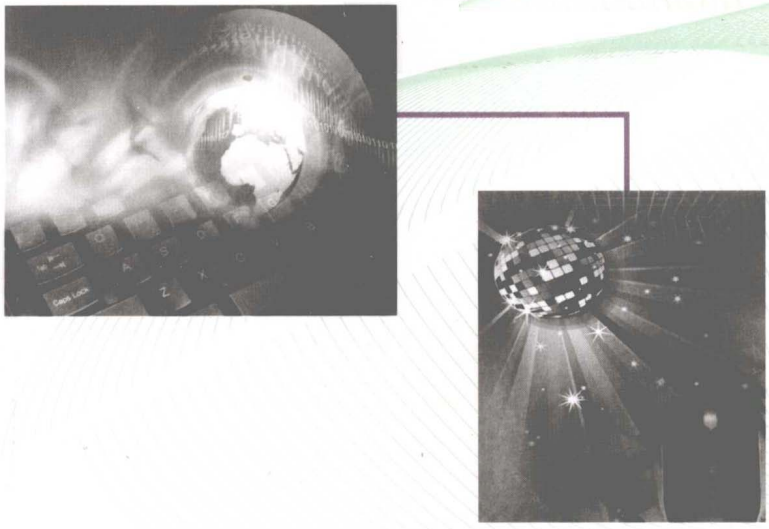
高等学校规划教材·计算机实用软件应用系列教程

JSP Web 应用程序开发教程

主　编　杨占胜

副主编　刘士彩　王　鸽　任国强



西北工业大学出版社

高等学校规划教材·计算机实用软件应用系列教程

JSP Web 应用程序开发教程

主　编　杨占胜

副主编　刘士彩　王　鸽　任国强

西北工业大学出版社

【内容简介】本书分 4 篇 16 章，系统地介绍了 Tomcat 服务器的使用、JSP 的基本语法、JSP 的内置对象、Servlet 技术、Java Bean 的使用、JDBC 数据库应用开发等 JSP 基本技术；Servlet 事件监听器、过滤器、表达式语言、自定义标签、标准标签库（JSTL）等 JSP 高级技术；以及使用 JSP 进行 Web 应用程序开发中的常见问题和常用组件：页面间数据的传递、JSP 中文问题、日志组件、文件上传组件、安全设置等。本书内容丰富，突出应用，例程详尽，对 JSP 中的疑难点进行了辨析；讲解力求简洁深入、例程精练典型，是系统地学习 JSP 技术的教材和参考手册。

本书可作为高等学校本、专科的教材及各类培训班的教材，也可供从事计算机应用和开发的各类人员学习参考。

图书在版编目（CIP）数据

JSP Web 应用程序开发教程/杨占胜主编；刘士彩，王鸽，任国强编．—西安：西北工业大学出版社，2010.8

高等学校规划教材·计算机实用软件应用系列教程

ISBN 978-7-5612-2882-1

Ⅰ．①J…　Ⅱ．①杨…②刘…③王…④任…　Ⅲ．①JAVA 语言—主页制作—程序设计—高等学校—教材　Ⅳ．①TP393.092

中国版本图书馆 CIP 数据核字（2010）第 158527 号

出版发行：西北工业大学出版社

通信地址：西安市友谊西路 127 号　　**邮编**：710072

电　　话：（029）88493844　88491757

网　　址：www.nwpup.com

电子邮箱：computer@nwpup.com

印 刷 者：陕西兴平报社印刷厂

开　　本：787 mm×1 092 mm　1/16

印　　张：24.75

字　　数：666 千字

版　　次：2010 年 8 月第 1 版　　2010 年 8 月第 1 次印刷

定　　价：40.00 元

前　言

程序设计可分为系统程序设计和应用程序设计，系统程序如操作系统、编译程序、数据库管理系统、驱动程序等。应用程序按应用范围有多媒体程序、网络程序、数据库程序等；按其运行方式有：命令方式程序、图形用户界面（窗口）程序、Web 应用程序、手机及 PDA 程序等。命令方式程序是我们在学习 C、C++、汇编、Java 等各种计算机语言时常遇到的例子和练习程序，这类程序通过在命令方式下的输入输出与用户进行交互，人—机界面单一，设计较为简单，实际应用中已很少将其作为单独的一类程序使用。图形用户界面（GUI）程序是我们最常见的，因为 Windows 是一个图形用户界面的操作系统，其中的大部分程序都是图形界面的。这类程序是事件驱动的，以菜单、工具栏、对话框等窗体元素作为人—机交互的界面。如果单纯使用操作系统的 API，用任何一种语言设计一个图形界面的程序都是很复杂、很低效的，即使用面向对象的程序设计语言，再辅以类库也还是有一定的复杂性。所幸的是组件技术的出现，大大地降低了窗口程序的设计难度；再加上功能强大的集成开发环境（IDE），使窗口程序设计达到了简单高效的地步。手机及 PDA 程序目前很热门，这类程序与一般的计算机程序原理基本一致，只是运行环境受限，比如 CPU 的运算速度、存储器的容量等，一般使用专有类库及 API 进行设计。Internet 的迅速普及，电子商务的广泛应用，促使 Web 应用程序的爆炸式发展，Web 程序已经成为企业应用的主要形式。在 Web 应用程序流行之前，有一种基于客户机/服务器（C/S）的网络程序，这类程序都有一个专用的服务器程序和一个客户端程序，双方通过底层的 TCP/IP 协议通信。编程时主要使用 Socket 技术，Socket 是操作系统 TCP/IP API 的高层抽象。C/S 程序的缺点是每个客户端必须安装专用的客户程序，在使用的方便性和安全性方面存在很大的问题，使用底层的传输协议，常被防火墙拦截，在安全性要求高的领域不能应用。Web 应用程序也是基于客户机/服务器模式的，但它的服务器统一为 Web 服务器，客户机统一为浏览器，所以又称为 B/S（Browser/Server）结构的程序。B/S 程序使用 HTTP 协议进行通信，利用 Web 服务器和浏览器的安全机制，克服了 C/S 程序的缺点。Web 应用程序是一个服务器端动态的网站，它通过浏览器与用户交互，最终返回给用户的是一个 HTML 文档，所以学习 Web 应用程序开发，需要“Web 技术基础”方面的知识：HTML，JavaScript，CSS；JSP Web 应用程序使用 Java 语言编写服务器端动态程序，所以又需要“Java 语言程序设计”方面的知识；Web 应用程序通常要访问数据库，数据库访问是 Web 应用程序设计的重点和难点，所以还需要“数据库基础与应用”方面的知识，如结构化查询语言（SQL）、数据库的安全机制与权限设置等，JSP 中的大量配置文件都使用 XML 格式，所以也需要“XML 基础与应用”方面的知识。本书介绍的 JSP Web 应用程序开发技术是以上述先修课程的知识为基础的。

JSP 应用开发涉及的软件技术多，综合性强。JSP 技术本身的知识也较繁杂，各知识点之间相互关联，如自定义标签中会用到表达式语言（EL），而 EL 函数又会用到自定义标签中的标签库描述。要系统地介绍 JSP 技术体系，合理地组织和编排 JSP 的知识结构很重要。在 JSP 技术的发展过程中，是先有 Servlet，后有 JSP 的。介绍 JSP 技术的教材主要有两种顺序，一种是先介绍

JSP 基本语法，然后介绍 Servlet；一种是先介绍 Servlet，后引入 JSP。鉴于 Servlet 的复杂性和几年来的教学实践，本书在结构安排上，从动态网页的发展入手引入 JSP，然后介绍 Servlet，但将 Servlet 提到了 Java Bean 之前。在 JSP 高级技术部分，由于表达式语言较为简单，标准标签库（JSTL）使用到了表达式语言，而自定义标签与 JSTL 应有连贯性，所以将表达式语言提到了自定义标签之前。在内容方面，本书力求对各种技术进行简练的、实质性的解释，如指令是对 JSP 解析器环境的设置；动作是服务器端的动态标记，JSP 解析器调用特定的方法处理相应的动作标记；内置对象是服务器提供的 API 类库；Servlet 是一个实现了特定接口，运行在服务器环境中的特殊 Java 类；监听器是 Web 方式的 Java 事件处理机制；过滤器是对 HTTP 请求的预处理和后处理；分页显示是对数据的再次选择等。JSP 语法中有些概念容易混淆且较难理解，如 page 指令中有关页面编码的 contentType 属性与 pageEncoding 属性；声明中的变量定义与 Scriptlet 中变量的定义；include 指令与 include 动作；forward 动作转向与 response 内置对象 sendRedirect 方法转向；request 内置对象的 getParameter 方法与 getAttribute 方法等，本书都进行了详细的辨析。对于 JSP 技术中的两大难点，数据库访问和自定义标签，本书重点进行了讲解。数据库访问部分按访问方式和要访问的数据库类型，分别安排了大量示例，并对各种可能出现的问题列出了注意事项。阐述自定义标签中的属性时，提出了属性在何处声明、何处定义、何处赋值、何处使用几个问题来帮助理解。其中数据分页、中文乱码、容器可控安全性等章节写得较有新意。书中的例程主要来源于教学实践，在本书的写作过程中，又重新进行了选择设计，尽可能使每个程序都很典型。通过阅读源代码是学习程序设计语言的最有效途径，为了鼓励和方便读者阅读源程序，对于书中各个例程所代表的知识点都作了详细的注释。所有的程序都在 Tomcat 6.0 中进行了调试运行。

本书是按照 21 世纪高等学校计算机规划教材及 Java 系列应用型教材的要求编写的。临沂师范学院刘士彩编写了第 1，2，3，5 章；临沂师范学院杨占胜编写了第 6，8，9，12，13，14 章；山东科技大学王鸽编写了第 4，7，10，11 章；山东科技大学任国强编写了第 15，16 章。临沂师范学院信息学院 Java Web 应用程序开发课程团队的聂志强、李信利、何淑庆、张雪飞、许作萍等，参与了实验指导书的编写，并对部分章节提出了许多宝贵的意见和建议。全书由临沂师范学院杨占胜统稿。临沂师范学院信息学院党委书记赵铭建教授对本书编写给予了高度的关心和热情帮助，并对全书进行了审核。

为方便读者阅读和示例验证，作者将提供本书所有例程的源代码和支持软件，并提供课件下载（E-mail：zsyong@sohu.com）。

由于本书覆盖面宽，篇幅紧凑，加之编者水平有限，书中难免存在不妥之处，诚恳读者不吝指正。

编　者

2010 年 5 月

目　录

第 1 篇　JSP 基本技术

第 2 篇 JSP 应用开发专题

第 3 篇 JSP 高级技术

第 4 篇 JSP 常用组件

第 1 篇

JSP 基本技术

第 1 章　JSP 运行环境

1.1　动态网页技术

Internet 的传统应用有远程登录（Telnet）、文件传输（FTP）、Web 应用、电子邮件（E-mail）、网络聊天（NetChat）、网络新闻（NetNews）等，其中电子邮件应用较为广泛，而目前 Web 应用在 Internet 上也是比较流行的。Web 是 World Wide Web（WWW）环球信息网或万维网的简称，是一张附着在 Internet 上的覆盖全球的信息“蜘蛛网”，镶嵌着无数以超文本形式存在的信息。Web 成为人们共享信息的主要手段，WWW 几乎成为 Internet 的代名词。Web 应用由 Web 服务器发布，客户端用浏览器（如 IE，Navigator 等）进行浏览，使用 HTTP 通过 Internet 进行信息传输。

1.1.1　Web 发展的三个阶段

1．静态网页

静态网页是早期单纯以 HTML 编写的网页。静态页面的请求处理过程比较简单，如图 1-1 所示。

图 1-1　静态网页的请求处理过程

静态网页以 HTML 语言编写，保存在 Web 服务器上，客户端浏览器根据用户输入的网址向服务器发出请求，服务器接受浏览器的请求后，查找所请求的页面文件，并进行权限验证，如果验证通过，将该网页发回给浏览器显示。请求与应答在网络上使用的传输协议为 HTTP。

网页的请求处理过程有两个重要的参与者，客户端的浏览器和服务端的 Web 服务器，可以说这两个软件是 Web 技术的核心体现。浏览器的主要功能是发起 HTTP 请求，解析与显示 HTML 网页。进一步讲，浏览器是客户端应用层协议 HTTP 的实现者和 HTML 标记解析器。目前常用的浏览器有 Microsoft（微软）的 IE（Internet Explorer）、Netscape（网景）的 NN（Netscape Navigator）、Mozilla 基金会的 Firefox（火狐狸）、傲游的 Maxthon、凤凰工作室（Phoenix Studio）的 The World（世界之窗）、腾讯的 TT（Tencent Traveler）。Web 服务器又称 HTTP 服务器，是典型的实现应用层协议 HTTP 的软件。Web 服务器的主要功能是处理 HTTP 请求，管理 Web 页面。评价 Web 服务器的因素有承载力、效率、稳定性、安全性、日志和统计、虚拟主机、代理服务器、缓存服务和集成应用程序等。常见的 Web 服务器有 Microsoft（微软）的 IIS（Internet Information Server），运行于 Windows 平台，支持 ASP 及 ASP.NET；The Apache Software Foundation 的 Apache，最流行的 HTTP 服务器，早期运行于 Unix 类系统上，目前也有运行于 Windows 系统的版本；IBM 的 Websphear，功能完善、开放的大型 Web 应用程序服务器，支持 JSP；W3C（World Wide Web Consortium）的 Jagsaw；使用 Java 语言，采用完全的面向对象架构，以最新的 Web 技术协议为标准设计的开放源码服务器，支持 JSP；AOL（America On Line 美国在线）的 AOL Server，高效能、高稳定性、高扩充性的开源 Web 服务器，运行于 Unix 类系统平台。Kerio 的 WebSTAR，Kerio 是专业生产防火墙的厂家，WebSTAR 运行于 Apple

（苹果）Mac OS 平台，其特点是高安全性。

2．客户端动态网页

客户端动态网页是以 DHTML 和其他客户端交互技术编写的网页。DHTML（Dynamic HTML）是一种通过结合 HTML、客户端脚本语言（JavaScript、VBScript）、层叠样式表（CSS）和文档对象模型（DOM）来创建动态网页内容的技术总称，其他客户端交互技术有 Flash，ActiveX，Java Applet 等。客户端动态技术需浏览器的支持，对浏览器的可扩展功能有了更高的要求，此时的浏览器不仅仅是一个标记解析器，而且已嵌入了脚本解析等各种功能模块。客户端动态网页的请求处理过程基本与早期的静态网页一致。

3．Web 应用程序

Web 应用程序即服务器端动态网页。浏览器请求服务器端动态的网页时，服务器须调用相应的解析器对页面进行处理，一般是运行其中嵌入的程序代码，将处理结果返回给浏览器显示，其处理过程如图 1-2 所示。

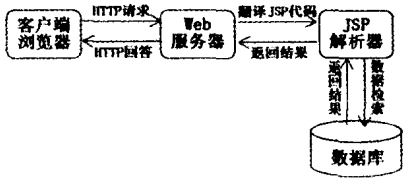


图 1-2 动态网页的请求处理过程

服务器端动态技术需要 Web 服务器的支持，对 Web 服务器的技术发展提出了新的要求，不同的服务器端动态网页设计技术要有特定的 Web 服务器，或者在 Web 服务器上安装特定的功能模块来支持。

1.1.2 Web 应用程序开发的三个阶段

1．代码输出

最初的服务器端动态网页技术是 CGI（Common Gateway Interface，通用网关接口），CGI 是 Web 服务器支持的允许客户端浏览器请求调用服务器上特定程序的技术，这个程序又称为 CGI 程序。它接受客户端提交的数据，CGI 描述了浏览器和所请求程序之间传输数据的标准，一般输出 HTML 代码给浏览器，所以称这样的动态网页设计为代码输出。CGI 程序以编译后可执行代码的形式发布，所以 CGI 程序是语言独立的，可以用任何编程语言实现，Perl 是使用最广泛的 CGI 程序设计语言。

2．代码混合

CGI 程序要输出各种 HTML 标记，编程很复杂。所以发展了在 HTML 文件中使用服务器端标记嵌入服务器端代码的网页，请求这些网页时，服务器调用特定的解析器（程序）对其进行处理，将其中的 HTML 标记直接输出，执行其中的服务端代码，在网页中代码所在处输出运行的结果。这样的动态网页中 HTML 标记与服务器端代码混合在一起，所以称为代码混合形式的 Web 应用程序开发。常用的代码混合动态网页编程技术有 ASP，PHP，JSP 三种。

（1）ASP（Active Server Page）：ASP 是一种服务器端脚本编写环境，可以用来创建和运行动态网页或 Web 应用程序。ASP 网页可以包含 HTML 标记、普通文本、脚本命令以及 COM 组件等，ASP 的强大不在于它的脚本语言，而在于它后台的 COM 组件，这些组件无限地扩充了 ASP 的功能。ASP

已经升级为 ASP.NET，ASP.NET 采用组件技术，以代码分离的方式来开发 Web 应用程序，已完全不同于 ASP。

（2）PHP（PHP Hypertext Process，PHP 原指 Personal Home Page）：是一种跨平台的服务器端嵌入式脚本语言。它大量地借用 C，Java 和 Perl 语言的语法，并结合 PHP 自己的特性，使 Web 开发者能够快速地写出动态页面。它支持绝大部分数据库，而且是完全免费的。

（3）JSP（Java Server Page）：是基于 Java 语言的一种 Web 应用开发技术，利用这一技术可以建立安全、跨平台的先进动态网站。利用 JSP 技术创建的 Web 应用程序，可以实现动态页面与静态页面的分离。与其他 Web 技术相比，JSP 具有跨平台、编译后运行等特点。

3．代码分离

代码分离指页面的程序逻辑与表现形式相互分离的动态网页设计机制，ASP.NET 以服务器控件和页面模型抽象方式实现代码分离，JSP 以 Bean 和自定义标签的方式实现代码分离，以组件方式构建 Java Web 应用程序的技术为 JSF（Java Server Faces），其他语言大多以模板方式实现代码分离，如 PHP Template。动态网页设计技术正在迅速发展中，从代码分离、Web 组件到 Web Service。

1.1.3 HTTP 请求地址

Web 服务器发布的资源在 Internet 上是以 URL（Uniform Resource Locator，统一资源定位器）标识的，客户端浏览器请求时，必须在地址栏中输入 URL，即必须提供所请求网页等目标的位置。URL 的格式为：

protocol://hostname[:port]/website/path/ [file][?query][#fragment]

协议://主机名:端口号/网站名称/目录/文件名?查询参数#信息片断

例如：http://www.lytu.edu.cn:80/chpage/index.html?str=abc#a1

protocol（协议）：http, ftp, file, gopher, https, mailto, news。

hostname（主机名）：机器名+域名+域树+域林。

port（端口号）：http 的默认端口为 80，可以省略。

其他常用协议的默认端口：telnet: 23，ftp: 21，smtp: 25，pop3: 110，dns: 53。

website（网站名称）：Web 应用程序上下文，虚拟目录名，网站根目录。

path/file（目录/文件）：网页相对于网站根目录的子目录和文件名。

?query（查询参数）：?名 1=值 1&名 2=值 2。

#fragment（信息片断）：网页锚点，使用<a>标记 name 属性在网页内部定义的位置标记。

1.1.4 HTTP 状态码

Web 服务器对客户端的响应一般包含一个状态行，一些响应报头，一个空行和相应的内容文档。

（1）状态行：状态行由 HTTP 版本，一个状态代码以及一段对应状态代码的简短说明信息组成，表示请求是否被理解或被满足。HTTP 版本由服务器决定。请求被正常响应时，状态码一般由系统自动设置为 200。也可以在页面中用代码设置状态码，说明信息也可自定义。

（2）一些响应报头（几个应答头）：HTTP 头消息，对应于 HTTP 协议的头部，在大多数情况下，除了 Content-Type 之外的所有应答头都是可选的。

（3）空行：起分隔、标识作用。

（4）内容文档：数据报内容，封装在 HTTP 协议的体内。

下面是一个最简单的应答：

```
HTTP/1.1 200 OK
Content-Type: text/plain

Hello World
```

HTTP 1.1 中的状态码（见表 1-1）分为五大类：

- 100～199 信息性的标示用户应该采取的其他动作。
- 200～299 表示请求成功。
- 300～399 用于那些已经移走的文件，常常包括 Location 报头，指出新的地址。
- 400～499 表明客户引发的错误。
- 500～599 指出由服务器引发的错误。

表 1-1 常见 HTTP 1.1 状态代码以及对应的状态信息和含义

代码	HttpServletResponse 符号常量	信 息	含 义
100	SC_CONTINUE	Continue	继续
101	SC_SWITCHING_PROTOCOLS	Switching Protocols	转换协议
200	SC_OK	OK	一切正常
201	SC_CREATED	Created	创建
202	SC_ACCEPTED	Accepted	接收
203	SC_NON_AUTHORITATIVE_INFORMATION	Non authoritative Information	非授权信息
204	SC_NO_CONTENT	No Content	无内容
205	SC_REST_CONTENT	Reset Content	重置内容
206	SC_PARTIAL_CONTENT	Partial Content	部分内容
300	SC_MULTIPLE_CHOICES	Multiple Choices	多选
301	SC_MOVED_PERMANENTLY	Moved Permanently	永久移动
302	SC_MOVED_PERMANENTLY	Moved Temporarily	暂时移动
304	SC_NOT_MODIFIED	Not Modified	未更改
305	SC_USE_PROXY	Use Proxy	使用代理服务器
400	SC_BAD_REQUEST	Bad Request	错误请求
401	SC_UNAUTHORIZED	Unauthorized	未授权
402	SC_PAYMENT_REQUIRED	Payment Required	要求支付
403	SC_FORBIDDEN	Forbidden	禁止
404	SC_NOT_FOUND	Not Found	未找到
405	SC_METHOD_NOT_ALLOWED	Method Not Allowed	不可用方法
406	SC_NOT_ACCEPTABLE	Not Acceptable	不接受
407	SC_PROXY_AUTHENTICATION_REQUIRED	Proxy Authentication Required	需确认代理服务器
408	SC_REQUEST_TIMEOUT	Request Time Out	请求超时
409	SC_CONFLICT	Conflict	冲突
410	SC_GONE	Gone	离开
411	SC_LENGTH_REQUIRED	Length Required	需要长度
412	SC_PRECONDITION_FAILED	Precondition Failed	预处理失败
413	SC_REQUEST_ENTITY_TOO_LARGE	Request Entity Too Large	请求实体过大
414	SC_REQUEST_URI_TOO_LONG	Request URL Too Large	请求 URL 过长
415	SC_UNSUPPORTED_MEDIA_TYPE	Unsupported Media Type	不支持的媒体类型
500	SC_INTERNAL_SERVER_ERROR	Server Error	服务器错误
501	SC_NOT_IMPLEMENTED	Not Implemented	未执行
502	SC_BAD_GATEWAY	Bad Gateway	网关坏
503	SC_SERVICE_UNAVAILABLE	Out of Resources	超出资源
504	SC_GATEWAY_TIME_OUT	Gateway Time Out	网关超时
505	SC_HTTP_VERSION_NOT_SUPPORTED	HTTP Version Not Supported	不支持 HTTP 版本

在 JSP Web 应用程序开发过程中常遇到的错误是 404 和 500。404 表示无法找到客户端所给地址的任何资源，即没有所请求的页面。这是最简单、最容易解决的错误，主要原因是请求地址错误，此时应仔细核对输入的路径和文件名是否正确，是否存在这样的路径和文件；另一个原因是虚拟路径不起作用，即 Web 应用程序未加载，此时应仔细核对虚拟路径配置标记是否正确，Web 应用程序的配置文件 web.xml 的格式是否正确，也可从服务器控制台显示的信息或日志文件中记录的信息查找原因。500 表示程序逻辑有错误，此时应根据错误提示调试程序代码。有时浏览器中显示不能加载类的错误提示，这是由于页面有错误，服务器在将 JSP 转化为 Servlet 类或进行编译时出现错误，所以服务器无法加载这个类，这种错误信息提示不明确，此时应将浏览器刷新几次，直到出现页面错误信息提示。

1.1.5 JSP 动态网页的处理过程

JSP Web 服务器在处理 JSP 网页时，首先由 JSP 引擎将 JSP 文件转化为 Servlet（一种 Java 类），其次将该 Servlet 编译为.class 文件，然后调用 Servlet 引擎执行 class 文件，输出 HTML 网页发送到客户端的浏览器中显示。其处理过程如图 1-3 所示。

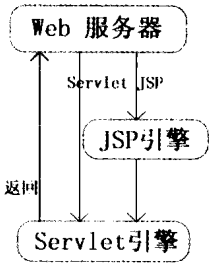


图 1-3 JSP 页面的处理过程

JSP 动态网页需要能够解析处理 JSP 代码的 Web 服务器支持，支持 JSP 的 Web 服务器有 Apache Tomcat，SUN JSWDK (Java Server Web Development Kit)，caucho Resin，W3C Jigsaw，IBM Websphear，BEA Weblogic，Jboss.org Jboss，Allaire Jrun 等。前四个是轻量级的 JSP 服务器，并且是开源软件。后四个是全功能的 Java EE 服务器，除 Jboss 外都是商业软件，其中 Tomcat 是最常用的 JSP Web Server，Jboss 也是通过内置 Tomcat 支持 JSP 的。本书中的 JSP Web 服务器使用 Tomcat。

1.2 Tomcat 服务器的安装与配置

Tomcat 是一个免费的开放源代码的 Servlet 容器，它是 Apache 软件基金会（Apache Software Foundation）的一个顶级项目，由 Apache，Sun 和其他一些公司及个人共同开发而成。由于有了 Sun 的参与和支持，最新的 Servlet 和 JSP 规范总是能在 Tomcat 中得到体现，Tomcat 6 支持最新的 Servlet 2.5 和 JSP 2.1 规范。因为 Tomcat 技术先进，性能稳定，而且免费，因而深受 Java 爱好者的喜爱，并得到了部分软件开发商的认可，成为目前比较流行的 Web 服务器。

Tomcat 需要 Java 运行环境。Tomcat 6.x 需要的 Java SE 版本最低为 5.0。Tomcat 6.x 不再需要 JDK 的支持，只需要安装 JRE 就可以了，这是因为 Tomcat 6.x 使用 Eclipse JDT Java 编译器来编译 JSP 页

面，而不再使用 JDK 中的编译器。Eclipse JDT Java 编译器已经绑定到 Tomcat 的发行版中。如果使用 Tomcat 5.0.x 或者 Tomcat 5.5.x，那么仍然需要安装 JDK。

1.2.1 安装 Java SE

Java SE 可从 http://java.sun.com/javase/downloads/index.jsp 下载，目录的版本是 JDK6。运行所下载的安装程序，按默认设置进行安装，也可自定义路径。

添加环境变量：我的电脑→属性→高级→环境变量→系统变量中添加以下环境变量：

JAVA_HOME=C:\Program Files\Java\jdk1.6.0

CLASSPATH=.;%JAVA_HOME%\lib\dt.jar;%JAVA_HOME%\lib\tools.jar;

在 Path 环境变量中添加%JAVA_HOME%\bin；即

Path=%JAVA_HOME%\bin;原来的内容

用记事本（Notpad）写一个简单的 Java 程序来测试 JDK 是否已安装成功。

```
public class JreTest {
    public static void main(String args[]) {
        System.out.println("Hello! The Java running enviorenment is OK! ");
    }
}
```

将程序保存为文件名是 JreTest.java 的文件。

打开命令提示符窗口（开始→运行 cmd 命令），进入到 JreTest.java 所在目录，键入下面的命令：

```
javac JreTest.java
java JreTest
```

此时若在命令窗口中显示出来 Hello! The Java running enviorenment is OK!，则 JDK 安装成功，环境变量设置正确。若没有显示出该字符串，请仔细检查以上配置是否正确。也可在命令提示符窗口中输入 javac 以及 java 命令直接运行 Java 编译器和虚拟机，如果显示这两个程序的帮助说明，则 JDK 安装成功，环境变量设置正确。Java 编译器与虚拟机易受病毒感染，必须保证机器无病毒才能正常运行。

：要显示记事本程序的扩展名，须将文件夹选项中“隐藏已知文件类型的扩展名”（我的电脑→工具→文件夹选项→查看→隐藏已知文件类型的扩展名）不勾选。

1.2.2 安装 Tomcat

Tomcat 可从 http://tomcat.apache.org 下载。目前的版本为 6.0。有安装版和非安装版两种，注意其文件扩展名分别是.exe 和.zip。

如果使用安装版，运行下载的.exe 文件，按向导提示点击下一步进行安装，安装后可以从开始菜单启动 Tomcat 的服务管理器，此时服务管理器的图标将显示在任务栏右侧的通知区域，右击图标通过其右键菜单来启动、关闭 Tomcat 服务器。也可以运行 Tomcat 的安装目录 bin 子目录下的 tomcat6.exe 或 tomcat6w.exe 启动 Tomcat 服务器。