

## Java基础之数组

原创 2017年07月29日 13:05:53

33 0 1

在生活中，我们可能会碰到如下的场景。

现在需要统计某公司员工的工资情况，例如计算平均工资、最高工资等。假设该公司有50名员工，用前面所学的知识完成，那么程序首先需要声明50个变量来分别记住每位员工的工资，这样做会显得很麻烦。其实在Java中，我们可以使用一个数组来记住这50名员工的工资。数组是指一组数据的集合，数组中的每个数据被称作元素。在数组中可以存放任意类型的元素，但同一个数组里存放的元素类型必须一致。

### 1.1 数组的定义

在Java中，可以使用以下格式来定义一个数组。如下

数据类型[] 数组名 = new 数据类型[元素个数或数组长度];

int[] x = new int[100]; 喜欢

上述语句就相当于在内存中定义了100个int类型的变量，第一个变量的名称为x[0]，第二个变量的名称为x[1]，以此类推，第100个变量的名称为x[99]，这些变量的初始值都是0。为了更好地理解数组的这种定义方式，可以将上面的一句代码分成两行来写，具体如下：

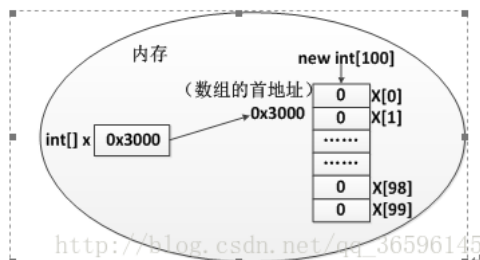
```
int[] x;           // 声明一个int[]类型的变量
x = new int[100];  // 创建一个长度为100的数组
```

接下来，通过两张内存图来详细地说明数组在创建过程中内存的分配情况。

第一行代码 int[] x; 声明了一个变量x，该变量的类型为int[]，即一个int类型的数组。变量x会占用一块内存单元，它没有被分配初始值。内存中的状态如下图所示。



第二行代码 x = new int[100]; 创建了一个数组，将数组的地址赋值给变量x。在程序运行期间可以使用变量x来引用数组，这时内存中的状态会发生变化，如下图所示。



在上图中描述了变量x引用数组的情况。该数组中有100个元素，初始值都为0。数组中的每个元素都有一个索引(也可称为角标)，要想访问数组中的元素可以通过“x[0]、x[1]、.....、x[98]、x[99]”的形式。需要注意的是，数组中最小的索引是0，最大的索引是“数组的长度-1”。在Java中，为了方便我们获得数组的



克己的博客 ([http://blog.csdn.net/qq\\_36596145](http://blog.csdn.net/qq_36596145))

+ 关注

([http://blog.csdn.net/qq\\_36596145](http://blog.csdn.net/qq_36596145))

原创 粉丝 喜欢

14 5 0

- > idea报错：BeanFactory not initialized or already closed - call 'refresh' before accessing beans via ...  
([/qq\\_36596145/article/details/76728787](http://blog.csdn.net/qq_36596145/article/details/76728787))
  - > Java基础之引用数据类型（类）  
([/qq\\_36596145/article/details/76310135](http://blog.csdn.net/qq_36596145/article/details/76310135))
  - > Java基础之方法  
([/qq\\_36596145/article/details/76308385](http://blog.csdn.net/qq_36596145/article/details/76308385))
  - > Java基础之数组  
([/qq\\_36596145/article/details/76305576](http://blog.csdn.net/qq_36596145/article/details/76305576))
- 更多文章 ([http://blog.csdn.net/qq\\_36596145](http://blog.csdn.net/qq_36596145))

在线课程



([http://edu.csdn.net/huiyiCourse/series\\_detail?utm\\_source=blog7](http://edu.csdn.net/huiyiCourse/series_detail?utm_source=blog7))

【直播】机器学习&数据挖掘7周实训--韦玮

([http://edu.csdn.net/huiyiCourse/series\\_detail/54?utm\\_source=blog7](http://edu.csdn.net/huiyiCourse/series_detail/54?utm_source=blog7))



([http://edu.csdn.net/combo/detail/471?utm\\_source=blog7](http://edu.csdn.net/combo/detail/471?utm_source=blog7))

【套餐】系统集成项目管理工程师顺利通关--徐朋

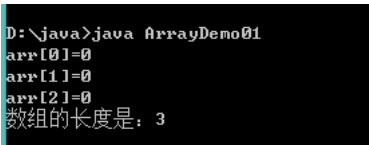
([http://edu.csdn.net/combo/detail/471?utm\\_source=blog7](http://edu.csdn.net/combo/detail/471?utm_source=blog7))

长度，提供了一个length属性，在程序中可以通过“数组名.length”的方式来获得数组的长度，即元素的个数。

接下来，通过一个案例来演示如何定义数组以及访问数组中的元素，如下所示。ArrayDemo01.java

```
1 public class ArrayDemo01 {
2     public static void main(String[] args) {
3         int[] arr; // 声明变量
4         arr = new int[3]; // 创建数组对象
5         System.out.println("arr[0]=" + arr[0]); // 访问数组中的第一个元素
6         System.out.println("arr[1]=" + arr[1]); // 访问数组中的第二个元素
7         System.out.println("arr[2]=" + arr[2]); // 访问数组中的第三个元素
8         System.out.println("数组的长度是: " + arr.length); // 打印数组长度
9     }
10 }
```

运行结果如下图所示。



在上述代码中声明了一个int[]类型变量arr，并将数组在内存中的地址赋值给它。在5~7行代码中通过角标来访问数组中的元素，在第8行代码中通过length属性访问数组中元素的个数。从打印结果可以看出，数组中的三个元素初始值都为0，这是因为当数组被成功创建后，数组中元素会被自动赋予一个默认值，根据元素类型的不同，默认初始化的值也是不一样的。具体如下表所示。

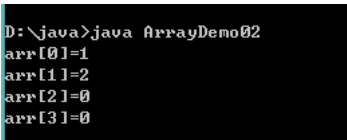
表2-1 元素默认值

数据类型	默认初始化值
byte、short、int、long	0
float、double	0.0
char	一个空字符（空格），即'\u0000'
boolean	false
引用数据类型	null，表示变量不引用任何对象

如果在使用数组时，不想使用这些默认初始值，也可以显式地为这些元素赋值。接下来通过一个程序来学习如何为数组的元素赋值，如下所示。ArrayDemo02.java

```
1 public class ArrayDemo02 {
2     public static void main(String[] args) {
3         int[] arr = new int[4]; // 定义可以存储4个整数的数组
4         arr[0] = 1; // 为第1个元素赋值1
5         arr[1] = 2; // 为第2个元素赋值2
6         // 下面的代码是打印数组中每个元素的值
7         System.out.println("arr[0]=" + arr[0]);
8         System.out.println("arr[1]=" + arr[1]);
9         System.out.println("arr[2]=" + arr[2]);
10        System.out.println("arr[3]=" + arr[3]);
11    }
12 }
```

运行结果如下图所示。



在上述代码中，第3行代码定义了一个数组，此时数组中每个元素都为默认初始值0。第2、3行代码通过赋值语句将数组中的元素arr[0]和arr[1]分别赋值为1和2，而元素arr[2]和arr[3]没有赋值，其值仍为0，因此打印结果中四个元素的值依次为1、2、0、0。

在定义数组时只指定数组的长度，由系统自动为元素赋初值的方式称作动态初始化。  
在初始化数组时还有一种方式叫做静态初始化，就是在定义数组的同时就为数组的每个元素赋值。数组的静态初始化有两种方式，具体格式如下：

- 1、类型[] 数组名 = new 类型[] {元素, 元素, .....};
- 2、类型[] 数组名 = {元素, 元素, 元素, .....};

上面的两种方式都可以实现数组的静态初始化，但是为了简便，建议采用第二种方式。接下来通过一段代码来演示数组静态初始化的效果，如下所示。ArrayDemo03.java

```
1 public class ArrayDemo03 {
2     public static void main(String[] args) {
3         int[] arr = { 1, 2, 3, 4 }; // 静态初始化
4         // 下面的代码是依次访问数组中的元素
5         System.out.println("arr[0] = " + arr[0]);
6         System.out.println("arr[1] = " + arr[1]);
7         System.out.println("arr[2] = " + arr[2]);
8         System.out.println("arr[3] = " + arr[3]);
9     }
10 }
```

运行结果如下图所示。

```
D:\java>java ArrayDemo03
arr[0] = 1
arr[1] = 2
arr[2] = 3
arr[3] = 4
```

上述代码中采用静态初始化的方式为数组每个元素赋予初值，分别是1、2、3、4。需要注意的是，第3行代码千万不可写成`int[] arr = new int[4]{1,2,3,4};`，这样写编译器会报错。原因在于编译器会认为数组限定的元素个数[4]与实际存储的元素{1,2,3,4}个数有可能不一致，存在一定的安全隐患。

## 1.2 数组遍历

在操作数组时，经常需要依次访问数组中的每个元素，这种操作称作数组的遍历。接下来通过一个案例来学习如何使用for循环来遍历数组，如下所示。ArrayDemo04.java

```
public class ArrayDemo04 {
    public static void main(String[] args) {
        int[] arr = { 1, 2, 3, 4, 5 }; // 定义数组
        // 使用for循环遍历数组的元素
        for (int i = 0; i < arr.length; i++) {
            System.out.println(arr[i]); // 通过索引访问元素
        }
    }
}
```

运行结果如下图所示。

```
D:\java>java ArrayDemo04
1
2
3
4
5
```

上述代码中，定义一个长度为5的数组arr，数组的角标为0~4。由于for循环中定义的变量i的值在循环过程中为0~4，因此可以作为索引，依次去访问数组中的元素，并将元素的值打印出来。

## 1.3 数组的常见问题

数组在编写程序时应用非常广泛，灵活地使用数组对实际开发很重要。接下来，本节将针对数组的常见操作进行详细地讲解，如数组的遍历、最值的获取、数组的排序等。

### 1.3.1 数组最值

在操作数组时，经常需要获取数组中元素的最值。接下来通过一个案例来演示如何获取数组中元素的最大值，如下所示。ArrayDemo05.java

```

public class ArrayDemo05 {
    public static void main(String[] args) {
        int[] arr = { 4, 1, 6, 3, 9, 8 }; // 定义一个数组
        int max = arr[0]; // 定义变量max用于记住最大数，首先假设第一个元素为最大值
        // 下面通过一个for循环遍历数组中的元素
        for (int x = 1; x < arr.length; x++) {
            if (arr[x] > max) { // 比较 arr[x]的值是否大于max
                max = arr[x]; // 条件成立，将arr[x]的值赋给max
            }
        }
        System.out.println("max=" + max); // 打印最大值
    }
}

```

运行结果如下图所示。

```

D:\java>java ArrayDemo05
max=9

```

上述代码中，定义了一个临时变量max，用于记住数组的最大值。通过for 循环获取数组中的最大值，赋值给max变量。

首先假设数组中第一个元素arr[0]为最大值，然后使用for循环对数组进行遍历，在遍历的过程中只要遇到比max值还大的元素，就将该元素赋值给max。这样一来，变量max就能够在循环结束时记住数组中的最大值。需要注意的是，在for循环中的变量i是从1开始的，这样写的原因是程序已经假设第一个元素为最大值，for循环中只需要从第二个元素开始比较，从而提高程序的运行效率。

### 1.3.2 数组异常

#### 1.3.2.1 数组越界异常

每个数组的索引都有一个范围，即0~length-1。在访问数组的元素时，索引不能超出这个范围，否则程序会报错，如下所示。ArrayDemo06.java

```

1 public class ArrayDemo06 {
2     public static void main(String[] args) {
3         int[] arr = new int[4]; // 定义一个长度为4的数组
4         System.out.println("arr[0]=" + arr[4]); // 通过角标4访问数组元素
5     }
6 }

```

运行结果如下图所示。

```

D:\java>java ArrayDemo06
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: 4
    at ArrayDemo06.main(ArrayDemo06.java:4)

```

上图运行结果中所提示的错误信息是数组越界异常ArrayIndexOutOfBoundsException，出现这个异常的原因是数组的长度为4，其索引范围为0~3，而上述代码中的第4行代码使用索引4来访问元素时超出了数组的索引范围。

所谓异常指程序中出现的错误，它会报告出错的异常类型、出错的行号以及出错的原因，关于异常在后面的章节会有详细地讲解。

#### 1.3.2.2 空指针异常

在使用变量引用一个数组时，变量必须指向一个有效的数组对象，如果该变量的值为null，则意味着没有指向任何数组，此时通过该变量访问数组的元素会出现空指针异常，接下来通过一个案例来演示这种异常，如下所示。ArrayDemo07.java

```

1 public class ArrayDemo07 {
2     public static void main(String[] args) {
3         int[] arr = new int[3]; // 定义一个长度为3的数组
4         arr[0] = 5; // 为数组的第一个元素赋值
5         System.out.println("arr[0]=" + arr[0]); // 访问数组的元素
6         arr = null; // 将变量arr置为null
7         System.out.println("arr[0]=" + arr[0]); // 访问数组的元素
8     }
9 }

```

运行结果如下图所示。

```
D:\java>java ArrayDemo07
arr[0]=5
Exception in thread "main" java.lang.NullPointerException
    at ArrayDemo07.main(ArrayDemo07.java:7)
```

通过上图所示的运行结果可以看出，上述代码中第4、5行代码都能通过变量arr正常地操作数组。第6行代码将变量置为null，当第7行代码再次访问数组时就出现了空指针异常NullPointerException。

### 1.4 二维数组

在程序中可以通过一个数组来保存某个班级学生的考试成绩，试想一下，如果要统计一个学校各个班级学生的考试成绩，又该如何实现呢？这时就需要用到多维数组，多维数组可以简单地理解为在数组中嵌套数组。在程序中比较常见的就是二维数组，接下来针对二维数组进行详细地讲解。

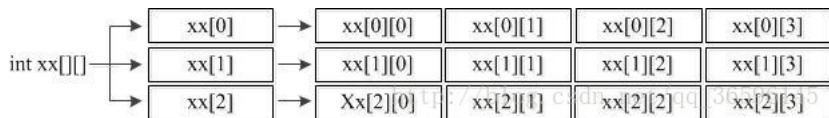
#### 1.4.1 二维数组的定义格式

二维数组的定义有很多方式，接下来针对几种常见的方式进行详细地讲解，具体如下：

第一种方式：

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

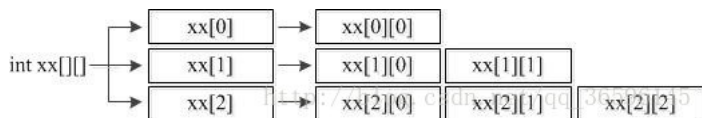
上面的代码相当于定义了一个3\*4的二维数组，即二维数组的长度为3，二维数组中的每个元素又是一个长度为4的数组，接下来通过一个图来表示这种情况，如下图所示。



第二种方式：

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

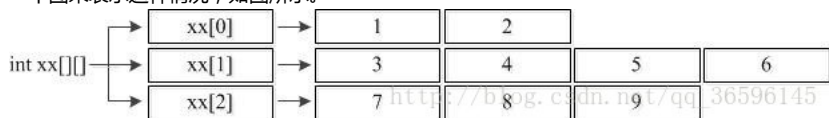
第二种方式和第一种类似，只是数组中每个元素的长度不确定，接下来通过一个图来表示这种情况，如下图所示。



第三种方式：

```
int[][] arr = {{1, 2}, {3, 4, 5, 6}, {7, 8, 9}};
```

上面的二维数组中定义了三个元素，这三个元素都是数组，分别为{1,2}、{3,4,5,6}、{7,8,9}，接下来通过一个图来表示这种情况，如图所示。



对二维数组中元素的访问也是通过角标的方式，如需访问二维数组中第一个元素数组的第二个元素，具体代码如下：

```
arr[0][1];
```

#### 1.4.2 二维数组元素的访问

操作二维数组时，经常需要获取数组中元素的值。接下来通过一个案例来演示如何获取数组中元素值，如下所示。ArrayDemo08.java

```

class ArrayDemo08 {
    public static void main(String[] args){

        //定义二维数组的方式
        int[][] arr = new int[3][4];

        System.out.println( arr );
        System.out.println("二维数组的长度: " + arr.length);
        //获取二维数组的3个元素
        System.out.println( arr[0] );
        System.out.println( arr[1] );
        System.out.println( arr[2] );

        System.out.println("打印第一个一维数组的元素值");
        System.out.println( arr[0][0] );
        System.out.println( arr[0][1] );//访问的为二维数组中第1个一维数组的第2个元素
        System.out.println( arr[0][2] );
        System.out.println( arr[0][3] );

        System.out.println("打印第二个一维数组的元素值");
        System.out.println( arr[1][0] );
        System.out.println( arr[1][1] );
        System.out.println( arr[1][2] );
        System.out.println( arr[1][3] );

        System.out.println("打印第三个一维数组的元素值");
        System.out.println( arr[2][0] );
        System.out.println( arr[2][1] );
        System.out.println( arr[2][2] );
        System.out.println( arr[2][3] );

    }
}

```

运行结果如下图所示:

![这里写图片描述] (<http://img.blog.csdn.net/20170729130324366?watermark/2/text/aHR0cDovL2Jsb2cuY3Nkt>)

### 1.4.3 二维数组元素遍历与数组元素累加和

学习完了数组元素的访问，我们来学习下数组的遍历及数组的元素累加和操作。

```

class ArrayDemo09 {
    public static void main(String[] args){
        //一维数组的求累加和并遍历
        int[] arr = {10, 20, 30, 40, 50};
        int sum = 0;
        for (int i=0; i<arr.length; i++) {
            //System.out.println(arr[i]);
            sum += arr[i];
        }
        System.out.println("sum= " + sum);
        System.out.println("-----");

        //二维数组的求累加和并遍历
        int[][] arr2 = { {1, 2}, {3, 4, 5}, {6, 7, 8, 9, 10} };
        int sum2 = 0;
        for (int i=0; i<arr2.length; i++) {
            for (int j=0; j<arr2[i].length; j++) {
                //System.out.println(arr2[i][j])
                sum2 += arr2[i][j];
            }
        }
        System.out.println("sum2= " + sum2);

    }
}

```

运行结果如下图所示:

```
D:\java>java ArrayDemo09
sun= 150
-----
sun2= 55
```

#### 1.4.4 公司销售总和练习

接下来通过一个案例来熟悉二维数组的使用。

例如要统计一个公司三个销售小组中每个小组的总销售额以及整个公司的销售额。如下所示

- 第一小组销售额为{11, 12}万元
- 第二小组销售额为{21, 22, 23}万元
- 第三小组销售额为{31, 32, 33, 34}万元。

```
ArrayDemo08.java
1 public class ArrayDemo10 {
2     public static void main(String[] args) {
3         int[][] arr = new int[3][]; // 定义一个长度为3的二维数组
4         arr[0] = new int[] { 11, 12 }; // 为数组的元素赋值
5         arr[1] = new int[] { 21, 22, 23 };
6         arr[2] = new int[] { 31, 32, 33, 34 };
7
8         int sum = 0; // 定义变量记录总销售额
9         for (int i = 0; i < arr.length; i++) { // 遍历数组元素
10             int groupSum = 0; // 定义变量记录小组销售总额
11             for (int j = 0; j < arr[i].length; j++) { // 遍历小组内每个人的销售额
12                 groupSum = groupSum + arr[i][j];
13             }
14             sum = sum + groupSum; // 累加小组销售额
15             System.out.println("第" + (i + 1) + "小组销售额为: " + groupSum + " 万元");
16         }
17         System.out.println("总销售额为: " + sum + " 万元");
18     }
19 }
```

运行结果如下图所示。

```
D:\java>java ArrayDemo08
第1小组销售额为: 23 万元。
第2小组销售额为: 66 万元。
第3小组销售额为: 130 万元。
总销售额为: 219 万元。
```

版权声明：本文为博主原创文章，欢迎转载,但请注明出处



标签：java (<http://so.csdn.net/so/search/s.do?q=java&t=blog>) /

数据 (<http://so.csdn.net/so/search/s.do?q=数据&t=blog>) /

数组 (<http://so.csdn.net/so/search/s.do?q=数组&t=blog>) /

本文已收录于以下专栏：

0 条评论



qq\_36596145 ([http://my.csdn.net/qq\\_36596145](http://my.csdn.net/qq_36596145))

([http://my.csdn.net/qq\\_36596145](http://my.csdn.net/qq_36596145))



发表评论

暂无评论

---

## 相关文章推荐

### Java基础（函数、数组） ([/haoyunpeng123/article/details/50750132](http://haoyunpeng123/article/details/50750132))

一、函数 函数的定义 什么是函数？函数就是定义在类中的具有特定功能的一段独立小程序，也可以称之为方法  
函数的格式 [java] view p...



haoyunpeng123 2016-02-26 15:27 173

---

### 黑马学习日记\_Java基础篇（基本数据类型、数组、函数） (<http://itheima.iteye.com/blog/2040298>)

<span style="line-height: 21px; background-color: #ffffff; font-family: 'Microsoft Yahei', 微软雅黑,



小驴变黑马 2014-04-03 03:03 225

---

### 黑马程序员--Java学习日记之基础知识(循环结构与数组) ([/memor\\_/article/details/46582793](http://memor_/article/details/46582793))

Java语言基础之运算符与流程控制语句 03.01\_Java语言基础(逻辑运算符的基本用法) A.逻辑运算符有哪些  
&(并且), |(或者), !(非), ^(异或...



Memor\_ 2015-06-21 18:50 479

---

### Java编程那些事儿44—数组基础语法 (<http://java2000-net.iteye.com/blog/241704>)

Java编程那些事儿44—数组基础语法 郑州游戏学院 陈跃峰 出自：<http://blog.csdn.net/mailbomb> 6.2 数组基本语法  
了解了数组的概念以后，下面来看一下数组的语法格式



java2000.net 2008-06-22 12:10 346

---

### Java基础复习笔记 数组，内存形式，父子，内存控制01 ([/hashsound\\_wang/article/details/8111456](http://hashsound_wang/article/details/8111456))

1. 前言 写这基础复习系列是觉得工作中自己的基础太差了，很多东西都没想透彻，没研究明白。看了《Java基础16课》总结出其中的一些知识点，用于以后自己复习用，以前的一些知识盲点也明白了...



hashsound\_Wang 2012-10-25 15:54 614

---

### （基础）Java中存储数据之数组,排序,查询,进制等 (<http://schy-hqh.iteye.com/blog/1948447>)

其实编写程序就是把数据拿在手中玩儿，对不？Java用来干嘛？<p style="padding-left: 30px



schy\_hqh 2013-09-27 22:44 518

---

### 黑马程序员 - (5) Java基础语法之数组 ([/wp1198571320/article/details/8877545](http://wp1198571320/article/details/8877545))

----- android培训、java培训、期待与您交流！ ----- 数组的定义：概念：同一种类型数据的集合，就是一个容器。数组优点：可以自动从0开始...



wp1198571320 2013-05-02 20:38 424

---

### JAVA基础——数组 (<http://953514429.iteye.com/blog/2097444>)

JAVA基础——数组 <p class="p0" style="





殇瓶-MIN 2014-07-28 19:32 101

## Java之美[从菜鸟到高手演变]之数据结构基础、线性表、栈和队列、数组和字符串 (/lililiii/article/details/40657483)

原文链接：<http://blog.csdn.net/zhangerqing/article/details/8796518> Java面试宝典之数据结构基础——线性表篇 作者...



lililiii 2014-10-31 16:58 523

## java笔记：关于复杂数据存储的问题--基础篇：数组以及浅拷贝与深拷贝的问题 (http://201111181922.iteye.com/blog/1335921)

记得我在写javascript笔记时候说过：程序就是由数据和运算组成。所以对数据存储以及读取方式的研究是熟练掌握语言精髓的重要途径。我在上篇文章里说道我想重新回顾一些知识，这些知识就是数据存储的问题，而且是复杂数据存储的问题。我个人认为一名优秀的程序员应该有四个主要指标：一是项目经验，二是程序优化的能力，三是良好的设计理念，四是快速准确定位程序bug的能力。项目经验不说，这个需要积累，而其他的能力都是可以通过学习而不断强化的。而语言中数据存储能力掌握的优劣是你优化程序的水平的高低的重要指标，你想让自己的程序越来越快，按什么数据模型快速存储数据，并且能很快的检索被存储的数据才是程序优化的本质，因



java软件乐园 2012-01-05 13:37 1014

## java基础——数组 (/funneies/article/details/9310869)

数组(Array)是Java 语言中内置的一种基本数据存储结构，通俗的理解，就是一组数的集合，目的是用来一次存储多个数据。数组是程序中实现很多算法的基础，可以在一定程度上简化代码的书写。Java 语言...



funneies 2013-07-12 14:10 1572

## Java编程那些事儿48——多维数组基础 (http://java2000-net.iteye.com/blog/241700)

Java编程那些事儿48——多维数组基础 郑州游戏学院 陈跃峰 出自：<http://blog.csdn.net/mailbomb> 6.4 多维数组基础 在学校里，由于一个班的人数不多，所以按照顺序编号



java2000.net 2008-06-27 14:58 576

## 黑马程序员——JAVA基础----语法（三）-- 方法和数组 (/du\_jing/article/details/49665503)

-----android培训、java培训、java学习型技术博客、期待与您交流！----- 语法（三）01\_Java语言基础(方法概述和格式说明) ...



Du\_jing 2015-11-05 17:22 197

## Java基础小知识——基本数据类型和字符串以及字节数组的转换 (http://jasonliao.iteye.com/blog/367724)

Java基础小知识——基本数据类型和字符串以及字节数组的转换 <span style=



Jasonliao 2005-08-20 21:12 839

## Java之美[从菜鸟到高手演变]之数据结构基础、线性表、栈和队列、数组和字符串 (/zhangerqing/article/details/8796518)

Java面试宝典之数据结构基础——线性表篇作者：egg邮箱：[xtfggef@gmail.com](mailto:xtfggef@gmail.com)微博：<http://weibo.com/xtfggef>博客：<http://blog.csdn.net/...>



zhangerqing 2013-04-14 22:37 32639

## Java基础小知识——基本数据类型和字符串以及字节数组的转换 (<http://java2000-net.iteye.com/blog/242046>)

Java基础小知识——基本数据类型和字符串以及字节数组的转换



java2000.net 2005-08-20 21:12 459

## [阶段一]Java基础语法—运算符、流程控制、数组 (/seafishyls/article/details/51534830)

广州近几天的天气非常闷热，整个人都不好了，中午的时候有点提不起精神来。每次上课前都有个习惯，就是喝杯红牛提提神。原因很简单，面对无精打采的老师，学生怎么能提起精神来听课呢。于是乎使劲让自己清醒，争取...



seafishyls 2016-05-29 23:47 355

## Java编程那些事儿44—数组基础语法 (<http://jasonliao.iteye.com/blog/367542>)

Java编程那些事儿44—数组基础语法 郑州游戏学院陈跃峰 出自：<http://blog.csdn.net/mailbomb> 6.2 数组基本语法 了解了数组的概念以后，下面来看一下数组的语法格式。数组



Jasonliao 2008-06-22 12:10 363

## Java小程序之自定义数组队列的实现（高级画板重绘基础） (/bluesky\_usc/article/details/53166403)

Java小程序之自定义数组队列的实现（高级画板重绘基础） 前言：经过一番努力，我们的画板已经完成了很大一部分，很多功能都已经通过我们自己的分析和代码实现了，但是还是有小bug呢？你找到小bug了吗？...



BlueSky\_USC 2016-11-15 00:25 828

## [黑马程序员][java基础学习]05——方法和数组 (<http://lqg5522.iteye.com/blog/2206602>)

-----Java培训、Android培训、iOS培训、.Net培训、期待与您交流！ ----- 1.方法 1)概念： 就是完成特定功能的代码块。 2)格式 <



不上课也要睡觉 2015-04-27 22:07 178