



Departemen Teknologi Informasi - Politeknik Negeri Malang

## Lembar Pekerjaan-05: PHP 2

Kursus: Pemrograman Web / Desain dan Pemrograman Web

Instruktur: Tim Pengajar Desain dan Pemrograman Web

September 2024

Github : <https://github.com/muhammadyusuf081205/jobsheet5pemogramanweb>

### **Topik**

1. Konsep *array* dalam pemrograman PHP
2. Konsep *fungsi* dalam pemrograman PHP

### **Obyektif**

Siswa diharapkan untuk:

1. Memahami konsep *array* dalam pemrograman PHP
2. Memahami konsep fungsi dalam pemrograman PHP

## INTRODUCTION

### **Pengantar Array dan Fungsi**

*Array* dan fungsi adalah konsep dasar dalam pemrograman yang membantu mengatur dan mengoptimalkan kode.

#### ➤ *Array*

*Array*, atau list, adalah salah satu tipe data. *Array* bukanlah tipe data dasar seperti bilangan bulat atau boolean, melainkan tipe data yang terdiri dari kumpulan tipe data lainnya. *Array* memudahkan untuk mengelompokkan data, menghemat penulisan, dan membuat penggunaan variabel lebih efisien. *Array* adalah struktur data yang menyimpan kumpulan elemen, ***biasanya*** dari tipe data yang sama, di bawah satu variabel. *Array* menyederhanakan penanganan kumpulan data yang besar, memungkinkan beberapa nilai disimpan dan diakses dengan mudah menggunakan indeks atau kunci. Dalam PHP, *array* diklasifikasikan menjadi tiga jenis:

- *Indexed Arrays*: *Array* dengan indeks numerik.
- *Associative Arrays*: *Array* dengan kunci (*key*) yang menetapkan nilai (*value*) pada tiap elemennya.
- *Multidimensional Arrays*: *Array* yang berisi *array* lain sebagai elemennya.

#### ➤ *Fungsi*

Fungsi adalah blok kode yang dirancang untuk melakukan tugas tertentu, yang dapat digunakan kembali beberapa kali di seluruh program. Fungsi mengurangi redundansi, membuat kode lebih mudah dibaca, dan meningkatkan efisiensi. Dengan memecah tugas-tugas berulang seperti kueri database atau perhitungan matematis menjadi fungsi, program menjadi modular, dan pemeliharaan menjadi lebih mudah.

### **Praktikum 1. *Indexed Arrays***

***Indexed Arrays*** PHP adalah *array* di mana elemen disimpan dengan indeks numerik, mulai dari 0 secara default. Setiap elemen dalam *array* dikaitkan dengan nomor indeks, yang digunakan untuk mengakses atau mereferensikan elemen tersebut.

```

<?php
// Creating an indexed array
$variable = array("Value0", "Value1", "Value2", "Value3");

// Accessing the elements of the indexed array
echo $variable[0]; // Outputs: Value0
echo $variable[1]; // Outputs: Value1

// Adding a new element
$variable[] = "Value4"; // Adds Value4 to the array
?>

```

Bisa juga ditulis sebagai berikut:

```

<?php
// Creating an indexed array
$variable[0] = "Value0";
$variable[1] = "Value1";
$variable[2] = "Value2";
$variable[3] = "Value3";

// Accessing the elements of the indexed array
echo $variable[0]; // Outputs: Value0
echo $variable[1]; // Outputs: Value1

// Adding a new element
$variable[] = "Value4"; // Adds Value4 to the array
?>

```

Dalam bentuk ini, *array* dibuat secara langsung menetapkan nilai ke indeks tertentu tanpa menggunakan **fungsi array()**. **PHP secara otomatis menetapkan indeks berikutnya** yang tersedia jika elemen baru ditambahkan tanpa menentukan indeks, seperti yang ditunjukkan saat menambahkan "Value4".

Ikuti langkah-langkah ini untuk memahami *array* yang diindeks di PHP:

Langkah	Deskripsi
1	Buat file baru bernama <a href="#"><u>array_1.php</u></a> di dalam direktori <a href="#"><u>JS05_PHP-2</u></a> , lalu ketik kode berikut: <pre> &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt; &lt;/head&gt; &lt;body&gt; &lt;h2&gt;Array Terindeks&lt;/h2&gt; &lt;?php     \$Listdosen=["Elok Nur Hamdana", "Unggul Pamenang", "Bagas Nugraha"];     echo \$Listdosen[2] . "&lt;br&gt;";     echo \$Listdosen[0] . "&lt;br&gt;";     echo \$Listdosen[1] . "&lt;br&gt;"; ?&gt; &lt;/body&gt; &lt;/html&gt; </pre>
2	Simpan file dan jalankan program Kita di browser. Ketik tautan ini ke browser Kita <a href="http://localhost/dasarWeb/JS05_PHP-2/array_1.php"><u>localhost/dasarWeb/JS05_PHP-2/array_1.php</u></a>
3	Amati <i>output</i> yang ditampilkan

4

Untuk menampilkan *array*, selain menggunakan indeks, kita juga bisa menggunakan *loop*. Coba tampilkan *output* program di atas menggunakan *loop*.  
**(Pertanyaan No.1)**

## Praktikum 2. Associative Array

*Array* asosiatif PHP adalah *array* di mana kuncinya bukan numerik melainkan *string*, memungkinkan kita untuk mengaitkan nilai tertentu dengan kunci yang bermakna (sesuai keinginan kita). Hal ini dapat membuat lebih mudah untuk mengakses dan memanipulasi data berdasarkan nama kunci daripada indeks numerik.

Komponen *array* asosiatif terdiri dari pasangan kunci-nilai (**key-value**). Kunci menunjukkan posisi di mana nilai disimpan. PHP menggunakan simbol panah (**=>**) untuk menetapkan nilai ke kunci. Berikut adalah sintaks untuk menulis *array* asosiatif:

```
<?php
// Creating an indexed array
$variable = array(
    'Key0' => 'value0',
    'Key1' => 'value1',
    'Key2' => 'value2',
    'Key3' => 'value3'
);

// Accessing the elements of the indexed array
echo $variable['Key0']; // Outputs: Value0
echo $variable['Key1']; // Outputs: Value1

// Adding a new element
$variable['Key4'] = "Value4"; // Adds Value4 to the array
?>
```

Bisa juga ditulis sebagai berikut:

```
<?php
// Creating an indexed array
$variable['Key0'] = 'value0';
$variable['Key1'] = 'value1';
$variable['Key2'] = 'value2';
$variable['Key3'] = 'value3';

// Accessing the elements of the indexed array
echo $variable['Key0']; // Outputs: Value0
echo $variable['Key1']; // Outputs: Value1

// Adding a new element
$variable['Key4'] = "Value4"; // Adds Value4 to the array
?>
```

Ikuti langkah-langkah ini untuk memahami *array* asosiatif di PHP:

Langkah	Deskripsi
1	Buat file baru bernama <b>array_2.php</b> di dalam direktori JS05_PHP-2, lalu ketik kode berikut: <pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt;     &lt;meta charset="utf-8"&gt;     &lt;meta name="viewport" content="width=device-width, initial-scale=1"&gt;     &lt;title&gt;&lt;/title&gt; &lt;/head&gt; &lt;body&gt;     &lt;?php         \$Dosen = [             'nama' =&gt; 'Elok Nur Hamdana',             'domisili' =&gt; 'Malang',             'jenis_kelamin' =&gt; 'Perempuan' ];          echo "Nama : {\$Dosen ['nama']} &lt;br&gt;";         echo "Domisili : {\$Dosen ['domisili']} &lt;br&gt;";         echo "Jenis Kelamin : {\$Dosen ['jenis_kelamin']} &lt;br&gt;      ?&gt;  &lt;/body&gt; &lt;/html&gt;</pre>
2	Simpan file dan jalankan program Kita di browser. Ketik tautan ini ke browser Kita <b>localhost/dasarWeb/JS05_PHP-2/array_2.php</b>
3	Amati hasil yang ditampilkan. Kemudian tambahkan style tabel pada output tampilan tersebut supaya lebih menarik. <b>(Pertanyaan No.2)</b> *Untuk penggunaan style bebas boleh internal atau eksternal file.

### Praktikum 3. Multidimensional Array

*Array* multidimensi di PHP adalah *array* yang berisi satu atau lebih *array* sebagai elemennya. Ini berarti bahwa setiap elemen dalam *array* dapat berupa *array* itu sendiri, memungkinkan Kita menyimpan data dalam struktur seperti kisi atau matriks. *Array* multidimensi sering digunakan untuk mewakili struktur data yang kompleks seperti tabel, matriks, atau database.

Contoh *Array* Multidimensi di PHP:

```

<?php
// Creating an multidimensional array
$variable = array(
    array('value00', 'value01', 'value02'),
    array('value10', 'value11', 'value12'),
    array('value20', 'value21', 'value22')
);

// Accessing the elements
echo $variable[0][0]; // Outputs: value00
echo $variable[1][1]; // Outputs: value11
echo $variable[2][2]; // Outputs: value22
?>

```

Atau bisa juga ditulis sebagai berikut:

```

<?php
// Creating a multidimensional array
$students = array(
    array("name" => "John", "age" => 20, "grade" => "A"),
    array("name" => "Sarah", "age" => 19, "grade" => "B"),
    array("name" => "Mike", "age" => 21, "grade" => "A")
);

// Accessing elements of the multidimensional array
echo $students[0]["name"]; // Outputs: John
echo $students[1]["name"]; // Outputs: Sarah
echo $students[2]["name"]; // Outputs: Mike
echo $students[0]["age"]; // Outputs: 20
echo $students[1]["age"]; // Outputs: 19
echo $students[2]["age"]; // Outputs: 21
?>

```

Ikuti langkah-langkah ini untuk memahami *array* multidimensi di PHP:

Langkah	Deskripsi
1	Buat file baru bernama <b>style.css</b> di dalam direktori JS05_PHP-2, lalu ketik kode berikut: <pre> 1  table { 2      border-collapse: collapse; 3      border-spacing: 0; 4      width: 100%; 5      border: 1px solid #ddd; 6  } 7 8  th, td { 9      text-align: left; 10     padding: 16px; 11 } 12 13 tr:nth-child(even) { 14     background-color: #f2f2f2 15 } </pre>
2	Buat file baru bernama <b>array_3.php</b> di dalam direktori JS05_PHP-2, lalu ketik kode berikut:

	<pre> 1  &lt;!DOCTYPE HTML&gt; 2  &lt;html&gt; 3      &lt;head&gt; 4          &lt;link rel="stylesheet" type="text/css" href="style.css"/&gt; 5      &lt;/head&gt; 6      &lt;body&gt; 7          &lt;h2&gt; Multidimensional Array &lt;/h2&gt; 8          &lt;table&gt; 9              &lt;tr&gt; 10                 &lt;th&gt;Judul Film&lt;/th&gt; 11                 &lt;th&gt;Tahun&lt;/th&gt; 12                 &lt;th&gt;Rating&lt;/th&gt; 13             &lt;/tr&gt; 14             &lt;?php 15                 \$movie = array( 16                     array("Avengers: Invinty War", 2018, 8.7), 17                     array("The Avengers", 2012, 8.1), 18                     array("Guardians of the Galaxy", 2014, 8.1), 19                     array("Iron Man", 2008, 7.9) 20                 ); 21                 echo "&lt;tr&gt;"; 22                 echo "&lt;td&gt;". \$movie[0][0] . "&lt;/td&gt;"; 23                 echo "&lt;td&gt;". \$movie[0][1] . "&lt;/td&gt;"; 24                 echo "&lt;td&gt;". \$movie[0][2] . "&lt;/td&gt;"; 25                 echo "&lt;/tr&gt;"; 26                 echo "&lt;tr&gt;"; 27                 echo "&lt;td&gt;". \$movie[1][0] . "&lt;/td&gt;"; 28                 echo "&lt;td&gt;". \$movie[1][1] . "&lt;/td&gt;"; 29                 echo "&lt;td&gt;". \$movie[1][2] . "&lt;/td&gt;"; 30                 echo "&lt;/tr&gt;"; 31                 echo "&lt;tr&gt;"; 32                     echo "&lt;td&gt;". \$movie[2][0] . "&lt;/td&gt;"; 33                     echo "&lt;td&gt;". \$movie[2][1] . "&lt;/td&gt;"; 34                     echo "&lt;td&gt;". \$movie[2][2] . "&lt;/td&gt;"; 35                 echo "&lt;/tr&gt;"; 36                 echo "&lt;tr&gt;"; 37                     echo "&lt;td&gt;". \$movie[3][0] . "&lt;/td&gt;"; 38                     echo "&lt;td&gt;". \$movie[3][1] . "&lt;/td&gt;"; 39                     echo "&lt;td&gt;". \$movie[3][2] . "&lt;/td&gt;"; 40                 echo "&lt;/tr&gt;"; 41             &lt;?&gt; 42         &lt;/table&gt; 43     &lt;/body&gt; 44 &lt;/html&gt;</pre>
3	Simpan file dan jalankan program Kita di browser. Ketik tautan ini ke browser Kita <a href="localhost/dasarWeb/JS05_PHP-2/array_3.php">localhost/dasarWeb/JS05_PHP-2/array_3.php</a>
4	Amati <i>output</i> yang ditampilkan dan jelaskan hasil pengamatanmu (Pertanyaan No 3)

The screenshot shows a code editor interface with several files open. On the left, the Explorer sidebar lists files like array\_1.php, array\_2.php, style.css, and array\_3.php. The main editor area contains PHP code that defines a multidimensional array \$movie and prints it to the screen. Below the code editor is a toolbar with various icons and a status bar at the bottom.

```

<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="text/css" href="style.css"/>
</head>
<body>
<h2>Multidimensional Array</h2>
<table>
<tr>
<th>Judul Film</th>
<th>Tahun</th>
<th>Rating</th>
</tr>
<?php
$movie = array(
    array("Avengers: Infinity War", 2018, 8.7),
    array("The Avengers", 2012, 8.1),
    array("Guardians of the Galaxy", 2014, 8.1),
    array("Iron Man", 2008, 7.9)
);

echo "<tr>";
echo "<td>". $movie[0][0] . "</td>";
echo "<td>". $movie[0][1] . "</td>";
echo "<td>". $movie[0][2] . "</td>";
echo "</tr>";
echo "<tr>";
echo "<td>". $movie[1][0] . "</td>";
echo "<td>". $movie[1][1] . "</td>";
echo "<td>". $movie[1][2] . "</td>";
echo "</tr>";
echo "<tr>";
echo "<td>". $movie[2][0] . "</td>";
echo "<td>". $movie[2][1] . "</td>";
echo "<td>". $movie[2][2] . "</td>";
echo "</tr>";
echo "<tr>";
echo "<td>". $movie[3][0] . "</td>";
echo "<td>". $movie[3][1] . "</td>";
echo "<td>". $movie[3][2] . "</td>";
echo "</tr>";
</table>

```

## Fungsi

Ada banyak fungsi PHP bawaan yang sering kita gunakan, seperti `print()`, `print_r()`, `unset()`, dll. Selain fungsi-fungsi tersebut, kita juga dapat membuat fungsi kustom kita sendiri sesuai dengan kebutuhan kita. Fungsi adalah sekumpulan instruksi yang dibungkus dalam blok. Fungsi dapat digunakan kembali tanpa harus menulis ulang instruksi di dalamnya. Di PHP, fungsi dapat dibuat menggunakan kata kunci `function`, diikuti dengan nama fungsi.

Contoh:

```

function namaFungsi(){
    //...
}

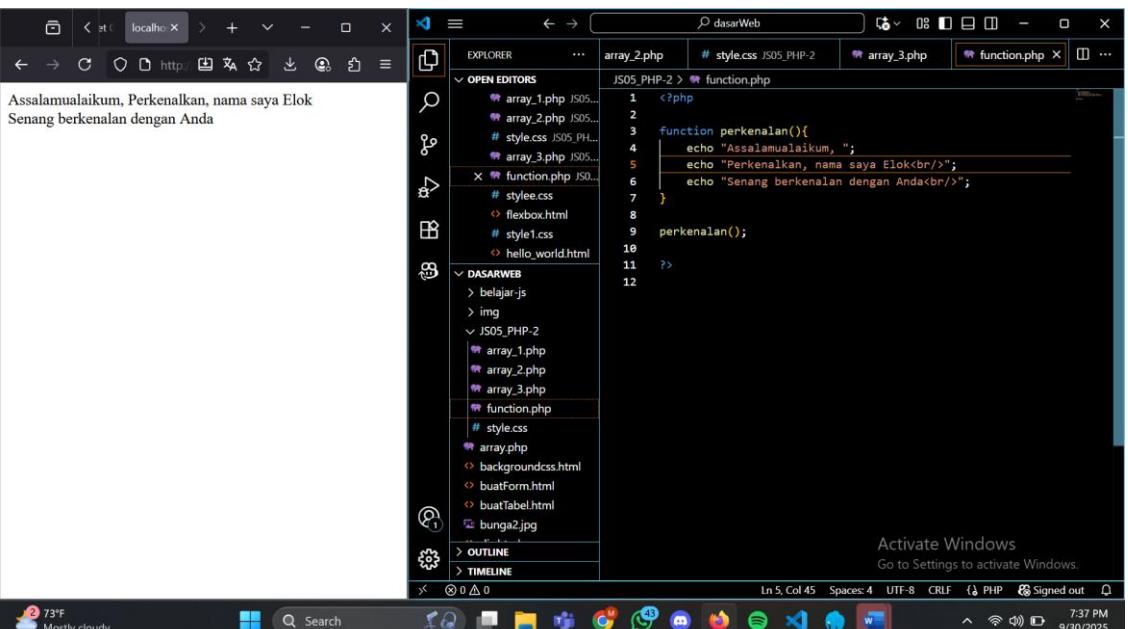
```

Kode instruksi dapat ditulis di dalam kurung kurawal (`{ ... }`). Nama fungsi di PHP harus dimulai dengan huruf atau garis bawah dan tidak boleh dimulai dengan angka. Penamaan fungsi di PHP tidak peka huruf besar/kecil (**case-insensitive**). Ikuti langkah-langkah ini untuk memahami penggunaan fungsi di PHP:

```
<?php
// Defining a function in PHP
function greet() {
    echo "Hello, welcome to PHP functions!";
}

// Calling the function
greet();
?>
```

## Praktikum 4. Fungsi

Langkah	Deskripsi
1	Buat file baru di dalam direktori JS05_PHP-2 dan beri nama <b>function.php</b> <pre>&lt;?php  function perkenalan(){     echo "Assalamualaikum, ";     echo "Perkenalkan, nama saya Elok&lt;br/&gt;"; //Tulis sesuai nama kalian     echo "Senang berkenalan dengan Anda&lt;br/&gt;";  }  //memanggil fungsi yang sudah dibuat perkenalan();  ?&gt;</pre>
2	Simpan file dan jalankan kodennya. Ubah program sehingga dapat menampilkan <i>output</i> dua kali. Jelaskan pengamatan Kita! <b>(Pertanyaan No 4)</b>  <p>The screenshot shows a Windows desktop environment. On the left, a Microsoft Edge browser window is open, displaying the output of the PHP code: "Assalamualaikum, Perkenalkan, nama saya Elok Senang berkenalan dengan Anda". On the right, a Visual Studio Code (VS Code) interface is visible. The Explorer sidebar shows files like array_1.php, array_2.php, array_3.php, style.css, and function.php. The editor pane shows the PHP code for defining and calling the perkenalan() function. The status bar at the bottom indicates the file is saved, has 5 lines and 45 columns, and is in UTF-8 encoding.</p>

	Fungsi perkenalan() digunakan untuk menampilkan teks sapaan sederhana. Setelah dipanggil, fungsi ini mencetak tiga baris teks ke halaman web: ucapan salam, perkenalan dengan nama (dalam contoh "Elok"), dan kalimat penutup
--	---

### Fungsi dengan Parameter

3	Untuk membuat instruksi di dalam fungsi lebih dinamis, kita dapat menggunakan parameter untuk meneruskan nilai ke dalam fungsi. Nilai-nilai ini akan diproses dalam fungsi. Misalnya, pada fungsi sebelumnya, tidak ideal jika nama yang dicetak selalu "Elok" dan sapaannya selalu "Assalamualaikum". Kita dapat mengubah nama orang dan sapaan menjadi kata lain.
4	Tambahkan parameter seperti pada kode program berikut:

```

<?php
//membuat fungsi
function perkenalan($nama, $salam){
    echo $salam." ";
    echo "Perkenalkan, nama saya ".$nama."<br/>";
    echo "Senang berkenalan dengan Anda<br/>";
}

//memanggil fungsi yang sudah dibuat
perkenalan("Hamdana","Hallo");

echo "<hr>";

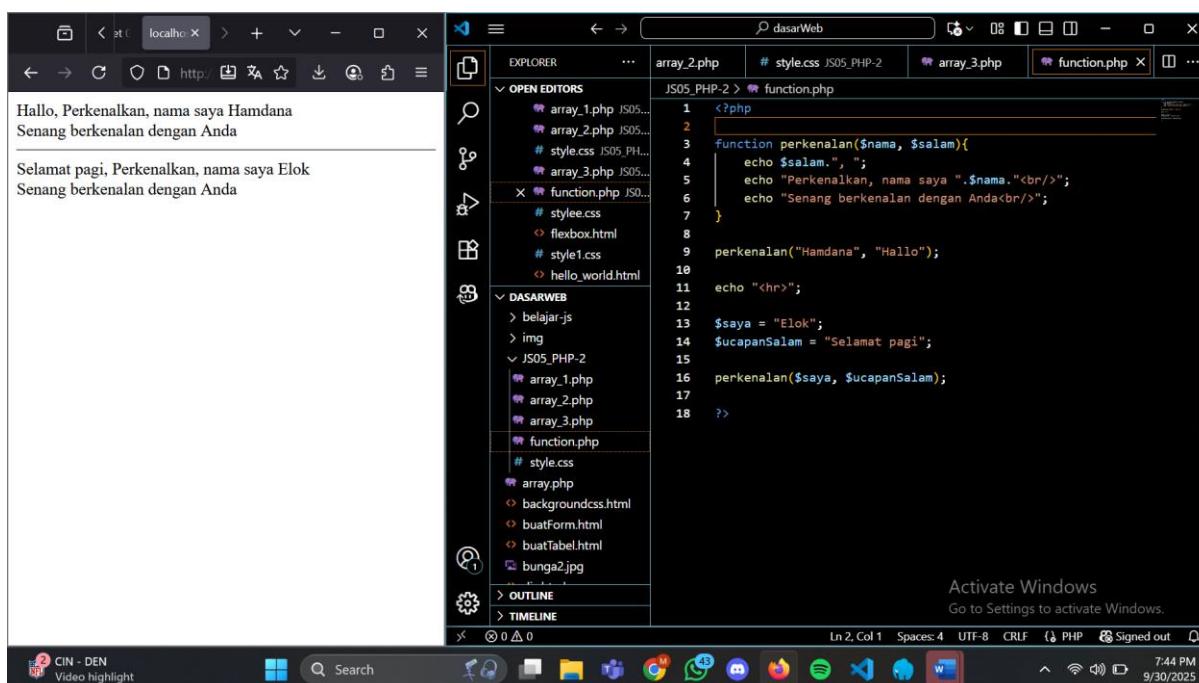
$saya = "Elok";
$ucapanSalam = "Selamat pagi";

//memanggil lagi
perkenalan($saya,$ucapanSalam);
?>

```

Amati *output* yang ditampilkan dan jelaskan hasil pengamatanmu!  
(Pertanyaan No 5)

5

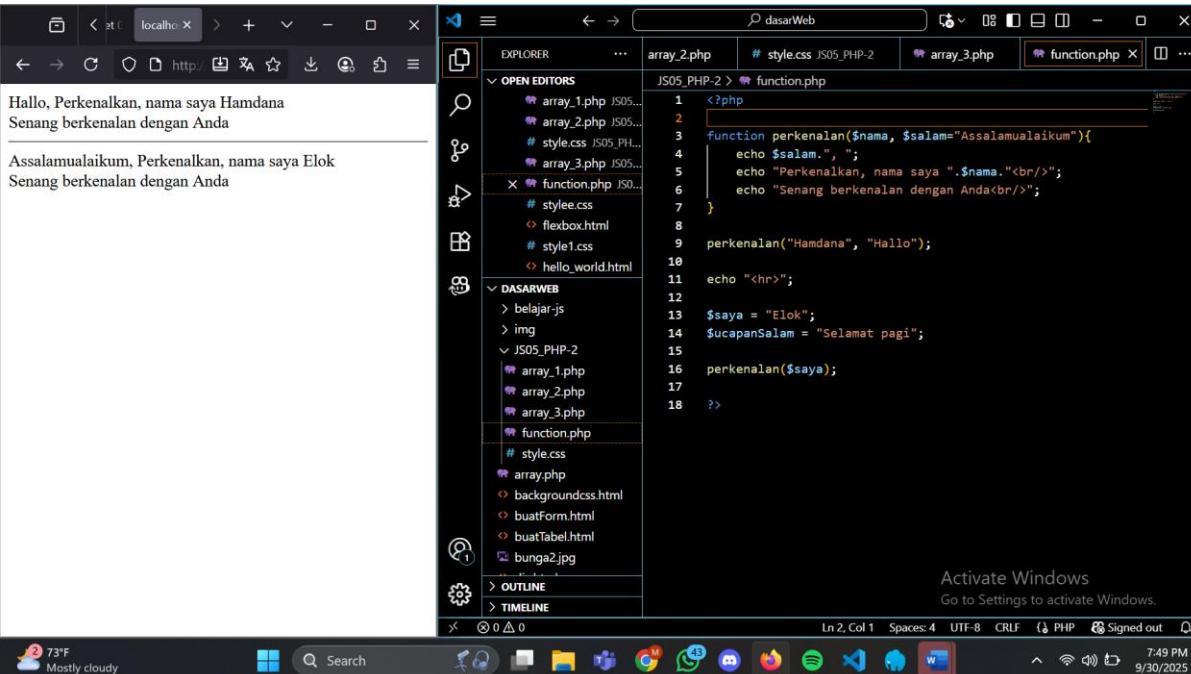


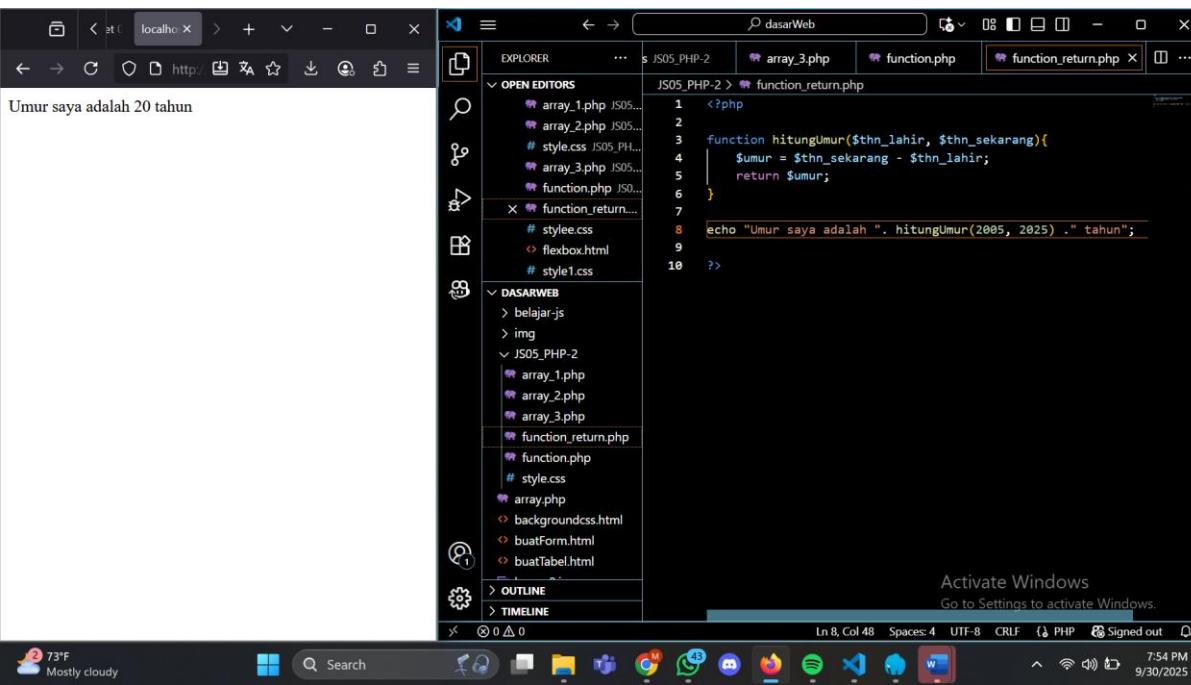
mendemonstrasikan prinsip dasar modularitas melalui penggunaan fungsi yang dapat digunakan kembali. Fungsi bernama perkenalan didefinisikan untuk mencetak ucapan salam dan perkenalan berdasarkan dua parameter, yaitu \$nama dan \$salam. Kode kemudian memanggil fungsi ini sebanyak dua kali: pertama, dengan memberikan nilai string statis ("Hamdana" dan "Hallo"); dan kedua, dengan melewatkannya variabel (\$saya dan \$ucapanSalam) yang telah diinisialisasi sebelumnya dengan nilai lain ("Elok" dan "Selamat pagi")

### Fungsi dengan Parameter dan menggunakan nilai default

6

Kita dapat menetapkan ***nilai default*** ke parameter. ***Nilai default*** berfungsi untuk memberikan nilai untuk parameter jika tidak disediakan. Misalnya: jika kita lupa memberikan parameter salam, program biasanya akan melemparkan kesalahan. Oleh karena itu, kita perlu menetapkan nilai default untuk menghindari kesalahan ini.

	Tulis kode ini ke program Kita
7	<pre>&lt;?php //membuat fungsi function perkenalan(\$nama, \$salam="Assalamualaikum"){     echo \$salam.", ";     echo "Perkenalkan, nama saya ".\$nama."&lt;br/&gt;";     echo "Senang berkenalan dengan Anda&lt;br/&gt;"; }  //memanggil fungsi yang sudah dibuat perkenalan("Hamdana","Hallo");  echo "&lt;hr&gt;";  \$saya = "Elok"; \$ucapanSalam = "Selamat pagi";  //memanggil lagi tanpa mengisi parameter salam perkenalan(\$saya); ?&gt;</pre>
8	<p>Amati <i>output</i> yang ditampilkan dan jelaskan hasil pengamatanmu!</p> <p>(Pertanyaan No 6)</p>  <p>demonstrasi fitur <b>nilai parameter <i>default</i></b> pada fungsi PHP. Fungsi perkenalan didefinisikan dengan parameter salam yang memiliki nilai <i>default</i> "Assalamualaikum". Poin pentingnya adalah, saat fungsi dipanggil dengan argumen lengkap ("Hallo"), nilai <i>default</i> akan ditimpa, tetapi saat parameter salam <b>dihilangkan</b> dalam pemanggilan kedua, fungsi secara otomatis akan kembali menggunakan nilai <i>default</i>* "Assalamualaikum". Hal ini menunjukkan bagaimana parameter <i>default</i> meningkatkan <b>fleksibilitas</b> fungsi, memungkinkannya dipanggil tanpa harus selalu menyediakan setiap argumen.</p>
	<b>Fungsi dengan nilai pengembalian</b>
9	Fungsi <b>dengan mengembalikan nilai</b> adalah fungsi yang dirancang untuk memproses data dan mengirim hasilnya kembali ke titik di mana ia dipanggil. Di PHP, Kita dapat

	menggunakan pernyataan return untuk mengembalikan nilai dari suatu fungsi. Ini berguna ketika Kita memerlukan fungsi untuk melakukan perhitungan atau operasi dan memberikan hasilnya kembali ke program utama untuk digunakan lebih lanjut.
10	Buat file baru di dalam direktori JS05_PHP-2 dan beri nama <b>function_return.php</b> <pre>&lt;?php //membuat fungsi function hitungUmur(\$thn_lahir, \$thn_sekarang){     \$umur = \$thn_sekarang - \$thn_lahir;     return \$umur; }  echo "Umur saya adalah ". hitungUmur(1988, 2023) ."tahun" //isi sesuai dengan tahun lahir kalian  ?&gt;</pre>
11	Amati <i>output</i> yang ditampilkan dan jelaskan hasil pengamatanmu! <b>(Pertanyaan No 7)</b>  <p>mendemonstrasikan pentingnya penggunaan <b>keyword return</b> dalam sebuah fungsi PHP. Fungsi hitungUmur bertugas menghitung selisih tahun dan mengembalikan hasilnya (\$umur) sebagai nilai numerik kepada kode yang memanggilnya, alih-alih langsung mencetak hasil di dalamnya. Nilai yang dikembalikan tersebut, yaitu 20 (hasil dari 2025–2005), kemudian ditangkap oleh pernyataan echo di luar fungsi. Hal ini membuktikan bahwa <b>return</b> memungkinkan output dari sebuah fungsi untuk digunakan dan dimanipulasi lebih lanjut di bagian lain dari kode</p>
	<b>Memanggil fungsi di dalam fungsi lain</b>
12	<b>Memanggil Fungsi Di Dalam Fungsi Lain</b> adalah praktik pemrograman umum dalam PHP dan bahasa lain. Ini memungkinkan Kita untuk memecah tugas kompleks menjadi bagian-bagian yang lebih kecil dan dapat digunakan kembali dengan mengaturnya menjadi fungsi terpisah dan kemudian memanggil satu fungsi dari fungsi lainnya.

Ubah `function_return.php` seperti kode ini

```
<?php
//membuat fungsi
function hitungUmur($thn_lahir, $thn_sekarang){
    $umur = $thn_sekarang - $thn_lahir;
    return $umur;
}
function perkenalan ($nama, $salam="Assalamualaikum") {
    echo $salam.;
    echo "Perkenalkan, nama saya ".$nama."<br/>";

    //memanggil fungsi lain
    echo "Saya berusia ". hitungUmur(1988, 2023) ." tahun<br/>";
    echo "Senang berkenalan dengan anda<br/>";
}

//memanggil fungsi perkenalan
perkenalan ("Elok");

?>
```

13

Amati *output* yang ditampilkan dan jelaskan hasil pengamatanmu!  
(Pertanyaan No 8)

14

The screenshot shows a Windows desktop environment. On the left is a code editor window titled 'dasarWeb' containing PHP code. On the right is a web browser window titled 'localhost' displaying the output of the PHP code. The browser output is:

```
Assalamualaikum, Perkenalkan, nama saya Yusuf
Saya berusia 20 tahun
Senang berkenalan dengan anda
```

The code editor window shows the following PHP code:

```
1 <?php
2
3 function hitungUmur($thn_lahir, $thn_sekarang){
4     $umur = $thn_sekarang - $thn_lahir;
5     return $umur;
6 }
7
8 function perkenalan ($nama, $salam="Assalamualaikum") {
9     echo $salam.;
10    echo "Perkenalkan, nama saya ".$nama."<br/>";
11
12    echo "Saya berusia ". hitungUmur(2005, 2025) ." tahun<br/>";
13
14    echo "Senang berkenalan dengan anda<br/>";
15 }
16
17 perkenalan ("Yusuf");
18
19 ?>
```

The status bar at the bottom of the code editor shows: Activate Windows Go to Settings to activate Windows. The system tray icons include: NYJ - MIA Video highlight, Search, Task View, File Explorer, Control Panel, Google Chrome, Spotify, and Microsoft Edge.

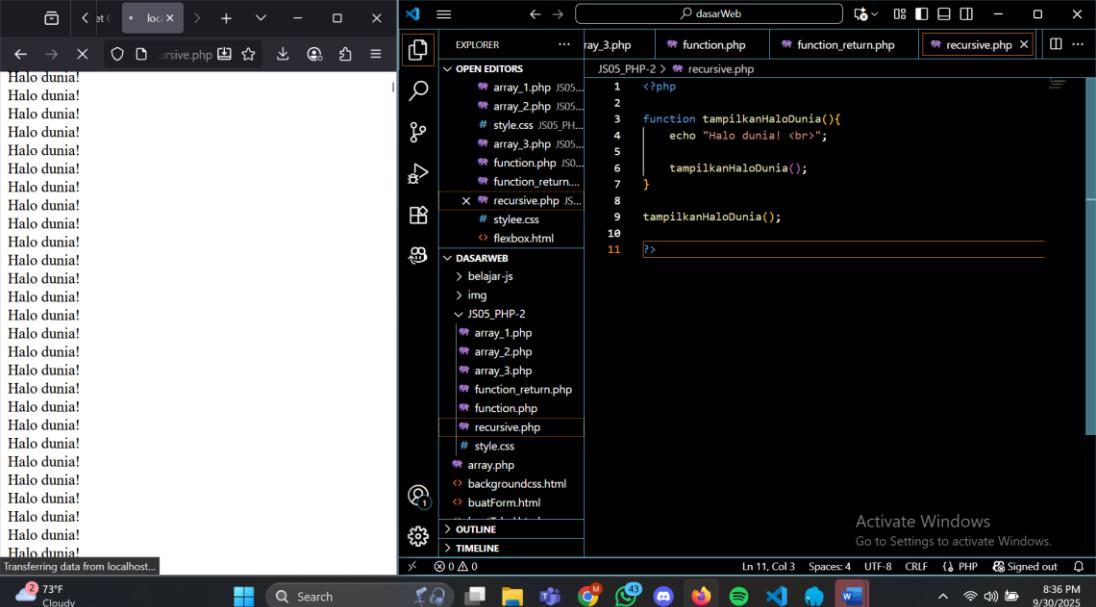
Modularitas: Tugas perhitungan (`hitungUmur`) dipisahkan dari tugas tampilan (`perkenalan`).

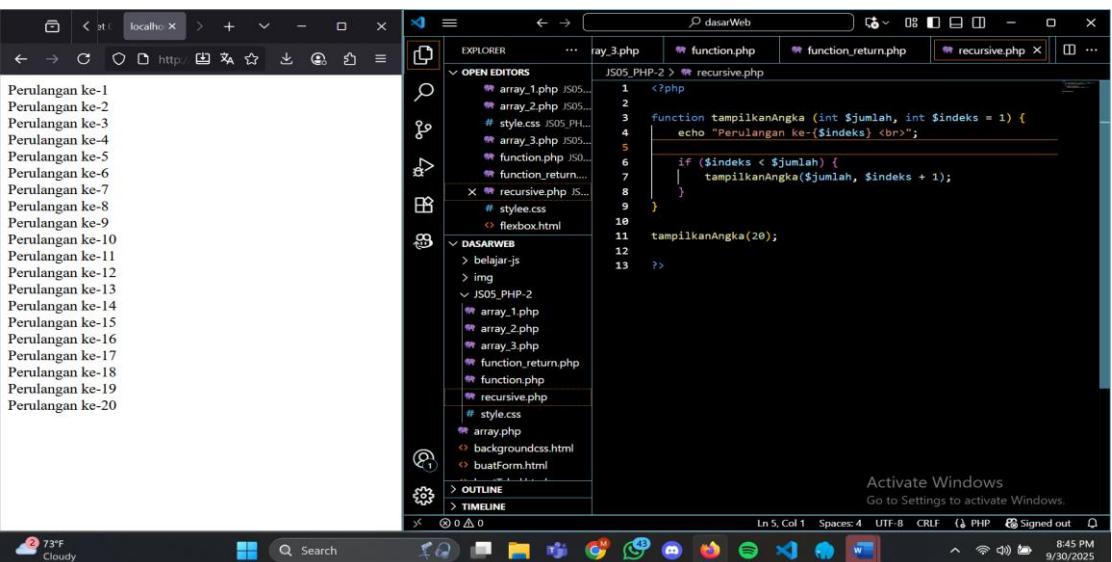
return: Fungsi `hitungUmur` menggunakan `return` untuk mengembalikan nilai usia (20 tahun), yang kemudian disisipkan ke dalam *output* perkenalan.

Parameter *Default*: Fungsi `perkenalan` menggunakan parameter *default* "Assalamualaikum" yang terpakai karena argumen salam dihilangkan saat pemanggilan.

## Praktikum 5. Fungsi Rekursif

**Fungsi Rekursif** di PHP adalah fungsi yang memanggil dirinya sendiri selama eksekusi. Teknik ini berguna untuk memecahkan masalah yang dapat dibagi menjadi submasalah yang lebih kecil dan serupa, sering disebut sebagai membagi dan menaklukkan. Fungsi rekursif biasanya digunakan untuk memecahkan masalah seperti menghitung faktorial, bilangan Fibonacci, dan pemrograman dinamis.

Langkah	Deskripsi
1	Buat file baru bernama <b>recursive.php</b> di dalam direktori JS05_PHP-2, lalu ketik kode berikut: <pre>&lt;?php function tampilanHaloDunia(){     echo "Halo dunia! &lt;br&gt;";     tampilanHaloDunia(); }  tampilanHaloDunia(); ?&gt;</pre>
2	Jika kode program di atas dieksekusi, apa yang akan terjadi dan apa dampaknya dari melakukannya? Silakan bagikan pendapat Kita! <b>(Pertanyaan No 9)</b>  <ol style="list-style-type: none"><li>1. Rekursi: Fungsi tampilanHaloDunia memanggil dirinya sendiri di dalamnya.</li><li>2. Masalah: Kode tidak memiliki kondisi berhenti (<i>base case</i>), menyebabkan fungsi memanggil dirinya sendiri secara terus-menerus.</li><li>3. Dampak: Hasilnya adalah pencetakan <i>string</i> "Halo dunia!" berulang kali hingga PHP mencapai batas maksimum kedalaman tumpukan fungsi (<i>maximum function nesting level</i>), yang akan menyebabkan program berhenti dengan <i>fatal error</i> di luar layar.</li></ol>

	Untuk menampilkan angka 1 hingga 25, kita dapat dengan mudah menggunakan <i>loop for</i> sebagai berikut:
3	<pre>&lt;?php for (\$i=1; \$i &lt;=25; \$i++){     echo "Perulangan ke-{\$i} &lt;br&gt;"; }  ?&gt;</pre>
4	Kita dapat membuat tampilan angka 1 hingga 25 menggunakan fungsi rekursif (tanpa <i>loop for</i> ). <pre>&lt;?php function tampilkanAngka (int \$jumlah, int \$indeks = 1) {     echo "Perulangan ke-{\$indeks} &lt;br&gt;";      // panggil diri sendiri selama \$indeks &lt;= \$jumlah     if (\$indeks &lt; \$jumlah) {         tampilkanAngka(\$jumlah, \$indeks + 1);     } } tampilkanAngka(20); ?&gt;</pre>
5	Jalankan kode program di atas dan jelaskan <i>output</i> nya, lalu jelaskan mengapa ia berperilaku seperti itu.
	<p>(Pertanyaan No 10)</p>  <p>The screenshot shows a Windows desktop environment. In the center, there's a Microsoft Edge browser window displaying the output of a PHP script. The output consists of 20 lines of text, each starting with 'Perulangan ke-' followed by a number from 1 to 20. To the left of the browser, a file explorer window is open, showing a directory structure under 'JS05_PHP-2'. Inside this directory, there are several PHP files like 'array_1.php', 'array_2.php', etc., and some CSS files. The file 'recursive.php' is also visible. The desktop taskbar at the bottom shows various pinned icons and the date/time as 9/30/2025.</p> <ol style="list-style-type: none"> <li>1. Rekursi Terkontrol: Fungsi tampilkanAngka berhasil melakukan perulangan dengan memanggil dirinya sendiri.</li> <li>2. Kondisi Berhenti: Adanya pernyataan <code>if (\$indeks &lt; \$jumlah)</code> berfungsi sebagai kondisi berhenti (<i>base case</i>), yang mencegah <i>infinite loop</i>.</li> <li>3. Hasil: Rekursi ini berhasil mencetak string "Perulangan ke-" dari 1 hingga 20 (sesuai dengan nilai yang dilewatkan <code>tampilkanAngka(20)</code>), membuktikan bahwa fungsi rekursif dapat digunakan sebagai alternatif yang valid untuk <i>loop</i> tradisional seperti <i>for</i> atau <i>while</i>, asalkan memiliki <i>base case</i> yang jelas.</li> </ol>

## Menu Multi-Level menggunakan Array

Menu Multi-Level menggunakan *Array* di PHP mengacu pada pembuatan struktur menu hierarkis atau bersarang di mana setiap item menu dapat memiliki sub-item. Hal ini berguna untuk membuat menu navigasi di situs web yang memiliki struktur yang lebih kompleks, seperti bagian dengan sub-bagian atau kategori dengan subkategori.

### Contoh Menu Multi-Level menggunakan *Array* di PHP:

Untuk membuat struktur menu multi-level atau bersarang, *array* dapat digunakan untuk mewakili item menu dan sub-itemnya. Berikut adalah contoh yang menunjukkan bagaimana Kita dapat menggunakan *array* multidimensi untuk membuat menu semacam itu.

```
<?php
// Menentukan menu multi-level menggunakan array asosiatif
$menu = array(
    "Rumah" => "#home",
    "Tentang Kami" => array(
        "Tim kami" => "#team",
        "Kisah Kami" => "#story",
        "Misi & Visi" => "#mission"
    ),
    "Layanan" => array(
        "Pengembangan Web" => "#web",
        "Pengembangan Seluler" => "#mobile",
        "Optimasi SEO" => "#seo"
    ),
    "Kontak" => "#contact"
);

// Fungsi untuk menampilkan menu
function displayMenu($menu) {
    echo "<ul>";
    foreach ($menu as $key => $value) {
        // Periksa apakah item menu adalah array (artinya memiliki sub-item)
        if (is_array($value)) {
            echo "<li>$key";
            displayMenu($value); // Tampilkan sub-menu secara rekursif
            echo "</li>";
        } else {
            echo "<li><a href='$value'>$key</a></li>";
        }
    }
    echo "</ul>";
}

// Panggil fungsi untuk menampilkan menu
displayMenu($menu);
?>
```

Kita dapat mencoba kode di atas di file [array\\_menu.php](#), dan menjalankannya di browser.

## Praktikum 6. Menu Multi-Level

Langkah	Deskripsi
1	Buat <b>\$menu</b> variabel. Variabel ini adalah kombinasi dari <i>array</i> terindeks dan <i>array</i> asosiatif multidimensi. <b>Ini disebut multidimensi karena merupakan <i>array</i> yang berisi <i>array</i> lain</b>
	<b>di dalamnya.</b> Selanjutnya, kita akan mencoba menampilkan semua item dari <a href="#">\$menu</a> menggunakan fungsi rekursif

	<p>Tulis kode ini ke dalam file <b>multi_menu.php</b></p> <pre>&lt;?php \$menu = [     [         "nama" =&gt; "Beranda"     ],     [         "nama" =&gt; "Berita",         "subMenu" =&gt; [             [                 "nama" =&gt; "Wisata",                 "subMenu" =&gt; [                     [                         "nama" =&gt; "Pantai"                     ],                     [                         "nama" =&gt; "Gunung"                     ]                 ]             ],             [                 "nama" =&gt; "Kuliner"             ],             [                 "nama" =&gt; "Hiburan"             ]         ]     ],     [         "nama" =&gt; "Tentang"     ],     [         "nama" =&gt; "Kontak"     ], ]; </pre>
2	<p>Setelah menulis kode di langkah 2, tulis kode ini setelahnya, untuk menampilkan menu di browser</p> <pre>function tampilanMenuBertingkat (array \$menu) {     echo "&lt;ul&gt;";     foreach (\$menu as \$key =&gt; \$item) {         echo "&lt;li&gt;{\$item['nama']}&lt;/li&gt;";     }     echo "&lt;/ul&gt;"; }  tampilanMenuBertingkat(\$menu); ?&gt;</pre>
3	<p>Jalankan program di atas dan jelaskan <i>output</i> nya.  <b>(Pertanyaan No 11)</b></p>
4	

```

<?php
$menu = [
    ["nama" => "Beranda"],
    [
        "nama" => "Berita",
        "subMenu" => [
            [
                "nama" => "Wisata",
                "subMenu" => [
                    ["nama" => "Pantai"],
                    ["nama" => "Gunung"]
                ]
            ],
            [
                "nama" => "Kuliner"
            ],
            [
                "nama" => "Hiburan"
            ]
        ]
    ],
    [
        "nama" => "Tentang"
    ],
    [
        "nama" => "Kontak"
    ]
];

```

menampilkan menu bertingkat dari array \$menu telah berhasil diimplementasikan di file multi\_menu.php. Namun, output di sisi browser saat ini hanya menampilkan item menu level atas ("Beranda", "Berita", "Tentang", "Kontak") dalam format *unordered list* (<ul>), dan belum menampilkan submenu di dalamnya ("Wisata", "Kuliner", "Hiburan", "Pantai", "Gunung"). Ini menandakan bahwa implementasi fungsi rekursif untuk memproses dan menampilkan submenu pada array belum sepenuhnya tercermin di *output* akhir, atau *output* yang ditampilkan masih merupakan hasil dari kode fungsi yang lama dan belum diperbarui dengan kode rekursif yang telah dibuat.

- 5 Selanjutnya, buat fungsi di atas rekursif dengan memanggil dirinya sendiri ketika item dari menu memiliki atribut submenu. Ini akan menghasilkan tampilan seperti berikut.

- Beranda
- Berita
  - Wisata
    - Pantai
    - Gunung
  - Kuliner
  - Hiburan
- Tentang
- Kontak

(Pertanyaan No 12)

```

EXPLORER ... irr.php recursive.php multi_menu.php # stylee.css flexbox ...
OPEN EDITORS JS05_PHP-2 > multi_menu.php
array_1.php JS05...
array_2.php JS05...
# style.css JS05...
array_3.php JS05...
function.php JS05...
function_return.php JS05...
recursive.php JS05...
multi_menu.php ...
# stylee.css
DASARWEB
> belajar.js
> img
> JS05_PHP-2
array_1.php
array_2.php
array_3.php
function_return.php
function.php
multi_menu.php
recursive.php
# style.css
array.php
backgroundcss.html
OUTLINE
TIMELINE
Ln 45, Col 10 Spaces: 4 UTF-8 CRLF ( PHP Signed out
Activate Windows
Go to Settings to activate Windows.
11:13 PM
10/1/2025

```

fungsi rekursif **tampilkanMenuBertingkat** pada file multi\_menu.php telah berhasil diimplementasikan untuk memproses dan menampilkan array menu bertingkat. Tampilan di browser menunjukkan keberhasilan fungsi tersebut dalam membuat struktur HTML bersarang (<ul> di dalam <li>), yang secara visual menghasilkan menu yang menjorok (berjenjang) sesuai kedalaman array, seperti "**Wisata**" yang menjorok dari item utama, dan "**Pantai**" serta "**Gunung**" yang menjorok lebih dalam lagi. Namun, terdapat keanehan di mana nama item level atas yang memiliki submenu, seperti "**Berita**", tampak tidak ikut tercetak, meskipun semua submenu-nya sudah ditampilkan dengan benar, yang mengindikasikan adanya *bug* kecil pada baris kode yang mencetak nama item tersebut sebelum proses rekursi dimulai.

## Tali

Dalam PHP, string dapat didefinisikan menggunakan **tanda kutip ganda** (" ") atau **tanda kutip tunggal** (' '). Meskipun keduanya memungkinkan Kita untuk membuat variabel string, ada perbedaan penting dalam cara mereka menangani variabel dan karakter khusus.

### 1. Tanda Kutip Ganda (" ") :

- Tanda kutip ganda memungkinkan **interpolasi variabel**, yang berarti bahwa variabel dalam string akan dievaluasi dan diganti dengan nilainya.
- Karakter khusus (urutan escape) seperti \n (baris baru) atau \t (tab) dikenali dan diproses.
- **Contoh:**

```

<?php
$name = "John Wick";
echo "Halo, $name!"; // Keluaran: Halo, John!
echo "Ini adalah baris baru.\nbaris berikutnya."; // Output: Ini adalah baris baru

```

### 2. Tanda Kutip Tunggal (' ') :

- Tanda kutip tunggal memperlakukan string secara **harfiah**, artinya variabel di dalam string tidak dievaluasi.
- Urutan escape tidak diproses, kecuali untuk \\ (garis miring terbalik) dan \' (tanda kutip tunggal).
- **Contoh:**

```

<?php
$name = 'Yohanes';
echo "Halo, $name!"; // Keluaran: Halo, $name!
echo 'Ini adalah baris baru.\nbaris berikutnya.'; // Output: Ini adalah baris baru.\nBaris berikutnya.

```

### Perbedaan Utama:

- **Penguraian Variabel:** Tanda kutip ganda akan menggantikan variabel dengan nilainya, sedangkan tanda kutip tunggal akan menampilkan nama variabel sebagai teks biasa.
- **Karakter Escape:** Tanda kutip ganda mengenali urutan escape khusus seperti `\n`, `\t`, sedangkan tanda kutip tunggal hanya mengenali `\'` dan `\\"`.

Secara umum, **gunakan tanda kutip ganda** saat Kita membutuhkan interpolasi variabel atau karakter khusus. Jika Kita tidak memerlukan fitur ini, **tanda kutip tunggal** lebih cepat dan lebih efisien untuk definisi string sederhana.

Beberapa operasi dapat dilakukan pada data tipe string. PHP menyediakan fungsi bawaan yang siap digunakan untuk operasi string

Fungsi	Deskripsi
<code>strlen()</code>	Untuk mengetahui panjang string
<code>str_word_count()</code>	Untuk menghitung jumlah kata dalam string
<code>strpos()</code>	Untuk menemukan posisi substring dalam string
<code>strrev()</code>	Untuk membalikkan urutan string
<code>strstr()</code>	Untuk mencari substring dalam string
<code>substr()</code>	Untuk mengekstrak substring dari posisi awal hingga akhir dalam string
<code>trim()</code>	Untuk menghapus spasi kosong dari awal dan akhir string
<code>ltrim()</code>	Untuk menghapus spasi kosong dari awal string
<code>rtrim()</code>	Untuk menghapus spasi kosong dari akhir string
<code>strtoupper()</code>	Untuk mengonversi semua karakter dalam string menjadi huruf besar
<code>strtolower()</code>	Untuk mengonversi semua karakter dalam string menjadi huruf kecil
<code>str_replace()</code>	Untuk mengganti bagian string dengan string lain
<code>ucwords()</code>	Untuk menggunakan huruf besar huruf pertama dari setiap kata dalam string
<code>meledak()</code>	Untuk membagi string menjadi <i>array</i> berdasarkan karakter tertentu

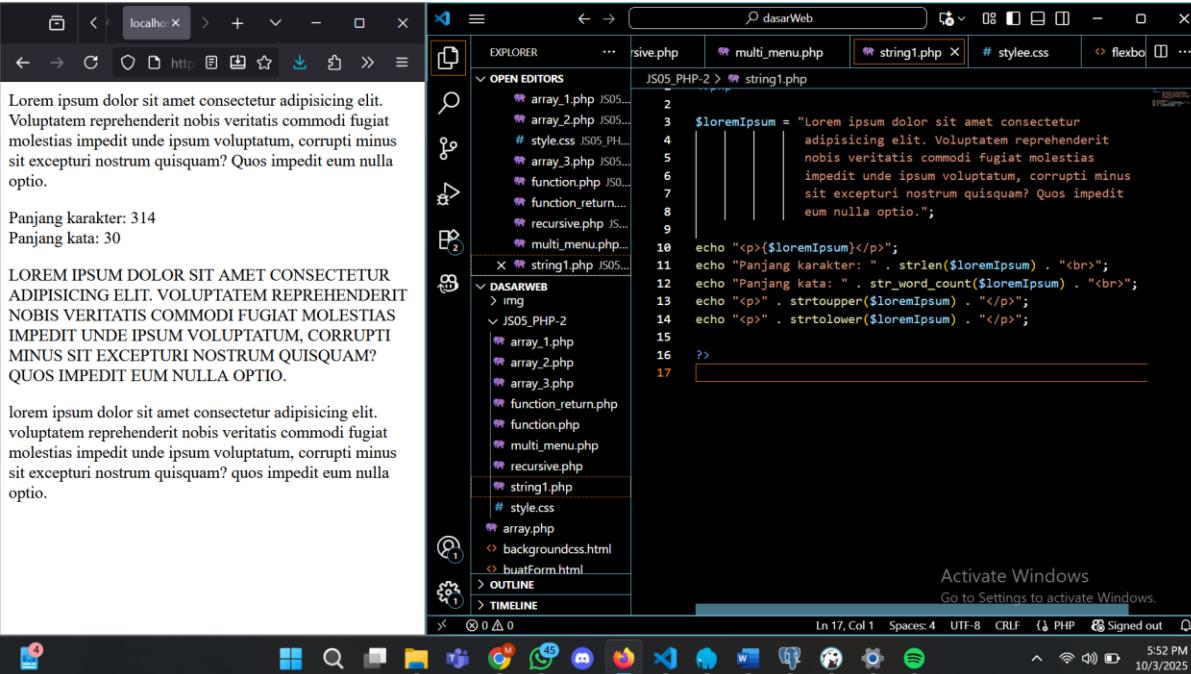
### Karakter Melarikan Diri

Karakter khusus yang tidak dapat ditampilkan secara langsung harus didahului dengan garis miring terbalik (\). String yang diapit dalam tanda kutip ganda akan mengganti karakter escape dengan karakter yang diwakilinya. Ini berbeda dari string yang diapit dalam tanda kutip tunggal, di mana ia hanya akan menampilkan konten apa adanya tanpa mengganti apa pun (dengan beberapa pengecualian).

Karakter escape di PHP adalah:

Fungsi	Deskripsi
<code>\n</code>	Baris baru
<code>\r</code>	Karakter kereta kembali
<code>\t</code>	Karakter tab
<code>\\$</code>	Karakter \$ itu sendiri
<code>\"</code>	Untuk menampilkan tanda kutip ganda
<code>\\"</code>	Untuk menampilkan garis miring terbalik (\) itu sendiri

## Praktikum 7. String

Langkah	Deskripsi
1	Buat file bernama <b>string1.php</b> di dalam direktori JS05_PHP-2, lalu ketik kode berikut: <pre>&lt;?php  \$loremIpsum = "Lorem ipsum dolor sit amet consectetur adipisicing elit. Voluptatem reprehenderit nobis veritatis commodi fugiat molestias impedit unde ipsum voluptatum, corrupti minus sit excepturi nostrum quisquam? Quos impedit eum nulla optio.";  echo "&lt;p&gt;{\$loremIpsum}&lt;/p&gt;"; echo "Panjang karakter: " . strlen(\$loremIpsum) . "&lt;br&gt;"; echo "Panjang kata: " . str_word_count(\$loremIpsum) . "&lt;br&gt;"; echo "&lt;p&gt;" . strtoupper(\$loremIpsum) . "&lt;/p&gt;"; echo "&lt;p&gt;" . strtolower(\$loremIpsum) . "&lt;/p&gt;";  ?&gt;</pre>
2	Amati <i>output</i> yang ditampilkan dan jelaskan hasil pengamatannya. <b>(Pertanyaan No 13)</b>  <p>The screenshot shows a code editor with the file 'string1.php' open. The code defines a variable \$loremIpsum containing a paragraph of Lorem Ipsum text. It then uses echo statements to print the text, its length, its word count, and both uppercase and lowercase versions of the text. Below the code editor is a browser window showing the resulting output. The output includes the original text, followed by three lines of statistics: 'Panjang karakter: 314', 'Panjang kata: 30', and two versions of the text where all letters are either uppercase or lowercase.</p> <p>menghitung jumlah karakter dengan <code>strlen()</code> yang menghasilkan 314 karakter, menghitung jumlah kata dengan <code>str_word_count()</code> yang menghasilkan 30 kata, mengkonversi teks menjadi huruf besar dengan <code>strtoupper()</code>, dan mengkonversi teks menjadi huruf kecil dengan <code>strtolower()</code></p>

### Karakter Melarikan Diri

3	Buat file bernama <b>string2.php</b> di dalam direktori JS05_PHP-2, lalu ketik kode berikut: <pre>&lt;?php  echo "Baris\nbaru &lt;br&gt;"; //soal 10.a echo 'Baris\nbaru &lt;br&gt;'; //soal 10.b echo "Halo\rDunia &lt;br&gt;"; //soal 10.c echo 'Halo\rDunia &lt;br&gt;'; //soal 10.d  echo "&lt;pre&gt;Halo\tDunia!&lt;/pre&gt;"; //soal 10.e echo '&lt;pre&gt;Halo\tDunia!&lt;/pre&gt;'; //soal 10.f  echo "Katakanlah \"Tidak pada narkoba!\" &lt;br&gt;"; //soal 10.g echo 'Katakanlah \'Tidak pada narkoba!\' &lt;br&gt;'; //soal 10.h  ?&gt;</pre>
---	---

4

Dari kode program di atas, Kita dapat mengamati perbedaan antara tanda kutip ganda dan tanda kutip tunggal dalam hal bagaimana mereka menangani string escape. Amati *output* dan jelaskan hasil dari setiap *output*. Kesimpulan apa yang dapat Kita tarik dari percobaan ini? (Pertanyaan No 14)

```

<?php
echo "Baris baru<br>"; //soal 10.a
echo 'Baris\nbaru<br>'; //soal 10.b
echo "Halo\rDunia<br>"; //soal 10.c
echo 'Halo\tDunia<br>'; //soal 10.d
echo "<pre>Halo\tDunia!</pre>"; //soal 10.e
echo '<pre>Halo\tDunia!</pre>'; //soal 10.f
echo "Katakanlah \"Tidak pada narkoba!\"<br>"; //soal 10.g
echo 'Katakanlah \'Tidak pada narkoba\'<br>'; //soal 10.h
?>

```

tanda kutip ganda ("") menginterpretasikan karakter *escape* khusus seperti \n (baris baru), \r (carriage return), \t (tab), dan membutuhkan \" untuk mencetak tanda kutip ganda.

Sebaliknya, tanda kutip tunggal ('') memperlakukan hampir semua karakter *escape* sebagai teks literal (\n, \r, \t) dan hanya menginterpretasikan \' dan \\. Perbedaan ini menyebabkan output yang berbeda secara visual, seperti terlihat pada perbandingan soal 10.a/10.b dan 10.c/10.d.

### Membalikkan String menggunakan fungsi strrev()

5

Buat file bernama **string3.php** di dalam direktori JS05\_PHP-2, lalu ketik kode berikut:

```

<?php
$pesan = "Saya arek malang";
echo strrev($pesan) . "<br>";
?>

```

6

Amati *output* yang ditampilkan dan jelaskan hasil pengamatannya (Pertanyaan No 15)

The screenshot shows a Windows desktop environment. On the left is a code editor window titled 'dasarWeb' with several tabs open. The main editor area contains the following PHP code:

```
<?php
$pesan = "Saya arek malang";
echo strrev($pesan) . "<br>";
?>
```

To the right of the code editor is a web browser window showing the result of the code execution: 'gnalam kera ayaS'. The browser's address bar shows 'localhost'.

mendemonstrasikan penggunaan fungsi `strrev()` untuk membalikkan urutan karakter dalam sebuah string. Program menyimpan teks "Saya arek malang" ke dalam variabel `$pesan`, kemudian menggunakan fungsi `strrev()` untuk membalik urutannya sehingga menghasilkan output "gnalam kera ayaS". Ini merupakan contoh sederhana untuk memahami cara kerja fungsi pembalikan string di PHP, yang dapat berguna dalam berbagai keperluan seperti enkripsi sederhana, validasi palindrom, atau manipulasi teks lainnya

	Untuk membalikkan string kata demi kata, ketik kode program berikut:
7	<pre>&lt;?php \$pesan = "saya arek malang"; # ubah variabel \$pesan menjadi array dengan perintah explode \$pesanPerKata = explode(" ", \$pesan); # ubah setiap kata dalam array menjadi kebalikannya \$pesanPerKata = array_map(fn(\$pesan) =&gt; strrev(\$pesan), \$pesanPerKata); # gabungkan kembali array menjadi string \$pesan = implode(" ", \$pesanPerKata);  echo \$pesan . "&lt;br&gt;"; ?&gt;</pre>
8	Amati <i>output</i> yang ditampilkan dan jelaskan hasil pengamatanmu <b>(Pertanyaan No 16)</b>

The screenshot shows a code editor interface with multiple tabs open. The active tab is 'string3.php' containing the following PHP code:

```

1 <?php
2
3 $pesan = "saya arek malang";
4 $pesanPerKata = explode(" ", $pesan);
5 $pesanPerKata = array_map('fn($pesan) => strrev($pesan)', $pesanPerKata);
6 $pesan = implode(" ", $pesanPerKata);
7
8 echo $pesan . "<br>";
9 ?>

```

The code demonstrates how to reverse each word in a sentence using PHP's explode(), array\_map(), and implode() functions.

mendemonstrasikan teknik manipulasi string tingkat lanjut yang membalikkan urutan kata dalam sebuah kalimat. Program memproses teks "saya arek malang" melalui beberapa tahap: pertama memecah kalimat menjadi array kata menggunakan explode(), kemudian membalik setiap kata secara individual dengan array\_map() dan strrev(), dan terakhir menggabungkan kembali kata-kata tersebut menggunakan implode() sehingga menghasilkan output "ayas kera gnalam". Ini merupakan contoh penerapan fungsi array dan string secara bersamaan untuk mencapai transformasi teks yang kompleks, yang menunjukkan bagaimana beberapa fungsi PHP dapat dikombinasikan untuk mengolah data string secara lebih canggih.

## Menggabungkan HTML dan PHP

**Menggabungkan HTML dan PHP** adalah praktik umum dalam pengembangan web di mana PHP disematkan dalam HTML untuk menghasilkan konten secara dinamis di halaman web. Kode PHP dapat dimasukkan ke dalam HTML untuk memproses data, menangani input pengguna, atau menampilkan konten dinamis.

### Praktikum 8. HTML dan PHP

Langkah	Deskripsi
1	Metode pertama adalah PHP di dalam HTML. Berikut adalah contoh kodennya: <pre> &lt;html&gt; &lt;head&gt;     &lt;title&gt;Cara 01&lt;/title&gt; &lt;/head&gt; &lt;body&gt; &lt;p&gt;Tanggal Hari ini : &lt;?php echo date("d M Y")?&gt;&lt;/p&gt; &lt;/body&gt; &lt;/html&gt; </pre>

2	Kode di atas adalah kode HTML yang berisi kode PHP untuk menampilkan tanggal server, ditandai dengan tag <code>&lt;?php</code> dan <code>?&gt;</code>
3	Metode kedua adalah HTML di dalam PHP. Dalam PHP, tag HTML diperlakukan sebagai string yang diapit dalam tanda kutip, dan berbagai fungsi dapat diterapkan untuk memanipulasi string, seperti penggabungan, dll. Berikut adalah contoh cuplikan kode: <pre style="background-color: #2e3436; color: #eeeeec; padding: 10px;">&lt;?php echo '&lt;html&gt;'; echo '&lt;head&gt;&lt;title&gt;Cara02&lt;/title&gt;&lt;/head&gt;'; echo '&lt;body&gt;'; echo '&lt;p&gt;Tanggal Hari ini : '.date('d M Y').'&lt;/p&gt;'; echo '&lt;/body&gt;'; echo '&lt;/html&gt;'; ?&gt;</pre>
4	Kode di atas menghasilkan <i>output</i> yang sama dengan cuplikan kode sebelumnya. Namun, perbedaannya terletak pada cara kode ditulis, di mana HTML berada di dalam PHP sebagai string, dan untuk menampilkannya menggunakan tag <code>echo</code> .
5	Manakah dari dua metode yang menurut Kalian lebih mudah? Berikan jawabanmu bersama dengan alasan. <b>(Pertanyaan No 17)</b> Menurut saya metode pertama lebih praktis dan merupakan best practice dalam pengembangan web PHP, karena memisahkan logika PHP dengan presentasi HTML secara natural, sehingga kode lebih maintainable dan mudah dipahami.

## Entitas HTML

**Entitas HTML** digunakan untuk menampilkan karakter yang dicadangkan dalam HTML atau karakter yang memiliki arti khusus, seperti `<`, `>`, dan `&`. Karakter ini harus ditulis sebagai entitas untuk mencegahnya ditafsirkan sebagai kode HTML.

Entitas HTML Umum:

Nama Entitas	Nomor Entitas	Deskripsi	Hasil
<code>&amp;menyalin;</code>	<code>&amp;#169;</code>	Hak cipta	©
<code>&amp;reg;</code>	<code>&amp;#174;</code>	Terdaftar	®
<code>-</code>	<code>&amp;#8482;</code>	Merek dagang	™
<code>&amp;ampnbsp</code>	<code>&amp;#161;</code>	Ruang tanpa putus	
<code>&amp;</code>	<code>&amp;</code>	Tanda dan	&
<code>&amp;laquo;</code>	<code>&amp;#171;</code>	Kutipan sudut kiri	«
<code>&amp;raquo;</code>	<code>&amp;#187;</code>	Kutipan sudut kanan	»
<code>"</code>	<code>"</code>	Tanda kutip ganda	”
<code>'</code>	<code>'</code>	Tanda kutip tunggal	’
<code>&lt;</code>	<code>&lt;</code>	Kurang	<
<code>&gt;</code>	<code>&amp;#61;</code>	Lebih besar dari	>
<code>&amp;kali;</code>	<code>&amp;#215;</code>	Tanda perkalian	×

&membagi;	&#247;	Tanda pembagian	÷
-----------	--------	-----------------	---

## Praktikum 9. Entitas HTML

Langkah	Deskripsi
1	Buat file bernama <b>entities.php</b> di dalam direktori JS05_PHP-2, lalu ketik kode berikut:

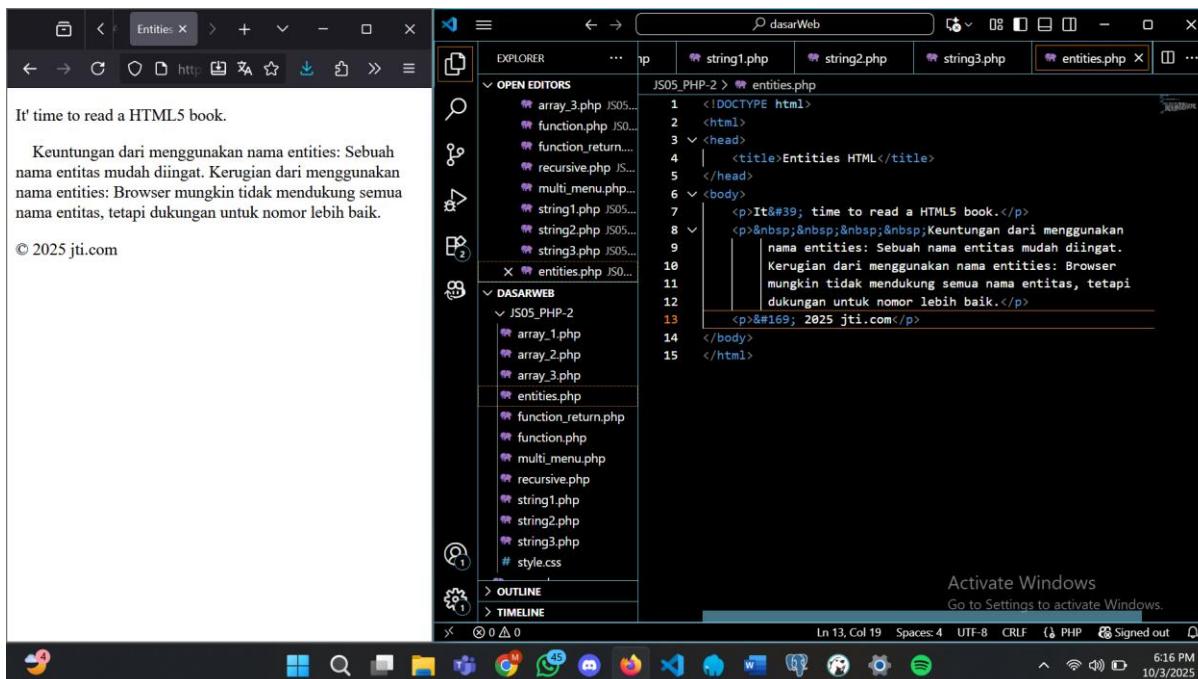
```

<!DOCTYPE html>
<html>
<head>
    <title>Entities HTML</title>
</head>
<body>
    <p>It's time to read a HTML5 book.</p>
    <p>&nbsp;&nbsp;&nbsp;Keuntungan dari menggunakan nama entities: Sebuah nama entitas mudah diingat. Kerugian dari menggunakan nama entities: Browser mungkin tidak mendukung semua nama entitas, tetapi dukungan untuk nomor lebih baik.</p>
    <p>© 2023 jti.com</p>
</body>
</html>

```

2

Amati *output* yang ditampilkan dan jelaskan hasil pengamatanmu  
(Pertanyaan No 18)



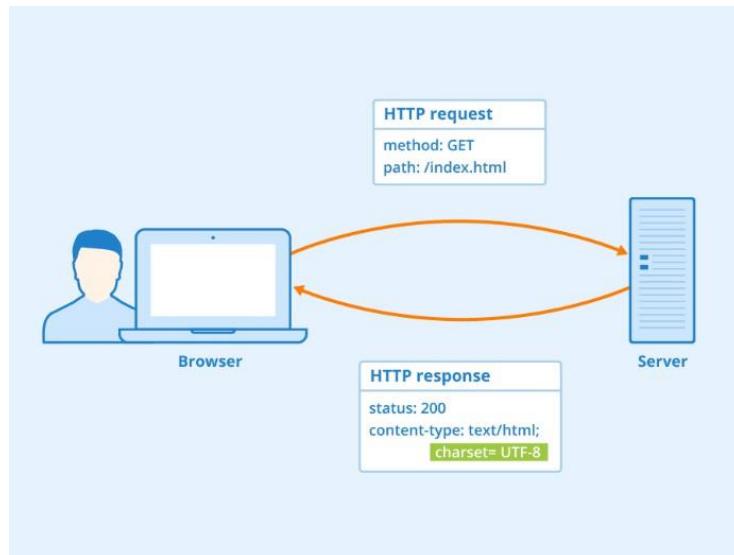
mendemonstrasikan penggunaan HTML entities untuk menampilkan karakter khusus di halaman web. Program menampilkan teks yang menjelaskan keuntungan penggunaan entities, dengan menggunakan &#39; untuk karakter apostrof () pada kata "It's", &nbsp; untuk spasi non-breaking yang menjaga format teks tetap rapi, dan &#169; untuk simbol copyright (©) pada footer "© 2025 jti.com". HTML entities sangat berguna untuk menampilkan karakter-karakter khusus yang mungkin tidak didukung oleh browser atau berpotensi konflik dengan syntax HTML, sehingga konten dapat ditampilkan dengan benar dan konsisten di berbagai browser tanpa masalah rendering.

## Header HTTP

Header HTTP adalah data yang dikirim antara browser web dan server web sebagai sarana komunikasi antara keduanya. Header HTTP berisi informasi tentang cara menangani file yang dikirim atau diminta.

Siklus **permintaan-respons** untuk halaman web: Saat kita mengakses halaman web, browser web secara otomatis mengirim permintaan HTTP ke server web. Permintaan HTTP berisi banyak informasi, salah

satunya adalah header HTTP. Di header HTTP (dikirim selama proses permintaan), ada informasi tentang file yang diminta (apakah itu file HTML, file PHP, file PDF, atau yang lainnya), serta info tambahan seperti jenis browser web yang digunakan, sistem operasi, dan alamat IP. Setelah mencapai server web, informasi di header HTTP dibaca, dan server web menyiapkan file yang diminta. Setelah itu, server web mengirim file-file tersebut kembali ke browser web. Proses pengembalian ini juga dikenal sebagai respons HTTP.



*Header HTTP*

Respons HTTP ini terdiri dari **dua bagian: header HTTP dan file web**. Header HTTP berisi informasi tentang file web yang dikirim, seperti tipe data, tanggal pengiriman, nama server web, dan sistem operasi

yang digunakan oleh server web. **File web** itu sendiri terdiri dari file HTML yang membentuk halaman web, termasuk file gambar apa pun (jika ada).

Sebagai analogi, jika presiden mengunjungi rumah Kita, akan ada tim keamanan (paspampres) yang datang sebelumnya. Mereka akan memberi tahu Kita bahwa presiden akan tiba pada waktu tertentu, dengan jumlah orang tertentu, dan memberikan informasi relevan lainnya. Tim keamanan ini dapat dibandingkan dengan header HTTP, yang tiba sebelum file yang sebenarnya dikirim. Dalam praktiknya, kita sering tidak mengetahui header HTTP, dan banyak yang bahkan mungkin belum pernah mendengar istilah tersebut. Ini normal, karena konten header HTTP dimaksudkan untuk pemrosesan browser web, bukan untuk pengunjung situs web.

Bagaimana Kita bisa melihat header HTTP di browser web Kita? Jelaskan dan sertakan langkah-langkahnya.

#### (Pertanyaan No 19)

Cara Melihat Header HTTP di Browser Web

Header HTTP dapat dilihat menggunakan Developer Tools yang tersedia di semua browser modern. Berikut langkah-langkahnya:

Metode Umum (Berlaku untuk Semua Browser):

##### 1. Buka Developer Tools

- Google Chrome / Microsoft Edge / Brave: Tekan F12 atau Ctrl + Shift + I (Windows/Linux) atau Cmd + Option + I (Mac)
- Mozilla Firefox: Tekan F12 atau Ctrl + Shift + I (Windows/Linux) atau Cmd + Option + I (Mac)
- Safari: Aktifkan dulu Developer Menu di Preferences > Advanced > "Show Develop menu", lalu tekan Cmd + Option + I

##### 2. Buka Tab Network

- Setelah Developer Tools terbuka, klik tab "Network" atau "Jaringan"
- Jika tab Network kosong, refresh halaman dengan menekan F5 atau Ctrl + R

##### 3. Pilih Request yang Ingin Dilihat

- Setelah halaman di-refresh, akan muncul daftar semua request HTTP
- Klik pada salah satu file/request (biasanya file HTML utama ada di paling atas)

##### 4. Lihat Header

- Setelah mengklik request, akan muncul panel detail di sebelah kanan atau bawah
- Cari tab "Headers" atau "Kepala"
- Di sini Anda akan melihat:
  - General: Informasi umum (URL, Status Code, Method)
  - Response Headers: Header yang dikirim server ke browser
  - Request Headers: Header yang dikirim browser ke server

## Tanggal dan Waktu

Fungsi **date()** di PHP digunakan untuk menampilkan tanggal dan waktu. Sintaks fungsi date() adalah sebagai berikut:

```
<?php  
date(format, timestamp);
```

- ✓ **format**: Parameter wajib yang menentukan bagaimana tanggal/waktu harus diformat. Ini dapat mencakup karakter untuk hari, bulan, tahun, jam, menit, dan detik.
- ✓ **timestamp**: Parameter opsional yang menentukan stempel waktu. Jika dihilangkan, tanggal dan

waktu saat ini akan digunakan.

Parameter **format** diperlukan. Parameter **format** digunakan untuk menentukan bagaimana tanggal dan/atau waktu akan diformat. Di bawah ini adalah beberapa karakter umum yang digunakan untuk pemformatan tanggal:

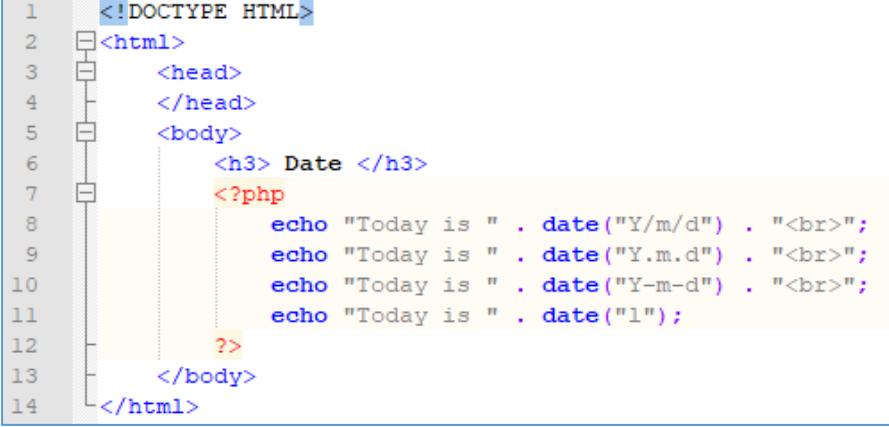
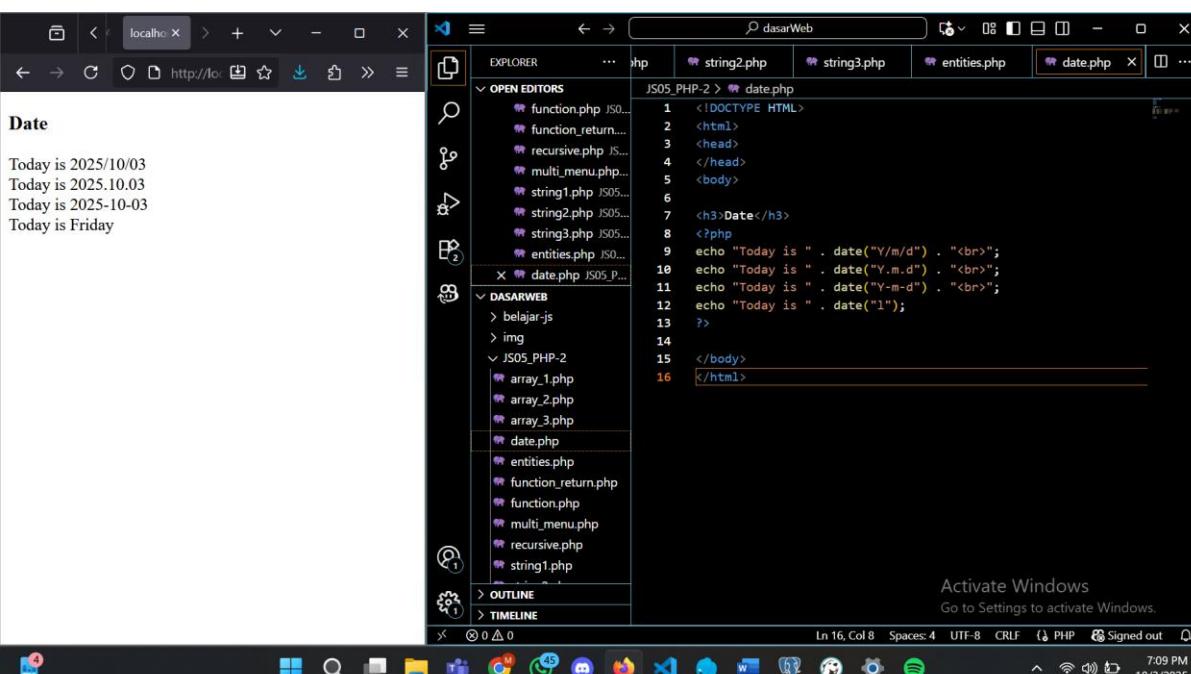
1. **d**: Mewakili hari (01 hingga 31)
2. **m**: Mewakili bulan (01 hingga 12)
3. **y**: Mewakili tahun (dalam 4 digit)
4. **l**: Mewakili hari dalam seminggu

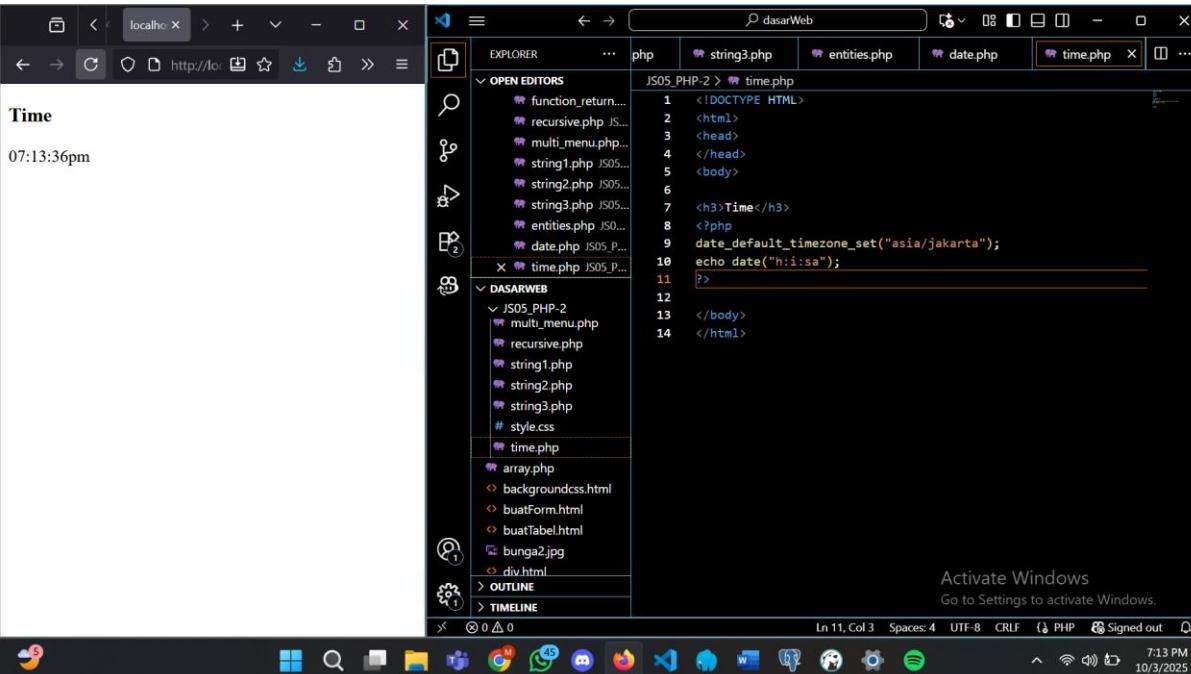
Selain menampilkan tanggal, fungsi ini juga dapat menampilkan waktu. Berikut adalah beberapa karakter umum yang digunakan untuk pemformatan waktu:

1. **H**: Mewakili jam dalam format 24 jam
2. **h**: Mewakili jam dalam format 12 jam
3. **i**: Mewakili menit (00 hingga 59)
4. **s**: Mewakili detik (00 hingga 59)
5. **a**: Mewakili ante meridiem (am) atau post meridiem (pm).

## Praktikum 10. Tanggal

Ikuti langkah-langkah ini untuk memahami cara menggunakan fungsi `date()`:

Langkah	Deskripsi
1	Buat file baru bernama <code>date.php</code> di dalam direktori JS05_PHP-2, lalu ketik kode berikut:  <pre>1 &lt;!DOCTYPE HTML&gt; 2 &lt;html&gt; 3   &lt;head&gt; 4   &lt;/head&gt; 5   &lt;body&gt; 6     &lt;h3&gt; Date &lt;/h3&gt; 7     &lt;?php 8       echo "Today is " . date("Y/m/d") . "&lt;br&gt;"; 9       echo "Today is " . date("Y.m.d") . "&lt;br&gt;"; 10      echo "Today is " . date("Y-m-d") . "&lt;br&gt;"; 11      echo "Today is " . date("l"); 12 13   &lt;/body&gt; 14 &lt;/html&gt;</pre>
2	Simpan file dan jalankan program
3	Amati <i>output</i> yang ditampilkan dan jelaskan hasil pengamatamu! <b>(Pertanyaan No 19)</b>  <p>The screenshot shows a Windows desktop environment. In the center, there is a Microsoft Edge browser window displaying the local host address. The page content shows the output of the PHP script: "Today is 2025/10/03", "Today is 2025.10.03", "Today is 2025-10-03", and "Today is Friday". To the right of the browser is a Visual Studio Code editor window. The Explorer sidebar shows several PHP files in the 'JS05_PHP-2' folder, including 'date.php'. The 'date.php' file in the editor has its code highlighted. The code itself is identical to the one shown in the first screenshot. The status bar at the bottom of the VS Code window indicates: 'Activate Windows Go to Settings to activate Windows.', 'Ln 16, Col 8 Spaces: 4 UTF-8 CRLF', and the date and time '10/3/2025 7:09 PM'.</p>
	mendemonstrasikan penggunaan fungsi <code>date()</code> untuk menampilkan tanggal dengan berbagai format. Program menampilkan tanggal yang sama (3 Oktober 2025) dalam empat format berbeda: "Y/m/d" menghasilkan format "2025/10/03" dengan tahun empat digit dan pemisah slash, "Y.m.d" menghasilkan "2025.10.03" dengan pemisah titik, "Y-m-d" menghasilkan "2025-10-03" dengan pemisah strip (format ISO standar), dan "l" menghasilkan "Friday" yang menampilkan nama hari dalam bahasa Inggris. Program ini menunjukkan fleksibilitas fungsi <code>date()</code> dalam PHP untuk memformat tanggal sesuai kebutuhan, yang sangat berguna dalam aplikasi web untuk menampilkan informasi waktu dengan cara yang user-friendly dan sesuai dengan konvensi format tanggal yang berbeda-beda.

	Buat file baru bernama <b>time.php</b> di dalam direktori JS05_PHP-2, lalu ketik kode berikut:
4	<pre> 1  &lt;!DOCTYPE HTML&gt; 2  &lt;html&gt; 3  &lt;head&gt; 4  &lt;/head&gt; 5  &lt;body&gt; 6      &lt;h3&gt; Time &lt;/h3&gt; 7      &lt;?php 8          date_default_timezone_set("asia/jakarta"); 9          echo date("h:i:s"); 10     ?&gt; 11     &lt;/body&gt; 12 &lt;/html&gt;</pre>
5	Simpan file dan jalankan program
6	<p>Amati <i>output</i> yang ditampilkan dan jelaskan hasil pengamatannya!</p> <p>(Pertanyaan No 20)</p>  <p>mendemonstrasikan cara menampilkan waktu saat ini dengan timezone yang spesifik menggunakan fungsi <code>date_default_timezone_set()</code> dan <code>date()</code>.      Program pertama-tama mengatur zona waktu ke "Asia/Jakarta" menggunakan <code>date_default_timezone_set()</code>, kemudian menampilkan waktu dalam format 12 jam dengan detik menggunakan <code>date("h:i:s")</code> yang menghasilkan output seperti "07:13:36pm". Fungsi ini sangat penting dalam aplikasi web untuk memastikan waktu yang ditampilkan sesuai dengan lokasi geografis pengguna atau server, sehingga informasi waktu yang ditampilkan akurat dan relevan dengan zona waktu yang diinginkan, terutama untuk aplikasi yang digunakan oleh pengguna dari berbagai wilayah atau negara.</p>

## Variabel Superglobal PHP

Variabel Superglobal PHP adalah variabel yang telah ditentukan sebelumnya dalam PHP yang dapat diakses dari mana saja dalam skrip. Mereka tersedia secara global, artinya mereka dapat digunakan dalam

fungsi, kelas, atau file apa pun tanpa perlu mendeklarasikannya sebagai global. Superglobal digunakan untuk menangani berbagai jenis data, seperti input formulir, detail server, informasi sesi, dan banyak lagi.

Variabel superglobal menyimpan banyak data penting dan berguna yang dapat kita gunakan untuk menyelesaikan proyek yang sedang kita kerjakan. Ada 9 variabel superglobal di PHP.

## Praktikum 11. Variabel Superglobal

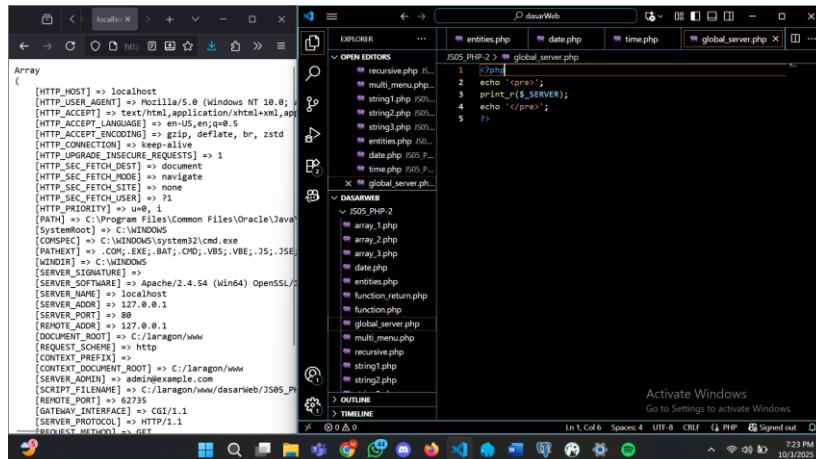
### 1. `$_SERVER`

Variabel pertama dan terpenting adalah variabel `$_SERVER`. Ini adalah *array* asosiatif yang menyediakan berbagai jenis informasi tentang permintaan yang ditangkap oleh server. Data tersebut mencakup header, jalur, lokasi skrip, dan lainnya.

Nilai yang disimpan dalam `variabel $_SERVER` disediakan oleh server web, yang berarti tidak ada jaminan khusus bahwa setiap server web yang kami gunakan akan menyediakan semua data standar yang tersedia.

Untuk mengetahui nilai apa yang tersedia dalam `variabel $_SERVER`, kita dapat menjalankan

Jalankan kode program di atas, lalu jelaskan *output* dari setiap perintah `echo`.  
**(Soal no.22)**



The screenshot shows a Windows desktop environment with a code editor window titled "desarWeb". The code editor displays a file named "global\_server.php" containing the following PHP code:

```
1 <?php
2 echo '<pre>';
3 print_r($_SERVER);
4 echo '</pre>';
5 ?>
```

The output of this code is shown in the terminal window below the code editor, displaying a large array of server information. Some key entries from the output include:

- [HTTP\_HOST] => localhost
- [HTTP\_USER\_AGENT] => Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4453.102 Safari/537.36
- [HTTP\_ACCEPT] => text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,\*/\*;q=0.8
- [HTTP\_ACCEPT\_LANGUAGE] => en-US,en;q=0.5
- [HTTP\_CONNECTION] => keep-alive
- [HTTP\_UPGRADE\_INSECURE\_REQUESTS] => 1
- [HTTP\_SEC\_FETCH\_DEST] => document
- [HTTP\_SEC\_FETCH\_MODE] => navigate
- [HTTP\_SEC\_FETCH\_SITE] => none
- [HTTP\_SEC\_FETCH\_USER] => ?1
- [HTTP\_PRIORITY] => 1
- [HTTP\_X\_AMZ\_TRACE\_ID] => 111111111111Common Files\Oracle\Java\SystemRoot\ => C:\Windows
- [COMSPEC] => C:\Windows\system32\cmd.exe
- [PATH] => .COM;EXE;BAT;CMD;VBS;VBE;.JS;.JSE
- [WINDIR] => C:\Windows
- [SERVER\_SOFTWARE] => Apache/2.4.54 (Win64) OpenSSL/1.1.1 PHP/8.1.12
- [SERVER\_NAME] => localhost
- [SERVER\_ADDR] => 127.0.0.1
- [SERVER\_PORT] => 80
- [REMOTE\_ADDR] => 127.0.0.1
- [DOCUMENT\_ROOT] => C:/laragon/www
- [REQUEST\_SCHEME] => http
- [CONTEXT\_PREFIX] => /
- [CONTEXT\_DOCUMENT\_ROOT] => C:/laragon/www
- [SERVER\_ADMIN] => admin@example.com
- [SCRIPT\_FILENAME] => C:/laragon/www/dasarweb/3505\_Php/index.php
- [HTTP\_HOST] => 127.0.0.1:6379
- [GATEWAY\_INTERFACE] => CGI/1.1
- [SERVER\_PROTOCOL] => HTTP/1.1
- [REQUEST\_METHOD] => GET

mendemonstrasikan penggunaan variabel superglobal `$_SERVER` yang menampilkan informasi lengkap tentang lingkungan server dan request HTTP. Program menggunakan `print_r($_SERVER)` untuk menampilkan seluruh isi array `$_SERVER` dalam format yang mudah dibaca, yang berisi informasi penting seperti hostname (localhost), user agent browser (Mozilla/5.0), tipe konten yang diterima, alamat IP server dan client (127.0.0.1), port (80), software server (Apache/2.4.54), document root (C:/laragon/www), script filename, protocol HTTP yang digunakan (HTTP/1.1), dan banyak informasi metadata lainnya tentang request dan environment server. Variabel `$_SERVER` sangat berguna dalam pengembangan web PHP untuk debugging, logging, keamanan, dan pengambilan keputusan berdasarkan informasi request seperti deteksi browser, validasi origin request, atau konfigurasi path aplikasi secara dinamis.

perintah berikut dan menyimpannya di `global_server.php`

```
<?php
// Menampilkan semua informasi yang tersedia dalam array $_SERVER
echo '<pre>';
print_r($_SERVER);
echo '</pre>';
?>
```

Berikut adalah beberapa contoh data dari `variabel $_SERVER` yang sering dibutuhkan:

Tidak	Variabel	Deskripsi
1	<code>\$_SERVER['PHP_SELF']</code>	Berisi nama file yang sedang dijalankan, diambil dari akar dokumen.
2	<code>\$_SERVER['SERVER_ADDR']</code>	Alamat IP server tempat file sedang dieksekusi.
3	<code>\$_SERVER['SERVER_NAME']</code>	Nama host server tempat file PHP sedang dijalankan. Nama host biasanya adalah nama PC dalam jaringan. Jika skrip PHP dijalankan pada Host Virtual, nama host virtual akan digunakan sebagai nama server.
4	<code>\$_SERVER['SERVER_PROTOCOL']</code>	Protokol komunikasi yang saat ini digunakan, seperti HTTP atau HTTPS. Contoh: 'HTTP/0.1'.
5	<code>\$_SERVER['REQUEST_METHOD']</code>	Berisi metode permintaan file PHP yang sedang dieksekusi, seperti GET, POST, PUT, DELETE, OPTIONS.
6	<code>\$_SERVER['QUERY_STRING']</code>	Mengembalikan string kueri dari file PHP yang sedang dieksekusi. Misalnya, jika pengguna mengakses
Tidak	Variabel	Deskripsi
		<code>http://localhost/halo-dunia?nama=Budi&amp;umur=20&amp;asal=Surabaya</code> , variabel ini akan mengembalikan nilai <code>nama=Budi</code> <code>umur=20</code> <code>asal=Surabaya</code>
7	<code>\$_SERVER['DOCUMENT_ROOT']</code>	Direktori akar dokumen dari file PHP yang sedang dieksekusi, dikembalikan berdasarkan pengaturan server.
8	<code>\$_SERVER['HTTP_HOST']</code>	Mengembalikan konten host, seperti header (jika tersedia).
9	<code>\$_SERVER['HTTP_REFERER']</code>	URL halaman yang merujuk ke halaman saat ini yang sedang dijalankan. Jika tidak ada, nilainya kosong.
10	<code>\$_SERVER['HTTP_USER_AGENT']</code>	Berisi informasi tentang pengguna yang membuat permintaan, termasuk browser, bahasa, dan sistem operasi. Contoh: Mozilla/4.5 [id] (X11; U; Linux 2.2.9 i586).
11	<code>\$_SERVER['REMOTE_ADDR']</code>	Alamat IP pengguna yang mengakses halaman PHP.
12	<code>\$_SERVER['SCRIPT_FILENAME']</code>	Nama jalur absolut dari file yang sedang dieksekusi.
13	<code>\$_SERVER['REQUEST_URI']</code>	URI file yang sedang dieksekusi. Contoh: "/php/halo-dunia".

## 2. `$_GET`

Variabel `$_GET` adalah *array* asosiatif yang berisi nilai dari string kueri. Misalnya, Buat file `global_get.php`, dan tulis kode sebagai berikut:

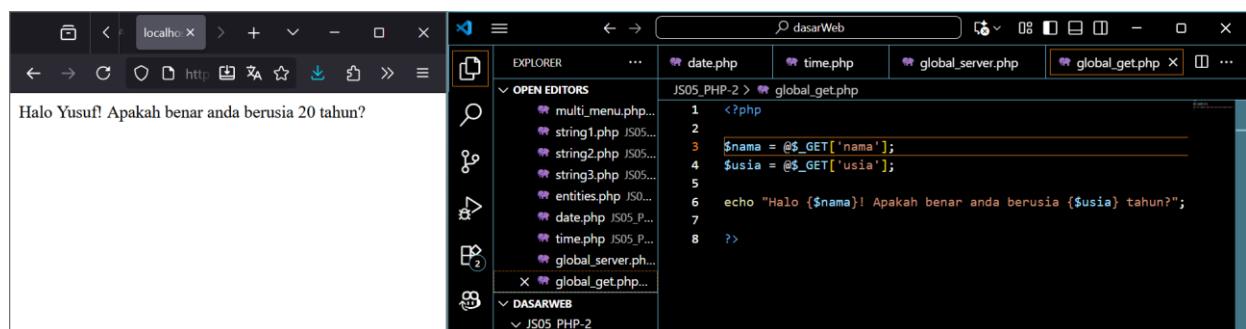
```
<?php  
  
$nama = @$_GET['nama']; //tanda @ agar tidak ada peringatan error  
ketika key-nya kosong  
$usia = @$_GET['usia']; //tanda @ agar tidak ada peringatan error  
ketika key-nya kosong  
  
echo "Halo {$nama}! Apakah benar anda berusia {$usia} tahun?";  
?>
```

Jalankan url ini di browser Kita

`localhost/dasarWeb/JS05_PHP-2/global_get.php?nama=Elok&usia=37`

Output apa yang dihasilkan, amati, dan jelaskan hasilnya.

(Pertanyaan No 23)



mendemonstrasikan penggunaan variabel superglobal `$_GET` untuk menerima data dari URL query string. Program mengambil dua parameter dari URL yaitu 'nama' dan 'usia' menggunakan `$_GET['nama']` dan `$_GET['usia']`, kemudian menampilkan pesan yang dipersonalisasi dengan format "Halo {nama}! Apakah benar anda berusia {usia} tahun?". Metode GET ini sangat umum digunakan dalam aplikasi web untuk mengirim data melalui URL, seperti pada form pencarian, filter data, atau navigasi dengan parameter. Data yang dikirim melalui GET akan terlihat di address bar browser (misalnya: `global_get.php?nama=Yusuf&usia=20`), sehingga cocok untuk data yang tidak sensitif dan perlu di-bookmark atau dibagikan, namun tidak disarankan untuk data rahasia seperti password karena terekspos di URL.

### 3. `$_POST`

Variabel `$_POST` mirip dengan variabel `$_GET`. Namun, data **tidak diteruskan melalui** string kueri di URL, melainkan di **isi permintaan**. Selain itu, metode permintaan yang digunakan harus POST.

Buat file `global_post.php`, dan tulis kode sebagai berikut:

```

<html>
<body>

<form method="post" action=<?php echo $_SERVER['PHP_SELF'];?>>
    Name: <input type="text" name="fname">
    <input type="submit">
</form>

<?php
if ($_SERVER['REQUEST_METHOD'] == "POST") {
    // collect value of input field
    $name = $_POST['fname'];
    if (empty($name)) {
        echo "Name is empty";
    } else {
        echo $name;
    }
}
?>

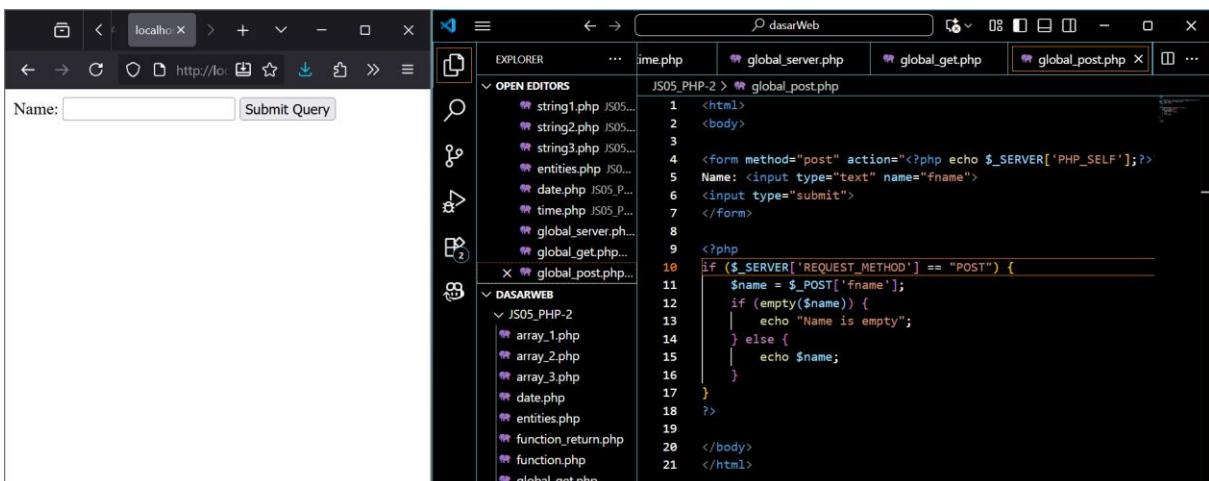
</body>
</html>

```

Jalankan url ini di browser Kita

[localhost/dasarWeb/JS05\\_PHP-2/global\\_post.php](http://localhost/dasarWeb/JS05_PHP-2/global_post.php)

Kirim formulir dan *output* apa yang dihasilkan? Amati dan jelaskan hasilnya.  
**(Pertanyaan No 24)**



mendemonstrasikan penggunaan metode POST untuk mengirim dan memproses data form secara lebih aman dibandingkan metode GET. Program membuat form HTML dengan method="post" yang mengirim data input nama ke file yang sama menggunakan \$\_SERVER['PHP\_SELF'], kemudian memproses data dengan \$\_POST['fname']. Program juga menerapkan validasi sederhana dengan fungsi empty() untuk mengecek apakah field nama kosong atau tidak, jika kosong akan menampilkan pesan "Name is empty", jika terisi akan menampilkan nilai yang diinput. Metode POST lebih aman untuk mengirim data sensitif karena data tidak terlihat di URL browser melainkan dikirim melalui HTTP request body, sehingga cocok untuk form login, registrasi, atau pengiriman data pribadi lainnya. Kondisi if(\$\_SERVER['REQUEST\_METHOD'] == "POST") memastikan kode pemrosesan hanya berjalan ketika form benar-benar di-submit, mencegah error saat halaman pertama kali dibuka.

#### 4. **`$_SESSION`**

Variabel **`$_SESSION`** adalah *array* asosiatif yang menyimpan data sesi pengguna. Variabel ini dapat digunakan untuk menyimpan pengguna yang masuk untuk sesi tertentu. Ini juga dapat digunakan untuk menyimpan data keranjang di toko online. Secara default, masa pakai sesi di PHP adalah 1440 detik atau 24 menit.

#### 5. **`$_COOKIE`**

Mirip dengan **`$_SESSION`**, variabel **`$_COOKIE`** dapat digunakan untuk menyimpan data yang terkait dengan pengguna, seperti informasi login, detail keranjang di toko online, dan sebagainya. Perbedaannya adalah bahwa cookie adalah file kecil yang disimpan di browser pengguna. File ini dikirim setiap kali browser membuat permintaan ke server. Masa pakai cookie umumnya lebih lama daripada sesi.

#### 6. **`$_REQUEST`**

Variabel **`$_REQUEST`** adalah *array* asosiatif yang menyimpan gabungan nilai dari variabel **`$_GET`**, **`$_POST`**, dan **`$_COOKIE`** yang kesemuanya berhubungan dengan data yang dikirim bersamaan dengan *request* user.

Buat file **`global_request.php`**, dan tulis kode sebagai berikut:

```

<html>
<body>

<form method="post" action=<?php echo $_SERVER['PHP_SELF'];?>>
  Name: <input type="text" name="fname">
  <input type="submit">
</form>

<?php
if ($_REQUEST["REQUEST_METHOD"] == "POST") {
  // collect value of input field
  $name = $_REQUEST['fname'];
  if (empty($name)) {
    echo "Name is empty";
  } else {
    echo $name;
  }
}

</body>
</html>

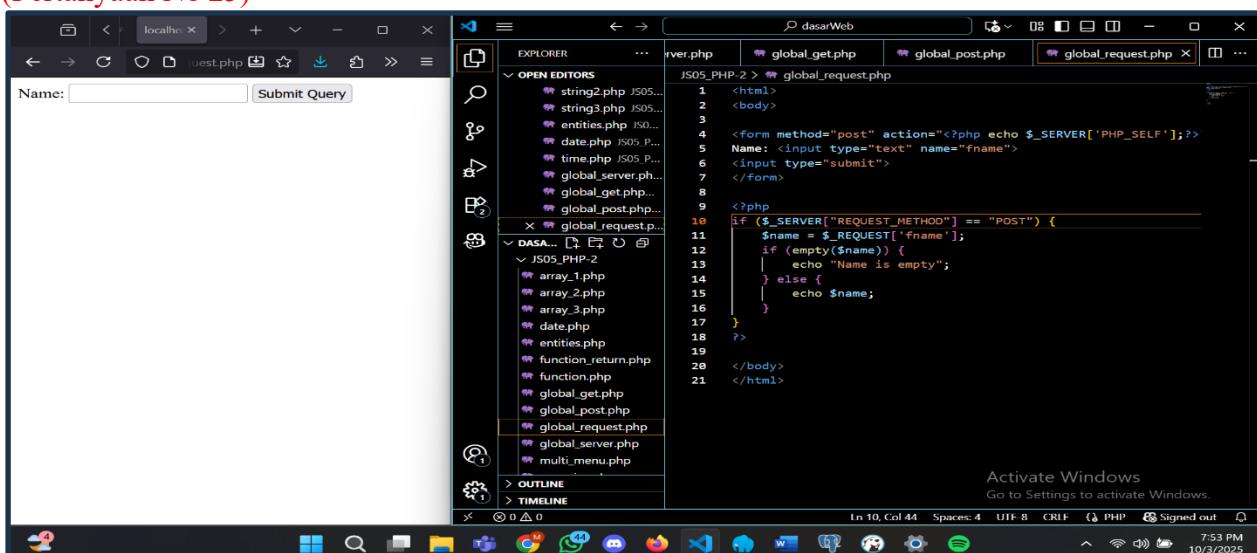
```

Jalankan url ini di browser Kita

[localhost/dasarWeb/JS05\\_PHP-2/global\\_request.php](http://localhost/dasarWeb/JS05_PHP-2/global_request.php)

Kirim formulir dan *output* apa yang dihasilkan? Amati dan jelaskan hasilnya. Dan apa bedanya dengan variabel global **\$\_POST**?

(Pertanyaan No 25)



Program ini menampilkan form HTML yang meminta input nama melalui field bernama fname. Ketika form dikirimkan, data yang dimasukkan akan diambil menggunakan variabel superglobal **\$\_REQUEST**. Kemudian, program akan memeriksa apakah input tersebut kosong. Jika kosong, maka akan ditampilkan pesan "Name is empty", sedangkan jika diisi, nama yang dimasukkan akan ditampilkan ke layar. Penggunaan **\$\_REQUEST** memungkinkan pengambilan data dari berbagai metode pengiriman (GET, POST, COOKIE), meskipun dalam kasus ini hanya metode POST yang digunakan. Program ini cocok sebagai latihan dasar dalam memahami cara kerja form handling dan validasi input di PHP.

## 7. **\$\_FILES**

Variabel **\$\_FILES** adalah *array* asosiatif yang menyimpan data tentang file yang diunggah oleh pengguna dalam satu permintaan menggunakan **metode POST** atau **PUT**.

## 8. **\$\_ENV**

Variabel `$_ENV` adalah *array* asosiatif yang berisi data tentang lingkungan tempat skrip PHP berjalan. Variabel `$_ENV` disediakan oleh shell yang menjalankan skrip PHP, sehingga nilainya dapat bervariasi tergantung pada sistem operasi yang digunakan.

Dalam kerangka kerja PHP modern seperti Laravel, `variabel $_ENV` juga digunakan untuk menyimpan informasi terkait lingkungan, seperti nama database, kata sandi database, dan nilai lain yang diperlukan untuk mengonfigurasi kerangka kerja.

#### 9. `$GLOBALS`

Variabel `$GLOBALS` adalah *array* asosiatif yang menyimpan semua variabel global yang ditentukan saat program berjalan. Variabel `$GLOBALS` adalah superglobal PHP yang digunakan untuk mengakses variabel global dari mana saja dalam skrip PHP (termasuk fungsi atau metode di dalam).

Buat file `global_globals.php`, dan tulis kode sebagai berikut:

```

<?php
$x = 75;
$y = 25;

function addition() {
    $GLOBALS['z'] = $GLOBALS['x'] + $GLOBALS['y'];
}

addition();
echo $z;
?>

```

Jalankan url ini di browser Kita

[localhost/dasarWeb/JS05\\_PHP-2/global\\_globals.php](http://localhost/dasarWeb/JS05_PHP-2/global_globals.php)

Kirim formulir dan *output* apa yang dihasilkan? Amati dan jelaskan hasilnya.  
**(Pertanyaan No 26)**

The screenshot shows a code editor interface with several files listed in the sidebar under 'OPEN EDITORS'. The main editor window displays the following PHP code:

```

<?php
$x = 75;
$y = 25;

function addition() {
    $GLOBALS['z'] = $GLOBALS['x'] + $GLOBALS['y'];
}

addition();
echo $z;
?>

```

merupakan contoh penggunaan variabel superglobal **\$GLOBALS** dalam PHP. Pada script tersebut, terdapat dua variabel global yaitu **\$x** dengan nilai 75 dan **\$y** dengan nilai 25. Di dalam fungsi **addition()**, kedua variabel ini dijumlahkan menggunakan array **\$GLOBALS**, dan hasilnya disimpan ke dalam **\$GLOBALS['z']**. Variabel **\$z** kemudian dicetak di luar fungsi dengan **echo \$z;**, yang menghasilkan output **100** di browser. Kesimpulannya, kode ini menunjukkan bahwa superglobal **\$GLOBALS** memungkinkan akses dan manipulasi variabel global dari dalam fungsi tanpa perlu menggunakan global keyword, dan dapat digunakan untuk berbagi data lintas lingkup dalam program PHP.

## Referensi:

- 1) Nixon, Robin. (2018). Belajar PHP, MySQL, JavaScript, CSS & HTML: Panduan Langkah demi Langkah untuk Membuat Situs Web Dinamis, Edisi ke-5. O'Reilly Media, Inc.
- 2) Forbes, Alan. (2012). Kegembiraan PHP: Panduan Pemula untuk Pemrograman Aplikasi Web Interaktif dengan PHP dan MySQL, Edisi <sup>ke-5</sup>. Penerbitan Pulau Plum