

PERTEMUAN 5

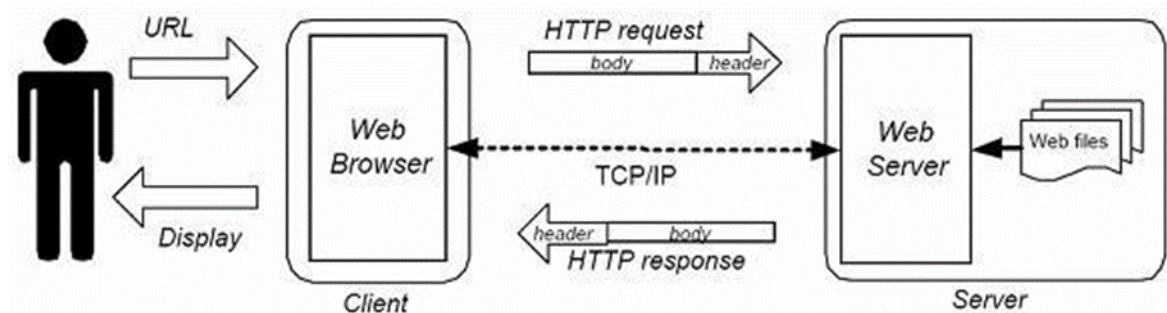
Pengantar PHP, Array, Fungsi, String

A. Tujuan Pembelajaran

Pada pertemuan ini akan dijelaskan penggunaan CSS. Setelah selesai materi pada pertemuan ini, mahasiswa mampu memahami konsep dasar web server, memahami konsep php (*array* dan *fungsi*), memahami fungsi *built-in* pada PHP untuk string dan waktu (*date and time*), dan mengimplementasikan konsep PHP ke dalam *web* dinamis

B. Materi

Server atau *web server* adalah sebuah perangkat lunak (*software*) yang berfungsi untuk menerima permintaan (*request*) dari *client* (*web browser*) berupa halaman web melalui *protocol* HTTP/HTTPS, lalu merespon permintaan tersebut dalam bentuk halaman *web* berupa dokumen HTML. Dalam bahasa sederhananya *web server* diartikan sebagai *software* yang berfungsi sebagai pemberi layanan kepada *web client* (*browser*) seperti Chrome, Mozilla, Opera, Safari, dan lainnya, agar *browser* tersebut dapat menampilkan halaman *website* yang diminta oleh *user*. Salah satu contoh dari *Web Server* adalah Apache. Cara kerja web server diilustrasikan pada Gambar 1.



Gambar 1. Ilustrasi cara kerja web server (sumber gambar: google.com)

Seperti yang sudah disebutkan sebelumnya, *client* adalah pengguna internet yang memakai *browser* (Chrome, Mozilla, Opera, Safari, dan lainnya) untuk terhubung dengan web server melalui internet, sedangkan web server adalah software yang melayani permintaan web *client*. Ketika *client* (*browser*) melakukan permintaan data halaman *website* kepada server, maka permintaan tersebut dikemas oleh *browser* di dalam TCP (Transmission Control Protocol) yang merupakan protokol transport. Lalu permintaan tersebut dikirim ke alamat data, dalam hal ini merupakan protokol HTTP atau HTTPS. Selanjutnya, data tersebut akan

dicari oleh web server di dalam komputer server. Jika data ditemukan maka data tersebut akan dikemas oleh web server dalam TCP lalu dikirim dan ditampilkan ke *browser*. Jika data yang dikirim diminta web *client* tidak ditemukan, maka web server akan menolak permintaan dan menampilkan halaman “Error 404” atau “Page Not Found” di *browser*.

Pengenalan PHP

Bahasa pemrograman PHP dikembangkan pada tahun 1995 oleh Rasmus Lerdorf. Sejarah dan perkembangan bahasa pemrograman PHP dapat dibaca lebih lanjut pada situs resmi PHP yang beralamat di <http://www.php.net>. PHP berasal dari kata Hypertext Preprocessor. PHP disebut juga sebagai server-side programming/script, yaitu bahasa pemrograman yang akan dijalankan/diproses oleh web server. PHP sebenarnya bukan bahasa pemrograman yang wajib digunakan dalam membuat halaman web. Kita bisa saja membuat *website* hanya menggunakan HTML saja. Web yang dihasilkan dengan HTML (dan CSS) ini dikenal dengan *website* statis, dimana konten dan halaman web bersifat tetap. PHP digunakan untuk membuat *website* yang dinamis, dimana halaman web bisa menyesuaikan tampilan konten tergantung kondisi. *Website* dinamis juga bisa menyimpan data ke dalam database, membuat halaman yang berubah-ubah sesuai input dari *user*, memproses form, dan sebagainya.

Untuk menjalankan kode program PHP, setidaknya dibutuhkan 3 jenis program, yaitu: web server, teks editor, dan web *browser*. Kode program PHP bisa diletakkan di dalam HTML atau dalam file yang terpisah. Agar web server dapat mengeksekusi kode program PHP, file harus disimpan sebagai file PHP, yaitu dengan akhiran file *.php. Sintak dasar kode program PHP adalah sebagai berikut:

```
<?php
  baris kode program
?>
```

Kode program PHP memiliki beberapa aturan dasar, yaitu:

1. *case sensitive*

PHP tidak membedakan huruf besar dan kecil untuk penamaan fungsi, class, maupun keyword bawaan PHP, seperti echo (perintah untuk menampilkan output ke layar), while (pengulangan), dan if (kondisi). Namun, PHP membedakan huruf besar dan huruf kecil (*case sensitive*) untuk penamaan variabel, sehingga variabel \$nama dan \$NAMA akan dianggap sebagai 2 variabel yang berbeda. Berikut contoh penulisan variable:

```
<?php
  $nama_pt = "polinema";
  echo $nama_pt; //output: polinema
  ECHO $nama_pt; //output: polinema
  echo $nama_PT; //output: error, karena tidak ada variabel $nama_PT
?>
```

2. Penulisan baris perintah dalam PHP

Baris perintah (statement) di dalam PHP adalah kumpulan perintah PHP yang menginstruksikan PHP untuk melakukan sesuatu. Setiap akhir baris perintah diakhiri dengan tanda semicolon atau titik koma (;). Kumpulan baris perintah yang menggunakan tanda kurung kurawal seperti kondisi (if) atau pengulangan (loop) tidak dibutuhkan tanda titik koma setelah kurung penutup.

```
<?php
    if(true){
        echo "perintah dijalankan";
    }
?>
```

3. Karakter spasi dan tab

Di dalam kode program PHP, karakter spasi dan tab diabaikan dalam eksekusi program.

4. Penulisan komentar

PHP memberikan beberapa cara untuk menuliskan komentar, yaitu metode C/C++ dengan tanda (//) dan UNIX Shel dengan tanda (#). Berikut adalah contoh penulisan komentar:

```
<?php
    $luas = 1/2 * $alas * $tinggi; //mengitung luas segitiga
    echo $luas; #menampilkan hasil
?>
```

Praktikum Bagian 1. Menjalankan Kode Program PHP

Agar halaman web yang dimaksud dapat ditampilkan pada *browser*, diperlukan suatu cara agar web server dapat membedakan kode PHP dan HTML. Ikuti langkah-langkah berikut untuk memahami cara kerja web server menjalankan kode program PHP:

Langkah	Keterangan
1	Jalankan Apache pada XAMPP
2	Buatlah direktori baru bernama "praktik_php" di dalam direktori "dasarWeb" yang telah Anda buat pada praktikum sebelumnya.
3	Buat file baru dengan nama <code>cara_kerja.php</code> di dalam direktori <code>praktik_php</code> , kemudian ketikkan kode berikut: <pre>1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <p> kalimat dengan menggunakan HTML </p> 7 <?php 8 echo "kalimat dengan menggunakan PHP"; 9 ?> 10 </body> 11 </html></pre>

4	Buat file baru dengan nama <code>cara_kerja.html</code> , kemudian salin kode program yang sama dengan langkah 2.
5	Simpan kedua file tersebut. Buka 2 tab/jendela <i>browser</i> dan jalankan kedua kode program tersebut dengan mengetikkan <code>localhost/dasarWeb/praktik_php/cara_kerja.php</code> dan <code>localhost/dasarWeb/praktik_php/cara_kerja.html</code>
6	Amati hasil yang ditampilkan dan jelaskan dengan bahasa Anda sendiri cara kerja web server setelah menjalankan kedua kode program tersebut! (soal no 1)

Praktikum Bagian 2. Echo dan Print

Di dalam PHP untuk menampilkan data ke layar digunakan perintah `echo` dan `print`. Ikuti langkah-langkah berikut untuk memahami penggunaan perintah `echo` dan `print`:

Langkah	Keterangan
1	<p>Buat file <code>cetak.php</code> di dalam direktori <code>praktik_php</code>, kemudian ketikkan kode berikut:</p> <pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$txt1 = "Praktikum PHP"; 8 \$txt2 = "Variabel, Tipe Data, String"; 9 echo "<h2> \$txt1 </h2>"; 10 echo "Materi yang dipelajari " . \$txt2; 11 12 print "<h2> \$txt1 </h2>"; 13 print "Materi yang dipelajari " . \$txt2; 14 ?> 15 </body> 16 </html> </pre>
2	Simpan file tersebut dan jalankan kode program dengan mengetikkan <code>localhost/dasarWeb/praktik_php/cetak.php</code>
3	Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu (soal no 2)

Praktikum Bagian 3. Variabel

Variable adalah kode program yang digunakan untuk menampung nilai tertentu. Nilai yang disimpan di dalam variable selanjutnya dapat dipindahkan ke dalam database atau ditampilkan kembali ke pengguna. Sebuah variable memiliki nama, yang digunakan untuk mengakses nilai dari variable itu. Nilai dari variabel dapat diisi dengan informasi yang diinginkan dan dapat diubah nilainya pada saat kode program sedang berjalan.

Berikut adalah aturan-aturan variable dalam PHP:

1. Penulisan variable diawali dengan tanda dollar (\$), kemudian diikuti dengan nama variabel.
2. Nama variable berisi karakter alfanumerik (A-z, 0-9, dan garis bawah atau *underscore*). Nama variabel harus diawali dengan huruf atau karakter garis bawah (_), tidak boleh diawali dengan angka.
3. Penulisan nama variable dalam PHP adalah *case sensitive*, sehingga perlu diperhatikan besar dan kecil huruf ketika menuliskannya.
4. Variable dalam PHP tidak perlu dideklarasikan terlebih dahulu

Ikuti langkah-langkah berikut untuk memahami penggunaan variabel PHP:

Langkah	Keterangan
1	<p>Buat file <code>variabel.php</code> di dalam direktori <code>praktik_php</code>, kemudian ketikkan kode berikut:</p> <pre>1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 hello = "Hello World!"; 8 echo \$hello; 9 ?> 10 </body> 11 </html></pre>
2	<p>Simpan file tersebut dan jalankan kode program dengan mengetikkan <code>localhost/dasarWeb/praktik_php/variabel.php</code></p>
3	<p>Modifikasi kode program pada langkah 1 dengan menambahkan tanda \$ pada baris ke-7, sehingga kode program menjadi seperti berikut:</p> <pre>1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$hello = "Hello World!"; 8 echo \$hello; 9 ?> 10 </body> 11 </html></pre>
4	<p>Ulangi langkah ke-2</p>
5	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu (soal no 3)</p>

Praktikum Bagian 4. Konstanta

Berbeda dengan variabel, konstanta berisikan nilai yang sifatnya tetap dan tidak bisa diubah sepanjang program dijalankan. Dalam PHP mendefinisikan konstanta menggunakan fungsi `define()`. Sintaks dasar konstanta adalah:

```
<?php
    define(name, value, case-insensitive)
?>
```

- Parameter `name`, adalah nama konstanta
- Parameter `value`, adalah nilai konstanta
- `case-insensitive`, berarti nama konstanta harus *case-insensitive*, default bernilai `false`

Ikuti langkah-langkah berikut untuk memahami penggunaan variabel dan konstanta PHP:

Langkah	Keterangan
1	<p>Buat file <code>konstanta.php</code> di dalam direktori <code>praktik_php</code>, kemudian ketikkan kode berikut:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$var = "ini adalah contoh variabel"; 8 echo \$var; 9 echo "

"; 10 define("konstanta", "ini adalah contoh konstanta"); 11 echo konstanta; 12 echo "
"; 13 echo KONSTANTA; 14 ?> 15 </body> 16 </html></pre>
2	<p>Simpan file tersebut dan jalankan kode program dengan mengetikkan <code>localhost/dasarWeb/praktik_php/konstanta.php</code>. kemudian amati hasilnya</p>
3	<p>Modifikasi kode program langkah 1 dengan menambahkan parameter <code>"true"</code> di baris ke-10, sehingga kode program menjadi seperti berikut:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$var = "ini adalah contoh variabel"; 8 echo \$var; 9 echo "

"; 10 define("konstanta", "ini adalah contoh konstanta", true); 11 echo konstanta; 12 echo "
"; 13 echo KONSTANTA; 14 ?> 15 </body> 16 </html></pre>
4	Ulangi langkah ke-2
5	Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu (soal no 4)

Praktikum Bagian 5. Tipe Data

Variabel dapat menyimpan data dengan tipe data yang berbeda. PHP mendukung beberapa tipe data, yaitu nilai skalar (integer, float, string, dan boolean), array, tipe data khusus (resource dan NULL). Berikut ini adalah penjelasan masing-masing tipe data:

Tipe Data	Keterangan
Integer	Integer adalah tipe data yang merepresentasikan bilangan bulat, positif maupun negatif, dan bukan pecahan. Rentang bilangan integer adalah -2,147,483,648 sampai 2,147,483,647.
Float	Float disebut juga sebagai bilangan pecahan atau sering juga disebut sebagai bilangan real yang mengandung angka decimal di belakang koma. Contoh bilangan float adalah 3,14.
String	String adalah tipe data karakter yang biasanya diekspresikan dengan diapit oleh tanda petik ganda (" ") atau petik tunggal (' '). Contoh string adalah "ini adalah string".
Boolean	Boolean adalah tipe data yang merepresentasikan nilai kebenaran (<i>truth value</i>). Sebaran nilai Boolean adalah <i>true</i> dan <i>false</i> .
Array	Array adalah variabel yang menyimpan sekelompok nilai, yang dapat diidentifikasi berdasarkan posisinya atau indeksinya.
Resource	Resource adalah tipe data yang menyimpan referensi resource eksternal.
NULL	NULL menunjukkan sebuah variabel belum memiliki nilai/data.

Dalam bahasa pemrograman PHP tidak seperti bahasa pemrograman lainnya, tidak perlu menuliskan tipe variabel di depan nama variabel. Di dalam PHP terdapat istilah type juggling, yaitu proses untuk menentukan jenis tipe data yang “cocok” untuk operasi saat itu dan PHP akan otomatis mengkonversinya. Ikuti langkah-langkah berikut untuk memahami tipe data dalam PHP:

Langkah	Keterangan
1	<p>Buat file <code>tipedata.php</code> di dalam direktori <code>praktik_php</code>, kemudian ketikkan kode berikut:</p> <pre>1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$a = "Hello World!"; 8 \$b = 99; 9 \$c = 8.8; 10 \$d = NULL; 11 var_dump (\$a,\$b,\$c,\$d); 12 ?> 13 </body> 14 </html></pre>

2	Simpan file tersebut dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/tipedata.php
3	Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu (soal no 5)
4	<p>Buat file typedata2.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 </head> 5 <body> 6 <h2> Konversi Tipe Data</h2> 7 <?php 8 \$a = 10.5; 9 \$b = "9 kucing"; 10 echo \$a; 11 echo "
".(integer) \$a; 12 echo "
". \$b; 13 echo "
".(integer)\$b; 14 ?> 15 </body> 16 </html> </pre>
5	Simpan file tersebut dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/tipedata2.php
6	Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu (soal no 6)

Praktikum Bagian 6. Operator

Dalam PHP terdapat empat jenis operator, yaitu assignment operator, operator aritmatik, operator relasional, dan operator logika. Assignment operator adalah operator yang digunakan untuk memberikan sebuah nilai yang akan dimasukkan ke dalam variabel. Assignment operator adalah tanda sama dengan “=”. Operator aritmatik adalah operator yang digunakan dalam operasi aritmatika. Berikut ini adalah contoh operator aritmatik:

Operator	Keterangan
+	Operator penjumlahan
-	Operator pengurangan
*	Operator perkalian
/	Operator pembagian
%	Operator modulus (siswa hasil pembagian)
++	Operator untuk menaikkan nilai variabel sebesar 1
--	Operator untuk menurunkan nilai variabel sebesar 1

Operator relasional adalah operator yang digunakan untuk membandingkan 2 buah nilai. Hasil operasi hanya ada dua, yaitu true dan false. Berikut adalah contoh operator relasional:

Operator	Keterangan
<code>\$a == \$b</code>	Memeriksa apakah \$a sama dengan \$b
<code>\$a != \$b</code>	Memeriksa apakah \$a tidak sama dengan \$b
<code>\$a < \$b</code>	Memeriksa apakah \$a kurang dari \$b
<code>\$a > \$b</code>	Memeriksa apakah \$a lebih besar dari \$b
<code>\$a <= \$b</code>	Memeriksa apakah \$a kurang dari sama dengan \$b
<code>\$a >= \$b</code>	Memeriksa apakah \$a lebih dari sama dengan \$b

Operator logika adalah operator yang digunakan untuk mengoperasikan dua buah operand yang bertipe Boolean. Contoh dari operator logika adalah:

Operator	Keterangan
<code>\$a and \$b</code>	Bernilai benar jika \$a dan \$b keduanya benar
<code>\$a or \$b</code>	Bernilai benar jika \$a atau \$b bernilai benar
<code>\$a xor \$b</code>	Bernilai benar jika \$a atau \$b bernilai benar, tetapi tidak keduanya
<code>\$a && \$b</code>	Logika AND
<code>\$a \$b</code>	Logika OR
<code>!\$a</code>	Logika NOT

Ikuti langkah-langkah berikut untuk lebih memahami operator di dalam PHP:

Langkah	Keterangan
1	<p>Buat file <code>operator.php</code> di dalam direktori <code>praktik_php</code>, kemudian ketikkan kode berikut:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$a =5; 8 \$b =2; 9 echo "hasil penambahan \$a dan \$b adalah "; echo \$a + \$b; 10 echo "
 hasil pengurangan \$a dan \$b adalah "; echo \$a - \$b; 11 echo "
 hasil perkalian \$a dan \$b adalah "; echo \$a * \$b; 12 echo "
 hasil pembagian \$a dan \$b adalah "; echo \$a / \$b; 13 echo "
 hasil sisa pembagian \$a dan \$b adalah "; echo \$a % \$b; 14 ?> 15 </body> 16 </html> </pre>
2	Simpan file tersebut dan jalankan kode program dengan mengetikkan <code>localhost/dasarWeb/praktik_php/operator.php</code>
3	Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu (soal no 7)

4	<p>Buat file operator_2.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$a = 12; 8 \$b = 14; 9 echo "\$a < \$b adalah "; var_dump(\$a < \$b); 10 echo "
 \$a > \$b adalah "; var_dump(\$a > \$b); 11 echo "
 \$a == \$b adalah "; var_dump(\$a == \$b); 12 echo "
 \$a != \$b adalah "; var_dump(\$a != \$b); 13 ?> 14 </body> 15 </html> </pre>
5	<p>Simpan file tersebut dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/operator_2.php</p>
6	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu (soal no 8)</p>
7	<p>Buat file operator_3.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$a=true; 8 \$b=false; 9 10 echo "nilai a AND b adalah "; var_dump(\$a and \$b); 11 echo "
 nilai a OR b adalah "; var_dump(\$a or \$b); 12 echo "
 nilai a XOR b adalah "; var_dump(\$a xor \$b); 13 echo "

 nilai a && b adalah "; var_dump(\$a && \$b); 14 echo "
 nilai a b adalah "; var_dump(\$a \$b); 15 echo "
 nilai !a b adalah "; var_dump(!\$a \$b); 16 ?> 17 </body> 18 </html> </pre>
8	<p>Simpan file tersebut dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/operator_3.php</p>
9	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu (soal no 9)</p>
10	<p>Buat file operator_4.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p>

	<pre> 1 <?DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$x = 4; 8 \$x += 3; 9 echo "hasil operasi tersebut adalah = \$x"; 10 ?> 11 </body> 12 </html> </pre>
11	Simpan file tersebut dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/operator_4.php
12	Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu (soal no 10)
13	<p>Modifikasi kode program langkah ke-4, ganti operator “+=” pada baris ke-8 dengan operator-operator berikut ini:</p> <ol style="list-style-type: none"> “-=” “*=” “/=” “%=” “.= ”
14	Amati hasil dari operator-operator pada langkah ke-12, lalu simpulkan hasil pengamatanmu (soal no 11)
15	<p>Buat file operator_5.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> 1 <?DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$nomor = 1; 8 while(\$nomor <= 5) { 9 echo \$nomor++; 10 } 11 ?> 12 </body> 13 </html> </pre>
16	Simpan file tersebut dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/operator_5.php
17	Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu (soal no 12)

18	<p>Modifikasi kode program pada langkah ke-14 dengan mengubah baris ke-9 dengan pre-increment, sehingga kode program menjadi seperti berikut:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$nomor = 1; 8 while(\$nomor <= 5) { 9 echo ++\$nomor; 10 } 11 ?> 12 </body> 13 </html> </pre>
19	<p>Simpan file tersebut dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/operator_5.php</p>
20	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu (soal no 13)</p>

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 7. 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 array_1.php di dalam direktori praktik_php,</p>  <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <h2> Indexed Array </h2> 7 <?php 8 \$buah = array("Mangga", "Apel", "Jeruk"); 9 10 echo "Buah ", \$buah[0], " dan ", \$buah[1], " rasanya manis sekali"; 11 ?> 12 </body> 13 </html> </pre> <p>kemudian ketikkan kode berikut:</p>
2	Simpan file dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/array_1.php
3	Amati hasil yang ditampilkan
4	Buat file baru dengan nama array_2.php, kemudian ketikkan kode berikut ini:

	<pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <h2> Indexed Array </h2> 7 <?php 8 \$buah[0] = "Mangga"; 9 \$buah[1] = "Apel"; 10 \$buah[2] = "Jeruk"; 11 12 echo "Buah ". \$buah[0]. " dan ". \$buah[1] . " rasanya manis sekali"; 13 ?> 14 15 </body> 16 </html> </pre>
5	Simpan file dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/array_2.php
6	Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 14)

Praktikum Bagian 8. Associative Array

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 dituliskan 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	Buat file baru dengan nama array_3.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:

	<pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 <style> 5 table,tr,td{ 6 border: 1px solid black; 7 } 8 </style> 9 </head> 10 <body> 11 <h2> Associative Array </h2> 12 <?php 13 \$mobil = array(14 'merk' => 'Toyota', 15 'type' => 'Fortuner', 16 'year' => 2017 17); 18 echo '<table>' 19 <tr> 20 <th>Key</th> 21 <th>Value</th> 22 </tr>'; 23 foreach (\$mobil as \$key => \$value){ 24 echo '<tr>' 25 <td>'. \$key .'</td> 26 <td>'. \$value .'</td> 27 </tr>'; 28 } 29 echo '</table>'; 30 <?> 31 </body> 32 </html> </pre>
2	<p>Simpan file dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/array_3.php</p>
3	<p>Amati hasil yang ditampilkan</p>
4	<p>Buat file baru dengan nama array_4.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 <style> 5 table,tr,td{ 6 border: 1px solid black; 7 } 8 </style> 9 </head> 10 <body> 11 <h2> Associative Array </h2> 12 <?php 13 \$mobil['merk'] = 'Toyota'; 14 \$mobil['type'] = 'Fortuner'; 15 \$mobil['year'] = 2018; 16 17 echo '<table>' 18 <tr> 19 <th>Key</th> 20 <th>Value</th> </pre>

	<pre> 21 </tr>'; 22 foreach (\$mobil as \$key => \$value){ 23 echo ' <tr> 24 <td>'. \$key .'</td> 25 <td>'. \$value .'</td> 26 </tr>'; 27 } 28 echo '</table>'; 29 ?> 30 </body> 31 </html> </pre>
5	Simpan file dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/array_4.php
6	Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 15)

Praktikum Bagian 9. 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:

Langkah	Keterangan
1	<p>Buat file baru dengan nama style.css di dalam direktori praktik_php, 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	Buat file baru dengan nama array_5.php di dalam direktori praktik_php, kemudian

ketikkan kode berikut:

```
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>
```

3

Amati hasil yang ditampilkan

4	<p>Buat file baru dengan nama array_4.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 <style> 5 table, tr, td { 6 border: 1px solid black; 7 } 8 </style> 9 </head> 10 <body> 11 <h2> Associative Array </h2> 12 <?php 13 \$mobil['merk'] = 'Toyota'; 14 \$mobil['type'] = 'Fortuner'; 15 \$mobil['year'] = 2018; 16 17 echo '<table> 18 <tr> 19 <th>Key</th> 20 <th>Value</th> 21 </tr>'; 22 foreach (\$mobil as \$key => \$value) { 23 echo '<tr> 24 <td>'. \$key . '</td> 25 <td>'. \$value . '</td> 26 </tr>'; 27 } 28 echo '</table>'; 29 ?> 30 </body> 31 </html> </pre>
5	Simpan file dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/array_5.php
6	Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 16)

Praktikum Bagian 10. Fungsi

Fungsi (atau Function) dalam bahasa pemrograman adalah kode program yang dirancang untuk menyelesaikan sebuah tugas tertentu, dan merupakan bagian dari program utama. Tujuan memisahkan kode program menjadi fungsi adalah untuk alasan kepraktisan dan kemudahan membuat program utama. Karena jika dijadikan fungsi, maka untuk menjalankan tugas yang sama, kita tinggal memanggil fungsi tersebut, tanpa perlu membuat kembali kode programnya. Kita dapat membuat fungsi sendiri atau menggunakan fungsi yang dibuat oleh programmer lain (lazy programming). PHP menyediakan ribuan fungsi bawaan yang tersedia untuk membantu kita dalam merancang program.

Menggunakan fungsi dalam teori pemrograman sering juga disebut dengan istilah ‘memanggil fungsi’ (calling a function). Fungsi dipanggil dengan menulis nama dari fungsi tersebut, dan diikuti dengan argumen (jika ada). Argumen ditulis di dalam tanda kurung, dan jika jumlah argumen lebih dari satu, maka diantaranya dipisahkan oleh karakter koma. Setelah memproses

nilai inputan, hampir semua fungsi akan memberikan nilai hasil pemrosesan tersebut (walaupun ada fungsi yang tidak memberikan nilai). Cara fungsi memberikan nilainya ini sering disebut dengan ‘mengembalikan nilai’ (return a value). Nilai yang dikembalikan oleh sebuah fungsi dapat ditampung ke dalam variabel, atau langsung ditampilkan ke web browser.

Berikut adalah sintaks penulisan fungsi di dalam PHP:

```
<?php
function nama_fungsi($parameter1, $parameter2){
    kode program fungsi;
    return $nilai_akhir
}
?>
```

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>Buat file baru dengan nama fungsi_1.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <h2> Fungsi </h2> 7 <?php 8 function writeMsg(){ 9 echo "Hello World!"; 10 } 11 12 writeMsg(); 13 ?> 14 </body> 15 </html> </pre>
2	<p>Simpan file dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/fungsi_1.php</p>
3	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 17)</p>

4	<p>Buat file baru dengan nama fungsi_2.php, kemudian ketikkan kode berikut ini:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <h2> Fungsi </h2> 7 <?php 8 function familyName(\$fname, \$year) { 9 echo "\$fname Refsnes. Born in \$year
"; 10 } 11 12 familyName("Hege", "1975"); 13 familyName("Stale", "1978"); 14 familyName("Kai Jim", "1983"); 15 ?> 16 </body> 17 </html> </pre>
5	<p>Simpan file dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/fungsi_2.php</p>
6	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 18)</p>
7	<p>Buat file baru dengan nama fungsi_3.php, kemudian ketikkan kode berikut ini:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <h2> Fungsi Menghitung Luas Lingkaran</h2> 7 <?php 8 echo "Luas Lingkaran dengan jari-jari 7cm = ".luas_lingkaran(7)." cm"; 9 10 function luas_lingkaran(\$jari2){ 11 return 3.14*\$jari2*\$jari2; 12 } 13 ?> 14 </body> 15 </html> </pre>
8	<p>Simpan file dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/fungsi_3.php</p>
9	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 19)</p>

Praktikum Bagian 11. Mengetahui Jumlah Elemen Array

Fungsi count() dapat digunakan untuk mengetahui berapa jumlah elemen di dalam sebuah array. Ikuti langkah-langkah berikut untuk memahami fungsi count() di dalam PHP:

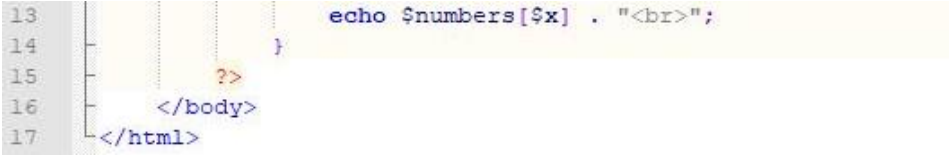
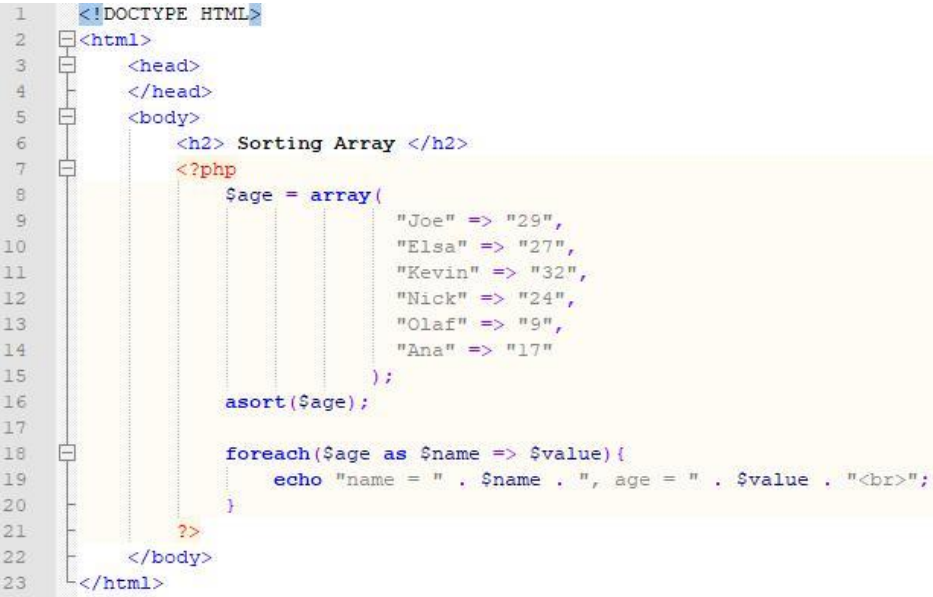
Langkah	Keterangan
1	Buat file baru dengan nama array_6.php di dalam direktori praktik_php, kemudian

	<p>ketikkan kode berikut:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <h2> Fungsi count() </h2> 7 <?php 8 \$menu = array("rawon", "sate", "nasi goreng"); 9 \$arrLength = count(\$menu); 10 11 echo "Menu hari ini adalah:
 "; 12 for(\$x = 0; \$x < \$arrLength; \$x++){ 13 echo \$menu[\$x] . "
"; 14 } 15 echo "
 Saya lapar, saya ingin makan " . "\$menu[2]"; 16 ?> 17 </body> 18 </html> </pre>
2	<p>Simpan file dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/array_6.php</p>
3	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 20)</p>

Praktikum Bagian 12. Mengurutkan Elemen Array

Elemen-elemen di dalam array dapat diurutkan berdasarkan abjad atau numerik. Pengurutan elemen-elemen array dapat dilakukan secara descending (menurun) atau ascending (naik). Beberapa fungsi yang dapat digunakan untuk mengurutkan array adalah sort(), rsort(), asort(), ksort(), arsort(), dan krsort(). Ikuti langkah-langkah berikut untuk memahami kegunaan dari fungsi-fungsi tersebut:

Langkah	Keterangan
1	<p>Buat file baru dengan nama array_7.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <h2> Sorting Array </h2> 7 <?php 8 \$numbers = array(8, 4, 1, 9, 23, 54, 17, 30); 9 sort(\$numbers); 10 11 \$arrLength = count(\$numbers); 12 for(\$x = 0; \$x < \$arrLength; \$x++){ </pre>

	
2	Simpan file dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/array_7.php
3	Amati hasil yang ditampilkan
4	Modifikasi kode program langkah ke-1 dengan mengubah baris 9 dengan fungsi rsort()
5	Ulangi langkah ke-2
6	Amati hasil yang ditampilkan dan jelaskan kegunaan fungsi asort() dan rsort() (soal no 21)
7	<p>Buat file baru dengan nama array_8.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> 
8	Simpan file dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/array_8.php
9	Amati hasil yang ditampilkan
10	<p>Modifikasi kode program langkah ke-7 dengan mengubah baris 16 dengan fungsi berikut ini:</p> <p>a. rsort()</p>

	b. ksort() c. arsort() d. krsort()
11	Ulangi langkah ke-8
12	Amati hasil yang ditampilkan dan jelaskan kegunaan kegunaan dari fungsi <code>assort()</code> , <code>rsort()</code> , <code>ksort()</code> , <code>arsort()</code> , dan <code>krsort()</code> (soal no 22)

Praktikum Bagian 13. String

String tipe data karakter yang biasanya diekspresikan dengan diapit oleh tanda petik ganda (“ ”) atau petik tunggal (‘ ’). 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 (lowercase)
<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

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

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

	<pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 echo strlen("hello world!"); 8 echo "
"; 9 echo str_word_count("hello world!"); 10 echo "
"; 11 echo strrev("hello world!"); 12 echo "
"; 13 echo strtoupper("hello world!"); 14 echo "
"; 15 echo strtolower("hello world!"); 16 echo "
"; 17 echo ucwords("hello world!"); 18 ?> 19 </body> 20 </html> </pre>
2	<p>Simpan file tersebut dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/string1.php</p>
3	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu (soal no 23)</p>
4	<p>Buat file string2.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$numberedString = "123456789012345678901234567890"; 8 \$fivePos = strpos(\$numberedString, "5"); 9 echo "Posisi angka 5 di dalam string adalah \$fivePos"; 10 \$fivePos2 = strpos(\$numberedString, "5", \$fivePos + 1); 11 echo "
 Posisi angka 5 yang kedua di dalam string adalah \$fivePos2"; 12 ?> 13 </body> 14 </html> </pre>
5	<p>Simpan file tersebut dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/string2.php</p>
6	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu (soal no 24)</p>
7	<p>Buat file string3.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 </head> </pre>

	<pre> 5 <body> 6 <?php 7 \$rawString = "Welcome Birmingham Parent. 8 Your replacement is a pleasure to have!"; 9 10 \$malestr = str_replace("replacement", "son", \$rawString); 11 \$femalestr = str_replace("replacement", "daughter", \$rawString); 12 13 echo "Son: ". \$malestr. "
"; 14 echo "Daughter: ". \$femalestr. "
"; 15 16 ?> 17 </body> </html> </pre>
8	Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu (soal no 25)

Praktikum Bagian 8. *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 date.php di dalam direktori praktik_php, 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	<p>Simpan file dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/date.php</p>
3	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 26)</p>
4	<p>Buat file baru dengan nama time.php di dalam direktori praktik_php, 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 dengan mengetikkan localhost/dasarWeb/praktik_php/time.php</p>
6	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 27)</p>

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.