

# 后端

## Java07

前言：至少在我着手做07题的时候它是后端最后一道题了，这7道题做过来好像尽是新鲜事物，据学长所说好像往前进也尽是新鲜事物，我等着。

### task1

#### Q1&Q2:

请举例几个具体的Exception和Error类型以及它们发生的原因，当发生Exception和Error时，程序的处理态度分别应该是什么？

一般我会怎么回答？对！查资料然后自己转述！

#### 对于Error:

StackOverflowError(字面意思就是叠得太多了(意会吧))具体解释应该就是程序一直调用，可能是死循环到知道，可能是空间被耗尽导致的。

OutOfMemoryError(字面意思就是内存爆了)具体解释应该就是程序回头一看电脑的空间里已经装满了恶臭的代码，没有多的空间留出来给新的变量了。最可能的原因就是，对象太多了(不是)，还有内存泄漏。

对于这种错误，（其实看也看得出来），程序没法自己调整了，直接报错然后停止运行。

#### 于Exception:

FileNotFoundException（字面意思就是文件找不到了）具体解释应该就是电脑试图打开一个文件，但发现它不存在，大概率是路径不对或者文件莫名丢失了。

IndexOutOfBoundsException（字面意思就是插入超范围）具体解释应该就是我们访问一个数组或者集合的时候输入的索引超出了其有限的范围。（这是我以前经常犯的错，据以往经验很多时候是设计的索引不准确，或者条件没考虑全）

NullPointerException（字面意思就是指向无）具体解释应该就是我们试图去访问一个null的成员，别人都是null你还要访问一下，这不跟询问瞎子看不看得见一样可恶吗！！

对于这种错误，程序的处理态度比较宽松吧，它会自己适当处理来修复代码正常跑，也可能向我们报错。

Exception也分两个小类：

#### checked:

IOException（c++中有个头文件叫iostream输入输出流，这里也就是输入输出中的错误了）具体解释应该就是读取文件时文件损坏，网络出问题等等。

SQLException(SQL数据库语言，自然它时数据库报错)具体解释应该就是与数据库交互时发生的报错，具体原因嘛，我没碰到过...

此种错误会在编译器中显示出来，让我们去改错（头痛...）

#### unchecked:

ArithmeticException（算术错误）具体解释应该是计算上出现一些不合理的式子，比如这道题的10/0了

NumberFormatException(数字转换出错)具体解释应该是在进行字符串转整数的过程中混进去了非数字的字符

这种错误不强制要求你修改，但是多半是我这家伙又乱写代码导致的！！

## task2

### Q3:

好耶，开始读代码了，让我看看怎么个事（我把题目代码复制到这里，应该是会有批注的）

```
class InsufficientFundsException extends Exception {
    //如果英文没认错的话，这个应该是余额不足的报错
    public InsufficientFundsException(String message) {
        super(message);
    } //嗯，它会用到一个message，message代表啥呢？（查了一下是报错要用的信息）
}

class BankAccount {
    private double balance;
    //为什么是balance？它代表啥？
    public BankAccount(double initialBalance) {
        this.balance = initialBalance;
    } //最初的balance 初始化...

    public double getBalance() {
        return balance;
    } //获取balance，查了一下（balance有余额的意思啊）

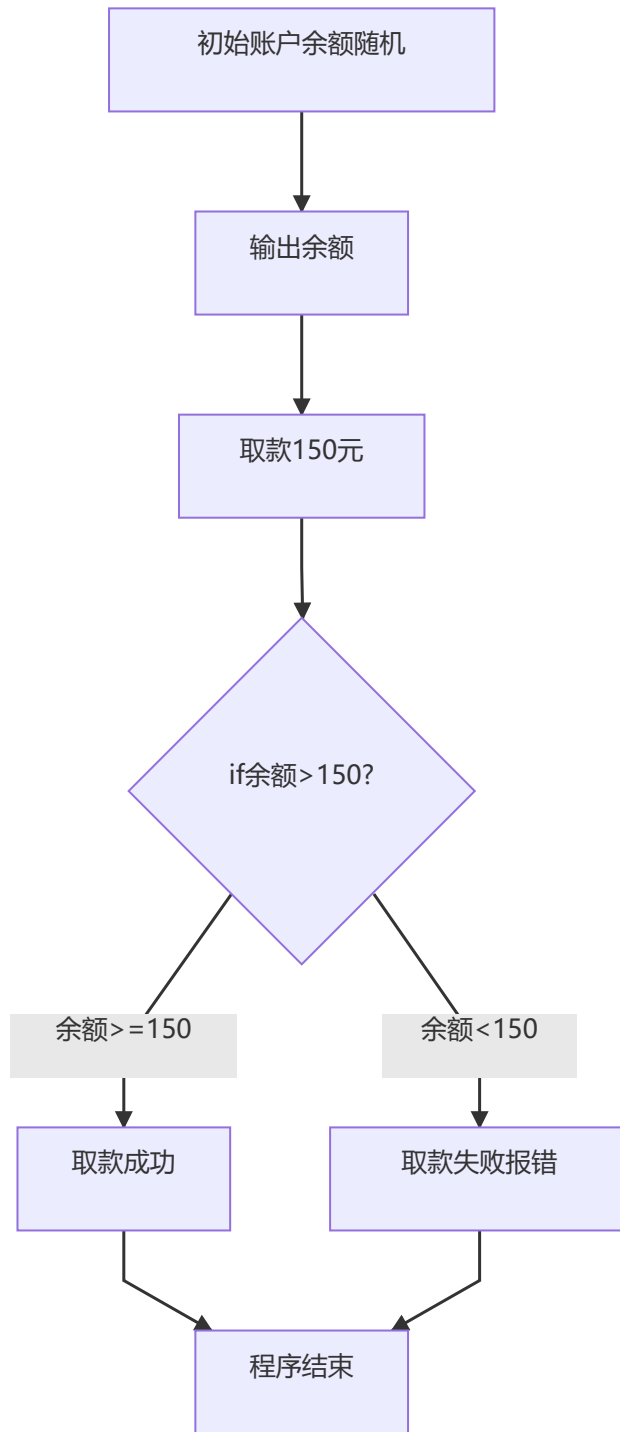
    public void withdraw(double amount) throws InsufficientFundsException {
        if (amount > balance) {
            throw new InsufficientFundsException("余额不足，无法取款。当前余额：" +
            balance); //这就是余额不足的报错的初始化
        }
        balance -= amount; //这是余额充足取出的结果
    }
}

public class BankAccountExample {
    public static void main(String[] args) {
        BankAccount account = new BankAccount(Math.random()*200); //初始银行里的钱随
        机

        try {
            System.out.println("当前余额：" + account.getBalance());
            account.withdraw(150.0);
            System.out.println("取款成功。"); //一次取钱
        } catch (InsufficientFundsException e) {
            System.err.println("错误：" + e.getMessage());
        }

        System.out.println("程序结束");
    }
}
```

为了更好的展示答案，姑且再去看一眼md的流程图语法



其实流程图简化了，其中throw报错的过程没有展示出来，无妨，下一个task就该我自己写了，学长你且看！

### task3

#### Q4:

好多好多的东西不懂，查到最后都不知道查了啥了，总的来说学长可以认为除了06题前出现的东西，其它都统统查了一遍，具体在做md答题整合的时候写到对应的地方我再讲讲有啥想法，嗯嗯（所有我这块贴的代码也是断开的）

```
import java.io.BufferedReader;
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.IOException;
```

因为确实查的很乱我就从这四个说起：

BufferedReader：它用于java的文本字符流输入，可以逐行的读取外部文件

这是我查到的它的一些缺点

由于BufferedReader是基于字符流的，对于读取二进制文件并不适用，不适合处理字节流。缓冲区的大小是固定的，如果输入流的内容超过了缓冲区大小，会导致缓冲区溢出。当BufferedReader类读取数据时，如果发生异常或读取过程中程序意外终止，可能会导致数据丢失

因为用法有很多，我这里就列出我用的那种

```
BufferedReader bufferedreader = new BufferedReader(new FileReader(file));
```

FileReader：这个也是能够以各种形式去读取文件，这里与bufferedReader连用，应该也能够不和bufferedReader连用，但什么时候用什么形式还有待我学习呀（学习不够深入，自罚三杯）

parseInt ()：看到这个函数我是很惊讶的，因为太tm的方便了！！c++我都不敢想几个字符就能给我把string转成整数！

而且这个parseInt () 还得用Integer装包才能赋值给int 这点反而是c更方便

FileNotFoundException与IOException就是查找的题目要求的两个报错用的

当然还有题目要求的NumberFormatException

## 说了引用的类，接下来是有关报错的结构try-catch

应该是把可能报错的代码放进try里 然后catch来捕捉错误并报错

不得不感叹有点高级

### finally:

这个是try-catch-finally中无论报不报错都会执行的代码语句，但我在这里不知道拿它干什么，就输出“文件读取完毕”吧

### 最后是自定义的报错类：

总结下步骤：

#### 1. 定义

```
static class EmptyFileException extends Exception {  
    public EmptyFileException(String message) {  
        super(message);  
    }  
}
```

#### 2. 发生错误时初始化这个错误类

```
throw new 错误("报错语句");
```

#### 3. catch中进行报错

```
catch(错误 e){  
    System.err.println("错误: " + e.getMessage());  
}
```

---

好了这就完成了最终的代码了，已上传github，文件名为Java07题答题代码)

！ 完结撒花 完结撒花！（虽然只是现阶段放出来的题做完了，但也算阶段性的完结啦！）

XY致上