

this与super的使用方法 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_this_E4_B8_8Esup_c104_278564.htm 通过用static来定义方法或成员，为我们编程提供了某种便利，从某种程度上可以说它类似于C语言中的全局函数和全局变量。但是，并不是说有了这种便利，你便可以随处使用，如果那样的话，你便需要认真考虑一下自己是否在用面向对象的思想编程，自己的程序是否是面向对象的。好了，现在开始讨论this&super这两个关键字的意义和用法。在Java中，this通常指当前对象，super则指父类的。当你想要引用当前对象的某种东西，比如当前对象的某个方法，或当前对象的某个成员，你便可以利用this来实现这个目的，当然，this的另一个用途是调用当前对象的另一个构造函数，这些马上就要讨论。如果你想引用父类的某种东西，则非super莫属。由于this与super有如此相似的一些特性和与生俱来的某种关系，所以我们在这一块儿来讨论，希望能帮助你区分和掌握它们两个。在一般方法中最普遍的情况就是，在你的方法中的某个形参名与当前对象的某个成员有相同的名字，这时为了不至于混淆，你便需要明确使用this关键字来指明你要使用某个成员，使用方法是“this.成员名”，而不带this的那个便是形参。另外，还可以用“this.方法名”来引用当前对象的某个方法，但这时this就不是必须的了，你可以直接用方法名来访问那个方法，编译器会知道你要调用的是那一个。下面的代码演示了上面的用法：

```
public class DemoThis{ private String name. private int age. DemoThis(String name,int age){ setName(name). //你可以加上this来调用方法，
```

像这样：`this.setName(name)`。但这并不是必须的 `setAge(age)`。
`this.print()`。 `br> } public void setName(String name){`
`this.name=name.//此处必须指明你要引用成员变量 } public void`
`setAge(int age){ this.age=age. } public void print(){`
`System.out.println("Name=" name " ge=" age). //在此行中并不需`
`要用this，因为没有会导致混淆的东西 } public static void`
`main(String[] args){ DemoThis dt=new DemoThis("Kevin","22").`
这段代码很简单，不用解释你也应该能看明白。在构造函数中你看到用 `this.print()`，你完全可以用 `print()` 来代替它，两者效果一样。下面我们修改这个程序，来演示 `super` 的用法。
`class`
`Person{ public int c. private String name. private int age. protected`
`void setName(String name){ this.name=name. } protected void`
`setAge(int age){ this.age=age. } protected void print(){`
`System.out.println("Name=" name " Age=" age). } }`
`public class`
`DemoSuper extends Person{ public void print(){`
`System.out.println("DemoSuper:"). super.print(). } public static`
`void main(String[] args){ DemoSuper ds=new DemoSuper().`
`ds.setName("kevin"). ds.setAge(22). ds.print(). } }`
在 `DemoSuper` 中，重新定义的 `print` 方法覆写了父类的 `print` 方法，它首先做一些自己的事情，然后调用父类的那个被覆写了的方法。输出结果说明了这一点：
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