

普通高等教育IT类工程创新能力培养规划教材

Java Web

程序设计

CHENGXU SHEJI

主 编 / 于 曦 鄢 涛 李 丹

副主编 / 刘永红 赵卫东 张海清



科学出版社

Java Web 程序设计

于 曦 鄢 涛 李 丹 主 编
刘永红 赵卫东 张海清 副主编

科 学 出 版 社

北 京

内 容 简 介

本书按工程认证的标准,以CDIO的理念为基础,分11个章节,对Java Web技术及其应用进行讲述,内容涵盖Web应用基础、HTTP与Servlet基础、Servlet核心接口、会话跟踪技术、数据库访问、JSP语法、JSP内置对象、JavaBean技术、表达语言、标准标签库、过滤器与监听器。

为帮助读者快速理解并掌握各项知识点,全面提升面对复杂工程问题时的分析、解决和实际编码能力,本书将一个实际项目的开发融入到全书各个章节,各个知识点。并在每章开篇处标明各知识点与能力点的对应关系,方便读者进行有选择的学习。

本书所有电子配套资料(电子课件、习题解答、源程序代码等)可以在科学出版社的网站上下载。

本书适合作为高等院校信息技术专业或培训机构的Java教材,也可以作为Web应用开发人员的入门参考书。

图书在版编目(CIP)数据

Java Web 程序设计 / 于曦, 鄢涛, 李丹主编. —北京: 科学出版社, 2017.9
ISBN 978-7-03-054603-6

I. ①J… II. ①于… ②鄢… ③李… III. ①JAVA 语言-程序设计
IV. ①TP312.8

中国版本图书馆 CIP 数据核字 (2017) 第 236592 号

责任编辑: 冯 钊 / 责任校对: 韩雨舟
封面设计: 墨创文化 / 责任印制: 罗 科

科学出版社出版

北京东黄城根北街16号
邮政编码: 100717
<http://www.sciencep.com>

四川煤田地质制图印刷厂印刷
科学出版社发行 各地新华书店经销

*

2017年9月第一版 开本: 787×1092 1/16

2017年9月第一次印刷 印张: 26.25

字数: 650千字

定价: 59.00元

(如有印装质量问题, 我社负责调换)

前 言

2016年6月2日,我国顺利成为《华盛顿协议》正式成员,这是促进我国工程师按照国际标准培养、提高工程技术人才培养质量的重要举措,是推进工程师资格国际互认的基础和关键,对我国工程技术领域应对国际竞争、走向世界具有重要意义。

本书是“普通高等教育IT类工程创新能力培养规划教材”中的一门,全书按工程认证的标准,以CDIO的理念为基础,将一个实际项目的开发融入全书的各个章节、各个知识点。通过本书的学习,让读者能快速理解并掌握各项知识点,全面提升面对复杂工程问题时的分析、解决和实际编码能力。

本门课程对应工程认证12项毕业要求指标点的关系如下:

| 能力点 | 覆盖度 |
|---|-------|
| 1.工程知识:能够将数学、自然科学、工程基础和专业知用于解决复杂工程问题 | ★★★★★ |
| 2.问题分析:能够应用数学、自然科学和工程科学的基本原理,识别、表达,并通过文献研究分析复杂工程问题,以获得有效结论 | ★★★ |
| 3.设计/开发解决方案:能够设计针对复杂工程问题的解决方案,设计满足特定需求的系统、单元(部件)或工艺流程,并能够在设计环节中体现创新意识,考虑社会、健康、安全、法律、文化以及环境等因素 | ★★★★★ |
| 4.研究:能够基于科学原理并采用科学方法对复杂工程问题进行研究,包括设计实验、分析与解释数据,并通过信息综合得到合理有效的结论 | ★★ |
| 5.使用现代工具:能够针对复杂工程问题,开发、选择与使用恰当的技术、资源、现代工程工具和信息技术工具,包括对复杂工程问题的预测与模拟,并能够理解其局限性 | ★★★★★ |
| 6.工程与社会:能够基于工程相关背景知识进行合理分析,评价专业工程实践和复杂工程问题解决方案对社会、健康、安全、法律以及文化的影响,并理解应承担的责任 | ★★★ |
| 7.环境和可持续发展:能够理解和评价针对复杂工程问题的专业工程实践对环境、社会可持续发展的影响 | ★★★ |
| 8.职业规范:具有人文社会科学素养、社会责任感,能够在工程实践中理解并遵守工程职业道德和规范,履行责任 | ★★★ |
| 9.个人和团队:能够在多学科背景下的团队中承担个体、团队成员以及负责人的角色 | ★★★★★ |
| 10.沟通:能够就复杂工程问题与业界同行及社会公众进行有效沟通和交流,包括撰写报告和设计文稿、陈述发言、清晰表达或回应指令。并具备一定的国际视野,能够在跨文化背景下进行沟通和交流 | ★★★★★ |
| 11.项目管理:理解并掌握工程管理原理与经济决策方法,并能在多学科环境中应用 | ★★★★★ |
| 12.终身学习:具有自主学习和终身学习的意识,有不断学习和适应发展的能力 | ★★ |

Java Web 技术是目前主流的 Web 应用开发技术之一,Java Web 程序设计已经成为一门综合性高、实践性强、应用领域广的技术类课程。对于从事计算机程序开发的人员,在当前互联网时代,掌握 Java Web 开发技术是非常必要的。本书主要特点如下:

(1)强调 CDIO 理念。将工程认证的理念贯穿所有章节,每章都明确所述知识点与能

力点的对应关系，方便读者有针对性地进行学习。

(2) 强调项目实施。将实际项目贯穿于全书所有章节对应的知识点，每一个章节都在前一章节的基础上进行实现，从而达到对项目的迭代、升级，最终帮助读者实现一个完整项目的开发。

(3) 强调实践性。除了贯穿全书的项目外，书中每个重要的知识点都配备实例，力求帮助读者在掌握知识的同时，能针对不同的需求解决实际问题。

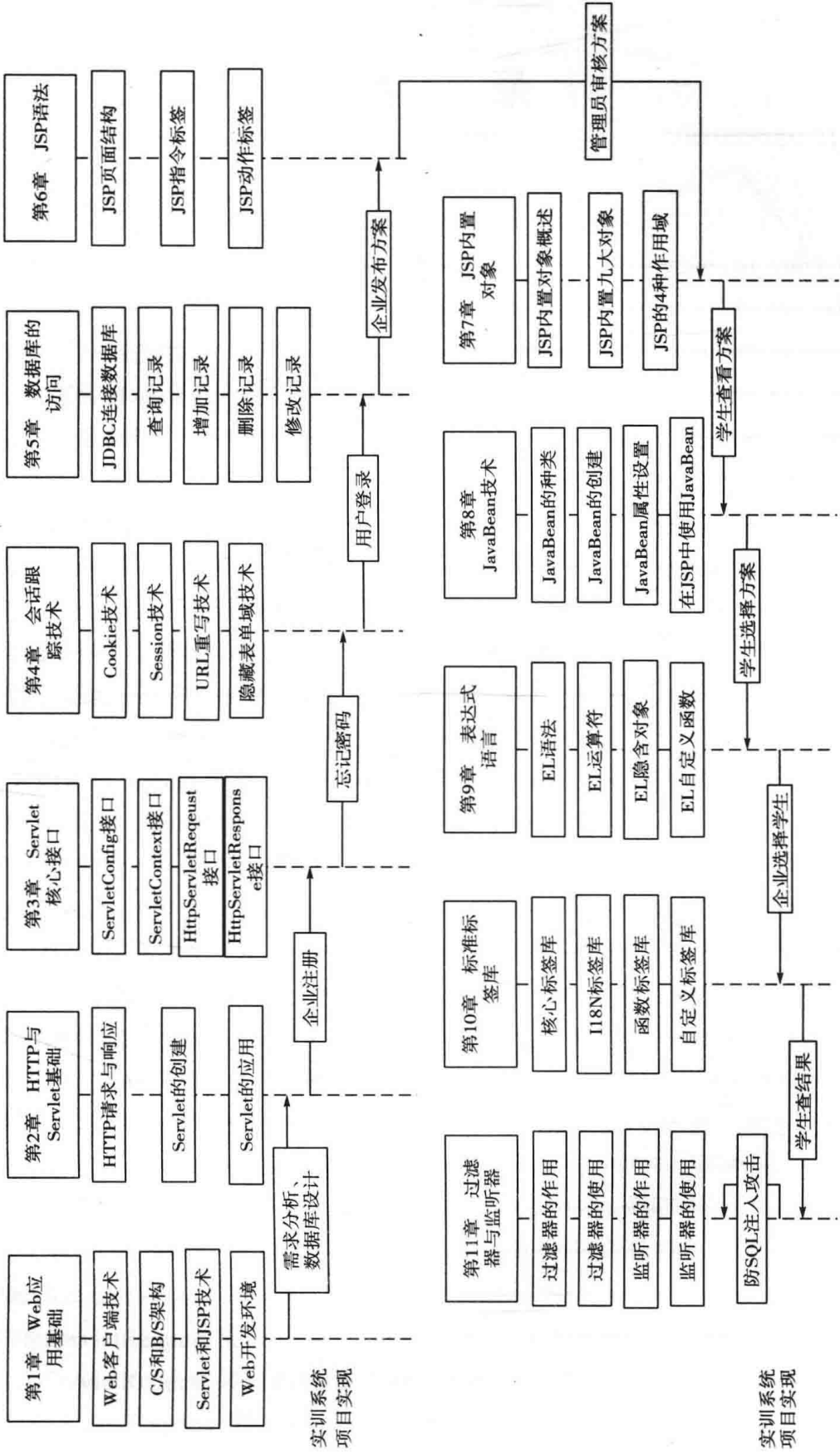
(4) 辅助教学和学习资料全。本书配有实验指导教程、电子课件。项目的源代码可以从科学出版社网站或者 github 上下载。

在使用本书进行教学时，建议采用过程化的考核方式，对每一章的项目任务完成情况进行记录，得到的成绩汇总平均后作为总成绩的 70%。期末再进行笔试，对书中的理论知识进行考核，得到的成绩作为总成绩的 20%。平时的签到和课堂提问成绩作为总成绩的 10%。

本书主编是于曦、鄢涛、李丹，副主编是刘永红、赵卫东、张海清，编委是郑加林、张莉、王兰、刘昶、李梅、聂莉莎等。另外，成都大学软件工程专业的高强、陈天雄、罗廷方、余悦等同学为本书的编写做出了大量的工作，锻炼了他们的写作、交流沟通和项目开发能力，体现了本书 CDIO 的教育理念。

由于作者水平有限，虽然对本书进行了反复审核与修订，但书中疏漏和不足之处在所难免，恳请广大读者及专家给予批评指正。

学习路线图



目 录

| | |
|------------------------------|-----|
| 第 1 章 Web 应用基础 | 1 |
| 1.1 Web 概述 | 2 |
| 1.2 Web 客户端技术 | 3 |
| 1.3 Web 服务器端技术 | 5 |
| 1.4 Web 应用架构 | 6 |
| 1.5 Web 工作机制 | 7 |
| 1.6 Java Web 应用的核心技术 | 7 |
| 1.7 Web 服务器 | 8 |
| 1.8 Java Web 应用开发环境的安装与配置 | 9 |
| 1.9 实例项目：成都大学信息科学与工程学院实训系统 | 14 |
| 1.10 课后练习 | 29 |
| 1.11 实践练习 | 30 |
| 第 2 章 HTTP 与 Servlet 基础 | 31 |
| 2.1 HTTP 请求与响应模型 | 32 |
| 2.2 Servlet 简介 | 40 |
| 2.3 Java Servlet API | 41 |
| 2.4 Servlet 的生命周期 | 43 |
| 2.5 Servlet 的创建 | 45 |
| 2.6 Servlet 的应用 | 53 |
| 2.7 实例项目 | 69 |
| 2.8 课后练习 | 81 |
| 2.9 实践练习 | 83 |
| 第 3 章 Servlet 核心接口 | 85 |
| 3.1 ServletConfig 接口 | 86 |
| 3.2 ServletContext 接口 | 88 |
| 3.3 HttpServletRequest 接口 | 99 |
| 3.4 HttpServletRequest 接口的应用 | 111 |
| 3.5 HttpServletResponse 接口 | 115 |

| | | |
|------------|-----------------|------------|
| 3.6 | 实例项目 | 121 |
| 3.7 | 课后练习 | 136 |
| 3.8 | 实践练习 | 137 |
| 第4章 | 会话跟踪技术 | 139 |
| 4.1 | 会话跟踪技术 | 140 |
| 4.2 | 实例项目 | 150 |
| 4.3 | 课后练习 | 161 |
| 4.4 | 实践练习 | 162 |
| 第5章 | 数据库的访问 | 163 |
| 5.1 | 数据源 | 163 |
| 5.2 | JDBC 连接数据库 | 166 |
| 5.3 | 增加、删除、更新记录 | 171 |
| 5.4 | 查询记录 | 172 |
| 5.5 | 实例项目 | 178 |
| 5.6 | 课后练习 | 194 |
| 5.7 | 实践练习 | 195 |
| 第6章 | JSP 语法 | 196 |
| 6.1 | JSP 页面的基本结构 | 197 |
| 6.2 | 变量和方法的声明 | 198 |
| 6.3 | JAVA 脚本 | 201 |
| 6.4 | 表达式 | 202 |
| 6.5 | JSP 中的注释 | 203 |
| 6.6 | JSP 指令标签 | 205 |
| 6.7 | JSP 动作标签 | 207 |
| 6.8 | JSP 页面元素小结 | 216 |
| 6.9 | 实例项目 | 217 |
| 6.10 | 课后练习 | 224 |
| 6.11 | 实践练习 | 225 |
| 第7章 | JSP 内置对象 | 226 |
| 7.1 | JSP 内置对象概述 | 227 |
| 7.2 | Request 对象 | 228 |
| 7.3 | Response 对象 | 233 |
| 7.4 | Session 对象 | 239 |
| 7.5 | Application 对象 | 244 |

| | | |
|---------------|--------------------|------------|
| 7.6 | Out 对象 | 248 |
| 7.7 | Page | 250 |
| 7.8 | PageContext | 251 |
| 7.9 | Config | 253 |
| 7.10 | Exception | 255 |
| 7.11 | JSP 的 4 种作用域 | 257 |
| 7.12 | 实例项目 | 262 |
| 7.13 | 课后练习 | 276 |
| 7.14 | 实践练习 | 277 |
| 第 8 章 | JavaBean 技术 | 279 |
| 8.1 | JavaBean 简介 | 280 |
| 8.2 | JavaBean 的种类 | 281 |
| 8.3 | JavaBean 的创建 | 281 |
| 8.4 | JavaBean 属性设置的原理 | 283 |
| 8.5 | 在 JSP 中使用 JavaBean | 294 |
| 8.6 | JavaBean 应用 | 298 |
| 8.7 | 实例项目 | 304 |
| 8.8 | 课后练习 | 312 |
| 8.9 | 实践练习 | 313 |
| 第 9 章 | 表达式语言 | 314 |
| 9.1 | 什么是表达式语言 | 314 |
| 9.2 | EL 语法 | 316 |
| 9.3 | EL 运算符 | 317 |
| 9.4 | EL 隐含对象 | 320 |
| 9.5 | EL 自定义函数 | 326 |
| 9.6 | 实例项目 | 330 |
| 9.7 | 课后练习 | 358 |
| 9.8 | 实践练习 | 359 |
| 第 10 章 | 标准标签库 | 360 |
| 10.1 | 认识 JSTL | 361 |
| 10.2 | 核心标签库 | 361 |
| 10.3 | I18N 标签库 | 374 |
| 10.4 | XML 标签库 | 375 |
| 10.5 | SQL 标签库 | 376 |

| | | |
|---------------|----------------|------------|
| 10.6 | 函数标签库 | 377 |
| 10.7 | 自定义标签库的开发 | 379 |
| 10.8 | 实例项目 | 384 |
| 10.9 | 课后练习 | 384 |
| 10.10 | 实践练习 | 385 |
| 第 11 章 | 过滤器与监听器 | 386 |
| 11.1 | 过滤器 | 386 |
| 11.2 | 监听器 | 398 |
| 11.3 | 实例项目 | 406 |
| 11.4 | 课后练习 | 408 |
| 11.5 | 实践练习 | 409 |

第 1 章 Web 应用基础

本章目标

| 知识点 | 理解 | 掌握 | 应用 |
|--------------------|----|----|----|
| 1.JSP 与 Servlet 技术 | ✓ | ✓ | |
| 2.C/S 与 B/S 结构 | ✓ | ✓ | |
| 3.B/S 结构应用运行过程 | ✓ | ✓ | |
| 4.Java Web 开发环境搭建 | ✓ | ✓ | ✓ |
| 5.Web 应用的基本结构 | ✓ | ✓ | |
| 6.需求分析 | ✓ | ✓ | ✓ |
| 7.数据库设计 | ✓ | ✓ | ✓ |

项目任务

完成成都大学信息科学与工程学院实训系统项目的需求分析、数据库设计、开发环境搭建的设计任务：

- 项目任务 1-1 需求分析
- 项目任务 1-2 数据库设计
- 项目任务 1-3 开发环境搭建
- 项目任务 1-4 工具类设计

知识能力点

| 知识点能力点 | 知识点 1 | 知识点 2 | 知识点 3 | 知识点 4 | 知识点 5 | 知识点 6 |
|-----------|-------|-------|-------|-------|-------|-------|
| 工程知识 | ✓ | ✓ | ✓ | | | |
| 问题分析 | | | | | ✓ | |
| 设计/开发解决方案 | | | | | ✓ | |
| 研究 | ✓ | | | | | |
| 使用现代工具 | | | | ✓ | | ✓ |
| 工程与社会 | | | | | ✓ | |
| 环境和可持续发展 | | | | | ✓ | |
| 职业规范 | | | | | | |
| 个人和团队 | | | | | ✓ | |
| 沟通 | | | | | ✓ | |
| 项目管理 | | | | ✓ | | |
| 终身学习 | ✓ | ✓ | ✓ | | ✓ | |

1.1 Web 概述

随着计算机网络技术的高速发展,社会各个行业的应用都进入了基于 Web 应用为核心的阶段,以 Web 方式进行信息处理与应用系统开发已经成为当前信息系统的主流。

1.1.1 Web 的概念

Web(World Wide Web),即全球广域网,也称为万维网。它是一种基于超文本和 HTTP 协议的,全球性的、动态交互的、跨平台的分布式图形信息系统。它是建立在 Internet 上的一种网络服务,为浏览者在 Internet 上查找和浏览信息提供了图形化的、易于访问的直观界面,其中的文档及超级链接将 Internet 上的信息节点组织成一个互为关联的网状结构。Web 技术具有以下特点。

1.图形化

在 Web 技术出现之前,Internet 上的信息只有文本形式,而 Web 技术能在网页上同时提供图形、音频、视频信息,为各类信息的传播提供坚实的基础,使得它自身快速流行起来。

2.易导航

Web 技术使页面之间的跳转变得轻松,用户只需要点击网页上的链接,就可以在各个网站、各个页面之间进行浏览。

3.平台无关

随着信息技术的发展,操作系统也在进行变革。DOS、Windows、Macintosh、UNIX 等操作系统相继出现,并拥有各自庞大的用户群体。传统的二进制可执行代码由于与操作系统绑定紧密,无法跨系统平台执行,这阻碍了信息的传播。而 Web 技术通过各种浏览器软件,可以轻松实现在不同操作系统下信息的展示。

4.分布式

Web 技术支持将图形、音频、视频等信息存放在不同的物理空间中。但可以通过将信息在一个站点上进行逻辑整合,保证信息对用户来说是一体的。这样,既便于用户访问信息,又便于开发人员对信息进行维护。

5.动态性

由于各 Web 站点的信息包含站点本身的信息,信息的提供者可以经常对站点上的信息进行更新。如某个协议的发展状况、公司的广告等。一般各信息站点都尽量保证信息的时

间性。所以 Web 站点上的信息是动态的、经常更新的，这一点是由信息的提供者保证的。

6. 交互性

Web 的交互性首先表现在它的超链接上，用户的浏览顺序和所到站点完全由他自己决定。另外，通过表单(FORM)的形式可以从服务器方获得动态的信息。用户通过填写 FORM，可以向服务器提交请求，服务器可以根据用户的请求返回相应信息。

1.2 Web 客户端技术

客户端技术是 Web 程序中最重要技术之一。客户端技术主要是用来描述在浏览器中显示的页面、利用 JavaScript 技术对页面进行控制、与服务器进行通信等。常用的客户端技术主要包括 HTML、CSS、DOM、JavaScript、AJAX 等。通过这些客户端技术，读者可以编写具有良好用户体验的 Web 程序。

1. HTML

超级文本标记语言是标准通用标记语言下的一个应用，也是一种规范、一种标准，它通过标记符号来标记要显示的网页中的各个部分。网页文件本身是一种文本文件，通过在文本文件中添加标记符，可以告诉浏览器如何显示其中的内容(如文字如何处理、画面如何安排、图片如何显示等)。浏览器按顺序阅读网页文件，然后根据标记符解释和显示其标记的内容，对书写出错的标记将不指出其错误，且不停止其解释执行过程，编制者只能通过显示效果来分析出错原因和出错部位。但需要注意的是，对于不同的浏览器，对同一标记符可能会有不完全相同的解释，因而可能会有不同的显示效果。

【示例 1-1】一个简单的 HTML 程序(example1-2.html)，在页面上呈现 HelloWorld。

```
1  <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
2  <html>
3  <head>
4  <title>example1-1.html</title>
5  </head>
6  <body>
7      Hello World! <br>
8  </body>
9  </html>
```

2. CSS

层叠样式表(Cascading Style Sheets)是一种用来表现 HTML(标准通用标记语言的一个应用)或 XML(标准通用标记语言的一个子集)等文件样式的计算机语言。CSS 不仅可以静态地修饰网页，还可以配合各种脚本语言动态地对网页各元素进行格式化。

3.DOM

文档对象模型 (Document Object Model, DOM), 是 W3C 组织推荐的处理可扩展标记语言的标准编程接口。在网页上, 组织页面(或文档)的对象被组织在一个如图 1-1 所示的树形结构中, 用来表示文档中对象的标准模型, 就称为 DOM。

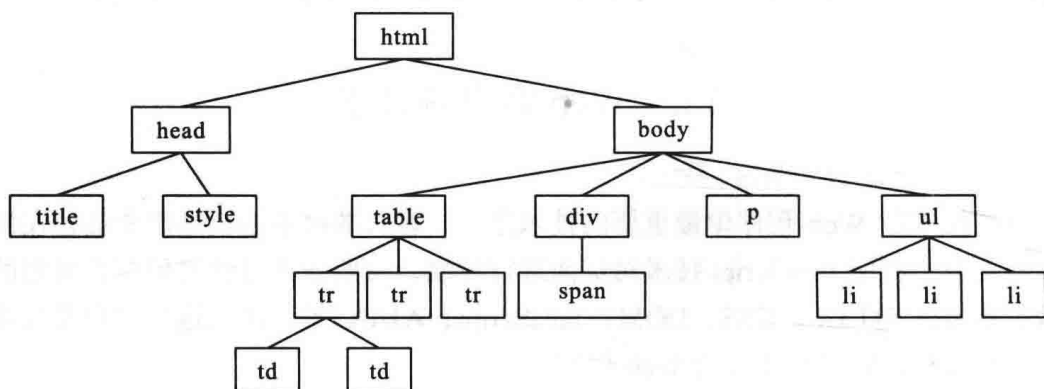


图 1-1 DOM 树状模型

4.JavaScript

JavaScript, 一种直译式脚本语言, 是一种动态类型、弱类型、基于原型的语言, 内置支持类型。它的解释器被称为 JavaScript 引擎, 是浏览器的一部分, 广泛用于客户端的脚本语言, 最早是在 HTML(标准通用标记语言下的一个应用)网页上使用, 用来给 HTML 网页增加动态功能。

【示例 1-2】一个简单的 JavaScript 程序(example2-1.html), 在页面上弹出提示窗口。

```
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
2 <html>
3 <head>
4 <title>example1-2.html</title>
5 <script type="text/javascript">
6     alert("Hello world!");
7 </script>
8 </head>
9 <body>
10 </body>
11 </html>
```

5.AJAX

AJAX, 即 “Asynchronous Javascript And XML” (异步 JavaScript 和 XML), 是指一种创建交互式网页应用的网页开发技术。传统的网页如果需要更新页面的内容, 需要重新

加载整个页面，而通过在后台与服务器进行少量数据交换，AJAX 可以使网页实现异步更新，即在不重新加载整个页面的情况下，对页面的某部分进行更新。

1.3 Web 服务器端技术

一个动态网站的开发，离不开服务器端技术。目前常用的服务器端技术主要有 ASP、ASP.NET、PHP、JSP 等。

1.ASP

ASP (Active Server Page, 动态服务器页面) 是微软公司开发的一种动态网页语言，它可以包含 HTML 标记、普通文本、脚本命令、COM 组件等。通过在页面代码中嵌入 VBScript 或 JavaScript 脚本语言，可以生成动态内容。在服务器端安装了适当的解释器后，页面的脚本语言可以被解释器执行，然后将执行结果与静态内容部分结合在一起后传送到客户端浏览器上。对复杂的操作，ASP 可以通过调用后台的 COM 组件完成，从而无限地扩充了其自身的能力。但由于 ASP 只能运行在 Windows 环境中，平台兼容性较差，限制了自身的发展，已经慢慢退出历史舞台。

2.ASP.NET

ASP.NET 是微软 .NET 框架的一部分，可以使用任何 .NET 兼容的语言来编写 ASP.NET 应用程序。它不是对 ASP 的简单升级，而是一种全新的交互式网页编程技术，是网站和 XML Web 服务的产物。使用 Visual Basic.NET、C#、J#、ASP.NET 页面 (Web Forms) 进行编译，可以提供比脚本语言更出色的性能表现。目前已经成为主流的动态网站技术之一。

3.PHP

PHP 是一种跨平台的服务器端嵌入式脚本语言。它的语法类似 C，并且混合了 Perl、C++ 和 Java 语言的一些特性，使得开发者能够快速写出动态页面。它是一种开源的 Web 服务器端脚本语言，支持目前绝大多数数据库。可以被多个平台支持，但广泛应用于 UNIX/Linux 平台。由于 PHP 本身的代码对外开放，并经过众多软件工程师的检测，所以其安全性能得到了业界的公认。

4.JSP

JSP (Java Server Page) 是新一代站点开发语言，完全解决了前面 ASP 和 PHP 的一个共同问题——脚本级执行。JSP 是以 Java 为基础开发的，可以在 Servlet 和 JavaBean 的支持下，完成功能强大的站点程序。JSP 可以被预编译，从而提高了程序的运行速度。同时，JSP 应用程序经过一次编译后可以随时随地运行。在绝大部分系统平台中，代码不需做任何修改即可在支持 JSP 的任何服务器上运行。

1.4 Web 应用架构

Web 应用的发展，带动了软件开发模式的变革。目前主流的 Web 应用架构有 C/S (Client/Server, 客户端/服务器) 结构和 B/S (Browser/Server, 浏览器/服务器) 结构。

1.4.1 C/S 结构

C/S 结构可以充分利用客户机和服务器两端的硬件环境优势，将任务合理分配到 Client 端和 Server 端来实现，降低了系统的通信开销。C/S 架构采用“功能分布”的原则：客户端负责数据处理、数据表示以及用户接口等功能；服务器端负责数据管理等核心功能。两端共同配合完成复杂的业务应用。这种结构能够充分发挥客户端 PC 的处理能力，业务在客户端处理后再提交给服务器也极大地提高了这个软件系统的响应速度。目前，涉及复杂业务逻辑的行业，例如银行内网系统、铁路航空售票系统、游戏软件等，常常选用 C/S 结构。其结构如图 1-2 所示。

1.4.2 B/S 结构

B/S 结构是 Web 兴起后的一种网络结构模式。考虑到 Web 浏览器是客户端最主要的应用软件。通过 Web 浏览器，用户不需要开发、安装任何客户端软件就可以进行软件业务流程的处理。这种模式就是 B/S 结构的核心：客户机上只要安装一个浏览器，如 Chrome、FireFox 或 Internet Explorer，服务器安装 SQL Server、Oracle、MYSQL 等数据库。用户通过浏览器向服务器发送请求，服务器接受请求，在数据库进行数据处理后，将结果响应给浏览器。B/S 结构通常应用于各大门户网站、各种信息管理系统和大型电子商务网站上。其结构如图 1-2 所示。

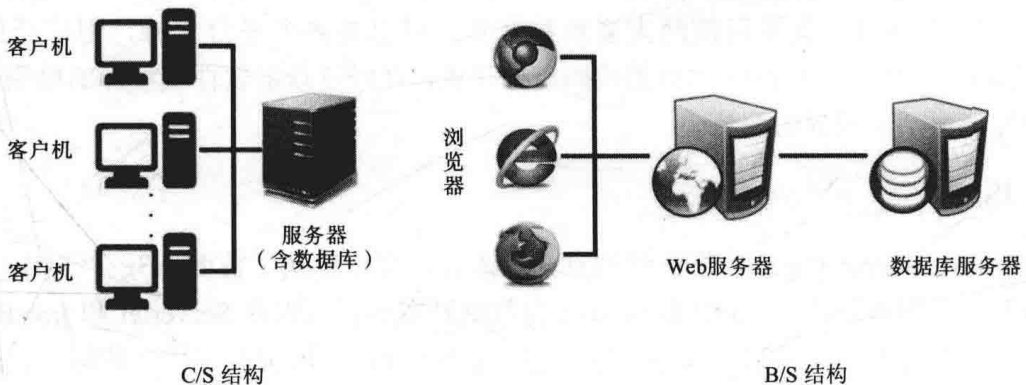


图 1-2 C/S 和 B/S 结构图

1.5 Web 工作机制

Web 的工作过程如下：

(1) 客户端通过浏览器发出要访问页面的 URL 地址，经过地址解析，找到服务器的 IP 地址，向该地址所指向的 Web 服务器发出请求。

(2) Web 服务器根据浏览器发送的请求，把 URL 地址转换成页面所在服务器上的文件名称，找到相应的文件。

(3) 如果 URL 指向 HTML 静态页面，Web 服务器使用 HTTP 协议将该文档直接送给客户端，由客户端浏览器负责解释执行。如果 HTML 文档中有 JSP、ASP、PHP 等动态代码，则由服务器运行这些程序。应用程序执行后的结果最后发送到客户端。

(4) 如果程序中包含对数据库的操作，则应用程序将查询指令发送给数据库驱动程序，由驱动程序对数据库进行操作。

(5) 数据库服务器将查询结果返回给数据库驱动程序，并由驱动程序返回给 Web 服务器。

(6) Web 服务器将结果数据嵌入到页面中相应的位置。

(7) Web 服务器将完成的页面以 HTML 格式发送给客户端。

(8) 客户端浏览器解释执行接受到的 HTML 文档，并显示结果。

其整个工作机制如图 1-3 所示。

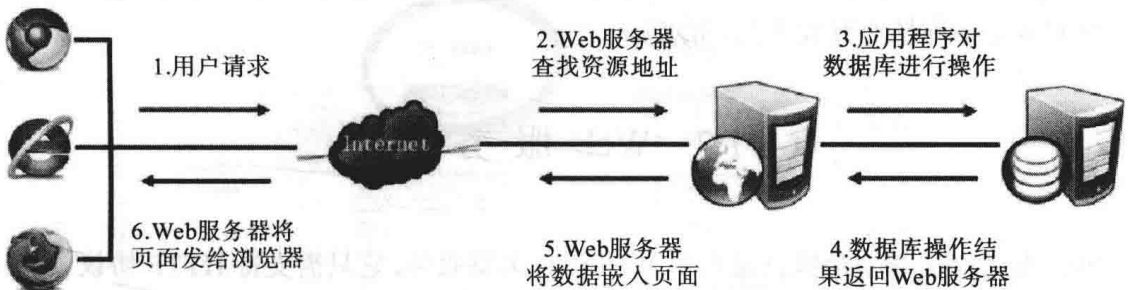


图 1-3 Web 工作机制

1.6 Java Web 应用的核心技术

Java Web 应用开发的核心技术主要有以下几种。

1.6.1 JSP

JSP (Java Server Page, Java 服务器页面) 是基于 Java 的技术，它在传统的网页 HTML