

JAVA教程第四讲Java的例外处理和I_O流4.2 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/138/2021_2022_JAVA_E6_95_99_E7_A8_8B_c97_138746.htm

4.2 例外的处理 java语言中有两种例外处理机制：捕获例外和声明抛弃例外。下面我们做详细介绍。

4.2.1 捕获例外

捕获例外是通过try-catch-finally语句实现的。try{ }catch(ExceptionName1 e){ }catch(ExceptionName2 e){ } }finally{ }

try 捕获例外的第一步是用try{...}选定捕获例外的范围，由try所限定的代码块中的语句在执行过程中可能会生成例外对象并抛弃。 catch 每个try代码块可以伴随一个或多个catch语句，用于处理try代码块中所生成的例外事件。catch语句只需要一个形式参数指明它能够捕获的例外类型,这个类必须是Throwable的子类,运行时系统通过参数值把被抛弃的例外对象传递给catch块。在catch块中是对例外对象进行处理的代码，与访问其它对象一样，可以访问一个例外对象的变量或调用它的方法。

getMessage()是类Throwable所提供的方法，用来得到有关异常事件的信息，类Throwable还提供了方法printStackTrace()用来跟踪异常事件发生时执行堆栈的内容。例如：

```
try{ .....
}catch( FileNotFoundException e ){ System.out.println( e ).
System.out.println( "message: " e.getMessage() ). e.printStackTrace(
System.out ). }catch( IOException e ){ System.out.println( e ). }
```

catch 语句的顺序：捕获例外的顺序和catch语句的顺序有关，当捕获到一个例外时，剩下的catch语句就不再匹配。因此，在安排catch语句的顺序时，首先应该捕获最特殊的例外，然后再逐渐一般化。也就是一般先安排子类，再安排父类

。 finally 捕获例外的最后一步是通过finally语句为例外处理提供一个统一的出口，使得在控制流转到程序的其它部分以前，能够对程序的状态作统一的管理。不论在try代码块中是否发生了异常事件，finally块中的语句都会被执行。 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