



情境1：异常概述

1.异常的定义

2.异常类



异常处理的概念

- 异常 (Exception) :
 - 指由于程序运行时发生的错误，从而导致程序错误结束。
- 异常处理：
 - 当程序运行过程中发生了某个异常现象，系统将产生一个相应的异常类对象，并把它交给系统处理，系统负责找到处理错误的代码并执行之。
- 如：

<pre>{ 输入数据x; 5/x; }</pre>	<pre>{ 输入数据x; if (x==0) {提示输入数据有误;} else 5/x; }</pre>
--	---

如果输入**X**的值为**0**，
产生异常。

一般的处理异常方法



利用JAVA语言提供的异常处理机制后的代码

```
try {  
    输入数据x;  
    5/x;  
} catch (ArithmeticException e)  
{  
    System.out.println(e);  
}
```

异常的概念

■ 和传统的方法比较异常的优点:

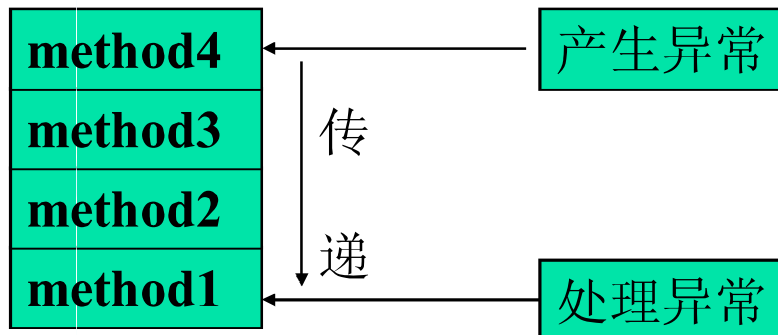
1. 把错误代码从常规代码中分离出来

2. 把错误传播给调用堆栈

3. 按错误类型和错误差别分组

4. 系统提供了对于一些无法预测的错误的捕获和处理

5. 克服了传统方法的错误信息有限的问题



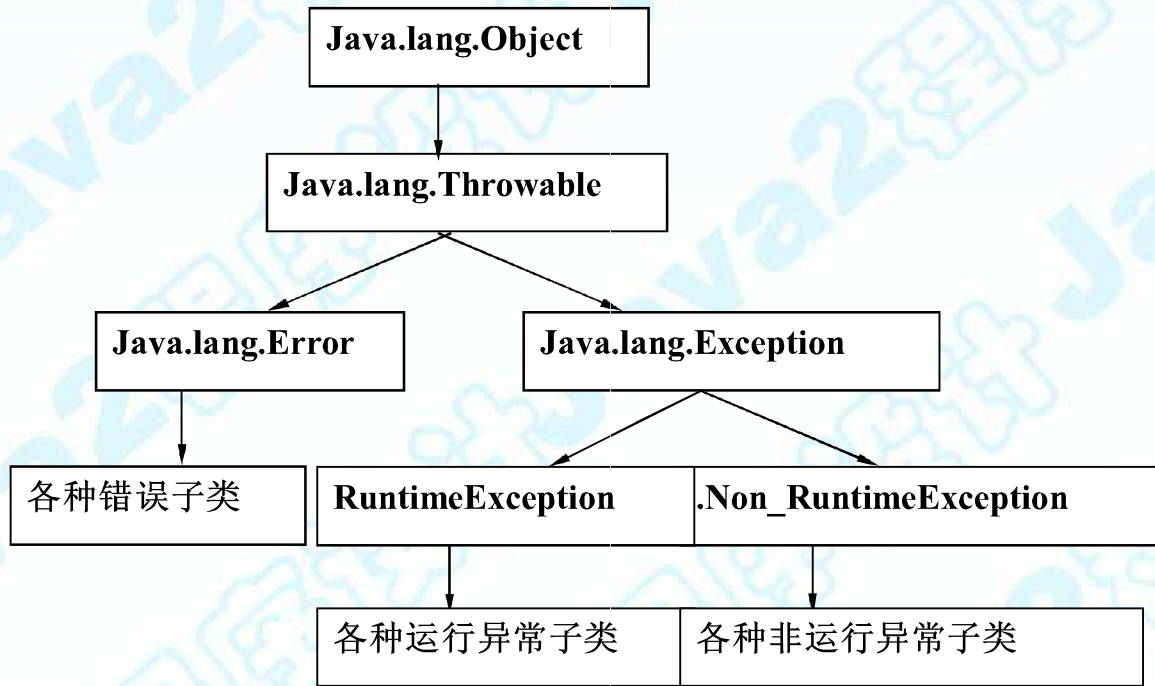


异常类

- 当程序发生异常时，会生成某个异常类的对象。
- 异常类对象包括关于异常的信息、类型和错误发生时程序的状态以及对该错误的详细描述。
- **Throwable**是**java.lang**包中一个专门用来处理异常的类。有两个子类：**Exception**（异常）类和**Error**（错误）类。
 - **Exception**类：是可以被捕捉并且可能恢复的异常类，也可以继承**Exception**类生成自己的异常类。
 - **Error**类：一般被认为是不可恢复和不可捕捉的异常类，用户程序不需要处理这类异常。



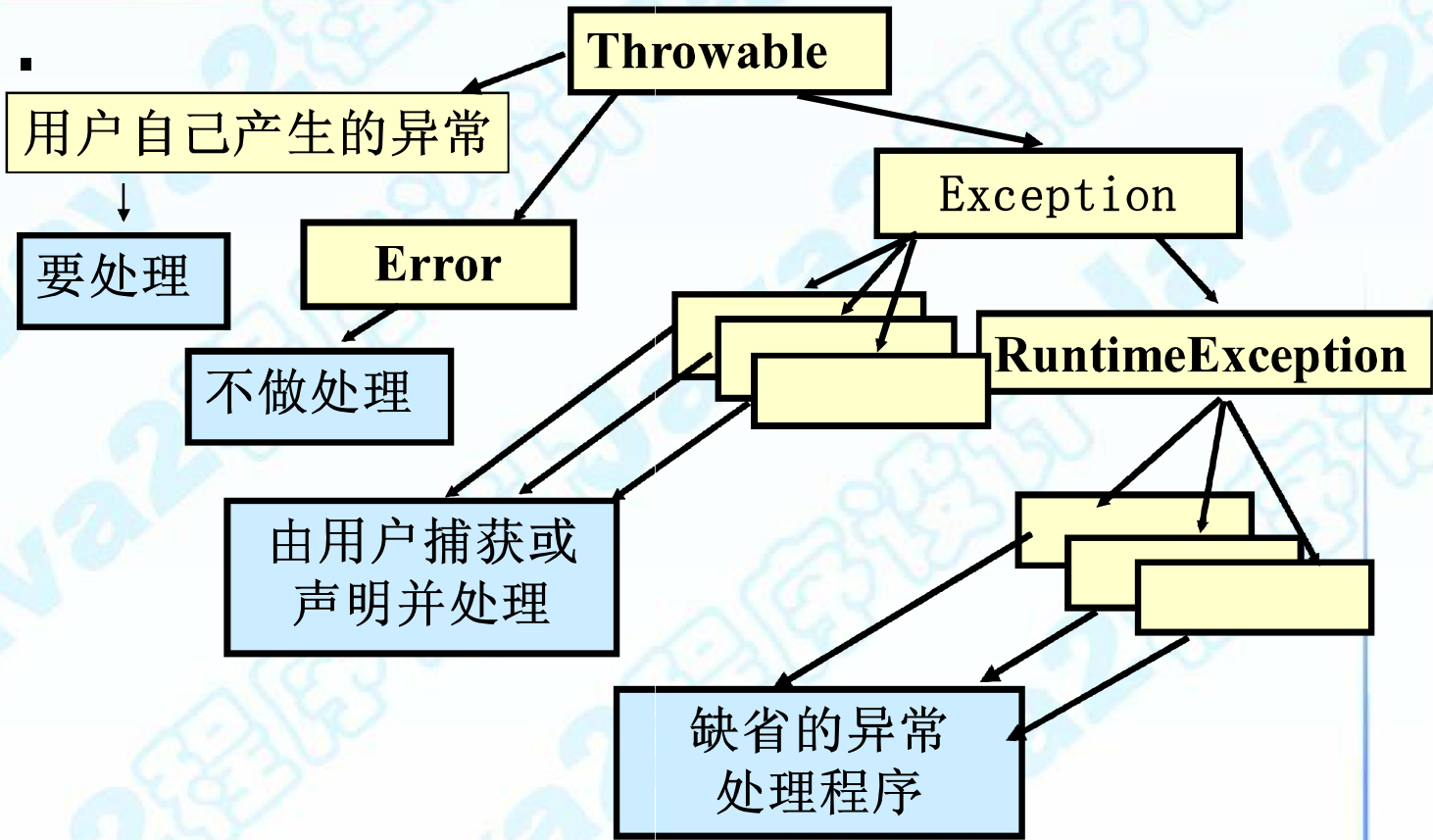
异常类的层次结构



异常类的层次



异常的分类





Exception类及其子类

- **Exception**类分为**RuntimeException**（运行异常）类和**Non-RuntimeException**（非运行异常）类两大类。

异常类名	功能说明
ArithmeticException	除数为零的异常
IndexOutOfBoundsException	下标越界异常
ClassCastException	类型转换异常
ArrayIndexOutOfBoundsException	访问数组元素的下标越界异常
NoClassDefFoundException	未找到类定义异常
FileNotFoundException	指定的文件没有发现
IOException	输入输出错误
NullPointerException	访问一个空对象中的成员时产生的异常

运行异常



异常类名	功能说明
ClassNotFoundException	指定类或接口不存在的异常
IllegalAccessException	非法访问异常
IOException	输入输出异常
FileNotFoundException	找不到指定文件的异常
ProtocolException	网络协议异常
SocketException	Socket操作异常
MalformedURLException	统一资源定位器（URL）的格式不正确的异常

非运行异常



Error类及其子类

- **Error**类定义了正常情况下不希望捕捉的错误。

Error异常类

异常类名	功能说明
LinkageError	动态链接失败
VirtualMachineError	虚拟机错误
AWTError	AWT错误



Thank you!

