

JSP

程序设计



主编 徐辉 陆璐



中国海洋大学出版社
CHINA OCEAN UNIVERSITY PRESS

JSP 程序设计

主 编 徐 辉 陆 璐 侯丽敏



中国海洋大学出版社

图书在版编目(CIP)数据

JSP 程序设计 / 徐辉, 陆璐, 侯丽敏主编. — 青岛:
中国海洋大学出版社, 2017.12

ISBN 978-7-5670-1567-8

I. ①J… II. ①徐… ②陆… ③侯… III. ①JAVA 语
言—网页制作工具 IV. ①TP312.8②TP393.092.2

中国版本图书馆 CIP 数据核字(2017)第 228347 号

出版发行 中国海洋大学出版社
社 址 青岛市香港东路 23 号
出 版 人 杨立敏
网 址 <http://www.ouc-press.com>
电子信箱 sjyybook@163.com
订购电话 010-60739092
责任编辑 王积庆
印 制 廊坊市广阳区九洲印刷厂
版 次 2017 年 12 月第 1 版
印 次 2017 年 12 月第 1 次印刷
成品尺寸 185mm × 260mm
印 张 19
字 数 440 千字
定 价 78.00 元

邮政编码 266071

电 话 0532-85901092

印 数 1 - 3000

前言

Preface

随着计算机及网络技术的迅速发展,Web 应用程序开发占据了主要的开发市场。JSP 是由 Sun Microsystems 公司倡导、许多公司参与一起建立的一种动态网页技术标准。它以 Java 语言为基础,与 HTML 文本语言、JSP 标记紧密结合,可以很好地实现 Web 页面设计和业务逻辑分离。JSP 程序不仅编写灵活,执行容易,而且具有很好的跨平台性,大大提高了系统的执行性能。

本书以项目为背景,从实战着手,以 JSP 知识为主线,由浅入深,循序渐进地讲述了使用 JSP 进行应用开发的流程与编程技巧。

本书课后提供了大量的习题,方便同学对基础知识的牢固掌握。

感谢读者使用本书,由于作者水平有限,本书不足之处在所难免,欢迎广大读者批评指正。

编 者

准备阶段——JSP 相关知识学习

第1章 JSP 概述

1.1 静态网页与动态网页技术概述	4
1.1.1 静态网页	4
1.1.2 动态网页	4
1.2 什么是 JSP	4
1.3 JSP 的优势.....	6
1.4 JSP 的劣势.....	6
1.5 应用程序体系结构	7
1.5.1 二层架构模型	7
1.5.2 三层架构模型	7
1.5.3 JSP 技术支持的架构模型.....	8
习题	9

第2章 JSP 运行环境

2.1 JSP 运行环境的构成	12
2.2 JDK 的安装与配置	12
2.2.1 下载 JDK	12
2.2.2 在 Windows 系统下安装 JDK	13
2.2.3 在 Windows 下配置环境变量	15
2.3 Tomcat 的安装与配置.....	16
2.3.1 下载 Tomcat	16
2.3.2 Tomcat 的安装	17
2.3.3 Tomcat 启动	19
2.3.4 测试 Tomcat	22
2.3.5 Tomcat 的目录结构	22
2.4 MyEclipse 的使用	23

2.5 使用 MyEclipse 开发 JSP 项目	25
2.5.1 在 MyEclipse 中配置 Tomcat	25
2.5.2 创建 JSP 项目	27
2.5.3 部署 Web 项目	27
2.5.4 运行 JSP 项目	29
习题	30

第3章 HTML 与 JavaScript 基础知识回顾

3.1 HTML 基础知识	34
3.1.1 HTML 概述	34
3.1.2 HTML 页面的基本结构	34
3.1.3 表格标记	36
3.1.4 HTML 表单标记	39
3.1.5 超链接标记	45
3.1.6 框架	47
3.2 JavaScript 基础知识回顾	50
3.2.1 在 Web 页面中使用 JavaScript	50
3.2.2 事件处理	52
3.2.3 window 对象	54
3.2.4 document 对象	59
3.2.5 document 对象应用案例	61
3.2.6 JavaScript 综合应用案例 - 客户端用户信息验证	62
习题	65

第4章 JSP 基础语法

4.1 JSP 页面的基本构成	70
4.1.1 注释	71
4.1.2 声明	72
4.1.3 表达式	75
4.1.4 Java 代码块	76
4.2 JSP 中多线程同步	77
4.3 JSP 标记	79
4.3.1 JSP 标记概述	79
4.3.2 JSP 指令标记	79
4.3.3 JSP 动作标记	84
4.3.4 综合案例	93

习题	97
----------	----

项目实战阶段——JSP 重点知识学习

第5章 综合实例—网络聊天室

5.1 项目背景	104
5.2 需求分析	104
5.3 总体设计	104
5.3.1 项目规划	104
5.3.2 系统功能结构图	105
5.4 系统设计	105
5.4.1 设计目标	105
5.4.2 开发及运行环境	106
5.4.3 逻辑结构设计	106
5.4.4 功能模块及对应页面介绍	108
5.5 界面展示	109
5.6 涉及知识点	114

第6章 JSP 内置对象

6.1 项目分解(一)利用 JSP 内置对象实现用户登录模块	116
6.2 理论知识	123
6.2.1 内置对象概述	123
6.2.2 request 对象	124
6.2.3 response 对象	131
6.2.4 session 对象	142
6.3 项目分解(二)利用 application 对象实现用户聊天模块	149
6.4 理论知识	153
6.4.1 application 对象简介	153
6.4.2 application 对象的常用方法	154
6.4.3 application 对象常用示例	155
6.4.4 out 对象与 exception 对象	159
6.4.5 一些内置对象的作用范围	160
习题	160

第7章 使用 JSP 访问数据库

7.1 项目分解(一)实现管理员权限中的更改用户信息功能	164
------------------------------------	-----

7.2 理论知识	165
7.2.1 JDBC 工作原理	165
7.2.2 JDBC 数据库连接方式	166
7.2.3 数据库事务处理	167
7.2.4 处理数据库返回的结果集并释放资源对象	168
7.2.5 数据库示例	169
习题	181

第8章 JSP 与 JavaBean

8.1 项目分解(一)实现管理员权限中的查询用户信息功能	188
8.2 项目分解(二)实现管理员权限中的查询用户聊天信息功能	189
8.3 理论知识	190
8.3.1 JavaBean 简介	190
8.3.2 编写 JavaBean	190
8.3.3 在 JSP 中使用 bean	193
8.3.4 综合实例	198
8.4 项目分解(三)实现管理员权限中的删除用户信息功能	206
8.5 项目分解(四)实现普通用户权限中的用户注册功能	210
习题	214

提高阶段——JSP 难点知识学习

第9章 Servlet 与 MVC 设计模式

9.1 案例设计:使用 MVC 设计模式,完成用户登录验证功能	222
9.2 理论知识	225
9.2.1 Servlet 简介	225
9.2.2 Servlet 的编写、部署与调用	228
9.2.3 MVC 开发模式介绍	235
9.3 MVC 模式应用案例实现——案例一的代码实现	238
习题	250
附录 在线聊天室源程序代码	255
参考文献	296

准备阶段

JSP相关知识学习

第1章

JSP概述

【 学习目标 】

1. 了解动态网页与静态网页的区别。
2. 了解 JSP, JSP 的优点与缺点。
3. 了解应用程序体系结构。

1.1

静态网页与动态网页技术概述

1.1.1 静态网页

在网站设计中,纯粹 HTML(标准通用标记语言下的一个应用)格式的网页通常被称为“静态网页”,静态网页是标准的 HTML 文件,它的文件扩展名是 .htm、.html,可以包含文本、图像、声音、FLASH 动画、客户端脚本和 ActiveX 控件及 Java 小程序等。静态网页是网站建设的基础,早期的网站一般都是由静态网页制作的。静态网页是相对于动态网页而言的,是指没有后台数据库、不含程序和不可交互的网页。静态网页相对更新起来比较麻烦,适用于一般更新较少的展示型网站。实际上静态网页也不是完全静态,他也可以出现各种动态的效果,如 GIF 格式的动画、FLASH、滚动字幕等。静态网页的基本特点如下:

- (1) 每个网页都有一个固定的 URL,且 URL 以 .htm、.html、.shtml 等常见形式为后缀,并且在 URL 中不包含“?”。
- (2) 网页内容一经发布,无论是否有用户访问,内容都是保存在网站服务器上的。
- (3) 静态网页的内容相对稳定,因此容易被搜索引擎检索。
- (4) 静态网页没有数据库的支持。
- (5) 静态网页的交互性较差,在功能方面有较大的限制。

1.1.2 动态网页

动态网页,是指跟静态网页相对的一种网页编程技术。静态网页,随着 html 代码的生成,页面的内容和显示效果就基本上不会发生变化了——除非你修改页面代码。而动态网页则不然,页面代码虽然没有变,但是显示的内容却可以随着时间、环境或者数据库操作的结果而发生改变。动态网页的基本特点如下:

- (1) 动态网页以数据库技术为基础,可以大大降低网站维护的工作量。
- (2) 采用动态网页技术的网站可以实现更多的功能。
- (3) 动态网页实际上并不是独立存在于服务器上的网页文件,只有当用户请求时服务器才返回一个完整的网页。
- (4) 动态网页中的“?”对搜索引擎检索存在一定的问题。

1.2

什么是 JSP

JSP 全名为 Java Server Pages,其根本是一个简化的 Servlet 设计,它是由 Sun Microsystems 公司倡导、许多公司参与一起建立的一种动态网页技术标准。JSP 技术是在传统的网页 HT-

ML(标准通用标记语言的子集)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。当一个JSP文件第一次被请求访问时,JSP引擎把该JSP文件转换成为一个Servlet。而这个引擎本身也是一个Servlet。JSP的运行过程如图1-1所示。

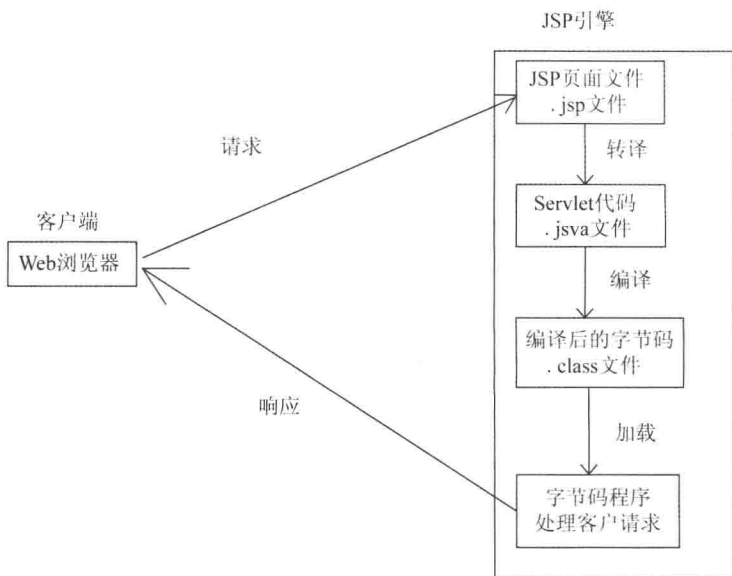


图 1-1 JSP 运行过程

(1) JSP引擎先把该JSP文件转换成一个Java源文件(Servlet),在转换时如果发现JSP文件有任何语法错误,转换过程将中断,并向服务端和客户端输出错误信息。

(2) 如果转换成功,JSP引擎用javac把该Java源文件编译成相应的.class文件。

(3) 创建一个该Servlet(JSP页面的转换结果)的实例,该Servlet的jspInit()方法被执行,jspInit()方法在Servlet的生命周期中只被执行一次。

(4) jspService()方法被调用来处理客户端的请求。对每一个请求,JSP引擎创建一个新的线程来处理该请求。如果有多个客户端同时请求该JSP文件,则JSP引擎会创建多个线程。每个客户端请求对应一个线程。以多线程方式执行可以大大降低对系统的资源需求,提高系统的并发量及响应时间。但不过也应该注意多线程的编程限制,由于该Servlet始终驻于内存,所以响应是非常快的。

(5) 如果.jsp文件被修改了,服务器将根据设置决定是否对该文件重新编译,如果需要重新编译,则将编译结果取代内存中的Servlet,并继续上述处理过程。

(6) 虽然JSP效率很高,但在第一次调用时由于需要转换和编译而有一些轻微的延迟。此外,在任何时候如果由于系统资源不足的原因,JSP引擎将以某种不确定的方式将Servlet从内存中移去。当这种情况发生时jspDestroy()方法首先被调用。

(7) 然后Servlet实例便被标记加入“垃圾收集”处理。可在jspInit()中进行一些初始化

工作,如建立与数据库的连接,或建立网络连接,从配置文件中取一些参数等,在 `jspDestory()` 中释放相应的资源。

1.3 JSP 的优势

JSP 的优势如下:

(1) 简便性和有效性。

JSP 动态网页的编写与一般的静态网页编写十分相似。只是在原来的 HTML 网页中加入一些 JSP 专有的标签,或是一些脚本程序。这样,一个熟悉 HTML 网页编写的设计人员可以很容易进行 JSP 网页的开发。而且开发人员自己完全可以不编写脚本程序,而只是通过 JSP 独有的标签使用别人已写好的部件来实现动态网页的编写。这样,一个不熟悉脚本语言的网页开发者,完全可以利用 JSP 做出漂亮的动态网页。而这在其他的动态网页开发中是无法实现的。

(2) 程序的独立性。

JSP 是 Java API 家族的一部分,它拥有一般的 Java 程序的跨平台的特性。换句话说,就是拥有程序对平台的独立性,即“Write once,Run anywhere!”一次编写,处处运行。

(3) 程序的兼容性。

JSP 中的动态内容可以各种形式进行显示,所以它可以为各种客户提供服务,即从使用 HTML/DHTML 的浏览器,到使用 WML(Wireless Markup Language,无线标记语言)的各种移动无线设备(例如:移动电话和个人数字设备 PDA),再到使用 XML(Extensible Markup Language,可扩展标记语言)的 B2B 应用,都可以使用 JSP 的动态页面。

(4) 程序的可重用性。

在 JSP 页面中可以不直接将脚本程序嵌入,而只是将动态的交互部分作为一个部件加以引用。一旦这样的一个部件写好,它可以为多个程序重复引用,实现了程序的可重用性。现在,大量的标准 JavaBeans 程序库就是一个很好的例证。

1.4 JSP 的劣势

JSP 的劣势如下:

(1) JSP 技术极大地增加了产品的复杂性。

为了获得系统的跨平台功能和产品伸缩能力,Java 系统开发了多种产品,如 JRE、JDK、J2EE、EJB、JSWDK、JavaBeans,这样极大地增加了产品的复杂性。

(2) Java 的高效率运行需要占用大量的内存和硬盘空间。

Java 程序的高速运行是通过 .class 文件常驻内存来实现的,另外,Java 程序还需要硬盘

空间来存储一系列的. java 文件和. class 文件以及对应的版本文件。

(3) JSP 程序调试困难。

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 事先被转换成. java 文件(Servlet)后再运行的,所以,浏览器中显示的代码出错行数并不是 JSP 源代码的行数,而是指转换后的. java 文件(Servlet)中程序代码的行数。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,即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。

1.5

应用程序体系结构

随着 Web 技术的出现,早期网络的集中计算逐渐被分布式计算所替代,Web 技术是一种分布式计算技术,使用这种技术构建企业应用时,通常需要开发大量的程序,把这些程序分布在不同的计算机上,在应用中承担不同的职责。例如:有的程序负责展示用户界面,有的程序负责进行逻辑计算,有的程序负责进行数据处理。在软件开发的实践过程中,人们总结出了一些常用的软件系统结构模式,以供应用系统设计时参考。这些模式包括两层、三层或 N 层架构等,下面开始介绍应用程序的二层和三层架构模型。

1.5.1 二层架构模型

二层架构模型是软件系统中最常见的一种,也称为客户端/服务器(Client/Server)结构。它将应用程序一分为二,客户端负责完成与用户的交互任务,数据库服务器端负责数据管理。它的缺点是表现层和事务层都放在客户端,而数据逻辑层和数据存储层则置于服务器端,所以客户端/服务器也被常称为“胖客户端”模式。

在实际的系统设计中,此类结构主要用于在局域网范围内的前台客户端+后台数据库管理系统,如图 1-2 所示。目前有很多数据库前端开发工具(如 Java、C#、Delphi、PowerBuilder 等)可以用来专门制作这种结构的软件系统。

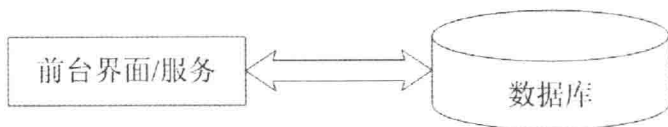


图 1-2 二层架构模型

1.5.2 三层架构模型

在三层架构中,按照程序承担的不同角色,把应用程序分为三层:

- (1) 数据表示层: 提供用户数据输入界面, 运行在客户端上。
- (2) 逻辑计算层: 提供数据计算功能, 运行在应用服务器上。
- (3) 数据处理层: 提供数据库处理功能, 运行在数据库服务器上。

应用程序的三层架构模型, 如图 1-3 所示。

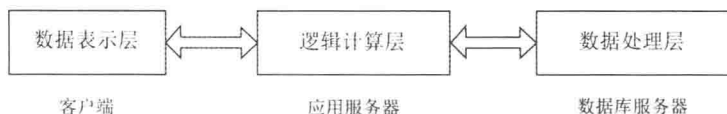


图 1-3 三层架构模型

客户端: 本地用户使用的计算机, 利用客户软件请求应用服务器提供服务。

应用服务器: 接受客户请求进行数据计算, 并把计算结果返回给客户。

数据库服务器: 提供数据处理和事务处理。

目前最典型的基于三层 B/S 结构的应用模式便是 B/S/S (Browser/ Web Server/DB Server, 浏览器/Web 服务器/数据库服务器) 模式, 在 B/S/S 结构中, 浏览器是一个用于文档检索和显示的客户应用程序, 并通过超文本传输协议 HTTP 与 Web 服务器相连。该模式下, 通用的、低成本的浏览器节省了两层结构的 C/S 模式客户端软件的开发和维护费用。这些浏览器大家都很熟悉, 包括 IE、Firefox、Chrome 等。

Web 服务器是指驻留于互联网上某种类型计算机的程序。当浏览器(客户端)连到服务器上并请求文件或数据时, 服务器将处理该请求, 并将文件或数据发送到该浏览器上, 附带的信息会告诉浏览器如何查看该文件。Web 服务器不仅能够存储信息, 还能在用户通过 Web 浏览器提供信息的基础上运行脚本和程序。

1.5.3 JSP 技术支持的架构模型

JSP 技术开发的程序结构只能是 B/S/S 结构或 B/S 结构。下面是 JSP 技术支持的一般层次模型, 如图 1-4 所示。通常情况下, JSP 页面由展示用户界面的 HTML 标记和进行数据计算两部分组成, 因此数据展示层代码和数据计算代码可能处在同一 JSP 页面, 它们都部署在 Web 服务器端。

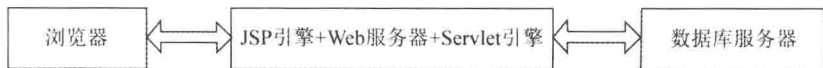


图 1-4 JSP 技术支持的一般层次模型

JSP 页面的三种形式如下:

- (1) JSP 页面 = HTML 标记 + java 程序片。
- (2) JSP 页面 = HTML 标记 + Servlet 模块。
- (3) JSP 页面 = HTML 标记 + Bean。

通常 Java 程序片、Servlet 模块实现逻辑计算功能, Bean 实现数据处理功能, HTML 标记

实现数据展示功能。JSP 页面中的 Java 程序片最终被 JSP 引擎转译为 Servlet 模块,当客户发出 Servlet 请求时,由 Servlet 引擎将这些应用 Servlet 模块载入内存运行,以处理客户请求。

习 题

一、选择题

- 当用户请求 JSP 页面时,JSP 引擎就会执行该页面的字节码文件响应客户的请求,执行字节码文件的结果是()。
 - 发送一个 JSP 源文件到客户端
 - 发送一个 Java 文件到客户端
 - 发送一个 HTML 页面到客户端
 - 什么都不做。
- 当多个用户请求同一个 JSP 页面时,Tomcat 服务器为每个客户启动一个()。
 - 进程
 - 线程
 - 程序
 - 服务
- 下列动态网页和静态网页的根本区别描述错误的是()。
 - 静态网页服务器端返回的 HTML 文件是事先存储好的
 - 动态网页服务器端返回的 HTML 文件是程序生成的
 - 静态网页文件里只有 HTML 标记,没有程序代码
 - 动态网页中只有程序,不能有 HTML 代码
- 不是 JSP 运行必须的是()。
 - 操作系统
 - JavaJDK
 - 支持 JSP 的 Web 服务器
 - 数据库
- URL 是 Internet 中资源的命名机制,URL 由三部分构成()。
 - 协议、主机 DNS 名或 IP 地址和文件名
 - 主机、DNS 名或 IP 地址和文件名、协议
 - 协议、文件名、主机名
 - 协议、文件名、IP 地址
- HTTP 的默认端口号是()。
 - 80
 - 8080
 - 70
 - 21
- 如果网页(),该网页是动态的。
 - 有 GIF 动画图片动来动去
 - 有动画广告飞来飞去
 - 能看影视
 - 是动态实时生成的
- 以下选项中()是不正确的 URL。