

LAPORAN JOBSHEET 6

DESAIN DAN PEMROGRAMAN WEB



Disusun Oleh:
Mohamad Dona Haryanto
22431760127 – SIB 2A

PROGRAM STUDI D-IV TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2023



Topik

1. konsep *function* pada pemrograman PHP
2. konsep penggabungan pada pemrograman PHP dan HTML

Tujuan

Mahasiswa diharapkan mampu:

1. Mahasiswa paham dengan konsep *function* pada pemrograman PHP
2. Mahasiswa paham dengan konsep penggabungan pada pemrograman PHP dan HTML

Pendahuluan

Pengenalan Array dan Fungsi

Array atau larik adalah salah satu tipe data. Array bukan merupakan tipe data dasar seperti integer atau boolean, tetapi array adalah sebuah tipe data yang terdiri dari kumpulan tipe data lainnya. Array memudahkan dalam membuat kelompok data, menghemat penulisan, dan penggunaan variabel. Di dalam PHP terdapat 3 jenis array, yaitu *indexed array*, *associative array*, dan *multidimensional array*.

Selain tipe data, pemahaman tentang fungsi adalah salah satu yang dibutuhkan ketika membuat program. Dalam bahasa pemrograman, fungsi didefinisikan sebagai kode program yang dirancang untuk menyelesaikan sebuah tugas tertentu, dan merupakan bagian dari program utama. Ketika merancang kode program kadang kita sering membuat kode yang melakukan tugas yang sama secara berulang-ulang, seperti membaca tabel dari database, menampilkan penjumlahan, dan lain-lain. Tugas yang sama ini akan lebih efektif jika dipisahkan dari program utama, dan dirancang menjadi sebuah fungsi.

Praktikum Bagian 1. *Indexed Array*

Indexed array adalah array dengan indeks numerik. Penulisan *indexed array* bisa dilakukan secara otomatis atau diberikan indeks secara manual. Indeks array selalu dimulai dari 0. Berikut adalah cara penulisan *indexed array*:

```
<?php
    $variable = array(
        "value0",
        "value1",
        "value2",
        :
        "value-n"
    );
?>
```

atau bisa juga dituliskan seperti berikut:

```
<?php
    $variable[0] = "value0";
    $variable[1] = "value1";
```

```
$variable[2] = "value2";
:
$variable[n] = "value-n";
?>
```

Ikuti langkah-langkah berikut untuk memahami *indexed* array di dalam PHP:

Langkah	Keterangan
1	<p>Buat file baru dengan nama <code>array_1.php</code> di dalam direktori <code>praktik_php</code>, kemudian ketikkan kode berikut:</p> <pre><!DOCTYPE html> <html> <head> </head> <body> <h2>Array Terindeks</h2> <?php \$Listdosen=["Elok Nur Hamdana","Unggul Pamenang", "Bagas Nugraha"]; echo \$Listdosen[2] . "
"; echo \$Listdosen[0] . "
"; echo \$Listdosen[1] . "
"; ?> </body> </html></pre>
2	Simpan file dan jalankan kode program dengan mengetikkan <code>localhost/dasarWeb/praktik_php/array_1.php</code>
3	Amati hasil yang ditampilkan
4	<p>Untuk menampilkan array, selain menggunakan indeks kita juga bisa menggunakan perulangan. Coba tampilan hasil dari kode program diatas dengan menggunakan perulangan. Tampilkan kode program dan hasilnya. (soal no.1)</p> <pre>for(\$i = 0; \$i < count(\$Listdosen); \$i++){ echo \$Listdosen[\$i] . "
"; }</pre> 

Komponen *associative* array terdiri dari pasangan kunci (*key*) dan nilai (*value*). Kunci menunjukkan posisi dimana nilai disimpan. PHP menggunakan tanda panah (=>) untuk mendefinisikan nilai kepada kunci. Berikut adalah cara penulisan *associative* array:

```
<?php
    $variable = array(
        "Key0" => "value0",
        "key1" => "value1",
        "key2" => "value2",
        :
        "key-n" => "value-n"
    );
?>
```

atau bisa juga ditulis seperti berikut:

```
<?php
    $variable['key0'] = "value0";
    $variable['key1'] = "value1";
    $variable['key2'] = "value2";
    :
    $variable['key-n'] = "value-n";
?>
```

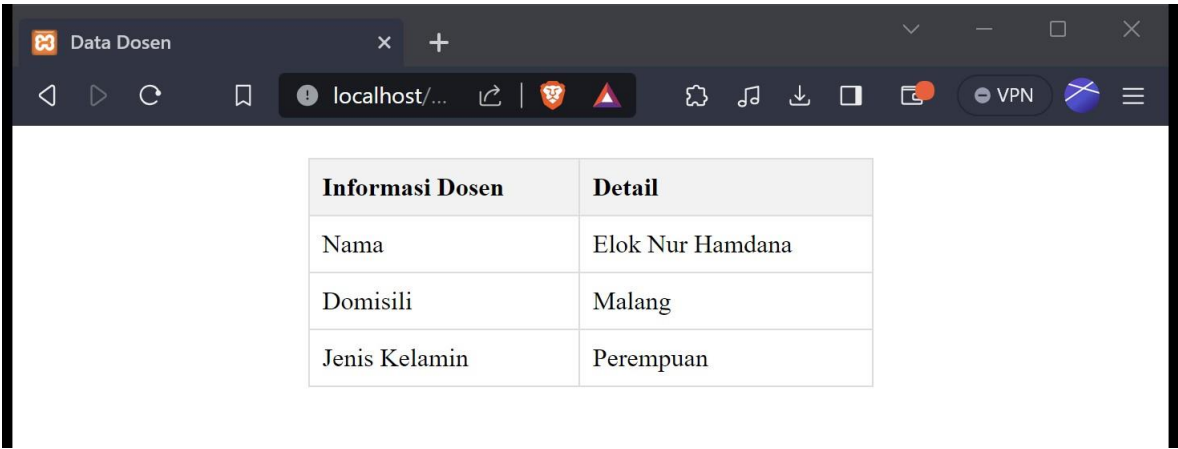
Ikuti langkah-langkah berikut untuk memahami *associative* array di dalam PHP:

Langkah	Keterangan
1	<p>Buat file baru dengan nama <code>array_2.php</code> di dalam direktori <code>praktik_php</code>, kemudian ketikkan kode berikut:</p> <pre><!DOCTYPE html> <html> <head> <meta charset="utf-8"> <meta name="viewport" content="width=device-width, initial-scale=1"> <title></title> </head> <body> <?php \$Dosen = ['nama' => 'Elok Nur Hamdana', 'domisili' => 'Malang', 'jenis_kelamin' => 'Perempuan']; echo "Nama : {\$Dosen ['nama']}
"; echo "Domisili : {\$Dosen ['domisili']}
"; echo "Jenis Kelamin : {\$Dosen ['jenis_kelamin']}
"; ?> </body> </html></pre>
2	<p>Simpan file dan jalankan kode program dengan mengetikkan <code>localhost/dasarWeb/praktik_php/array_2.php</code></p>
3	<p>Amati hasil yang ditampilkan. Kemudian tambahkan style tabel pada output tampilan tersebut supaya lebih menarik. (soal no.2) *Untuk penggunaan style bebas boleh internal atau eksternal file.</p>

Source Code

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <title>Data Dosen</title>
7     <style>
8         table {
9             border-collapse: collapse;
10            width: 50%;
11            margin: 20px auto;
12        }
13
14        th, td {
15            border: 1px solid #ddd;
16            padding: 8px;
17            text-align: left;
18        }
19
20        th {
21            background-color: #f2f2f2;
22        }
23    </style>
24 </head>
25 <body>
26     <?php
27         $Dosen = [
28             'nama' => 'Elok Nur Hamdana',
29             'domisili' => 'Malang',
30             'jenis_kelamin' => 'Perempuan'
31         ];
32     >
33
34     <table>
35         <tr>
36             <th>Informasi Dosen</th>
37             <th>Detail</th>
38         </tr>
39         <tr>
40             <td>Nama</td>
41             <td><?php echo $Dosen['nama']; ?></td>
42         </tr>
43         <tr>
44             <td>Domisili</td>
45             <td><?php echo $Dosen['domisili']; ?></td>
46         </tr>
47         <tr>
48             <td>Jenis Kelamin</td>
49             <td><?php echo $Dosen['jenis_kelamin']; ?></td>
50         </tr>
51     </table>
52 </body>
53 </html>
```

Output

	
--	--

Praktikum Bagian 3. *Multidimensional Array*

Multidimensional array adalah array yang terdiri dari satu atau lebih array. Berikut adalah contoh penulisan array dengan dimensi 2:

```
<?php
$variable = array(
    array ("value00", "value01", .., "value-nm"),
    array ("value10", "value11", .., "value-nm"),
    array ("value20", "value21", .., "value-nm"),
);
?>
```

Ikuti langkah-langkah berikut untuk memahami multidimensional array di dalam PHP:

Langka h	Keterangan
1	<p>Buat file baru dengan nama <code>style.css</code> di dalam direktori <code>praktik_php</code>, kemudian ketikkan kode berikut:</p> <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	<p>Buat file baru dengan nama <code>array_3.php</code> di dalam direktori <code>praktik_php</code>, kemudian ketikkan kode berikut:</p>

	<pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 <link rel="stylesheet" type="text/css" href="style.css"/> 5 </head> 6 <body> 7 <h2> Multidimensional Array </h2> 8 <table> 9 <tr> 10 <th>Judul Film</th> 11 <th>Tahun</th> 12 <th>Rating</th> 13 </tr> 14 <?php 15 \$movie = array(16 array("Avengers: Invinity 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 "<tr>"; 22 echo "<td>". \$movie[0][0] . "</td>"; 23 echo "<td>". \$movie[0][1] . "</td>"; 24 echo "<td>". \$movie[0][2] . "</td>"; 25 echo "</tr>"; 26 echo "<tr>"; 27 echo "<td>". \$movie[1][0] . "</td>"; 28 echo "<td>". \$movie[1][1] . "</td>"; 29 echo "<td>". \$movie[1][2] . "</td>"; 30 echo "</tr>"; 31 echo "<tr>"; 32 echo "<td>". \$movie[2][0] . "</td>"; 33 echo "<td>". \$movie[2][1] . "</td>"; 34 echo "<td>". \$movie[2][2] . "</td>"; 35 echo "</tr>"; 36 echo "<tr>"; 37 echo "<td>". \$movie[3][0] . "</td>"; 38 echo "<td>". \$movie[3][1] . "</td>"; 39 echo "<td>". \$movie[3][2] . "</td>"; 40 echo "</tr>"; 41 ?> 42 </table> 43 </body> 44 </html> </pre>
3	<p>Simpan file dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/array_3.php</p>
4	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 3)</p>

Document

localhost/...

VPN

Multidimensional Array

Judul Film	Tahun	Rating
Anvengers: Invinity War	2018	8.7
The Avengers	2012	8.1
Guardians of the Galaxy	2014	8.1
Iron Man	2008	7.9

Dari hasil yang dihasilkan pada gambar diatas. Kita bisa melihat bahwa kita bisa menggunakan array dua dimensi dengan dengan membuat sebuah array didalam array. Lalu, kita tampilkan berdasarkan indeks yang kita inginkan ditambah dengan penggunaan tabel agar tampilan terlihat lebih menarik.

Fungsi

Banyak fungsi *build-in* dari php yang sering kita gunakan, seperti `print()`, `print_r()`, `unset()`, dll. Selain fungsi-fungsi tersebut, kita juga dapat membuat fungsi sendiri sesuai kebutuhan.

Fungsi adalah sekumpulan intruksi yang dibungkus dalam sebuah blok. Fungsi dapat digunakan ulang tanpa harus menulis ulang instruksi di dalamnya.

Fungsi pada PHP dapat dibuat dngan kata kunci *function*, lalu diikuti dengan nama fungsinya.

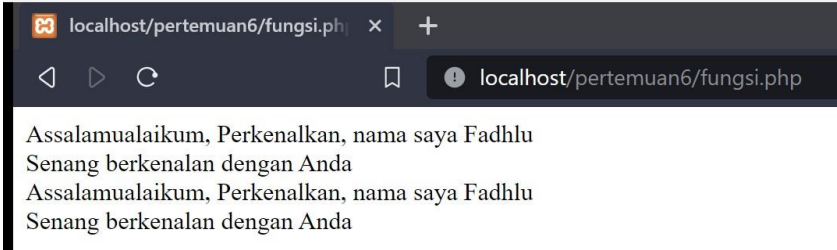
Contoh:

```
function namaFungsi(){
    //...
}
```

Kode instruksi dapat di tulis di dalam kurung kurawal ({...})

Penamaan fungsi di dalam PHP diawali dengan huruf atau tanda garis bawah dan tidak boleh diawali dengan angka. Penulisan nama fungsi ini tidak *case-sensitive*. Ikuti langkah-langkah berikut untuk memahami penggunaan fungsi di dalam PHP:

Langkah	Keterangan
1	<p>Buatlah satu file baru di dalam direktori praktik_php , beri nama fungsi.php</p> <pre><?php function perkenalan(){ echo "Assalamualaikum, "; echo "Perkenalkan, nama saya Elok
"; //Tulis sesuai nama kalian echo "Senang berkenalan dengan Anda
"; } //memanggil fungsi yang sudah dibuat perkenalan(); ?></pre>

2	<p>Simpan file dan jalankan kode program dengan cetak sebanyak 2 kali, Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 4)</p>  <p>Hasil yang didapatkan adalah teks yang dimasukkan ke dalam function atau method atau fungsi ketika dipanggil nama fungsi tersebut. Maka, akan ditampilkan secara keseluruhan dari isi fungsi tersebut.</p>
---	---

Fungsi dengan Parameter

3	<p>Supaya instruksi yang di dalam fungsi lebih dinamis, kita dapat menggunakan parameter untuk memasukkan sebuah nilai ke dalam fungsi. Nilai tersebut akan diolah di dalam fungsi. Misalkan, pada contoh fungsi yang tadi, tidak mungkin nama yang dicetak adalah <i>elok</i> saja dan salam yang dipakai tidak selalu <i>assalamualaikum</i>.</p>
4	<p>Tambahkan parameter seperti pada kode program berikut ini:</p> <pre data-bbox="277 1048 855 1514"><?php //membuat fungsi function perkenalan(\$nama, \$salam){ echo \$salam.", "; echo "Perkenalkan, nama saya ".\$nama."
"; echo "Senang berkenalan dengan Anda
"; } //memanggil fungsi yang sudah dibuat perkenalan("Hamdana","Hallo"); echo "<hr>"; \$saya = "Elok"; \$ucapanSalam = "Selamat pagi"; //memanggil lagi perkenalan(\$saya,\$ucapanSalam); ?></pre>
5	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 5)</p> <div data-bbox="502 1626 1034 1702"> <p>Halo, Perkenalkan, nama saya Hamdana Senang berkenalan dengan Anda</p> <hr/> </div> <div data-bbox="502 1736 1106 1812"> <p>Selamat Pagi, Perkenalkan, nama saya Fadhlu Senang berkenalan dengan Anda</p> </div> <p>Dari penerapan fungsi diatas kita bisa memanfaatkan fungsi dengan parameter langsung menggunakan value dari variabelnya dan kita menggunakan variabel nya. Namun, dari kedua cara tersebut akan menghasilkan output yang sama. Hanya berbeda cara pemanggilannya saja.</p>

Parameter dengan Nilai Default

6	<p>Nilai <i>default</i> dapat kita berikan di parameter. Nilai <i>default</i> berfungsi untuk mengisi nilai sebuah parameter, kalau parameter tersebut tidak diisi nilainya.</p> <p>Misalnya: lupa mengisi parameter salam, maka program akan <i>error</i>. Oleh karena itu, kita perlu memberikan nilai <i>default</i> supaya tidak error.</p>
7	<p>Ketikkan kode program berikut</p> <pre><?php //membuat fungsi function perkenalan(\$nama, \$salam="Assalamualaikum"){ echo \$salam.", "; echo "Perkenalkan, nama saya ".\$nama."
"; echo "Senang berkenalan dengan Anda
"; } //memanggil fungsi yang sudah dibuat perkenalan("Hamdana","Hallo"); echo "<hr>"; \$saya = "Elok"; \$ucapanSalam = "Selamat pagi"; //memanggil lagi tanpa mengisi parameter salam perkenalan(\$saya); ?></pre>
8	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 6)</p> <div style="border: 1px solid black; padding: 10px; margin: 10px 0;"> <p>Hallo, Perkenalkan, nama saya Hamdana Senang berkenalan dengan Anda</p> <hr/> <p>Assalamualaikum, Perkenalkan, nama saya Elok Senang berkenalan dengan Anda</p> </div> <p>Dari hasil tersebut kita bisa memanfaatkan fungsi tanpa harus memiliki parameter yang diinput atau kita bisa menggunakan nilai default. Sehingga pada parameter bila ada yang tidak diinput kita masih mempunyai nilai default sehingga ketika pada pemanggilan tidak error dan masih bisa berjalan.</p>
Fungsi yang Mengembalikan Nilai	
9	<p>Hasil pengolahan nilai dari fungsi mungkin saja kita butuhkan untuk pemrosesan berikutnya. Oleh karena itu, kita harus membuat fungsi yang dapat mengembalikan nilai.</p> <p>Pengembalian nilai dalam fungsi dapat menggunakan kata kunci return.</p>
10	<p>Ketikkan kode program berikut</p> <pre><?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 ?></pre>
11	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 7)</p> <div style="border: 1px solid black; padding: 10px; margin: 10px 0;"> <p>Umur saya adalah 19 tahun</p> </div>

	<p>Dari hasil yang didapatkan dari kode program diatas. Kita bisa mengembalikan nilai dari sebuah method atau mengambil nilai dari method tersebut untuk kita tampilkan. Pada saat ingin menampilkan hasil nilai method tersebut kita hanya perlu memanggil method atau fungsi tersebut jika memang digunakan parameter maka kita masukkan parameter nya pada fungsi tersebut. Maka, nilai dari fungsi tersebut akan ditampilkan. Konsep pemanggilan fungsi ini sama seperti kita memanggil variabel tertentu untuk menampilkan nilai dari variabel itu sendiri.</p>
Memanggil Funs di dalam fungsi	
12	Fungsi yang sudah kita buat, dapat juga dipanggil di dalam fungsi lain.
13	<p>Ketikkan kode program berikut</p> <pre> <?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."
"; //memanggil fungsi lain echo "Saya berusia ". hitungUmur(1988, 2023) ." tahun
"; echo "Senang berkenalan dengan anda
"; } //memanggil fungsi perkenalan perkenalan ("Elok"); ?> </pre>
14	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 8)</p> <div style="border: 1px solid black; padding: 10px; margin: 10px 0;"> <p>Assalamualaikum,Perkenalkan, nama saya Fadhlu Saya berusia 19 tahun Senang berkenalan dengan Anda</p> </div> <p>Dari hasil yang didapatkan pada kode program diatas. Kita bisa mengetahui bahwa fungsi juga bisa saling memanggil. Ketika pada suatu method atau fungsi menginginkan nilai yang dibutuhkan untuk ditampilkan atau diproses kita bisa memanfaatkan fungsi lain yang memang memiliki nilai yang kita butuhkan untuk ditampilkan.</p>

Fungsi Rekursif

Fungsi rekursif adalah fungsi yang memanggil dirinya sendiri. Fungsi ini biasanya digunakan untuk menyelesaikan masalah seperti faktorial, bilangan fibbonaci, pemrograman dinamis

Langkah	Keterangan
1	Buat file baru dengan nama <code>rekursif.php</code> di dalam direktori <code>praktik_php</code> , kemudian ketikkan kode berikut:

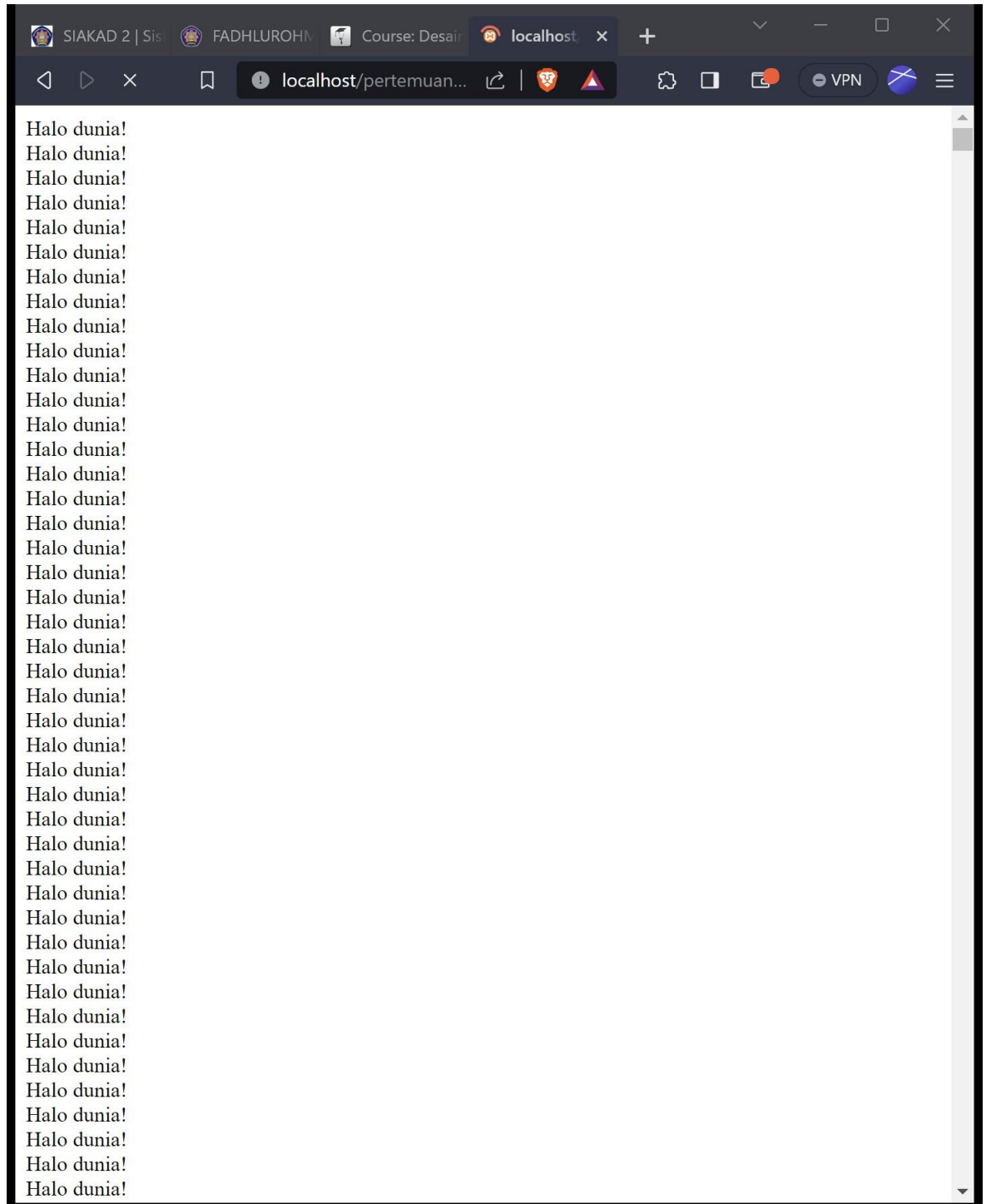
```
<?php
function tampilkanHaloDunia(){
    echo "Halo dunia! <br>";

    tampilkanHaloDunia();
}

tampilkanHaloDunia();
?>
```

Jika kode program di atas dijalankan, apa yang akan terjadi dan bagaimana dampaknya jika itu di lakukan kemukakan pendapat kalian! (soal no 9)

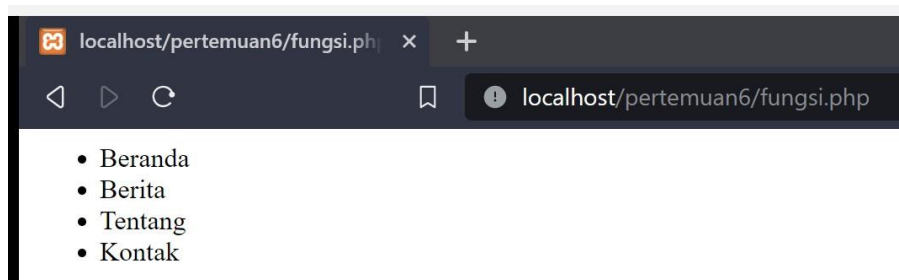
2



Akan terjadi looping secara terus-menerus tanpa henti. Dikarenakan pada fungsi tersebut meanggil dirinya sendiri tanpa diketahui batas pemanggilannya. Maka program akan terus berjalan dan tidak akan berhenti.

3	<p>Untuk menampilkan angka 1 sampai 25, kita bisa dengan mudah menggunakan perulangan for seperti berikut:</p> <pre data-bbox="277 219 821 409"><?php for (\$i=1; \$i <=25; \$i++){ echo "Perulangan ke-{\$i}
"; } ?></pre>
4	<p>Akan tetapi jika kita ingin menggunakan konsep fungsi rekursif untuk menjalankan tugas yang sama, ketikkan kode program berikut.</p> <pre data-bbox="277 506 1083 813"><?php function tampilkanAngka (int \$jumlah, int \$indeks = 1) { echo "Perulangan ke-{\$indeks}
"; //panggil diri sendiri selama \$indeks <= \$jumlah if (\$indeks < \$jumlah) { tampilkanAngka(\$jumlah, \$indeks + 1); } } tampilkanAngka(20); ?></pre>
5	<p>Jalankan kode program di atas dan bagaimana outputnya kemudian jelaskan kenapa bisa seperti itu. (soal no 10)</p> <div data-bbox="724 965 957 1704" data-label="Text"> <pre> Perulangan ke-1 Perulangan ke-2 Perulangan ke-3 Perulangan ke-4 Perulangan ke-5 Perulangan ke-6 Perulangan ke-7 Perulangan ke-8 Perulangan ke-9 Perulangan ke-10 Perulangan ke-11 Perulangan ke-12 Perulangan ke-13 Perulangan ke-14 Perulangan ke-15 Perulangan ke-16 Perulangan ke-17 Perulangan ke-18 Perulangan ke-19 Perulangan ke-20 </pre> </div> <p>Dari hasil yang didapatkan, fungsi rekursif ini sama seperti dengan perulangan. Namun, yang membedakannya adalah dalam fungsi ini dia memanggil dirinya sendiri dan untuk mengupdate nilai penentu nya didalam parameternya harus diterapkan sebuah rumus agar nilai penentu terus diupdate hingga mencapai nilai batas.</p>

Contoh Kasus Menu Bertingkat

Langkah	Keterangan
1	<p>Buat variabel \$menu. Variable ini adalah gabungan antara <i>array</i> terindeks dan <i>array</i> asosiatif multidimensi. Dikatakan multidimensi karena ia adalah suatu <i>array</i> yang memiliki <i>array</i> lain di dalamnya.</p> <p>Selanjutnya kita akan coba menampilkan semua item dari <i>array</i> \$menu menggunakan fungsi rekursif.</p>
2	<p>Buatlah kode program untuk variabel \$menu berikut</p> <pre><?php \$menu = [["nama" => "Beranda"], ["nama" => "Berita", "subMenu" => [["nama" => "Wisata", "subMenu" => [["nama" => "Pantai"], ["nama" => "Gunung"]]]], ["nama" => "Kuliner"], ["nama" => "Hiburan"]], ["nama" => "Tentang"], ["nama" => "Kontak"]];</pre>
3	<p>Kemudian buatlah fungsi untuk menampilkan <i>array</i> utama</p> <pre>function tampilkanMenuBertingkat (array \$menu) { echo ""; foreach (\$menu as \$key => \$item) { echo "{\$item['nama']}"; } echo ""; } tampilkanMenuBertingkat(\$menu); ?></pre>
4	<p>Jalankan program diatas dan bagaimana hasil outputnya (soal no 11)</p> 

	<p>Dari hasil program diatas kita dapat melihat pemanggilan masing-masing array utama dengan menggunakan perulangan 1 tingkat. Sehingga hanya array utama saja yang ditampilkan lalu, dibuat menggunakan list seperti pada gambar diatas.</p>
5	<p>Selanjutnya buatlah fungsi di atas menjadi rekursif dengan memanggil dirinya sendiri ketika suatu item dari menu memiliki atribut subMenu. Sehingga tampilanya menjadi seperti berikut. (soal no 12)</p> <div data-bbox="284 358 705 761" data-label="List-Group"> <ul style="list-style-type: none"> • Beranda • Berita <ul style="list-style-type: none"> ◦ Wisata <ul style="list-style-type: none"> ▪ Pantai ▪ Gunung ◦ Kuliner ◦ Hiburan • Tentang • Kontak </div> <p>Source Code</p> <pre>function tampilkanMenuBertingkat(array \$menu){ echo ""; foreach (\$menu as \$key => \$item){ echo "{\$item['nama']}"; // Periksa apakah ada submenu dan apakah submenu tersebut adalah array if (isset(\$item['subMenu']) && is_array(\$item ['subMenu'])) { // Panggil fungsi rekursif untuk menampilkan submenu tampilkanMenuBertingkat(\$item ['subMenu']); } echo ""; } echo ""; } tampilkanMenuBertingkat(\$menu);</pre>

String

String tipe data karakter yang biasanya diekspresikan dengan diapit oleh tanda petik ganda (“ ”) atau petik tunggal (‘ ’). Keduanya hampir sama akan tetapi cara kerjanya sedikit berbeda. Tanda petik dua akan mengisi variabel dengan nilai aslinya, ada pun dengan tanda petik satu, kita tidak bisa melakukan hal tersebut.

Contoh strings adalah “Hello world!”. Beberapa operasi dapat dilakukan pada data bertipe string. PHP menyediakan fungsi-fungsi *built-in* yang siap untuk digunakan dalam operasi string, yaitu:

Fungsi	Keterangan
<code>strlen()</code>	Untuk mengetahui panjang string
<code>str_word_count()</code>	Untuk mengetahui jumlah kata di dalam string
<code>strpos()</code>	Untuk mengetahui posisi suatu string dalam string
<code>strrev()</code>	Untuk membalik urutan string
<code>strstr()</code>	Untuk mencari substring suatu string
<code>substr()</code>	Untuk mengambil substring dari posisi awal dan akhir dalam sebuah string
<code>trim()</code>	Menghilangkan karakter spasi di awal dan akhir string
<code>ltrim()</code>	Menghilangkan karakter spasi di awal string
<code>rtrim()</code>	Menghilangkan karakter spasi di akhir string
<code>strtoupper()</code>	Mengubah huruf menjadi huruf capital
<code>strtolower()</code>	Mengubah huruf menjadi huruf kecil (<i>lowercase</i>)
<code>str_replace()</code>	Menggantikan nilai dari beberapa bagian dari string dengan string yang lain
<code>ucwords()</code>	Mengubah huruf awal dari sebuah kata dengan huruf besar
<code>explode()</code>	Memecah sebuah kalimat berdasarkan sebuah karakter dan mengubah menjadi array

Escape Character

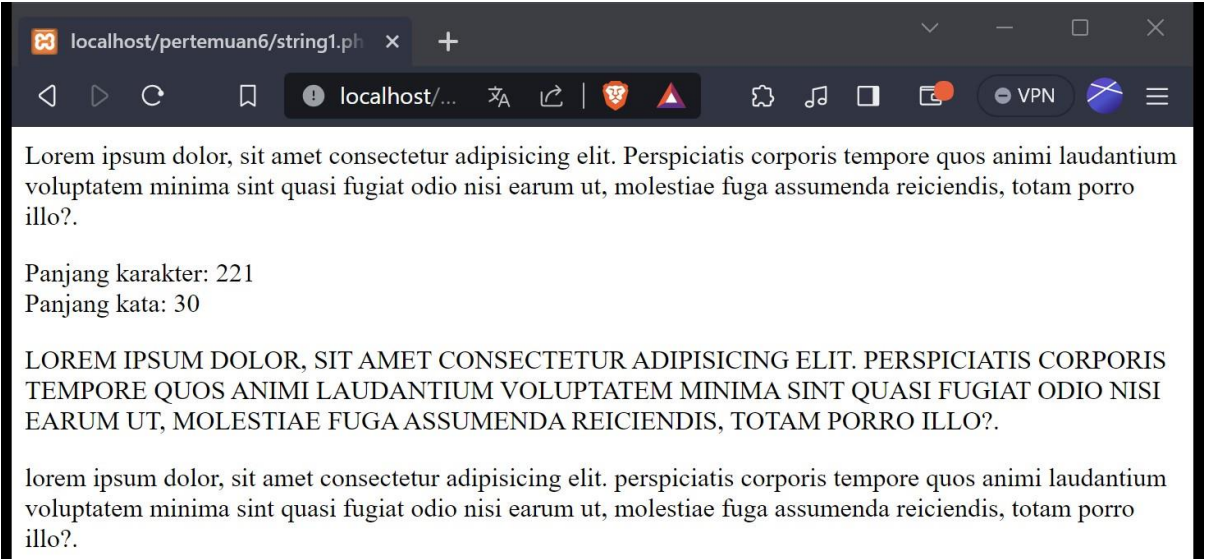
Karakter-karakter khusus yang tidak bisa ditampilkan secara langsung, melainkan harus diikuti dengan tanda \. String yang dirangkai dengan tanda petik dua akan mengganti *escape character* dengan karakter yang merepresentasikannya. Hal ini berbeda dengan string yang dirangkai dengan tanda petik satu. Yang mana ia hanya akan menampilkan apa adanya tanpa mereplace apa pun dengan apa pun (kecuali sedikit kasus).

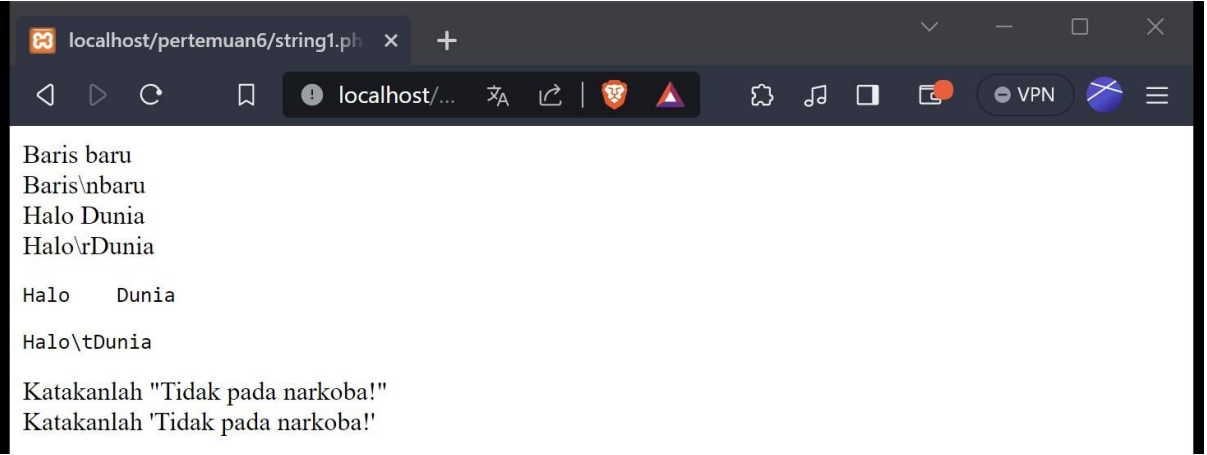
Ada pun *escape character* pada PHP adalah:

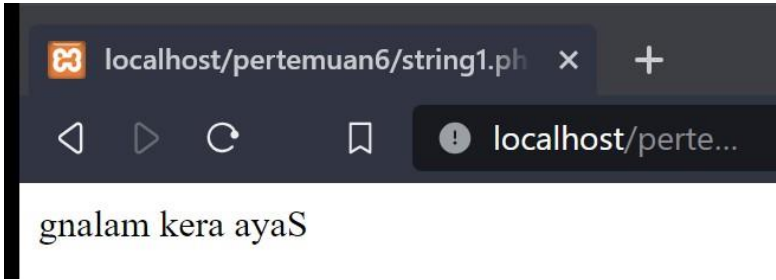
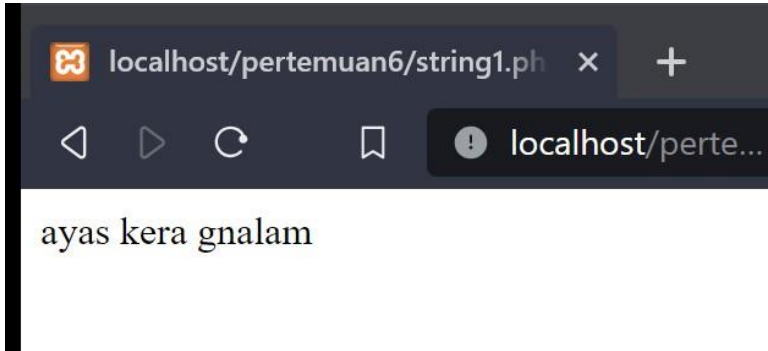
Fungsi	Keterangan
<code>\n</code>	Baris baru
<code>\r</code>	Karakter carriage-return
<code>\t</code>	Karakter tab
<code>\\$</code>	Karakter \$ itu sendiri
<code>\"</code>	Untuk menampilkan tanda petik dua
<code>\\</code>	Untuk menampilkan tanda slash \ itu sendiri

Ikuti langkah-langkah praktikum berikut ini:

Langka h	Keterangan
1	Buat file <code>string1.php</code> di dalam direktori <code>praktik_php</code> , kemudian ketikkan kode berikut:

	<pre> <?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 "<p>{\$loremIpsum}</p>"; echo "Panjang karakter: " . strlen(\$loremIpsum) . "
"; echo "Panjang kata: " . str_word_count(\$loremIpsum) . "
"; echo "<p>" . strtoupper(\$loremIpsum) . "</p>"; echo "<p>" . strtolower(\$loremIpsum) . "</p>"; ?> </pre>
2	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu (soal no 13)</p>  <p> Lorem ipsum dolor, sit amet consectetur adipisicing elit. Perspiciatis corporis tempore quos animi laudantium voluptatem minima sint quasi fugiat odio nisi earum ut, molestiae fuga assumenda reiciendis, totam porro illo?. Panjang karakter: 221 Panjang kata: 30 LOREM IPSUM DOLOR, SIT AMET CONSECTETUR ADIPISICING ELIT. PERSPICIATIS CORPORIS TEMPORE QUOS ANIMI LAUDANTIAM VOLUPTATEM MINIMA SINT QUASI FUGIAT ODIO NISI EARUM UT, MOLESTIAE FUGA ASSUMENDA REICIENDIS, TOTAM PORRO ILLO?. lorem ipsum dolor, sit amet consectetur adipisicing elit. perspiciatis corporis tempore quos animi laudantium voluptatem minima sint quasi fugiat odio nisi earum ut, molestiae fuga assumenda reiciendis, totam porro illo?. </p> <p>Dari hasil diatas kita dapat mengetahui jumlah karakter serta kata dari sebuah kalimat yang kita buat dengan menggunakan strlen untuk mengetahui panjang karakter serta str_word_count untuk mengetahui jumlah kata yang ada pada kalimat. Lalu, kita juga bisa mengubah sebuah kalimat yang kita buat untuk menampilkan dalam bentuk kapital ataupun lowercase dengan menggunakan strtoupper untuk menampilkan kalimat dalam bentuk kapital dan strtolower untuk menampilkan kalimat dalam bentuk lowercase.</p>
Escape Character	
3	<p>Buat file string2.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> <?php echo "Baris\nbaru
"; //soal 10.a echo 'Baris\nbaru
'; //soal 10.b echo "Halo\rDunia
"; //soal 10.c echo 'Halo\rDunia
'; ///soal 10.d echo "<pre>Halo\tDunia!</pre>"; //soal 10.e echo '<pre>Halo\tDunia!</pre>'; //soal 10.f echo "Katakanlah \"Tidak pada narkoba!\"
"; //soal 10.g echo 'Katakanlah \"Tidak pada narkoba!\"
'; //soal 10.h ?> </pre>
4	<p>Dari kode program di atas, kalian bisa mengetahui perbedaan antara tanda petik dua dan tanda petik satu dari segi cara kerjanya menangani <i>escape string</i>. Amati hasil yang ditampilkan dan jelaskan hasil dari masing-masing outputnya dan apa yang dapat kalian simpulkan dari soal percobaan tersebut (soal no 14)</p>

	 <p>Dari hasil program diatas kita bisa melihat</p> <ul style="list-style-type: none"> - 10.a : Kata “Baris” dan “baru” terdapat sebuah baris baru yang dimana itu merupakan hasil dari “\n” dengan menggunakan tanda petik dua, kita bisa menghasilkan itu. - 10.b : Pada hasil No. 10.b hanya menampilkan apa yang ada pada program. Tidak memberikan hasil yang sesuai dengan nilai seperti pada soal No. 10.a. - 10.c : Pada kalimat di soal No.10.c hampir sama dengan soal No.10.a. Namun, sebenarnya itu akan digunakan untuk mengendalikan penempatan kursor dalam teks untuk “/r”. Sehingga tertampil seperti itu. Itu dikarenakan pada penggunaan tanda petik dua. - 10.d : Pada hasil No.10.d hanya menampilkan apa yang ada pada program. Tidak memberikan hasil yang sesuai dengan nilai seperti pada soal No.10.c. - 10.e : Pada kalimat di soal No.10.e kata “Halo” dan “Dunia” memiliki jarak yang dihasilkan dari penggunaan “\t” yang memberikan tab jarak antar kedua kata tersebut. Lalu, itu juga bisa dihasilkan sesuai itu karena menggunakan tanda petik dua. - 10.f : Pada hasil No.10.f hanya menampilkan apa yang ada pada program. Kecuali, penggunaan tag bisa ditampilkan. Namun penggunaan “/” pada tanda petik satu tidak bisa digunakan. - 10.g : Pada hasil No.10.g kita bisa melihat bahwa tanda petik dua bisa ditampilkan dengan menggunakan “/”. - 10.h : Pada hasil No.10.h sama saja dengan hasil No.10.g yaitu menampilkan tanda petik ke layar. Namun, pada hasil No.10.h menampilkan tanda petik satu.
	<p>Memalik String menggunakan perintah <i>strrev()</i>.</p>
5	<p>Buat file <code>string3.php</code> di dalam direktori <code>praktik_php</code>, kemudian ketikkan kode berikut:</p> <pre data-bbox="279 1680 686 1848"><?php \$pesan = "Saya arek malang"; echo strrev(\$pesan) . "
"; ?></pre>
6	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu (soal no 15)</p>

	 <p>Dari hasil yang diatas terlihat bahwa kalimat yang kita tampilkan terbalik dari yang paling belakang. Yang seharusnya “Saya arek malang” sehingga menjadi “gnalam nera ayaS”.</p>
8	<p>untuk membalik string per kata, ketikkan kode program berikut:</p> <pre><?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) => strrev(\$pesan), \$pesanPerKata); # gabungkan kembali array menjadi string \$pesan = implode(" ", \$pesanPerKata); echo \$pesan . "
"; ?></pre>
8	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu (soal no 16)</p>  <p>Dari hasil yang terjadi adalah kalimat yang kita masukkan menjadi terbalik setiap katanya. Jika, pada soal sebelumnya benar-benar terbalik hingga ke urutan katanya. Sekarang hanya per katanya saja yang terbalik namun urutan katanya tetap benar.</p>

Menggabungkan HTML dan PHP

Ada dua cara menggabungkan HTML dan PHP yaitu PHP yang berada didalam HTML, dan HTML yang ada di dalam PHP.

Langkah	Keterangan
1	<p>Cara pertama adalah php di dalam HTML. Seperti kode berikut.</p> <pre><html> <head> <title>Cara 01</title> </head> <body> <p>Tanggal Hari ini : <?php echo date("d M Y")?></p> </body> </html></pre>

2	Pada kode diatas merupakan kode html yang berisi kode php untuk menampilkan tanggal server yang ditandai dengan tag <?php dan ?> .
3	<p>Cara kedua adalah HTML di dalam PHP. Di dalam PHP Tag HTML diperlakukan sebagai <i>string</i> yang di apit dengan tanda petik dan bisa dilakukan berbagai fungsi untuk memanipulasi <i>string</i> seperti menyambung dll.</p> <p>Contoh kode seperti pada potongan kode berikut.</p> <pre><?php echo '<html>'; echo '<head><title>Cara02</title></head>'; echo '<body>'; echo '<p>Tanggal Hari ini : '.date('d M Y').'</p>'; echo '</body>'; echo '</html>'; ?></pre>
4	Kode diatas mengeluarkan output sama dengan potongan kode sebelumnya. Namun yang berbeda adalah penulisan kode dimana HTML berada di dalam PHP sebagai string dan untuk menampilkannya menggunakan <i>tag echo</i> .
5	<p>Dari dua cara tersebut mana yang lebih mudah menurut kalian, kemukakan jawaban disertai dengan alasan (soal no 17)</p> <p>Dari kedua cara tersebut yang lebih mudah adalah pada cara pertama. Kita hanya menambahkan sintaks php echo date untuk menampilkan tanggal. Jika dibanding dengan kita full menggunakan echo pada php dan penulisan yang lebih memakan waktu yang lama.</p>

Entities HTML

Ketika membuat konten website, tidak akan terlepas dari menggunakan karakter khusus seperti simbol copyright (©), ampersand (&), lebih kecil (<), lebih besar (>), titik koma (;) dan sebagainya. Untuk menampilkan simbol-simbol tersebut di HTML perlu menggunakan entitas karakter (*character entities*). Entitas Karakter mempunyai 3 bagian yaitu: sebuah *ampersand* (&), nama dan nomor entitas (#), dan titik koma/semicolon (;).

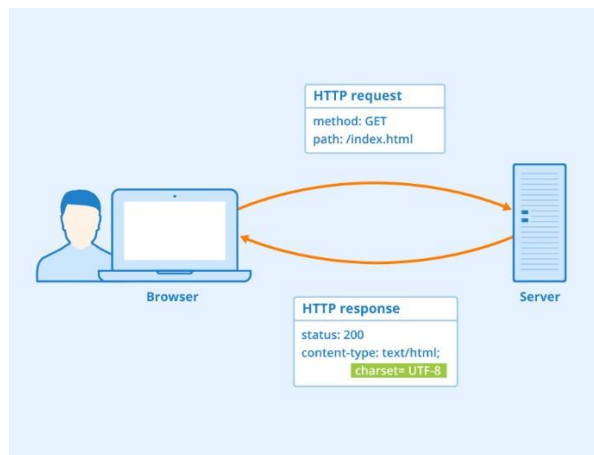
Sebagai contoh, untuk menampilkan simbol copyright (©) → © atau ©

Berikut ini adalah tabel beberapa entitas yang umum digunakan:

Nama Entitas	Nomor Entitas	Keterangan	Hasil
©	©	Copyright	©
®	®	Registered	®
-	™	Trademark	™
 	¡	Non break space	
&	&	Ampersand	&
«	«	Angle question left	«

HTTP *header* adalah sebuah data yang dikirim antara web browser dengan web server sebagai sarana komunikasi antar keduanya. Di dalam HTTP header terdapat informasi tentang bagaimana cara menangani file yang dikirim/diminta.

Siklus *request respond* untuk halaman web: Ketika kita mengakses suatu halaman web, web browser secara otomatis mengirim sebuah HTTP *request* kepada web server. HTTP *request* berisi banyak informasi, salah satunya adalah HTTP *header*. Di dalam HTTP *header* (yang dikirim pada saat proses *request*) terdapat informasi tentang file apa yang diminta (apakah file HTML, file PHP, file PDF, atau yang lain), serta berbagai info tambahan seperti jenis web browser yang dipakai, sistem operasi, dan alamat IP. Setelah sampai di web server, informasi yang ada di HTTP *header* akan dibaca dan web server mempersiapkan file-file yang diminta. Setelah itu, web server mengirim kembali file-file tersebut kepada web browser. Proses pengembalian ini dikenal juga dengan sebutan HTTP *Response*.



HTTP Header

HTTP *response* ini terdiri dari 2 bagian: HTTP header dan file web. HTTP *header* berisi informasi mengenai file web yang dikirim, seperti tipe data, tanggal dikirim, nama web server, dan sistem operasi yang digunakan oleh web server. Sedangkan file web sendiri terdiri dari file HTML yang menyusun halaman web, termasuk juga file gambar (jika ada).

Sebagai analogi, jika presiden ingin berkunjung ke rumah kalian, tentu akan ada tim paspampres yang datang lebih awal. Mereka menginformasikan bahwa presiden akan datang pada jam sekian, dengan membawa sekian orang, dan informasi-informasi lain. Tim paspampres ini bisa disamakan dengan HTTP header yang datang lebih dahulu sebelum file asli dikirim. Dalam prakteknya, kita tidak menyadari adanya HTTP header, bahkan mungkin tidak pernah mendengar istilah ini. Dan itu tidak salah, karena isi dari HTTP header ditujukan kepada pemrosesan di web browser, bukan untuk pengunjung web.

Bagaimana cara melihat http header pada web browser kalian jelaskan disertai Langkah-langkahnya (Soal no 19)

Date and Time

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

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

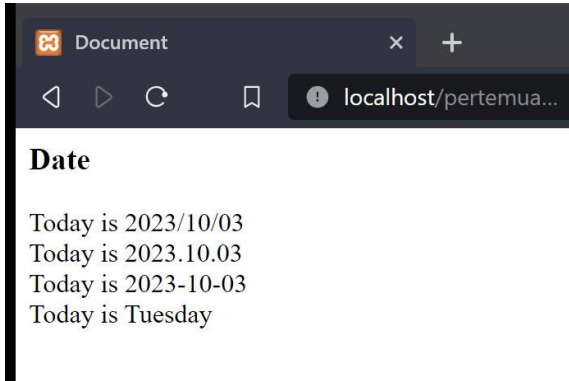
Parameter format bersifat wajib (*required*). Parameter format digunakan untuk menentukan bagaimana format tanggal dan/atau waktu yang akan digunakan. Berikut beberapa contoh karakter yang umum digunakan untuk format tanggal:

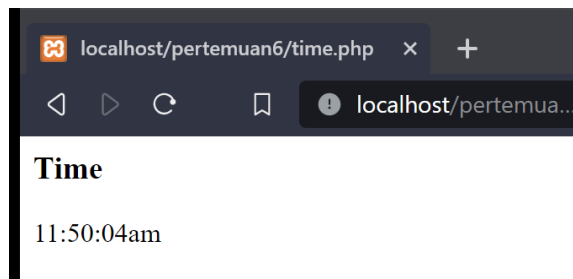
1. d, merepresentasikan hari (01 sampai 31)
2. m, merepresentasikan bulan (01 sampai 12)
3. Y, merepresentasikan tahun (dalam 4 digit)
4. l, merepresentasikan hari dalam 1 minggu

Selain menampilkan tanggal, fungsi tersebut juga menampilkan waktu. Berikut adalah beberapa karakter yang umum digunakan untuk format waktu:

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

Ikuti langkah-langkah berikut untuk memahami bagaimana penggunaan fungsi `date()`

Langkah	Keterangan
1	<p>Buat file baru dengan nama <code>date.php</code> di dalam direktori <code>dasarWeb</code>, kemudian ketikkan kode berikut:</p> <pre> 1 <?DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <h3> Date </h3> 7 <?php 8 echo "Today is " . date("Y/m/d") . "
"; 9 echo "Today is " . date("Y.m.d") . "
"; 10 echo "Today is " . date("Y-m-d") . "
"; 11 echo "Today is " . date("l") ; 12 ?> 13 </body> 14 </html> </pre>
2	Simpan file dan jalankan kode program
3	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 19)</p> 

	<p>Dari hasil diatas kita bisa mengetahui bahwa kita bisa menampilkan date atau tanggal. Bisa dalam bentuk "/", ".", "-" yang paling penting adalah penentuan formatan Year adalah Y, Month adalah m, dan d adalah Day. Sedangkan kita bisa mentarget hari ini yaitu dengan formatan "l" yang akan mentarget hari ini.</p>
4	<p>Buat file baru dengan nama time.php di dalam direktori dasarWeb, kemudian ketikkan kode berikut:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <h3> Time </h3> 7 <?php 8 date_default_timezone_set("asia/jakarta"); 9 echo date("h:i:sa"); 10 ?> 11 </body> 12 </html> </pre>
5	<p>Simpan file dan jalankan kode program</p>
6	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 20)</p>  <p>Dari hasil diatas kita bisa untuk set waktu kita dalam zona waktu apa. Disini kita mengatur waktu dalam zona asia/Jakarta. Disini kita akan menggunakan waktu Jakarta. Yang dimana bisa kita tampilkan waktu saat ini pukul berapa dengan menggunakan echo lalu kita tambahkan date dan formatan untuk menampilkan jam, menit, hingga detik nya. Seperti pada gambar diatas.</p>

Variabel Superglobal

Variabel super global di PHP adalah variabel bawaan yang bersifat global. Variabel bawaan yang dimaksud adalah: variabel yang sudah otomatis ada tanpa perlu kita definisikan sendiri. Dan ia bersifat global dalam artian bisa kita akses dari mana pun dan kapan pun.

Variabel super global menyimpan banyak sekali data penting dan juga bermanfaat yang bisa kita gunakan dalam menyelesaikan projek yang sedang kita kerjakan. Terdapat **9 variabel** super global di PHP.

1. Variabel \$_SERVER

Variabel yang pertama dan utama adalah variabel \$_SERVER. Ia adalah sebuah array asosiatif yang menyediakan berbagai macam informasi tentang request yang ditangkap oleh server. Data yang dimuat berupa *headers*, *paths*, lokasi skrip, dan sebagainya.

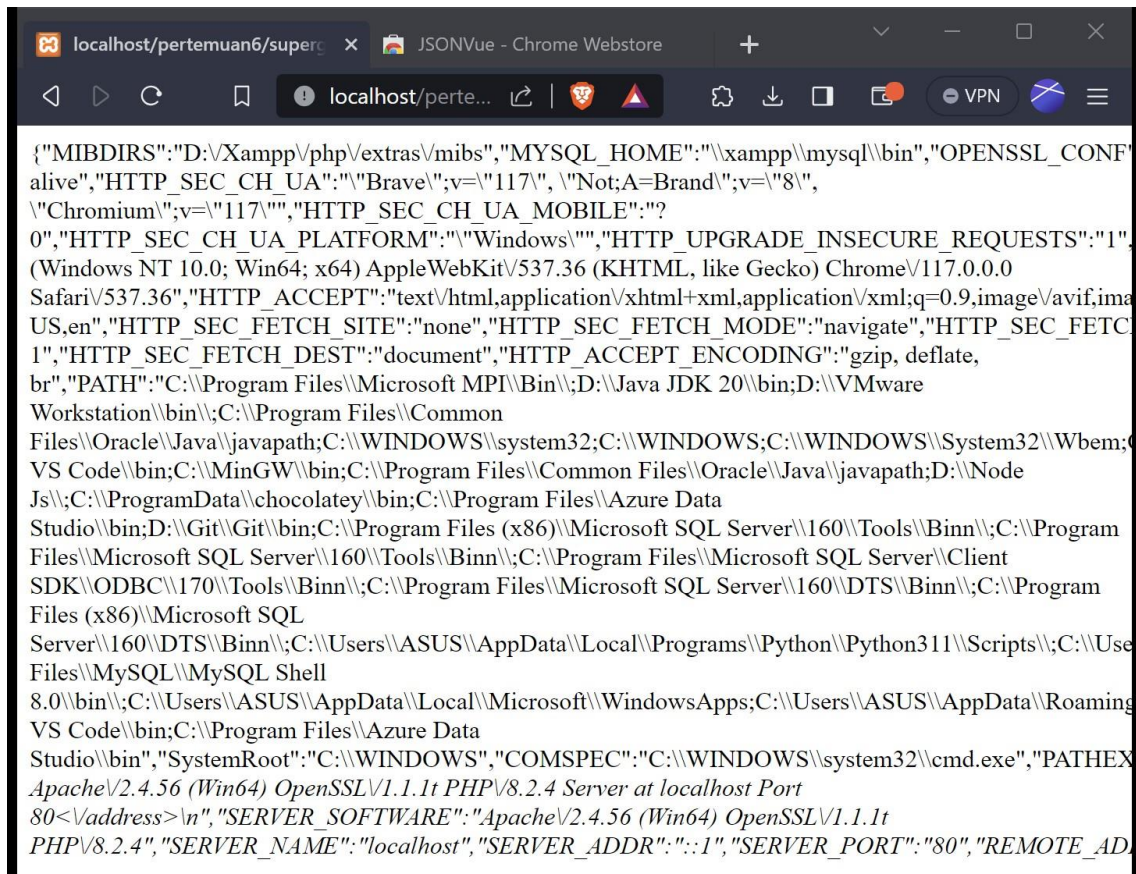
Nilai yang tersimpan pada variabel \$_SERVER disediakan oleh web server, oleh karena itu tidak ada jaminan khusus bahwa setiap web server yang kita gunakan akan memberikan semua data-data standar yang ada.

Untuk bisa mengetahui nilai apa saja yang tersedia pada variabel \$_SERVER, kita bisa mengeksekusi perintah berikut:

```
<?php
```

```
echo json_encode($_SERVER);
```

Bagaimana output yang didapatkan (**Soal no 21**) sebelum dijalankan silakan install ekstensi JSON View pada link berikut ini



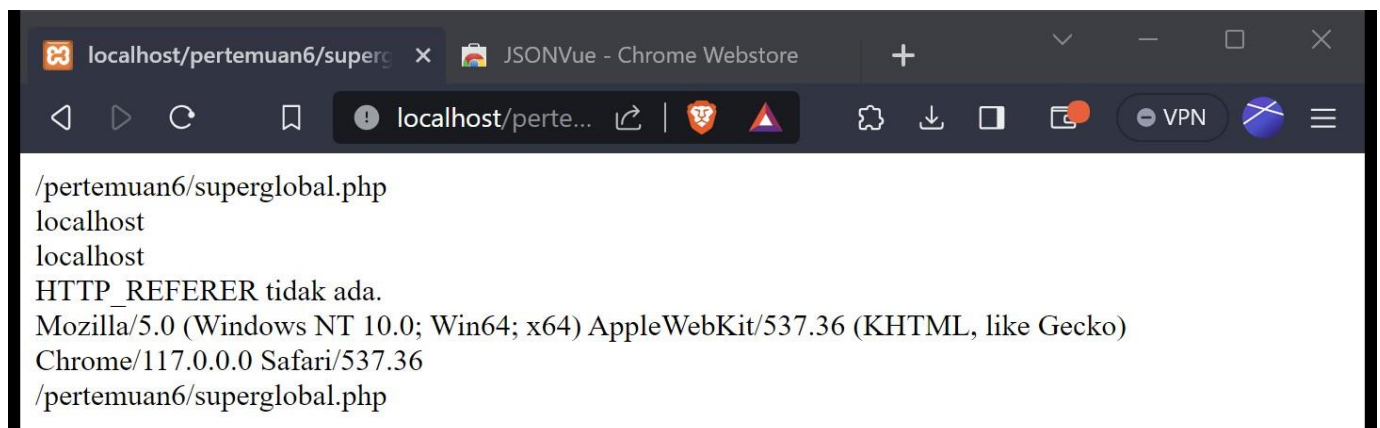
<https://chrome.google.com/webstore/detail/jsonvue/chklaanhfefbnpoihckbnefhakgolnmc?hl=id>

```

1  <?php
2  echo $_SERVER['PHP_SELF'];
3  echo "<br>";
4  echo $_SERVER['SERVER_NAME'];
5  echo "<br>";
6  echo $_SERVER['HTTP_HOST'];
7  echo "<br>";
8  echo $_SERVER['HTTP_REFERER'];
9  echo "<br>";
10 echo $_SERVER['HTTP_USER_AGENT'];
11 echo "<br>";
12 echo $_SERVER['SCRIPT_NAME'];
13 ?>

```

Jalankan kode program diatas kemudian jelaskan output dari masing-masing perintah echonya (Soal no.22)



- `$_SERVER['PHP_SELF']` : variabel yang berisi alamat URL dari skrip php saat ini yang sedang dieksekusi oleh server.
- `$_SERVER['SERVER_NAME']` : variabel yang berisi nama server yang digunakan untuk mengakses halaman saat ini.
- `$_SERVER['HTTP_HOST']` : variabel yang berisi nama host HTTP yang digunakan dalam permintaan HTTP saat ini. Ini sering digunakan untuk mengidentifikasi domain atau host yang digunakan oleh pengguna.
- `$_SERVER['HTTP_REFERER']` : variabel yang berisi URL referer (URL halaman yang merujuk ke skrip ini, jika ada) dari permintaan HTTP saat ini.
- `$_SERVER['HTTP_USER_AGENT']` : variabel yang berisi informasi tentang agen pengguna (browser yang digunakan oleh pengguna) yang mengakses halaman saat ini.
- `$_SERVER['SCRIPT_NAME']` : variabel yang berisi nama skrip (file PHP) yang sedang dieksekusi.

Beberapa contoh data pada variabel `$_SERVER` yang akan sering dibutuhkan:

No	Variabel dan Deskripsi
1	<code>\$_SERVER['PHP_SELF']</code> Berisi informasi nama file yang sedang dieksekusi, nama file diambil berdasarkan dari dokumen <i>root</i>
2	<code>\$_SERVER['SERVER_ADDR']</code> Alamat IP server dari file yang sedang dieksekusi.
3	<code>\$_SERVER['SERVER_NAME']</code> Hostname dari server dari file php yang sedang dieksekusi. Hostname biasanya adalah nama PC yang terlihat di dalam jaringan. Jika skrip PHP dijalankan di atas Virtual Host, maka nama virtual host tersebut akan dijadikan sebagai <i>server name</i>
4	<code>\$_SERVER['SERVER_PROTOCOL']</code> Ini adalah protokol komunikasi yang sedang berjalan. Apakah HTTP atau HTTPS. Misal: 'HTTP/0.1'
5	<code>\$_SERVER['REQUEST_METHOD']</code> Berisi jenis metode <i>request</i> dari file PHP yang sedang dieksekusi. Misalkan: <code>GET POST PUT DELETE OPTIONS</code>
6	<code>\$_SERVER['QUERY_STRING']</code> Akan mengembalikan query string dari file atau skrip PHP yang sedang dieksekusi. Misalkan user mengakses halaman <code>http://localhost/halo-dunia?nama=Budi&umur=20&asal=Surabaya</code> , maka variabel tersebut akan mengembalikan nilai <code>nama=Budi&umur=20&asal=Surabaya</code> .
7	<code>\$_SERVER['DOCUMENT_ROOT']</code> Nama direktori root dari dokumen/file PHP yang sedang dieksekusi. Hasil yang dikembalikan berdasarkan dari pengaturan server.
8	<code>\$_SERVER['HTTP_HOST']</code> Mengembalikan konten dari host, seperti misalnya: header (jika memang ada).
9	<code>\$_SERVER['HTTP_REFERER']</code> Halaman url yang menjadi referensi ke halaman yang sedang dieksekusi. Jika tidak ada, nilainya kosong.
10	<code>\$_SERVER['HTTP_USER_AGENT']</code> Variabel ini berisi informasi dari user yang sedang melakukan <i>request</i> konten: mulai dari browser yang digunakan, bahasa, hingga OS. Contoh nilai dari variabel ini adalah: Mozilla/4.5 [en] (X11; U; Linux 2.2.9 i586
11	<code>\$_SERVER['REMOTE_ADDR']</code> Berisi alamat IP dari user yang sedang mengakses halaman web PHP
12	<code>\$_SERVER['SCRIPT_FILENAME']</code> Nama path absolut dari file yang sedang dieksekusi.
13	<code>\$_SERVER['REQUEST_URI']</code> Alamat uri dari file yang sedang dieksekusi. Misalkan: <code>"/php/halo-dunia"</code> .

2. Variabel \$_GET

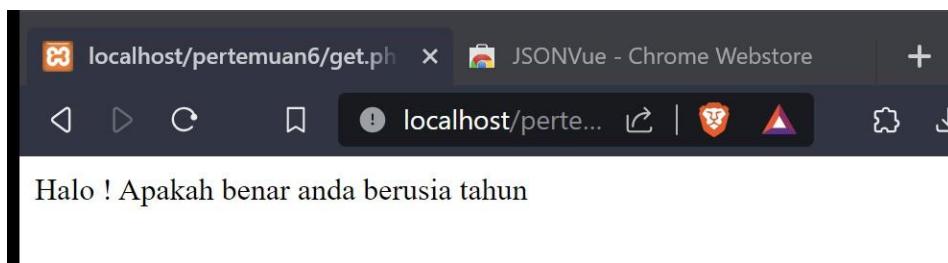
Variabel \$_GET adalah array asosiatif yang berisi nilai dari *query string*. Misalkan kita memiliki file halo-dunia.php sebagaimana berikut:

```
<?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?";
?>
```

Jika kita mengakses file tersebut dengan *query string* dibawah ini, output apa yang dihasilkan, amati dan bagaimana hasilnya (soal no 23)



Dari hasil diatas kita bisa melihat bahwa kita bisa menggunakan \$_GET untuk mengambil nilai dari query string jika ada. Jika tidak ada kita bisa memanfaatkan @ agar tidak menampilkan pesan error pada websitenya.

<http://localhost/halo-dunia.php?nama=Elok&usia=30>

3. Variabel \$_POST

Variabel \$_POST mirip dengan variabel \$_GET. Hanya saja data yang di-*passing* tidaklah melalui query string pada URL, akan tetapi pada *body request*. Dan *request method* yang dilakukan haruslah dengan metode **POST**.

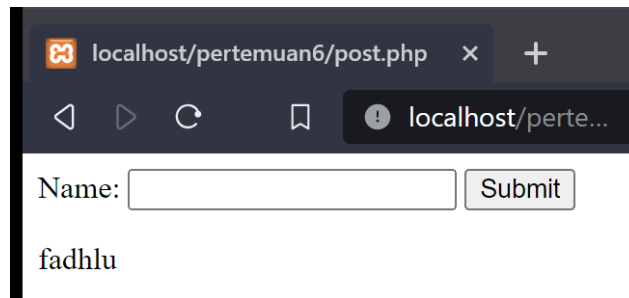
```
<html>
<body>

<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 kode program berikut ini. Apa yang bisa kalian simpulkan dari output yang dihasilkan (Soal no 24)



Dari hasil diatas kita bisa melihat bahwa terdapat formulir sebuah inputan yang dibuat dari hasil `<form>` lalu memberikan sebuah action. Setelah pengguna mengisi nama mereka dan mengklik tombol "Submit," data formulir dikirimkan kembali ke halaman yang sama. Kode PHP kemudian memeriksa apakah data formulir telah dikirimkan dengan metode POST. Lalu, akan dilakukan sebuah pengecekan tentang nama yang diinput jika kosong maka tidak ditampilkan. Jika ada maka nama tersebut akan ditampilkan seperti pada gambar diatas.

4. Variabel `$_SESSION`

Variabel `$_SESSION` adalah array asosiatif yang menyimpan data sesi pengguna. Variabel ini bisa kita gunakan untuk menyimpan user yang login pada satu sesi tertentu. Atau juga bisa digunakan untuk menyimpan data *cart* pada toko online. Secara *default*, umur sesi pada PHP adalah **1440 detik** atau **24 menit**.

5. Variabel `$_COOKIE`

Mirip dengan `$_SESSION`, variabel `$_COOKIE` bisa kita gunakan untuk menyimpan suatu data yang berkaitan dengan user: misal informasi login, informasi *cart* pada toko online, dan sebagainya.

Bedanya, *cookie* adalah file berukuran kecil yang disimpan pada browser pengguna. File tersebut akan senantiasa dikirim setiap kali browser mengirimkan *request* ke server. Umur *cookie* umumnya lebih panjang dari pada umur sesi.

6. Variabel `$_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.

```

<html>
<body>

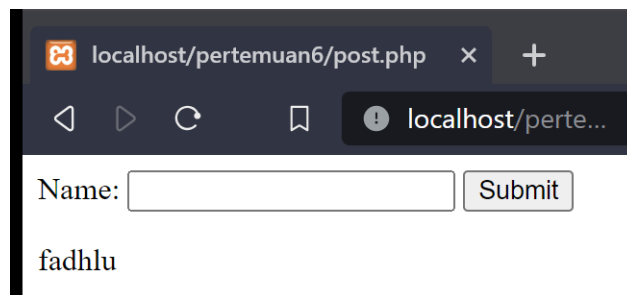
<form method="post" action="<?php echo $_SERVER['PHP_SELF'];?>">
  Name: <input type="text" name="fname">
  <input type="submit">
</form>

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

</body>
</html>

```

Jalankan kode program berikut ini. Apa yang bisa kalian simpulkan dari output yang dihasilkan. Dan apa bedanya dengan variable global \$POST(Soal no 25)



- Metode POST digunakan untuk mengirimkan data formulir secara “aman” karena data tersebut tidak muncul dalam URL dan tidak bisa terlihat oleh orang lain saat data dikirimkan.
- REQUEST adalah variabel superglobal yang menggabungkan data dari metode POST, GET, dan COOKIE. Ini berarti jika data tidak ditemukan dalam \$_POST, maka \$_REQUEST akan mencoba mencarinya dalam \$_GET.
- Perbedaan antara keduanya terletak pada sumber data yang diakses. Jika kita hanya ingin mengakses data formulir yang dikirimkan melalui metode POST, lebih baik menggunakan \$_POST. Jika kita ingin mengakses data dari berbagai sumber (POST, GET, COOKIE), maka kita dapat menggunakan \$_REQUEST

7. Variabel \$_FILES

Variabel \$_FILES adalah array asosiatif yang menyimpan data file yang diunggah pengguna dalam satu *request* dengan metode **POST** atau **PUT**.

8. Variabel \$_ENV

Variabel \$_ENV adalah array asosiatif yang berisi data tentang *environment* yang skrip PHP berjalan di atasnya. Variabel \$_ENV disediakan oleh *shell* yang menjalankan skrip PHP, sehingga nilainya bisa bervariasi tergantung dengan sistem operasi yang digunakan.

Di dalam *framework* PHP modern seperti laravel, variabel \$_ENV juga digunakan untuk menyimpan hal-hal yang berkaitan dengan *environment* seperti nama database, password database, dan nilai lainnya untuk melakukan konfigurasi *framework*.

9. Variabel \$GLOBALS

Variabel \$GLOBALS adalah array asosiatif yang menyimpan semua variabel global yang didefinisikan saat program dijalankan. Variabel \$GLOBALS merupakan variabel super global PHP yang digunakan untuk mengakses variabel global dari mana saja dalam scrip PHP (juga dari dalam fungsi atau metode).

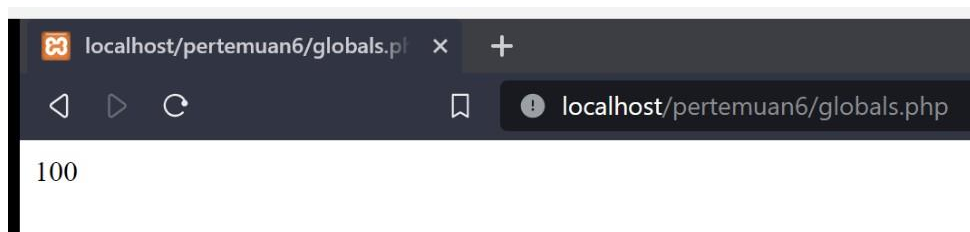
Cara menggunakan variabel super global \$GLOBALS:

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

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

addition();
echo $z;
?>
```

Bagaimana output dari kode soal di atas kemudian jelaskan! (Soal no 26)



Dari program diatas kita bisa melihat penggunaan variabel globals. Yang dimana variabel yang bisa diakses dari dalam fungsi ataupun diluar fungsi. Sehingga kita tidak perlu membuat sebuah return pada fungsi untuk mendapatkan nilai dari hasil fungsi. Kita bisa memanfaatkan variabel global.

Referensi:

- 1) Nixon, Robin. (2018). Learning PHP, MySQL, JavaScript, CSS & HTML: A Step-by-step Guide to Creating Dynamic Websites, 5th Edition. O'Reilly Media, Inc.
- 2) Forbes, Alan. (2012). The Joy of PHP: A Beginners's Guide to Programming Interactive Web Applications with PHP and MySQL, 5th Edition. Plum Island Publishing