

Live Session 5

- ❖ Mengimplementasikan Pemrograman Terstruktur
- ❖ Menggunakan Library atau Komponen Preexisting

Vocational School Graduate Academy



Skema Pelatihan :
Junior Web Developer

<i>Output</i> Pelatihan	<p>Setelah mengikuti pelatihan ini, diharapkan peserta kompeten:</p> <ol style="list-style-type: none"> 1. Mengimplementasikan user interface 2. Menerapkan perintah eksekusi bahasa pemrograman berbasis teks, grafik, dan multimedia 3. Menyusun fungsi, file atau sumber daya pemrograman yang lain dalam organisasi yang rapi 4. Menulis kode dengan prinsip sesuai guidelines dan best practices 5. Mengimplementasikan pemrograman terstruktur 6. Menggunakan library atau komponen pre-existing
Jam Pelatihan	67 JP / 9 hari
Jenis Pelatihan	Daring (<i>Online</i>)

Profil Pengajar

Enny Indasyah, S.ST, MT., M.Sc.

NIP : 199103302018072001

Email: enny_indasyah@its.ac.id

- **Jabatan Akademik :**
Dosen ITS
- **Riwayat Pendidikan :**
 - **Sarjana Terapan (S.ST) Teknik Telekomunikasi
Politeknik Elektronika Negeri Surabaya (PENS)**
 - **Master of Science (M.Sc) Department of Computer Science,
NTUST, Taiwan**
 - **Magister Teknik (M.T) Teknik Elektro – Jaringan Cerdas
Multimedia, ITS**



Motivasi



Masa depanmu
 diciptakan oleh apa yang
 kamu lakukan hari ini,
 bukan besok

Aspersepsi

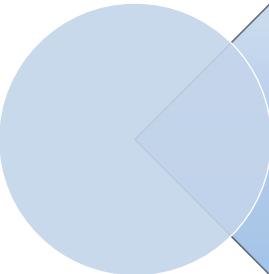


Menggunakan tipe data dan control program

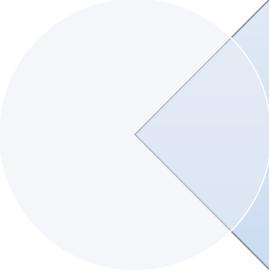
Ringkasan Mata Pelatihan

- Unit Kompetensi Acuan: Mengimplementasikan Pemrograman Terstruktur
- Kode Unit Kompetensi Acuan: J.620100.017.02
- Deskripsi singkat: Mata pelatihan ini menentukan kompetensi, pengetahuan dan sikap kerja yang diperlukan dalam membuat program terstruktur atau prosedural.
- Tujuan Pembelajaran: Peserta dapat membuat program terstruktur atau procedural sesuai dengan paradigma yang berlaku.

Agenda



Tipe Data, Variable &
Syntax PHP



Struktur PHP

Tipe Data, Variable & Sytax PHP

Apa itu tipe data ?

- tipe data adalah **klasifikasi jenis data atau bentukan dari suatu data.** Ia menjelaskan suatu data: dari jenis apakah ia tersusun? Apakah bilangan riil? Atau kah bilangan pecahan? Atau kah ia data yang tersusun dari bentukan karakter? Dsb.

Macam-Macam Tipe Data Dalam PHP

- Ada berbagai macam tipe data yaitu :

Tipe Data	Keterangan
Integer	Berisi bilangan bulat
Float	Berisi bilangan desimal
Boolean	Berisi 2 nilai saja: <code>true</code> dan <code>false</code>
String	Berisi data teks yang diapit oleh tanda <code>''</code> atau <code>'''</code>
Array	Berisi himpunan data

Tipe data pada PHP

- PHP adalah bahasa pemrograman yang bersifat *dinamic typing*, yang **artinya** ia tidak memiliki aturan ketat terhadap pendefinisian tipe data pada setiap variabel.
- PHP akan otomatis menentukan tipe data dari suatu variabel tertentu ketika program dijalankan.

Tipe Data Integer

- Tipe data berupa bilangan bulat
- Misalnya untuk menyimpan data jumlah stock, umur, tinggi badan, nomor sepatu
- Dapat bernilai positif (+) maupun negatif (-)
- Dapat dilakukan operasi matematis (tambah, kurang, kali, bagi)

Tipe Data Integer

- Contoh

```
<?php  
$umur=21;  
$harga=15000;  
$rugi=-500000;  
  
echo $umur; //21  
echo "<br />";  
echo $harga;  
//15000  
echo "<br />";  
echo $rugi; //  
-500000
```

- Contoh dalam operasi matematis

```
<?php  
$a=14;  
$b=16;  
$c= $a + $b;  
echo $c; // 30  
  
$d=$a * $b;  
echo $d; // 224  
?>
```

Tipe Data Float

- Tipe data floating point/real number berupa bilangan decimal (pecahan)
- Digunakan untuk variable yang akan berisi angka pecahan seperti nilai IPK, hasil pembagian, atau hasil komputasi numeric yang angkanya tidak bias ditampung oleh tipe data integer
- Contoh angka float: 0.9 , 2.80 , 3.14 , 0.314E1

Tipe Data Float

- Contoh <?php

```
$angka_float1= 0.78;  
$angka_float2= 14.99;  
$angka_scientific1=0.314E1;  
$angka_scientific2=0.3365E-3;
```

```
echo $angka_float1; // 0.78  
echo "<br />";  
echo $angka_float2; //14.99  
echo "<br />";  
echo $angka_scientific1; //3.14  
echo "<br />";  
echo $angka_scientific2; //0.0003365  
?>
```

Tipe Data Float

- Contoh dalam operasi matematis

```
<?php  
    $a=10.66;  
    $b=12.4;  
    $c= $a + $b;  
    echo $c; // 23.06  
  
    $d=$a / $b;  
    echo $d; // 0.85967741935484  
?>
```

Tipe Data String

- Tipe data yang berisi text, kalimat, atau kumpulan karakter
- Penulisannya diapit oleh single quoted/petik satu ('') atau double quoted/petik ganda ("")
- Contoh:
- 'a' , 'saya sedang belajar PHP' , 'emailku19@gmail.com'
- "a" , "saya sedang belajar PHP" , "emailku19@gmail.com"

Tipe Data String

1

- 2 • Contoh

3

```
4 <?php  
5     $string1='Ini adalah string sederhana';  
6     $string2='Ini adalah string yang bisa memiliki beberapa baris';  
7     $string3='Dia berkata: "I'll be back"';  
8     $string4="Dia berkata: \"I'll be back\"";  
9     $string5="Variabel akan otomatis ditampilkan: $string1 dan  
10    $string3";  
11  
12    echo $string1; echo "<br>";  
13    echo $string2; echo "<br>";  
14    echo $string3; echo "<br>";  
15    echo $string4; echo "<br>";  
16    echo $string5; echo "<br>";  
17    ?>  
18  
19
```

Tipe Data String



localhost/digitalent/pelatihan.php

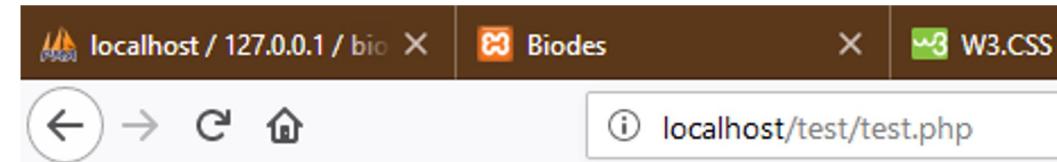
```
echo $string1; echo "<br>" ini adalah string sederhana
echo $string2; echo "<br>" ini adalah string yang bisa memiliki beberapa baris
echo $string3; echo "<br>" dia berkata: "I'll be back"
echo $string4; echo "<br>" Dia berkata: "I'll be back"
echo $string5; echo "<br>" Variabel akan otomatis ditampilkan: ini adalah string sederhana dan dia berkata: "I'll be back"
```

Tipe Data Boolean

- Hanya memiliki 2 nilai : true dan false
- Digunakan dalam operasi logika
- Contoh:

```
<?php  
    $benar=true;  
    $salah=false;  
  
    echo "benar = $benar, salah = $salah";  
    // hasil output: benar = 1, salah =  
  
    $x = FALSE; // false  
    $x = ""; // string kosong dianggap false  
    $x = " "; // string dengan karakter adalah true;  
    $x = 0; // false  
    $x = 1; // true  
?>
```

Tampilan di web browser



Apa itu Variabel?

- Variabel adalah suatu “*wadah*” yang digunakan untuk **menyimpan suatu data atau nilai**. Kita bisa menyimpan berbagai macam data dari berbagai macam tipe, misalkan kita menyimpan data teks untuk nama mahasiswa, atau data desimal untuk nilai ipk mahasiswa, dan seterusnya

Variabel

Aturan :

- Diawali dengan karakter dolar (\$) dan diikuti dengan nama pengenal
- Nama pengenal dimulai dengan huruf atau garis bawah (_), tidak boleh diawali dengan angka
- Tidak boleh mengandung spasi dan terdiri dari minimal satu karakter
- Bersifat case sensitive (huruf besar dan kecil dibedakan)
- Tidak diperlukan deklarasi type variable, tetapi type variable akan mengikuti type nilai yg diberikan
- Setiap variabel terbentuk dalam tipe data variant (dapat menampung jenis data apapun)

Variabel

Contoh penulisan Variabel

```
<?php  
    $i;  
    $nama;  
    $Umur;  
    $_lokasi_memori;  
    $ANGKA_MAKSIMUM;  
?>
```

Contoh penulisan yang salah

```
<?php  
    $4ever; //variabel tidak boleh diawali dengan angka  
    $_salah satu; //varibel tidak boleh mengandung spasi  
    $nama*^; //variabel tidak boleh mengandung karakter  
    khusus: * dan ^  
?>
```

Peraturan Penamaan Variabel

- Untuk menyimpan suatu data pada variabel, kita perlu memberi nama terhadap variabel tersebut.
- Dalam PHP, terdapat beberapa peraturan dalam pemberian nama variabel sebagaimana pada tabel berikut:
- Penamaan suatu variabel disarankan agar menggunakan nama yang sesuai dengan tugas variabel tersebut

Peraturan	Keterangan
diawali tanda <code>\$</code>	Setiap nama variabel dalam bahasa pemrograman PHP didefinisikan dengan tanda <code>\$</code> lalu diikuti oleh nama variabel itu sendiri
nama diawali huruf atau <i>underscore</i> (<code>_</code>)	Nama variabel PHP harus diawali huruf, atau tanda underscore (<code>_</code>). Kita bisa membuat variabel dengan nama <code>\$_nilai</code> atau <code>\$nilai123</code> akan tetapi tidak bisa membuat variabel dengan nama <code>\$1nilai</code>
case sensitive	PHP membedakan huruf besar dan kecil dalam penamaan variabel. Maka variabel <code>\$nilai</code> , <code>\$nilAi</code> dan juga <code>\$nIlAi</code> dianggap 3 variabel yang berbeda
hanya boleh huruf dan angka	nama variabel hanya boleh tersusun dari huruf <code>[a-z]</code> atau <code>[A-Z]</code> , dan juga angka <code>[0-9]</code> . Kita tidak bisa memberi nama variabel misalkan dengan tanda <code>^</code> atau <code>&</code> dan lain sebagainya. Kita juga tidak bisa menggunakan <code>spasi</code> dalam penamaan variabel di PHP
inisiasi dengan tanda <code>=</code>	Kita bisa memberi nilai terhadap suatu variabel dengan menggunakan operator sama dengan (<code>=</code>). Contoh: <code>\$nama = "Nurul Huda";</code>

Pembuatan Variabel

- Untuk pembuatan variabel dalam PHP, kombinasi sintaksnya adalah sebagai berikut:

```
<?php  
  
$namaVariabel = [nilai variabel];
```

- Bagian **\$namaVariabel** adalah nama yang anda berikan untuk variabel tersebut, sedangkan **[nilai variabel]** adalah nilai yang akan anda masukkan ke dalam variabel tersebut.
- Nilai variabel ini lah yang menentukan tipe data dari variabel itu sendiri. Jika anda memasukkan angka, maka tipe datanya menjadi tipe data numeric (integer/float), jika anda masukkan true/false maka jadinya adalah boolean, dan seterusnya.

Tipe Data Integer

- Tipe data integer adalah tipe data yang digunakan untuk menyimpan bilangan bulat. Contoh :

```
<?php  
# inisiasi dan inisialisasi variabel  
$a = 10;  
$b = 5;  
$c = $a + 5;  
$d = $b + (10 * 5);  
$e = $d - $c;  
  
# Tampilkan data dengan perintah echo  
echo "Variabel a: {$a} <br>";  
echo "Variabel b: {$b} <br>";  
echo "Variabel c: {$c} <br>";  
echo "Variabel d: {$d} <br>";  
echo "Variabel e: {$e} <br>";  
  
# mengetahui tipe data dari variabel  
var_dump($e);
```

Hasilnya :

```
Variable a: 10  
Variable b: 5  
Variable c: 15  
Variable d: 55  
Variable e: 40  
int(40)
```

Tipe Data Float

- Untuk tipe data float, caranya sama saja dengan integer. Hanya saja, ia menerima data desimal dengan angka . sebagai pembaginya. Contoh sebagai berikut :

```
<?php

$nilaiMatematika = 5.1;
$nilaiIPA = 6.7;
$nilaiBahasaIndonesia = 9.3;

# hitung nilai rata-rata
$rataRata = ($nilaiMatematika + $nilaiIPA + $nilaiBahasaIndonesia) / 3;

# Tampilkan data

echo "Matematika: {$nilaiMatematika} <br>";
echo "IPA: {$nilaiIPA} <br>";
echo "Bahasa Indonesia: {$nilaiBahasaIndonesia} <br>";
echo "Rata-rata: {$rataRata} <br>";

# lihat tipe data dari variabel $rataRata
var_dump($rataRata);
```

Hasilnya :

```
Matematika: 5.1
IPA: 6.7
Bahasa Indonesia: 9.3
Rata-rata: 7.03333333333333
float(7.03333333333333)
```

Tipe Data Boolean

- Tipe data boolean hanya bisa menampung nilai true atau false.
- Tipe data ini adalah tipe data yang paling simpel, akan tetapi butuh logika yang kuat untuk bisa memanfaatkannya dengan benar.

```
<?php  
  
$apakahSiswaLulus = true;  
$apakahSiswaSudahUjian = false;  
  
var_dump($apakahSiswaLulus);  
echo "<br>";  
var_dump($apakahSiswaSudahUjian);
```

Hasilnya :

```
bool(true)  
bool(false)
```

Tipe Data String

- Tipe data string adalah tipe data yang digunakan untuk menyimpan teks. Semua teks tersebut diapit oleh tanda petik satu ('') atau tanda petik dua ("").

```
<?php

$namaDepan = "Ibnu"; # pakai tanda petik dua
$namaBelakang = 'Jakaria'; # pakai tanda petik satu

# menggabungkan dua variabel dengan tanda
# petik dua
$namaLengkap = "{$namaDepan} {$namaBelakang}";

# anda juga bisa menggabungkan string dengan menggunakan tanda titik (.)
$namaLengkap2 = $namaDepan . ' ' . $namaBelakang;

# [Tampilkan Data]
# kita bisa memasukkan variabel lain jika menggunakan tanda petik dua
echo "Nama Depan: {$namaDepan} <br>";
# ada pun jika pakai tanda petik satu, kita tidak bisa memasukkan variabel
# di dalam string akan tetapi menggabungkannya dengan operator titik (.)
echo 'Nama Belakang: ' . $namaBelakang . '<br>';

echo $namaLengkap;
```

Hasilnya :

```
Nama Depan: Ibnu
Nama Belakang: Jakaria
Ibnu Jakaria
```

Tipe Data Array

- Tipe data array berfungsi untuk menyimpan himpunan data. Himpunan data tersebut diapit oleh tanda kurung siku ([]).
- Sebagai contoh, saya memiliki 3 mahasiswa, dan saya ingin menyimpan ketiga nama mahasiswa dalam variabel. Maka saya bisa melakukan hal tersebut dengan menggunakan tipe data array sebagai berikut:

```
<?php  
  
$listMahasiswa = ["Wahid Abdullah", "Elmo Bachtiar", "Lendis Fabri"];
```

Lanj.

- Untuk mengakses isi dari variabel array, kita bisa menggunakan indeks. Indeks dimulai dari 0. Sehingga jika saya akan menampilkan nama pertama dari variabel \$listMahasiswa, saya akan menggunakan indeks 0 seperti di bawah:

```
<?php  
  
echo $listMahasiswa[0]; // "Wahid Abdullah"
```

Operator Variable

Operator adalah simbol-simbol yang digunakan untuk melakukan operasi terhadap suatu nilai dan variabel.

- Ada 6 Jenis operator dalam pemrograman PHP yang harus kita ketahui:
- Operator Aritmatika;
- Operator Penugasan atau *Assignment*;
- Operator *Increment & Decrement*;
- Operator Relasi atau pembanding;
- Operator Logika;
- dan Operator Ternary.

Operator Aritmatika

- Operator aritmatika merupakan operator untuk melakukan operasi aritmatika.
- Operator aritmatika terdiri dari:

Nama Operator	Simbol
Penjumlahan	+
Pengurangan	-
Perkalian	*
Pemangkatan	**
Pembagian	/
Sisa Bagi	%

Operator Aritmatika

Contoh Operator Aritmatika :

```
<?php  
  
$a = 5;  
$b = 2;  
  
// penjumlahan  
$c = $a + $b;  
echo "$a + $b = $c";  
echo "<hr>";  
  
// pengurangan  
$c = $a - $b;  
echo "$a - $b = $c";  
echo "<hr>";  
  
// Perkalian  
$c = $a * $b;  
echo "$a * $b = $c";  
echo "<hr>";  
  
// Pembagian  
$c = $a / $b;  
echo "$a / $b = $c";  
echo "<hr>";  
  
// Sisa bagi  
$c = $a % $b;  
echo "$a % $b = $c";  
echo "<hr>";  
  
// Pangkat  
$c = $a ** $b;  
echo "$a ** $b = $c";  
echo "<hr>";
```

Operator Penugasan

- Operator berikutnya yang harus kamu ketahui adalah operator penugasan atau *assignment*.
- Operator ini adalah operator untuk memberikan tugas kepada variabel.
- Biasanya digunakan untuk mengisi nilai.

Nama Operator	Simbol
Pengisian Nilai	=
Pengisian dan Penambahan	+=
Pengisian dan Pengurangan	-=
Pengisian dan Perkalian	*=
Pengisian dan Pemangkatan	**=
Pengisian dan Pembagian	/=
Pengisian dan Sisa bagi	%=
Pengisian dan Peggabungan (string)	.=

Apa bedanya dengan operator aritmatika?

- Bisa dibilang, operator penugasan adalah bentuk yang lebih sederhana dari ekspresi Aritmetika
- Sebagai contoh:

```
$speed = 83;  
  
// ini operotor aritmatika  
$speed = $speed + 10;  
  
// maka nilai speed akan samadengan 83 + 10 = 93  
  
// ini operator penugasan  
$speed += 10;  
  
// sekarang nilai speed akan menjadi 93 + 10 = 103
```

Opearator Increment & Decrement

- Operator increment dan decrement merupakan operator yang digunakan untuk menambah +1 (tambah satu) dan mengurangi -1 (kurangi dengan satu).
- Opertor increment menggunakan simbol ++, sedangkan decrement menggunakan simbol --. Contohnya :

```
$score = 0;  
  
$score++;  
$score++;  
$score++;  
  
echo $score;
```

Operator Relasi

- Operator relasi adalah operator untuk membandingkan dua buah nilai.
- Berikut ini daftar operator relasi:

Nama Operator	Simbol
Lebih Besar	>
Lebih Kecil	<
Sama Dengan	== atau ===
Tidak Sama dengan	!= atau !==
Lebih Besar Sama dengan	>=
Lebih Kecil Sama dengan	<=

Lanj.

- Hasil operasi dari operator relasi akan menghasilkan nilai dengan tipe data boolean, yaitu true (benar) dan false (salah).
- Dari hasil di samping kita mendapatkan nilai 1 untuk true sedangkan false tidak ditampilkan atau 0.

```
<?php  
  
$a = 6;  
$b = 2;  
  
// menggunakan operator relasi lebih besar  
$c = $a > $b;  
echo "$a > $b: $c";  
echo "<hr>";  
  
// menggunakan operator relasi lebih kecil  
$c = $a < $b;  
echo "$a < $b: $c";  
echo "<hr>";  
  
// menggunakan operator relasi lebih sama dengan  
$c = $a == $b;  
echo "$a == $b: $c";  
echo "<hr>";  
  
// menggunakan operator relasi lebih tidak sama dengan  
$c = $a != $b;  
echo "$a != $b: $c";  
echo "<hr>";  
  
// menggunakan operator relasi lebih besar sama dengan  
$c = $a >= $b;  
echo "$a >= $b: $c";  
echo "<hr>";  
  
// menggunakan operator relasi lebih kecil sama dengan  
$c = $a <= $b;  
echo "$a <= $b: $c";  
echo "<hr>";
```

6 > 2: 1 true

6 < 2: false

6 == 2: false

6 != 2: 1 true

6 >= 2: 1 true

6 <= 2: false

Operator Logika

- Operator logika adalah operator untuk melakukan operasi logika seperti AND, OR, dan NOT.
- Operator logika terdiri dari:

Nama Operator	Simbol
Logika AND	&&
Logika OR	
Negasi/kebalikan/ NOT	!

Lanj

- Operator logika sama seperti operator relasi, ia akan menghasilkan nilai dengan tipe data boolean.
- Perhatikanlah hasil yang di dapatkan ketika menggunakan operator **&&** (AND), **||** (OR), dan **!** (NOT).

```
<?php

$a = true;
$b = false;

// variabel $c akan bernilai false
$c = $a && $b;
printf("%b && %b = %b", $a,$b,$c);
echo "<br>";

// variabel $c akan bernilai true
$c = $a || $b;
printf("%b || %b = %b", $a,$b,$c);
echo "<br>";

// variabel $c akan bernilai false
$c = !$a;
printf("!%b = %b", $a, $c);
echo "<br>";
```

$$1 \&\& 0 = 0$$

$$1 \parallel 0 = 1$$

$$!1 = 0$$

Lanj

- Operator `&&` akan menghasilkan true apabila nilai kiri dan kanan bernilai true. Sedangkan operator `||` akan menghasilkan false saat nilai kiri dan kanan bernilai false.
- Coba cek kembali hukum logika AND, OR, dan NOT.

AND		Hasil
true	true	true
true	false	false
false	true	false
false	false	false

OR		Hasil
true	true	true
true	false	true
false	true	true
false	false	false

NOT		Hasil
true		false
false		true

www.petanikode.com

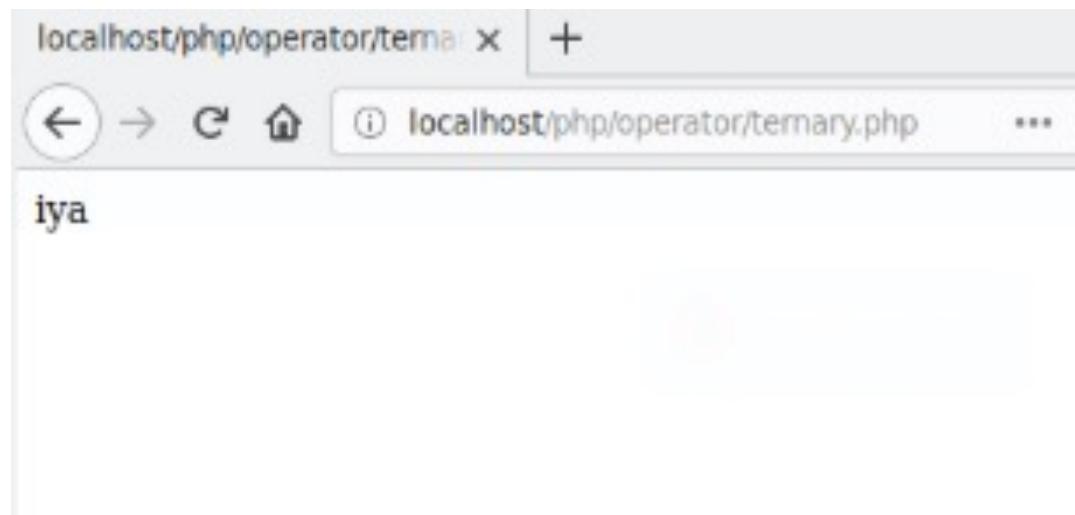
Operator Ternary

- Operator ternary adalah operator untuk membuat sebuah kondisi. Simbol yang digunakan adalah tanda tanya (?) dan titik dua (:).

Operator Ternary
kamu suka aku ? ya : tidak;
jawaban benar jawaban salah

Contoh

```
<?php  
  
$suka = true;  
  
// menggunakan operator ternary  
$jawab = $suka ? "iya": "tidak";  
  
// menampilkan jawaban  
echo $jawab;
```



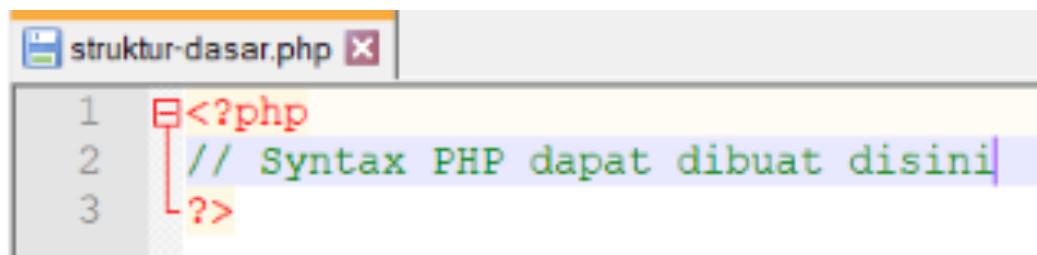
Struktur PHP

PHP Pemrosesan data berbasis server side

Didalam membangun sebuah web terdiri dari beberapa komponen diantaranya adalah **HTML** (HyperText Markup Language) sebagai bahasa web yang pemrosesannya di proses di sisi klien. Kemudian ada juga **CSS** (Cascading Style Sheets) yang sama dengan HTML pemrosesannya dari sisi klien. Sedangkan **PHP** pemrosesan data berada di sisi server (server side).

Struktur Dasar penulisan PHP

- PHP mempunyai struktur yang sederhana. Syntax PHP dimulai dengan tanda **<?php** dan di akhiri dengan **?>**



```
struktur-dasar.php
1 <?php
2 // Syntax PHP dapat dibuat disini
3 ?>
```

Ekstensi file PHP bisa dibuat dengan ekstensi .php (dot php) seperti pada contoh gambar diatas membuat nama filenya **struktur-dasar.php**

Komentar pada PHP

- Saat kita membuat baris-baris program didalam PHP kita dapat menggunakan fungsi komentar untuk menjelaskan maksud dari setiap baris atau function yang kita buat. Sehingga terdapat dokumentasi yang jelas dari setiap syntax PHP yang kita buat.
- Komentar di PHP diawali dengan tanda `//`. PHP tidak akan mengeksekusi setiap karakter yang terdapat dibelakang tanda `//`. Karena PHP akan membaca hal tersebut merupakan komentar. Contoh seperti dibawah ini:

Sensivitas didalam PHP

- Didalam PHP fungsi-fungsi seperti if, for, while, echo, print dan lain-lain tidak case sensitive. Penggunaan fungsi-fungsi tersebut dapat kita gunakan dengan huruf besar maupun kecil. Contohnya fungsi echo pada PHP ketika kita membuat fungsi echo dan ECHO, PHP akan membaca kedua syntax tersebut adalah sama.
- Sedangkan penamaan variabel di dalam PHP bersifat adalah case sensitive.

Menggabungkan PHP Dalam HTML

- Secara default, dokumen PHP berakhir dengan ekstensi .php. Tapi, beberapa pengembang web memilih untuk menggunakan ekstensi .html dimana html juga bisa diurai oleh prosesor PHP, biasanya karena mereka ingin menyembunyikan fakta bahwa mereka menggunakan PHP
- Dokumen PHP akan menampilkan output seperti dokumen HTML. File index.html dan index.php akan memiliki tampilan yang identik.

Contoh

```
<!DOCTYPE html>
<html>
  <head>
    <title><?php echo "Belajar PHP" ?></title>
  </head>
  <body>
    <?php
      echo "saya sedang belajar PHP<br>";
      echo "<p>Belajar PHP hingga jadi master</p>";
    ?>
  </body>
</html>
```

Kesimpulan

Kesimpulan

- Tipe data **berperan penting** dalam Bahasa pemrograman karena berkaitan dengan variable yang digunakan dalam membuat perangkat lunak.
- Struktur pemrograman php membantu dalam memahami proses eksekusi program dan memudahkan dalam pembuatan sebuah perangkat lunak.

Referensi / Bacaan Lebih Lanjut

Referensi / Bacaan Lebih Lanjut

- <https://kelasprogrammer.com/syntax-dasar-php-belajar-struktur-php/>
- <https://www.w3schools.com/php/>

Motivasi



Inovasi membedakan antara pemimpin dan pengikut

Aspersepsi

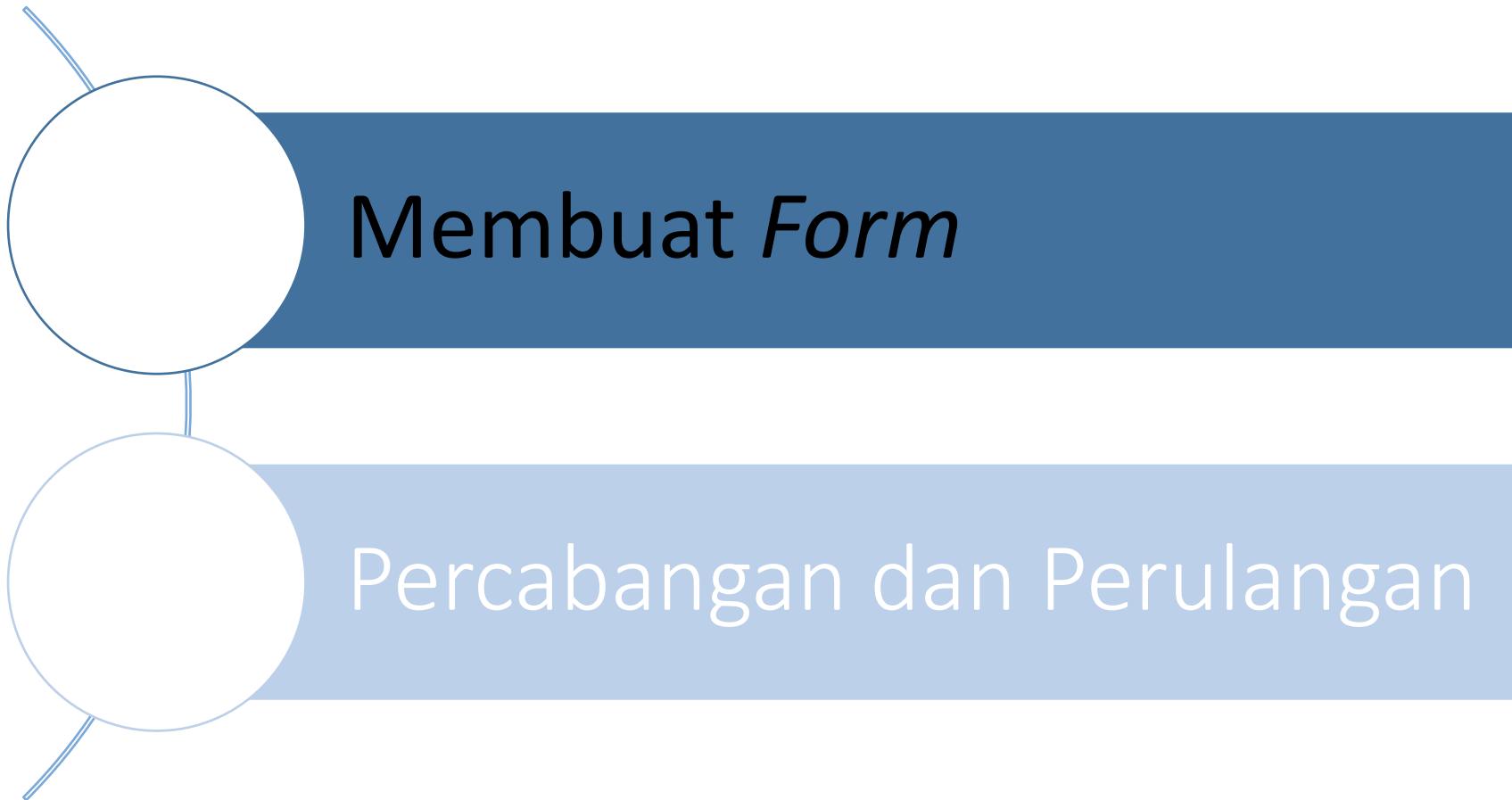


Membuat Program Sederhana

Ringkasan Mata Pelatihan

- Unit Kompetensi Acuan: Mengimplementasikan Pemrograman Terstruktur
- Kode Unit Kompetensi Acuan: J.620100.017.02
- Deskripsi singkat: Mata pelatihan ini menentukan kompetensi, pengetahuan dan sikap kerja yang diperlukan dalam membuat program terstruktur atau prosedural.
- Tujuan Pembelajaran: Peserta dapat membuat program terstruktur atau procedural sesuai dengan paradigma yang berlaku.

Agenda



Membuat Form

Mengenal Struktur Dasar Form HTML

```
<form action="proses.php" method="get">
    Nama: <input type="text" name="nama" />
    <br />
    E-Mail: <input type="text" name="email" />
    <br />
    <input type="submit" value="Proses Data" >
</form>
```

- Dari struktur dasar diatas tersebut, di dalam tag **<form>** terdapat 2 buah atribut. Yakni atribut **action** dan atribut **method**

Atribut Action

- Atribut pertama adalah **action**. Atribut **action** ini diisi dengan nilai berupa alamat halaman PHP dimana kita akan memproses isi form tersebut.
- Isi atribut action adalah alamat dari halaman PHP. Karena atribut *action* pada contoh diatas ditulis **action="proses.php"**, maka file **proses.php** harus berada di dalam 1 folder dengan halaman HTML yang berisi form ini. Namun alamat **proses.php** ini dapat dirubah tergantung dimana file tersebut berada, misalnya menjadi alamat relatif seperti **action="file_php/proses.php"**

Atribut Method

- Atribut kedua yang berkaitan dengan pemrosesan form HTML adalah atribut method. Atribut inilah yang akan menentukan bagaimana cara form ‘dikirim’ ke dalam halaman proses.php. Nilai dari atribut method hanya bisa diisi dengan 1 dari 2 pilihan, yakni get atau post.

Mengirimkan Nilai Form HTML ke dalam PHP

- Untuk mengambil nilai form HTML, PHP menyediakan 2 buah variabel global yaitu variabel **`$_GET`** dan **`$_POST`**. Kita menggunakan variabel **`$_GET`** jika pada saat pembuatan form menggunakan atribut **`method=get`**, dan menggunakan variabel **`$_POST`** jika form dibuat dengan **`method=post`**.
- Kedua variabel ini sebenarnya adalah array, sehingga cara mengakses nilai dari form adalah dengan cara: **`$_GET['nama_objek_form']`**.
- '***nama_objek_form***' adalah nilai dari atribut **`name`** di dalam **`form`**. Jika kita memiliki tag dengan kode HTML **`<input type="text" name="nama" />`**, maka untuk mengakses nilainya adalah dengan **`$_GET['nama']`**, dan untuk tag **`<input type="text" name="email" />`** diakses dengan nilai **`$_GET['email']`**.

Contoh

Belajar.php

```
<!DOCTYPE html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>Belajar Form PHP</title>
</head>

<body>
    <h2>Tutorial Belajar Form HTML - PHP </h2>
    <form action="proses.php" method="post">
        Nama: <input type="text" name="nama" />
        <br />
        E-Mail: <input type="text" name="email" />
        <br />
        <input type="submit" value="Proses Data" >
    </form>
</body>
</html>
```

Proses.php

```
<?php
    echo $_POST['nama'];
    echo "<br />";
    echo $_POST['email'];
?>
```

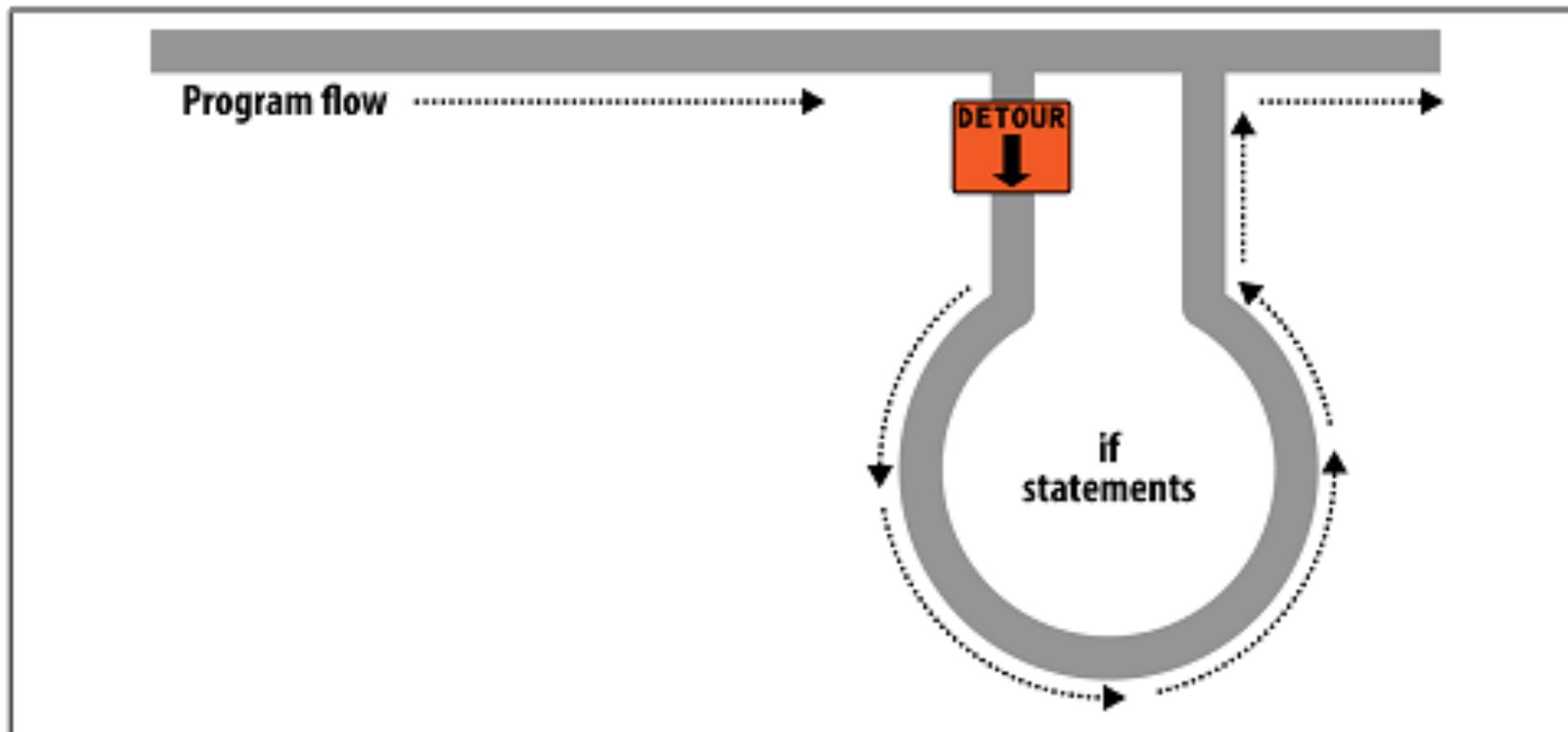
Percabangan dan Perulangan

Pencabangan

- Memungkinkan program melakukan aksi yang berbeda berdasarkan kondisi tertentu.
 - if
 - if...else
 - if...elseif...else
 - switch



Percabangan if



Percabangan if

- Menjalankan suatu perintah jika kondisi benar

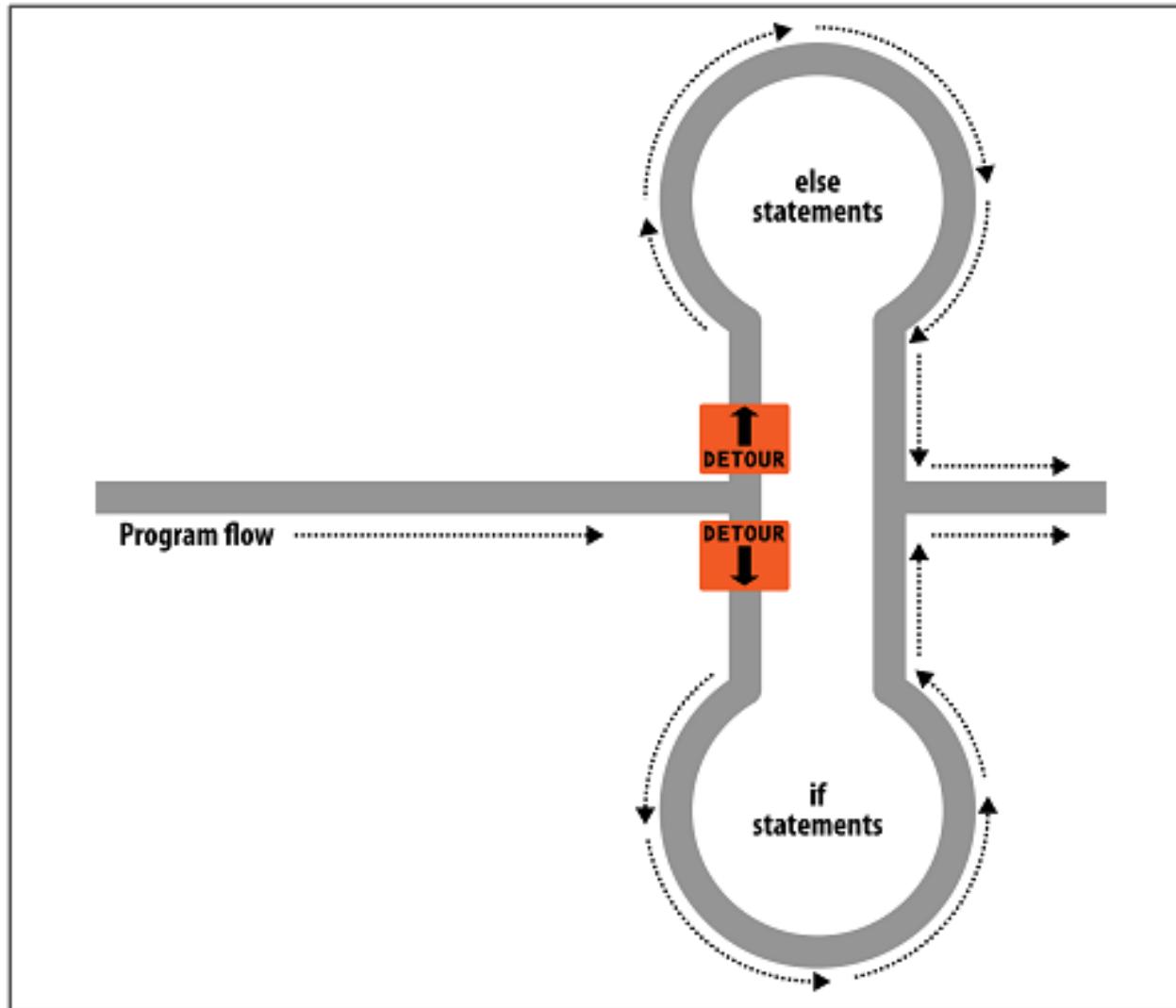
```
if (kondisi benar) {  
    //perintah yang dijalankan jika kondisi benar  
}
```

Contoh

```
echo "<br>";  
$angka = 3;  
if($angka >= 0){  
    echo "Bilangan Positif";  
}
```

Bilangan Positif

Percabangan If...else



Percabangan If...else

- Menjalankan suatu perintah jika kondisi benar dan perintah lain jika kondisi salah

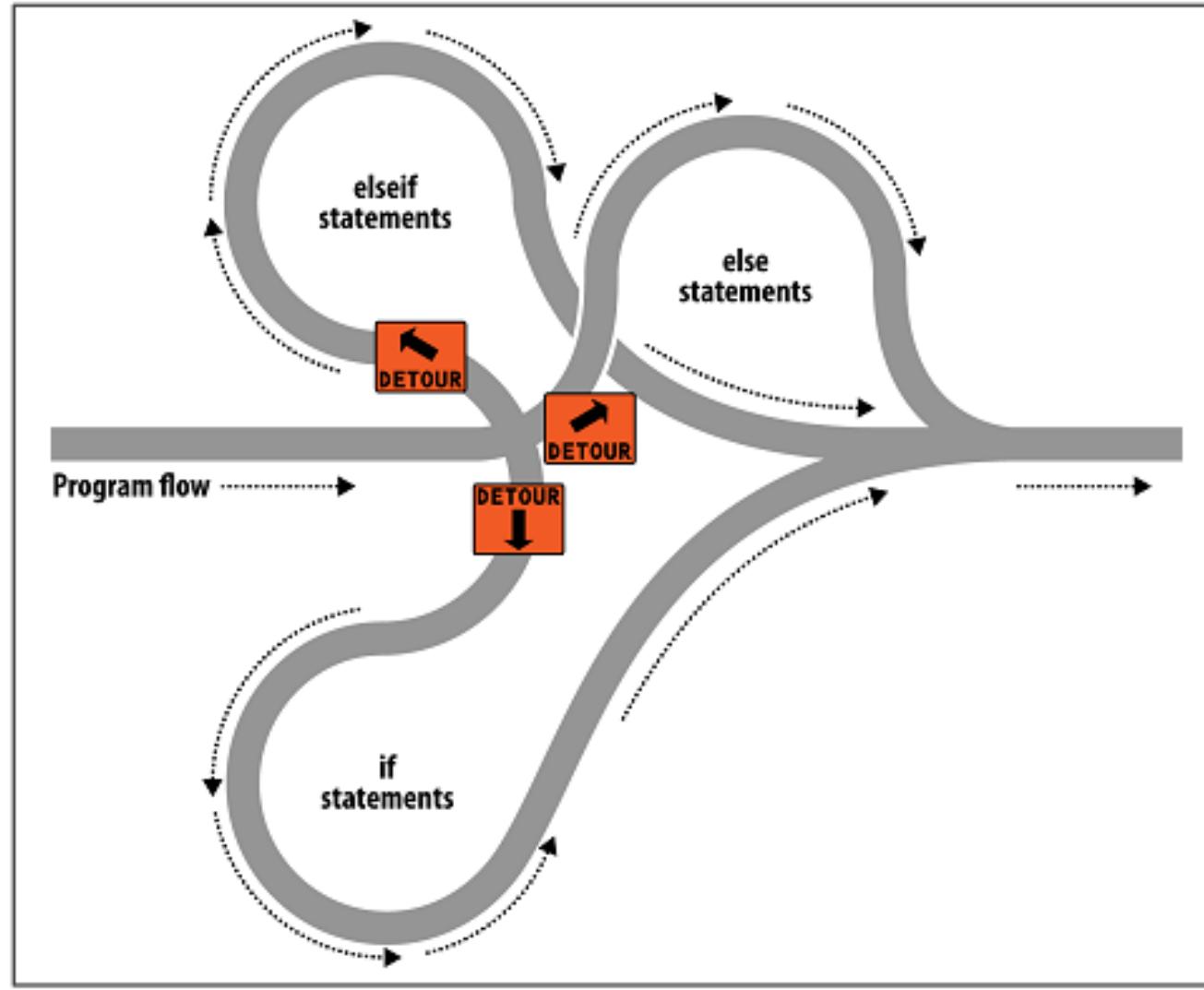
```
if (kondisi benar) {  
    //perintah yang dijalankan jika kondisi benar  
} else {  
    //perintah yang dijalankan jika kondisi salah  
}
```

Contoh

```
echo "<br>";  
$angka = -1;  
if($angka >= 0){  
    echo "Bilangan Positif";  
} else{  
    echo "Bilangan Negatif";  
}
```

Bilangan Negatif

Percabangan if...elseif...else



Percabangan if...elseif...else

- Menjalankan perintah yang berbeda untuk lebih dari dua kondisi

```
if (kondisi1 benar) {  
    //perintah yang dijalankan jika kondisi1 benar  
} elseif (kondisi2 benar) {  
    //perintah yang dijalankan jika kondisi2 benar  
} else{  
    //perintah yang dijalankan jika semua kondisi  
    salah  
}
```

Contoh

```
echo "<br>";  
$lampa = "hijau";  
if($lampa == "hijau"){  
    echo "jalan";  
} elseif ($lampa == "merah"){  
    echo "berhenti";  
} else{  
    echo "hati-hati";  
}
```

jalan

Video Learning Konsep Percabangan *if-else statement*

- Berikut merupakan *link* video mengenai konsep percabangan *if-else statement*

<https://youtu.be/OPU5Xei20U8>

Percabangan Switch/Case

- Menjalankan perintah yang berbeda berdasarkan pada kondisi yang berbeda
- Gunakan switch untuk memilih salah satu dari banyak kode yang akan dieksekusi

Percabangan Switch/Case

```
switch (n) {  
    case label1:  
        //perintah yang dijalankan jika n=label1  
        break;  
    case label2:  
        //perintah yang dijalankan jika n=label2  
        break;  
    case label3:  
        //perintah yang dijalankan jika n=label3  
        break;  
    ...  
    default:  
        //perintah yang dijalankan jika n berbeda dari semua label  
}
```

Contoh

```
$halaman = "berita";
switch ($halaman) {
case "home":
    echo "Anda memilih home";
    break;
case "berita":
    echo "Anda memilih berita";
    break;
case "artikel":
    echo "Anda memilih artikel";
    break;
default:
    echo "Halaman yang ada cari tidak tersedia";
}
```

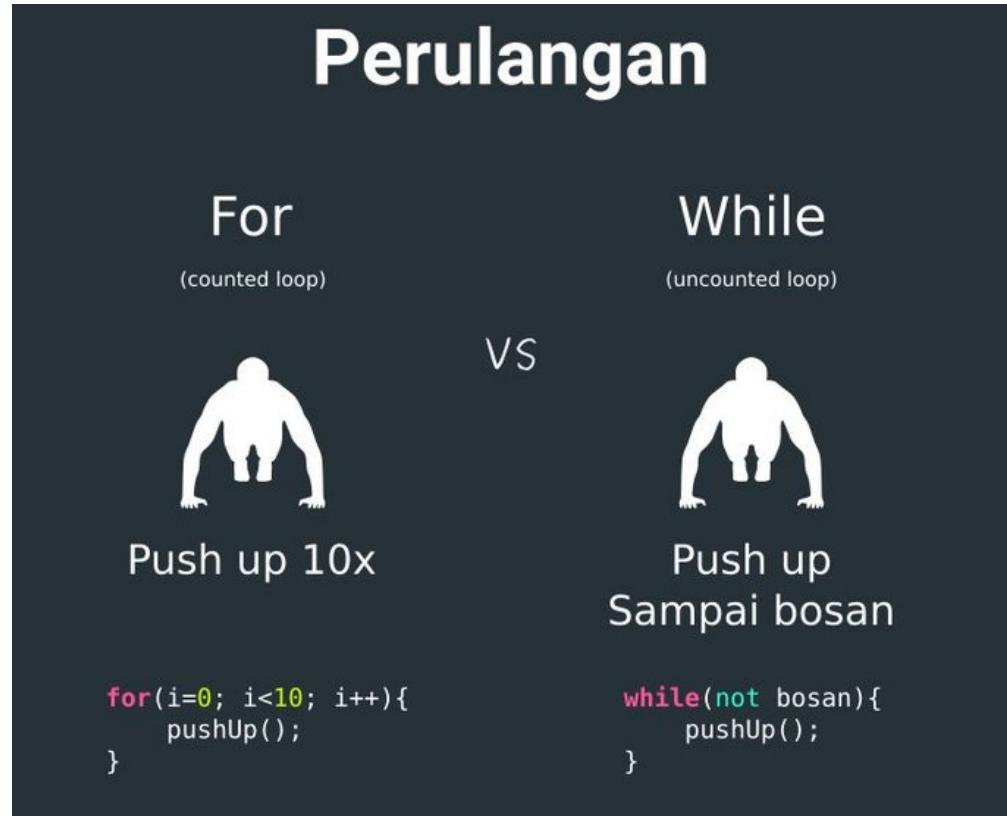
Perulangan (Loop)

Terdapat 4 komponen :

- . *inisialisasi* : kondisi awal perulangan, inisialisasi (pemberian nilai awal) pada variabel counter
- . *condition* : kondisi yang harus dipenuhi agar perulangan dijalankan
- . *increment* : bagian untuk memproses variable counter agar bisa memenuhi kondisi akhir perulangan
- . *statement* : baris perintah yang akan diulang

Perulangan

- Memungkinkan program mengulangi sederetan aksi berdasarkan kondisi tertentu.
- Ada dua jenis perulangan dalam pemrograman:
 - ✓ Counted loop;
 - ✓ Uncounted loop.



Lanj.

Pada PHP ada 4 jenis perulangan yang bisa kita gunakan:

- Perulangan For
- Perulangan While
- Perulangan Do/While
- Perulangan Foreach

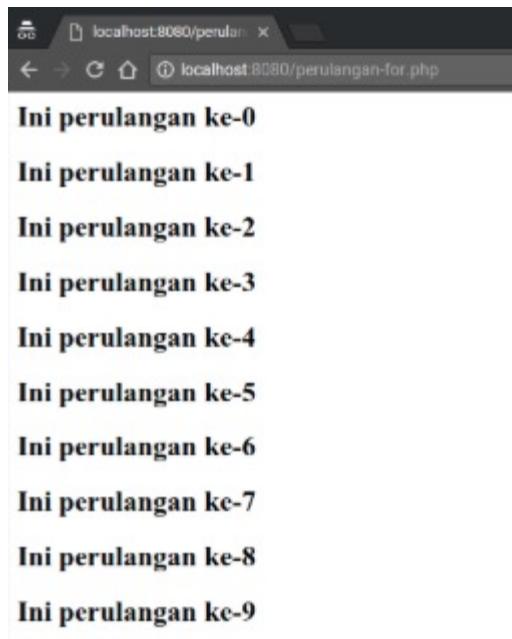
Perulangan For

- Perulangan *For* adalah perulangan yang termasuk dalam *counted loop*, karena kita bisa menentukan jumlah perulangannya.

```
for (init counter; test counter; increment counter) {  
    //perintah yang dijalankan  
}
```

Contoh

```
<?php  
  
for($i = 0; $i < 10; $i++){  
    echo "<h2>Ini perulangan ke-$i</h2>";  
}  
  
?>
```



- Variabel \$i dalam perulangan For berfungsi sebagai counter yang menghitung berapa kali ia akan mengulang.
- Hitungan akan dimulai dari nol (0), karena kita memberikan nilai \$i = 0.
- Lalu, perulangan akan diulang selama nilai \$i lebih kecil dari 10. Artinya, perulangan ini akan mengulang sebanyak 10x.
- Maksud dari \$i++ adalah nilai \$i akan ditambah 1 disetiap kali melakukan perulangan.

while Loop

- . Dikenal juga dengan *indeterminate loop*.
- . Penentuan jumlah perulangan tidak ditentukan sebelumnya.
- . Perulangan akan dilakukan terus menerus sampai dengan kondisi yang menjadi prasyarat bernilai *false*.

Perulangan While

- Perulangan *while* adalah perulangan yang termasuk dalam *uncounted loop*. Karena biasanya digunakan untuk mengulang sesuatu yang belum jelas jumlah pengulangannya.
- Namun, perulangan *while* juga bisa digunakan seperti perulangan *for* sebagai *counted loop*.

```
while (kondisi benar) {  
    //perintah yang dijalankan selama kondisi benar  
}
```

while Loop

Format:

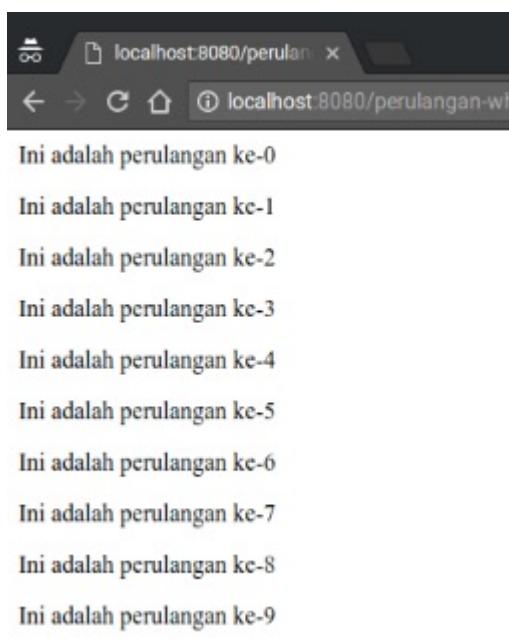
```
inisialisasi;  
while(condition)  
{  
    statement;  
    ...  
    increment;  
}
```

Contoh:

```
<?php  
    $i=1;  
    while($i<=10) {  
        echo "Perulangan ke-".$i."<br />";  
        $i++;  
    }  
?>
```

Contoh

```
<?php  
  
$ulangi = 0;  
  
while($ulangi < 10){  
    echo "<p>Ini adalah perulangan ke-$ulangi</p>";  
    $ulangi++;  
}  
  
?>
```



- Perulangan while akan terus mengulang selama nilai \$ulangi lebih kecil dari 10.
- Lalu di dalam perulangan kita melakukan increment nilai \$ulangi dengan \$ulangi++. Artinya: Tambah 1 disetiap pengulangan.
- Hati-hati, jangan sampai lupa menambahkan increment, atau kode yang akan mempengaruhi pengulangan. Karena kalau tidak, pengulangannya tidak akan pernah berhenti dan akan membuat komputer kita hang.

Video Learning Konsep *Looping* atau Perulangan (*For*) dalam Pemrograman

- Berikut merupakan *link* video mengenai konsep *looping* atau perulangan (*For*) dalam pemrograman

<https://youtu.be/u44gnktW4-U>

do-while Loop

- . Prinsip kerja sama dengan perulangan `while`.
- . Perbedaannya : pengecekan kondisi perulangan dilakukan pada akhir perulangan
- . Karena pengecekan kondisi dilakukan di akhir perulangan, maka walaupun kondisi adalah FALSE, perulangan akan tetap dijalankan minimal 1 kali.

Perulangan Do/While

- Perulangan *Do/While* sama seperti perulangan *while*. Ia juga tergolong dalam *uncontrolled loop*.
- Perbedaan *Do/While* dengan *while* terletak pada cara iya memulai pengulangan.

```
do {  
    //perintah yang dijalankan  
} while (kondisi benar)
```

do-while Loop

Format:

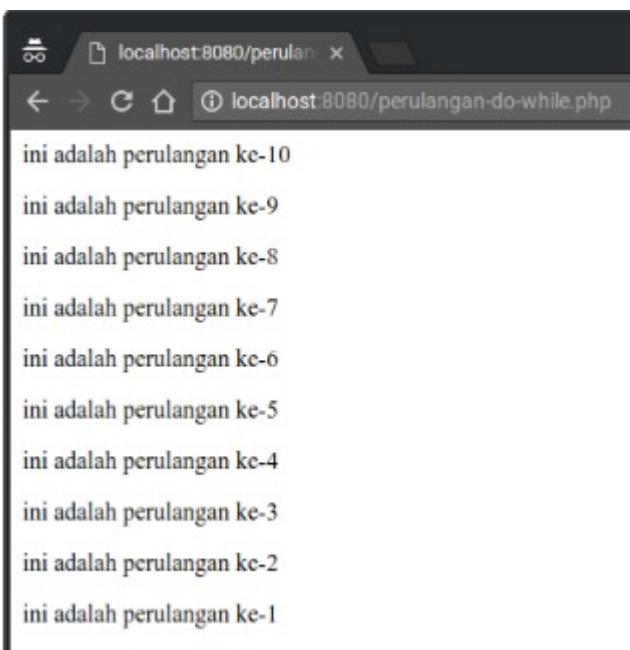
```
inisialisasi;  
do  
{  
    statement;  
    ...  
    increment;  
} while(condition)
```

Contoh:

```
<?php  
    $i=1;  
    do{  
        echo "Perulangan ke-".$i."<br/>";  
        $i++;  
    } while($i<=10);  
?>
```

Contoh

```
<?php  
  
$ulangi = 10;  
  
do {  
    echo "<p>ini adalah perulangan ke-$ulangi</p>";  
    $ulangi--;  
} while ($ulangi > 0);  
  
?>
```



- Perulangan *Do/While* akan selalu melakukan pengulangan sebanyak 1 kali, kemudian melakukan pengecekan kondisi.
- Sedangkan perulangan *while* akan mengecek kondisi terlebih dahulu, baru melakukan pengulangan.

Perulangan Foreach

- Perulangan *foreach* sama seperti perulangan *for*. Namun, ia lebih khusus digunakan untuk mencetak array.

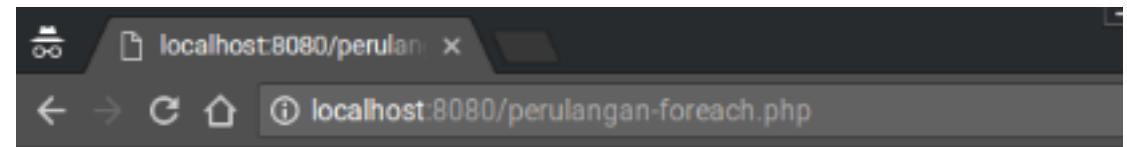
```
<?php  
foreach ($array as $data) {  
    echo $data;  
}
```

Contoh

```
<?php
$books = [
    "Panduan Belajar PHP untuk Pemula",
    "Membangun Aplikasi Web dengan PHP",
    "Tutorial PHP dan MySQL",
    "Membuat Chat Bot dengan PHP"
];

echo "<h5>Judul Buku PHP:</h5>";
echo "<ul>";
foreach($books as $buku){
    echo "<li>$buku</li>";
}
echo "</ul>";

?>
```



Judul Buku PHP:

- Panduan Belajar PHP untuk Pemula
- Membangun Aplikasi Web dengan PHP
- Tutorial PHP dan MySQL
- Membuat Chat Bot dengan PHP

Lompatan (Jumping)

Bentuk:

- .Break
- .Continue
- .Exit

Lompatan sebenarnya tidak termasuk dalam struktur control, tetapi turut digunakan juga pada struktur control percabangan dan perulangan untuk melompati / berpindah dari suatu bagian program/percabangan/perulangan.

Break

- . Fungsinya untuk keluar dari sebuah perulangan.
- . Contoh:

```
<?php
for ($i=1; $i<=6; $i++) {
    if ($i == 2) {
        break;
    }
    echo "Nilai i : ".$i."<br />";
}
?>
```

Continue

- . Digunakan untuk melewati satu siklus perulangan / iterasi.
- . Contoh:

```
<?php
for ($i=1; $i<=6; $i++) {
    if ($i == 3) {
        continue;
    }
    echo "Nilai i : ".$i."<br />";
}
?>
```

Exit()

- . Digunakan untuk keluar dari sebuah program. Walaupun terdapat statements lain dibawah baris kode `exit` yang belum dikerjakan.
- . Contoh:

```
<?php
for ($i=1; $i<=6; $i++) {
    if ($i == 3) {
        exit();
    }
    echo "Nilai i : ".$i."<br />";
}
// statement dibawah ini tidak akan dijalankan
echo "Contoh Penggunaan Exit Pada Program";
?>
```

Kesimpulan

Kesimpulan

- *Form* mempunyai 2 buah atribut yaitu action dan method yang harus disesuaikan dengan kebutuhan
- Semakin kompleks percabangan atau perulangan maka akan semakin memperlambat proses loading time server.

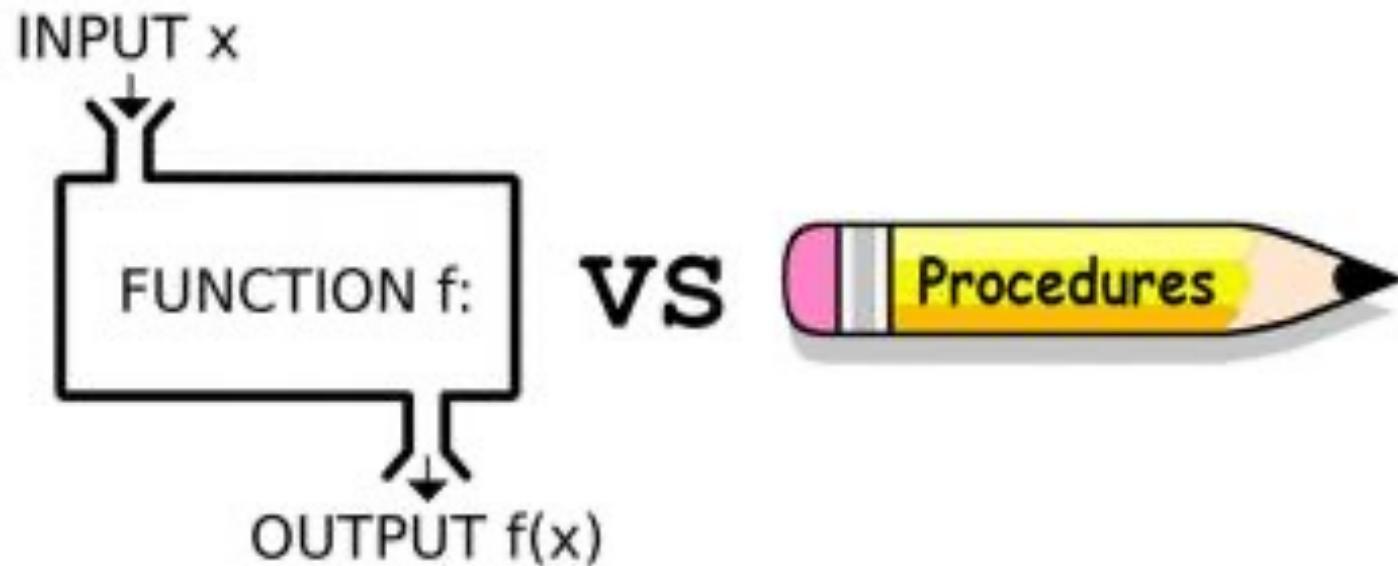
Referensi / Bacaan Lebih Lanjut

Referensi / Bacaan Lebih Lanjut

- <https://www.duniailkom.com/tutorial-form-php-cara-menampilkan-hasil-form-html-dengan-php/>
- <https://www.w3schools.com/php/>

Fungsi dan Prosedur

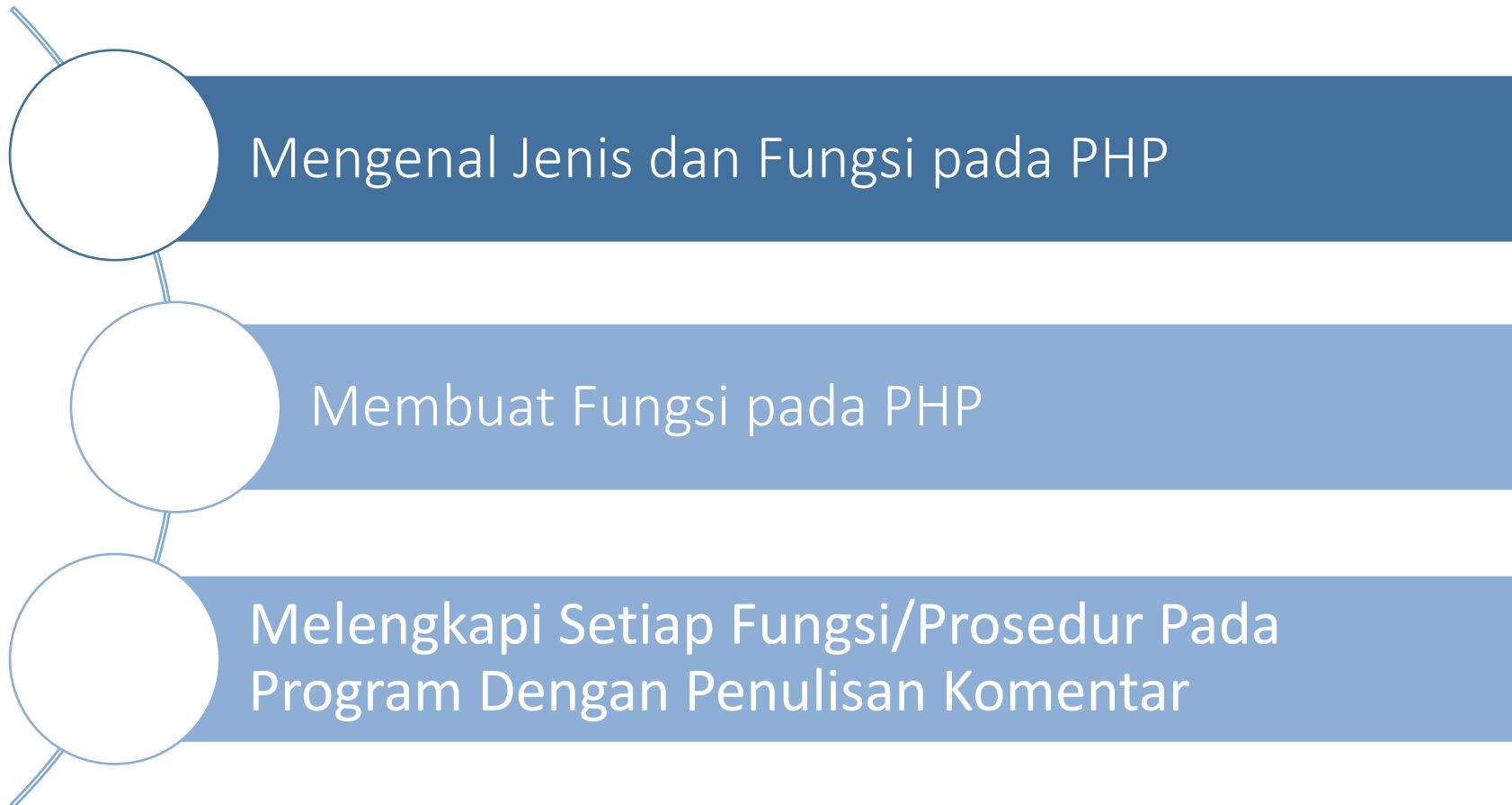
Aspersepsi



Ringkasan Mata Pelatihan

- Unit Kompetensi Acuan: Mengimplementasikan Pemrograman Terstruktur
- Kode Unit Kompetensi Acuan: J.620100.017.02
- Deskripsi singkat: Mata pelatihan ini menentukan kompetensi, pengetahuan dan sikap kerja yang diperlukan dalam membuat program terstruktur atau prosedural.
- Tujuan Pembelajaran: Peserta dapat membuat program terstruktur atau procedural sesuai dengan paradigma yang berlaku.

Agenda



Mengenal Jenis dan Fungsi pada PHP

Memahami Fungsi Pada PHP

- Fungsi adalah sekumpulan instruksi yang dibungkus dalam sebuah blok. Fungsi dapat digunakan ulang tanpa harus menulis ulang instruksi di dalamnya.

Jenis Fungsi Pada PHP

- Fungsi pada PHP dibagi menjadi dua yaitu *built-in function* yang merupakan fungsi bawaan PHP dan *user-defined function*, dimana kita membuat fungsi sendiri.
- Banyak sekali fungsi yang disediakan php, seperti substr(), dll yang dapat kita gunakan langsung, list lengkapnya dapat dilihat pada halaman: [PHP: Function and Method Listing – Manual](#)

Perbedaan Fungsi dan Prosedur pada PHP

- Prosedur: tidak mengembalikan nilai
- Fungsi: mengembalikan nilai
- Pada PHP, prosedur dan fungsi menggunakan sintaks function

Membuat Fungsi pada PHP

Struktur Penulisan Fungsi

```
function cetak ( $text, $callback ) {  
    echo $text;  
}  
  
Function Name  
Arguments  
} statement
```

Contoh

```
function perkenalan(){
    echo "Assalamualaikmu, ";
    echo "Perkenalkan, nama saya Ardianta<br/>";
    echo "Senang berkenalan dengan anda<br/>";
}
```

Fungsi yang sudah dibuat tidak akan menghasilkan apapun kalau tidak dipanggil. Kita dapat memanggil fungsi dengan menuliskan namanya.

Code Lengkapnya

```
<?php  
// membuat fungsi  
function perkenalan(){  
    echo "Assalamulaikmu, ";  
    echo "Perkenalkan, nama saya Ardianta<br/>";  
    echo "Senang berkenalan dengan anda<br/>";  
}  
  
// memanggil fungsi yang sudah dibuat  
perkenalan();  
  
echo "<hr>";  
  
// memanggilnya lagi  
perkenalan();  
?>
```

Assalamulaikmu,
Perkenalkan, nama saya Ardianta
Senang berkenalan dengan anda

Assalamulaikmu,
Perkenalkan, nama saya Ardianta
Senang berkenalan dengan anda

Fungsi dengan Parameter atau Arguments

- Supaya intruksi yang di dalam fungsi lebih dinamis, kita dapat menggunakan parameter untuk memasukkan sebuah nilai ke dalam fungsi. Nilai tersebut akan diolah di dalam fungsi.

```
<?php
// membuat fungsi
function perkenalan($nama, $salam){
    echo $salam.", ";
    echo "Perkenalkan, nama saya ".$nama."<br/>";
    echo "Senang berkenalan dengan anda<br/>";
}

// memanggil fungsi yang sudah dibuat
perkenalan("Muhardian", "Hi");

echo "<br>";

$saya = "Indry";
$ucapanSalam = "Selamat pagi";
// memanggilnya lagi
perkenalan($saya, $ucapanSalam);
?>
```

Hi,
Perkenalkan, nama saya Muhardian
Senang berkenalan dengan anda

Selamat pagi,
Perkenalkan, nama saya Indry
Senang berkenalan dengan anda

Paramter dengan Nilai Default

- Nilai *default* dapat kita berikan di parameter. Nilai *default* berfungsi untuk mengisi nilai sebuah parameter, kalau parameter tersebut tidak diisi nilainya.

```
<?php
// membuat fungsi
function perkenalan($nama, $salam="Assalamualaikum"){
    echo $salam . ", ";
    echo "Perkenalkan, nama saya " . $nama . "<br/>";
    echo "Senang berkenalan dengan anda<br/>";
}

// memanggil fungsi yang sudah dibuat
perkenalan("Muhardian", "Hi");

echo "<hr>";

$saya = "Indry";
$ucapanSalam = "Selamat pagi";
// memanggilnya lagi tanpa mengisi parameter salam
perkenalan($saya);
?>
```

Hi,
Perkenalkan, nama saya Muhardian
Senang berkenalan dengan anda

Assalamualaikum,
Perkenalkan, nama saya Indry
Senang berkenalan dengan anda

Fungsi yang Mengembalikan Nilai

- Hasil pengolahan nilai dari fungsi mungkin saja kita butuhkan untuk pemrosesan berikutnya. Oleh karena itu, kita harus membuat fungsi yang dapat mengembalikan nilai.

```
<?php
// membuat fungsi
function hitungUmur($thn_lahir, $thn_sekarang){
    $umur = $thn_sekarang - $thn_lahir;
    return $umur;
}

echo "Umur saya adalah ". hitungUmur(1994, 2015) ." tahun";
?>
```

Umur saya adalah 21 tahun

Memanggil Fungsi di dalam Fungsi

- Fungsi yang sudah kita buat, dapat juga dipanggil di dalam fungsi lain.

```
<?php
// membuat fungsi
function hitungUmur($thn_Lahir, $thn_sekarang){
    $umur = $thn_sekarang - $thn_Lahir;
    return $umur;
}

function perkenalan($nama, $salam="Assalamualaikum"){
    echo $salam.", ";
    echo "Perkenalkan, nama saya ".$nama."<br/>";
    // memanggil fungsi lain
    echo "Saya berusia ". hitungUmur(1994, 2015) ." tahun<br/>";
    echo "Senang berkenalan dengan anda<br/>";
}

// memanggil fungsi perkenalan
perkenalan("Ardianta");

?>
```

Assalamualaikum,
Perkenalkan, nama saya Ardianta
Saya berusia 21 tahun
Senang berkenalan dengan anda

Fungsi rekursif

- Fungsi rekursif adalah fungsi yang memanggil dirinya sendiri. Fungsi ini biasanya digunakan untuk menyelesaikan masalah seperti faktorial, bilangan fibbonaci, pemrograman dinamis, dll.

```
<?php

function faktorial($angka) {
    if ($angka < 2) {
        return 1;
    } else {
        // memanggil dirinya sendiri
        return ($angka * faktorial($angka-1));
    }
}

// memanggil fungsi
echo "faktorial 5 adalah " . faktorial(5);

?>
```

faktorial 5 adalah 120

Melengkapi Setiap
Fungsi/Prosedur
Pada Program
Dengan Penulisan
Komentar

Komentar pada PHP

- Saat kita membuat baris-baris program didalam PHP kita dapat menggunakan fungsi komentar untuk menjelaskan maksud dari setiap baris atau function yang kita buat. Sehingga terdapat dokumentasi yang jelas dari setiap syntax PHP yang kita buat.
- Komentar di PHP diawali dengan tanda `//`. PHP tidak akan mengeksekusi setiap karakter yang terdapat dibelakang tanda `//`. Karena PHP akan membaca hal tersebut merupakan komentar. Contoh seperti dibawah ini:

Contoh

```
<?php  
echo "Selamat datang di Kelasprogrammer.com "; // Menampilkan output ke layar monitor  
?>
```



Pemberian Komentar pada fungsi

- Biasanya fungsi/prosedur/program dilengkapi dengan penulisan komentar di awal mengenai deskripsi fungsi/ prosedur/program tersebut; initial state dan final state; author (pembuat); versi dan/atau tanggal.

```
1  <?php
2  /*
3   * @desc class ini digunakan untuk menyimpan fungsi untuk interaksi user
4   * @author Jake Rocheleau (jakerocheleau@gmail.com)
5   * @required config.php
6   * @version 1.2
7   * @date : Maret 2019
8   */
9
10 class userInteraction{
11
12
13 /*
14  * @desc method digunakan untuk memproses login |
15  * @param $userID berisi id user dalam bentuk string
16  * @param $pass berisi pass dari user dalam bentuk string minimal 8 character
17  * @return $loginID yang berisi id login dalam bentuk string
18  */
19 public function login($userID, $pass){ ... }
20
21 }
22
23 ?>
```

Kesimpulan

Kesimpulan

- Fungsi dan Prosedure dalam PHP hampir sama yang membedakan hanyalah jika fungsi mengembalikan nilai, sebaliknya prosedur tidak mengembalikan nilai

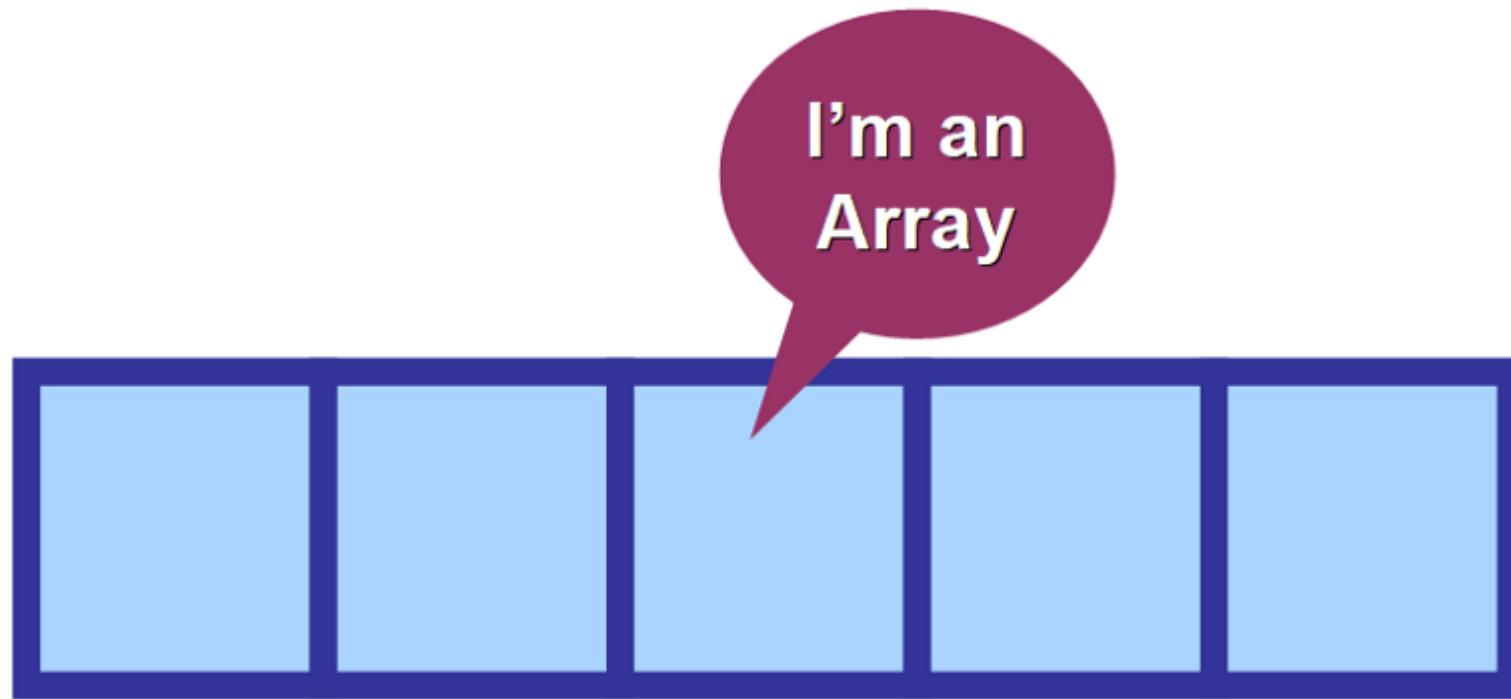
Referensi / Bacaan Lebih Lanjut

Referensi / Bacaan Lebih Lanjut

- <https://www.petanikode.com/php-prosedur-fungsi/>
- <https://www.w3schools.com/php/>

Membuat Program Menggunakan Array

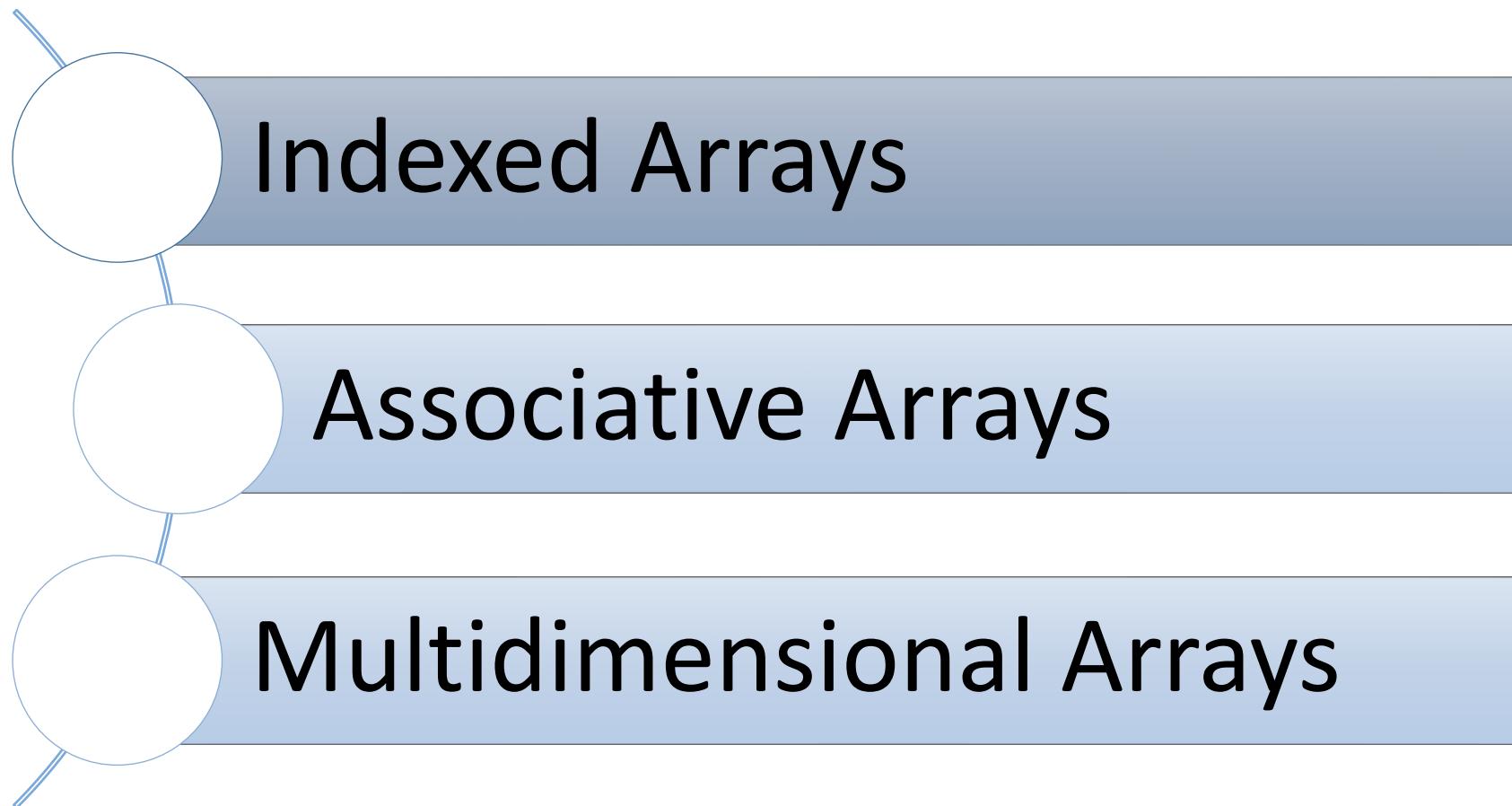
Aspersepsi



Ringkasan Mata Pelatihan

- Unit Kompetensi Acuan: Mengimplementasikan Pemrograman Terstruktur
- Kode Unit Kompetensi Acuan: J.620100.017.02
- Deskripsi singkat: Mata pelatihan ini menentukan kompetensi, pengetahuan dan sikap kerja yang diperlukan dalam membuat program terstruktur atau prosedural.
- Tujuan Pembelajaran: Peserta dapat membuat program terstruktur atau procedural sesuai dengan paradigma yang berlaku.

Agenda

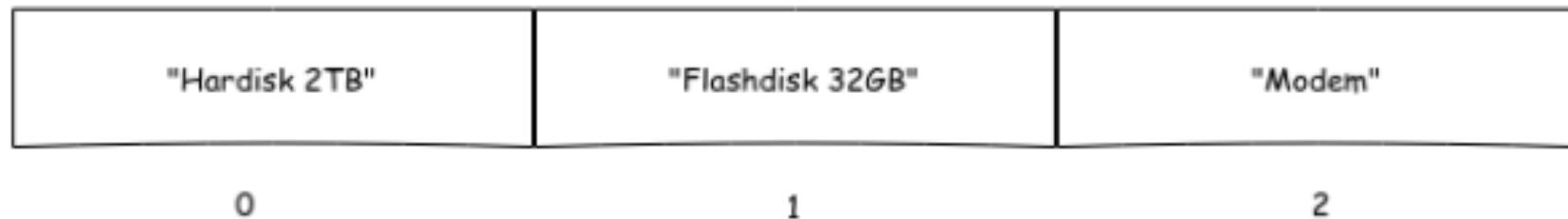


Indexed Array

Apa itu Array ?

Array adalah salah satu struktur data yang berisi sekumpulan data dan memiliki indeks. Indeks digunakan untuk mengakses nilai array.

Indeks array selalu dimulai dari nol (0).



Membuat Array di PHP

- Array di PHP dapat kita buat dengan fungsi array() dan tanda kurung kotak [].
- Array dapat kita isi dengan tipe data apa saja.

```
<?php

// membuat array kosong
$buah = array();
$hobi = [];

// membuat array sekaligus mengisinya
$minuman = array("Kopi", "Teh", "Jus Jeruk");
$makanan = ["Nasi Goreng", "Soto", "Bubur"];

// membuat array dengan mengisi indeks tertentu
$anggota[1] = "Dian";
$anggota[2] = "Muhar";
$anggota[0] = "Petani Kode";
```

```
<?php

$item = ["Bunga", 123, 39.12, true];
```

Menampilkan isi Array

- Untuk menampilkan isi array, kita bisa mengaksesnya melalui indeks.

```
<?php  
// membuat array  
$barang = ["Buku Tulis", "Penghapus", "Spidol"];  
  
// menampilkan isi array  
echo $barang[0]."<br>";  
echo $barang[1]."<br>";  
echo $barang[2]."<br>";
```

Buku Tulis
Penghapus
Spidol

- Bisa juga menggunakan perulangan

```
<?php  
// membuat array  
$barang = ["Buku Tulis", "Penghapus", "Spidol"];  
  
// menampilkan isi array dengan perulangan for  
for($i=0; $i < count($barang); $i++){  
    echo $barang[$i]."<br>";  
}
```

Menghapus isi Array

- Untuk menghapus isi array, kita bisa menggunakan fungsi `unset()`. Fungsi ini juga dapat digunakan untuk menghapus variable.

```
<?php

// membuat array
$hewan = [
    "Burung",
    "Kucing",
    "Ikan"
];

// menghapus kucing
unset($hewan[1]);

echo $hewan[0]."<br>";
echo $hewan[1]."<br>";
echo $hewan[2]."<br>";

echo "<hr>";

echo "<pre>";
print_r($hewan);
echo "</pre>";
```

Burung

Ikan

Array
(
 [0] => Burung
 [2] => Ikan
)

Menambahkan isi Array

Ada dua cara yang bisa dilakukan untuk menambah isi array:

1. Mengisi langsung ke nomer indeks yang ingin ditambahkan
2. Mengisi langsung ke indeks terakhir

```
<?php
// membuat array
$hobi = [
    "Membaca",
    "Menulis",
    "Ngeblog"
];

// menambahkan isi pada indeks ke-3
$hobi[3] = "Coding";

// menambahkan isi pada indeks terakhir
$hobi[] = "Olahraga";

// cetak array dengan perulangan
foreach($hobi as $hobiku){
    echo $hobiku."<br>";
}
?>
```

Membaca
Menulis
Ngeblog
Coding
Olahraga

Lanj

Apabila kita menambahkan pada indeks yang sudah memiliki isi, maka isinya akan ditindih dengan yang baru.

```
<?php
// membuat array
$user = [
    "dian",
    "muhar",
    "petanikode"
];

// mengisi array pada indek ke-1 ("muhar")
$user[1] = "ardianta";

// mencetak isi array
echo "<pre>";
print_r($user);
echo "</pre>";
?>
```

```
Array
(
    [0] => dian
    [1] => ardianta
    [2] => petanikode
)
```

Mengurutkan Array

```
$colors = array("red", "green", "blue", "yellow");

sort($colors);

foreach ($colors as $value) {
    echo "$value <br>";
}
```

Mengurutkan Array (terbalik)

```
$colors = array("red", "green", "blue", "yellow");

rsort($colors);

foreach ($colors as $value) {
    echo "$value <br>";
}
```

Associative Arrays

Apa Itu Array Asosiatif ?

- Array asosiatif adalah array yang indeksnya tidak menggunakan nomer atau angka. Indeks array asosiatif berbentuk kata kunci.

```
<?php
// membuat array asosiatif
$artikel = [
    "judul" => "Belajar Pemrograman PHP",
    "penulis" => "petanikode",
    "view" => 128
];

// mencetak isi array assosiatif
echo "<h2>".$artikel["judul"]."</h2>";
echo "<p>oleh: ".$artikel["penulis"]."</p>";
echo "<p>View: ".$artikel["view"]."</p>";
```

Belajar Pemrograman PHP

oleh: petanikode

View: 128

Lanj.

- Pada array asosiatif, kita menggunakan tanda => untuk mengasosiasikan sebuah kata kunci dengan isi array. Selain menggunakan tanda =>, kita juga bisa membuat array asosiatif seperti ini:

```
<?php
$email["subjek"] = "Apa Kabar?";
$email["pengirim"] = "dian@petanikode.com";
$email["isi"] = "Apa kabar? sudah lama tidak berjumpa";

echo "<pre>";
print_r($email);
echo "</pre>";
```

```
Array
(
    [subjek] => Apa Kabar?
    [pengirim] => dian@petanikode.com
    [isi] => Apa kabar? sudah lama tidak berjumpa
)
```

Multidimensional Arrays

Apa itu Array Multidimensi ?

- Array multidimensi adalah array yang memiliki dimensi lebih dari satu. Biasanya digunakan untuk membuat matrik, graph, dan stuktur data rumit lainnya.

```
<?php
// ini adalah array dua dimensi
$matrik = [
    [2,3,4],
    [7,5,0],
    [4,3,8],
];
// cara mengakses isinya
echo $matrik[1][0]; //-> output: 7
```

Array Multidimensi

Name	Stock	Sold
Volvo	22	18
BMW	15	13
Saab	5	2
Land Rover	17	15

Array Multidimensi

```
$cars = array
(
    array("Volvo",22,18),
    array("BMW",15,13),
    array("Saab",5,2),
    array("Land Rover",17,15)
);
```

Array Multidimensi

```
echo $cars[0][0].": In stock: ".$cars[0][1].", sold: ".$cars[0][2].".<br>";  
echo $cars[1][0].": In stock: ".$cars[1][1].", sold: ".$cars[1][2].".<br>";  
echo $cars[2][0].": In stock: ".$cars[2][1].", sold: ".$cars[2][2].".<br>";  
echo $cars[3][0].": In stock: ".$cars[3][1].", sold: ".$cars[3][2].".<br>";
```

Menampilkan Array Multidimensi

```
for ($row = 0; $row < 4; $row++) {  
    echo "<p><b>Row number $row</b></p>";  
    echo "<ul>";  
    for ($col = 0; $col < 3; $col++) {  
        echo "<li>".$cars[$row][$col]."</li>";  
    }  
    echo "</ul>";  
}
```

Kesimpulan

Kesimpulan

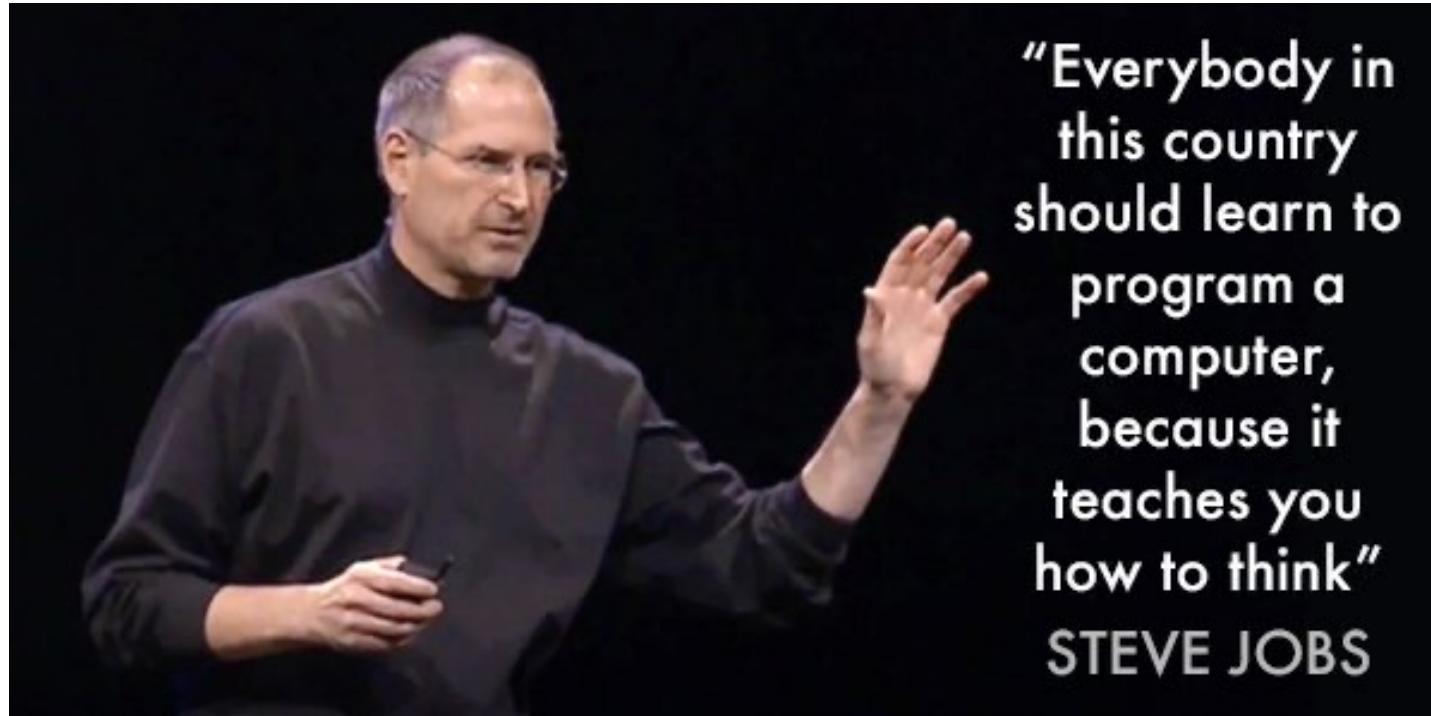
- Array dapat digunakan untuk menyimpan beberapa tipe data sehingga sangat fleksibel dan mudah di organisir.

Referensi / Bacaan Lebih Lanjut

Referensi / Bacaan Lebih Lanjut

- <https://www.petanikode.com/php-array/>
- <https://www.w3schools.com/php/>

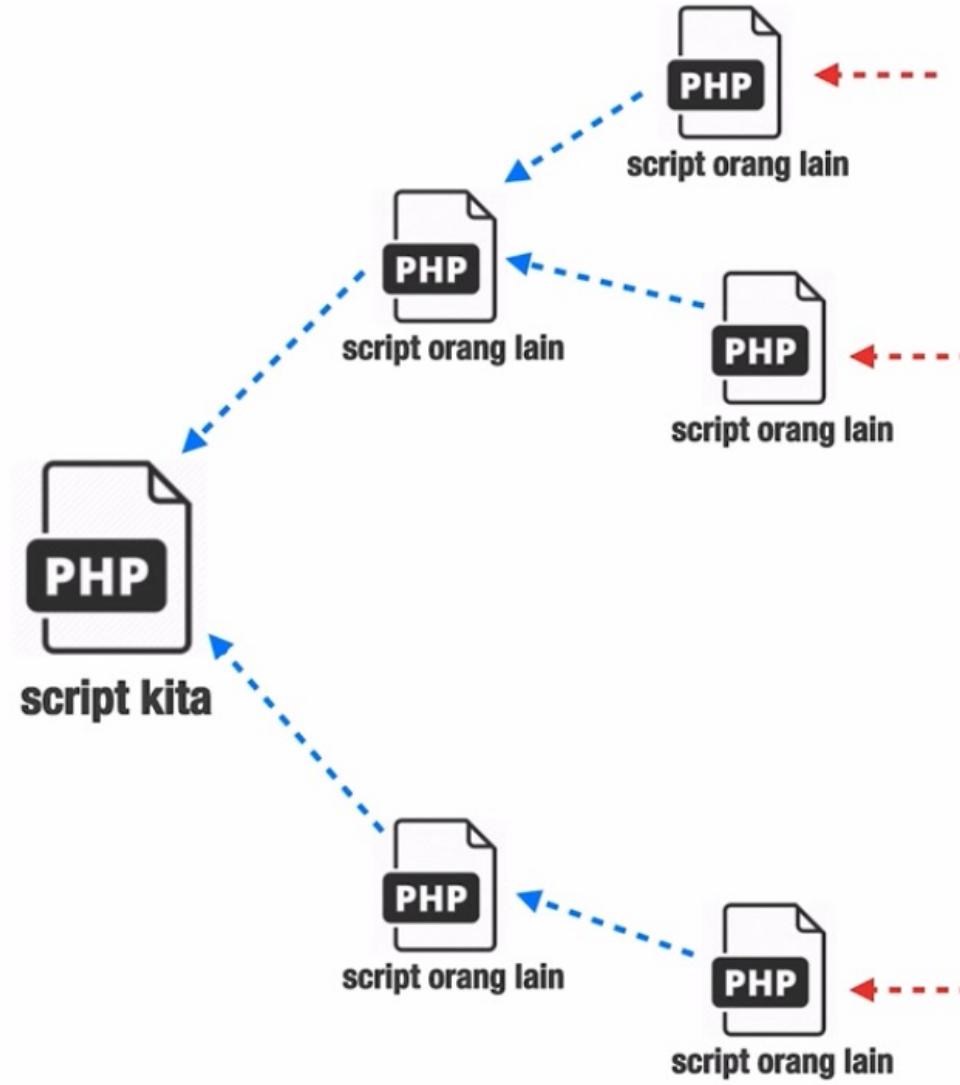
Motivasi



**"Everybody in
this country
should learn to
program a
computer,
because it
teaches you
how to think"**
STEVE JOBS

semua orang di negara ini
harus belajar memprogram
komputer, karena itu
mengajarkan Anda cara
berpikir

Apersepsi



Sumber: Web Programming Unpas

Penggunaan Library atau Komponen Pre- Existing

Ringkasan Mata Pelatihan

Unit Kompetensi Acuan	Menggunakan library atau komponen pre-existing
Kode Unit Kompetensi Acuan	J.620100.019.02
Deskripsi singkat	Mata Pelatihan ini memfasilitasi pembentukan kompetensi dalam menggunakan komponen-komponen reuse (yang dapat dipergunakan secara berulang untuk mendukung pengembangan aplikasi yang efisien)
Tujuan Pembelajaran	
Hasil Belajar	Setelah mengikuti seluruh rangkaian pembelajaran pada mata pelatihan Penggunaan Library atau Komponen Pre-Existing, peserta mampu menggunakan komponen-komponen reuse (yang dapat dipergunakan secara berulang untuk mendukung pengembangan aplikasi yang efisien).
Indikator Hasil Belajar	Setelah mengikuti pembelajaran ini, peserta dapat: <ol style="list-style-type: none">1. Melakukan pemilihan unit-unit reuse yang potensial2. Melakukan integrasi library atau komponen pre-existing dengan source code yang ada3. Melakukan pembaharuan library atau komponen library atau komponen pre-existing yang digunakan

Agenda

1. Memilih Unit-Unit Reuse yang Potensial
2. Mengintegrasikan Library atau komponen pre-existing dengan source code yang ada
3. Melakukan pembaharuan library atau komponen library atau komponen pre-existing yang digunakan

Memilih Unit-Unit Reuse yang Potensial

Setelah mengikuti pembelajaran ini, peserta dapat melakukan pemilihan unit-unit reuse yang potensial

Reuse, Library, dan Pre-Existing

- Reuse adalah penggunaan kembali suatu kesatuan kode tanpa menulis ulang atau mengubah kode tersebut.
- Library adalah pemaketan kode yang dapat digunakan untuk spesifikasi tertentu.
- Pre-Existing merupakan istilah untuk library atau komponen yang sudah ada dari sebelumnya.

Keuntungan Menerapkan ‘Reuse’

Mengurangi redundansi (pengulangan) penulisan kode

Menghemat waktu

Menghemat sumber daya

Pendekatan Reuse

- Pemilihan Desain dan Kode
- Software Libraries
- Software Frameworks
- Component-based Development
- Application Generators
- Design Patterns
- Product Lines

Hambatan dalam Menerapkan Reuse

1. Hambatan yang Disadari
 1. Persoalan Lisensi/Legal
 2. Tidak sesuai kebutuhan non-fungsional
 3. Masalah dalam mengadaptasi solusi
 4. Kehilangan kontrol
2. Hambatan yang Tidak Disadari
 1. Eksistensi solusi belum diantisipasi
 2. Masalah dalam mencari solusi

Jenis-jenis Library

- Pembuatan User interface (Bootstrap, DataTables, JQueryUI)
- Image editing (Imagine, PHP Graphics Works, Zebra Image)
- Impor/ekspor ke bentuk dokumen tertentu (TCPDF, PHPOffice)
- Algoritma kompleks (Artificial Neural Network Library, Artificial Neural Network Library)
- Dan lain-lain.

Lisensi Library

Hak yang diberikan	Public domain	Permissive FOSS license	Copyleft FOSS license	Freeware/ Shareware/ Freemium	Proprietary license	Rahasia dagang
Contoh lisensi		BSD License, MIT License	GPL, LGPL			
Hak Cipta Dilindungi	Tidak	Iya	Iya	Iya	Iya	Sangat Ketat
Hak untuk ditampilkan	Iya	Iya	Iya	Iya	Iya	Tidak
Hak untuk menyalin	Iya	Iya	Iya	Seringkali Iya	Tidak	Tidak
Hak untuk memodifikasi	Iya	Iya	Iya	Tidak	Tidak	Tidak
Hak untuk mendistribusikan	Iya	Ya, di bawah lisensi yang sama	Ya, di bawah lisensi yang sama	Seringkali Iya	Tidak	Tidak
hak untuk sublisensi	Iya	Iya	Tidak	Tidak	Tidak	Tidak
Contoh software	SQLite, ImageJ	Apache web server, ToyBox	Linux kernel, GIMP, OBS	Winamp, League of Legends	Windows, Spotify	Online Games PlayStation Network

Cara Mendapatkan Library

Mengunduh file langsung dari website penyedia library

1. Ketergantungan antar library atau unit dilakukan manual.
2. Memperbaharui library dilakukan dengan memeriksa di website/repository.
3. Tidak membutuhkan software tambahan.

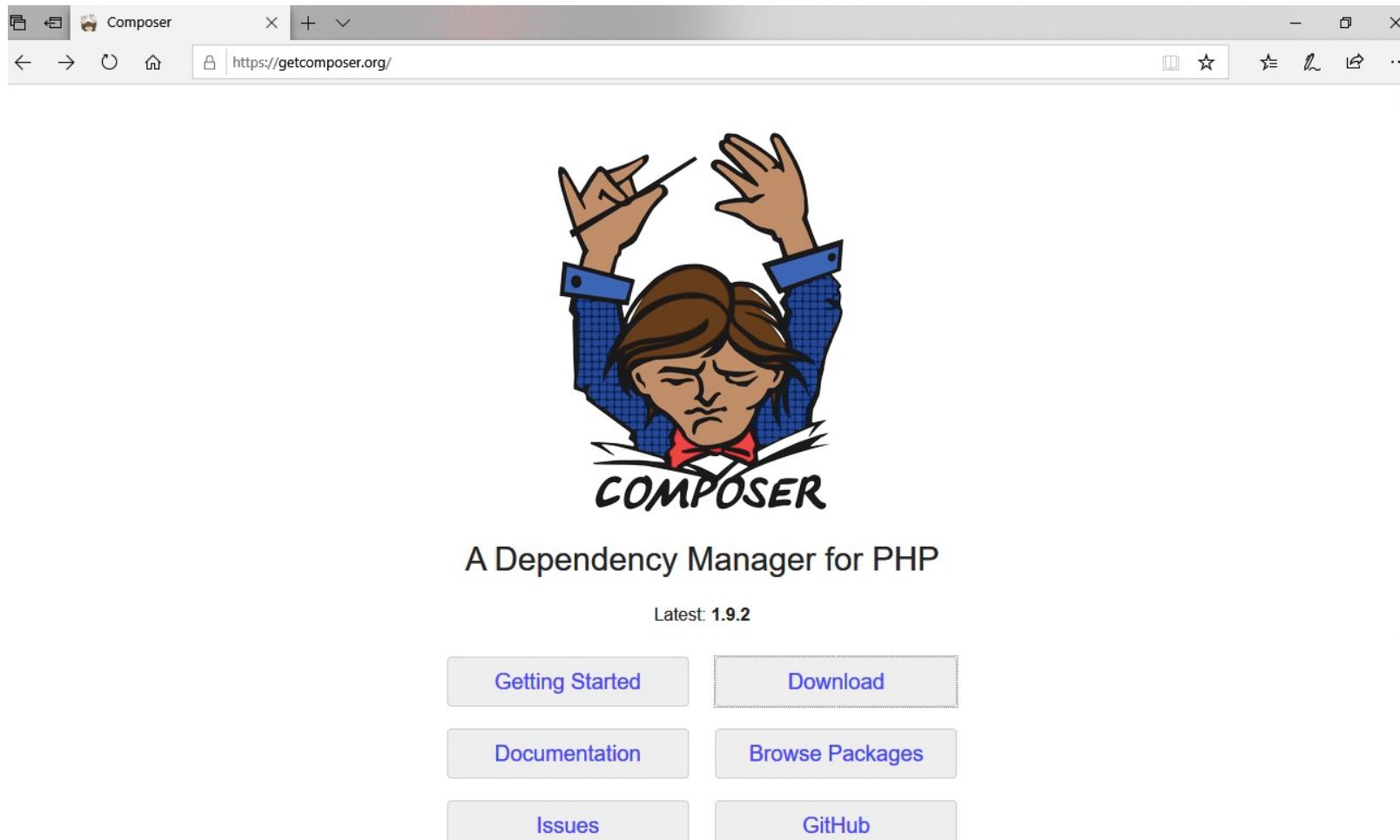
Menggunakan library manager (Composer)

1. Ketergantungan antar library atau unit dapat dilakukan secara otomatis
2. Memperbaharui library dapat dilakukan dengan mudah
3. Membutuhkan software tambahan

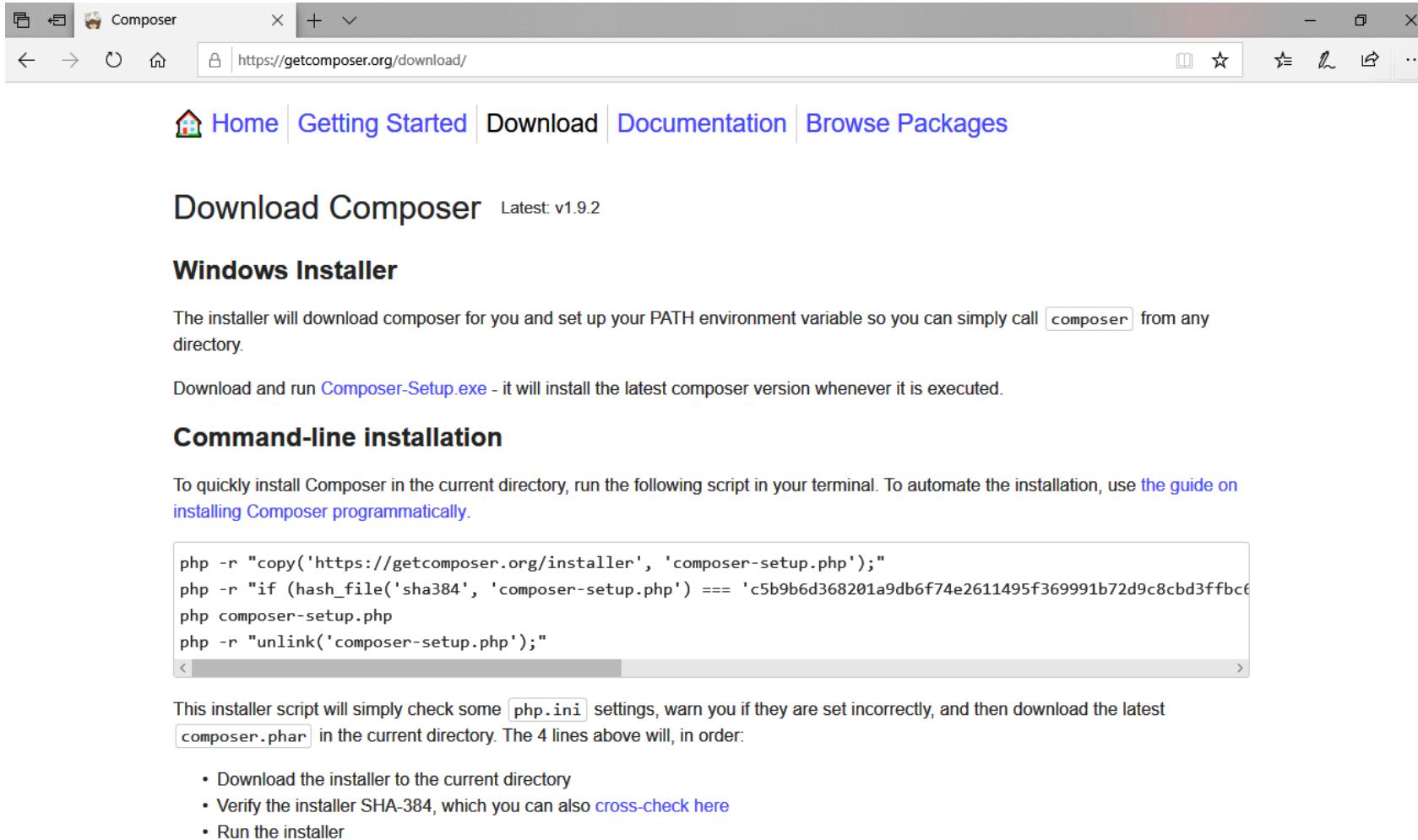
Mengintegrasikan Library atau komponen pre-existing dengan source code yang ada

Setelah mengikuti
pembelajaran ini, peserta
dapat melakukan integrasi
library atau komponen pre-
existing dengan source code
yang ada

Menginstall Composer (1)



Menginstall Composer (2)



The screenshot shows a Microsoft Edge browser window displaying the Composer download page at <https://getcomposer.org/download/>. The page features a navigation bar with links for Home, Getting Started, Download, Documentation, and Browse Packages. Below this, a section titled "Download Composer" indicates the latest version is v1.9.2. Two installation methods are shown: "Windows Installer" and "Command-line installation". The "Windows Installer" section describes how the installer will download Composer and set up the PATH environment variable. The "Command-line installation" section provides a shell script for quick installation in the current directory. A note below the script explains its purpose: it checks `php.ini` settings, warns about incorrect configurations, and then downloads the latest `composer.phar` file. A bulleted list at the bottom details the steps: download the installer, verify its SHA-384 hash, and run the installer.

Home | Getting Started | Download | Documentation | Browse Packages

Download Composer Latest: v1.9.2

Windows Installer

The installer will download composer for you and set up your PATH environment variable so you can simply call `composer` from any directory.

Download and run [Composer-Setup.exe](#) - it will install the latest composer version whenever it is executed.

Command-line installation

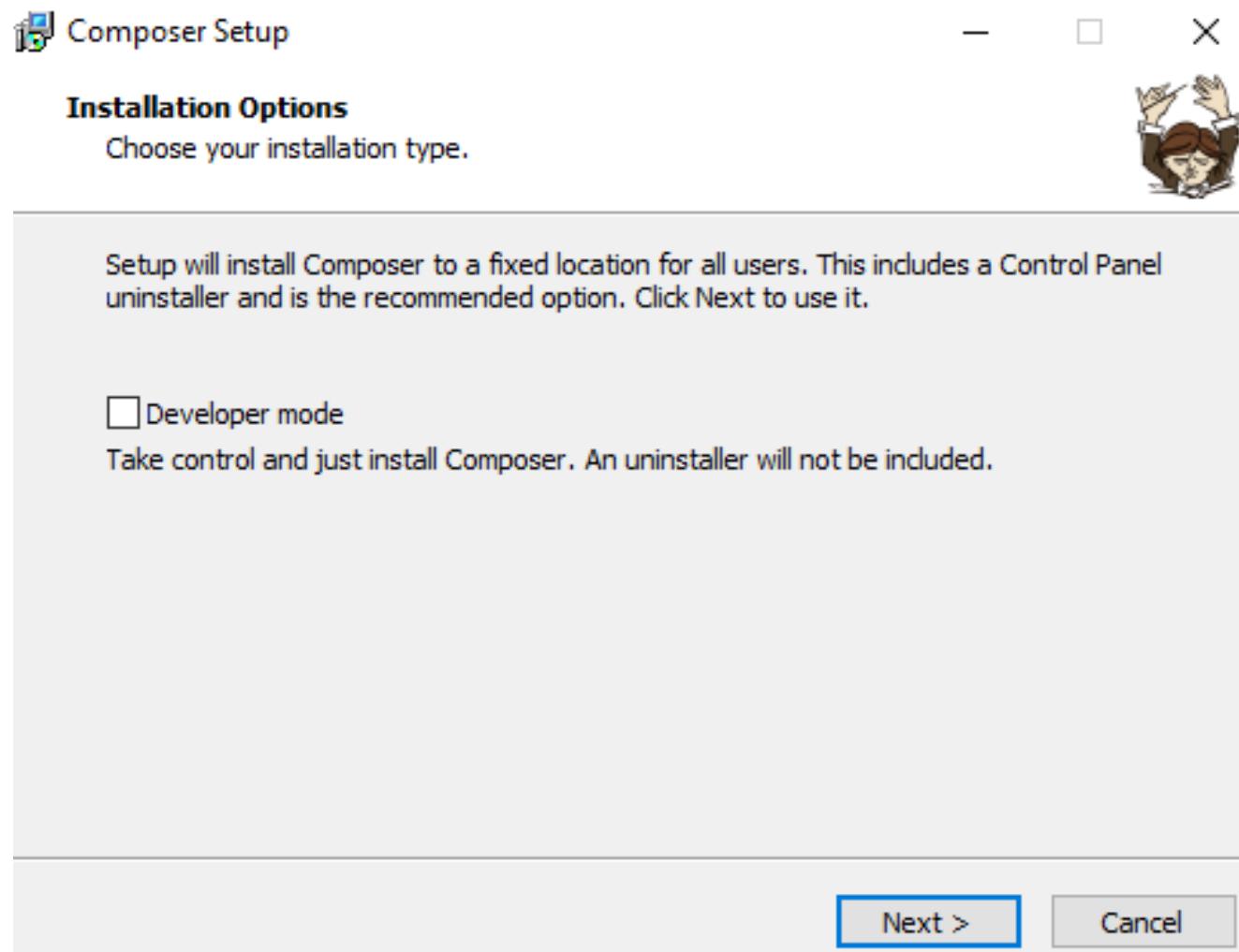
To quickly install Composer in the current directory, run the following script in your terminal. To automate the installation, use [the guide on installing Composer programmatically](#).

```
php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"
php -r "if (hash_file('sha384', 'composer-setup.php') === 'c5b9b6d368201a9db6f74e2611495f369991b72d9c8cbd3ffbc6
php composer-setup.php
php -r "unlink('composer-setup.php');"
```

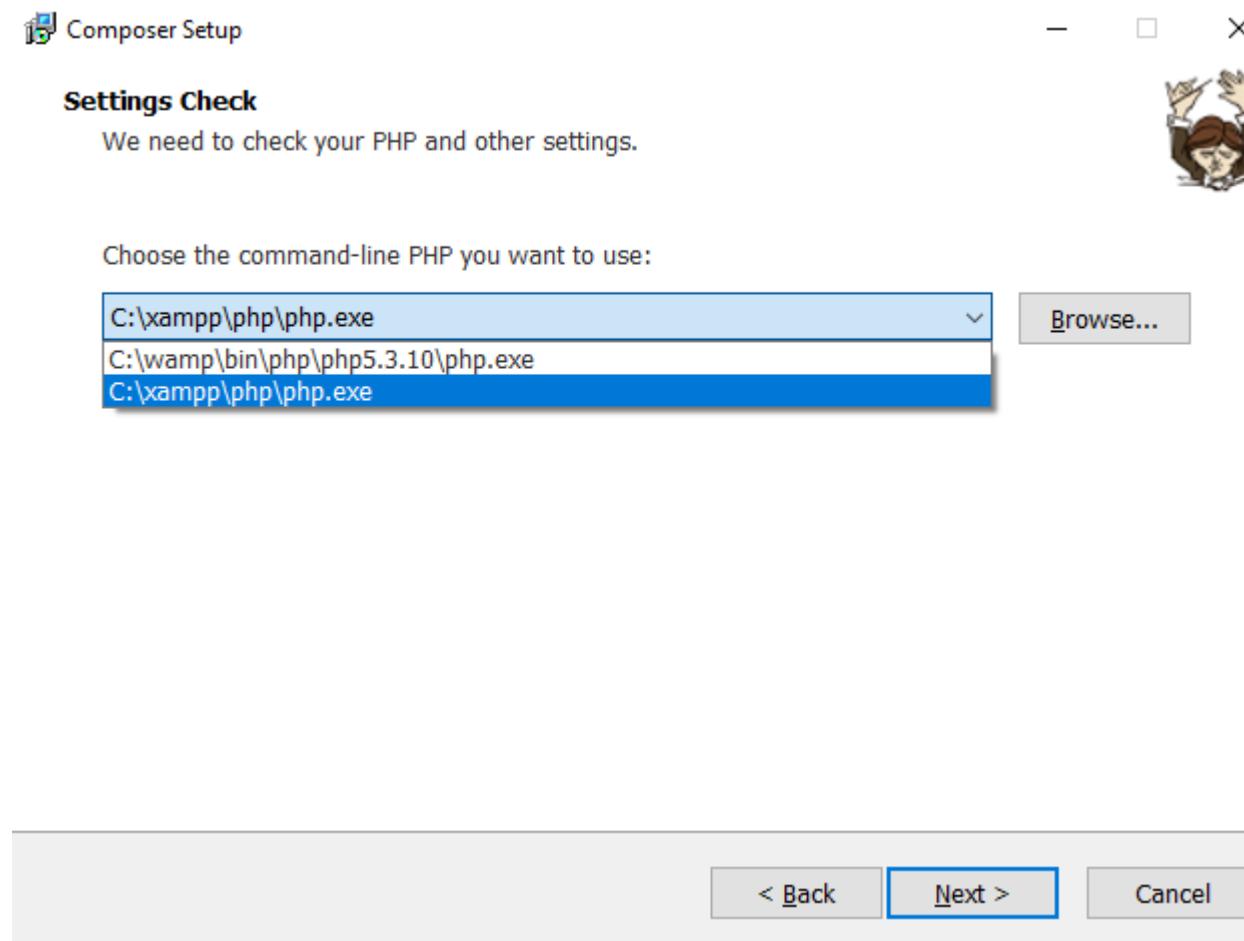
This installer script will simply check some `php.ini` settings, warn you if they are set incorrectly, and then download the latest `composer.phar` in the current directory. The 4 lines above will, in order:

- Download the installer to the current directory
- Verify the installer SHA-384, which you can also [cross-check here](#)
- Run the installer

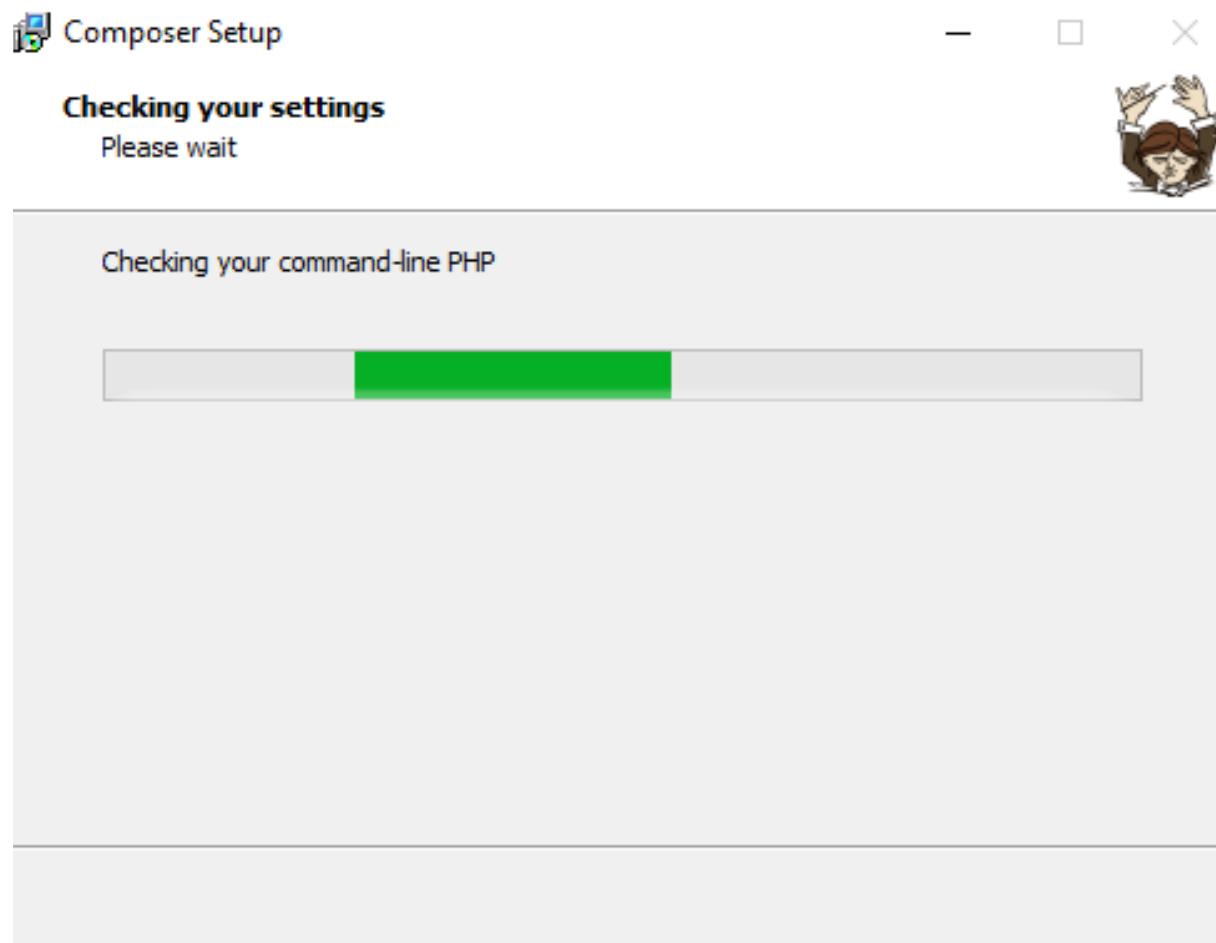
Menginstall Composer (3)



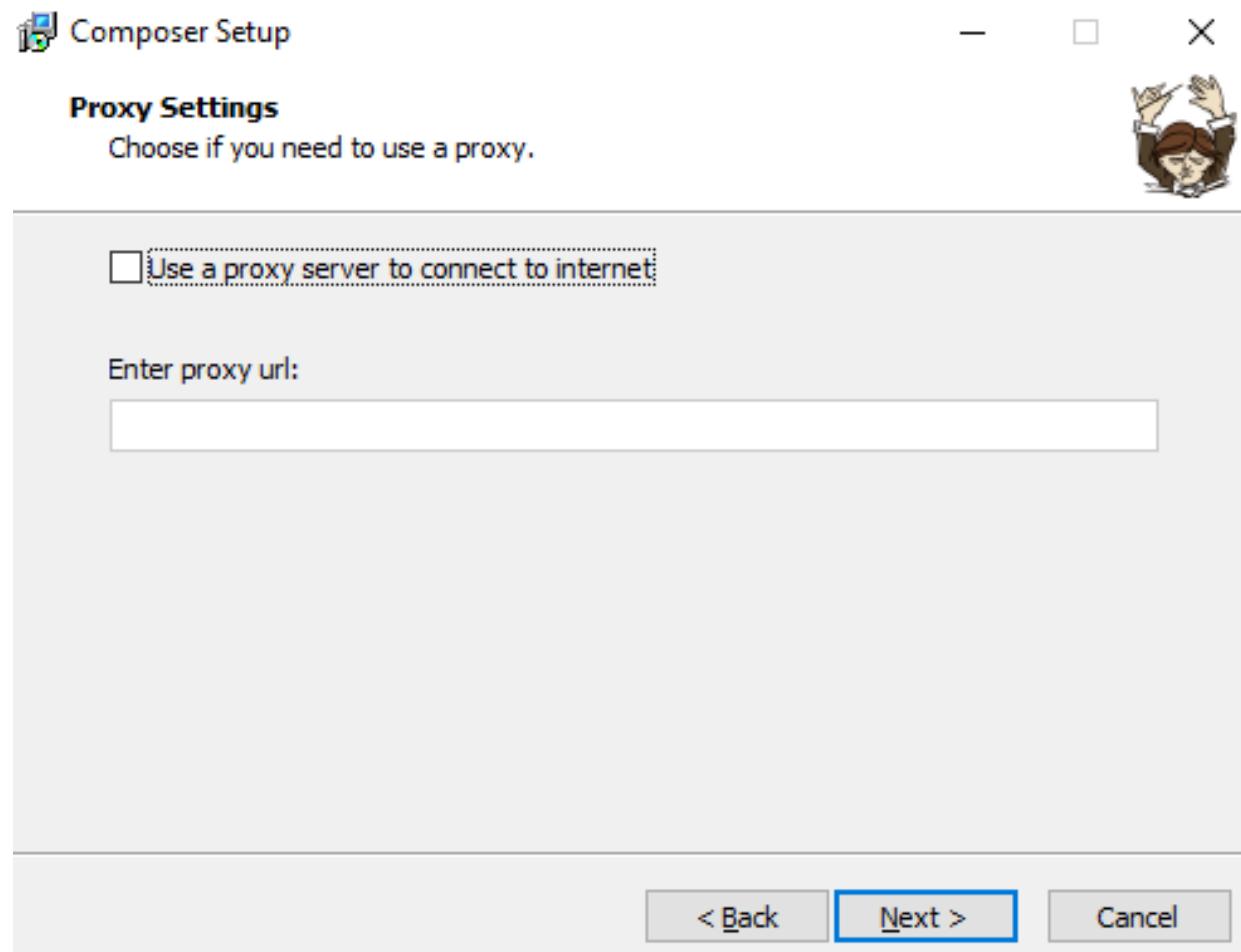
Menginstall Composer (4)



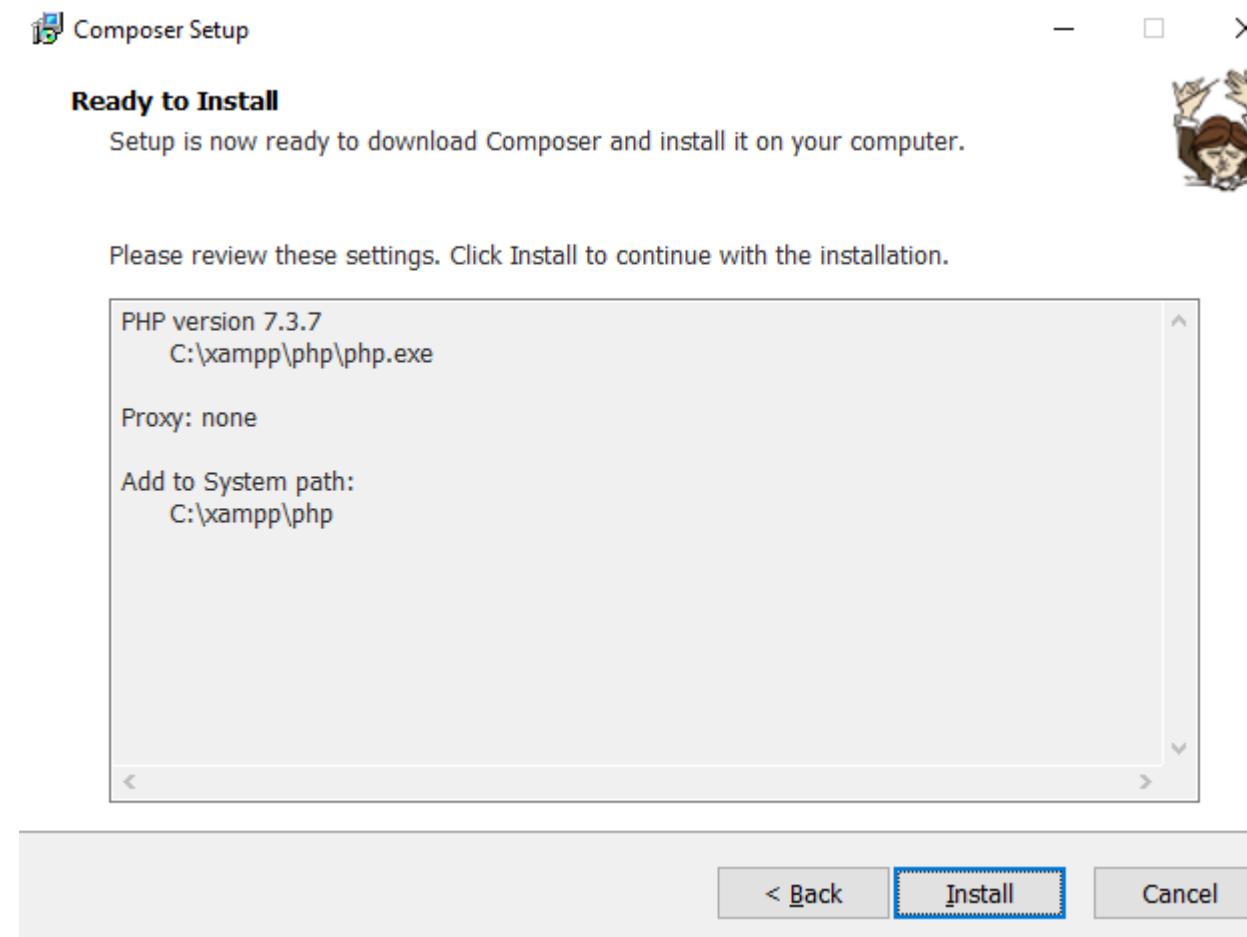
Menginstall Composer (5)



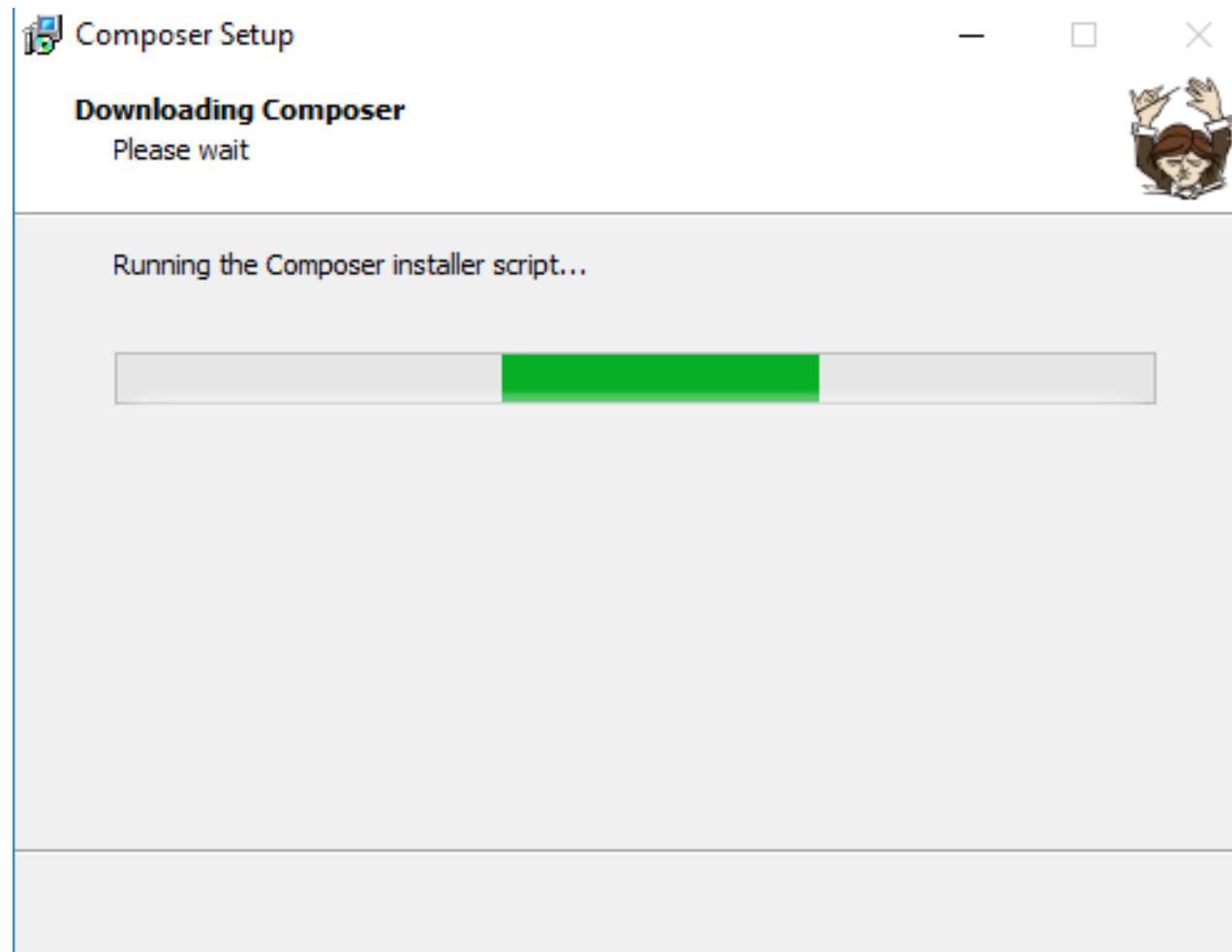
Menginstall Composer (6)



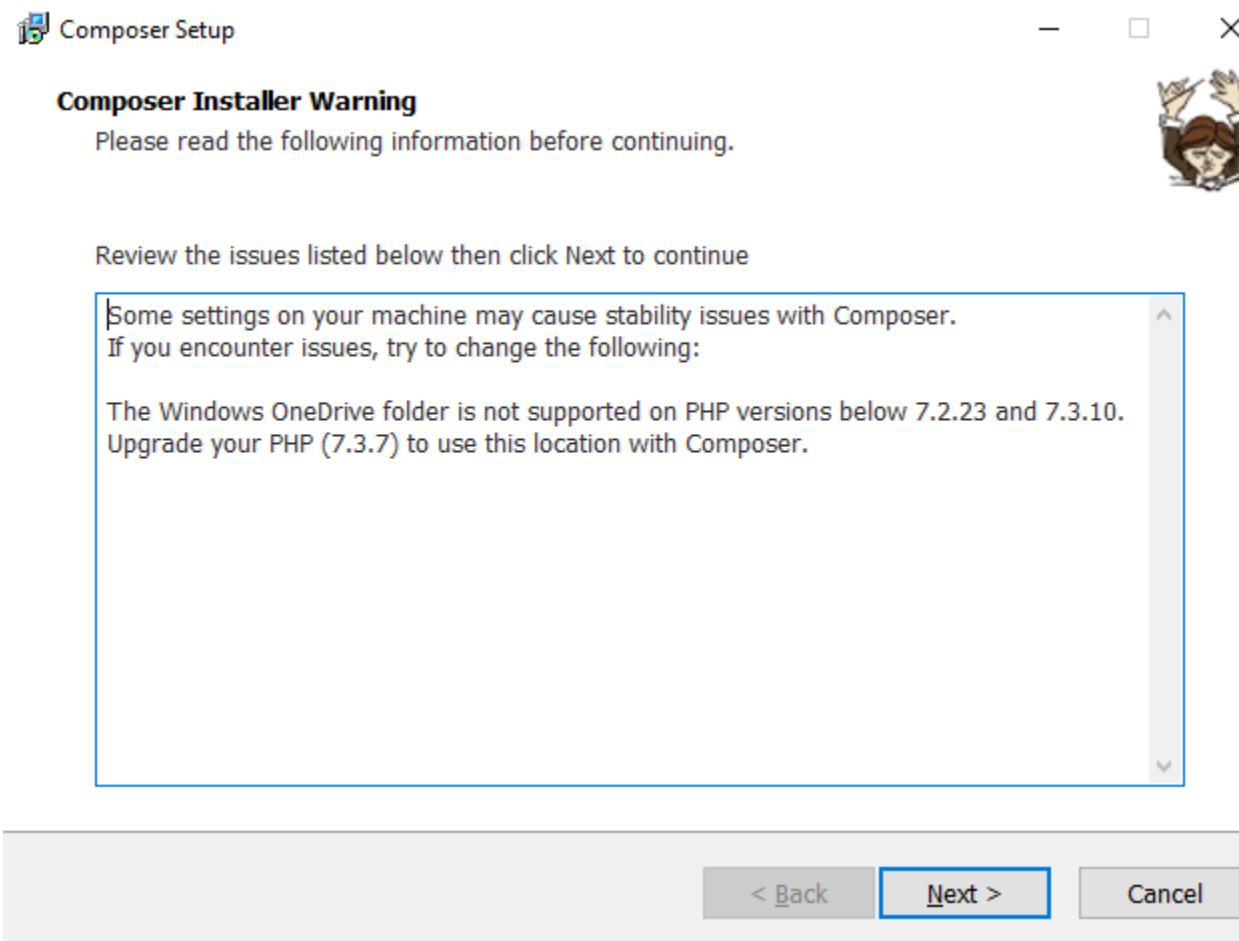
Menginstall Composer (7)



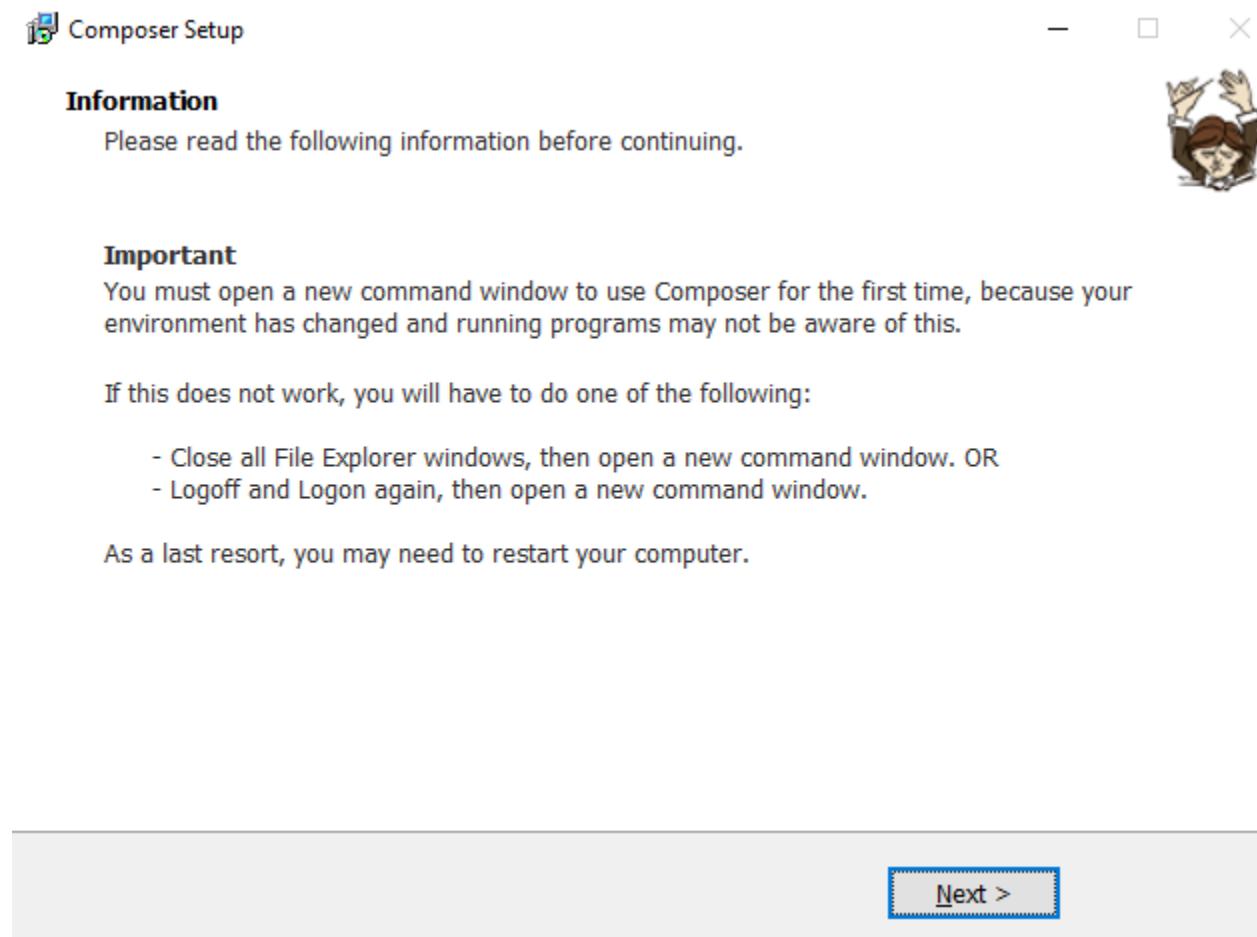
Menginstall Composer (8)



Menginstall Composer (9)



Menginstall Composer (10)



Menginstall Composer (11)



Menginstall Composer (12)

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.18362.778]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Roni>composer
The logo consists of a grid of 16 squares arranged in a 4x4 pattern. The squares are filled with various symbols: some have diagonal lines, some have vertical lines, and some have horizontal lines, creating a stylized representation of a person or a logo.

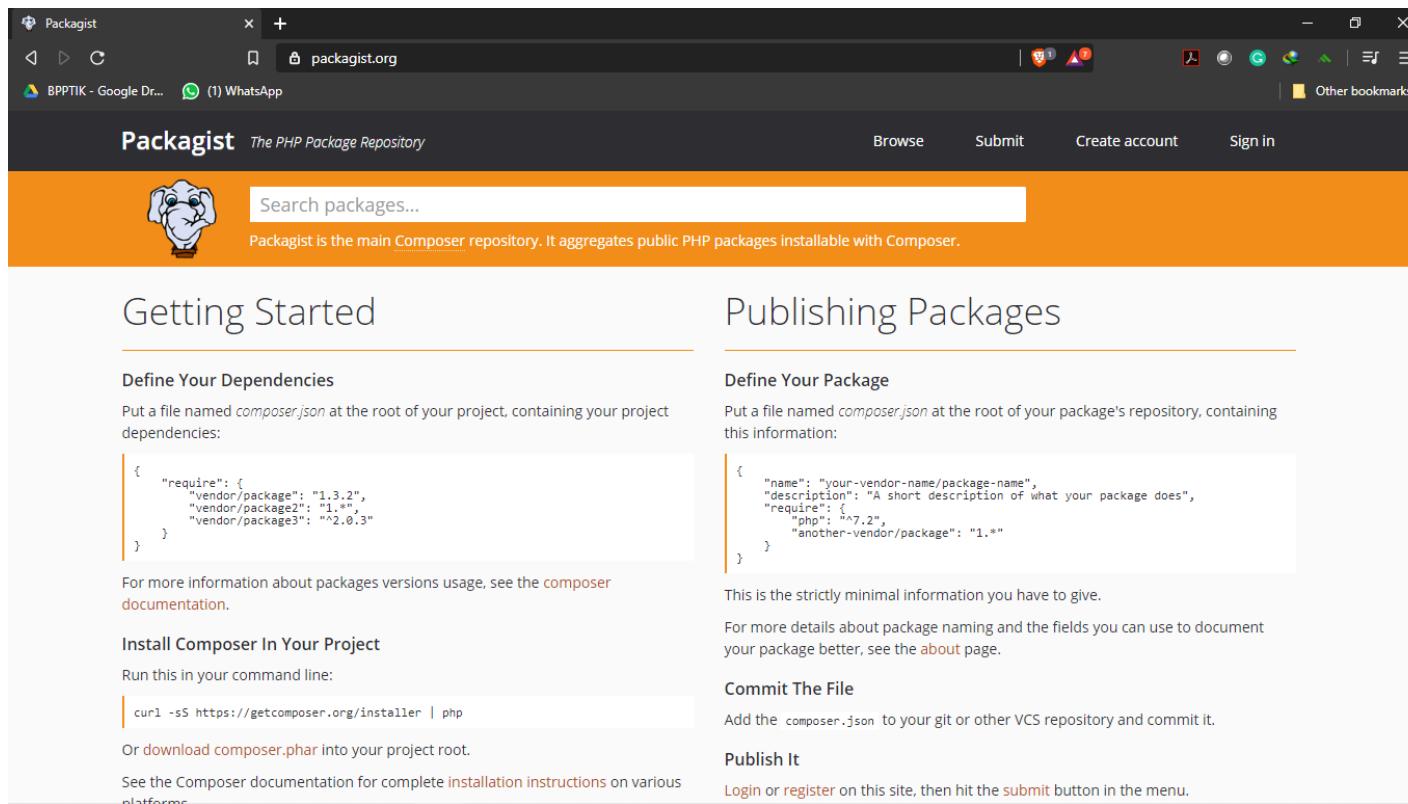
Composer version 1.10.5 2020-04-10 11:44:22

Usage:
  command [options] [arguments]

Options:
  -h, --help          Display this help message
  -q, --quiet         Do not output any message
  -V, --version       Display this application version
  --ansi             Force ANSI output
  --no-ansi          Disable ANSI output
  -n, --no-interaction Do not ask any interactive question
  --profile          Display timing and memory usage information
  --no-plugins        Whether to disable plugins.
  -d, --working-dir=WORKING-DIR If specified, use the given directory as working directory.
  --no-cache          Prevent use of the cache
  -v|vv|vvv, --verbose Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and
  3 for debug

Available commands:
```

Menggunakan Composer



The screenshot shows the homepage of Packagist.org. At the top, there's a navigation bar with links for 'Browse', 'Submit', 'Create account', and 'Sign in'. Below the navigation is a search bar with the placeholder 'Search packages...'. To the left of the search bar is a logo of a white elephant. A banner below the search bar states: 'Packagist is the main Composer repository. It aggregates public PHP packages installable with Composer.' The page is divided into two main sections: 'Getting Started' on the left and 'Publishing Packages' on the right.

Getting Started

Define Your Dependencies

Put a file named `composer.json` at the root of your project, containing your project dependencies:

```
{ "require": { "vendor/package": "1.3.2", "vendor/package2": "1.*", "vendor/package3": "2.0.3" } }
```

For more information about packages versions usage, see the [composer documentation](#).

Install Composer In Your Project

Run this in your command line:

```
curl -sS https://getcomposer.org/installer | php
```

Or download `composer.phar` into your project root.

See the Composer documentation for complete [installation instructions](#) on various platforms.

Publishing Packages

Define Your Package

Put a file named `composer.json` at the root of your package's repository, containing this information:

```
{ "name": "your-vendor-name/package-name", "description": "A short description of what your package does", "require": { "php": ">=7.2", "another-vendor/package": "1.*" } }
```

This is the strictly minimal information you have to give.

For more details about package naming and the fields you can use to document your package better, see the [about](#) page.

Commit The File

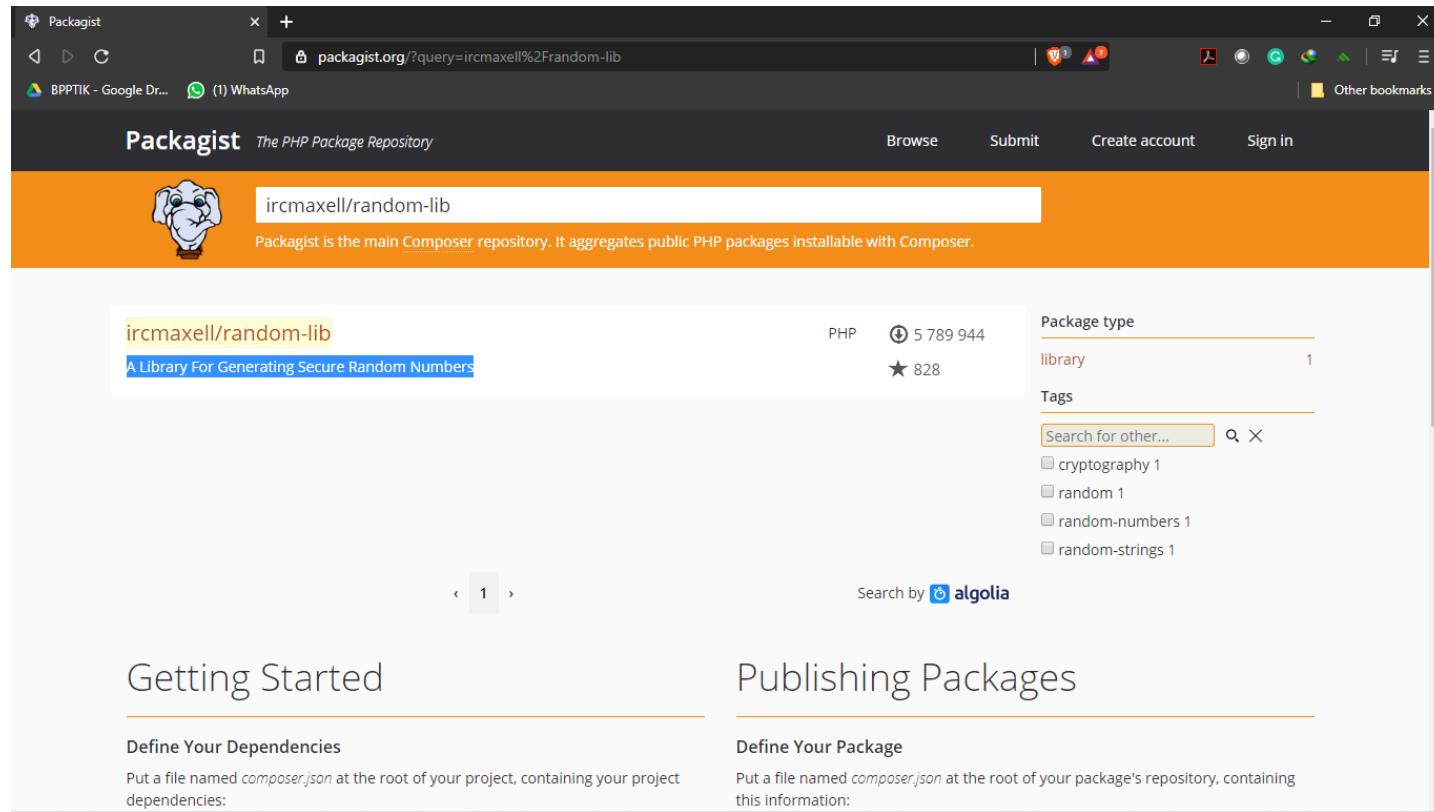
Add the `composer.json` to your git or other VCS repository and commit it.

Publish It

Login or register on this site, then hit the submit button in the menu.

- Buka packagist.org
- Cari package yang akan digunakan

Menggunakan Composer



- Sebagai contoh kita akan menggunakan packagist ircmaxell/random-lib
- Dimana packagist ini digunakan untuk men-generate secure random numbers

Menggunakan Composer

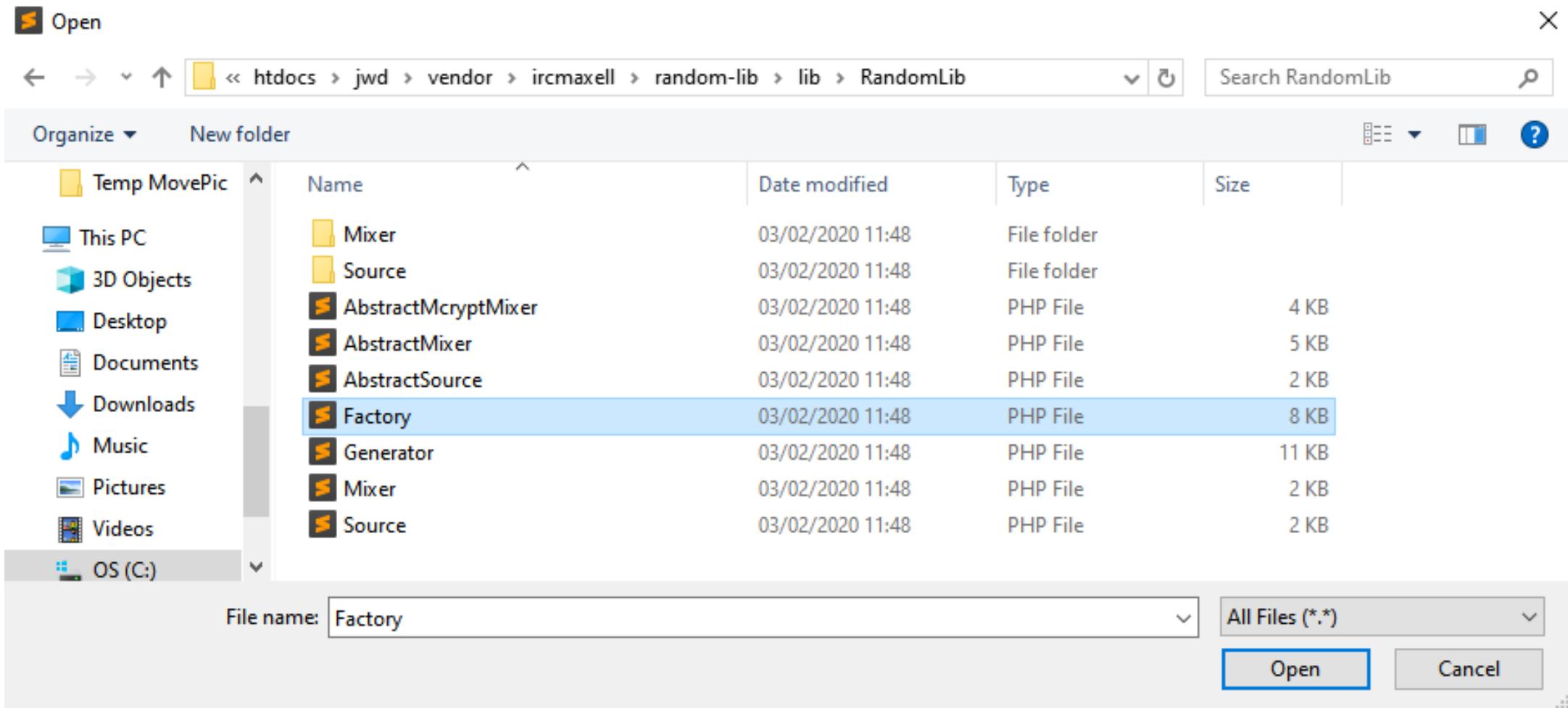
```
C:\WINDOWS\system32\cmd.exe

C:\xampp\htdocs\jwd>composer require ircmaxell/random-lib
Using version ^1.2 for ircmaxell/random-lib
./composer.json has been created
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 2 installs, 0 updates, 0 removals
- Installing ircmaxell/security-lib (v1.1.0): Downloading (100%)
- Installing ircmaxell/random-lib (v1.2.0): Downloading (100%)
Writing lock file
Generating autoload files

C:\xampp\htdocs\jwd>
```

- Tentukan lokasi packagist akan di install, contoh :
C:\xampp\htdocs\jwd
- Gunakan perintah untuk mendownload packagist
> composer require ircmaxell/random-lib
- Tunggu hingga proses instalasi selesai

Contoh Class dalam Sebuah Library



C:\xampp\htdocs\jwd\vendor\ircmaxell\random-lib\lib\RandomLib\Factory.php - Sublime Text (UNREGISTERED)

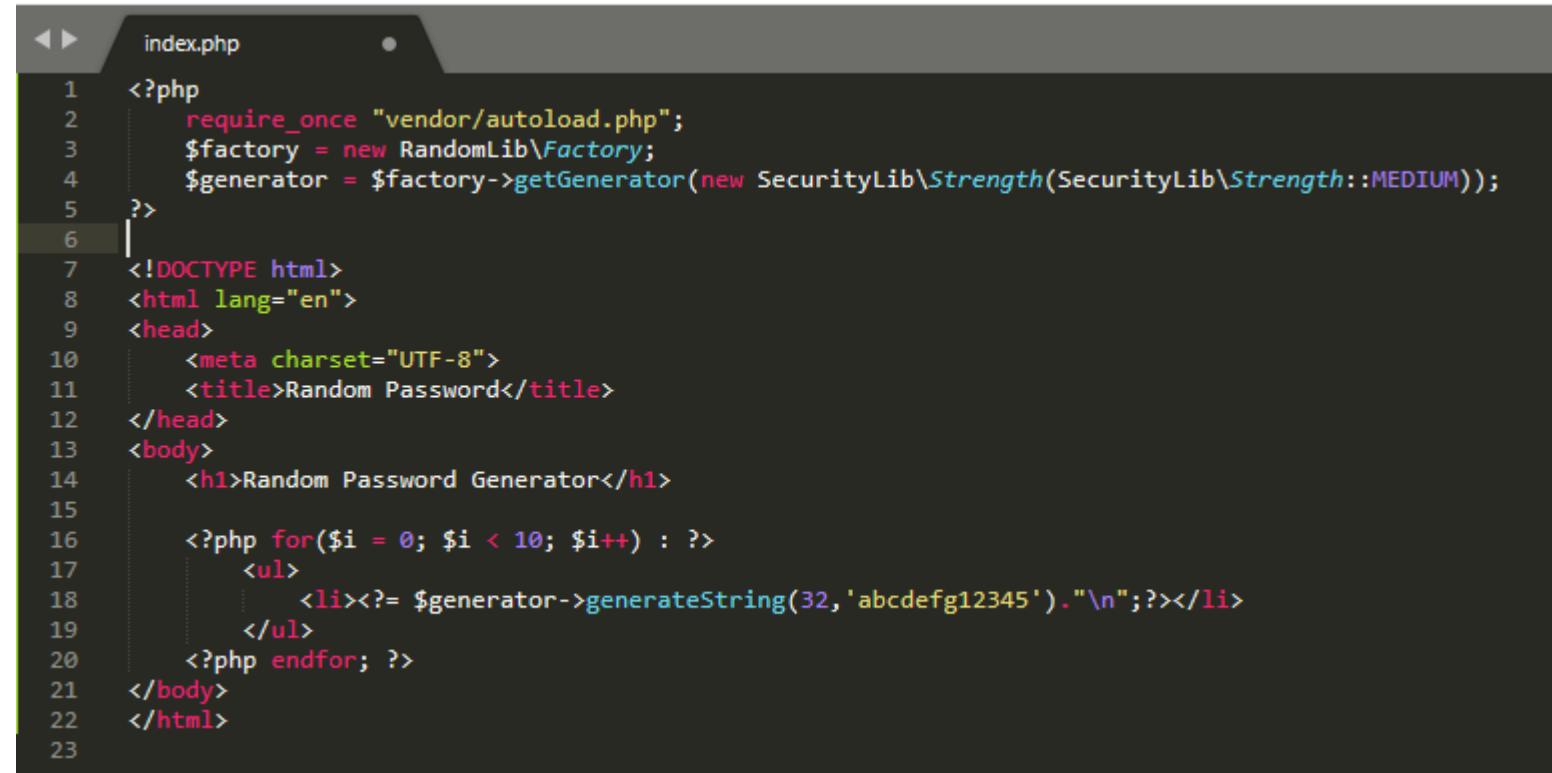
File Edit Selection Find View Goto Tools Project Preferences Help

Factory.php

```
10     */
11
12 /**
13 * The Random Factory
14 *
15 * Use this factory to instantiate random number generators, sources and mixers.
16 *
17 * PHP version 5.3
18 *
19 * @category PHPPasswordLib
20 * @package Random
21 *
22 * @author Anthony Ferrara <ircmaxell@ircmaxell.com>
23 * @copyright 2011 The Authors
24 * @license http://www.opensource.org/licenses/mit-license.html MIT License
25 *
26 * @version Build @version@*
27 */
28 namespace RandomLib;
29
30 use SecurityLib\Strength;
31
32 /**
33 * The Random Factory
34 *
35 * Use this factory to instantiate random number generators, sources and mixers.
36 *
37 * @category PHPPasswordLib
38 * @package Random
39 *
40 * @author Anthony Ferrara <ircmaxell@ircmaxell.com>
41 */
42 class Factory extends \SecurityLib\AbstractFactory
43 {
44
45 /**
46 * @var array A list of available random number mixing strategies
47 */
48 protected $mixers = array();
49
50 /**
51 * @var array A list of available random number sources
52 */
53 protected $sources = array();
54
55 /**
56 * Build a new instance of the factory, loading core mixers and sources
57 *
58 * @return void
59 */
60 public function __construct()
61 {
```

- Perhatikan namespace, use, dan nama class
- Perhatikan ketergantungan antar ‘unit’

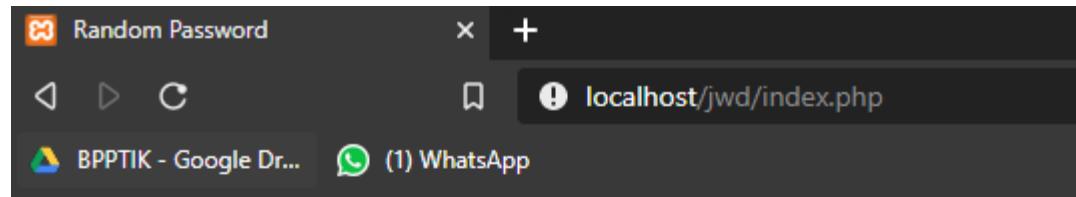
Menjalankan fungsi pada library composer



```
index.php
1 <?php
2     require_once "vendor/autoload.php";
3     $factory = new RandomLib\Factory;
4     $generator = $factory->getGenerator(new SecurityLib\Strength(SecurityLib\Strength::MEDIUM));
5 ?>
6
7 <!DOCTYPE html>
8 <html lang="en">
9 <head>
10    <meta charset="UTF-8">
11    <title>Random Password</title>
12 </head>
13 <body>
14    <h1>Random Password Generator</h1>
15
16    <?php for($i = 0; $i < 10; $i++) : ?>
17        <ul>
18            <li><?= $generator->generateString(32, 'abcdefg12345')."\n";?></li>
19        </ul>
20    <?php endfor; ?>
21 </body>
22 </html>
23
```

- Buat file baru pada folder jwd
- Masukan kode berikut

Menjalankan fungsi pada library composer



Random Password Generator

- f24fe515341d41acfdfgb14g3becd14
- bbgcdaffd4bb1afa1db2b51f3f5c4g5a
- ab341ca5b4c21a3eg41144dg22ecee1
- cdcf5fga42425e425afa2afc3fd52cf3
- 23ebf43fafc2452gf53g2a1e2ga4dd2b
- 1b42542e43g4bf42gad4c1253c4f2c3c
- a45112dbe2f22c23cdefefff2e3c2e1a
- 5542a5fb1ec11gb5fgedbbd3f4423443
- 2b4g44gg344ba3dcb2aa45geg4gb4c14
- 33b5cc3gdcccebfceced1f1f1efaeag2d1

- Program akan men-Generate 10 Password yang berbeda

Melakukan Pembaharuan Library Atau Komponen Library Atau Komponen Pre- existing Yang Digunakan

Setelah mengikuti pembelajaran ini, peserta dapat melakukan pembaharuan library atau komponen library atau komponen pre-existing yang digunakan

Contoh Cara Memperbaharui Sebuah Library

Memperbaharui library dengan composer, dapat dilakukan dengan perintah yang dijalankan dalam direktori project:

```
# composer update
```

```
C:\xampp\htdocs\jwd>composer update ircmaxell/random-lib
Loading composer repositories with package information
Updating dependencies (including require-dev)
Nothing to install or update
Generating autoload files

C:\xampp\htdocs\jwd>_
```

Memeriksa Library yang Sudah ‘Obsolete’

- Hindari penggunaan library yang sudah *obsolete* atau usang.
- Gunakan command: **composer outdated** untuk memeriksa ketergantungan library yang sudah out of date dan dapat diupdate
- Pembaharuan library yang diunduh secara langsung, proses pembaharuan dilakukan dengan memeriksa langsung ke website/repository library terkait.

Kesimpulan

Kesimpulan

1. Memilih artefak (unit-unit) dalam penerapan reuse, harus disesuaikan dengan kebutuhan.
2. Integrasi library atau komponen pre-existing perlu memperhatikan ketergantungan antar unit dan komponen yang sudah obsolete.
3. Pembaharuan library atau komponen library atau komponen pre-existing dapat secara mudah dilakukan dengan memanfaatkan Composer

Output form html:

Liv

Form Pendaftaran

localhost:1235/01-memb...

Nama

Email

Usia

Tanggal Lahir
 mm/dd/yyyy

Alamat

Jenis Kelamin

Laki-Laki

Perempuan

Status

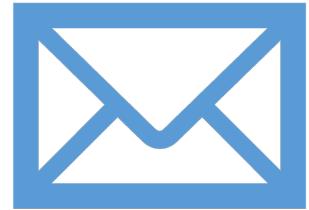
Hobi

Berenang

Sepak Bola

Bulu Tangkis

Ngoding



Kantor:

Balai Pelatihan dan Pengembangan
Teknologi Informasi dan Komunikasi
Kementerian Kominfo

Website: <https://bpptik.keminfo.go.id>

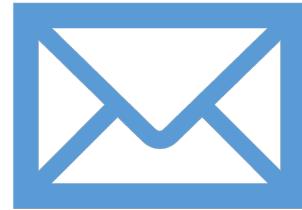
Email: bpptik@kominfo.go.id

Twitter: @bpptik

Facebook: @bpptik

Instagram: @bpptik

Google Plus: +bpptikkemkominfo



Kantor:

Balai Pelatihan dan Pengembangan
Teknologi Informasi dan Komunikasi

Kementerian Kominfo

Website: <https://bpptik.keminfo.go.id>

Email: bpptik@keminfo.go.id

Twitter: @bpptik

Facebook: @bpptik

Instagram: @bpptik

Google Plus: +bpptikkemkominfo

Terima Kasih

BPPTIK