



Sams Teach Yourself
JavaServer Pages in 24 Hours

JSP

- 利用JavaServer Pages开发模型
- 从JSP访问数据库
- 用标签库扩展JSP功能
- 给出错误处理页
- 使用JavaServer Pages创建工作级Web应用程序
- 创建动态用户接口
- 配置JSP应用服务器
- 跟踪用户在应用中的移动
- 同时使用多个JSP

速成教程

时光飞逝 时不我待



(美) Jose Annunziato 等著

闻道工作室 译



机械工业出版社
China Machine Press

SAMS

JSP 速成教程

(美) Jose Annunziato 等著

闻道工作室 译



机械工业出版社
China Machine Press

本书首先介绍 JSP 基础知识,如它的结构、应用模型以及如何用于在 Web 中创建动态内容;然后讨论一些高级特性,如在 JSP 中与数据库的连接和使用 JavaBean 访问数据库的方法;最后用实例介绍 JSP 的 Web 应用。

Jose Annunziato, et al: Sams Teach Yourself JavaServer Pages in 24 Hours.

Authorized translation from the English language edition published by Sams, an imprint of Macmillan Computer Publishing U.S.A.

Copyright © 2001 by Sams Publishing. All rights reserved.

Chinese simplified language edition published by China Machine Press.

Copyright © 2002 by China Machine Press.

本书中文简体字版由美国麦克米兰公司授权机械工业出版社独家出版。未经出版者书面许可,不得以任何方式复制或抄袭本书内容。

版权所有,侵权必究。

本书版权登记号:图字:01-2001-0690

图书在版编目(CIP)数据

JSP 速成教程/(美)昂纳日托(Annunziato, J.)等著;闻道工作室译. - 北京:机械工业出版社, 2002.1

书名原文: Sams Teach Yourself JavaServer Pages in 24 Hours

ISBN 7-111-09457-3

I. J... II. ①昂... ②闻... III. 主页制作-技术, JSP-教材 IV. TP393.092

中国版本图书馆 CIP 数据核字(2001)第 071604 号

机械工业出版社(北京市西城区百万庄大街 22 号 邮政编码 100037)

责任编辑:李云静

北京第二外国语学院印刷厂印刷·新华书店北京发行所发行

2002 年 1 月第 1 版第 1 次印刷

787mm×1092mm 1/16·21 印张

印数:0 001-4 000 册

定价:48.00 元(附光盘)

凡购本书,如有倒页、脱页、缺页,由本社发行部调换

向 JSP 中插入 Java 的基本标签

scriptlet 标签

用途:用于向 JSP 中嵌入 Java 代码块。

语法:

```
< % //embed Java Code % >
```

声明标签

用途:用于声明变量和方法。

语法:

```
< %! variable declaration;  
      method declaration; % >
```

表达式标签

用途:用于向 JSP 中插入 Java 表达式。

语法:

```
< % = Java Expression % >
```

JSP 指令

指令的通用语法:

```
< % directive_type directive_attributes % >  
      directive_type::= page | include | taglib  
      directive_attributes::= dependent on the directive_type
```

include 指令

用途:用于包含另一个文件的文本。

语法:

```
< % @ include file = "Relative_URL_to_File" % >
```

taglib 指令

用途:用于通过导入定制行为标签扩展 JSP 的功能。

语法:

```
< % @ taglib uri = "URI_to_Tag_Library" prefix = "Custom_actions" % >
```

page 指令

用途:用于设置 JSP 的属性,如它导入什么软件包,它是否为会话的一部分,以及是否存在一个错误页面。

语法:

```
< % @ page page_attributes % >  
      page_attributes::= { language = "scriptingLanguage" }  
                        { extends = "className" }  
                        { import = "importList" }  
                        { session = "true | false" }  
                        { buffer = "none | sizekb" }
```

```

{ autoFlush = "true|false" |
  isThreadSafe = "true|false" |
  info = "info_Text" |
  errorPage = "errorPageURL" |
  isErrorPage = "true|false" |
  contentType = "contentType" |

```

预定义行为标签

行为标签的通用语法：

```

<jsp:action_Name action_attributes /> or
<jsp:action_Name action_attributes > </jsp:action_Name >
  action_Name:: = useBean | getProperty | setProperty
                  include | forward | plugin

```

useBean 行为

用途：用于访问位于服务器的 JavaBean 工具类。

语法：

```

<jsp:useBean id = "name" scope = "page|request|session|application"
              class = "type_Spec" />

```

或：

```

<jsp:useBean id = "name" scope = "page|request|session|application"
              class = "type_Spec" >

```

```

    //body

```

```

</jsp:useBean >

```

```

  type_Spec:: = class = "className" |
                class = "className" type = "typeName" |
                type = "typeName" class = "className" |
                beanName = "beanName" type = "typeName" |
                type = "typeName" beanName = "beanName" |
                type = "typeName"

```

NJS304/07

setProperty 行为

用途：用于设置 JavaBean 类的属性。

语法：

```

<jsp:setProperty name = "beanName" property_expression />
  property_expression:: = property = " * " |
                        property = "propertyName" |
                        property = "propName" param = "paramName" |
                        property = "propName" value = "valueName"

```

getProperty 行为

用途：用于取得 JavaBean 类的一个属性值。

语法:

```
<jsp:getProperty name = "beanName" property = "propName" />
```

Include 行为

用途:用于包含另一个 JSP 的输出。

语法:

```
<jsp:include page = "Relative_URL" />
```

forward 行为

用途:用于向另一个 JSP 转发请求。

语法:

```
<jsp:forward page = "Relative_URL" />
```

plugin 行为

用途:用于插入一个 Java applet 或 JavaBean 组件(用于客户端的外观)。

语法:

```
<jsp:plugin type = "bean|applet" code = "objectCode"
```

```
    codebase = "objectCodebase"
```

```
    { align = "alignment"          }
```

```
    { archive = "archiveList"      }
```

```
    { height = "height"            }
```

```
    { hspace = "hspace"            }
```

```
    { jreversion = "jreversion"    }
```

```
    { name = "componentName"       }
```

```
    { vspace = "vspace"            }
```

```
    { width = "width"              }
```

```
    { nspluginurl = "url"          }
```

```
    { iepluginurl = "url"          } >
```

```
< params >
```

```
    < param name = "name" value = "paramValue" />
```

```
< /params >
```

```
< fallback > text < /fallback >
```

```
< /jsp:plugin >
```

译 者 序

当今社会,科学技术的发展真可谓一日千里。“活到老、学到老”的终身教育理念深入人心。尤其是在 IT 领域,技术的更新换代更为频繁;因此,对于该领域的从业者,或者有志于投身该行业的年轻人,无论其具备何种学业背景,为了更好地胜任自己的本职工作,进行以自学为主的各种形式的“充电”都势在必行。所谓“工欲善其事,必先利其器”,对于这些自学大军而言,拥有一套合适的教材尤为重要。机械工业出版社推出的这套适合自学的速成教程,为大家提供了一种不错的选择。

JSP 是一种基于内置 Java 代码的、用于创建动态内容的脚本语言,它天然地具备 Java 技术的跨平台特性,从而在广泛的基于网络的应用中迅速得到应用,成为当前软件领域最热门的技术之一。JSP 技术涉及多种软件技术和标准,诸如 Java、HTML、数据库等,并且广泛应用于电子商务等动态网站建设中。

本书首先介绍 JSP 基础,如它的结构、应用模型以及如何用于在 Web 中创建动态内容。然后讨论一些高级特性,如在 JSP 中与数据库的连接和使用 JavaBean 访问数据库。最后,将要编写一个模拟公司内部网的人力资源网站建设的 Web 应用。

本书选材精当,重点突出,从而确保在一种 24 学时的自学教程中,得以合理地组织这种涉及众多相关知识并且应用广泛的技术。

本书的另一特色是,实例丰富,剖析详细。在众多计算机论著中,提供大量实例代码者并不鲜见;但是,对代码进行几乎是逐行分析解释者则并不多见。

本书中还以新术语的形式,以简练的语言对涉及的新概念进行明确定义,这有利于读者对新知识有一个清晰的认识。

本书由闻道工作室组织翻译,但由于时间紧迫,加之水平有限,错误或不当之处在所难免,敬请广大读者批评指正。

前 言

本书将向你介绍 JSP 基础,如它的结构、应用模型以及如何用于在 Web 中创建动态内容。读者也将学习一些高级特性,如在 JSP 中与数据库的连接和使用 JavaBean 访问数据库。最后,将要编写一个 Web 应用,该应用模拟一个公司内部网的人力资源网站。听起来内容非常丰富!的确如此,而且读者将在 24 学时课程之内学完这些内容。

读者对象

本书针对的是熟悉 HTML 和 Java 的读者。不要求读者是一位 HTML 专家;但是,应该熟悉常见的标签(或称为标记)并且可以创建简单的 HTML 文档。同样,不要求读者是一位专家级 Java 程序员,尽管关于任何编程语言方面的经验都将有助于你学好本书。

本书内容

读者将学习什么是 JSP,以及 JSP 如何适合于 Sun 公司的 Java 2 企业版平台(Java 2 Enterprise Edition Platform)。读者还将了解 JSP 与其他类似技术的比较,如 CGI 程序和活动服务器页面(ASP)。

首先,将向读者介绍 JSP 基础,诸如允许在 HTML 页面中嵌入 Java 的标签和如何在几个 JSP 之间进行分工。本书也将探讨诸如在 JSP 中与数据库的连接、JavaBean 或服务器端类的用法以及使用定制标签库扩展 JSP 的功能。最后,将以一个项目开发结束本书,即编写一个公司内部网的人力资源网站。

读者所需要的工具

在开始你的“征程”之前,需要准备几项工具。读者需要一个文本编辑器,如 Windows 操作系统中的 Notepad(记事本)。还需要一个 Sun 公司的 Java 开发工具包,即 JDK(Java Development Kit)。此 JDK 可以从 Sun 公司的网站 <http://java.sun.com/products/jdk/1.2/index.html> 下载,可以在 Production Releases 标题下选择下载的平台。最佳的 JDK 版本为 1.2.2,它非常容易下载和安装。你下载的将是一个 zip 文件,通常称为 jdk1_2_2-win.zip,它包含所需要的文件。你所需要做的不过是将它解压至根目录,如 c:\。

本书使用的惯例

本书使用不同的字体来区分代码和正常的文本,这有助于读者识别重要的概念。

请不要输入出现在代码清单每行开始处的行号!行号只是为了在解释代码时引用该行。



注意 给出与周围的讨论相关的值得注意的信息。



技巧 提供建议或者传授一种更方便的做事方法。



警告 提供关于潜在问题的忠告并且帮助读者避开灾难。

新术语 新术语提供一种新的或者关键术语的清晰定义。

本书的其他信息

原书名: Sams Teach Yourself JavaServer Pages in 24 Hours

原书书号: ISBN 0-672-32023-1

原出版社邮箱地址: java@mcp.com

原出版社网址: www.sampublishing.com

目 录

译者序
前言

第一部分 概 述

第 1 学时 Web 应用简介	1	2.7 课外作业	24
1.1 什么是 JSP	1	第 3 学时 Servlet 入门	25
1.2 Web 的演变	3	3.1 什么是 Servlet	25
1.2.1 静态 Web	3	3.2 JSP 如何转换为 Servlet	26
1.2.2 插入 Web	4	3.2.1 JSP/Servlet 的生命期	26
1.2.3 动态 Web	6	3.2.2 JSP 的转换过程	26
1.2.4 N 层 Web	9	3.3 HttpServlet 的基本结构	27
1.3 什么是 Java 2 企业版平台	10	3.3.1 init (...) 方法	27
1.3.1 谁开发了此规范	10	3.3.2 service (...) 方法	28
1.3.2 此规范定义了什么	10	3.3.3 doPost (...) 和 doGet (...) 方法	28
1.3.3 谁受益于此规范	11	3.4 把一个简单 JSP 转换为 Servlet	30
1.3.4 什么技术构成了此平台	11	3.5 课时小结	33
1.4 什么是应用服务器	12	3.6 专家答疑	33
1.4.1 应用服务器的需求	12	3.7 课外作业	33
1.4.2 与 J2EE 兼容意味着什么	12	第 4 学时 配置应用服务器	34
1.4.3 应用服务器供货商列表	13	4.1 如何配置 BEA WebLogic Server	34
1.5 课时小结	13	4.1.1 安装 BEA WebLogic Server	34
1.6 专家答疑	14	4.1.2 WebLogic Server 的目录结构	35
1.7 课外作业	14	4.1.3 weblogic.properties 文件	36
第 2 学时 创建一个 JSP	15	4.1.4 注册 JSPServlet	38
2.1 HTML 回顾	15	4.1.5 设置 JSPServlet 的初始参数	38
2.2 JSP 与 CGI 程序比较	16	4.2 在 WLS 中部署 JSP	38
2.3 JSP 与 ASP 比较	16	4.3 课时小结	40
2.4 创建 JSP 的步骤	19	4.4 专家答疑	40
2.4.1 配置 Apache 的 Tomcat	19	4.5 课外作业	40
2.4.2 在文本编辑器中输入 JSP	22		
2.4.3 以 .jsp 为扩展名进行保存	23	第二部分 创建简单的 JSP	
2.4.4 在浏览器中观察页面	23	第 5 学时 理解 JSP 应用模型	41
2.5 课时小结	24	5.1 什么是应用模型	41
2.6 专家答疑	24	5.2 简单模型	41
		5.3 N 层模型	43
		5.4 松散耦合模型	44
		5.5 包含请求模型	45
		5.6 转发请求模型	47
		5.7 课时小结	50

5.8 专家答疑	51	7.6 课外作业	75
5.9 课外作业	51	第 8 学时 在 JSP 中插入 Java 表达式	77
第 6 学时 理解 JSP 的布局	52	8.1 JSP 表达式标签	77
6.1 标签和属性值的用法	52	8.2 在表达式中引用一个变量	78
6.1.1 理解单一的和成对的标签	52	8.3 从表达式中调用方法	80
6.1.2 用属性配置标签	52	8.3.1 调用一个局部方法	81
6.2 向代码中添加注释	53	8.3.2 调用一个对象的方法	82
6.2.1 添加 HTML 注释	53	8.4 课时小结	84
6.2.2 添加 JSP 注释	53	8.5 专家答疑	84
6.2.3 添加 Java 注释	54	8.6 课外作业	84
6.3 JSP 剖析	54	第 9 学时 用 Java scriptlet 控制 JSP	85
6.3.1 编写 HTML 代码	54	9.1 scriptlet	85
6.3.2 理解 JSP 标签元素	55	9.2 选择控制结构	85
6.4 JSP 内置对象的用法	58	9.2.1 使用 if 选择语句	85
6.4.1 理解 request 对象	58	9.2.2 使用 switch 语句	88
6.4.2 理解 response 对象	59	9.2.3 使用 break 语句	90
6.4.3 理解 session 对象	59	9.3 循环控制结构	91
6.4.4 理解 config 对象	59	9.3.1 使用 for 循环	91
6.4.5 其他 JSP 内置对象	59	9.3.2 使用 while 循环	93
6.5 理解 JSP 和 JavaBean	59	9.3.3 使用 continue 语句	95
6.5.1 JavaBean 概述	60	9.3.4 使用 do/while 循环	96
6.5.2 JavaBean 规范	60	9.4 课时小结	98
6.5.3 实例化 JavaBean	60	9.5 课外作业	98
6.5.4 set 和 get 属性标签的用法	61	第 10 学时 处理用户的请求	100
6.5.5 一个使用 JavaBean 的简单例子	61	10.1 理解用户请求	100
6.6 课时小结	61	10.2 创建用户请求	101
6.7 专家答疑	61	10.2.1 剖析 URL	101
6.8 课外作业	61	10.2.2 从一个超级链接中产生请求	102
第 7 学时 JSP 中的变量声明与方法		10.3 提取路径和 URL 信息	102
声明	63	10.4 提取主机和端口信息	103
7.1 声明标签的用法	63	10.5 从查询串中解析参数	105
7.2 JSP 中的变量声明	63	10.6 使用 HTML 表单	106
7.2.1 理解数据类型	64	10.7 一个处理信用卡信息的 HTML 表单	
7.2.2 初始化变量	65	和 JSP	108
7.2.3 声明数组与集合	66	10.8 课时小结	111
7.3 JSP 中的方法声明	70	10.9 专家答疑	111
7.3.1 方法调用	71	10.10 课外作业	111
7.3.2 理解方法过载	73	第 11 学时 为用户生成动态响应	113
7.4 课时小结	74	11.1 生成动态内容	113
7.5 专家答疑	74	11.1.1 动态标题	113

11.1.2 动态表格	114	14.3.2 exception 隐含对象	156
11.1.3 动态表单	117	14.4 课时小结	157
11.2 可配置主页	119	14.5 专家答疑	158
11.3 课时小结	123	14.6 课外作业	158
11.4 专家答疑	123	第 15 学时 在 JSP 中插入 applet	159
11.5 课外作业	123	15.1 JSP 行为标签的一般语法	159
第三部分 向 JSP 中增加功能		15.2 用 plugin 行为插入 applet	160
第 12 学时 跟踪用户会话	125	15.3 用 param 标签定义 applet 的初始 参数	164
12.1 HTTP 是一种无状态协议	125	15.4 综合应用 plugin 行为	168
12.2 保存和取出 cookie	126	15.5 课时小结	173
12.3 用户会话	129	15.6 专家答疑	173
12.4 编码 URL	133	15.7 课外作业	173
12.5 课时小结	136	第 16 学时 在 JSP 中应用 JavaBean	174
12.6 专家答疑	136	16.1 理解 JavaBean	174
12.7 课外作业	136	16.2 理解 JavaBean 作用域	178
第 13 学时 配置与初始化 JSP	137	16.3 用 JavaBean 实现购物车	179
13.1 理解 JSP 的注册和初始化	137	16.4 设置和获取 JavaBean 属性	183
13.2 打印全部初始化参数的简单 JSP	137	16.5 使用带 JavaBean 的表单	185
13.3 编译 JSP 至 Servlet	138	16.6 课时小结	187
13.4 用应用服务器注册与初始化 JSP	140	16.7 专家答疑	187
13.5 config 对象的用法	140	16.8 课外作业	187
13.6 实例: 配置登录页面	141	第 17 学时 从 JSP 中访问数据库	189
13.7 ServletContext 对象	142	17.1 创建数据库	189
13.8 课时小结	143	17.1.1 使用 Cloudscape 的 Cloudview	190
13.9 专家答疑	143	17.1.2 创建表	191
13.10 课外作业	143	17.1.3 创建记录	192
第 14 学时 建立错误页面	145	17.2 Java 数据库连接	192
14.1 概述	145	17.2.1 JDBC 驱动器	193
14.1.1 Exception 和 Throwable 类	145	17.2.2 取得数据库连接	195
14.1.2 理解栈跟踪	147	17.2.3 用 Statement 对象和数据库 交互	195
14.1.3 捕获异常	147	17.2.4 用 ResultSet 对象定位查询 结果	196
14.2 处理异常	148	17.3 JSP 和数据库交互	196
14.2.1 创建定制异常	148	17.3.1 用 JSP 打印数据库的内容	197
14.2.2 在 JSP 中生成异常	149	17.3.2 用 JSP 更新新数据库	198
14.2.3 在 JSP 中捕获异常	150	17.4 课时小结	200
14.2.4 在 Java 类中产生异常并在 JSP 中 捕获	152	17.5 专家答疑	200
14.3 实现错误页面	154	17.6 课外作业	201
14.3.1 决定错误页面	154		

第 18 学时 在 JSP 中替换文本	203	21.4 专家答疑	251
18.1 理解 include 指令	203	21.5 课外作业	252
18.2 include 指令的例子	204	第 22 学时 创建表示层	253
18.2.1 包含一个文本文档	204	22.1 建立应用程序	253
18.2.2 包含另一个 JSP	206	22.1.1 制作 CompanyBanner.html	253
18.2.3 被包含的文件不存在	208	22.1.2 制作 SiteNavigator.html	254
18.3 课时小结	211	22.1.3 制作主页	256
18.4 专家答疑	211	22.1.4 编写 PresentCompanyHealthCareIn- fo.jsp	258
18.5 课外作业	211	22.1.5 编写 JSP 介绍公司策略	261
第 19 学时 用标签库扩展 JSP 功能	213	22.1.6 允许员工浏览个人津贴费	264
19.1 理解标签库	213	22.2 课时小结	271
19.2 使用标签库的例子	215	22.3 专家答疑	271
19.2.1 用一个定制行为跟踪页面 点击	215	22.4 课外作业	271
19.2.2 显示 session 对象中的元素	218	第 23 学时 设计事务逻辑	272
19.2.3 使用带属性的定制行为标签	221	23.1 概述	272
19.3 课时小结	225	23.2 通过 Husky World 人力资源 Web 站点 增加新员工	272
19.4 专家答疑	225	23.3 编写修改员工信息的功能	282
19.5 课外作业	225	23.4 课时小结	289
第 20 学时 用多个 JSP 处理请求	226	23.5 专家答疑	289
20.1 回顾包含请求应用模型	226	23.6 课外作业	289
20.1.1 回顾 JSP 行为	226	第 24 学时 打包和部署企业级应用 程序	291
20.1.2 理解 include 行为与 include 指令 的区别	227	24.1 什么是 Web 档案文件	291
20.2 实现 include 行为	227	24.2 建立 Web 档案文件的目录结构	292
20.3 回顾转发请求应用模型	234	24.3 创建部署描述符	308
20.4 实现 forward 行为	235	24.4 生成 .war 文件	311
20.5 课时小结	245	24.5 向应用程序发送请求	312
20.6 专家答疑	245	24.6 课时小结	312
20.7 课外作业	245	24.7 专家答疑	313
第 21 学时 设计企业应用	246	24.8 课外作业	313
21.1 收集设计需求	246	附录 A 思考题答案	314
21.2 设计应用程序	250		
21.3 课时小结	251		

第一部分 概 述

第 1 学时 Web 应用简介

这里以一个年轻小姐的故事作为开始。Marian 对将赴佛罗里达度假充满期待,她是一位网络通,因此她访问一家航空公司的 Web 站点,并输入她的目的地——佛罗里达,得到了一张飞往佛罗里达的航班列表,并以一种便于阅读的方式显示于她的浏览器中。她的工作伙伴,John 也在筹划自己即将到来的假期,但是他想去夏威夷。因为他对 Internet 也很在行,他进入同一家航空公司的站点,并输入他的目的地,得到了一张飞往夏威夷的航班列表。他们从相同的 Web 站点接收到如此不同的信息,是什么使之成为可能呢?该航空公司的 Web 站点拥有一种称为 Web 应用的应用。此应用接受他们的请求,并生成一种适合于每个特定请求的响应。由于 Web 应用需要立即创建这些文档,因此需要动态地返回内容。这意味着,来自于这些网站的响应,诸如飞往全国各地的航班,在输入特定请求时需动态创建。

Web 应用 Web 应用是一种运行于 Web 服务器上的程序。

Marian 和 John 都找到了他们度假所需搭程的航班。关于 Marain 和 John 的故事也结束了,但是本书的历程才刚刚开始。读者将深入研究 Web 结构并且理解 Web 应用如何生成各种请求的不同响应。本书主要探讨 Sun 公司的 JavaServer Page (JSP)技术,用于向 Web 客户端发送动态内容。因此,让我们开始此历程吧。

本学时将学习:

- 什么是 JavaServer Page。
- Web 的演变。
- 创建 Java 2 企业版(J2EE)平台的理由。
- 什么是应用服务器。

1.1 什么是 JSP

JSP 是由 Sun 公司为创建动态 Web 内容而定义的一种技术。它们是与 Java 相关的一种 HTML 文档,Java 提供动态内容。JSP 是服务器端应用;它们接受请求并生成响应。此请求通常产生于一个 Web 客户端,而响应则生成一种 HTML 文档,需要返回给 Web 客户端。由于 JSP 是服务器端应用,它们拥有对服务器端资源的访问,诸如 Servlet、JavaBean、EJB 和数据库。在第 5 学时,读者将学习 JSP 可以实现的各种设计模式,如同 Sun 的规范所定义的一样。

Web 客户端 Web 客户端是一种使用 HTTP 协议与 Web 服务器交互的客户端。此

客户端运行一个 Web 浏览器,诸如 Internet Explorer 或 Netscape Navigator,并且通过在浏览器中指定一个 URL 来请求文档。

使用 JSP 有多种好处。因为 JSP 使用 Java 编程语言,并遵循一次写就,处处运行的策略。这意味着,一个 JSP 可以运行于支持 JSP 的任何应用服务器,而不需要对代码做任何修改(应用服务器将在 1.4 节讨论)。JSP 的另一优点在于,此规范是按照鼓励与集成开发环境(IDE)相集成的方式编写的。



JSP 可以用任何编辑器编写;只需要确保以 .jsp 的扩展名保存此文档。这意味着,读者可以用 Windows 操作系统的 Notepad 编写 JSP,或者用 UNIX 操作系统的 emacs 编写 JSP。Dreamweaver 是一种运行编写 JSP 的 IDE,它是 WebGain Studio 的组成部分。

JSP 的另一个优势是标签(tag,或称为标记)库的使用。JSP 使用类似如 HTML 或 XML 的标签,来插入动态内容。可以通过导入标签库向 JSP 添加额外的功能。标签库定义可用于替换代码节的额外标签。

JSP 的另一个主要好处是角色的分离。JSP 规范允许将工作量分为两类:页面的图形内容和页面的动态内容。不具备 Java 编程语言知识的创新性团队可以创建页面的图形内容,然后 Java 程序员向此 HTML 文档插入 Java 代码,实现动态内容。



在编写 JSP 时,先编写 HTML 代码,然后插入 Java 代码创建动态内容。因此,如果读者不计划编写 HTML,就可让你的图形设计师发挥他们的想象力创建一个漂亮的页面,然后插入自己的 Java 代码,实现 JSP 的动态化。

清单 1-1 是一个 JSP 示例,显示 Java 代码如何插入 JSP。此 JSP 所做的一切是向一个 HTML 页面中插入当前的时间和日期(见图 1-1)。如果读者不能马上理解此部分,请不要担心——本书将详尽地探讨 JSP 的生成。在第 2 学时,读者将创建并部署自己的第一个 JSP。

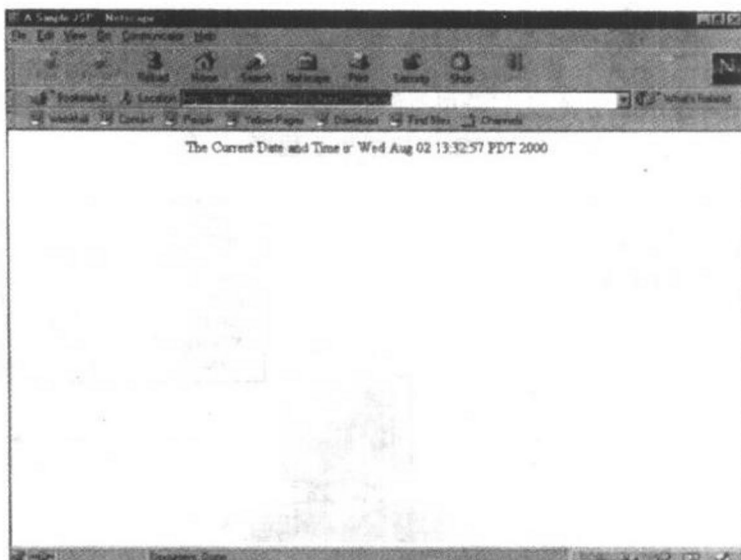
清单 1-1 一个向 HTML 文档插入当前时间与日期的示例 JSP

```

1: <HTML>
2: <HEAD><TITLE>A Simple JSP </TITLE> </HEAD>
3: < BODY>
4:   < FONT COLOR = "blue"FACE = "Trebuchet">
5:     < CENTER>
6:       The Current Date and Time is: < % = new java.util.Date()% >
7:     </CENTER>
8:   </FONT>
9: </BODY>
10: </HTML>

```

图 1-1 在一个浏览器中观察此示例 JSP



1.2 Web 的演变

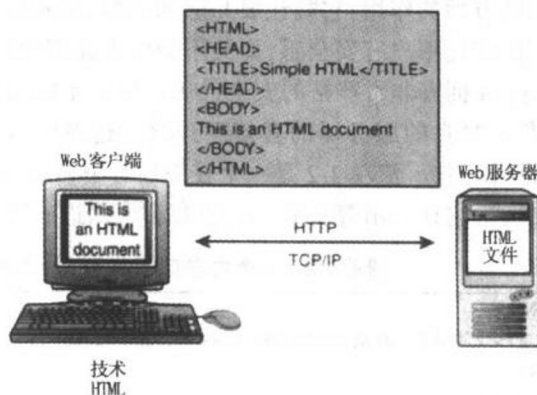
在过去的几年中,Web 技术已经经历了重大演变。在不久以前,Web 还只由静态文档构成。现在,程序已经被嵌入 HTML,而且可以在运行时向 HTML 文档添加动态内容。

1.2.1 静态 Web

静态 Web 是最简单的 Web 结构。图 1-2 显示了一张静态 Web 框图。

Web 客户端使用 TCP/IP 接连至 Web 服务器,并且使用 HTTP 产生请求。服务器向 Web 客户端发送一个已生成的 HTML 文档。此文档包含文本,可能还有超级链接和格式化标签。它不包含任何动态内容,也不包含一种用户与之交互的方式。在这种结构中,HTML 文档是缺少变化的,而且不向客户端提供交互。

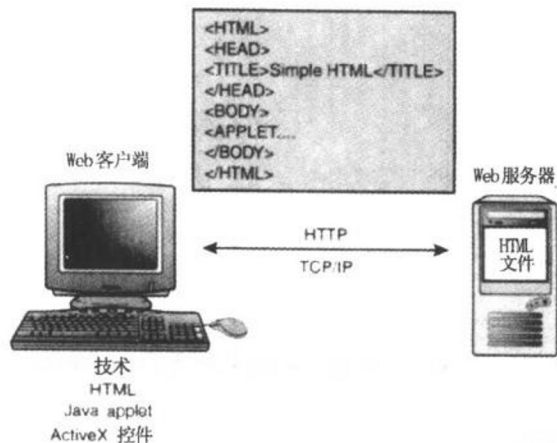
图 1-2 静态 Web 结构不包括动态内容和用户交互



1.2.2 插入 Web

随着技术的发展,人们希望他们的 Web 页面具备更多的功能。插入 Web(Plug-In Web),如图 1-3 所示,允许用户向 HTML 文档添加小程序。

图 1-3 插入 Web 结构提供被包含的小程序



此小程序的添加意味着需要新的 HTML 标签,而且浏览器将必须理解如何处理此新标签。可以运行于浏览器的两类程序是 Java applet 和微软的 ActiveX 控件。

Java applet 是插入于 HTML 文档的、用 Java 编程语言编写的小程序,以便增加更多的功能。由于 applet 是用 Java 编程语言编写的,因此极具移植性。Java 程序被编译为运行于 Java 虚拟机的字节码。applet 可以运行于内置有虚拟机的浏览器。

1.1 Java 虚拟机 Java 虚拟机接受编译过的字节码并转换为具体机器语言,以便执行。对于现存的每种类型的计算机,几乎都存在相应的 Java 虚拟机。Internet Explorer 和 Netscape Navigator 都配置有 Java 虚拟机。

applet 运行于一种称为 sandbox 的环境。读者在考虑 sandbox 时,可以将它看做婴儿的一种“安全乐园”,有两方面的作用:它既要婴儿感到安全,又隔离了沙子。用于 applet 的 sandbox 与此非常类似;它应该让用户有安全感,而又不会危害他们的机器,同时它又提供一种运行 applet 的环境。Java applet 拥有非常严格的安全代码。它们不能接触你的本地文件或硬驱,而且只能创建一个与它们所来自的服务器的套接字连接。这意味着,applet 不会变成恶意的和开始创建全球范围的网络连接。清单 1-2 是一个在 HTML 中调用的 Java applet 示例,它允许用户输入一个十进制数;然后它计算出等价的二进制数和十六进制数。

清单 1-2 一个内嵌有 Java applet 的 HTML 文档

```
1: < HTML >
2: < HEAD > < TITLE > Java Applet Example </TITLE> </HEAD >
3: < BODY >
4: < CENTER >
```