

# 异常作业

## 一、填空题

1. \_\_\_\_\_ 机制是一种非常有用的辅助性程序设计方法。采用这种方法可以使得在程序设计时将程序的正常流程与错误处理分开，有利于代码的编写和维护。
2. 在 Java 异常处理中可以使用多个 `catch` 子句，此时包含异常类的父类 `Exception` 的 `catch` 子句的位置应该是在\_\_\_\_\_。
3. 异常处理中 `finally` 块可确保无论是否发生异常，该块中代码总能被执行。`finally` 块不执行的唯一情况是在异常处理代码中执行\_\_\_\_\_语句退出 Java 虚拟机。
4. 异常是由 Java 应用程序抛出和处理的非严重错误，比如所需文件没有找到、零作除数，数组下标越界等，可分为两类：`Checked` 异常和\_\_\_\_\_。
5. 在 Java 中对于程序可能出现的检查时异常，要么用 `try...catch` 语句捕获并处理它，要么使用\_\_\_\_\_语句抛出它，由上一级调用者来处理。
6. Java 异常处理中，如果一个方法中出现了多个 `Checked` 异常，可以在方法声明中使用关键字\_\_\_\_\_声明抛出，各异常类型之间使用逗号分隔。
7. Java 中，处理异常两大方式\_\_\_\_\_。

## 二、选择题

1. 以下关于异常的代码的执行结果是（ ）。(选择一项)

```
public class Test {  
    public static void main(String args[]) {  
        try {  
            System.out.println("try");  
            return;  
        } catch (Exception e) {  
            System.out.println("catch");  
        } finally {  
            System.out.println("finally");  
        }  
    }  
}
```

- A. try  
catch  
finally
- B. catch  
finally
- C. try  
finally
- D. try
2. 在异常处理中，如释放资源、关闭文件等由（ ）来完成。(选择一项)
- A try 子句

- B. catch 子句  
C. finally 子句  
D. throw 子句
3. 编译并运行如下 Java 程序, 将输出 ( )。(选择一项)
- ```
public static void main(String[] args) {  
    try {  
        int num1 = 2;    int num2 = 0;  
        int result = num1 / num2;  
        System.out.println(result);  
        throw new NumberFormatException( );  
    } catch (ArrayIndexOutOfBoundsException e) {  
        System.out.print("1");  
    } catch (NumberFormatException e) {  
        System.out.print("2");  
    } catch (Exception e) {  
        System.out.print("3");  
    } finally {  
        System.out.print("4");  
    }  
    System.out.print("5");  
}
```
- A. 134  
B. 2345  
C. 1345  
D. 345
4. 阅读如下Java代码, 其中错误的行是 ( )。(选择二项)
- ```
public class Student {  
    private String stuId;  
    public void setStuId(String stuId) throw Exception { // 1  
        if (stuId.length() != 4) { // 2  
            throws new Exception("学号必须为4位!"); // 3  
        } else {  
            this.stuId = stuId; //4  
        }  
    }  
}
```
- A. 1  
B. 2  
C. 3  
D. 全部正确
5. 下面选项中有关 Java 异常处理模型的说法错误的是 ( )。(选择二项)
- A. 一个 try 块只能有一条 catch 语句

- B. 一个 try 块中可以不使用 catch 语句  
C. catch 块不能单独使用，必须始终与 try 块在一起  
D. finally 块可以单独使用，不是必须与 try 块在一起
6. 下面选项中属于运行时异常的是（ ）。(选择二项)  
A. Exception 和 SexException  
B. NullPointerException 和 InputMismatchException  
C. ArithmeticException 和 ArrayIndexOutOfBoundsException  
D. ClassNotFoundException 和 ClassCastException
7. 阅读如下 Java 代码，在控制台输入"-1",执行结果是（ ）。(选择一项)
- ```
public class Demo {  
    public static void main(String[] args) {  
        Scanner input = new Scanner(System.in);  
        System.out.print("请输入数字: ");  
        try {  
            int num = input.nextInt();  
            if (num < 1 || num > 4) {  
                throw new Exception("必须在1-4之间!");  
            }  
        } catch (InputMismatchException e) {  
            System.out.println("InputMismatchException");  
        } catch (Exception e) {  
            System.out.println(e.getMessage());  
        }  
    }  
}
```
- A. 输出: InputMismatchException  
B. 输出: 必须在1-4之间!  
C. 什么也没输出  
D. 编译错误

### 三、简答题

1. Error 和 Exception 的区别
2. Checked 异常和 Runtime 异常的区别。
3. Java 异常处理中，关键字 try、catch、finally、throw、throws 分别代表什么含义？
4. throws 和 throw 的区别

### 四、编码题

1. 编写程序接收用户输入分数信息，如果分数在 0—100 之间，输出成绩。如果成绩

不在该范围内，抛出异常信息，提示分数必须在 0—100 之间。

要求：使用自定义异常实现

2. 写一个方法 `void isTriangle(int a,int b,int c)`，判断三个参数是否能构成一个三角形，如果不能则抛出异常 `IllegalArgumentException`，显示异常信息 “a,b,c 不能构成三角形”，如果可以构成则显示三角形三个边长，在主方法中得到命令行输入的三个整数，调用此方法，并捕获异常。

## 五、可选题

1. 编写一个计算 `N` 个整数平均值的程序。程序应该提示用户输入 `N` 的值，如何必须输入所有 `N` 个数。如果用户输入的值是一个负数，则应该抛出一个异常并捕获，提示 “`N` 必须是正数或者 0”。并提示用户再次输入该数