

关于java数组的深度思考 PDF转换可能丢失图片或格式，建议  
阅读原文

[https://www.100test.com/kao\\_ti2020/138/2021\\_2022\\_\\_E5\\_85\\_B3\\_E4\\_BA\\_8Ejava\\_c97\\_138723.htm](https://www.100test.com/kao_ti2020/138/2021_2022__E5_85_B3_E4_BA_8Ejava_c97_138723.htm) 刚刚开始接触java数组的人都会听到一句类似的话：java是纯面向对象的语言，他的数组也是一个对象。于是乎，笔者就按照一个对象的方式来使用数组，心安理得。直到我接触到C的数组后，才发现将数组作为一个类来使用在实现上是多么的“不自然”。首先我们看一下表面现象，数组创建的时候采用的是如下语句：`MyClass[] arr = new MyClass[9]`. 而普通类采用的是如下语句：`MyClass obj = new MyClass()`. 就是说，创建数组的时候不使用小括号传参。使得数组和普通类看起来就有很多不同，因为小括号里的参数是传递给构造方法的，进而让人感觉数组类是没有构造方法的。再往深了想，还有很多让人感觉不自然的东西。可以肯定的是，java确实将数组作为了一个类来处理。还是用上面的例子说明：可以通过以下方法得到`MyClass[]`的Class实例：`arr.getClass()`或`MyClass[].class`。这样，我就可以向数组类里面“窥探”了。`Class clazz = MyClass[].class`.

`System.out.println(clazz.getConstructors().length)`. 打印出来的结果是0；证明数组类确实没有构造方法。如果强行执行`clazz.newInstance()`.就会得到下面的错误。

`java.lang.InstantiationException: [Larraytest.MyClass`. 证明数组类不能够通过普通的反射方式来创建一个实例。再看看数组类的“庐山真面目”：`System.out.println(clazz)`. 输出是：

`[Larraytest.MyClass` 对Java Class文件结构稍有了结就知道，这个字符串的意思就是一个元素类型为`arraytest.MyClass`的一维

数组。也就是说，数组类型不是和普通类一样，以一个全限定路径名 类名来作为自己的唯一标示的，而是以[ 一个或者多个L 数组元素类全限定路径 类来最为唯一标示的。这个()也是数组和普通类的区别。而这个区别似乎在某种程度上说明数组和普通java类在实现上有很大区别。因为java虚拟机(java指令集)在处理数组类和普通类的时候，肯定会做出区分。笔者猜想，可能会有专门的java虚拟机指令来处理数组。

100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)