



新世纪应用型高等教育
网络专业系列规划教材

Web高级程序设计 (Java&JSP)

新世纪应用型高等教育教材编审委员会 组编

主编 丁一凡 姚远



大连理工大学出版社



新世纪应用型高等教育
网络专业系列规划教材

Web高级程序设计 (Java&JSP)

新世纪应用型高等教育教材编审委员会 组编

主编 丁一凡 姚远



大连理工大学出版社

图书在版编目(CIP)数据

Web 高级程序设计:Java&JSP/丁一凡,姚远
主编. — 大连:大连理工大学出版社,2011.2
新世纪应用型高等教育网络专业系列规划教材
ISBN 978-7-5611-6029-9

I. ①W… II. ①丁… ②姚… III. ①
JAVA 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(2011)第 020554 号

大连理工大学出版社出版

地址:大连市软件园路 80 号 邮政编码:116023

发行:0411-84708842 邮购:0411-84703636 传真:0411-84701466

E-mail:dutp@dutp.cn URL:<http://www.dutp.cn>

大连美跃彩色印刷有限公司印刷 大连理工大学出版社发行

幅面尺寸:185mm×260mm 印张:17.25 字数:398 千字

印数:1~2500

2011 年 2 月第 1 版

2011 年 2 月第 1 次印刷

责任编辑:马 双

责任校对:李 红

封面设计:张 莹

ISBN 978-7-5611-6029-9

定 价:33.00 元

Java 是于 1991 年起源于美国的一种面向对象的程序设计语言。Java 语言具有简单、面向对象、分布式、解释执行、鲁棒、安全、体系结构中立、可移植、高性能、多线程以及动态性等诸多特点,在国外早已广泛应用于各种软件开发工程项目,特别是在互联网普及的 3G 时代,Java 已大量应用于 Web 应用程序开发方面。目前,国内的各类软件外包企业和开发企业也已经纷纷选择 Java 作为软件开发的首选工具之一。可以说 Java 不仅是一个广泛使用的网络编程语言,同时它代表了一种新的计算概念。

本书作为普通高等学校培养应用型人才的实用型教材,围绕计算机及相关专业 Web 程序设计课程的教学实际设计教学思路,以改革计算机应用课程教学为出发点,以培养学生适应新时代软件开发技术的发展需要为目标,以面向对象、数据库的连接应用和网络应用为知识讲授的核心来构建教材的体系结构。

在本书的编写设计中,我们尽可能改变传统程序设计语言教材重视语言的词法语法,忽视配套工具、环境、中间件等整体应用的编写方式,尽可能精简基本词法语法的篇幅,力争在有限的课时内,围绕一个真实可行的 Web 程序设计项目,将 Java 的应用作为一个完整的 Web 程序设计平台进行系统的描述。

Java 平台是基于 Java 语言的一系列技术及相关产品,一共包括 3 个版本:Java 平台的标准版本 Java SE、Java 平台的企业级版本 Java EE 和使用 Java 平台的 ME 版本。

Java SE 版本包括两部分:JDK 和 JRE(可分别下载和单独使用)。JDK 包含了 JRE,还包括一些工具,如开发 Java application 类型程序和 applet 类型程序的编译器、调试器等。JRE 中包含了 Java 的核心类库、Java 虚拟机以及运行 Java application 类型程序和 applet 类型程序的一些组件等。在本书中将 Java 语言及 Java SE 的技术、工具等作为 Web 应用程序开发的基础进行讲述。



使用 Java 平台的标准版本 Java SE 可以开发出 Java application 类型程序和 applet 类型程序,使用 Java 平台的企业级版本 Java EE 可开发出 Web 应用程序,这是本书的重点部分,而使用基于 Java 平台的 ME 版本技术可开发出手机应用程序等。无论何种类型的程序,一般都以 JDK 中的核心类库为基础进行开发。

本书的 1~6 章讲述了 Java 语言面向对象的特点、Java SE 中的核心类库的使用、Java 的异常处理机制、Java 的基本输入输出、文件处理等;在读者充分掌握了 Java 核心类库的使用,对于面向对象语言开发有了较为全面的认识之后,在 7~10 章中重点讲解了 Java 语言在 Web 应用程序开发中的应用,包括 JSP 基础、Servlet 开发、Struts 开发等;而由于 Web 应用程序是与数据库访问密切相关的,所以在本书的 11~12 章讲解了 Java 平台与数据库连接访问的 JDBC、Hibernate 数据访问技术,使读者对于 Java 语言在 Web 开发中的使用能够有一个完整的系统性的理解,达到学完本课程后能够利用所学知识完成完整的 Web 应用系统开发的目标。

本书在内容编排上,力争做到体系完整、结构清晰、实例丰富、理论知识讲解循序渐进,具有普遍的适用性。本书可作为应用型高等学校本、专科学生的教材,也可作为希望学习 Java 语言程序设计和使用 Java 作为 Web 软件开发平台的读者们的自学或参考用书。

在本书的编写过程中,丁一凡老师完成了全书的总体规划,主持且参与了 1~6 章的编写,并承担了全书的校改任务;姚远老师提供了 1~6 章的案例和代码,完成了 7~12 章的编写以及 1~6 章的审核及部分编写任务。此外,武汉工业学院工商学院信息工程系的张勤、张樊、孙敏等老师为本书的编写提出了宝贵的意见和建议,王晶老师为书中的部分案例代码进行了核实与验证。在此,编者向上述人员表示由衷的感谢。

由于编者的水平与经验有限,书中难免有疏漏之处,恳请广大读者给予批评指正。

所有意见和建议请发往:dutpbk@163.com

欢迎访问我们的网站:<http://www.dutpgz.cn>

联系电话:0411-84707492 84706104

编 者

2011 年 2 月



第 1 章 Java 语言概述与语言基础	1
1.1 Java 概述	1
1.1.1 Java 的面向对象特性	1
1.1.2 Java 的跨平台与可移植特性	1
1.2 Java 的开发工具	2
1.2.1 JDK 概述	2
1.2.2 JDK 的安装与配置	3
1.2.3 开发工具 Eclipse	5
1.3 Java 语言基础	7
1.3.1 一个简单的 Java 程序	8
1.3.2 main 方法的参数	9
1.3.3 基本数据类型	10
1.3.4 常量与变量	10
1.3.5 流程控制语句	12
1.3.6 枚举	15
1.3.7 数组	16
习 题	19
第 2 章 类和对象	21
2.1 定义 Java 的类	21
2.1.1 类的基本结构	21
2.1.2 定义初始化	22
2.1.3 构造方法	23
2.1.4 方法重载	25
2.2 创建与使用对象	25
2.2.1 创建对象	25
2.2.2 使用对象	26
2.3 方法调用	28
2.3.1 方法的返回值	28
2.3.2 实例方法的调用	30
2.3.3 this 引用	30
2.3.4 静态方法的调用	32
2.4 访问数据成员	33
2.4.1 静态变量与实例变量	34
2.4.2 局部变量、全局变量、静态变量、实例变量的区别	36
2.4.3 final 变量	37
2.5 包的概念	40
2.5.1 创建包	40
2.5.2 使用包	40

2.5.3 包在类的访问控制上的意义与用途	41
2.5.4 类成员的缺省修饰符	42
习 题	44
第 3 章 类的层次结构	47
3.1 类的继承	47
3.1.1 使用 extends 继承	47
3.1.2 super 与派生类对象的构造	49
3.1.3 继承中的方法覆盖	54
3.1.4 super 的其他用法	55
3.1.5 super 关于变量隐藏的优势	57
3.1.6 final 方法	58
3.2 final 类	59
3.3 多 态	60
3.3.1 子类覆盖父类的方法	60
3.3.2 方法重载	60
3.3.3 向上类型转换	61
3.4 抽象类	62
3.4.1 抽象类与抽象方法	62
3.4.2 抽象类的继承	63
3.5 接 口	65
3.5.1 接口的定义	65
3.5.2 接口与继承	65
3.5.3 抽象方法的实现	65
3.5.4 接口与多继承	66
3.6 内部类	67
3.7 匿名类	69
3.7.1 匿名类实现某个接口	69
3.7.2 继承的匿名类	70
3.8 实例:登录程序的接口设计与类实现	71
3.9 实例:简单的高校工资管理程序的类层次结构设计 with 实现	74
习 题	79
第 4 章 Java 资源库	81
4.1 String	81
4.1.1 String 对象的创建	81
4.1.2 字符串的比较	82
4.2 包裹类	83
4.2.1 Character	83
4.2.2 Float	84
4.2.3 Boolean	84
4.2.4 Integer	85
4.2.5 字符串与数值数据间的相互转换	87
4.3 StringBuffer	91
4.4 Math	91
4.5 Vector	92

4.6	Stack	95
4.7	ArrayList	96
4.8	日期类	98
4.9	实例:Vector 应用于工资管理程序	99
	习 题	100
第 5 章	文件处理	102
5.1	概 述	102
5.2	标准输入输出流	102
5.2.1	System 类	102
5.2.2	标准输入流与标准输出流	103
5.3	文件创建与查看	105
5.3.1	创建文件	106
5.3.2	输出指定目录(文件)的相关信息	107
5.3.3	创建目录	109
5.4	读写文件	110
5.4.1	使用 FileReader 对象读文件	110
5.4.2	使用 FileWriter 对象写文件	112
5.4.3	使用 FileInputStream 对象从文件读取数据	112
5.4.4	使用 FileOutputStream 对象将文件内容写入到另一个文件	113
5.5	读写文件实例	114
5.5.1	实例:输入英文写入到文件	114
5.5.2	实例:输入中文汉字写入到文件	115
5.6	实例:工资管理程序中的工资数据文件存储	116
	习 题	118
第 6 章	异 常	120
6.1	概 述	120
6.2	抛出异常	121
6.2.1	throw 语句抛出异常	121
6.2.2	系统抛出异常	122
6.3	异常的类型	123
6.4	异常处理结构	123
6.4.1	try-catch	124
6.4.2	try-catch-finally	125
6.4.3	关于 catch 匹配	126
6.4.4	用特定异常类型代替通用异常类型	127
6.5	throws 异常声明	127
6.6	自定义异常	129
	习 题	132
第 7 章	HTML 基础	133
7.1	认识 HTML	133
7.2	HTML 文件结构介绍	134
7.3	HTML 文件的常见标记	135
7.3.1	文档结构标记	135
7.3.2	内容标记	137

7.3.3	一般布局类标记	142
7.3.4	表格与相关标记	144
7.3.5	表单与相关标记	154
7.3.6	辅助标记	160
7.4	HTML 文件显示风格简述	162
7.4.1	层叠样式表风格	162
7.4.2	老式的显示风格方案	162
7.4.3	层叠样式表的应用	162
7.4.4	外部文件	168
习 题		168
第 8 章	JSP 开发基础	169
8.1	JSP 的概念与作用	169
8.1.1	JSP 页面简介	169
8.1.2	JSP 页面指令	170
8.1.3	JSP 标签行为	172
8.1.4	JSP 代码片断	173
8.1.5	静态模板及变量获取语言 EL 简介	174
8.2	Tomcat 的配置与安装	175
8.2.1	安装:下载并配置环境变量	176
8.2.2	测试 Tomcat 是否安装正确	176
8.2.3	在 Eclipse 中建立 Web 项目	178
8.3	一个简单的 JSP 页面的编写与运行	179
8.3.1	在 Eclipse 中新建 JSP 页面	179
8.3.2	运行 JSP 程序	180
8.4	JSP 页面处理表单数据	180
8.4.1	使用表单和输入标签完成页面	180
8.4.2	从 URL 请求中获取数据	181
8.4.3	从请求中获取数据实例	182
8.4.4	认识 request	183
8.5	JSP 内置对象	184
8.5.1	输出对象 out	184
8.5.2	响应对象 response	184
8.5.3	会话对象 session	184
8.6	实例:使用 JSP 页面来进行输入输出的工资计算	185
8.6.1	功能简述	185
8.6.2	具体步骤及源码	186
习 题		188
第 9 章	Servlet 基础	189
9.1	Servlet 概述	189
9.1.1	Web 服务器	189
9.1.2	Servlet 容器	189
9.1.3	Servlet 概念	189
9.2	Servlet 的生命周期	190
9.2.1	生命周期概述	190

9.2.2 生命周期实例	191
9.2.3 接口 Servlet 与类 HttpServlet	192
9.2.4 Servlet 的基本配置	192
9.3 Servlet 发送页面到客户端	193
9.3.1 Servlet 的调用过程	193
9.3.2 发送页面到客户端实例	194
9.3.3 web.xml 配置 Servlet	195
9.4 Servlet 处理表单数据	196
9.4.1 表单提交 get 请求 Servlet 处理	196
9.4.2 从请求中获取数据	197
9.4.3 通过 Response 处理响应	197
9.4.4 实例	197
9.5 Servlet 请求转发	201
9.5.1 ServletContext 对象	201
9.5.2 RequestDispatcher 对象	201
9.5.3 HttpServletResponse 对象的 sendRedirect 方法	201
9.6 实例:使用 Servlet 完成工资计算	202
9.6.1 功能简述	202
9.6.2 源码清单	203
习 题	206
第 10 章 Struts 基础	207
10.1 什么是 Struts	207
10.1.1 Struts 的版本	207
10.1.2 Struts 应用程序的架构	208
10.2 使用 Eclipse 开发 Struts 2 网站	209
10.2.1 Struts 2 的运行环境设置	210
10.2.2 Struts 2 的库文件	210
10.2.3 修改 web.xml 配置文件	211
10.2.4 添加 Struts.xml 配置文件	212
10.2.5 添加 Action 代码类	212
10.2.6 编辑网页生成模板	213
10.2.7 网站打包部署	214
10.3 Struts 的配置文件	215
10.3.1 <package> 结点	215
10.3.2 <namespace> 属性	216
10.3.3 <constant> 结点	216
10.3.4 <action> 结点	216
10.3.5 <result> 结点	218
10.4 编写 Struts 的 Action	219
10.4.1 ActionSupport 类	219
10.4.2 ModelDriven 接口	221
10.5 网页生成模板	223
10.5.1 数据使用	223
10.5.2 条件生成	225

10.5.3 集合对象	226
10.5.4 表单标签	228
10.6 实例:使用 Struts 完成工资计算	228
10.6.1 功能简述	229
10.6.2 源码清单	229
习 题	233
第 11 章 JDBC 数据库访问技术	234
11.1 JDBC 与关系数据库	234
11.1.1 JDBC 概述	234
11.1.2 数据库连接类	235
11.1.3 JDBC 数据库驱动程序	235
11.1.4 数据库访问类	236
11.1.5 数据库结果类	236
11.1.6 建立实例数据库	236
11.2 应用 JDBC 完成数据库访问	237
11.2.1 建立数据库连接	237
11.2.2 查询数据库	239
11.2.3 更新数据库	240
11.2.4 预准备语句	243
11.2.5 结果集	245
11.2.6 事务处理	246
11.3 使用 JDBC 操作数据库实例	247
11.3.1 在 JSP 页面中查询数据	247
11.3.2 分页	251
习 题	251
第 12 章 Hibernate 应用实例与 Spring 简述	252
12.1 Hibernate 技术简介	252
12.1.1 Hibernate 的基本对象	253
12.1.2 Hibernate 开发环境设置	255
12.1.3 编辑 Hibernate 配置文件	255
12.1.4 在 Eclipse 中安装 Hibernate 工具	257
12.1.5 使用反向生成工具生成映射类	258
12.2 Hibernate 构建应用系统实例	261
12.2.1 基本框架代码	261
12.2.2 显示单个教师信息	262
12.2.3 新建教师信息	263
12.2.4 修改教师信息	263
12.2.5 删除教师信息	264
12.2.6 查看所有教师信息	265
12.3 Spring 简述	265
习 题	266

1.1 Java 概述

1.1.1 Java 的面向对象特性

Java 是一种完全面向对象的语言。

Alan Kay 总结了 Smalltalk 的五项基本特征。Smalltalk 是最早获得成功的面向对象的编程语言,也是 Java 所依赖的基础之一。这些特征代表了纯面向对象的编程方法:

1. 万物皆对象。可以想象对象是一片存储区域,它能够存储数据,还能够提供某些操作。所以可以把所有待解决的问题域中的概念、需要存储的事件等均变为程序里的对象。

2. 程序就是一组相互之间传递消息,告诉对方该干些什么的对象。在有了对象的基础上,无论需要做什么,都可以看作是对某个对象发送了一个消息,也即是提出请求,请求去调用某个对象的方法。

3. 每个对象都利用其他对象来组建它自己的记忆,即在已有对象的基础上去构建新的对象。程序的复杂性隐藏在对象的简单性之下。

4. 对象都有类型。任何对象都是某个类的实例。可以看作类能够提供功能,每个类都提供了一些方法。所以类和类之间的区分在于它们各自能提供哪些方法。

5. 所有属于同一类型的对象能接受相同的消息。一个 circle 型的对象也是一个 shape 型的对象,所以可以保证 circle 能接受 shape 的消息。即写给 shape 的代码能自动交由任何符合 shape 描述的事物去处理。

上述特征和我们常说的面向对象的类、对象、数据封装、继承、多态等实质是相同的。站在语言的层面上,类是编程中的基本单位,类中包含方法成员、数据成员。对象是类的实例。数据封装使得外界看不到对象内部的实现细节,只能通过对象提供的方法来和对象进行通信。外部无法访问对象内部的数据成员。继承使得程序的复用性大大提高,子类对父类的定义作出某些修改,要么改变同一方法的内部操作;要么增加新的操作。同时因为继承,父类可以提供的方法,子类也具备。通过使用对象可以使得类的扩展效率大大提高,使得用不同的结构去搭建复杂的应用程序成为可能。

1.1.2 Java 的跨平台与可移植特性

一段 Java 源程序编辑完毕后,以 *.java 的文件名形式保存。通过编译器编译成二进制代码,以 *.class 的文件名形式存在。通过 JRE,能使之运行在不同的操作系统之上。如图 1-1 所示。

这种运行方式使得跨平台成为可能。

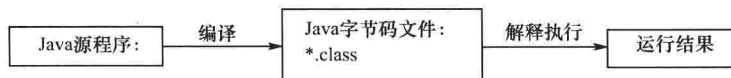


图 1-1 Java 程序执行步骤

Java 虚拟机 (JVM) 不是一个独立的程序,它是 Java 运行时环境(JRE)的一部分。目前有基于各种不同操作系统的 JRE,安装 JRE 并进行相关配置以后,Java 程序才能被执行起来。

正如,EXE 文件在 Windows 操作系统下可以被执行,是因为 Windows 提供了这种 EXE 文件的执行环境。而, class 文件不能被 Windows 直接“接受”,需要有 Java 运行时环境,即 JRE。所以可以使用同一个, class 文件,在不同操作系统的机器上运行。当然前提是这些机器都安装了和自己操作系统配套的 JRE。

1.2 Java 的开发工具

1.2.1 JDK 概述

使用 Java 语言开发的 Java application 类型程序和 applet 类型程序在运行状态下需要有 Java 运行时环境(Java Runtime Environment,JRE)。而开发 Java application 类型程序和 applet 类型程序则需要 Java 开发工具包(Java Development Kit,JDK)。JDK 作为 Java 平台标准版本产品 JavaSE(Java Platform,Standard Edition)中的一部分,目前已发布到 JDK 7。鉴于 JDK7 仍属于发布的早期版本,所以本书仍以 JDK6 为主。

图 1-2 是 JavaSE 平台的所有组件结合的概念图。

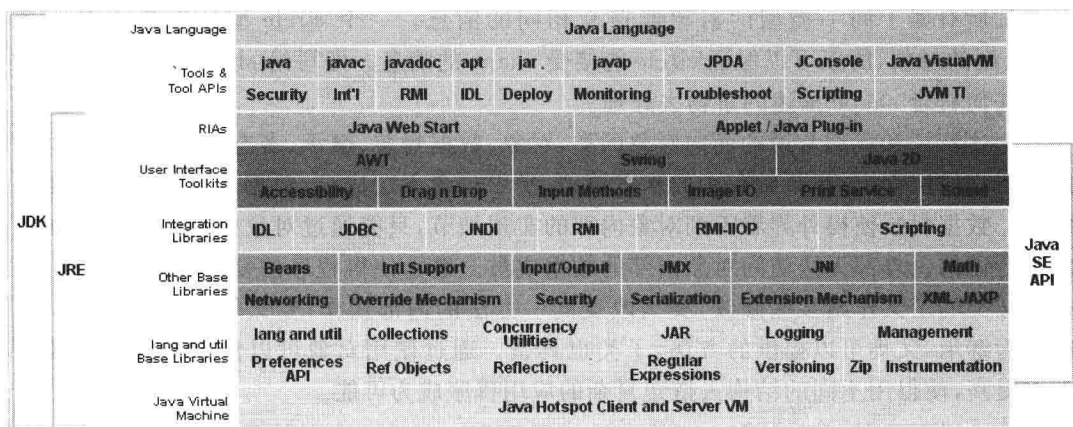


图 1-2 JavaSE 平台组件关系图

从图中可以看出,JavaSE 产品发布了很多供应用程序编程的 API(JavaSE API),这些是使用 Java 开发任何类型程序的基础。Java 虚拟机提供了解释执行 Java 二进制代码等重要功能。在 JRE 之外,是一些编译、解释执行工具。

1.2.2 JDK 的安装与配置

JDK 的系列版本可于 Oracle 网站下载。

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

点击上述链接将会看到包含最新版本信息的页面,如图 1-3 所示:

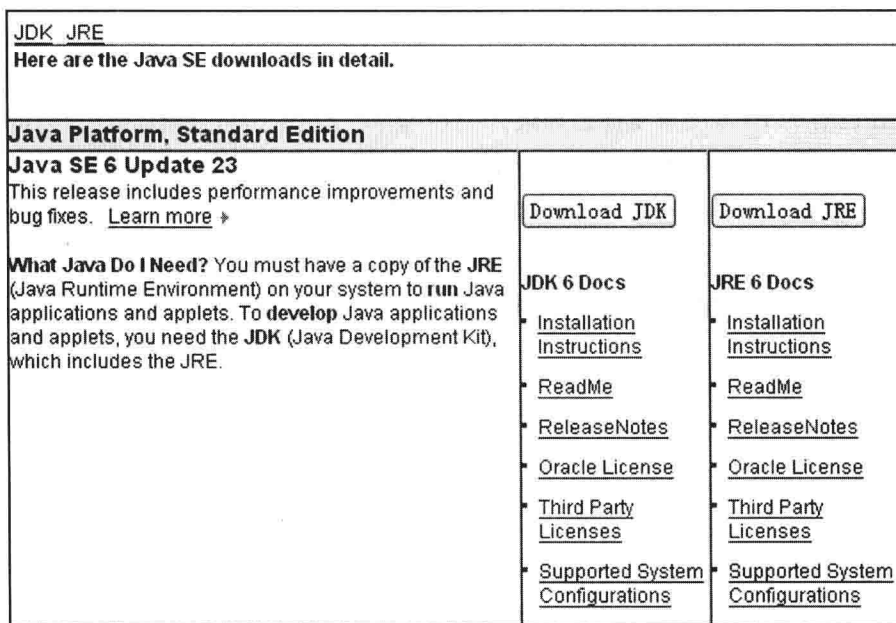


图 1-3 所示 最新版本信息

单击图 1-3 页面中的 Download JDK 按钮,将出现如图 1-4 所示页面内容:

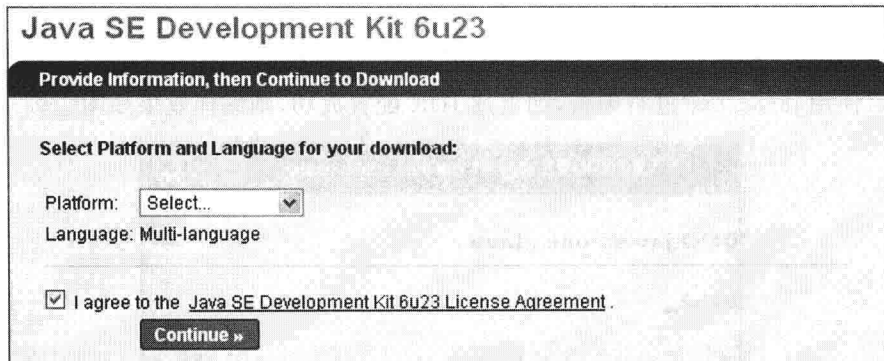


图 1-4 版本选择页面

在 Platform 下拉框中选择适用于不同操作系统的 JDK 6 版本,如使用的是 Windows 系列的操作系统,则下载支持 Windows 的 JDK 版本,即在下拉框中选择 Windows。下载后得到一个 jdk-6u23-windows-i586.exe 文件,这个文件即是安装 JDK 的可执行文件。执行过程中如选择默认路径安装,则安装成功后在 C:\Program Files\Java\路径下将看到 jdk1.6.0_23 文件夹。

文件夹内包含 bin 目录,内有 Java.exe 等;lib 目录,包含一些诸如 dt.jar 的文件等。bin 目录中即是用于开发、编译 Java 源程序文件的工具等;而 lib 目录中是提供给开发人员的核心类。为了使用此 JDK,需要进行安装配置。如该文件夹在磁盘中的位置为:C:\Program Files\Java\jdk1.6.0_23。

具体的安装步骤为:右键点击“我的电脑”→“属性”→“高级”→“环境变量”,新建一个变量,classpath,并设置其值:

```
.;C:\Program Files\java\jdk1.6.0_23\lib;
```

并在已有的 Path 变量里面追加值:C:\Program Files\java\jdk1.6.0_23\bin;

配置完毕重启计算机,可以试着编一个 Java 简单程序,然后使用刚才配置后的 Javac.exe 和 java.exe 进行编译和解释执行。在解释执行时,将会加载所需要的类,从配置的 classpath 中找到相应的.class 文件进行解释、执行。

本例中,可以使用简单的文本编辑器记事本,编辑一个简单的 Java 源程序,Pro1_1:源文件 one.java:

```
public class one
{
    public static void main(String[] args)
    {
        System.out.println("hello! Everyone");
    }
}
```

将 one.java 放置在 d:\。该文件的全名为 d:\one.java

使用 javac.exe 进行编译:

通过“开始”→“运行”,输入“cmd”,单击“确定”进入控制台命令窗口。

然后使用 javac.exe 进行编译,如上述 JDK 配置成功,则编译效果如图 1-5:

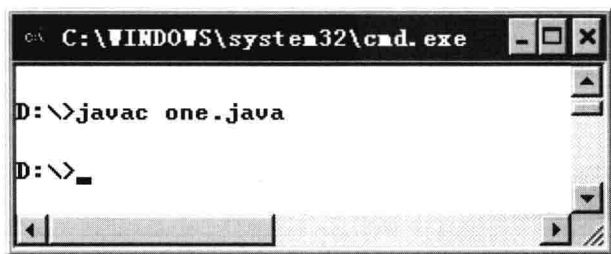


图 1-5 编译效果

在 d 盘中将出现 one.class 文件。

使用 java.exe 运行程序:

输出:hello! Everyone(如图 1-6)

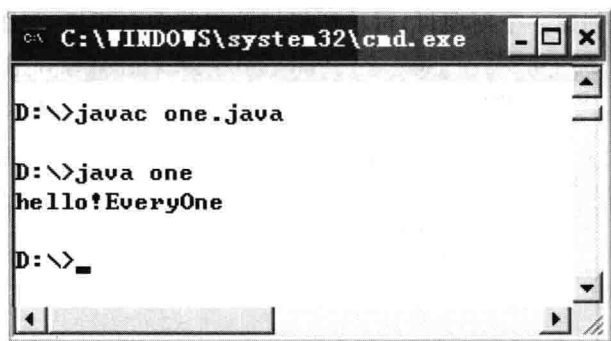


图 1-6 运行结果

1.2.3 开发工具 Eclipse

本教材选择的版本类别为:Eclipse IDE for Java EE Developers。

Eclipse 产品的主界面如图 1-7 所示:

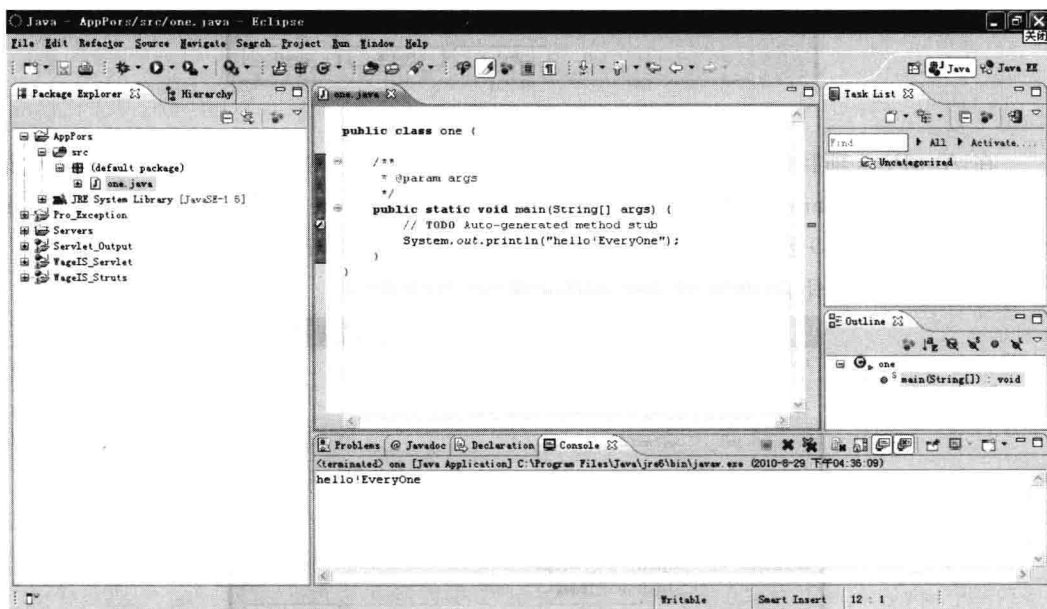


图 1-7 Eclipse 产品主界面

左部 Package Explorer 面板组织、管理所有指定目录下的所有程序文件。中间区域为 Java 源程序编辑区。下方若干个面板是一系列调试工具等。其中在 Console 面板可以看到 Java 程序写入到控制台的数据。

Eclipse 可以完成新建、编辑、运行 java 程序的过程,下面将程序 Pro1_1 使用 Eclipse 重新做一遍。

1. 新建源程序文件 one.java

具体操作步骤:在 Project Explorer 面板中,选择其中一个已经存在的 Java Application 项目,如 AppPors(第一次使用时,将不存在任何 Java Application 项目,需要

使用新建项目功能创建一个新项目,可按照标识符规范命名项目,如图 1-8)。



图 1-8 新建 Java Application 类型的项目

在 AppPros 项目的右键弹出菜单中选择 New→Class。出现 New Java Class 对话框,如图 1-9。在 Name 文本框中键入待定义的类的名字:one,并勾选 public static void main(String[] args)选项框。注意,对话框中其他选项可不作改动。

单击 Finish 按钮。



图 1-9 新建 Java 类