

1. 异常:

- 1. Checked Exception (受检异常):
 - 例子及原因:
 - java.io.FileNotFoundException : 当程序试图打开一个不存在的文件时发生或路径错误的文件
 - java.sql.SQLException : 在数据库操作中可能出现, 比如数据库连接失败、执行 SQL 语句错误等。
 - 处理态度: 编译器会强制要求开发者对 Checked Exception 进行处理, 可以通过 try-catch 块捕获并处理异常, 或使用 throws 关键字将异常抛给上层调用者处理。
- 2. Unchecked Exception (非受检异常):
 - 例子及原因:
 - java.lang.NullPointerException : 当程序试图访问一个 null 对象的成员 (方法或属性) 时发生。比如, 一个变量未初始化就被使用。
 - java.lang.ArithmeticException : 进行算术运算时出现错误, 如除数为零。
 - **两者区别:**
 - Checked Exception 是在编译阶段就需要被处理的异常, 通常表示可以预期的、可恢复的错误情况, 如文件不存在、数据库连接问题等。
 - Unchecked Exception 通常是由程序员错误导致的, 在编译阶段不强制处理, 但在运行时可能导致程序不正常而终止。
 - 处理态度: 通常由程序员的错误导致, 编译器不强制要求处理。在代码中尽量避免出现这种异常, 也可以使用 try-catch 块进行捕获处理, 以便程序能够处理错误而不是突然崩溃。

2. 错误:

- 例子及原因:
 - java.lang.OutOfMemoryError : 当 JVM 内存不足时发生。
 - java.lang.StackOverflowError : 当递归调用过深, 栈空间被耗尽时发生。
- 处理态度: Error 是严重的问题, 通常不建议在应用程序层面进行处理, 如果出现 Error, 通常意味着程序可能需要重新设计或者检查系统配置等。

task2:

一:取款成功:

1. 首先创建银行账户对象 account , 随机初始余额大于等于 150。
 2. 输出“当前余额: [具体余额数值]”。
 3. 调用 withdraw 方法尝试取款 150, 由于余额充足, 成功减少余额。
 4. 输出“取款成功。”。
 5. 输出“程序结束”。
- 二: 取款失败:
6. 创建银行账户对象 account , 随机初始余额小于 150。
 7. 输出“当前余额: [具体余额数值]”。
 8. 调用 withdraw 方法尝试取款 150, 因为余额不足, 抛出 InsufficientFundsException 异常。
 9. 异常被捕获, 输出“错误: 余额不足, 无法取款。当前余额: [具体余额数值]”。
 10. 输出“程序结束”。

文件读取与数据处理: 文件路径, 数字格式, 代码都要正确时才能求出平均值。



