

**LAPORAN PRAKTIKUM
DESAIN WEB MODUL 5
PERCABANGAN, PERULANGAN**



**ANNISA NABILA PUTRI
2131710016
MI1F-05**

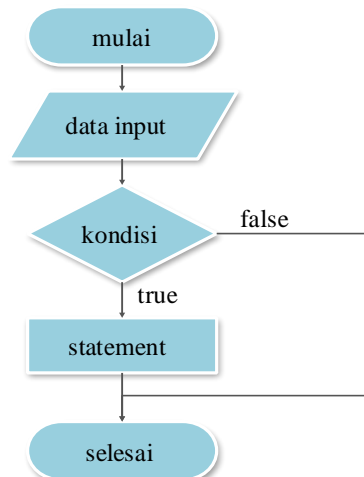
**POLITEKNIK NEGERI MALANG
2022**

PERTEMUAN 6

Percabangan dan Perulangan

Praktikum Bagian 1. Pecabangan (IF)

Salah satu struktur percabangan adalah menggunakan perintah IF. Struktur percabangan IF ditunjukkan pada diagram alur (flowchart) berikut ini:

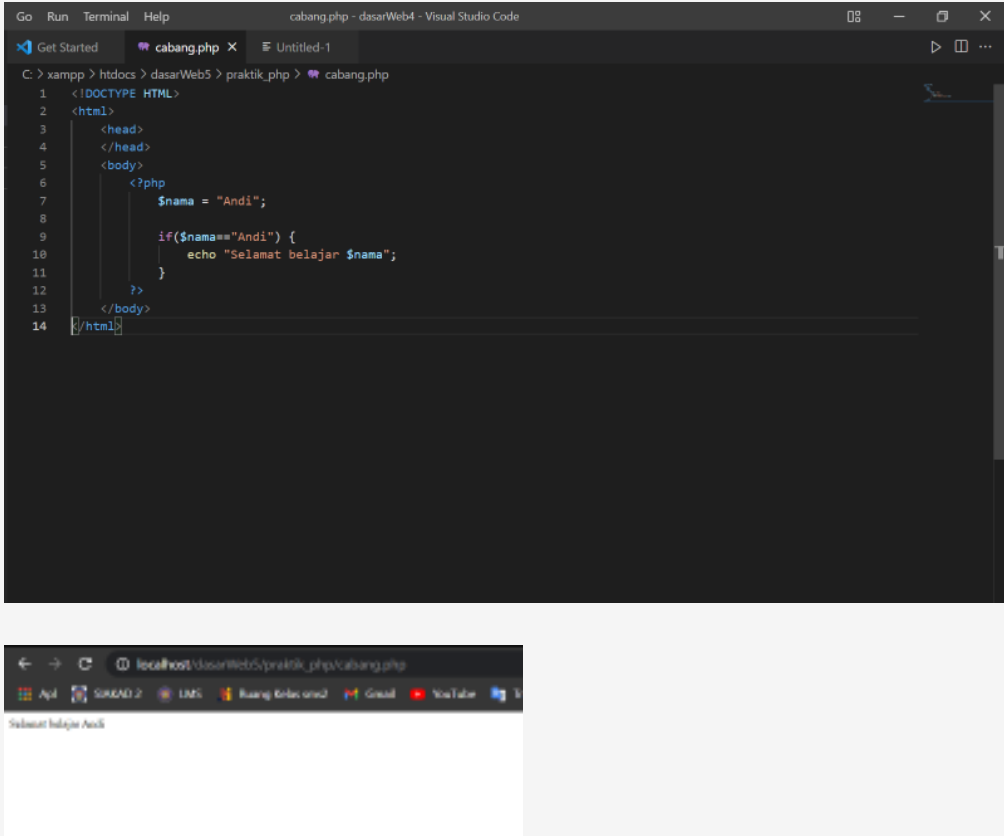
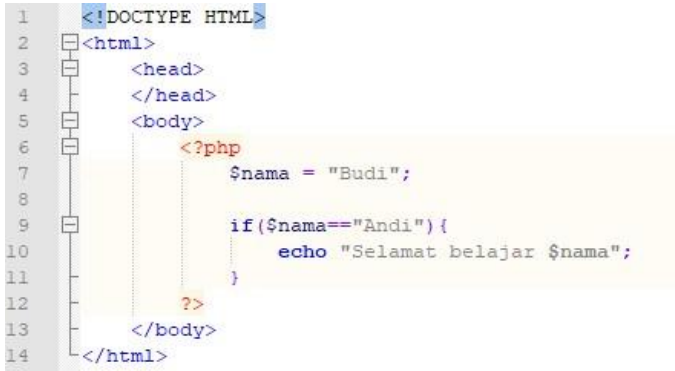


Berdasarkan diagram alur diatas, percabangan dengan menggunakan perintah IF akan mengeksekusi kode program jika kondisi benar. Jika data input sesuai dengan kondisi (bernilai TRUE), maka statement akan dieksekusi. Sintaks perintah IF adalah sebagai berikut:

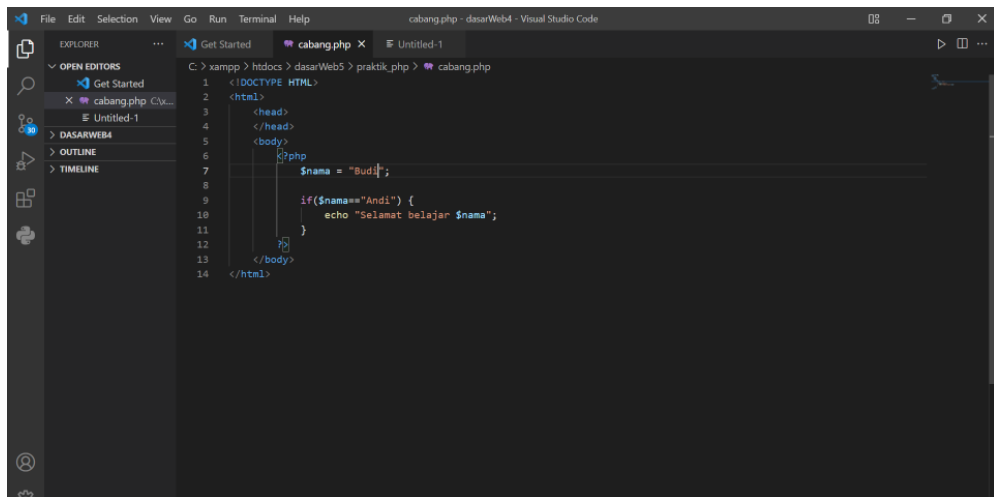
```
<?php
if(kondisi){
statement;
} ?>
```

Selanjutnya untuk memahami bagaimana menerapkan percabangan if, ikuti langkah-langkah praktikum berikut:

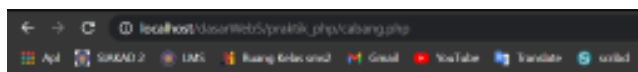
Langkah	Keterangan
1	<p>Buat file baru dengan nama <code>cabang.php</code> di dalam direktori <code>praktik_php</code>, kemudian ketikkan kode berikut:</p> <pre>1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$nama = "Andi"; 8 9 if(\$nama=="Andi"){ 10 echo "Selamat belajar \$nama"; 11 } 12 ?> 13 </body> 14 </html></pre>

2	<p>Simpan file dan jalankan kode program tersebut dengan mengetikkan <code>localhost/dasarWeb/praktik_php/cabang.php</code></p>
3	<p>Amati hasil yang ditampilkan</p> <p>percabangan dengan menggunakan perintah IF akan mengeksekusi kode program jika kondisi benar. Jika data input sesuai dengan kondisi (bernilai TRUE), maka statement akan dieksekusi.</p>  <p>The screenshot shows the Visual Studio Code editor with a file named <code>cabang.php</code> open. The code is as follows:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$nama = "Andi"; 8 9 if(\$nama=="Andi") { 10 echo "Selamat belajar \$nama"; 11 } 12 ?> 13 </body> 14 </html> </pre> <p>Below the editor, a browser window is shown displaying the output: "Selamat belajar Andi".</p>
4	<p>Lakukan modifikasi kode program langkah ke-1 dengan mengganti nilai variabel <code>\$nama</code> pada baris ke-7 menjadi "Budi", sehingga kode program menjadi seperti berikut:</p>  <p>The screenshot shows the same code as before, but with the value of <code>\$nama</code> changed to "Budi" on line 7:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$nama = "Budi"; 8 9 if(\$nama=="Andi") { 10 echo "Selamat belajar \$nama"; 11 } 12 ?> 13 </body> 14 </html> </pre>
5	<p>Ulangi langkah ke-2</p>
6	<p>Amati perbedaan hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 1)</p>

percabangan dengan menggunakan perintah IF akan mengeksekusi kode program jika kondisi benar. Karena data input berbeda dengan kondisi (bernilai FALSE), maka statement tidak dieksekusi.

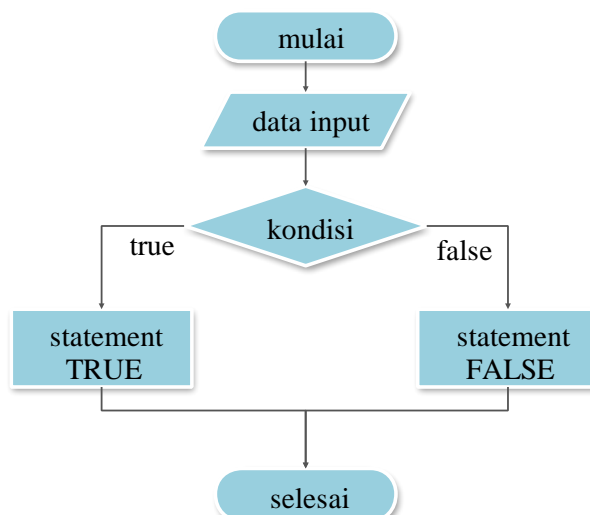


```
1 <!DOCTYPE HTML>
2 <html>
3 <head>
4 </head>
5 <body>
6 <?php
7     $nama = "Budji";
8
9     if($nama="Andi") {
10         echo "Selamat belajar $nama";
11     }
12
13 </body>
14 </html>
```



Praktikum Bagian 2. Percabangan (IF-ELSE)


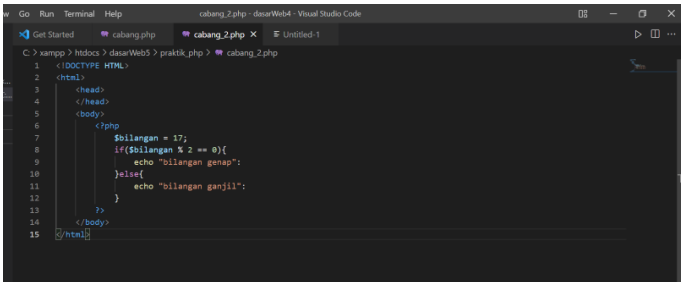
Struktur lainnya dari percabangan adalah IF – ELSE. Jika percabangan IF digunakan untuk percabangan alur program dengan 1 pilihan saja, maka dengan percabangan IF – ELSE kita dapat membuat percabangan kedua. Percabangan ketika kondisi IF tidak terpenuhi, atau expressi IF menghasilkan nilai FALSE. Struktur percabangan IF – ELSE ditunjukkan pada diagram alur (flowchart) berikut ini:

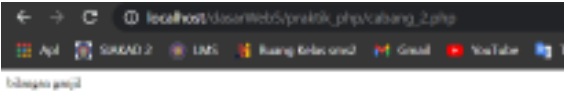
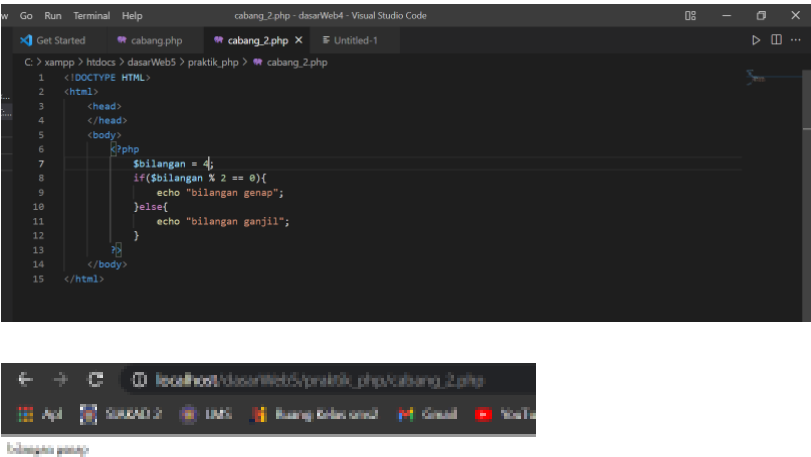


Berdasarkan diagram alur diatas, jika data input memenuhi kondisi (bernilai TRUE), maka statement TRUE akan dieksekusi. Jika data input tidak memenuhi kondisi (bernilai FALSE), maka statement FALSE yang akan dieksekusi. Struktur ELSE adalah kondisi yang bernilai FALSE. Sintaks percabangan IF – ELSE adalah sebagai berikut:

```
<?php
if(kondisi){
statement1;
}else{
statement2;
} ?>
```

Selanjutnya untuk memahami bagaimana menerapkan percabangan if-else, ikuti langkahlangkah praktikum berikut:

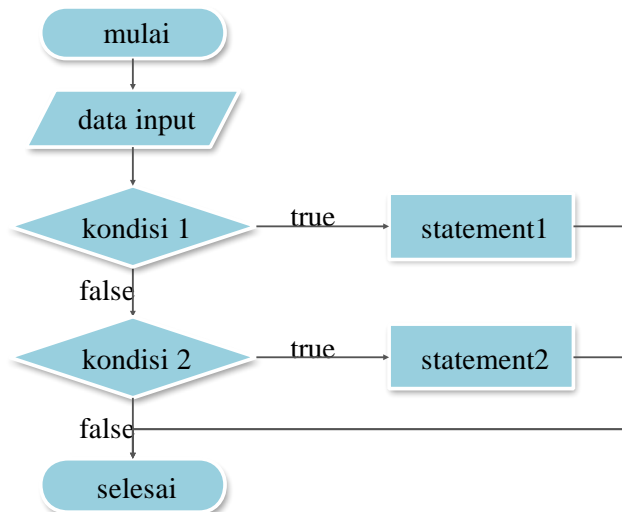
Langkah	Keterangan
1	Buat file baru dengan nama cabang_2.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:
	 <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$bilangan = 17; 8 if(\$bilangan % 2 == 0){ 9 echo "bilangan genap"; 10 }else{ 11 echo "bilangan ganjil"; 12 } 13 ?> 14 </body> 15 </html> </pre>
2	Simpan file dan jalankan kode program tersebut dengan mengetikkan <code>localhost/dasarWeb/praktik_php/cabang_2.php</code>
3	<p>Amati hasil yang ditampilkan</p> <p>Karena kondisi mengetikkan angka 17 maka outputnya menghasilkan bilangan ganjil</p> 

	
4	<p>Lakukan modifikasi kode program langkah ke-1 dengan mengganti nilai variabel \$bilangan pada baris ke-7 menjadi 4, sehingga kode program menjadi seperti berikut:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$bilangan = 4; 8 if(\$bilangan % 2 == 0){ 9 echo "bilangan genap"; 10 }else{ 11 echo "bilangan ganjil"; 12 } 13 ?> 14 </body> 15 </html> </pre>
5	<p>Ulangi langkah ke-2</p>
6	<p>Amati perbedaan hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 2)</p> <p>Karena kondisi mengetikkan 4 maka outputnya menghasilkan bilangan genap</p> 

Praktikum Bagian 3. Percabangan (IF-ELSE IF-ELSE)

Struktur lainnya dari percabangan adalah IF – ELSE IF – ELSE (nested if). Jika percabangan IF digunakan untuk percabangan alur program dengan 1 pilihan saja dan percabangan IF – ELSE kita dapat membuat percabangan kedua, maka nested if dapat digunakan untuk

percabangan dengan berbagai kemungkinan kondisi. Struktur percabangan IF – ELSE IF – ELSE ditunjukkan pada diagram alur (flowchart) berikut ini:

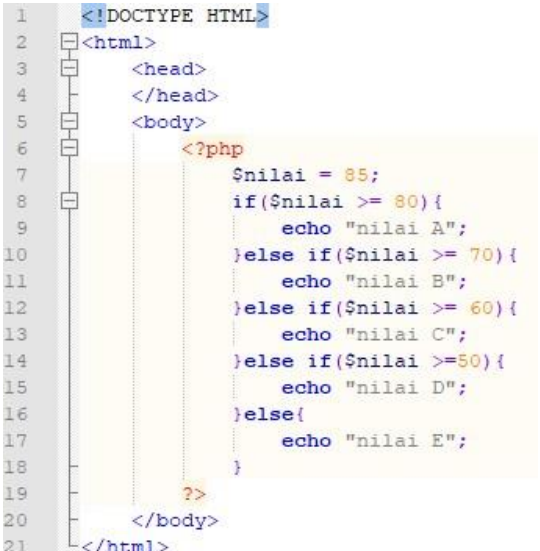
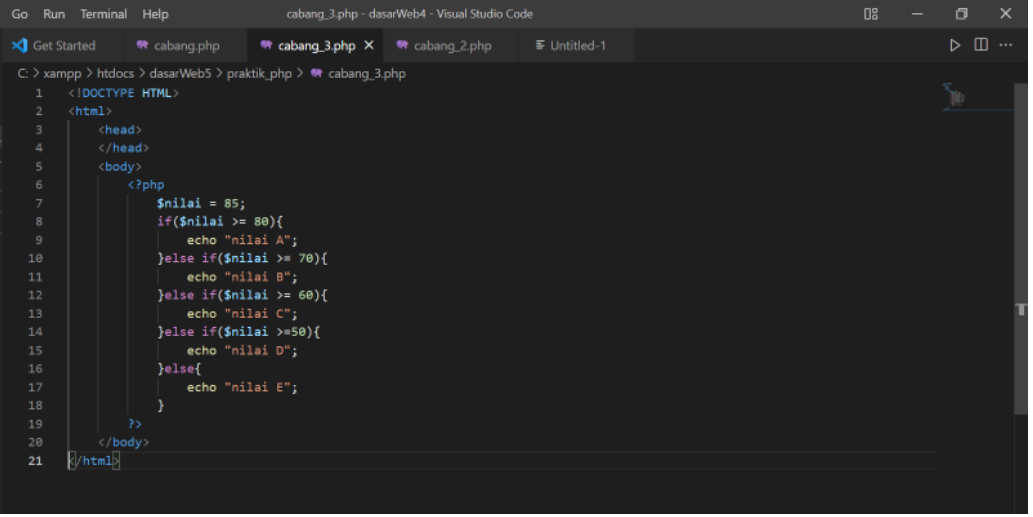
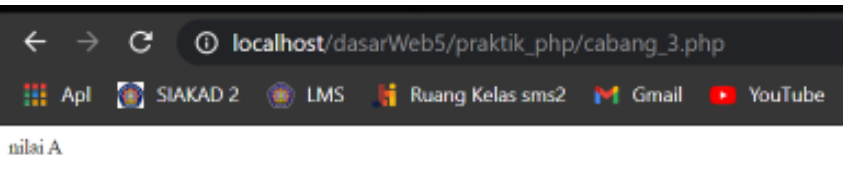


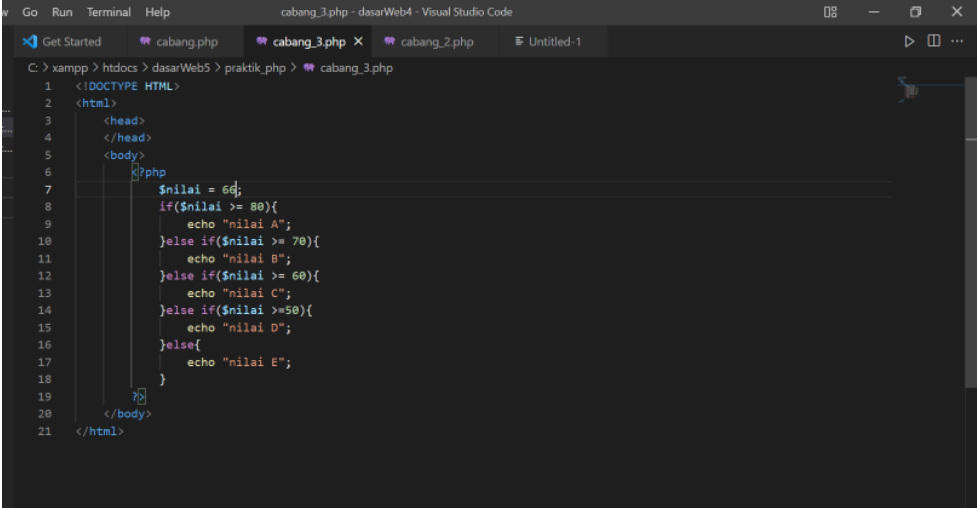
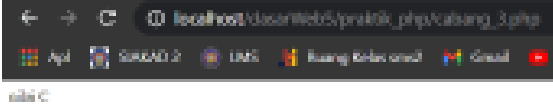
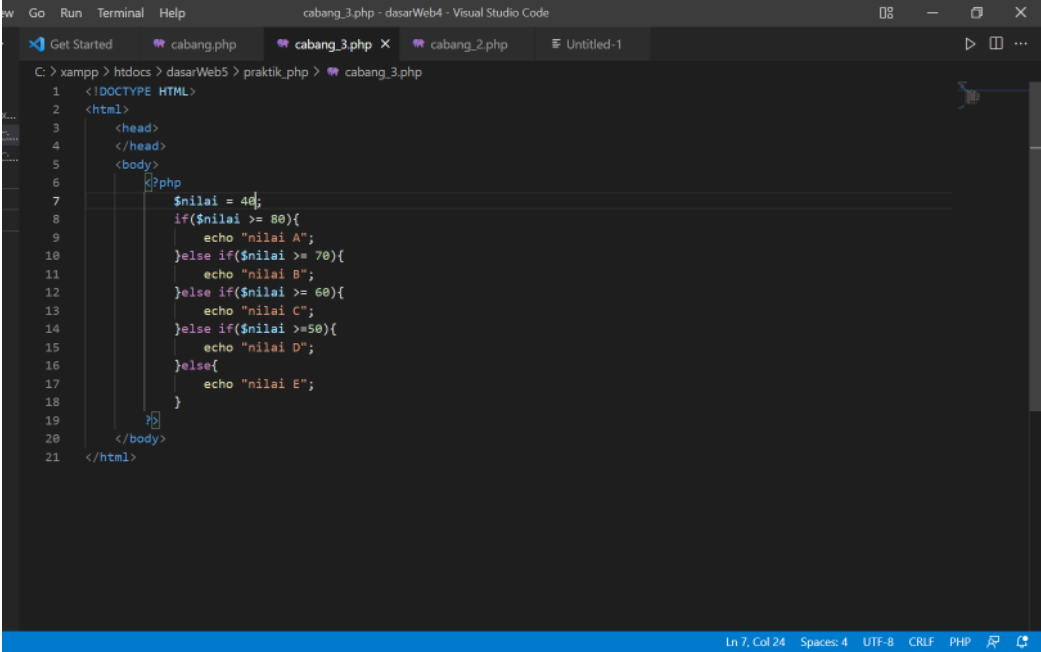
Berdasarkan diagram alur diatas, jika data input memenuhi kondisi1 (bernilai TRUE), maka statement1 akan dieksekusi. Jika data input tidak memenuhi kondisi1 (bernilai FALSE), maka data akan diseleksi berdasarkan kondisi2. Jika data input memenuhi kondisi2 (bernilai TRUE), maka statement2 akan dieksekusi. Sintaks percabangan IF – ELSE IF – ELSE adalah sebagai berikut:

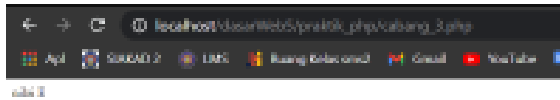
```
<?php
if(kondisi1) {
statement1;
}else
if(kondisi2) {
statement2;
} ?>
```

Selanjutnya untuk memahami bagaimana menerapkan percabangan if-elseif-else, ikuti langkah-langkah praktikum berikut:

Langkah	Keterangan
1	Buat file baru dengan nama cabang_3.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:

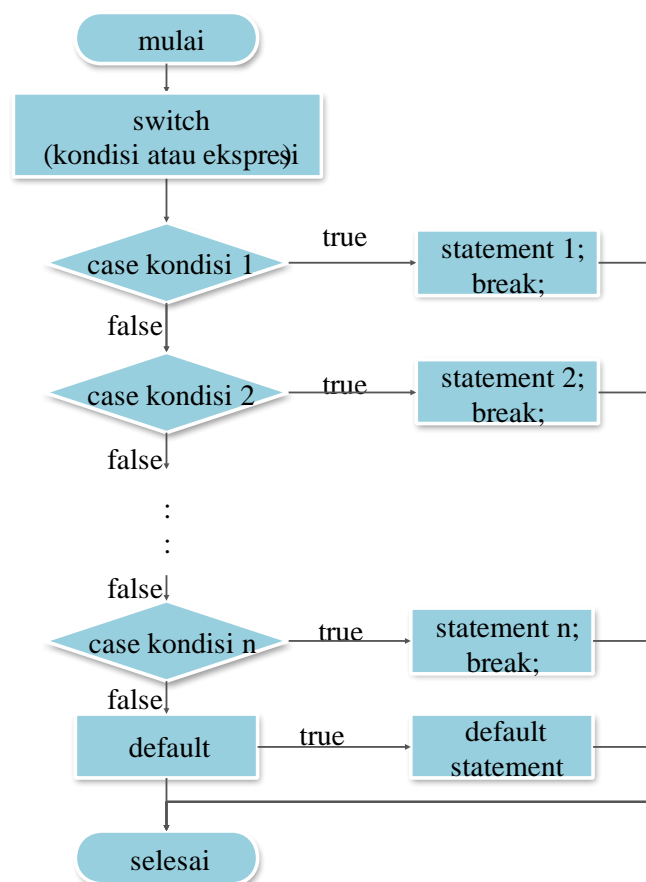
	 <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$nilai = 85; 8 if(\$nilai >= 80){ 9 echo "nilai A"; 10 }else if(\$nilai >= 70){ 11 echo "nilai B"; 12 }else if(\$nilai >= 60){ 13 echo "nilai C"; 14 }else if(\$nilai >= 50){ 15 echo "nilai D"; 16 }else{ 17 echo "nilai E"; 18 } 19 ?> 20 </body> 21 </html> </pre>
2	<p>Simpan file dan jalankan kode program tersebut dengan mengetikkan localhost/dasarWeb/praktik_php/cabang_3.php</p>
3	<p>Amati hasil yang ditampilkan</p> <p>Karena kondisi \$nilai bernilai di atas 80 maka menampilkan statement nilai A</p>  

4	<p>Lakukan modifikasi kode program langkah ke-1 dengan mengganti nilai variabel \$nilai pada baris ke-7 dengan beberapa nilai berikut: a. 66 b. 40</p>
5	<p>Ulangi langkah ke-2</p>
6	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 3)</p> <p>Karena kondisi \$nilai bernilai di atas 60 maka menampilkan statement nilai C</p>   <p>Karena kondisi \$nilai bernilai di bawah 50 maka menampilkan statement nilai E</p> 



Praktikum Bagian 4. Percabangan (SWITCH-CASE)

Struktur lainnya dari percabangan adalah SWITCH – CASE. Selain menggunakan percabangan IF – ELSE IF – ELSE, percabangan SWITCH – CASE dapat digunakan ketika kita membutuhkan banyak sekali kondisi. Struktur percabangan SWITCH – CASE ditunjukkan pada diagram alur (flowchart) berikut ini:



Berdasarkan diagram alur di atas, percabangan SWITCH – CASE digunakan jika kondisi yang ada sangat banyak, sehingga kondisi diwakili oleh setiap CASE. Program akan memilih CASE mana yang sesuai dengan data input berdasarkan kondisi atau ekspresi SWITCH. Statement DEFAULT dieksekusi jika data input tidak memenuhi seluruh CASE yang ada. Sintaks percabangan SWITCH - CASE adalah sebagai berikut:

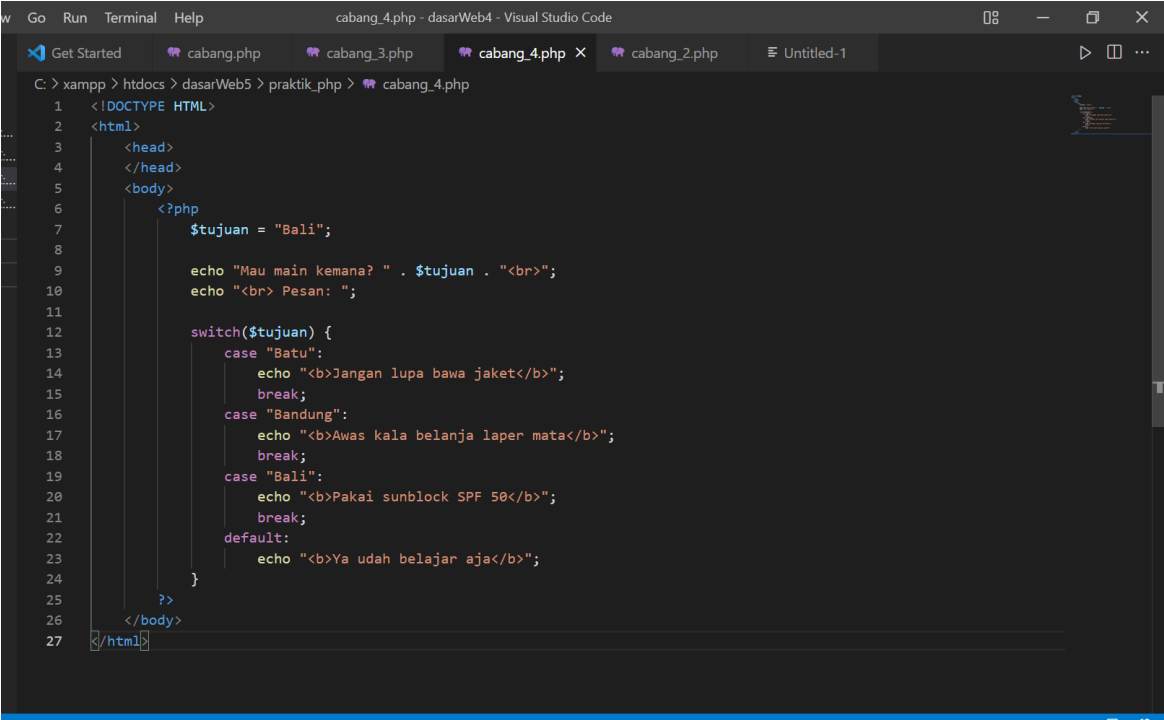
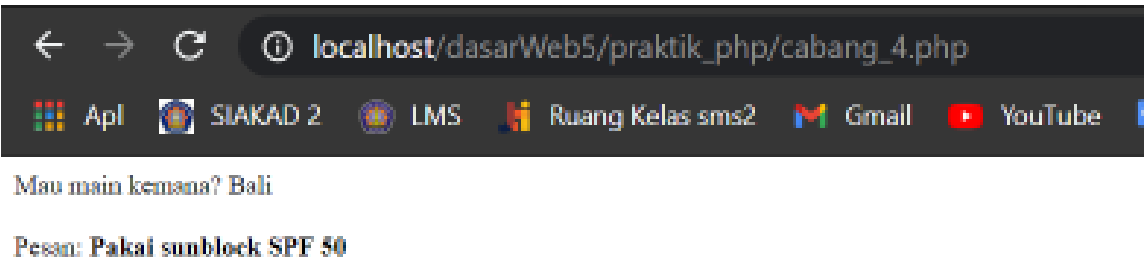
```

<?php
    switch(variabel atau ekspresi)
    {
        case
        kondisi 1:
        statement 1;
        break;
        case
        kondisi 2:
        statement 2;
        break;
        :
        case kondisi n:
        statement n;
        break;
        default:
        statement default;
    }
?>

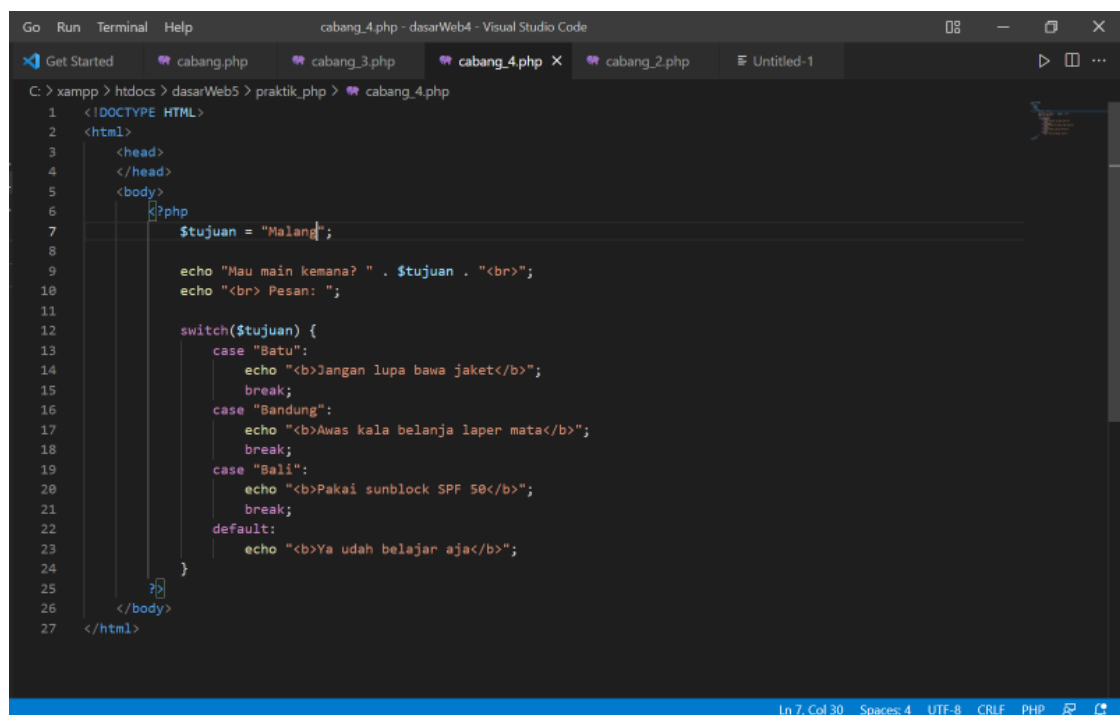
```

Selanjutnya untuk memahami bagaimana menerapkan percabangan if-elseif-else, ikuti langkah-langkah praktikum berikut:

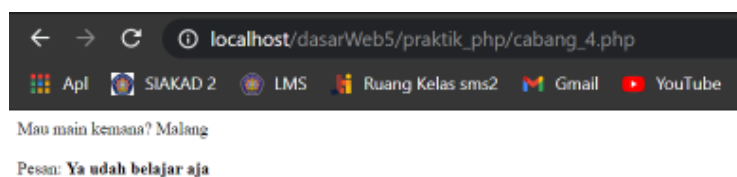
Langkah	Keterangan
1	<p>Buat file baru dengan nama cabang_4.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$tujuan = "Bali"; 8 9 echo "Mau main kemana? " . \$tujuan . "
"; 10 echo "
 Pesan: "; 11 12 switch(\$tujuan){ 13 case "Batu": 14 echo "Jangan lupa bawa jaket"; 15 break; 16 case "Bandung": 17 echo "Awat kalau belanja laper mata"; 18 break; 19 case "Bali": 20 echo "Pakai sunblock SPF 50"; 21 break; 22 default: 23 echo "Ya udah belajar aja"; 24 } 25 ?> 26 </body> 27 </html> </pre>

2	<p>Simpan file dan jalankan kode program tersebut dengan mengetikkan</p> <pre>localhost/dasarWeb/praktik_php/cabang_4.php</pre>
3	<p>Amati hasil yang ditampilkan</p> <p>Data input menginputkan \$tujuan bali maka Program akan memilih CASE mana yang sesuai dengan data input berdasarkan kondisi atau ekspresi SWITCH</p>  <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$tujuan = "Bali"; 8 9 echo "Mau main kemana? " . \$tujuan . "
"; 10 echo "
 Pesan: "; 11 12 switch(\$tujuan) { 13 case "Batu": 14 echo "Jangan lupa bawa jaket"; 15 break; 16 case "Bandung": 17 echo "Awat kala belanja laper mata"; 18 break; 19 case "Bali": 20 echo "Pakai sunblock SPF 50"; 21 break; 22 default: 23 echo "Ya udah belajar aja"; 24 } 25 ?> 26 </body> 27 </html> </pre>  <p>Mau main kemana? Bali</p> <p>Pesan: Pakai sunblock SPF 50</p>
4	<p>Lakukan modifikasi kode program langkah ke-1 dengan mengganti nilai variabel \$tujuan pada baris ke-7 dengan "Malang"</p>
5	<p>Ulangi langkah ke-2</p>
6	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 4)</p>

Program akan memilih CASE mana yang sesuai dengan data input berdasarkan kondisi atau ekspresi SWITCH

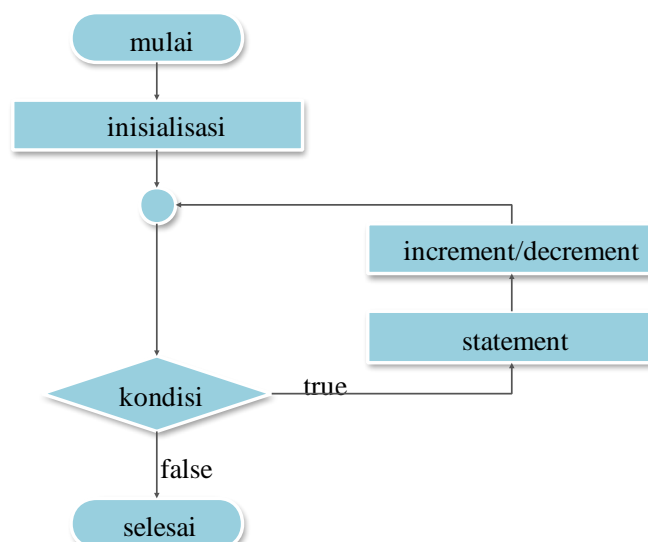


```
1 <!DOCTYPE HTML>
2 <html>
3   <head>
4   </head>
5   <body>
6     <?php
7       $tujuan = "Malang";
8
9       echo "Mau main kemana? " . $tujuan . "<br>";
10      echo "<br> Pesan: ";
11
12      switch($tujuan) {
13        case "Batu":
14          echo "<b>Jangan lupa bawa jaket</b>";
15          break;
16        case "Bandung":
17          echo "<b>Awat kala belanja laper mata</b>";
18          break;
19        case "Bali":
20          echo "<b>Pakai sunblock SPF 50</b>";
21          break;
22        default:
23          echo "<b>Ya udah belajar aja</b>";
24      }
25    <?php
26  </body>
27 </html>
```



Praktikum Bagian 5. Perulangan (FOR)

Dalam pembuatan sebuah aplikasi web, kemungkinan kita dihadapkan kasus yang menuntut kita untuk melakukan perulangan terhadap statement-statement tertentu. Salah satu struktur perulangan adalah dengan menggunakan perintah FOR.



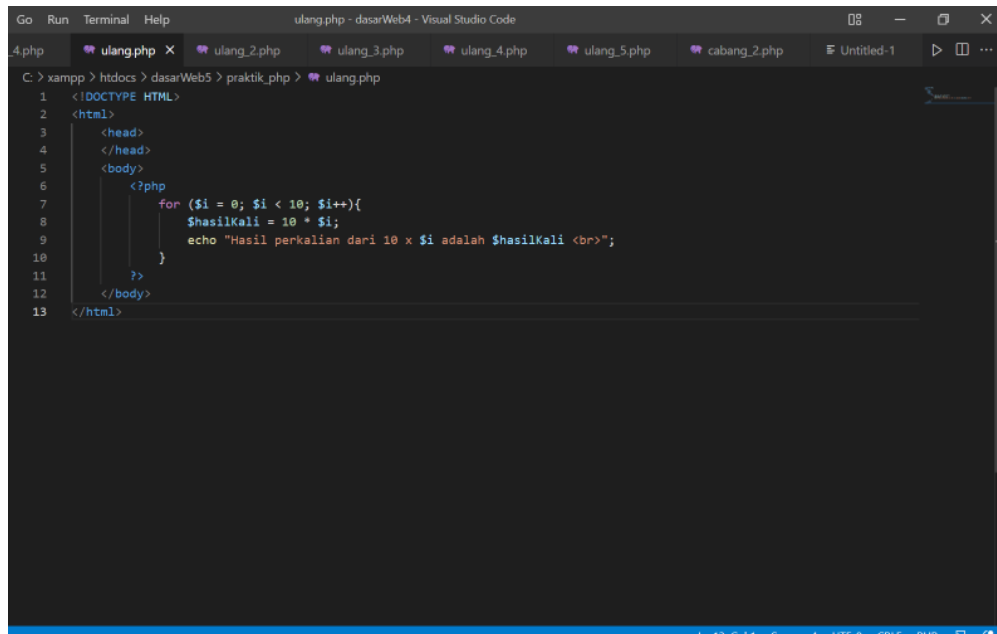
Sintaks perulangan FOR adalah sebagai berikut:

```
<?php

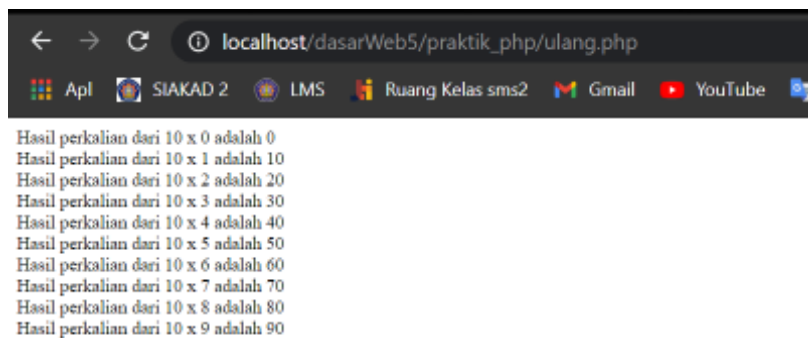
for (inisi
    alisasi;
    kondisi;
    increment
    /decremen
    t) {
    statement
    ;
    } ?>
```

Untuk memahami bagaimana menerapkan perulangan FOR, ikuti langkah-langkah praktikum berikut:

Langkah	Keterangan
1	<p>Buat file baru dengan nama <code>ulang.php</code> di dalam direktori <code>praktik_php</code>, kemudian ketikkan kode berikut:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 for (\$i = 0; \$i < 10; \$i++){ 8 \$hasilKali = 10 * \$i; 9 echo "Hasil perkalian dari 10 x \$i adalah \$hasilKali
"; 10 } 11 ?> 12 </body> 13 </html></pre>
2	<p>Simpan file dan jalankan kode program tersebut dengan mengetikkan</p> <pre>localhost/dasarWeb/praktik_php/ulang.php</pre>
3	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 5)</p> <p>Hitungan dimulai dari 0 karena memberikan nilai $\\$i = 0$, perulangan diulang selama nilai $\\$i$ lebih kecil dari 10 yang berarti perulangan mengulang sebanyak 10 kali. $\\$i++$ merupakan nilai $\\$i$ akan ditambah 1 setiap kali melakukan perulangan</p>

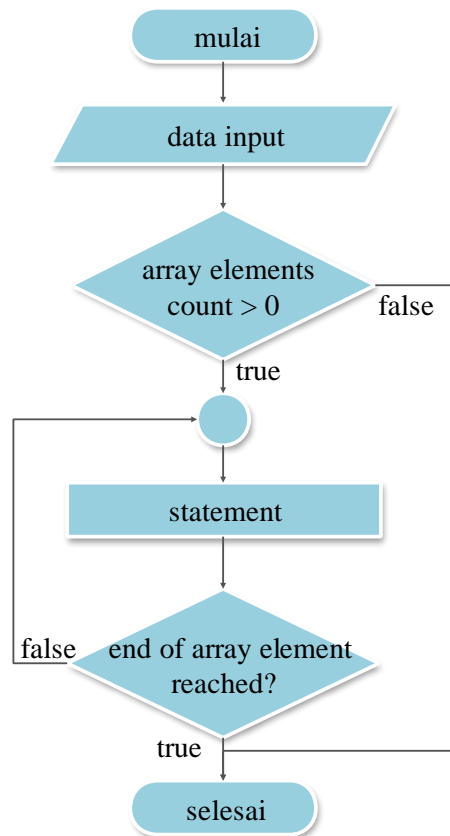


```
1 <!DOCTYPE HTML>
2 <html>
3   <head>
4   </head>
5   <body>
6     <?php
7       for ($i = 0; $i < 10; $i++){
8         $hasilKali = 10 * $i;
9         echo "Hasil perkalian dari 10 x $i adalah $hasilKali <br>";
10      }
11    <?>
12  </body>
13 </html>
```



Praktikum Bagian 6. Perulangan (FOREACH)

FOREACH adalah perulangan yang digunakan untuk tipe data array. FOREACH digunakan untuk mengulangi setiap pasangan indeks – nilai di dalam array. Struktur perulangan FOREACH ditunjukkan dengan diagram alur (flowchart) berikut ini:



Sintaks perulangan FOREACH adalah sebagai berikut:

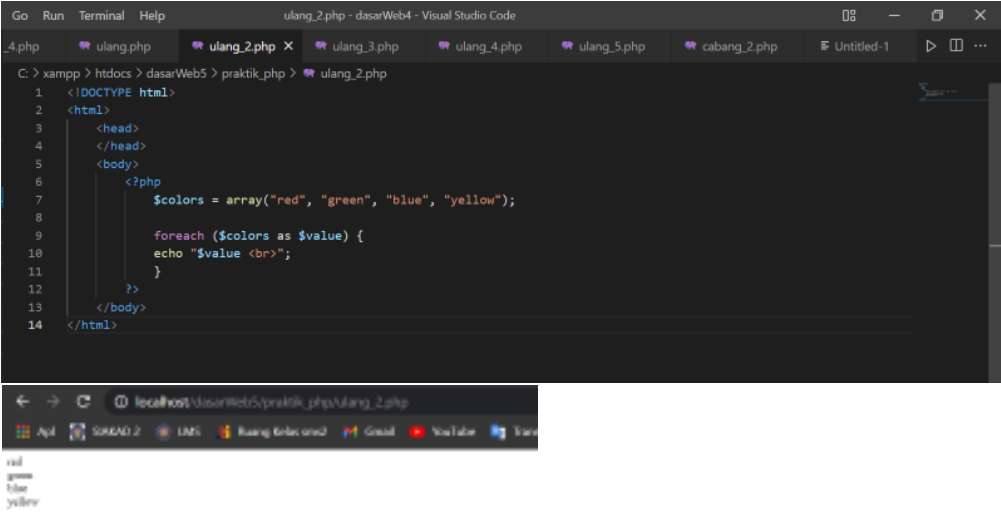
```

<?php

foreach ($
array as
$value) {
statement
;
    } ?>
  
```

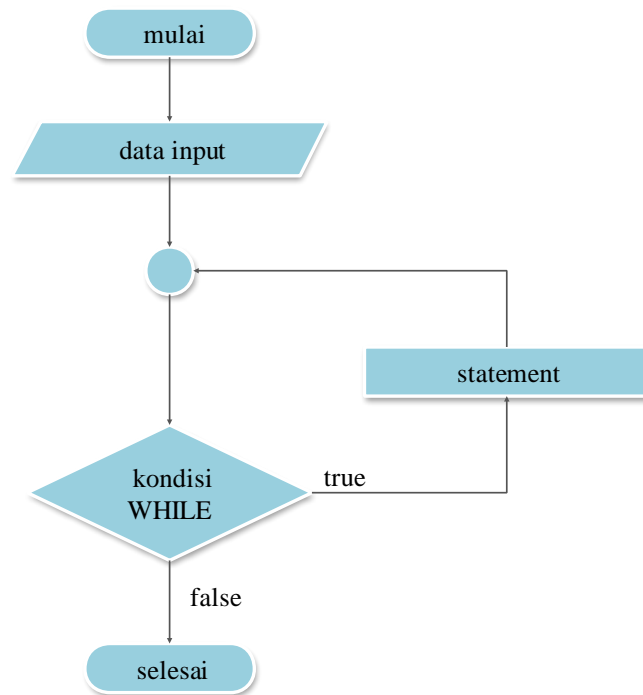
Untuk memahami bagaimana menerapkan perulangan FOR, ikuti langkah-langkah praktikum berikut:

Langkah	Keterangan
1	Buat file baru dengan nama <code>ulang_2.php</code> di dalam direktori <code>praktik_php</code> , kemudian ketikkan kode berikut:

	 <pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$colors = array("red", "green", "blue", "yellow"); 8 9 foreach (\$colors as \$value) { 10 echo "\$value
"; 11 } 12 ?> 13 </body> 14 </html> </pre>
2	<p>Simpan file dan jalankan kode program tersebut dengan mengetikkan localhost/dasarWeb/praktik_php/ulang_2.php</p>
3	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 6)</p> <p>Foreach digunakan untuk mengulangi setiap pasangan indeks – nilai di dalam array. Jadi outputnya menjadi red green blue yellow</p> 

Praktikum Bagian 7. Perulangan (WHILE)

Selain FOR dan FOREACH, untuk melakukan eksekusi statement berulang kali dapat digunakan perintah WHILE. Dengan menggunakan perintah WHILE, statement akan dieksekusi berulang kali selama memenuhi kondisi (kondisi bernilai TRUE. Struktur WHILE ditunjukkan dengan diagram alur (flowchart) berikut ini:



Sintaks perulangan WHILE adalah sebagai berikut:

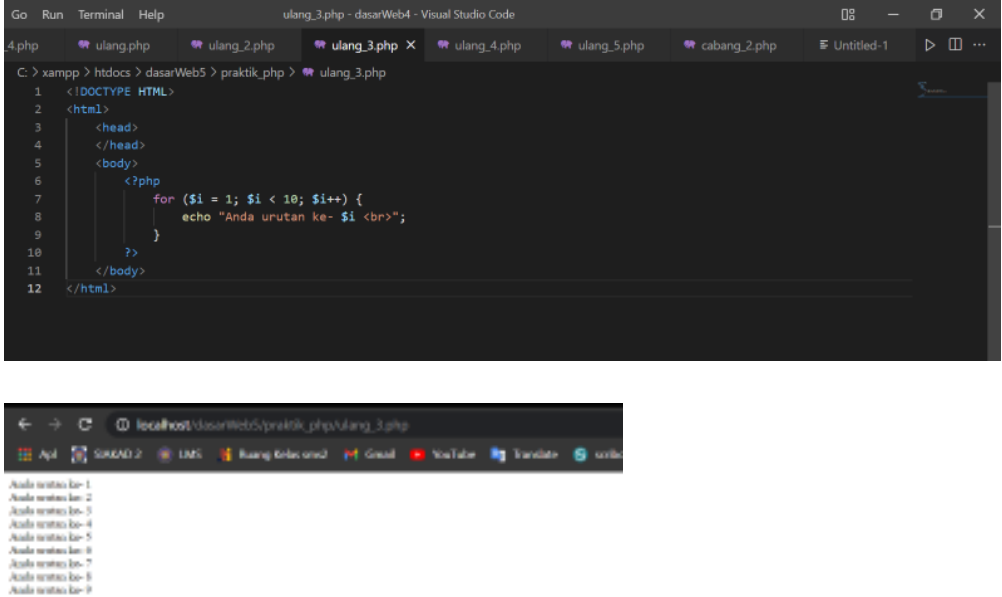
```

<?php
while(kon
disi
TRUE){
statement
;
} ?>

```

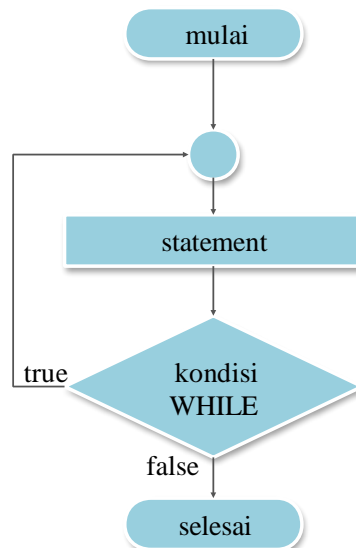
Selanjutnya untuk memahami bagaimana menerapkan perulangan WHILE, ikuti langkahlangkah praktikum berikut:

Langkah	Keterangan
1	<p>Buat file baru dengan nama ulang_3.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 for (\$i = 1; \$i < 10; \$i++){ 8 echo "Anda urutan ke- \$i
"; 9 } 10 ?> 11 </body> 12 </html> </pre>

2	Simpan file dan jalankan kode program tersebut dengan mengetikkan localhost/dasarWeb/praktik_php/ulang_3.php
3	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 7)</p> <p>Hasilnya akan menampilkan urutan ke 1 sampai ke 9</p>  <p>The screenshot shows the Visual Studio Code editor with the file 'ulang_3.php' open. The code is as follows:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 for (\$i = 1; \$i < 10; \$i++) { 8 echo "Anda urutan ke- \$i
"; 9 } 10 <?> 11 </body> 12 </html> </pre> <p>Below the code editor, a browser preview is shown at the URL 'localhost/dasarWeb5/praktik_php/ulang_3.php'. The output of the script is displayed as follows:</p> <pre> Anda urutan ke-1 Anda urutan ke-2 Anda urutan ke-3 Anda urutan ke-4 Anda urutan ke-5 Anda urutan ke-6 Anda urutan ke-7 Anda urutan ke-8 Anda urutan ke-9 </pre>

Praktikum Bagian 8. Perulangan (DO – WHILE)

Perintah DO – WHILE juga dapat digunakan untuk melakukan perulangan. Perbedaannya dengan perulangan WHILE, perintah DO – WHILE mengeksekusi statement setidaknya satu kali sebelum melakukan pemeriksaan apakah memenuhi kondisi WHILE. Diagram alur (flowchart) berikut ini menunjukkan struktur perintah DO – WHILE:



Sintaks perulangan DO – WHILE adalah sebagai berikut:

```

<?php
do{
    statement;
} while(kondisi TRUE); ?>
  
```

Untuk memahami bagaimana menerapkan perulangan DO – WHILE, ikuti langkah-langkah praktikum berikut:

Langkah	Keterangan
1	<p>Buat file baru dengan nama ulang_4.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 \$number = 5; 8 \$factorial = 1; 9 10 do { 11 \$factorial *= \$number; 12 \$number = \$number - 1; 13 } while (\$number > 0); 14 echo \$factorial; 15 ?> 16 </body> 17 </html> </pre>
2	<p>Simpan file dan jalankan kode program tersebut dengan mengetikkan</p>
	<p>localhost/dasarWeb/praktik_php/ulang_4.php</p>

3

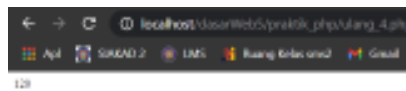
Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 8)

DO – WHILE juga dapat digunakan untuk melakukan perulangan. Perbedaannya dengan perulangan **WHILE**, perintah **DO – WHILE** mengeksekusi statement setidaknya satu kali sebelum melakukan pemeriksaan apakah memenuhi kondisi **WHILE**. Faktorial dari 5 adalah 120

```

1 <!DOCTYPE HTML>
2 <html>
3   <head>
4   </head>
5   <body>
6     <?php
7       $number = 5;
8       $factorial = 1;
9
10      do {
11        $factorial *= $number;
12        $number = $number - 1;
13      } while ($number > 0);
14      echo $factorial;
15    <?>
16  </body>
17 </html>

```



Praktikum Bagian 9. Perulangan Bersarang (Nested Loop)

Nested loop atau perulangan bersarang adalah istilah pemrograman yang berarti membuat perulangan di dalam perulangan. Untuk memahami bagaimana menerapkan perulangan bersarang, ikuti langkah-langkah praktikum berikut:

Langkah	Keterangan
1	<p>Buat file baru dengan nama ulang_5.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> 1 <!DOCTYPE HTML> 2 <html> 3 <head> 4 </head> 5 <body> 6 <?php 7 for(\$i = 0; \$i < 5; \$i++){ 8 for(\$j = 0; \$j < 10; \$j++){ 9 echo "Ini perulangan ke (\$i, \$j)
"; 10 } 11 } 12 <?> 13 </body> 14 </html> </pre>

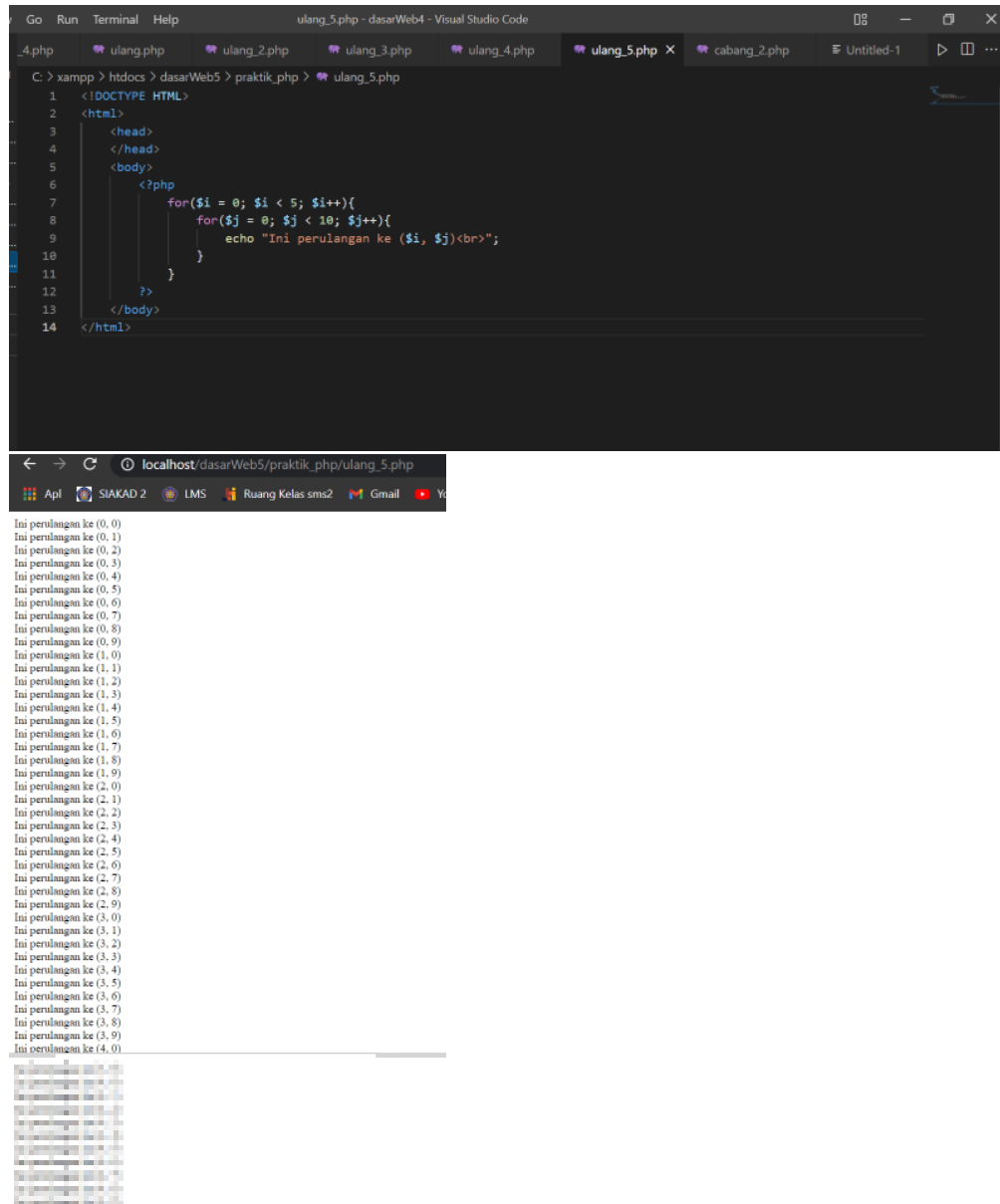
2

Simpan file dan jalankan kode program tersebut dengan mengetikkan
localhost/dasarWeb/praktik_php/ulang_5.php

3

Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 9)

Nested loop adalah perulangan bersarang yang di maksud perulangan yang masih terdapat perulangan lagi. Sehingga outputnya dimulai dari 0 dan berakhir kurang dari 5 jadi 4,9



```
Go Run Terminal Help ulang_5.php - dasarWeb4 - Visual Studio Code
ulang_5.php x cabang_2.php Untitled-1
C: > xampp > htdocs > dasarWeb5 > praktik_php > ulang_5.php
1 <!DOCTYPE HTML>
2 <html>
3 <head>
4 </head>
5 <body>
6 <?php
7     for($i = 0; $i < 5; $i++){
8         for($j = 0; $j < 10; $j++){
9             echo "Ini perulangan ke ($i, $j)<br>";
10        }
11    }
12 >?
13 </body>
14 </html>
```

localhost/dasarWeb5/praktik_php/ulang_5.php

Apl SIKAD 2 LMS Ruang Kelas sms2 Gmail

Ini perulangan ke (0, 0)
Ini perulangan ke (0, 1)
Ini perulangan ke (0, 2)
Ini perulangan ke (0, 3)
Ini perulangan ke (0, 4)
Ini perulangan ke (0, 5)
Ini perulangan ke (0, 6)
Ini perulangan ke (0, 7)
Ini perulangan ke (0, 8)
Ini perulangan ke (0, 9)
Ini perulangan ke (1, 0)
Ini perulangan ke (1, 1)
Ini perulangan ke (1, 2)
Ini perulangan ke (1, 3)
Ini perulangan ke (1, 4)
Ini perulangan ke (1, 5)
Ini perulangan ke (1, 6)
Ini perulangan ke (1, 7)
Ini perulangan ke (1, 8)
Ini perulangan ke (1, 9)
Ini perulangan ke (2, 0)
Ini perulangan ke (2, 1)
Ini perulangan ke (2, 2)
Ini perulangan ke (2, 3)
Ini perulangan ke (2, 4)
Ini perulangan ke (2, 5)
Ini perulangan ke (2, 6)
Ini perulangan ke (2, 7)
Ini perulangan ke (2, 8)
Ini perulangan ke (2, 9)
Ini perulangan ke (3, 0)
Ini perulangan ke (3, 1)
Ini perulangan ke (3, 2)
Ini perulangan ke (3, 3)
Ini perulangan ke (3, 4)
Ini perulangan ke (3, 5)
Ini perulangan ke (3, 6)
Ini perulangan ke (3, 7)
Ini perulangan ke (3, 8)
Ini perulangan ke (3, 9)
Ini perulangan ke (4, 0)