

# **JOB SHEET 6 Percabangan,Perulangan**



**Oleh :**

**Muhammad Brilliant Krisnadi**

**2131710024**

**PRODI D-III MANAJEMEN INFORMATIKA  
JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

**2021/2022**

**Jl. Soekarno Hatta No. 9, Jatimulyo , Kec. Lowokwaru, Kota Malang, Jawa Timur**

## PERTEMUAN 6

### Percabangan dan Perulangan

---

#### A. Tujuan Pembelajaran

Pada pertemuan ini akan dijelaskan penggunaan Percabangan dan Perulangan pada PHP. Setelah selesai materi pada pertemuan ini, mahasiswa mampu memahami konsep php (percabangan dan perulangan), dan mengimplementasikan konsep php ke dalam web dinamis.

#### B. Materi

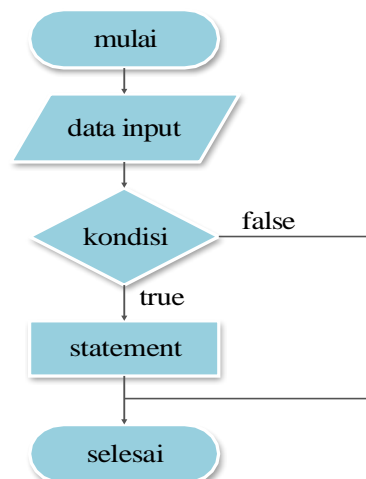
##### **Alur Program**

Di dalam berbagai bahasa pemrograman metode percabangan dan perulangan merupakan metode yang digunakan untuk mengatur dan mengarahkan urutan jalannya suatu program. Percabangan dan perulangan digunakan agar menghasilkan suatu program yang dinamis. Sama seperti bahasa pemrograman lainnya, PHP juga mendukung metode percabangan dan perulangan.

Percabangan digunakan untuk menentukan kode program mana yang akan dieksekusi berdasarkan suatu kondisi. Percabangan diawali dengan pengecekan parameter kondisi untuk melakukan seleksi kode program yang akan di eksekusi sesuai dengan parameter kondisi tersebut. Beberapa fungsi percabangan adalah `if`, `if-else`, `if-elseif-else`, dan `switch-case`. Perulangan digunakan untuk mengeksekusi kode program berulang-ulang berdasarkan suatu kondisi tertentu. PHP menyediakan beberapa keyword untuk melakukan perulangan, yaitu `for`, `foreach`, `while`, dan `do-while`.

##### **Praktikum Bagian 1. Pecabangan (IF)**

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

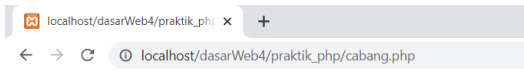


Berdasarkan diagram alur diatas, percabangan dengan menggunakan perintah IF akan mengeksekusi kode program jika kondisi benar. Jika data input sesuai dengan kondisi (bernilai TRUE), maka statement akan dieksekusi. Sintaks perintah IF adalah sebagai berikut:

```
<?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 &lt;!DOCTYPE HTML&gt; 2 &lt;html&gt; 3   &lt;head&gt; 4   &lt;/head&gt; 5   &lt;body&gt; 6     &lt;?php 7       \$nama = "Andi"; 8 9       if(\$nama=="Andi"){ 10        echo "Selamat belajar \$nama"; 11      } 12    ?&gt; 13  &lt;/body&gt; 14 &lt;/html&gt; </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> 
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> <pre> 1 &lt;!DOCTYPE HTML&gt; 2 &lt;html&gt; 3   &lt;head&gt; 4   &lt;/head&gt; 5   &lt;body&gt; 6     &lt;?php 7       \$nama = "Budi"; 8 9       if(\$nama=="Andi"){ 10        echo "Selamat belajar \$nama"; 11      } 12    ?&gt; 13  &lt;/body&gt; 14 &lt;/html&gt; </pre>
5	<p>Ulangi langkah ke-2</p>

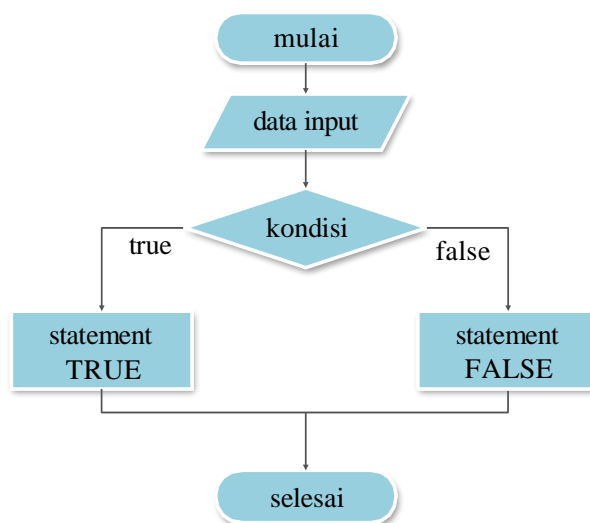


Hasilnya Tidak Ada

6	Amati perbedaan hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 1) Perbedaannya ada di kode bagian if. Jika kode di \$nama dan di bagian if disamakan maka output akan keluar. Namun jika sebaliknya tidak akan keluar
---	--

### 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:

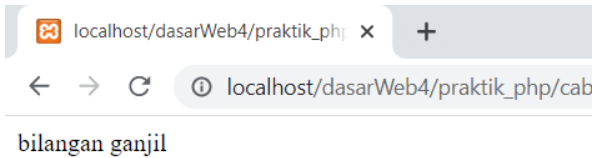
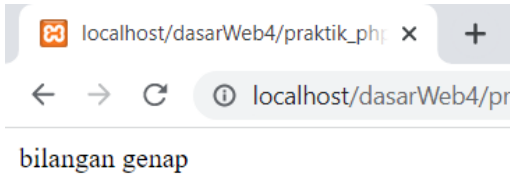
```

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

```

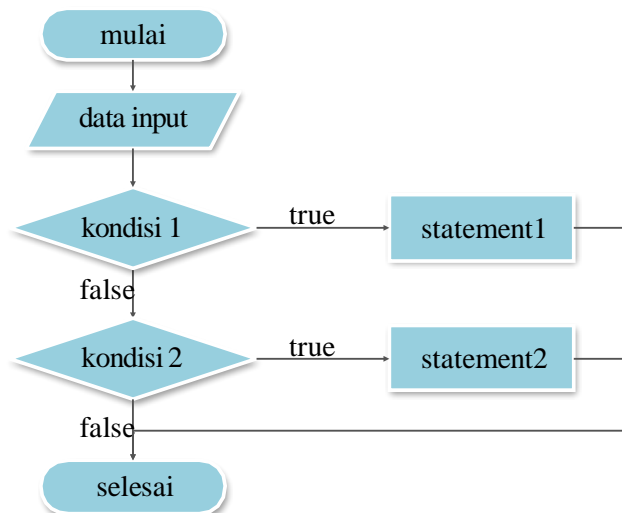
Selanjutnya untuk memahami bagaimana menerapkan percabangan if-else, ikuti langkah-langkah praktikum berikut:

Langkah	Keterangan
1	Buat file baru dengan nama <code>cabang_2.php</code> di dalam direktori <code>praktik_php</code> , kemudian ketikkan kode berikut:

	<pre> 1 &lt;!DOCTYPE HTML&gt; 2 &lt;html&gt; 3 &lt;head&gt; 4 &lt;/head&gt; 5 &lt;body&gt; 6 &lt;?php 7     \$bilangan = 17; 8     if(\$bilangan % 2 == 0){ 9         echo "bilangan genap"; 10    }else{ 11        echo "bilangan ganjil"; 12    } 13 &lt;?&gt; 14 &lt;/body&gt; 15 &lt;/html&gt; </pre>
2	<p>Simpan file dan jalankan kode program tersebut dengan mengetikkan</p> <p>localhost/dasarWeb/praktik_php/cabang_2.php</p>
3	<p>Amati hasil yang ditampilkan</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 &lt;!DOCTYPE HTML&gt; 2 &lt;html&gt; 3 &lt;head&gt; 4 &lt;/head&gt; 5 &lt;body&gt; 6 &lt;?php 7     \$bilangan = 4; 8     if(\$bilangan % 2 == 0){ 9         echo "bilangan genap"; 10    }else{ 11        echo "bilangan ganjil"; 12    } 13 &lt;?&gt; 14 &lt;/body&gt; 15 &lt;/html&gt; </pre>
5	<p>Ulangi langkah ke-2</p> 
6	<p>Amati perbedaan hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 2)</p> <p>Perbedaanya jika pada kode \$bilangan diinputkan bilangan ganjil. Maka outputnya adalah “bilangan ganjil” karena rumusnya dibagi 2. Begitupun sebaliknya pada 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:

```

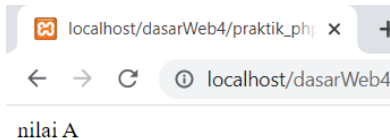
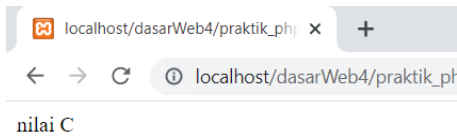
<?php
    if(kondisi1){
        statement1;
    }else if(kondisi2){
        statement2;
    }
?>

```

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

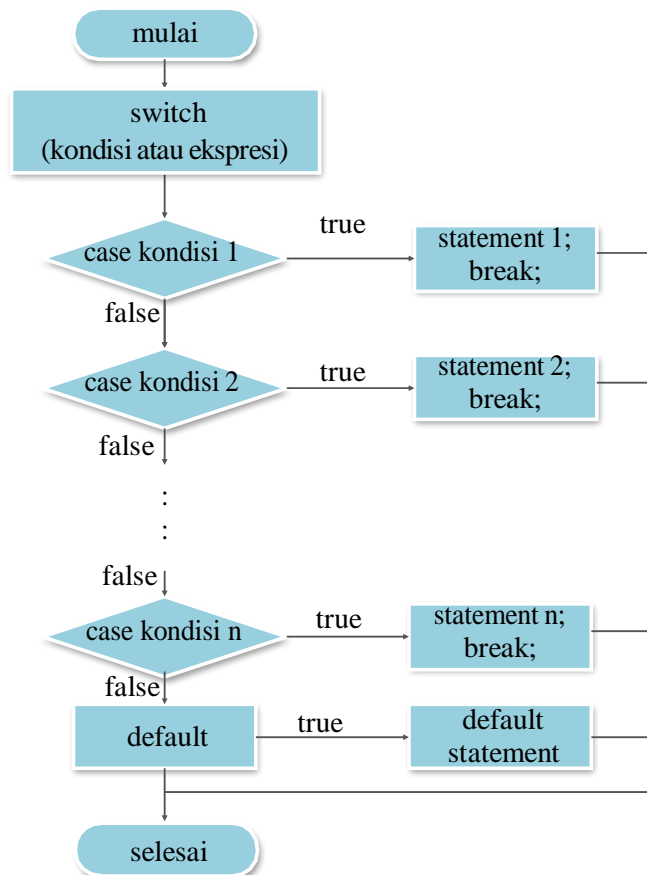
Langkah	Keterangan
1	Buat file baru dengan nama <code>cabang_3.php</code> di dalam direktori <code>praktik_php</code> , kemudian ketikkan kode berikut:



	<pre> 1 &lt;!DOCTYPE HTML&gt; 2 &lt;html&gt; 3 &lt;head&gt; 4 &lt;/head&gt; 5 &lt;body&gt; 6 &lt;?php 7     \$nilai = 85; 8     if(\$nilai &gt;= 80){ 9         echo "nilai A"; 10    }else if(\$nilai &gt;= 70){ 11        echo "nilai B"; 12    }else if(\$nilai &gt;= 60){ 13        echo "nilai C"; 14    }else if(\$nilai &gt;=50){ 15        echo "nilai D"; 16    }else{ 17        echo "nilai E"; 18    } 19 ?&gt; 20 &lt;/body&gt; 21 &lt;/html&gt; </pre>
2	<p>Simpan file dan jalankan kode program tersebut dengan mengetikkan</p> <p>localhost/dasarWeb/praktik_php/cabang_3.php</p>
3	<p>Amati hasil yang ditampilkan</p> 
4	<p>Lakukan modifikasi kode program langkah ke-1 dengan mengganti nilai variabel \$nilai pada baris ke-7 dengan beberapa nilai berikut:</p> <p>a. 66</p> <p>b. 40</p>
5	<p>Ulangi langkah ke-2</p> <p>A. 66</p>  <p>B. 40</p> 
6	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 3)</p> <p>jika variable \$nilai dirubah menjadi 66 maka hasilnya “nilai C” dikarenakan nilainya &gt;=60 tetapi kurang dari 70.</p> <p>jika variable \$nilai dirubah menjadi 40 maka hasilnya “nilai E” dikarenakan nilainya kurang dari semua kondisi.</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;
}
?>

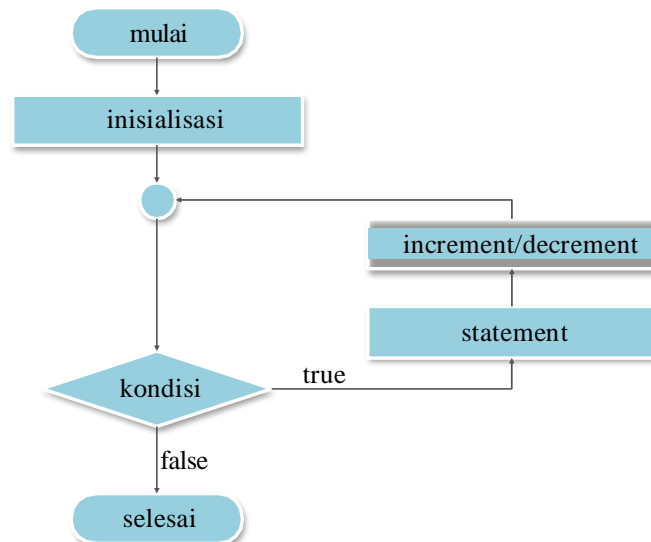
```

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 &lt;!DOCTYPE HTML&gt; 2 &lt;html&gt; 3   &lt;head&gt; 4   &lt;/head&gt; 5   &lt;body&gt; 6     &lt;?php 7       \$tujuan = "Bali"; 8 9       echo "Mau main kemana? " . \$tujuan . "&lt;br&gt;"; 10      echo "&lt;br&gt; Pesan: "; 11 12      switch(\$tujuan){ 13        case "Batu": 14          echo "&lt;b&gt;Jangan lupa bawa jaket&lt;/b&gt;"; 15          break; 16        case "Bandung": 17          echo "&lt;b&gt;Awat kalau belanja laper mata&lt;/b&gt;"; 18          break; 19        case "Bali": 20          echo "&lt;b&gt;Pakai sunblock SPF 50&lt;/b&gt;"; 21          break; 22        default: 23          echo "&lt;b&gt;Ya udah belajar aja&lt;/b&gt;"; 24      } 25    &lt;?&gt; 26  &lt;/body&gt; 27 &lt;/html&gt; </pre>
2	<p>Simpan file dan jalankan kode program tersebut dengan mengetikkan localhost/dasarWeb/praktik_php/cabang_4.php</p>
3	<p>Amati hasil yang ditampilkan</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>  <p>Pesan yg keluar adalah "Ya udah belajar aja" karena kata "Malang" tidak ada pada kondisi</p>

#### **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.



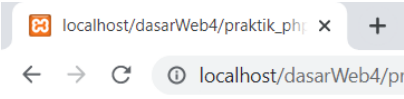
Sintaks perulangan FOR adalah sebagai berikut:

```

<?php
    for(inisialisasi; kondisi; increment/decrement){
        statement;
    }
?>

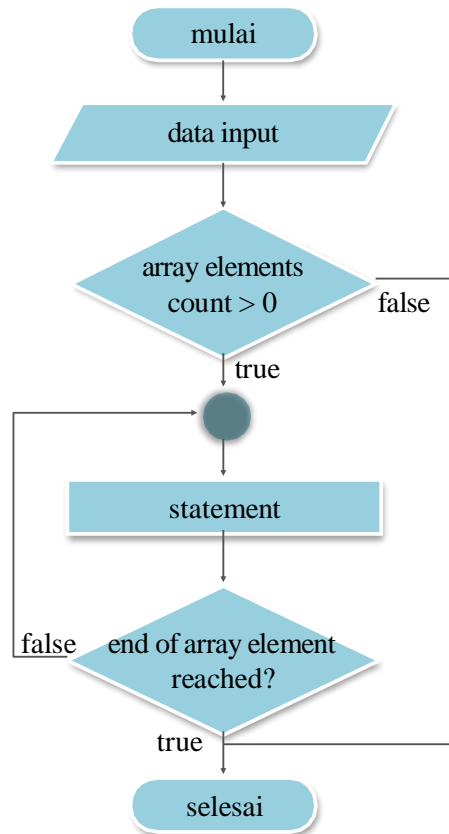
```

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 &lt;!DOCTYPE HTML&gt; 2 &lt;html&gt; 3   &lt;head&gt; 4   &lt;/head&gt; 5   &lt;body&gt; 6 7       &lt;?php 8           for (\$i = 0; \$i &lt; 10; \$i++){ 9               \$hasilKali = 10 * \$i; 10              echo "Hasil perkalian dari 10 x \$i adalah \$hasilKali &lt;br&gt;"; 11          } 12      } 13  &lt;/body&gt; 14  &lt;/html&gt; </pre>
2	<p>Simpan file dan jalankan kode program tersebut dengan mengetikkan <code>localhost/dasarWeb/praktik_php/ulang.php</code></p>
3	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 5)</p>  <pre> Hasil perkalian dari 10 x 0 adalah 0 Hasil perkalian dari 10 x 1 adalah 10 Hasil perkalian dari 10 x 2 adalah 20 Hasil perkalian dari 10 x 3 adalah 30 Hasil perkalian dari 10 x 4 adalah 40 Hasil perkalian dari 10 x 5 adalah 50 Hasil perkalian dari 10 x 6 adalah 60 Hasil perkalian dari 10 x 7 adalah 70 Hasil perkalian dari 10 x 8 adalah 80 Hasil perkalian dari 10 x 9 adalah 90 </pre>

## 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;
    }
?>
```

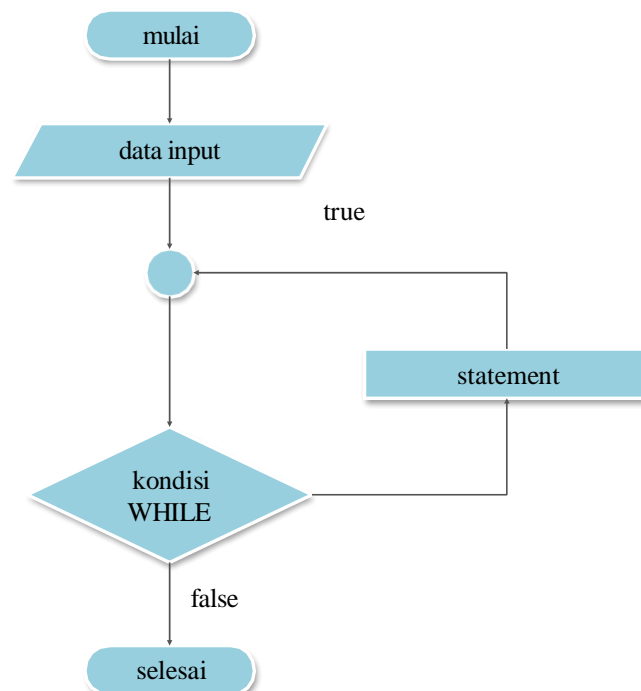
Untuk memahami bagaimana menerapkan perulangan FOR, ikuti langkah-langkah praktikum berikut:

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  &lt;!DOCTYPE html&gt; 2  &lt;html&gt; 3    &lt;head&gt; 4    &lt;/head&gt; 5    &lt;body&gt; 6      &lt;?php 7        \$colors = array("red", "green", "blue", "yellow"); 8 9        foreach (\$colors as \$value) { 10         echo "\$value &lt;br&gt;"; 11        } 12      &lt;?&gt; 13    &lt;/body&gt; 14  &lt;/html&gt; </pre>
2	Simpan file dan jalankan kode program tersebut dengan mengetikkan localhost/dasarWeb/praktik_php/ulang_2.php
3	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 6)</p>  <p>localhost/dasarWeb4/praktik_php x</p> <p>← → ↻ ⓘ localhost/dasarWe</p> <p>red green blue yellow</p> <p>Hasilnya Sesuai Pada perintah Kode foreach</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:

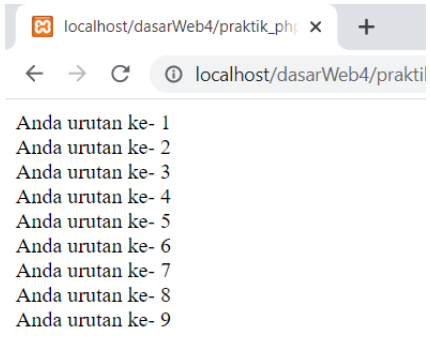




Sintaks perulangan WHILE adalah sebagai berikut:

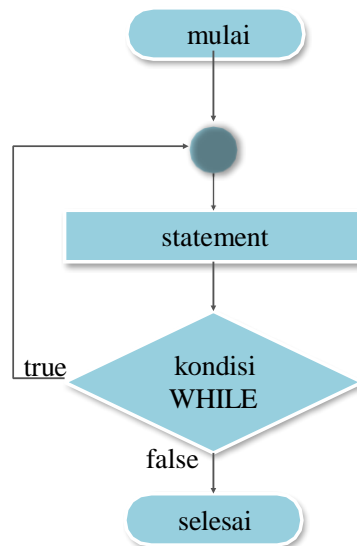
```
<?php
    while(kondisi TRUE) {
        statement;
    }
?>
```

Selanjutnya untuk memahami bagaimana menerapkan perulangan WHILE, ikuti langkah-langkah 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  &lt;!DOCTYPE HTML&gt; 2  &lt;html&gt; 3      &lt;head&gt; 4      &lt;/head&gt; 5      &lt;body&gt; 6          &lt;?php 7              for (\$i = 1; \$i &lt; 10; \$i++){ 8                  echo "Anda urutan ke- \$i &lt;br&gt;"; 9              } 10         ?&gt; 11     &lt;/body&gt; 12 &lt;/html&gt; </pre>
2	<p>Simpan file dan jalankan kode program tersebut dengan mengetikkan localhost/dasarWeb/praktik_php/ulang_3.php</p>
3	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 7)</p>  <p>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</p> <p>Outpunya Sesuai perintah pada kode</p> <pre> for (\$i = 1; \$i &lt; 10; \$i++) {     echo "Anda urutan ke- \$i &lt;br&gt;"; } </pre> <p>Output yg diminta mulai dari 1 dan berakhir di 9. Dikarenakan \$i &lt; 10</p>

### 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:

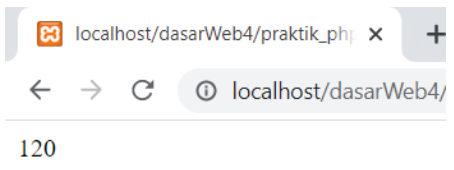
```

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

```

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

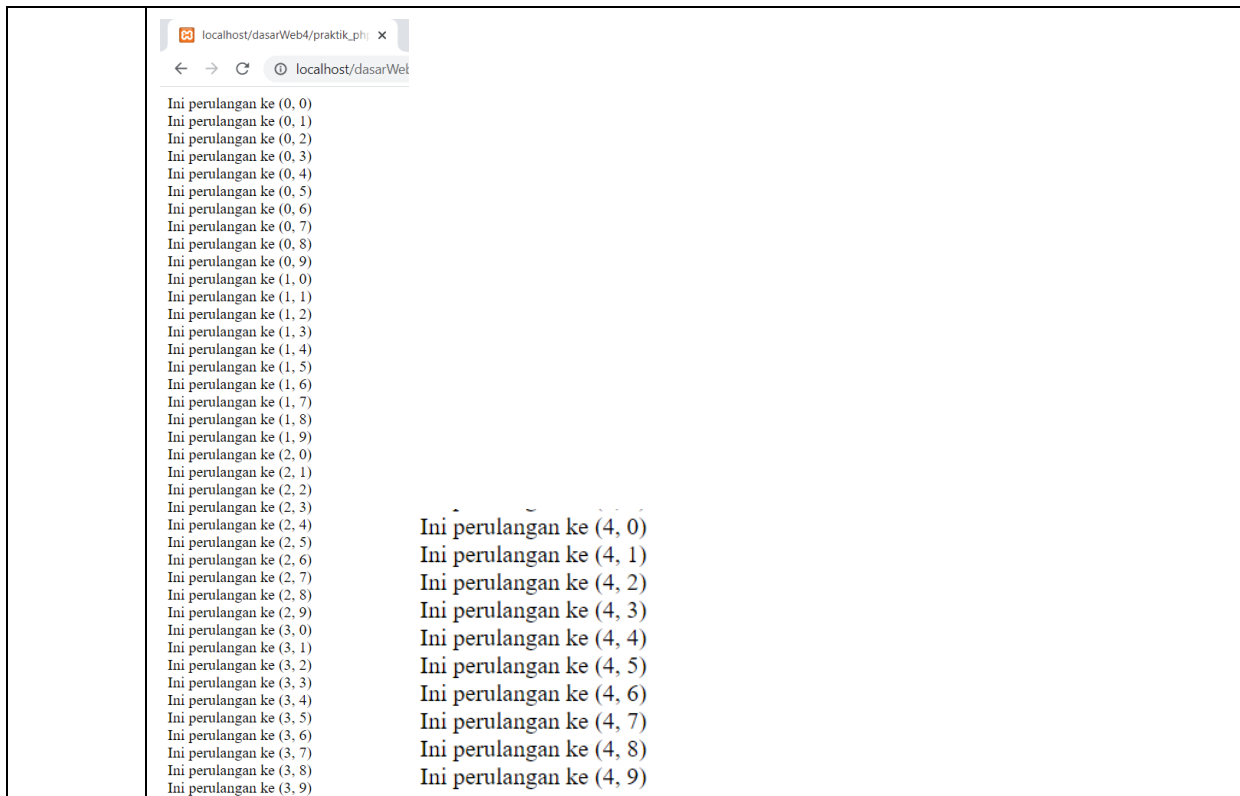
Langkah	Keterangan
1	<p>Buat file baru dengan nama <code>ulang_4.php</code> di dalam direktori <code>praktik_php</code>, kemudian ketikkan kode berikut:</p> <pre> 1  &lt;!DOCTYPE HTML&gt; 2  &lt;html&gt; 3  &lt;head&gt; 4  &lt;/head&gt; 5  &lt;body&gt; 6  &lt;?php 7      \$number = 5; 8      \$factorial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

	localhost/dasarWeb/praktik_php/ulang_4.php
3	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 8)</p> 

### 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 <code>ulang_5.php</code> di dalam direktori <code>praktik_php</code>, kemudian ketikkan kode berikut:</p> <pre> 1  &lt;!DOCTYPE HTML&gt; 2  &lt;html&gt; 3      &lt;head&gt; 4      &lt;/head&gt; 5      &lt;body&gt; 6          &lt;?php 7              for(\$i = 0; \$i &lt; 5; \$i++){ 8                  for(\$j = 0; \$j &lt; 10; \$j++){ 9                      echo "Ini perulangan ke (\$i, \$j)&lt;br&gt;"; 10                     } 11                 } 12             ?&gt; 13         &lt;/body&gt; 14     &lt;/html&gt; </pre>
2	<p>Simpan file dan jalankan kode program tersebut dengan mengetikkan <code>localhost/dasarWeb/praktik_php/ulang_5.php</code></p>
3	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu! (soal no 9)</p>



#### Referensi :

- 1) Nixon, Robin. (2018). Learning PHP, MySQL, JavaScript, CSS & HTML: A Step-by-step Guide to Creating Dynamic Websites, 5th Edition. O'Reilly Media, Inc.
- 2) Forbes, Alan. (2012). The Joy of PHP: A Beginners's Guide to Programming Interactive Web Applications with PHP and MySQL, 5th Edition. Plum Island Publishing.