

Java07 异常处理

Task 1:

异常 Exception 和 Error 的概念:

什么叫异常:

异常就是在程序运行过程中发生的不正常的事件，会中断正在运行的程序。异常本身就是 Java 对可能出现问题进行描述的一种对象体现。

常见的异常有:

1. 除数不能为 0 (ArithmeticException):

例如: `System.out.println(10/0)` 在编译时不会报错, 但是在运行时会报错, 原因就是除数为 0;

2. 空指针异常 (NullPointerException):

例如:

```
String a = null;
```

```
a.length();
```

就会出现空指针异常, 原因通常有:

1. 未初始化的对象引用;
2. 对象引用为 null
3. 数组未初始化或越界
4. 外部资源或数据库查询返回 null;
5. 方法返回 null

3. 数组下标越界异常 (ArrayIndexOutOfBoundsException);
是指在访问数组时使用了超出数组长度的索引值, 通常由于:

- a. 使用负数作为数组下标;
- b. 使用超出数组长度的值作为数组下标;

4. 类型转换异常 (ClassCastException);

是指进行了非法的数据类型转换。主要指向下转换和将一个对象转换为于原类型不兼容的类型 (例如将一个 String 对象转换成 Integer 类型)

常见 Error 类型:

1. 内存溢出错误 (OutOfMemoryError):

是指 JVM 在运行过程中, 由于堆内存不足, 无法为对象分配内存, 导致程序抛出错误, 具体可分为:

- a. Java Heap Space: 堆内存不足以分配对象;
- b. GC Overhead Limit Exceeded, 由于大量的对象频繁创建和销毁。
- c. Metaspace Metaspace 空间不足;

2. 栈内存溢出错误 (StackOverflowError):

在程序运行过程中栈的深度超过了虚拟机容许的最大深度, 主要因为递归调用过深和大量方法和局部变量被调用。

Checked 和 Unchecked 异常

概念:

Checked 异常

在编译时必须检查的异常，否则编译器会报错，例如
IOException

得在代码中显示处理

例如：IO 错误 (IOException):

输入输出操作时发生的异常

1. 文件不存在或者无法访问
2. 网络连接问题
3. 资源关闭异常
4. 不支持的字符编码

Unchecked 异常

不需要在编译时进行检查或处理!

例如空指针异常和数组越界异常，是 RuntimeException 或者
其子类

其例子见上文;

程序对 Exception 和 Error 的态度区别:

Exception 是程序运行时可能预料得到的异常情况，应该捕获并处理，而 Error 表示程序遇到了无法解决的状态，不需要也不应该被捕获。

