

Browser handling power of PHP

The super global array `$_SERVER` contains great deal of information about what's going on with the web application. For example `$_SERVER['PHP_SELF']` is used to get the name of the current script, `$_SERVER['REQUEST_METHOD']` is holding the request method used ('get', or 'post') etc. see the following program.

Phpserver.php

```
<html>
<head>
<title> browser handling of PHP</title>
</head>
<body>
<h1> browser handling of PHP</h1>
<?php
echo "<br>You have accessed ", $_SERVER["PHP_SELF"];
echo "<br>on port ", $_SERVER["SERVER_PORT"];
?>
</body>
</html>
```

output



Using HTTP headers

In addition to the server variables there are HTTP headers we can use with `$_SERVER` .

These headers are sent by the browser and contain information about the browser.

The following table lists the most important elements that can go inside \$_SERVER:

Element/Code	Description
<code>\$_SERVER['PHP_SELF']</code>	Returns the filename of the currently executing script
<code>\$_SERVER['GATEWAY_INTERFACE']</code>	Returns the version of the Common Gateway Interface (CGI) the server is using
<code>\$_SERVER['SERVER_ADDR']</code>	Returns the IP address of the host server
<code>\$_SERVER['SERVER_NAME']</code>	Returns the name of the host server (such as www.w3schools.com)
<code>\$_SERVER['SERVER_SOFTWARE']</code>	Returns the server identification string (such as Apache/2.2.24)
<code>\$_SERVER['SERVER_PROTOCOL']</code>	Returns the name and revision of the information protocol (such as HTTP/1.1)
<code>\$_SERVER['REQUEST_METHOD']</code>	Returns the request method used to access the page (such as POST)
<code>\$_SERVER['REQUEST_TIME']</code>	Returns the timestamp of the start of the request (such as 1377687496)
<code>\$_SERVER['QUERY_STRING']</code>	Returns the query string if the page is accessed via a query string
<code>\$_SERVER['HTTP_ACCEPT']</code>	Returns the Accept header from the current request
<code>\$_SERVER['HTTP_ACCEPT_CHARSET']</code>	Returns the Accept_Charset header from the current request (such as utf-8,ISO-8859-1)
<code>\$_SERVER['HTTP_HOST']</code>	Returns the Host header from the current request
<code>\$_SERVER['HTTP_REFERER']</code>	Returns the complete URL of the current page (not reliable because not all user-agents support it)
<code>\$_SERVER['HTTPS']</code>	Is the script queried through a secure HTTP protocol
<code>\$_SERVER['REMOTE_ADDR']</code>	Returns the IP address from where the user is viewing the current page
<code>\$_SERVER['REMOTE_HOST']</code>	Returns the Host name from where the user is viewing the current page
<code>\$_SERVER['REMOTE_PORT']</code>	Returns the port being used on the user's machine to communicate with the web server
<code>\$_SERVER['SCRIPT_FILENAME']</code>	Returns the absolute pathname of the currently executing script
<code>\$_SERVER['SERVER_ADMIN']</code>	Returns the value given to the SERVER_ADMIN directive in the web server configuration file (if your script runs on a virtual host, it will be the value defined for that virtual host) (such as someone@w3schools.com)
<code>\$_SERVER['SERVER_PORT']</code>	Returns the port on the server machine being used

<code>\$_SERVER['SERVER_SIGNATURE']</code>	by the web server for communication (such as 80) Returns the server version and virtual host name which are added to server-generated pages
<code>\$_SERVER['PATH_TRANSLATED']</code>	Returns the file system based path to the current script
<code>\$_SERVER['SCRIPT_NAME']</code>	Returns the path of the current script
<code>\$_SERVER['SCRIPT_URI']</code>	Returns the URI of the current page

```
<html>
<head>
<title> Getting browser type etc </title>
</head>
<body>
<h1> Getting browser type etc</h1>
<?php
echo "You are using", $_SERVER["HTTP_USER_AGENT"];
echo "<br>on port ", $_SERVER["SERVER_PORT"];
?>
</body>
</html>
```



Redirecting browsers with HTTP headers

We can create our own HTTP headers and send them back to the browser.

Suppose we create two buttons and if one button is pressed it should go to hello.html and if another button is pressed it should go to welcome.html. For this we write phpredirect.html file like below:

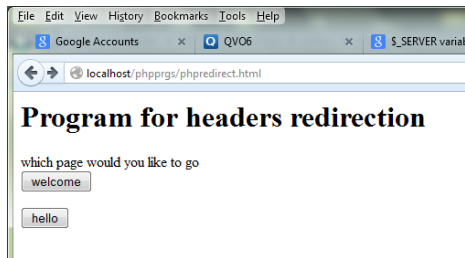
Phpredirect.php

```
<html>
<head>
<title> Program for headers redirection</title>
```

```
</head>
<body>
<h1> Program for headers redirection</h1>
which page would you like to go
<form name="form1" action="phpredirect.php" method="post">
<input type="submit" name="button" value="welcome">
</form>
<form name="form2" action="phpredirect.php" method="post">
<input type="submit" name="button" value="hello">
</form>

</body>
</html>
```

output



The coding for hello.html is

```
<html>
<head>
<title> Program for headers redirection</title>
</head>
<body>
<h1> Program for headers redirection</h1>
Hello from this application
</body>
</html>
```

The coding for welcome.html is

```
<html>
<head>
<title> Program for headers redirection</title>
</head>
<body>
<h1> Program for headers redirection</h1>
Welcome from this application
</body>
</html>
```

Phredirect.php

```
<?php
$redirect="Location:" . $_REQUEST['button'] . ".html";
echo header($redirect);
?>
```

When the above phredirect.html is run the php part of program will take care of redirecting the respective files based on which button the user has pressed.

The below program will display all the server variables values

```
<?php
foreach($_SERVER as $key_name => $key_value) {
print $key_name . " = " . $key_value . "<br>";
}
?>
```

output

```
MIBDIRS = D:/xampp/php/extras/mibs
MYSQL_HOME = \xampp\mysql\bin
OPENSSL_CONF = D:/xampp/apache/bin/openssl.cnf
PHP_PEAR_SYSCONF_DIR = \xampp\php
PHPRC = \xampp\php
TMP = \xampp\tmp
HTTP_HOST = localhost
HTTP_USER_AGENT = Mozilla/5.0 (Windows NT 6.1; rv:29.0) Gecko/20100101 Firefox/29.0
HTTP_ACCEPT = text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
HTTP_ACCEPT_LANGUAGE = en-US,en;q=0.5
HTTP_ACCEPT_ENCODING = gzip, deflate
HTTP_CONNECTION = keep-alive
PATH = C:\Program Files\PC Connectivity Solution\;C:\Program Files\Common Files\Microsoft Shared\Windows Live;C:\Python27\Lib\site-packages\PyQt4\bin;C:\Program Files\MiKTeX 2.9\miktex\bin;C:\Python27\Lib\site-packages\PyQt4\bin;C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShell\v1.0\;C:\Program Files\Microsoft SQL Server\100\Tools\Binn\;C:\Program Files\Microsoft SQL Server\100\DTS\Binn\;E:\matlab2010\runtime\win32;E:\matlab2010\bin;C:\Program Files\Microsoft SQL Server\100\Tools\Binn\Redist\;C:\Users\WELCOME\AppData\Roaming\npm;C:\Program Files\nodejs\;C:\Program Files\Microsoft SQL Server\110\Tools\Binn\;C:\Program Files\Microsoft\Web Platform Installer\;F:\TCWIN45\BIN;C:\Program Files\Common
```

Files\Ahead\Lib\;C:\Program Files\Common Files\Ahead\Lib\;C:\Program Files\iis
express\PHP\v5.3;C:\Program Files\Windows Live\Shared;

SystemRoot = C:\Windows

COMSPEC = C:\Windows\system32\cmd.exe

PATHEXT = .COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC

WINDIR = C:\Windows

SERVER_SIGNATURE =

Apache/2.4.2 (Win32) OpenSSL/1.0.1c PHP/5.4.4 Server at localhost Port 80

SERVER_SOFTWARE = Apache/2.4.2 (Win32) OpenSSL/1.0.1c PHP/5.4.4

SERVER_NAME = localhost

SERVER_ADDR = 127.0.0.1

SERVER_PORT = 80

REMOTE_ADDR = 127.0.0.1

DOCUMENT_ROOT = D:/xampp/htdocs

REQUEST_SCHEME = http

CONTEXT_PREFIX =

CONTEXT_DOCUMENT_ROOT = D:/xampp/htdocs

SERVER_ADMIN = postmaster@localhost

SCRIPT_FILENAME = D:/xampp/htdocs/phpprgs/phpserver_var.php

REMOTE_PORT = 51040

GATEWAY_INTERFACE = CGI/1.1

SERVER_PROTOCOL = HTTP/1.1

REQUEST_METHOD = GET

QUERY_STRING =

REQUEST_URI = /phpprgs/phpserver_var.php

SCRIPT_NAME = /phpprgs/phpserver_var.php

PHP_SELF = /phpprgs/phpserver_var.php

PHP_AUTH_USER = admin

PHP_AUTH_PW = admin_pass

REQUEST_TIME_FLOAT = 1396097720.281

REQUEST_TIME = 1396097720