

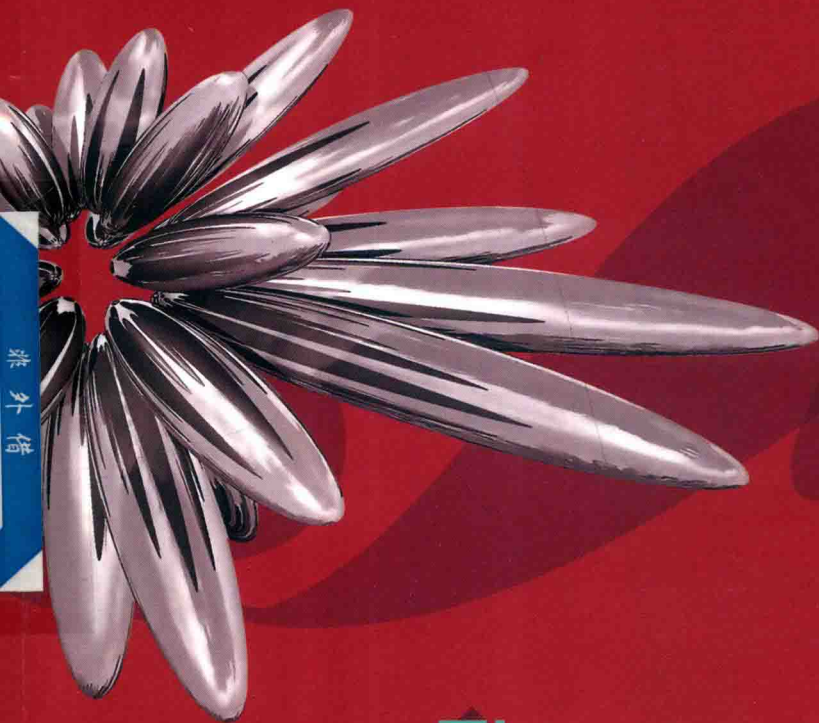


普通高等教育“十三五”规划教材

高等学校计算机规划教材

JSP程序设计 实例教程

◆ 谷志峰 李同伟 主编



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

普通高等教育“十三五”规划教材
高等学校计算机规划教材

JSP 程序设计实例教程

谷志峰 李同伟 主编

琚伟伟 副主编

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书系统地讲解了JSP程序设计所涉及的基本语法,并以在线图书销售平台项目贯穿始终,真正做到项目驱动。全书共9章,内容包括JSP概述、在线图书销售平台项目案例设计、Java数据库编程技术、HTML和CSS简介、JSP基本语法详解、JSP内置对象详解、EL表达式和JSTL标签、MVC设计模式和Servlet技术详解、过滤器和监听器。本书提供配套电子课件。

本书的特色是通俗易懂、案例翔实、项目驱动、体系结构合理、章节设置得当,可作为高等院校计算机及信息工程类专业相关课程的教材,也可作为JSP初学者自学的入门教材。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

JSP 程序设计实例教程 / 谷志峰, 李同伟主编. —北京: 电子工业出版社, 2017.10

普通高等教育“十三五”规划教材

ISBN 978-7-121-32609-7

I. ①J… II. ①谷… ②李… III. ①JAVA 语言—网页制作工具—高等学校—教材 IV. ①TP312
②TP393.092

中国版本图书馆 CIP 数据核字(2017)第 213107 号

策划编辑: 王晓庆

责任编辑: 郝黎明 特约编辑: 张燕虹

印 刷: 三河市君旺印务有限公司

装 订: 三河市君旺印务有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编: 100036

开 本: 787×1092 1/16 印张: 14.25 字数: 364 千字

版 次: 2017 年 10 月第 1 版

印 次: 2017 年 11 月第 1 次印刷

定 价: 39.80 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888, 88258888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式: (010)88254113, wangxq@phei.com.cn。

前 言

在 JavaEE 方向的课程体系中, JSP 程序设计是一门非常重要的课程。JSP 是一种完全与平台无关的开发技术, 它克服了目前 ASP 和 PHP 脚本级执行的缺点。它将极高的运行效率、较短的开发周期、超强的扩展能力、完全开放的技术标准和自由的开发方式等众多的完美特性集于一身。JSP 可以在 Servlet 和 JavaBean 的支持下, 完成功能强大的动态网站程序的开发, 使构造基于 Web 的应用程序更加容易和快捷。

本书共 9 章, 内容包括 JSP 概述、在线图书销售平台项目案例设计、Java 数据库编程技术、HTML 和 CSS 简介、JSP 基本语法详解、JSP 内置对象详解、EL 表达式和 JSTL 标签、MVC 设计模式和 Servlet 技术详解、过滤器和监听器。在各章语法知识点介绍中, 引用相关的案例, 将复杂的知识点寓于案例中, 力求做到案例教学。在每章的最后还附有习题, 用于检验学习效果 and 巩固本章所学内容。

本书的特色在于将“在线图书销售平台”项目融入各个章节, 贯穿全书; 在第 2 章中对该项目进行设计, 在后续章节中利用各章所学知识完成项目的各个功能。这样的设计使得本书的教学真正做到了项目驱动。

本书可作为普通院校计算机及信息工程或相关专业本科生的教材或参考书, 也可供相关领域的读者参考。本书的参考教学时数在 72 学时以内。

本书提供配套电子课件, 请登录华信教育资源网 (<http://www.hxedu.com.cn>) 注册下载, 也可联系本书编辑 (010-88254113, wangxq@phei.com.cn) 索取。

本书由谷志峰、李同伟任主编, 负责全书统稿; 由琚伟伟任副主编。具体分工为: 第 4 章、第 5 章、第 6 章、第 7 章由谷志峰负责编写; 第 1 章、第 2 章、第 3 章由李同伟负责编写; 第 8 章、第 9 章由琚伟伟负责编写。

本书的出版得到了河南科技大学软件学院以及教务处的大力支持, 软件学院的霍华、刘欣亮、叶传奇、张虎老师对本书的编写提出了很多宝贵的意见。在此, 我们一并表示衷心的感谢。

尽管在编写本书过程中, 我们本着科学严谨的态度, 力求精益求精, 但错误、疏忽之处在所难免, 敬请广大读者批评指正。

编 者

目 录

第 1 章 JSP 概述	1
1.1 JSP 简介	1
1.2 JSP 工作原理	1
1.3 JSP 程序体系结构	2
1.3.1 比较 C/S 结构与 B/S 结构	2
1.3.2 三层架构	4
1.3.3 两层架构	5
1.4 搭建 JSP 的运行环境	5
1.4.1 JDK 的安装与配置	5
1.4.2 Tomcat 的安装、运行与目录结构	7
1.4.3 开发工具的选择	11
1.5 第一个 JSP 应用	12
1.5.1 创建 JSP 页面	12
1.5.2 运行 JSP 程序	14
习题 1	16
第 2 章 在线图书销售平台项目案例设计	18
2.1 系统需求分析	18
2.1.1 系统需求及权限分析	18
2.1.2 系统功能详细介绍	18
2.2 数据库设计	23
2.2.1 数据库设计的三大范式	23
2.2.2 数据表结构详细介绍	25
2.3 系统编写要求及分工	29
2.3.1 系统总体架构	29
2.3.2 系统分工及要求	29
习题 2	30
第 3 章 Java 数据库编程技术	31
3.1 常用数据库介绍	31
3.1.1 Oracle 数据库	31
3.1.2 MySQL 数据库	32
3.2 JDBC 概述	33
3.2.1 JDBC 简介	33

3.2.2	JDBC 驱动程序	33
3.3	数据库操作常用接口简介	34
3.3.1	驱动程序接口 Driver	34
3.3.2	驱动程序管理器 DriverManager 类	34
3.3.3	数据库连接接口 Connection	35
3.3.4	执行 SQL 语句接口 Statement	35
3.3.5	执行动态 SQL 语句接口 PreparedStatement	35
3.3.6	执行存储过程接口 CallableStatement	35
3.3.7	访问结果集接口 ResultSet	35
3.4	Java 数据库操作技术	36
3.4.1	加载驱动	36
3.4.2	建立连接	37
3.4.3	执行 SQL 语句	38
3.4.4	获取结果集	41
3.4.5	关闭资源	42
3.5	Dao 模式	42
3.6	Java 单元测试技术	46
3.7	应用实例	49
3.7.1	浏览图书信息功能数据层代码	49
3.7.2	浏览图书明细信息功能数据层代码	51
习题 3		53
第 4 章	HTML 和 CSS 简介	55
4.1	HTML 基础	55
4.1.1	HTML 简介	55
4.1.2	表格	56
4.1.3	表单	59
4.1.4	其他常用标签	62
4.2	CSS 基础	63
4.2.1	CSS 概念及作用	63
4.2.2	CSS 基本语法	63
4.2.3	页面布局及排版	66
4.3	应用实例	68
习题 4		72
第 5 章	JSP 基本语法详解	74
5.1	JSP 程序的基本结构	74
5.2	JSP 指令	74
5.2.1	page 指令	75
5.2.2	include 指令	76

5.2.3	taglib 指令	79
5.3	JSP 脚本程序	79
5.4	JSP 表达式	82
5.5	JSP 声明标识	85
5.6	JSP 注释	86
5.7	JSP 标准动作简介	87
5.7.1	jsp:include 动作	88
5.7.2	jsp:forward 动作	89
5.7.3	操作 JavaBean 所用到的三个标准动作	90
5.8	应用实例	94
5.8.1	完成系统主要功能页面 JSP 模板的创建	94
5.8.2	实现浏览图书类别功能	96
习题 5		99
第 6 章	JSP 内置对象详解	101
6.1	内置对象简介	101
6.2	out 对象	101
6.3	request 对象	103
6.3.1	请求方式简介	103
6.3.2	接收请求参数	104
6.3.3	request 属性管理	112
6.4	response 对象	113
6.4.1	实现重定向页面	113
6.4.2	处理 HTTP 文件头	114
6.4.3	设置输出缓冲区	115
6.5	session 对象	115
6.5.1	session 对象的特点和概念	115
6.5.2	session 对象的常用方法介绍	117
6.6	application 对象	125
6.7	cookie 技术	127
6.7.1	cookie 使用初步	128
6.7.2	cookie 使用进阶	129
6.8	其他内置对象简介	131
6.9	应用实例	132
6.9.1	登录功能	132
6.9.2	注册功能	135
习题 6		139
第 7 章	EL 表达式和 JSTL 标签	142
7.1	EL 表达式	142

7.1.1	EL 表达式的概念及用法	142
7.1.2	EL 隐藏对象	144
7.2	JSTL 入门	147
7.2.1	JSTL 概述	147
7.2.2	JSTL 用法	148
7.3	JSTL 常用标签	148
7.3.1	表达式操作标签	149
7.3.2	条件标签	151
7.3.3	迭代标签	153
7.4	JSTL 其他标签	155
7.4.1	URL 标签相关	155
7.4.2	国际化格式标签简介	156
7.5	应用实例	157
7.5.1	浏览图书信息	157
7.5.2	查询图书信息	160
习题 7		164
第 8 章	MVC 模式和 Servlet 技术详解	165
8.1	MVC 模式	165
8.1.1	JSP 程序开发模式	165
8.1.2	MVC 模式	166
8.2	Servlet 简介	167
8.2.1	Servlet 概述	167
8.2.2	Servlet 生命周期	168
8.3	Servlet 创建及使用	172
8.3.1	Servlet 创建	172
8.3.2	Servlet 实现请求转发和重定向	175
8.3.3	Servlet 接收 get/post 请求	177
8.4	Servlet 获取初始化参数及上下文参数	179
8.4.1	获取初始化参数	179
8.4.2	获取上下文参数	180
8.5	Servlet 获取 JSP 内置对象	181
8.5.1	Servlet 获得 JSP 中的 out 对象	181
8.5.2	Servlet 获得 JSP 中的 request 对象	181
8.5.3	Servlet 获得 JSP 中的 response 对象	182
8.5.4	Servlet 获得 JSP 中的 session 对象	182
8.5.5	Servlet 获得 JSP 中的 application 对象	184
8.6	Servlet 中的异常处理	187
8.7	应用实例	188

8.7.1	购物车添加	191
8.7.2	购物车移除	194
8.7.3	购物车更新	195
习题 8		197
第 9 章	过滤器和监听器	199
9.1	过滤器	199
9.1.1	过滤器概述	199
9.1.2	过滤器的生命周期	199
9.1.3	过滤器的创建和使用	201
9.1.4	过滤器链	203
9.1.5	利用过滤器实现禁用 IP 问题	206
9.2	监听器	207
9.2.1	监听器概述	207
9.2.2	监听器接口简介	208
9.2.3	监听器的创建和使用	209
9.3	过滤器和监听器在 JavaEE 框架中的运用	210
9.4	应用实例	212
习题 9		214
参考文献		215

第1章 JSP 概述

1.1 JSP 简介

早期的网站一般都是由静态网页制作的，静态网页指网页的内容是固定的，不会根据浏览者的不同需求而改变。静态网页一般使用 HTML（超文本标记语言）进行编写。静态网页一般是运行于客户端的程序、网页、插件、组件，显示内容属于静态的，是永远不变的。在信息技术日新月异的今天，静态网页已无法满足人们对信息丰富性和多样性的强烈需求。现在大部分的网页都是动态网页，动态网页是指在接到用户访问要求后动态生成的页面，页面内容会随着访问时间和访问者发生变化，动态网页一般是在服务器端运行的程序、网页等。目前，解决 Web 动态网站的开发技术很多，而 JSP 是目前应用最为广泛的一种。

JSP 的全称是 Java Server Page，是 Sun 公司于 1999 年推出的动态网页技术。利用这一技术可以建立安全、跨平台的 Web 应用程序。JSP 以 Java 编程语言作为脚本语言，JSP 的安全性和跨平台得益于 Java 语言，这是因为 Java 语言具有不依赖于平台、面向对象和安全等优良特性，已经成为网络程序设计的佼佼者。许多和 Java 有关的技术得到了广泛的应用和认可，JSP 技术就是其中之一。

目前，Web 动态网站开发技术有 ASP、PHP、JSP 等，ASP 是微软（Microsoft）公司开发的动态网页语言，只能运行于微软公司的服务器产品和 PWS 之上。在 UNIX 环境下可以通过 ChiliSoft 的插件来支持 ASP，但是 ASP 本身的功能有限，必须通过 ASP+COM 的组合来扩充，UNIX 下的 COM 实现起来非常困难。PHP3.0 可在 Windows、UNIX 和 Linux 的 Web 服务器上正常运行，还支持 IIS 和 Apache 等通用 Web 服务器，用户更换平台时，无须变换 PHP 代码即可使用。与 ASP、PHP 相比，JSP 是一种完全与平台无关的开发技术，完全克服了目前 ASP 和 PHP 的脚本级执行的缺点。它将极高的运行效率、较短的开发周期、超强的扩展能力、完全开放的技术标准和自由的开发方式等众多的完美特性集于一身。JSP 可以在 Servlet 和 JavaBean 的支持下，完成功能强大的动态网站程序的开发，使构造基于 Web 的应用程序更加容易和快捷。

JSP 应该是未来发展的趋势，已广泛地应用于电子商务、电子政务等各个行业的管理应用软件中，一些大的电子商务解决方案提供商都采用 JSP。比较著名的有 IBM 公司的 E-business，它的核心技术采用 JSP 的 Web Sphere。

1.2 JSP 工作原理

当客户端浏览器向服务器发出请求访问一个 JSP 页面时，服务器根据该请求加载相应的 JSP 页面，并对该页面进行编译，然后执行。

JSP 的具体处理过程如下所示。

(1) 客户端通过 Web 浏览器向 JSP 服务器发送请求。

(2) JSP 服务器检查是否已经存在 JSP 页面对应的 Servlet 源代码，若存在则进行第 (3) 步，否则转至 (4)。

(3) JSP 服务器检查 JSP 页面是否有更新修改, 若存在更新修改则进行第(4)步, 否则转(5)。

(4) JSP 服务器将 JSP 代码转译为 Servlet 源代码。

(5) JSP 服务器将 Servlet 源代码编译后执行。

(6) 将产生的结果返回到客户端。

JSP 工作原理如图 1-1 所示。

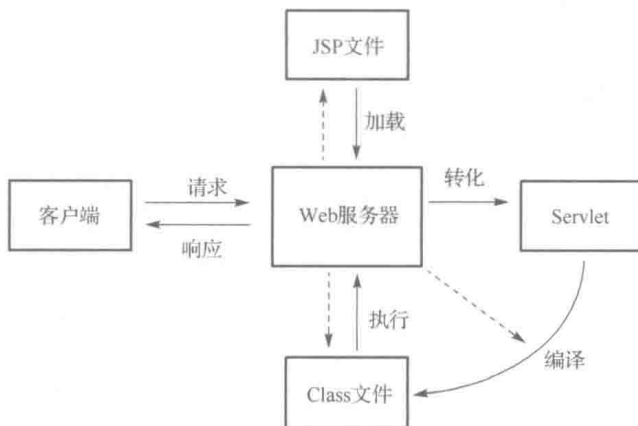


图 1-1 JSP 工作原理

从前面的介绍中, 可以知道服务器在获取了客户端发送的请求后, 依据请求调用相关的 JSP 处理页面。如果该页面是第一次执行, 则需要把 JSP 页面中的代码转换为 Servlet 代码形式, 转换完毕后, 需要转换的 Java 文件即编译成了 class 文件, 编译完成后, 使用 JVM 执行编译过的文件, 并将执行结果返回到客户端。如果该页面不是第一次执行, 就会直接调用该页面的 class 文件执行。

可以看出, JSP 页面的第一次执行需要耗费一些时间, 这些时间是耗费在 JSP 文件到 Servlet 文件转换并编译的过程中, 而再次访问时会感觉快了很多。如果被请求的页面经过修改, 服务器将会重新编译这个文件, 然后执行。

由 JSP 的处理过程可以看到, JSP 和 Servlet 有很深的渊源关系, Servlet 是 JSP 技术的发展前身, 它是 Java 技术对 CGI 编程的回应。Servlet 具有更高的效率, 更容易使用, 功能更强大, 具有更好的可移植性、更节省投资等优点, 因此通常用来做 MVC 模式中的控制器, 但 Servlet 也有自身的不足之处, 即所有响应代码都是通过 Servlet 程序生成的, 如 HTML 标记。一个 Servlet 程序, 其中大量的代码都用于生成这些 HTML 标记响应代码, 只有少部分代码用于数据的处理和响应。并且, 开发 Servlet 程序起点要求较高, Servlet 产生之后, 没有像 PHP 和 ASP 那样, 快速得到应用。因此, Sun 公司在结合了 Servlet 技术和 ASP 技术等特点后, 又推出了 JSP 技术, JSP 技术完全继承了 Servlet 技术的优势, 并具备了一些新的优势。

1.3 JSP 程序体系结构

1.3.1 比较 C/S 结构与 B/S 结构

软件开发中, 目前最常用的体系结构有两种, 即 C/S 结构和 B/S 结构。下面介绍这两种结构, 并进行比较。

1. C/S 结构

C/S 结构即 Client/Server(客户/服务器)结构,它通过将任务合理分配到 Client 端和 Server 端,降低了系统的通信开销,可以充分利用两端硬件环境的优势。C/S 结构的出现是为了解决费用和性能的矛盾,最简单的 C/S 体系结构的数据库应用由两部分组成,即客户应用程序和数据库服务器程序。二者可分别称为前台程序与后台程序。运行数据库服务器程序的机器称为应用服务器,一旦服务器程序被启动,就随时等待响应客户程序发来的请求;客户程序运行在用户自己的计算机上,对应于服务器计算机,可称为客户计算机。当需要对数据库中的数据进行操作时,客户程序就自动地寻找服务器程序,并向其发出请求,服务器程序根据预定的规则做出应答,返回结果。

C/S 结构的优点是能充分发挥客户端 PC 的处理能力,很多工作可以在客户端处理后再提交给服务器,对应的优点就是客户端响应速度快。但它也有自身的局限性,其缺点如下:

(1) 只适用于局域网。而随着互联网的飞速发展,移动办公和分布式办公越来越普及,这需要我们系统具有扩展性。这种方式远程访问需要专门的技术,同时要对系统进行专门的设计来处理分布式数据。

(2) 客户端需要安装专用的客户端软件。首先涉及安装的工作量,其次任何一台计算机出问题,如病毒、硬件损坏,都需要进行安装或维护。特别是有很多分部或专卖店的情况,不是工作量的问题,而是路程的问题。另外,在系统软件升级时,每一台客户机需要重新安装,其维护和升级成本非常高。

(3) 对客户端的操作系统一般也会有限制。可能适应于 Windows 98,但不能用于 Windows 2000 或 Windows XP。或者不适用于微软新的操作系统等,更不用说 Linux、UNIX 等。

虽然 C/S 结构有一定的缺点,但目前仍有大量的软件开发采用的是该结构,例如 QQ、MSN、PP Live、迅雷等、eMule 等。

2. B/S 结构

B/S 结构即 Browser/Server(浏览器/服务器)结构,是随着 Internet 技术的兴起,对 C/S 体系结构的一种变化或者改进的结构。在 B/S 体系结构下,用户界面完全通过 WWW 浏览器实现,一部分事务逻辑在前端实现,但是主要事务逻辑在服务器端实现。B/S 结构利用不断成熟和普及的浏览器技术实现原来需要复杂专用软件才能实现的强大功能,并节约了开发成本,是一种全新的软件系统构造技术。

基于 B/S 体系结构的软件,系统安装、修改和维护都在服务器端解决。Web 应用程序的访问不需要安装客户端程序,可以通过任一款浏览器(例如 IE 或者 Firefox)来访问各类 Web 应用程序。当 Web 应用程序进行升级时,并不需要在客户端做任何更改。和 C/S 结构的应用程序相比,Web 应用程序可以在网络上更加广泛地进行传播和使用。一般的网站都是 B/S 结构的,例如 Google、Baidu。

虽然 B/S 有诸多优点,但也是存在一些缺点的,例如 B/S 结构程序在跨浏览器的使用上总是不能尽如人意。另外,因为 B/S 结构的程序的大量工作都是由服务器完成的,所以如何设计算法使得访问效率得到保证也是一个很大的问题。

本书所介绍的 JSP 程序的体系结构将要采用的结构就是 B/S 结构,B/S 结构的程序是非常注重程序架构的,常用的程序架构有三层架构和两层架构,具体内容在下面介绍。

1.3.2 三层架构

传统的 JSP 开发中通常将业务处理的代码与 JSP 代码混在一起，这样程序的可读性很差，不易于阅读，更不易于代码维护。如何解决这个弊端？通常采用分层模式的设计理念来解决这个问题。分层模式是最常见的一种架构模式。分层模式是很多架构模式的基础，通过分层模式将解决方案的组件分隔到不同的层中，实现在同一个层中组件之间保持内聚性，并保持层与层之间松耦合，如何进行分层呢？

一般的做法是在客户端与数据库之间加入一个“中间层”，从而形成三层架构。这里所说的三层架构，不是指物理上的三层，而是逻辑上的三层，即把这三个层放置到一台机器上。三层架构的三层指的是表示层、业务逻辑层、数据层，各层的作用如下。

（1）表示层：主要作用为数据显示或者和后台进行交互，因此表示层通常对应于 HTML 页面或 JSP 页面。

（2）业务逻辑层：主要是针对具体问题的操作，也可以理解成对数据层的操作，对数据进行业务逻辑处理。

（3）数据层：主要是对非原始数据（数据库或者文本文件等存放数据的形式）的操作层，而不是指原始数据。也就是说，是对数据库的操作，而不是数据，具体为业务逻辑层或表示层提供数据服务。

注意：这里所说的数据层并不是数据库，而是与数据库密切相关的操作代码。

表示层、业务逻辑层、数据访问层之间的访问关系如图 1-2 所示。

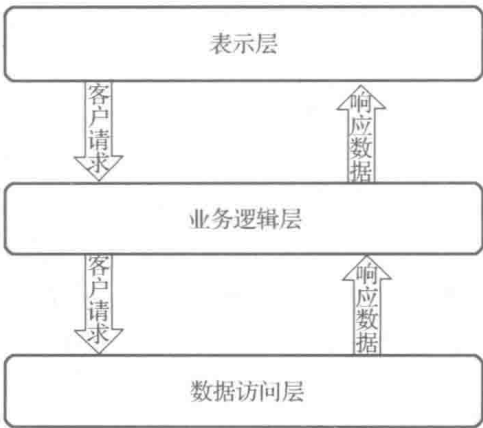


图 1-2 三层架构

表示层访问业务层，业务层为表示层的访问提供数据或相应的方法；业务层访问数据层，数据层为业务层的访问提供数据或方法。也可以说，表示层依赖业务层，业务层依赖数据层，层和层之间是单向的依赖关系，下层不知道上层的存在，仅完成自身的功能，而不关心结果如何使用，每一层仅知道其下层的存在，忽略其他层的存在，只关心结果的取得，而不关心结果的实现过程。

这种单向的依赖关系使得在同一个层中组件之间保持内聚性，并保持层与层之间的松耦合，从而实现软件工程要求的高内聚和低耦合的设计目标，提高程序的可复用性。

1.3.3 两层架构

三层架构虽然优秀，但理念相对复杂，不利于初学者掌握，因此在本书的 JSP 程序设计中将采用两层架构。所谓两层架构就是由同一程序来实现逻辑计算和数据处理，即把逻辑层与数据处理层合并为一层。这时，前台就是表示层页面，后台就是业务层和数据层的 java 代码。两层架构如图 1-3 所示。



图 1-3 两层架构

1.4 搭建 JSP 的运行环境

1.4.1 JDK 的安装与配置

JSP 程序中的脚本语言是 Java 语言，Java 程序是运行在 Java 的虚拟机即 JVM 上的，所以 JSP 程序的开发首先需要搭建 Java 开发环境。搭建 Java 开发环境主要包括如下步骤：下载并安装 Java Develop Kit (JDK)；配置环境变量。下面分别进行详细介绍。

安装 JDK 及配置环境变量

在 Windows 下安装 JDK 与安装其他程序的步骤基本相同，请选择默认项直到单击“下一步”按钮，还需要进行系统环境变量的配置。

所谓环境变量，就是在操作系统中一个具有特定名字的对象，它包含了一个或多个应用程序将使用到的信息。如果安装完 JDK 之后，不配置 Java 的环境变量，那么在 DOS 命令行环境下就找不到 Java 的编译程序和 Java 的运行程序，也就不能在 DOS 环境下进行 Java 编译与运行程序了。与 JDK 或 JRE 的使用有关的是 path、classpath 两个环境变量。path 变量中存储的是 JDK 命令文件的路径，path 变量用来告诉操作系统到哪里去查找某个命令，只有设置好 path 变量，才能正常地编译和运行 Java 程序。classpath 则表示的是“类”路径，classpath 变量中存储的是 JDK 的类文件的路径，classpath 变量用来告诉 Java 执行环境，在哪些目录下可以找到执行 Java 程序所需要的类或包，在这些包中包含了常用的 Java 方法和常量。path 变量的值是 JDK 命令文件的路径，它的值应该设置成：“C:\Program Files (x86)\Java\jdk1.8.0_11\bin;”。classpath 变量的值是 JDK 类文件的路径，它的值应该设置成：“.;C:\Program Files (x86)\Java\jdk1.8.0_11\lib;”。注意，C:\Program Files (x86)是根路径，用户可以根据自己 JDK 的安装位置，调整 C:\Program Files (x86)的值。下面分别对这两个环境变量进行设置。

在环境变量设置窗口（如图 1-4 所示），单击“新建”按钮，添加一个名字是 path 的环境变量，变量的值是：“C:\Program Files (x86)\Java\jdk1.8.0_11\bin;”，如图 1-5 所示。

输入完成后，单击“确定”按钮，即可进行保存。path 变量就出现在系统变量列表中了，如图 1-6 所示。



图 1-4 环境变量设置窗口



图 1-5 新建系统变量



图 1-6 系统变量列表

注意：因为安装某些其他软件也需要配置 path 变量，可以选中 path 变量，单击“编辑”按钮对该变量进行编辑。如果因为其他软件 path 变量问题使得 JDK 运行异常，可以将 Java 的 path 变量的值放在其他软件 path 变量的值的前面，最后以分号结束，这样就能解决这个问题。

然后配置 classpath 环境变量，单击“新建”按钮，添加一个名字是 classpath 的环境变量，变量的值是：“.;C:\\Program Files (x86)\\Java\\jdk1.8.0_11\\lib;”，如图 1-7 所示。



图 1-7 新建系统变量

输入完成后，单击“确定”按钮，即可进行保存。至此，环境变量配置完成，可以编写 Java 程序来测试环境变量的配置是否正确。

1.4.2 Tomcat 的安装、运行与目录结构

JSP 程序的运行必须依赖于服务器。目前能够运行 JSP 的服务器软件有很多，例如 Tomcat、JBoss、Resin、WebLogic 等。每个服务器都有自己的特点，其应用方面也不相同。本书将选择 Tomcat 服务器，Tomcat 是 Apache 公司的产品，目前版本已经升级到 9.x。Tomcat 服务器在中、小型的 JSP 网站上应用比较广泛，并且是完全开源免费的。

1. 下载 Tomcat

获取 Tomcat 非常容易，可以直接在网上搜索或者从 Tomcat 官方网站获取。访问“<http://tomcat.apache.org/>”，下载 Tomcat 软件“apache-tomcat-7.0.27.exe”，下载完毕后，就可以使用 Tomcat 服务器了。

2. 安装 Tomcat

单击下载的可执行程序，会弹出一个如图 1-8 所示的窗口，在该窗口中单击 Next 按钮，会弹出如图 1-9 所示的窗口。



图 1-8 Tomcat 安装启动窗口

在如图 1-9 所示的窗口中单击 I Agree 按钮，进入安装选项窗口，如图 1-10 所示。在该窗口中需要对相关的插件进行选择，在这里把所有的插件全部选中，即选择 Full 选项，选择好后单击 Next 按钮，会显示如图 1-11 所示的窗口。

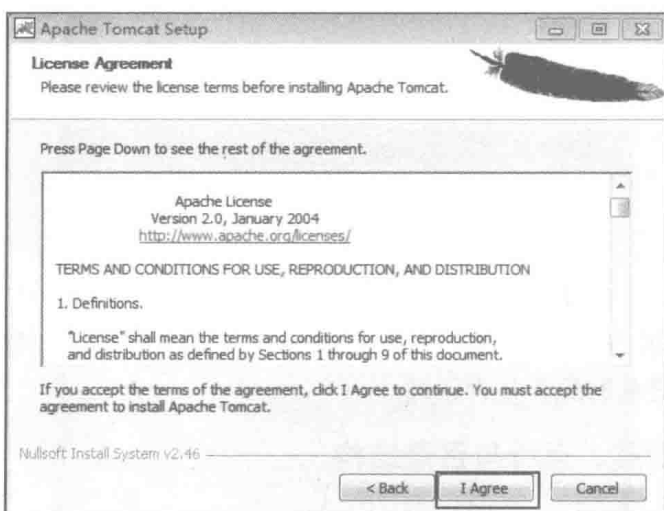


图 1-9 Tomcat 安装显示窗口

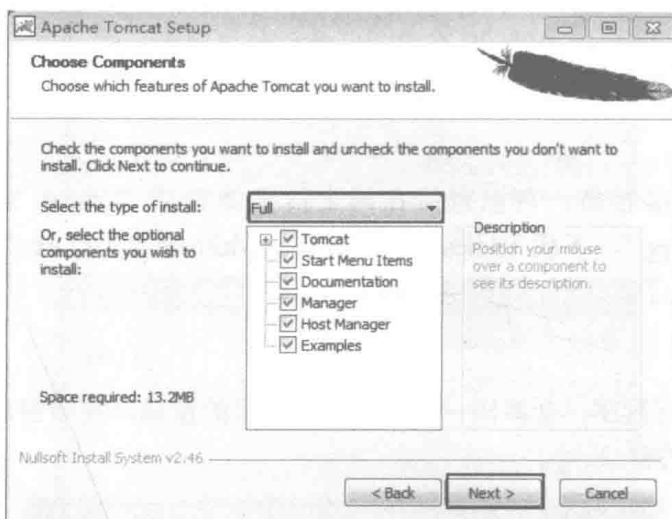


图 1-10 安装选项窗口



图 1-11 进行端口的配置