

第六章 数组

主要内容

1. 一维数组
2. 一维数组引用举例
3. 二维数组
4. 二维数组的引用

1. 一维数组

1.1 一维数组的声明

声明数组需要确定数组名、数组的维数和数组元素的数据类型。

类型标识符 数组名[]; 或 类型标识符[] 数组名;

例 【6-1】

```
int abc[];
```

```
double[] example2;
```


1.2 一维数组初始化

(1) 直接指定初值的方式

声明一个数组的同时将数组元素的初值依次写入赋值号后的一对花括号内，给这个数组的所有数组元素赋上初始值。

```
int[] a1 = {23, -9, 38};
```

Java编译器可通过初值的个数确定数组元素的个数，为它分配足够的存储空间并将这些值写入相应的存储单元。

1.2 一维数组初始化

(2) 用关键字 `new` 初始化数组

`new` 初始化数组，只为数组分配存储空间而不对数组元素赋初值。

`new` 初始化数组有两种方式：

- A. 先声明数组，再初始化数组
- B. 声明数组同时用 `new` 初始化数组

● 方式1:

类型标识符 数组名[];

数组名 = new 类型标识符[数组长度];

● 方式2:

类型标识符 数组名 = new 类型标识符[数组长度];

或

类型标识符[] 数组名 = new 类型标识符[数组长度];

1.3 一维数组的引用

- 当数组初始化后可通过数组名与下标来引用数组中的每一个元素。

数组名[数组下标];

【例6-2】

```
int[] a = new int[10];
```

```
a[3] = 30;
```

```
a[3+6] = 90;
```

```
a[10] = 8; 越界，错误引用
```


2 一维数组引用举例

1. 测定数组的长度

在Java语言中，数组也是一种对象。

数组经初始化后，就确定了它的长度，对于每个已分配存储空间的数组，Java用一个数据成员`length`来存储这个数组的长度值。

2. 数组间相互赋值

Java语言允许类型相同，长度不同的两个数组互相赋值。赋值的结果是两个数组名指向同一数组。

3. 向成员方法传递数组元素

数组元素作为成员方法的实参，是单向值传递，即只能由数组元素传递给形参，程序中对形参的任何修改并不改变数组元素的值。

4. 向成员方法传递数组名

数组名作为成员方法的实参时，把实参数组的起始地址传递给形参数组，即两个数组共用同一段内存单元，在成员方法中对形参数组中各元素值的修改，都会使实参数组元素的值也发生同样的变化。传递方式被称为“双向地址传递”。

5. 同类型、不同长度的数组引用同一个成员方法

在Java语言中，数组也是一种对象，只要数组元素的数据类型相同，而不论其长度是否一致，都可以引用同一个成员方法，完成所需要的操作。

6. 对象数组

实际问题中往往需要把不同类型的数据组合成一个有机的整体，以便于引用。

姓名	性别	年龄	数学	英语	计算机
Li	F	19	89	86	69
He	M	18	90	83	76
Zhang	M	20	78	91	80

如果一个类有若干个对象，可把这一系列同类的对象用一个数组来存放。

3 二维数组

$a[0][0]$	$a[0][1]$	$a[0][2]$	$a[0][3]$
$a[1][0]$	$a[1][1]$	$a[1][2]$	$a[1][3]$
$a[2][0]$	$a[2][1]$	$a[2][2]$	$a[2][3]$

数组名

列下标

行下标

3.1 二维数组的声明

- 二维数组的声明与一维数组类似，只需给出两对方括号。

类型说明符 数组名[] []

【例6-3】

```
int arr[] [];
```

```
int [] [] arr;
```


3.2 二维数组初始化

1. 用new操作符初始化数组

1) 先声明数组再初始化数组

```
int arr[][];
```

```
arr= new int[3][4];
```


2) 声明数组同时初始化数组

```
int[][] arr = new int[4][];
```

```
int arr[][] = new int[4][3];
```

不指定行数，而指定列数是错误的

```
int[][] arr = new int[][4]; 错误
```


2. 直接指定初值的方式

在数组声明时，对数据元素赋初值。使用指定的初值对数组初始化。

```
int[][] arr = {{3, -9, 6}, {8, 0, 1}};
```

用指定初值的方式对数组初始化时，各子数组元素的个数可以不同。

```
int[][] arr = {{3, -9}, {8, 0, 1}};
```


3.3 二维数组的引用

1. 测定数组的长度及数组赋值

与一维数组一样，也可以用`.length`成员方法测定二维数组的长度，即元素的个数。

使用“数组名`.length`”的形式测定数组的行数；

使用“数组名`[i].length`”的形式测定该行列数。

2. 数组名作为成员方法的参数

与一维数组类似，二维数组的数组名也可以作为参数传递给成员方法。