实验课程名称：Java语言程序设计A

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **实验项目名称** | **实验6：异常处理** | | | **实验成绩** |  |
| **实 验 者** |  | **专业班级** | **信管1502** | **组 别** | **1人1组** |
| **同 组 者** | **无** | | | **开始日期** | **2018年5月15日** |
| 第一部分：实验预习报告（包括实验目的及意义，实验基本原理与方法，主要仪器设备及耗材，实验内容及要求，实验方案与技术路线）  一．实验目的及意义  掌握最基本的异常处理。  二．实验基本原理与方法  1．异常处理机制。  2．try-catch代码块的合理使用。  三．主要仪器设备及耗材  1．PC及其兼容机。  2．计算机操作系统。  3．程序编辑器EditPlus。  4．Java开发工具JDK。  四．实验内容及要求  某同学编写的一个程序如下：  import java.util.Scanner;  class ExceptionDemo{  public static void main(String[] args) {  Scanner scanner = new Scanner(System.in);  System.out.print("Enter a interger：");  int number = scanner.nextInt();  System.out.println("The number entered is：" + number);  }  }  该程序运行时，等待输入一个整数，并显示输出该整数。但如果你输入的不是整数（比如说浮点数），程序运行时系统将抛出异常并非正常终止。请你重写该程序，增加异常处理功能。  实验报告要求：实验6的实验报告，必须包含设计思路、代码、运行结果、问题及解决过程、综合分析、总结、体会。  五．实验方案及技术路线（设计思路、算法、注意事项）  技术路线：  这次实验具有修改性质，需要在程序中对风险代码进行异常处理。  具有风险的代码有两行：  **int** number = scanner.nextInt();  System.***out***.println("The number entered is:" + number);  于是这两行代码就丢入try-catch语句块中，异常类型选用所有异常的父类Exception接住。  第二部分：实验过程记录（可加页）（代码、运行结果、实验中出现的问题及解决过程）  一、代码  import java.util.\*;  public class ExceptionDemo  {  public static void main(String[] args)  {  Scanner scanner = new Scanner(System.in);  System.out.print("Enter a interger:");  try  {  int number = scanner.nextInt();  System.out.println("The number entered is:"+number);  }  catch (Exception e)  {  e.printStackTrace();  System.out.println("因为没有按整数格式输入，程序报错终止");  }  }  }    二、运行结果  Enter a interger:3.14159  Enter a interger:由于没有输入要求的整数，我报错终止了  java.util.InputMismatchException  at java.util.Scanner.throwFor(Scanner.java:864)  at java.util.Scanner.next(Scanner.java:1485)  at java.util.Scanner.nextInt(Scanner.java:2117)  at java.util.Scanner.nextInt(Scanner.java:2076)  at ExceptionDemo.main(file.java:10)  第三部分 结果与讨论（可加页）  一、实验结果分析  Java语言对错误有很好的处理机制，其公共父类Exception涵盖了所有的基本错误，当不清楚可能会出现什么错误的时候，用Exception当作父类，就可以找到最基本的错误类型。  二、小结、建议及体会  本次实验主要是对Java异常处理机制的学习，当我们在开发的过程中，需要避免高风险的操作，当我们把代码部署到服务器上的时候，对一些小错误都需要直接跳过，这时候Java的错误处理机制就十分有用。  第四部分 实验评分标准（教师自行设计）及成绩   |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | 观测点 | 考核目标 | 权重 | 得分 | | 实验预习 | 程序设计方案、技术路线（设计思路、算法、注意事项）科学、合理、可行 | 实验方案设计能力 | 20% |  | | 实验过程 | 1. 按时参加实验，不迟到，不早退 2. 源代码正确、完整、可读性强、编码规范，运行结果正确，实验中出现的问题得到解决 | 学生的实验态度、UML面向对象建模与设计能力、分析并解决问题的专业素养及编程技能、良好的编程风格与习惯 | 50% |  | | 结果分析 | 实验结果分析到位、总结完善、体会深刻 | 学生对实验数据处理与分析的能力；对专业知识的综合应用能力；事实求实的精神 | 30% |  | | 该项实验报告最终得分 | | |  | |   教师签名： | | | | | |