

时尚百例丛书



JSP

时尚编程百例

 机械工业出版社
China Machine Press

●网冠科技 编著

100

时尚百例丛书

JSP

时尚编程百例

网冠科技 编著

光盘包含本书素材、效果文件



机械工业出版社

Java Server Page(简称为 JSP)是一种用于开发动态网页的编程工具,它为创建动态内容的 Web 页面提供了一个简捷而且便利的方法。JSP 技术是由 SUN 公司倡导、众多业界公司参与制定的一种标准技术。

本书精心制作了 100 个实例,全面讲解 JSP 语言的开发方法和技巧。全书共分六篇:基础页面和例外处理篇;数据输入篇;Cookie 与 session 篇;文件操作篇;数据库操作篇;使用 Java Bean 篇。

本书适用于不同层次的编程人员阅读和参考。

图书在版编目(CIP)数据

JSP 时尚编程百例 / 网冠科技编著.

—北京:机械工业出版社,2001.9

(时尚百例丛书)

ISBN 7-111-09317-8

I. J … II. 网… III. 主页制作—软件工具, JSP IV. TP311.56

中国版本图书馆 CIP 数据核字(2001)第 061098 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

策 划:胡毓坚

责任编辑:王 虹

责任印制:付方敏

北京市密云县印刷厂印刷·新华书店北京发行所发行

2001 年 9 月第 1 版·第 1 次印刷

787mm × 1092mm $\frac{1}{16}$ · 20.5 印张 · 504 千字

0001-6000 册

定价:36.00 元(1CD)

凡购本图书,如有缺页、倒页、脱页,由本社发行部调换

本社购书热线电话:(010) 68993821、68326677-2527

时尚百例丛书

追求时尚 追求完美

出版说明

随着 21 世纪的到来,人们更深切地感受到了计算机在生活和工作中的作用越来越重要,越来越多的职业需要具有计算机的应用技能。掌握计算机是职业的需要,更是事业发展的需要。

目前计算机技术不但广泛地应用在办公自动化中,它还全面渗透到各行各业。如果要从事平面设计的相关行业,就应该学会平面设计软件,如 Photoshop、CorelDRAW、FreeHand 等;如果要从事三维设计的相关行业,就应该学会三维设计软件,如 3DS MAX、Maya、Poser 等;如果要从事多媒体设计的相关行业,就应该学会多媒体制作软件,如 Authorware、Director、Premiere 等;如果要从事与网络相关的行业,就应该学会 Flash、Dreamweaver、Fireworks、ASP、PHP、JavaScript 等;如果要从事建筑产品、工业产品设计的相关行业,就应该学会 AutoCAD、3DS VIZ、Protel 等;如果要从事软件开发的相关行业,就应该学会 VB、VC、VFP、Delphi、PowerBuilder 等编程。

所有与计算机相关的职业都要求工作者有很强的计算机操作技能,做到运用自如,熟练而且深入地掌握软件的应用。而要做到这一点,必须从软件的各个方面入手,通过实例演练的方式训练自己,而且要反复练习,做到举一反三。

为了让大家能深入而且熟练地掌握相关软件的应用方法,机械工业出版社特别为广大读者推出了这套时尚百例丛书。本丛书对每一个应用软件精心制作了 100 个实例,其宗旨就是让读者全方位掌握软件的应用,为广大读者提供一条快速掌握计算机应用技能的捷径。

本丛书采用新颖的版式,将知识和实例紧密结合,通过对各种实例的详细讲解,使读者不必事先学习各种软件,而从实例的制作过程中体会到每个软件每项功能的使用方法,并自己做出各种实例效果,这样既节省了大量时间,同时也使读者有身临其境的感觉,并可以反复演练,将所学知识运用到职业工作中去。

书山有路勤为径。愿广大读者能通过本丛书的学习掌握计算机技能,并应用到自己的工作和事业中去。

机械工业出版社



前 言

《JSP 时尚编程百例》是“时尚百例丛书”中的一本。

Java Server Page (简称为 JSP) 技术是由 SUN 公司倡导、众多业界公司参与制定的一种标准技术。它为创建动态内容的 Web 页面提供了一个简捷而且便利的方法。毫不夸张地说, JSP 从诞生之日起, 就注定了将会成为最流行的 Web 开发技术。JSP 技术的目的主要是设计一个开放的、可扩展的、建立动态 Web 页面的标准。

JSP 集成了 Java 面向对象的程序语言能力与跨平台的优势, 并且与超文本标记语言紧密结合, 与传统的 CGI 程序相比, JSP 不仅编写、执行容易, 也大幅提高了系统的执行性能。

本书的例子都是在以下环境中编写和运行的 (如果要正确运行, 最好能在相同的环境下进行):

操作系统: Windows 2000 Professional

系统设置: 1024×800

Java 编译器: J2sdk1.3

JSP 编译器: Jrun

浏览器: Microsoft 的 IE5.0

数据库: Microsoft 的 Access2000

本书以实例的形式向读者讲解如何使用 JSP。全书制作了 100 个实例, 满足不同层次编程人员的需要。对于初学者, 通过本书实例的学习, 可以迅速了解 JSP 的开发思想和方法, 并可以快速创建自己的编程应用程序, 从而可以熟悉掌握 JSP 的使用方法。对于已经有一定基础的中级水平的编程开发人员, 可以通过本书的实例, 学习到一些具有实际意义的开发方法。对于高级开发人员, 本书讲解了许多处理问题的方法和技巧, 进一步提高开发应用程序的能力。

相信通过本书 100 个实例的学习, 可以提高您开发 JSP 应用程序的效率。真诚希望本书对读者有所帮助。



网冠科技

本书光盘含配套素材, 技术支持请点击网冠科技站点 <http://netking.163.com>。E-mail:

netking_@yeah.net。  **NETEASE** 是网易公司的标志。
WWW.163.COM

目 录

出版说明 前 言

第一篇 基础页面和例外处理

实例 1	依此变小的文字	2
实例 2	获得当前时间	4
实例 3	简单计数器	6
实例 4	设置页面属性	8
实例 5	加载文件	10
实例 6	动态加载文件	12
实例 7	利用<Applet>载入 Applet	14
实例 8	利用<jsp:plugin>载入 Applet	16
实例 9	获得客户端浏览器信息	18
实例 10	错误检测	20
实例 11	自行投掷异常	22
实例 12	从函数中投掷异常	24
实例 13	建立错误处理网页	26
实例 14	发送邮件错误的处理	29

第二篇 数据输入

实例 15	文本框数据的 post 传递	33
实例 16	文本框数据的 get 传递	35
实例 17	将字符串转换为数值	37
实例 18	密码输入	40
实例 19	汉字处理	43
实例 20	下拉列表框	46
实例 21	单选型列表框	49
实例 22	多选型列表框	52
实例 23	选择钮	56
实例 24	复选框	59



实例 25	群组检查框	62
实例 26	网上测试	65
实例 27	文本框数据检查	69
实例 28	选择钮数据检查	72
实例 29	光标位置控制	75
实例 30	逐渐显示文字	78
实例 31	网页引导	80
实例 32	选择型计算器	82
实例 33	指定用户计数器	86

第三篇 Cookie 与 session

实例 34	存取 Cookie 数据	90
实例 35	保存日期到 Cookie	93
实例 36	Cookie 有效时间设置	95
实例 37	Cookie 存取路径设置	100
实例 38	应用 Cookie 的网上测试	104
实例 39	猜数字	110
实例 40	存取 session 数据	114
实例 41	session 有效时间的设置	116
实例 42	session 时间与编号	118
实例 43	url 重组	120
实例 44	管理 session 变量	123
实例 45	网上测试	127
实例 46	分页显示问卷调查	131
实例 47	application 数据的存取	135
实例 48	月历	137
实例 49	计算器	141
实例 50	简易聊天室	146

第四篇 文件操作

实例 51	取得当前的磁盘路径	150
实例 52	文件的建立与删除	152
实例 53	目录的建立与删除	154
实例 54	取得文件属性	156
实例 55	显示目录中的文件	158
实例 56	判断文件是否空白	160

实例 57	读取文件中的字符	162
实例 58	略过字符不读取	164
实例 59	读取文件内容	166
实例 60	读取整行字符	168
实例 61	将数据写入文件	170
实例 62	分行写入数据	172
实例 63	文字型计数器	175
实例 64	用户注册	177

第五篇 数据库操作

实例 65	数据库链接的建立与关闭	182
实例 66	顺序取得数据	184
实例 67	逆序取得数据	187
实例 68	取得数据库中各栏名称	190
实例 69	取得记录数	193
实例 70	设置记录指针位置	196
实例 71	参数式查询	199
实例 72	用通配符查询	203
实例 73	数据完整查询	206
实例 74	设置显示页	209
实例 75	分页显示	213
实例 76	高级搜索	218
实例 77	添加数据	224
实例 78	添加表单中数据	227
实例 79	删除数据库中的数据	231
实例 80	修改数据	235
实例 81	整笔记录的修改	239

第六篇 使用 Java Bean

实例 82	色彩变换	246
实例 83	用 Bean 传递数据	250
实例 84	使用 Beans	252
实例 85	简易计算器	254
实例 86	用 Bean 链接数据库	257

实例 87	图形计数器	261
实例 88	随机大小的文字	264
实例 89	显示目录中内容	267
实例 90	读文件内容	270
实例 91	写内容到文件	274
实例 92	追加内容到文件	277
实例 93	二十一点	280
实例 94	猜数字	285
实例 95	网络订餐服务	290
实例 96	成绩单系统	295
实例 97	网上软件商店	300
实例 98	记事本	306
实例 99	发送邮件	310
实例 100	通信录	314

第一篇

基础页面和例外处理

本篇总览

本篇主要介绍 JSP 中构建页面中的最基本的语法和编写 JSP 页面必不可少的部分例外处理。

通过本篇中各个实例的制作,读者可以对 JSP 中,如何构建最基本的页面和经常使用的一些语法有一个大概的了解。而熟悉了例外处理的几个例子的制作,读者便可认识到 JSP 的错误处理机制,以及在以后的程序编写过程中如何通过例外处理去解决程序中的错误。

实例 1 依此变小的文字

实例说明

本例制作了一个显示不同大小文字的例子，如图 1-1 所示。

本例中在显示文字的同时，利用 Java 语言的 for 循环，向客户端输出了六行逐渐变小的文字，达到了能显示不同大小文字的效果。

本例知识点有：Java 语言的 for 循环语句。



图 1-1 效果图

编程思路

本实例的目的就是向客户端输出一段文字：Hello World!。我们在通过 print 语句向客户端输出信息的同时，又利用了 for 循环语句来控制输出语句的文字大小，每次输出语句的文字依次变小，就形成了不同大小的文字。

创作步骤

一、脚本代码的编写

以下就是所用到的控制文字大小的 for 循环语句：

```
//控制文字大小的循环语句
```

```
for( int i=1; i<=6; i++ )
```

```
out.println( "<h" + i + ">Hello World!</h" + i + ">" )
```

二、源代码

```
<%@ page language="Java" %>
```

```
<%@ page contentType="text/html; charset=GB2312"%>
```

```
<html>
```

```
<head>
```

```
<title>Hello World</title>
```

```
<meta http-equiv="Content-Type" content="content=text/html; charset=gb2312">
```

```
</head>
```

```
<body bgcolor="#FFFFFF">
```

```
<center>
```

```
<font size= 5 color=blue>多种字体大小显示</font>
```

```
<center>
```

```
<br>
```

```
<hr>
```

```
<br>
```

```
<div align="center">
```

```
<%
```

这里插入上面的脚本代码。

```
%>
```

```
</div>
```

```
</body>
```

```
</html>
```

实例 2 获得当前时间

实例说明

本例制作一个获得当前时间的例子,如图 2-1 所示。

本例中,通过对获得的小时、星期、日期、月份、年份等时间信息的处理,输出当前的时间。

本例知识点有:Java 语言的 Date 函数和字符串数组。

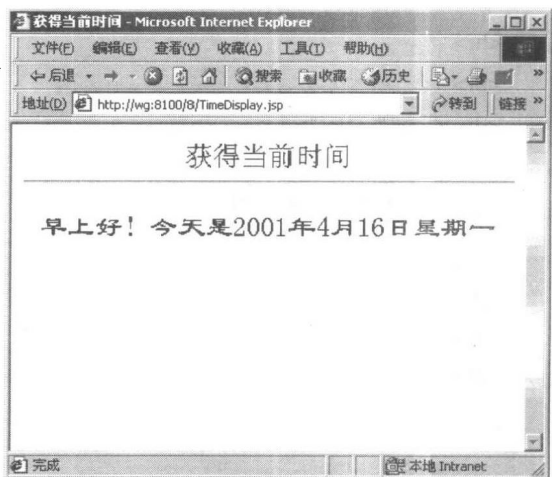


图 2-1 效果图

编程思路

本实例的目的就是向客户端显示当前的时间。首先,用 Date 函数获得所有需要的信息,包括小时、星期、日期、月份、年份。对小时进行判断,当它在 0~12 之间,就输出“早上好!”,否则,输出“下午好!”。对获得的星期,和字符串数组结合起来使用后,就获得了星期几的信息。把所有的信息处理完后,将其输出到客户端,我们就获得了当前时间。

创作步骤

一、脚本代码的编写

以下就是用到的脚本代码:

```
Date today = new Date();
int hours = today.getHours();
int minute = today.getMinutes();
//判断是上午还是下午
if(hours>=0 && hours<12){
    out.println("早上好!");
}else{
    out.println("下午好!");
}
//把当前的星期转换为中文所用到的数组
String[] weekdays = {"日","一","二","三","四","五","六"};
out.println("\n 今天是"
```

```
+ (today.getYear()+1900) + "年"  
+ (today.getMonth()+1) + "月"  
+ (today.getDate()) + "日星期"  
+ weekdays[today.getDay()];
```

二、源代码

```
<%@ page contentType="text/html; charset=gb2312" %>  
<%@ page import="Java.util.*" %>  
<HTML>  
<HEAD>  
<META HTTP-EQUIV="Content-Type" content="text/html; charset=gb2312">  
<TITLE>获得当前时间</TITLE>  
</HEAD>  
<BODY>  
<center>  
<font size=5 color=blue>获得当前时间</font>  
<br>  
<hr>  
<br>  
<font color=blue size=5 face="隶书">  
<%  
这里插入上面的脚本代码。  
>%>  
</font>  
</center>  
</BODY>  
</HTML>
```

实例 3 简单计数器

实例说明

本例制作了一个简单计数器的例子,如图 3-1 所示。

本例中,只要打开 counter.JSP 这个文件或者访问这个文件,计数器的数字~都会加 1,从而记录访问的人数。

本例知识点有:同步处理函数和 application 对象。

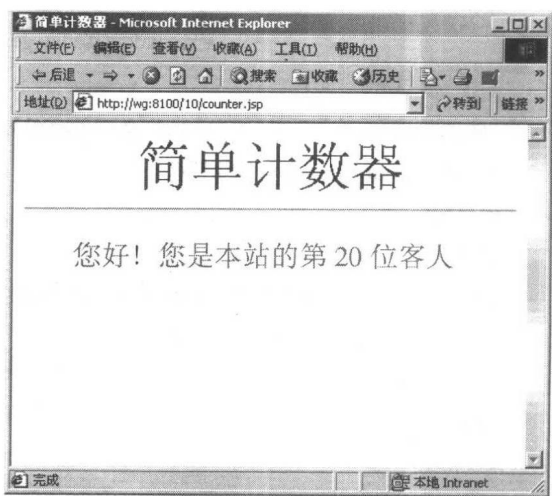


图 3-1 效果图

编程思路

本实例的作用是对访问量进行简单的统计。访问 counter.JSP 这个文件时,就将访问量加 1,并保存到内存中,再访问时,调出保存访问量的变量,加 1,再保存到内存中。为了防止一个以上的访客同时访问,又利用了同步处理。

创作步骤

一、脚本代码的编写

以下就是用到的脚本代码:

```
int count = null;
//同步处理
synchronized (application)
{
    //从内存当中读取访问量
    count = (Integer) application.getAttribute("basic.counter");
    if (count == null)
        count = new Integer(0);
    count = new Integer(count.intValue() + 1);
    //将访问量保存到内存当中
    application.setAttribute("basic.counter", count);
}
```

二、源代码

```
<%@ page contentType="text/html; charset=gb2312" %>
<%@ page language="Java" %>
<%
    Integer count = null;
    //同步处理
    synchronized (application)
    {
        //从内存当中读取访问量
        count = (Integer) application.getAttribute("basic.counter");
        if (count == null)
            count = new Integer(0);
        count = new Integer(count.intValue() + 1);
        //将访问量保存到内存当中
        application.setAttribute("basic.counter", count);
    }
%>
<html>
<head>
<title>简单计数器</title>
</head>
<body>
<center>
<font size=10 color=blue>简单计数器</font>
<br>
<hr>
<br>
<font size=5 color=blue>您好! 您是本站的第 <%= count %> 位客人</font>
</center>
</body>
</html>
```


实例 4 设置页面属性

实例说明

本例制作了设置页面属性的例子,如图 4-1 所示。

本例中,我们首先利用 `info` 指令设置了页面的属性,然后调出页面属性,并把它显示到客户端。

本例知识点有: `info` 指令的使用。

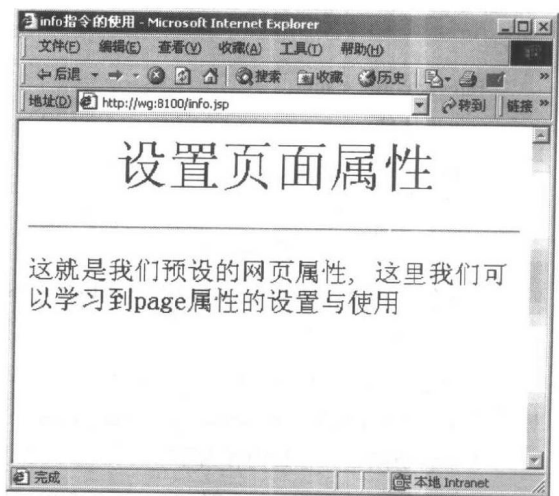


图 4-1 效果图

编程思路

本实例的作用是设置网页的页面属性,也就是网页的说明信息。首先用 `info` 指令设置网页的页面属性。设置语法为: `<%@ page info="说明信息" %>`,接着,调用 `Servlet` 类的 `getServletInfo` 方法获得设置的页面属性。

创作步骤

一、脚本代码的编写

以下就是用到的重点脚本代码:

```
<!-- 设置页面属性 -->  
<%@ page info="网页页面属性" %>  
<!--获得设置的页面属性-->  
    <%= getServletInfo() %>
```

二、源代码

```
<%@ page contentType="text/html; charset=GB2312" %>  
<%@ page language="Java" %>  
以下就是用到的脚本代码。  
<HTML>  
<HEAD>  
<TITLE>设置页面属性</TITLE>  
</HEAD>
```