

实例讲解main()中方法调用的问题计算机等级考试 PDF转换  
可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/645/2021\\_2022\\_\\_E5\\_AE\\_9E\\_E4\\_BE\\_8B\\_E8\\_AE\\_B2\\_E8\\_c97\\_645406.htm](https://www.100test.com/kao_ti2020/645/2021_2022__E5_AE_9E_E4_BE_8B_E8_AE_B2_E8_c97_645406.htm) public class

```
invokeMethod{ public void showMsg(){ System.out.println("this is showMsg"). } public void callOther(){ showMsg(). } public static void main(String args[]){ invokeMethod ob=new invokeMethod().ob.callOther(). } }
```

在上例中，方法callOther()和方法showMsg()处在同一个类中，所以调用后者时直接使用方法名就可以了。

令人比较疑惑的地方是在main()方法中，此处调用

callOther()方法使用了看似比较麻烦的办法，先创建一个对象ob，再用对象名.方法的格式来调用方法，这似乎是多此一举实际上这里做是必须的。main()方法是一个静态方法，它由系统来调用，系统在调用它的时候并没有创建一个

invokeMethod的对象，而callOther和showMsg方法都是实例方法，他们被调用时必须要有对象存在，所以必须在main()方法中先创建一个对象ob才能调用这两个方法，从这点来看

，main()方法虽然在invokeMethod类的内部，但它的表现如同在类的外部一样。来源：[www.examda.com](http://www.examda.com) 这么解释可能有的

同学还有疑惑，为什么callOther方法又能够直接调用

showMsg()，难道它能保证在调用后者时对象已经存在，答案却是如此，因为callOther本身是实例方法，它在执行时一定有对象存在，基于这个前提，它能够直接调用showMsg方法。

编辑特别推荐: Java每日一题系列 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)