
SERVER SIDE PROGRAMMING (PART I)

WEB SERVER, PHP, SINTAKS DASAR

MATERI KE-7



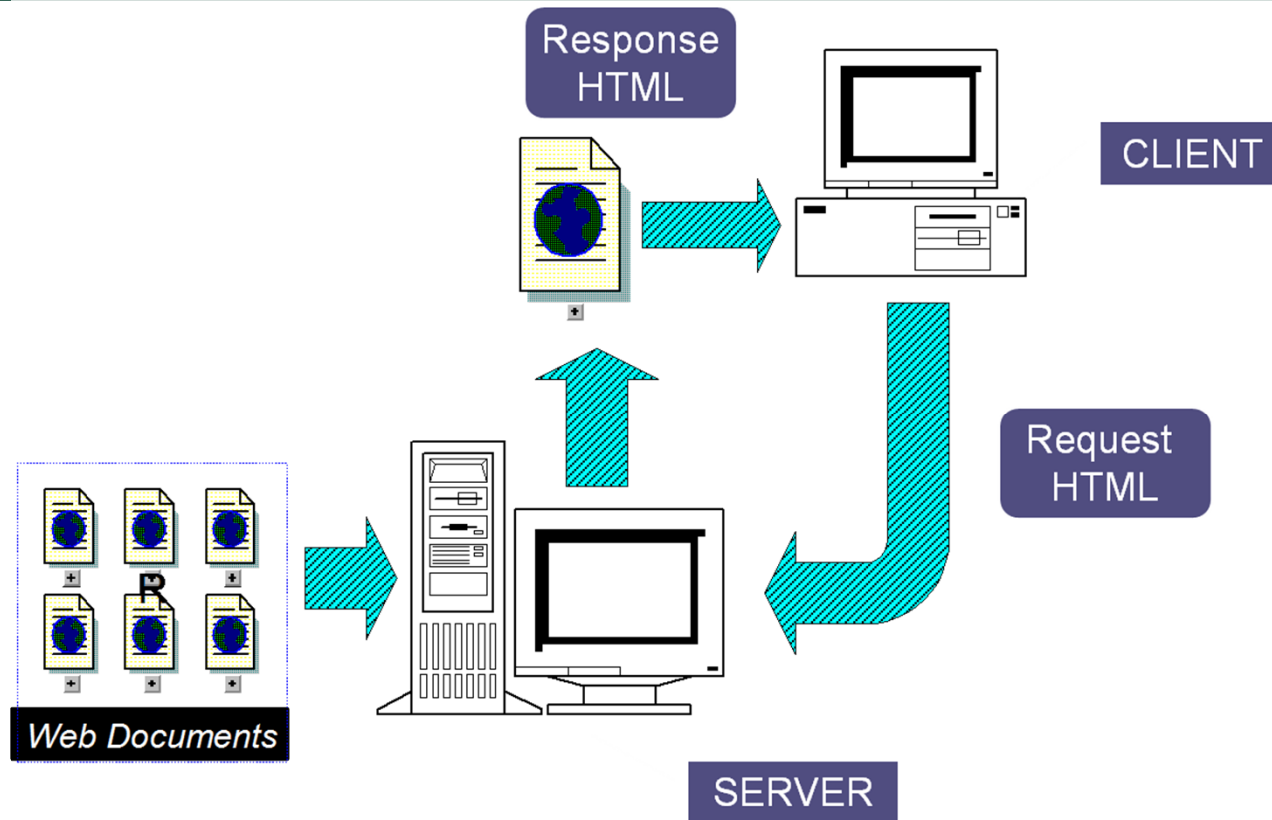
SERVER SIDE SCRIPTING

- ASP (Active Server Page) dan ASP.NET
- ColdFusion (<https://www.adobe.com/products/coldfusion-family.html>)
- JSP (Java Server Pages)
(<https://www.oracle.com/technetwork/java/javaee/jsp/index.html>)
- Perl (<http://www.perl.org>)
- Python (<http://www.python.org>)
- PHP (<http://www.php.net>)

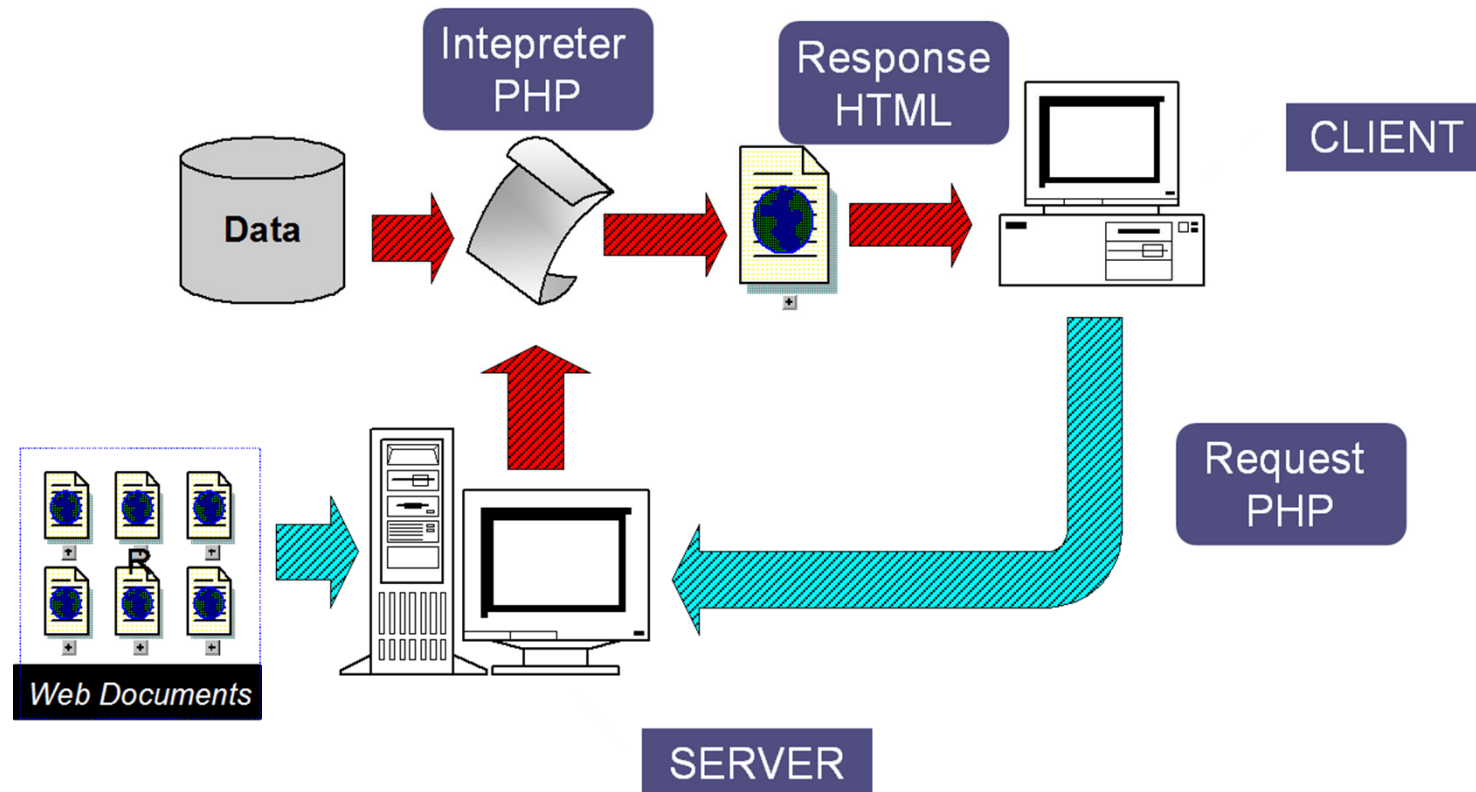


WEB STATIS VS WEB DINAMIS ?

ARSITEKTUR WEB STATIS



ARSITEKTUR WEB DINAMIS (PHP)





APA ITU PHP?



TENTANG PHP

- PHP singkatan *recursive* dari "PHP: Hypertext Preprocessor"
- Pertama kali dibuat oleh **Rasmus Lerdorf** pada tahun 1994.
- PHP merupakan bahasa *scripting* yang bersifat *open source*
- *Script* PHP dijalankan di server
- PHP banyak digunakan dalam pengembangan aplikasi berbasis web



APA ITU FILE PHP?

- File PHP mempunyai ekstensi ".php"
- File PHP bisa memuat text, HTML, CSS, JavaScript dan kode PHP
- Kode PHP dieksekusi di server dan hasilnya dikembalikan ke browser sebagai *plain* HTML

APA SAJA YANG BISA DILAKUKAN PHP?

- PHP bisa menghasilkan konten dinamis *
- PHP bisa membuat, membaca, menulis, menghapus, dan menutup file di web server
- PHP bisa mengumpulkan data yang dikirim dari form
- PHP bisa mengirim dan menerima *cookies*
- PHP bisa menambah, menghapus atau merubah data di database
- PHP bisa digunakan untuk mengontrol *user-access*
- PHP bisa meng-enkripsi data

* PHP tidak hanya bisa menghasilkan konten HTML tetapi juga gambar, file PDF, bahkan Flash, XML, XHTML dan lainnya.



BASIC PHP – SINTAKS DASAR



SINTAKS PHP

- Script PHP dapat diletakkan dimanapun dalam dokumen
- Harus diapit oleh `<?php` dan `?>`

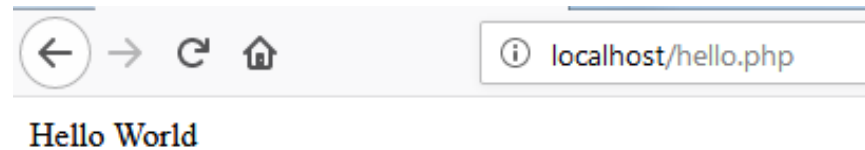
```
<?php  
// script PHP ditulis di sini  
?>
```

- Setiap *statement* PHP diakhiri dengan semicolon (;)

CONTOH : HELLO WORLD

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

```
<html>  
<body>  
  
<?php  
echo "Hello World";  
?>  
  
</body>  
</html>
```



VARIABEL

- Digunakan untuk menyimpan sebuah value, data atau informasi
- Nama variabel diawali dengan tanda \$
- Panjang tidak terbatas
- Setelah tanda \$ diawali oleh huruf atau *under-score* (_).
Karakter berikutnya bisa terdiri dari huruf, angka, dan karakter tertentu yang diperbolehkan (karakter ASCII dari 127 – 255).
- Bersifat **case-sensitive**.
- Tidak perlu dideklarasikan.
- Tidak boleh mengandung spasi.

CONTOH PEMBUATAN VARIABEL

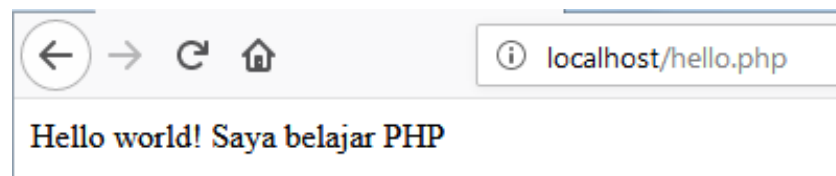
```
<?php  
$txt = "Hello world!";  
$x = 5;  
$y = 10.5;  
?>
```

- PHP tidak mempunyai *command* untuk mendeklarasikan variable
- Pada saat eksekusi variabel akan mempunyai tipe data sesuai dengan data yang di-assign

OUTPUT VARIABEL

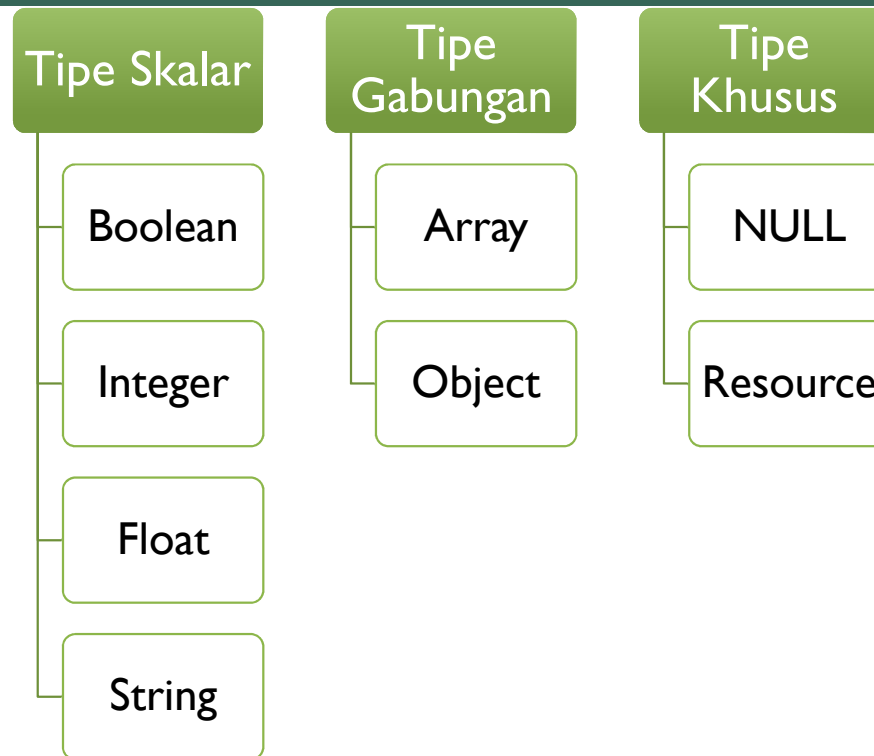
```
<?php  
$txt = "Hello world!";  
echo "$txt Saya belajar PHP";  
?>
```

```
<?php  
$txt = "Hello world!";  
echo $txt . " Saya belajar PHP";  
?>
```



Kedua code di atas akan menghasilkan output yang sama

TIPE DATA



Fungsi `var_dump` digunakan untuk mengembalikan tipe data dan nilai variabel

Tipe Skalar

Boolean

```
<?php  
$x = true;  
$y = false;  
?>
```

Integer

```
<?php  
$x = 5985;  
?>
```

Float

```
<?php  
$x = 10.365;  
?>
```

String

```
<?php  
$x = "Hello world!";  
?>
```

Tipe Gabungan

Array

```
<?php
$cars = array("Volvo", "BMW", "Toyota");
var_dump($cars);
?>
```

Object

```
<?php
class Mobil{
    function Mobil(){
        $this->model = "Toyota";
    }
}
// membuat object baru
$Yaris = new Mobil();
// mengakses object property
echo $Yaris->model;
?>
```

Tipe Khusus

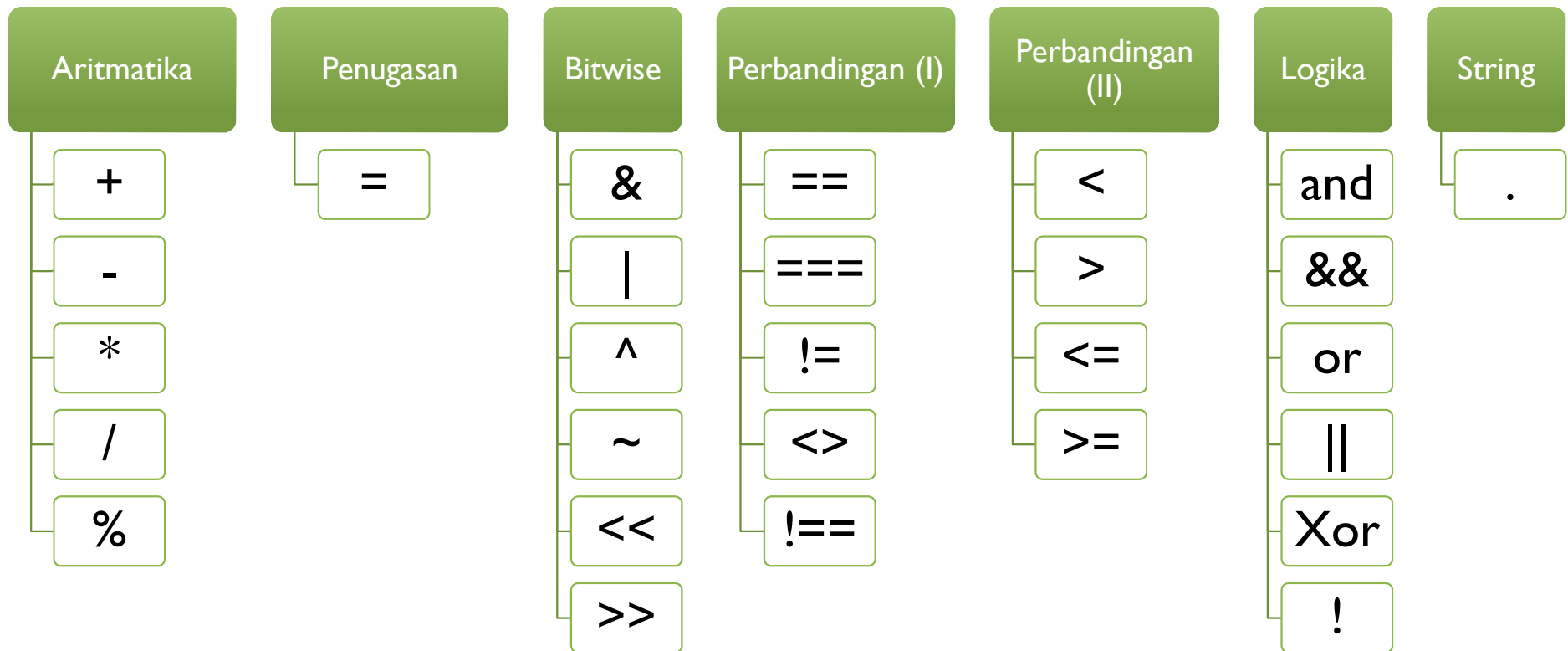
NULL

```
<?php
$x = "Hello world!";
$x = null;      // mengosongkan variabel
var_dump($x);
?>
```

Resource

- Resource sebenarnya bukan merupakan tipe data aktual.
- Biasanya digunakan untuk referensi ke fungsi atau resource eksternal
- Contohnya adalah untuk pemanggilan database
- Dibahas pada kuliah web lanjutan

OPERATOR



VARIABLE SCOPE

- Di PHP variabel dapat dideklarasikan di mana saja
- Scope dari sebuah variabel adalah bagian di mana sebuah variabel dapat digunakan
- Scope:
 - local
 - global
 - static

VARIABEL LOCAL VS GLOBAL

```
<?php
function angka() {
    $x = 5; // local scope
    echo "<p>nilai x di dalam fungsi: $x</p>";
}
angka();
echo "<p>nilai x di luar fungsi: $x</p>"; // error
?>
```

Variabel yang dideklarasikan di **dalam** fungsi mempunyai scope LOCAL dan hanya bisa diakses di dalam fungsi

```
<?php
$x = 5; // global scope
function angka() {
    echo "<p> nilai x di dalam fungsi: $x</p>"; // error
}
angka();
echo "<p> nilai x di luar fungsi : $x</p>";
?>
```

Variabel yang dideklarasikan di **luar** fungsi mempunyai scope GLOBAL dan hanya bisa diakses di luar fungsi

KEYWORD `global`

- Cara untuk mengakses variabel dengan scope global dari dalam fungsi menggunakan `global`

```
<?php
$x = 5;
$y = 10;
function angka() {
    global $x, $y;
    $y = $x + $y;
}
angka();
echo $y;
?>
```

KEYWORD `static`

- Setelah fungsi selesai dieksekusi, semua variabel biasanya dihapus. Supaya variabel local tidak dihapus setelah eksekusi dapat digunakan keyword `static`

```
<?php
function angka() {
    static $x = 0;
    echo $x . "<br>";
    $x++;
}
angka();
angka();
angka();
?>
```


KOMENTAR (REMARK)

Cara menulis komentar

- /*

komentar

di sini

*/

- // komentar di sini

- # komentar di sini



BERSAMBUNG...

CLIENT SIDE PROGRAMMING (PART 2)

