

PERNYATAAN KONTROL PENGULANGAN

Adi Rachmanto, S.Kom – Prodi Akuntansi UNIKOM

Pernyataan Kontrol Pada Program PHP

PENGAMBILAN KEPUTUSAN

- Pernyataan IF
 - - If saja
 - - If – else
 - - if – elseif
- Pernyataan Switch

PENGULANGAN

- For
- While
- Do-While

FUNGSI PENGULANGAN (LOOP)

- 
- For
 - While
 - Do-While

- ✓ Struktur pengulangan (Loop) digunakan untuk mengulang sebagian dari suatu yang rutin (bagian dari suatu kode program tertentu) berdasarkan suatu kondisi tertentu.
- ✓ Dengan demikian kita tidak perlu menulis ulang rutin sebanyak pengulangan yang kita inginkan. Banyak pengulangan dapat ditentukan sesuai dengan kondisi yang diberikan.
- ✓ Dengan kata lain, pengulangan tersebut akan dilakukan selama kondisi tersebut tidak memenuhi syarat maka pengulangan akan dihentikan.

Pernyataan FOR

Bentuk Pernyataan For

```
for (ekspresi1;ekspresi2;ekspresi3){  
  Pernyataan_pernyataan;  
}
```

- **ekspresi1** → memberi nilai awal terhadap variable yang akan digunakan untuk melakukan pencacahan pengulangan.
- **ekspresi2** → menentukan pengulangan terhadap pernyataan yang ada dalam tanda 2 kurung kurawal ({...}) akan dilakukan atau tidak.
- **ekspresi3** → mengatur nilai variabel yang digunakan dalam ekspresi

Contoh Pengulangan “For” 1

```
<html>
<head>
    <title>Contoh Pengulangan 1</title>
</head>
<body>
    <?php
        for($i=1;$i<=100;$i++)
        {
            printf("%s <Br>", $i);
        }
    ?>
</body>
</html>
```

Simpan di drive E:Xampp/Htdocs/Folder Masing-masing/For1.php

Penjelasan Skrip For1.php

- Baris 1-5 : Awal dari HTML
- Baris 6 : Awal dari script php dimulai dengan tag `<?php`
- Baris 7- 8 : Perulangan dari 1 sampai 100 untuk menghasilkan deret angka 1 sampai 100.
- Baris 9 : Akhir dari script php diakhiri dengan tab `?>`
- Baris 10 - 11 : Akhir dari HTML

Pernyataan WHILE

□ Bentuk Pernyataan While :

```
while (ekspresi)  
{  
    Pernyataan_pernyataan  
}
```

- ❖ Pernyataan while akan memeriksa nilai ekspresi terlebih dahulu.
- ❖ Jika bernilai benar maka pernyataan-pernyataan yang terdapat dalam { } akan dijalankan dan kemudian ekspresi dievaluasi lagi.
- ❖ Proses ini diulang terus-menerus sampai ekspresi bernilai salah.

Contoh Pengulangan “While 1”

```
□ <html>
□ <head>
□   <title>Contoh Pengulangan While</title>
□ </head>
□ <body>
□ <center>
□ <?php
□   $bilangan = 1;
□   while ($bilangan <=10)
□   {
□     print ("Baris nomor $bilangan<br>");
□     $bilangan = $bilangan + 1;
□   }
□ ?>
```

```
</center>
</body>
</html>
```

```
Simpan di drive
E:Xampp/Htdocs/
Folder Masing-masing
/While1.php
```


Penjelasan While1.php

- Pada contoh skrip while1.php, isi variabel **bilangan** berperan dalam melakukan pengulangan perintah yang berada { }.
- Pada keadaan seperti ini harus dipastikan bahwa ada pernyataan yang mengubah nilai **bilangan** sehingga suatu ketika dalam **while** (yaitu pada contoh ini, \$bilangan <=10) bernilai salah. Jika tidak maka akan terjadi pengulangan selamanya.

Pernyataan DO - WHILE

□ Bentuk Pernyataan do-while

```
Do  
{  
    pernyataan_pernyataan  
} while (ekspresi);
```

- Pengulangan akan berakhir jika ekspresi (yang diuji sesudah pernyataan-pernyataan dijalankan) bernilai salah.
- Pernyataan yang berada dalam { }, paling tidak akan dijalankan dieksekusi sekali.

Contoh Pengulangan “do-while”

```
<html>
<head>
  <title>contoh Pengulangan do-while</title>
</head>
<body>
</body>
</html>
```

```
<?php
    $bilangan = 1;
    do
    {
        print (" $bilangan <BR> " );
        $bilangan++;
    }
    while ( $bilangan < 26 );
?>
```

Simpan di drive D:Xampp/Htdocs/Folder Masing-masing/dowhile1.php

Pernyataan “Break”

- <HTML>
- <HEAD>
- <TITLE>Contoh untuk Memperlihatkan Efek break</TITLE>
- </HEAD>
- <BODY>

```
<?php
for($i = 1; $i <= 25; $i++)
{
    print("$i <BR>");

    if ($i == 10)
        break;
}
print("Selesai <BR>");
?>
</BODY>
</HTML>
```

Simpan file di drive
D:\xampp\htdocs\folder
masing-
masing/Break1.php