

LOOPING FOR

Pengertian Struktur Perulangan For (Looping For)

instruksi program yang bertujuan untuk mengulang beberapa baris perintah. Di dalam bahasa pemrograman, terdapat beberapa jenis instruksi perulangan, salah satunya perulangan for.

Loop for pada PHP digunakan untuk melakukan perulangan sejumlah nilai yang telah ditentukan.

Ketika merancang perulangan, setidaknya kita harus mengetahui 3 komponen: kondisi awal dari perulangan, perintah program yang akan diulang, serta kondisi akhir dimana perulangan akan berhenti.

Cara Penulisan Perulangan For

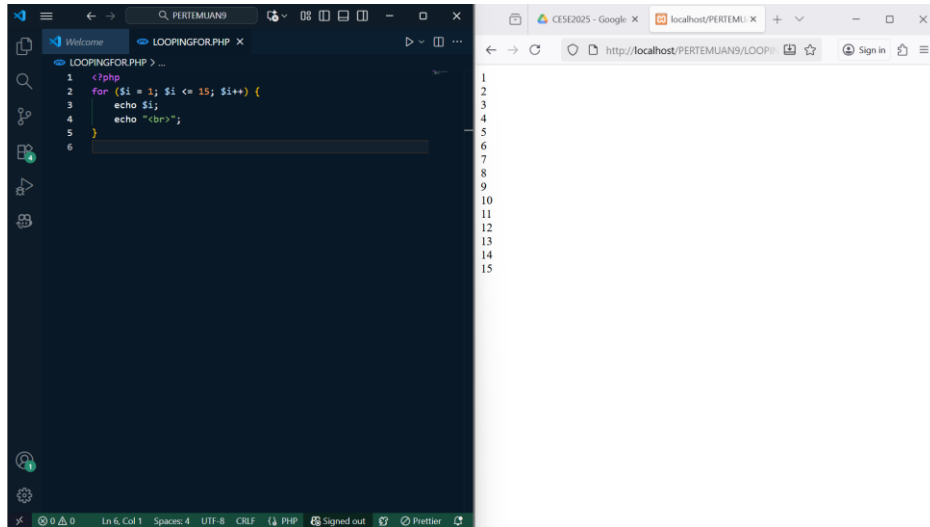
```
for (start; condition; increment) {  
    statement;  
}
```

- Start adalah kondisi awal perulangan. Biasanya kondisi ini dipakai untuk membuat dan memberikan nilai awal ke variabel counter. Variabel counter adalah sebutan untuk variabel yang dipakai untuk mengontrol perulangan. Misalkan kita akan membuat variabel counter \$i, maka untuk kondisi start harus memberi nilai awal seperti \$i = 1.
- Condition adalah kondisi yang harus dipenuhi agar perulangan bisa berjalan. Selama kondisi ini terpenuhi, maka PHP akan terus melakukan perulangan. Misalnya kita ingin menghentikan perulangan jika variabel \$i telah mencapai 20, maka pada bagian condition ini tulis perintah \$i<=20. Ini berarti berarti selama nilai \$i kurang atau sama dengan 20, terus lakukan perulangan.
- Increment adalah bagian yang dipakai untuk memproses variabel counter agar bisa memenuhi kondisi akhir perulangan.
- Statement adalah bagian kode program yang akan diproses secara terus-menerus selama proses perulangan berlangsung.

Sebagai contoh lain, berikut kode program untuk menampilkan angka 1-15 memakai perulangan for PHP:

```
<?php  
for ($i = 1; $i <= 15; $i++) {  
    echo $i;  
    echo "<br>";  
}  
?>
```

Hasil Kode Program



The screenshot shows a code editor on the left and a web browser on the right. The code editor displays a PHP file named 'LOOPINGFOR.PHP' with the following code:

```
1 <?php
2 for ($i = 1; $i <= 15; $i++) {
3     echo $i;
4     echo "<br>";
5 }
6
```

The web browser on the right shows the output of the script, displaying the numbers 1 through 15, each on a new line.

Kondisi awal dari perulangan adalah **$\$i = 1$** , dimana saya memberikan nilai 1 kepada variabel **$\$i$** . Variabel **$\i** inilah yang akan menjadi **counter** atau penghitung dari perulangan for.

Untuk kondisi akhir, ditulis sebagai **$\$i \leq 15$** . Jadi selama variabel **$\$i$** bernilai kurang atau sama dengan 15, perulangan akan terus dijalankan.

Sebagai **increment**, terdapat perintah **$\$i++$** . Perintah ini sama dengan **$\$i = \$i + 1$** , sehingga dalam setiap proses perulangan, **$\$i$** akan bertambah 1 angka.

Selain berfungsi sebagai **counter**, variabel **$\$i$** juga dapat diakses dalam proses perulangan seperti di echo pada baris 3.

Penjelasan Lengkap Syntax Dasar For (Loop For)

Contoh code

```
<?php
for ($i = 1; $i <= 5; $i++) {
    echo $i;
    echo "<br>";
}
?>
```

Penjelasan code

- **for = Kata kunci untuk memulai perulangan (looping).**

Kata kunci *for* digunakan untuk memulai proses perulangan dalam PHP. Struktur ini memungkinkan program menjalankan satu atau beberapa baris perintah secara berulang, sesuai dengan kondisi yang telah ditentukan.

- **\$i = 1 = Inisialisasi (nilai awal).**

Pada bagian ini dibuat variabel dengan nilai awal tertentu, misalnya \$i = 1. Bagian inisialisasi ini hanya dijalankan satu kali di awal perulangan dan berfungsi sebagai titik awal pengulangan yang akan digunakan sebagai penghitung atau *counter*.

- **\$i <= 5 = Kondisi (syarat perulangan).**

Kondisi ini menentukan kapan perulangan akan terus dijalankan atau berhenti. Selama nilai \$i kurang dari atau sama dengan 5, maka perulangan akan terus berjalan. Kondisi ini selalu dicek sebelum perulangan dimulai pada setiap siklusnya.

- **\$i++ = Increment (penambahan nilai).**

Increment berfungsi untuk menambah nilai variabel penghitung setiap kali satu kali perulangan selesai dilakukan. Penulisan \$i++ memiliki arti yang sama dengan \$i = \$i + 1, artinya nilai \$i akan bertambah satu pada setiap proses perulangan sehingga mencegah terjadinya perulangan tanpa akhir.

- **echo "Perulangan ke-\$i"; = Perintah yang dijalankan.**

Bagian ini berisi perintah yang akan dijalankan selama perulangan berlangsung. Misalnya, perintah echo "Perulangan ke-\$i
"; akan menampilkan teks “Perulangan ke-1” hingga “Perulangan ke-5” pada layar. Tag
 digunakan untuk membuat tampilan hasil berpindah ke baris baru.