

Web Techniques
T.Y.B.Sc.(Computer Science)
Semester – IV

Mr. Abhijeet D. Mankar
Department of Computer Science
Tuljaram Chaturchand College, Baramati.

PHP Global Variables - Superglobals

- Always accessible, regardless of scope
 - Accessible from any function, class or file without having to do anything special
1. `$GLOBALS`
 2. `$_SERVER`
 3. `$_REQUEST`
 4. `$_POST`
 5. `$_GET`
 6. `$_FILES`
 7. `$_ENV`
 8. `$_COOKIE`
 9. `$_SESSION`

Server Information

ELEMENT	DESCRIPTION
\$_SERVER['PHP_SELF']	Returns the filename of the currently executing script
\$_SERVER['GATEWAY_INTERFACE']	Returns the version of the Common Gateway Interface (CGI) the server is using
\$_SERVER['SERVER_ADDR']	Returns the IP address of the host server
\$_SERVER['SERVER_NAME']	Returns the name of the host server (such as www.w3schools.com)
\$_SERVER['SERVER_SOFTWARE']	Returns the server identification string (such as Apache/2.2.24)

Server Information(contd.)

ELEMENT	DESCRIPTION
\$_SERVER['SERVER_PROTOCOL']	Returns the name and revision of the information protocol (such as HTTP/1.1)
\$_SERVER['REQUEST_METHOD']	Returns the request method used to access the page (such as POST)
\$_SERVER['REQUEST_TIME']	Returns the timestamp of the start of the request (such as 1377687496)
\$_SERVER['QUERY_STRING']	Returns the query string if the page is accessed via a query string
\$_SERVER['HTTP_ACCEPT']	Returns the Accept header from the current request
\$_SERVER['HTTP_ACCEPT_CHARSET']	Returns the Accept_Charset header from the current request (such as utf-8,ISO-8859-1)
\$_SERVER['HTTP_HOST']	Returns the Host header from the current request
\$_SERVER['HTTP_REFERER']	Returns the complete URL of the page from which the current page was called
\$_SERVER['HTTPS']	Is the script queried through a secure HTTP protocol

Server Information(contd.)

ELEMENT	DESCRIPTION
\$_SERVER['REMOTE_ADDR']	Returns the IP address from where the user is viewing the current page
\$_SERVER['REMOTE_HOST']	Returns the Host name from where the user is viewing the current page
\$_SERVER['REMOTE_PORT']	Returns the port being used on the user's machine to communicate with the web server
\$_SERVER['SCRIPT_FILENAME']	Returns the absolute pathname of the currently executing script
\$_SERVER['SERVER_ADMIN']	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)
\$_SERVER['SERVER_PORT']	Returns the port on the server machine being used by the web server for communication (such as 80)
\$_SERVER['SERVER_SIGNATURE']	Returns the server version and virtual host name which are added to server-generated pages
\$_SERVER['PATH_TRANSLATED']	Returns the file system based path to the current script
\$_SERVER['SCRIPT_NAME']	Returns the path of the current script
\$_SERVER['SCRIPT_URI']	Returns the URI of the current page

GET Vs POST

GET

- ❖ GET requests can be cached
- ❖ GET requests remain in the browser history
- ❖ GET requests can be bookmarked
- ❖ GET requests should never be used when dealing with sensitive data
- ❖ GET requests have length restrictions
- ❖ GET requests is only used to request data (not modify)

POST

- ❖ POST requests are never cached
- ❖ POST requests do not remain in the browser history
- ❖ POST requests cannot be bookmarked
- ❖ POST requests have no restrictions on data length

Maintaining States

HTTP is a stateless protocol.

To maintain state we use

- Sessions
- Cookies

Maintaining State using Session

Program which accepts name and password in first form, on the second form accepts address and contact number. If the address and contact number is not entered in the specified time interval, then on the third form it displays session time out message otherwise displays the details of the user. (In this program time interval is 60 seconds)

File Name : sess_1.html

```
<html>
<head><title>Maintaining State - Using Session</title></head>
<body>
<form action="sess_2.php" method="POST">
<pre>
    Enter Name    <input type="text" name="uname">
    Enter Password <input type="password" name="passw">
    <input type="submit" value="Submit"> <input type="reset" value="Clear">
</pre>
</form>
</body>
</html>
```


Maintaining State using Session(contd.)

File Name : sess_2.php

```
<?php
    session_start();
    $_SESSION['un']=$_POST['uname'];
    $_SESSION['pw']=$_POST['passw'];
    $_SESSION['start']=time();
?>
<html>
<head><title>Maintaining State - Using Session</title></head>
<body>
<form action="sess_3.php" method="POST">
<pre>
    Enter Address          <input type="text" name="taddr">
    Enter Contact No.    <input type="text" name="tcontact">
    <input type="submit" value="Submit">
    <input type="reset" value="Clear">
</pre>
</form>
</body>
</html>
```

Maintaining State using Session(contd.)

File Name : sess_3.php

```
<?php
    session_start();
    $_SESSION['end']=time();
    if($_SESSION['end']-$_SESSION['start']>60)
    {
        echo "Session timed out";
    }
    else
    {
        echo $_SESSION['un']."<br>";
        echo $_SESSION['pw']."<br>";
        echo $_POST['taddr']."<br>";
        echo $_POST['tcontact']."<br>";
    }
    session_unset();
    session_destroy();
?>
```

COOKIES

Cookies are text files stored on the client computer.

Steps

- ❖ Server script sends a set of cookies to the browser. For example user name, contact number etc.
- ❖ Browser stores this information on local machine for future.
- ❖ When browser sends any request next time to web server then it sends those cookies information to the server and server uses that information to identify the user.

Maintaining State using Cookies

Program which accepts name, password and font colour in first form, on the second form accepts address and contact number and city. On the third form it displays all the details of the user in the font colour (which is specified on the first form).

File Name : cookie_1.html

```
<html>
<head><title>Maintaining State-using Cookies</title></head>
<body>
<form action="cookie_2.php" method="POST">
<pre>
    User Name  <input type="text" name="uname">
    Password   <input type="text" name="passw">
    Font Colour <input type="text" name="fnt">
    <input type="submit" value="Click"> <input type="reset" value="Reset">
</pre>
</form>
</body>
</html>
```

Maintaining State using Cookies(contd.)

File Name : cookie_2.php

```
<?php
    setcookie('unm',$_POST['uname']);
    setcookie('pws',$_POST['passw']);
    setcookie('font',$_POST['fnt']);
?>
<html>
<head><title>Other Information</title></head>
<body>
<form action="cookie_3.php" method="POST">
    <pre>
    Enter Address   <input type="text" name="taddr">
    Enter Contact No. <input type="text" name="tcontact">
    Enter City     <input type="text" name="tcity">
    <input type="submit" value="Submit"> <input type="reset" value="Cancel">
    </pre>
</form>
</body>
</html>
```

Maintaining State using Cookies(contd.)

File Name : cookie_3.php

```
<html>
<head><title>User Preferences</title></head>
<body>
  <font color="<?php echo $_COOKIE['font'];?>">
    <?php
      echo $_COOKIE['unm']."<br>";
      echo $_COOKIE['pws']."<br>";
      echo $_POST['taddr']."<br>";
      echo $_POST['tcity']."<br>";
      echo $_POST['tcontact']."<br>";
    ?>
  </font>
</body>
</html>
```

Additional References

How to Start a Session in PHP

<https://www.youtube.com/watch?v=3CS-eQdcMLU>

Session and Cookies in PHP

https://www.youtube.com/watch?v=jort8_4U-88