



第三章 数组

讲授思路

- 数组概述
- 一维数组
- Arrays类
- 二维数组

数组的概念

- 数组是一组具有相同数据类型的数据元素的有序集合。
- 数组的特点
 - 在整个生命周期中长度固定不可变。
 - 数组可以存储基本数据类型和引用类型元素。
 - 同一个数组中必须存储相同类型的元素。
 - 数组中的元素有先后顺序，其顺序位置由数组下标决定。
- Java中数组作为数组类（Array）的实例来处理的。

一维数组的声明

- 数组声明的基本语法：
 - array_type array_Name[];
 - array_type[] array_Name;
- 举例：

```
int ids[];  
String[] stuNo;
```

一维数组的初始化

- 数组只是声明的话，JVM并没有给数组分配内存空间。在使用数组前还必须对其进行初始化(即为其分配内存空间)。给数组元素分配内存并为数组元素赋初值的过程称为数组初始化。
- 初始化可分为
 - 静态初始化
 - 动态初始化

静态初始化

- 当数组元素的初始化值直接由括在大括号 “{ }” 之间的数据给出时，就称为静态初始化。该方法适用于数组的元素不多且初始元素有限时。静态初始化往往和声明结合在一起使用，其格式如下：
 - `array_type array_Name={element1[, element2...]};`
- 例如：

```
int ids[]={ 1,2,3,4,5,6,7,8};  
String[] stuNo={"2011010345", "2011010346", "2011010347"};  
double scores[]={62,45.5,88.1,76.4};
```

静态初始化

```
int num1[] = {1,2,3};  
int num2[3] = {1,2,3};    // 编译错误，不能在[]中指定数字长度  
int num3[];  
int k;  
System.out.println(num1.length);  
System.out.println(num3.length); // 编译错误，未初始化不能使用  
System.out.println(k);    // 编译错误，未初始化不能使用
```

动态初始化

- 动态初始化先用new操作符为数组分配内存，然后才为每一个元素赋初值。其一般格式如下：

- array_Name = new array_type [arraySize];

- 例如：

```
int series[ ]= new int[4];  
for (int i=0;i<4;i++){  
    series[ i ]=i*3;  
}
```


一维数组的使用

- 有了数组的声明和初始化后，就可以在程序中引用数组的元素了。
- 数组元素的引用是通过数组名和下标值来进行的，其一般格式如下：
 - array_Name [arrayIndex]
- 例如：

```
int sale[]={1,2,3,4,5};  
int i = sale[1];  
System.out.println("您选择的是" + i);
```

二维数组

- Java中多维数组可以看成是数组的数组，即多维数组中的每一个元素可以看成是低维数组，所以多维数组的声明、初始化、引用与一维数组非常类似。
- 在多维数组中二维数组应用最为广泛，所以我们着重讨论二维数组的属性和特性，多维以此类推即可。

二维数组的声明

- 二维数组的声明的一般格式：
 - `array_type array_Name[][];`
 - `array_type [][] array_Name;`
 - `array_type [] array_Name[];`
- 例如：

```
int[ ][ ] arrays;  
int[ ] arrays[ ];  
int arrays[ ][ ];
```

二维数组的初始化

- 二维数组的初始化与一维数组类似，也分为静态初始化和动态初始化
- 静态初始化
 - `int number[][] = { {1,2},{3,4},{5,6}};`
 - `char ch[][] = { {'a', 'b'},{'c', 'd'},{'e', 'f'}};`
- 动态初始化
 - `long[][] array_long = new long[5][5];`
 - `String[][] names = new String[3][4];`

Arrays类的引入

- 数组是一种非常常用的存储数据的数据结构，并且在程序中会大量对数组中的元素进行查找、排序等操作，使用循环，条件分支等流程控制可以满足需求，但程序会显得比较凌乱。
- Java中提供了实用的工具类Arrays，辅助编程人员简化数组的常用操作。

Arrays

- `java.util.Arrays`类能方便地操作数组，它提供的所有方法都是静态的
- 常用方法：
 - `copyOf`实现数组的复制
 - `Fill`实现数组元素的初始化
 - `Sort`实现数组的排序

总结

- 一维数组
- 二维数组
- Arrays工具类



Thank You