

Tgl kumpul	
Nilai	
Ttd aslab	

LAPORAN PRAKTIKUM PEMROGRAMAN WEB
MATERI KE – 3 : Cascading Style Sheet (CSS)& JavaScript



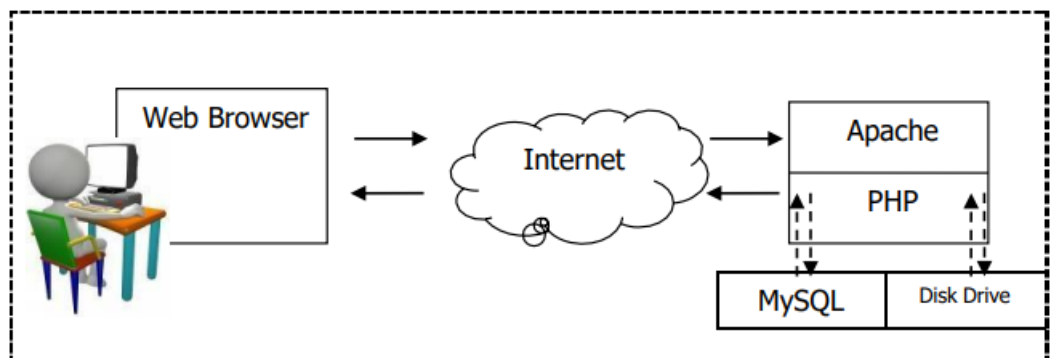
OLEH:
Muhammad Abdul Ghani
200605110185
KELAS J

TEKNIK INFORMATIKA
UIN MAULANA MALIK IBRAHIM MALANG
2023

1. Dasar Teori

a) PHP

PHP adalah salah satu bahasa pemrograman yang dirancang untuk membangun aplikasi web dinamis. PHP juga disebut sebagai server-side programming/scripting yang berarti script/program yang berjalan di sisi server. Kode PHP di-parse di web server oleh interpreter dan diterjemahkan ke dalam bentuk HTML (Hypertext Markup Language) dan pada akhirnya dikirimkan dan ditampilkan pada browser yang dibuka oleh user. Gambar 4.1 memberikan gambaran singkat alur kerja PHP dalam web server.



Sintaks kode PHP biasanya diletakkan di dalam HTML, walaupun bisa juga dibuat dokumen dengan PHP murni. Kode PHP diawali dengan tag pembuka PHP “<?php” dan diakhiri dengan tag penutup PHP “?” atau bisa juga dengan bentuk yang lebih singkat yaitu dengan awalan “<?” dan diakhiri “?””. Kode PHP harus disimpan di dalam file berekstensi .php. Jika tidak, maka kode PHP tidak akan dieksekusi oleh web server walaupun ditulis di antara tag pembuka dan penutup PHP. Berikut satu contoh kode PHP sederhana.

```
1 <?php
2     echo "Hello World";
3     echo 10000;
```

Perintah echo berfungsi untuk mencetak nilai baik teks maupun numerik ke browser. Keywordkeyword perintah PHP seperti echo dan juga tag pembuka dan penutup PHP

(<?php dan ?>) tidak akan ikut tercetak di browser. Selain echo, kita juga bisa menggunakan perintah print:

```
1 <?php
2 print "Hello World";
```

Jika suatu file PHP hanya berisi kode PHP saja tanpa HTML atau kode PHP terletak di akhir file, maka tag penutup PHP (?>) tidak perlu ditulis. Contoh:

```
1 <?php
2     print "Hello World";
3     <!-- kode HTML -->
4 </html>
5
6 <?php
7     print "Hello World";
8     // akhir file
```

Setiap perintah atau statemen di dalam kode PHP harus diakhiri dengan tanda titik koma (;).

Ada beberapa cara untuk meletakkan kode PHP dalam sebuah dokumen, yaitu dengan menyisipkan PHP dalam HTML atau HTML dalam PHP. Berikut contoh PHP dalam HTML.

```
1 <html>
2 <head>
3     <title>PHP dalam HTML</title>
4 </head>
5 <body>
6     <?php echo "<p>Teks dari kode PHP</p>"; ?>
7 </body>
8 </html>
```

Berikut contoh HTML dalam PHP.

```
1 <?php
2     echo "<html>";
3     echo "<head>";
4     echo "<title>HTML dalam PHP</title>";
5     echo "</head>";
6     echo "<body>";
7     echo "<p>Paragraf 2: Teks dari kode PHP</p>";
8     echo "</body>";
```

```
9     echo "</html>";
```

b) Tipe Data, Variabel, dan Operator

PHP mendukung delapan tipe data. Empat nilai skalar (single-value): integer, float, string, dan boolean; nilai koleksi: array. Sisanya adalah tipe khusus yaitu resource dan null.

Integer	<i>Integer</i> adalah tipe data yang merepresentasikan bilangan utuh atau bulat, positif maupun negatif dan bukan pecahan. Rentang dari bilangan integer adalah - 2.147.483.648 sampai 2.147.483.647.
Float	<i>Float</i> disebut sebagai bilangan pecahan atau sering juga disebut sebagai bilangan <i>real</i> yang mengandung angka desimal di belakang koma. Contohnya adalah 3,14; 2,54; dan 9,99.
String	PHP mendukung tipe data <i>string</i> yaitu tipe data karakter yang biasanya diekspresikan dengan diapit oleh petik ganda atau petik tunggal. Contoh: <code>'ini adalah string' atau "ini adalah string"</code>
Boolean	Tipe <i>boolean</i> adalah tipe data yang merepresentasikan nilai kebenaran (<i>truth value</i>). Nilai yang mungkin pada tipe <i>boolean</i> adalah <i>true</i> dan <i>false</i> . Penulisan nilai <i>true</i> dan <i>false</i> bersifat <i>case-insensitive</i> sehingga <code>"true"</code> , <code>"True"</code> , dan <code>"TRUE"</code> adalah sama dan <code>"false"</code> , <code>"False"</code> , dan <code>"FALSE"</code> adalah sama. Contoh: <div style="border: 1px solid black; padding: 10px; margin: 10px 0;"> <pre><?php \$found = true; if (\$found) { echo "Data telah ditemukan"; }</pre> </div> <p>Maksud dari kode diatas adalah jika variabel <code>\$found</code> bernilai <i>true</i>, maka dilakukan kode di dalam blok <code>if</code> akan dieksekusi.</p>
Array	<i>Array</i> adalah variabel yang menyimpan sekelompok nilai yang dapat diidentifikasi berdasarkan posisinya atau indeksnya. Contoh: <div style="border: 1px solid black; padding: 10px; margin: 10px 0;"> <pre>\$person = array('Dadang', 'Jajang', 'Udin', 'Rendra');</pre> </div> <p>Penulisan <i>array</i> juga dapat dilakukan dengan sintaks yang lebih ringkas. Contoh:</p> <div style="border: 1px solid black; padding: 10px; margin: 10px 0;"> <pre>\$person = ['Dadang', 'Jajang', 'Udin', 'Rendra'];</pre> </div>

Pendefinisian Variabel

Variabel di PHP ditulis dengan awalan karakter dolar (\$).

Contoh:

1	<code>\$username = "Hendro";</code>
2	<code>\$usia = 25;</code>

PHP tidak seperti bahasa pemrograman lain yang mengharuskan kita untuk memberikan atau menuliskan tipe variabel. Artinya PHP mempunyai kemampuan untuk mendeteksi secara otomatis apa tipe variabel yang kita tulis.

Operator

Dalam PHP operator dibagi menjadi 4 yaitu operator assignment, operator aritmetika, operator relasional, dan operator logika. Operator assignment adalah operator yang digunakan untuk memberikan sebuah nilai yang akan

dimasukkan ke dalam variabel. Operator assignment ditulis dengan karakter sama dengan (“=”). Operator aritmetika digunakan untuk operasi aritmetika dengan daftar berikut.

Operator	Keterangan
+	Operator penjumlahan
-	Operator pengurangan
*	Operator perkalian
/	Operator pembagian
%	Operator modulus (sisanya dari pembagian)
++	Operator untuk menaikkan nilai variabel sebesar 1
--	Operator untuk menurunkan nilai variabel sebesar 1

Operator relasional adalah operator yang digunakan untuk membandingkan 2 buah nilai. Hasil operasi hanya ada dua yaitu true dan false. Berikut adalah contohnya.

Operator	Keterangan
\$a == \$b	Memeriksa apakah \$a sama dengan \$b
\$a != \$b	Memeriksa apakah \$a tidak sama dengan \$b
\$a < \$b	Memeriksa apakah \$a kurang dari \$b
\$a > \$b	Memeriksa apakah \$a lebih besar dari \$b
\$a <= \$b	Memeriksa apakah \$a kurang dari sama dengan \$b
\$a >= \$b	Memeriksa apakah \$a lebih dari sama dengan \$b

Sedangkan operator logika adalah operator yang digunakan untuk mengoperasikan dua buah operand yang bertipe boolean. Contoh dari operator logika adalah sebagai berikut.

Operator	Keterangan
&&	Logika AND
	Logika OR
!	Logika NOT

- c) Alur adalah sintaks yang didefinisikan untuk mengatur alur program. Pembahasan ini dibagi menjadi dua, yaitu :
- Kondisi
 - Perulangan

Kondisi

Kondisi digunakan untuk menentukan kode mana yang akan dijalankan. Biasanya kondisi diawali dengan pengecekan beberapa parameter untuk diseleksi. Pada kondisi terdapat beberapa sintak yang bisa kita gunakan.

Yang pertama adalah kondisi if. Berikut adalah contoh kodenya.

```
1  if (kondisi) {  
2      statemen yang dieksekusi  
3  }
```

Jika kita tidak menemukan hasil pada seleksi pertama maka kita bisa menambahkan else seperti contoh berikut.

```
1  if (kondisi) {  
2      statemen yang dieksekusi  
3  } else {  
4      statemen lainnya  
5  }
```

Atau kita bisa gunakan model seperti di bawah ini.

```
1  if (kondisi1){  
2      // statemen untuk kondisi 1;  
3  } elseif (kondisi2){  
4      // statemen untuk kondisi 2;  
5  } elseif (kondisi3){  
6      // statemen untuk kondisi 3;  
7  }
```

Selain if-else, kita juga bisa menggunakan kondisional switch, kondisional ini digunakan Ketika kita membutuhkan banyak kondisi. Contohnya adalah sebagai berikut.

```
1  <?php  
2      switch ($bulan){  
3          case 1:  
4              echo "Januari";  
5              break;  
6          case 2:  
7              echo "Februari";  
8              break;  
9          case 3:  
10             echo "Maret";  
11             break;  
12          case 4:  
13             echo "April";  
14             break;  
15          case 5:  
16             echo "Mei";  
17             break;  
18          case 6:  
19             echo "Juni";  
20             break;  
21          case 7:  
22             echo "Juli";  
23             break;  
24          case 8:  
25             echo "Agustus";  
26             break;
```

```

27     case 9:
28         echo "September";
29         break;
30     case 10:
31         echo "Oktober";
32         break;
33     case 11:
34         echo "November";
35         break;
36     case 12:
37         echo "Desember";
38         break;
39     default:
40         echo "Ini default statemen";
41 }
42 ?>

```

Perulangan

Dalam pembuatan sebuah aplikasi web, seringkali kita harus melakukan pengulangan terhadap statemen-statemen tertentu. PHP menyediakan kode yang dapat kita gunakan untuk mengulang statemen, antara lain while, do-while, for, dan foreach. Berikut sintaks masing-masing kode.

While

```

1 while(ekspresi) {
2     Statemen;
3 }

```

Do-While

```

1 do {
2     statemen
3 } while (ekspresi);

```

For

```

1 for (inisialisasi; ekspresi; increment/decrement) {
2     statemen
3 }

```

Foreach

```

1 foreach ($array as $current) {
2     statemen
3 }

```

2. Prosedur Praktikum

a) Latihan 1

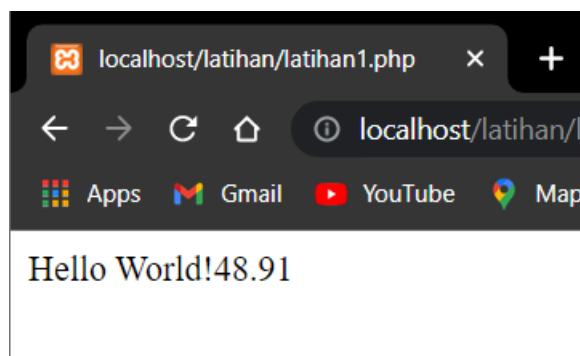
Tulis kode di bawah ini.

```
1 <?php
2 $hello = "Hello World!";
3 $a_number = 4;
4 $anotherNumber = 8.9;
5 $bool = true;
```

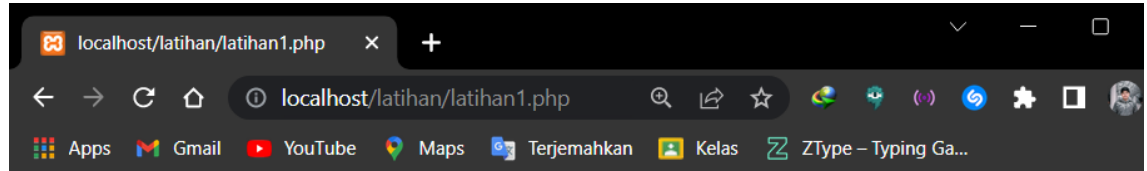
Hapus notasi \$, lalu jalankan kode diatas. Apa yang terjadi? Lalu simpulkan.

Jawab :

Output sebelum di hapus :



Output setelah di hapus :



Parse error: syntax error, unexpected '=' in C:\xampp\htdocs\LATIHAN\latihan1.php on line 2

Penjelasan :

Jika notasi \$ dihapus, maka kode akan menghasilkan error karena \$ merupakan karakter khusus dalam bahasa pemrograman PHP yang menandakan bahwa variabel yang dideklarasikan diikuti oleh nama variabel tersebut. Oleh karena itu, jika notasi \$ dihapus, maka PHP tidak akan mengenali variabel-variabel tersebut dan akan menghasilkan error.

Secara umum, variabel dalam bahasa pemrograman PHP diawali dengan karakter \$ dan diikuti oleh nama variabel yang dapat terdiri dari huruf, angka, dan underscore. Karakter pertama dari nama variabel biasanya merupakan huruf atau underscore dan tidak dapat dimulai dengan angka. Variabel dapat menyimpan nilai berupa teks, angka, atau nilai boolean.

Ketika variabel dipanggil atau digunakan, maka notasi \$ harus digunakan untuk mengakses nilai yang tersimpan dalam variabel tersebut.

b) **Latihan 2**

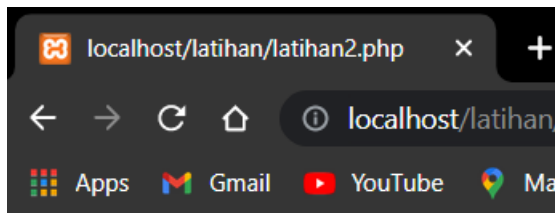
Tulis kode di bawah ini.

```
1 <?php
2     echo "<h1>Variabel</h1></br>";
3
4     $angka = 99;
5     echo "Ini adalah angka = $angka </br>";
6
7     $angka_2 = $angka + 1;
8     echo "Ini hasilnya = $angka_2 </br>";
```

Amati apa hasil dari kode di atas lalu simpulkan.

Jawab :

Output yang ditampilkan :



Variabel

Ini adalah angka = 99

Ini hasilnya = 100

Penjelasan :

Pada kode di atas, terdapat dua variabel yang dideklarasikan, yaitu \$angka dan \$angka_2. Variabel \$angka diberi nilai 99, sedangkan variabel \$angka_2 diberi nilai yang dihasilkan dari operasi penjumlahan antara variabel \$angka dengan angka 1.

Kemudian, nilai dari kedua variabel tersebut ditampilkan ke layar menggunakan perintah echo, di mana nilai variabel \$angka akan ditampilkan pada baris kedua, dan nilai variabel \$angka_2 akan ditampilkan pada baris ketiga. Dari kode di atas, dapat disimpulkan bahwa variabel dapat digunakan untuk menyimpan nilai dan melakukan operasi matematika pada nilai tersebut. Variabel juga dapat digunakan untuk menampilkan nilai ke layar menggunakan

perintah echo atau untuk melakukan operasi matematika dengan variabel lainnya.

c) Latihan 3

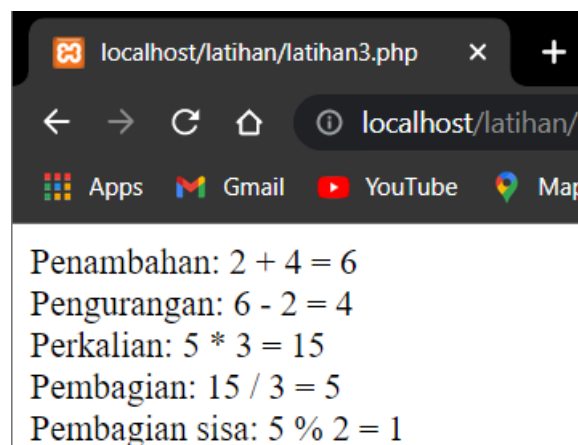
Tulis kode di bawah ini.

```
1 <?php
2 $addition = 2 + 4;
3 $subtraction = 6 - 2;
4 $multiplication = 5 * 3;
5 $division = 15 / 3;
6 $modulus = 5 % 2;
7
8 echo "Penambahan: 2 + 4 = $addition <br>";
9 echo "Pengurangan: 6 - 2 = $subtraction <br>";
10 echo "Perkalian: 5 * 3 = $multiplication <br>";
11 echo "Pembagian: 15 / 3 = $division <br>";
12 echo "Pembagian sisa: 5 % 2 = $modulus";
```

Amati apa hasil dari kode di atas lalu simpulkan.

Jawab :

Output yang di tampilkan :



Penjelasan :

Pada kode di atas, terdapat lima variabel yang dideklarasikan untuk menyimpan hasil dari operasi aritmatika yang berbeda, yaitu \$addition untuk operasi penjumlahan, \$subtraction untuk operasi pengurangan, \$multiplication untuk operasi perkalian, \$division untuk operasi pembagian, dan \$modulus untuk operasi modulus (pembagian sisa).

Kemudian, nilai dari masing-masing variabel tersebut ditampilkan ke layar menggunakan perintah echo.

Dari kode di atas, dapat disimpulkan bahwa bahasa pemrograman PHP mendukung operasi aritmatika dasar seperti penjumlahan, pengurangan, perkalian, pembagian, dan modulus. Variabel juga dapat digunakan untuk menyimpan

hasil dari operasi-aritmatika tersebut dan menampilkan hasil tersebut ke layar menggunakan perintah echo.

d) **Latihan 4**

Tulis kode di bawah ini.

```
1 <?php
2 $x = 4;
3 $x += 3;
4 echo "Hasil dari operasi tersebut adalah = $x";
```

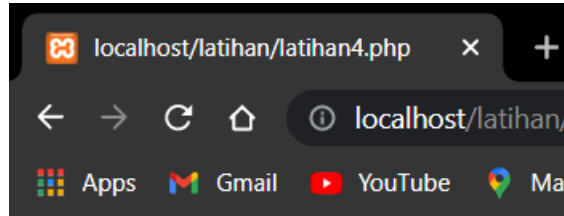
Gantikan operator "+=" dengan :

- "--"
- "*="
- "/="
- "%="
- ".*="

Amati apa hasil dari operator tersebut lalu simpulkan.

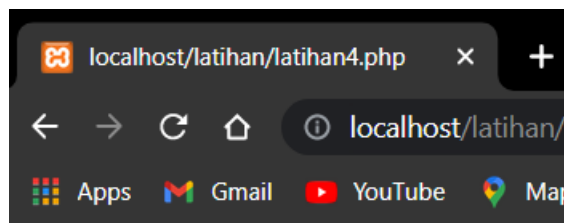
Jawab :

Output yang ditampilkan pada kode diatas adalah :



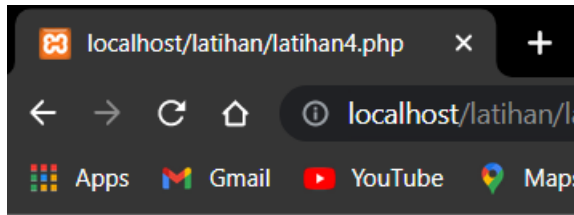
Hasil dari operasi tersebut adalah = 7

Output yang ditampilkan setelah diganti dengan "--" adalah:



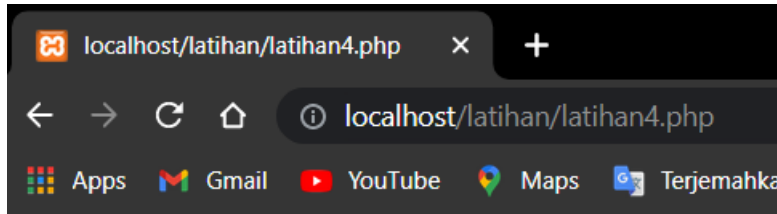
Hasil dari operasi tersebut adalah = 1

Output yang ditampilkan setelah diganti dengan "*" adalah:



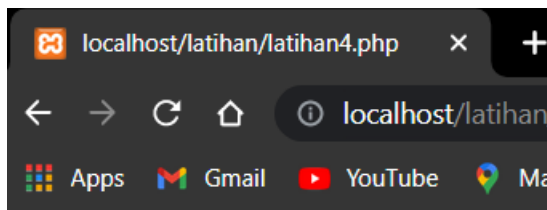
Hasil dari operasi tersebut adalah = 12

Output yang ditampilkan setelah diganti dengan "/" adalah:



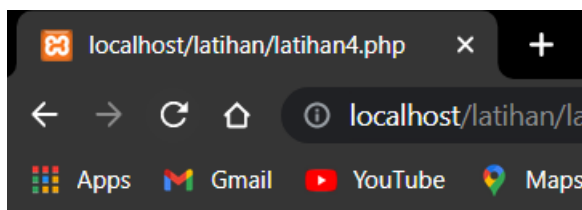
Hasil dari operasi tersebut adalah = 1.33333333333333

Output yang ditampilkan setelah diganti dengan "%" adalah:



Hasil dari operasi tersebut adalah = 1

Output yang ditampilkan setelah diganti dengan "." adalah:



Hasil dari operasi tersebut adalah = 43

Penjelasan :

Dari kode di atas, dapat disimpulkan bahwa:

- Operator "-" digunakan untuk mengurangi nilai sebuah variabel dengan nilai tertentu, sehingga nilai variabel tersebut berkurang.
- Operator "/" digunakan untuk mengalikan nilai sebuah variabel dengan nilai tertentu, sehingga nilai

variabel tersebut menjadi hasil kali dari nilai awal dengan nilai tertentu.

- Operator "/"= digunakan untuk membagi nilai sebuah variabel dengan nilai tertentu, sehingga nilai variabel tersebut menjadi hasil bagi dari nilai awal dengan nilai tertentu.
- Operator "%=" digunakan untuk membagi nilai sebuah variabel dengan nilai tertentu, dan menyimpan sisa hasil bagi tersebut ke dalam variabel.
- Operator ".=" digunakan untuk menggabungkan dua buah string menjadi satu string.

Perlu diingat bahwa operator-operator ini hanya berlaku untuk tipe data angka (integer dan float) dan string. Operasi aritmatika dengan tipe data lain seperti boolean dan array tidak bisa menggunakan operator-operator tersebut.

e) **Latihan 5**

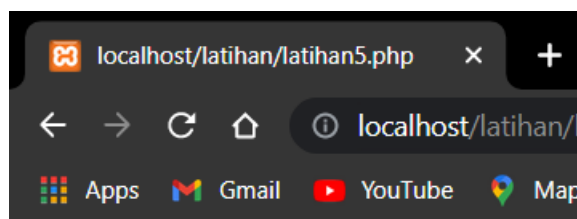
Tulis kode di bawah ini.

```
1 <?php
2     $my_name = "anotherguy";
3
4     if ($my_name == "someguy") {
5         echo "Your name is someguy!<br>";
6     }
7
8     echo "Welcome to my homepage!";
```

Isikan variabel \$my_name dengan nilai yang lain dan simpulkan.

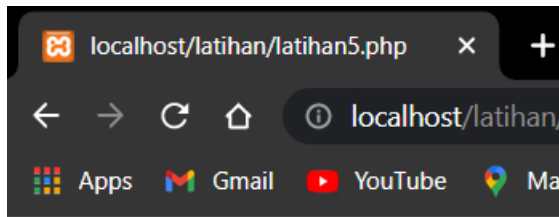
Jawab :

Output yang ditampilkan kode di atas :



Welcome to my homepage!

Output yang ditampilkan setelah \$my_name diganti dengan nilai yang lain :



Your name is someguy!
Welcome to my homepage!

Penjelasan :

Karena kondisi pada if statement tidak terpenuhi (nilai dari \$my_name tidak sama dengan "someguy"), maka blok kode di dalam kurung kurawal {} tidak dijalankan, dan yang dijalankan adalah perintah echo yang berada di luar blok kode tersebut.

Dengan demikian, dapat disimpulkan bahwa hasil output kode di atas bergantung pada nilai yang diberikan kepada variabel \$my_name. Jika nilai tersebut sama dengan "someguy", maka pesan "Your name is someguy!" akan ditampilkan, sedangkan jika nilai tersebut tidak sama dengan "someguy", maka hanya pesan "Welcome to my homepage!" yang akan ditampilkan.

f) **Latihan 6**

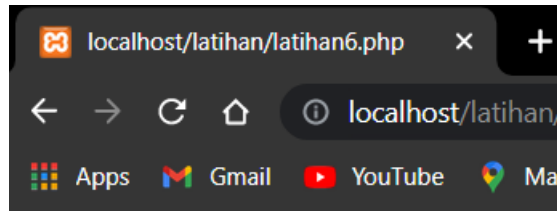
Tulis kode di bawah ini.

```
1  <?php
2      $destination = "Tokyo";
3      echo "Traveling to $destination<br>";
4
5      switch ($destination){
6          case "Las Vegas":
7              echo "Bring an extra $500";
8              break;
9          case "Amsterdam":
10             echo "Bring an open mind";
11             break;
12          case "Egypt":
13             echo "Bring 15 bottles of SPF 50 Sunscreen";
14             break;
15          case "Tokyo":
16             echo "Bring lots of money";
17             break;
18          case "Caribbean Islands":
19             echo "Bring a swimsuit";
20             break;
21     }
```

Gantikan nilai dari variabel `$destination` dengan "Amsterdam". Apa yang terjadi? Jelaskan alur kerja dari alur switch.

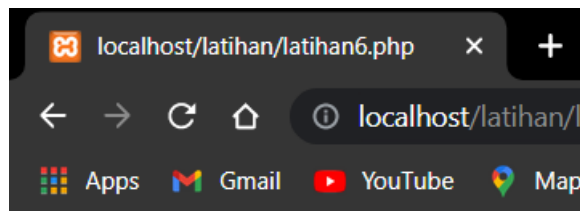
Jawab :

Output yang ditampilkan kode diatas :



Traveling to Tokyo
Bring lots of money

Output yang ditampilkan setelah nilai `$destination` diganti dengan Amsterdam :



Traveling to Amsterdam
Bring an open mind

Penjelasan :

Pada dasarnya, struktur switch digunakan untuk membandingkan sebuah variabel dengan serangkaian nilai yang mungkin (dalam hal ini menggunakan case) dan mengeksekusi kode yang sesuai dengan nilai tersebut. Alur kerja dari switch adalah sebagai berikut:

- Ekspresi yang akan diperiksa (dalam hal ini variabel `$destination`) dievaluasi.
- Hasil evaluasi ini dibandingkan dengan setiap nilai kasus (case) dalam struktur switch.
- Jika nilai kasus cocok dengan hasil evaluasi, maka blok kode di bawahnya (setelah titik dua) akan dieksekusi. Jika tidak ada nilai kasus yang cocok, maka blok kode di bawah default (jika ada) akan

dieksekusi. Jika tidak ada blok kode default, maka switch akan keluar tanpa melakukan apa pun.

Dalam kasus kode di atas, variabel \$destination dievaluasi dan dibandingkan dengan setiap kasus (case) yang ada.

Ketika nilai variabel adalah "Tokyo", maka kode di bawah kasus "Tokyo" dieksekusi, yaitu "Bring lots of money".

Sedangkan ketika nilai variabel diubah menjadi "Amsterdam", maka kode di bawah kasus "Amsterdam" dieksekusi, yaitu "Bring an open mind".

g) **Latihan 7**

Tulis kode di bawah ini.

```
1  <?php
2      $brush_price = 5;
3
4      echo "<table border='1' align='center'>";
5      echo "<tr><th>Quantity</th>";
6      echo "<th>Price</th></tr>";
7
8      for ( $counter = 10; $counter <= 100; $counter += 10) {
9
10         echo "<tr><td>";
11         echo $counter;
12         echo "</td><td>";
13         echo $brush_price * $counter;
14         echo "</td></tr>";
15     }
16     echo "</table>";
```

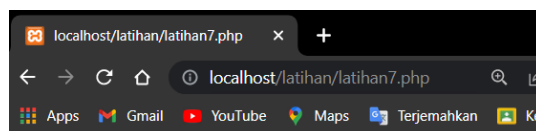
Ganti inkremen dengan \$counter += 5, apa yang terjadi?

Ganti looping for dengan menggunakan while, dan do-while?

Jelaskan bagaimana alur kerja dari looping for, while, dan do-while!

Jawab :

Output yang ditampilkan kode diatas adalah :



Quantity	Price
10	50
20	100
30	150
40	200
50	250
60	300
70	350
80	400
90	450
100	500

Output yang ditampilkan menggunakan while :

Quantity	Price
10	50
20	100
30	150
40	200
50	250
60	300
70	350
80	400
90	450
100	500

Output yang ditampilkan menggunakan do-while :

Quantity	Price
10	50
20	100
30	150
40	200
50	250
60	300
70	350
80	400
90	450
100	500

Penjelasan :

Alur kerja dari for, while, dan do-while adalah sebagai berikut:

- for: Variabel kontrol (biasanya bernama i) dideklarasikan dan diinisialisasi pada awal perulangan. Kemudian, kondisi evaluasi dievaluasi. Jika kondisi tersebut benar, maka blok kode di dalam perulangan akan dieksekusi. Setelah itu, inkremen dilakukan untuk variabel kontrol, dan kondisi

evaluasi dievaluasi lagi. Proses ini diulangi sampai kondisi evaluasi tidak lagi benar.

- while: Kondisi evaluasi dievaluasi terlebih dahulu. Jika kondisi tersebut benar, maka blok kode di dalam perulangan akan dieksekusi. Setelah itu, kondisi evaluasi dievaluasi lagi, dan jika masih benar, maka proses akan diulangi. Proses ini diulangi sampai kondisi evaluasi tidak lagi benar.
- do-while: Pertama-tama, blok kode di dalam perulangan akan dieksekusi. Setelah itu, kondisi evaluasi dievaluasi. Jika kondisi tersebut benar, maka proses akan diulangi. Proses ini diulangi sampai kondisi evaluasi tidak lagi benar. Sedangkan untuk do-while, blok kode di dalam perulangan akan dieksekusi minimal satu kali, meskipun kondisi evaluasi awalnya salah.