**EXPERIMENT 08(B)**

**Aim:** Implement PHP GET and POST methods.

**Lab Objective:** Students will be able to:

 Understand and implement PHP GET and POST methods.

**Theory:**

**The GET Method**

The GET method sends the encoded user information appended to the page request. The page and the encoded information are separated by the ? character.

<http://www.test.com/index.htm?name1=value1&name2=value2>

The GET method produces a long string that appears in your server logs, in the browser's Location: box.

* The GET method is restricted to send upto 1024 characters only.
* Never use GET method if you have password or other sensitive information to be sent to the server.
* GET can't be used to send binary data, like images or word documents, to the server.
* The data sent by GET method can be accessed using QUERY\_STRING environment variable.
* The PHP provides $\_GET associative array to access all the sent information using GET method.

**The POST Method**

The POST method transfers information via HTTP headers. The information is encoded as decribed in case of GET method and put into a header called QUERY\_STRING.

* The POST method does not have any restriction on data size to be sent.
* The POST method can be used to send ASCII as well as binary data.
* 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 PHP provides $\_POST associative array to access all the sent information using POST method.

**Source code and output:**

**Get:**

<?php

if( $\_GET["name"] || $\_GET["age"] ) {

echo "Welcome ". $\_GET['name']. "<br />";

echo "You are ". $\_GET['age']. " years old.";

exit();

}

?>

<html>

<body>

<form action = "<?php $\_PHP\_SELF ?>" method = "GET">

Name: <input type = "text" name = "name" />

Age: <input type = "text" name = "age" />

<input type = "submit" />

</form>

</body>

</html>



**Post:**

<?php

if( $\_POST["name"] || $\_POST["age"] ) {

if (preg\_match("/[^A-Za-z'-]/",$\_POST['name'] )) {

die ("invalid name and name should be alpha");

}

echo "Welcome ". $\_POST['name']. "<br />";

echo "You are ". $\_POST['age']. " years old.";

exit();

}

?>

<html>

<body>

<form action = "<?php $\_PHP\_SELF ?>" method = "POST">

Name: <input type = "text" name = "name" />

Age: <input type = "text" name = "age" />

<input type = "submit" />

</form>

</body>

</html>



**Lab Outcome**: Students were able to:

Implement PHP GET and POST methods.

**Conclusion:**

Both the GET and POST methods have their own advantages and disadvantages. It is important to choose the right method for the job. If you are unsure which method to use, it is always best to consult with a more experienced web developer.

**COs attained:**

**POs attained:**

PO 1: ENGINEERING KNOWLEDGE (Apply Knowledge of Mathematics, Science,

engineering fundamentals and an engineering specialization to the solution of complex engineering problems.)

PO 2: PROBLEM ANALYSIS (Identify, formulate, research literature and analyse complex

engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences and engineering sciences.)

PO 3: DESIGN / DEVELOPMENT OF SOLUTIONS (Design solutions for complex engineering problems and design system components or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal and environmental considerations.)

PO 5:MODERN TOOL USAGE (Create, select and apply appropriate techniques, resources and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.)

**PEOs achieved:**

PEO 1: To prepare learners with a strong foundation in the area of Information Technology required solving real life problems arising from software technology. (Knowledge)(CURRICULAR)

PEO 3: To prepare learners to understand the need for lifelong learning with effective written

and oral communication skills and to be able to readily adapt to new software engineering environments. (PRESENTATION AND GROWTH) **.**