

TUGAS WEBGIS Pertemuan 12



M Gilang Martiansyah M

121450056

RC

Dosen Pengampu : Ilyas, S.Si., M.T.

PROGRAM STUDI Teknik Geomatika

JURUSAN Teknik Industri

INSTITUT TEKNOLOGI SUMATERA

2024/2025

Daftar isi

TUGAS !!	3
SWITCH	3
Input Code nya	3
Penjelasan Code :	3
Output	4
LOOPING	4
Input Codenya	4
Penjelasan Code :	5
Output	5
WHILE	6
Input Code nya	6
Penjelasan Code :	6
Output	6
DO WHILE	7
Input Code nya	7
Penjelasan Code :	7
Output	7
FOR	8
Input Code nya	8
Penjelasan Code :	8
Output	9
Kombinasi Contoh	9
Input Code nya	9
Output:	10
Kesimpulan	10

TUGAS !!

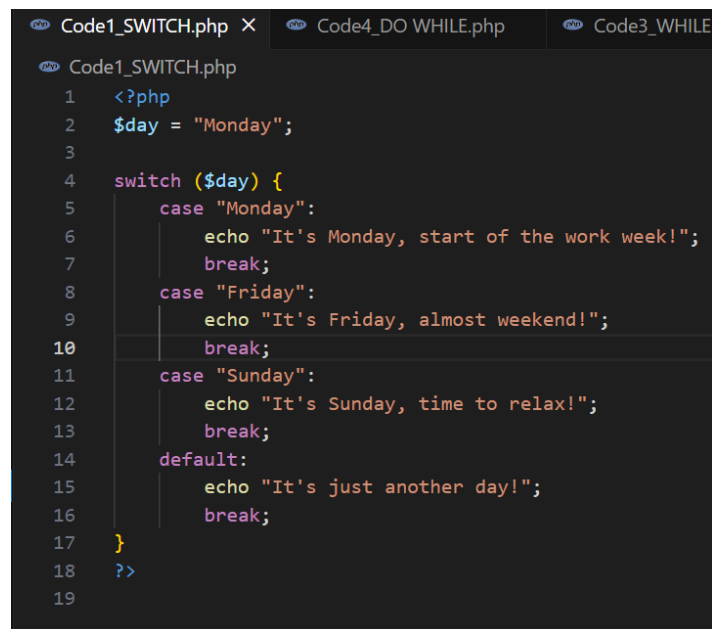
Mengimplementasikan Bahasa PHP dari Fungsi fungsi berikut dan memberikan penjelasannya:

1. SWITCH
2. LOOPING
3. WHILE
4. DO WHILE
5. FOR

SWITCH

SWITCH digunakan untuk memilih salah satu dari beberapa blok kode berdasarkan nilai ekspresi tertentu.

Input Code nya



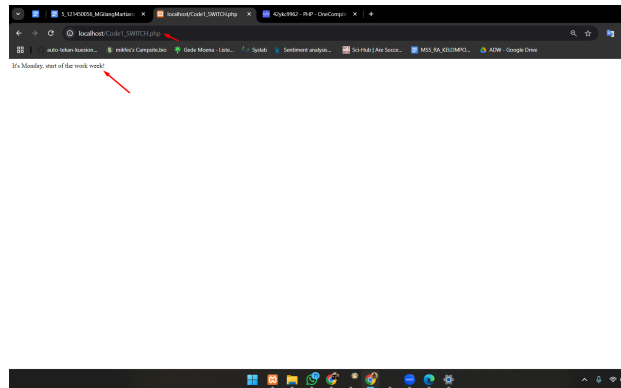
```
Code1_SWITCH.php X Code4_DO WHILE.php Code3_WHILE.php
Code1_SWITCH.php
1  <?php
2  $day = "Monday";
3
4  switch ($day) {
5      case "Monday":
6          echo "It's Monday, start of the work week!";
7          break;
8      case "Friday":
9          echo "It's Friday, almost weekend!";
10         break;
11     case "Sunday":
12         echo "It's Sunday, time to relax!";
13         break;
14     default:
15         echo "It's just another day!";
16         break;
17 }
18 ?>
19
```

Penjelasan Code :

- Nilai variabel \$day diperiksa menggunakan switch.
- Berdasarkan nilai \$day, kode dalam blok case yang sesuai akan dijalankan.
- break menghentikan eksekusi switch setelah blok kode dijalankan, mencegah eksekusi ke blok berikutnya.
- default akan dijalankan jika tidak ada nilai yang cocok.

Output

Output dari kode tersebut adalah:



It's Monday, start of the work week!

Output menunjukkan bahwa program mendeteksi hari sebagai "Monday" berdasarkan nilai variabel \$day. Karena "Monday" adalah awal minggu kerja dalam banyak konteks, program memberikan pesan motivasi yang relevan, menandai dimulainya aktivitas rutin setelah akhir pekan. Dalam kode PHP ini, nilai \$day diperiksa menggunakan struktur switch dengan beberapa pilihan seperti "Monday", "Friday", "Sunday", atau lainnya melalui blok default. Karena nilai \$day adalah "Monday", program mencetak pesan yang sesuai dan berhenti memeriksa pilihan lain. Blok default hanya dijalankan jika tidak ada kasus yang cocok. Pesan yang dihasilkan dirancang untuk menyesuaikan suasana hari Senin sebagai hari transisi menuju produktivitas kerja.

LOOPING

Looping adalah proses yang digunakan untuk mengulangi eksekusi blok kode tertentu selama suatu kondisi terpenuhi. Dalam PHP, terdapat beberapa jenis looping, yaitu:

Input Codenya

```
Code1_SWITCH.php | Code4_DO WHILE.php | Code3_WHILE.php | Code2_LOOPING.php X
Code2_LOOPING.php
1  <?php
2  // Looping menggunakan for
3  for ($i = 1; $i <= 10; $i++) {
4      echo "Juara $i, ";
5  }
6  ?>
```

Penjelasan Code :

Inisialisasi: $\$i = 1$

- Variabel $\$i$ diinisialisasi dengan nilai awal 1. Ini adalah titik awal untuk perulangan.

Kondisi: $\$i \leq 10$

- Perulangan akan terus berjalan selama nilai $\$i$ kurang dari atau sama dengan 10.
- Jika kondisi ini bernilai **false**, perulangan berhenti.

Increment: $\$i++$

- Setelah setiap iterasi, nilai $\$i$ akan bertambah 1 (increment).

Eksekusi Blok Kode:

- Pada setiap iterasi, perintah berikut dijalankan:

```
echo "Juara $i, ";
```

Hasilnya: String "Juara $\$i$, " akan dicetak ke layar, di mana $\$i$ adalah nilai iterasi saat ini.

Output



Hasil output dari kode tersebut adalah:

Juara 1, Juara 2, Juara 3, Juara 4, Juara 5, Juara 6, Juara 7, Juara 8, Juara 9, Juara 10,

Output menunjukkan bahwa program mencetak daftar peringkat dari 1 hingga 10 dengan format "Juara" diikuti angka peringkatnya. Penggunaan loop **for** memungkinkan pengulangan otomatis untuk mencetak setiap peringkat tanpa perlu menulis kode secara manual untuk setiap angka. Program ini secara sistematis menghasilkan daftar tersebut, memberikan kesan bahwa setiap peringkat memiliki nilai yang sama pentingnya dan terstruktur.

WHILE

WHILE adalah jenis loop yang terus berjalan selama kondisinya bernilai true

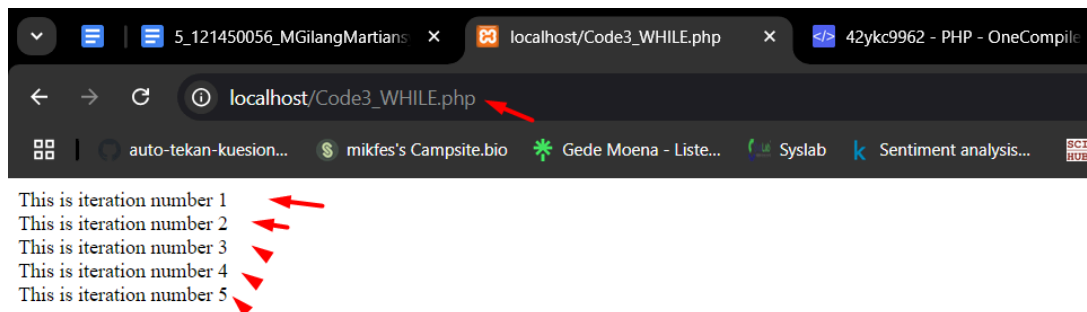
Input Code nya

```
Code1_SWITCH.php Code4_DO WHILE.php Code3_WHILE.php X Code2
Code3_WHILE.php
1  <?php
2  $count = 1;
3
4  while ($count <= 5) {
5      echo "This is iteration number $count<br>";
6      $count++;
7  }
8  ?>
9
```

Penjelasan Code :

- Loop dimulai dengan mengevaluasi kondisi `$count <= 5`.
- Selama kondisi bernilai `true`, kode dalam loop akan dijalankan.
- Setelah setiap iterasi, nilai `$count` ditingkatkan menggunakan `$count++`.

Output



```
localhost/Code3_WHILE.php
This is iteration number 1
This is iteration number 2
This is iteration number 3
This is iteration number 4
This is iteration number 5
```

Hasil outputnya dari kode tersebut:

This is iteration number 1
This is iteration number 2
This is iteration number 3
This is iteration number 4
This is iteration number 5

Hasil output dari kode tersebut adalah lima baris teks yang mencetak informasi tentang nomor iterasi, dimulai dari iterasi pertama hingga iterasi kelima. Setiap baris mencantumkan pesan seperti "**This is iteration number X**" dengan nilai X sesuai dengan nomor iterasi saat ini, yaitu dari 1 hingga 5. Setiap baris dipisahkan oleh tag HTML `
`, yang memastikan bahwa setiap output muncul di baris baru saat ditampilkan di browser. Output ini mencerminkan penggunaan loop 'while' yang efektif untuk mencetak pesan berulang dengan nilai yang meningkat setiap iterasi, memberikan hasil yang sistematis dan terstruktur sesuai logika program.

DO WHILE

DO WHILE mirip dengan while, tetapi memastikan blok kode dijalankan setidaknya sekali sebelum memeriksa kondisinya.

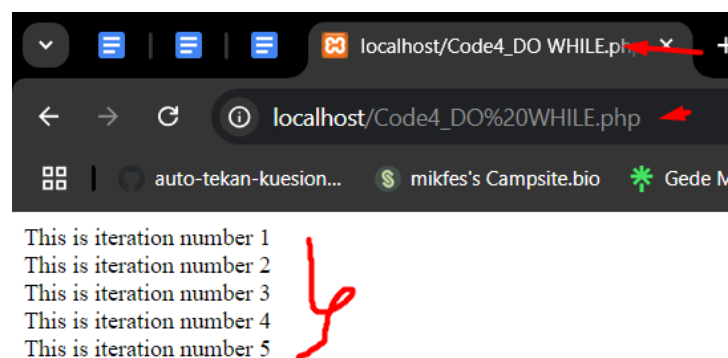
Input Code nya

```
Code4_DO WHILE.php
1  <?php
2  $count = 1;
3
4  do {
5      echo "This is iteration number $count<br>";
6      $count++;
7  } while ($count <= 5);
8  ?>
9
```

Penjelasan Code :

- Loop dimulai dengan mengeksekusi kode di dalam blok **do**.
- Setelah itu, kondisi **while (\$count <= 5)** diperiksa. Jika **true**, loop berlanjut.
- Jika kondisi **false**, loop berhenti.

Output



This is iteration number 1
This is iteration number 2
This is iteration number 3
This is iteration number 4
This is iteration number 5

Hasil output dari kode tersebut adalah:

This is iteration number 1
This is iteration number 2
This is iteration number 3
This is iteration number 4
This is iteration number 5

Output dari kode tersebut adalah lima baris teks yang mencantumkan informasi tentang nomor iterasi, dimulai dari *"This is iteration number 1"* hingga *"This is iteration number 5"*. Kode ini menggunakan loop **do-while**, yang memastikan bahwa blok kode akan dieksekusi setidaknya sekali, meskipun kondisi pada akhirnya tidak terpenuhi. Dalam hal ini, variabel **\$count** dimulai dengan nilai 1, dan pesan dicetak pada setiap iterasi, diikuti dengan tag HTML **
** untuk memisahkan setiap baris. Nilai **\$count** bertambah 1 pada setiap iterasi, hingga mencapai 6, yang menyebabkan loop berhenti karena kondisi **\$count <= 5** tidak lagi terpenuhi. Output ini mencerminkan proses iterasi yang terstruktur, di mana setiap iterasi menghasilkan teks dengan nomor yang sesuai, memberikan daftar hasil yang terorganisir dengan baik.

FOR

FOR digunakan untuk mengulangi blok kode dengan jumlah iterasi yang ditentukan.

Input Code nya

```
Code5_FOR.php
1  <?php
2  for ($i = 1; $i <= 5; $i++) {
3      echo "This is iteration number $i<br>";
4  }
5  ?>
6
```

Penjelasan Code :

- **for** memiliki tiga bagian:
 1. **Inisialisasi**: **\$i = 1** (nilai awal variabel loop).
 2. **Kondisi**: **\$i <= 5** (loop berjalan selama kondisi **true**).
 3. **Increment/Decrement**: **\$i++** (nilai **\$i** bertambah setelah setiap iterasi).
- Blok kode akan dijalankan selama kondisi **true**.

Output

Output dari kode tersebut adalah **"It's Monday, start of the work week!"**, yang menunjukkan bahwa program mengenali nilai variabel `\$day` sebagai "Monday". Program menggunakan struktur `switch` untuk mencocokkan nilai `\$day` dengan beberapa kemungkinan kasus, yaitu "Monday", "Friday", "Sunday", atau opsi lainnya melalui blok `default`. Karena nilai `\$day` adalah "Monday", program mencetak pesan yang relevan, yaitu pengingat bahwa hari Senin adalah awal minggu kerja. Setelah mencetak pesan tersebut, program menghentikan proses dengan perintah `break`, sehingga kasus lainnya tidak dieksekusi. Output ini mencerminkan logika seleksi yang digunakan untuk memberikan pesan khusus berdasarkan nilai hari yang telah ditentukan.

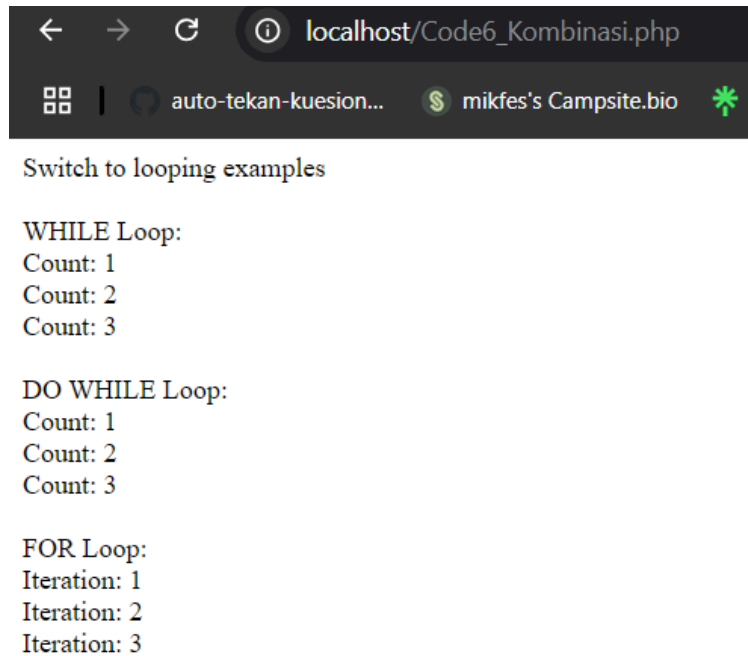
Kombinasi Contoh

Jika digabungkan, berikut contoh sederhana untuk memahami implementasi semua fungsi tersebut:

Input Code nya

```
Code6_Kombinasi.php
1  <?php
2  // SWITCH
3  $command = "loop";
4  switch ($command) {
5      case "loop":
6          echo "Switch to looping e
7          break;
8      default:
9          echo "Unknown command<br>
10         break;
11 }
12
13 // WHILE
14 echo "WHILE Loop:<br>";
15 $count = 1;
16 while ($count <= 3) {
17     echo "Count: $count<br>";
18     $count++;
19 }
20
21 // DO WHILE
22 echo "<br>DO WHILE Loop:<br>";
23 $count = 1;
24 do {
25     echo "Count: $count<br>";
26     $count++;
27 } while ($count <= 3);
28
29 // FOR
30 echo "<br>FOR Loop:<br>";
31 for ($i = 1; $i <= 3; $i++) {
32     echo "Iteration: $i<br>";
33 }
34 ?>
```

Output:



```
Switch to looping examples

WHILE Loop:
Count: 1
Count: 2
Count: 3

DO WHILE Loop:
Count: 1
Count: 2
Count: 3

FOR Loop:
Iteration: 1
Iteration: 2
Iteration: 3
```

1. SWITCH akan mencetak pesan berdasarkan nilai `$command`.
2. WHILE akan mencetak iterasi dengan batas tertentu.
3. DO WHILE memastikan eksekusi setidaknya sekali.
4. FOR digunakan untuk iterasi dengan batas yang jelas.

Kesimpulan

Tugas ini menjelaskan dan mengimplementasikan fungsi **SWITCH**, **FOR**, **WHILE**, dan **DO WHILE** untuk memahami looping dalam PHP. Setiap jenis memiliki karakteristik yang unik dan aplikasi berbeda. Berikut adalah kesimpulannya:

1. **SWITCH** digunakan untuk seleksi logika berdasarkan beberapa kondisi. Ini berguna untuk meminimalkan kode panjang dalam banyak pilihan.
2. **FOR** cocok untuk iterasi dengan jumlah yang sudah ditentukan, seperti mencetak daftar angka atau string berulang.
3. **WHILE** digunakan untuk menjalankan blok kode selama kondisi tetap bernilai `true`. Efektif untuk kasus tanpa batas iterasi yang pasti.
4. **DO WHILE** memastikan setidaknya satu kali eksekusi, meskipun kondisi tidak terpenuhi sejak awal.

Penggunaan fungsi-fungsi ini membantu menyederhanakan logika kompleks dan membuat kode lebih terstruktur, terorganisir, dan dapat digunakan kembali. Hal ini sangat penting untuk pengembangan aplikasi berbasis **WebGIS**, di mana iterasi data seperti koordinat atau fitur peta sangat sering dilakukan.