



Pemrograman Web

# **OPERATORS, CONTROL STRUCTURES**



## Tujuan Pembelajaran

- Penerapan operator untuk melakukan operasi pada variabel dan nilai
- Penerapan struktur kontrol untuk melakukan tindakan yang berbeda

# Operator

Operator digunakan untuk melakukan operasi pada variabel dan nilai.

PHP membagi operator dalam beberapa kelompok berikut ini:

- Operator aritmatika
- Operator penugasan
- Operator perbandingan
- Operator Increment / Decrement
- Operator logika
- Operator string
- Operator larik
- Operator penugasan bersyarat

# Operator Aritmatika

Operator	Name	Example	Result
+	Addition	$\$x + \$y$	Sum of $\$x$ and $\$y$
-	Subtraction	$\$x - \$y$	Difference of $\$x$ and $\$y$
*	Multiplication	$\$x * \$y$	Product of $\$x$ and $\$y$
/	Division	$\$x / \$y$	Quotient of $\$x$ and $\$y$
%	Modulus	$\$x \% \$y$	Remainder of $\$x$ divided by $\$y$
**	Exponentiation	$\$x ** \$y$	Result of raising $\$x$ to the $\$y$ 'th power

# Contoh Operator Aritmatika



```
<?php
```

```
    $x = 10;
```

```
    $y = 6;
```

```
    echo $x + $y;
```

```
?>
```

# Operator Penugasan

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

# Contoh Operator Penugasan



```
<?php  
    $x = 10;  
    echo $x;  
?>
```

# Operator Perbandingan

Operator	Name	Example	Result
==	Equal	\$x == \$y	Returns true if \$x is equal to \$y
===	Identical	\$x === \$y	Returns true if \$x is equal to \$y, and they are of the same type
!=	Not equal	\$x != \$y	Returns true if \$x is not equal to \$y
<>	Not equal	\$x <> \$y	Returns true if \$x is not equal to \$y
!==	Not identical	\$x !== \$y	Returns true if \$x is not equal to \$y, or they are not of the same type
>	Greater than	\$x > \$y	Returns true if \$x is greater than \$y
<	Less than	\$x < \$y	Returns true if \$x is less than \$y
>=	Greater than or equal to	\$x >= \$y	Returns true if \$x is greater than or equal to \$y
<=	Less than or equal to	\$x <= \$y	Returns true if \$x is less than or equal to \$y
<=>	Spaceship	\$x <=> \$y	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.



# Contoh Operator Perbandingan



```
<?php
```

```
    $x = 100;
```

```
    $y = "100";
```

```
    var_dump($x == $y); // returns true because values are equal  
?>
```

# Operator Increment / Decrement

Operator	Name	Description
<code>++\$x</code>	Pre-increment	Increments <code>\$x</code> by one, then returns <code>\$x</code>
<code>\$x++</code>	Post-increment	Returns <code>\$x</code> , then increments <code>\$x</code> by one
<code>--\$x</code>	Pre-decrement	Decrements <code>\$x</code> by one, then returns <code>\$x</code>
<code>\$x--</code>	Post-decrement	Returns <code>\$x</code> , then decrements <code>\$x</code> by one

# Contoh Operator Increment / Decrement



```
<?php  
    $x = 10;  
    echo ++$x;  
?>
```

# Operator Logika

Operator	Name	Example	Result
and	And	\$x and \$y	True if both \$x and \$y are true
or	Or	\$x or \$y	True if either \$x or \$y is true
xor	Xor	\$x xor \$y	True if either \$x or \$y is true, but not both
&&	And	\$x && \$y	True if both \$x and \$y are true
	Or	\$x    \$y	True if either \$x or \$y is true
!	Not	!\$x	True if \$x is not true

# Contoh Operator Logika



```
<?php
```

```
    $x = 100;
```

```
    $y = 50;
```

```
    if ($x = 100 and $y = 50) {
```

```
        echo "Hello world!";
```

```
    }
```

```
?>
```

# Operator String

Operator	Name	Example	Result
.	Concatenation	\$txt1 . \$txt2	Concatenation of \$txt1 and \$txt2
.=	Concatenation assignment	\$txt1 .= \$txt2	Appends \$txt2 to \$txt1

# Contoh Operator String



```
<?php
$txt1 = "Hello";
$txt2 = " world!";
echo $txt1 . $txt2;
?>
```

# Operator Larik/Array

Operator	Name	Example	Result
+	Union	<code>\$x + \$y</code>	Union of <code>\$x</code> and <code>\$y</code>
==	Equality	<code>\$x == \$y</code>	Returns true if <code>\$x</code> and <code>\$y</code> have the same key/value pairs
===	Identity	<code>\$x === \$y</code>	Returns true if <code>\$x</code> and <code>\$y</code> have the same key/value pairs in the same order and of the same types
!=	Inequality	<code>\$x != \$y</code>	Returns true if <code>\$x</code> is not equal to <code>\$y</code>
<>	Inequality	<code>\$x &lt;&gt; \$y</code>	Returns true if <code>\$x</code> is not equal to <code>\$y</code>
!==	Non-identity	<code>\$x !== \$y</code>	Returns true if <code>\$x</code> is not identical to <code>\$y</code>



# Contoh Operator Larik/Array



```
<?php
    $x = array("a" => "red", "b" => "green");
    $y = array("c" => "blue", "d" => "yellow");

    print_r($x + $y); // union of $x and $y
?>
```

# Operator Penugasan Bersyarat

Operator	Name	Example	Result
<code>?:</code>	Ternary	<code>\$x = <i>expr1</i> ? <i>expr2</i> : <i>expr3</i></code>	Returns the value of <code>\$x</code> . The value of <code>\$x</code> is <code><i>expr2</i></code> if <code><i>expr1</i> = TRUE</code> . The value of <code>\$x</code> is <code><i>expr3</i></code> if <code><i>expr1</i> = FALSE</code>
<code>??</code>	Null coalescing	<code>\$x = <i>expr1</i> ?? <i>expr2</i></code>	Returns the value of <code>\$x</code> . The value of <code>\$x</code> is <code><i>expr1</i></code> if <code><i>expr1</i></code> exists, and is not NULL. If <code><i>expr1</i></code> does not exist, or is NULL, the value of <code>\$x</code> is <code><i>expr2</i></code> . Introduced in PHP 7

# Contoh Operator Penugasan Bersyarat

```
<?php
    // if empty($user) = TRUE, set $status = "anonymous"
    echo $status = (empty($user)) ? "anonymous" : "logged in";
    echo("<br>");

    $user = "John Doe";
    // if empty($user) = FALSE, set $status = "logged in"
    echo $status = (empty($user)) ? "anonymous" : "logged in";
?>
```

# Contoh Operator Penugasan Bersyarat



```
<?php
    // if empty($user) = TRUE, set $status = "anonymous"
    echo $status = (empty($user)) ? "anonymous" : "logged in";
    echo("<br>");

    $user = "John Doe";
    // if empty($user) = FALSE, set $status = "logged in"
    echo $status = (empty($user)) ? "anonymous" : "logged in";
?>
```

# Conditional Statements

Pernyataan bersyarat digunakan untuk melakukan tindakan yang berbeda berdasarkan kondisi yang berbeda.

Dalam PHP, kita memiliki pernyataan kondisional berikut ini:

- if - mengeksekusi beberapa kode jika satu kondisi bernilai benar
- if...else - mengeksekusi beberapa kode jika suatu kondisi benar dan kode lain jika kondisi tersebut salah
- if...elseif...else - mengeksekusi kode yang berbeda untuk lebih dari dua kondisi
- switch - memilih salah satu dari banyak blok kode yang akan dieksekusi

# Contoh if



```
<?php
```

```
    $t = date("H");
```

```
    if ($t < "20") {
```

```
        echo "Have a good day!";
```

```
    }
```

```
?>
```

## Contoh if...else

```

<?php
    $t = date("H");

    if ($t < "20") {
        echo "Have a good day!";
    } else {
        echo "Have a good night!";
    }
?>
```

# Contoh if...elseif...else

```
<?php
    $t = date("H");

    if ($t < "10") {
        echo "Have a good morning!";
    } elseif ($t < "20") {
        echo "Have a good day!";
    } else {
        echo "Have a good night!";
    }
?>
```



# Contoh switch

```
<?php
    $favcolor = "red";

    switch ($favcolor) {
        case "red":
            echo "Your favorite color is red!";
            break;
        case "blue":
            echo "Your favorite color is blue!";
            break;
        case "green":
            echo "Your favorite color is green!";
            break;
        default:
            echo "Your favorite color is neither red, blue, nor green!";
    }
?>
```

# Perulangan

Perulangan digunakan untuk mengeksekusi blok kode yang sama berulang kali, selama kondisi tertentu benar.

Dalam PHP, kita memiliki beberapa jenis perulangan berikut ini:

- while - mengulang blok kode selama kondisi yang ditentukan benar
- do...while - mengulang sebuah blok kode sekali, lalu mengulang perulangan tersebut selama kondisi yang ditentukan benar
- for - mengulang melalui blok kode beberapa kali
- foreach - mengulang blok kode untuk setiap elemen dalam larik

# Contoh while



```
<?php
    $x = 1;

    while($x ≤ 5) {
        echo "The number is: $x <br>";
        $x++;
    }
?>
```

# Contoh do...while

```

<?php
    $x = 1;

    do {
        echo "The number is: $x <br>";
        $x++;
    } while ($x ≤ 5);
?>
```

# Contoh for



```
<?php
    for ($x = 0; $x ≤ 10; $x++) {
        echo "The number is: $x <br>";
    }
?>
```

## Contoh foreach



```
<?php
    $colors = array("red", "green", "blue", "yellow");

    foreach ($colors as $value) {
        echo "$value <br>";
    }
?>
```

# Referensi

- <https://www.php.net/manual/en/>
- <https://www.w3schools.com/php/DEFAULT.asp>