

public E set(int index,E element):修改指定索引处的元素，返回被修改的元素

#### 案例代码三:

**package** com.itheima\_01;

**import** java.util.ArrayList;

/\*

\* 获取元素

\* public E get(int index):返回指定索引处的元素

\* 集合长度

\* public int size():返回集合中的元素的个数

\* 删除元素

\* public boolean remove(Object o):删除指定的元素，返回删除是否成功

\* public E remove(int index):删除指定索引处的元素，返回被删除的元素

\* 修改元素

\* public E set(int index,E element):修改指定索引处的元素，返回被修改的元素

\*/

**public** **class** ArrayListDemo2 {

**public** **static** **void** main(String[] args) {

//创建集合对象

ArrayList<String> array = **new** ArrayList<String>();

//添加元素

array.add("hello");

array.add("world");

array.add("java");

//public E get(int index):返回指定索引处的元素

//System.out.println("get:"+array.get(0));

//System.out.println("get:"+array.get(1));

//System.out.println("get:"+array.get(2));

//public int size():返回集合中的元素的个数

//System.out.println("size:"+array.size());

//public boolean remove(Object o):删除指定的元素，返回删除是否成功

//System.out.println("remove:"+array.remove("world"));//true

//System.out.println("remove:"+array.remove("world"));//false

//public E remove(int index):删除指定索引处的元素，返回被删除的元素

//System.out.println("remove:"+array.remove(0));

//public E set(int index,E element):修改指定索引处的元素，返回被修改的元素

System.*out*.println("set:"+array.set(1, "android"));

//输出

System.*out*.println("array:"+array);

}

}

### ArrayList遍历

集合的遍历思想和数组的遍历思想相同

循环遍历容器,依次取出里面的元素即可

#### 案例代码四:

**package** com.itheima\_01;

**import** java.util.ArrayList;

/\*

\* ArrayList集合的遍历

\* 通过size()和get()配合实现的

\*/

**public** **class** ArrayListDemo3 {

**public** **static** **void** main(String[] args) {

//创建集合对象

ArrayList<String> array = **new** ArrayList<String>();

//添加元素

array.add("hello");

array.add("world");

array.add("java");

//获取元素

//原始做法

System.*out*.println(array.get(0));

System.*out*.println(array.get(1));

System.*out*.println(array.get(2));

System.*out*.println("----------");

**for**(**int** x=0; x<3; x++) {

System.*out*.println(array.get(x));

}

System.*out*.println("----------");

//如何知道集合中元素的个数呢?size()

**for**(**int** x=0; x<array.size(); x++) {

System.*out*.println(array.get(x));

}

System.*out*.println("----------");

//最标准的用法

**for**(**int** x=0; x<array.size(); x++) {

String s = array.get(x);

System.*out*.println(s);

}

}

}

## ArrayList集合案例

### ArrayList练习之存储字符串并遍历

向集合中添加任意四个字符串,遍历集合,依次打印取出的字符串

#### 案例代码五:

**package** com.itheima\_02;

**import** java.util.ArrayList;

/\*

\* 存储字符串并遍历

\*

\* 分析：

\* A:创建集合对象

\* B:添加字符串元素

\* C:遍历集合

\*/

**public** **class** ArrayListTest {

**public** **static** **void** main(String[] args) {

//创建集合对象

ArrayList<String> array = **new** ArrayList<String>();

//添加字符串元素

array.add("向问天");

array.add("刘正风");

array.add("左冷禅");

array.add("风清扬");

//遍历集合

**for**(**int** x=0; x<array.size(); x++) {

String s = array.get(x);

System.*out*.println(s);

}

}

}

### ArrayList练习之获取满足要求的元素

给定一个字符串数组：{“张三丰”,“宋远桥”,“张无忌”,“殷梨亭”“张翠山”,“莫声谷”}，将数组中的元素添加到集合中，并把所有姓张的人员打印到控制台上

#### 案例代码六:

**package** com.itheima\_02;

**import** java.util.ArrayList;

/\*

\* 给定一个字符串数组：{“张三丰”,“宋远桥”,“张无忌”,“殷梨亭”,“张翠山”,“莫声谷”}，将数组中的元素添加到集合中，并把所有姓张的人员打印到控制台上。

\*

\* 分析：

\* A:定义字符串数组

\* B:创建集合对象

\* C:遍历字符串数组，获取到每一个字符串元素

\* D:把获取到的字符串元素添加到集合

\* E:遍历集合

\* 要判断每一个字符串元素是否以"张"开头，如果是，就输出在控制台

\*/

**public** **class** ArrayListTest2 {

**public** **static** **void** main(String[] args) {

//定义字符串数组

String[] strArray = {"张三丰","宋远桥","张无忌","殷梨亭","张翠山","莫声谷"};

//创建集合对象

ArrayList<String> array = **new** ArrayList<String>();

//遍历字符串数组，获取到每一个字符串元素

**for**(**int** x=0; x<strArray.length; x++) {

//把获取到的字符串元素添加到集合

array.add(strArray[x]);

}

//遍历集合

**for**(**int** x=0; x<array.size(); x++) {

String s = array.get(x);

//要判断每一个字符串元素是否以"张"开头，如果是，就输出在控制台

**if**(s.startsWith("张")) {

System.*out*.println(s);

}

}

}

}

### ArrayList练习之存储自定义对象并遍历

A:自定义一个学生类,学生中有姓名和年龄属性,生成满参构造与空参构造

生成属性对应的getter/setter方法

B:在测试类中使用满参构造创建三个学生对象,然后将每个学生对象均添加到ArrayList集合中

C:遍历这个ArrayList集合,依次打印出每个学生的姓名和年龄

#### 案例代码七:

**package** com.itheima\_02;

**public** **class** Student {

**private** String name;

**private** **int** age;

**public** Student() {

}

**public** Student(String name, **int** age) {

**this**.name = name;

**this**.age = age;

}

**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;

}

}

**package** com.itheima\_02;

**import** java.util.ArrayList;

/\*

\* 存储自定义对象并遍历

\*

\* 分析：

\* A:定义学生类

\* B:创建集合对象

\* C:创建学生对象

\* D:把学生对象作为元素添加到集合中

\* E:遍历集合

\*/

**public** **class** ArrayListTest3 {

**public** **static** **void** main(String[] args) {

//创建集合对象

ArrayList<Student> array = **new** ArrayList<Student>();

//创建学生对象

Student s1 = **new** Student("林青霞",28);

Student s2 = **new** Student("张曼玉",30);

Student s3 = **new** Student("景甜",25);

Student s4 = **new** Student("柳岩",18);

//把学生对象作为元素添加到集合中

array.add(s1);

array.add(s2);

array.add(s3);

array.add(s4);

//遍历集合

**for**(**int** x=0; x<array.size(); x++) {

Student s = array.get(x);

System.*out*.println(s.getName()+"---"+s.getAge());

}

}

}

### ArrayList练习之键盘录入数据存储并遍历

创建一个Student类包含姓名和年龄属性

创建一个ArrayList集合

向集合中添加三个Student对象Student对象中姓名和年龄的数据均来自与键盘录入

最终遍历这个集合,取出Student对象以及里面属性的值

#### 案例代码八:

**package** com.itheima\_03;

**public** **class** Student {

**private** String name;

**private** String age;

**public** Student() {

}

**public** Student(String name, String age) {

**this**.name = name;

**this**.age = age;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** String getAge() {

**return** age;

}

**public** **void** setAge(String age) {

**this**.age = age;

}

}

package com.itheima\_03;

import java.util.ArrayList;

import java.util.Scanner;

/\*

\* 创建一个集合，存储学生对象，学生对象的数据来自于键盘录入，最后，遍历集合

\*

\* 注意：为了方便使用，我把学生类中的所有成员定义为String类型

\*

\* 分析：

\* A:定义学生类

\* B:创建集合对象

\* C:键盘录入数据,创建学生对象,把键盘录入的数据赋值给学生对象的成员变量

\* D:把学生对象作为元素存储到集合中

\* E:遍历集合

\*

\*/

public class StudentDemo {

public static void main(String[] args) {

//创建集合对象

ArrayList<Student> array = new ArrayList<Student>();

/\*

//键盘录入数据,创建学生对象,把键盘录入的数据赋值给学生对象的成员变量

Scanner sc = new Scanner(System.in);

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();

Student s = new Student();

s.setName(name);

s.setAge(age);

//把学生对象作为元素存储到集合中

array.add(s);

\*/

//为了提高代码的复用性，我把键盘录入数据给学生对象，并存储到集合中的动作用一个方法来实现

//调用方法

addStudent(array);

addStudent(array);

addStudent(array);

//遍历集合

for(int x=0; x<array.size(); x++) {

Student s = array.get(x);

System.out.println(s.getName()+"---"+s.getAge());

}

}

/\*

\* 两个明确：

\* 返回值类型：void

\* 参数列表：ArrayList<Student> array

\*/

public static void addStudent(ArrayList<Student> array) {

//键盘录入数据,创建学生对象,把键盘录入的数据赋值给学生对象的成员变量

Scanner sc = new Scanner(System.in);

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();

Student s = new Student();

s.setName(name);

s.setAge(age);

//把学生对象作为元素存储到集合中

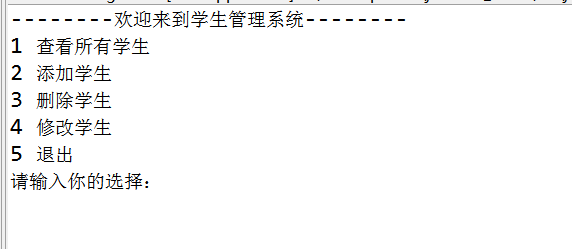
array.add(s);

}

}

# 学生管理系统案例

## 学生管理系统案例需求



利用集合完成对学生的增删改查四个功能

## 学生管理系统案例实现

### 创建学生类:

#### 案例代码九:

**package** com.itheima;

/\*

\* 这是我的学生类

\*/

**public** **class** Student {

//学号

**private** String id;

//姓名

**private** String name;

//年龄

**private** String age;

//居住地

**private** String address;

**public** Student() {

}

**public** Student(String id, String name, String 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** **void** setName(String name) {

**this**.name = name;

}

**public** String getAge() {

**return** age;

}

**public** **void** setAge(String age) {

**this**.age = age;

}

**public** String getAddress() {

**return** address;

}

**public** **void** setAddress(String address) {

**this**.address = address;

}

}

### 学生管理系统界面实现:

#### 案例代码十:

**package** com.itheima;

**import** java.util.ArrayList;

**import** java.util.Scanner;

/\*

\* 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

**break**;

**case** "2":

// 添加学生

**break**;

**case** "3":

// 删除学生

**break**;

**case** "4":

// 修改学生

**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**;

}

}

}

}

### 学生管理系统之查询所有学生功能

#### 案例代码十一:

**package** com.itheima.test;

**import** java.util.ArrayList;

**import** java.util.Scanner;

/\*

\* 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

**break**;

**case** "3":

//删除学生

**break**;

**case** "4":

//修改学生

**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**;

}

}

}

//查看所有学生

**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\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());

}

}

}

### 学生管理系统之添加学生功能

#### 案例代码十二:

**package** com.itheima;

**import** java.util.ArrayList;

**import** java.util.Scanner;

/\*

\* 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

**break**;

**case** "2":

//添加学生

*addStudent*(array);

**break**;

**case** "3":

//删除学生

**break**;

**case** "4":

//修改学生

**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**;

}

}

}

/\*

//添加学生

public static void addStudent(ArrayList<Student> array) {

//创建键盘录入对象

Scanner sc = new Scanner(System.in);

System.out.println("请输入学生学号：");

String id = sc.nextLine();

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");

}

\*/

//添加学生

**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");

}

}

### 学生管理系统之删除学生功能

#### 案例代码十三:

**package** com.itheima;

**import** java.util.ArrayList;

**import** java.util.Scanner;

/\*

\* 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

**break**;

**case** "2":

//添加学生

**break**;

**case** "3":

//删除学生

*deleteStudent*(array);

**break**;

**case** "4":

//修改学生

**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**;

}

}

}

//删除学生

**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();

/\*

//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)) {

array.remove(x); //根据索引删除

break;

}

}

//给出提示

System.out.println("删除学生成功");

\*/

//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");

}

}

}

### 学生管理系统之修改学生功能

#### 案例代码十四:

**package** com.itheima;

**import** java.util.ArrayList;

**import** java.util.Scanner;

/\*

\* 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

**break**;

**case** "2":

//添加学生

**break**;

**case** "3":

//删除学生

**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**;

}

}

}

//修改学生

**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");

}

}

}