

# Motivasi

"YOUR **FUTURE** IS CREATED BY WHAT  
YOU DO { TODAY }  
NOT { TOMORROW }"

# Aspersepsi

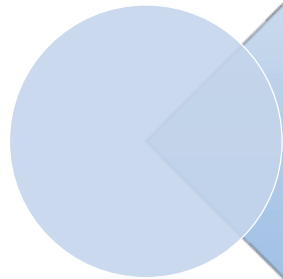


# Menggunakan tipe data dan control program

# 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 &  
Sytax 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 *dynamic 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.

# 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

# 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:

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 underscore ( <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 <b>tidak bisa membuat variabel dengan nama <code>\$1nilai</code></b>
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>&amp;</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: 7.0333333333333
float(7.0333333333333)
```

# 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 (') mau pun 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 pembandingan;
- 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	Symbol
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 operator 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 "<hr>";

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

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

1 && 0 = 0

---

1 || 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

# 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

The diagram shows the text 'kamu suka aku ? ya : tidak;'. An arrow points from the label 'Operator Ternary' to the question mark '?'. Another arrow points from the label 'jawaban benar' to the word 'ya'. A third arrow points from the label 'jawaban salah' to the word 'tidak'.

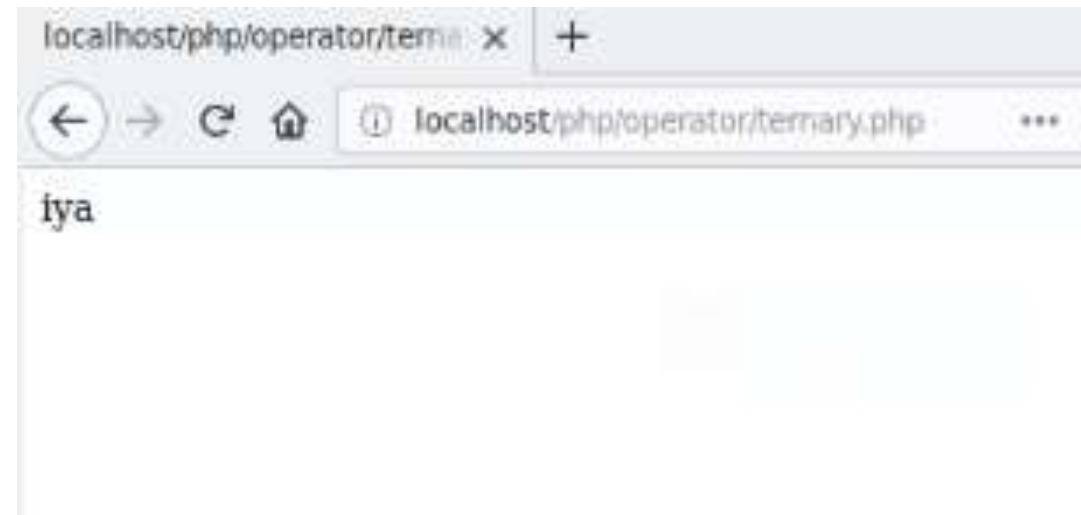
# Contoh

```
<?php

$suka = true;

// menggunakan operator ternary
$jawab = $suka ? "iya" : "tidak";

// menampilkan jawaban
echo $jawab;
```



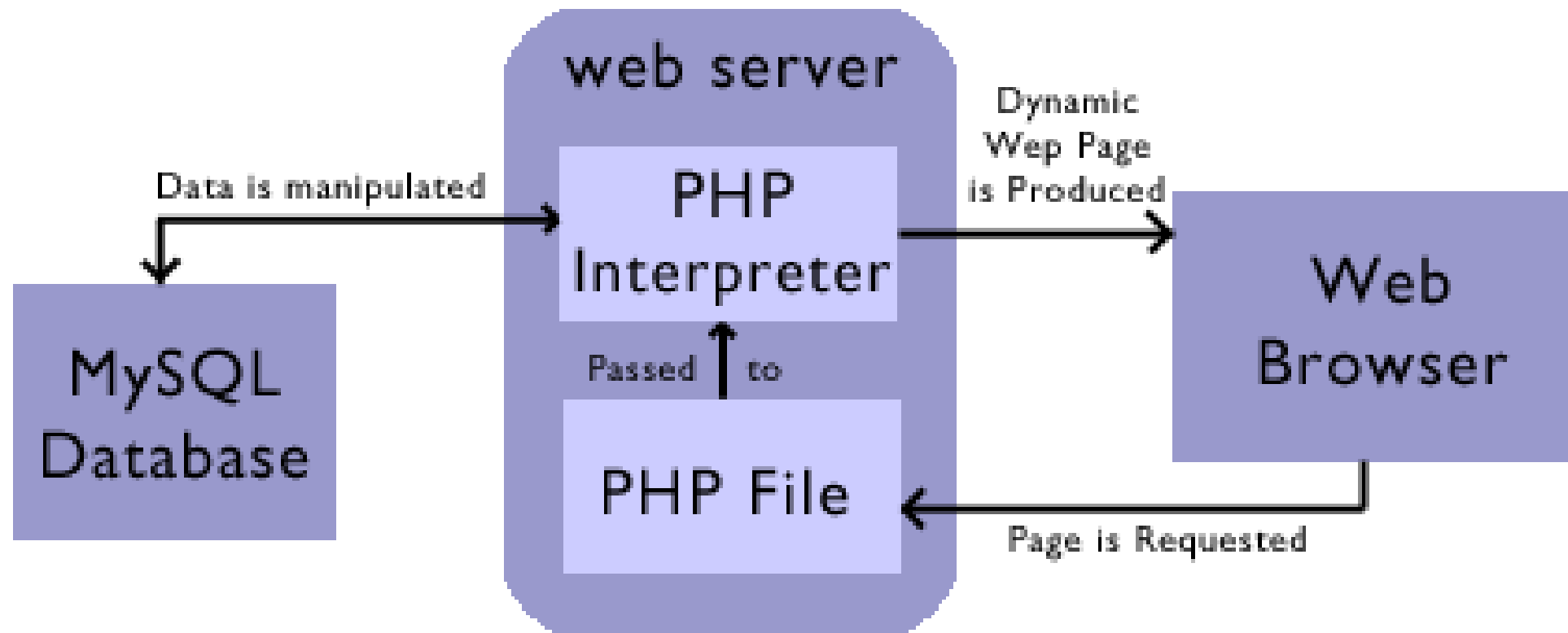
# Struktur PHP



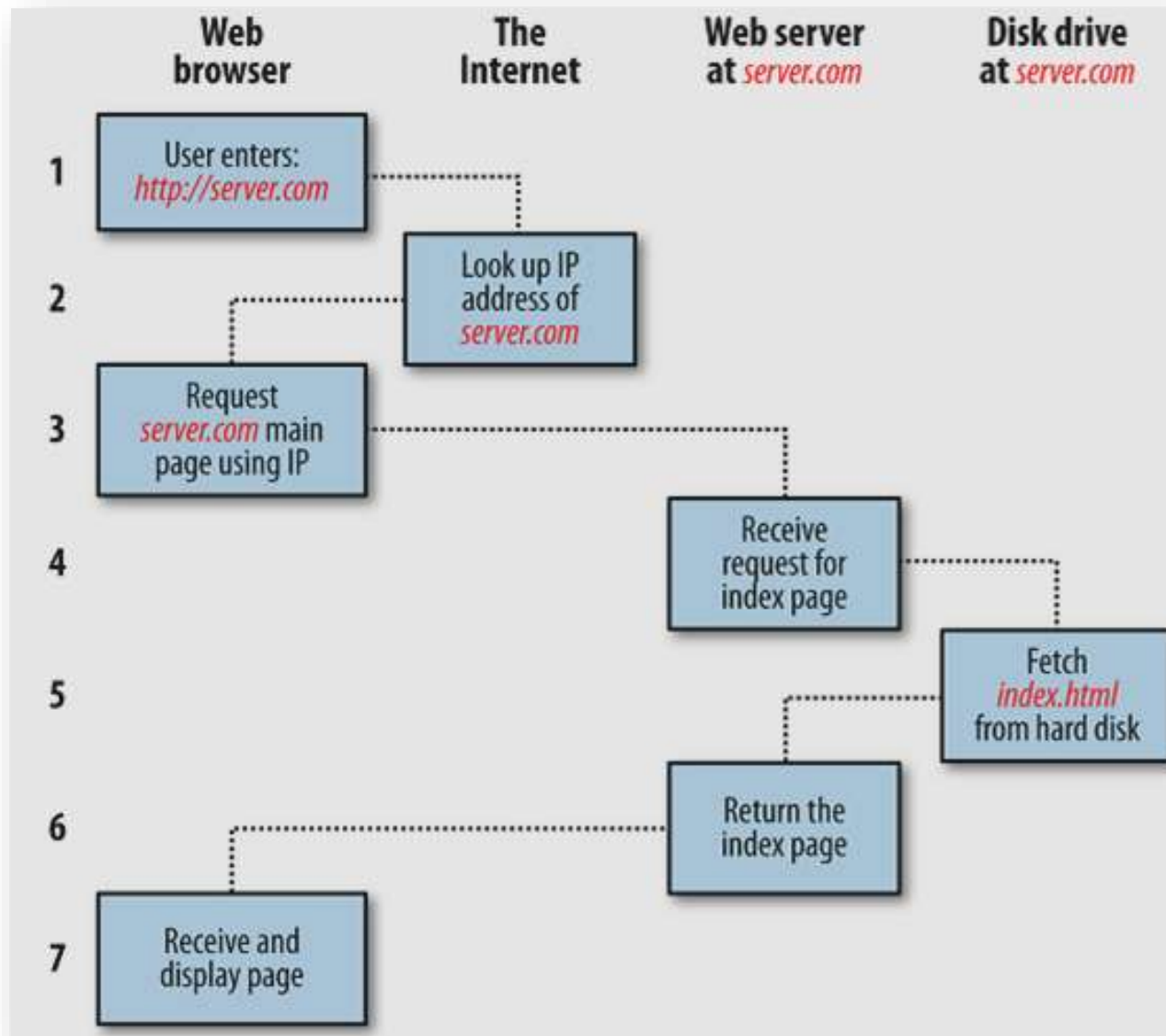
# PHP Pemrosesan data berbasis server side

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).

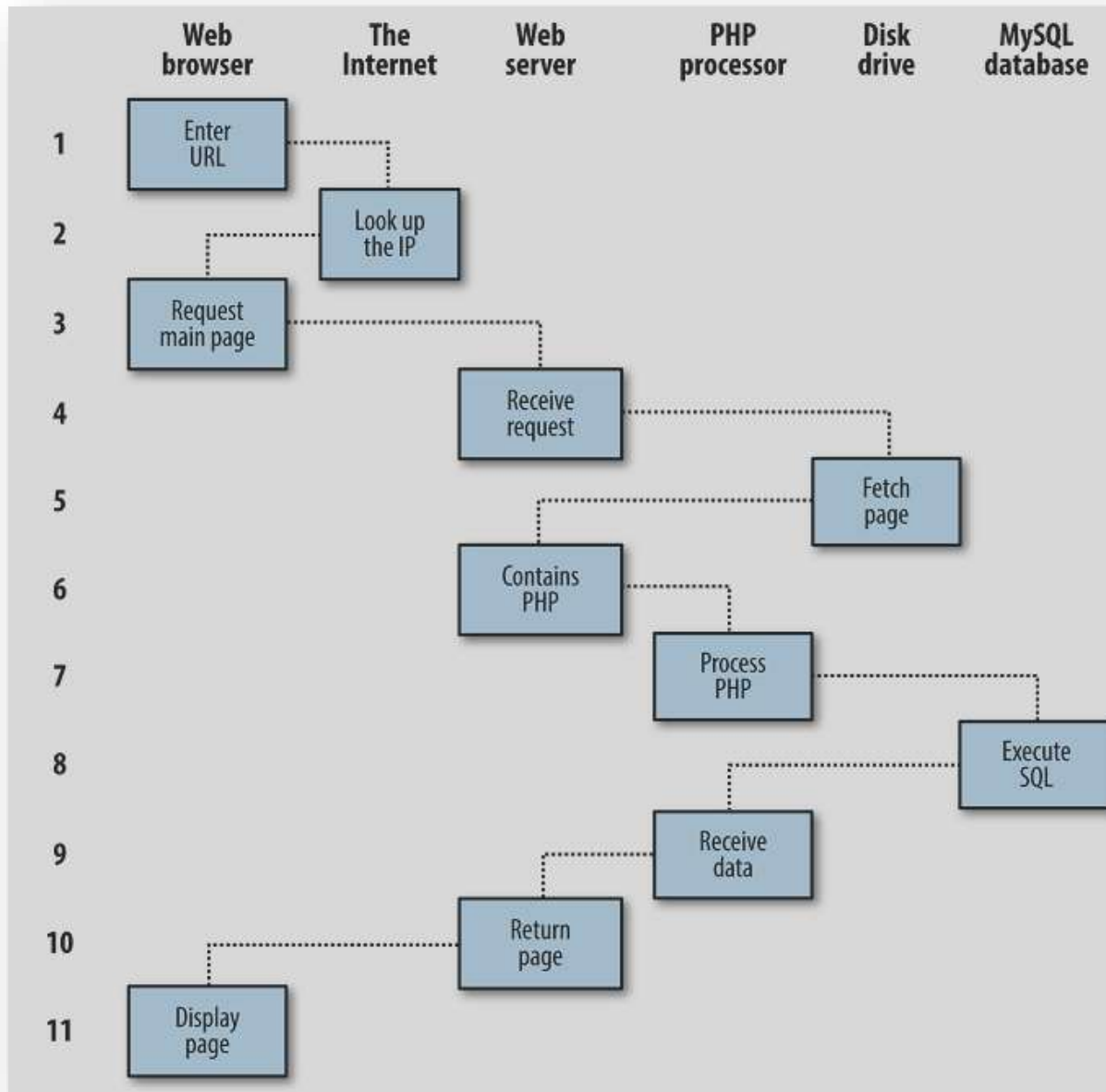
# Skema Hubungan Web Browser-Web Server-PHP



# Request/Response Procedure

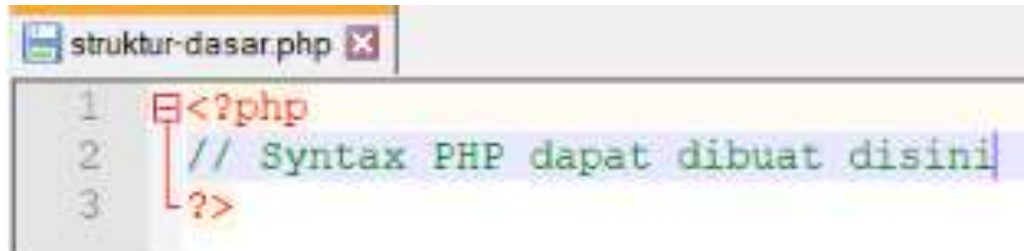


# Request/Response Procedure



# Struktur Dasar penulisan PHP

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

A screenshot of a code editor window titled 'struktur-dasar.php'. The editor shows three lines of code: line 1 contains '<?php', line 2 contains '// Syntax PHP dapat dibuat disini', and line 3 contains '?>'. The code is color-coded: '<?php' is red, the comment is green, and '?>' is red. A red bracket on the left side of the code block indicates the structure from line 1 to line 3.

```
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 dalah 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/>



Kantor:

Balai Pelatihan dan Pengembangan  
Teknologi Informasi dan Komunikasi  
Kementerian Kominfo

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

Email: [bpptik@kominfo.go.id](mailto:bpptik@kominfo.go.id)

Twitter: @bpptik

Facebook: @bpptik

Instagram: @bpptik

Google Plus: +bpptikkemkominfo

# Terima Kasih

## BPPTIK