**教 学 设 计**

**科目: Java编程技术基础**

**教师:**

**部门:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **课题** | 项目11 实现除法计算  任务 实现一个除法计算器 | | | **备课时间** |  | |
| **课型** | 理实一体 | **课时** | 4课时 | **授课班级** | 2020级 | |
| **教学目标** | 1. 理解异常的概念和用途；  2. 掌握使用try-catch-finally语句结构；  3. 理解finally语句的用法 | | | | | |
| **教材**  **分析** | **重点** | 掌握使用try-catch-finally语句结构 | | | | |
| **难点** | 理解finally语句的用法 | | | | |
| **教具** | PPT、课本 | | | | | |
| **教法** | 案例分析、合作学习、电脑模拟等方法 | | | | | |
| **学法** | 学生在教师的指导下课前复习课本的内容，课后及时巩固上课所学的知识，完成上课布置的作业 | | | | | |
| **教 学 过 程** | | | | | | |
| **教学环节** | **教 学 内 容** | | | | | **教师活动、学生活动** |
| **导入新课** | 编写一个除法计算器，程序要求在出现除数为零和除数、被除数中有一个不是数字的情况时进行相应的处理。当调用存放在数组中的计算结果，数组有可能产生索引越界，对这种情况进行捕捉和处理。其运行结果如下：  请输入除数:0  请输入被除数:10  异常2:除数不能为零!  最后要执行的内容5! | | | | | 先让学生自己思考，小组讨论，然后教师再讲解，采用启发式教学 |
| **教学过程** | 【知识准备】  1.1 异常概念及处理机制  在生活中，发生异常我们懂得如何处理，那么在Java程序中，又是如何处理异常的呢？异常处理就像我们平时可能会遇到的意外情况，预想好了一些处理的办法。也就是说，在程序执行代码的时候，万一发生了异常，程序会按照预定的处理方法对异常进行异常处理，异常处理完毕之后，程序继续运行。但异常处理的机制需要落实到具体的处理代码上，Java的异常处理方式有两种：“捕捉异常”的异常处理方式对受检异常、运行时异常均适用，捕捉异常处理语句是try-catch；“上报异常”是当前的代码不能处理产生的异常，将异常交给调用它的上级进行处理的异常处理的方法。  1.2 异常的分类  Java异常分为系统异常和自定义异常。  1. 系统异常  在Java的系统异常中，throwable是它们的父类，其子类有Error和Exception。前者表示程序运行时发生较少内部的异常，程序员无法处理。后者是程序运行和环境产生的异常，可以捕获和处理。在开发中遇到的异常绝大部分是Exception异常。  Java中几个常见的异常如下：  1）ArithmeticException当出现异常算术条件时产生；  2）NullPointerException当应用程序企图使用需要的对象处为空时产生；  3）ArrayIndexOutOfBoundsException 数组下标越界时产生；  4）ArrayStoreException 当程序试图存储数组中错误的类型数据时产生；  5）FileNotFoundException 试图访问的文件不存在时产生；  6）IOException 由于一般I/O故障而引起的，如读文件故障；  7）NumberFormatException 当把字符串转换为数值型数据失败时产生；  8）OutOfMemoryException 内存不足时产生；  9）StackOverflowException 当系统的堆栈空间用完时产生。  2. 用户自定义异常  Java内置的异常能够处理大多数常见的运行时错误，但也可以自己定义，自定义异常来处理系统异常无法捕获的异常。  1.3 异常的捕获与处理  捕获处理方式主要是使用try...catch，将可能出现的错误用try语句包绕，当try中的语句出现异常时，就停止当前程序的执行，转入到catch中执行语句处理异常。也就是说try语句用来发现异常，而catch语句用来处理异常。  1. catch语句的结构格式      2. 多个catch语句  当某个程序块可能出现错个异常时，可以用多个catch语句，每个catch语句捕获一种异常，捕获异常的顺序和catch语句的顺序有关，当捕获到一个异常时，剩下的catch语句就不再进行匹配。  3. finally语句  finally语句是为异常处理事件提供的一个清理机制，一般是用来关闭文件或释放其他系统资源。作为try-catch-finally结构的一部分，可以没有finally语句，如果存在finally语句，不论try块中是否发生异常，是否执行过catch语句，都执行finally语句。  【实践拓展】  实践操作，除法计算器程序设计。提示：  1）打开Eclipse，创建一个类；  2）在类main方法中输入两个数相除；  3）给输入的两个数以及两个数相除会产生异常，对这段代码进行异常处理；  4）编写测试类，运行程序 | | | | | 提问查看当堂掌握情况  对学生操作进行点评 |
| **作业布置** | 编写一个类ExceptionTest，在main方法中使用try、catch、finally关键字实现：在try块中，编写被两个数相除操作，其中除法的两个操作数要求运行时用户输入；  在catch块中，捕获被0除所产生的异常，并且打印异常信息；  在finally块中，打印一条语句 | | | | | |
| **板书设计** | 项目11 实现除法计算  任务 实现一个除法计算器  1.1 异常概念及处理机制  1.2 异常的分类  1.3 异常的捕获与处理 | | | | | |
| **教学反思** |  | | | | | |