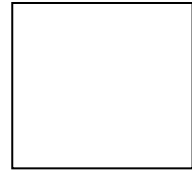


## LATIHAN PRAKTIKUM PEMROGRAMAN WEB

BAB : DASAR PHP  
NAMA : WAHYU SEPTIADI  
NIM : 175150201111015  
ASISTEN : 1. DESE NARFA FIRMANSYAH  
2. IQBAL SANTOSO PUTRA  
TGL PRAKTIKUM : 19 FEBRUARI 2019



### LATIHAN 1

#### A. Soal

Tulis kode di bawah ini.

```
<?php
    $hello = "Hello World!";
    $a_number = 4;
    $anotherNumber = 8.9;
    $bool = true;
?>
```

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

#### B. Screenshoot



#### C. Syntax

1	
---	--

#### D. Penjelasan

Jika kita hapus notasi \$ pada kode php diatas, maka akan menghasilkan error seperti screenshot diatas, hal tersebut disebabkan oleh kesalahan dalam pembuatan variabel. Pembuatan variabel dalam PHP harus diawali dengan karakter dolar (\$).

### LATIHAN 2

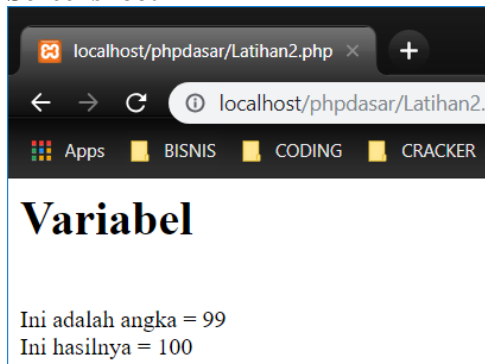
#### A. Soal

Tulis kode di bawah ini.

```
<?php
echo "<h1>Variabel</h1></br>";
$angka = 99;
echo "Ini adalah angka = $angka </br>";
$angka_2 = $angka + 1;
echo "Ini hasilnya = $angka 2 </br>";
```

Amati apa hasil dari kode di atas lalu simpulkan.

## B. Screenshoot



## C. Syntax

1	
---	--

## D. Penjelasan

Jika file / syntax diatas dijalankan akan menghasilkan output seperti Screenshot diatas, dimana syntax diatas menjumlahkan variabel angka dengan 1 yang disimpan ke variabel angka\_2 dan mencetaknya dengan perintah “echo”.

# LATIHAN 3

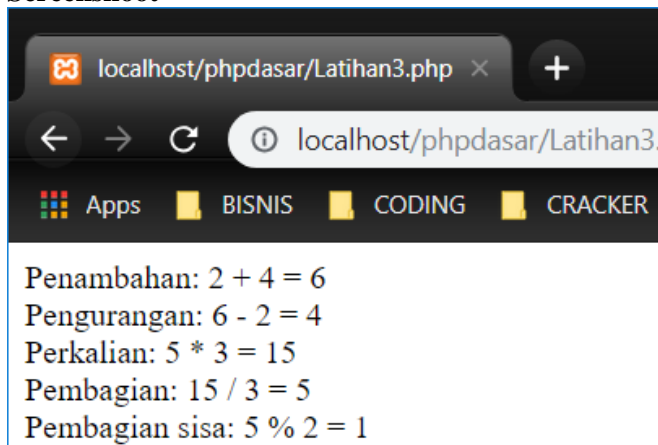
## A. Soal

Tulis kode di bawah ini.

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

Amati apa hasil dari kode di atas lalu simpulkan.

## B. Screenshoot



### C. Syntax

1	
---	--

### D. Penjelasan

Dari syntax diatas dapat kita ketahui, bahwa syntax diatas hanya memanggil variabel yang telah diberi operasi aritmatika dari 2 bilangan yang berbeda-beda.

## LATIHAN 4

### A. Soal

Tulis kode di bawah ini.

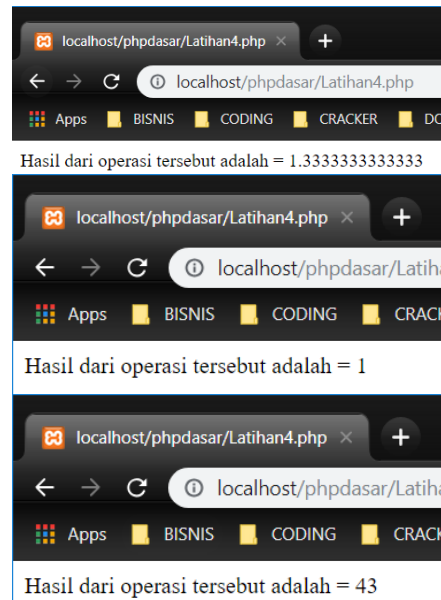
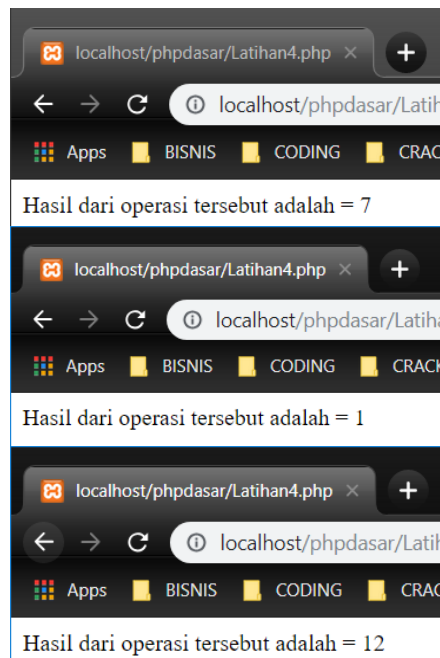
```
<?php
    $x = 4;
    $x += 3;
    echo "Hasil dari operasi tersebut adalah = $x";
```

Gantikan operator “+=” dengan:

1. “=”
2. “\*=”
3. “/=”
4. “%=”
5. “.=

Amati apa hasil dari operator tersebut lalu simpulkan.

### B. Screenshoot



### C. Syntax

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

```

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

```

```

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

```

```

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

```

```

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

```

#### D. Penjelasan

Operasi diatas sama saja dengan operasi seperti berikut ini :

- $4 + 3 = 7$
- 1.  $4 - 3 = 1$
- 2.  $4 \times 3 = 12$
- 3.  $4 : 3 = 1,33333$
- 4.  $4 \% 3 = 1$  (sisa hasil bagi)
- 5.  $4 . 3 = 43$  (fungsi titik(.) tersebut yaitu menggabungkan 2 buah variabel)

Dapat disimpulkan bahwa pada syntax diatas hanya mempunyai 1 variabel saja yaitu variabel x, dimana variabel x tersebut awalnya diinisialisasi dengan nilai 4 kemudian dilakukan oleh beberapa operasi dengan nilai 3, sehingga nilai variabel x tersebut berubah secara berkelanjutan (Increment nilai variabel).

## LATIHAN 5

#### A. Soal

Tulis kode di bawah ini.

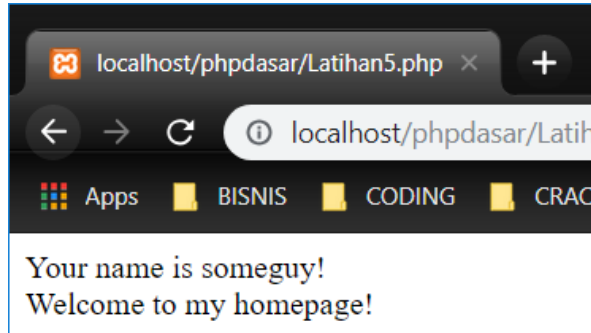
```

<?php
    $my_name = "anotherguy";
    if ($my_name == "someguy") {
        echo "Your name is someguy!<br>";
    }
    echo "Welcome to my homepage!";

```

Isikan variabel \$my\_name dengan nilai yang lain dan simpulkan.

## B. Screenshoot



## C. Syntax

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

## D. Penjelasan

Kita dapat menyimpulkan bahwa pada syntax diatas, memberikan seleksi kondisi jika variabel `my_name = someguy` maka akan mencetak “Your name is someguy” dan pada proses mencetak String ke dua, pada nomer 6 tersebut akan otomatis dicetak, karena tidak termasuk kedalam seleksi kondisi.

# LATIHAN 6

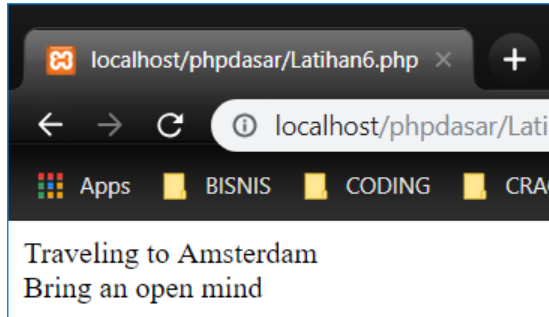
## A. Soal

Tulis kode di bawah ini.

```
<?php
    $destination = "Tokyo";
    echo "Traveling to $destination<br>";
    switch ($destination){
        case "Las Vegas":
            echo "Bring an extra $500";
            break;
        case "Amsterdam":
            echo "Bring an open mind";
            break;
        case "Egypt":
            echo "Bring 15 bottles of SPF 50 Sunscreen";
            break;
        case "Tokyo":
            echo "Bring lots of money";
            break;
        case "Caribbean Islands":
            echo "Bring a swimsuit";
            break;
    }
```

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

## B. Screenshoot



## C. Syntax

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

## D. Penjelasan

Alur kerja pada syntax diatas yaitu pertama program akan mencetak “Traveling to” kemudian mencetak nilai yang ada di variabel destination yaitu Amsterdam, Kemudian program menseleksi nilai pada varaibel destination, dimana pada pernyataan (case) 1 yaitu Las Vegas yang mana tidak sesuai pada variabel destination, maka akan dilanjutkan ke case 2 yaitu Amsterdam, pada case ini sesuai pada nilai variabel destination, maka dari itu program akan mencetak nilai / perintah yang ada didalam case 2 ini yaitu mencetak “Bring an open mind”, kemudian break / program diberhentikan.

# LATIHAN 7

## A. Soal

Tulis kode di bawah ini.

```
<?php
    $brush_price = 5;
    echo "<table border=\"1\" align=\"center\">";
    echo "<tr><th>Quantity</th>";
    echo "<th>Price</th></tr>";
    for ( $counter = 10; $counter <= 100; $counter += 10){
        echo "<tr><td>";
```

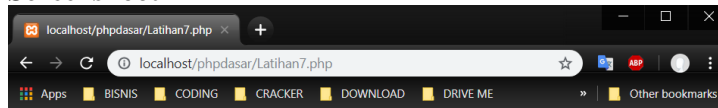
```

        echo $counter;
        echo "</td><td>";
        echo $brush_price * $counter;
        echo "</td></tr>";
    }
    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?

## B. Screenshoot



Quantity	Price
10	50
15	75
20	100
25	125
30	150
35	175
40	200
45	225
50	250
55	275
60	300
65	325
70	350
75	375
80	400
85	425
90	450
95	475
100	500

## C. Syntax

```

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

```

```

1 <?php
2     $brush_price = 5;
3     echo "<table border=\"1\" align=\"center\">";
4     echo "<tr><th>Quantity</th>";
5     echo "<th>Price</th></tr>";
6     $counter = 10;
7     do {
8         echo "<tr><td>";
9         echo $counter;

```

10	echo "</td><td>";
11	echo \$brush_price * \$counter;
12	echo "</td></tr>";
13	\$counter += 5;
14	} while (\$counter <= 100);
15	echo "</table>";

#### D. Penjelasan

Setelah kita ganti increment dari variabel \$counter menjadi 5, maka dari quantity sebelumnya kelipatan 10 dan sekarang quantity berubah menjadi kelipatan 5 yang dimulai dari angka 10 (10, 15, 20, 25, ...).

Penulisan pada syntax for yaitu kita harus memberikan beberapa ekspresi / batasan-batasan perulangan (iterasi) yang diletakkan didalam kurung setelah for. Batasan-batasan tersebut antara lain : Statement 1 yaitu inisialisasi variabel, statement 2 yaitu batas akhir dari perulangan, dan statement 3 yaitu increment/decrement. Sedangkan cara kerjanya yaitu program mengecek apakah nilai variabel counter kurang dari atau sama dengan 100? jika benar, maka statement didalam for akan dijalankan sampai counter menjadi sama dengan 100, setelah itu program akan berhenti dijalankan.

Penulisan pada syntax while yaitu kita harus inisialisasi variabelnya diluar syntax while, kemudian variabel untuk batas akhir dari perulangan kita letakkan didalam kurung setelah while, sedangkan untuk increment/decrement kita letakkan didalam perulangan while. Untuk cara kerja pada perulangan while ini yaitu hampir sama seperti perulangan for yaitu variabel counter diseleksi dahulu, baru kemudian di lakukan statement yang ada didalam perulangan tersebut.

Penulisan pada syntax do-while yaitu hampir sama seperti perulangan while, dimana inisialisasi variabel dilakukan diluar do-while, dan untuk increment juga berada di dalam perulangan, akan tetapi untuk batasan akhir perulangan diletakkan pada akhir yaitu pada dalam kurung setelah while. Untuk cara kerja pada perulangan do-while ini yaitu berbeda dengan kedua perulangan sebelumnya. Dimana program akan menjalankan statement yang ada di dalam perulangan dahulu, kemudian baru dicek apakah nilai variabel counter kurang dari atau sama dengan 100? Jika benar, maka statement didalam perulangan tetap dijalankan, sedangkan jika variabel counter sudah mencapai sama dengan batas akhirnya yaitu 100, maka program akan berhenti dijalankan.