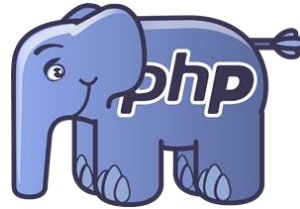


Pemrograman PHP



Pesantren PeTIK II YBM PLN

Jl. KH. Bisri Syansuri RT/01 RW/05, Plosogeneng,
Kec. Jombang, Kabupaten Jombang, Jawa Timur



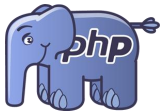
Pertemuan Ke-5





Materi

1. Pengantar Pemrograman PHP
2. Variabel dan Tipe Data
3. Array dan Struktur Data
4. PHP Form Processing
5. PHP Control Structure
6. PHP Function
7. Implementasi Template Web
9. PHP Object Oriented Programming
10. PHP Object Oriented Programming 2
11. PHP Database Connection
12. CRUD Database Operation 1
13. CRUD Database Operation 2
14. PHP Operasi file
15. Manajemen Session User



5. PHP Control Structure





PHP Control Structures

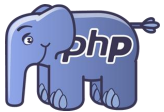
- `if (expression) {block} elseif {block} else {block}`
- `while (expression) {block} OR`
- `do {block} while (expression)`
- `for (expression) {block}`
- `foreach var (list) {block}`
- `switch (variable) //variable can be int or string`
 - `case 0:`
 - `// do something;`
 - `break;`
 - `case 1:`
 - `// do something else;`
 - `break;`
- `break;`
- `continue;`
- `return;`





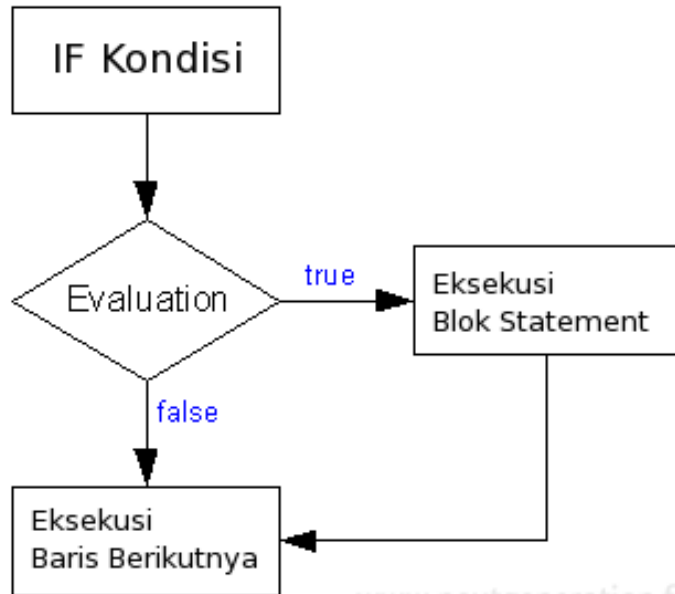
Control Structure : IF

If merupakan struktur kontrol percabangan yang hanya memiliki satu kondisi. Pernyataan if juga merupakan percabangan bersyarat yang berfungsi untuk melewati suatu proses. Jika pernyataan benar (terpenuhi) maka perintah yang kita buat akan dijalankan dan dilanjutkan ke proses yang selanjutnya, sedangkan jika pernyataan salah (tidak terpenuhi) maka perintah yang kita buat akan diabaikan.





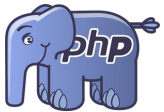
Control Structure : IF



www.nextgeneration.fr

```
<?php
    if(condition)
    {
        // Blok Statement
    }
?>

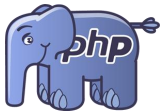
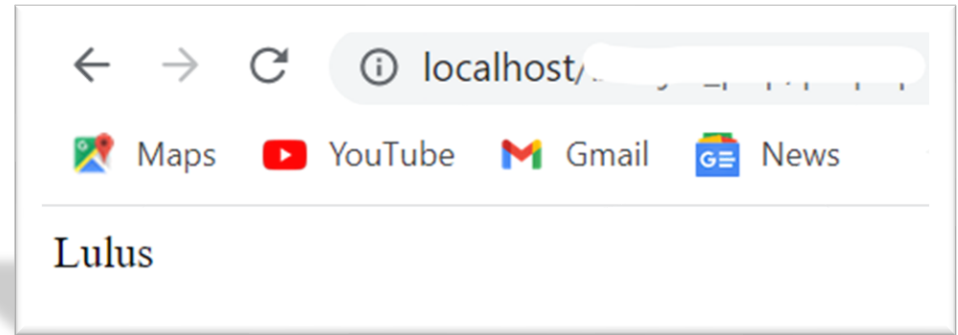
<?php
$
    if (!empty($nama))
    {
        echo 'Nama ' . $nama;
    }
?>
```





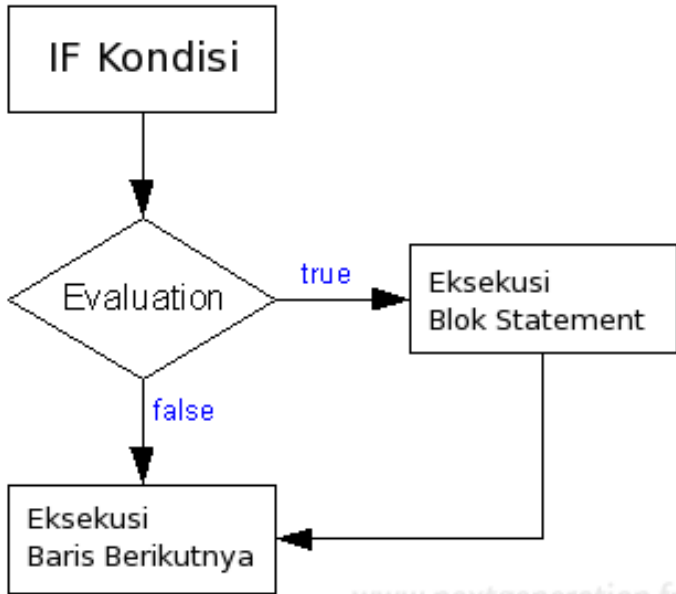
Control Structure : IF

```
<?php
    $nilai = 70;
    if($nilai > 60){
        echo 'Lulus';
    } else{
        echo 'mengulang';
    }
?>
```





Control Structure : IF



www.nextgeneration.fr

```
<?php
    $secretNumber = 453;
    if ($_POST['guess'] == $secretNumber) {
        echo "<p>Congratulations!</p>";
    }
?>
```

```
<?php
    $secretNumber = 453;
    if ($_POST['guess'] == $secretNumber) echo"<p>Congratulations!</p>";
?>
```

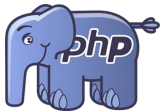
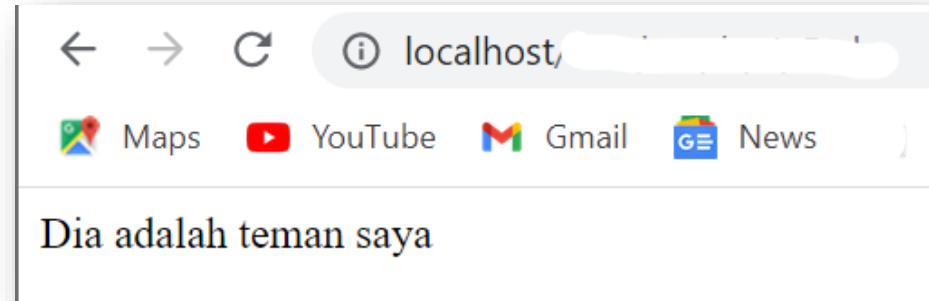




Control Structure : Else If

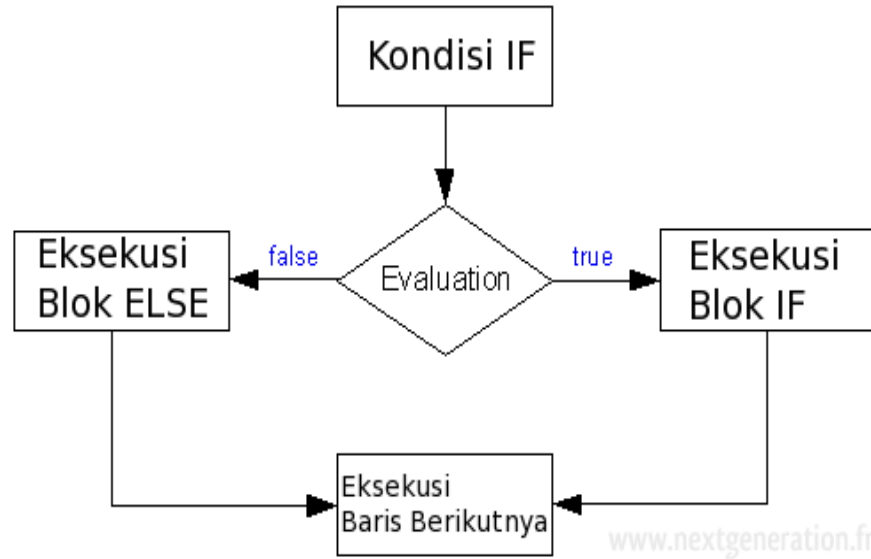
Kondisi else if ini berfungsi untuk memeriksa suatu nilai untuk melakukan suatu aksi. Berikut merupakan struktur penggunaan else if:

```
<?php
// Struktur Else If
$teman = "Daus";
if($teman == "Daus"){
    echo "Dia adalah teman saya";
} else{
    echo "Dia bukan teman saya";
}
?>
```



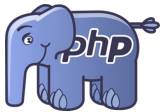


Control Structure : IF .. ELSE



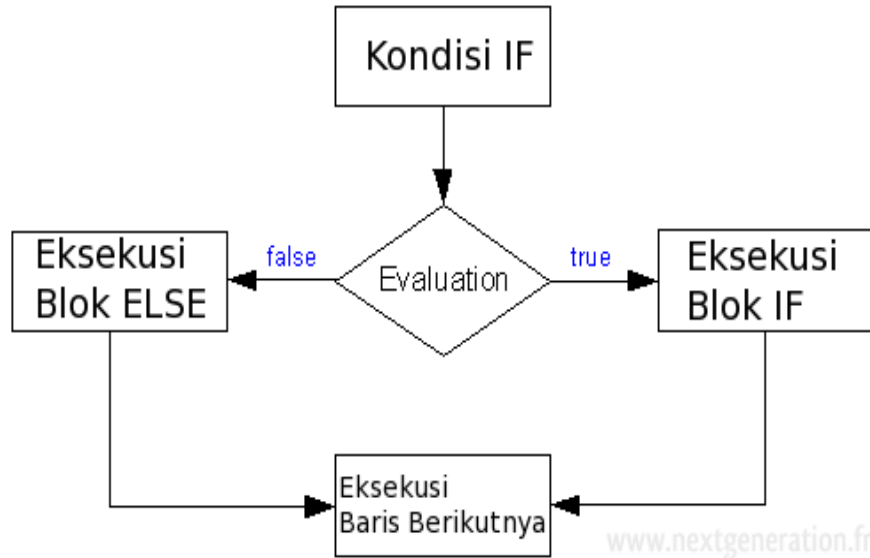
```
if (condition) {  
    code to be executed if condition is true;  
} else {  
    code to be executed if condition is false;  
}
```

```
<?php  
    if ( $_nilai > 55 ) {  
        echo 'LULUS';  
    } else {  
        echo 'TIDAK LULUS';  
    }  
?>
```



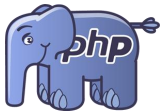


Control Structure : IF .. ELSE



www.nextgeneration.fr

```
<?php
    $secretNumber = 453;
    if ($_POST['guess'] == $secretNumber) {
        echo "<p>Congratulations!!</p>";
    } else {
        echo "<p>Sorry!</p>";
    }
?>
```





Control Structure : If Else If

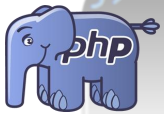
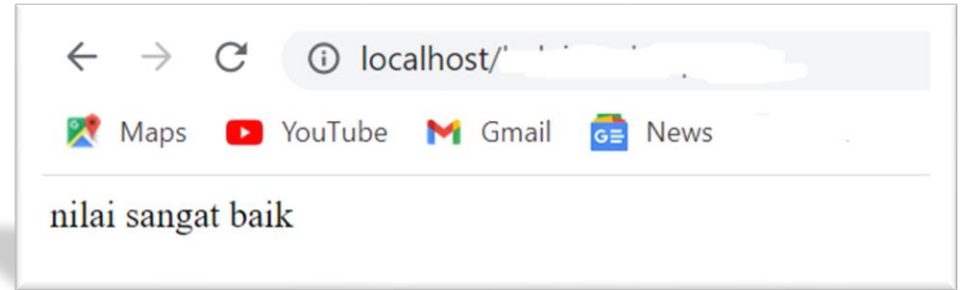
Else if ini berbeda hal nya dengan if, dimana if hanya memiliki satu kondisi, sedangkan if else if ini memiliki kondisi yang lebih dari 1 kondisi dan dari sekian banyaknya kondisi hanya akan dipilih salah satu yang benar (memenuhi).





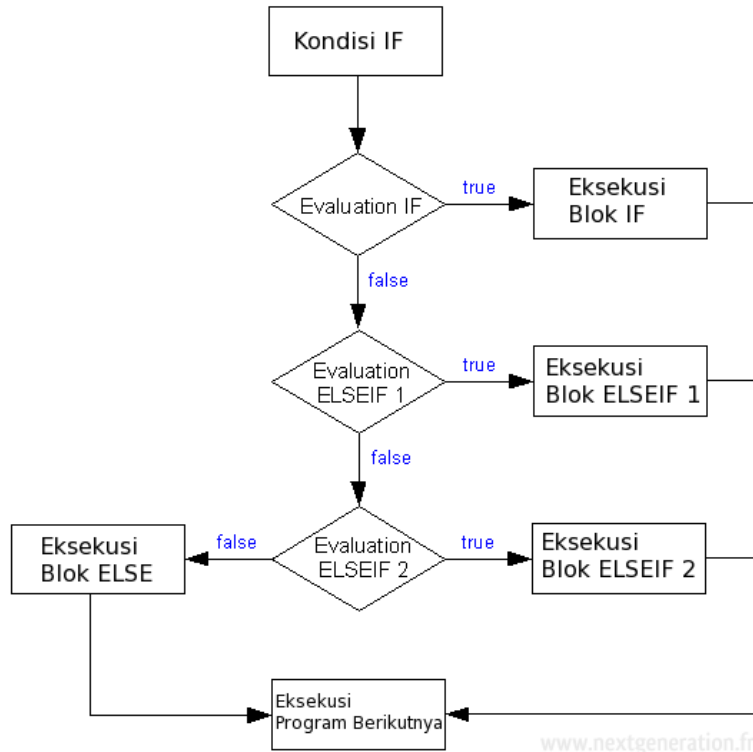
Control Structure : IF Else If

```
<?php
$nilai = 90;
if($nilai <= 60) {
    echo 'nilai cukup';
} elseif ($nilai <= 75) {
    echo 'nilai baik';
} elseif ($nilai >= 85) {
    echo 'nilai sangat baik';
} else {
    echo 'nilai kurang';
}
```





Control Structure : IF .. ELSE



www.nextgeneration.fr

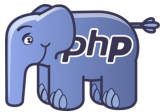
```

if (condition) {
    code to be executed if this condition is true;
} elseif (condition) {
    code to be executed if this condition is true;
} else {
    code to be executed if all conditions are false;
}
    
```

```

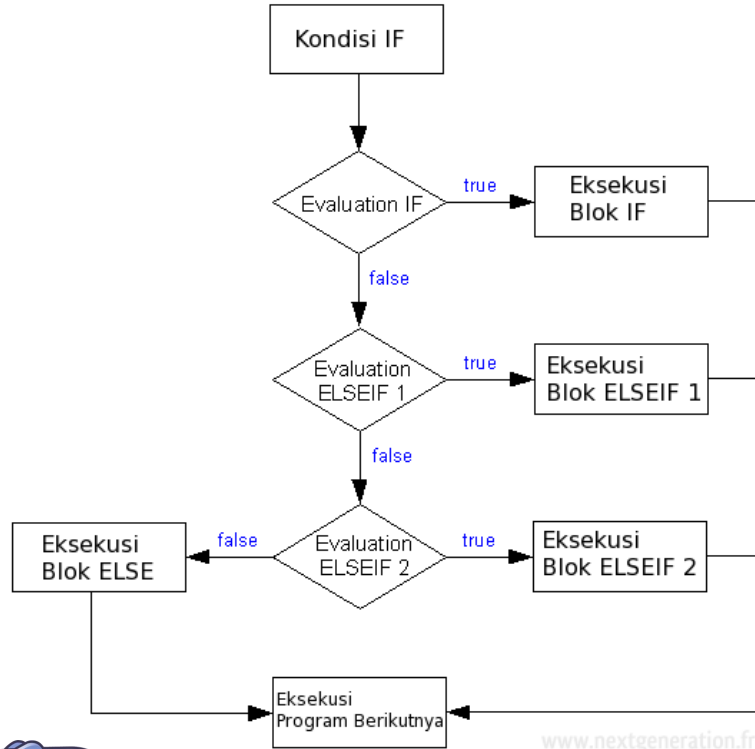
<?php
    if ($_nilai <= 35 ){
        $_grade='E';
    }elseif ($_nilai <= 55 ){
        $_grade='D';
    }
    ...
    else {
        $_grade='I';
    } ...

?>
    
```





Control Structure : IF .. ELSE



```
<?php
```

```
$secretNumber = 453;
```

```
$_POST['guess'] = 442;
```

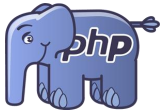
```
if ($_POST['guess'] == $secretNumber) {  
    echo "<p>Congratulations!</p>";
```

```
} elseif (abs ($_POST['guess'] - $secretNumber) < 10) {  
    echo "<p>You're getting close!</p>";
```

```
} else {  
    echo "<p>Sorry!</p>";
```

```
}
```

```
?>
```





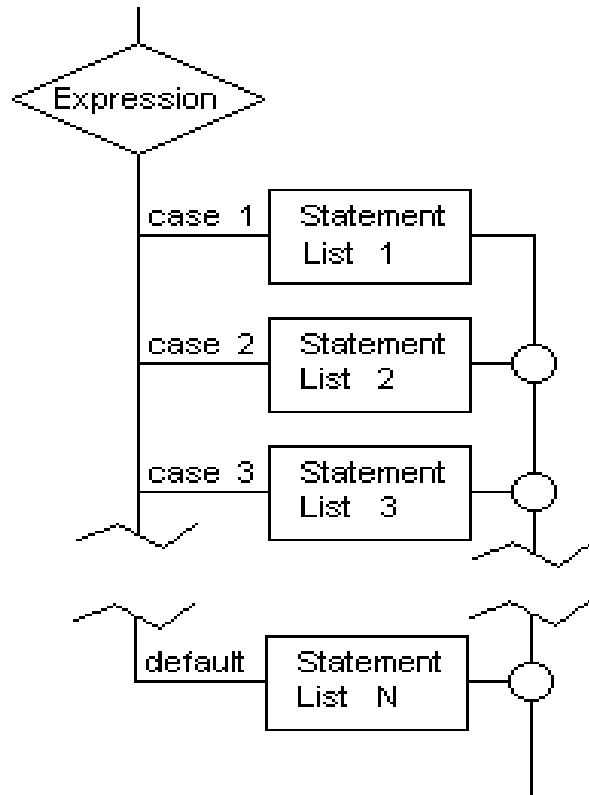
Control Structure : Switch

Pernyataan switch agak mirip dengan pernyataan else if, dimana pernyataan else if memiliki kondisi yang berbeda dan tindakan yang sesuai akan dijalankan berdasarkan suatu kondisi. Disisi lain, jika kita ingin membandingkan variabel dengan nilai yang beda, maka kita bisa menggunakan switch.





Control Structure : Switch



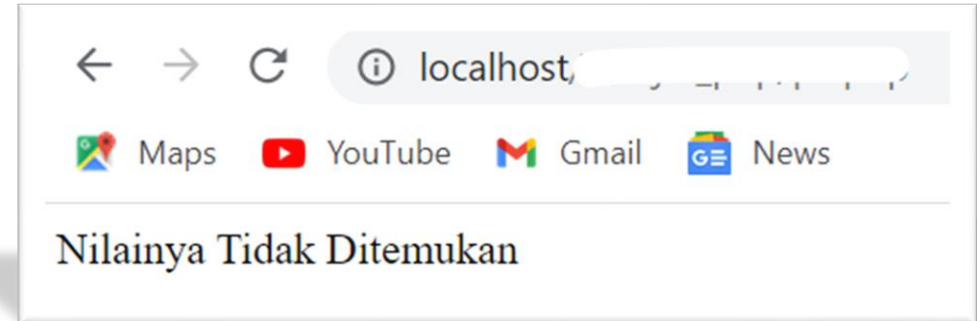
```
switch (n) {  
    case label1:  
        code to be executed if n=label1;  
        break;  
    case label2:  
        code to be executed if n=label2;  
        break;  
    case label3:  
        code to be executed if n=label3;  
        break;  
    ...  
    default:  
        code to be executed if n is different from all labels;  
}
```





Control Structure : Switch

```
<?php
$nilai = 85;
switch ($nilai) {
    case 50:
        echo "Nilainya Adalah 50";
        break;
    case 70:
        echo "Nilainya Adalah 70";
        break;
    case 90:
        echo "Nilainya Adalah 90";
        break;
    case 100:
        echo "Nilainya Adalah 100";
        break;
    default:
        echo "Nilainya Tidak Ditemukan";
        break;
}
```

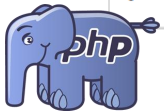




Control Structure : Switch

```
switch (n) {  
    case label1:  
        code to be executed if n=label1;  
        break;  
    case label2:  
        code to be executed if n=label2;  
        break;  
    case label3:  
        code to be executed if n=label3;  
        break;  
    ...  
    default:  
        code to be executed if n is different from all labels;  
}
```

```
switch ( $_grade )  
{  
    case 'E' :  
        $_predikat = 'SANGAT  
        KURANG';  
        break;  
    case 'D' :  
        $_predikat = 'KURANG';  
        break;  
    --  
    default :  
        $_predikat = 'TIDAK ADA';  
}
```

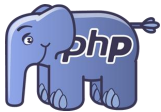




Control Structure : Switch

```
switch (n) {  
    case label1:  
        code to be executed if n=label1;  
        break;  
    case label2:  
        code to be executed if n=label2;  
        break;  
    case label3:  
        code to be executed if n=label3;  
        break;  
    ...  
    default:  
        code to be executed if n is different from all labels;  
}
```

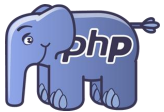
```
<?php  
switch($category) {  
    case "news":  
        print "<p>What's happening around the World</p>";  
        break;  
    case "weather":  
        print "<p>Your weekly forecast</p>";  
        break;  
    case "sports":  
        print "<p>Latest sports highlights</p>";  
        break;  
    default:  
        print "<p>Welcome to my Web site</p>";  
}  
?>
```





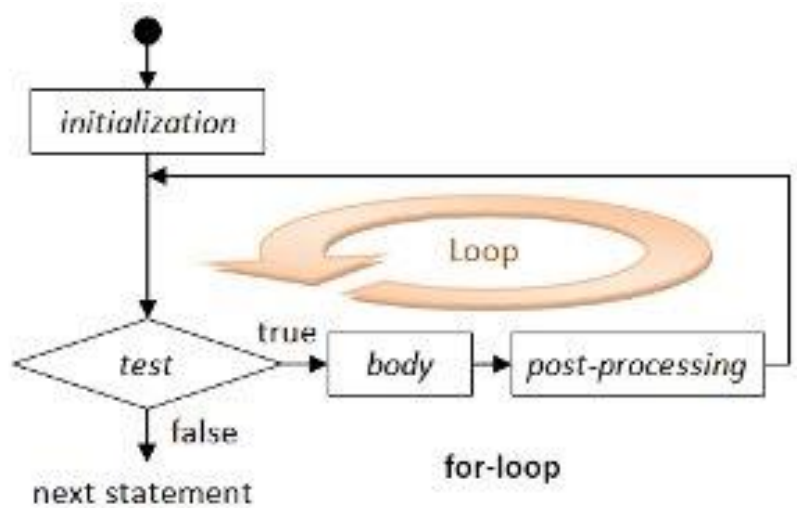
Looping For

for loop digunakan untuk mengeksekusi sepotong kode beberapa kali. Dengan kata lain, jika Anda sudah mengetahui berapa kali Anda ingin mengeksekusi blok kode, perulangan for adalah pilihan terbaik.



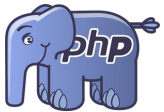


Looping For



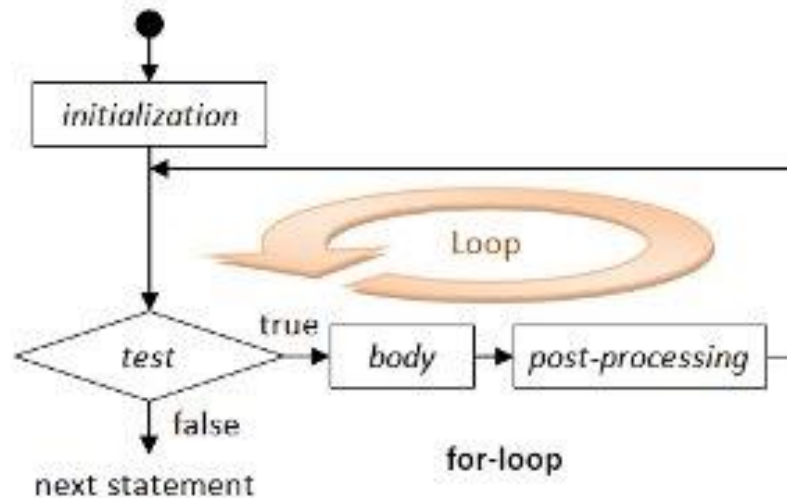
```
for (expr1 ; expr2 ; expr3)
{
    blok statement
}
```

```
<select name="tgl">
<?php
for ( $x = 1 ; $x <= 30 ; $x++)
{
    echo '<option value=" '.$x.' ">'.$x.
    '</option>';
}
?>
</select>
```





Looping For



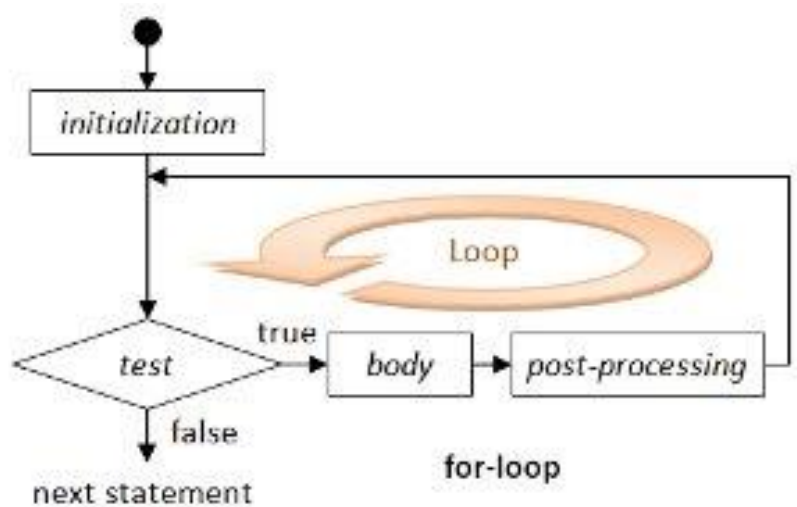
- Expression 1 dievaluasi pada saat iterasi pertama
- Expression 2 dievaluasi pada saat awal disetiap iterasi, bagian ini menentukan apakah akan dilanjutkan iterasi berikutnya
- Expression 3 akan di eksekusi hasil dari logika untuk setiap loop

```
for (expression1; expression2; expression3) {  
    statements  
}
```



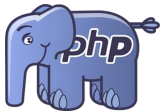


Looping For



```
for (expr1 ; expr2 ; expr3)
{
    blok statement
}
```

```
<select name="tgl">
<?php
for ( $x = 1 ; $x <= 30 ; $x++ )
{
    echo '<option value=" ' . $x . '">' . $x .
    '</option>';
}
?>
</select>
```





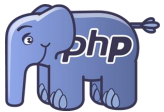
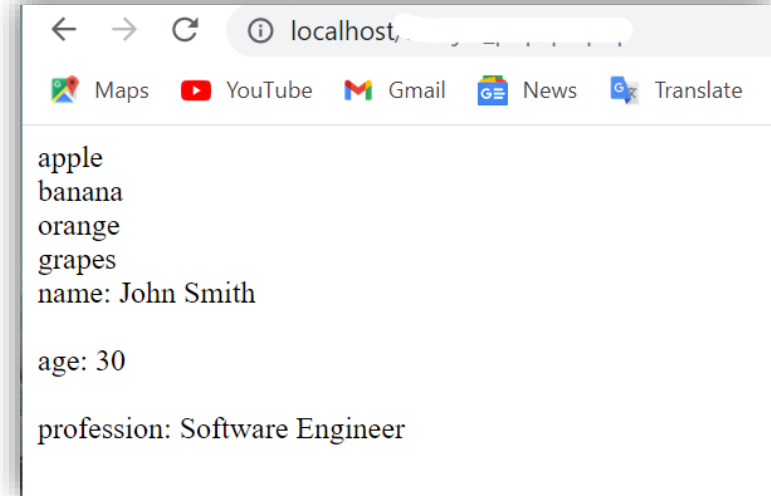
Looping Foreach

Loop foreach digunakan untuk mengulangi variabel array. Jika Anda memiliki variabel array, dan Anda ingin menelusuri setiap elemen array itu, loop foreach adalah pilihan terbaik.



Looping Foreach

```
<?php
// foreach
$fruits = ["apple","banana","orange","grapes"];
foreach($fruits as $fruit)
{
    echo $fruit;
    echo "<br/>";
}
$employee = [
    "name" => "John Smith",
    "age" => 30,
    "profession" => "Software Engineer"
];
foreach($employee as $key => $value)
{
    echo sprintf("%s: %s<br>", $key, $value);
    echo "<br/>";
}
?>
```





Looping Foreach : Array

- Looping khusus untuk Array

```
foreach (array_expr as $value) {  
    statement  
}
```

- Looping Array Asosiatif

```
foreach (array_expr as $key => $value) {  
    statement  
}
```



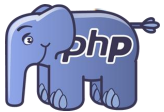


Looping Foreach : Array

- Looping khusus untuk Array

```
foreach (array_expr as $value) {  
    statement  
}
```

```
<?php  
$links = array("www.apress.com", "www.php.net", "www.apache.org");  
echo "<b>Online Resources</b>:<br />";  
foreach($links as $link) {  
    echo "<a href=\"http://$link\">$link</a><br />";  
}
```





Looping Foreach : Array Asosiatif

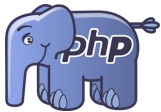
```
<?php
$ar_hari = [1=>'Senin','Selasa','Rabu','Kamis','Jumat','Sabtu','Minggu' ] ;
?>
<select name="hari">
<?php
foreach ( $ar_hari as $k => $v)
{
    echo '<option value=" '.$k.' ">'.$v.' </option>';
}
?>
</select>
```





Looping While

While merupakan struktur perulangan yang berfungsi melakukan tugas berulang selama banyaknya pernyataan yang bernilai benar (terpenuhi). Loop while digunakan ketika Anda ingin mengeksekusi sepotong kode berulang kali hingga kondisi while bernilai false.





Looping While

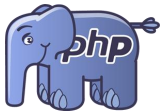
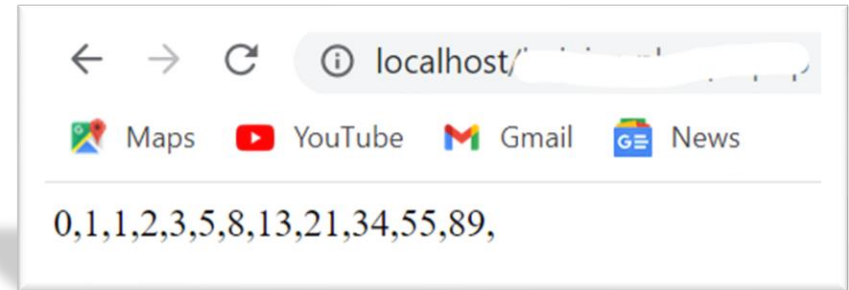
```
<?php
$max = 0;
echo $i = 0;
echo ",";
echo $j = 1;
echo ",";
$result = 0;

while($max < 10)
{
    $result = $i + $j;

    $i = $j;
    $j = $result;

    $max = $max + 1;
    echo $result;
    echo ",";
}

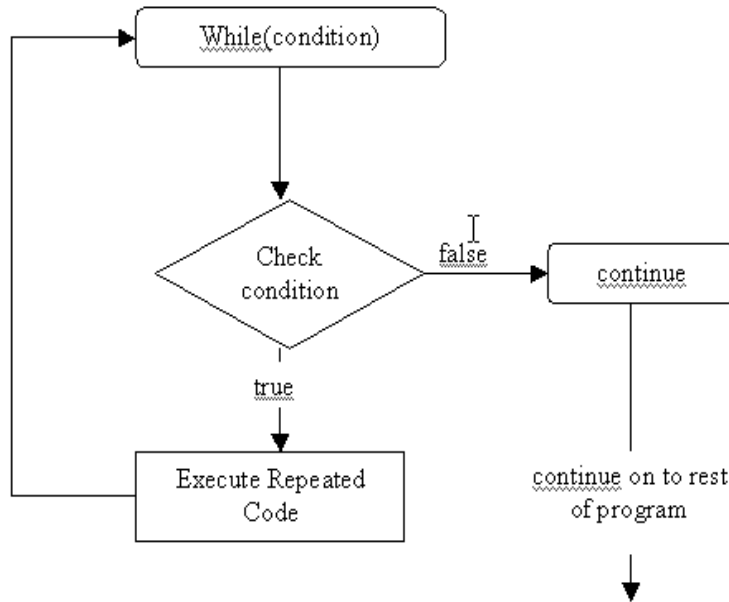
?>
```





Looping While

Flow Diagram of a while loop

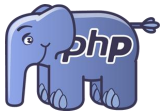


// Contoh : cetak bilangan 1 s/d 12, 14 s/d 17

```
$x = 0;
while ( $x <= 20)
{
    $x++;
    if ( $x == 13 ) continue;
    echo '<br>Bilangan : '.$x;
    if ( $x == 17 ) break;
}
echo '<br>Selesai';
```

Keyword break : digunakan untuk keluar dari looping

Keyword continue : digunakan untuk melanjutkan looping berikutnya dengan mengabaikan baris dibawahnya





Looping Do-While

Perulangan do-while sangat mirip dengan perulangan while, dengan satu-satunya perbedaan adalah kondisi while diperiksa pada akhir iterasi pertama. Dengan demikian, kami dapat menjamin bahwa kode loop dieksekusi setidaknya sekali, terlepas dari hasil ekspresi while.





Looping Do-While

```
<?php
    $x = 1;

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

?>
```

← → ↻ ⓘ localhost

 Maps  YouTube  Gmail  News

The number is: 1
The number is: 2
The number is: 3
The number is: 4
The number is: 5

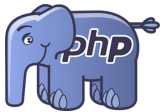




Looping Break

Ada kalanya Anda mungkin ingin keluar dari lingkaran sebelum berjalan dengan sendirinya. Ini dapat dicapai dengan mudah menggunakan kata kunci break. Ini akan membuat Anda keluar dari struktur for, foreach, while, do-while, atau switch saat ini.

Anda juga dapat menggunakan break untuk keluar dari beberapa loop bersarang dengan memberikan argumen numerik. Misalnya, menggunakan break 3 akan mengeluarkan Anda dari 3 loop bersarang.





Looping - Break

- Menghentikan secara paksa saat looping

```
<?php
```

```
$primes = array(2,3,5,7,11,13,17,19,23,29,31,37,41,43,47);
```

```
for($count = 1; $count++; $count < 1000) {
```

```
    $randomNumber = rand(1,50);
```

```
    if (in_array($randomNumber,$primes)) {
```

```
        break;
```

```
    } else {
```

```
        echo "<p>Non-prime number encountered: $randomNumber</p>";
```

```
    }
```

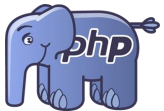
```
}
```

```
?>
```

Non-prime number encountered: 48

Non-prime number encountered: 42

Prime number encountered: 17

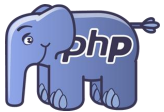
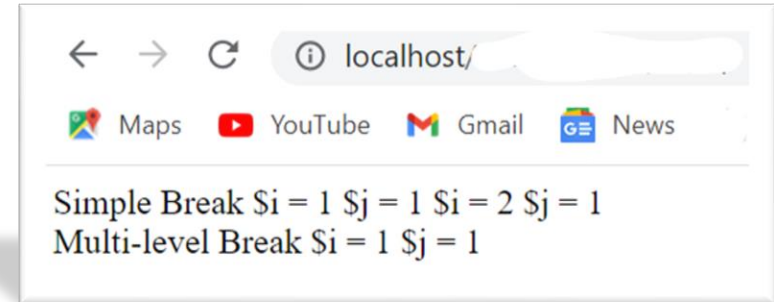


Looping Break

```
<?php
echo 'Simple Break';
for($i = 1; $i <= 2; $i++) {
    echo "\n".'$i = '.$i.' ';
    for($j = 1; $j <= 5; $j++) {
        if($j == 2) {
            break;
        }
        echo '$j = '.$j.' ';
    }
}

echo "<br/>";

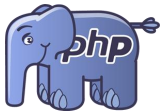
echo 'Multi-level Break';
for($i = 1; $i <= 2; $i++) {
    echo "\n".'$i = '.$i.' ';
    for($j = 1; $j <= 5; $j++) {
        if($j == 2) {
            break 2;
        }
        echo '$j = '.$j.' ';
    }
}
?>
```





Looping Continue

Kata kunci lain yang dapat menginterupsi loop di PHP adalah `continue`. Namun, ini hanya melewatkan sisa iterasi loop saat ini alih-alih keluar dari loop sama sekali. Sama seperti `break`, Anda juga dapat menggunakan nilai numerik dengan terus menentukan berapa banyak loop bersarang yang harus dilewati untuk iterasi saat ini.





Looping - Continue

- Lanjutkan ke looping berikutnya dengan mengabaikan baris dibawahnya (body looping tidak dieksekusi)

```
<?php
```

```
$usernames = array("grace","doris","gary","nate","missing","tom");
```

```
for ($x=0; $x < count($usernames); $x++) {
```

```
    if ($usernames[$x] == "missing") continue;
```

```
    echo "Staff member: $usernames[$x] <br />";
```

```
}
```

```
?>
```

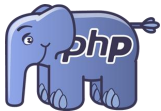
Staff member: grace

Staff member: doris

Staff member: gary

Staff member: nate

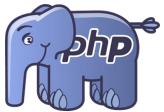
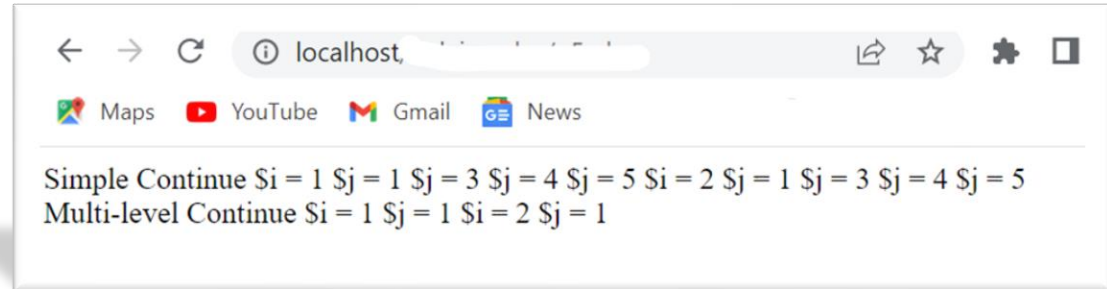
Staff member: tom



Looping Break

```
<?php
echo 'Simple Continue';
for($i = 1; $i <= 2; $i++) {
    echo "\n".'$i = '.$i.' ';
    for($j = 1; $j <= 5; $j++) {
        if($j == 2) {
            continue;
        }
        echo '$j = '.$j.' ';
    }
}

echo "<br/>";
echo 'Multi-level Continue';
for($i = 1; $i <= 2; $i++) {
    echo "\n".'$i = '.$i.' ';
    for($j = 1; $j <= 5; $j++) {
        if($j == 2) {
            continue 2;
        }
        echo '$j = '.$j.' ';
    }
}
?>
```





Form Validation : Control Logic IF

- Memastikan data yang di isi melalui Form adalah valid !
- Tidak ada data yang kosong/spasi kosong (required field)
- Data sesuai dengan tipe data dan format yang diinginkan
 - Contoh : number – uang , date – tanggal, string - kode





Contoh: PHP Form Validation

PHP Form Validation Example

* required field

Name: *

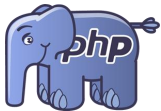
E-mail: *

Website:

Comment:

Gender: ☐ Female ☐ Male ☐ Other *

Submit



```
<?php
// define variables and set to empty values
$name = $email = $gender = $comment = $website = "";

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $name = test_input($_POST["name"]);
    $email = test_input($_POST["email"]);
    $website = test_input($_POST["website"]);
    $comment = test_input($_POST["comment"]);
    $gender = test_input($_POST["gender"]);
}

function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
?>
```

https://www.w3schools.com/php/php_form_validation.asp



Contoh: PHP Form Required

```
<?php
// define variables and set to empty values
$nameErr = $emailErr = $genderErr = $websiteErr = "";
$name = $email = $gender = $comment = $website = "";

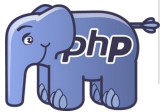
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    if (empty($_POST["name"])) {
        $nameErr = "Name is required";
    } else {
        $name = test_input($_POST["name"]);
    }

    if (empty($_POST["email"])) {
        $emailErr = "Email is required";
    } else {
        $email = test_input($_POST["email"]);
    }

    if (empty($_POST["website"])) {
        $website = "";
    } else {
        $website = test_input($_POST["website"]);
    }
}
```

```
if (empty($_POST["comment"])) {
    $comment = "";
} else {
    $comment = test_input($_POST["comment"]);
}

if (empty($_POST["gender"])) {
    $genderErr = "Gender is required";
} else {
    $gender = test_input($_POST["gender"]);
}
?>
```



**TERIMA KASIH
ATAS SEGALA PERHATIAN
SEMOGA BERMANFAAT...**

