

JSP动态WEB技术 与开发实例

杨生举 史宝明 贺元香 编著



JSP 是一种动态网页技术标准，可以无缝地运行在 UNIX、Linux 及 Windows 操作系统平台上，可以在多种 Web 服务器下使用。本书全面、翔实地介绍了应用 JSP 进行 Web 程序开发所需的各种知识和技能，主要包括：JSP 技术概述，JSP 语法基础，JSP 内置对象，JSP 中的文件操作，JSP 数据库连接技术，JavaBean 技术，Servlet 技术以及 MVC 框架技术。



甘肃人民出版社

JSP动态WEB技术与 与 开发实例

杨生举 史宝明 贺元香 编著



甘肃人民出版社

图书在版编目 (CIP) 数据

JSP动态WEB技术与开发实例 / 杨生举, 史宝明, 贺元香编著. -- 兰州: 甘肃人民出版社, 2014.4
ISBN 978-7-226-04573-2

I. ①J… II. ①杨… ②史… ③贺… III. ①JAVA语言—网页制作工具 IV. ①TP312②TP393.092

中国版本图书馆CIP数据核字 (2014) 第 068495 号

出 版 人: 吉西平

责任编辑: 肖林霞

封面设计: 马吉庆

JSP 动态WEB技术与开发实例

杨生举 史宝明 贺元香 编著

甘肃人民出版社出版发行

(730030 兰州市读者大道 568 号)

甘肃北辰印务有限公司印刷

开本787毫米×1092毫米 1/16 印张19 插页2 字数376千

2014年6月第1版 2014年6月第1次印刷

印 数: 1~1 000

ISBN 978-7-226-04573-2 定 价: 48.00 元

前言

JSP(Java Server Pages)是在Sun Microsystems公司的倡导下,由许多公司共同参与建立的一种新的动态网页技术标准。它在动态网页的建设方面具有强大而特殊的功能。

为了满足广大网络编程爱好者的需要,作者在总结以往开发经验和教学经验的基础上,注重理论和实践相结合,编写了本书。编著者杨生举是甘肃省科学技术情报研究所网络中心技术开发主管,具有十多年的JSP技术开发和培训经历,在省级以上刊物发表论文18篇,主持技术开发项目6项。编著者史宝明、贺元香是兰州文理学院教师,多年来一直担任《JSP动态Web技术》等课程的教学工作,主持和参与多项教改科研项目,在省级以上期刊发表论文20余篇,具有丰富的JSP理论知识和项目开发经验。

本书中我们强调编程思想和方法,淡化枯燥的语言讲解;内容的取舍以实用为主,讲述JSP运用中的问题与难点;并且描述了Web应用程序开发的方式与流程,让读者有法可循。本书中穿插了大量的实例,使读者在阅读这本书时,既能学到理论知识,又能掌握解决实际问题的方法。

为使读者加深理解并掌握每章的重点,在每章的末尾均做了小结,并给出了学习的建议。本书无论是对一个网络新手,还是一个网络高手,都能够起到一定的指导作用,都可以从本书中找到一些有益的知识。

本书的主要内容安排如下:

第1章,“JSP技术概述”,介绍了什么是JSP,JSP与其他动态网页技术的比较,JSP系统环境的安装与配置,JSP程序运行的原理及运行程序。

第2章,“JSP语法基础”,介绍了JSP页面的基本结构,数据类型,变量和

方法的声明, JAVA程序片, JSP注释, JSP指令, JSP动作指令标签。

第3章, “JSP内置对象”, 介绍了Request, Response, Out, Session, Application, Config, page, pagecontext, exception等对象的基本知识和实际应用。

第4章, “JSP中的文件操作”, 介绍了File类, 字节流, 字符流, RandomAccessFile类, 文件上传和下载, 分页读取文件等。

第5章, “JSP数据库连接技术”, 介绍了数据库的安装与创建, JDBC技术, 数据的增删改查的基本操作技术。

第6章, “JavaBean技术”, 介绍了JavaBean的概念和优点, JavaBean的部署和编译以及在JSP中应用JavaBean。

第7章, “Servlet技术”, 介绍了Servlet概念、实现过程和特点, Servlet基本结构和应用示例, 读取Http请求头、Http应答状态和设置Http应答头。

第8章, “MVC模式和SSH技术”, 介绍了MVC模式和SSH技术相关知识, 包括JSP设计模式, MVC结构和SSH框架技术。

本书在编写过程中得到很多人的帮助, 特别是兰州大学管会生教授认真地审阅了全书, 并提出了很多宝贵的修改意见, 在此表示衷心感谢。编著者所在单位甘肃省科学技术情报研究所和兰州文理学院也在本书的出版中给予了极大支持, 在此表示衷心感谢。

本书第1章、第5章、第7章由杨生举编写, 第2章、第3章、第6章由史宝明编写, 第4章、第8章由贺元香编写。本书若有疏漏之处, 敬请各位专家、朋友指正。

本书著作权属于甘肃省科学技术情报研究所。

编者

2014年6月

目 录

第1章 JSP技术概述	(001)
1.1 JSP介绍	(001)
1.1.1 什么是JSP	(001)
1.1.2 JSP与其他动态网页技术的比较	(002)
1.2 JSP系统环境的安装与配置	(004)
1.2.1 安装和配置JDK	(004)
1.2.2 安装配置Tomcat服务器	(009)
1.3 运行JSP程序	(013)
1.3.1 JSP程序运行的原理	(013)
1.3.2 第一个JSP程序	(014)
1.3.3 虚拟目录的配置	(015)
1.4 本章小结	(016)
第2章 JSP语法基础	(017)
2.1 JSP页面的基本结构	(017)
2.2 数据类型	(019)
2.2.1 数据类型概述	(019)
2.2.2 标识符	(020)
2.2.3 简单数据类型	(020)
2.2.4 Java中简单数据类型的转换	(022)
2.2.5 简单类的包装类	(023)
2.2.6 类	(024)
2.3 变量和方法的声明	(025)
2.3.1 声明变量	(025)

2.3.2 声明方法	(027)
2.4 Java程序片	(029)
2.4.1 实体定义	(029)
2.4.2 表达式	(030)
2.4.3 Java代码块	(031)
2.5 JSP注释	(033)
2.6 JSP指令	(035)
2.6.1 Page指令标签	(035)
2.6.2 Include指令标签	(038)
2.6.3 Taglib指令标签	(040)
2.7 JSP动作指令标签	(041)
2.7.1 Include 动作指令	(041)
2.7.2 Forward 动作指令	(043)
2.7.3 Plugin动作指令	(044)
2.7.4 Param动作指令	(046)
2.7.5 UseBean动作标记	(047)
2.8 本章小结	(048)
第3章 JSP的内置对象	(049)
3.1 内置对象概述	(049)
3.1.1 JSP内置对象概述	(049)
3.1.2 JSP内置对象的特点	(051)
3.1.3 JSP内置对象的分类	(051)
3.1.4 JSP对象的作用域	(052)
3.2 Request对象	(054)
3.2.1 Request对象的常用方法	(055)
3.2.2 获取表单数据	(059)
3.2.3 中文乱码处理	(064)
3.2.4 Request对象应用示例	(066)
3.3 Response对象	(071)
3.3.1 Response对象的常用方法	(072)
3.3.2 动态响应 contentType 属性	(075)
3.3.3 使用cookie	(076)

3.3.4 Response 的HTTP 文件头	(078)
3.3.5 Respose 重定向	(079)
3.4 Out对象	(081)
3.5 Session对象	(083)
3.5.1 什么是Session对象	(083)
3.5.2 Session对象的Id	(084)
3.5.3 Session对象的常用方法	(087)
3.5.4 Session 对象与URL重写	(090)
3.5.5 Session对象应用示例	(092)
3.6 Application对象	(098)
3.6.1 什么是application对象	(098)
3.6.2 Application对象的常用方法	(099)
3.6.3 Application对象应用示例	(100)
3.7 Config对象	(103)
3.8 Page对象	(104)
3.8.1 Page对象	(104)
3.8.2 Page对象的方法	(105)
3.9 PageContext对象	(105)
3.10 Exception对象	(109)
3.11 本章小结	(110)
第4章 JSP中的文件操作	(111)
4.1 File 类	(111)
4.1.1 获取文件的属性	(112)
4.1.2 用File类操作目录	(114)
4.1.3 删除文件和目录	(118)
4.2 字节流	(119)
4.2.1 FileInputStream 和FileOutputStream 类	(120)
4.2.2 BufferedInputStream 和BufferedOutputStream 类	(125)
4.3 字符流	(127)
4.3.1 FileReader 和FileWriter 类	(128)
4.3.2 BufferedReader 和BufferedWriter 类	(129)
4.4 RandomAccessFile 类	(130)

4.4.1 RandomAccessFile 类的方法	(130)
4.5 文件上传	(138)
4.6 文件下载	(144)
4.6.1 超链接下载	(144)
4.6.2 使用文件输出流的方式下载	(145)
4.6.3 两种下载方式的比较	(148)
4.7 用组件实现文件的上传与下载	(149)
4.7.1 JspSmartUpload组件的安装	(150)
4.7.2 文件上传	(155)
4.7.3 文件下载	(160)
4.8 分页读取文件	(164)
4.9 本章小结	(167)
第5章 JSP数据库连接技术	(168)
5.1 数据库的安装与创建	(168)
5.1.1 下载及安装MySQL	(168)
5.1.2 创建数据库	(176)
5.2 JDBC技术	(180)
5.2.1 JDBC简介	(180)
5.2.2 JDBC的四种驱动	(181)
5.2.3 JDBC接口	(183)
5.2.4 通过JDBC—ODBC桥连接数据库	(196)
5.3 查询记录	(199)
5.3.1 顺序查询	(200)
5.3.2 游动查询	(204)
5.3.3 随机查询	(207)
5.3.4 参数查询	(210)
5.3.5 排序查询	(218)
5.4 添加记录	(221)
5.5 更新记录	(225)
5.6 删除记录	(232)
5.7 本章小结	(234)

第6章 JavaBean技术	(235)
6.1 什么是JavaBean	(235)
6.1.1 JavaBean的概念	(236)
6.1.2 JavaBean的优点	(236)
6.1.3 编写第一个JavaBean	(236)
6.2 编译和部署JavaBean	(238)
6.2.1 编译JavaBean	(238)
6.2.2 关于MANIFEST.MF文件	(240)
6.2.3 部署JavaBean	(242)
6.3 在JSP中应用JavaBean	(242)
6.3.1 useBean动作指令标签	(243)
6.3.2 JavaBean的作用范围	(243)
6.3.3 JSP与JavaBean结合的简单例子	(244)
6.3.4 设置JavaBean属性	(246)
6.3.5 获得JavaBean属性	(247)
6.4 数据库操作JavaBean	(248)
6.5 本章小结	(252)
第7章 Servlet技术	(253)
7.1 Servlet概述	(253)
7.1.1 Servlet的概念	(253)
7.1.2 Servlet的实现过程	(253)
7.1.3 Servlet的特点	(254)
7.2 Servlet基本结构	(255)
7.2.1 GenericServlet类	(255)
7.2.2 HttpServlet类	(255)
7.2.3 编写Servlet类的步骤	(257)
7.2.4 Servlet的基本结构	(257)
7.2.5 简单Servlet应用示例	(258)
7.3 处理表单数据	(261)
7.4 读取HTTP请求头	(269)
7.4.1 HTTP请求头概述	(269)
7.4.2 在Servlet中读取请求头	(270)

7.4.3 读取HTTP请求头实例	(270)
7.5 HTTP应答状态	(272)
7.5.1 状态代码概述	(272)
7.5.2 设置状态代码	(273)
7.5.3 HTTP 1.1状态代码及其含义	(273)
7.5.4 HTTP应答状态举例	(276)
7.6 设置HTTP应答头	(280)
7.6.1 HTTP应答头概述	(280)
7.6.2 常见应答头及其含义	(281)
7.6.3 HTTP应答头实例	(282)
7.7 本章小结	(286)
第8章 MVC模式和SSH技术	(287)
8.1 JSP设计模式	(287)
8.1.1 模式一(Model1)	(288)
8.1.2 模式二(Model2)	(288)
8.2 MVC结构	(289)
8.2.1 MVC简介	(289)
8.2.2 框架内容	(289)
8.2.3 框架模式和设计模式的区别	(290)
8.3 SSH技术	(291)
8.3.1 什么是SSH	(291)
8.3.2 Struts	(292)
8.3.3 Spring	(293)
8.3.4 Hibernate	(294)
8.4 本章小结	(295)
主要参考文献	(296)

第1章 JSP 技术概述

JSP(Java Server Pages)是由 Sun Microsystems 公司倡导、许多公司参与共同建立的一种动态网页技术标准。JSP 技术有点类似 ASP 技术,它是在传统的网页 HTML 文件(*.htm,*.html)中插入 Java 程序段(Scriptlet)和 JSP 标记(tag),从而形成 JSP 文件,后缀名为(*.jsp)。用 JSP 开发的 Web 应用是跨平台的,既能在 Linux 下运行,也能在其他操作系统上运行。本章主要对 JSP 的一些基本概念和 JSP 的开发环境进行介绍。

1.1 JSP 介绍

1.1.1 什么是 JSP

JSP 技术使用 Java 编程语言编写类 XML 的 tags 和 scriptlets,来封装产生动态网页的处理逻辑。网页还能通过 tags 和 scriptlets 访问存在于服务端的资源的应用逻辑。JSP 将网页逻辑与网页设计和显示分离,支持可重用的基于组件的设计,使基于 Web 的应用程序的开发变得迅速和容易。

JSP 是英文 Java Server Pages 的缩写,它是 Sun 公司首创的一种新型动态网页编程技术。JSP 不但完全拥有 Servlet 技术的优点,而且 JSP 源代码能够直接嵌入 HTML 之中,使得程序的开发更为简化和快速。由于 JSP 技术是从 Java 编程语言中的 Servlet 技术扩展而来,JSP 也就当然实现了 Servlet 所具有的动态页面与静态页面分离以及程序代码与硬件平台无关性。

Web 服务器在遇到访问 JSP 网页的请求时,首先执行其中的程序段,然后将执行结果连同 JSP 文件中的 HTML 代码一起返回给客户。插入的 Java 程序段可以操作数据库、重新定向网页等,以实现建立动态网页所需要的功能。JSP 与 JavaServlet 一样,是在

服务器端执行的,通常返回给客户端的就是一个 HTML 文本,因此客户端只要有浏览器就能浏览。

1.1.2 JSP 与其他动态网页技术的比较

现在主流的网站开发语言主要有 ASP、PHP、ASP.net、JSP 等。网页从开始简单的 HTML 到复杂的服务语言,走过了 10 多个年头,各种技术层出不穷,每一种主流技术也在不断翻新其版本,现在分析一下各种语言的区别、优势、劣势、开发注意事项。

HTML 当然是网页最基本的语言,每一个服务器语言都需要它的支持,要学习,这个肯定是开始,也是大家学习网页技术的基础,必须掌握。

(1) ASP 和 ASP.NET 技术

ASP 是微软(Microsoft)所开发的一种后台脚本语言,它的语法和 Visual BASIC 类似,可以像 SSI(Server Side Include)那样把后台脚本代码内嵌到 HTML 页面中。虽然 ASP 简单易用,但是它自身存在着许多缺陷,最重要的就是安全性问题。目前在微软的 .net 战略中新推出的 ASP.net 借鉴了 Java 技术的优点,使用 C Sharp (C#) 语言作为 ASP.net 的推荐语言,同时改进了以前 ASP 的安全性差等缺点。但是,使用 ASP/ASP.net 仍有一定的局限性,因为从某种角度来说它们只能在微软的 Windows NT/2000/XP + IIS 的服务器平台上良好运行(虽然像 ChiliSoft 提供了在 UNIX/Linux 上运行 ASP 的解决方案,但是目前 ASP 在 UNIX/Linux 上的应用可以说几乎为 0)。所以平台的局限性和 ASP 自身的安全性限制了 ASP 的广泛应用。

ASP 在执行的时候,是由 IIS 调用程序引擎,解释执行嵌在 HTML 中的 ASP 代码,最终将结果和原来的 HTML 一同送往客户端。

ASP.net 是 Microsoft.net 的一部分,作为战略产品,不仅仅是 Active Server Page (ASP) 的下一个版本;它还提供了一个统一的 Web 开发模型,其中包括开发人员生成企业级 Web 应用程序所需的各种服务。ASP.NET 的语法在很大程度上与 ASP 兼容,同时它还提供一种新的编程模型和结构,可生成伸缩性和稳定性更好的应用程序,并提供更好的安全保护。可以通过在现有 ASP 应用程序中逐渐添加 ASP.NET 功能,随时增强 ASP 应用程序的功能。ASP.NET 是一个已编译的、基于 NET 的环境,可以用任何与 .NET 兼容的语言(包括 Visual Basic.NET、C# 和 JScript .NET.)创作应用程序。另外,任何 ASP.NET 应用程序都可以使用整个 .NET 框架。开发人员可以方便地获得这些技术的优点,其中包括托管的公共语言运行库环境、类型安全、继承等等。ASP.NET 可以无缝地与 WYSIWYG HTML 编辑器和其他编程工具(包括 Microsoft Visual Studio.NET)一起工作。这不仅使得 Web 开发更加方

便,而且还能提供这些工具必须提供的所有优点,包括开发人员可以用来将服务器控件拖放到 Web 页的 GUI 和完全集成的调试支持。微软为 ASP.net 设计了这样一些策略:易于写出结构清晰的代码、代码易于重用和共享、可用编译类语言编写等等,目的是让程序员更容易开发出 Web 应用,满足计算向 Web 转移的战略需要。

(2) PHP 技术

PHP 的全名非常有趣,它是一个巢状的缩写名称——“PHP: Hypertext Preprocessor”,打开缩写还是缩写。PHP 是一种 HTML 内嵌式的语言(就像上面讲的 ASP 那样)。而 PHP 独特的语法混合了 C、Java、Perl 以及 PHP 式的新语法。它可以比 CGI 或者 Perl 更快速地执行动态网页。

PHP 的源代码完全公开,在 Open Source 意识抬头的今天,它更是这方面的中流砥柱。不断地有新的函数库加入以及不停地更新,使得 PHP 无论在 UNIX 或是 Win32 平台上都可以有更多新的功能。它提供丰富的函数,使得在程式设计方面有更好的资源。目前 PHP 的最新版本为 5.2.17,它可以在 Win32 以及 UNIX/Linux 等几乎所有的平台上良好工作。PHP 在 4.0 版后使用了全新的 Zend 引擎,其在最佳化之后的效率,比传统 CGI 或者 ASP 等技术有了更好的表现。

平台无关性是 PHP 的最大优点,但是在优点的背后,还是有一些小小的缺点的。如果在 PHP 中不使用 ODBC,而用其自带的数据库函数(这样的效率要比使用 ODBC 高)来连接数据库的话,使用不同的数据库,PHP 的函数名不能统一。这样,使得程序的移植变得有些麻烦。不过,作为目前应用最为广泛的一种后台语言,PHP 的优点还是非常明显的。

(3) JSP 技术

JSP 和 Servlet 要放在一起讲,是因为它们都是 Sun 公司的 J2EE (Java 2 platform Enterprise Edition)应用体系的一部分。

Servlet 的形式和 CGI 差不多,它的 HTML 代码和后台程序是分开的。它们的启动原理也差不多,都是服务器接到客户端的请求后,进行应答。不同的是,CGI 对每个客户请求都打开一个进程 (Process),而 Servlet 却在响应第一个请求的时候被载入,一旦 Servlet 被载入,便处于已执行状态。对于以后其他用户的请求,它并不打开进程,而是打开一个线程(Thread),将结果发送给客户。由于线程与线程之间可以通过生成自己的父线程(Parent Thread)来实现资源共享,这样就减轻了服务器的负担,所以,Java Servlet 可以用来做大规模的应用服务。

虽然在形式上 JSP 和 ASP 或 PHP 看上去很相似——都可以被内嵌在 HTML 代码中。但是,它的执行方式和 ASP 或 PHP 完全不同。在 JSP 被执行的时候,JSP 文件被 JSP 解释器(JSP Parser)转换成 Servlet 代码,然后 Servlet 代码被 Java 编译器编译成 .class

字节文件,这样就由生成的 Servlet 来对客户端应答。所以,JSP 可以看做是 Servlet 的脚本语言(Script Language)版。

由于 JSP/Servlet 都是基于 Java 的,所以它们也有 Java 语言的最大优点——平台无关性,也就是所谓的“一次编写,随处运行(WORA - Write Once, Run Anywhere)”。除了这个优点,JSP/Servlet 的效率以及安全性也是相当惊人的。因此,JSP/Servlet 虽然在国内目前的应用并不广泛,但是其前途不可限量。

在调试 JSP 代码时,如果程序出错,JSP 服务器会返回出错信息,并在浏览器中显示。这时,由于 JSP 是先被转换成 Servlet 后再运行的,所以,浏览器中所显示的代码出错的行数并不是 JSP 源代码的行数,而是指转换后的 Servlet 程序代码的行数。这给调试代码带来一定困难。所以,在排除错误时,可以采取分段排除的方法(在可能出错的代码前后输出一些字符串,用字符串是否被输出来确定代码段从哪里开始出错),逐步缩小出错代码段的范围,最终确定错误代码的位置。

如今主流的 Web 服务器软件主要由 IIS 或 Apache 组成。IIS 支持 ASP 且只能在 Windows 平台下运行,Apache 支持 PHP,JSP 可运行于多种平台,虽然 Apache 是世界使用排名第一的 Web 服务器平台,但是众所周知,Windows 以易用而出名,也因此占据不少的服务器市场。

1.2 JSP 系统环境的安装与配置

编写 JSP 程序的第一步就是要搭建环境,本节将引导大家一步一步搭建一个 JSP 运行的基本环境。这是后续章节学习的基础,也是初学者要学习的基础知识。

一个服务器上可以有很多基于 JSP 的 Web 应用程序,以满足各种不同的应用需求。而这些应用程序需要一个统一的软件来管理和运行,这个软件就称为 JSP 引擎。安装 JSP 引擎的计算机称做支持 JSP 的 Web 服务器。

目前比较常用的 JSP 引擎主要有 Tomcat、WebLogic、WebSphere 等,其中 Tomcat 是应用最为广泛的一个,本书以 Tomcat 为例来介绍 JSP 运行环境的搭建。

1.2.1 安装和配置 JDK

(1) 安装 JDK

安装 Tomcat 之前,首先要安装 JDK。如果没有 JDK,可以从网站 <http://www.oracle>.

com/technetwork/java/javase/downloads/index.html 上免费下载。

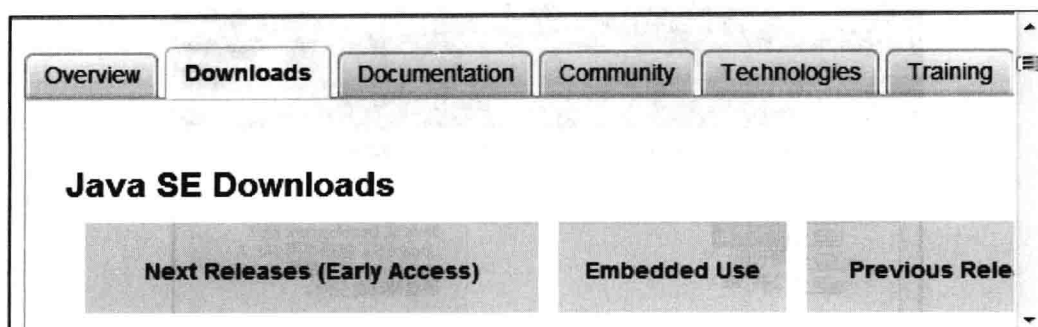


图 1-1 JDK 下载页面

目前最新的 JDK 版本是 JDK1.7, 下载地址:

<http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1637583.html>

下载完成后双击下载的 JDK 安装程序 jdk-7u5-windows-i586.exe, 进入安装界面, 如图 1-2 所示。



图 1-2 JDK 安装向导

安装过程中注意安装路径的选择,如图 1-3 所示。

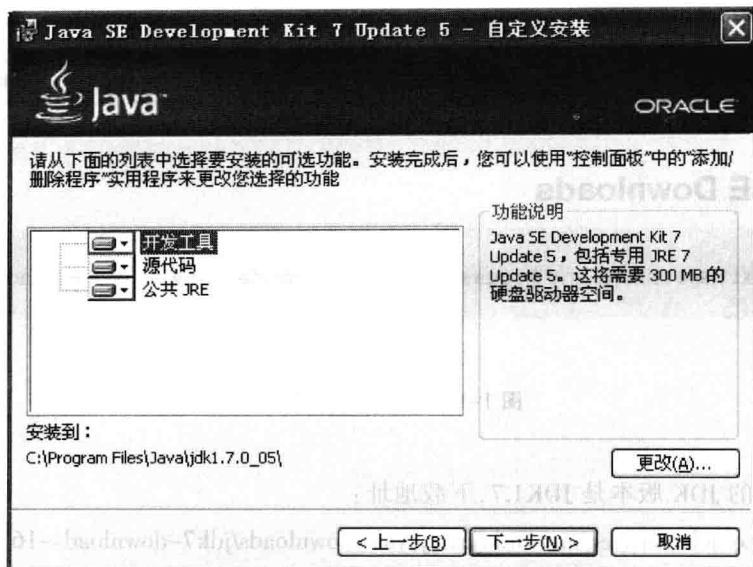


图 1-3 JDK 安装路径选择

其余选项都选择默认即可,安装完成后显示如图 1-4。

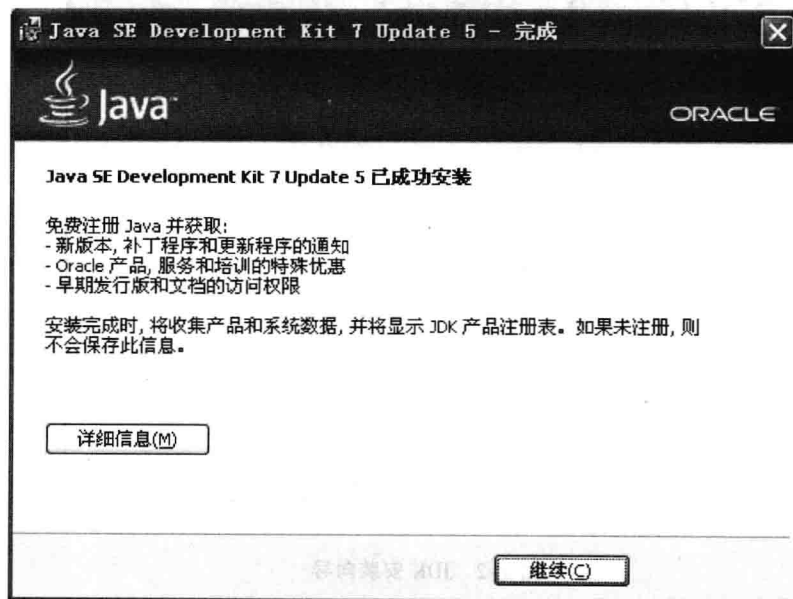


图 1-4 JDK 安装成功