

Web Application Development

LAB 6: Introduction to PHP

Main objectives

At the end of this activity, students should be able to:

- Perform arithmetic operations, implement conditional if, else statements, and use for loops in PHP.
- Analyze the differences in data transmission between the GET and POST methods.

Exercise 1:

- 1) Create a folder named siteweb inside the www directory of your WAMP server (e.g., **C:\wamp64\www\TP6**). Inside the TP6 folder, create a file named **index.php** with the following code:

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>PHP page</title>
</head>
<body>
    <h1>Test PHP</h1>
    <p><?php echo "Hello World !";?></p>

    <h3>addition</h3>
    <?php
        $a = 12;
        $b = 13;
        $somme = $a + $b;
        echo "<p> $a + $b = $somme</p>"; ?>

</body>
</html>
```

- 2) Launch the Apache server by starting WAMP. Open your browser and navigate to **<http://localhost/TP6/index.php>** to test your dynamic website.
- 3) Modify the code to include subtraction, multiplication, and division, and then display the results for all four operations.

- 4) Refine the division part of your code by adding an **if, else** statement to handle the division by zero error.

Basic syntax :

```
if (condition)  
{ // Code to execute if condition is true}  
else  
{ // Code to execute if condition is false }
```

- 5) Add a **for** loop to the script that generates and displays the multiplication table for variable \$a (from 1 to 10). Each line should follow the format:

\$a X \$i = result.

Basic syntax :

```
for (initialization; condition; increment)  
{  
  // Code to be executed for each iteration  
}
```

Exercise 2:

- 1) Create a new file named Exo2.php in the directory C:\wamp64\www\TP6. Copy the following code into it:

```
<!DOCTYPE html>  
<html>  
<body>  
    <form method="GET" action="">  
        <input type="text" name="name" placeholder="Enter Name" required>  
        <input type="number" name="age" placeholder="Enter Age" required>  
        <button type="submit" name="submit">Submit</button>  
    </form>  
    <?php  
    if (isset($_GET['submit'])) {  
        $name = htmlspecialchars($_GET['name']);  
        $age = $_GET['age'];  
        echo "<p> Hello $name, you are $age years old.</p>";  
    }  
    ?>  
</body>  
</html>
```

- 2) Test the form in your browser. After clicking "Submit", what do you notice exactly in the URL?
- 3) Modify the <form> tag to use method="POST". Then, update the PHP script to use the \$_POST instead of \$_GET.
- 4) Test the form again. What is the difference between the two methods.