

高等学校计算机应用规划教材

# Java Web 程序设计教程

张永宾 辛 宇 王 攀 编著

清华大学出版社

北 京

## 内 容 简 介

本书是关于 Java Web 开发的基础教程。全书共分为 13 章,包括 Web 应用开发概述、Java EE 开发及运行环境、JSP 与 Servlet、JSTL、SQL 与 JDBC、MVC 与 Struts 2、Hibernate 与 ORM、Spring 框架、Spring 与 Hibernate 的整合以及 RESTful Web 服务等内容。

本教程内容丰富、结构合理、思路清晰、语言简练流畅、案例新颖、针对性强,主要面向 Java Web 应用开发的初学者,适合作为各种 Java Web 开发培训班的培训教材、高等院校的 Java Web 程序设计教材,还可作为 Java Web 应用开发人员的参考资料。

本书的电子课件、习题答案和实例源文件可以到 <http://www.tupwk.com.cn> 网站下载。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

### 图书在版编目(CIP)数据

Java Web 程序设计教程 / 张永宾, 辛宇, 王攀 编著. —北京: 清华大学出版社, 2017  
(高等学校计算机应用规划教材)

ISBN 978-7-302-47116-5

I. ①J… II. ①张… ②辛… ③王… III. ①JAVA 语言—程序设计—高等学校—教材 IV. ①TP312.8

中国版本图书馆 CIP 数据核字(2017)第 108577 号

责任编辑: 胡辰浩 李维杰

装帧设计: 孔祥峰

责任校对: 曹 阳

责任印制:

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座 邮 编: 100084

社 总 机: 010-62770175 邮 购: 010-62786544

投稿与读者服务: 010-62776969, [c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

质 量 反 馈: 010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

印 刷 者:

装 订 者:

经 销: 全国新华书店

开 本: 185mm×260mm 印 张: 23.5 字 数: 557 千字

版 次: 2017 年 5 月第 1 版 印 次: 2017 年 5 月第 1 次印刷

印 数: 1~4000

定 价: 48.00 元

---

产品编号:

# 前 言

随着 Internet 的迅猛发展, 以及“互联网+”的横空出世, Java 已经成为全球最流行、使用最广泛的 Web 开发语言之一。随着 Java 语言的推广和应用, 基于 Java 的各种针对 Web 开发的框架技术应运而生。本书以 Java Web 开发环境为背景, 从开发环境的搭建讲起, 遵循“从简单到复杂”、“从抽象到具体”的原则, 介绍了 Java Web 开发的核心技术, 以及 Web 开发的基本步骤和基本方法。

本书共 13 章, 第 1 章是 Web 应用开发概述, 主要介绍 Web 开发的基本概念和使用 Java 开发 Web 应用的相关技术与常用开发环境; 第 2 章介绍 Java EE 开发环境的搭建, 包括 JDK、Tomcat 以及 Eclipse 的下载、安装与配置; 第 3 章介绍了 JSP 和 Servlet, 包括 JSP 的 3 个编译指令、7 个动作指令和 9 个内置对象, 以及 Servlet 的工作原理、过滤器与监听器等。JSP 和 Servlet 是开发 Java Web 应用程序的两种基本技术, 一些主流框架都是以这些基本技术为基础演变而来的; 第 4 章介绍了 JSTL 标签库的使用; 第 5 章介绍了 Struts 2 框架的基本知识, 包括 Struts 2 与 MVC、Struts 2 的工作流程、Action 以及拦截器等; 第 6 章介绍了 SQL 与 JDBC, 这是应用程序与数据库交互的基础; 第 7 章和第 8 章介绍了 Hibernate 框架, 包括 Hibernate 环境的搭建、Hibernate 的工作流程、事务控制以及缓存机制等; 第 9 章开始学习 Spring 框架, 首先介绍的是 Spring 的 IoC 与 AOP; 第 10 章介绍 Spring Web MVC, 它是 Spring 框架中用于 Web 应用开发的一个模块, 是当今流行的 Web 开发框架之一, 也是本书学习的重点; 第 11 章介绍了 Spring MVC 与 Hibernate 的整合, 这也是实际项目中应用最广泛的一种框架组件; 第 12 章介绍了 RESTful Web 服务, 包括如何创建和测试 RESTful Web 服务; 第 13 章是一个综合实例, 使用 Spring MVC + Hibernate 框架开发了一个简易的图书馆管理系统。

本书内容丰富、结构合理、思路清晰、语言简练流畅、案例新颖、针对性强。每一章的开始部分概述了该章的作用和内容, 指出该章的学习目标。正文部分结合每章的知识点和关键技术, 穿插了大量极富实用价值的示例, 所有示例都在 Eclipse + Tomcat 8.5 + JDK 1.8 环境下调试运行通过。每一章的末尾有本章小结, 总结本章的内容、重点与难点; 同时安排了有针对性的思考和练习, 帮助读者巩固本章所学内容, 提高读者的实际动手能力。

本书主要面向 Java Web 应用开发的初学者, 适合作为各种 Web 开发培训班的培训教材、高等院校 Java Web 开发相关课程的教材及 Web 应用程序开发人员的参考资料。

本书分为 13 章, 其中黑龙江科技大学的张永宾编写了第 1~第 7 章, 哈尔滨理工大学的辛宇编写了第 8~第 11 章, 黑龙江科技大学的王攀编写了第 12 和第 13 章。另外, 参加本书编写的人员还有杨春元、周高翔、荆双燕、徐泉楠、余泉灵、臧俊丽、卢华林、肖乐

民、王朋丽、李文静、冯波、马金帅、木德朋、董文燕、张亚威、陈梦麟、霍利凡、杨爱提、马玉芬、靳守燕、崔增玉、许素芳等。由于作者水平有限，本书难免有不足之处，欢迎广大读者批评指正。我们的 email 是 [huchenhao@263.net](mailto:huchenhao@263.net)，联系电话是 010-62796045。

本书的电子课件、习题答案和实例源文件可以到 <http://www.tupwk.com.cn> 网站下载。

作 者  
2017年3月

# 目 录

第 1 章 Web 应用开发概述.....1	2.5 思考和练习 .....35
1.1 Web 应用概述.....1	第 3 章 JSP 与 Servlet ..... 36
1.1.1 Web 与 Web 应用.....1	3.1 一个简单的JSP+Servlet应用.....36
1.1.2 Web 应用是如何运行的 .....3	3.1.1 创建 Servlet 类.....36
1.1.3 服务端开发技术对比 .....4	3.1.2 部署 Servlet .....39
1.2 使用 Java 开发 Web 应用.....5	3.1.3 创建 JSP 文件.....41
1.2.1 Java 语言简介.....6	3.2 JSP 技术初步 .....42
1.2.2 丰富的框架技术.....8	3.2.1 JSP 的工作原理 .....42
1.2.3 Java Web 应用的核心技术.....9	3.2.2 JSP 文件中的内容.....43
1.2.4 使用 Ajax 和 jQuery 提升 用户体验 .....10	3.2.3 JSP 的内置对象 .....49
1.3 集成开发环境介绍.....11	3.2.4 JSP 中的中文显示问题.....54
1.3.1 常用的 Java IDE.....11	3.3 Servlet 的开发与应用 .....60
1.3.2 Web 服务器汇总 .....12	3.3.1 Servlet 的生命周期.....60
1.4 本章小结.....13	3.3.2 使用 HttpServletRequest .....62
1.5 思考和练习.....14	3.3.3 使用 HttpServletResponse .....64
第 2 章 Java EE 开发及运行环境 ..... 15	3.3.4 使用过滤器.....66
2.1 下载并安装 JDK.....15	3.3.5 使用监听器.....71
2.1.1 安装 JDK .....15	3.4 使用 jQuery 异步请求数据 .....73
2.1.2 配置环境变量 .....16	3.4.1 下载 jQuery 库 .....74
2.2 Tomcat 的安装与配置 .....18	3.4.2 JSON 简介.....74
2.2.1 下载并安装 Tomcat .....18	3.4.3 jQuery 与 Ajax .....75
2.2.2 在 Tomcat 中部署和卸载 应用程序 .....21	3.5 本章小结.....79
2.3 Eclipse 开发环境介绍 .....24	3.6 思考和练习 .....80
2.3.1 安装 Eclipse .....24	第 4 章 使用 JSP 标签库 ..... 81
2.3.2 使用 Eclipse 新建 Java EE 应用 .....24	4.1 JSP 标准标签库(JSTL).....81
2.3.3 在 Eclipse 中使用 Tomcat .....27	4.1.1 下载 JSTL 安装包 .....82
2.3.4 Eclipse 的常用快捷键.....33	4.1.2 表达式语言 EL .....82
2.4 本章小结.....35	4.1.3 使用核心标签库 (C 名称空间).....86
	4.1.4 使用国际化和格式化 标签库(FMT 命名空间).....94

4.1.5 使用 SQL 标签库 (SQL 名称空间).....	96	6.1.1 下载并安装 MySQL.....	133
4.1.6 使用 XML 标签库 (X 名称空间).....	97	6.1.2 安装 MySQL JDBC 驱动.....	135
4.1.7 使用 JSTL 函数.....	97	6.2 SQL 语言简介.....	135
4.2 用 JSTL 开发用户管理系统.....	98	6.2.1 SQL 概述.....	135
4.2.1 数据库设计.....	98	6.2.2 SQL 数据类型.....	136
4.2.2 开发与实现.....	99	6.2.3 常有 SQL 语句.....	137
4.2.3 部署并测试应用.....	103	6.3 JDBC.....	141
4.3 自定义标签库.....	104	6.3.1 JDBC 概述.....	142
4.3.1 一个最简单的自定义 标签.....	104	6.3.2 JDBC 驱动程序.....	143
4.3.2 访问标签体.....	107	6.3.3 使用 JDBC 操作数据库.....	143
4.3.3 自定义标签属性.....	108	6.3.4 JDBC 示例.....	146
4.4 本章小结.....	109	6.4 本章小结.....	154
4.5 思考和练习.....	109	6.5 思考和练习.....	154
<b>第 5 章 Struts 2 框架基础.....</b>	<b>110</b>	<b>第 7 章 Hibernate 框架基础.....</b>	<b>155</b>
5.1 MVC 框架.....	110	7.1 ORM 简介.....	155
5.1.1 框架内容.....	110	7.1.1 应用 ORM 的意义.....	155
5.1.2 框架和设计模式的区别.....	111	7.1.2 流行的 ORM 框架.....	156
5.1.3 MVC 的优点.....	112	7.1.3 使用 Hibernate ORM 的 原因.....	157
5.2 Struts 2 基础.....	113	7.2 一个简单的 Hibernate 应用.....	158
5.2.1 Struts 2 中的 MVC.....	113	7.2.1 下载 Hibernate 资源包.....	158
5.2.2 Struts 2 的工作流程.....	114	7.2.2 在 Eclipse 中引入 Hibernate 的 JAR 包.....	158
5.2.3 一个简单的 Struts 2 应用.....	115	7.2.3 使用 Hibernate 代替 JDBC.....	161
5.2.4 Action 详解.....	118	7.3 认识 Hibernate ORM.....	167
5.2.5 struts.xml 配置详解.....	119	7.3.1 Hibernate 的框架结构.....	167
5.2.6 Struts 2 标签库.....	123	7.3.2 Hibernate 配置文件详解.....	170
5.3 拦截器.....	126	7.3.3 使用 Hibernate 映射文件.....	171
5.3.1 拦截器的工作机制.....	126	7.3.4 Hibernate 的工作流程.....	174
5.3.2 Struts 2 内置拦截器.....	127	7.4 Hibernate 查询.....	175
5.3.3 自定义拦截器.....	130	7.4.1 Hibernate 的检索策略.....	175
5.4 本章小结.....	131	7.4.2 Hibernate 的关联查询.....	181
5.5 思考和练习.....	132	7.4.3 Hibernate 的查询方式.....	189
<b>第 6 章 SQL 与 JDBC.....</b>	<b>133</b>	7.5 本章小结.....	192
6.1 准备关系数据库.....	133	7.6 思考和练习.....	192

<b>第 8 章 Hibernate 性能优化</b> .....	<b>194</b>		
8.1 Hibernate 事务与并发 .....	194		
8.1.1 什么是事务 .....	194		
8.1.2 Hibernate Session 和 事务范围 .....	195		
8.1.3 并发控制 .....	196		
8.2 Hibernate 缓存 .....	202		
8.2.1 Hibernate 缓存概述 .....	202		
8.2.2 应用一级缓存 .....	203		
8.2.3 应用二级缓存 .....	205		
8.2.4 查询缓存 .....	208		
8.3 本章小结 .....	210		
8.4 思考和练习 .....	211		
<b>第 9 章 Spring 框架基础</b> .....	<b>212</b>		
9.1 Spring 框架简介 .....	212		
9.1.1 概述 .....	212		
9.1.2 Spring 框架的模块结构 .....	214		
9.1.3 Spring 4.x 新特性 .....	216		
9.2 从 Hello World 开始 .....	218		
9.2.1 下载 Spring 资源包 .....	218		
9.2.2 基于 Spring 的 Hello World .....	220		
9.3 Spring IoC .....	222		
9.3.1 IoC 基础 .....	222		
9.3.2 IoC 容器 .....	223		
9.3.3 bean 的装配 .....	225		
9.4 Spring AOP .....	232		
9.4.1 什么是 AOP .....	232		
9.4.2 AOP 相关概念 .....	233		
9.4.3 使用 Spring 的通知 .....	233		
9.5 本章小结 .....	236		
9.6 思考和练习 .....	236		
<b>第 10 章 Spring Web MVC</b> .....	<b>237</b>		
10.1 Spring Web MVC 入门 .....	237		
10.1.1 Spring Web MVC 是什么 .....	237		
10.1.2 为什么使用 Spring Web MVC .....	238		
10.1.3 Spring Web MVC 的 工作流程 .....	239		
10.1.4 Spring MVC 的 Hello World 程序 .....	240		
10.2 深入学习 Spring Web MVC .....	242		
10.2.1 启动 Spring MVC .....	242		
10.2.2 DispatcherServlet 组件类 .....	245		
10.2.3 使用 @RequestMapping .....	248		
10.2.4 控制器方法的参数 .....	251		
10.2.5 控制器方法的返回 类型 .....	259		
10.2.6 模型与视图 .....	261		
10.2.7 Spring MVC 的表单 标签库 .....	272		
10.3 本章小结 .....	277		
10.4 思考和练习 .....	277		
<b>第 11 章 Spring MVC 整合 Hibernate</b> .....	<b>278</b>		
11.1 Spring 提供的 DAO 支持 .....	278		
11.1.1 J2EE 应用的 3 层架构 .....	278		
11.1.2 Spring 的 DAO 理念 .....	279		
11.1.3 使用 @Repository 注解 .....	280		
11.2 Spring MVC 整合 Hibernate 5 .....	281		
11.2.1 新建工程 .....	281		
11.2.2 创建实体类 .....	281		
11.2.3 创建 Dao 层 .....	286		
11.2.4 创建 Service 层 .....	291		
11.2.5 创建 Controller 控制器 .....	292		
11.2.6 创建 JSP 页面 .....	294		
11.2.7 配置 Spring 和 Hibernate .....	297		

11.2.8 引入 aspectjweaver.JAR 包.....	303
11.2.9 项目运行结果 .....	304
11.3 本章小结.....	305
11.4 思考和练习 .....	305
<b>第 12 章 创建 RESTful Web 服务.....</b>	<b>306</b>
12.1 Web 服务概述.....	306
12.1.1 基于 SOAP 的 Web 服务 .....	306
12.1.2 RESTful Web 服务 .....	307
12.2 创建 RESTful Web 服务.....	313
12.2.1 新建工程 .....	313
12.2.2 创建实体类、DAO 层 和 Service 层.....	314
12.2.3 基于 REST 的控制器.....	316
12.2.4 添加配置信息 .....	318
12.3 测试 RESTful Web 服务.....	319
12.3.1 使用 RESTClient Firefox 插件 .....	319
12.3.2 使用 REST 模板编写 REST 客户端 .....	323
12.4 本章小结.....	325
12.5 思考和练习 .....	326
<b>第 13 章 图书馆管理系统.....</b>	<b>327</b>
13.1 系统概述.....	327
13.1.1 项目背景 .....	327
13.1.2 需求分析 .....	327
13.2 数据库设计 .....	328
13.2.1 系统 E-R 图 .....	328
13.2.2 数据表设计.....	329
13.3 系统设计与实现.....	331
13.3.1 搭建系统框架 .....	332
13.3.2 配置 Spring 与 Hibernate .....	333
13.3.3 创建实体类.....	334
13.3.4 管理员登录功能.....	334
13.3.5 管理员管理功能.....	339
13.3.6 读者管理功能 .....	347
13.3.7 图书管理功能 .....	350
13.3.8 读者登录及操作功能 .....	357
13.4 系统运行结果 .....	360
13.4.1 管理员操作页面.....	360
13.4.2 读者操作页面 .....	363
13.5 本章小结.....	364
13.6 思考和练习 .....	364
<b>参考文献.....</b>	<b>365</b>

# 第1章 Web应用开发概述

Web使用超文本技术将Internet上的资源以页面的形式展示出来，Web应用是一种使用HTTP作为核心通信协议，通过Internet让Web浏览器和服务器通信的计算机程序。本章将从Web应用的基本概念开始，讲述Java Web应用开发相关的技术和常用框架，最后介绍常用的集成开发环境与Web服务器。

**本章学习目标：**

- 理解Web和Web应用的基本概念
- 掌握Web应用的工作原理
- 了解服务器端开发技术
- 了解Java语言的特点与发展前景
- 熟悉常用的Java Web框架技术
- 了解常用的集成开发环境

## 1.1 Web应用概述

Internet采用超文本和超媒体的信息组织方式，将信息的链接扩展到整个Internet上。而Web就是一种超文本信息系统，它使得文本不再像一本书一样是固定的、线性的，而是可以从一个位置跳到另一个位置并从中获取更多的信息。

### 1.1.1 Web与Web应用

Web应用是一种可以通过Web访问的应用程序。那么Web是如何发展而来的呢？什么样的应用才是Web应用呢？

#### 1. Web的发展

Web是World Wide Web的简称，中文译为万维网，是Internet上的一种服务。Web使用超文本技术将Internet上的资源以页面的形式展示出来，以供用户使用。Web上的资源十分丰富，包括图片、文本、声音、视频等多媒体元素。通常所说的网页是一个包含HTML标签的纯文本文件(文件扩展名为.html、.htm、.asp、.aspx、.php或.jsp等)，它可以存放在世界某个角落的某一台计算机中，是万维网中的一“页”。

Web在组成上包括以下两部分：

- **服务器：**通常包括物理设备和软件应用程序，物理设备是指存放供用户访问的信息资源的远程计算机；软件应用程序是指能够根据用户的请求将信息资源传递给

用户的应用程序，如Apache服务器。

- 客户端：通常是指客户使用的本地计算机，通过客户端浏览器向服务器发送请求，然后接收并显示服务器传递过来的信息资源。

发展到今天，Web共经历了两个阶段：Web 1.0和Web 2.0。其中Web 1.0被称为Internet第一代，指的是2003年以前的Internet模式。在Web 1.0时代，Internet采用的是技术创新主导模式。而Web 2.0则是以Internet为平台，以用户为灵魂，允许许多人参与，以可读、可写的模式成为Internet新的发展趋势。如果说Web 1.0的本质是联合，那么Web 2.0的本质就是互动，它让网民更多地参与信息产品的创造、传播和分享，而这个过程是有价值的。

在Web 2.0的基础上，又有人提出了Web 3.0，不过对Web 3.0的争议比较大，常见的对Web 3.0的解读是：网站内的信息可以直接和其他网站的相关信息交互，能通过第三方信息平台同时对多家网站的信息进行整合使用；用户在互联网上拥有自己的数据，并能不同网站上使用。不管Web 3.0最终将向何处发展，但不可否认的是，从Web诞生至今，它不仅改变着人们联系、交流、获取知识的方式，而且也在改变着商业运行模式，已经成为人们日常生活和工作不可缺少的一部分。

## 2. 什么是Web应用

我们通过浏览器可以访问百度、淘宝、网易等网站，这些就是Web应用程序(简称Web应用)。最初，这些网站上的内容都是由静态页面组成的，页面上包含一些文本、图片等信息资源，用户可以通过链接来浏览信息。采用静态页面的缺陷非常多，如不能与用户进行交互，不能实时更新Web上的内容，因此像搜索引擎、股票行情等许多功能无法实现。于是出现了动态页面，即根据不同的用户或在不同的时间，呈现给用户的信息资源也不相同。这里的动态内容就是由Web应用程序来实现的。

Web应用程序是一种使用HTTP(HyperText Transfer Protocol，超文本传输协议)作为核心通信协议，通过Internet让Web浏览器和服务器通信的计算机程序。

一个Web应用程序是由完成特定任务的各种Web组件构成的并通过Web将服务展示给外界。在实际应用中，Web应用程序由多个Servlet、JSP页面、HTML文件以及图像文件等组成。所有这些组件相互协调，从而为用户提供一组完整的服务。

Web应用中的每一次数据交换都要涉及客户端和服务器端两个层面。因此，Web应用程序的开发技术分为客户端开发技术和服务器端开发技术两种。

## 3. 客户端开发技术

常用的客户端开发技术包括如下内容：

- HTML：超文本标记语言，是Web的描述语言。无论哪种动态页面开发技术，都无法摆脱HTML的影子。HTML是所有动态页面开发技术的基础，这些动态页面开发技术无非是在静态HTML页面的基础上添加了动态的可以交互的内容。
- CSS：层叠样式表(Cascading Style Sheets)，也就是通常所说的样式表，用于增强控制网页样式并允许将样式信息与网页内容分离的一种标记性语言。通过使用CSS，可以方便、灵活地设置网页中不同元素的外观属性，通过这些设置可以使网页在外观上达到一个更高的级别。

- **ActiveX**: 一个集成平台, 使用ActiveX可以方便地在Web页中插入多媒体效果、交互式对象、复杂程序等。
- **JavaScript**: JavaScript 是一种简单的脚本语言, 可以在浏览器中直接运行, JavaScript的出现给静态的HTML网页带来很大的变化。JavaScript增加了HTML网页的互动性, 使以前单调的静态页面变得富有交互性。它可以在浏览器端实现一系列动态的功能, 仅仅依靠浏览器就可以完成一些与用户的互动。
- **jQuery**: 一个快速、简洁的JavaScript框架, 是继Prototype之后又一个优秀的JavaScript代码库。jQuery具有独特的链式语法和短小清晰的多功能接口; 具有高效灵活的CSS选择器, 并且可对CSS选择器进行扩展; 拥有便捷的插件扩展机制和丰富的插件。
- 其他: VBScript、Applet等。

#### 4. 服务器端开发技术

常用的服务端开发技术包括如下内容:

- **JSP/Servlet**: 服务器端的Java应用程序, 可以生成动态的Web页面。
- **PHP**: 在服务器端执行的嵌入HTML文档的脚本语言。
- **ASP/ASP.NET**: 微软公司推出的用于构建Windows服务器平台上的Web应用程序。
- 其他: CGI、Perl、ISAPI等。

应用程序的工作模式分为两种: C/S模式和B/S模式。在C/S模式(Client/Server, 客户端/服务器模式)中, 客户端需要安装专用的客户端软件, 程序的维护和升级成本比较高; B/S模式(Browser/Server, 浏览器/服务器模式)的应用程序则需要通过浏览器来运行。Web应用程序一般采用的是B/S模式。

采用B/S模式的Web应用程序分为3层结构:

- **表示层**: 采用Web浏览器实现。
- **业务逻辑层**: 由位于Web服务器上的各种服务器端程序实现。
- **数据服务层**: 由数据库服务器提供, 数据库服务器有MySQL、Oracle等。

随着Internet和手机上网的普及, Web应用程序已经成为目前最流行的应用程序类型。

### 1.1.2 Web应用是如何运行的

随着网络的普及, 大家对上网的过程越来越熟悉, 上网的一般过程如下:

- (1) 打开浏览器。
- (2) 输入某个网址。
- (3) 经过一段时间的等待, 浏览器显示要访问的信息。

然后可以在网页上继续进行其他操作, 例如: 在网页上单击超链接, 访问其他内容; 或者在网页中输入一些信息, 然后单击按钮, 等待浏览器中内容的再次更新。

不管是在地址栏中输入地址, 还是单击超链接或者单击按钮, 都需要等待浏览器中内容的更新。等待浏览器内容更新的过程, 实际上就是浏览器访问 Web 应用的过程。这个过程如下:

(1) 浏览器根据我们输入的地址找到相应的服务器，不同的网站对应不同的服务器。这个服务器通常称为 Web 服务器，可以接收浏览器发送的请求。

(2) Web 服务器根据请求的内容调用不同的服务器端程序，服务器端程序通常也是一个服务器，称为应用服务器。

(3) 应用服务器接收到请求之后，查找相应的文件，加载并执行相应的任务。如果涉及数据处理，则需要与数据库服务器交互。

(4) 处理完数据，将处理结果返回给应用服务器，服务器端程序的执行结果通常是 HTML 文档。

(5) 应用服务器把执行的结果返回给 Web 服务器，Web 服务器再把这个结果返回给客户端浏览器。

(6) 浏览器解析 HTML 文档，然后把解析后的网页显示给最终用户。

Web 应用的工作原理如图 1-1 所示。

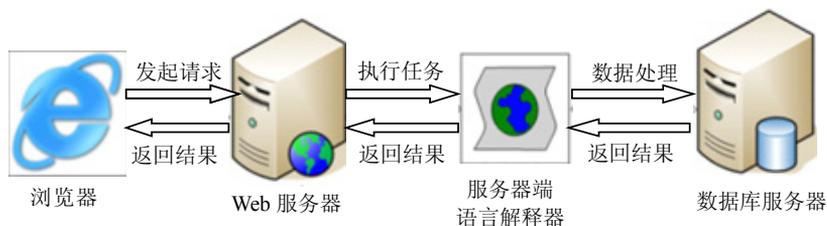


图1-1 Web应用的工作原理

### 1.1.3 服务端开发技术对比

在Web发展的最初阶段，所有网页都是静态的HTML网页。在这种情况下，网站所能实现的任务仅仅是静态的信息展示，而不能与客户进行互动，当然这样的网站是不能满足用户不同需要的。

为了满足不同用户各种各样的需求，需要网站或Web应用程序具有收集并处理用户需求的功能，这就有了后来一系列的动态页面语言。

所谓动态页面，是指可以和用户进行交互，能根据用户的输入信息产生相应响应的页面，能满足这种需求的语言可以称之为动态语言。

较早的动态网页技术主要使用CGI，现在常用的动态网页技术有ASP/ASP.NET、JSP、PHP等，下面分别介绍这几种动态语言。

#### 1. CGI

在互联网发展的早期，动态网页技术主要使用CGI(Common Gate Interface，通用网关接口)。CGI程序被用来解释处理表单中的输入信息，并在服务器中产生对应的操作处理，或是把处理结果返回给客户端的浏览器，从而可以给静态的HTML网页添加动态的功能。但是由于CGI程序在编程中比较困难、效率低下，而且修改维护也比较复杂，因此在一段时间以后，CGI逐渐被其他新的动态网页技术所替代。

## 2. ASP/ASP.NET

ASP是微软公司推出的一种动态网页语言，它可以将用户的HTTP请求传入到ASP的解释器中。这个解释器对这些ASP脚本进行分析和执行，然后从服务器返回处理的结果，从而实现了与用户交互的功能。ASP的语法比较简单，对编程基础没有很高的要求，所以很容易上手，而且微软提供的开发环境的功能十分强大，这更是降低了ASP程序开发的难度。但是ASP也有自身的缺点。ASP在本质上还是一种脚本语言，除了使用大量的组件，没有其他办法提高效率，而且ASP还只能运行在Windows环境中，这样 Windows自身的一些限制就制约了ASP的发挥，这些都是使用ASP无法回避的弊端。

ASP.NET是微软推出的用于替代ASP的一项技术，它不是ASP的升级版本，而是基于.NET框架的用于Web应用开发的技术。ASP.NET属于编译语言，这是相对ASP的最大不同之处。ASP.NET的运行需要.NET环境，适用于中小型网站的开发。

## 3. JSP

JSP(Java Server Page)是Sun公司开发的一种服务器端的脚本语言，自从1999年推出以来，逐步发展为开发Web应用的一项重要技术。JSP可以嵌套在HTML中，而且支持多个操作系统平台。一个用JSP开发的Web应用系统，不用做什么改动就可以在不同的操作系统中运行。

JSP本质上就是把Java代码嵌套到HTML中，然后经过JSP容器的编译执行，可以根据这些动态代码的运行结果生成对应的HTML代码，从而可以在客户端的浏览器中正常显示。

由于JSP中使用的是Java语法，因此Java语言的所有优势都可以在JSP中体现出来。尤其是J2EE中的强大功能，更是成为JSP语言发展的强大后盾。

## 4. PHP

PHP和JSP类似，都是可以嵌套到HTML中的语言，不同之处在于：PHP的语法比较独特，它混合了C、Java等多种语法中的优秀部分，而且PHP网页的执行速度要比CGI和ASP页面等快很多。在PHP中，提供了对常见数据库的支持，例如SQL Server 2000、MySQL、Oracle、Sybase等，这种内置的方法使PHP中的数据库操作变得异常简单。而且PHP程序可以在IIS和Apache中运行，提供对多种操作系统平台的支持。

但是PHP也存在一些劣势，PHP开发运行环境的配置比较复杂，而且PHP是开源的产品，缺乏正规的商业支持。这些因素在一定程度上限制了PHP的进一步发展。

总之，各种动态语言都有着自身的优势和劣势，可以根据客户的需求来选择具体的语言。只要能够保证系统的性能和功能，选择什么语言无关紧要。

# 1.2 使用Java开发Web应用

Java提供的JSP和Servlet是开发Web应用的两项引人注目的技术，同时它的开源项目也是层出不穷，如Web框架Struts、Struts 2等，持久层框架Hibernate、iBATIS等，J2EE框架

Spring, 模板引擎Velocity、FreeMarker等。

### 1.2.1 Java语言简介

Java是一种跨平台的面向对象编程语言,由Sun公司于1995年推出。Java语言自从问世以来,受到越来越多开发者的喜爱。在Java语言出现以前,很难想象在Window环境下编写的程序可以不加修改就在Linux系统中运行,因为计算机硬件只识别机器指令,而不同操作系统中的机器指令是有所不同的。所以,要把一种平台下的程序迁移到另一种平台,必须针对目标平台进行修改。如果想要程序运行在不同的操作系统上,就要求程序设计语言能够跨平台,可以跨越不同的硬件、软件环境,而Java语言就能够满足这种要求。

#### 1. Java语言的特点

Java 是一种优秀的面向对象语言。在Java语言中,有着健壮的安全设计,它的结构是中立的,可以移植到不同的系统平台。优秀的多线程设计也是Java语言的一大特色。

目前,Java语言最大的用途就是Web应用的开发。使用Java语言可以不用考虑系统平台的差异,在一种系统下开发的应用系统,可以不加任何修改就能运行在另一种不同的系统中。例如开发人员在Windows平台下开发的Web应用程序,可以直接部署在Linux或Unix服务器系统中。

Java语言之所以如此受欢迎,是由其自身的优点决定的,以下简单介绍Java语言的特性:

- **平台无关性:** 在Java中,并不是直接把源文件编译成硬件可以识别的机器指令。Java的编译器把Java源代码编译为字节码文件,这种字节码文件就是编译Java源程序时得到的class类文件,执行这种类文件的是Java虚拟机。Java虚拟机是软件模拟出的计算机,可以执行编译Java源文件得到的中间码文件,而各种平台的差异就是由Java虚拟机来处理的,从而实现了可以在各种平台上运行Java程序的目的。
- **安全性:** Java语言放弃了C/C++中的指针操作。在Java中,没有显式提供指针操作,不提供对存储器空间直接访问的方法,这样就可以保证系统的地址空间不会被有意或无意破坏。而且经过这样的处理,也可以避免系统资源的泄漏。例如在 C/C++中,如果指针不及时释放,就会占用系统内存空间,而Java提供了一套有效的资源回收策略,会自动回收不再使用的系统资源,从而保证了系统的安全性和稳定性。
- **面向对象:** 面向对象是现代软件开发中的主流技术,Java语言继承了C++面向对象的理论,并简化了这种面向对象的技术,去掉了一些复杂的技术,例如多继承、运算符的重载等功能。在Java程序中,所有的操作都是在对象的基础上实现的,为了实现模块化和信息的隐藏,Java语言采用了将功能代码封装的处理方法,Java语言对继承性的实现使功能代码可以重复利用。用户可以把具体的功能代码封装成自定义的类,从而实现对代码的重用。
- **异常处理:** Java中的异常处理可以帮助用户定位处理各种错误,从而大大缩短了Java应用程序的开发周期。而且这种异常策略可以捕捉到程序中的所有异常,针对

不同的异常，用户可以采取具体的处理方法，从而保证了应用程序在用户的控制下运行，保证了程序的稳定和健壮。

## 2. Java语言的发展

Java语言和Java平台的发展是一段漫长而富于传奇的历史。从20世纪90年代中期发明开始，Java已经经历了许多变化，也遇到过许多争论。在早期，Java被称为Java开发工具包或JDK，是一门与平台(由一组必需的应用程序编程接口(API)组成)紧密耦合的语言。

Java语言和Java SE(Standard Edition)平台一直是共同发展的——它们的新版本总是会同时发布，并且彼此紧密耦合。从1997年的1.1版本开始，该平台被称为JDK，但到了1.2版本，JDK和平台不再是同一技术。从1998年底的1.2版本开始，Java技术栈被分割为如下关键部分：

- **Java**：它是一门包含了严格和强类型语法的面向对象编程语言。
- **Java 2平台标准版本**：也被称为J2SE，指的是平台以及java.lang和java.io包中包含的类。它是构建Java应用程序的基础。
- **Java虚拟机或JVM**：它是一个可以运行Java程序的软件虚拟机。因为被编译过的Java代码只是字节码，JVM将在运行代码之前，把字节码编译成机器码(通常被称作即时编译器或JIT编译器)。JVM还负责管理内存，从而实现了应用程序代码的简化。
- **Java开发工具包或JDK**：Java开发者创建应用程序所需的软件。它包含了Java语言编译器、文档生成器、与本地代码协作的工具和用于调试平台类的Java源代码。
- **Java运行时环境或JRE**：终端用户用于运行编译后Java应用程序的软件。它包含了JVM但不含任何JDK中的开发工具。不过，JDK中也包含JRE。

这5个组件曾经都只是规范，而不是实现。任何公司都可以创建自己的Java技术栈实现，并且许多公司已经这样做了。尽管Sun提供了Java、J2SE、JVM、JDK和JRE的标准实现，但IBM、Oracle和Apple仍然创建了包含不同特性的实现。

## 3. 企业级Java的诞生

随着Internet的发展和Web应用程序的流行，Sun公司已经意识到应用程序开发对高级开发工具的需求。1998年，就在J2SE 1.2发布之前，Sun宣布正在开发一个称为Java专业版本或JPE的产品。同时Sun还研发了一门称为Servlet的技术，这是一个能够处理HTTP请求的小型应用程序。

Servlet和JPE在经历过几次内部迭代过程之后，Sun于1999年12月12日发布了Java 2平台的企业版(J2EE, Java 2 Enterprise Edition)，版本为1.2。J2EE包含J2SE中的类，并且还包含用于开发企业级应用的类，比如EJB、Servlet、JSP、XML、事务控制等。在随后发布的版本中，J2EE迅速成为对J2SE的补充，并且随着多年的发展，一些组件已经被认为必须从J2EE迁移到J2SE中。随着版本的不断升级，从JDK 5.0开始，不再叫J2SE和J2EE了，而改名为Java SE和Java EE了，因为那个“2”已经失去了其本应该有的意义。

## 4. Java语言的发展前景

虽然说Java语言并不是为网络环境设计的，但是Java语言目前还是主要被用于网络环

境中，尤其是在服务器端的程序设计中，Java语言的地位是其他动态语言所无法替代的。尤其是在B/S开发模式盛行的今天，Java语言的地位更是举足轻重。在Java EE中，提供了优秀的B/S应用程序的解决方案。再加上Java语言跨平台、简单易用等特性，用户自然会选择Java语言进行开发。随着网络技术的急速发展，Java语言必然会取得更大的发展，在这个复杂的网络环境中，Java语言有着广阔的前景。

Java语言在不断发展和完善，现在各大厂商都在努力推动Java技术的发展。在这些厂商中间，Sun(2010年被Oracle公司收购)、IBM、Sybase等做的都相当出色，而且在Java领域中，还有开源力量的支持，例如Apache、JBoss等。这些开源力量给Java的发展带来巨大的推动作用，很多优秀的Java框架都是由这些开源力量开发和维护的。

## 1.2.2 丰富的框架技术

框架其实就是可重用的设计架构，应用框架强调的是软件的设计重用性和系统的可扩充性，以缩短大型应用软件系统的开发周期，提高开发效率和质量。

使用Java开发Web应用的常用框架有很多，下面简要介绍一些比较常用的框架。

### 1. Struts/Struts 2

Struts是Apache基金会Jakarta项目组的一个开源项目，是一个基于Java EE平台的MVC(模型-视图-控制器)框架，它将Servlet和JSP标签作为实现自身功能的一部分。

Struts 2建立在Struts框架和WebWork框架基础之上，集成了二者的优点，是早期Java开发比较流行的Web框架。

### 2. WebWork

WebWork是由OpenSymphony组织开发的，是一个基于Web的MVC框架。它在运行时通过Interceptor(拦截器)自动应用，因此脱离了Action类。

### 3. Spring

Spring是一个以IoC(Inversion of Control，控制反转)和AOP(Aspect Oriented Programming，面向切面编程)为核心的轻量级容器框架。它提供了一系列的Java EE开发解决方案，包括表示层的Spring MVC、持久层的Spring JDBC、业务层事务管理等众多的企业级应用技术。目前，Java Web应用开发的主流框架就是Spring。

### 4. Hibernate

Hibernate是一个ORM(对象-关系映射)框架，它对JDBC进行了轻量级封装。通过使用Hibernate框架，开发人员能够以面向对象的思维方式来操作数据库。

### 5. iBATIS

相对于Hibernate而言，iBATIS是一个“半自动化”的ORM实现框架，它主要致力于POJO(Plain Ordinary Java Object，简单的Java对象)与SQL之间的映射关系，是对“全自动化”ORM框架的一种有益补充。

## 6. EasyJWeb

EasyJWeb是一个基于模板技术实现的MVC框架,主要致力于Java Web应用程序的快速开发。

## 7. Apache Shiro

Apache Shiro是功能强大并且容易集成的开源权限框架,它能够完成认证、授权、加密、会话管理等功能。Shiro自身提供了对Spring的良好支持。

## 8. SiteMesh

SiteMesh是一个用来在JSP中实现页面布局和装饰的框架组件,利用它可以将网页的内容和页面结构分离,以达到共享页面结构的目的。

SiteMesh基于Servlet的过滤流。它通过截取响应,并进行装饰后再交付给客户。

除了上面介绍的这些框架,Java中还有很多框架,在应用开发中,可根据实际的需求来选择使用。

### 1.2.3 Java Web应用的核心技术

Java Web 应用的核心技术包括以下几个方面:

- **JSP:** 进行输入和输出的基本手段。JSP 以脚本文件的形式存在,主要由HTML代码、客户端脚本(JavaScript等)、JSP的标签和指令、自定义标签库构成。下面是一个典型的JSP示例:

```

<%@ page contentType="text/html;charset=gb2312"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" prefix="c"%> } 指令
学生信息如下:
<table>
  <tr>
    <th>学号</th>
    <th>姓名</th>
    <th>电话</th>
  </tr>
  <c:forEach items="{stulist}" var="stu"> ----- 标准标签库标签
    <tr> ----- HTML代码
      <td>${stu.sid}</td>
      <td>${stu.sname}</td>
      <td>${stu.phone}</td>
    </tr> ----- HTML代码
  </c:forEach> ----- 标准标签库标签
</table> ----- HTML代码

```

- **JavaBean:** 完成功能的处理。JavaBean就是Java中普通的Java类,所以没有特殊的地方。Java Web技术中提供了多个与JavaBean操作相关的标签。
- **Servlet:** 对应用的流程进行控制。Servlet以Java文件的形式存在。所以Servlet也是Java类,是特殊的Java类,在Java Web技术中主要完成控制功能,负责协调JSP页面和完成功能的JavaBean之间的关系。

- **JDBC**: 与数据库进行交互不可缺少的技术。严格来讲, JDBC不属于Java Web技术,但是在Java Web中不可避免地要使用JDBC,所以JDBC也算是Java Web开发中比较重要的技术之一。
- **JSTL和表达式语言EL**: 完成对JSP页面中各种信息的控制和输出。JSTL和表达式语言是在JSP 2.0之后引入的,主要是为了方便用户在JSP页面中使用常用功能。其典型应用是信息的输出,因为JSP界面的主要功能就是展示信息,使用表达式语言使得信息的显示非常简单。例如,上面的 JSP 代码中的`{stu.sid}`,完成的功能是从请求中获取stu对象的sid属性。如果使用Java代码,就没有这么简单了。另外, JSTL中提供了大量常用的功能,例如选择结构和循环结构,在上面的 JSP 例子中就使用`<c:forEach>`标签来完成循环控制。

## 1.2.4 使用Ajax和jQuery提升用户体验

JavaScript的出现使网页和用户之间出现了一种实时性、动态的、交互性的关系,一定程度上减轻了服务器的负载量,为客户提供更流畅的浏览效果。

Ajax(Asynchronous JavaScript and XML)被称为异步的JavaScript与XML,它是一种支持异步请求的技术,可以使用JavaScript向服务器提出请求并处理响应,而不阻塞用户。它最大的优点是在不重新加载整个页面的情况下,可以与服务器交换数据并更新部分网页内容。让用户感觉不到与服务器的交互过程,从而获得更好的用户体验。

Ajax的核心是XMLHttpRequest对象,XMLHttpRequest是由微软开发的可以在不刷新页面的情况下直接进行脚本与服务器间通信的技术。

XMLHttpRequest在发送请求的时候,有两种方式:同步与异步。同步方式是请求发出后,一直到收到服务器返回的数据为止,浏览器进程被阻塞,在页面上什么事也做不了。而异步方式则不会阻塞浏览器进程,在服务端返回数据并触发回调函数之前,用户依然可以在该页面上进行其他操作。Ajax的核心是异步方式,而同步方式只有在极其特殊的情况下才会被用到。

Ajax的基本流程可以概括为:在页面上由JavaScript脚本设置好服务器端的URL、必要的查询参数和回调函数之后,向服务器发出请求,服务器在处理请求之后将处理结果返回给页面,触发事先绑定的回调函数。这样,页面脚本如果想要改变一个区域的内容,只需要通过Ajax向服务器请求与该区域有关的少量数据,在回调函数中将该区域的内容替换即可,不需要刷新整个页面。

jQuery是一个快速、简洁的JavaScript框架,它简化了JavaScript开发。jQuery中的j代表JavaScript,Query是“查询”的意思。也就是说,这个库的意图是基于JavaScript的查询。查询的目标是DOM(文档对象模型)结构中的Node(节点)。网页上的所有内容都是节点,包括文档节点、元素节点、文本节点、注释节点、属性节点等。而jQuery查询主要针对的是元素节点,如段落(`<p>`)、表格(`<table>`)等,同时jQuery还可以用`attr`方法方便地对元素节点的属性进行读取/设置。

此外, jQuery还提供了浏览器兼容、样式读写、事件绑定与执行、动画等特性,后来又加入了Ajax、Promise等,再加上方便的插件编写机制,对整个JavaScript生态圈产生了重

大影响，可以说是JavaScript历史上影响力最大的一个库。

## 1.3 集成开发环境介绍

通常情况下，开发应用程序都要使用IDE(Integrated Development Environment, 集成开发环境), IDE能提高应用程序的开发效率。本节将介绍Java Web应用开发常用的IDE和Web服务器。

### 1.3.1 常用的Java IDE

IDE是一种用于辅助开发人员开发应用程序的应用软件，一般包括代码编辑器、编译器、调试器和图形用户界面。有的还包括版本控制系统、性能分析器等更多辅助工具，因此IDE都具有编写、编译、调试等多种功能。正是基于这些功能，使用IDE开发应用程序才能够大大减轻程序员的工作，缩短项目的开发周期，从而提高应用程序的开发效率。

IDE的种类非常多，有的IDE能同时支持多种应用程序的开发，例如，Eclipse能用于Java、PHP、C++等多种语言开发；有的IDE只针对特定语言的开发，如JBuilder只能用于Java开发，Zend Studio只能用于PHP开发。本节将介绍Java应用开发常用的IDE。

#### 1. IntelliJ IDEA

IntelliJ IDEA是JetBrains公司的产品，是Java语言开发的集成环境。IntelliJ在业界被公认为最好的Java开发工具之一，尤其在智能代码助手、代码自动提示、重构、J2EE支持、各类版本工具(git、svn、GitHub等)、JUnit、CVS整合、代码分析、创新的GUI设计等方面，功能可以说是超强的。

该软件的官方下载地址为<http://www.jetbrains.com/idea/download/index.html>。它的旗舰版本还支持HTML、CSS、PHP、MySQL、Python等。免费版只支持Java等少数语言。IntelliJ IDEA的开发界面如图1-2所示。

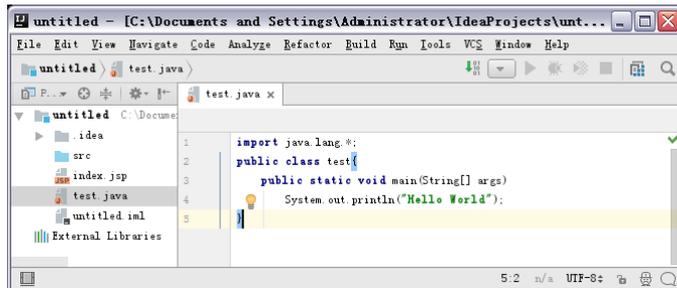


图1-2 IntelliJ IDEA的开发界面

#### 2. Eclipse

Eclipse是一个开放源代码的、基于Java的可扩展开发平台。就其本身而言，它只是一个框架和一组服务，用于通过插件组件构建开发环境。它的最大特点就是其扩展性，几乎能够集成开发人员编写的任何开发源代码的插件。Eclipse最早是由IBM开发的，后来IBM

将Eclipse作为一个开发源代码的项目，献给了开源组织Eclipse.org，但仍由IBM的子公司OTI(主要从事Eclipse开发)继续Eclipse的开发。其官方下载地址为<http://www.eclipse.org/downloads/>，开发界面如图1-3所示。

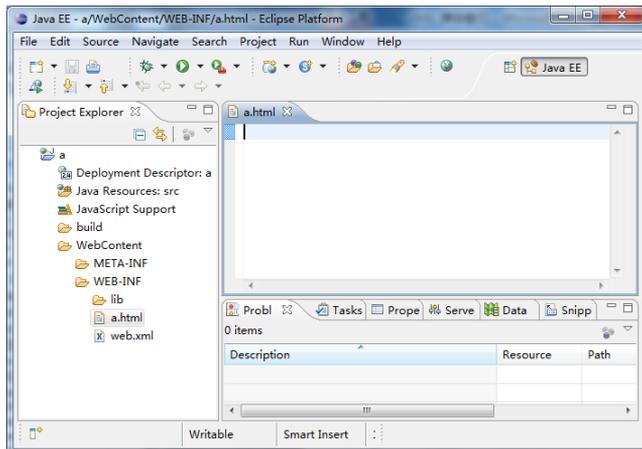


图1-3 Eclipse的开发界面

MyEclipse是一个十分优秀的用于开发Java EE的Eclipse插件集合，MyEclipse的功能非常强大，支持也十分广泛，尤其是对各种开源产品的支持十分不错。MyEclipse可以支持Java Servlet、Ajax、JSP、JSF、Struts、Spring、Hibernate、EJB 3、JDBC数据库链接工具等多项功能。可以说MyEclipse是几乎囊括了目前所有主流开源产品的专属Eclipse开发工具。

### 3. JBuilder

JBuilder最初是由Borland公司开发的产品，目前已归Embarcadero公司所有。它是针对Java的开发工具，官方下载地址为<http://www.embarcadero.com/products/jbuilder>。JBuilder具有专业化的图形调试界面，支持远程调试和多线程调试。它能够简化团队合作，适合企业级Java EE开发，其开发界面如图1-4所示。

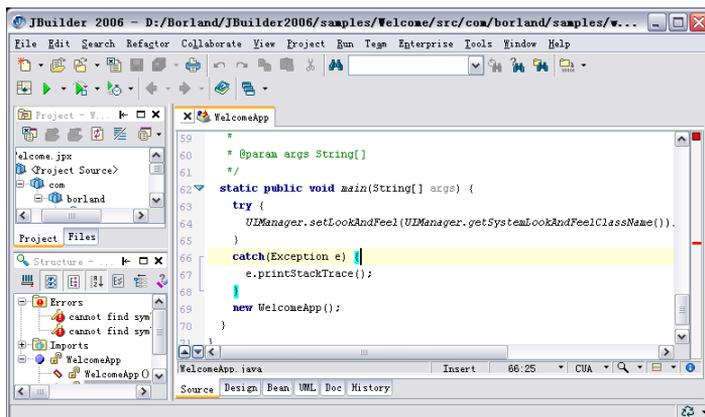


图1-4 JBuilder的开发界面

## 1.3.2 Web服务器汇总

Web服务器是运行及发布Web应用的容器，只有将开发的Web项目放置到该容器中，

才能使网络中的所有用户通过浏览器进行访问。开发Java Web应用所采用的服务器主要是与JSP/Servlet兼容的Web服务器，常用的有如下几种：

- **Apache服务器**：是由Apache基金组织提供的一种Web服务器，其特点是处理静态页面，对静态页面的处理效率非常高。
- **Tomcat服务器**：也是由Apache基金组织提供的一种Web服务器，提供对JSP和Servlet的支持，通过安装的插件，同样可以提供对PHP语言的支持。但是Tomcat只是一个轻量级的Java Web容器，像EJB这样的服务在Tomcat中是不能运行的。它是初学者学习开发JSP应用的首选。
- **Resin服务器**：Resin是Caucho公司的产品，是一个非常流行的支持Servlet和JSP的服务器，速度非常快。Resin本身包含了一个支持HTML的Web服务器，这使它不仅可以显示动态内容，而且显示静态内容的能力也毫不逊色，因此许多网站都使用Resin服务器来构建。
- **JBoss服务器**：JBoss是一个开源的重量级Java Web服务器，是一个遵从Java EE规范、开放源代码的、纯JavaEJB服务器，对J2EE有很好的支持。JBoss采用JML API实现软件模块的集成与管理，其核心服务又是提供EJB服务器，不包含Servlet和JSP的Web容器，不过它可以和Tomcat完美结合。
- **WebSphere服务器**：WebSphere是IBM公司的产品，可进一步细分为WebSphere Performance Pack、Cache Manager和WebSphere Application Server等系列。其中WebSphere Application Server是基于Java的应用环境，可以运行于Sun Solaris、Windows NT等多种操作系统平台上，用于建立、部署和管理Internet和Intranet Web应用程序。
- **WebLogic服务器**：WebLogic是BEA公司的产品，可进一步细分为WebLogic Server、WebLogic Enterprise和WebLogic Portal等系列。其中WebLogic Server的功能特别强大。WebLogic支持企业级的、多层次的和完全分布式的Web应用，并且服务器的配置简单、界面友好。对于那些正在寻求能够提供Java平台所拥有的一切应用服务器的用户来说，WebLogic是十分理想的选择。

## 1.4 本章小结

本章对Java Web应用开发的一些基本知识做了简要介绍。首先介绍了Web和Web应用的概念，以及Web应用的工作原理，接下来讲解了使用Java开发Web应用的优势和核心技术，最后对使用Java开发Web应用的常用集成开发环境进行了对比和简介。通过本章的学习，读者应对Web应用开发有个基本的理解，掌握Web应用的工作原理，熟悉常用的Java Web应用的核心技术，了解常用的Java IDE。

## 1.5 思考和练习

1. 什么是Web应用？
2. 简述Web应用的工作原理。
3. 常用的客户端开发技术有哪些？
4. 简述Java语言的特点。
5. Java Web应用的核心技术有哪些？
6. Web服务器的用途是什么？说出几个常见的Java Web服务器。