

PHP: Hypertext Preprocessor Sessions

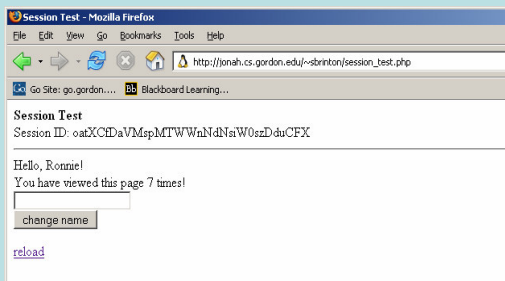
Gordon College
CS374 Internet Programming

Core PHP Programming, Third Edition By Leon Atkinson
PHP and MySQL Web Development By Luke Welling and Laura Thomson
Web Applications By Craig Knuckles and David Yuen

Sessions

- uses the `_SESSION` array (an associative array)
- PHP saves this array in files on the server. When the user requests another page, PHP restores the array.
- session identifier is a long series of numbers and letters sent to the user as a cookie

Session Example



Sessions Example

```
<?php
session_start();

if(isset($_REQUEST['inputName'])) $_SESSION['Name'] =
    $_REQUEST['inputName'];

if(isset($_SESSION['Count'])) $_SESSION['Count']++;
else $_SESSION['Count'] = 1;
?>
```

Sessions Example

```
<html>
<head>
<title>Session Test</title>
</head>
<body>
<b>Session Test</b><br>
Session ID: <?php print $_SESSION['sid'] ?> <br>
<hr>
<?php
if(isset($_SESSION['Name'])) print("Hello,
".stripslashes($_SESSION['Name'])."!<br>\n"); ?>
You have viewed this page
<?php print $_SESSION['Count']?> times!<br>
```

string **stripslashes** (string str)

Returns a string with backslashes stripped off. (\' becomes \' and so on.) Double backslashes (\\) are made into a single backslash (\)

An example use of **stripslashes()** is when the PHP directive `magic_quotes_gpc` is *on* (it's on by default), and you aren't inserting this data into a place (such as a database) that requires escaping. For example, if you're simply outputting data straight from an HTML form.

Sessions Example

```
<form action=" <?php print $_SERVER['PHP_SELF'] ?> "
  method="post">
  <input type="text" name="inputName" value=""><br>
  <input type="submit" value="change name"><br>
</form>

<a href=" <?php print $_SERVER['PHP_SELF'] ?>
">reload</a><br>

</body>
</html>
```

Sessions

- PHP sends session identifiers with cookies
- PHP can detect when a browser does not accept cookies - in this situation it modifies all forms and links to include the session identifier.

Redo - Session Example

```
<?php
include("session_lib.php");

//start session
start_session();

//Set variable based on form input
if(isset($_REQUEST['inputName']))
{
    $stateHash['Name'] = $_REQUEST['inputName'];
}
}
```

Redo Session Example

```
//Increment counter with each page load
if(isset($stateHash['Count']))
{
    $stateHash['Count']++;
}
else
{
    //start with count of 1
    $stateHash['Count'] = 1;
}
?>
```

Sessions Example

```
<html>
<head>
<title>Session Test</title>
</head>
<body>
<b>Session Test</b><br>
  Session ID: <?php print $stateHash['sid'] ?> <br>
  <hr>
  <?php
  if(isset($stateHash['Name'])) print("Hello,
  ".stripslashes($stateHash['Name'])."!<br>\n");
  ?>
  You have viewed this page
  <?php print $stateHash['Count']?> times!<br>
```

Sessions Example

```
<form action=" <?php print $_SERVER['PHP_SELF'] ?> "
  method="post">
  <input type="text" name="inputName" value=""><br>
  <input type="submit" value="change name"><br>
</form>

<a href=" <?php print $_SERVER['PHP_SELF'] ?>
">reload</a><br>

</body>
<?php update_session() ?>
</html>
```

Session Library

- Basic Configuration Variables

```
<?php
srand(time());
$stateDir = "states";           # directory for state files
$cache_limit = 300;            # max number of files in cache
$file_life_span = 1;          # kill files older than one day
```

```
$stateDir  folder location of state files
$cache_limit  number of files in state cache
$file_life_span  state files are cleared out after this
many days.
```

Session Library

```
function start_session()
{
    global $stateHash;
    if(isset($_COOKIE['sid']))
    {
        read_state($_COOKIE['sid']);
    } else
    {
        $sessionID = get_long_id();
        $stateHash['sid'] = $sessionID;
        setcookie("sid", "$sessionID");
    }
}
```

Cookie

Set cookie using the setcookie function:

```
bool setcookie ( string name [, string value [, int expire
[, string path [, string domain [, bool secure]]]] )
```

Example:

```
setcookie('mycookie', 'value');
```

Cookie

Access cookie using:

```
$_COOKIE['mycookie']
```

```
$HTTP_COOKIE_VARS['mycookie'];
```

Cookie

Delete cookie by:

Calling setcookie() with same cookie name and an expire time in the past.

```
setcookie('mycookie', 'value', time() - 3600);
```

Cookie

More Examples

```
<?php
$value = 'something from somewhere';

setcookie("TestCookie", $value);
setcookie("TestCookie", $value, time()+3600);
setcookie("TestCookie", $value, time()+3600,
"/~rasmus/", ".example.com", 1);

?>
```

Session Library

```
function start_session()
{
    global $stateHash;
    if(isset($_COOKIE['sid']))
    {
        read_state($_COOKIE['sid']);
    } else
    {
        $sessionID = get_long_id();
        $stateHash['sid'] = $sessionID;
        setcookie("sid", "$sessionID");
    }
}
```

Session Library

```
function get_long_id (){
    global $stateDir; global $cache_limit;
    global $file_life_span; global $stateHash;

    get_files($entryName, $stateDir);
    if(count($entryName) >= $cache_limit) {
        foreach($entryName as $file){
            if(abs(date("z", filetime($stateDir.$file))-date("z")) >
                $file_life_span)
                unlink($stateDir.$file); #delete the file
        }
        get_files($entryName, $stateDir);
    }

    if(count($entryName) >= $cache_limit)
        errorPage("Site busy. Please try again later.");
    return generate_random_string(32);
}
```

Session Library

```
function get_files(&$entryName, $dir)
{
    $myDirectory = opendir($dir);
    // get each entry
    while(FALSE != ($entryName[] = readdir($myDirectory))){}
    array_pop($entryName); //take off the last item
    closedir($myDirectory);
    // rip out everything but the files
    $i=0;
    while($i<count($entryName)
    {
        if (!is_file($entryName[$i]))
        {
            array_splice($entryName, $i, 1);
        } else $i++;
    }
}
```

Session Library

```
function get_long_id (){
    global $stateDir; global $cache_limit;
    global $file_life_span; global $stateHash;

    get_files($entryName, $stateDir);
    if(count($entryName) >= $cache_limit) {
        foreach($entryName as $file){
            if(abs(date("z", filetime($stateDir.$file))-date("z")) >
                $file_life_span)
                unlink($stateDir.$file); #delete the file
        }
        get_files($entryName, $stateDir);
    }

    if(count($entryName) >= $cache_limit)
        errorPage("Site busy. Please try again later.");
    return generate_random_string(32);
}
```

Session Library

```
function generate_random_string($length) {
    $result = "";
    $chars=range('0','9');
    array_splice($chars, 0, 0, range('a','z'));
    array_splice($chars, 0, 0, range('A','Z'));

    for($i = 1 ; $i <= $length ; $i++) {
        $which=rand(1,62);
        $result = $result . $chars[$which];
    }
    return $result;
}
```

Encoding and Decoding

• string urlencode (string str)

Returns a string in which all non-alphanumeric characters except `_-.` have been replaced with a percent (%) sign followed by two hex digits and spaces encoded as plus (+) signs

```
<?php
echo '<a href="mycgi?foo=
        ',urlencode($userinput), ">';
?>
```

Encoding and Decoding

- string **urlencode** (string str)

Decodes any %## encoding in the given string.

```
<?php
$a = explode('&', $QUERY_STRING);
$i = 0;
while ($i < count($a)) {
    $b = split('=', $a[$i]);
    echo "Value for parameter ";
    htmlspecialchars(urldecode($b[0]),
        'is ', htmlspecialchars(urldecode($b[1])), "<br />\n";
    $i++;
}
?>
```

Session Library

```
function read_state($fileroot){
    global $stateDir;
    global $stateHash;
    $filename = "$fileroot.state";

    if(!($inFile = fopen($stateDir.$filename, "r"))
        errorPage("Error reading state file.");
    while(!feof($inFile))
    {
        $myLine = fgets($inFile, 255);
        $temp = split("=", $myLine, 2);
        if (urldecode($temp[0])!="")
            $stateHash[urldecode($temp[0])]
                =rtrim(urldecode($temp[1]));
    }
    fclose($inFile);
}
```

Session Library

```
function write_state ($fileroot){
    global $stateDir;
    global $stateHash;

    $filename = "$fileroot.state";
    if(!($outFile = fopen($stateDir.$filename, "w"))
        errorPage("Error opening state file to write.");

    foreach($stateHash as $key=>$value)
    {
        $value = urlencode($stateHash[$key]);
        $name = urlencode($key);
        fputs($outFile, "$name=$value\n");
    }

    fclose($outFile);
}
```

Session Library

```
function update_session()
{
    global $stateHash;
    write_state($stateHash['sid']);
}
```