



# Statemen Kontrol – Kondisional

# Sub CPMK:

Setelah mengikuti modul ini, mahasiswa diharapkan mampu:

- Memahami dan mengimplentasikan statement control IF dan SWITCH

# Statement Kontrol

- Dalam dunia pemrograman umumnya, terdapat 2 jenis statement kontrol yaitu: **statement kontrol kondisional (bersyarat)** dan **statement kontrol perulangan (looping)**.
- Statement kontrol kondisional → statement kontrol yang digunakan untuk mengatur kapan suatu perintah akan dijalankan. Dengan statement ini kita bisa mengatur kapan suatu perintah akan dijalankan, yaitu ketika telah dipenuhinya suatu syarat tertentu.
- Sedangkan statement kontrol perulangan → digunakan untuk mengatur perintah yang dijalankan secara berulang-ulang.
- Dalam PHP, terdapat dua buah statement kontrol yang termasuk statement control kondisional, yaitu IF dan CASE OF. Sedangkan yang termasuk statement control perulangan adalah: FOR, WHILE, DO WHILE dan FOREACH.

# Statement Kontrol - IF

- Misalkan ada sebuah kalimat:

Jika sore ini hujan, maka saya akan menghabiskan waktu di rumah.

- Pernyataan di atas mengandung sebab akibat. Bila Anda lihat, keterangan 'sore ini hujan' adalah merupakan suatu syarat. Apa akibatnya bila syarat tersebut terpenuhi? Ya... 'saya akan menghabiskan waktu di rumah'. Dengan demikian, 'saya akan menghabiskan waktu di rumah' ini hanya akan dilakukan apabila syaratnya terpenuhi atau 'sore ini hujan'.
- Dalam dunia logika, istilah 'terpenuhinya suatu syarat' dapat dikatakan sebagai 'syarat tersebut bernilai BENAR atau TRUE'.
- Bentuk pernyataan 'jika... maka...' pada contoh di atas dapat diadopsi dalam programming. Untuk menyatakan pernyataan tersebut dalam programming, khususnya PHP, kita bisa menggunakan statement kontrol IF

# Statement Kontrol - IF

- Adapun sintaks atau aturan penulisan IF nya adalah sebagai berikut:

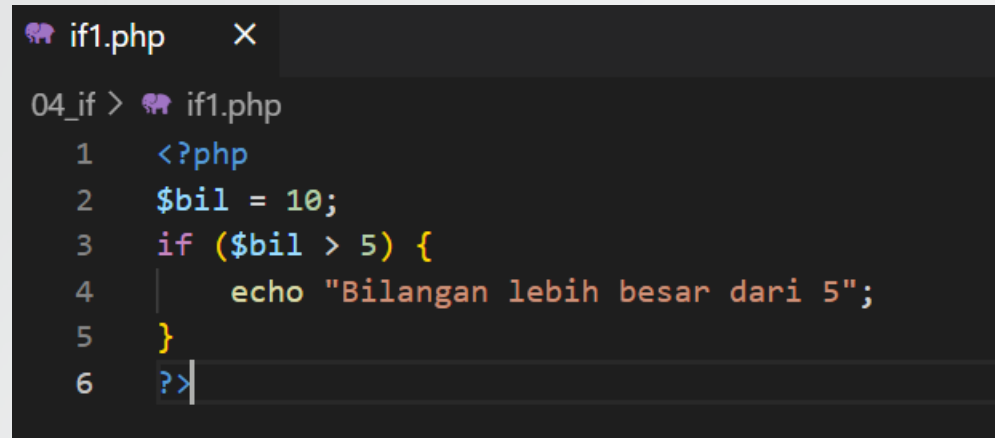
```
if (syarat)
{
    statement;
}
```

- Dalam sintaks di atas, bagian 'statement' akan dijalankan atau dilakukan jika 'syarat' terpenuhi atau 'syarat' bernilai benar/true. Bagaimana bila 'syarat' tidak terpenuhi? Maka bagian 'statement' tidak akan dijalankan atau dalam hal ini tidak melakukan apa-apa.
- Dengan demikian, nilai 'syarat' haruslah hanya ada dua kemungkinan, yaitu BENAR atau SALAH itu saja.

# Operator relational

- Terkadang untuk menyatakan suatu syarat, kita menggunakan operator pembandingan atau relasional. Hasil penggunaan operator relasional ini akan diperoleh nilai BENAR atau SALAH

Simbol	Makna
<	Lebih kecil dari
>	Lebih besar dari
>=	Lebih besar atau sama dengan
<=	Lebih kecil atau sama dengan
==	Sama dengan
!=	Tidak sama dengan



```
if1.php X
04_if > if1.php
1  <?php
2  $bil = 10;
3  if ($bil > 5) {
4      echo "Bilangan lebih besar dari 5";
5  }
6  ?>
```

- Pada script di atas, nilai dari variabel \$bil adalah 10.
- Selanjutnya terdapat statement IF yang di dalamnya terdapat syarat: jika nilai \$bil lebih besar dari 5, maka tampilkan teks 'Bilangan lebih besar dari 5'.
- Dalam hal ini syarat bernilai BENAR, karena  $10 > 5$ . Karena syarat bernilai BENAR, maka perintah echo " Bilangan lebih besar dari 5"; akan dijalankan.

```
if2.php ×
04_if > if2.php
1  <?php
2  $bil = 10;
3  if ($bil == 5) {
4      echo "Bilangan sama dengan 5";
5  }
6  ?>
```

- Pada script di atas, syarat bernilai SALAH, karena 10 tidak sama dengan 5.
- Dengan demikian perintah echo "Bilangan sama dengan 5"; tidak akan dijalankan.



# Operator logika

- Dalam pemrograman PHP, terdapat beberapa operator logika yang bisa digunakan

Opeartor	Makna
&&	AND
	OR
!	NEGASI

- Tabel kebenaran logika

Operand	Hasil (NOT)
Benar	Salah
Salah	Benar

Operand1	Operand2	Hasil (AND)	Hasil (OR)
Benar	Benar	Benar	Benar
Benar	Salah	Salah	Benar
Salah	Benar	Salah	Benar
Salah	Salah	Salah	Salah

04\_if &gt; logika1.php

```
1  <?php
2  $hobi = "makan";
3  if (($hobi == "makan") || ($hobi == "ngemil")){
4      echo "Hati-hati, jangan terlalu banyak. <br />";
5      echo "Jangan lupa olahraga ..";
6  }
7  ?>
```

- Pada script di atas, nilai \$hobi adalah "makan", sedangkan untuk syarat terdapat dua keadaan yaitu \$hobi == "makan" serta \$hobi == "ngemil".
- Diantara kedua keadaan tersebut terdapat operator OR. Keadaan pertama dalam hal ini bernilai BENAR sedangkan keadaan kedua bernilai SALAH.
- Sehingga apabila kita lihat tabel kebenaran, BENAR OR SALAH dihasilkan BENAR. Dengan demikian syarat bernilai BENAR dan akibatnya perintah echo "Hati-hati,, jangan terlalu banyak"; akan dijalankan.

## Statement Kontrol – IF (2)

- Selain bentuk IF di atas, terdapat pula bentuk IF seperti di bawah ini

```
if (syarat)
{
    statement 1
}
else
{
    statement 2
}
```

- Bentuk if IF di atas, statement 1 akan dijalankan apabila nilai 'syarat' adalah BENAR. Sedangkan bila 'syarat' bernilai SALAH maka yang dijalankan adalah statement 2.

```
logika2.php X
04_if > logika2.php
1  <?php
2  $bil = 10;
3  if ($bil == 10) {
4      echo "Bilangan sama dengan 10";
5  }
6  else {
7      echo "Bilangan tidak sama dengan 10";
8  }
9  ?>
```

- Script di atas akan menampilkan "Bilangan sama dengan 10", karena syarat terpenuhi

logika3.php X

04\_if > logika3.php

```
1  <?php
2  $bil = 12;
3  if ($bil == 10) {
4      echo "Bilangan sama dengan 10";
5  }
6  else {
7      echo "Bilangan tidak sama dengan 10";
8  }
9  ?>
```

- Script di atas akan menampilkan "Bilangan tidak sama dengan 10", karena syarat tidak terpenuhi (bil tidak sama dengan 10)

# Statement Kontrol – IF (3)

- Selain bentuk kedua IF di atas, terdapat pula bentuk IF seperti berikut:
- Pada bentuk IF di atas, Statement 1 akan dijalankan apabila 'syarat 1' bernilai BENAR.
- Jika 'syarat 1' bernilai SALAH, maka akan dicek 'syarat 2'. Jika 'syarat 2' BENAR maka akan dijalankan Statement 2, begitu seterusnya.
- Dan apabila ternyata tidak ada satupun syarat yang terpenuhi, barulah Statement X akan dikerjakan.

```
if (syarat 1)
{
    Statement 1;
}
else if (syarat 2)
{
    Statement 2
}
else if (syarat 3)
{
    Statement 3
}
.
.
else {
    Statement X
}
```

```
logika4.php X
04_if > logika4.php
1  <?php
2  $nilai = 70;
3  if ($nilai > 50) {
4      echo "Nilai lebih besar dari 50";
5  }
6      else if ($nilai < 50) {
7          echo "Nilai lebih kecil dari 50";
8      }
9      else {
10         echo "Nilai sama dengan 50";
11     }
12  ?>
```

- Script di atas akan menghasilkan "Nilai lebih besar dari 50" karena syarat '\$nilai > 50' bernilai BENAR. Begitu terdapat syarat yang bernilai benar, maka syarat yang terletak di bawahnya tidak akan dicek.

```
logika5.php X
04_if > logika5.php
1  <?php
2  $nilai = 50;
3  if ($nilai > 50)
4  {
5      echo "Nilai lebih besar dari 50";
6  }
7  else
8      if ($nilai < 50) {
9          echo "Nilai lebih kecil dari 50";
10     }
11  ?>
```

- Pada script di atas, syarat ' $\$nilai > 50$ ' bernilai SALAH. Karena syarat ini bernilai SALAH, maka selanjutnya akan dicek syarat ' $\$nilai < 50$ '. Syarat inipun juga SALAH.
- Dengan demikian hasil script di atas tidak akan muncul apa-apa, karena dalam hal ini tidak terdapat alternatif terakhir 'else'



```
logika6.php X
04_if > logika6.php
1  <?php
2  $nilai = 50;
3  if ($nilai > 50)
4  {
5      echo "Nilai lebih besar dari 50";
6  }
7  else
8      if ($nilai < 50)
9      {
10         echo "Nilai lebih kecil dari 50";
11     }
12     else
13     {
14         echo "Nilai sama dengan 50";
15     }
16  ?>
```

- Pada script di atas, hasil dari script adalah "Nilai sama dengan 50".

# Contoh-contoh script dengan IF

- Script untuk menentukan apakah suatu huruf termasuk huruf vokal (asumsi: huruf yang digunakan adalah kapital)

logika\_huruf.php X

04\_if > logika\_huruf.php

```
1  <?php
2  $huruf = "A";
3  if (($huruf == "A") || ($huruf == "I") || ($huruf == "U") || ($huruf == "E") || ($huruf == "O")) {
4      echo "Termasuk huruf vokal";
5  }
6  else {
7      echo "Tidak termasuk huruf vokal";
8  }
9  ?>
```

# Contoh-contoh script dengan IF

- Script untuk mencari bilangan terbesar dari 3 buah bilangan: 6, 9 dan 10.
- Langkahnya:
  1. Kita cari terlebih dahulu bilangan terbesar di antara bilangan pertama dan kedua (misalkan dinamakan 'max') Untuk mencari 'max' caranya dengan membandingkan bilangan pertama dengan kedua. Jika bilangan pertama > bilangan kedua, maka nilai 'max' adalah bilangan pertama. Jika tidak, maka 'max' adalah bilangan kedua.
  2. Setelah kita tahu bilangan terbesar di antara bilangan pertama dan kedua ('max'), langkah berikutnya kita cari bilangan terbesar di antara 'max' dengan bilangan ketiga. Nilai terbesar dari ketiga bilangan adalah nilai yang terbesar di antara bilangan ketiga dengan 'max'. Untuk mencari bilangan terbesar antara 'max' dengan bilangan ketiga, caranya sama seperti langkah pertama.

# Contoh-contoh script dengan IF

- Script untuk mencari bilangan terbesar dari 3 buah bilangan: 6, 9 dan 10.

```
logika_max.php X
04_if > logika_max.php
1  <?php
2  $bil1 = 6;
3  $bil2 = 9;
4  $bil3 = 10;
5  if ($bil1 > $bil2) {
6      $max = $bil1;
7  }
8  else {
9      $max = $bil2;
10 }
11 if ($max > $bil3) {
12     $maxSemua = $max;
13 }
14 else {
15     $maxSemua = $bil3;
16 }
17 echo "Nilai terbesar dari ketiga bilangan adalah ".$maxSemua;
18 ?>
```

# Contoh-contoh script dengan IF

- Script untuk mengilustrasikan proses login
  - Script berikut ini akan mengilustrasikan proses login. Sebagai contoh, misalkan user yang bisa login hanyalah user bernama 'joko' (password: 'passjoko') serta user bernama 'amir' (password: 'passamir').
  - Konsep login pada prinsipnya pengecekan nama user dan password saja. Jika nama user dan passwordnya cocok maka login sukses.
  - Pertama kita buat terlebih dahulu form loginnya, selanjutnya kita buat script login.php

# Contoh-contoh script dengan IF

- Script untuk mengilustrasikan proses login

login.php X

04\_if > login.php

```
1  <?php
2  $username = $_POST['username'];
3  $password = $_POST['pass'];
4  if (($username == "joko") && ($password == "passjoko")) ||
5     (($username == "amir") && ($password == "passamir")) {
6     echo "Login sukses";
7  }
8  else {
9     echo "Login gagal";
10 }
11 ?>
```

input.html X

04\_if > input.html > form

```
1  <form method="post" action="login.php">
2      Username <input type="text" name="username"> <br />
3      Password <input type="password" name="pass"> <br />
4      <input type="submit" name="submit" value="Submit">
5  </form>
```

# Statement Kontrol - Switch

- SWITCH termasuk statement yang dapat dipergunakan untuk menyatakan suatu pernyataan kondisional atau bersyarat seperti halnya IF

```
switch (var)
{
    case value1 : statement A
        .
        .
        break;
    case value2 : statement B
        .
        .
        break;
    case value3 : statement C
        .
        .
        break;
    .
    .
    default : statement D
        .
        .
}
```

# Statement Kontrol - Switch

- Maksud dari sintaks tersebut adalah: dalam hal ini nilai yang ada pada variabel ('var') akan dicek. Diantara value – value yang ada sebagai pilihan, dalam hal ini value1, value2, value3, ... dst manakah yang sesuai dengan value yang dimiliki oleh 'var'.
- Misalkan nilai var = value1, maka blok yang ada di statement A lah yang dijalankan. Bila nilai var = value2, maka blok statement B yang akan dijalankan.
- Namun bila tidak ada value yang cocok, maka barulah blok yang ada dibagian DEFAULT (statement D) yang akan dijalankan.

```
switch (var)
{
case value1 : statement A
    .
    .
    break;
case value2 : statement B
    .
    .
    break;
case value3 : statement C
    .
    .
    break;
.
.
default : statement D
    .
    .
}
```



# Statement Kontrol - Switch

- Bentuk sintaks SWITCH di atas ekuivalen dengan statement IF berikut ini
- Apabila kita perhatikan dari konsep SWITCH, maka kita bisa menyimpulkan bahwa proses pengecekan atau pengujian persyaratan yang terjadi hanyalah proses perbandingan dalam bentuk 'sama dengan' atau '=='.
- SWITCH tidak bisa digunakan untuk kondisional yang di dalamnya terdapat proses perbandingan seperti 'lebih besar dari', 'lebih kecil dari', 'lebih kecil atau sama dengan', 'lebih besar atau sama dengan' dan 'tidak sama dengan'. Melihat hal ini memang penggunaan statement IF untuk menyatakan kondisional lebih fleksibel daripada SWITCH.

```
if (var == value1) {  
    statement A  
.  
}  
else if (var == value2) {  
    statement B  
.  
}  
else if (var == value3) {  
    statement C  
.  
}  
.  
Else {  
    statement D  
.  
}
```

# Contoh-contoh script dengan SWITCH

- Script untuk menentukan bulan sekarang (fungsi date("n") untuk mengambil angka bulan sekarang)

```
switch01.php X
05_switch > switch01.php
1  <?php
2  $angkaBln = date("n");
3  switch($angkaBln)
4  {
5      case 1 : $namaBln = "Januari";
6          break;
7      case 2 : $namaBln = "Pebruari";
8          break;
9      case 3 : $namaBln = "Maret";
10         break;
11     case 4 : $namaBln = "April";
12         break;
13     case 5 : $namaBln = "Mei";
14         break;
15     case 6 : $namaBln = "Juni";
16         break;
```

# Contoh-contoh script dengan SWITCH

- Script untuk menentukan bulan sekarang (fungsi date("n") untuk mengambil angka bulan sekarang)

```
switch01.php X
05_switch > switch01.php
16         break;
17     case 7 : $namaBln = "Juli";
18         break;
19     case 8 : $namaBln = "Agustus";
20         break;
21     case 9 : $namaBln = "September";
22         break;
23     case 10: $namaBln = "Oktober";
24         break;
25     case 11: $namaBln = "Nopember";
26         break;
27     case 12: $namaBln = "Desember";
28         break;
29 }
30 echo "Nama bulan sekarang adalah : ".$namaBln;
31 ?>
```



# Contoh studi kasus

script7-1.php X

04\_if > script7-1.php

```
1  <html>
2  <head>
3  |   <title>Mencari Bilangan Terbesar dari 3 Bilangan</title>
4  </head>
5  <body>
6  <h1>Mencari Bilangan Terbesar dari 3 Bilangan</h1>
7
8  <form method="post" action="script7-1proses.php">
9  <table>
10 |   <tr><td>Masukkan bilangan ke-1</td><td>:</td><td><input type="text" name="bil1"></td></tr>
11 |   <tr><td>Masukkan bilangan ke-2</td><td>:</td><td><input type="text" name="bil2"></td></tr>
12 |   <tr><td>Masukkan bilangan ke-3</td><td>:</td><td><input type="text" name="bil3"></td></tr>
13 </table>
14 <input type="submit" name="submit" value="Submit"><input type="reset" name="reset" value="Hapus">
15 </form>
16
17 </body>
18 </html>
```

script7-1.php X script7-1proses.php ●

04\_if > script7-1proses.php

```
1  <html>
2  <head>
3  |   <title>Mencari Bilangan Terbesar dari 3 Bilangan</title>
4  </head>
5  <body>
6  <h1>Mencari Bilangan Terbesar dari 3 Bilangan</h1>
7
8  <?php
9
10 $bilangan1 = $_POST['bil1']; // membaca input bilangan pertama
11 $bilangan2 = $_POST['bil2']; // membaca input bilangan kedua
12 $bilangan3 = $_POST['bil3']; // membaca input bilangan ketiga
13
14 // cara ke - 1
15 // idenya adalah mencari bilangan x yang terbesar dari bil 1 dan bil2. lalu x dibandingkan dengan
16 // bilangan ke-3. Jika x > dari bil 3, maka x adalah max dari semua bilangan. Jika tidak, maka bil 3 sbg max nya
17
18 if ($bilangan1 > $bilangan2)
19 |   $maxSementara = $bilangan1;
20 else
21 |   $maxSementara = $bilangan2;
22
```

script7-1.php × script7-1proses.php ●

04\_if > script7-1proses.php

```
23  if ($bilangan3 > $maxSementara)
24      $maxAkhir = $bilangan3;
25  else
26      $maxAkhir = $maxSementara;
27
28  echo "<p>Nilai maksimum dari ".$bilangan1.", ".$bilangan2.", dan ".$bilangan3." adalah : ".$maxAkhir."</p>";
29
30  // cara ke - 2
31  // idenya adalah langsung membandingkan masing-masing bilangan dengan bilangan yang lain.
32  // misal: jika bil 1 > bil2 dan bil 1 > bil 3, maka max adalah bil 1.
33
34  if (($bilangan1 > $bilangan2) && ($bilangan1 > $bilangan3))
35      $maxAkhir = $bilangan1;
36  else
37      if (($bilangan2 > $bilangan1) && ($bilangan2 > $bilangan3))
38          $maxAkhir = $bilangan2;
39      else
40          if (($bilangan3 > $bilangan1) && ($bilangan3 > $bilangan2))
41              $maxAkhir = $bilangan3;
42
43  echo "<p>Nilai maksimum dari ".$bilangan1.", ".$bilangan2.", dan ".$bilangan3." adalah : ".$maxAkhir."</p>";
44
```

script7-1.php

script7-1proses.php

04\_if > script7-1proses.php

```
45 // cara ke - 3
46 // idenya hampir sama dengan cara ke- 1, tapi tidak dengan mencari max sementara.
47
48 if ($bilangan1 > $bilangan2)
49 {
50     if ($bilangan1 > $bilangan3)
51     |     $maxAkhir = $bilangan1;
52     else
53     |     $maxAkhir = $bilangan3;
54 }
55 else
56 {
57     if ($bilangan2 > $bilangan3)
58     |     $maxAkhir = $bilangan2;
59     else
60     |     $maxAkhir = $bilangan3;
61 }
62
63 echo "<p>Nilai maksimum dari ".$bilangan1.", ".$bilangan2.", dan ".$bilangan3." adalah : ".$maxAkhir."</p>";
64 ?>
65 </body>
66 </html>
```



script7-2.php X

script7-1proses.php ●

04\_if > script7-2.php

```
1  <html>
2  <head>
3  |   <title>Mencari Status Bilangan</title>
4  </head>
5  <body>
6  <h1>Mencari Status Bilangan</h1>
7
8  <form method="post" action="script7-2proses.php">
9  Masukkan sebuah bilangan <input type="text" name="bil">
10 <input type="submit" name="submit" value="Submit">
11 </form>
12
13 </body>
14 </html>
```

script7-1.php

script7-2proses.php X

04\_if > script7-2proses.php

```
1  <html>
2  <head>
3  |  <title>Mencari Status Bilangan</title>
4  </head>
5  <body>
6  <h1>Mencari Status Bilangan</h1>
7
8  <?php
9
10 $bilangan = $_POST['bil'];
11
12 // cara ke - 1
13
14 if ($bilangan > 0)
15 |     echo "<p>Bilangan ".$bilangan." adalah positif</p>";
16 else
17 |     if ($bilangan < 0)
18 |         echo "<p>Bilangan ".$bilangan." adalah negatif</p>";
19 |     else
20 |         echo "<p>Bilangan ".$bilangan." adalah nol</p>";
21
```

script7-1.php

script7-2proses.php X

04\_if > script7-2proses.php

```
21 |
22 | // cara ke - 2
23 |
24 | if ($bilangan > 0)
25 |     $status = "positif";
26 | else
27 |     if ($bilangan < 0)
28 |         $status = "negatif";
29 |     else
30 |         $status = "nol";
31 |
32 | echo "<p>Bilangan ".$bilangan." adalah bilangan ".$status."</p>";
33 |
34 | ?>
35 |
36 | </body>
37 | </html>
```