

Read this first:

While working with the dynamic web application user required to send information to server for either calculation or for permanent storage like registration details need to be stored at server. Sometimes user also needs to navigate from one page to another page and use some information from previous page to current page also. For example in online shopping site all item which are selected by user need to display on check out page. This is works are generally referred as passing information between pages.

PHP Superglobals (Global Variables of PHP)

PHP has several predefined variables which are called "superglobals". As they are available in all scopes throughout a script they are known as superglobals. There is no need to do global \$variable; to access them within functions or methods.

\$GLOBALS	PHP stores all global variables in an array called \$GLOBALS[index]. The index holds the name of the variable
\$_SERVER	\$_SERVER is a PHP super global variable which holds information about headers, paths, and script locations.
\$_REQUEST	PHP \$_REQUEST is used to collect data after submitting an HTML form.
\$_POST	PHP \$_POST is widely used to collect form data after submitting an HTML form with method="post". \$_POST is also widely used to pass variables.
\$_GET	PHP \$_GET can also be used to collect form data after submitting an HTML form with method="get".
\$_FILES	Contains all information of uploaded file.
\$_ENV	An associative array of variables passed to the current script via the environment method.
\$_COOKIE	Containing session variables available to the current script.
\$_SESSION	Array of variables passed to the current script via HTTP Cookies.

isset() and unset()

- **isset()**

isset(name of variable)

isset() method is used to determine if a variable is set and is not NULL. If variable has assigned any value than it will return TRUE either it returns FALSE.

- **unset()**

unset(name of variable)

unset() method is used to unset a given variable. Unset() will destroy the specified variable.

Answer below questions:

Why predefined variables known as “Superglobals”?

Passing Information between Pages

this information is not specific to PHP but is a consequence of the PHP/HTML interaction or of the HTTP protocol itself.

The most important thing to recall about the way the Web works is that the HTTP protocol itself is stateless. Stateless means it doesn’t store any information about the request. Once request is served it will destroyed itself. Each request initiates a discrete process.

In simple way, the HTTP protocol will never know that someone browsing Page2 must have come from Page1. You cannot set the value of a variable on Page1 and expect it to be imported to Page2 by the exigencies of HTML itself. But in any dynamic web application development it is necessary to pass information between the pages. And as HTTP is unable to provide this mechanism PHP provides various ways to pass the information between the pages. Like; GET and POST method, Cookies, Session etc.

GET & POST Method

The GET method passes the information with producing a long string that appears in URL.	The data sent by POST method goes through HTTP header so security depends on HTTP protocol. By using Secure HTTP you can make sure that your information is secure.
-----------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------

The GET method is restricted to send upto 1024 characters only.	The POST method does not have any restriction on data size to be sent.
Not secure as data is visible in URL. We should never use it send password or other sensitive information to the server.	This method is very secure and suitable to pass sensitive information to the server.
GET can't be used to send binary data, like images or word documents, to the server.	The POST method can be used to send ASCII as well as binary data
The data sent by GET method can be accessed using QUERY_STRING environment variable.	The PHP provides \$_POST associative array to access all the sent information using POST method.
The PHP provides \$_GET associative array to access all the sent information using GET method.	
<pre> <?php if(\$_GET["name"] \$_GET["age"]) { echo "Welcome ". \$_GET['name']. "
"; echo "You are ". \$_GET['age']. " years old."; exit(); } ?> <html> <body> <form action = "<?php \$_PHP_SELF </pre>	<pre> <?php if(\$_POST["name"] \$_POST["age"]) { echo "Welcome ". \$_POST['name']. "
"; echo "You are ". \$_POST['age']. " years old."; exit(); } ?> <html> <body> <form action = "<?php \$_PHP_SELF </pre>

<pre> ?>" method = "GET"> Name: <input type = "text" name = "name" /> Age: <input type = "text" name = "age" /> <input type = "submit" /> </form> </body> </html> </pre>	<pre> ?>" method = "POST"> Name: <input type = "text" name = "name" /> Age: <input type = "text" name = "age" /> <input type = "submit" /> </form> </body> </html> </pre>
<p>Above both script has same output except one difference in get method value of variables are visible in URL</p>	

Important Note:

When we need to send the data to server don't forget to specify the **method** and **action** attributes in **form** tag. Action attribute specifies which page should be loaded next after pressing on submit and method attribute defines how data should be sent to server (through URL or HTTP header).

<form action="p1.php" method="POST">

Button or input tag **type** must be **"submit"** which will send all form data to server and later it will be interpreted by PHP engine and send response back to client.

<button type="submit" name="reg">Register Now</button>

<input type="submit" value="Register Now" name="reg"/>

Answer below questions:

Why “get” method is not preferable to send sensitive data to server?

Why should we do not use “get” method to upload file?

Lab Assignment

1. Write a PHP script to display all server information in tabular format.
2. Write a PHP script to create a registration form. On submit display all the information on next page in tabular form.