



Chapter 6 异常处理

An exception is an event, which occurs during the execution of a program, that disrupts the normal flow of the program's instructions.

1. 异常的产生

- o `java.lang.ArithmeticException`
- o `java.lang.StringIndexOutOfBoundsException`
- o `java.lang.ArrayIndexOutOfBoundsException`
- o `java.lang.NumberFormatException`
- o `java.lang.NullPointerException`

2. 异常的处理

```
try {  
    // 可能产生异常的语句块  
} catch (ExceptionType exceptionType) {  
    // 异常的处理  
} finally {  
    // 其他的处理  
}  
...
```

- `try` 后面必须有 `catch` 或 `finally`
- `catch` 可以有多个
- `finally` 最多只能有一个
- `finally` 位于 `catch` 的后面
- `try` 内一旦发生异常, `try` 内这条语句后的代码都不再执行, 无论异常有没有被 `catch`
- `finally` 语句块总是会被执行
- 异常的处理方式
 - 输出异常信息 `e.printStackTrace();`
 - 退出程序 `System.exit(1)`
 - 针对特定异常的更积极的处理方式

3. 异常的分类

```
``java  
java.lang.Object  
    java.lang.Throwable  
        java.lang.Error  
            java.lang.Exception  
                java.lang.RuntimeException  
                java.io.IOException  
                *.*.Exception  
````
```

- 非受检异常 `unchecked exception`
  - > `RuntimeException`类及其子类是非受检异常
- 受检异常 `checked exception`
  - > `Exception`类中除了`RuntimeException`之外的其他异常类及其子类

- 非受检异常可以通过编译，可能在运行时产生
- 受检异常不能通过编译，必须对其进行处理
  - 受检异常的处理方式
    1. 使用 `try` / `catch`` 包围
      - 在方法中声明这个异常 `throws``

#### 4. 显式抛出异常 `throw``

#### 5. `try-with-resources`

> since JDK 7

```
```java
try (RandomAccessFile raf = new RandomAccessFile("path-to-file", "rw")){
    System.out.println();
} catch (IOException e) {
    e.printStackTrace();
}
```