

LATIHAN PRAKTIKUM PENGEMBANGAN APLIKASI WEB

BAB : DASAR PHP
NAMA : DANI ADRIAN
NIM : 225150201111009
ASISTEN : - IBAR HUTTAQI SULTHON
 - MUHAMMAD AMMAR
TANGGAL PRAKTIKUM : 21/03/2024

LATIHAN 1

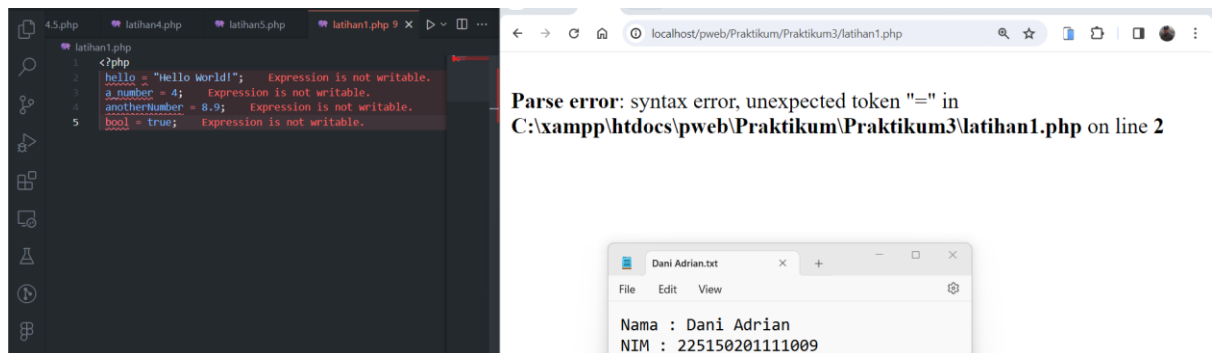
A. Soal

Tulis kode dibawah ini

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

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

B. Screenshoot



C. Syntax

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

D. Penjelasan

Ketika menjalankan kode diatas, PHP akan menghasilkan pesan error seperti "Parse error: syntax error, unexpected '='". Hal ini terjadi karena PHP tidak dapat mengenali variabel tanpa notasi \$, dan menggunakan operator = untuk mendeklarasikan variabel tanpa \$ bukanlah sintaks yang valid dalam PHP.

Notasi \$ sangat penting dalam PHP karena digunakan untuk mendeklarasikan variabel. Tanpa notasi \$, PHP tidak dapat mengenali variabel yang ingin digunakan, sehingga kode akan menghasilkan error.

LATIHAN 2

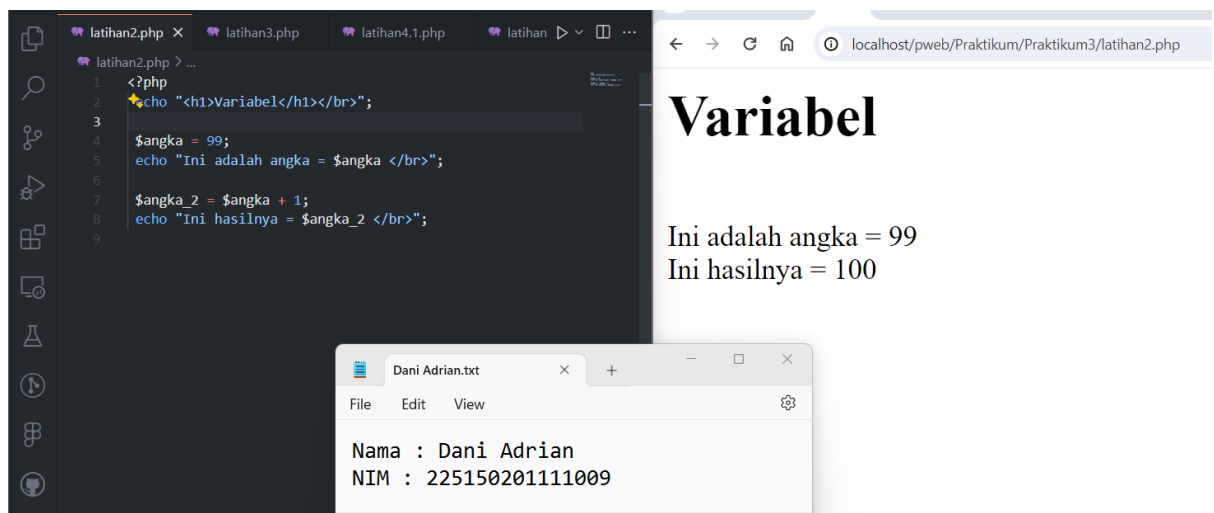
A. Soal

Tulis kode dibawah 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.

B. Screenshoot



C. Syntax

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>";

D. Penjelasan

Syntax di atas mencetak teks yang berisi sebuah tag HTML `<h1>` dan dua variabel yang dideklarasikan dan diinisialisasi dengan nilai tertentu. Variabel `$angka` diinisialisasi dengan nilai 99, kemudian dicetak bersama dengan teks "Ini adalah angka = " menggunakan `echo`. Selanjutnya, nilai variabel `$angka` ditambah satu dan disimpan dalam variabel `$angka_2`, kemudian nilai variabel `$angka_2` juga dicetak.

Syntax diatas mendemonstrasikan penggunaan variabel dalam PHP untuk menyimpan nilai dan melakukan operasi matematika sederhana, serta mencetak hasilnya menggunakan fungsi `echo`.

LATIHAN 3

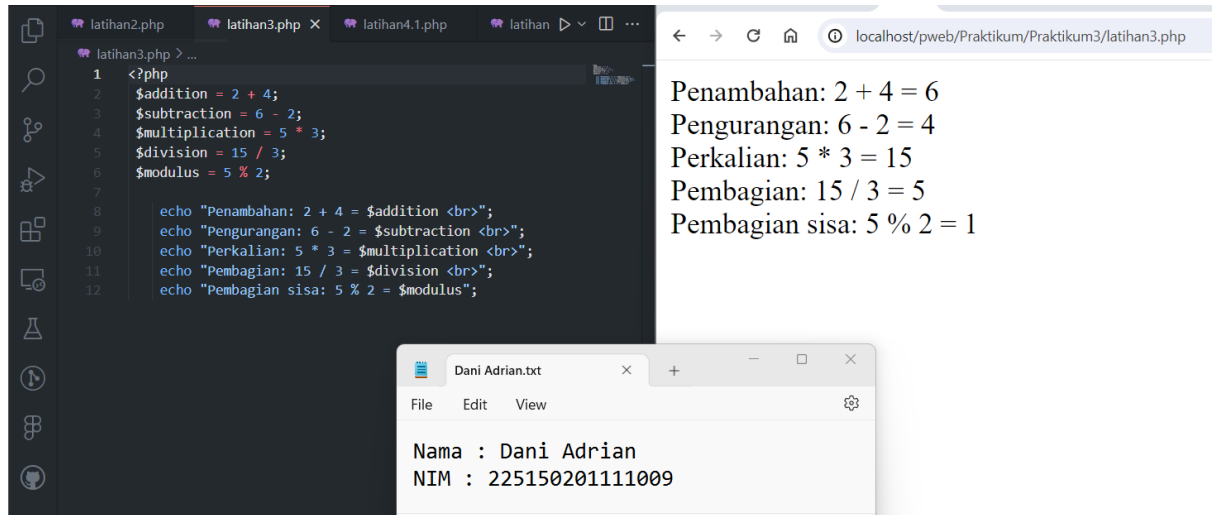
A. Soal

Tulis kode dibawah ini

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

Amati apa hasil dari kode di atas lalu simpulkan.

B. Screenshoot



C. Syntax

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

D. Penjelasan

Syntax di atas melakukan beberapa operasi aritmatika dasar dalam PHP, yaitu penambahan, pengurangan, perkalian, pembagian, dan modulus. Setelah operasi-operasi tersebut dilakukan, hasilnya dicetak menggunakan fungsi `echo`.

- Penambahan: $2 + 4 = 6$
- Pengurangan: $6 - 2 = 4$
- Perkalian: $5 * 3 = 15$
- Pembagian: $15 / 3 = 5$
- Pembagian sisa: $5 \% 2 = 1$

Syntax tersebut menunjukkan penggunaan operator aritmatika dalam PHP untuk melakukan operasi matematika dasar seperti penambahan, pengurangan, perkalian, pembagian, dan modulus, dan kemudian mencetak hasilnya.

LATIHAN 4

A. Soal

Tulis kode dibawah ini

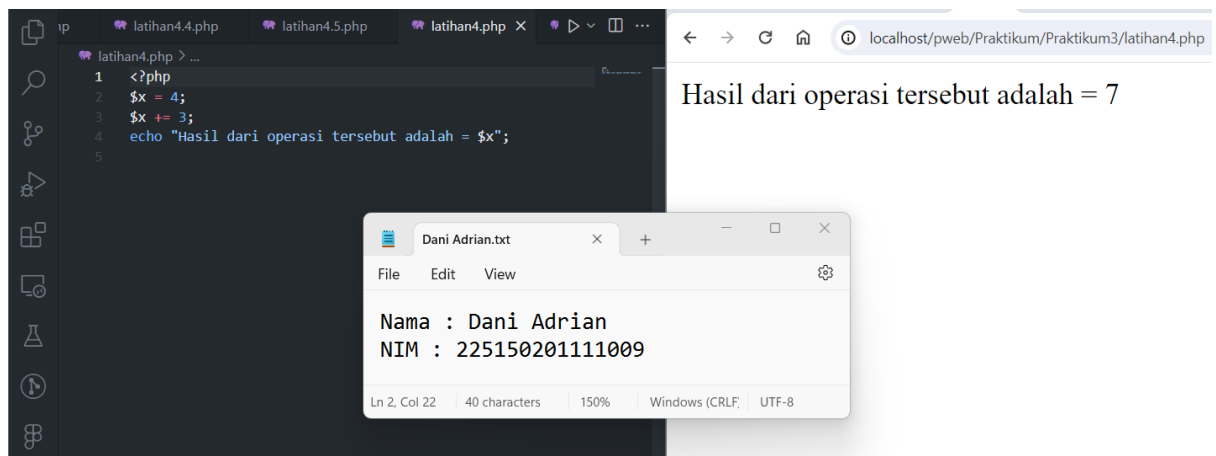
1.	<?php
2.	\$x = 4;
3.	\$x += 3;
4.	echo "Hasil dari operasi tersebut adalah = \$x";

Gantikan operator "+=" dengan:

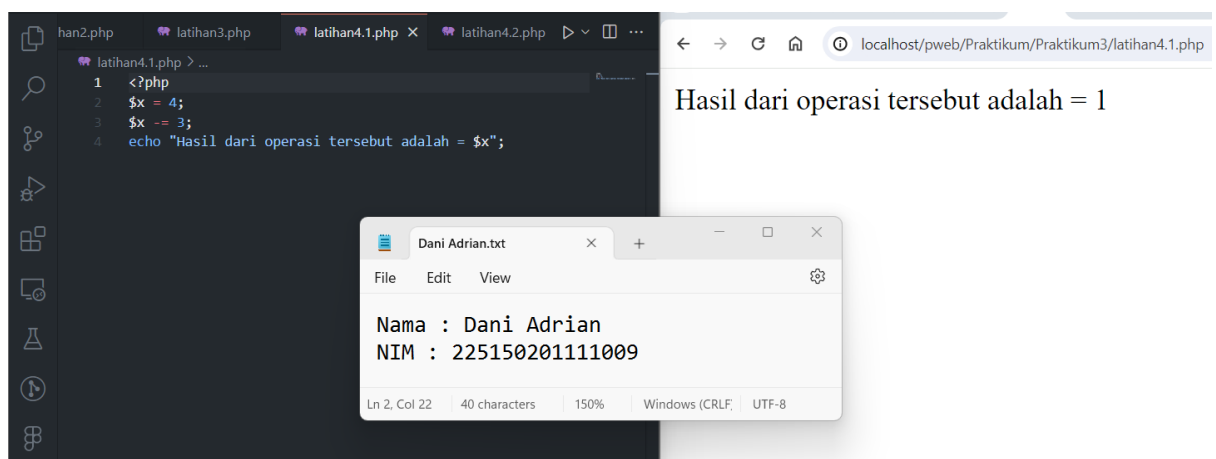
1. "--="
2. "*="
3. "/="
4. "%="
5. ".="

B. Screenshoot

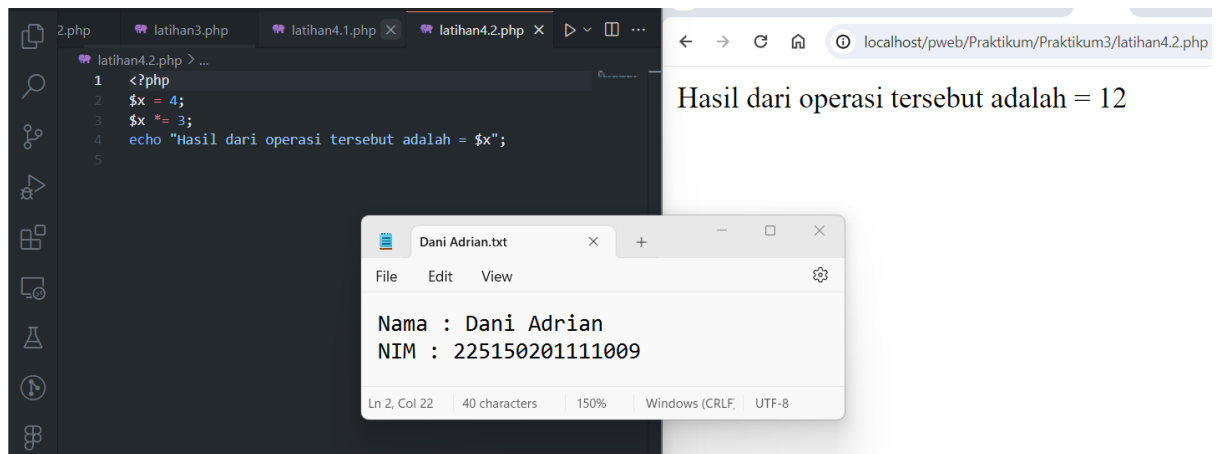
Operator "+="



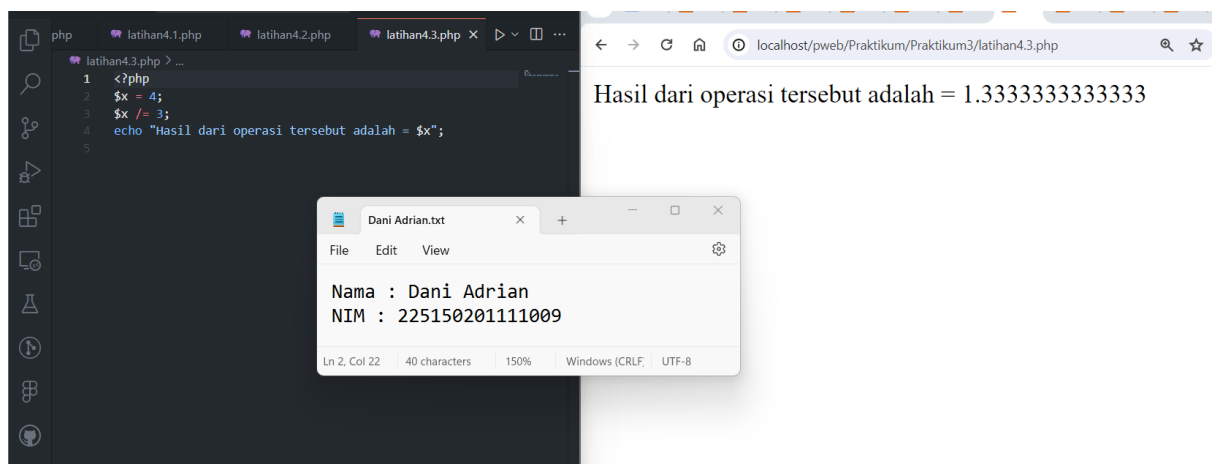
Operator "--="



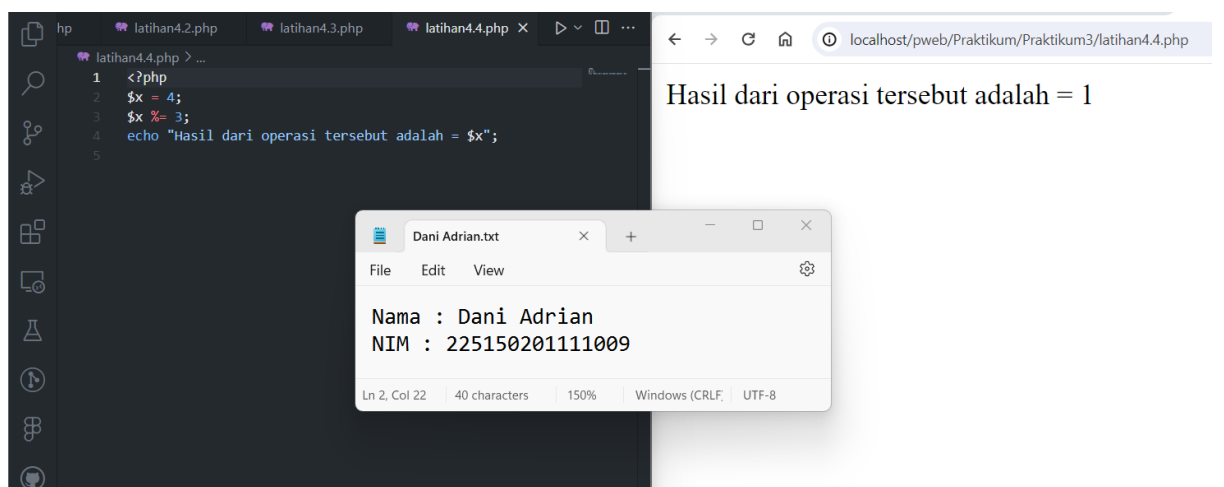
Operator "*"="



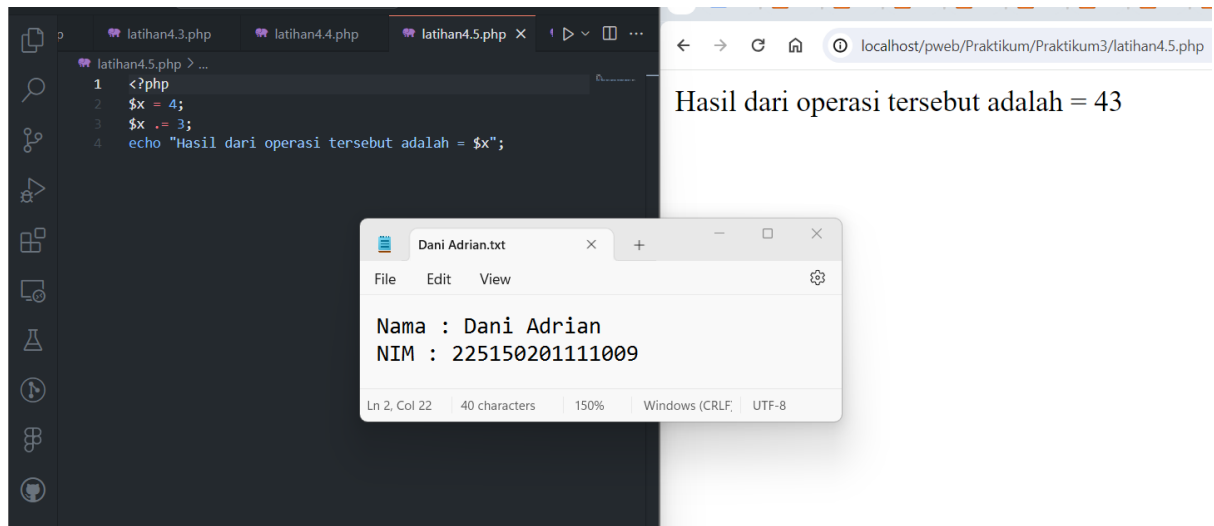
Operator "/"="



Operator "%"="



Operator "."="



C. Syntax

1.	Operator "+="
2.	<?php
3.	\$x = 4;
4.	\$x += 3;
5.	echo "Hasil dari operasi tersebut adalah = \$x";
6.	
7.	Operator "-="
8.	<?php
9.	\$x = 4;
10.	\$x -= 3;
11.	echo "Hasil dari operasi tersebut adalah = \$x";
12.	
13.	Operator "*="
14.	<?php
15.	\$x = 4;
16.	\$x *= 3;
17.	echo "Hasil dari operasi tersebut adalah = \$x";
18.	
19.	Operator "/="
20.	<?php
21.	\$x = 4;
22.	\$x /= 3;
23.	echo "Hasil dari operasi tersebut adalah = \$x";
24.	
25.	Operator "%="
26.	<?php
27.	\$x = 4;
28.	\$x %= 3;
29.	echo "Hasil dari operasi tersebut adalah = \$x";
30.	
31.	Operator ".="
32.	<?php
33.	\$x = 4;
34.	\$x .= 3;
35.	echo "Hasil dari operasi tersebut adalah = \$x";

D. Penjelasan

Penggantian dengan -= (Pengurangan):

- Hasil: Hasil dari operasi tersebut adalah = 1
- Operator -= digunakan untuk mengurangi nilai variabel.

Penggantian dengan *= (Perkalian):

- Hasil: Hasil dari operasi tersebut adalah = 12
- Operator *= digunakan untuk mengalikan nilai variabel.

Penggantian dengan /= (Pembagian):

- Hasil: Hasil dari operasi tersebut adalah = 1.3333333333333
- Operator /= digunakan untuk membagi nilai variabel.

Penggantian dengan %= (Modulus):

- Hasil: Hasil dari operasi tersebut adalah = 1
- Operator %= digunakan untuk mengambil sisa bagi dari nilai variabel.

Penggantian dengan .= (Concatenation):

- Hasil: Hasil dari operasi tersebut adalah = 43
- Operator .= digunakan untuk menggabungkan string atau menambahkan ke akhir string.

LATIHAN 5

A. Soal

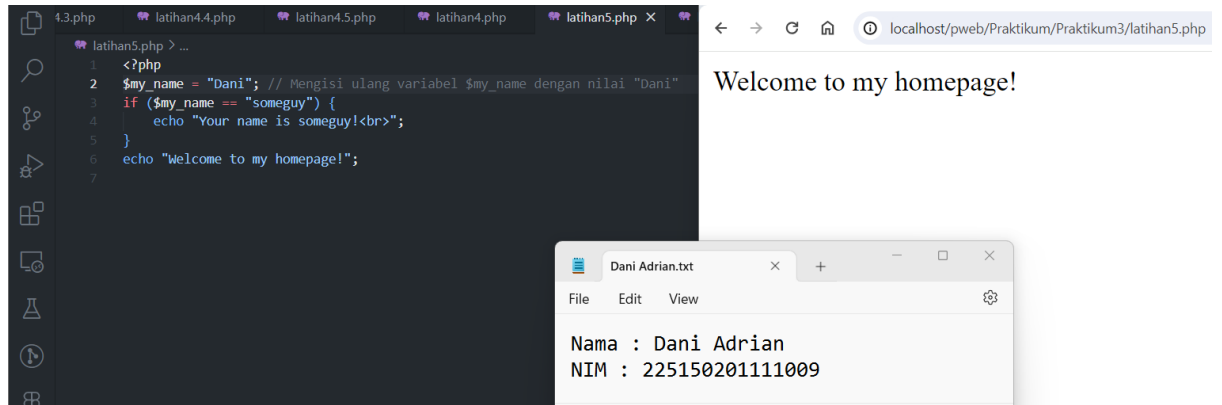
Tulis kode dibawah ini

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

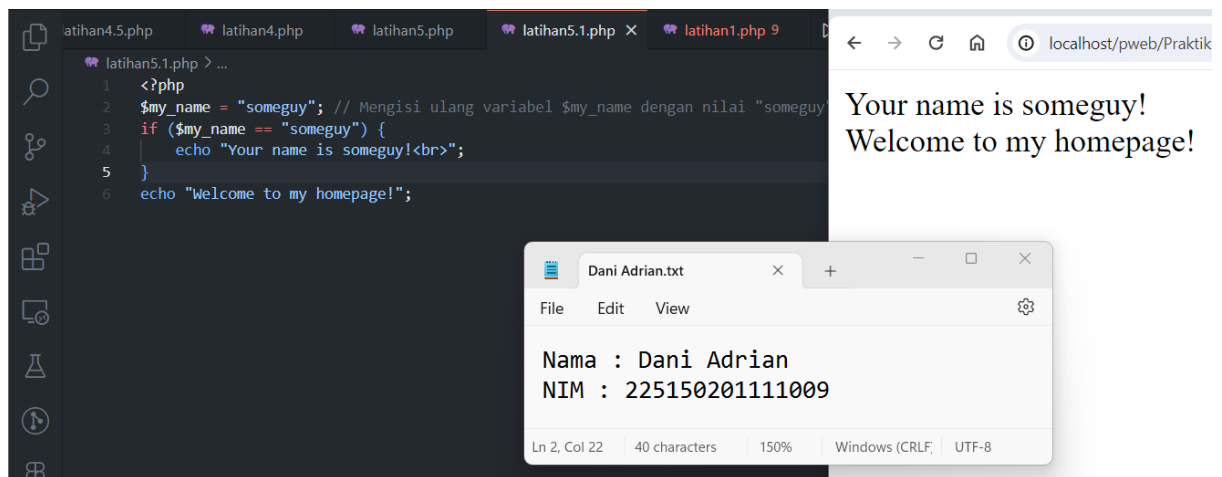
Isikan variabel \$my_name dengan nilai yang lain dan simpulkan.

B. Screenshoot

Ketika variable \$my_name diisi dengan nilai "Dani"



Ketika variable `$my_name` diisi dengan nilai "someguy"



C. Syntax

1.	// Ketika variable <code>\$my_name</code> diisi dengan nilai "Dani"
2.	<code><?php</code>
3.	<code>\$my_name = "someguy"; // Mengisi ulang variabel \$my_name dengan nilai "Dani"</code>
4.	<code>if (\$my_name == "someguy") {</code>
5.	<code> echo "Your name is someguy!
";</code>
6.	<code>}</code>
7.	<code>echo "Welcome to my homepage!";</code>
8.	
9.	//Ketika variable <code>\$my_name</code> diisi dengan nilai "someguy"
10.	<code><?php</code>
11.	<code>\$my_name = "someguy"; // Mengisi ulang variabel \$my_name dengan nilai "someguy"</code>
12.	<code>if (\$my_name == "someguy") {</code>
13.	<code> echo "Your name is someguy!
";</code>
14.	<code>}</code>
15.	<code>echo "Welcome to my homepage!";</code>

D. Penjelasan

Ketika variable `$my_name` diisi dengan nilai "Dani"

Karena nilai `$my_name` tidak sama dengan "someguy", kondisi pada baris 3 tidak terpenuhi, sehingga pesan "Your name is someguy!" tidak akan ditampilkan.

Pesan "Welcome to my homepage!" akan tetap ditampilkan karena pesan tersebut berada di luar blok if. Jadi, saat nilai variabel `$my_name` adalah "Dani", hanya pesan selamat datang yang akan ditampilkan.

Ketika variable `$my_name` diisi dengan nilai "someguy"

Karena nilai `$my_name` sama dengan "someguy", kondisi pada baris 3 terpenuhi, sehingga pesan "Your name is someguy!" akan ditampilkan.

Sama seperti sebelumnya, pesan "Welcome to my homepage!" juga akan tetap ditampilkan karena pesan tersebut berada di luar blok if. Jadi, ketika nilai variabel `$my_name` adalah "someguy", pesan selamat datang dan pesan "Your name is someguy!" akan ditampilkan.

LATIHAN 6

A. Soal

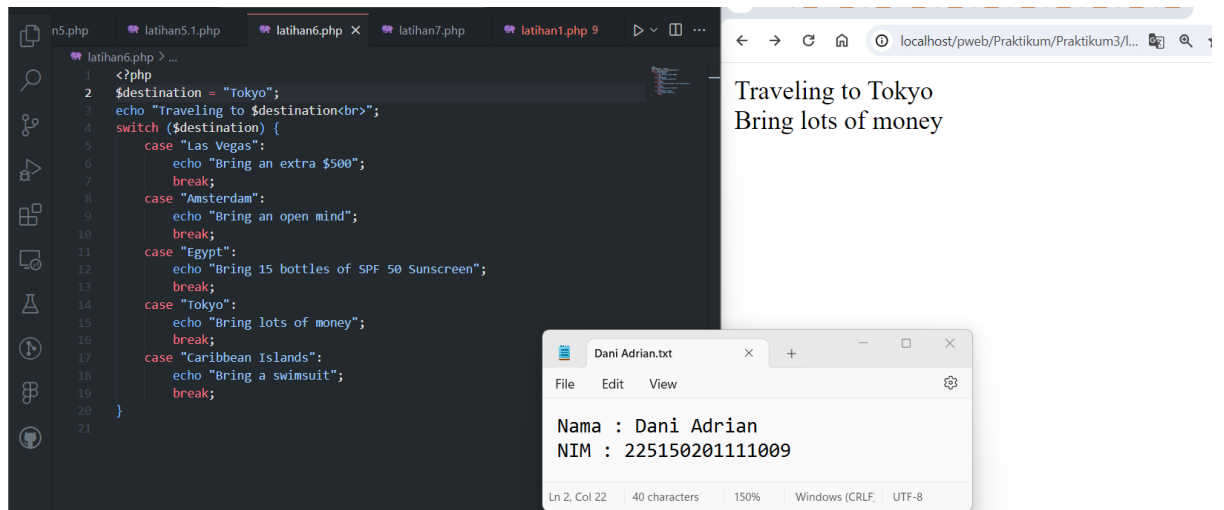
Tulis kode dibawah ini

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

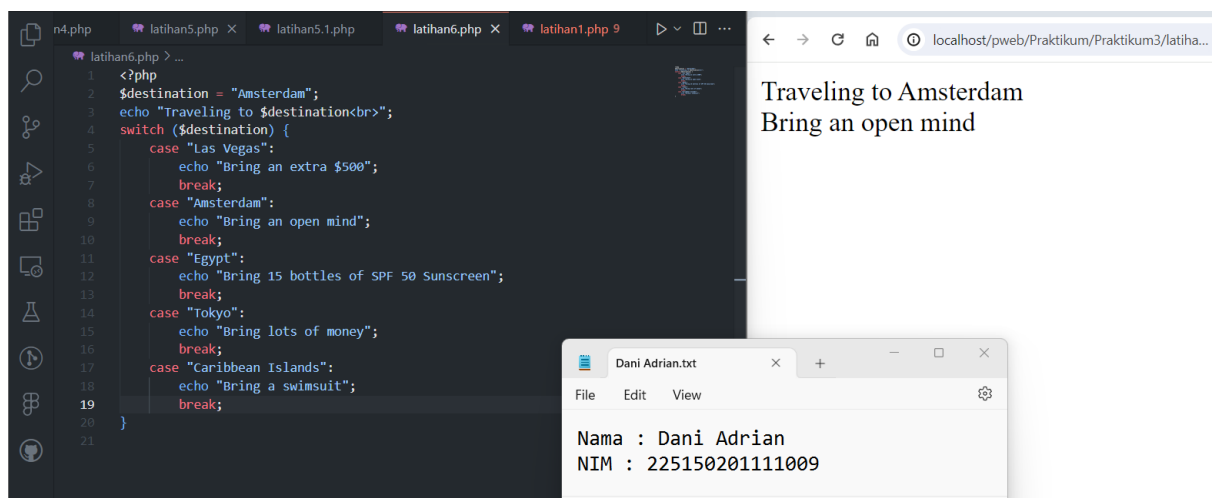
Gantikan nilai dari variabel `$destination` dengan "Amsterdam". Apa yang terjadi? Jelaskan alur kerja dari alur switch.

B. Screenshoot

```
$destination = "Tokyo";
```



`$destination = "Amsterdam";`



C. Syntax

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

19.	break;
20.	}

D. Penjelasan

Ketika menjalankan kode tersebut dengan nilai variabel `$destination` tetap "Tokyo", maka hasilnya akan menjadi:

```
Traveling to Tokyo
Bring lots of money
```

- Pada baris 2, pesan "Traveling to `$destination`" akan dicetak, di mana `$destination` adalah nilai variabel yang digunakan dalam struktur switch.
- Struktur switch pada baris 3-15 akan mengevaluasi nilai dari variabel `$destination`.
- Ketika nilai `$destination` sama dengan salah satu kasus yang didefinisikan dalam switch, kode yang sesuai dengan kasus tersebut akan dieksekusi.
- Jika tidak ada kasus yang cocok dengan nilai variabel `$destination`, maka tidak ada kode yang akan dieksekusi.
- Pada kasus yang cocok, kode yang ada setelah kata kunci case akan dieksekusi hingga ditemukan kata kunci break, yang menandakan akhir dari kasus tersebut.
- Setelah itu, eksekusi akan keluar dari struktur switch.
- Dalam contoh ini, karena nilai variabel `$destination` adalah "Tokyo", maka kasus yang cocok adalah kasus "Tokyo", sehingga pesan "Bring lots of money" akan dicetak setelah pesan "Traveling to Tokyo". Setelah itu, eksekusi keluar dari struktur switch.

Kemudian ganti nilai variabel `$destination` dengan "Amsterdam", maka hasilnya akan menjadi:

```
Traveling to Amsterdam
Bring an open mind
```

Karena nilai variabel `$destination` adalah "Amsterdam", maka kasus yang cocok adalah kasus "Amsterdam", sehingga pesan "Bring an open mind" akan dicetak setelah pesan "Traveling to Amsterdam". Setelah itu, eksekusi keluar dari struktur switch.

LATIHAN 7

1. Soal

Tulis kode dibawah ini

1.	<code><?php</code>
2.	<code>\$brush_price = 5;</code>
3.	<code>echo "<table border=\"1\" align=\"center\">";</code>
4.	<code>echo "<tr><th>Quantity</th>";</code>

5.	echo "<th>Price</th></tr>";
6.	for (\$counter = 10; \$counter <= 100; \$counter += 10) {
7.	echo "<tr><td>";
8.	echo \$counter;
9.	echo "</td><td>";
10.	echo \$brush_price * \$counter;
11.	echo "</td></tr>";
12.	}
13.	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?

2. Screenshoot

```

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

```

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

Dani Adrian.txt
File Edit View
Nama : Dani Adrian
NIM : 225150201111009
Ln 1, Col 19 40 characters 150% Windows (CRLF) UTF-8

Mengganti inkremen dalam looping for menjadi \$counter += 5 :

```

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

```

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

Dani Adrian.txt
File Edit View
Nama : Dani Adrian
NIM : 225150201111009
Ln 1, Col 19 40 characters 150% Windows (CRLF) UTF-8

Mengubah looping for menjadi while:

The screenshot shows a code editor with a PHP script using a `while` loop. The script outputs a table with two columns: Quantity and Price. The Quantity ranges from 10 to 100 in increments of 10, and the Price is calculated as Quantity multiplied by 5. A text box below the table displays the user's name and NIM.

```
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 += 10;
14 }
15 echo "</table>";
16
```

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

Nama : Dani Adrian
NIM : 225150201111009

Mengubahnya menjadi do-while:

The screenshot shows a code editor with a PHP script using a `do-while` loop. The script outputs a table with two columns: Quantity and Price. The Quantity ranges from 10 to 100 in increments of 10, and the Price is calculated as Quantity multiplied by 5. A text box below the table displays the user's name and NIM.

```
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    $counter += 10;
14 } while ($counter <= 100);
15 echo "</table>";
16
```

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

Nama : Dani Adrian
NIM : 225150201111009

3. Syntax

1.	// Mengganti inkremen dalam looping for menjadi \$counter += 5 :
2.	<?php
3.	\$brush_price = 5;
4.	echo "<table border='1' align='center'>";
5.	echo "<tr><th>Quantity</th>";
6.	echo "<th>Price</th></tr>";
7.	for (\$counter = 10; \$counter <= 100; \$counter += 5)
8.	{
9.	echo "<tr><td>";
10.	echo \$counter;
11.	echo "</td><td>";
	echo \$brush_price * \$counter;

12.	echo "</td></tr>";
13.	}
14.	echo "</table>";
15.	
16.	// Mengubah looping for menjadi while:
17.	<?php
18.	\$brush_price = 5;
19.	echo "<table border=\"1\" align=\"center\">";
20.	echo "<tr><th>Quantity</th>";
21.	echo "<th>Price</th></tr>";
22.	\$counter = 10;
23.	while (\$counter <= 100) {
24.	echo "<tr><td>";
25.	echo \$counter;
26.	echo "</td><td>";
27.	echo \$brush_price * \$counter;
28.	echo "</td></tr>";
29.	\$counter += 10;
30.	}
31.	echo "</table>";
32.	
33.	// Mengubahnya menjadi do-while:
34.	<?php
35.	\$brush_price = 5;
36.	echo "<table border=\"1\" align=\"center\">";
37.	echo "<tr><th>Quantity</th>";
38.	echo "<th>Price</th></tr>";
39.	\$counter = 10;
40.	do {
41.	echo "<tr><td>";
42.	echo \$counter;
43.	echo "</td><td>";
44.	echo \$brush_price * \$counter;
45.	echo "</td></tr>";
46.	\$counter += 10;
47.	} while (\$counter <= 100);
48.	echo "</table>";

4. Penjelasan

- **Looping for:** Looping ini memiliki tiga bagian: inisialisasi, kondisi, dan inkremen. Inisialisasi dilakukan sekali pada awal loop. Kondisi dievaluasi sebelum setiap iterasi. Jika kondisi terpenuhi, iterasi dilakukan. Setelah setiap iterasi, inkremen dieksekusi. Jika kondisi tidak terpenuhi, loop berhenti.
- **Looping while:** Looping ini hanya memiliki satu bagian yaitu kondisi. Kondisi dievaluasi sebelum setiap iterasi. Jika kondisi terpenuhi, iterasi dilakukan. Jika kondisi tidak terpenuhi, loop berhenti.
- **Looping do-while:** Looping ini juga hanya memiliki satu bagian yaitu kondisi. Bedanya, iterasi pertama dilakukan sebelum kondisi dievaluasi. Jika kondisi terpenuhi setelah iterasi pertama, iterasi akan terus dilakukan. Loop berhenti setelah kondisi tidak terpenuhi lagi.