

# Modul Web Lanjut

---

*Arif Dwi Laksito, S.Kom*

## Table of Contents

1	Web Lanjut.....	3
2	PHP .....	3
2.1	PHP Introduction .....	3
2.2	PHP syntax .....	4
2.3	PHP Operator .....	4
2.3.1	PHP Arithmetic Operator.....	4
2.3.2	PHP Assignment Operator .....	5
2.3.3	PHP Comparison dan Logical Operator .....	6
2.4	Form PHP .....	7
2.5	PHP Quote.....	8
2.6	Conditional Statement .....	8
2.6.1	If Statement .....	9
2.6.2	If...else Statement .....	9
2.6.3	If...elseif....else Statement .....	9
2.6.4	Switch Statement .....	9
2.7	Perulangan .....	10
2.7.1	Perulangan while.....	10
2.7.2	Perulangan do..while.....	10
2.7.3	Perulangan for.....	10
2.7.4	Perulangan foreach .....	10
2.8	PHP Array .....	11
2.9	PHP Function .....	13
2.10	PHP Socket .....	14
2.11	Upload File .....	16
2.12	Session dan Cookies .....	17
2.12.1	Cookies.....	17

2.12.2	Session .....	18
2.13	OOP .....	19
2.14	Akses Database .....	20
2.14.1	PHP – Mysql .....	20
2.14.2	PHP Adodb .....	21
2.15	PHP Framework.....	22
2.15.1	CodeIgniter Framework.....	23
2.16	PHP Security .....	24
3	Web Service .....	25

## 1 Web Lanjut

Untuk perkuliahan web lanjut ini diharapkan nanti mahasiswa/i dapat membuat suatu aplikasi web dinamis. Ya.. dinamis, agak berbeda dengan mata kuliah yang saya ajarkan sebelumnya yaitu Pemrograman Web, untuk web dinamis disini kita akan menggunakan bahasa pemrograman web yang berjalan di sisi Server, yaitu PHP. Untuk dapat menjalankan PHP, maka kita juga harus melakukan instalasi Web Server (kita gunakan web server Apache, bisa di download disini).

Jika ingin lebih mudah saat instalasi, bisa menggunakan XAMPP, yaitu aplikasi installer yang sudah menyediakan Web Server(Apache) + (Server Side Language Programming)PHP + DBMS(Mysql) + FTP Server(Filezilla). Untuk lebih jelasnya bisa dilihat disini. XAMPP bisa digunakan di Windows, Linux atau MacOS. Jadi klo mau download, silahkan pilih OS yang sesuai

Aplikasi web biasanya identik dengan Internet, sebenarnya banyak juga aplikasi web yang berjalan di jaringan lokal ataupun standalone PC, seperti aplikasi Sistem Informasi Kepegawaian berbasis web yang berjalan di suatu instansi atau Aplikasi SMS gateway (Kalkun, Open Source Aplikasi) yang berjalan di Standalone PC. Perbedaan aplikasi web di internet, jaringan lokal atau standalone PC adalah url yang di inputkan di browser, seperti:

Internet, akses menggunakan nama domain: [www.detik.com](http://www.detik.com), [www.amikom.ac.id](http://www.amikom.ac.id), [www.google.com](http://www.google.com), dll

Jaringan lokal, akses menggunakan IP address lokal: 172.16.15.3, 192.168.1.10, dll

Standalone PC, akses menggunakan IP local: localhost atau 127.0.0.1

Sebenarnya aplikasi web tidak hanya web company profile, web social networking ataupun web blog. Aplikasi web bisa juga seperti Web GPS tracking, Web SMS Gateway, Web Aplikasi Keuangan, dll.

## 2 PHP

### 2.1 PHP Introduction

PHP is a server-side scripting Language, atau bahasa pemrograman yang berada di sisi server. Untuk mempelajari PHP, anda sebelumnya harus mengetahui HTML, CSS dan Javascript.

PHP kepanjangan dari PHP: Hypertext Preprocessor, PHP adalah bahasa pemrograman yang berada di sisi server seperti ASP atau JSP. PHP mendukung banyak database seperti Mysql, Oracle, SQL Server, dll. File PHP dapat terdiri dari text, HTML atau script PHP itu sendiri, tetapi hasil request akan dikembalikan ke browser dalam bentuk plain HTML. Ekstensi PHP bisa beberapa macam seperti .php, .php4, .phtml tergantung konfigurasi dari Web Server.

Sebelum memulai coding PHP, pastikan anda mempunyai file PHP manual untuk panduan, dapat di download disini.

## 2.2 PHP syntax

Menuliskan string PHP atau membuat comment

```
<?php
echo "Hello World!";
print "Hello World again";

// ini adalah comment untuk satu baris
/* atau ini comment untuk beberapa baris..
lanjutan comment sebelumnya */
?>
```

Membuat variabel di PHP, fungsi variabel di PHP sama dengan pemrograman yang lain yaitu untuk menyimpan data String, Integer, atau array. Nama variabel harus diawali dengan string atau underscore “\_”. Penulisan variabel juga tidak boleh menggunakan spasi, jika memerlukan 2 kata bisa digunakan underscore “\_”.

```
<?php
$var = "Hello Bob.."; // penulisan variabel benar
$_num = 453; // penulisan variabel benar
$4int = 40.32; // penulisan variabel salah, diawali number
$my_var = "its me"; // penulisan variabel benar
?>
```

Penggabungan String dan variabel

```
<?php
$var = "Bob";
$lst = "Jonson";
echo "Hello ".$var; // menggabungkan string dan variabel menggunakan "."
echo $var.$lst; // akan menampilkan text "BobJonson"
?>
```

## 2.3 PHP Operator

### 2.3.1 PHP Arithmetic Operator

Pada artikel kali ini kita akan pelajari PHP Operator, yang pertama adalah Arithmetic Operator, sama dengan pemrograman yang lain, beberapa macam arithmetic operator seperti

- ✓ Penambahan +
- ✓ Pengurangan -
- ✓ perkalian \*
- ✓ Pembagian /
- ✓ Sisa Hasil bagi %

- ✓ Increment(kenaikan) ++
- ✓ Decrement(penurunan) –

Langsung saja, contoh penggunaannya di syntax PHP adalah sbb:

```
<?php
$x = 10;
echo $x + 5; echo '<br />';
echo $x - 3; echo '<br />';
echo $x * 3; echo '<br />';
echo $x / 2; echo '<br />';
echo $x % 3; echo '<br />';

$x++; // nilai x dinaikkan 1
echo $x; echo '<br />'; // menampilkan nilai x setelah dinaikkan

$x--; // nilai x diturunkan 1
echo $x; echo '<br />'; // menampilkan nilai x setelah diturunkan
?>
```

Tugas selanjutnya adalah melakukan analisis di code diatas.. Kenapa hasil output paling bawah bernilai 10 tidak ??? Bagaimana caranya supaya output nya menjadi 9??

### 2.3.2 PHP Assignment Operator

Pengertiannya, operator ini digunakan untuk menetapkan variabel sama dengan nilai atau set variabel ke nilai variabel lain, contoh beberapa Assignment Operator di PHP

- ✓ Plus Equal +=
- ✓ Minus Equal -=
- ✓ Multiply Equal \*=
- ✓ Divide Equal /=
- ✓ Modulus Equal %=
- ✓ Concatenate Equal .=

```
<?php
$x = 10;
echo $x += 2; echo "<br />"; // hasilnya 12
echo $x -= 5; echo "<br />"; // hasilnya 7
echo $x *= 2; echo "<br />"; // hasilnya 14
?>
```

Kenapa hasil dari baris ke-2 dan ke-3 adalah 7 dan 14, silahkan dipahami konsep assignment operator tersebut.

### 2.3.3 PHP Comparison dan Logical Operator

Kedua Operator ini biasanya digubakan untuk percabangan yaitu untuk membandingkan suatu nilai satu dengan yang lainnya, beberapa macam comparison operator yaitu :

- ✓ is equal to ==
- ✓ is not equal (biasa digunakan dlm String/Text) !=
- ✓ is not equal (biasa digunakan dml int/numeric) <>
- ✓ is greater than >
- ✓ is less than <
- ✓ is greater than or equal to >=
- ✓ is less than or equal to <=

Contoh script-nya sebagai berikut

```
<?php
$x='jon';
$y=10;
$z=5;

if($x == $y){
    echo 'match';
}else{
    echo 'no-match';
}

echo '<br />';
echo (($y/5) > 8)?'match':'no-match';
// penulisan percabangan sederhana bisa menggunakan
// style seperti diatas (java)

echo '<br />';
$hasil = ($x != 'jon')?'match':'no-match';
echo $hasil;
// percabangan diatas membandingkan text dgn text
// dan hasilnya disimpan di variabel sebelum di tampilkan
?>
```

Logical Operator digunakan untuk melakukan pengecekan lebih dari satu kondisi, Penggunaanya sama dengan pemrograman yang lain, dan berikut contoh Logical Operator di PHP :

- ✓ And &&
- ✓ Or ||

✓ Not !

Contoh script penggunaan logical operator sebagai berikut:

```
<?php
$a = 5;
$b = 17;

if($a>10 && $a<10) echo 'match';
else echo 'no-match';
// percabangan yg menghasilkan 1 baris perintah
// bisa ditulis seperti diatas, tanpa tanda { dan }

echo '<br />';
echo ($b>15 || $b< 15)?'match':'no-match';

echo '<br />';
if(!($a>10)) echo 'match';
else 'no-match';

?>
```

- ✓ Logika AND (&&), Menghasilkan nilai true jika kedua operand menghasilkan nilai true.
- ✓ Logika OR (||), Menghasilkan nilai true jika salah satu operand menghasilkan nilai true.
- ✓ Logika NOT (!), Menghasilkan nilai true jika operand menghasilkan nilai false dan menghasilkan nilai false jika operand menghasilkan nilai true.

## 2.4 Form PHP

Aplikasi web pasti banyak sekali berhubungan dengan Form, karena form digunakan sebagai input dari user untuk dapat berinteraksi dengan aplikasi web. Metode dalam membaca data dari inputan form ada 2 macam, yaitu POST dan GET. Perbedaan utama keduanya adalah, data yang dikirim menggunakan GET akan terlihat di address bar, sedangkan post tidak.

```
<form action="" method="post">
Nama Lengkap: <input type="text" name="fullname" /><br />

Hobby: <br />
<input type="checkbox" name="hobby[]" value="sport" />Sport<br />
<input type="checkbox" name="hobby[]" value="read" />Reading<br />
<input type="checkbox" name="hobby[]" value="hangout" />Hang Out<br />
<input type="checkbox" name="hobby[]" value="shopping" />Shopping<br />

Desc: <br />
```

```
<textarea name="desc"></textarea><br />
<input type="submit" name="kirim" value="Kirim" />

</form>
```

Script diatas menggunakan metode POST, maka saat membaca isi data yang dikirim dari form menggunakan variabel `$_POST` sedangkan metode GET menggunakan variabel `$_GET`

```
<?php
if(isset($_POST['kirim'])){ // validasi saat submit tombol kirim
    print_r($_POST); // untuk melihat array data yg dikirim
    echo "<br />";
    echo $_POST['fullname']."<br />";
    print_r($_POST['hobby']); echo "<br />";
    echo $_POST['desc']."<br />";
}
?>
```

## 2.5 PHP Quote

Single quote(') dan double quote(") sering di gunakan dalam pemrograman PHP, terutama dalam menuliskan String. Terdapat perbedaan kedua penggunaan quote tersebut, single quote tidak dapat membaca variabel di dalam String, jadi harus menggunakan syntax (.) untuk menggabungkannya.

```
<?php
$var = 'Budi';
echo 'Ini ibu $var'; // hasil output adalah: Ini ibu $var
echo '<br />';
echo 'Ini ibu '.$var // hasil output adalah: Ini ibu Budi
?>
```

Berbeda dengan menggunakan double quote, bisa digunakan untuk menampilkan variabel di dalam String tanpa harus menggunakan syntax penghubung.

```
<?php
$var = 'Budi';
echo "Ini ibu $var"; // hasil output adalah: Ini ibu Budi
?>
```

## 2.6 Conditional Statement

Bentuk percabangan sering digunakan dalam pemrograman PHP, ada beberapa bentuk percabangan di PHP yaitu:

- ✓ if statement -digunakan untuk kondisi jika true saja
- ✓ if...else statement - digunakan untuk kondisi jika true atau false saja



- ✓ if...elseif....else statement - digunakan untuk kondisi true lebih dari satu
- ✓ switch statement - digunakan untuk kondisi true atau false, atau true lebih dari satu

### 2.6.1 If Statement

Bentuk code :

```
if (condition) {  
    //dijalankan jika kondisi true  
}
```

### 2.6.2 If...else Statement

Bentuk code :

```
if (condition){  
    //dijalankan jika kondisi true  
}else {  
    //dijalankan jika kondisi false  
}
```

### 2.6.3 If...elseif....else Statement

Bentuk code :

```
if (condition){  
    //dijalankan jika kondisi true  
}elseif (condition) {  
    //dijalankan jika kondisi true  
}else {  
    //dijalankan jika kondisi false  
}
```

### 2.6.4 Switch Statement

Bentuk code :

```
switch (n)  
{  
case label1:  
    //dijalankan jika n=label1;  
    break;  
case label2:  
    //dijalankan jika n=label2;  
    break;  
default:  
    //dijalankan jika n tidak sama label1 dan label2;  
}
```

## 2.7 Perulangan

Biasanya dalam menuliskan blok kode kita sering menuliskan kode yang sama diulang-ulang, perulangan digunakan untuk mengatasi masalah tersebut.

Di PHP ada beberapa macam perulangan yang bisa digunakan, yaitu:

- ✓ while -perulangan blok kode dengan kondisi true
- ✓ do...while - melakukan perulangan sekali, kemudian perulangan selanjutnya jika kondisi true
- ✓ for - perulangan blok kode dengan awal dan akhir nomer tertentu
- ✓ foreach - perulangan dalam tiap elemen array

### 2.7.1 Perulangan while

```
<?php
$i=1;
while($i<=5)
{
    echo "The number is " . $i . "<br />";
    $i++;
}
?>
```

### 2.7.2 Perulangan do..while

```
<?php
$i=1;
do
{
    $i++;
    echo "The number is " . $i . "<br />";
}
while ($i<=5);
?>
```

### 2.7.3 Perulangan for

```
<?php
for ($i=1; $i<=5; $i++)
{
    echo "The number is " . $i . "<br />";
}
?>
```

### 2.7.4 Perulangan foreach

```
<?php
```

```

$x=array("one","two","three");
foreach ($x as $value)
{
    echo $value . "<br />";
}

// foreach lebih simple digunakan dalam melakukan perulangan array
// bandingkan dengan perulangan for berikut..
$x=array("one","two","three");
for($i=0; $i<count($x); $i++)
{
    echo $x[$i]. "<br />";
}

?>

```

## 2.8 PHP Array

Dalam banyak aplikasi Web, seringkali kita membutuhkan sebuah variabel yang mampu menyimpan beberapa nilai sekaligus secara terorganisir, sehingga lain saat kita dapat memperoleh kembali nilai-nilai tersebut dengan mudah dan dalam bentuk yang teratur dan terorganisir pula. Tidak, bukan database yang saya maksud, karena saya sedang bicara variabel, dan database bukan variabel. Mengapa ini sangat penting? Karena dalam banyak kasus kita hanya membutuhkan penyimpanan nilai sementara yang terorganisir dengan baik, dan database sangat mahal untuk melakukan hal tersebut. Bayangkan tentang koneksi, query dan seterusnya yang akan memperlambat proses apabila kita menggunakan database.

Di sinilah kita membutuhkan array, sebuah tipe variabel yang mampu menyimpan beberapa nilai sekaligus bahkan dalam tipe berbeda yang terorganisir dengan baik. Sehingga dalam bahasa pemrograman manapun—termasuk PHP tentunya—Anda akan senantiasa menemukan array sebagai salah satu tipe variabelnya. Karena—maaf saya menggunakan bahasa Inggris—array is powerful. (sumber: [www.master.web.id](http://www.master.web.id))

Inisialisasi Array, dalam melakukan inisialisasi array di PHP kita bisa menggunakan fungsi `array()` atau langsung mengisikan ke dalam variabel array dengan index tertentu

```

<?php
// inisialisasi dengan fungsi array(), index dimulai dari 0
$var = array('pedrosa','stoner','lorenzo','rossi','simoncelli');
echo $var[3]. ' - ' . $var[0];

// inisialisai array langsung mengisikan data ke index tertentu
$tim[1]='van persie';
$tim[0]='gervinho';
$tim[3]='wilshere';

```

```
$tim[4]='ramsey';
echo $tim[1].' - '.$var[2];
// data $tim[2] tidak tampil, karena index 2 blm kita inisialisasi
?>
```

Untuk keperluan develop web, kita bisa menampilkan seluruh index dan data array menggunakan fungsi `print_r`

```
<?php
print_r($var);
?>
```

Beberapa contoh array diatas menggunakan Numeric Array, yaitu index array tersebut menggunakan angka, sedangkan kita juga bisa menggunakan index array yang saling terkait atau menggunakan text yang disebut juga Associative Array

```
<?php
$num['Pedrosa'] = "26";
$num['Rossi'] = "46";
$num['Simoncelli'] = "58";

echo "Pedrosa number is " . $num['Pedrosa'];
?>
```

Dalam array multidimensi, setiap elemen dalam array utama juga dapat sebuah array. Dan setiap elemen dalam sub-array dapat juga sebuah array, dan seterusnya.

Berikut contoh inisialisasi array multidimensi

```
<?php

$race = array(
    'Honda' => array('Pedrosa','Stoner','Dovizioso'),
    'Yamaha' => array('Lorenzo','Spies'),
    'Ducati' => array('Rossi','Hayden')
);

print_r($race);
echo 'The Best Honda rider is '.$race['Honda'][0];

?>
```

## 2.9 PHP Function

PHP sudah mempunyai fungsi yang built-in di dalamnya, meski pun sudah ada fungsi bawaan kita dapat membuat fungsi sendiri. Fungsi yang kita buat harus berbeda nama dengan fungsi yang sudah ada, meskipun kegunaannya sama.

Berdasarkan parameternya, fungsi di PHP bisa kita bagi menjadi 2 yaitu: fungsi dengan parameter dan fungsi tanpa parameter. Untuk fungsi yang membutuhkan parameter, kita bisa menggunakan 1 parameter atau lebih dan tidak terbatas sesuai kebutuhan penggunaannya.

Sedangkan Fungsi berdasar nilai kembalinya, dibedakan menjadi 2 juga yaitu: fungsi dengan nilai kembalian(dapat berupa string, int, array, dsb) dan fungsi tanpa nilai kembalian.

Berikut contoh fungsi dengan parameter dan tanpa parameter

```
<?php
function writeName()
{
    echo "James Bond";
}

echo "My name is ";
writeName();
?>
```

```
<?php
function writeName($fname)
{
    echo $fname . " Refsnes.<br />";
}

echo "My name is ".writeName("James Bond");
echo "<br />";
echo "My sister's name is ".writeName("Emma Watson");
echo "<br />";
echo "My brother's name is ".writeName("Bean");
?>
```

Contoh kedua fungsi writeName() diatas tidak mengembalikan nilai, hanya menuliskan string saja menggunakan echo. Berikut contoh fungsi yang mengembalikan nilai

```

<?php
function add($x,$y)
{
$total=$x+$y;
return $total;
}

echo "1 + 16 = " . add(1,16);
?>

```

## 2.10 PHP Socket

Dalam socket programming, pasti berhubungan dengan port dan service. Misal kita mengakses port 80, dan service yang berjalan dari port 80 tersebut adalah web server. Port 22 untuk service SSH, port 21 untuk service FTP dan sebagainya.

Socket merupakan literal Connector antara port dan Program, dan socket programming adalah suatu programming untuk mengakses, menggunakan dan menyediakan service dalam suatu komunikasi antara server dan client menggunakan port tertentu.

Ada 2 macam protokol komunikasi yang dapat digunakan untuk socket programming, yaitu:

1. Datagram Communication, atau biasa dikenal dengan istilah UDP(User Datagram Protocol). Datagram berupa connectionless protocol, yang berarti setiap saat kita dapat mengirim data ke server dan tidak akan dicek data tersebut sampai atau rusak.
2. Stream Communication, atau biasa dikenal dengan istilah TCP(Transfer Control Protocol). Tidak seperti UDP, TCP merupakan connection-oriented protocol yaitu saat terjadi komunikasi harus ada pairing(terbentuk koneksi antar server-client) setelah itu pengiriman data baru dapat dilakukan. Kerusakan atau kehilangan data dapat ditangani karena terdapat flag/tanda untuk pengecekan data.

Berikut contoh Script Socket Client, yang digunakan untuk mengakses domain amikom.ac.id menggunakan parameter GET

```

$fp = fsockopen ("amikom.ac.id", 80);
if ($fp) {
    fwrite($fp, "GET / HTTP/1.1\r\nHOST: amikom.ac.id\r\n\r\n");
    while (!feof($fp)) {
        print fread($fp,256);
    }
    fclose ($fp);
} else {
    print "Fatal error\n";
}

```

fungsi `fsockopen()` di atas akan membuka sebuah alamat server dengan port yang kita tentukan, kemudian setelah terjadi koneksi dengan server kita dapat meminta request dengan menggunakan protokol HTTP. Kita gunakan fungsi `fwrite()` untuk mengirim request tsb, pada contoh script diatas kita gunakan GET untuk meminta request ke server.

Berikut script Socket server menggunakan mode TCP, dengan fungsi membalik kata yang diinputkan dari user

```
<?php

$socket = socket_create_listen("9000");

if (!$socket) {
    print "Failed to create socket!\n";
    exit;
}

while (true) {
    $client = socket_accept($socket);
    $welcome = "\nWelcome to the Local machine.\n
    Type '!close' to close this connection, or type '!halt'
    to halt the server.\n";

    socket_write($client, $welcome);
    while (true) {
        $input = trim(socket_read ($client, 256));
        if ($input == '!close') {
            break;
        }
        if ($input == '!halt') {
            socket_close ($client);
            break;
        }

        $output = strrev($input) . "\n";
        socket_write($client, $output);
        print "In: $input, Out: $output\n";
    }
    socket_close ($client);
}
```

```
socket_close ($socket);

?>
```

Programming socket ini bisa digunakan untuk berkomunikasi antara low level programming dengan PHP, misal seperti device sensor yang dapat mengirim data ke server melalui Internet atau Device GPS yang mengirim data NMEA ke server melalui GPRS dan sebagainya.

## 2.11 Upload File

Suatu aplikasi web pasti banyak sekali berhubungan dengan upload file ke server, Cara upload suatu file ke server menggunakan PHP ada beberapa cara, antara lain menggunakan fungsi `move_uploaded_file()` atau menggunakan fungsi FTP.

Untuk membuat aplikasi upload, tambahkan atribut berikut `enctype="multipart/form-data"` pada tag form html. Dan input untuk menampung file dari media penyimpanan gunakan `type="file"`. Script html untuk form seperti berikut:

```
<form enctype="multipart/form-data" action="uploader.php" method="POST">
Choose a file to upload: <input name="uploadedfile" type="file" /><br />
<input type="submit" value="Upload File" />
</form>
```

Script PHP untuk menerima beberapa komponen yang dikirim melalui form tersebut bisa dilihat menggunakan fungsi `print_r()` berikut:

```
<?php

print_r($_FILES['uploadedfile']);

?>
```

Setelah data dari form tersebut dapat kita baca, selanjutnya tinggal kita proses untuk di upload menggunakan FTP atau fungsi `move_uploaded_file()`. Berikut contoh script menggunakan fungsi `move_uploaded_file()` yang disimpan dengan nama file `uploader.php` sesuai action dari form HTML diatas

```
<?php
$target_path = "uploads/";

$target_path = $target_path . basename( $_FILES['uploadedfile']['name']);

if(move_uploaded_file($_FILES['uploadedfile']['tmp_name'], $target_path)) {
    echo "The file ". basename( $_FILES['uploadedfile']['name']).
    " has been uploaded";
} else{
```



```

    echo "There was an error uploading the file, please try again!";
}
?>

```

## 2.12 Session dan Cookies

### 2.12.1 Cookies

Cookie biasanya digunakan untuk mengidentifikasi user, Setiap user melalui browser yang meminta request ke server akan membawa juga cookie dari browser tersebut. Melalui PHP kita dapat membuat cookie ataupun menerima data cookie.

Dalam mengatur cookie terdapat 3 hal penting yang harus diperhatikan, yaitu:

- ✓ membuat variabel cookie dan memberi nilai awal cookie,
- ✓ mengambil data dari cookie yang telah kita buat sebelumnya,
- ✓ menghapus variabel cookie

Konsep cookie dalam pemrograman web gambarannya seperti berikut, Saat user login melalui browser dan login tersebut berhasil, maka dari server akan mengirim data cookie yang akan disimpan di client. Saat cookie tersebut tersimpan, kita dapat membuka halaman2 web yang membutuhkan validasi cookie. Saat kita logout, maka data cookie tersebut akan di hapus dan kita tidak dapat lagi membuka halaman2 web yang tervalidasi oleh cookie. Cookie dapat dihapus juga menggunakan waktu expired, jadi jika user tidak logout dalam batas waktu tertentu cookie akan otomatis terhapus, pengaturan tersebut dapat kita lakukan melalui PHP.

Berikut script untuk membuat cookie dan memberi nilai awal cookie, Pastikan menuliskan script tersebut sebelum tag <html>

```

<?php
// cookie dengan variabel user, bernilai Jack Wilshere, batas waktu 1 jam
setcookie("user", "Jack Wilshere", time()+3600);

// jika ingin menghilangkan batas waktu, tdk perlu parameter terakhir
setcookie("user", "Jack Wilshere");
?>

```

Dan untuk mengambil nilai cookie, dapat menggunakan fungsi berikut.. dan tentu saja fungsi print\_r() tetap bisa digunakan untuk melihat data cookie apa saja yang tersimpan di client

```

<?php
// menampilkan data cookie
echo $_COOKIE["user"];

// cara untuk melihat semua data cookie
print_r($_COOKIE);

```

```
?>
Dan berikut script untuk menghapus data cookie
<?php
// set expire untuk session
setcookie("user", "", time()-3600);
?>
```

### 2.12.2 Session

Suatu variabel session PHP digunakan untuk menyimpan informasi tentang user, dan dapat di kenali di semua halaman web dalam satu aplikasi. Session di simpan di sisi server dan bersifat sementara, yaitu selama user belum keluar dari aplikasi atau session tersebut telah habis waktu gunanya (expired).

Session bekerja dengan menciptakan sebuah id unik (UID) untuk setiap user dan menyimpan di server berdasarkan UID tersebut.

Sebelum menyimpan variabel session di PHP, session harus di inisialisasi terlebih dahulu menggunakan fungsi `session_start()` dan fungsi tersebut harus dituliskan sebelum tag `<html>`

```
<?php session_start(); ?>

<html>
<body>

</body>
</html>
```

Fungsi tersebut akan mendaftarkan user ke server dan menetapkan id session (UID) untuk user.

Cara di PHP untuk menyimpan dan mengambil variabel session adalah dengan menggunakan variabel `$_SESSION`

```
<?php
session_start();
//Menyimpan data sesi
$_SESSION['user'] = 'jack';
?>
```

Sedangkan jika ingin menghapus variabel session di server dapat menggunakan fungsi `unset()` atau `session_destroy()`

```
<?php
unset ($_SESSION['user']);
session_destroy();
?>
```

Fungsi unset() akan menghapus variabel session yang telah ditentukan saja, sedangkan fungsi session\_destroy() akan menghapus semua session dan mengatur ulang session dari awal

## 2.13 OOP

Pada bahasa pemrograman PHP di kenal juga istilah Object Oriented Programming (OOP), disini akan di bahasa mulai dari class, function dan penggunaan OOP di PHP. PHP 5 memberikan performa yang lebih baik dan fitur baru dibandingkan versi sebelumnya. Class atau Object Modul PHP banyak tersedia di internet, seperti di [phpclasses](#), di sana kita bisa mendapatkan class-class siap pakai. Untuk dapat menggunakan class tersebut kita harus memahami konsep OOP.

Kita dapat membuat suatu class php di file tersendiri dan menggunakannya di beberapa file yang lain. Misalkan kita buat file class\_lib.php dengan kode sebagai berikut

```
<?php
    class person {
        var $name;

        function set_name($new_name) {
            $this->name = $new_name;
        }

        function get_name() {
            return $this->name;
        }
    }
?>
```

Dan kita dapat menggunakan class tersebut dengan cara sbb:

```
<?php include("class_lib.php"); ?>

<?php
    $jack = new person();
    $robin = new person;

    $jack->set_name("Jack Wilshere");
    $robin->set_name("Robin van Persie");

    echo "Jack's full name: " . $jack->get_name();
    echo "Robin's full name: " . $robin->get_name();
?>
```

## 2.14 Akses Database

### 2.14.1 PHP – Mysql

Database Mysql paling sering digunakan untuk penyimpanan data di PHP, artikel berikut akan dibahas step-by-step menghubungkan PHP dan database Mysql.

Sebelum mengakses data-data yang ada di Mysql, kita harus membuat koneksi terlebih dahulu ke Mysql dengan cara seperti ini:

```
mysql_connect(servername,username,password);
```

Setelah script koneksi diatas kita, gunakan script berikut untuk memilih nama database

```
mysql_select_db(dbname);
```

Untuk melakukan query ke database seperti select, update, insert dan delete kita bisa menggunakan fungsi berikut:

```
mysql_query(string-query);
```

Khusus untuk query “select” kita dapat gunakan beberapa fungsi untuk mengambil data hasil dari query tersebut, yaitu:

```
mysql_fetch_array(var-query);
mysql_fetch_assoc(var-query);
mysql_fetch_row(var-query);
```

Ketiga fungsi tersebut dapat kita tampilkan menggunakan perulangan di PHP, berikut penjelasan untuk masing-masing fungsi tersebut:

- ✓ `mysql_fetch_array()`, mengambil data hasil query dalam bentuk array asosiasi atau numeric array
- ✓ `mysql_fetch_assoc()`, mengambil data hasil query dalam bentuk array asosiasi
- ✓ `mysql_fetch_row()`, mengambil data hasil query dalam bentuk numeric array. Saran saya jangan gunakan fungsi ini, karena array harus sesuai dengan select field di query

Dan terakhir, untuk menutup query kita gunakan fungsi

```
mysql_close(var-koneksi);
```

Berikut script lengkap dari penjelasan diatas

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}
```

```
mysql_select_db("my_db", $con);

$result = mysql_query("SELECT FirstName, LastName FROM Persons");

while($row = mysql_fetch_array($result))
{
    echo $row['FirstName'] . " " . $row['LastName'];
    echo "<br />";
}

mysql_close($con);

?>
```

### 2.14.2 PHP Adodb

Sebelumnya telah kita pelajari OOP di PHP dan PHP-Mysql, kali ini kita akan menggunakan suatu library menggunakan OOP di PHP untuk mengakses Database. Kali ini kita gunakan PHP ADOdb dan dapat didownload di <http://sourceforge.net/projects/adodb/files/> atau dokumentasi dari library tersebut di <http://phplens.com/adodb/index.html>

Keuntungan kita menggunakan library ADOdb adalah:

Support beberapa macam Database, seperti MySQL, Oracle, Microsoft SQL Server, Sybase, Sybase SQL Anywhere, Informix, PostgreSQL, FrontBase, Interbase (Firebird and Borland variants), Foxpro, Access, ADO and ODBC.

- ✓ Tidak hanya fokus ke select statement saja, juga support untuk insert, delete, dan update
- ✓ Konfigurasi mudah
- ✓ Efisien dalam development

Tahapan menggunakan ADOdb

1. Download dari sourceforge.
2. Extract file zip ke directori web.
3. Gunakan database manager (ex, phpmyadmin, sqlyog, mysqlfront).
4. Buat database untuk contoh (ex, "phonebook").
5. Buat table (ex, "people").
6. Tuliskan kode berikut dan simpan sebagai file php. Letakkan di directori yang sama dengan folder adodb

```
<?php
    include('adodb/adodb.inc.php');

    $databasetype = 'mysql';
    $server = 'localhost';
```

```

$user = 'root';
$password = 'root';
$database = 'inventory';

$db = ADONewConnection($databasetype);
$db->debug = true;
$db->Connect($server, $user, $password, $database);
$rs = $db->Execute('select firstname,lastname,mobile,work,email from people');

print_r($rs->GetRows());

?>

```

Referensi selengkapnya silahkan akses url berikut <http://www.phpeveryday.com/articles/PHP-ADODB-Tutorial-P844.html>

## 2.15 PHP Framework

Framework adalah sebuah lingkungan pengembangan aplikasi berbasis php yang berisi sejumlah class-class yang telah dibuat agar bisa digunakan kembali untuk membuat aplikasi dengan cepat dan efisien. Jenis class-class tsb tergantung tujuan dari php framework tersebut, phpframework yang satu dengan framework yang lain kemungkinan berbeda. Tetapi sebagian besar, php framework yang tersedia saat ini berbasis MVC (model view controller), ini model php framework yang modern.

Jika kita bekerja dengan suatu framework, maka harus mengikuti aturan dari framework tersebut. Karena inilah perusahaan lebih senang menggunakan framework daripada membiarkan si programmer dengan style nya sendiri.

Keuntungan Menggunakan Framework :

- ✓ Mempercepat dan mempermudah pembangunan sebuah aplikasi PHP.
- ✓ Relatif memudahkan dalam proses maintenance karena sudah ada pola tertentu dalam sebuah framework (dengan syarat programmer mengikuti pola standar yang ada)
- ✓ Umumnya framework menyediakan fasilitas-fasilitas
- ✓ yang umum dipakai sehingga kita tidak perlu membangun dari awal (misalnya validasi, ORM, pagination, multiple database, scaffolding, pengaturan session, error handling, dll)

MVC Pattern :

- ✓ MVC (Model View Controller) merupakan suatu metode untuk memisahkan pengedali logika dan pengendali tampilan.
- ✓ Model : mengandung fungsi-fungsi untuk berkomunikasi dengan database
- ✓ View : Berurusan dengan tampilan yang terlihat user
- ✓ Controller : Proses logika, pemeriksaan input, relasi view dan model, dan pengaturan hal-hal yang dilakukan diatur di sini

### 2.15.1 CodeIgniter Framework

Codeigniter(CI) adalah Framework PHP yang populer dan banyak digunakan oleh perusahaan2 seperti amikom.ac.id, detik.com, Kompas.com atau tvone.co.id

CI sudah menggunakan teknologi MVC, yaitu suatu konsep yang cukup populer dalam pembangunan aplikasi web. MVC memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, user interface, dan bagian yang menjadi kontrol aplikasi.

Beberapa fitur yang diberikan CI antara lain: Interaksi dengan database apapun dengan satu bahasa tunggal, manajemen session dan cookies, validasi user input, Membangun html seperti tabel, form, link, dan lainnya dengan kode minimal, komunikasi dengan xmlrpc, ftp, captcha, rss dan teknologi lainnya.

Selain beberapa fitur yang sudah ada tersebut, kita juga bisa menambahkan class-class library di CI dengan mudah.

Dan beberapa kelebihan CI dibanding Framework PHP yang lain adalah:

- ✓ Performa sangat cepat : salah satu alasan tidak menggunakan framework adalah karena eksekusinya yang lebih lambat daripada PHP from the scratch, tapi Codeigniter sangat cepat bahkan mungkin bisa dibanding codeigniter merupakan framework yang paling cepat dibanding framework yang lain.
- ✓ Konfigurasi yang sangat minim (nearly zero configuration) : tentu saja untuk menyesuaikan dengan database dan keleluasaan routing tetap diizinkan melakukan konfigurasi dengan mengubah beberapa file konfigurasi seperti database.php atau autoload.php, namun untuk menggunakan codeigniter dengan setting standard, anda hanya perlu merubah sedikit saja file pada folder config.
- ✓ Banyak komunitas: dengan banyaknya komunitas CI ini, memudahkan kita untuk berinteraksi dengan yang lain, baik itu bertanya atau teknologi terbaru.
- ✓ Dokumentasi yang sangat lengkap : Setiap paket instalasi codeigniter sudah disertai user guide yang sangat bagus dan lengkap untuk dijadikan permulaan, bahasanya pun mudah dipahami.

Langkah2 Instalasi atau Setup awal menggunakan CI

1. Ekstrack file .zip dari hasil download CI
2. Optional edit file index.php ubah baris ke 60 dan baris ke 76, \$system\_path bisa anda ubah sesuai nama folder di CI anda dan \$application\_folder juga dapat dirubah dengan nama lain.
3. Edit file /applicaiton/config/config.php baris ke 17, edit base\_url sesuai dengan alamat web project anda
4. Default controller adalah /application/controllers/welcome.php dapat anda ubah dari file /application/config/routes.php
5. akses alamat web project anda melalui browser
6. Untuk penambahan file2 php dapat di lakukan di folder /application/controllers, /application/models dan /application/views

## 2.16 PHP Security

PHP merupakan bahasa pemrograman yang paling banyak digunakan untuk develop aplikasi web, karena kemudahannya, simple, dan support multi platform. Oleh karena itu banyak sekali serangan-serangan ke web dengan bahasa pemrograman PHP dibanding dengan bahasa pemrograman yang lain. Berikut beberapa kemungkinan serangan yang dilakukan ke Web PHP:

- ✓ SQL Injection, adalah salah satu cara yang paling sering digunakan oleh hacker untuk menyerang web dengan memanfaatkan kelemahan coding di script PHP. Cara ini dilakukan dengan menyerang url dan form yang akan melakukan eksekusi script sql tanpa melindungi variabel yang dikirim.
- ✓ Posting Spam, adalah dengan mengirim data ke form dalam jumlah yang sangat banyak, sehingga membuat isi tabel di database menjadi sangat besar. Cara ini dapat dilakukan dengan menggunakan script yang langsung mengeksekusi insert ke form.
- ✓ PHP Shell, adalah menanam script PHP di server yang dapat di eksekusi sewaktu-waktu untuk merusak content, DB dan script web itu sendiri, Cara ini dapat dilakukan dengan memanfaatkan form upload yang belum tervalidasi atau dapat juga memanfaatkan hak akses direktori.
- ✓ HTML Injection, adalah memposting script2 html, css dan javascript melalui form yang belum tervalidasi untuk merusak tampilan/content web.

Masih banyak jenis serangan-serangan ke web PHP selain ke-4 cara diatas, yang mungkin tidak kita persiapkan untuk mencegahnya. Berikut beberapa cara untuk menghindari beberapa serangan tersebut, paling tidak untuk meminimalkan serangan ke web kita:

1. Error reporting, adalah cara untuk tidak menampilkan pesan saat web kita error, karena sangat berbahaya jika sampai nama tabel-tabel kita bisa terbaca, akan memungkinkan untuk melakukan sql injection atau serangan yang lain. Caranya adalah dengan setting di php.ini yaitu : `display_errors = Off` saat web sudah online, meskipun saat development di localhost bisa kita set `display_errors = On` untuk mempermudah develop web tsb.
2. Validasi variabel yang akan dieksekusi sql, digunakan untuk menghindari sql injection yaitu dengan cara menggunakan fungsi `mysql_real_escape_string()`

```
$studentName = mysql_real_escape_string($_POST['student_name'], $db);
$queryResult = mysql_query("INSERT INTO Students (name) VALUE ('{$studentName}')");
```

3. Direktori akses, untuk direktori tujuan upload gunakan hak akses 755 jangan sampai membuka akses menjadi 777 karena sangat mudah sekali akses public untuk menaruh file PHP shell.
4. Tambahkan file .htaccess untuk men disable direktori upload untuk eksekusi file .php

```
<FilesMatch "\.(php|htm|css|js)$">
    Deny from all
</FilesMatch>
```

5. Validasi upload form, jika aplikasi anda memperbolehkan upload file ke server, lakukan validasi jenis tipe file apa saja yg boleh diupload, tutup koneksi upload untuk file type php, html, css atau js. Untuk mengambil type file gunakan perintah `$_FILES['type']` di php.



- Gunakan captcha untuk script login admin atau form comment yang bersifat public.

Beberapa cara diatas paling tidak dapat meminimalkan serangan-serangan ke web, meskipun masih banyak cara lain yang dapat digunakan untuk mencegahnya.. Jika ada tambahan cara-cara yang lain silahkan ditambahkan di comment.

### 3 Web Service

Web service adalah suatu sistem yang mendukung interoperabilitas dan interaksi antar sistem pada suatu jaringan. Teknologi web service menawarkan kemudahan menjembatani setiap informasi tanpa mempermasalahkan perbedaan teknologi yang digunakan oleh masing-masing sumber.

Web service menggunakan format XML dalam melakukan pertukaran data, sehingga dapat diakses oleh sistem lain walaupun berbeda platform, sistem operasi maupun berbeda bahasa pemrograman.

Web service bertujuan meningkatkan kolaborasi antar pemrogram dan perusahaan, yang memungkinkan sebuah fungsi di dalam web service dapat dipinjam oleh aplikasi lain tanpa perlu mengetahui detail pemrograman yang terdapat di dalamnya.

Ada beberapa pendekatan mengimplementasikan web service. Tiga yang paling banyak digunakan adalah Representational State Transfer (REST), XML-RPC, dan SOAP. Namun kebanyakan aplikasi enterprise menggunakan SOAP. Yang akan saya bahas dalam artikel kali ini adalah SOAP saja..

SOAP (Simple Object Access Protocol) merupakan protokol yang digunakan untuk mempertukarkan data atau informasi dalam format XML. SOAP dapat dikatakan sebagai gabungan antara HTTP dengan XML karena SOAP umumnya menggunakan protocol HTTP sebagai sarana transport datanya dan data akan dipertukarkan ditulis dalam format XML. Karena SOAP menggunakan HTTP dan XML maka SOAP memungkinkan pihak-pihak yang mempunyai platform, system operasi dan perangkat lunak yang berbeda dapat saling mempertukarkan datanya.

Dalam PHP sendiri terdapat suatu file library yang dapat digunakan untuk mencreate SOAP Sever yaitu menggunakan NuSOAP (<http://sourceforge.net/projects/nusoap/>)

NuSOAP adalah salah satu dari sekian toolkit yang tersedia untuk programmer PHP yang ingin bekerja dengan layanan SOAP. Keunggulan yang ditawarkan oleh NuSOAP adalah karena kesederhanaan sebab justru karena sederhana dan mudah sehingga kecepatannya menjadi lebih baik. Adanya NuSOAP menyebabkan peningkatan keuntungan menggunakan bahasa PHP sebagai bahasa pemrograman.

#### Code NuSOAP Serverr

```
<?php
// Pull in the NuSOAP code
require_once('nusoap.php');
// Create the server instance
$server = new soap_server;
```

```
// Register the method to expose
$server->register('hello');
// Define the method as a PHP function
function hello($name) {
    return 'Hello, ' . $name;
}
// Use the request to (try to) invoke the service
$HTTP_RAW_POST_DATA = isset($HTTP_RAW_POST_DATA) ? $HTTP_RAW_POST_DATA : '';
$server->service($HTTP_RAW_POST_DATA);
?>
```

### Code NuSOAP Client

```
<?php
// Pull in the NuSOAP code
require_once('nusoap.php');
// Create the client instance
$client = new soapclient('http://localhost/phphack/helloworld.php');
// Call the SOAP method
$result = $client->call('hello', array('name' => 'Scott'));
// Display the result
print_r($result);
?>
```