

第七题

Task1

1

Error 表示严重的问题，通常是系统级的问题，通常无法通过代码来处理或恢复
通常情况下，程序无法处理 Errors。它们往往代表了一个严重的问题，通常无法用常规手段来恢复，一般建议记录日志、通知用户或采取备份措施后退出程序

Exception 表示可预见的问题，是在程序运行过程中可能发生的问题的信号
通过try-catch-finally语句块来捕获和处理异常。

error

OutOfMemoryError（内存溢出）
VirtualMachineError（虚拟机错误）

exception

FileNotFoundException：尝试打开一个不存在的文件
SQLException：数据库操作失败时，比如查询语法错误或者连接问题
IOException：数据输入/输出操作出现问题，例如磁盘满、文件损坏等
NullPointerException：忘记初始化变量或者引用了null对象引起的
ArrayIndexOutOfBoundsException：尝试访问数组的一个越界索引
ArithmeticException：如除数为零

2

checked:在编译阶段就出现（标红）。
unchecked:在编译阶段不会报错，在运行阶段出错，导致的程序出现问题。
我感觉这两类与Non-Runtime Exception和Runtime Exception其实非常相像，感觉就是叫法不一样。

Checked Exception:

FileNotFoundException：尝试打开一个不存在的文件
SQLException：数据库操作失败时，比如查询语法错误或者连接问题
IOException：数据输入/输出操作出现问题，例如磁盘满、文件损坏等

Unchecked Exception:

NullPointerException：忘记初始化变量或者引用了null对象引起的
ArrayIndexOutOfBoundsException：尝试访问数组的一个越界索引
ArithmeticException：如除数为零

Task2

进入main方法，创建了一个BankAccount对象，随机生成的初始余额（0到200）。

进入try-catch语句，程序打印出当前的账户余额；调用withdraw方。

进入withdraw方法，判断余额和取钱的大小，
如果amount < balance,余额减少，输出取款成功。
如果amount > balance，使用throw关键字，抛出InsufficientFundsException异常对象， throws关键字将（"余额不足，无法取款。当前余额: " + balance)传递至InsufficientFundsException方法， catch开始执行，捕获了InsufficientFundsException异常，并打印出错误信息（来自e.getMessage()，之前储存在InsufficientFundsException方法的内容）。

最后，无论是否发生异常，程序都会打印出"程序结束"，表示程序已经正常退出。

当前余额： 129.17538846724423
程序结束
错误： 余额不足，无法取款。当前余额： 129.17538846724423

当前余额： 199.51855327724058
取款成功。
程序结束

4

Duqu

```
package tuqu;
import java.io.*;
import java.io.IOException;
public class Duqu {

    private static void calculateAverage() {
        try (BufferedReader reader = new BufferedReader(new FileReader("data.txt"))) {
            int sum = 0;
            int count = 0;

            String line;
            while ((line = reader.readLine()) != null) {
                try {
                    int num = Integer.parseInt(line);
                    sum += num;
                    count++;
                } catch (NumberFormatException e) {
```

```

        throw new IllegalArgumentException("读取到的内容格式错误: " + line);
    }
}

if (count == 0) {
    throw new EmptyFileException("文件为空");
}

double average = (double) sum / count;
System.out.println("文件 " + " 的平均值是: " + average);
} catch (FileNotFoundException e) {
    System.err.println(e.getMessage());
} catch (EmptyFileException e) {
    System.err.println(e.getMessage());
} catch (IOException e) {
    throw new RuntimeException(e);
}
}

public static void main (String[] args){
    calculateAverage();
}
}

```

EmptyFileException

```

package tuqu;

public class EmptyFileException extends RuntimeException {
    public EmptyFileException(String message) {
        super(message);
    }
}

```

FileNotFoundException

```

package tuqu;

import java.io.IOException;

public class FileNotFoundException extends IOException{
    public FileNotFoundException(String message) {

        super(message);
    }
}

```

写完了这个还要去写前四题的plus，想si