

Pertemuan 2

SINTAKS DASAR PHP

Yunia Ikawati

Workshop Pemrograman Web

Topik

PHP

Sintak Dasar PHP

PHP Variabel

PHP Output

Tipe Data PHP

PHP String

Casting

Fungsi Matematika

Fungsi Aritmatika

Fungsi Logika

Fungsi
Increment/Decrement

PHP

PHP adalah akronim untuk "PHP: Hypertext Preprocessor"

PHP adalah bahasa skrip sumber terbuka yang banyak digunakan

Skrip PHP dijalankan di server

PHP dapat diunduh dan digunakan secara gratis

Apa itu File PHP?

- File PHP dapat berisi teks, HTML, CSS, JavaScript, dan kode PHP
- Kode PHP dieksekusi di server, dan hasilnya dikembalikan ke browser sebagai HTML biasa
- File PHP memiliki ekstensi ".php"

Apa yang Dapat Dilakukan PHP?

- PHP dapat menghasilkan konten halaman yang dinamis
- PHP dapat membuat, membuka, membaca, menulis, menghapus, dan menutup file di server
- PHP dapat mengumpulkan data formulir
- PHP dapat mengirim dan menerima cookie
- PHP dapat menambahkan, menghapus, mengubah data di basis data Anda
- PHP dapat digunakan untuk mengontrol akses pengguna
- PHP dapat mengenkripsi data
- Dengan PHP, Anda tidak terbatas pada keluaran HTML. Anda dapat mengeluarkan gambar atau file PDF. Anda juga dapat mengeluarkan teks apa pun, seperti XHTML dan XML.

Mengapa PHP?

PHP berjalan di berbagai platform (Windows, Linux, Unix, Mac OS X, dll.)

PHP kompatibel dengan hampir semua server yang digunakan saat ini (Apache, IIS, dll.)

PHP mendukung berbagai macam basis data

PHP gratis. Unduh dari sumber resmi PHP: www.php.net

PHP mudah dipelajari dan berjalan secara efisien di sisi server

Sintaksis Dasar PHP

- Skrip PHP dapat ditempatkan di mana saja dalam dokumen.
- Skrip PHP dimulai dengan `<?php` dan diakhiri dengan `?>`
- Ekstensi file default untuk file PHP adalah ".php".
- File PHP biasanya berisi tag HTML, dan beberapa kode skrip PHP.

```
<?php  
// PHP code goes here  
?>
```

Contoh File PHP

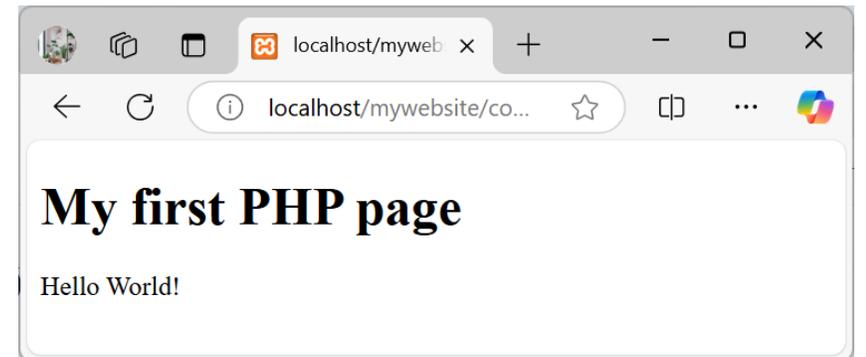
- Di samping ini, contoh file PHP sederhana, dengan skrip PHP yang menggunakan fungsi PHP bawaan "echo" untuk menampilkan teks "Hello World!" di halaman web:

```
<!DOCTYPE html>
<html>
<body>

<h1>My first PHP page</h1>

<?php
echo "Hello World!";
?>

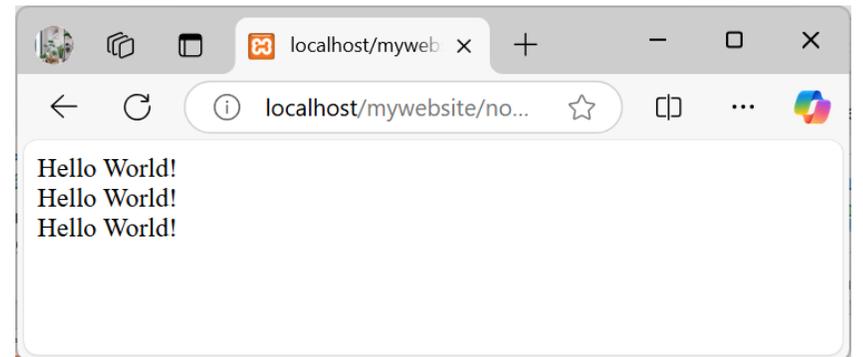
</body>
</html>
```



PHP Case Sensitive

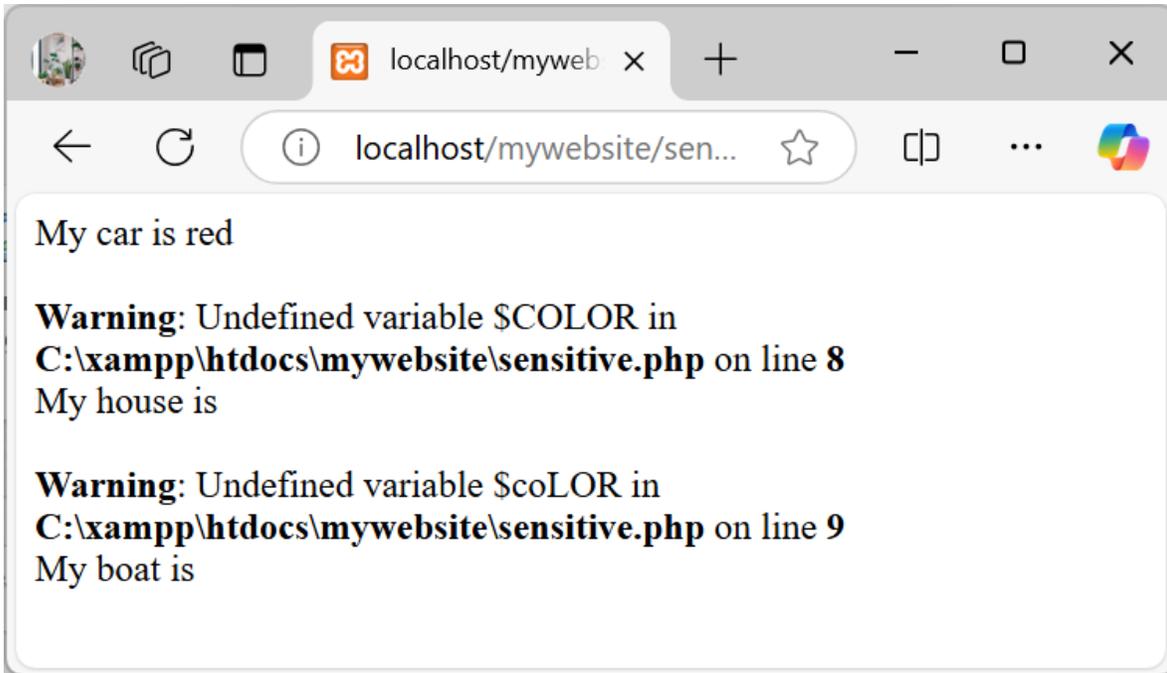
- Pada PHP keywords (e.g. **if**, **else**, **while**, **echo**, etc.), classes, functions, dan fungsi yang ditentukan pengguna tidak case-sensitive.

```
<!DOCTYPE html>
<html>
<body>
<?php
ECHO "Hello World!<br>";
echo "Hello World!<br>";
EcHo "Hello World!<br>";
?>
</body>
</html>
```



PHP Case Sensitive

Jika bukan keyword/ fungsi maka PHP bersifat case-sensitive.



A screenshot of a web browser window. The address bar shows 'localhost/mywebsite/sen...'. The page content displays 'My car is red' followed by two warning messages:

```
Warning: Undefined variable $COLOR in  
C:\xampp\htdocs\mywebsite\sensitive.php on line 8  
My house is
```

```
Warning: Undefined variable $coLOR in  
C:\xampp\htdocs\mywebsite\sensitive.php on line 9  
My boat is
```

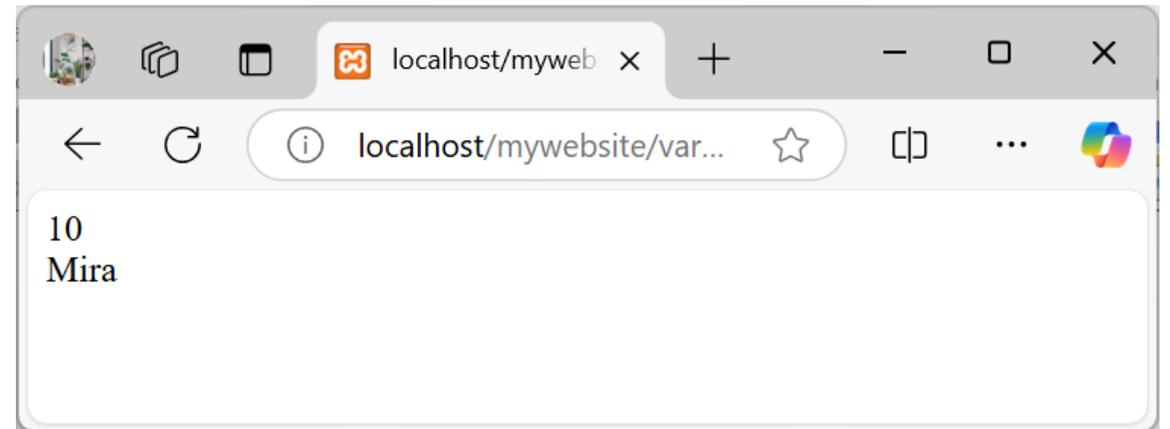
```
<!DOCTYPE html>  
<html>  
<body>  
<?php  
    $color = "red";  
    echo "My car is " . $color . "<br>";  
    echo "My house is " . $COLOR . "<br>";  
    echo "My boat is " . $coLOR . "<br>";  
?>  
</body>  
</html>
```

PHP Variables

- Variabel adalah "wadah" untuk menyimpan informasi.
- Dalam PHP, sebuah variabel dimulai dengan tanda \$, diikuti dengan nama variabel:

```
<!DOCTYPE html>
<html>
<body>
<?php
    $x = 10;
    $y = "Mira";

    echo $x;
    echo "<br>";
    echo $y;
?>
</body>
</html>
```



Aturan untuk variabel PHP

Variabel dimulai dengan tanda \$, diikuti dengan nama variabel

Nama variabel harus dimulai dengan huruf atau karakter garis bawah

Nama variabel tidak dapat dimulai dengan angka

Nama variabel hanya dapat berisi karakter alfanumerik dan garis bawah (A-z, 0-9, dan _)

Nama variabel peka huruf besar-kecil (\$age dan \$AGE adalah dua variabel yang berbeda)

PHP Output

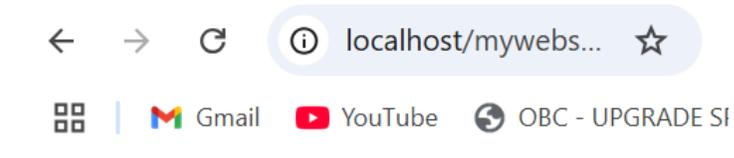
- Dengan PHP, ada dua cara dasar untuk mendapatkan output: **echo** dan **print**.
- **Echo** dan **print** digunakan untuk menampilkan data ke layar.
- Echo tidak memiliki nilai balik
- Print memiliki nilai balik 1 sehingga dapat digunakan dalam ekspresi.
- Echo dapat mengambil beberapa parameter (meskipun penggunaan seperti itu jarang) sementara print dapat mengambil satu argumen.
- Echo sedikit lebih cepat daripada print.

PHP Output Print

```
<!DOCTYPE html>
<html>
<body>
<?php
$txt1 = "Belajar PHP";
$txt2 = "PENS";

print "<h2>$txt1</h2>";
print "<p>Study PHP at $txt2</p>";

print '<h2>' . $txt1 . '</h2>';
print '<p>Study PHP at ' . $txt2 . '</p>';
?>
</body>
</html>
```



Belajar PHP

Study PHP at PENS

Belajar PHP

Study PHP at PENS

- Saat menggunakan tanda kutip ganda, variabel dapat disisipkan ke string seperti pada contoh disamping.
- Saat menggunakan tanda kutip tunggal, variabel harus disisipkan menggunakan operator .

PHP Output Echo

```
<?php
$txt1 = "Belajar PHP";
$txt2 = "PENS";

echo "<h2>$txt1</h2>";
echo "<p>Study PHP at $txt2</p>";

echo '<h2>' . $txt1 . '</h2>';
echo '<p>Study PHP at ' . $txt2 . '</p>';
?>
```

Belajar PHP

Study PHP at PENS

Belajar PHP

Study PHP at PENS

Tipe Data PHP

Variabel dapat menyimpan data dengan tipe yang berbeda, dan tipe data yang berbeda dapat melakukan hal yang berbeda.

PHP mendukung tipe data berikut:

- String
- Integer
- Float (floating point numbers - also called double)
- Boolean
- Array
- Object
- NULL
- Resource

Tipe Data PHP

```
<!DOCTYPE html>
<html>
<body>
<?php
$a = 5;
$b = "Hello world!";
$c = 'Hello world!';
$d = 5985;
$e = 10.365;
$f = true;
$cars = array("Volvo", "BMW", "Toyota");

var_dump($a);
echo "<br>";
var_dump($b);
echo "<br>";
var_dump($c);
echo "<br>";
var_dump($d);
echo "<br>";
var_dump($e);
echo "<br>";
var_dump($f);
echo "<br>";
var_dump($cars);
-?>
</body>
</html>
```

```
int(5)
string(12) "Hello world!"
string(12) "Hello world!"
int(5985)
float(10.365)
bool(true)
array(3) { [0]=> string(5) "Volvo" [1]=> string(3) "BMW" [2]=> string(6) "Toyota" }
```

PHP String

- Fungsi PHP **strlen()** mengembalikan panjang string.
- Fungsi PHP **str_word_count()** menghitung jumlah kata dalam string.
- Fungsi PHP **strpos()** mencari teks tertentu dalam string.

```
<!DOCTYPE html>
<html>
<body>
<?php
    $x = "John";
    echo "Hello $x";
    echo "<br>";
    echo 'Hello $x';
    echo "<br>";
    echo strlen("Hello world!");
    echo "<br>";
    echo str_word_count("Hello world!");
    echo "<br>";
    echo strpos("Hello world!", "world");
?>
</body>
</html>
```

Hello John
Hello \$x
12
2
6

Modifikasi PHP String

- Fungsi **strtoupper()** mengembalikan string dalam huruf kapital
- Fungsi **strtolower()** mengembalikan string dalam huruf kecil
- Fungsi PHP **str_replace()** mengganti beberapa karakter dengan beberapa karakter lain dalam string.
- Fungsi PHP **strrev()** membalikkan string.
- **Trim()** menghapus spasi dari awal atau akhir
- Fungsi PHP **explode()** membagi string menjadi array.
 - Parameter pertama fungsi **explode()** mewakili "pemisah". "Pemisah" menentukan tempat untuk membagi string. Gunakan karakter spasi sebagai pemisah

```
<!DOCTYPE html>
<html>
<body>
<?php
    $x = "Hello World!";
    echo strtoupper($x);
    echo "<br>";
    echo strtolower($x);
    echo "<br>";
    echo strrev($x);
    echo "<br>";
    echo str_replace("World", "Rony", $x);
    echo "<br>";
    echo trim($x);
    echo "<br>";
    $y = explode(" ", $x);
    print_r($y);
?>
</body>
</html>
```

```
HELLO WORLD!
hello world!
!dlroW olleH
Hello Rony!
Hello World!
Array ( [0] => Hello [1] => World! )
```

Casting

mengubah variabel dari satu tipe data ke tipe data lain dengan casting.

```
<?php
$a = 5;           // Integer
$b = "5.34";     // Float
$c = "25 kilometers"; // String
$d = "kilometers 25"; // String
$e = "hello";    // String
$f = true;       // Boolean
$g = NULL;       // NULL

$a = (string) $a;
$b = (int) $b;
$c = (float) $c;
$d = (bool) $d;
$e = (array) $e;
$f = (int) $f;
$g = (int) $g;

//To verify the type of any object in PHP, use the var_dump() function:
var_dump($a);
var_dump($b);
var_dump($c);
var_dump($d);
var_dump($e);
var_dump($f);
var_dump($g);
-?>
```

```
string(1) "5"
int(5)
float(25)
bool(true)
array(1) {
    [0]=>
        string(5) "hello"
}
int(1)
int(0)
```

Fungsi Matematika

PHP memiliki serangkaian fungsi matematika untuk melakukan tugas matematika pada angka.

Fungsi pi() mengembalikan nilai PI

Fungsi min() dan max() dapat digunakan untuk menemukan nilai terendah atau tertinggi dalam daftar argumen

Fungsi abs() mengembalikan nilai absolut (positif) dari sebuah angka

Fungsi sqrt() mengembalikan akar kuadrat dari sebuah angka

Fungsi round() membulatkan angka floating-point ke bilangan bulat terdekat

Fungsi rand() menghasilkan angka acak

```
<?php
echo("Nilai pi: " . pi());
echo "<br>";
echo("Nilai min: " . min(0, 150, 30, 20, -8, -200) . "<br>");
echo("Nilai max: " . max(0, 150, 30, 20, -8, -200));
echo "<br>";
echo("Nilai abs: " . abs(-6.7));
echo "<br>";
echo("Nilai sqrt: " . sqrt(64));
echo "<br>";
echo("Nilai round: " . round(0.49) . "<br>");
echo("Nilai round: " . round(-4.40) . "<br>");
echo("Nilai rand: " . rand(10, 100));
?>
```

Nilai pi: 3.1415926535898

Nilai min: -200

Nilai max: 150

Nilai abs: 6.7

Nilai sqrt: 8

Nilai round: 0

Nilai round: -4

Nilai rand: 81

Konstanta

- Konstanta mirip dengan sebuah variabel.
- Konstanta adalah pengenalan (nama) untuk nilai sederhana. Nilai konstanta tidak dapat diubah.
- Nama konstanta yang valid dimulai dengan huruf atau garis bawah (tanpa tanda \$ sebelum nama konstanta).

```
<?php
// case-sensitive constant name
define("GREETING", "Welcome to PENS!");
echo "1 " . GREETING;
echo "<br>";
const Jurusanku = "Teknik Informatika";
echo "2 " . Jurusanku . "<br>";
define("cars", [
    "Alfa Romeo",
    "BMW",
    "Toyota"
]);
echo "3 " . cars[0]. "<br>";
define("PANGGIL", "Memanggil Fungsi!");
function myTest() {
    echo "4 " . PANGGIL;
}
myTest();
?>
```

1 Welcome to PENS!
2 Teknik Informatika
3 Alfa Romeo
4 Memanggil Fungsi!

Konstanta

- Untuk membuat konstanta, gunakan fungsi define().
- Dapat membuat konstanta dengan menggunakan kata kunci const.
- const vs. define()
 - const tidak dapat dibuat di dalam cakupan blok lain, seperti di dalam fungsi atau di dalam pernyataan if.
 - define dapat dibuat di dalam cakupan blok lain.
- Pada PHP7 dapat membuat konstanta Array menggunakan fungsi define().
- Konstanta secara otomatis bersifat global dan dapat digunakan di seluruh skrip.

Operator di PHP

Operator digunakan untuk menentukan sebuah nilai dengan melakukan suatu prosedur, atau dengan suatu operasi dari beberapa nilai.

Nilai yang digunakan dalam sebuah operasi disebut operand.

Arithmetic Operator (Operator Aritmatika)

Operator aritmatika PHP digunakan dengan nilai numerik untuk melakukan operasi aritmatika umum, seperti penjumlahan, pengurangan, perkalian, dan sebagainya.

Operator	Contoh	Deskripsi	Hasil
+	$\$a + \b	Penjumlahan	Jumlah $\$a$ dan $\$b$.
-	$\$a - \b	Pengurangan	Selisih $\$a$ dan $\$b$.
*	$\$a * \b	Perkalian	Hasil kali $\$a$ dan $\$b$.
/	$\$a / \b	Pembagian	Quotient dari $\$a$ dan $\$b$.
%	$\$a \% \b	Modulus	Remainder dari $\$a$ dibagi $\$b$.

```
<?php
$x = 10;
$y = 6;

echo $x * $y . "<br>";
echo $x + $y . "<br>";
echo $x / $y ;
?>
```

```
60
16
1.66666666666667
```

Assignment Operator

- Tanda sama dengan (=) digunakan untuk mengatur atau menetapkan nilai suatu variabel. Oleh karena itu tanda tersebut dikenal sebagai operator penugasan.

Assignment	Same as...	Description
<code>x = y</code>	<code>x = y</code>	The left operand gets set to the value of the expression on the right
<code>x += y</code>	<code>x = x + y</code>	Addition
<code>x -= y</code>	<code>x = x - y</code>	Subtraction
<code>x *= y</code>	<code>x = x * y</code>	Multiplication
<code>x /= y</code>	<code>x = x / y</code>	Division
<code>x %= y</code>	<code>x = x % y</code>	Modulus

```
<?php
$x = 50;
$x -= 30;
$y = 5;
$y *= 6;

echo $x . "<br>";
echo $y;
```

20

30

Comparison Operator (Operator Perbandingan)

- Operator perbandingan digunakan untuk menguji suatu kondisi atau membandingkan dua nilai (angka atau string). Ekspresi yang menggunakan operator perbandingan akan selalu menghasilkan nilai boolean, yaitu antara true (benar) atau false (salah).

Operator	Name	Example	Result
==	Equal	<code>\$x == \$y</code>	Returns true if \$x is equal to \$y
===	Identical	<code>\$x === \$y</code>	Returns true if \$x is equal to \$y, and they are of the same type
!=	Not equal	<code>\$x != \$y</code>	Returns true if \$x is not equal to \$y
<>	Not equal	<code>\$x <> \$y</code>	Returns true if \$x is not equal to \$y
!==	Not identical	<code>\$x !== \$y</code>	Returns true if \$x is not equal to \$y, or they are not of the same type
>	Greater than	<code>\$x > \$y</code>	Returns true if \$x is greater than \$y
<	Less than	<code>\$x < \$y</code>	Returns true if \$x is less than \$y
>=	Greater than or equal to	<code>\$x >= \$y</code>	Returns true if \$x is greater than or equal to \$y
<=	Less than or equal to	<code>\$x <= \$y</code>	Returns true if \$x is less than or equal to \$y
<=>	Spaceship	<code>\$x <=> \$y</code>	Returns an integer less than, equal to, or greater than zero, depending on if \$x is less than, equal to, or greater than \$y. Introduced in PHP 7.

```
<?php
$x = 100;
$y = "100";
$a = 10;
$b = 50;

var_dump($x == $y);
echo "<br>";
var_dump($x != $y);
echo "<br>";
var_dump($a < $b);

?>
```

bool(true)
bool(false)
bool(true)

Operator logika

Operator logika PHP digunakan untuk menggabungkan pernyataan kondisional.

Operator	Contoh	Name	Hasil
And	<code>\$a and \$b</code>	And	TRUE jika <code>\$a</code> dan <code>\$b</code> TRUE .
Or	<code>\$a or \$b</code>	Or	TRUE jika <code>\$a</code> atau <code>\$b</code> TRUE .
Xor	<code>\$a xor \$b</code>	Xor	TRUE jika salah satu dari <code>\$a</code> atau <code>\$b</code> TRUE
!	<code>! \$a</code>	Not	TRUE jika <code>\$a</code> bukan TRUE .
&&	<code>\$a && \$b</code>	And	TRUE jika <code>\$a</code> dan <code>\$b</code> TRUE .
	<code>\$a \$b</code>	Or	TRUE jika <code>\$a</code> atau <code>\$b</code> TRUE .

Operator Increment / Decrement PHP

- Operator increment PHP digunakan untuk menambah nilai variabel.
- Operator decrement PHP digunakan untuk mengurangi nilai variabel.

Operator	Same as...	Description
++\$x	Pre-increment	Increments \$x by one, then returns \$x
\$x++	Post-increment	Returns \$x, then increments \$x by one
--\$x	Pre-decrement	Decrements \$x by one, then returns \$x
\$x--	Post-decrement	Returns \$x, then decrements \$x by one

```
<?php
$x = 10;
$y = 5;
echo (++$x . "<br>");
echo ($x++ . "<br>");
echo (--$y . "<br>");
echo ($y--);
?>
```

11
11
4
4

PHP String Operators

- PHP memiliki dua operator yang dirancang khusus untuk string.

Operator	Name	Example	Result
.	Concatenation	<code>\$txt1 . \$txt2</code>	Menggabungkan \$txt1 and \$txt2
.=	Concatenation assignment	<code>\$txt1 .= \$txt2</code>	Menambahkan \$txt2 to \$txt1

```
<?php
$txt1 = "Kota";
$txt2 = " Lamongan!";
echo ("1" . $txt1 . $txt2 . "<br>");
$txt1 .= $txt2;
echo ("2" . $txt1);
?>
```

1Kota Lamongan!
2Kota Lamongan!

PRAKTIKUM

1. Bahan dan Peralatan

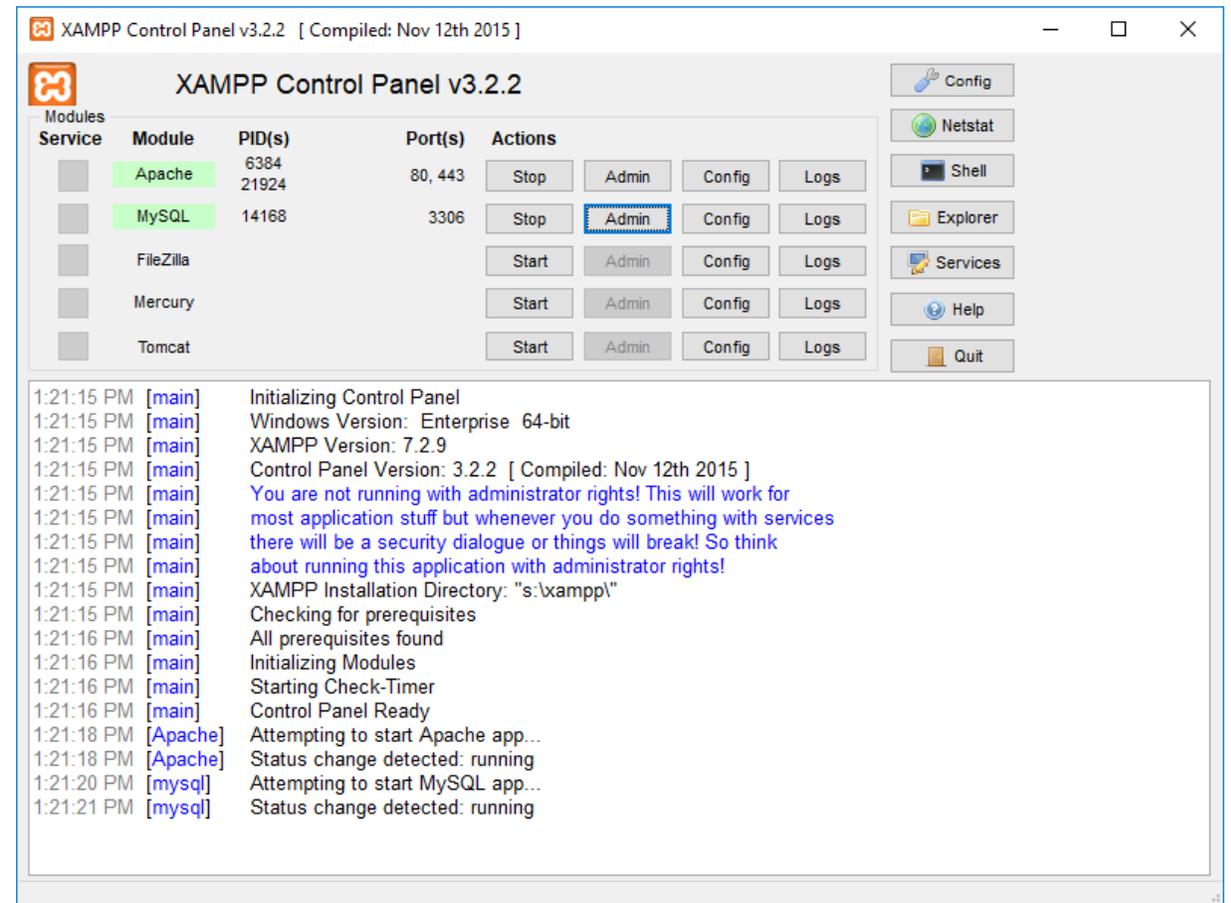
- a. Komputer atau laptop
- b. *Notepad/Notepad++*
- c. Browser (Mozilla, Chrome dll)
- d. Software XAMPP

2. Langkah Praktikum

- a. Bukalah notepad/notepad++/editplus/visual code atau program text editor lain
- b. Ketikkan Script masing-masing percobaan dan simpan dengan ekstensi **.php**
- c. Jika menggunakan **XAMPP** maka simpan file tersebut di drive :c , pilih folder **xampp** lalu pilih folder **htdocs**, simpan file .php tersebut pada folder tersebut
- d. Tampilkan Pada Browser (<http://localhost/file.php>)
- e. Pastikan XAMPP(Apache,MySql,Filezilla)
- f. Capture hasil percobaan tersebut
- g. Analisa hasil tampilan percobaannya

Menjalankan File PHP dengan XAMPP

- Install XAMPP
- Jalankan XAMPP Control Panel untuk mengaktifkan service Apache dan MySQL dengan pilih actions start
- Buka Browser lalu ketikkan (<http://localhost/file.php>)



Percobaan 1

Perintah Menampilkan Kalimat

```
<html>
  <head>
    <title>Latihan-01</title>
  </head>
<body>
  <?php
    echo "Semangat Belajar PHP!!!";
    echo "<br>";
    print "Pasti Bisa";
    print "<br>";
    printf "Sukses Selalu";
  ?>
</body>
</html>
```

Percobaan 2

Menampilkan nilai variabel PHP

```
<html>
  <head>
    <title>Latihan-02</title>
  </head>
  <body>
    <?php
      $i="PENS (Politeknik Elektronika Negeri
      Surabaya)";
      echo $i;
    ?>
  </body>
</html>
```

Percobaan 3

Menggabungkan nilai variabel PHP

```
<html>
<body>
    <?php
        $txt1="Mahasiswa PENS Lamongan";
        $txt2="Pasti Bisa";
        echo $txt1 . " " . $txt2 ;
    ?>
</body>
</html>
```

Percobaan 4

Memberikan Perintah Komentar Pada PHP

```
<html>
<body>
    <?php
        //Ini hanya sebuah komentar
        print "Teknik Informatika";
        /* Hanya komentar
           Dalam dua baris*/
    ?>
</body> </html>
```

Percobaan 5

Memberikan Perintah Komentar Pada PHP

```
<?php
    echo "First line <br>"; // You won't see me in the output
    // I'm a one liner comment

    /* Same thing, you won't
    see this in the output file
    */
    echo "The above comment spans two lines <br>";

    # Hi i'm a Unix shell-style comment
    # Too bad you can't see me
    echo "View the source of this file, you'll see no comments here
    <br>";
?>
```

Percobaan 6

Menampilkan Pengaturan Nama Variabel Pada PHP

```
<?php
```

```
    $kampus = "PENS"; // valid
```

```
    echo $kampus;
```

```
    echo "<br>";
```

```
    $jurusan_MM2 = "Multimedia"; // valid
```

```
    echo $ jurusan_MM2;
```

```
    echo "<br>";
```

```
    $123Jurusan = "Teknik Informatika"; // invalid, starts with number
```

```
    echo $123Jurusan;
```

```
    echo "<br>";
```

```
    $Jurusan-123 = "Teknik Komputer"; // invalid
```

```
    echo $Jurusan-123;
```

```
?>
```

Percobaan 7

Perintah Aritmatic Sederhana

```
<?php
    $angka1=10;
    $angka2=2;
    $hasilTambah=$angka1+$angka2;
    $hasilKurang=$angka1-$angka2;
    $hasilKali=$angka1*$angka2;
    echo $hasilTambah;
    echo "<br>";
    echo $hasilKurang;
    echo "<br>";
    echo $hasilKali;
?>
```

Percobaan 8

Perintah Aritmatic

```
<?php
    $x=20;
    $y=5;
    echo "x = ".$x."<br>";
    echo "y = ".$y."<br>";
    $h = $x / $y;
    echo $x." / ".$y." = ".$h."<br>";
    $x++;
    echo "x = ".$x."<br>";
    $x += 3;
    echo "x = ".$x."<br>";
    $x /=4;
    echo "x = ".$x;
?>
```

Percobaan 9

Perintah Pengaturan Type Data

```
<?php
    $angka="469.80";
    echo"angkanya adalah : $angka <br> ";
    settype($angka,"Double");
    print("type Double = $angka<br>");
    settype($angka,"Integer");
    print("type Integer = $angka");
?>
```

Percobaan 10

```
<?php
$a = 1; // $a is a global variable
function Test()
{
    echo $a; // try to print $a, but $a is not defined here
}
Test();
?>
```

Percobaan 11

```
<?php
$a = 1; // $a is defined in global scope ...
$b = 2; // $b too

function Sum()
{
    global $a, $b; // now $a and $b are available in Sum()
    $b = $a + $b;
}

Sum();
echo $b;
?>
```

Percobaan 12

```
<?php
$numbers = array(1, 2, 3, 4, 5, 6);
$age = array("mom" => 45, "pop" => 50, "bro" => 25);
$mixed = array("hello" => "World", 2 => "It's two");

echo "numbers[4] = {$numbers[4]} <br>";
echo "My mom's age is {$age['mom']} <br>";
echo "mixed['hello'] = {$mixed['hello']} <br>";
echo "mixed[2] = {$mixed[2]}";
?>
```

Percobaan 13

```
<?php
$abc = 10; // $abc is an integer
$xyz = (boolean) $abc; // $xyz is a boolean
echo "abc is $abc and xyz is $xyz <br>";
?>
```

Percobaan 14

```
<?php
$fruit = 'jamblang';

echo "My favourite fruit is $fruit <br>";
echo "I lied, actually I hate $fruit <br>";

echo "\r\n My first line \r\n and my second line <br>\r\n";
echo " Though I use \r\n this string is still on one line <br>";
?>
```

TUGAS

- Kerjakan seluruh program pada contoh dan percobaan
- Berikan penjelasan singkat tentang programnya
- Buat laporannya dan dikumpulkan di ethol