

河南中医药大学课堂教学设计

授课章节	第 05 章：程序调试		授课学时	2 学时
所属课程	计算机程序设计	授课年级	2023 级	
设计者	阮晓龙、黄子杰	授课专业	智能医学工程专业	
1.教学目标：含知识、技能（能力）、学习态度与价值观（情感）目标				
知识目标： 1. 理解什么是异常； 2. 掌握调试程序的方法。 能力目标： 1. 逻辑推导能力； 2. 语言表达能力； 3. 复杂问题简化分析能力。 素质目标： 1. 提升自主学习能力； 2. 激发创新思维与问题解决能力； 3. 强调团队合作与沟通能力； 4. 培养逻辑思维与清晰表达能力。 思政目标： 1. 弘扬科学精神，推动社会进步：通过学习 Java 编程语言，激发学生对科学技术的兴趣和追求，树立终身学习的理念，鼓励学生以科学的态度探索技术问题，积极参与到科技创新中去； 2. 增强国家认同感，致力于自主创新：培养学生为国家自主创新和技术突破贡献力量的责任意识。				
2.教学内容：依据教学大纲；含教学重点难点				
教学重点： 1. 调试程序的几步曲； 2. 异常概述； 3. 使用 try 和 catch 捕获异常； 4. getMessage 和 printStackTrace 方法。 教学难点： 1. 使用 try 和 catch 捕获异常； 2. getMessage 和 printStackTrace 方法。				

课堂教学内容：

1、异常概述（15 分钟）

异常（Exception）是指程序在运行过程中发生的不正常情况，通常会导致程序的中断。Java 使用异常处理机制来捕获和处理这些异常，以便程序能够继续运行或安全地终止。Java 中的异常类主要分为两类：受检异常（Checked Exception）和非受检异常（Unchecked Exception）。

受检异常、非受检异常。

2、使用 try 和 catch 捕获异常（15 分钟）

try 和 catch 是 Java 中用于捕获和处理异常的关键字。try 块包含可能抛出异常的代码，而 catch 块用于捕获和处理该异常。

包括：try-catch 语句的作用、try-catch 语句的使用场景。

3、使用 throw 和 throws 引发异常（10 分钟）

throw 关键字用于显式抛出一个异常，通常在方法内部使用；而 throws 关键字用于在方法声明时指明该方法可能抛出的异常。

包括：显式抛出异常、throw 的使用场景、throw 的注意事项、throws 的注意事项、声明方法可能抛出的异常、throws 的使用场景。

4、finally 关键字（5 分钟）

finally 块是与 try-catch 结构一起使用的，它包含的代码在异常处理后总是执行，无论是否抛出异常。finally 通常用于释放资源，如关闭文件或网络连接。

finally 块的作用：确保代码总是执行、finally 块的执行时机、finally 块的使用场景、finally 块的注意事项。

5、getMessage 和 printStackTrace 方法（15 分钟）

getMessage() 方法用于获取异常的简要描述信息，而 printStackTrace() 方法用于打印异常的堆栈跟踪信息，这有助于调试程序。

getMessage() 方法包括：获取异常的简要描述、getMessage 的使用场景、getMessage 的注意事项。

printStackTrace() 方法包括：打印异常的堆栈跟踪信息、printStackTrace 的使用场景、printStackTrace 的注意事项。

6、多重 catch（10 分钟）

在 try 块后可以跟随多个 catch 块，以捕获不同类型的异常。每个 catch 块处理一种特定的异常类型。

包括：多重 catch 的语法、多重 catch 的使用场景、多重 catch 的注意事项。

多重 catch 的优势：优化异常处理逻辑、提高代码的可读性、多重 catch 的限制。

7、自定义异常类（10 分钟）

Java 允许开发者创建自定义异常类，以满足特定的异常处理需求。自定义异常类通常继承自 Exception 或 RuntimeException 类。

包括：继承 Exception 或 RuntimeException、自定义异常类的构造方法、自定义异常类的使用场景。

自定义异常类的优势：提高异常处理的精确度、增强代码的可维护性、自定义异常类的注意事项。

课堂教学内容：

3.思政知识点：

课程思政案例	思政点映射
在量子计算、人工智能等前沿技术的研发中，研究人员通过调试和优化程序，不断突破现有技术的瓶颈，推动技术的进步。每一次调试过程中的反馈，都是一次技术突破的机会，正如我国在科技创新道路上的不断努力和坚持，面对挑战不退缩，寻找创新的路径。	科技创新道路不断努力和坚持、自主创新、精益求精

4.学情分析及教学预测：	
学生的知识基础： 1. 编程基础； 学生的认知特点： 1. 对程序调试没有清晰的理解。 学生的学习风格： 1. 不熟悉 Java 程序的基本知识，但学习的热情和积极性较高； 2. 不理解程序调试的基本概念，对计算机程序设计课程比较期待。 教学预测： 1. 通过案例式教学和探究式教学等方法，培养学生的创新意识和思维能力； 2. 学生的学习兴趣 and 动机提升：通过引导学生进行实际的操作和互动交流； 3. 学生的合作与沟通能力培养：在课程中鼓励学生进行小组合作，分享经验和解决问题。	
5.教学策略与方法：	
教学策略： 1. 通过多媒体演示文稿进行讲解，并结合板书进行关键难点的介绍和原理过程的讲解； 2. 课后留练习题目或作业，引导学生对课程内容进一步巩固和复习。 教学方法： 1. 通过课前预习，让学生对相关基础知识及概念有基本的了解。 2. 理论课通过讲解、与学生互动了解学生知识掌握情况，对学生较为薄弱的环节进一步强化介绍。	
6.板书设计：	
① 黑板（白板）设计： 异常 Try getMessage 和 printStackTrace Catch	② 现代信息媒体设计： 使用多媒体教学课件开展。
7.教学互动环节设计：	
课堂上的提问和互动交流： 1. 问题一：什么是异常？Java 中的异常如何分类？ 2. 问题二：try 和 catch 的作用是什么？ 3. 问题三：如何创建自定义异常类？为什么需要自定义异常？	

8.学习资源，课外自主学习设计：

自建学习资源：

1. 课程学习平台：<https://webdev.hactcm.edu.cn/program>
2. 课堂派：<https://www.ketangpai.com>

网络学习资源：

1. Java 官网：<https://www.java.com/zh-CN/>

9.教学测量与评价：

课堂教学测量评价：

1. 课堂测试：使用课堂派开展阶段性测试。
2. 课堂提问：通过提问及利用课堂派与学生互动，及时了解学生知识点掌握情况。

课外学习测量评价：

1. 课前预习：通过课程学习平台开展预习。
2. 课后作业：通过课堂派布置作业，每个章节 1 个作业，内容见课堂派。

10.教学反思与改进：（授课后教师总结）

11.授课教师认为尚未包含在内的设计内容：（授课后教师总结）