

JAVA更多的类谜题72：终极危难 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_JAVA_E6_9B_B4_E5_A4_9A_c97_138573.htm 本谜题旨在检验当你试图隐藏

一个final域时将要发生的事情。下面的程序将做些什么呢？

```
class Jeopardy { public static final String PRIZE = "$64,000". } public class DoubleJeopardy extends Jeopardy { public static final String PRIZE = "2 cents". public static void main(String[ ] args) { System.out.println(DoubleJeopardy.PRIZE). } }
```

因为在Jeopardy中的PRIZE域被声明为是public和final的，你可能会认为Java语言将阻止你在子类中重用该域名。毕竟，final类型的方法不能被覆写或隐藏。如果你尝试着运行该程序，就会发现它可以毫无问题地通过编译，并且将打印2 cents。出什么错了呢？

可以证明，final修饰符对方法和域而言，意味着某些完全不同的事情。对于方法，final意味着该方法不能被覆写（对实例方法而言）或者隐藏（对静态方法而言）[JLS 8.4.3.3]。对于域，final意味着该域不能被赋值超过一次[JLS 8.3.1.2]。关键字相同，但是其行为却完全不相关。在该程序中，final

域DoubleJeopardy.PRIZE隐藏了final域Jeopardy.PRIZE，其净损失达到了\$63,999.98。尽管我们可以隐藏一个域，但是通常这都是一个不好的念头。就像我们在谜题66中所讨论的，隐藏域可能会违反包容性，并且会混淆我们对类型与其成员之间的关系所产生的直觉。如果你想保证在Jeopardy类中的奖金可以保留到子类中，那么你应该用一个final方法来代替final域

```
： class Jeopardy { private static final String PRIZE = "$64,000". public static final String prize() { return PRIZE. } }
```

对语言设计者来

说，其教训是应该避免在不相关的概念之间重用关键字。一个关键字应该只在密切相关的概念之间重用，这样可以帮助程序员构建关于易混淆的语言特性之间的关系的印象。在Java的final关键字这一案例中，重用就导致了混乱。应该注意的是，作为一种有年头的语言来说，在无关的概念之间重用关键字是它的一种自然趋势，这样做可以避免引入新的关键字，而引入新的关键字会对语言的稳定性造成极大的损害。当语言设计者在考虑该怎么做时，总是在两害相权取其轻。总之，要避免在无关的变量或无关的概念之间重用名字。对无关的概念使用有区别的名字有助于让读者和程序员区分这些概念。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