

Java中测试基本异常计算机等级考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/638/2021_2022_Java_E4_B8_AD_E6_B5_8B_c97_638823.htm 编辑特别推荐: 全国计算机等级考试(等考)指定教材 全国计算机等级考试学习视频 全国计算机等级考试网上辅导招生 全国计算机等级考试时间及科目预告 百考试题教育全国计算机等级考试在线测试平台 全国计算机等级考试资料下载 全国计算机等级考试论坛 1、第一个JAVA例程：hello world 执行时的问题：problems Executing Hello 如果你看到下面的错误提示: java is not recognized as an internal or external command, ... or java: Command not found. 那么你或许没有安装java或者在环境变量中并没有设置path路径java\bin目录。请检查你的java安装以及环境变量的正确设置。如果你看到: Exception in thread "main"

java.lang.NoClassDefFoundError: hello (wrong name: Hello) ... 那么你很有可能是因为没有注意大小写或是把名称中的字母打错了。如：java hello 键入正确的命令重新运行即可。如果你看到: Exception in thread "main" java.lang.NoClassDefFoundError: Hello /class 那么你需要考虑是否去掉名字中的.class部分。 2、分析出现java.lang.NoClassDefFoundError异常的具体情况 这个问题在上面调试第一个例程出现过，但是如果我们输入的名称是正确的，仍然提示这样的问题时怎么办呢？装

了JDK1.4.0版，而且JAVA_HOME，PATH和CLASSPATH都设置好了。如下 JAVA_HOME=/home/jdk

CLASSPATH=\$JAVA_HOME/jre/lib/tools.jar:\$JAVA_HOME/lib/dt.jar:\$JAVA_HOME/jre/lib/:\$JAVA_HOME/lib

PATH=\$PATH:\$JAVA_HOME/bin 用JAVAC命令生成
了hello.class,当使用java hello时，仍然提示：Exception in thread
"main" java.lang.NoClassDefFoundError:XXX (wrong name: XXX
) 解决提示：(1)、Add (current directory) to your classpath或类
定义前面加public (2)、path环境变量中设置的路径中是否有另
一个java.exe文件，默认执行的是这个目录下的java.exe，虽然
版本与JDK下的java.exe一致，但是在此目录下执行就会出错
。方法：安装了JDK的情况下，应该删除winnt下的java.exe，
确保默认调用的java.exe是位于jdk中bin目录下的java.exe。(3)
、假如你使用了packet 那么检查你在编译了没有将指
定package包在你文件中的java文件。3、通常异常捕获处理：
JAVA中几个基本异常简单的描述一下：ArithmeticException
当出现异常算术条件时产生 NullPointerException 当应用程序
企图使用需要的对象处为空时产生
ArrayIndexOutOfBoundsException 数组下标越界时产生
ArrayStoreException 当程序试图存储数组中错误的类型数据时
产生 FileNotFoundException 试图访问的文件不存在时产生
IOException 由于一般I/O故障而引起的，如读文件故障
NumberFormatException 当把字符串转换为数值型数据失败时
产生 OutOfMemoryException 内存不足时产生
SecurityException 当小应用程序（Applet）试图执行由于浏览
器的安全设置而不允许的动作时产生 StackOverflowException
当系统的堆栈空间用完时产生
StringIndexOutOfBoundsException 当程序试图访问串中不存在的
的字符位置时产生 上述出现的异常基本上都是JAVA中我们可
能会遇到的异常。类Throwable有两个直接子类：Error

和Exception。Exception类对象是Java程序处理或抛弃的对象。Java提供了两种Exception的模式，一种是执行的时候所产生的Exception (Runtime Exception)，另外一种则是受控制的Exception (Checked Exception)。所有的Checked Exception 均从java.lang.Exception 继承而来，而Runtime Exception 则继承java.lang.RuntimeException 或java.lang.Error (实际上java.lang.RuntimeException 的上一层也是java.lang.Exception)。它有各种不同的子类分别对应于不同类型的例外。其中类RuntimeException代表运行时由Java虚拟机生成的例外。从程序的运作机制上看，Runtime Exception与Checked Exception 不一样，从逻辑上看，Runtime Exception 与Checked Exception 在使用的目的上也不一样。一般而言，Checked Exception 表示这个Exception 必须要被处理，也就是说程序设计者应该已经知道可能会收到某个Exception(因为要try catch住)，所以程序设计者应该能针对这些不同的Checked Exception 做出不同的处理。而Runtime Exception 通常会暗示着程序上的错误，这种错误会导致程序设计者无法处理，而造成程序无法继续执行下去。Java的可控制异常处理是通过5个关键字来实现的：try，catch，throw，throws，finally。JB的在线帮助中对这几个关键字是这样解释的：Throws: Lists the exceptions a method could throw. Throw: Transfers control of the method to the exception handler. Try：Opening exception-handling statement. Catch：Captures the exception. Finally：Runs its code before terminating the program. try语句 try语句用大括号{}指定了一段代码，该段代码可能会抛弃一个或多个例外。100Test 下载频道开通，各类考试题目直接下载。详细请访问

