

二维数组

目 录

- 1. 二维数组的定义2
- 2. 二维数组及其使用3



1. 二维数组的定义

Java 中定义和操作多维数组的语法与一维数组类似。在实际应用中，三维及以上的数组很少使用，主要使用二维数组。下面主要了解一下二维数组。

二维数组定义语法如下：

数据类型 [][] 数组名；

或者：

数据类型 数组名 [][]；

在语法中：

- 数据类型为数组元素的类型
- “[][]”用于表明定义了一个二维数组，通过多个下标进行数据访问

例如：

```
int[][] scores;           //定义二维数组
scores=new int[5][50];    //分配内存空间
//或者
int[][] scores = new int[5][50];
```

需要强调的是，虽然从语法上Java支持多维数组，但从内存分配原理的角度讲，Java中只有一维数组，没有多维数组。或者说，表面上是多维数组，实质上都是一维数组。

下面这行代码定义一个整型二维数组，并为其分配内存空间。

```
int[][] s = new int[3][5];
```

表面看来是定义了一个二维数组，但是从内存分配原理角度讲，实际上是定义了一个一维数组，数组名是s，包括三个元素，分别为s[0]、s[1]、s[2]，每个元素是整型数组类型，即一维数组类型。而s[0]又是一个数组的名称，包括5个元素，分别为s[0][0]、s[0][1]、s[0][2]、s[0][3]、s[0][4]，每个元素都是整数类型。s[1]、s[2]与s[0]的情况相同，其存储方式如下图所示。

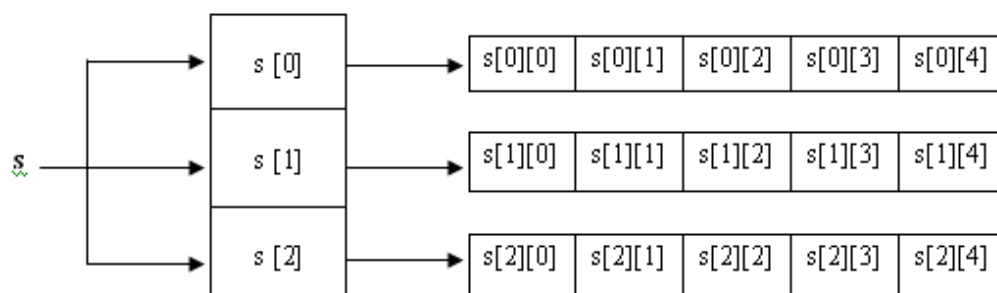


图4.13 二维数组存储方式示意图

二维数组实际上是一个一维数组，它的每个元素又是一个一维数组。

2. 二维数组及其使用

(1) 初始化二维数组

二维数组也可以进行初始化操作，与一维数组类似，同样采用如下两种方式，要注意大括号的结构及书写顺序。

下面是定义二维数组并初始化数组元素的两种方法。

```
int[][] scores=new int[][]{ { 90, 85, 92,78,54 }, { 76, 63,80 }, { 87 } };
//或者
int scores[][] = {{ 90, 85, 92,78,54 }, { 76, 63,80 }, { 87 } };
```

(2) 二维数组的遍历

示例 1

分别计算每个班级的学员总成绩。

实现步骤：

- (1) 初始化整型二维数组。
- (2) 定义保存总成绩的变量。
- (3) 使用for循环遍历二维数组。

关键代码：

```
public static void main(String[] args) {
    int [][] array = new int[][]{ {80,66},{70,54,98},{77,59} };
    int total;                                     //保存总成绩
    for (int i = 0; i < array.length; i++) {
        String str = (i+1) + "班";
        total = 0;
        for (int j = 0; j < array[i].length; j++) {
            total += array[i][j];                  //成绩叠加
        }
        System.out.println(str+"总成绩: " + total);
    }
}
```

```
}  
}
```

运行结果：

1 班总成绩：146

2 班总成绩：222

3 班总成绩：136

