

Java中四个关键字用法的简要介绍Java认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/644/2021_2022_Java_E4_B8_AD_E5_9B_9B_c104_644620.htm 一:transient 使用对象：属性 字段不是对象持久状态的一部分，不应该把字段和对象一起串起。当一个对象被串行化的时候，transient型变量的值不包括在串行化的表示中，然而非transient型的变量是被包括进去的。当串行化某个对象时，如果该对象的某个变量是transient，那么这个变量不会被串行化进去。也就是说，假设某个类的成员变量是transient，那么当通过ObjectOutputStream把这个类的某个实例保存到磁盘上时，实际上transient变量的值是不会保存的。

二:strictfp 使用对象：类、方法 自Java2以来，Java语言增加了一个关键字strictfp，虽然这个关键字在大多数场合比较少用，但是还是有必要了解一下。strictfp的意思是FP-strict，也就是说精确浮点的意思。在Java虚拟机进行浮点运算时，如果没有指定strictfp关键字时，Java的编译器以及运行环境在对浮点运算的表达式是采取一种近似于我行我素的行为来完成这些操作，以致于得到的结果往往无法令你满意。而一旦使用了strictfp来声明一个类、接口或者方法时，那么所声明的范围内Java的编译器以及运行环境会完全依照浮点规范IEEE-754来执行。因此如果你想让你的浮点运算更加精确，而且不会因为不同的硬件平台所执行的结果不一致的话，那就请用关键字strictfp。你可以将一个类、接口以及方法声明为strictfp，但是不允许对接口中的方法以及构造函数声明strictfp关键字，例如下面的代码：

1. 合法的使用关键字strictfp

```
strictfp interface A {} public strictfp class FpDemo1 {
```

`strictfp void f() {} }` 2. 错误的使用方法 `interface A { strictfp void f(). }` `public class FpDemo2 { strictfp FpDemo2() {} }` 一旦使用了关键字`strictfp`来声明某个类、接口或者方法时，那么在这个关键字所声明的范围内所有浮点运算都是精确的，符合IEEE-754规范的。例如一个类被声明为`strictfp`，那么该类中所有的方法都是`strictfp`的。百考试题 - 全国最大教育类网站(100test.com) 三: `volatile` 使用对象：属性介绍：因为异步线程可以访问字段，所以有些优化操作是一定不能作用在字段上的。`volatile`有时可以代替`synchronized`。四: `final` 对与java中的`final`变量，java编译器是进行了优化的。每个使用了`final`类型变量的地方都不会通过连接而进行访问。比如说 `Test`类中使用了`Data`类中一个`final`的`int`数字`fNumber=77`，这时候，java编译器会将77这个常数编译到`Test`类的指令码或者常量池中。这样，每次`Test`类用到`fNumber`的时候，不会通过引用连接到`Data`类中进行读取，而是直接使用自己保存在类文件中的副本。用程序说话：C:/ 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