数组

黑马程序员
www.itheima.com

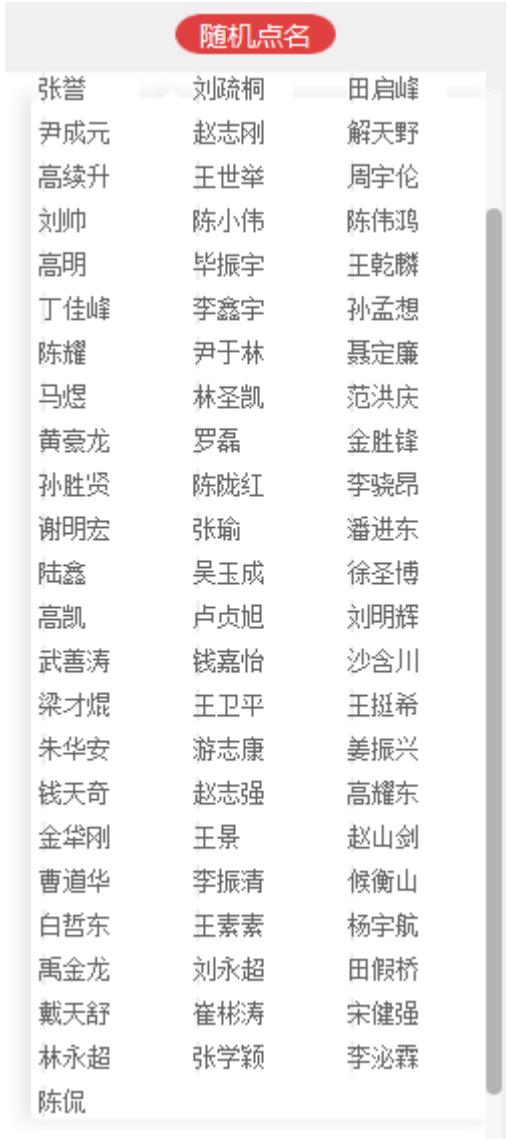
传智教育旗下
高端IT教育品牌

数组是什么

- 数组就是用来存储一批同种类型数据的容器。

例子

```
20, 10, 80, 60, 90

int[] arr = {20, 10, 80, 60, 90};



牛二，西门，全蛋

String[] names = {"牛二", "西门", "全蛋"};
```



假如用变量存储这些名字，然后完成随机点名功能，怎么实现？存在有什么问题？

```
String name1 = "张誉";
String name2 = "刘疏桐";
String name3 = "田启峰";
...
...
String name68= "张学颖";
String name69= "李沁霖";
String name70= "陈侃";
```

- **代码繁琐：大量变量的定义。**
- **业务功能实现麻烦。**

其实你真的很好
只是我们不合适

使用数组完成：

```
String[] names = {“张誉”, “刘疏桐”, “田启峰”，… “张学颖”, “李沁霖”, “陈侃”,};
```

结论：数组适合做一批同种类型数据的存储。

我超爱你的

关于数组同学们需要学会什么

怎么定义数组存储数据	怎么操作数组元素	怎么解决实际问题	数组内存原理	数组使用的注意点
在程序中如何去定义数组存储数据，具体格式是什么样的	怎么去获取，修改数组中的数据	怎么结合数组解决比如随机点名等一些实际业务	数组在内存中具体是怎么去工作的	避免入坑，同时可以在出现问题后立即排查问题所在

目录

Contents

静态初始化数组

- 定义数组的时候直接给数组赋值。

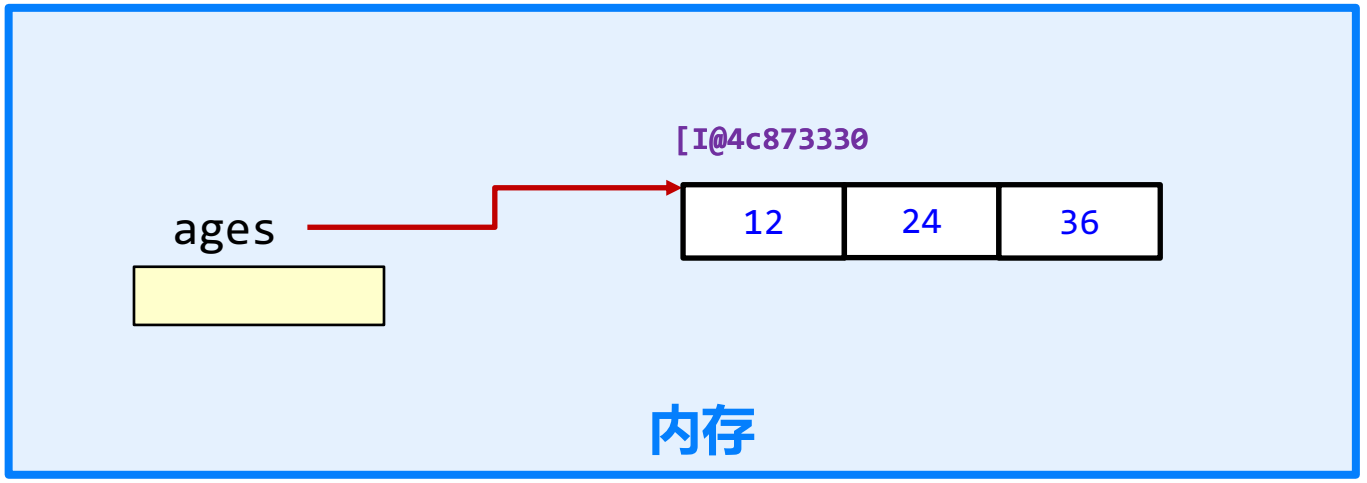
静态初始化数组的格式：

```
//完整格式
数据类型[]  数组名 = new 数据类型[]{元素1，元素2 ，元素3… };
double[] scores = new double[]{89.9, 99.5, 59.5, 88.0};
int[] ages = new int[]{12, 24, 36}
```

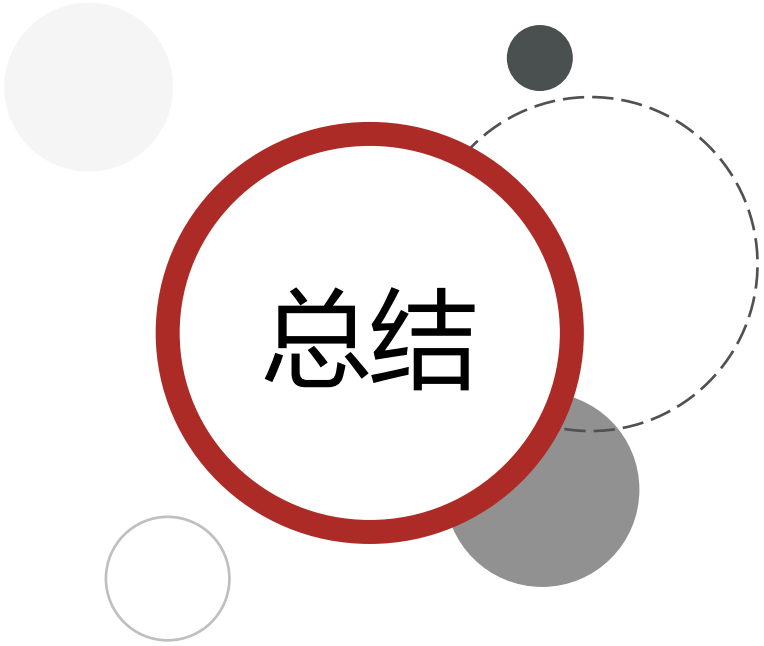
```
//简化格式
数据类型[] 数组名 = { 元素1，元素2 ，元素3，… };
int[] ages = {12, 24, 36};
```

数组的基本原理

```
int[] ages = {12，24，36};
```



注意：数组变量名中存储的是数组在内存中的地址，数组是引用类型。



1．数组的静态初始化的写法和特点什么样的？

```
数据类型[]  数组名 = { 元素1，元素2 ，元素3，… };
int[] ages = {12, 24, 36, 48, 60};
double[] scores = {89.9, 99.5, 59.5};

//完整格式
数据类型[]  数组名 = new 数据类型[]{ 元素1，元素2 ，元素3… };
int[] ages = new int[]{12, 24, 36, 48, 60};
```

2．数组是属于什么类型，数组变量名中存储的是什么？

- **引用数据类型，存储的数组在内存中的地址信息。**

目录

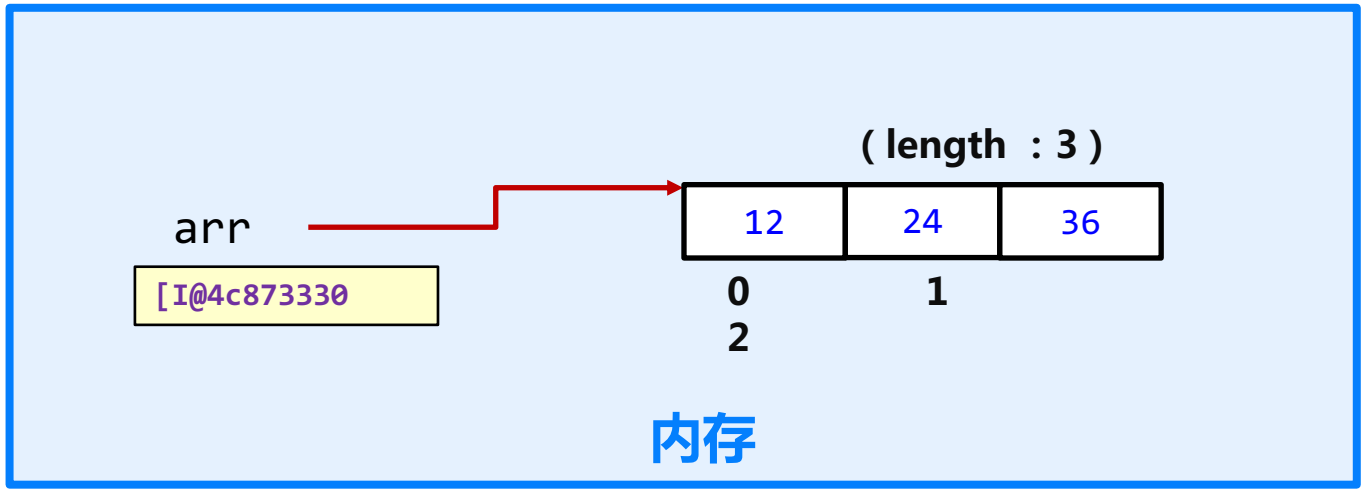
Contents

数组的访问

数组名称[索引]

```
// 取值
System.out.println(arr[0]); // 12

// 赋值
arr[2] = 100;
System.out.println(arr[2]); // 100
```

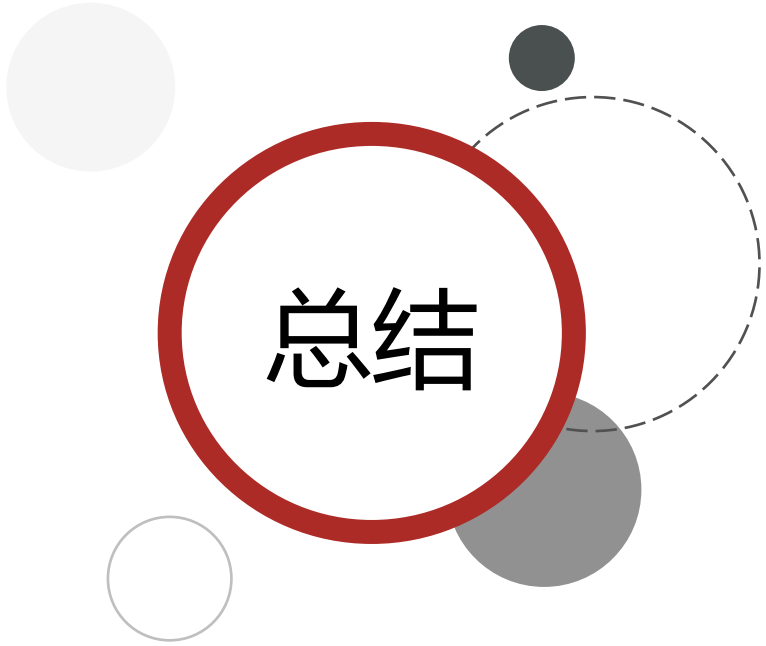
int[] arr = {12，24，36};



数组的长度属性：length

```
// 获取数组的长度（就是数组元素的个数）
System.out.println(arr.length); // 3
```

问题：数组的最大索引可以怎么表示？ `数组名. length - 1 // 前提：元素个数大于0`



1．如何访问数组的元素？

```
数组名称[索引]
```

2．如何访问数组的长度？

- **数组名称.length**

3．数组的最大索引怎么获取？

```
数组名.length - 1 // 前提：元素个数大于0
```

目录

Contents

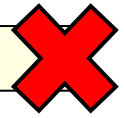
数组的几个注意事项：

- “数据类型[] 数组名”也可以写成 “数据类型 数组名[] ”。

```
int[] ages =...;
int ages[] =...;

double[] scores = ...;
double scores[] = ...;
```

- 什么类型的数组存放什么类型的数据，否则报错。

```
int[] arrs = new int[]{30, 40, "黑马"};
```



- 数组一旦定义出来，程序执行的过程中，长度、类型就固定了。

目录

Contents

数组的动态初始化

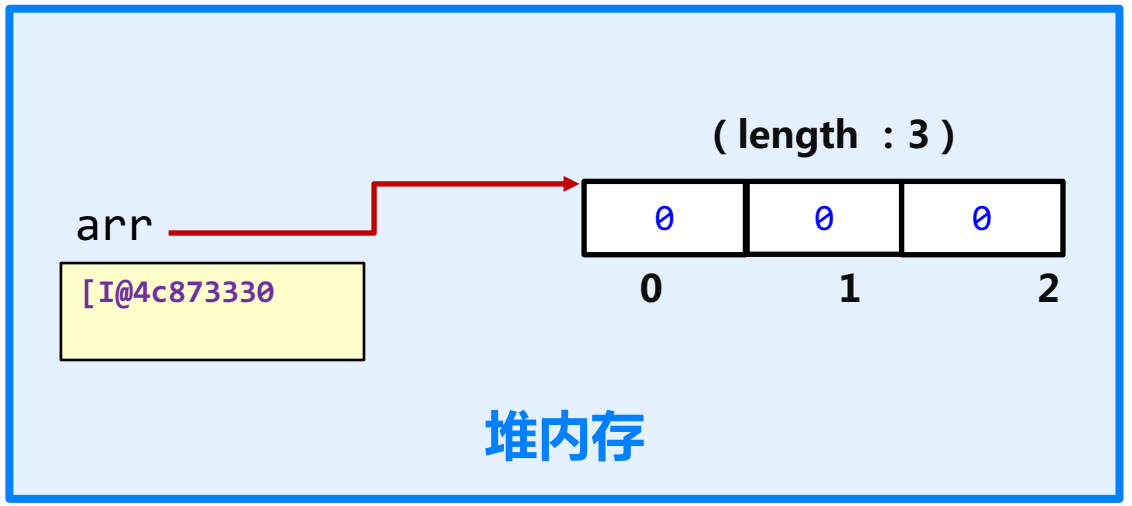
- 定义数组的时候只确定元素的类型和数组的长度，之后再存入具体数据。

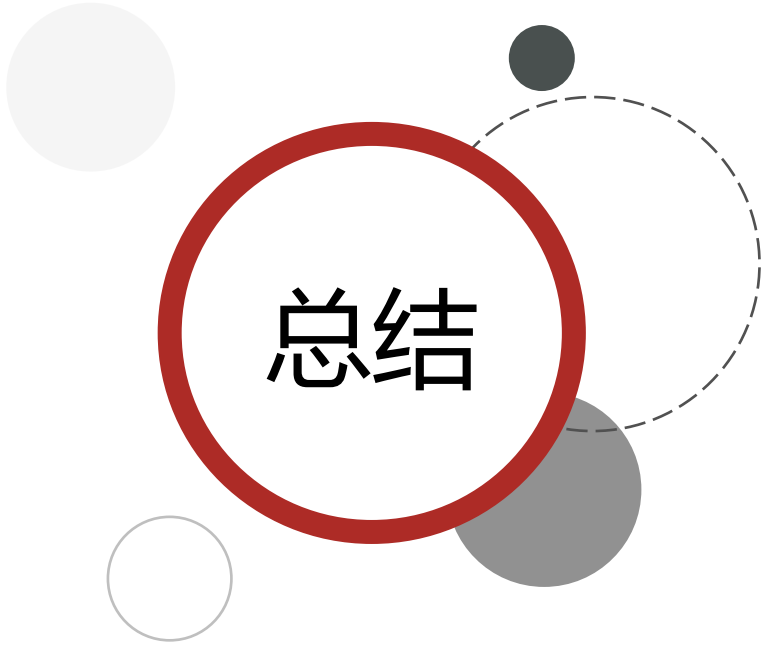
```
int[] arr = new int[3]
```

数组的动态初始化格式：

```
数据类型[]  数组名 = new 数据类型[长度];
int[] arr = new int[3];
```

```
// 后赋值
arr[0] = 10;
System.out.println(arr[0]); // 10
```





1、动态初始化的写法是什么样的？

```
数据类型[]  数组名 = new 数据类型[长度];
int[] ages = new int[4];
```

2、两种数组定义时的特点和场景有什么区别

- **当前已经知道存入的元素值，用静态初始化。**
- **当前还不清楚要存入哪些数据，用动态初始化。**

目录

Contents

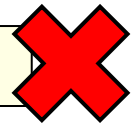
元素默认值规则

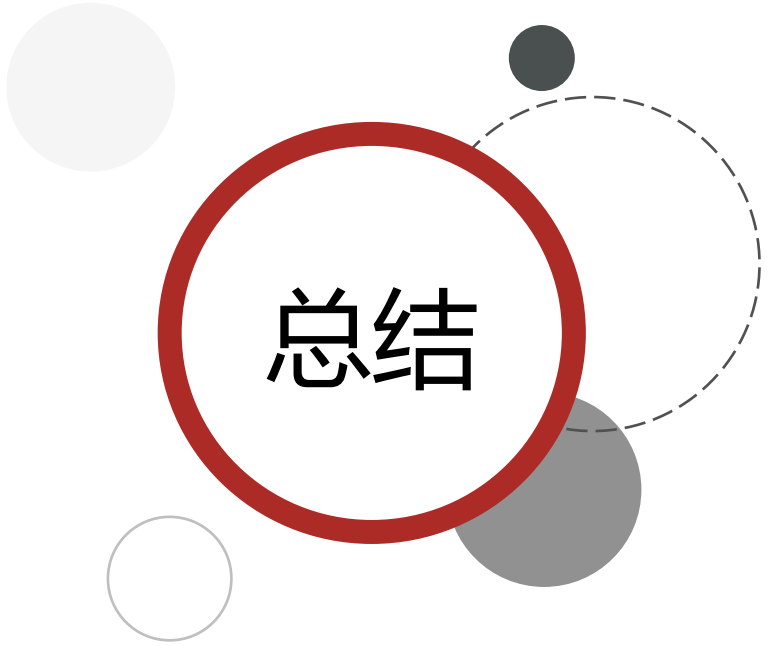
数据类型	明细	默认值
基本类型	byte、short、char、int、long	0
	float、double	0.0
	boolean	false
引用类型	类、接口、数组、String	null

两种初始化的的使用场景总结、注意事项说明：

- 动态初始化：只指定数组长度，后期赋值，适合开始知道数据的数量，但是不确定具体元素值的业务场景。
- 静态初始化：开始就存入元素值，适合一开始就能确定元素值的业务场景。
- 两种格式的写法是独立的，不可以混用。

```
int[] arrs = new int[3]{30，40,50};
```





1. 动态初始化数组后元素的默认值是什么样的？

- byte、short、int 、char、long类型数组元素的默认值都是0
- float、double类型数组元素的默认值都是0.0
- boolean类型数组元素的默认值是false、String类型数组元素的默认值是null

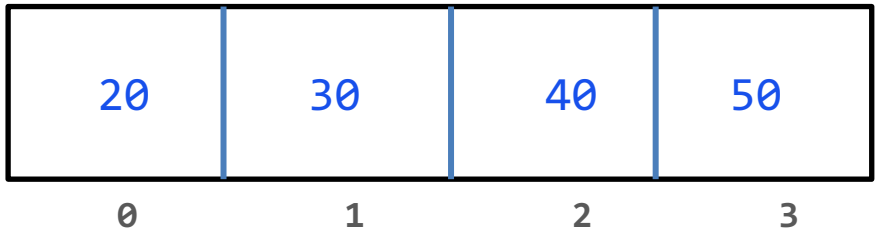
目录

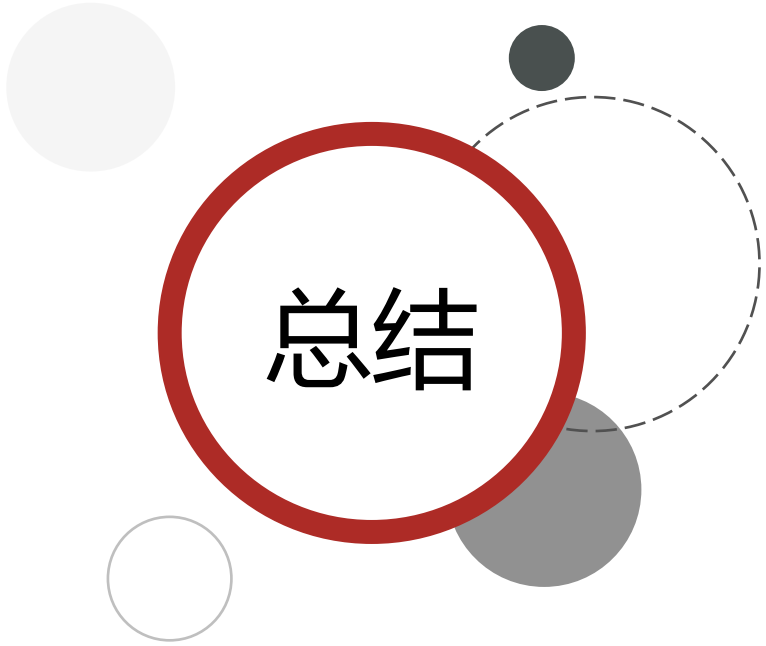
Contents

数组遍历介绍

- 遍历：就是一个一个数据的访问。
- 为什么要遍历？

```
int[] ages = {20, 30, 40, 50};
for (int i = 0; i < ages.length; i++) {
    System.out.println(ages[i]);
}
```





1. 什么是数组的遍历？

- **一个一个的访问数组中的数据。**

2. 如何遍历数组？

```
int[] ages = {20, 30, 40, 50};
for (int i = 0; i < ages.length; i++) {
    System.out.println(ages[i]);
}
```

目录

Contents

案例 数组遍历-求和

需求：某部门5名员工的销售额分别是：16、26、36、6、100，请计算出他们部门的总销售额。

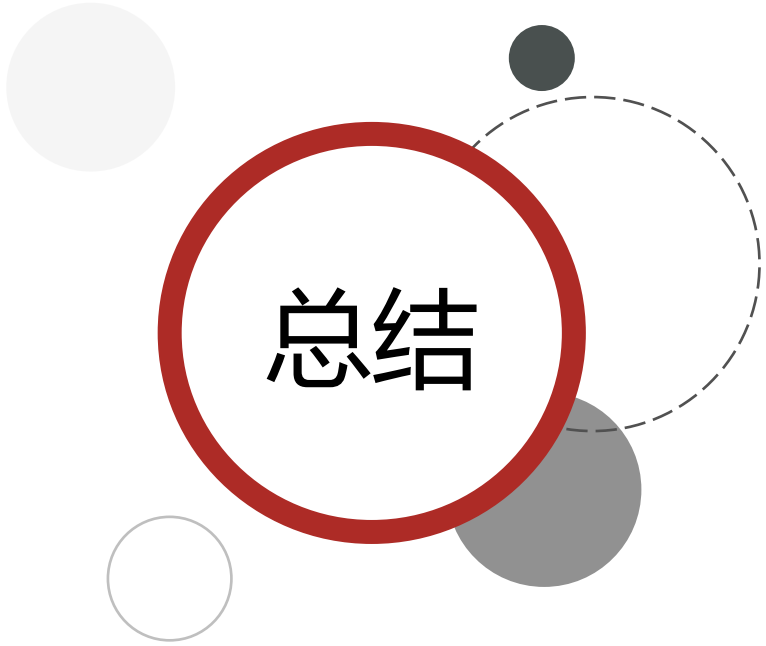
分析：

① 把这5个数据拿到程序中去 ---> 使用数组

```
int[] money = {16, 26, 36, 6, 100};
```

② 遍历数组中的每个数据，然后在外面定义求和变量把他们累加起来。

```
int sum = 0;
for (int i = 0; i < money.length; i++) {
    // i = 0 1 2 3 4
    sum += money[i];
}
```



1. 如何实现批量数据的求和?

- **使用数组存储批量数据**

```
int[] money = {16, 26, 36, 6, 100};
```

- **遍历数组中的每个数据，然后定义变量把他们累加起来。**

```
int sum = 0;
for (int i = 0; i < money.length; i++) {
    // i = 0 1 2 3 4
    sum += money[i];
}
```

目录

Contents

案例 数组元素求最大值

					
颜值：15	颜值：9000	颜值：10000	颜值：20000	颜值：9500	颜值：-5

案例 数组元素求最大值

颜值：15	颜值：9000	颜值：10000	颜值：20000	颜值：9500	颜值：-5

分析：

① 把颜值数据拿到程序中去，用数组装起来。

```
int[] faceScores = {15, 9000, 10000, 20000, 9500, -5};
```

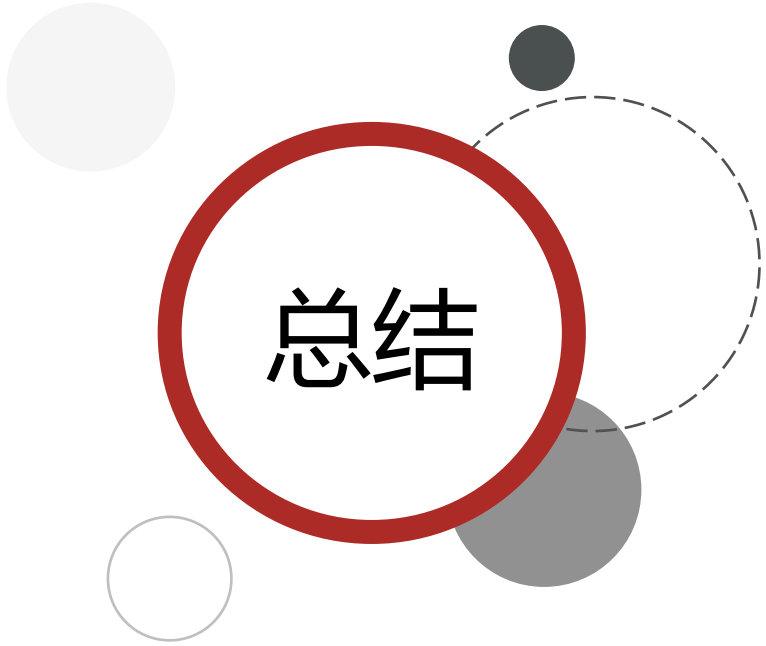
② 定义一个变量用于记录最大值，这个变量建议默认存储第一个元素值作为参照。

```
int max = faceScores[0];
```

③ 遍历数组的元素，如果该元素大于变量存储的元素，则替换变量存储的值为该元素。

```
for (int i = 1; i < faceScores.length; i++) {
    if(faceScores[i] > max) {
        // 替换
        max = faceScores[i];
    }
}
```

④ 循环结束后输出最大值变量即可。



1. 数组元素求最大值如何实现的？

① 数据拿到程序中去，用数组装起来。

② 定义一个变量用于记录最大值，这个变量建议默认存储第一个元素值作为参照。

③ 遍历数组的元素，如果该元素大于变量存储的元素，则替换变量存储的值为该元素。

④ 循环结束后输出最大值变量即可。

目录

Contents

案例

猜数字游戏

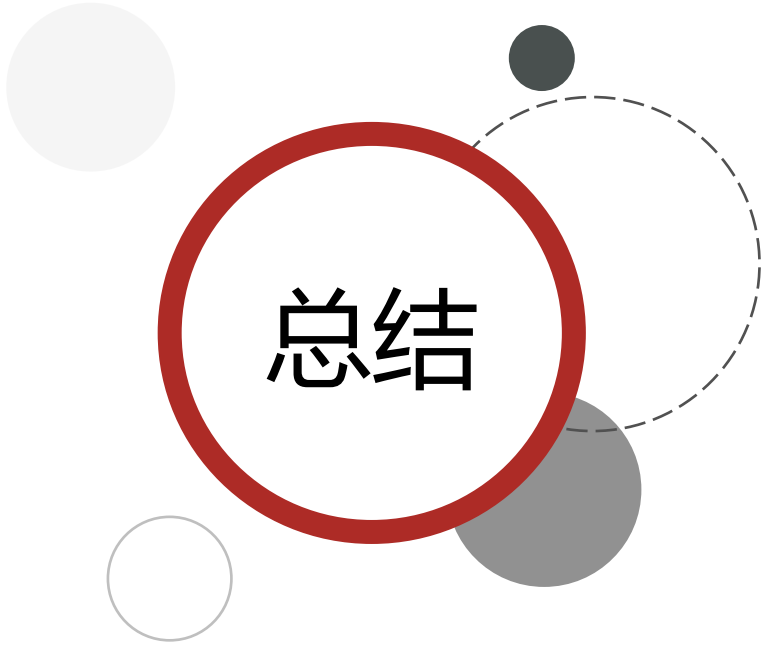
需求

- 开发一个幸运小游戏，游戏规则如下：
- 游戏后台随机生成1-20之间的5个数（无所谓是否重复），然后让大家来猜数字：
  - **未猜中提示：“未命中”，并继续猜测**
  - **猜中提示：“运气不错，猜中了”，并输出该数据第一次出现的索引位置，最后把数组中的5个数据都输出看以下，然后结束本游戏。**

分析

① 随机生成5个1-20之间的数据存储起来 ---> 使用数组

② 定义一个死循环，输入数据猜测，遍历数组，判断数据是否在数组中，如果在，进行对应提示并结束死循环；如果没有猜中，提示继续猜测直到猜中为止。



1. 猜数字游戏的实现步骤？

   ① 动态初始化数组，存入5个随机的1-20之间的数据。

   ② 定义一个死循环，不断的猜数据。

   ③ 遍历数组，判断数据是否在数组中，如果在，进行对应提示并结束死循环；如果没有猜中，提示继续。

目录

Contents

案例 随机排名

需求

某公司开发部5名开发人员，要进行项目进展汇报演讲，现在采取随机排名后进行汇报。

请先依次录入5名员工的工号，然后展示出一组随机的排名顺序。

分析

① 在程序中录入5名员工的工号存储起来 ---> 使用数组。

② 依次遍历数组中的每个元素，随机一个索引数据，让当前元素与该索引位置处的元素进行交换。

案例 元素交换的原理图

临时变量

22 33 35 13

88

随机交换排名的其他使用场景

总结

1. 如何实现随机排名的？

① **定义一个动态初始化的数组用于录入数据。**

② **遍历数组中的每个元素，每次随机一个索引值，让当前元素与该索引位置处的元素进行交换。**

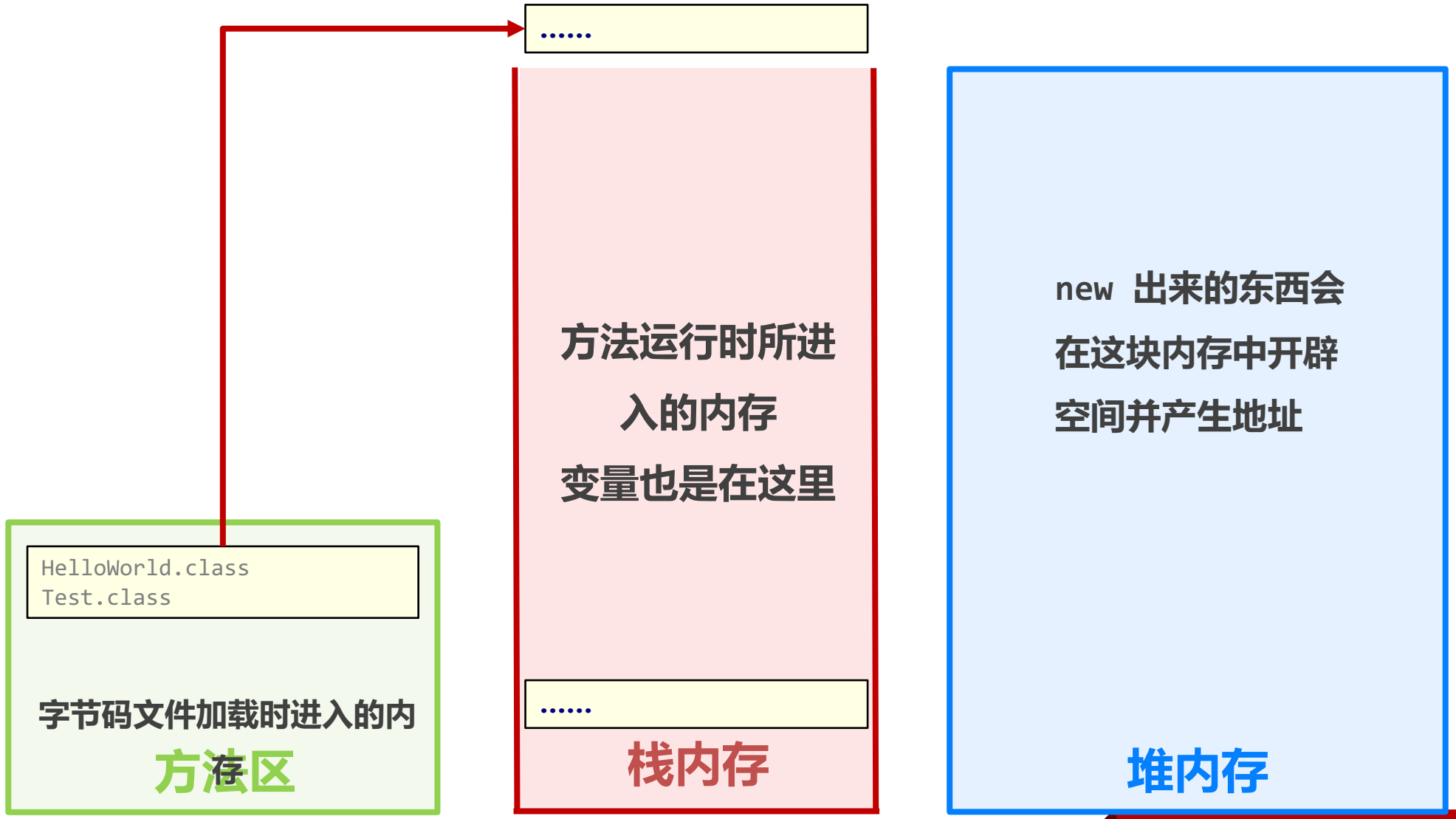
③ **遍历输出数组中的内容即可。**

目录

Contents

Java 内存分配介绍

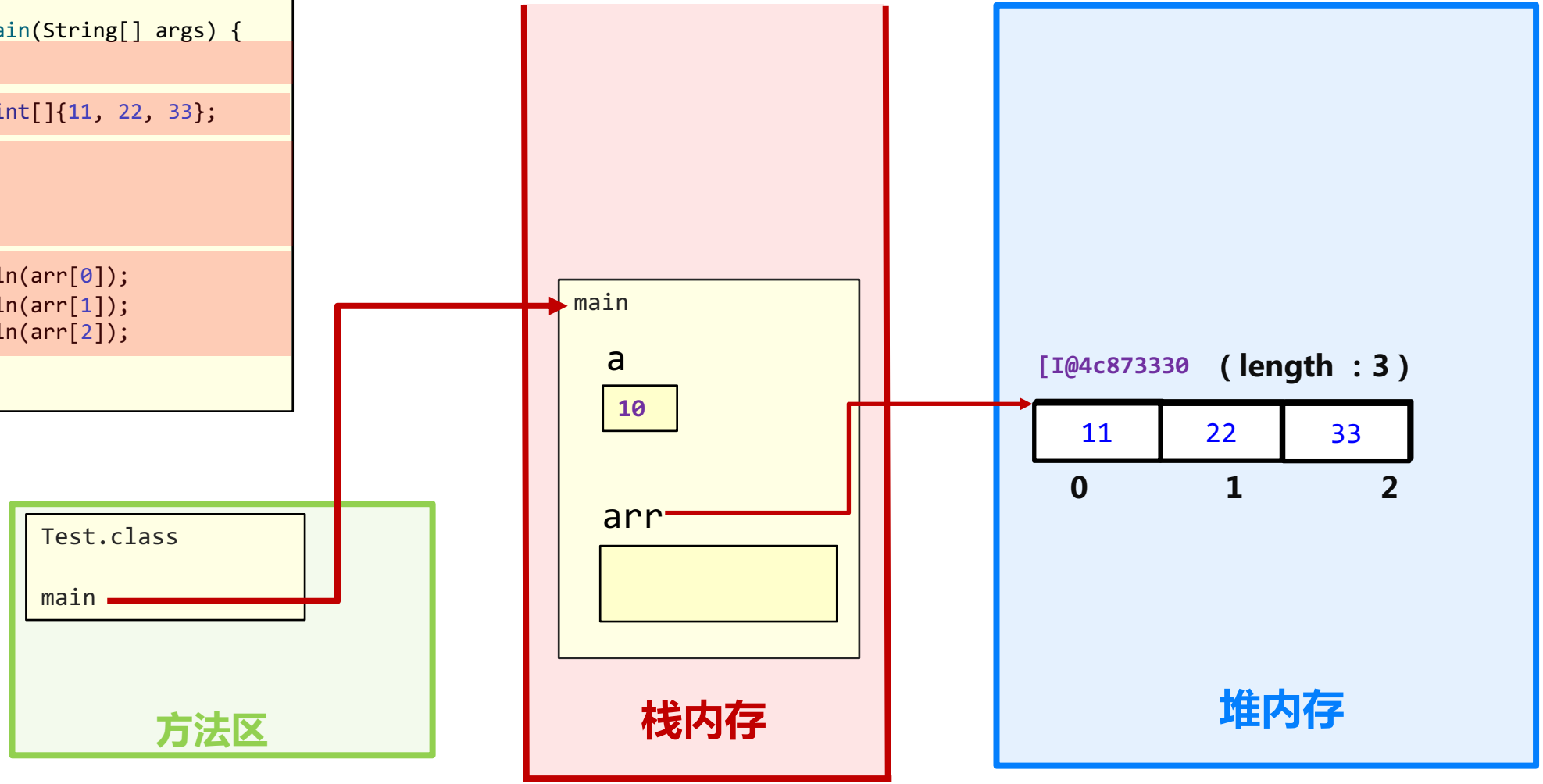
- 栈
- 堆
- 方法区
- 本地方法栈
- 寄存器



Java内存分配介绍

```
public class Test {
    public static void main(String[] args) {
        int a = 10;

        int[] arr = new int[]{11, 22, 33};

        arr[0] = 44;
        arr[1] = 55;
        arr[2] = 66;

        System.out.println(arr[0]);
        System.out.println(arr[1]);
        System.out.println(arr[2]);
    }
}
```

```
44
55
66
```



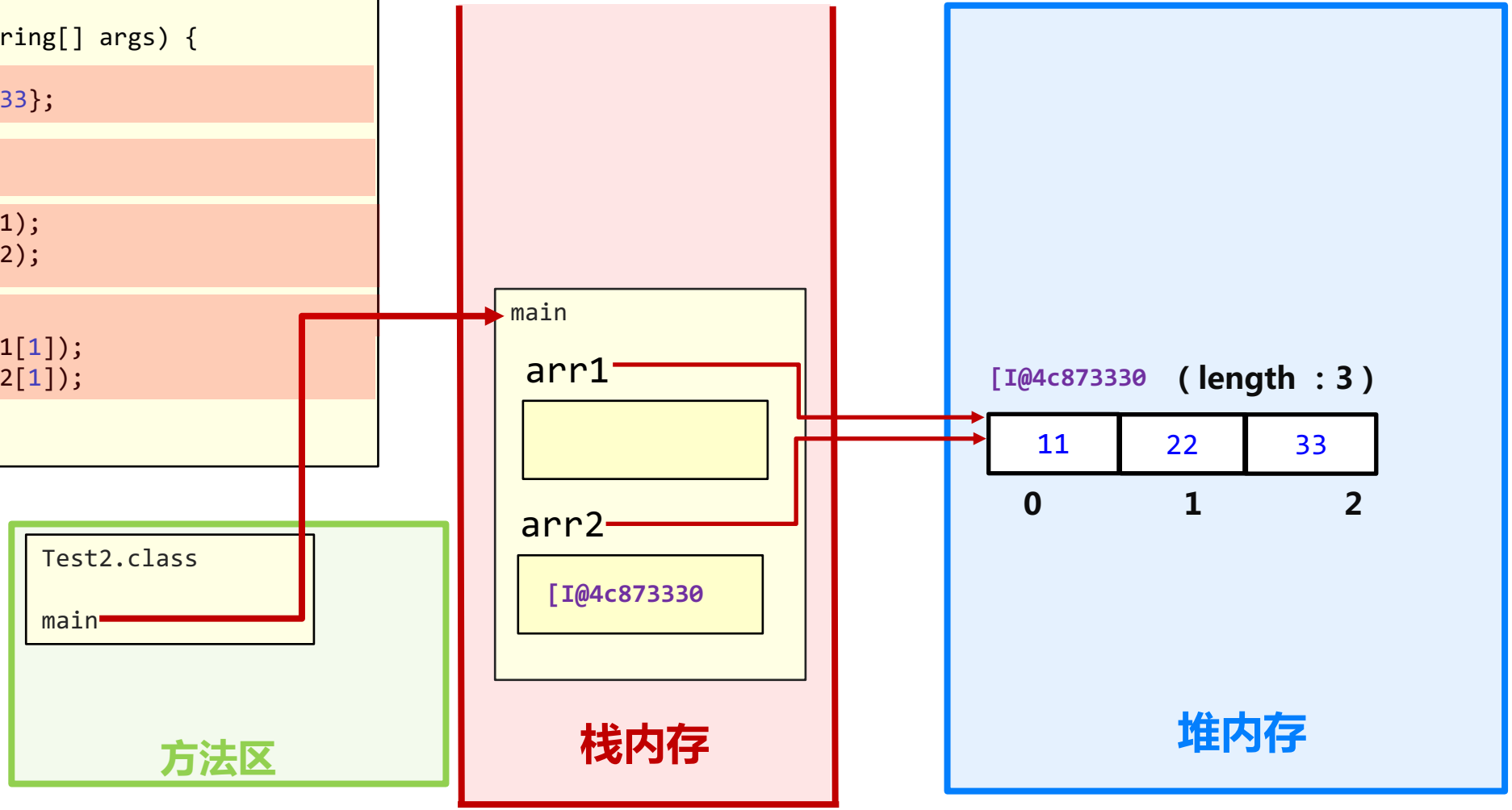
目录

Contents

两个数组变量指向同一个数组对象

```
public class Test2{
    public static void main(String[] args) {

        int[] arr1 = {11, 22, 33};

        int[] arr2 = arr1;

        System.out.println(arr1);
        System.out.println(arr2);

        arr2[1] = 99;
        System.out.println(arr1[1]);
        System.out.println(arr2[1]);
    }
}
```

```
[I@4c873330
[I@4c873330
99
99
```



目录

Contents

数组使用常见问题

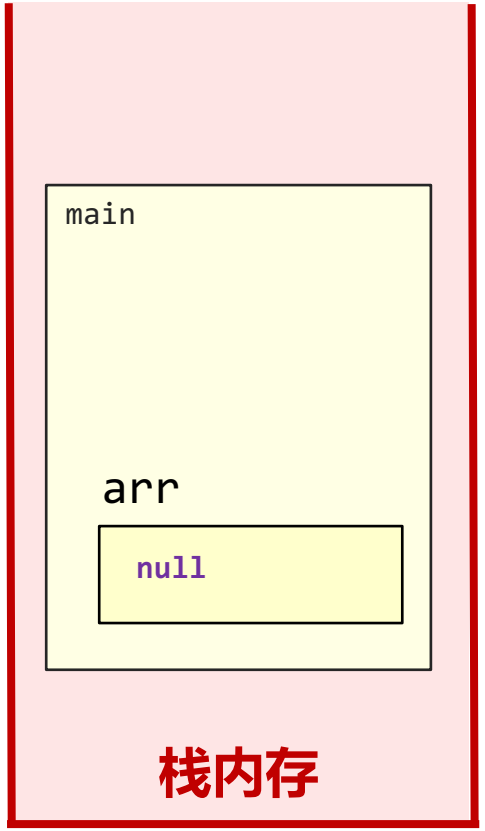
- 问题1：如果访问的元素位置超过最大索引，执行时会出现ArrayIndexOutOfBoundsException(数组索引越界异常)
- 问题2：如果数组变量中没有存储数组的地址，而是null, 在访问数组信息时会出现NullPointerException(空指针异常)

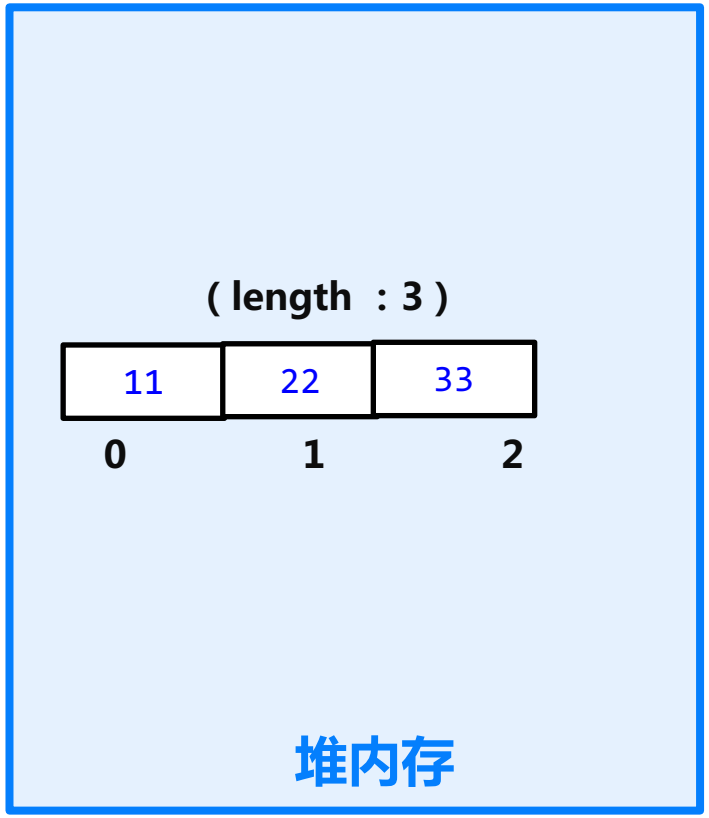
```
int[] arr = new int[]{11, 22, 33};

System.out.println(arr[2]);

// System.out.println(arr[3]) // 出现异常
```

出现ArrayIndexOutOfBoundsException异常

```
arr = null;
System.out.println(arr); // null
System.out.println(arr.length) // 出现异常
```

出现NullPointerException异常





目录

Contents

Debug工具

- IDEA自带的断点调试(排错)工具，可以控制代码从断点开始一行一行的执行，然后详细观看程序执行的情况

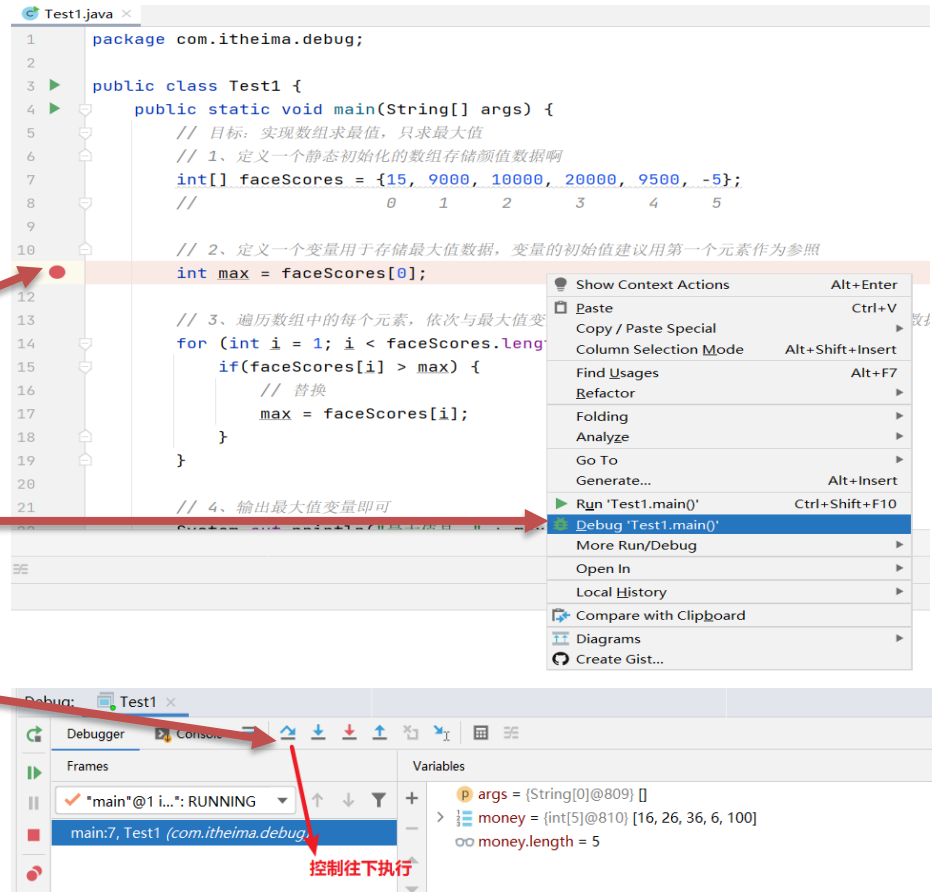
。

DEBUG工具基本使用步骤

① 在需要控制的代码行左侧，点击一下，形成断点

② 选择使用Debug方式启动程序，启动后程序会在断点暂停

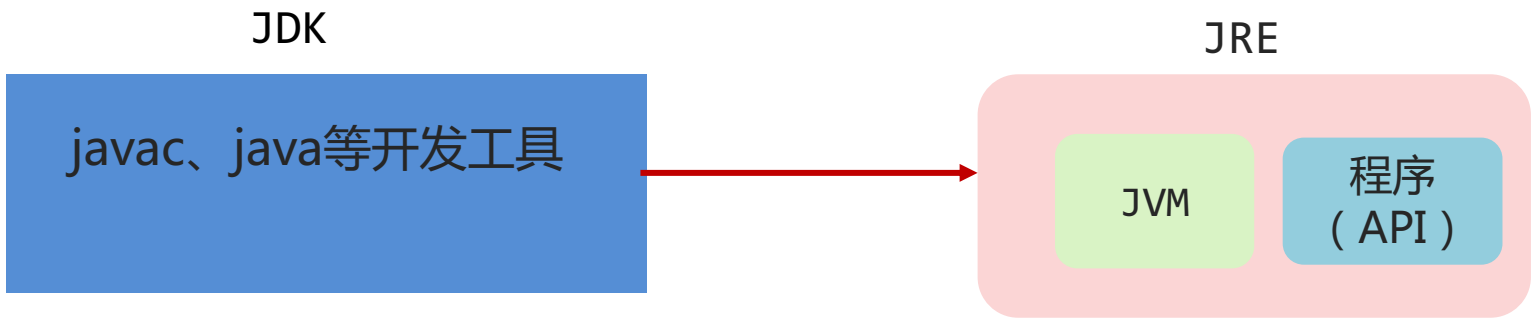
③ 控制代码一行一行的往下执行



- 类型转换
- 运算符
- 案例知识：键盘录入技术

需求：

- 请完成Java程序与用户交互，比如录入用户输入的名称、年龄，怎么办？。



API（Application Programming Interface,应用程序编程接口）

- Java写好的程序(功能代码)，咱们可以直接调用。
- Oracle 也为Java写好的程序提供了相应的 API文档(技术使用说明书) 。

下载API文档：

http://www.oracle.com/technetwork/java/javase/downloads/index.html

键盘录入功能实现的三个步骤：

①：导包：告诉程序去JDK的哪个包中找扫描器技术

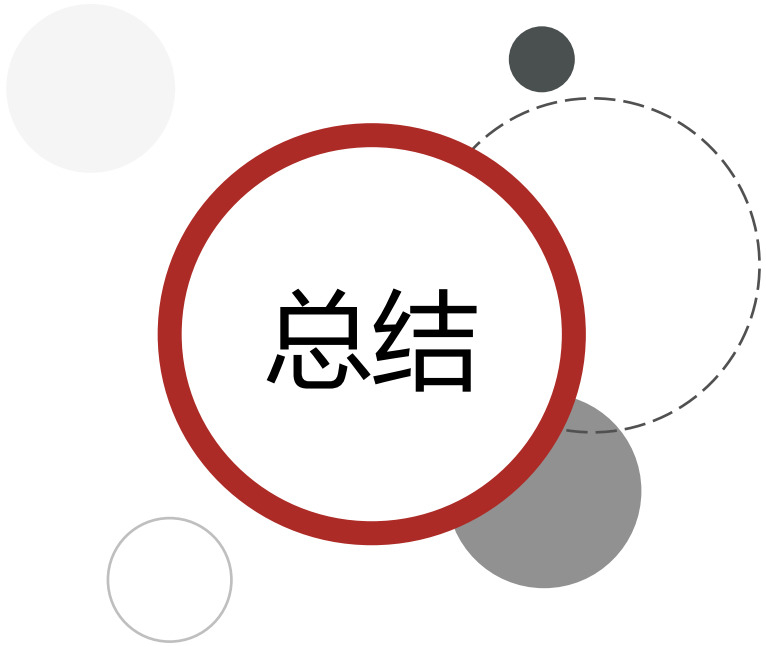
②：写一行代码代表得到键盘扫描器对象。

③：等待接收用户输入数据。

```
package com.itheima.scanner;
import java.util.Scanner;
public class Test {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.println("请输入您的年龄：");
        int age = sc.nextInt();
        System.out.println("年龄是：" + age);

        System.out.println("请输入您的名称：");
        String name = sc.next();
        System.out.println("欢迎：" + name);
    }
}
```

注意：

- **System、String在JDK中的Java.lang包下**
- **lang包不需要我们导包，是默认的包。**



1．API是什么？

- **Application Programming Interface,应用程序编程接口。**
- **Java写好的程序，咱们可以直接调用。**

2．键盘录入的开发步骤

- **导包：import java.util.Scanner;**
- **抄写代码得到扫描器对象: Scanner sc = new Scanner(System.in)**
- **抄写代码等待接收用户输入的数据: sc.nextInt() 、sc.next()**

传智教育旗下高端IT教育品牌