

Chapter 25 - Active Server Pages (ASP)

Outline

- 25.1 Introduction
- 25.2 How Active Server Pages Work
- 25.3 Setup
- 25.4 Active Server Page Objects
- 25.5 Simple ASP Examples
- 25.6 File System Objects
- 25.7 Session Tracking and Cookies
- 25.8 Accessing a Database from an Active Server Page
- 25.9 Server-Side ActiveX Components
- 25.10 Internet and World Wide Web Resources



25.1 Introduction

- Server-side technologies
 - Dynamically creates Web pages
 - Uses client information, server information and information from the Internet
 - Active Server Pages (ASP)
 - Microsoft Server-side technology
 - Dynamically builds documents in response to client requests
 - Delivers dynamic Web content
 - XHTML, DHTML, ActiveX controls, client-side scripts and Java applets



25.2 How Active Server Pages Work

- Active Server Pages
 - Processed by scripting engine
 - Server-side ActiveX control
 - **.asp** file extension
 - Can contain XHTML tags
 - Scripting written with VBScript
 - JavaScript also used
 - Others (Independent Software Vendors)
 - Communication with Server
 - Client HTTP request to server
 - Active server page processes request and returns results



25.2 How Active Server Pages Work

- Active Server Pages, cont.
 - Communication with Server, cont.
 - ASP document is loaded into memory
 - **asp.dll** scripting engine on server
 - Parses (top to bottom)



25.3 Setup

- Web Server
 - Need to run Web server to use Active Server Pages
 - IIS (Internet Information Services) or
 - PWS 4.0 (Personal Web Server 4.0)
 - Create a virtual directory
 - Copy files to `c:\InetPub\Wwwroot` or `c:\Webshare\Wwwroot`



25.4 Active Server Page Objects

- Built-in objects
 - Communicate with a Web browser
 - Gather data sent by HTTP request
 - Distinguish between users
 - **Request**
 - Get or post information
 - Data provided by the user in an XHTML form
 - Access to information stored on client machine
 - Cookies
 - File upload (binary)
 - **Response**
 - Sends information to client
 - XHTML, text
 - **Server**
 - Access to server methods or properties



25.4 Active Server Page Objects

Object Name	Description
Request	Used to access information passed by an HTTP request.
Response	Used to control the information sent to the client.
Server	Used to access methods and properties on the server.

Fig. 25.1 Commonly used ASP objects.



```

1  <% @LANGUAGE = VBScript %>
2
3  <%
4      ' Fig. 25.2 : clock.asp
5      ' A simple ASP example
6      Option Explicit
7  %>
8
9  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
10     "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
11
12  <html xmlns = "http://www.w3.org/1999/xhtml">
13
14      <head>
15          <title>A Simple ASP Example</title>
16
17          <style type = "text/css">
18              td { background-color: black;
19                  color: yellow }
20              strong { font-family: arial, sans-serif;
21                      font-size: 14pt; color: yellow }
22              p { font-size: 14pt }
23          </style>
24      </head>
25
26      <body>
27
28          <p><strong>A Simple ASP Example</strong></p>
29          <table border = "6">
30              <tr>
31                  <td>
32                      <% =FormatDateTime( Now, vbLongDate ) %>
33                  </td>
34              </tr>
35          </table>

```

Scripting delimiter wraps around code executed on the server.

Processing directive Requires programmer explicitly define all variables

Returns server date and time (Now) as a string formatted in vbLongDate format

"<%= " is short for Response.write



Outline

Clock.asp

sends Web server's date and time to the client as XHTML markup

<% %>
scripting delimiter

@LANGUAGE processing directive

Option Explicit

FormatDateTime

Now
vbLongDate format

Response.write

Time

```

36         <td>
37             <% =Time() %>
38         </td>
39     </tr>

```

Statement is short for:

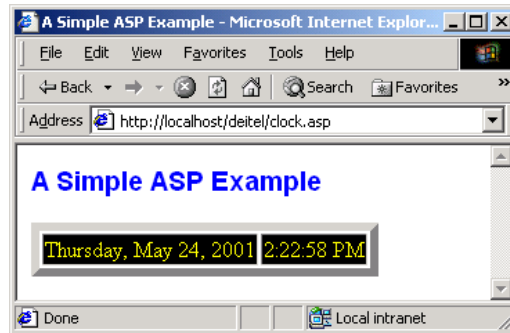
```
<% Call Response.write( Time()) %>
```



Outline

Clock.asp

Time



Program Output

© 2001 Prentice Hall, Inc.
All rights reserved.

```

1  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
2  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3
4  <html xmlns = "http://www.w3.org/1999/xhtml">
5
6      <head>
7          <title>A Simple ASP Example</title>
8
9          <style type = "text/css">
10             td        { background-color: black;
11                         color: yellow }
12             strong    { font-family: arial, sans-serif;
13                         font-size: 14pt; color: blue }
14             p          { font-size: 14pt }
15          </style>
16
17      </head>
18
19      <body>
20
21          <p><strong>A Simple ASP Example</strong></p>
22          <table border = "6">
23              <tr>
24                  <td>
25                      Thursday, May 24, 2001
26                  </td>
27
28                  <td>
29                      2:22:58 PM
30                  </td>
31              </tr>
32          </table>
33      </body>
34
35  </html>

```



Outline

Fig. 25.3

XHTML generated by
clock.asp.

© 2001 Prentice Hall, Inc.
All rights reserved.

25.5 Simple ASP Examples

- ASP processes input
 - Form information sent by client
 - E-commerce Web site
 - Use to verify customer information
 - Server responds to process request
 - Form
 - Using **post** method
 - **action** attribute indicates the .asp file to which the form information is posted
 - **request** object retrieves form data

© 2001 Prentice Hall, Inc. All rights reserved.



12



Outline

Name.html

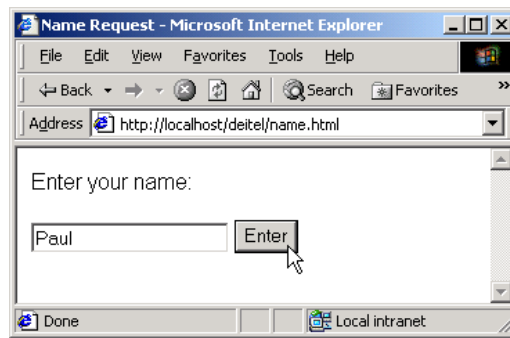
passes information into an ASP document using the **post** method.

```

1  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
2    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3
4  <!-- Fig. 25.4 : name.html -->
5  <!-- XHTML document that request an ASP document -->
6
7  <html xmlns = "http://www.w3.org/1999/xhtml">
8
9    <head>
10     <title>Name Request</title>
11   </head>
12
13   <body>
14
15     <p style = "font-family: arial, sans-serif">
16       Enter your name:
17     </p>
18
19     <!-- request name.asp when posted -->
20     <form action = "name.asp" method = "post">
21       <input type = "text" name = "namebox" size = "20" />
22       <input type = "submit" name = "submitButton"
23         value = "Enter" />
24     </form>
25
26   </body>
27
28 </html>

```

action set to ASP file where information is posted.



Name.html

Program Output

© 2001 Prentice Hall, Inc.
All rights reserved.



```
1  <% @LANGUAGE = VBScript %>
2
3  <%
4      ' Fig. 25.5 : name.asp
5      ' Another simple ASP example
6      Option Explicit
7  %>
8
9  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
10     "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
11
12  <html xmlns = "http://www.w3.org/1999/xhtml">
13
14      <head>
15          <title>Name Information</title>
16
17          <style type = "text/css">
18              p      { font-family: arial,
19                      font-size: 14pt; co
20              .special { font-size: 20pt; co
21          </style>
22      </head>
23
24      <body>
25
26          <!-- retrieve and display namebox's value -->
27          <p>Hi <% =Request( "namebox" ) %>, </p><br />
28          <p class = "special">Welcome to ASP!</p>
29
30      </body>
31
32  </html>
```

Request object retrieves
form data from textfield
"namebox"

Name.asp

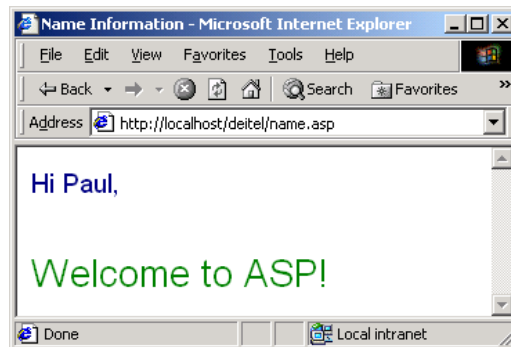
Uses the **Request**
method to retrieve the data
posted in **Name.html**.
Places this data within
XHTML tags for output.

© 2001 Prentice Hall, Inc.
All rights reserved.



Name.asp

Program Output



© 2001 Prentice Hall, Inc.
All rights reserved.

25.6 File System Objects

- File System Objects (FSOs)
 - Allow programmer to manipulate files, directories and drives
 - Read and write text
 - Microsoft Scripting Runtime Library (Fig 25.6)
 - **FileSystemObject**, **File**, **Folder**, **Drive** and **TextStream**
 - Use to create directories, move files, determine whether a **Drive** exists
 - **FileSystemObject** methods (Fig. 25.7)
 - **File** object
 - Allows programmer to gather info about files, manipulate files, open files
 - **File** properties and methods (Fig. 25.8)



25.6 File System Objects

Object type	Description
FileSystemObject	Allows the programmer to interact with Files , Folders and Drives .
File	Allows the programmer to manipulate Files of any type.
Folder	Allows the programmer to manipulate Folders (i.e., directories).
Drive	Allows the programmer to gather information about Drives (hard disks, RAM disks—computer memory used as a substitute for hard disks to allow high-speed file operations, CD-ROMs, etc.). Drives can be local or remote.
TextStream	Allows the programmer to read and write text files.
Fig. 25.6 File System Objects (FSOs).	



25.6 File System Objects

Methods	Description
CopyFile	Copies an existing File .
CopyFolder	Copies an existing Folder .
CreateFolder	Creates and returns a Folder .
CreateTextFile	Creates and returns a text File .
DeleteFile	Deletes a File .
DeleteFolder	Deletes a Folder .
DriveExists	Tests whether or not a Drive exists. Returns a boolean.
FileExists	Tests whether or not a File exists. Returns a boolean.
FolderExists	Tests whether or not a Folder exists. Returns a boolean.
GetAbsolutePathName	Returns the absolute path as a string.
Fig. 25.7 FileSystemObject methods. (Part 1 of 2)	



25.6 File System Objects

Methods	Description
GetDrive	Returns the specified Drive .
GetDriveName	Returns the Drive drive name.
GetFile	Returns the specified File .
GetFileName	Returns the File file name.
GetFolder	Returns the specified Folder .
GetParentFolderName	Returns a string representing the parent folder name.
GetTempName	Creates and returns a string representing a file name.
MoveFile	Moves a File .
MoveFolder	Moves a Folder .
OpenTextFile	Opens an existing text File . Returns a TextStream .

Fig. 25.7 **FileSystemObject** methods. (Part 2 of 2)



25.6 File System Objects

Property/method	Description
<i>Properties</i>	
DateCreated	Date. The date the File was created.
DateLastAccessed	Date. The date the File was last accessed.
DateLastModified	Date. The date the File was last modified.
Drive	Drive . The Drive where the file is located.
Name	String. The File name.
ParentFolder	String. The File 's parent folder name.
Path	String. The File 's path.
ShortName	String. The File 's name expressed as a short name.
Size	Variant. The size of the File in bytes.
<i>Methods</i>	
Copy	Copy the File . Same as CopyFile of FileSystemObject .
Delete	Delete the File . Same as DeleteFile of FileSystemObject .
Move	Move the File . Same as MoveFile of FileSystemObject .
OpenAsTextStream	Opens an existing File as a text File . Returns TextStream .

Fig. 25.8 Some common **File** properties and methods.



25.6 File System Objects

- File System Objects (FSOs)
 - **Path** property
 - Contains **File** path in long name format
 - **ShortName** property
 - Contains **File** path in short name format
 - **Folder** object
 - Allows programmer to manipulate and gather info about directories
 - **Folder** properties (Fig. 25.9)



25.6 File System Objects

Property/method	Description
<i>Properties</i>	
Attributes	Integer. Value indicating Folder 's attributes (read only, hidden, etc.).
DateCreated	Date. The date the folder was created.
DateLastAccessed	Date. The date the folder was last accessed.
DateLastModified	Date. The date the folder was last modified.
Drive	Drive . The Drive where the folder is located.
IsRootFolder	Boolean. Indicates whether or not a Folder is a root folder.
Name	String. The Folder 's name.
ParentFolder	Folder . The Folder 's parent folder.
Path	String. The Folder 's path.
ShortName	String. The Folder 's name expressed as a short name.
ShortPath	String. The Folder 's path expressed as a short path.
Size	Variant. The total size in bytes of all subfolders and files.
Type	String. The Folder type.
Fig. 25.9 Some Folder properties and methods. (Part 1 of 2)	



25.6 File System Objects

Property/method	Description
<i>Methods</i>	
Delete	Delete the Folder . Same as DeleteFolder of FileSystemObject .
Move	Move the Folder . Same as MoveFolder of FileSystemObject .
Copy	Copy the Folder . Same as CopyFolder of FileSystemObject .
Fig. 25.9 Some Folder properties and methods. (Part 2 of 2)	



25.6 File System Objects

- File System Objects (FSOs)
 - **IsRootFolder** property
 - Indicates whether folder is the root folder for the Drive
 - If not:
 - Method **ParentFolder**
 - Retrieves parent folder
 - Method **Size**
 - Returns the total bytes the folder contains (includes subfolders)



25.6 File System Objects

- File System Objects (FSOs)
 - Drive object (Fig. 25.10)
 - Gather information about drives
 - Property DriveLetter
 - Contains drive letter
 - Property SerialNumber
 - Contains drive serial number
 - Property FreeSpace
 - Contains number of available bytes



25.6 File System Objects

Property	Description
AvailableSpace	Variant. The amount of available Drive space in bytes.
DriveLetter	String. The letter assigned to the Drive (e.g., “C”).
DriveType	Integer. The Drive type. Constants Unknown , Removable , Fixed , Remote , CDRom and RamDisk represent Drive types and have the values 0–5, respectively.
FileSystem	String. The file system Drive description (FAT, FAT32, NTFS, etc.).
FreeSpace	Variant. Same as AvailableSpace .
IsReady	Boolean. Indicates whether or not a Drive is ready for use.
Path	String. The Drive ’s path.
RootFolder	Folder object. The Drive ’s root Folder .
SerialNumber	Long. The Drive serial number.
TotalSize	Variant. The total Drive size in bytes.
VolumeName	String. The Drive volume name.
Fig. 25.10 Drive properties.	



25.6 File System Objects

- File System Objects (FSOs)
 - **TextStream** object (Fig. 25.11)
 - Manipulate text files



25.6 File System Objects

Property/Method	Description
<i>Properties</i>	
AtEndOfLine	Boolean. Indicates whether the end of a line has been encountered.
AtEndOfStream	Boolean. Indicates whether the end of file has been encountered.
Column	Integer. Returns the character's position in a line.
Line	Integer. Returns the current line number.
<i>Methods</i>	
Read	String. Returns a specified number of characters from the file referenced by the TextStream object.
ReadAll	String. Returns the entire contents of the file referenced by the TextStream object.
Fig. 25.11 TextStream methods and properties. (Part 1 of 2)	



25.6 File System Objects

Property/Method	Description
<i>Methods, cont.</i>	
ReadLine	String. Returns one line from the file referenced by the TextStream object.
Write	String. Writes text to the file referenced by the TextStream object.
WriteBlankLines	String. Writes newline characters to the file referenced by the TextStream object.
WriteLine	String. Writes one line to the file referenced by the TextStream object.
Skip	Variant. Skips a specified number of characters while reading from the file referenced by the TextStream object.
SkipLine	Variant. Skips a line of characters while reading from the file referenced by the TextStream object.
Close	Close the file referenced by the TextStream object.
Fig. 25.11 TextStream methods and properties. (Part 2 of 2)	



25.6 File System Objects

Key Name	Description
APPL_PHYSICAL_PATH	Returns the physical path.
HTTPS	Boolean. Determines if the request came in through SSL (Secure Sockets Layer).
REMOTE_ADDR	Client's DNS name or IP address.
REQUEST_METHOD	Request method (i.e., get and post).
SERVER_NAME	Server's hostname (DNS or IP address).
HTTP_USER_AGENT	Returns information about the client making the request.
HTTP_COOKIE	Returns cookies residing on the client.
Fig. 25.13 Some server variable keys.	





Outline

Guestbook.asp

Users enter name, email address and comments. FSOs write this data to a server file.

<form>

post action

hiddenInput

```

66      <!-- write form to the client -->
67      <form action = "guestbook.asp" method = "post">
68          <table>
69              <tr>
70                  <td>Your Name: </td>
71                  <td><input class = "font"
72                      type = "text" size = "60"
73                      name = "username" /></td>
74              </tr>
75              <tr>
76                  <td>Your email address:</td>
77                  <td><input class = "font"
78                      type = "text" size =
79                      name = "email"
80                      value = "user@isp.com" />
81              </td>
82          </tr>
83          <tr>
84              <td>Tell the world: </td>
85              <td><textarea name = "comment" rows = 5
86                  Replace this text with the information
87                  you would like to post.</textarea></td>
88              </tr>
89          </table>
90
91          <input type = "submit" value = "submit" />
92          <input type = "reset" value = "clear" />
93          <input type = "hidden" name = "hiddenInput"
94              value = "true" />
95      </form>
96
97
98
99
  
```

Form post action to .asp page

Form contains two text fields and text area for user input

Passes parameter hiddenInput value "true"

© 2001 Prentice Hall, Inc.
All rights reserved.



Outline

Guestbook.asp

Users enter name, email address and comments. FSOs write this data to a server file.

```

100  <%
101      ' check if the file exists
102      If fileObject.FileExists( guestBook ) = True Then
103
104
105          ' open the guestbook, "1" is for read
106          Set textFile = fileObject.OpenTextFile( guestBook, 1 )
107
108          ' read the entries from the file and write them to the client.
109          Call Response.Write( "Guestbook Entries:<br />" & _
110              textFile.ReadAll() )
111          Call textFile.Close()
112      End If
113  %>
114
115  </body>
116
117  </html>
118
119
  
```

FSO object FileExists checks if guestbook.txt exists.

Lines 100-115 execute every time the client connects to the ASP page

TextStream and ReadAll methods read entire contents of textFile

Read entries from guestbook.txt and write to client

Response.Write writes text to client. Contains XHTML markup rendered on client browser.

Response.Write

© 2001 Prentice Hall, Inc.
All rights reserved.



Program Output

GuestBook Example - Microsoft Internet Explorer

Address: http://localhost/deitel/guestbook.asp

Please leave a message in our guestbook.

Your Name:

Your email address:

Tell the world:

GuestBook Example - Microsoft Internet Explorer

Address: http://localhost/deitel/guestbook.asp

Thanks for your entry, tem!

Please leave a message in our guestbook.

Your Name:

Your email address:

Tell the world:

Guestbook Entries:

5/24/2001 [tem](#): ASP is a great server-side development tool.

© 2001 Prentice Hall, Inc.
All rights reserved.



Program Output

GuestBook Example - Microsoft Internet Explorer

Address: http://localhost/deitel/guestbook.asp

Thanks for your entry, tem!

Please leave a message in our guestbook.

Your Name:

Your email address:

Tell the world:

Guestbook Entries:

5/24/2001 [tem](#): ASP is a great server-side development tool.

GuestBook Example - Microsoft Internet Explorer

Address: http://localhost/deitel/guestbook.asp

Thanks for your entry, dan!

Please leave a message in our guestbook.

Your Name:

Your email address:

Tell the world:

Guestbook Entries:

5/24/2001 [tem](#): ASP is a great server-side development tool.

5/24/2001 [dan](#): ASP is my preferred server-side development tool.

© 2001 Prentice Hall, Inc.
All rights reserved.



```
1 <hr />5/24/2001 <a href = "mailto:tem@deitel.com">tem</a>: ASP is a
great tool for server-side development.
2 <hr />5/24/2001 <a href = "mailto:dan@bushycat.com">dan</a>: ASP is
my preferred server-side development tool.
```

XHTML generated by
guestbook.asp

© 2001 Prentice Hall, Inc.
All rights reserved.

25.7 Session Tracking and Cookies

- Session Tracking and Cookies
 - Helps server to distinguish between clients
 - Provide custom content
 - Shopping cart
 - Marketing/advertising
 - **SessionID**
 - Assigned by server to each unique session
 - Compared with sessionIDs stored in server
 - **Session** object
 - **Timeout** property
 - Length of session before it expires
 - **Abandon** property
 - Terminates individual session



25.7 Session Tracking and Cookies

- ASP application
 - Multiple ASP pages linked with session variables
 - Example
 - **instantpage.asp**
 - Form, requests information from the user
 - Posted to **process.asp**
 - **process.asp**
 - Redirects user to **instantpage.asp** if errors exist
 - Otherwise creates users ASP page
 - Stores message in session variable **welcomeBack**
 - Every time user submits the form



25.7 Session Tracking and Cookies

- Server-side include
 - Commands embedded in XHTML documents
 - Add dynamic content
 - Places a **.shtml** include file within another file
 - Physical or virtual path
 - `<!--#include file = "includes\file.shtml" -->`
 - Not all Web servers support
 - Written as comment
 - Performed before scripting code is interpreted.
 - ASP page cannot determine which includes to use
 - Can contain scripting code
 - Must use `<script>` tag or `<% %>` delimiters





Outline

Instantpage.asp

Receives error message from **process.asp**

Uses server side include to create dynamic content (embedded file)

```

1  <% @LANGUAGE = VBScript %>
2
3  <%
4      ' Fig. 25.15 : instantpage.asp
5      ' ASP document that posts data to process.asp
6      Option Explicit
7  %>
8
9  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
10     "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
11
12  <html xmlns = "http://www.w3.org/1999/xhtml">
13
14      <head>
15          <title>Instant Page Content Builder</title>
16
17          <style type = "text/css">
18              table { text-align: center;
19                  font-size: 12pt;
20                  color: blue;
21                  font-size: 12pt;
22                  font-family: arial, sans-serif }
23          </style>
24      </head>
25
26      <body>
27
28          <!-- include the header -->
29          <!-- #include virtual = "/includes/header.shtml" -->
30          <h2>Instant Page Content Builder</h2>
31
32  </body>
33  </html>

```

Server-side include

© 2001 Prentice Hall, Inc.
All rights reserved.



Outline

Instantpage.asp

session variable with no explicit value contains an empty string

```

33  <% ' if process.asp posted an error, print the error
34      ' message.
35  If Session( "errorMessage" ) <> "no error" Then
36      Call Response.Write( Session( "errorMessage" ) )
37      ' otherwise, print the welcome back message, if any
38  Else
39      Call Response.Write( Session( "welcome" ) )
40  End If
41  %>
42
43  <!-- a form to get the information for process.asp -->
44  <form action = "process.asp" method = "post">
45      <table>
46          <tr>
47              <td>Your Name: </td>
48              <td><input type = "text" size = "60"
49                  name = "username" value = "yourname" /></td>
50          </tr>
51      </table>
52
53      <tr>
54          <td>Enter the Filename:</td>
55          <td><input type = "text" size = "60"
56              name = "filename" value = "yourfilename" /></td>
57      </tr>
58
59      <tr>
60          <td>Enter the Title:</td>
61          <td><input type = "text" size = "60"
62              name = "doctitle" value = "document title" /></td>
63      </tr>
64  </table>
65  </form>
66
67  </body>
68  </html>

```

Session object retrieves variable value

When line 36 is executed **errorMessage** never has a value unless errors are encountered

Return True or False

Requests **process.asp** when form is posted

© 2001 Prentice Hall, Inc.
All rights reserved.



```

68
69         <tr>
70             <td>Enter the content:</td>
71
72             <td><textarea name = "content" rows = "3"
73                 cols = "50">
74 Replace this text with the
75 information you would like to post.</textarea></td>
76         </tr>
77     </table>
78
79     <input type = "submit" value = "submit" />
80     <input type = "reset" value = "clear" />
81 </form>
82
83     <!-- #include virtual = "/includes/footer.shtml" -->
84 </body>
85 </html>

```

Server-side include



Instant Page Content Builder - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites History

Address http://localhost/deitel/instantpage.asp

Instant Page Content Builder

Your Name:

Enter the Filename:

Enter the Title:

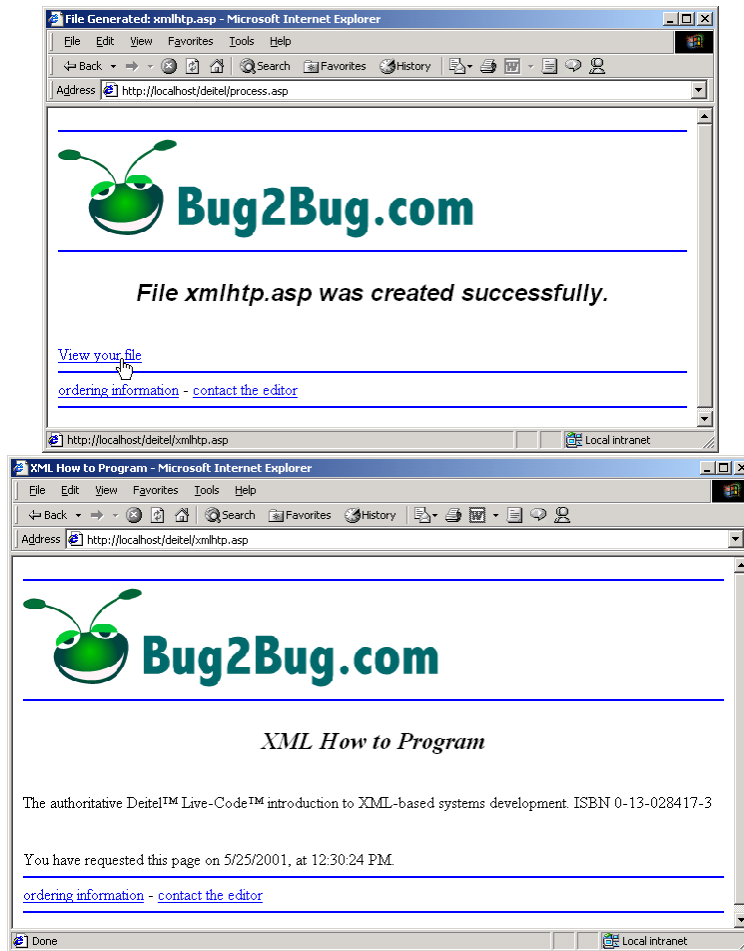
Enter the content:

[ordering information](#) - [contact the editor](#)

Done Local intranet



Program Output



© 2001 Prentice Hall, Inc.
All rights reserved.



Header.shtml

```
1  <!-- Fig. 25.16: header.shtml -->
2  <!-- Server-side include file containing XHTML -->
3  <hr style = "color: blue" />
4  <img height = "100" src = "/images/bug2bug.gif" alt = "Bug" />
5  <hr style = "color: blue" />
```

server-side include file for
the document header

© 2001 Prentice Hall, Inc.
All rights reserved.



```

1  <!-- Fig. 25.17: footer.shtml -->
2  <!-- Server-side include file containing XHTML -->
3  <hr style = "color: blue" />
4  <a style = "text-align: center"
5    href = "mailto:orders">ordering information</a> -
6  <a style = "text-align: center"
7    href = "mailto:editor">contact the editor</a><br />
8  <hr style = "color: blue" />

```

Footer.shtml

server-side include file for
the document footer



```

1  <% @LANGUAGE = VBScript %>
2
3  <%
4    ' Fig. 25.18 : process.asp
5    ' ASP document that creates user's ASP document
6    Option Explicit
7  %>
8
9  <%
10     Dim message, q
11
12     q = Chr( 34 ) ' assign quote character to q
13     Session( "errorMessage" ) = "no error"
14
15     ' check to make sure that they have entered
16     ' valid filename
17     If ( LCase( Request( "filename" ) ) = "you"
18       Or Request( "filename" ) = "" ) Then
19       message = "<p style = " & q & "color: red"
20         & ">" & "Please enter a valid name or
21         Session( "errorMessage" ) = message
22         Call Server.Transfer( "instantpage.asp" )
23     End If
24
25     Dim directoryPath, filePath, fileName
26
27     ' Create a FileSystem Object
28     Set fileObject = Server.CreateObject( _
29       "Scripting.FileSystemObject" )
30
31     directoryPath = _
32       Request.ServerVariables( "APPL_PHYSICAL_PATH" )
33
34     fileName = Request( "filename" ) & ".asp"
35

```

If field is empty or contains default string **yourfilename**, lines 19-21 assign XHTML error message to variable **errorMessage**.

Assign **message** value to session variable **errorMessage**

FSO object is created if valid user name is entered

Request.ServerVariables retrieves physical path

Builds **fileName** by concatenating "filename" to the **.asp** file extension

Process.asp

Creates the user's ASP document and presents a link to the user's page

Requested by
instantpage.asp



Outline

Process.asp

Creates the user's ASP document and presents a link to the user's page

Requested by
instantpage.asp

```

36 ' build path for text file
37 filePath = directoryPath & "\" & fileName
38
39 ' check if the file already exists
40 If fileObject.FileExists( filePath ) Then
41     message = "<p style = " & q & "color: red" & q &
42         ">" & "The file name is in use" & q & "Please use a different file name" & q & ">"
43     Session( "errorMessage" ) = message
44     Call Server.Transfer( "instantpage.asp" )
45 End If
46
47 ' save XHTML for the welcome back message in a session variable
48 message = "<p style = " & q & "color: red" & q & ">" & "Welcome Back, " & Request.QueryString( "name" ) & q & ">"
49
50 Session( "welcomeBack" ) = message
51
52 Dim header, footer, textFile, openMark, closeMark
53 openMark = "<" & q & " " & q & ">"
54 closeMark = ">" & q & ">"
55
56 ' build the header.
57 ' vbCrLf inserts a carriage return and line feed
58 ' string which makes the XHTML code valid
59 header = openMark & " @LANGUAGE = " & q & "VBScript" & q & " " & closeMark & vbCrLf &
60     & "<!DOCTYPE html PUBLIC " & q & "-//W3C//DTD XHTML 1.0 Transitional//EN" & q & " " &
61     & "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd" & q & ">" & vbCrLf &
62     & "<html xmlns = " & q & "http://www.w3.org/1999/xhtml" & q & ">" & vbCrLf &
63     & "<head>" & vbCrLf &
64     & "<meta name = " & q & "author" & q & " content = " & q & Request( "username" ) & q & " />" & vbCrLf &
65     & "<meta name = " & q & "pubdate" & q & " content = " & q & Date() & q & " />" & vbCrLf &
66     & "<title>" & Request( "doctype" ) & "</title>" & vbCrLf &
67     & "</head>" & vbCrLf &
68     & "<!-- #include " & q & "virtual = " & q & " /includes/header.shtml -->" & vbCrLf &
69     & "<h2 style = " & q & "text-align: center" & q & ">" & Request( "doctype" ) & "</h2>" & vbCrLf &
70     & "<br />" & vbCrLf &
71     & "<!-- #include " & q & "virtual = " & q & " /includes/footer.shtml -->" & vbCrLf &
72     & "</body>" & vbCrLf &
73     & "</html>" & vbCrLf &
74     & "</html>" & vbCrLf &
75     & "</html>" & vbCrLf &
76     & "</html>" & vbCrLf &
77     & "</html>" & vbCrLf &
78     & "</html>" & vbCrLf &
79     & "</html>" & vbCrLf &
80     & "</html>" & vbCrLf &
81     & "</html>" & vbCrLf &
82     & "</html>" & vbCrLf &
83     & "</html>" & vbCrLf &
84     & "</html>" & vbCrLf &
85     & "</html>" & vbCrLf &
86     & "</html>" & vbCrLf &
87     & "</html>" & vbCrLf &
88     & "</html>" & vbCrLf &
89     & "</html>" & vbCrLf &
90     & "</html>" & vbCrLf &
91     & "</html>" & vbCrLf &
92     & "</html>" & vbCrLf &
93     & "</html>" & vbCrLf &
94     & "</html>" & vbCrLf &
95     & "</html>" & vbCrLf &
96     & "</html>" & vbCrLf &
97     & "</html>" & vbCrLf &
98     & "</html>" & vbCrLf &
99     & "</html>" & vbCrLf &
100    & "</html>" & vbCrLf &
101    & "</html>" & vbCrLf &
102    & "</html>" & vbCrLf &
103    & "</html>" & vbCrLf &
104    & "</html>" & vbCrLf &
105    & "</html>" & vbCrLf &

```

Builds file path by concatenating
filePath passed FileExists
method. Determines if file exists.

If file exists, variable
Server Transfer method requests
instantpage.asp
message

Construct XHTML for header
back message to
welcomeBack session variable

Assign ASP scripting
delimiters to string variables
openMark and closeMark

© 2001 Prentice Hall, Inc.
All rights reserved.



Outline

Process.asp

Creates the user's ASP document and presents a link to the user's page

Requested by
instantpage.asp

```

71 & "<meta name = " & q & "author" & q & " content = " & q & Request( "username" ) & q & " />" & vbCrLf &
72 & "<meta name = " & q & "pubdate" & q & " content = " & q & Date() & q & " />" & vbCrLf &
73 & "<title>" & Request( "doctype" ) & "</title>" & vbCrLf &
74 & "</head>" & vbCrLf &
75 & "<!-- #include " & q & "virtual = " & q & " /includes/header.shtml -->" & vbCrLf &
76 & "<h2 style = " & q & "text-align: center" & q & ">" & Request( "doctype" ) & "</h2>" & vbCrLf &
77 & "<br />" & vbCrLf &
78 & "<!-- #include " & q & "virtual = " & q & " /includes/footer.shtml -->" & vbCrLf &
79 & "</body>" & vbCrLf &
80 & "</html>" & vbCrLf &
81 & "</html>" & vbCrLf &
82
83 ' build the footer using a different style
84 ' building the string
85 footer = vbCrLf & "<br /><br /><br />" & vbCrLf &
86     & "You have requested this page on " & q & " " & closeMark & " " &
87     & openMark & " =Date() " & closeMark & " " &
88     & vbCrLf & "at " & openMark & " =Time() " & closeMark & " " &
89     & closeMark & " " & vbCrLf &
90     & "<!-- #include " & q & "virtual = " & q & " /includes/footer.shtml -->" & vbCrLf &
91     & "</body>" & vbCrLf &
92     & "</html>" & vbCrLf &
93     & "</html>" & vbCrLf &
94
95 ' create the ASP file
96 Set textFile = fileObject.CreateTextFile( filePath, False )
97 With textFile
98     Call .WriteLine( header & Request( "doctype" ) & footer )
99     Call .Close()
100 End With
101 %>
102
103 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
104 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
105

```

footer variable assigned to XHTML footer contents

Form values retrieved using Request object

Write header, text area content's text and footer to text file

Lines 103-129 send XHTML to client that contains link to created page

© 2001 Prentice Hall, Inc.
All rights reserved.



Outline

Process.asp

Creates the user's ASP and presents a user's page

by
instantpage.asp

Lines 103-129 send XHTML to client that contains link to created page

```

106 <html xmlns = "http://www.w3.org/1999/xhtml">
107
108 <head>
109 <!-- use the title given by the user -->
110 <title>File Generated: <% =fileName %></title>
111
112 <style type = "text/css">
113     h2 { font-family: arial, sans-serif;
114         text-align: center }
115 </style>
116
117 </head>
118
119 <body>
120 <!-- #include virtual = "/includes/header.shtml" -->
121 <h2><em>File <% =fileName %>
122     was created successfully.</em>
123 </h2><br />
124
125 <!-- provide a link to the generated page -->
126 <a href = "<% =fileName %>">View your file</a>
127 <!-- #include virtual = "/includes/footer.shtml" -->
128 </body>
129 </html>

```

© 2001 Prentice Hall, Inc.
All rights reserved.



Outline

Test.asp

XHTML file created by
process.asp

```

1 <% @LANGUAGE = VBScript %>
2 <% ' test.asp %>
3
4 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
5 "http://www.w3c.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
6 <html xmlns = "http://www.w3.org/1999/xhtml">
7 <head>
8 <meta name = "author" content = "tem" />
9 <meta name = "pubdate" content = "2/27/2001" />
10 <title>XML How to Program</title>
11 </head>
12 <body>
13 <!-- Fig. 25.16: header.shtml -->
14 <!-- Server-side include file containing XHTML -->
15 <hr style = "color: blue" />
16 <img height = "100" src = "/images/bug2bug.gif" alt = "Bug" />
17 <hr style = "color: blue" />
18 <h2 style = "text-align: center"><em>XML How to Program</em></h2>
19 <br />
20
21 The authoritative Deitel&#8482; Live-Code&#8482;
22 introduction to XML-based systems development.
23 ISBN 0-13-028417-3
24
25 <br /><br /><br />
26 You have requested this page on 2/27/2001,
27 at 10:14:44 PM.
28 <!-- Fig. 25.17: footer.shtml -->
29 <!-- Server-side include file containing XHTML -->
30 <hr style = "color: blue" />
31 <a style = "text-align: center"
32     href = "mailto:orders">ordering information</a> -

```

© 2001 Prentice Hall, Inc.
All rights reserved.



```

33 <a style = "text-align: center"
34     href = "mailto:editor">contact the editor</a><br />
35 <hr style = "color: blue" />
36
37 </body>
38 </html>

```

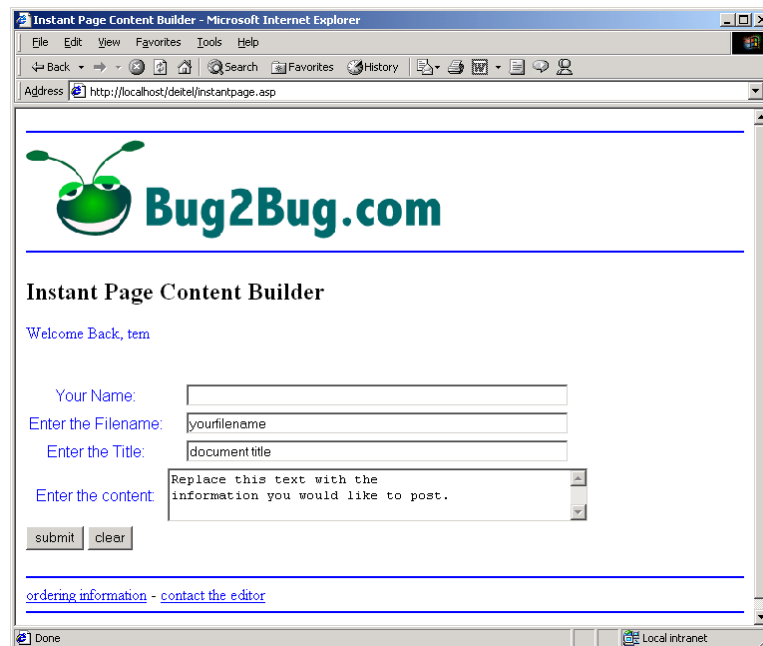


Fig. 25.20 Welcome back message displayed by **instantpage.asp**.

© 2001 Prentice Hall, Inc.
All rights reserved.

25.7 Session Tracking and Cookies

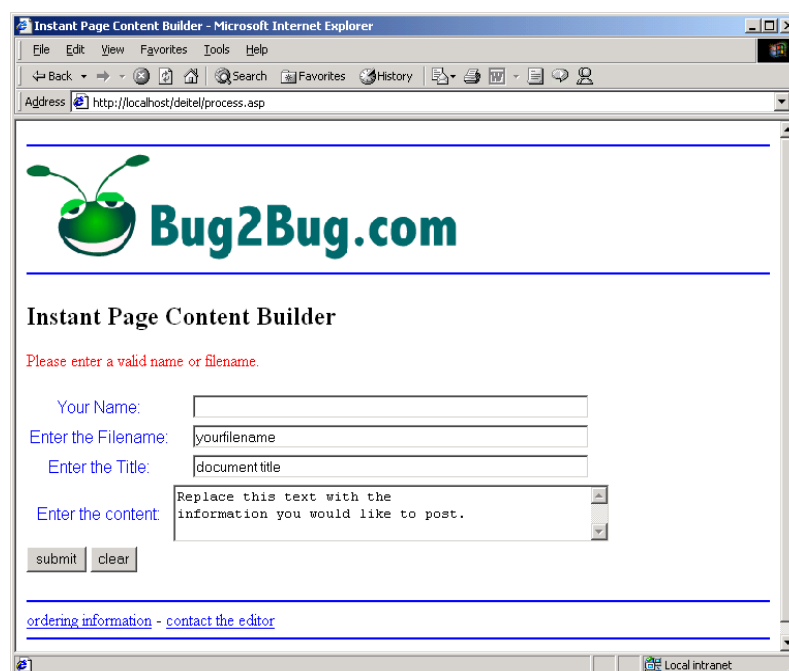


Fig. 25.21 Error message generated by **instantpage.asp**.



25.8 Accessing a Database from an Active Server Page

- Web applications
 - Communicating with databases
 - Use ActiveX Data Objects (ADO)
 - Provides uniform way for programs to connect with databases
 - Three-tier distributed applications
 - User interface
 - Created with XHTML, DHTML or XML
 - Contains ActiveX controls, client-side scripts, Java applets
 - Communicate directly with business logic
 - Business logic
 - Database access
 - May reside on separate computers



25.8 Accessing a Database from an Active Server Page

- Web applications, cont.
 - Three-tier distributed applications, cont.
 - Web servers build middle tier
 - Provide business logic
 - Manipulates databases
 - Communicates with client Web browsers
 - ASP communications with databases
 - Use SQL-based queries
 - ADO handled specifics
 - Through OLE DB
 - Databases provide data source for dynamic content
 - Allows users who are not familiar with Web languages to create Web pages
 - Restrict access
 - Password protection
 - Query Access database



```

1  <% @LANGUAGE = VBScript %>
2
3  <%
4  ' Fig. 25.22 : database.asp
5  ' ASP document for interacting with the database
6  Option Explicit
7
8  Dim connection, loginData
9
10 ' provide error handling code
11 On Error Resume Next
12 Session( "errorString" ) = ""
13
14 Set connection = Server.CreateObject( "ADODB.Connection" )
15 Call connection.Open( "login" )
16 Call errorHandlerLog()
17
18 ' create the record set
19 Set loginData = Server.CreateObject( "ADODB.Recordset" )
20 Call loginData.Open( Session( "query" ), connection, adOpenStatic, adLockOptimistic )
21 Set Session( "loginData" ) = loginData
22
23 Call errorHandlerLog()
24
25 Sub errorHandlerLog()
26 If Err.Number <> 0 Then
27 Dim errorString
28
29 errorString = Session( "errorString" ) & "Error number: " & Err.Number & " Error message: " & Err.Description & "<br>"
30 errorString = errorString & "Error number: " & Err.Number & " Error message: " & Err.Description & "<br>"
31 errorString = errorString & "Error number: " & Err.Number & " Error message: " & Err.Description & "<br>"
32 errorString = errorString & "Error number: " & Err.Number & " Error message: " & Err.Description & "<br>"
33 errorString = errorString & "Error number: " & Err.Number & " Error message: " & Err.Description & "<br>"
34 Session( "errorString" ) = errorString
35 End Sub
36 End Sub
37 %>

```

Ignores errors until
Server method
CreateObject creates
ADODB.Connection
 Method **Open** opens
 When **Open** finishes
 executing, points to first
 record or EOF if no
errorHandlerLog
 called again
Error object Number
 property contains VBScript
 If true **errorString**
 Error number and message
 concatenated to variable
errorString
 Sets session variable **loginData** to
 variable **loginData** which references
 the **ADODB.Recordset** containing
 all records matching SQL query

Database.asp

connects to, and queries an Access database

CreateObject

ADODB.Connection contains functionality necessary to connect to database

errorHandlerLog

© 2001 Prentice Hall, Inc.
All rights reserved.

```

1  <% @LANGUAGE = VBScript %>
2
3  <%
4  ' Fig. 25.23 : login.asp
5  ' ASP document to login to instantpage.asp
6  Option Explicit
7
8  ' create the SQL query
9  Session( "query" ) = "SELECT loginID FROM Users"
10 Call Server.Execute( "database.asp" )
11 %>
12
13 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
14 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
15
16 <html xmlns="http://www.w3.org/1999/xhtml">
17
18   <head>
19     <title>Login Page</title>
20
21     <style type="text/css">
22       table { text-align: center;
23         font-size: 12pt;
24         color: blue;
25         font-size: 12pt;
26         font-family: arial, sans-serif }
27       .error { color: red }
28     </style>
29
30   </head>
31
32   <body>
33
34     <!-- #include virtual="/includes/header.shtml" -->
35   </body>
36 </html>
37 %>

```

Assigns SQL query

Executes **database.asp** to retrieve login IDs from the database

Login.asp

Identifies users by prompting them for login name and password.

Data is stored in Access database **login.mdb**

submitlogin.asp validates the user's login

© 2001 Prentice Hall, Inc.
All rights reserved.



Outline

Login.asp

Prompts user for login name and password.

Information is stored in Access database opened in **database.asp**

```

36  If Session( "errorString" ) = "" Then
37      ' if this is a return after a failed attempt,
38      ' print an error
39      If Session( "loginFailure" ) = True Then %>
40          <p class = "error">Login attempt failed,
41              please try again
42      <% End If
43
44      ' begin the form %>
45      <p>Please select your name and enter
46          your password to login:</p><br />
47
48      <form action = "submitlog
49
50      <!-- format the form u
51      <table border = "0">
52          <tr>
53              <td>Name:</td>
54
55              <td>
56                  <select name = "loginID">
57                      <option value = "noSel
58                          Select your name</option>
59
60      <%
61      If Request.Cookies( "loginID" ) <> "" Th
62          Call BuildReturning()
63      Else
64          Call BuildNewUser()
65      End If
66      %>
67
68          </select>
69          </td>
70          </tr>

```

Tests if session variable **errorString** value is empty string

If false, line 89 prints

Lines 39-41 test is session variable **loginFailure** is True

select structure builds drop-down list of **loginIDs** prompts new login

Requests **loginID** cookie

Selects the returning user's login ID **option**

Build **loginID** options

© 2001 Prentice Hall, Inc.
All rights reserved.



Outline

Login.asp

Prompts user for login name and password.

Information is stored in Access database opened in **database.asp**

```

71      <tr>
72          <td>Password:</td>
73          <td><input type = "password"
74              name = "password" /></td>
75      </tr>
76
77      <tr>
78          <td></td>
79          <td align = "left">
80              <input type = "submit" value = "Log Me In" />
81          </td>
82      </tr>
83  </table>
84  </form>
85
86  <!-- #include virtual="/includes/footer.shtml" -->
87  <%
88      Else
89          Call Response.Write( Session( "errorString" ) )
90      End If
91  %>
92  </body>
93  </html>
94
95  <%
96      ' builds the option items for loginIDs and writes
97      ' selected for the loginID of the returning user
98      Sub BuildReturning()
99          Dim found, loginData
100
101          Set loginData = Session( "loginData" )
102
103          ' pull user names from the record set to populate the
104          ' dropdown list

```

© 2001 Prentice Hall, Inc.
All rights reserved.

Login.asp

Prompts user for login name and password.

Information is stored in access database opened in database.asp

```

105 found = False
106
107 While Not loginData.EOF
108     ' create this record's dropdown entry
109     <option
110     <%
111     ' if we did not write selected for
112     ' before
113     If ( Not found ) Then
114         ' if the current record's loginID is
115         ' the loginID cookie, then it is
116         ' the returning user, and thus
117         ' selected for this option; in
118         ' need to signal that we have w
119         ' for an option by setting found = True
120         If Request.Cookies( "loginID" ) =
121             = loginData( "loginID" ) Then
122             Call Response.Write( "selected = "
123             Chr( 34 ) & "selected" & Chr( 34 )
124             found = True
125         End If
126     End If
127     <%>
128     value = "<% =loginData( "loginID"
129     <% =loginData( "loginID" ) %></opt
130     Call loginData.MoveNext()
131 Wend
132 End Sub
133
134 ' builds the option items for loginIDs w
135 ' selected for any loginID
136 Sub BuildNewUser()
137     Dim loginData
138     Set loginData = Session( "loginData"
139

```

Tests for EOF

found set to false before loop

selected for an option

If statement tests whether option needs to be selected

Increments the record set pointer to next record

If true, lines 122-124 write selected and set found to true

while record's loginID field is equal to loginID cookie

Writes option display as current loginID

Sets option value to current loginID

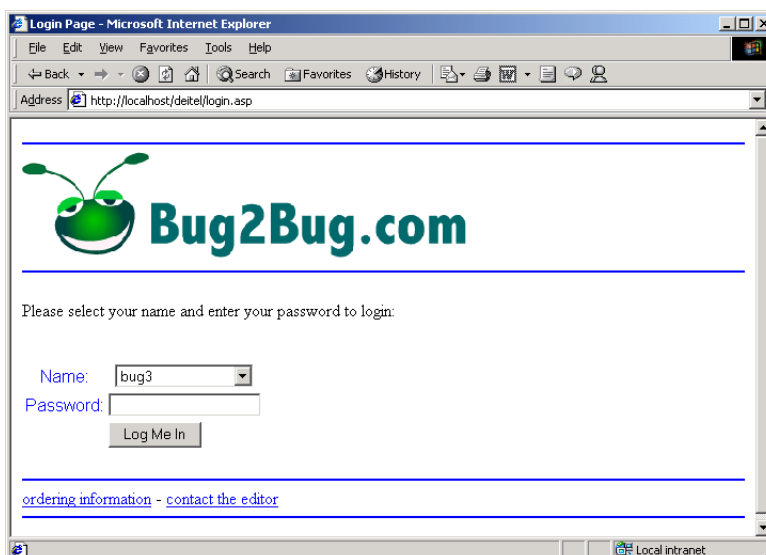
© 2001 Prentice Hall, Inc.
All rights reserved.

Login.asp

```

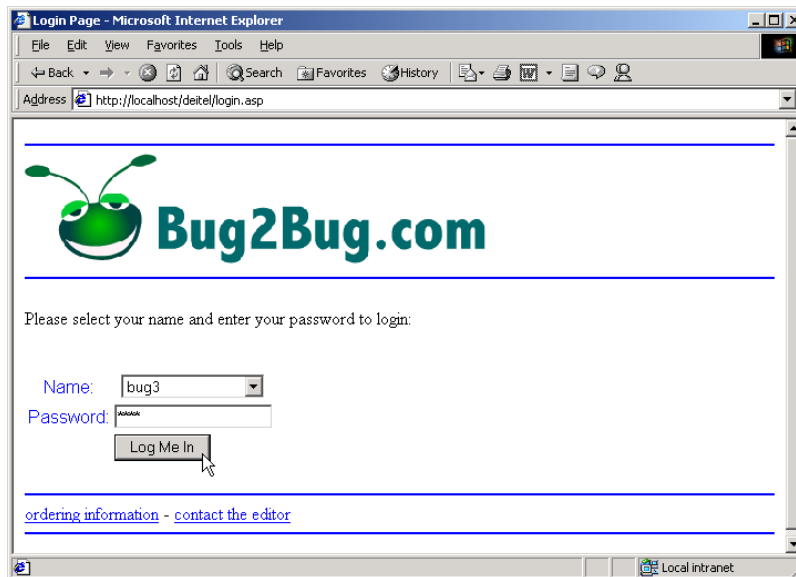
140 ' pull user names from the record set to populate the
141 ' dropdown list
142 While Not loginData.EOF
143     ' create this record's dropdown entry
144     <option value = "<% =loginData( "loginID" ) %>">
145     <% =loginData( "loginID" ) %></option>
146     Call loginData.MoveNext()
147 Wend
148 End Sub
149 <%>

```



Program Output

© 2001 Prentice Hall, Inc.
All rights reserved.



Program Output

© 2001 Prentice Hall, Inc.
All rights reserved.



```
1  <% @LANGUAGE = VBScript %>
2
3  <% ' Fig. 25.24 : submitlogin.asp
4  ' ASP document to check user's username and password
5  Option Explicit
6
7  ' test if a user name and a password were
8  ' entered. If not, transfer back to the login.asp
9  If Request( "password" ) = "" Or
10     Request( "loginID" ) = "noSelection" Then
11     Session( "loginFailure" ) = True
12     Call Server.Transfer( "login.asp" )
13 End If
14
15 Dim connect
16
17 ' create the connection to the database
18 Session( "loginData" ) = "SELECT * FROM Users WHERE loginID = " &
19     Request( "loginID" ) & ";"
20
21 Call Server.Execute( "database.asp" )
22
23 ' Sets session variable loginFailure value to False
24
25 ' Sets cookie's expiration date to current date plus 3 days
26
27 ' password is OK, adjust loginID
28 Session( "loginFailure" ) = False
29
30 ' write a cookie to recognize them the next time they
31 ' go to login.asp
32 Response.Cookies( "loginID" ) = Request( "loginID" )
33
34 ' give it three days to expire
35 Response.Cookies( "loginID" ).Expires = Date() + 3
```

Lines 9-13 check whether **password** field is empty or id **loginID** field contains default value

If so, variable **loginFailure** set to **true**, client redirected back to **login.asp** **WHERE** specifies a

checks password against the records in recordset **loginData** to session variable **loginData** (contains records matching query.)

Sets session variable **loginFailure** value to **False**

If true, line writes form's **loginID** value as cookie named **loginID**

Submitlogin.asp

Takes values passed by **login.asp** and checks values against **Users** table in database

If match is found, user is redirected to **instantpage.asp**. If not, user is redirected to **login.asp**.

© 2001 Prentice Hall, Inc.
All rights reserved.

Otherwise `loginFailure` set to `True`
and client is redirected to `login.asp`

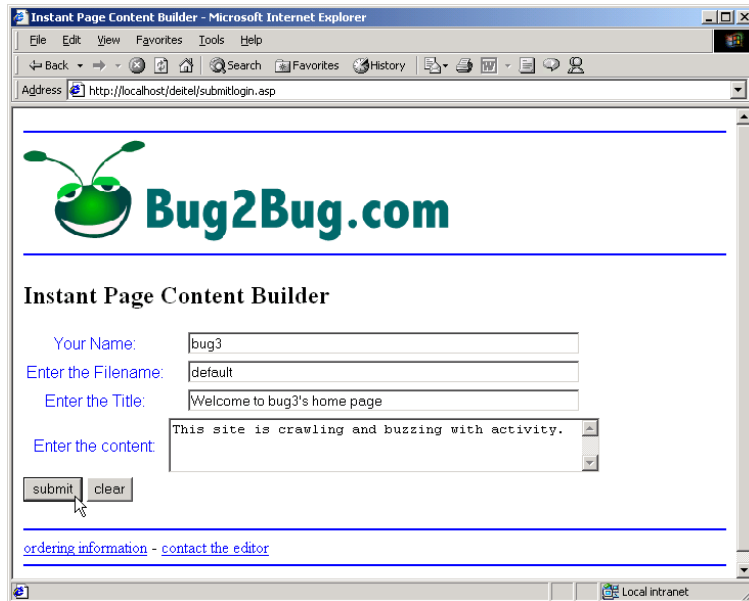
er method `Transfer` to
nt to `instantpage.asp`

```

39 Else
40     Session( "loginFailure" ) = True
41     Call Server.Transfer( "login.asp" )
42 End If
43 %>

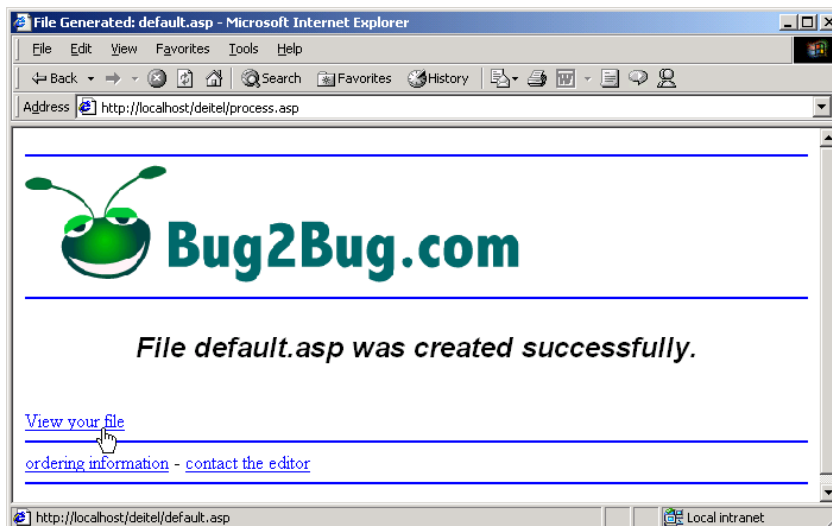
```

Program Output



© 2001 Prentice Hall, Inc.
All rights reserved.

Program Output



© 2001 Prentice Hall, Inc.
All rights reserved.

25.8 Accessing a Database from an Active Server Page

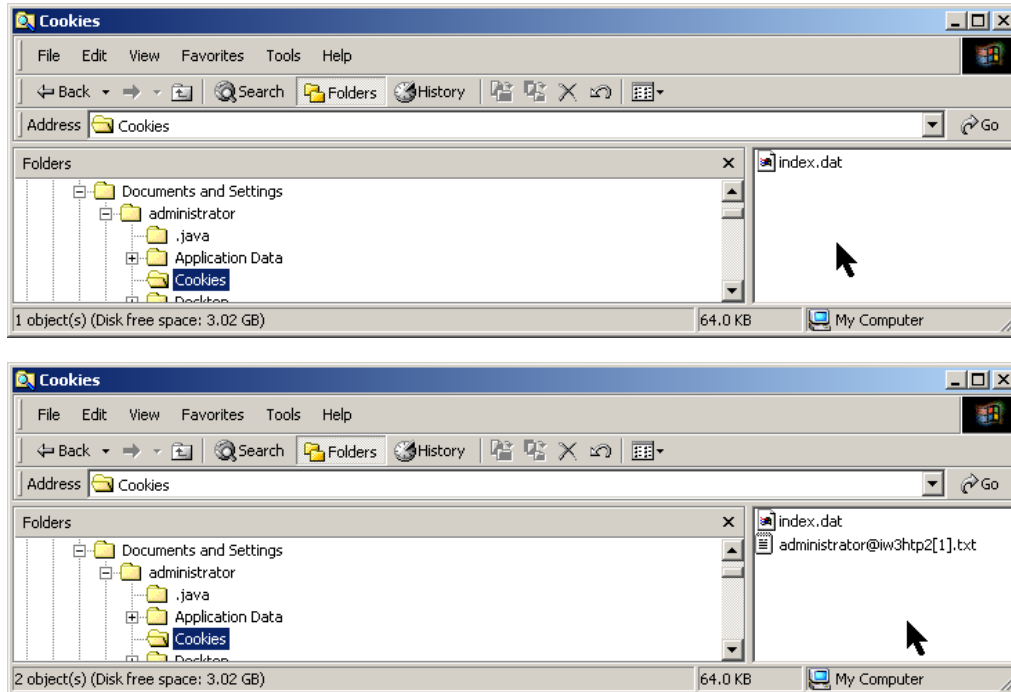


Fig. 25.25 Cookies folder before and after cookie creation.



25.8 Accessing a Database from an Active Server Page

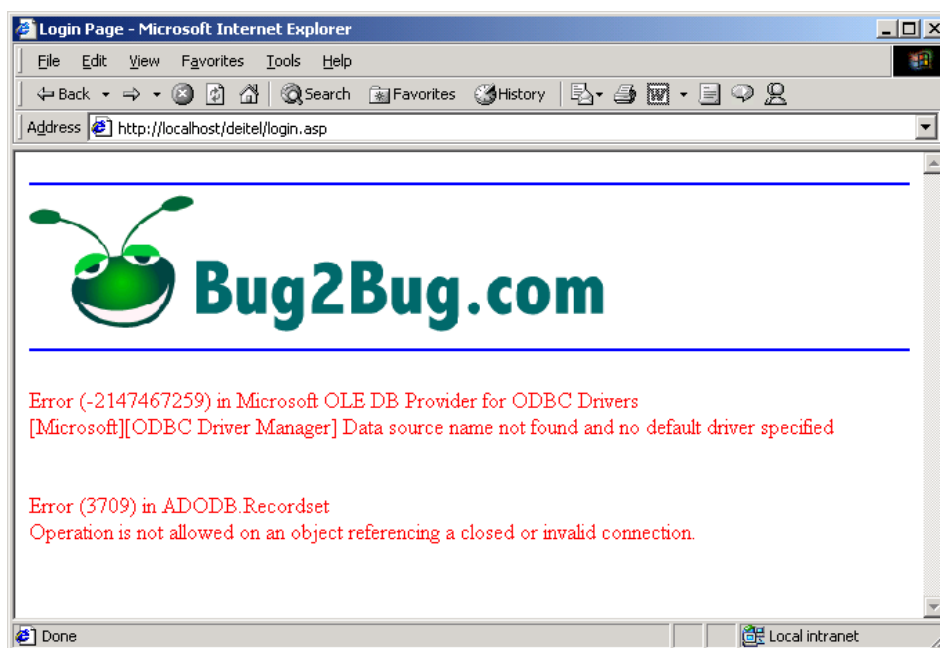


Fig. 25.26 Error messages sent to `login.asp` by `database.asp`.



25.9 Server-Side ActiveX Components

- ActiveX controls on the server with no GUI
 - Make features available in ASP
 - AdRotator ActiveX component
 - Rotates advertisements on a Web page
 - Randomly displays one of several advertisements
 - Minimizes space on a Web page committed to advertisements
 - Client does not have to support ActiveX technologies (on the server)
 - PageCounter ActiveX component
 - Page “hit” counter



25.9 Server-Side ActiveX Components

Component Name	Description
MSWC.BrowserType	ActiveX component for gathering information about the client's browser (e.g., type, version, etc.).
MSWC.AdRotator	ActiveX component for rotating advertisements on a Web page.
MSWC.NextLink	ActiveX component for linking Web pages together.
MSWC.ContentRotator	ActiveX component for rotating HTML content on a Web page.
MSWC.PageCounter	ActiveX component for storing the number of times a Web page has been requested.
MSWC.Counters	ActiveX component that provide general-purpose persistent counters.
MSWC.MyInfo	ActiveX component that provides information about a Web site (e.g., owner name, owner address, etc.).
Scripting.FileSystemObject	ActiveX component that provides an object library for accessing files on the server or on the server's network.
ActiveX Data Objects (ADO) Data Access Components	ActiveX components that provide an object library for accessing databases.

Fig. 25.27 Some server-side ActiveX components.





Component.asp

Uses AdRotator ActiveX component to rotate one of five flag images. When user clicks flag image, country's corresponding CIA Fact Book Web page displays.

```

1  <% @LANGUAGE = VBScript %>
2
3  <%
4      ' Fig. 25.28 : component.asp
5      ' Demonstrating Server-side ActiveX Components
6      Option Explicit
7  %>
8
9  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
10     "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
11
12  <html xmlns = "http://www.w3.org/1999/xhtml">
13
14      <head>
15          <title>ActiveX Component Example</title>
16      </head>
17
18      <body>
19
20          ' Creates AdRotator component instance,
21          ' Sends advertisement as HTML to client. Method
22          ' GetAdvertisement called using reference rotator.
23          ' Retrieves advertisements from config.txt
24
25      <%
26          Dim rotator, browser, information, counter
27
28          ' create an AdRotator object
29          Set rotator = Server.CreateObject( "MSWC.AdRotator" )
30
31          ' use config.txt to send an advertisement to the client
32          Call Response.Write( _
33              rotator.GetAdvertisement( "config.txt" ) )
34      %>

```

© 2001 Prentice Hall, Inc.
All rights reserved.



Component.asp

```

35      ' create a BrowserType object
36      Set browser = Server.CreateObject( "MSWC.BrowserType" )
37
38      If browser.VBScript = True Then
39      %>
40          <script language = "VBScript">
41              Call MsgBox( "Client browser supports VBScript!" )
42          </script>
43      <%
44      End If
45
46      If browser.JavaScript = True Then
47      %>
48          <script language = "JavaScript">
49              alert( "Client browser supports JavaScript!" )
50          </script>
51      <%
52      End If
53
54      ' BrowserType object's Browser
55      ' Version and MinorVer properties
56      ' can obtain similar client information
57
58      information = "<p>Your browser is " & Request.ServerVariables( "HTTP_USER_AGENT" ) & "
59      information = information & "<br />Browser: " & browser.Version & "
60      information = information & "<br />Cookies are " & browser.MinorVer
61
62      If browser.Cookies Then
63          information = information & "enabled<br />"
64      Else
65          information = information & "disabled<br />"
66      End If
67
68      Call Response.Write( information )

```

© 2001 Prentice Hall, Inc.
All rights reserved.



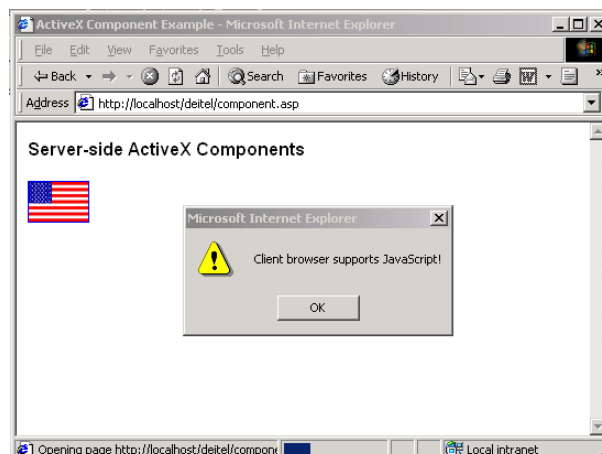
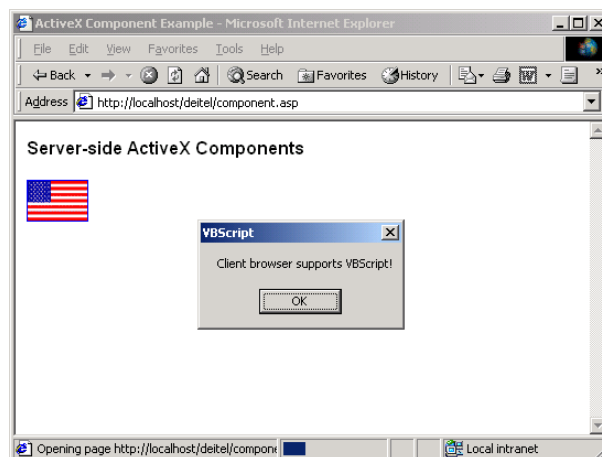
```

69 ' create Page Counter Object
70 Set counter = Server.CreateObject( "MSWC.PageCounter" )
71 Call counter.PageHit() ' page has been "hit"
72 %>
73 </p>
74
75 <p style = "color: blue" >
76     This page has been
77     times!</p>
78 </body>
79 </html>

```

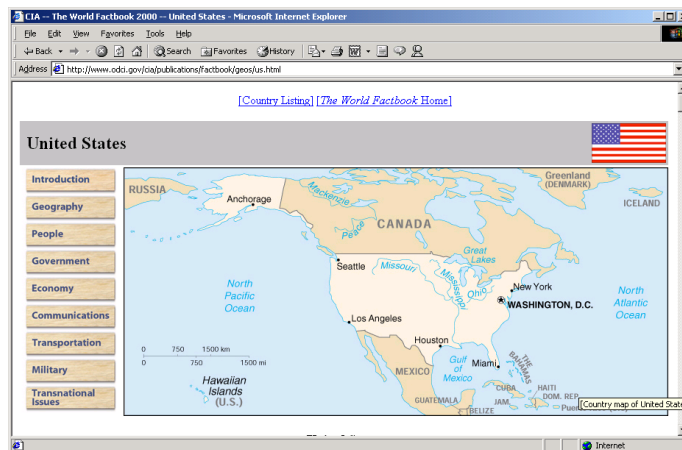
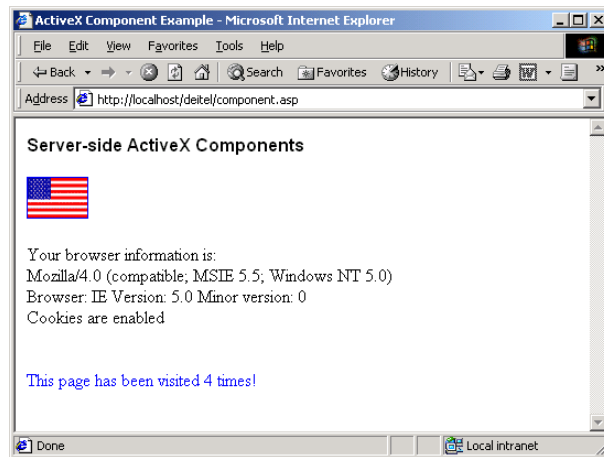
Returns number of "hits"

Increments number of "hits" by one





Program Output



© 2001 Prentice Hall, Inc.
All rights reserved.



Config.txt

Describes the
advertisements

```
1 REDIRECT redirect.asp
2 width 54
3 height 36
4 border 1
5 *
6 /images/us.gif
7 http://www.odci.gov/cia/publications/factbook/geos/us.html
8 United States Information
9 20
10 /images/france.gif
11 http://www.odci.gov/cia/publications/factbook/geos/fr.html
12 France Information
13 20
14 /images/germany.gif
15 http://www.odci.gov/cia/publications/factbook/geos/gm.html
16 Germany Information
17 20
18 /images/italy.gif
19 http://www.odci.gov/cia/publications/factbook/geos/it.html
20 Italy Information
21 20
22 /images/spain.gif
23 http://www.odci.gov/cia/publications/factbook/geos/sp.html
24 Spain Information
25 20
```

Header contains
Image URL, destination URL,
alt value, image display ratio
Advertisements
from advertisements

© 2001 Prentice Hall, Inc.
All rights reserved.

**Redirect.asp**

Redirects user to country's CIA page when ad is clicked

```
1  <% @LANGUAGE = VBScript %>
2
3  <%
4      ' Fig. 25.30 : redirect.asp
5      ' Redirection Page for AdRotator Component
6      Option Explicit
7
8      Call Response.Redirect( Request( "url" ) )
9  %>
```