

java静态内部类(嵌套类) PDF转换可能丢失图片或格式，建议
阅读原文

https://www.100test.com/kao_ti2020/278/2021_2022_java_E9_9D_99_E6_80_81_c104_278569.htm 如果你不需要内部类对象与其
外围类对象之间有联系，那你可以将内部类声明为static。这
通常称为嵌套类（nested class）。想要理解static应用于内部类
时的含义，你就必须记住，普通的内部类对象隐含地保存了
一个引用，指向创建它的外围类对象。然而，当内部类
是static的时，就不是这样了。嵌套类意味着：1. 要创建嵌套
类的对象，并不需要其外围类的对象。2. 不能从嵌套类的对
象中访问非静态的外围类对象。

```
public class Outer {  
    private static int i = 1; private int j = 10; public static void outer_f1() {}  
    public void outer_f2() {} // 静态内部类可以  
    // 用public,protected,private修饰 // 静态内部类中可以定义静态或  
    // 者非静态的成员 static class Inner { static int inner_i = 100; int inner_j  
    = 200; static void inner_f1() { //静态内部类只能访问外部类的静  
    态成员(包括静态变量和静态方法) System.out.println("Outer.i"  
    i). outer_f1(). } void inner_f2() { // 静态内部类不能访问外部类的  
    非静态成员(包括非静态变量和非静态方法) //  
    System.out.println("Outer.i" j). // outer_f2(). } }  
    public void outer_f3() { // 外部类访问内部类的静态成员：内部  
    类.静态成员 System.out.println(Inner.inner_i). Inner.inner_f1(). //  
    外部类访问内部类的非静态成员:实例化内部类即可 Inner  
    inner = new Inner(). inner.inner_f2(). } public static void  
    main(String[] args) { new Outer().outer_f3(). } }
```

生成一个静态内
部类不需要外部类成员：这是静态内部类和成员内部类的区

别。静态内部类的对象可以直接生成：`Outer.Inner in = new Outer.Inner()`.而不需要通过生成外部类对象来生成。这样实际上使静态内部类成为了一个顶级类(正常情况下，你不能在接口内部放置任何代码，但嵌套类可以作为接口的一部分，因为它是static的。只是将嵌套类置于接口的命名空间内，这并不违反接口的规则) 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