

企业级卓越人才培养（信息类专业集群）
解决方案“十三五”规划教材

Java Web 应用程序开发

天津滨海迅腾科技集团有限公司 主编

南开大学出版社

企业级卓越人才培养（信息类专业集群）解决方案“十三五”规划教材

Java Web 应用程序开发

天津滨海迅腾科技集团有限公司 主编

南开大学出版社

天 津

图书在版编目 (CIP) 数据

Java web 应用程序开发 / 天津滨海迅腾科技集团有限公司主编. — 天津: 南开大学出版社, 2017.5

ISBN 978-7-310-05326-1

I. ①J… II. ①天… III. ①JAVA语言—程序设计
IV. ①TP312.8

中国版本图书馆 CIP 数据核字 (2017) 第 012360 号

版权所有 侵权必究

南开大学出版社出版发行

出版人: 刘立松

地址: 天津市南开区卫津路 94 号 邮政编码: 300071

营销部电话: (022)23508339 23500755

营销部传真: (022)23508542 邮购部电话: (022)23502200

*

唐山新苑印务有限公司印刷

全国各地新华书店经销

*

2017 年 5 月第 1 版 2017 年 5 月第 1 次印刷

260×185 毫米 16 开本 18.5 印张 463 千字

定价: 55.00 元

如遇图书印装质量问题, 请与本社营销部联系调换, 电话: (022)23507125

企业级卓越人才培养(信息类专业集群)解决方案

“十三五”规划教材编写委员会

顾 问：朱耀庭 南开大学
邓 蓓 天津中德应用技术大学
张景强 天津职业大学
郭红旗 天津软件行业协会
周 鹏 天津市工业和信息化委员会教育中心
邵荣强 天津滨海迅腾科技集团有限公司

主 任：王新强 天津中德应用技术大学

副主任：杜树宇 山东铝业职业学院
陈章侠 德州职业技术学院
郭长庚 许昌职业技术学院
周仲文 四川华新现代职业学院
宋国庆 天津电子信息职业技术学院
刘 胜 天津城市职业学院
郭思延 山西旅游职业学院
刘效东 山东轻工职业学院
孙光明 河北交通职业技术学院
廉新宇 唐山工业职业技术学院
张 燕 南开大学出版社有限公司

编 者：刘品洁 郭思延 贾月辉 徐云燕 孙光明

企业级卓越人才培养(信息类专业集群)

解决方案简介

企业级卓越人才培养(信息类专业集群)解决方案(以下简称“解决方案”)是面向我国职业教育量身定制的应用型、技术技能型人才培养解决方案,以天津滨海迅腾科技集团技术研发为依托,联合国内职业教育领域相关行业、企业、职业院校共同研究与实践研发的科研成果。本解决方案坚持“创新产教融合协同育人,推进校企合作模式改革”的宗旨,消化吸收德国“双元制”应用型人才培养模式,深入践行“基于工作过程”的技术技能型人才培养,设立工程实践创新培养的企业化培养解决方案。在服务国家战略、京津冀教育协同发展、中国制造 2025(工业信息化)等领域培养不同层次及领域的信息化人才。为推进我国教育现代化发挥应有的作用。

该解决方案由“初、中、高级工程师”三个阶段构成,集技能型人才培养方案、专业教程、课程标准、数字资源包(标准课程包、企业项目包)、考评体系、认证体系、教学管理体系、就业管理体系等于一体。采用校企融合、产学研融合、师资融合的模式在高校内共建互联网学院、软件学院、工程师培养基地的方式,开展“卓越工程师培养计划”,开设系列“卓越工程师班”,“将企业人才需求标准、企业工作流程、企业研发项目、企业考评体系、企业一线工程师、准职业人才培养体系、企业管理体系引进课堂”,充分发挥校企双方特长,推动校企、校际合作,促进区域优质资源共建共享,实现卓越人才培养目标,达到企业人才培养及招录的标准。本解决方案已在全国近二十所高校开始实施,目前已形成企业、高校、学生三方共赢格局。未来五年将努力实现年培养能力达到万人的目标。

天津滨海迅腾科技集团是以 IT 产业为主导的高科技企业集团,总部设立在北方经济中心——天津,子公司和分支机构遍布全国近 20 个省市,集团旗下的迅腾国际、迅腾科技、迅腾网络、迅腾生物、迅腾日化分属于 IT 教育、软件研发、互联网服务、生物科技、快速消费品五大产业模块,形成了以科技为原动力的现代科技服务产业链。集团先后荣获“全国双爱双评先进单位”“天津市五一劳动奖状”“天津市政府授予 AAA 级和谐企业”“天津市文明单位”“高新技术企业”“骨干科技企业”等近百项殊荣。集团多年中自主研发天津市科技成果 2 项,具备自主知识产权的开发项目数十余项。现为国家工业和信息化部人才交流中心“全国信息化工程师”项目联合认证单位。

前 言

随着互联网的不断发展, Web 项目需求不断扩大, Java 工程师的需求也越来越大。当前大部分网站开发使用的技术为 Java Web。Java Web 作为目前最流行的动态网页开发技术之一, 吸引了众多软件开发人员的眼球。

本书的知识点由易到难, 以“理论+实践”的形式展现给读者, 读完本书后, 读者对 Java Web 的理念和思想可以初步了解, 并且具备创建类似各种电商平台项目的能力。

本书共十章, 分别介绍了 Java Web 的异常处理原则、I/O 流对程序的输入和输出的处理、通过 JDBC 对数据库的访问、项目开发过程中 Tomcat 服务器的作用、如何使用 JSP 技术创建 Web 页面、如何在 JSP 中使用 Java 代码、怎么实现 Web 应用等。

通过本书的学习, 读者可以了解异常处理机制、I/O 流之间数据的传递, 掌握 JDBC 对数据库的访问, Tomcat 服务器的搭建, JSP 内置函数等知识, 通过这些知识的学习, 读者能够自己创建类似于各种电商平台的项目, 进而适应市场的需求。

本书每个章节都按照 Java Web 知识体系, 循序渐进的讲解。都设有学习目标、课前准备、本章简介、具体知识点讲解、小结、英语角、作业、思考题、学员回顾内容等模块。此结构条理清晰、内容详细, 将相关知识、技能、最准确的信息传递给读者, 不仅有益于巩固掌握的知识, 还能提高实践能力。

本书由刘品洁、郭思延主编, 贾月辉、徐云燕、孙光明参与编写, 由刘品洁、郭思延负责全面内容的规划、编排。具体分工如下: 第一、二章由贾月辉编写; 第三、四章由郭思延编写; 第五、六、七章由刘品洁编写; 第八、九章由徐云燕编写; 第十章由孙光明编写。

本书理论内容简明扼要、通俗易懂、即学即用; 实例操作讲解细致, 步骤清晰。在本书中, 操作步骤后有相对应的效果图, 便于读者直观、清晰地看到操作效果, 牢记书中的操作步骤, 更重要的是与实际项目相结合, 提高读者的综合能力。

目 录

理论部分

第1章 异常	3
1.1 Java 异常处理机制概述	3
1.2 运用 Java 异常处理机制	6
1.3 异常处理原则	20
1.4 小结	23
1.5 英语角	23
1.6 作业	23
1.7 思考题	23
1.8 学员回顾内容	24
第2章 I/O 流	25
2.1 字节流	25
2.2 字符流	33
2.3 特殊的 I/O 流	38
2.4 小结	44
2.5 英语角	45
2.6 作业	45
2.7 思考题	45
2.8 学员回顾内容	45
第3章 JDBC(一)	46
3.1 JDBC	46
3.2 基本数据库访问	48
3.3 特殊处理	56
3.4 小结	61
3.5 英语角	61
3.6 作业	61
3.7 思考题	62
3.8 学员回顾内容	62

第4章 JDBC(二)	63
4.1 高级数据库访问	63
4.2 事务	71
4.3 封装数据访问	76
4.4 小结	83
4.5 英语角	83
4.6 作业	83
4.7 思考题	83
4.8 学员回顾内容	84
第5章 Web 运行模式:Tomcat	85
5.1 程序网络计算模式	85
5.2 B/S 模式技术介绍	87
5.3 JSP 运行原理	90
5.4 Web 服务器	92
5.5 Tomcat 样例程序	94
5.6 部署 JSP 文件	96
5.7 小结	97
5.8 作业	97
5.9 思考题	98
5.10 学员回顾内容	98
第6章 JSP(一)	99
6.1 剖析一个 JSP 页面	99
6.2 脚本元素	100
6.3 JSP 指令	104
6.4 实例	111
6.5 小结	117
6.6 英语角	117
6.7 作业	118
6.8 思考题	118
6.9 学员回顾内容	118
第7章 JSP(二)	119
7.1 内置对象	119
7.2 out 对象	120
7.3 request 对象	122
7.4 response 对象	130

7.5 实例	132
7.6 小结	139
7.7 作业	139
7.8 思考题	140
7.9 学员回顾内容	140
第8章 JSP(三)	141
8.1 session 对象	141
8.2 application 对象	148
8.3 pageContext 对象	150
8.4 cookie	153
8.5 小结	158
8.6 英语角	159
8.7 作业	159
8.8 思考题	159
8.9 学员回顾内容	159
第9章 JSP 标准动作	160
9.1 概述	160
9.2 文件包含动作	161
9.3 <jsp:useBean> 动作	162
9.4 <jsp:setProperty> 动作	165
9.5 <jsp:getProperty> 动作	169
9.6 请求重定向动作	171
9.7 实例	172
9.8 小结	178
9.9 作业	178
9.10 思考题	179
9.11 学员回顾内容	179
第10章 Java 实用技术	180
10.1 在 JSP 中上传文件	180
10.2 用 POI 与 Excel 交互	187
10.3 小结	196
10.4 英语角	196
10.5 作业	196
10.6 思考题	197
10.7 学员回顾内容	197

上机部分

第1章 异常	201
1.1 指导(1小时10分钟)	201
1.2 练习(50分钟)	205
1.3 作业	205
第2章 I/O 流	206
2.1 指导(1小时10分钟)	206
2.2 练习(50分钟)	212
2.3 作业	212
第3章 JDBC(一)	213
3.1 指导(1小时10分钟)	213
3.2 练习(50分钟)	217
3.3 作业	217
第4章 JDBC(二)	218
4.1 指导(1小时10分钟)	218
4.2 练习(50分钟)	223
4.3 作业	223
第5章 Web 运行模式:Tomcat	224
5.1 指导(1小时10分钟)	224
5.2 练习(50分钟)	227
5.3 作业	228
第6章 JSP(一)	230
6.1 指导(1小时10分钟)	230
6.2 page 指令	231
6.3 练习(50分钟)	241
6.4 作业	241
第7章 JSP(二)	242
7.1 指导(1小时10分钟)	242
7.2 练习(50分钟)	252
7.3 作业	252

第 8 章 JSP(三)	253
8.1 指导(1 小时 10 分钟)	253
8.2 练习(50 分钟)	261
8.3 作业	261
第 9 章 JSP 标准动作	262
9.1 指导(1 小时 10 分钟)	262
9.2 练习(50 分钟)	272
9.3 作业	272
第 10 章 Java 实用技术	273
10.1 指导(1 小时 10 分钟)	273
10.2 练习(50 分钟)	281
10.3 作业	281

理论部分

第 1 章 异常

学习目标

- ◇ 了解使用异常的原因。
- ◇ 理解 Java 的异常层次结构。
- ◇ 掌握 try、throw 和 catch 块检测、指出和处理异常。
- ◇ 掌握 finally 子句释放资源。
- ◇ 掌握声明新的异常类。

课前准备

掌握面向对象的基本概念，熟悉类的结构。

本章介绍

本章将介绍异常处理。异常是指程序运行期间出现的问题。这种问题不是严重的计算机硬件问题，也不是在程序编译时出现的问题，更不是因程序设计不良所导致的结果与预期不一致的问题。它是指：诸如用户的错误输入，或碰巧的计算结果导致一些程序运行时发生了致命的违背常理的错误，如“除零”。这些错误和软件本身的设计优劣无关，但是无法完全避免。它一旦发生会导致程序运行的停止，而不是以一种用户可控制的方式将问题通知用户。所以必须有一个机制来避免程序运行停止的现象，故引入异常。

Java 语言提供了一套完善的异常处理机制。正确运用这套机制，有助于提高程序的健壮性。所谓程序的健壮性，就是能采取周到的解决措施。而不健壮的程序则没有事先充分预计到可能出现的异常，或者没有提供强有力的异常解决措施，导致程序在运行时经常莫名其妙的终止，或者返回错误的运行结果，而且难以检测出现异常的原因。

本章首先概述异常处理的概念，然后举例说明基本的异常处理技术。我们将通过一个例子来说明这些技术。

1.1 Java 异常处理机制概述

要在程序中处理异常，主要考虑两个问题：

(1) 如何表示异常情况？(2) 如何控制处理异常的流程？

对于如何表示异常, Java 相对于传统的异常处理而言采用了面向对象的思想, 具有更好的可维护性。Java 异常处理机制具有以下优点:

- 把各种不同类型的异常情况进行分类, 用 Java 类来表示异常情况, 这种类被称为异常类。异常类使得处理异常具备可扩展和可重用的优势。
- 异常流程的代码和正常流程的代码分离, 提高了程序的可读性, 简化了程序的结构。
- 可以灵活地处理异常, 如果当前方法有能力处理异常, 就捕获并处理它, 否则只需要抛出异常, 由上一层来处理。

Java 中的一切都是对象。因此程序员可以创建异常类的层次结构。图 1-1 显示了 `Throwable` 类 (`Object` 类的子类) 的继承层次结构的一小部分, 该类是所有异常的超类。只有 `Throwable` 对象才可以用于异常处理机制。 `Throwable` 类有两个子类: `Exception` 和 `Error`。 `Exception` 类及其子类 (例如 `RuntimeException` 和 `IOException` 类, 每个类均位于 `java.lang` 包中) 代表 Java 程序中可能发生的异常情况, 并且应用程序可以捕获这些异常情况。 `Error` 类及其子类 (例如 `OutOfMemoryError` 类) 代表 Java 运行时系统中可能发生的异常情况, 但通常应用程序不应该捕获这些异常。

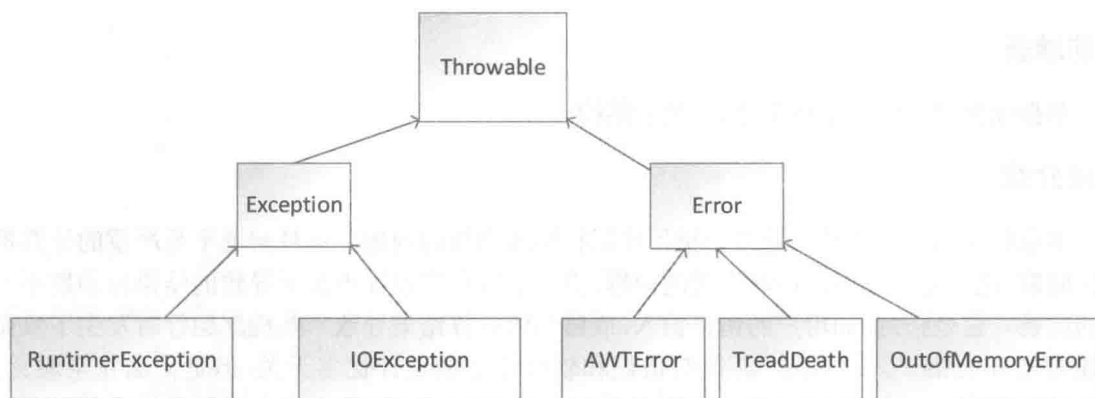


图 1-1 `Throwable` 类继承层次结构图

所有的异常都起源于 `Throwable`, 但是 `Throwable` 分为两个分支: `Error` 和 `Exception`。

`Error` 分支用于 Java 运行时系统内部错误和资源耗尽错误。你无法抛出一个这种类型的错误。当这种内部错误发生时, 你除了通知用户并试图终止程序以外基本无能为力。

`Exception` 也分为两个分支: 派生自 `RuntimeException` 的异常和普通异常。对这两类区分的原则一般为: 产生一个 `RuntimeException` 的原因是编程错误。而任何其他异常出现是因为程序碰上一个意外情况, 例如一个 I/O 错误。

见表 1-1 中所列几种情况会导致派生自 `RuntimeException` 和非派生自 `RuntimeException` 的异常。

表 1-1 两种异常发生情况

RuntimeException	非 RuntimeException
一个错误的类型转换	试图读取一个文件结尾后面的数据
一个越界数组访问	试图打开一个错误的 URL
一个空指针访问	试图根据一个不代表任何存在类的字符串来找到一个 Class 类

记住一个原则：“如果它是一个 RuntimeException，那么这是你的错”。你可以通过判断数组下标是否超过数组边界来避免出现数组越界异常 `ArrayIndexOutOfBoundsException`。如果在使用一个变量之前首先判断它是否为 null，则空指针异常 `NullPointerException` 就可以避免。常见的异常见表 1-2。

表 1-2 常见异常

异常类	说明
<code>Exception</code>	异常层次结构的根类
<code>ArithmeticException</code>	算术错误情形，如以零作除数
<code>IllegalArgumentException</code>	方法接收到非法参数
<code>ArrayIndexOutOfBoundsException</code>	数组大小小于或大于实际的数组大小
<code>NullPointerException</code>	尝试访问 null 对象成员
<code>ClassNotFoundException</code>	不能加载所需的类
<code>NumberFormatException</code>	数字转化格式异常，比如字符串到 float 型数字的转换无效
<code>IOException</code>	输入输出
<code>FileNotFoundException</code>	找不到文件
<code>EOFException</code>	文件结束

我们从上面的描述中清楚了怎么样去表示异常，现在来看看如何去处理异常：Java 采用抓抛模型来完成对异常的处理。即系统在执行 Java 程序，比如执行一个方法，遇到了一个错误但无法处理，这时，该方法会抛出一个异常类的对象。如果存在一个异常处理程序，则该异常处理程序会捕获这一异常，并加以处理，然后继续执行程序。但是如果没有异常处理程序，则异常最终会被 Java 虚拟机捕获，并由 Java 虚拟机提示用户，终止程序的运行。如图 1-2 所示。

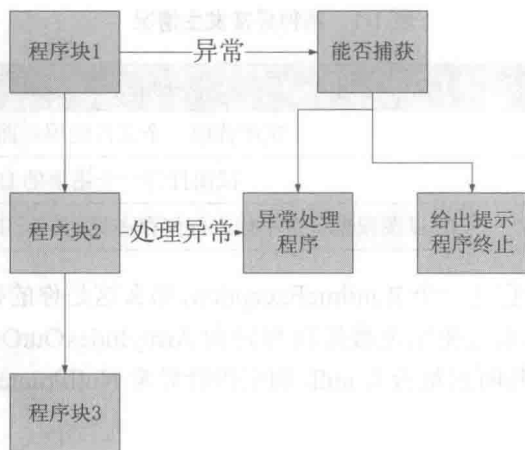


图 1-2 Java 异常处理流程图

1.2 运用 Java 异常处理机制

Java 为异常处理提供了 `try...catch...finally` 语句来支持异常处理。`try` 语句有关键字 `try` 和随后用于封装 `try` 块的大括号 `{}` 构成。`try` 块包含可能产生异常的语句，并且在 `try` 块后至少紧跟一条 `catch` 块或一条 `finally` 块。每个 `catch` 块在圆括号中指定一个异常类参数，用于表示该类 `catch` 块能够处理的异常类型。在最后一个 `catch` 块后是一个可选的 `finally` 块，无论异常是否发生，都会执行该块所提供的代码。

1.2.1 `try...catch` 语句：捕获异常

在 Java 语言中，用 `try...catch` 语言来捕获异常。格式如下：

```
try{
    可能会出现异常情况的代码
}catch(异常类型一 e1){
    处理出现的异常一类型的异常
}catch(异常类型二 e2){
    处理出现的异常二类型的异常
}
```

`try` 块中包含可能会出现异常情况的代码，随后紧跟 `catch` 块，并且两个块之间不允许出现其他语句。`try` 块后面可以跟多个 `catch` 块以处理不同类型的异常，在括号中的异常类型是 Java 中的系统定义异常类或自定义异常类。

程序发生异常的地方称为抛出点。如果某个 `try` 块发生异常，就会生成一个描述所发生异常的对象，然后抛出该对象。这个 `try` 块也立即终止执行，程序控制转移到该 `try` 块后面的第一