

Java编程中异常处理的特殊情况 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/206/2021_2022_Java_E7_BC_96_E7_A8_8B_c104_206904.htm 1、不能在finally块中执

行return,continue等语句，否则会把异常“吃掉”；2、在try，catch中如果有return语句，则在执行return之前先执行finally块 请大家看下面的例子：public class TryTest { public static void main(String[] args) { try { System.out.println(TryTest.test()).// 返回结果为true其没有任何异常 } catch (Exception e) {

```
System.out.println("Exception from main"). e.printStackTrace(). }
doThings(0). } public static boolean test() throws Exception { try {
throw new Exception("Something error").// 第1步.抛出异常 }
catch (Exception e) { // 第2步.捕获的异常匹配(声明类或其父类)
， 进入控制块 System.out.println("Exception from e").// 第3步.
打印 return false.// 第5步. return前控制转移到finally块,执行完后
再返回（这一步被吃掉了，不执行） } finally { return true. //
第4步. 控制转移，直接返回，吃掉了异常 } } public static void
doThings(int i) { try { if(i==0) { //在执行return之前会先执
行finally return. } int t=100/i. System.out.println(t).
} catch (Exception ex) { ex.printStackTrace(). } finally {
System.out.println("finally"). } } } 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
```