

Modul Pertemuan 1 & 2 PBO

Dasar-dasar PHP

Apa Itu PHP?

PHP adalah *bahasa pemrograman web yang digunakan untuk men-generate atau menghasilkan kode HTML* (Hyper Text Markup Language). PHP merupakan singkatan dari **PHP: Hypertext Preprocessor**. Singkatan ini disebut singkatan *rekursif*, yakni permainan kata dimana kepanjangan PHP juga terdiri dari singkatan PHP itu sendiri, yakni **PHP: Hypertext Preprocessor**. **Hypertext Preprocessor** bisa diterjemahkan sebagai '*pemroses hypertext*', atau '*pemroses HTML*'.

"PHP adalah bahasa pemrograman script server-side yang di desain untuk pengembangan web, dan juga bisa digunakan sebagai bahasa pemrograman umum."

HTML sendiri biasa kita gunakan untuk design tampilan halaman web, apakah kalian sudah pernah belajar HTML? Apabila belum saya harap kalian belajar mandiri mengenai HTML untuk mendukung materi ini. Apa hubungan antara PHP dan HTML akan kita lihat dalam contoh.

PHP sebagai Server Side Programming Language

Jika dibagi menurut tempat dimana kode program diproses, bahasa pemrograman web dapat dibagi menjadi 2 jenis: client side programming language dan server side programming language.

Client-side Programming Language

Client side programming language (bahasa pemrograman berbasis client) adalah jenis bahasa pemrograman yang berjalan dan di proses di sisi client. Ketika berbicara tentang pemrograman web, client ini mengacu ke pengunjung website (user), atau lebih spesifik lagi kepada web browser yang digunakan user ketika mengakses website.

HTML dan CSS merupakan contoh dari bahasa pemrograman client side ini (meskipun keduanya lebih pas disebut sebagai "bahasa kode" daripada "bahasa pemrograman"). Ketika kita membuka website yang berisi kode HTML dan CSS, kode tersebut akan diproses di komputer user (dalam web browser).

JavaScript adalah contoh lain dari bahasa pemrograman client side. Seluruh kode JavaScript akan diproses di web browser pengunjung, yakni di sisi client. Namun saat ini JavaScript juga ada yang berjalan di sisi server contoh NodeJs.

Server-side Programming Language

Di pihak lain, server side programming language (bahasa pemrograman berbasis server) adalah kelompok bahasa pemrograman yang prosesnya berada di server, bukan di komputer pengunjung.

Contoh dari bahasa server side adalah PHP dan ASP.NET. Karena termasuk bahasa pemrograman berbasis server, kita harus menjalankan kode PHP dari sebuah server.

Mengenal Istilah Client – Server

Dalam dunia komputer, **server** adalah sebuah software dan/atau hardware yang berfungsi memproses 'sesuatu' untuk dikembalikan hasilnya kepada **client**. Oleh karena itulah terdapat istilah **client-server**.

Analoginya bisa disamakan dengan memesan makanan di restoran. Dapur tempat masakan disiapkan adalah **server**. Kita sebagai customer adalah **client**. Setiap kali ada permintaan makanan dari kita (client), pramusaji akan meneruskannya ke dapur (server). Tergantung banyaknya permintaan, koki di dapur perlu beberapa saat untuk membuat masakan. Setelah selesai, pramusaji akan mengantar pesanan dari dapur (server) kembali ke kita (client).

Restoran juga memiliki keterbatasan seberapa banyak masakan yang bisa dilayani dalam satu waktu. Jika pesanan datang dalam jumlah banyak dan pada waktu yang bersamaan, koki di dapur bisa kewalahan (*server overload*), dan kita pun terpaksa menunggu lebih lama hingga makanan datang. Hal ini bisa diatasi dengan menambah koki atau memperbesar kapasitas dapur (*upgrade server*).

Setiap koki biasanya punya spesialisasi di bidang tertentu, ada yang spesialis *seafood*, spesialis *dessert*, spesialis nasi goreng, dst. Begitu juga server yang tersedia dalam berbagai fungsi, diantaranya:

- File Server**: menyimpan dan berbagi pakai file (sharing).
- Database Server**: menyimpan dan menampilkan data.
- Email Server**: menyimpan dan mengatur lalu lintas email.
- Web Server**: memproses dan menangani permintaan halaman web.

Secara fisik, server ini hanyalah komputer biasa dengan komponen 'spesial', misalnya menggunakan processor khusus server hingga 8 atau 16 core, memory RAM hingga 16GB atau 32GB, dan harddisk hingga puluhan TeraByte (tergantung kebutuhan). Untuk mengubah sebuah komputer menjadi server, tinggal menginstall aplikasi khusus server. Sebagai contoh, untuk database server tersedia aplikasi MySQL, MariaDB, MS SQL, atau Oracle. Untuk web server, tersedia aplikasi Apache, Nginx, IIS atau LiteSpeed Web Server. Web server inilah yang diperlukan untuk memproses kode program PHP.

Untuk kebutuhan praktikum ini, kita akan menginstall aplikasi XAMPP yang diantaranya terdiri dari web server Apache dan database server MySQL. Akan kita bahas selanjutnya.

Cara Kerja PHP dan Web Server

Sekarang kita akan bahas bagaimana cara kerja PHP dalam menghasilkan kode HTML. Agar bisa memahaminya, kita juga harus membahas fungsi dari web server. Memproses kode PHP hanya salah satu dari fungsi web server. Agar bisa diakses dari internet, file HTML, CSS, JavaScript, PHP serta berbagai file pembangun sebuah website harus ditempatkan ke dalam web server. Di sini web server berfungsi sebagai pengatur arus lalu lintas data dari dan ke web browser pengunjung.

Untuk situs online, fisik web server ini berada di komputer perusahaan web hosting. Sebagai contoh, situs www.komputeronline.com dibangun dengan PHP (menggunakan Wordpress), dimana file-file PHP penyusun website komputeronline berada di harddisk web hosting misal rumahweb.com

Paham ya dasar kerja webserver seperti apa????

Instalasi XAMPP, Web Browser dan Text Editor

OK kita masuk ke pembahasan teknis, lalu apa yang kita perlukan sebagai senjata perang untuk memulai belajar PHP??? Jawabannya adalah **PHP, web server, database MySQL, Text editor dan browser**.

Tahapannya kita bisa menginstal masing-masing perangkat diatas satu persatu, tapi bagi pemula proses ini akan sangat membingungkan yang bisa menyebabkan jadi males untuk memulai belajar PHP.

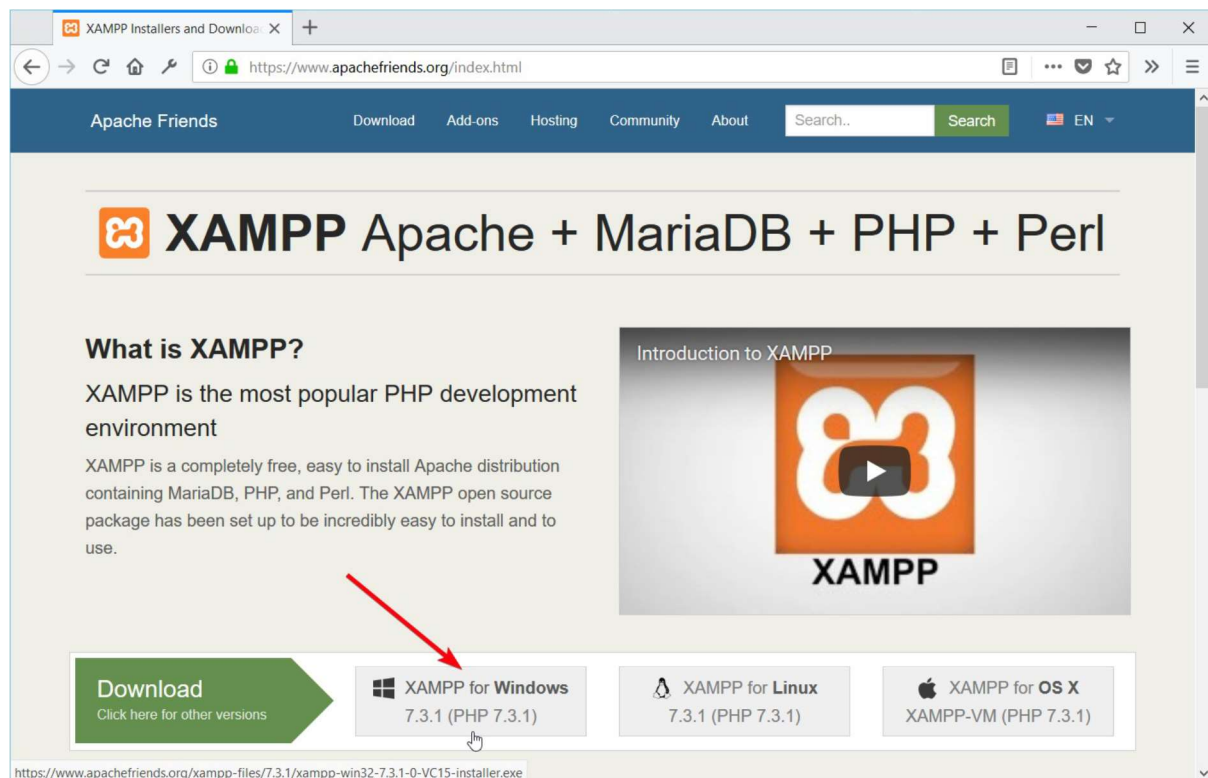
Untungnya, terdapat aplikasi yang "mem-paketkan" web server + PHP + MySQL ke dalam 1 kali proses instalasi. Dengan menginstall aplikasi paket ini, kita sudah mendapat 3 aplikasi lengkap dan sudah siap pakai tanpa perlu mengkonfigurasinya satu per satu. Salah satu aplikasi paket ini adalah **XAMPP**. Nama aplikasi XAMPP terdiri dari paket yang ada di dalamnya. X (berarti cross-platform, maksudnya tersedia dalam berbagai sistem operasi),

Apache Web Server, MySQL Database Server, PHP dan Perl. Selain aplikasi ini, XAMPP juga menyertakan aplikasi pelengkap lain seperti phpMyAdmin, File Zilla FTP Server, serta Mercury Mail Server. XAMPP tersedia gratis dan bisa di download melalui situs resminya di: **apachefriends.org**. Silakan sekarang download aplikasi ini!!!

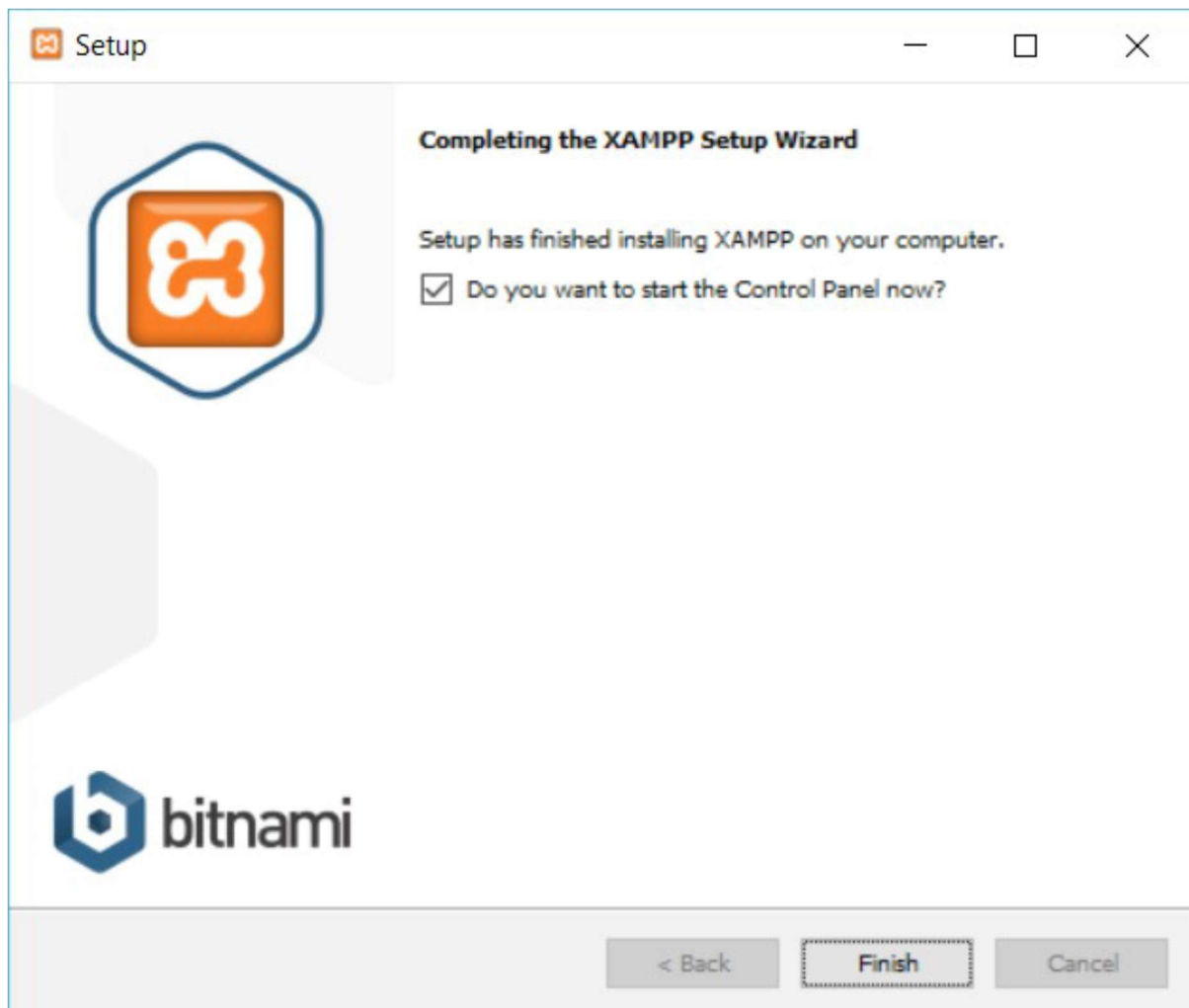
Apabila bertemu dengan istilah MariaDB dan MySQL sementara abaikan dulu, anggap saja itu adalah dua jenis database yang sama, karena sejarahnya panjang, klo penasaran silakan cari sendiri ya sejarah keduanya lahir ☺

Selama praktikum apabila sudah berhubungan dengan database saya akan tetap menyebut MySQL!

Instalasi XAMPP



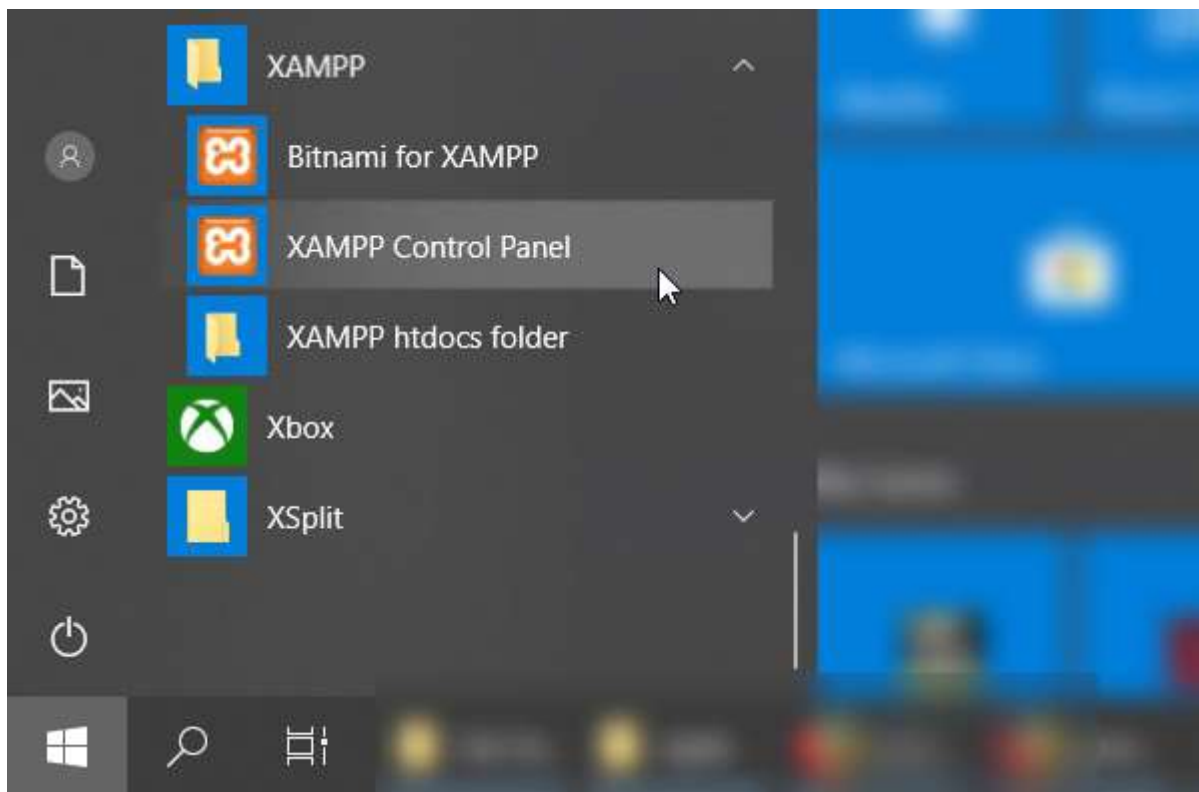
Setelah didownload install xampp seperti biasa, next next next finish, ok sampai akhirnya seperti dibawah



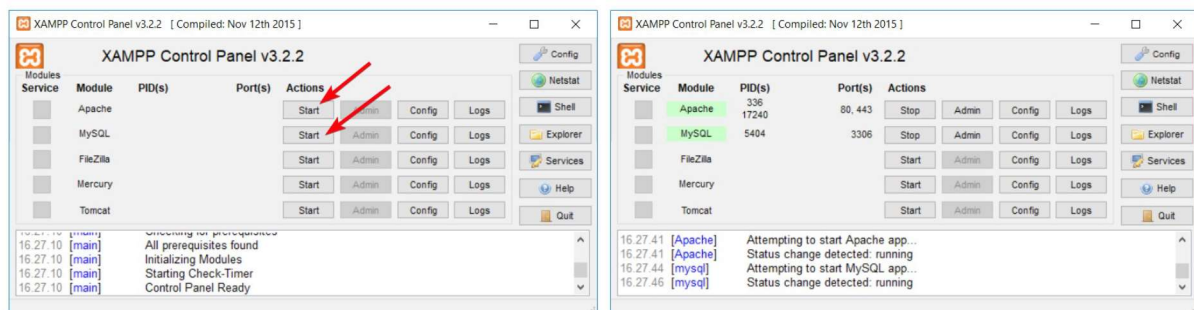
Setelah kita klik finish maka akan muncul halaman control panel berikut:



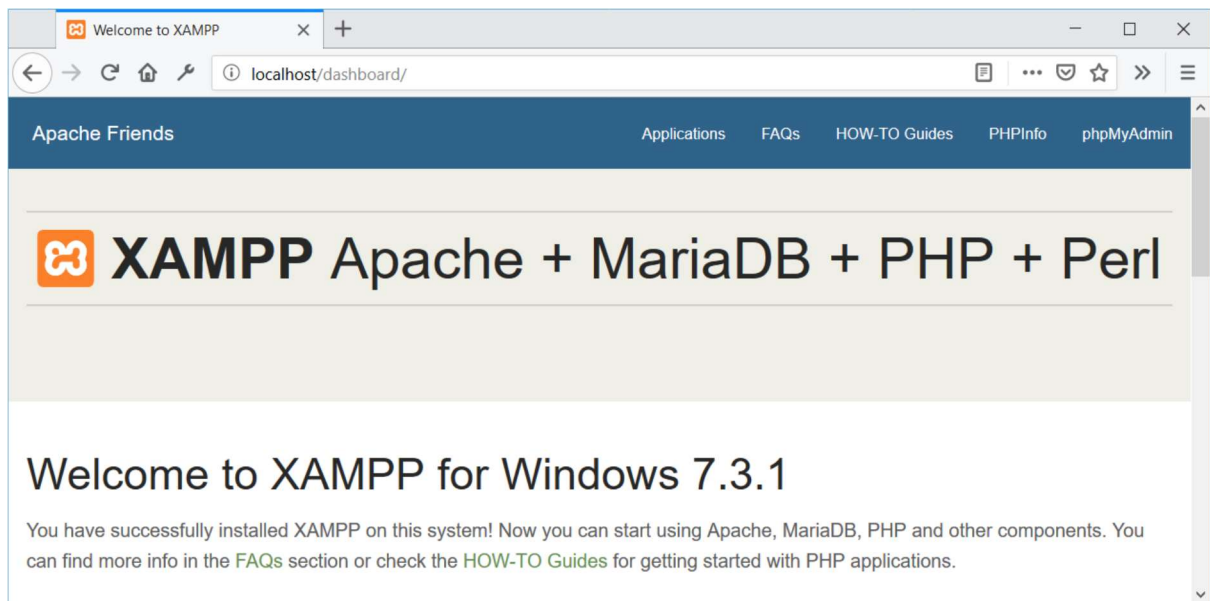
Apabila tidak muncul maka silakan buka program file dan buka xampp control panel dari situ



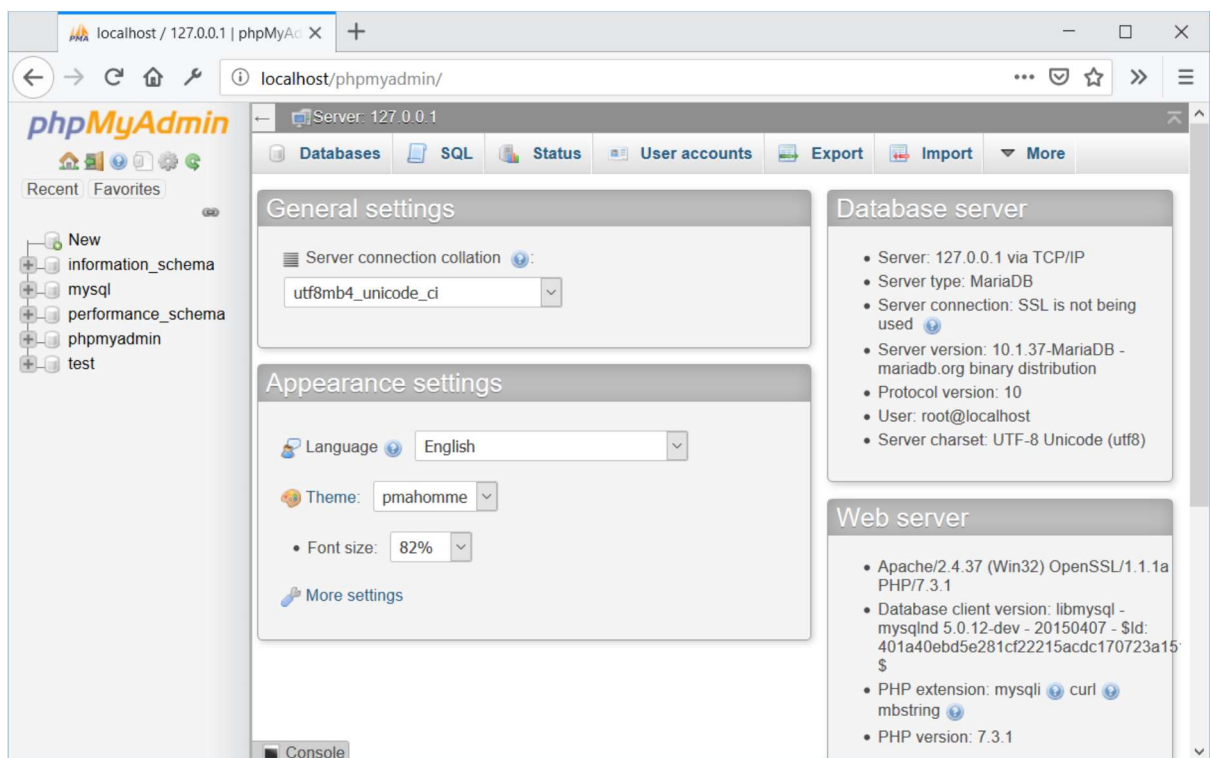
Tahap pengujian apakah proses instalasi kita berhasil adalah sebagai berikut:
Start Apache dan MySql hingga menampilkan status on berwarna hijau,



Selanjutnya, buka web browser dan ketik alamat <http://localhost> pada address bar dan tekan Enter. Jika tampil jendela pembuka XAMPP (halaman *dashboard*) seperti dibawah, maka semuanya telah terinstall dengan baik.
Apabila tidak, silakan diskusikan di forum atau bertanya kepada saya,

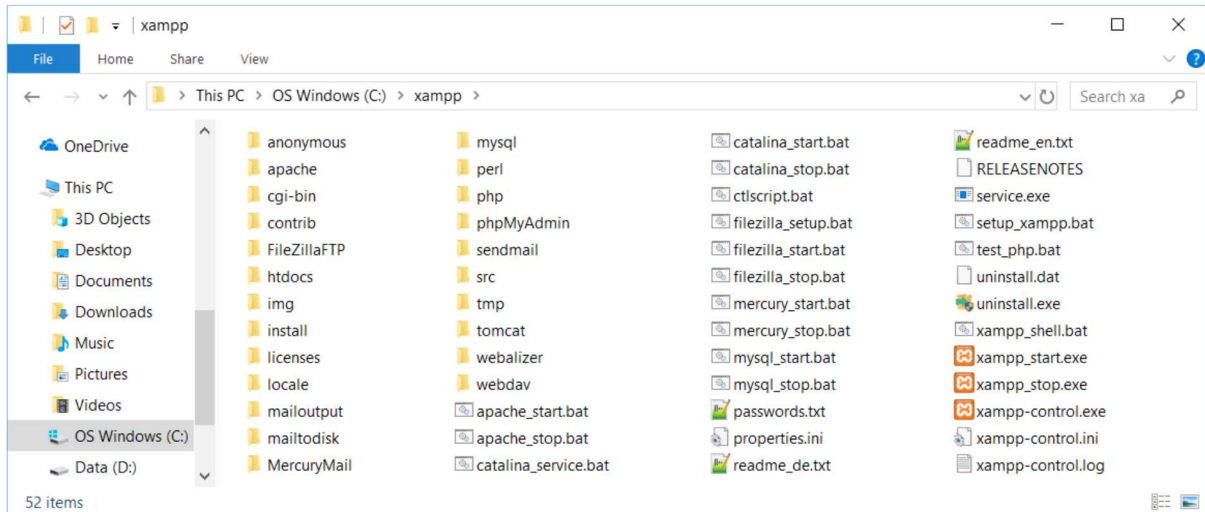


Untuk menguji apakah database MySQL (atau tepatnya MariaDB) sudah terkoneksi dan bisa diakses, ketik alamat <http://localhost/phpmyadmin> pada browser, apabila muncul tampilan seperti dibawah maka database MySQL sudah berjalan dengan baik.



Melihat Folder Instalasi XAMPP

Seperti yang telah kita bahas, XAMPP pada dasarnya 'mengumpulkan' berbagai aplikasi yang diperlukan untuk menjalankan PHP. Aplikasi ini ditempatkan dalam folder instalasi XAMPP. Silahkan buka folder instalasi XAMPP yang jika anda mengikuti settingan default, ini berada di C:\xampp\. Dalam folder inilah berbagai file dan folder penyusun XAMPP berada.



Folder htdocs

Folder htdocs menjadi folder yang paling penting dan yang akan paling sering kita akses nantinya. Karena di sinilah file PHP harus ditempatkan agar bisa diproses oleh XAMPP. Jika file PHP disimpan di folder lain (walaupun masih di dalam folder instalasi xampp), kode program tersebut TIDAK AKAN BERJALAN. Kita akan kembali membahas penggunaan folder htdocs ini dalam bab berikutnya.

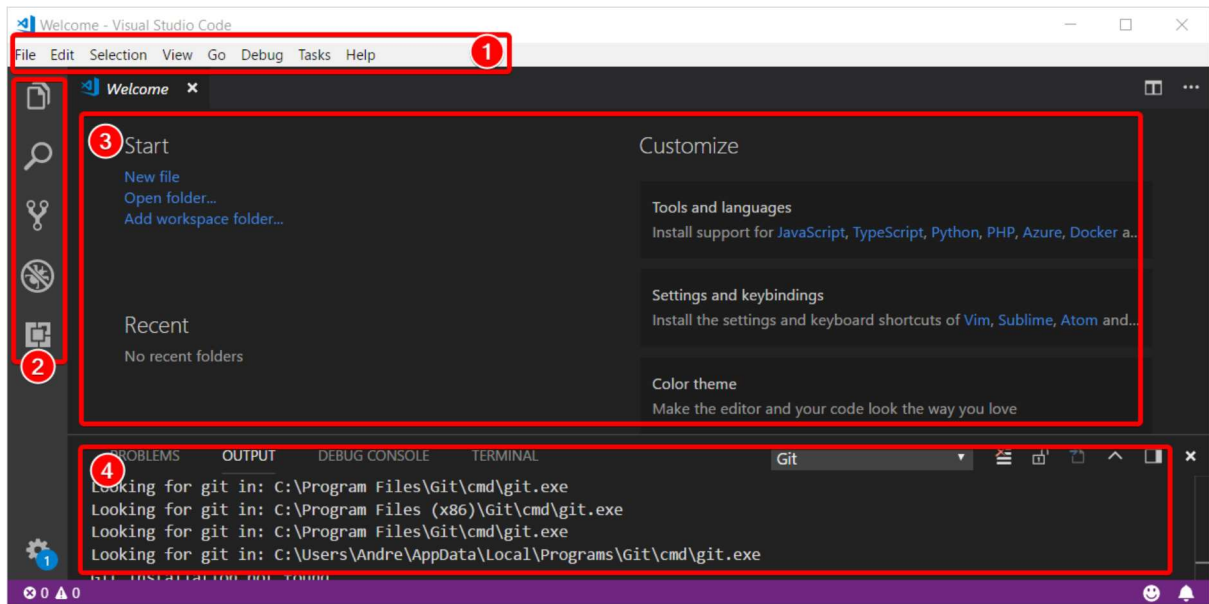
Instalasi Text Editor

File PHP pada dasarnya hanyalah file teks biasa dengan aturan penulisan khusus. Kurang lebih sama seperti file HTML. Untuk menulis kode PHP, kita bisa memakai aplikasi teks editor apapun. Bahkan aplikasi Notepad bawaan Windows juga bisa dipakai.

Untuk text editor kalian boleh menggunakan apapun, tapi yang saya gunakan secara pribadi adalah aplikasi VS Code. VS Code merupakan sebuah aplikasi teks editor yang sangat lengkap. Bahkan ada yang menyebut bahwa VS Code sebenarnya tidak cocok lagi disebut sebagai teks editor, tapi sudah sebagai IDE (Integrated Development Environment) karena banyaknya fitur yang tersedia. Walaupun dibuat oleh Microsoft (sebagai pengembang utama), VS Code dirilis sebagai project open source dan bisa di download gratis dari web resminya: <https://code.visualstudio.com>

Silakan download dan install apabila belum memiliki.

Tampilan jendelanya kira2 seperti dibawah ini

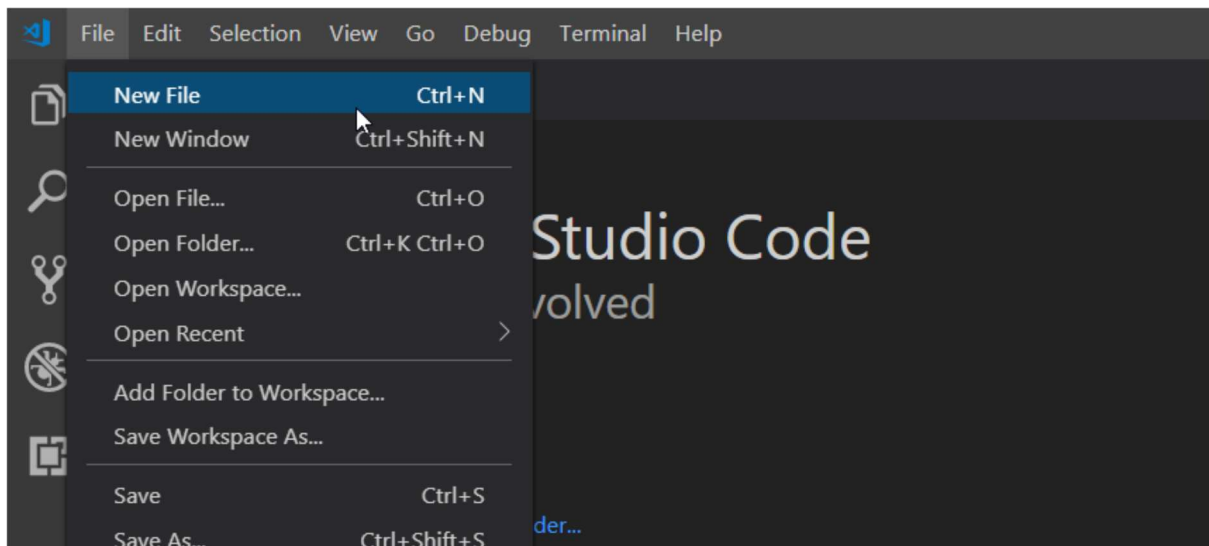


Ok sampai disini senjata yang kita persiapkan sudah lengkap, saatnya kita ngoding.PHP !!!

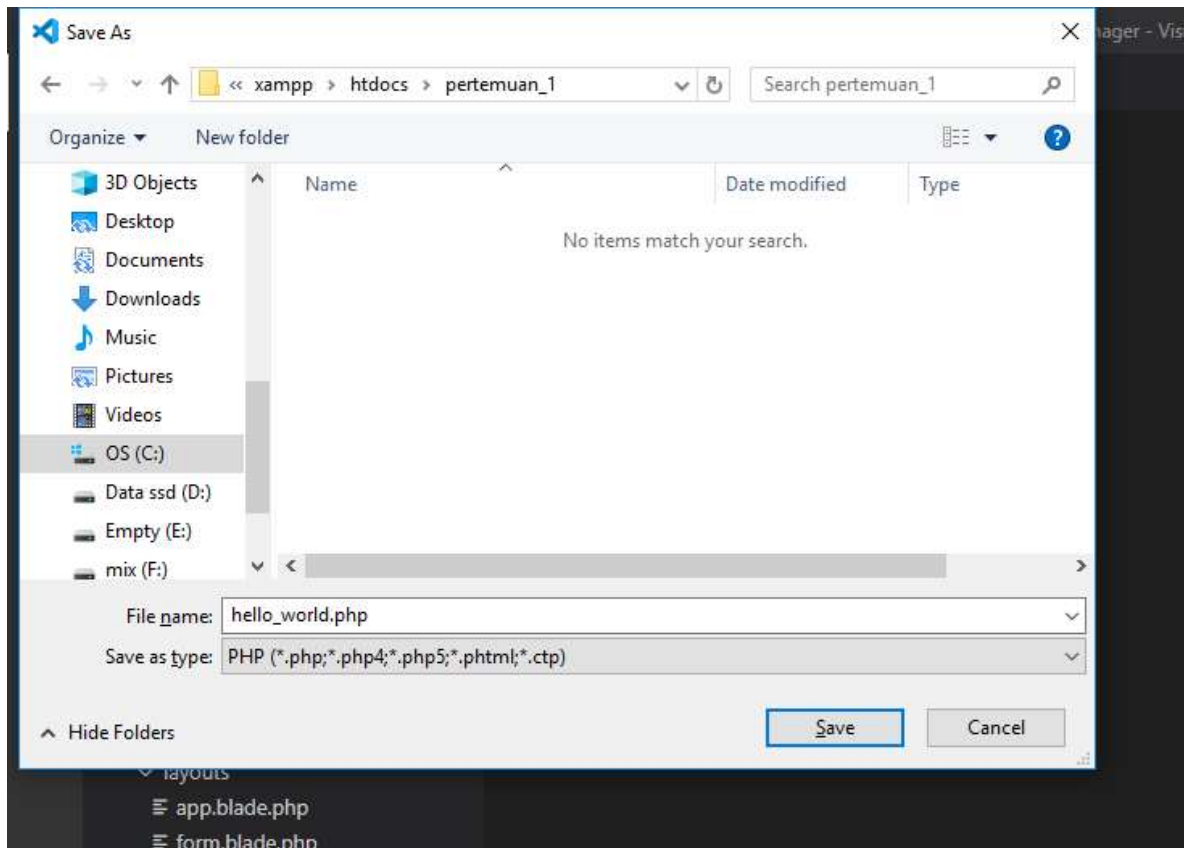
Menampilkan Teks "Hello World"

Sudah menjadi tradisi di dunia pemrograman untuk menampilkan teks "Hello World" ketika mempelajari bahasa pemrograman baru. Dan inilah yang akan kita lakukan.

Buat file baru, Langsung simpan filenya dalam kondisi kosong dengan nama **hello_world.php** di dalam folder **C:\xampp\htdocs\pertemuan_1** buat folder baru didalam htdocs dengan nama pertemuan_1 agar lebih rapih



Kenapa harus langsung disimpan sebelum menulis coding, agar fitur auto complete untuk syntax PHP di visual code aktif.



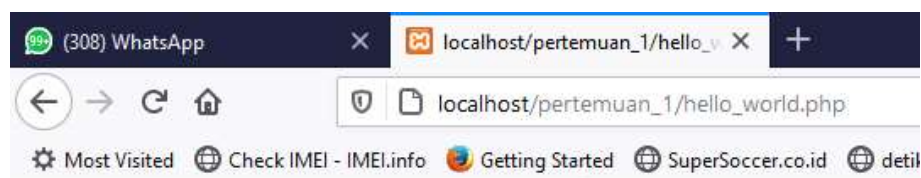
Ok sekarang kita buat kode PHP pertama kita

```
<?php
    echo "Hello World";
?>
```

Silakan simpan kembali file nya sebelum kita cek hasilnya di browser.

Ok sekarang kita kan lihat hasil kode PHP pertama kita, pastikan bahwa apache dan MySQL server pada XAMPP telah **running**. Apabila sudah silakan buka browser dan ketikan pada URL:

localhost/pertemuan_1/hello_world.php



Hello World

Apabila kalian meletakkan file `hello_world.php` langsung dibawah htdocs maka cara aksesnya `localhost/hello_world.php` jadi untuk akses perhatikan dimana letak file PHP kalian di dalam direktori `htdocs`. Paham??? Apabila ada yang berkendala silakan diskusikan digrup yang ada.

Dari contoh file `hello_world.php` ini, kita dapat melihat bagaimana cara menulis dan menjalankan sebuah file PHP:

- ✓ Setiap file PHP harus disimpan dengan extension `*.php`.
- ✓ PHP diawali dengan kode `<?php` dan ditutup dengan kode `?>`.
- ✓ Perintah `echo` digunakan untuk menampilkan teks ke dalam web browser.
- ✓ Seluruh file PHP harus berada di dalam folder `htdocs`.
- ✓ Untuk menjalankan file PHP, dapat diakses dengan alamat `localhost/nama_file.php`.

Mengenal `index.php`

File dengan nama `index.php` mendapat perlakuan khusus dari web server. Sebuah file dengan nama `index` akan menjadi halaman *default* ketika diakses hanya dengan memanggil nama folder saja (tanpa mencantumkan nama file).

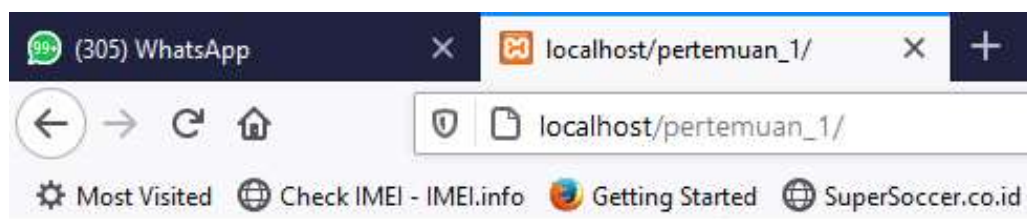
Sebagai contoh, jika di web browser anda mengetik alamat `http://localhost/pertemuan_1/`, secara otomatis file `index.php` akan tampil. Perhatikan bahwa yang kita ketik adalah alamat folder, bukan alamat lengkap file: `http://localhost/pertemuan_1/index.php`.

Ok agar lebih paham kita akan buat contohnya, sekarang kita tambahkan sebuah file dengan nama `index.php` yang akan kita simpan di folder `pertemuan_1`, berisi syntax sederhana.

New file pada visual code dan langsung simpan dengan nama `index.php`, simpan pada direktori `pertemuan_1`

```
<?php
    echo 'ini dari file index';
?>
```

Lalu akan kita coba akses melalui browser.



ini dari file index

Ok terlihat tanpa menuliskan file index.php pada url maka file index akan tetap diakses pertama kali, tapi apabila kita ingin mengakses file hello_world.php tetap harus menuliskan nama filenya. Itulah keunikan dari file index dibandingkan dengan file yang lain.

Pada implementasinya kode PHP akan sering disandingkan dengan HTML, apabila kalian sudah baca2 mengenai HTML, HTML berperan untuk mengatur tampilan dari halaman web yang kita buat, sedangkan PHP lebih untuk pemrosesan data yang akan ditampilkan.

Sehingga apabila kita hanya menggunakan PHP tampilan yang muncul pada halaman web yang kita buat tidak akan menarik sama sekali. Tolong HTML dipelajari juga ya mudah kok!!!

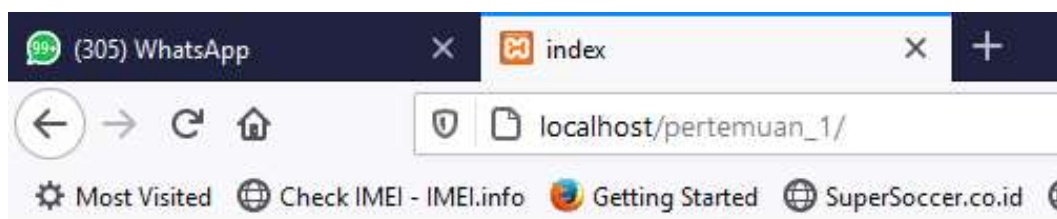
Sekarang kita akan coba mengubah file index.php yang kita buat dengan menambahkan sedikit dasar kode HTML.

Ubah menjadi berikut dan simpan masih dengan nama index.php:

```
<!DOCTYPE html>
<html lang="id">
<head>
  <title>index</title>
</head>
<body>
  <!-- saat mulai menulis php dalam html harus diawali dengan tag php-->
  <?php
    echo "kode gabungan HTML dan PHP";
    echo "<br>"; //cara menulis tag HTML dalam PHP
                //fungsi tag <br> untuk memberikan spasi antar baris
    echo "tidak terlalu susah";
  ?>
</body>
</html>
```

Diatas terlihat tag PHP berada didalam barisan tag HTML, please belajar mandiri untuk HTML nya ya, karena waktu tidak akan memungkinkan bila harus membahas HTML juga.

Bila dijalankan kode diatas pada browser akan tampil:



kode gabungan HTML dan PHP
tidak terlalu susah

Variabel dan Konstanta

Dalam bab kita akan membahas salah satu inti dari PHP (dan bahasa pemrograman secara umum). Kita akan mempelajari pengertian variabel, cara penulisan variabel serta perbedaannya dengan konstanta.

Pengertian Variabel

Dalam definisi sederhana, **variabel** adalah *kode program yang digunakan untuk menampung nilai dari sebuah data*. Nilai ini bisa berupa angka, teks, objek, dan lain-lain. Sebuah variabel memiliki **nama** yang digunakan untuk mengakses nilai dari variabel tersebut.

Aturan Penulisan Variabel

Untuk membuat variabel, PHP memiliki aturan penulisan sebagai berikut:

- ✓ Sebuah variabel harus diawali dengan tanda **dollar (\$)**
- ✓ Setelah tanda dollar, karakter pertama harus berupa huruf atau *underscore* (`_`).
- ✓ Karakter kedua dan seterusnya bisa berupa huruf, angka, atau *underscore* (`_`).
- ✓ Nama variabel bersifat **case sensitif** (huruf besar dan huruf kecil dianggap berbeda).
- ✓ Untuk memberikan nilai kepada sebuah variabel, PHP menggunakan karakter sama dengan `=`.
- ✓ Variabel dalam PHP tidak memerlukan deklarasi terlebih dahulu.

Untuk latihan variable silakan buat file baru dengan nama `variable.php` dan simpan pada direktori pertemuan_1

```
<?php
//contoh penulisan variable yang benar pada PHP
$i;
$nama;
$Umur;
$_lokasi_memori;
$ANGKA_MAKSIMUM;

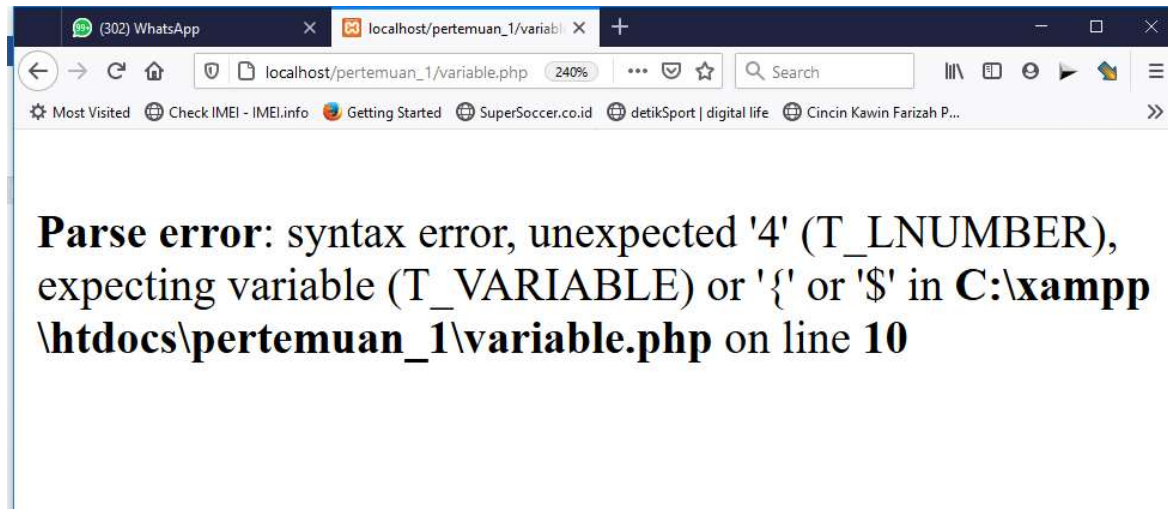
//contoh penulisan variable yang salah
$4ever; //variabel tidak boleh diawali dengan angka
$_salah satu; //varibel tidak boleh mengandung spasi
$nama*^; //variabel tidak boleh mengandung karakter khusus: * dan ^

//cara pengisian nilai pada sebuah variable
$nama = "andi"; //variable berisi teks maka harus diapit dengan ""
$umur = 17; //variable berupa angka maka tanpa tanda kutip ""
$pesan = "Saya sedang belajar PHP";
$jarak = 18.45; //penulisan nilai untuk bilangan desimal
?>
```

Apabila kode program diatas dijalankan tidak akan menampilkan apapun pada browser, KENAPA??? ADA YANG TAU???

Yup benar sekali, kita belum menambahkan perintah echo untuk menampilkan nilai dari variable pada browser, akan kita lakukan setelah ini.

Tapi silakan coba panggil file diatas pada browser



Akan muncul PESAN ERROR,

INGAT SAAT KALIAN BELAJAR PEMROGRAMAN KALIAN HARUS MULAI MEMBIASAKAN DIRI DENGAN ERROR PROGRAM, JANGAN MALES BACA ERROR KARENA INI BAGIAN TAK TERPISAHKAN DARI SEORANG PROGRAMMER, PEPATAH JAWA MENGATAKAN MEMBUAT PROGRAM TANPA BERTEMU ERROR ITU SEPERTI MEMBUAT SAYUR TANPA GARAM, HAMBAR!! 😊

Maaf capslock rusak.

Ok kembali ke error, apabila kita baca error diatas maka bisa kita baca secara sederhana, terjadi kesalahan penulisan 4 pada file variable.php baris ke 10. Apa baris ke 10 program kita tadi?

```
$4ever; //variabel tidak boleh diawali dengan angka
```

Nah, sesuai prediksi bahwa penulisan variable tidak boleh diawali dengan angka, otomatis akan ditampilkan sebagai error dan error akan hilang apabila kita memperbaiki syntax program kita.

Ok paham ya, SAMPAI BERTEMU DI ERROR ERROR SELANJUTNYA, jangan kapok.

Sekarang kita perbaiki errornya

Ubah bagian yang error menjadi

```
//contoh penulisan variable yang salah
/* $4ever; //variabel tidak boleh diawali dengan angka
$_salah satu; //variabel tidak boleh mengandung spasi
$nama*; //variabel tidak boleh mengandung karakter khusus: * dan ^ */
```

Kalian tambahkan fungsi komen agar baris program tidak dieksekusi.

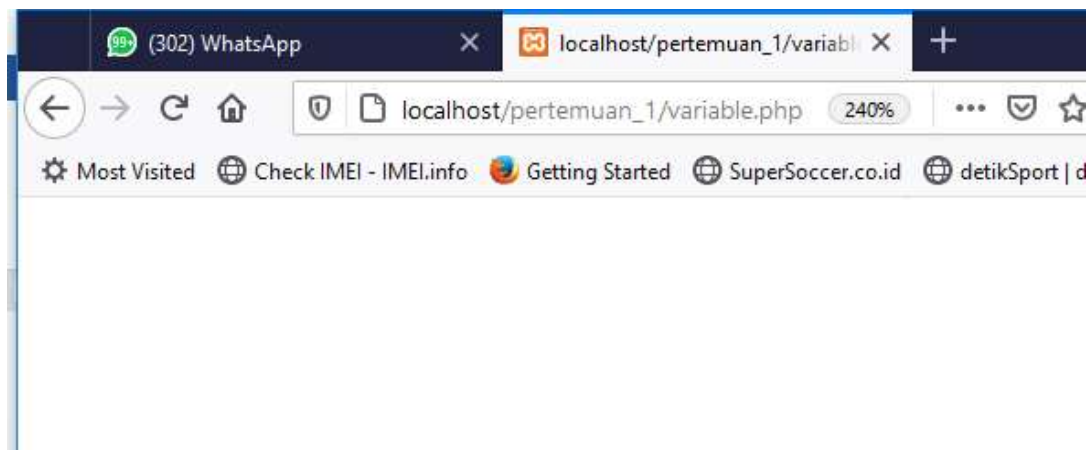
Oya teks berwarna hijau diatas artinya sebuah komentar, dan tidak akan dianggap sebagai baris program.

Dalam PHP ada beberapa cara penulisan komentar berikut beberapa diantaranya:

```
// ini komentar  
# ini komentar  
/* ini komentar untuk beberapa baris  
    Sekaligus */
```

Ok, kembali ke program

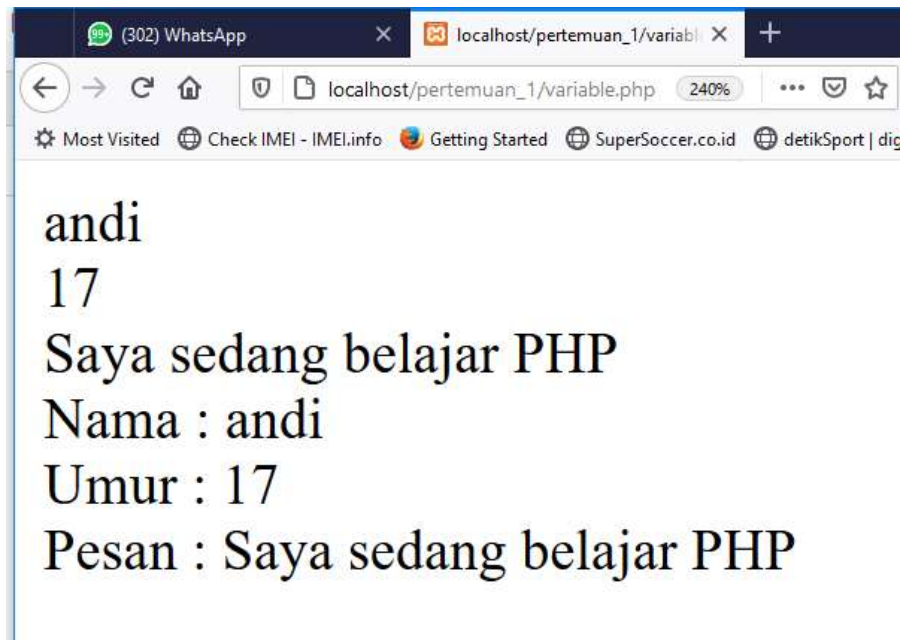
Sekarang kita coba panggil ulang melalui browser setelah kita perbaiki programnya.



Tidak ada error dan tidak menampilkan apapun juga, ingat kenapa?? Karena kita belum menambahkan fungsi echo untuk menampilkan hasilnya, ok sekarang kita tambahkan masih di file yang sama tambahkan baris program berikut, saya tidak menulis ulang kode program sebelumnya untuk menghemat space.

```
// menampilkan isi variable  
echo $nama;  
echo "<br>"; //untuk memberi spasi antar baris  
echo $umur."<br>"; // cara kedua menulis <br> dengan menambahkan .(concat)  
echo $pesan;  
  
echo "Nama : $nama <br> Umur : $umur <br> Pesan : $pesan"; //bisa seperti ini  
juga
```

lalu simpan dan panggil ulang melalui browser. INGAT TULIS BARIS PROGRAM DENGAN TELITI, KURANG PENULISAN KARAKTER BISA MENYEBABKAN ERROR.



SILAKAN RENUNGI DAN RESAPI SERTA PAHAMI DULU APA YANG TELAH KALIAN BUAT DIATAS, BILA ADA YANG PERLU DITANYAKAN BISA MELALUI FORUM.

Tipe Data PHP

Di dalam PHP, terdapat 8 jenis tipe data yang bisa dikelompokkan menjadi 3 bagian:

Tipe data dasar / tipe data *primitive* / tipe data *scalar*:

- ✓ **Integer** (angka bulat)
- ✓ **Float / Double** (angka pecahan)
- ✓ **Boolean** (logika true atau false)
- ✓ **String** (teks)

Tipe data *composite* / tipe data *compound*:

- ✓ **Array**
- ✓ **Object**

Tipe data khusus:

- ✓ **Resource**
- ✓ **Null**

kita akan bahas dulu 4 tipe data dasar

kita buat file baru lagi dengan nama **tipe_data.php** dan buat baris program berikut:

```
<?php
    $var1=12; //tipe integer
    $var2=15.18; //tipe float
    $var3="belajar PHP"; //tipe string
    $var4=-210; //tipe integer negative

    echo $var1."<br>";
```

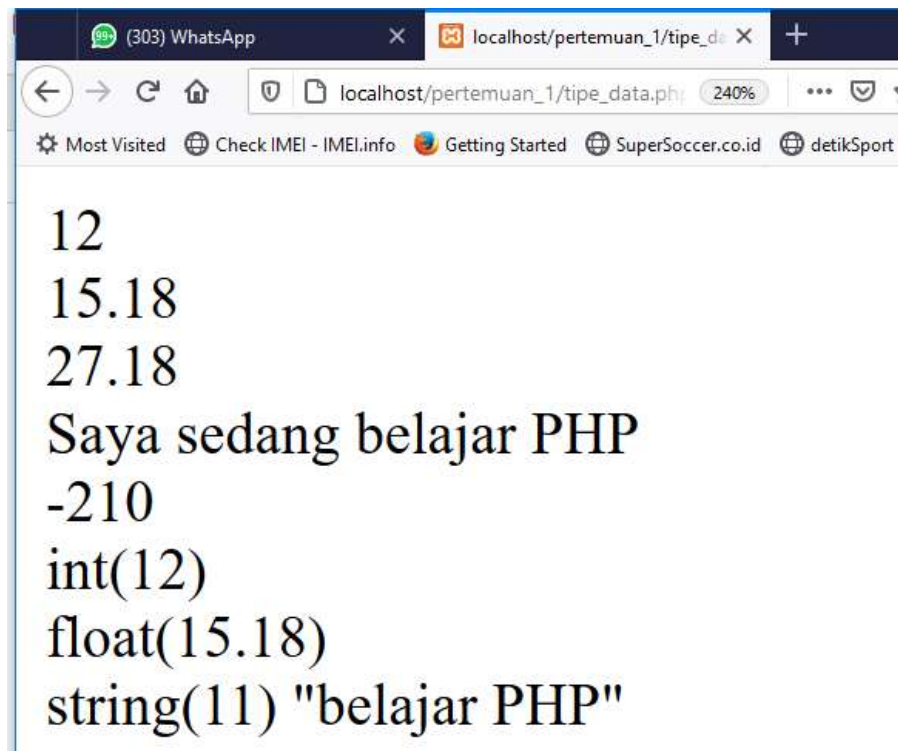
```

echo $var2."<br>";
echo $var1 + $var2."<br>"; //menjumlahkan 2 bilangan
echo "Saya sedang $var3 <br>"; //menggabungkan string dengan variable
echo $var4."<br>";

var_dump($var1); //melihat jenis tipe data suatu variable
echo "<br>";
var_dump($var2); //melihat jenis tipe data suatu variable
echo "<br>";
var_dump($var3); //melihat jenis tipe data suatu variable
echo "<br>";
?>

```

Lalu panggil melalui browser dan pahami hasil tampilannya.



Operator PHP

Operator adalah sesuatu yang menghasilkan nilai dari satu atau lebih data. Sebagai contoh, tanda tambah (+) adalah operator aritmatika yang menghasilkan nilai dari penambahan dua buah angka.

Operand adalah nilai asal yang digunakan oleh operator. Sebagai contoh, dalam operasi 5+2, angka 5 dan 2 disebut sebagai **operand**. Operasi tersebut membutuhkan 2 buah operand dan 1 operator.

Contoh:

A + B * C

maka A dan B adalah operand, + , * merupakan operator

operator dan operan tidak saya bahas lebih jauh karena asumsinya kalian sudah pernah mendapatkan materi ini pada mata kuliah dasar pemrograman

berikut tabel operator aritmatika dalam PHP

Nama Operator	Operator	Contoh	Hasil
positif / plus	+	+\$a	nilai positif dari \$a
negatif / minus	-	-\$a	nilai negatif dari \$a
penambahan	+	\$a + \$b	total dari \$a dan \$b
pengurangan	-	\$a - \$b	selisih dari \$a dan \$b
perkalian	*	\$a * \$b	hasil kali dari \$a dan \$b
div/pembagian	/	\$a / \$b	hasil bagi dari \$a dan \$b
modulus/sisa hasil bagi	%	\$a % \$b	sisa pembagian \$a bagi \$b
eksponensial*	**	\$a ** \$b	hasil dari \$a pangkat \$b

Contoh implementasi program, silakan dicoba buat file dengan nama **aritmatika.php**

```
<?php
$hasil1 = +11;
$hasil2 = -3;
$hasil3 = 3 + 5;
$hasil4 = 8 - 4.5;
$hasil5 = 2 * 5;
$hasil6 = 3 + 8 / 5 - 3;
$hasil7 = 10 % 4;
$hasil8 = 2 ** 4;

var_dump($hasil1); echo "<br>"; // int(11)
var_dump($hasil2); echo "<br>"; // int(-3)
var_dump($hasil3); echo "<br>"; // int(8)
var_dump($hasil4); echo "<br>"; // float(3.5)
var_dump($hasil5); echo "<br>"; // int(10)
var_dump($hasil6); echo "<br>"; // float(1.6)
var_dump($hasil7); echo "<br>"; // int(2)
var_dump($hasil8); // int(16)
?>
```

Fungsi `var_dump` pada PHP salah satunya untuk mengetahui isi dan tipe data dari sebuah variable.

Operator Increment dan Decrement

Nama Operator	Contoh	Hasil
pre-increment	<code>++\$a</code>	tambah nilai \$a dengan 1, lalu kirim nilai \$a
post-increment	<code>\$a++</code>	kirim nilai \$a, lalu tambah nilai \$a dengan 1
pre-decrement	<code>--\$a</code>	kurangi nilai \$a dengan 1, lalu kirim nilai \$a
post-decrement	<code>\$a--</code>	kirim nilai \$a, lalu kurangi nilai \$a dengan 1

Kerjakan contoh berikut untuk memahaminya, buat file dengan nama **inc_dec.php** simpan di direktori yang sama dengan latihan sebelumnya, jalankan dan pahami outputnya

```
<?php
$a = 5;
echo ++$a;    // nilai $a langsung ditambah menjadi 6
echo $a;      // 6
echo "<br>";

$b = 5;
echo $b++;    // nilai $a masih tetap 5
echo $b;      // akan bertambah menjadi 6 saat dicetak kembali
echo "<br>";

$a = 5;
echo --$a;    // 4
echo $a;      // 4
echo "<br>";

$b = 5;
echo $b--;    // 5
echo $b;      // 4
```

operator post increment dan pre-increment sering digunakan pada saat proses iterasi atau looping. **INGATT!!**

OPERATOR PERBANDINGAN

Nama Operator	Operator	Contoh	Hasil
Sama dengan	==	\$a == \$b	TRUE jika nilai \$a sama dengan \$b
Identik dengan	===	\$a === \$b	TRUE jika nilai \$a sama dengan \$b, dan memiliki tipe data yang sama
Tidak sama dengan	!=	\$a != \$b	TRUE jika nilai \$a tidak sama dengan \$b
Tidak sama dengan	<>	\$a <> \$b	TRUE jika nilai \$a tidak sama dengan \$b
Tidak identik dengan	!==	\$a !== \$b	TRUE jika nilai \$a tidak sama dengan \$b, dan memiliki tipe data yang tidak sama
Kurang dari	<	\$a < \$b	TRUE jika nilai \$a kurang dari \$b
Lebih dari	>	\$a > \$b	TRUE jika nilai \$a lebih dari \$b
Kurang dari atau sama dengan	<=	\$a <= \$b	TRUE jika nilai \$a kurang dari atau sama dengan \$b
Lebih dari atau sama dengan	>=	\$a >= \$b	TRUE jika nilai \$a lebih dari atau sama dengan \$b

Operator ini sering digunakan pada proses pengambilan keputusan, atau membandingkan suatu nilai. Dan **INGAT!!!** Output dari operator perbandingan adalah nilai TRUE atau FALSE, **INGATT INGATT!!**

Contoh implementasi, buat file dengan nama **perbandingan.php**

```
<?php
var_dump(12 < 14);    echo "<br />";    // bool(true)
var_dump(14 < 14);    echo "<br />";    // bool(false)
var_dump(14 <= 14);   echo "<br />";    // bool(true)
var_dump(10 <> 14);    echo "<br />";    // bool(true)
var_dump(15 == 10);   echo "<br />";    // bool(false)
var_dump(10 === 10); echo "<br />";    // bool(true)
var_dump(150 == 1.5e2); // bool(true)

//perbandingan teks/karakter
var_dump('andi' < 'andri'); // bool(true)
echo "<br>";
var_dump('anto' < 'anti'); // bool(false)
```

cara baca fungsi diatas misal: baris satu apakah 12 kurang dari 14 maka outpunya adalah **true**. Begitu seterusnya, **PAHAM YA??**

Jika yang dibandingkan adalah sesama string, PHP akan membandingkan urutan huruf berdasarkan abjad (atau lebih tepatnya berdasarkan urutan karakter ASCII). pada contoh kedua, huruf 'o' lebih belakangan urutannya dalam abjad dibandingkan huruf 'i' (anto vs anti) sehingga 'anto' < 'anti' akan menghasilkan **false**, **seharusnya anto>anti untuk jawaban true**

OPERATOR LOGIKA

Nama Operator	Operator	Contoh	Hasil
and	and atau &&	<code>\$a and \$b</code> atau <code>\$a && \$b</code>	true, jika \$a dan \$b sama-sama bernilai true
or	or atau	<code>\$a or \$b</code> atau <code>\$a \$b</code>	true, jika salah satu dari \$a atau \$b bernilai true
xor	xor	<code>\$a xor \$b</code>	true, jika salah satu dari \$a atau \$b bernilai true, tapi tidak keduanya.
not	!	<code>!\$a</code>	true, jika \$a bernilai false.

Untuk contoh buat file dengan nama **logika.php**

```
<?php
$hasil = (true and false);
var_dump($hasil); echo "<br/>";    // bool(false)

$hasil = (true or false);
var_dump($hasil); echo "<br/>";    // bool(true)

$hasil = (true xor false);
var_dump($hasil); echo "<br/>";    // bool(true)

$hasil = false;
var_dump(!$hasil); echo "<br/>";    // bool(true)

$hasil = (false or true && false);
var_dump($hasil); echo "<br/>";    // bool(false)

$hasil = ('000' or false);
var_dump($hasil);                // bool(true)
```


untuk memahami ini kalian ingat lagi materi logika matematika yang sudah pernah kalian pelajari ya.

Operator String

Dalam PHP, hanya terdapat 1 jenis operator string, yakni operasi penyambungan string (*string concatenation*). Operator ini menggunakan karakter titik (.). Jika operand bukan string, akan dikonversi menjadi string secara otomatis.

Berikut contoh operasi penyambungan string dalam PHP, buat file dengan nama **string.php**:

```
<?php
$hasil = "Belajar"."PHP";
echo $hasil; // BelajarPHP
echo "<br>";

$a = "Sstt!";
$b = " lagi";
$c = " serius";
$d = " belajar PHP";
$hasil = $a.$b.$c.$d;
echo $hasil; // Sstt! lagi serius belajar PHP
echo "<br>";

$hasil = 9 . " ekor anak beruang";
echo $hasil; // 9 ekor anak beruang
echo "<br>";

$hasil = true . " adalah data boolean";
echo $hasil; // 1 adalah data boolean
```

silakan coba, akses di browser dan pahami maksudnya,

Operator Assignment

Operator assignment adalah operator untuk memasukkan/menginput sebuah nilai ke dalam variabel. PHP memiliki 3 jenis operator assignment, yaitu :

- ☐ **Assignment nilai** (assignment by value).
- ☐ **Assignment referensi** (assignment by reference).
- ☐ **Assignment array**.

Assignment Nilai (assignment by value)

Assignment nilai, atau dalam bahasa inggris dikenal dengan istilah **assignment by value** adalah operator yang dipakai untuk menginput suatu nilai ke dalam variabel. Dari awal buku secara tidak langsung kita sudah sering menggunakannya, yakni tanda sama dengan ' = ' seperti contoh berikut:

```
<?php
$a = 99;
```

```
$b = "Belajar PHP";  
$c = 20;  
  
$d = $a - 15;  
$e = $a + $c;
```

Nah baris program diatas adalah contoh Assignment nilai, yang paling umum dan paling sering digunakan. Untuk assignment referensi dan array akan kita bahas di materi OOP

OK pertemuan 1 dan 2 sementara sampai disini, silakan dikembangkan dengan mencari contoh-contoh lain dari berbagai sumber yang ada

Tetap semangat jangan kasih kendor, jangan hanya berpatokan pada modul yang saya buat.!!!