

# Live Session 3

## Menyusun Fungsi, File atau Sumber Daya Pemrograman yang lain dalam Organisasi yang Rapi

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"> <li>1. Mengimplementasikan user interface</li> <li>2. Menerapkan perintah eksekusi bahasa pemrograman berbasis teks, grafik, dan multimedia</li> <li>3. Menyusun fungsi, file atau sumber daya pemrograman yang lain dalam organisasi yang rapi</li> <li>4. Menulis kode dengan prinsip sesuai guidelines dan best practices</li> <li>5. Mengimplementasikan pemrograman terstruktur</li> <li>6. Menggunakan library atau komponen pre-existing</li> </ol>
Jam Pelatihan	67 JP / 9 hari
Jenis Pelatihan	<b>Daring (<i>Online</i>)</b>

# 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**



# Organisasi Sumber Daya Pemrograman Sesuai Konteks

# Ringkasan Mata Pelatihan

- Unit Kompetensi Acuan: Menyusun fungsi, file atau sumber daya pemrograman yang lain dalam organisasi yang rapi
- Kompetensi Acuan: J. 620100.015.01
- Deskripsi singkat: mengelola dan mengorganisasikan sumber daya pemrograman.
- Tujuan Pembelajaran:
  - Hasil Belajar: bisa mengelola dan mengorganisasikan sumber daya pemrograman (*Maintainable*)

# Agenda

1. Menyusun folder dan sub-sub folder sesuai konteks dan isinya.
2. Membuat file “readme”, mengandung penjelasan mengenai struktur/hirarki folder serta penjelasan mengenai sumber daya pemrograman.

(materi pelatihan yang akan disampaikan mengacu pada dokumen Program Pelatihan)

# Fungsi & Prosedur

# Fungsi pada PHP

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

Fungsi pada PHP dapat dibuat dengan kata kunci function, lalu diikuti dengan nama fungsinya.

```
function namaFungsi() {  
    //...  
}
```

# Fungsi pada PHP

Kode intruksi dapat di tulis di dalam kurung kurawal ({}).

Contoh:



```
fungsi_1.php
1 <?php
2 function perkenalan(){
3     echo "Selamat Datang, ";
4     echo "Pada Acara VSGA Junior Web Developer BPPTIK Kominfo<br/>";
5     echo "2022<br/>";
6 }
7 ?>
8
```

# Fungsi pada PHP

Fungsi sebelumnya tidak akan menghasilkan output, untuk menampilkan outputnya harus memanggil nama fungsinya terlebih dahulu.

Seperti pada program dibawah ini:



```
<?php
//membuat fungsi
function perkenalan(){
    echo "Selamat Datang, ";
    echo "Pada Acara VSGA Junior Web Developer BPPTIK Kominfo<br/>";
    echo "2022<br/>";
}
//memanggil fungsi yang sudah dibuat
perkenalan();
?>
```

# Fungsi pada PHP

Hasil dari fungsi sebelumnya adalah sebagai berikut:



Selamat Datang, Pada Acara VSGA Junior Web Developer BPPTIK Kominfo  
2022

# Fungsi dengan Parameter

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

```
1 <?php
2 // membuat fungsi
3 ▶ function perkenalan($nama, $salam){
4     echo $salam.", ";
5     echo "Perkenalkan, nama saya ".$nama."<br/>";
6     echo "Senang berkenalan dengan anda<br/>";
7 }
8
9 // memanggil fungsi yang sudah dibuat
10 ▶ perkenalan("Komang", "Hi");
11
12 echo "<hr>";
13
14 $saya = "Medi";
15 $ucapanSalam = "Selamat pagi";
16 // memanggilnya lagi
17 perkenalan($saya, $ucapanSalam);
18 ?>
```

# Fungsi dengan Parameter

Hasil dari program diatas adalah:



Hi, Perkenalkan, nama saya Komang  
Senang berkenalan dengan anda

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

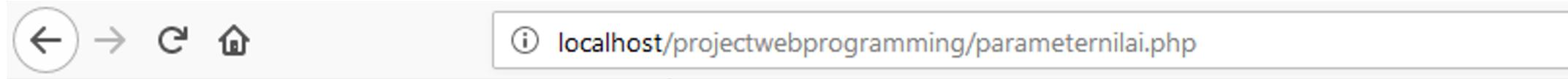
# Parameter dengan Nilai Default

Nilai default dapat kita berikan di parameter. Nilai default berfungsi untuk mengisi nilai sebuah parameter, kalau parameter tersebut tidak diisi nilainya. Misalnya: saya lupa mengisi parameter salam, maka program akan error. Oleh karena itu, kita perlu memberikan nilai default supaya tidak error.

```
1 <?php
2 // mmbuat fungsi
3 function perkenalan($nama, $salam="Selamat Datang"){
4     echo $salam.", ";
5     echo "Perkenalkan, nama saya ".$nama."<br/>";
6     echo "Senang berkenalan dengan anda<br/>";
7 }
8
9 // memanggil fungsi yang sudah dibuat
10 perkenalan("Komang", "Hi");
11
12 echo "<hr>";
13
14 $saya = "Medi";
15 $ucapanSalam = "Selamat pagi";
16 // memanggilnya lagi tanpa mengisi parameter salam
17 perkenalan($saya);
18 ?>
```

# Parameter dengan Nilai Default

Hasil dari program diatas adalah:



Hi, Perkenalkan, nama saya Komang  
Senang berkenalan dengan anda

---

Selamat Datang, Perkenalkan, nama saya Medi  
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.

Pengembalian nilai dalam fungsi dapat menggunakan kata kunci return.

Contoh:

```
1 <?php
2 // membuat fungsi
3 function hitungUmur($thn_lahir, $thn_sekarang){
4     $umur = $thn_sekarang - $thn_lahir;
5     return $umur;
6 }
7
8 echo "Umur saya adalah ". hitungUmur(1992, 2019) ." tahun";
9 ?>
```

# Fungsi yang Mengembalikan Nilai

Hasil dari program diatas adalah:



Umur saya adalah 27 tahun

# Memanggil Fungsi di dalam Fungsi

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

Contoh:

```
1 <?php
2 // membuat fungsi
3 function hitungUmur($thn_lahir, $thn_sekarang){
4     $umur = $thn_sekarang - $thn_lahir;
5     return $umur;
6 }
7
8 function perkenalan($nama, $salam="Selamat Datang"){
9     echo $salam.", ";
10    echo "Perkenalkan, nama saya ".$nama."<br/>";
11    // memanggil fungsi lain
12    echo "Saya berusia ". hitungUmur(1992, 2019) ." tahun<br/>";
13    echo "Senang berkenalan dengan anda<br/>";
14 }
15
16 // memanggil fungsi perkenalan
17 perkenalan("Komang");
18 ?>
```

# Memanggil Fungsi di dalam Fungsi

Hasil dari program diatas adalah:



Selamat Datang, Perkenalkan, nama saya Komang  
Saya berusia 27 tahun  
Senang berkenalan dengan anda

# Fungsi Rekursif

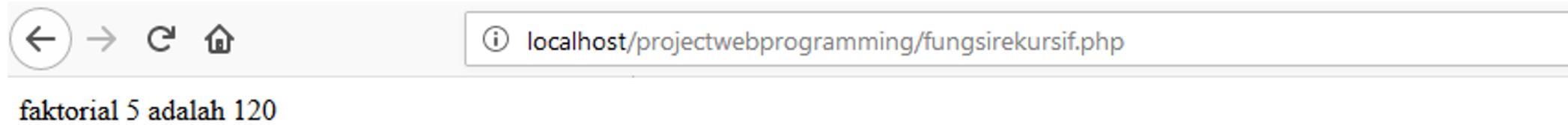
Fungsi rekursif adalah fungsi yang memanggil dirinya sendiri. Fungsi ini biasanya digunakan untuk menyelesaikan masalah seperti faktorial, bilangan fibbonaci, pemrograman dinamis, dan lain-lain.

Contoh:

```
1 <?php
2
3 function faktorial($angka) {
4     if ($angka < 2) {
5         return 1;
6     } else {
7         // memanggil dirinya sendiri
8         return ($angka * faktorial($angka-1));
9     }
10 }
11
12 // memanggil fungsi
13 echo "faktorial 5 adalah " . faktorial(5);
14
15 ?>
```

# Fungsi Rekursif

Hasil dari program diatas adalah:



# Prosedur dalam PHP

Sebuah perintah yang dapat digunakan untuk membagi beberapa kejadian dalam suatu kumpulan perintah yang lebih kecil dengan berbagai kelengkapan di dalamnya baik itu pengecekan kondisi, fungsi matematika maupun fungsi string.

Prosedur tidak dapat mengembalikan nilai.

Dengan menggunakan prosedur atau fungsi dapat menghemat banyak ruang atau ukuran program dan menghindari pengetikan kode yang berulang-ulang.

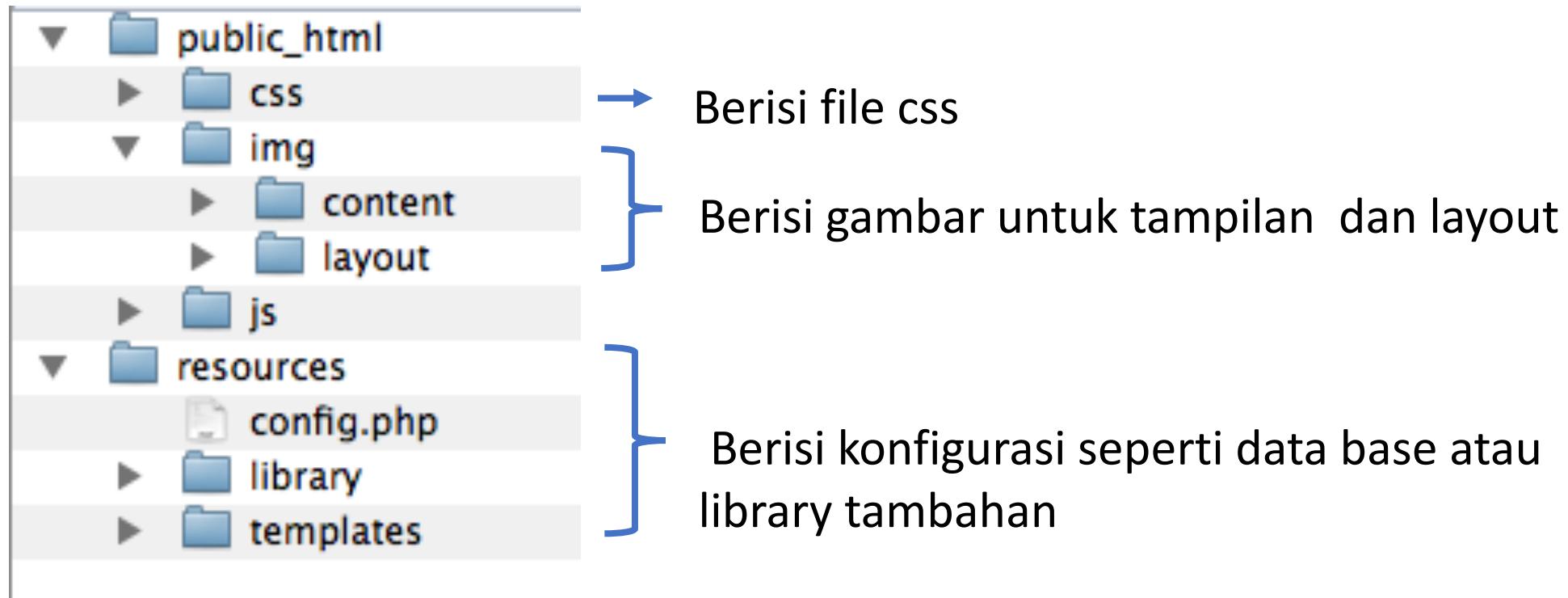
# Manfaat Fungsi dan Procedure

- ❖ Dapat menghemat banyak ruang atau ukuran program
- ❖ Menghindari pengetikan kode yang berulang-ulang.
- ❖ Pencarian kesalahan lebih mudah karena kesalahan dapat dilokalisasi dalam suatu sub routine tertentu saja.
- ❖ Jika ada aktifitas memodifikasi program, programmer fokus pada memodifikasi satu fungsi atau procedure saja tanpa khawatir mengganggu fungsi atau procedure yang lain
- ❖ Reusability, fungsi yang sudah dibuat dapat digunakan kembali

# Penyusunan Folder dan Sub-sub Folder

# Penyusunan Folder dan Sub-sub Folder

- Folder dan sub-sub folder disusun sesuai konteks dan isinya.



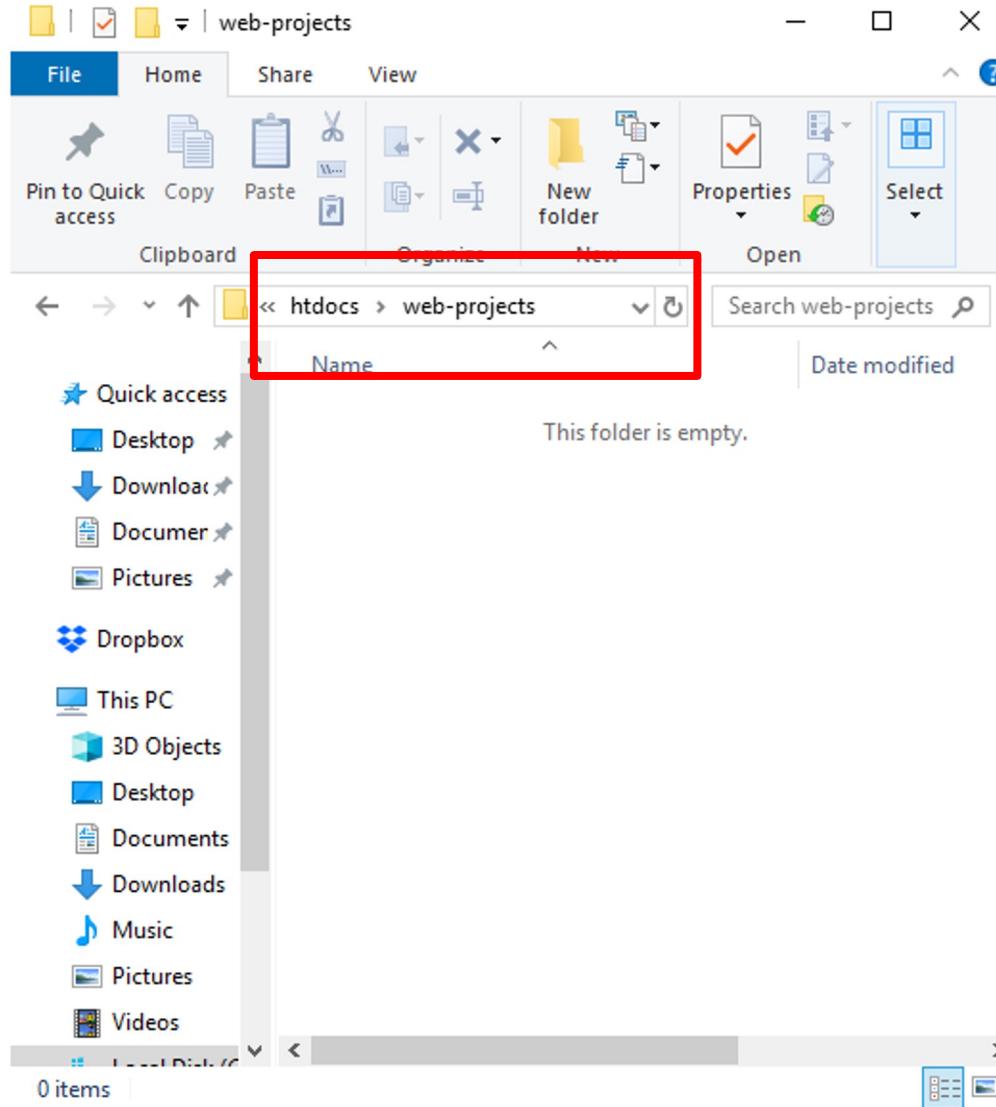
# Menyusun Struktur Folder dan File PHP

- ❖ Suatu website terdiri atas banyak file, seperti konten teks, kode, stylesheet, konten media, dan lain-lain.
- ❖ Ketika kamu sedang membuat sebuah website, kamu perlu merangkai file-file ini menjadi sebuah struktur yang sesuai pada komputermu.
- ❖ Memastikan file-file tersebut saling terhubung antara satu dengan lainnya, dan memastikan semua konten sudah benar sebelum akhirnya kamu unggah mereka ke suatu server.

## Menyusun Struktur Folder dan File PHP

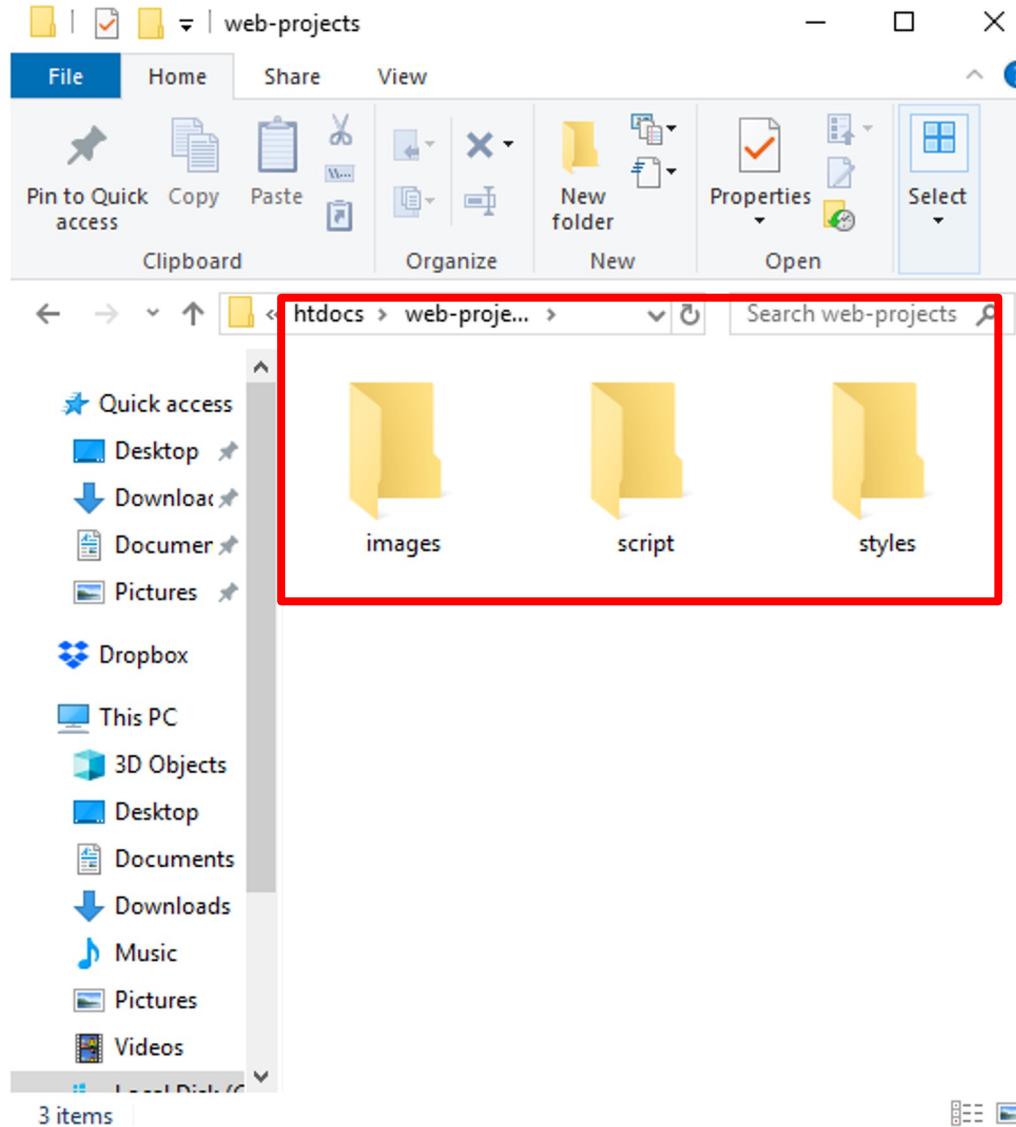
- ❖ Ketika kamu sedang mengerjakan satu website lokal pada komputer pribadimu, kamu harus menyimpan semua file terkait pada satu folder sesuai dengan struktur website yang diunggah ke server.
- ❖ Folder ini harus disimpan di dalam folder htdocs agar pada saat server local diaktifkan, kita dapat melihat hasil website kita sebelum di upload ke server.

# Menyusun Struktur Folder dan File PHP



- ❖ Buatlah sebuah folder dengan nama web-projects (atau nama lain yang serupa) didalam folder htdocs.
- ❖ Di sinilah proyek-proyek websitemu akan disimpan.

# Menyusun Struktur Folder dan File PHP



- ❖ Hal yang paling sering kamu temukan pada proyek website apapun yang kita buat adalah sebuah file indeks HTML/PHP dan folder untuk menyimpan gambar, file CSS, dan file script.
- ❖ Maka buat 3 folder didalam web-projects dengan nama sebagai berikut.

# Menyusun Struktur Folder dan File PHP

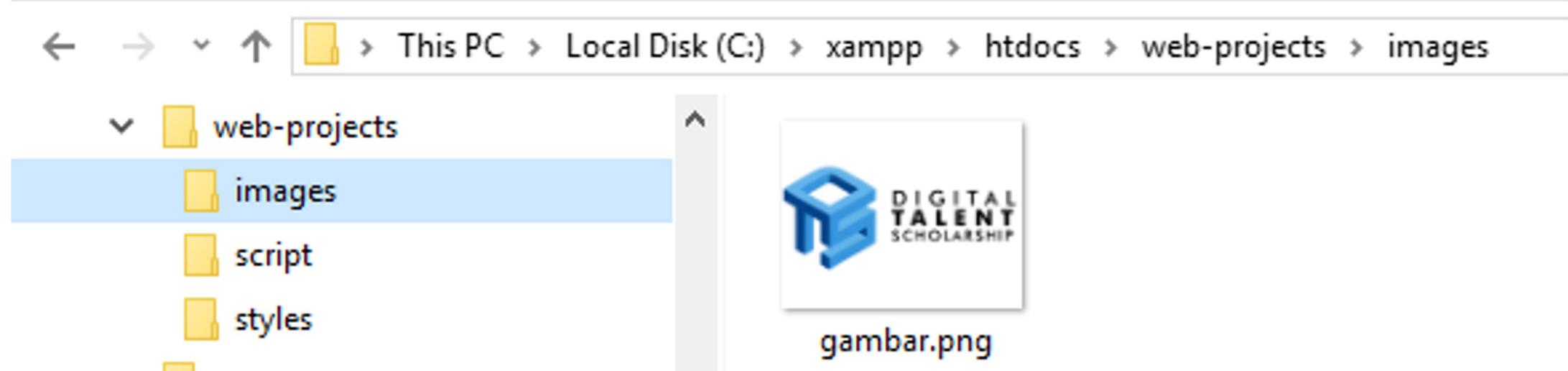
- ❖ Folder images: Folder ini berisi semua gambar yang akan digunakan pada websitemu.
- ❖ Folder styles: Folder ini berisi kode CSS yang digunakan untuk kontenmu (contoh: pengaturan teks dan warna latar belakang).
- ❖ Folder scripts: Folder ini berisi semua kode JavaScript yang digunakan untuk menambah fungsionalitas interaktif pada websitemu.

# Jalur/Path file

- ❖ Agar file-file dapat berkomunikasi antara satu dengan lainnya, kamu harus menyediakan sebuah path file — sebuah rute sehingga satu file dapat mengetahui di mana lokasi file yang lainnya.
- ❖ Untuk mendemonstrasikannya, kita akan membuat sebuah file dengan nama index.html, dan membuatnya menampilkan gambar.

# Jalur/Path file

- ❖ Salin/Copy gambar yang kamu pilih ke dalam folder image.



# Jalur/Path file

- ❖ Buat file index.html, dan tambahkan kode berikut ke dalam file tersebut.

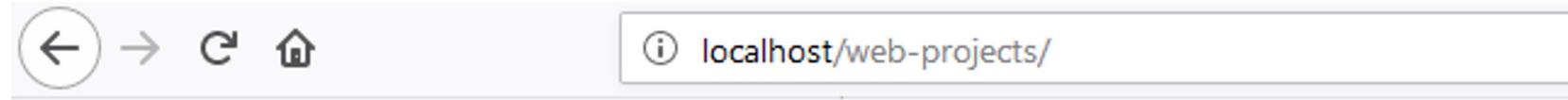
```
1 <html>
2   <head>
3     <meta charset="utf-8">
4     <title>Menampilkan Gambar</title>
5   </head>
6   <body>
7     
8   </body>
9 </html>
```

# Jalur/Path file

- ❖ Baris  adalah kode HTML yang menyisipkan gambar ke dalam halaman.
- ❖ Kita perlu memberitahu HTML di mana letak gambar tersebut.
- ❖ Gambar berada pada folder images yang sejajar dengan file index.html.
- ❖ Untuk menelusuri struktur file dari index.html ke gambar kita, nama file kita beri nama gambar.png, sehingga jalur file-nya menjadi images/gambar.png.
- ❖ Simpan file HTML, kemudian buka halaman tersebut pada web browser.

# Jalur/Path file

- ❖ Hasil dari program diatas adalah sebagai berikut:



DIGITAL  
TALENT  
SCHOLARSHIP

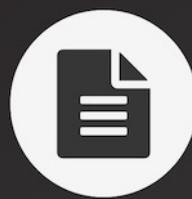
# Mengorganisasikan Sumberdaya Pemrograman

Yang Perlu diperhatikan dalam Mengorganisasikan Sumberdaya Pemrograman:

- ❖ Folder dan nama file diberi nama sesuai dengan fungsinya.
- ❖ Tempatkan File source code yang mempunyai fungsionalitas yang sama dalam satu folder
- ❖ Membuat file Readme.txt

# Membuat file “readme”

# Apersepsi



## Make a README

Because no one can read your mind (*yet*)

# Membuat File Readme.txt

- ❖ README adalah sebuah berkas atau file yang berisi informasi tentang berkas lain di dalam direktori atau arsip dari perangkat lunak komputer.
- ❖ File Readme berisikan informasi tentang :
  - Petunjuk Konfigurasi
  - Petunjuk Instalasi
  - Penjelasan Struktur/Hirarki Program
  - Informasi hak cipta dan perizinan menggunakan perangkat lunak
  - dll

# Organisasi Sumber Daya Pemrograman

- File “readme” dibuat, mengandung penjelasan mengenai struktur/hirarki folder serta penjelasan mengenai sumber daya pemrograman.
- Biasanya file README memiliki ekstensi md (markdown)

## FreeGeoIP for Laravel 4 ← Title

build passing stable v1.0.1 downloads 2.27 unstable dev-master license MIT ← Badges

Laravel 4 Library for calling <http://freegeoip.net/> API.

In contrary to all other packages wherein it requires that you have the geoip database in your filesystem, this library calls a free service So you dont really have to worry about downloading and maintaining geoip data from Maxmind in your own server.

Just install the package, add the config and it is ready to use! ← Description

### Requirements

← Requirements

- PHP >= 5.3.7
- cURL Extension

### Installation

← Installation

```
composer require buonzz/laravel-4-freegeoip:dev-master
```

Add the service provider and facade in your config/app.php

Service Provider

```
Buonzz\GeoIP\Laravel4\ServiceProvider
```

Facade

```
'GeoIP'      => 'Buonzz\GeoIP\Laravel4\Facades\GeoIP',
```

### Usage

← Usage

Get country of the visitor

## Bagian-bagian README file

- Title
- Badges
- Description
- Requirements
- Installation
- Usage
- Credits (Optional)

# Struktur Folder

## Directory Structure

ini adalah struktur folder yang terdapat dalam project ini :

```
|-- public_html
|   |-- css
|   |   |-- **/*.css
|   |-- favicon.ico
|   |-- images
|   |   |-- **/*.png
|   |-- js
|   |   |-- **/*.js
|   |-- index.php
|-- resource
|   |-- config.php
|   |-- library
|   |   |-- template
|-- Gruntfile.js/gulpfile.js
|-- README.md
|-- package.json
|-- bower.json (if using bower)
|-- .gitignore
```

Pada readme.md  
juga dapat  
dilengkapi dengan  
struktur folder dari  
project yang anda  
buat

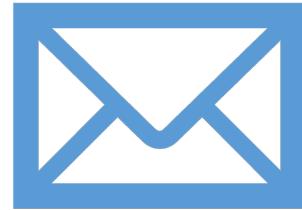
# Kesimpulan

# Kesimpulan

1. **Fungsi** dan **Prosedur** dipakai untuk **efisiensi** penulisan *source code* karena dapat digunakan berulang-ulang.
2. **Mengorganisasikan** sumber daya pemrograman memudahkan untuk pemetaan pengembangan program lanjutan maupun kerja tim
3. Penyusunan sumberdaya pemograman dilakukan sesuai konteks serta dapat sertai dengan readme yang mengandung penjelasan mengenai struktur/ hirarki folder dan sumber daya pemograman

# Referensi

1. Anonymous.(n.d.). *Apache HTTP Server Documentation Version 2.2*. Retrieved from <http://httpd.apache.org/docs/2.2/>.
2. Achour, M., Betz, F. (n.d.), *PHP Manual*. Retrieved from <http://www.php.net/download-docs.php>.
3. Anonymous. (n.d.). *MySQL Reference Manual*. Retrieved from <http://downloads.mysql.com/docs/>.
4. Naramore, E., Gerner, J., Le Scouarnec, Y., Stolz, J., Glass, M. K. (2005). *Beginning PHP5, Apache, and MySQL® Web Development*. Indianapolis, IN: Wiley Publishing, Inc.
5. PHP 5 Tutorial, diakses dari laman <https://www.w3schools.com/php/>, pada 26 April 2019
6. Tizag PHP, diakses dari laman <http://www.tizag.com/phpT/comment.php>, pada 27 April 2019



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