

MODUL 3

Struktur Kontrol



CAPAIAN PEMBELAJARAN

Membuat script program PHP untuk pemecahan masalah algoritma pemrograman dengan bahasa PHP



KEBUTUHAN ALAT/BAHAN/SOFTWARE

1. XAMPP for Windows
2. VS Code editor
3. Web Browser: Mozilla Firefox atau Chrome
4. Sorce Code Hasil Praktikum Modul 2



DASAR TEORI

Keputusan/Kondisional

Statement kondisional digunakan pada saat Anda akan melakukan eksekusi blok statement tertentu. Misalnya Anda memiliki beberapa blok statement dan Anda hanya ingin mengeksekusi salah satunya. PHP dapat membantu menjalankan block yang anda inginkan dengan adanya pengecekan kondisional diawal block statementnya. Sintak PHP berkaitan dengan statement kondisional adalah statemen **if else elseif** dan statemen **switch**.

a) Bentuk if

Perintah IF ini akan menjalankan statement jika kondisinya bernilai benar (True).

```
if (expr)
    statement
```

b) Bentuk if else

Dalam bentuk ini jika ekspresi bernilai benar maka statement_1 akan dijalankan, tetapi jika salah maka statement_2 yang akan dijalankan.

```

if (expr)
    statement_1
else
    statement_2

```

atau

```

if (expr)
{
    statement_1a
    statement_1b
}
else
{
    statement_2a
    statement_2b
}

```

c) Bentuk if – elseif

Bentuk if-elseif ini cocok untuk melakukan pengambilan keputusan yang melibatkan banyak alternative. PHP dapat mengetahui bentuk elseif dan else if.

d) Switch

Pada prinsipnya penggunaan switch hampir sama dengan penggunaan kondisi if – elseif. Dalam penggunaan switch juga disertakan perintah break. Tanpa perintah break semua pernyataan akan dijalankan.

Bentuk pernyataan switch yaitu :

```

switch (ekspresi)
{
    case ekspresi_case_1:
        pernyataan_1;
        break;
    case ekspresi_case_2:
        pernyataan_2;
        break;
    case ekspresi_case_n :
        pernyataan_n;
        break;
    default:
        pernyataan_default;
}

```

Perulangan

Terkadang pada saat scripting terdapat statement yang perlu dieksekusi secara berulang agar dapat memperoleh output yang dibutuhkan. Yang perlu diperhatikan jika ingin melakukan pengulangan adalah kondisi apa pengulangan dimulai, kondisi apa pengulangan berhenti dan seperti apa pengulangan dilakukan. Beberapa sintak PHP berkaitan dengan pengulangan adalah while, for, do-while, foreach.

a) while

Perulangan menggunakan while mempunyai bentuk yang mudah untuk digunakan. Perintah while akan terus diulang selama kondisi berisi **TRUE** dan akan berhenti jika kondisi berisi **FALSE**

```
while (expr) {  
    statement  
}
```

b) For

Perulangan menggunakan for mempunyai 3 ekspresi

1. Expr1 adalah ekspresi untuk memberikan nilai awal yang akan digunakan untuk perulangan.
2. Expr2 adalah ekspresi untuk memberikan kondisi dalam perulangan. Jika kondisi tersebut bernilai TRUE maka perulangan akan dilanjutkan. Jika kondisi tersebut bernilai FALSE maka perulangan akan berhenti.
3. Expr3 digunakan untuk mengatur nilai variabel yang digunakan pada expr1 (increment/decrement)

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

c) do – while

Perulangan menggunakan do – while ini akan berakhir jika ekspresi bernilai FALSE

```
do  
{  
    Statement;  
} while (expr)
```



PRAKTIK

1. Ubahlah script file **proses.php**, menjadi seperti berikut

```
1 File Proses <hr/>
2 <?php
3     $nilai = $_POST["angka"] ;
4     echo "Nilai Anda = " . $nilai . "<br/>" ;
5
6     if ($nilai > 100){
7         echo "Nilai Kelebihan" ;
8     } elseif ($nilai > 70) {
9         echo "Selamat Anda Lulus Ujian";
10    } elseif ($nilai >= 40) {
11        echo "Anda harus ujian lagi";
12    } else {
13        echo "Maaf, Gagal";
14    }
15 ?>
```

2. Jalankan file **form.php**, isilah data nilai angka kemudian klik **SUBMIT**,
 - Amati hasil yang ditampilkan !
 - Coba ulang pengisian formulir dengan memasukkan nilai angka yang lain, contoh 200, 90, 71, 70, 40, 39 atau 0, dan amati hasil yang ditampilkan serta jelaskan alasannya mengapa demikian ?
 - Tuliskan di laporan Anda !

3. Ubahlah script file **proses.php**, menjadi seperti berikut

```
1 File Proses <hr/>
2 <?php
3     $jurusan = $_POST["jurusan"] ;
4
5     switch ($jurusan){
6         case "TI":
7             echo "Jurusan anda Teknik Informatika <br/>";
8             break;
9         case "SI":
10            echo "Jurusan anda Sistem Informasi <br/>";
11            break;
12        case "MI":
13            echo "Jurusan anda Manajemen Informatika <br/>";
14            break;
15        case "TK":
16            echo "Jurusan anda Teknik Komputer <br/>";
17            break;
18        case "KA":
19            echo "Jurusan anda Komputerisasi Akuntansi <br/>";
20            break;
21        default :
22            echo "Jurusan tidak ada <br/>" ;
23    }
24
25    $nilai = $_POST["angka"] ;
26    echo "Nilai Anda = " . $nilai . "<br/>" ;
27
28    if ($nilai > 100){
29        echo "Nilai Kelebihan" ;
30    } elseif ($nilai > 70) {
31        echo "Selamat Anda Lulus Ujian";
32    } elseif ($nilai >= 40) {
33        echo "Anda harus ujian lagi";
34    } else {
35        echo "Maaf, Gagal";
36    }
37 ?>
```

4. Kemudian ubahlah script file **form.php**, menjadi seperti berikut !

```
1 <form name="form" action="proses.php" method="post">
2 Nilai : <input type="text" name="angka" /> <br/>
3 Jurusan :
4 <select name="jurusan">
5     <option value=" "> Pilih </option>
6     <option value="TI"> Teknik Informatika </option>
7     <option value="SI"> Sistem Informasi </option>
8     <option value="MI"> Manajemen Informatika </option>
9     <option value="TK"> Teknik Komputer </option>
10    <option value="KA"> Komputer Akuntansi </option>
11 </select> <br/>
12
13 <input type="submit" value="SUBMIT" />
14 </form>
```

- Jalankan file **form.php**, isilah data kemudian klik **Submit**.
- Amati hasil yang ditampilkan dan berikan alasan dari hasil yang ditampilkan !
- Tuliskan di Laporan Anda !

5. Ubah script file **proses.php**, dengan menghapus semua perintah **break** ;

- Jalankan file **form.php**, isilah data kemudian klik **Submit**.
- Amati hasil yang ditampilkan dan berikan alasan dari hasil tersebut!
- Tulis di Laporan Anda !

6. Ketikkan program berikut, simpan dengan nama **ulangfor.php**

ulangfor.php	
1	<?php
2	for (\$i=1;\$i<=5;\$i++){
3	echo " for ke \$i.";
4	}
5	?>

8. Buka browser, jalankan **ulangfor.php** (perintah <http://localhost/ulangfor.php>).

Amati hasil yang di tampilkan, jelaskan script yang tertulis di file **ulangfor.php**

9. Ubah $\$i \leq 5$ menjadi $\$i < 10$, jalankan kembali dan amati hasilnya

10. Ubah $\$i++$ menjadi $\$i+=2$, jalankan kembali dan amati hasilnya

Tuliskan di Laporan Anda!

11. Modifikasi file **ulangfor.php** menjadi bentuk seperti berikut, simpan kembali dengan nama **ulangwhile.php**

ulangwhile.php	
1	<?php
2	\$i=1;
3	while(\$i<=5)
4	{
5	echo " for ke \$i.";
6	\$i++;
7	}
8	?>

- Jalankan **ulangwhile.php** (perintah <http://localhost/ulangwhile.php>).
- Amati hasil yang di tampilkan, jelaskan script yang tertulis di file **ulangwhile.php**

12. Ubah $i \leq 5$ menjadi $i < 0$, jalankan kembali dan amati hasilnya
Tuliskan di Laporan Anda!

13. Modifikasi file ulangwhile.php menjadi bentuk seperti berikut, simpan kembali dengan nama **ulangdowhile.php**

ulangdowhile.php	
1	<?php
2	\$i=1;
3	do
4	{
5	echo " for ke \$i.";
6	\$i++;
7	} while(\$i<=5);
8	?>

- Jalankan **ulangdowhile.php** (perintah <http://localhost/ulangdowhile.php>).
- Amati hasil yang di tampilkan, jelaskan script yang tertulis di file **ulangdowhile.php**

14. Ubah $i \leq 5$ menjadi $i < 0$, jalankan kembali dan amati hasilnya

- Bandingkan hasil **ulangwhile.php** dengan **ulangdowhile.php** setelah dimodifikasi
- Bagaimana alur kerja perulangan while dan do-while, bandingkan
- Tuliskan di Laporan Anda!

15. Buatlah form dengan script seperti berikut dan simpat dengan nama file **formulang.php**

formulang.php	
1	<form name="form" action="ulangfor.php" method="post">
2	Awal <input type="text" name="awal" />
3	Akhir <input type="text" name="akhir" />
4	<input type="submit" value="Kirim" />
5	</form>

16. Modifikasi file **ulangfor.php** menjadi seperti berikut:

ulangfor.php	
1	<?php
2	\$awal = \$_POST["awal"] ;
3	\$akhir = \$_POST["akhir"] ;
4	for (\$i=\$awal ; \$i<=\$akhir ; \$i++){
5	echo " for ke \$i.";
6	}
7	?>

- Jalankan **formulang.php** (perintah <http://localhost/formulang.php>).
- Isikan Nilai Angka Awal dan Nilai Angka Akhir, dan amati hasil yang di tampilkan !
- Tuliskan di Laporan Anda!

17. Modifikasi file **ulangfor.php** menjadi seperti berikut:

ulangfor.php	
1	<?php
2	\$awal = \$_POST["awal"] ;
3	\$akhir = \$_POST["akhir"] ;
4	\$berhenti = \$_POST["henti"] ;
5	for (\$i=\$awal ; \$i<=\$akhir; \$i++){
6	echo " for ke \$i.";
7	if(\$i == \$berhenti){
8	break ;
9	}
10	}
11	?>

18. Kemudian modifikasi file **formulang.php** menjadi seperti berikut:

formulang.php	
1	<form name="form" action="ulangfor.php" method="post">
2	Awal <input type="text" name="awal" />
3	Akhir <input type="text" name="akhir" />
4	Berhentikan <input type="text" name="henti" />
5	<input type="submit" value="Kirim" />
6	</form>

- Jalankan **formulang.php** (perintah <http://localhost/formulang.php>).

- Isikan Nilai Angka Awal (misal 1), Nilai Angka Akhir (misal 10) dan Nilai Angka Berhentikan, dan amati hasil yang di tampilkan !
- Cobalah Anda isikan kembali Nilai Angka Awal, Nilai Angka Akhir dan Nilai Berhentikan dengan angka-angka yang lain, amati hasilnya!
- Apa yang anda dapat simpulkan dari penggunaan perintah **break**;
- Tuliskan di Laporan Anda!

19. Modifikasi file **ulangfor.php** menjadi seperti berikut:

ulangfor.php	
1	<?php
2	\$awal = \$_POST["awal"] ;
3	\$akhir = \$_POST["akhir"] ;
4	\$lanjutkan = \$_POST["lanjut"] ;
5	for (\$i=\$awal ; \$i<=\$akhir ; \$i++){
6	if(\$i == \$lanjutkan){
7	continue ;
8	}
9	echo " for ke \$i.";
10	}
11	?>

20. Kemudian modifikasi file **formulang.php** menjadi seperti berikut:

formulang.php	
1	<form name="form" action="ulangfor.php" method="post">
2	Awal <input type="text" name="awal" />
3	Akhir <input type="text" name="akhir" />
4	Lanjutkan <input type="text" name="lanjut" />
5	<input type="submit" value="Kirim" />
6	</form>

- Jalankan **formulang.php** (perintah <http://localhost/formulang.php>).
- Isikan Nilai Angka Awal (misal 1), Nilai Angka Akhir (misal 10), dan Nilai Angka Lanjutkan (misal 3), dan amati hasil yang di tampilkan !
- Cobalah Anda isikan kembali Nilai Angka Awal, Nilai Angka Akhir dan Nilai Lanjutkan dengan angka-angka yang lain, amati hasilnya!
- Apa yang anda dapat simpulkan dari penggunaan perintah **continue**; ?
- Tuliskan di Laporan Anda!



LATIHAN

diberikan oleh dosen atau asisten pengampu



TUGAS

diberikan oleh dosen atau asisten pengampu



REFERENSI

<https://php.net>