

高等教育质量工程信息技术系列示范教材

新概念

Java程序设计 大学教程

张基温 编著

清华大学出版社

高等教育质量工程信息技术系列示范教材

新概念 Java程序设计大学教程

张基温 编著

清华大学出版社
北京

内 容 简 介

本书结构新颖、概念清晰、面向应用,渗透了作者提出的“以思路训练为主线”的程序设计教育理念;书中例题丰富、习题题型多、覆盖面宽,为读者建立了一个全新概念的立体学习环境,并引导学生进行探索式学习。全书共分为4篇:第1篇为面向对象启步,用5个例子引导读者逐步建立面向对象的思维方式和基本的设计能力,将Java基本语法贯穿其中;第2篇为面向抽象编程,主要介绍抽象类、接口和设计模式;第3篇为基于API开发,包括多线程编程、图形用户界面、网络编程、JavaBean、持久化技术;第4篇为面向Web开发,介绍Servlet、JSP、Struts和Spring,并引出软件架构的概念。通过这4篇,可以达到夯实基础、面向应用、领略全貌的教学效果。

本书适合各种层次的计算机及相关专业作为程序设计教学的教材,也适合培训和自学。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

新概念Java程序设计大学教程 / 张基温编著. --北京:清华大学出版社, 2013.8

高等教育质量工程信息技术系列示范教材

ISBN 978-7-302-32916-9

I. ①新… II. ①张… III. ①Java语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆CIP数据核字(2013)第136268号

责任编辑:白立军 顾冰

封面设计:常雪影

责任校对:焦丽丽

责任印制:刘海龙

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦A座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载: <http://www.tup.com.cn>, 010-62795954

印 装 者:清华大学印刷厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:30 字 数:706千字

版 次:2013年8月第1版 印 次:2013年8月第1次印刷

印 数:1~2000

定 价:49.00元

产品编号:052835-01

前 言

这是我第 3 次写 Java 教材了。第 1 次是 2001 年,应清华大学出版社之约,写了《Java 程序开发教程》,并配有一本习题解答。第 2 次是 2010 年,应中国电力出版社之约,写了《新概念 Java 教程》。这一次又是应清华大学出版社之约,写了《新概念 Java 程序设计大学教程》。

近几年,我所写的程序设计教材都冠以“新概念”。所谓“新概念”,并非我能在一种程序设计语言中添加什么概念,而是企图建立一种新的程序设计教学的模式,来改变一下程序设计课程教学效率不高,甚至不成功的现状。

(一)

在多年的程序设计课程教学实践以及广泛地与国内同行的交流中,自己感到或听到的,都是认为程序设计课程不太成功的说法。

有的学校的系主任(院长)抱怨,学生学习了几门程序设计课程,可是到了课程设计、毕业设计,遇到问题还是下不了手。

多数程序设计课程老师都认为,教了 C++、Java,可是学生遇到的问题,写出来的代码还是面向过程的。

许多学生告诉我,不知道如何测试程序,甚至学习了软件工程以后,设计了程序也还是简单地试通一下,根本没有规范化测试的习惯。

企业界的朋友告诉我,企业界已经在关注设计模式和软件架构,而新入职的软件专业的大学生对此还完全没有概念。

.....

所有这些问题,都落到一点上:程序设计课程应当教什么?应当如何教?

(二)

教材是教学的剧本。程序设计课程的改革首先应当从教材改革开始。创新是发展的动力,教材的改革要求教材有所创新。

计算机程序设计教材的创新,首先要改变程序设计教材基于语法体系的结构。说到底,语法体系的程序设计教材都是程序设计语言手册的翻版。这种语法体系造就了重语法教学、轻思维训练的教学模式,是学习了程序设计课程却不会编写程序的祸根。

新概念系列力图在这个方面创出一条新路。本书没有沿袭 Java 教材从数据类型、控制结构的思路,而是在第 1 篇直接用 6 个实例按照“定义类—定义引用—创建对象—操作对象”的过程进行面向对象思维的训练,形成面向对象的思维主线,把数据类型、控制结构等语法嵌于其中。同时介绍基于组件的测试方法。

不了解设计模式,就没有掌握面向对象的真谛。本书第 2 篇以面向抽象为题,首先介绍抽象类和接口,然后引出面向对象的几个基本原则,接着通过 3 个例子引出 GoF 设计模式,

最后介绍反射、配置文件和程序打包发布。这一篇不仅让学生了解了面向对象的真正意义,也更贴近了应用实际。

第3篇通过多线程、图形用户界面、网络编程、JavaBean和持久化,加深对于API意义和应用的理解决,为开发应用程序奠定基础。

第4篇介绍Java Web开发和软件架构。

这4篇,有详有略。详者为夯实Java开发的坚实基础,并学到实用的本领;略者为读者了解Java技术的全貌,以便将来确定在何处突破。这4篇,也形成4个学习层次,便于有关教学单位根据教学对象和目的进行取舍。

(三)

我国著名教育家陶行知说道:“行动是老子,知识是儿子,创造是孙子”,并倡导“知行合一”。世界上第一所完全为发展现代设计教育而建立的学院——包豪斯(Bauhaus)的名言:“干中学(learning from doing)”已经成为现代教育的重要思想。所有这些都表明了实践在学习中的重要性。程序设计更是这样,仅仅学习了一些程序设计语言的语法,仅仅了解了一些程序设计的方法,仅仅有了“知”,而不一定“会”。要想会,就要实践。

由J. Piaget、O. Kernberg、R. J. Sternberg、D. Katz、Vogotsky等创建的建构主义(constructivism)学习理论认为,知识不是通过教师传授得到,而是学习者在一定的情境即社会文化背景下借助其他人(包括教师和学习伙伴)的帮助,利用必要的学习资料,通过意义建构的方式而获得的。在信息时代,人们获得知识的途径发生了根本性的变化,教师不再是单一的“传道、授业、解惑”者,帮助学习者构建一个良好的学习环境也成为其一项重要职责。当然,这也是现代教材的责任。本书充分考虑了这些问题。

本书的每一单元后面都安排了概念辨析、代码分析、开发实践和探索深究4种自测和训练实践环节,为学习者搭建了起一个立体化的实践环境。

1. 概念辨析

概念辨析主要提供选择和判断两类自测题目,帮助学习者理解本节学习过的有关概念,把当前学习内容所反映的事物尽量和自己已经知道的事物相联系,并认真思考这种联系,通过“自我协商”与“相互协商”,形成新知识的同化与顺应。对于这种类型的习题,读者应当按照如下顺序完成:

- (1) 先给出自己的判断。
- (2) 设计一个小的程序,验证自己的判断。
- (3) 结合自己的判断对验证结果进行分析,说明原因。

2. 代码分析

代码阅读是程序设计者的基本能力之一。代码分析部分的主要题型有4种:

- (1) 要求给出执行结果。
- (2) 要求找出错误。
- (3) 选择一个答案。

(4) 填写一个空白。

3. 开发实践

提高程序开发能力是本书的主要目标。本书在绝大多数单元后面都给出了相应的作业题目。但是,完成这些题目并非就是简单地写出其代码,而要将它看做是一个思维+语法+方法的工程训练。因此,要求在上机之前先写出准备文档。准备文档的内容包括:

- (1) 问题分析与建模。
- (2) 源代码设计。
- (3) 测试用例设计。

这些文档内容的准备应当作为是否可以上机作业的条件。没有这些准备,上机就比较盲目,收获不会太大。在学校中,学生进入机房之前,教师应当先检查学生是否已经准备了这些内容,并将准备情况作为该次上机作业成绩的一部分或不允许上机。

经过上机作业,学生还应当提交作业报告。作业报告内容包括如下内容:

- (1) 上机作业时发现的问题。
- (2) 对于发现的问题采取的调试方法。
- (3) 对自己准备文档中给出的测试用例的评价。
- (4) 程序运行结果分析。
- (5) 编程心得。

4. 探索思考

建构主义提倡,学习者要用探索法和发现法去建构知识的意义,学习者要在意义建构的过程中主动地搜集和分析有关的信息资料,对碰到的问题提出各种假设并努力加以验证。按照这一理论,本书还提供了一个探索思考栏目,以培养学习者获取知识的能力和不断探索的兴趣,同时可以从更深层次上探究 Java 语法。

(四)

写书难,写教材更难。一本专著,仅用于表达自己的见地;而一本好的教材,不仅是科学技术知识和方法的精华,还应当是先进教育理念的结晶。离这些,我还差得很远。好在有二十余年教学的经历和不断进行教学改革探索的积累,以及许多热心者的支持和帮助。

在此,我要衷心感谢在这本书的出版过程中付出了劳动的下列人员:姚威、陶利民、张展为、郎贵义、李磊、董兆军、张秋菊、史林娟、文明瑶、刘王敏娜、黄姝敏、陈觉、丁群、方晓、张友明、宋文炳、许剑生、贺竞峰等。此外,还要感谢清华大学出版社为本书出版所做的大量平凡而细致的工作。

本书就要出版了。它的出版,是我在这项教学改革工作跨上的一个新的台阶。我衷心希望能得到有关专家和读者的批评和建议,也希望结交一些志同道合者,把这项改革推向更新的境界。

张基温

2013年5月

目 录

第 1 篇 Java 面向对象启步

第 1 单元 职员类	3
1.1 从对象到类：类的定义	3
1.1.1 对象建模.....	3
1.1.2 类：对象的抽象	4
1.1.3 信息隐蔽与类成员的访问控制.....	6
1.1.4 成员变量与数据类型.....	6
1.1.5 成员方法：定义与调用	8
1.1.6 Java 关键词与标识符	9
1.2 从类到对象：对象的声明、建立与初始化	10
1.2.1 创建对象的过程	11
1.2.2 构造方法	12
1.2.3 对象成员的访问与 this	13
1.3 包.....	13
1.3.1 包的作用与结构	13
1.3.2 包的声明	14
1.3.3 包或类的导入	14
1.4 流与 Java 输入输出初步	15
1.4.1 流的概念	15
1.4.2 System 类与标准 I/O 流对象.....	15
1.4.3 使用 PrintStream 类的 println() 和 print() 方法输出	16
1.4.4 使用 Scanner 类进行键盘输入	16
1.5 源程序文件与 Java 平台	17
1.5.1 主方法	17
1.5.2 源程序文件	17
1.5.3 Java 程序的编译与解释	19
1.6 Java 开发运行环境	20
1.6.1 JDK	20
1.6.2 Eclipse 开发环境	22
习题 1	27
第 2 单元 计算器类	32
2.1 计算器类的定义.....	32
2.1.1 问题建模	32

2.1.2	Calculator 类的 Java 描述	32
2.1.3	运算符及其运算规则	33
2.1.4	程序的可靠性：程序测试和调试	33
2.1.5	整数除的风险与对策	35
2.2	异常处理——Calculator 类改进之一	36
2.2.1	Java 的隐含异常处理机制	37
2.2.2	Java 程序中的显式异常处理	37
2.2.3	进行异常处理的 4 种方式	38
2.3	用选择结构确定计算类型——Calculator 类改进之二	42
2.3.1	用 if-else 实现方法 calculate()	42
2.3.2	关系操作符	44
2.3.3	用 switch 结构实现计算方法 calculate()	44
2.4	判断结构的测试	46
2.4.1	逻辑覆盖测试及其策略	46
2.4.2	本例的测试用例设计	50
2.5	使用 static 成员变量存储计算结果——Calculator 类改进之三	50
2.5.1	static 变量的性质	50
2.5.2	带有 static 成员的 Calculator 类定义	51
	习题 2	52
第 3 单元	素数序列	60
3.1	问题描述与对象建模	60
3.1.1	对象建模	60
3.1.2	getPrimeSeries() 方法算法分析	60
3.1.3	isPrime() 方法的设计与静态方法的使用	61
3.2	变量的访问属性	63
3.2.1	变量的作用域	63
3.2.2	静态变量、实例变量与方法变量的存储与生命期	64
3.2.3	对象的生命期与 Java 垃圾回收	64
3.3	Java 的重复控制结构	65
3.3.1	while 结构和 do-while 结构	65
3.3.2	for 结构	66
3.3.3	continue 语句和 break 语句	67
3.3.4	PrimeSeries 类的定义代码	68
3.4	重复控制结构的测试	69
3.4.1	等价分类法与边值分析法	69
3.4.2	循环结构的测试用例设计	70
3.4.3	本例测试数据的设计	70

3.5	知识链接	71
3.5.1	Java 操作符的优先级别与结合性	71
3.5.2	表达式及其求值规则	72
	习题 3	72
第 4 单元	阶乘计算器	80
4.1	阶乘计算器类建模	80
4.2	用迭代方法实现的阶乘计算器	80
4.2.1	用迭代法实现 getFactorial() 方法	80
4.2.2	考虑特殊情况的 getFactorial() 方法算法	81
4.2.3	构造方法定义	82
4.2.4	测试用例设计	82
4.2.5	主方法设计	83
4.2.6	完整的阶乘类定义	83
4.3	用递归方法实现的阶乘计算器	84
4.3.1	问题的提出	84
4.3.2	递归过程	85
4.4	Hanoi 游戏机	86
4.4.1	问题描述	86
4.4.2	问题建模	86
4.4.3	成员方法 hanoiPlay() 的实现	86
4.4.4	HanoiGame 类的定义	87
4.4.5	递归过程分析	87
4.5	知识链接	88
4.5.1	直接数的表示形式	88
4.5.2	BigDecimal 和 BigInteger	90
4.5.3	基本类型的包装	91
4.5.4	类型转换	92
4.5.5	String 类	95
	习题 4	96
第 5 单元	扑克游戏	99
5.1	数组与扑克牌的表示和存储	99
5.1.1	数组的概念	99
5.1.2	数组引用变量的声明和创建	99
5.1.3	数组的显式初始化	100
5.1.4	数组元素的访问	101
5.1.5	ForEach 循环	101
5.2	随机数与扑克游戏洗牌	102
5.2.1	随机数的概念	102

5.2.2	Random 类	102
5.2.3	一次洗牌算法	103
5.2.4	洗牌方法的测试	104
5.2.5	n 次洗牌算法	105
5.2.6	Math 类	107
5.3	扑克整理与数组排序	108
5.3.1	冒泡排序算法	108
5.3.2	基于冒泡排序算法的 cardSort() 方法	109
5.4	扑克发牌与二维数组	109
5.4.1	基本的发牌算法	109
5.4.2	用二维数组表示玩家手中的牌	110
5.4.3	使用二维数组的发牌方法	111
5.5	数组实用类: java.util.Arrays	113
5.5.1	数组元素检索方法	113
5.5.2	数组比较方法	113
5.5.3	数组元素填充方法	114
5.5.4	数组元素升序排序方法	114
5.6	String 数组与命令行参数	115
5.6.1	引用数组和 String 数组	115
5.6.2	命令行参数	115
5.7	知识链接: Vector 类	115
	习题 5	118
第 6 单元	学生-研究生	122
6.1	学生类-研究生类层次结构	122
6.1.1	由 Student 类派生 GradStudent 类	122
6.1.2	关键词 this 和 super	124
6.1.3	Java 继承规则	126
6.2	访问权限控制	126
6.2.1	类成员的访问权限控制级别	126
6.2.2	类的访问权限控制级别	126
6.2.3	private 构造方法	127
6.3	类层次中的类型转换	127
6.3.1	类层次中的赋值兼容规则	127
6.3.2	里氏代换原则	128
6.3.3	类型转换与类型测试	128
6.4	方法覆盖与隐藏	129
6.4.1	派生类实例方法覆盖基类中原型相同的实例方法	129
6.4.2	派生类静态方法隐藏基类中原型相同的静态方法	130

6.4.3 JVM 的绑定机制	131
6.5 知识链接	132
6.5.1 Object 类	132
6.5.2 Java 异常类和错误类的继承体系	135
习题 6	137

第 2 篇 Java 面向抽象编程

第 7 单元 抽象类与接口	151
7.1 圆、三角形和矩形	151
7.1.1 三个独立的类: Circle、Rectangle 和 Triangle	151
7.1.2 final 关键字	153
7.1.3 Java 构件修饰符	154
7.1.4 枚举	154
7.2 抽象类	155
7.2.1 由具体类抽象出抽象类	155
7.2.2 抽象类的特征	160
7.3 接口	160
7.3.1 接口及其定义	160
7.3.2 接口的实现	161
7.3.3 关于接口的进一步讨论	163
习题 7	164
第 8 单元 面向对象程序设计的基本原则	172
8.1 从可重用说起: 合成/聚合优先原则	173
8.1.1 继承重用的特点	174
8.1.2 合成/聚合重用及其特点	174
8.1.3 合成/聚合优先原则	174
8.2 从可维护性说起: 开闭原则	176
8.2.1 软件的可维护性和可扩展性	176
8.2.2 开闭原则	177
8.3 面向抽象编程	177
8.3.1 具体与抽象	177
8.3.2 依赖倒转原则	178
8.3.3 接口与面向接口的编程	178
8.3.4 面向接口编程举例	179
8.4 单一职责原则	183
8.4.1 对象的职责	183
8.4.2 单一职责原则的概念	183
8.4.3 接口分离原则	184

8.5	不要和陌生人说话	188
8.5.1	狭义迪米特法则	189
8.5.2	广义迪米特法则	189
	习题 8	190
第 9 单元	设计模式举例	193
9.1	图形对象的创建问题	193
9.1.1	简单工厂模式	194
9.1.2	工厂方法模式	199
9.1.3	其他创建性设计模式	203
9.2	商场营销问题	204
9.2.1	策略模式的引入	205
9.2.2	策略模式与简单工厂模式结合	210
9.2.3	其他行为型设计模式	213
9.3	消息发送问题	218
9.3.1	不用模式的解决方案	218
9.3.2	采用桥接模式的解决方案	222
9.3.3	其他结构型设计模式	225
	习题 9	229
第 10 单元	反射、配置文件与程序打包发布	231
10.1	反射	231
10.1.1	类文件与类加载	231
10.1.2	Class 对象	232
10.1.3	反射 API	235
10.1.4	使用反射的简单工厂模式	237
10.2	Java 程序的属性配置文件	238
10.2.1	.properties 配置文件和 XML 配置文件	238
10.2.2	Java 应用程序读取配置文件	239
10.2.3	使用反射+配置文件的简单工厂模式	242
10.3	Java 程序的打包与发布	243
10.3.1	Java 程序打包与 JAR 文件包	243
10.3.2	manifest 文件	244
10.3.3	创建可执行 JAR 包	246
	习题 10	250

第 3 篇 基于 API 的 Java 开发

第 11 单元	Java 图形用户界面技术	255
11.1	图形用户界面的基本知识	255
11.1.1	组件与容器	255

11.1.2	布局管理器与坐标系	256
11.1.3	事件与事件监听器	258
11.1.4	GUI 程序开发的基本过程	259
11.2	AWT 及其应用	259
11.2.1	AWT 及其 API	259
11.2.2	抽象类 Component 与 AWT 组件	259
11.2.3	抽象类 Container 与 AWT 容器	262
11.2.4	AWT 布局管理器	264
11.3	事件处理	266
11.3.1	AWT 事件类	266
11.3.2	事件监听接口	266
11.3.3	组件注册事件监听器	269
11.3.4	事件适配器	271
11.3.5	内嵌类、匿名类及其在事件处理中的应用	271
11.4	Swing 及其应用	275
11.4.1	Swing 的特点	275
11.4.2	Swing 组件	276
11.4.3	Swing 布局管理器	281
11.4.4	Swing 事件处理	284
	习题 11	286
第 12 单元	Java 多线程	291
12.1	Java 多线程概述	291
12.1.1	进程与线程	291
12.1.2	线程的状态	292
12.1.3	Java 多线程机制	293
12.1.4	Thread 类	293
12.2	线程状态控制	295
12.2.1	创建线程	295
12.2.2	线程调度	298
12.2.3	线程暂停	299
12.3	多线程管理	299
12.3.1	多线程同步共享资源	299
12.3.2	线程死锁问题	301
12.3.3	线程组	301
	习题 12	302
第 13 单元	Java 网络程序设计	307
13.1	Java Socket 程序设计基础	307
13.1.1	IP 地址与 InetAddress 类	307

13.1.2	端口号与套接口地址	309
13.1.3	客户端/服务器工作模式	311
13.2	基于 TCP 的 Java Socket 程序设计	311
13.2.1	Socket 类和 ServerSocket 类	311
13.2.2	TCP Socket 通信过程	313
13.2.3	TCP Socket 程序设计	314
13.3	基于 UDP 的 Java Socket 程序设计	318
13.3.1	DatagramPacket 类和 DatagramSocket 类	318
13.3.2	DatagramSocket 类	319
13.3.3	UDP Socket 程序设计	321
13.4	基于 URL 的 Java 网络程序设计	324
13.4.1	URI、URL 和 URN	324
13.4.2	URL 类	325
13.4.3	URLConnection 类	326
	习题 13	326
第 14 单元	JavaBean	328
14.1	JavaBean 概述	328
14.1.1	软件组件与 JavaBean	328
14.1.2	JavaBean 结构	328
14.1.3	JavaBean 规范	331
14.2	开发 JavaBean	331
14.2.1	JavaBean 的 API	331
14.2.2	JavaBean 开发工具	332
	习题 14	334
第 15 单元	Java 数据持久化技术	336
15.1	数据库与数据持久化技术	336
15.1.1	数据库与 SQL	336
15.1.2	数据持久层的概念	339
15.2	JDBC 原理	340
15.2.1	JDBC 体系结构	340
15.2.2	JDBC 驱动程序	341
15.2.3	用 JDBC URL 命名数据源	342
15.3	JDBC 工作流程	344
15.3.1	加载数据库驱动程序	344
15.3.2	创建一个数据源连接	347
15.3.3	创建 SQL 工作空间	349
15.3.4	传送 SQL 语句,得到结果集	349
15.3.5	处理结果集	351

15.3.6	关闭数据库连接	352
15.3.7	实例	352
15.4	连接数据库的配置文件	354
15.4.1	从配置文件读取连接数据库字符串的程序结构	354
15.4.2	主流数据库配置文件示例	357
15.5	事务处理	358
15.5.1	事务的概念	358
15.5.2	JDBC 中的事务处理	359
15.5.3	实例	360
15.6	持久层架构 Hibernate	362
15.6.1	ORM	362
15.6.2	Hibernate 的基本原理	362
15.6.3	Hibernate 工作流程	364
15.6.4	Hibernate 开发流程	364
15.6.5	Hibernate 开发环境	365
习题 15		365

第 4 篇 面向 Web 开发

第 16 单元	Java Web 概述	369
16.1	Web 支持技术	369
16.1.1	超文本与超媒体	369
16.1.2	HTML	370
16.1.3	XML	371
16.1.4	HTTP	377
16.2	Java Web 技术演进	381
16.2.1	Java Applets——客户端 Java 动态 Web 技术	381
16.2.2	从 CGI 到 Java Servlet 再到 JSP	383
16.2.3	Java EE	385
习题 16		386
第 17 单元	Java Servlet	390
17.1	Servlet 环境	390
17.1.1	安装 JSDK	390
17.1.2	JSDK 的配置	390
17.1.3	Servlet 容器与 Tomcat 服务器	391
17.2	Servlet 语法	393
17.2.1	Servlet 类定义	393
17.2.2	Servlet 对象生命周期	394
17.2.3	HttpServlet 类的 doXxx() 方法	395

17.3	Servlet 运行	396
17.3.1	Servlet 类字节码文件的生成和保存	396
17.3.2	Servlet 的部署和运行	397
	习题 17	398
第 18 单元	JSP 技术	404
18.1	JSP 概要	404
18.1.1	JSP 页面结构	404
18.1.2	JSP 运行机制	405
18.1.3	JSP 环境引擎	405
18.1.4	JSP 注释元素	407
18.1.5	JSP 页面元素	407
18.2	JSP 脚本元素	408
18.2.1	JSP 声明	408
18.2.2	JSP 表达式	408
18.2.3	JSP 代码段	408
18.3	JSP 指令元素	409
18.3.1	include 指令	409
18.3.2	page 指令	409
18.4	JSP 动作元素	410
18.4.1	页面插入动作 include	411
18.4.2	页面切换动作 forward	412
18.5	JSP 内置对象	413
18.5.1	out 对象	414
18.5.2	request 对象	414
18.5.3	response 对象	415
18.5.4	session 对象	419
18.5.5	application 对象	421
18.6	JSP 与 JavaBean 交互	425
18.6.1	JSP 与 JavaBean 交互的动作元素	425
18.6.2	JSP 与 JavaBean 交互实例	427
	习题 18	429
第 19 单元	MVC 模式	434
19.1	MVC 概述	434
19.1.1	MVC 的概念	434
19.1.2	MVC 组件与工作原理	434
19.1.3	MVC 的优势与应用	435
19.2	JSP+Servlet+JavaBean 实现的 MVC	435
19.2.1	JSP+Servlet+JavaBean 实现的 MVC 模式结构	435

19.2.2	JSP+Servlet+JavaBean 实现的 MVC 实例	436
习题 19	440
第 20 单元	Java Web 架构简介	442
20.1	软件架构概述	442
20.1.1	架构的概念	442
20.1.2	软件架构及其目标	443
20.1.3	Java EE 应用分层模型	443
20.2	经典 MVC 支柱框架——Struts	444
20.2.1	Struts 的构成	444
20.2.2	Struts 的工作过程	446
20.2.3	Struts 应用	446
20.2.4	Struts 与 Hibernate 整合	446
20.2.5	Struts 2	447
20.3	基于控制反转和面向切面的轻量级框架——Spring	448
20.3.1	基本概念	448
20.3.2	Spring 的组件架构	451
20.3.3	Spring 的下载和安装	453
20.3.4	Spring 应用实例	453
20.3.5	SSH 整合	455
20.4	Java Web 用户界面标准——JSF 框架	455
20.4.1	JSF 的特点	455
20.4.2	JSF 的工作流程	456
20.4.3	JSF 应用程序结构	456
习题 20	457
附录 A	Java 运行时异常类和错误类	458
A.1	RuntimeException 类	458
A.2	Error 类	459
参考文献	462