

# PEMOGRAMAN WEB II



*Nama : Mega Lestari*

*NPM : 43A87006160309*

*S1/TI/6A Malam*

**Program Studi S1 Teknik Infomatika STMIK Bani Saleh**

**Bekasi**

**2019**

## DAFTAR ISI

DAFTAR ISI .....	
Pengenalan Dasar Array .....	
Pengenalan Sintak Dasar Object Oriented Programming(OOP).....	
Pengenalan Arsitektur Codeigniter.....	
Pengenalan Konfigurasi Database Mysql dengan CI .....	
Pengenalan Controller dari MVC .....	
Pengenalan View dari MVC .....	
Pengenalan Model dari MVC .....	
Pengenalan Autentifikasi & Autorisasi dari login hingga hak akses.....	
Pengenalan JQuery EasyUI .....	
Membuat Report PDF dan Excel pada Pemograman Web .....	

## BAB I

### a. Pengenalan Array

Array adalah sebuah tipe variabel yang dapat menyimpan beberapa data dan memanipulasinya dengan lebih efektif. Sebuah array akan menyimpan beberapa item data yang memiliki tipe data sama didalam sebuah blok memori yang berdekatan yang kemudian dibagi menjadi beberapa ruang. Array merupakan variabel/lokasi tertentu yang memiliki satu nama sebagai identifier, dan dapat menyimpan lebih dari sebuah nilai.

Number : 0 1 2 3

1 2 3 4

Gambar 1. Contoh dari Integer Array

### 2. Pendeklarasian Array

Pada saat mendeklarasikan array, kita harus membuat daftar dari tipe data yang diikuti oleh sepasang tanda kurung [ ], lalu diikuti nama identifier-nya.

```
tipe_array nama_array[ ];
```

```
tipe_array[ ] nama_array[ ];
```

```
contoh : int nilai[ ];
```

```
char[ ] huruf;
```

### 3. Pengaksesan sebuah Elemen Array

Untuk mengakses sebuah elemen dalam array, atau mengakses sebagian dari array, anda harus menggunakan sebuah angka atau yang disebut indeks atau subscript.

Adapun caranya dengan memakai new karena di dalam Java adalah suatu array dianggap suatu obyek.

Lembaga Pelatihan Kerja IT LINK TECH Makassar 1

Modul 6. Deklarasi Array

Modul Java Fundamental 2013/2014

Format penulisan :

```
nama_array = new tipe_array[total_elemen_array];
```

contoh :

```
int nilai[ ];
```

```
nilai = new int[5];
```

Untuk dapat mengakses elemen array dapat dilakukan dengan menyebutkan elemen ke berapa dari array yang akan diakses, seperti berikut :

```
nama_array[elemen_array]
```

Kita juga dapat melakukan deklarasi dan pembuatan array hanya pada satu baris

statement. Format penulisannya sebagai berikut :

```
tipe_array nama_array[ ] = new tipe_array[total_elemen_array];
```

```
contoh : int nilai[ ] = new int[5];
```

Untuk inisialisasi array dapat dilakukan dengan format berikut :

```
tipe_array nama_array[] = {nilai_indeks_0, nilai_indeks_1, ...,
```

```
nilai_indeks_n};
```

```
contoh : int nilai[ ] = {70, 65, 85};
```

#### 4. Panjang Array

Untuk mengetahui berapa banyak elemen di dalam sebuah array, kita dapat menggunakan atribut `length` dari array. Atribut ini akan mengembalikan ukuran dari array itu sendiri. Format penulisannya sebagai berikut :

`arrayName.length`

kita dapat menuliskannya kembali seperti berikut :

```
public class ArraySample
{
    public static void main (String[] args) {
        int[] ages = new int[100];
        for (int i = 0; i < ages.length; i++) {
            System.out.print(ages[i]);
        }
    }
}
```

Lembaga Pelatihan Kerja IT LINK TECH Makassar 2

Modul Java Fundamental 2013/2014

#### 5. Array Multidimensi

Array multidimensi diimplementasikan sebagai array yang terletak di dalam array.

Array multidimensi dideklarasikan dengan menambahkan jumlah tanda kurung setelah nama array. Sebagai contoh :

//Elemen 512 x 128 dari integer array

```
int[ ][ ] twoD = new int[512][128];
```

//karakter array 8 x 6 24

```
char[ ][ ][ ] threeD = new char [8][16][24];
```

//String array 4 baris x 2 kolom

```
String[ ][ ] dogs = { {"terry", "brown"}, {"Kristin", "White"},  
{"Toby", "Gray"}, {"Fido", "Black"} };
```

untuk mengakses sebuah elemen didalam array multidimensi, sama saja dengan mengakses array satu dimensi.

Latihan :

Percobaan 1

```
public class ContohArray1 {  
  
    public static void main (String[ ] args) {  
  
        int[ ] ages = new int[100];  
  
        for (int i = 0; i < 100; i++) {  
  
            System.out.print(ages[i]);  
  
        }  
  
    }  
}
```

Percobaan 2

```
public class Array1 {  
  
    public static void main (String[ ] args) {
```

```

int nilai[ ] = new int[3];

nilai[0] = 70;

nilai[1] = 80;

nilai[2] = 65;

double ratarata = 0.0;

for (int i = 0; i < nilai.length; i++) {

ratarata+ = nilai[i];

ratarata/=nilai.length;

System.out.print("Nilai rata-rata = "+ ratarata);

}

}

```

Lembaga Pelatihan Kerja IT LINK TECH Makassar 3

Modul Java Fundamental 2013/2014

Percobaan 3

```

import java.text.NumberFormat;

public class Array2 {

public static void main (String[ ] args) {

NumberFormat nf = NumberFormat.getInstance();

nf.setMaximumFractionDigits(3);

int nilai[][] = new int[2][3];

nilai[0][0] = 85;

```

```

nilai[0][1] = 81;

nilai[0][2] = 78;

nilai[1][0] = 65;

nilai[1][1] = 73;

nilai[1][2] = 71;

String MK[] = {"RPL", "PBO"};

double ratarataMK[] = new double[nilai.length];

for (int i = 0; i < nilai.length; i++) {

for (int j = 0; j < nilai[0].length; j++) {

ratarataMK[i] += nilai[i][j];

}

ratarataMK[i] /= nilai[0].length;

}

System.out.println("Nilai Mata Kuliah\n");

System.out.println("MK\tMinggu1\tMinggu2\tMinggu3\tRata-Rata");

for (int i=0; i<nilai.length; i++) {

System.out.print (MK[i] + "\t");

for (int j=0; j<nilai[0].length;j++) {

System.out.print(nilai[i][j] + "\t");

}

System.out.print(nf.format(ratarataMK[i]) + "\t");

```



}  
  
}  
  
}

Lembaga Pelatihan Kerja IT LINK TECH Makassar 4

Modul Java Fundamental 2013/2014

Soal :

Nomor 1 :

Buatlah sebuah String array yang akan menginisialisasi 7 hari dalam seminggu.

Contoh :

```
String days[ ] = {"Monday", "Tuesday", . . .} ;
```

Gunakan while-loop, kemudian print semua nilai dari array (gunakan juga untuk do while dan for-loop) using a while-loop.

Nomor 2 :

Diketahui daftar nilai siswa sebagai berikut :

Stambuk NamaMhs Java Basis Data Micro

1 Adhi 81 90 62

2 Ojhi 77 70 92

3 Erin 89 55 65

4 Afgan 50 83 87

Buatlah program untuk menampilkan laporan sebagai berikut :

NRP Rata-rata

---

1 77.67

2 79.67

3 69.67

4 73.33

---

## **b. Pengenalan Sintak Dasar Object Oriented Programming(OOP)**

### **Pengertian OOP**

OOP (Object Oriented Programming) adalah suatu metode pemrograman yang berorientasi kepada objek. Tujuan dari OOP diciptakan adalah untuk mempermudah pengembangan program dengan cara mengikuti model yang telah ada di kehidupan sehari-hari. Jadi setiap bagian dari suatu permasalahan adalah objek, nah objek itu sendiri merupakan gabungan dari beberapa objek yang lebih kecil lagi. Saya ambil contoh Pesawat, Pesawat adalah sebuah objek. Pesawat itu sendiri terbentuk dari beberapa objek yang lebih kecil lagi seperti mesin, roda, baling-baling, kursi, dll. Pesawat sebagai objek yang terbentuk dari objek-objek yang lebih kecil saling berhubungan, berinteraksi, berkomunikasi dan saling mengirim pesan kepada objek-objek yang lainnya. Begitu juga dengan program, sebuah objek yang besar dibentuk dari beberapa objek yang lebih kecil, objek-objek itu saling berkomunikasi, dan saling berkiriman pesan kepada objek yang lain.

### **➤ Konsep OOP**

#### **1. Abstrak Class**

Kelas merupakan deskripsi abstrak informasi dan tingkah laku dari sekumpulan data.

Kelas dapat diilustrasikan sebagai suatu cetak biru(blueprint) atau prototipe yang digunakan untuk menciptakan objek.

Kelas merupakan tipe data bagi objek yang mengenkapsulasi data dan operasi pada data dalam suatu unit tunggal.

Kelas mendefinisikan suatu struktur yang terdiri atas data kelas (data field), prosedur atau fungsi (method), dan sifat kelas (property).

## 2. Encapsulation

Istilah enkapsulasi sebenarnya adalah kombinasi data dan fungsionalitas dalam sebuah unit tunggal sebagai bentuk untuk menyembunyikan detail informasi.

Proses enkapsulasi memudahkan kita untuk menggunakan sebuah objek dari suatu kelas karena kita tidak perlu mengetahui segala hal secara rinci.

Enkapsulasi menekankan pada antarmuka suatu kelas, atau dengan kata lain bagaimana menggunakan objek kelas tertentu.

Contoh: kelas mobil menyediakan antarmuka fungsi untuk menjalankan mobil tersebut, tanpa kita perlu tahu komposisi bahan bakar, udara dan kalor yang diperlukan untuk proses tersebut.

## 3. Inheritance

Kita dapat mendefinisikan suatu kelas baru dengan mewarisi sifat dari kelas lain yang sudah ada.

Penurunan sifat ini bisa dilakukan secara bertingkattingkat, sehingga semakin ke bawah kelas tersebut menjadi semakin spesifik.

Sub kelas memungkinkan kita untuk melakukan spesifikasi detail dan perilaku khusus dari kelas supernya.

Dengan konsep pewarisan, seorang programmer dapat menggunakan kode yang telah ditulisnya pada kelas super berulang kali pada kelas-kelas turunannya tanpa harus menulis ulang semua kodekode itu.

## 4. Polymorphism

- 1) Polimorfisme merupakan kemampuan objekobjek yang berbeda kelas namun terkait dalam pewarisan untuk merespon secara berbeda terhadap suatu pesan yang sama.

- 2) Polimorfisme juga dapat dikatakan kemampuan sebuah objek untuk memutuskan method mana yang akan diterapkan padanya, tergantung letak objek tersebut pada jenjang pewarisan.
- 3) Method overriding.
- 4) Method name overloading.

### **c. Pengenalan Arsitektur Codeigniter**

Codeigniter merupakan suatu Web Application Framework (WAF) yang dirancang khusus untuk mempermudah para developer web dalam mengembangkan aplikasi berbasis web.

Codeigniter berisi kumpulan kode berupa pustaka (library) dan alat (tools) yang dipadukan sedemikian rupa menjadi suatu kerangka kerja (framework).

Codeigniter adalah framework web untuk bahasa pemrograman PHP, yang dibuat oleh Rick Ellis pada tahun 2006, penemu dan pendiri EllisLab ([www.ellislab.com](http://www.ellislab.com)).

EllisLab adalah tim kerja yang berdiri pada tahun 2002 dan bergerak di bidang pembuatan software dan tool untuk para pengembang web.

Sejak tahun 2014 sampai sekarang, EllisLab telah menyerahkan hak kepemilikan Codeigniter ke British Columbia Institute of Technology (BCIT) untuk proses pengembangan lebih lanjut.

Saat ini, situs web resmi dari Codeigniter telah berubah dari [www.ellislab.com](http://www.ellislab.com) menjadi [www.codeigniter.com](http://www.codeigniter.com).

Codeigniter memiliki banyak fitur (fasilitas) yang membantu pada pengembang PHP untuk dapat membuat aplikasi web secara mudah dan cepat.

Dibandingkan dengan framework web PHP lainnya, harus diakui bahwa Codeigniter memiliki desain yang lebih sederhana dan bersifat fleksibel (tidak kaku).

Codeigniter mengizinkan para pengembang untuk menggunakan framework secara parsial atau secara keseluruhan.

Ini berarti bahwa codeigniter masih memberi kebebasan kepada pengembang untuk menulis bagian-bagian kode tertentu di dalam aplikasi menggunakan cara konvensional (tanpa framework).

Codeigniter menganut pola desain atau arsitektur Model-View-Controller (MVC), yang memisahkan bagian kode untuk penanganan proses bisnis dengan bagian kode untuk keperluan presentasi (tampilan).

Pembahasan lebih lanjut tentang MVC klik disini.

Dengan menggunakan pola desain ini, memungkinkan para pengembang web untuk mengerjakan aplikasi berbasis web secara bersama (teamwork).

Dengan begitu para pengembang web lebih bisa berfokus pada bagiannya masing-masing tanpa mengganggu bagian yang lain. Sehingga aplikasi yang dibangun akan selesai lebih cepat.

#### **d. Pengenalan Konfigurasi Database Mysql dengan CI**

Cara mengakses database mysql pada codeigniter dengan konsep model view controller

Setelah membahas tentang Model, View dan Controller, kali ini kita mulai masuk pada pembahasan tentang cara mengakses database mysql pada CodeIgniter. Secara sepintas, fungsi - fungsi yang digunakan untuk mengakses database mysql berbeda dengan fungsi - fungsi standar php dalam kaitannya dengan mysql. Pada codeIgniter, fungsi - fungsi ini di ringkas dengan nama - nama lain sehingga terkesan lebih enak untuk dihafal.

Berikut akan kami sebutkan beberapa fungsi database dalam codeigniter dalam kaitannya dengan mysql

Fungsi query('perintah sql') » Sama dengan fungsi mysql\_query('perintah sql')

Fungsi num\_rows() » Sama dengan fungsi mysql\_num\_rows()

Fungsi result() » Sama dengan fungsi mysql\_fetch\_array()

Fungsi row() » Menampilkan hasil query satu baris

Setelah mengetahui fungsi - fungsi diatas, berikut beberapa hal yang harus kita lakukan pada CodeIgniter kita supaya bisa mengakses database MySQL

## 1. Membuat database dan tabel posting

Untuk memperringkas pembahasan kali ini, silahkan download contoh tabel postig pada link download di bagian bawah halaman ini

## 2. Setting autoload pada /application/config/autoload.php

Pada script autoload.php, carilah bagian `$autoload['libraries'] = array()` , kemudian tambahkan text database pada daftar arraynya seperti terlihat dibawah ini

```
$autoload['libraries'] = array('database');
```

## 3. Setting koneksi pada /application/config/database.php

Pada script database.php , carilah bagian seperti terlihat pada script dibawah ini. Masukkan host database, username, password dan nama database yang akan diakses

```
$active_group = 'default';
```

```
$active_record = TRUE;
```

```
$db['default']['hostname'] = 'localhost';
```

```
$db['default']['username'] = 'root';
```

```
$db['default']['password'] = 'kutukupret';
```

```
$db['default']['database'] = 'tutorial';
```

```
$db['default']['dbdriver'] = 'mysql';
```

```
$db['default']['dbprefix'] = '';
```

```
$db['default']['pconnect'] = TRUE;
```

```
$db['default']['db_debug'] = TRUE;
```

```
$db['default']['cache_on'] = FALSE;
```

```
$db['default']['cachedir'] = '';
```

```
$db['default']['char_set'] = 'utf8';
```

```
$db['default']['dbcollat'] = 'utf8_general_ci';
```

```
$db['default']['swap_pre'] = '';
```

```
$db['default']['autoinit'] = TRUE;
```

```
$db['default']['stricton'] = FALSE;
```

4. Membuat controller home.php /application/controllers/home.php

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
```

```
class Home extends CI_Controller {
```

```
    /**
```

```
    * Index Page for this controller.
```

```
    *
```

```
    * Maps to the following URL
```

```
    *          http://example.com/index.php/home
```

```
    *    - or -
```

```
    *          http://example.com/index.php/home/index
```

```
    *    - or -
```

```
    * Since this controller is set as the default controller in
```

```
    * config/routes.php, it's displayed at http://example.com/
```

```
    *
```

```
    * So any other public methods not prefixed with an underscore will
```

```
    * map to /index.php/home/<method_name>
```

```
    * @see http://codeigniter.com/user_guide/general/urls.html
```

```
    */
```

```
    public function index()
```

```
    {
```

```
        echo 'Selamat datang di CodeIgniter';
```

```
    }
```

```
    function posting()
```

```

    {
        $this->load->model('Webmodel');
        $data['posting'] = $this->Webmodel->list_posting();
        $this->load->view('posting',$data);
    }
}

```

/\* End of file home.php \*/

/\* Location: ./application/controllers/home.php \*/

## 5. Membuat model webmodel.php /application/models/webmodel.php

Pada bagian model inilah kita akan menggunakan fungsi - fungsi database untuk kemudian dikirimkan ke controller dan kemudian dicetak oleh view, contoh model yang kita buat untuk mengakses database adalah seperti berikut :

```
<?php
```

```
class Webmodel extends CI_Model {
```

```
    function __construct()
```

```
    {
```

```
    }
```

```
    /* Menampilkan posting */
```

```
    function list_posting()
```

```
    {
```

```
        $query = $this->db->query('select * from posting order by PID desc limit
20');
```

```
        //lihat apakah ada data dalam tabel
    }
}
```



```

        $num = $query->num_rows();
        if($num>0){
            //Mengirimkan data array hasil query
            return $query->result();
            //Function result() hampir sama dengan function
mysql_fetch_array()
        }
        else{
            return 0;
            //Kirimkan 0 jika tidak ada datanya
        }
    }
}

```

#### 6. Membuat view posting.php /application/views/posting.php

Setelah semua proses diatas, kita juga harus membuat script view yang akan menampilkan hasil query oleh model. script posting.php ini kita simpan pada direktory views.

```

<?php
//periksa apakah datanya array atau hanya 0
if(is_array($posting)){
    echo '<ol>';
    foreach($posting as $key){
        $judul = '<li>'.$key->Judul.'</li>';
        echo $judul;
    }
    echo '</ol>';
}
?>

```

Setelah selesai dengan semua langkah - langkah diatas, cobalah mengakses aplikasi kita dengan memasukkan alamat berikut pada browser kita

[http://localhost/nama\\_folder\\_ci/index.php/home/posting](http://localhost/nama_folder_ci/index.php/home/posting)

#### **e. Pengenalan Controller dari MVC**

Model View Controller merupakan suatu konsep yang cukup populer dalam pembangunan aplikasi web, berawal pada bahasa pemrograman Small Talk (Wikipedia), MVC memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, user interface, dan bagian yang menjadi kontrol aplikasi. Terdapat 3 jenis komponen yang membangun suatu MVC pattern dalam suatu aplikasi yaitu :

View, merupakan bagian yang menangani presentation logic. Pada suatu aplikasi web bagian ini biasanya berupa file template HTML, yang diatur oleh controller. View berfungsi untuk menerima dan merepresentasikan data kepada user. Bagian ini tidak memiliki akses langsung terhadap bagian model.

Model, biasanya berhubungan langsung dengan database untuk memanipulasi data (insert, update, delete, search), menangani validasi dari bagian controller, namun tidak dapat berhubungan langsung dengan bagian view.

Controller, merupakan bagian yang mengatur hubungan antara bagian model dan bagian view, controller berfungsi untuk menerima request dan data dari user kemudian menentukan apa yang akan diproses oleh aplikasi.

Singkat kata Model untuk mengatur alur database, View untuk menampilkan web, sedangkan Controller untuk mengatur alur kerja antara Model dan View. Jadi misalnya Anda ingin membuat akun facebook atau e-mail. Pertama Anda akan melihat tampilan sign-up / register itulah View, kemudian Anda menuliskan form pengisian seperti username, password, dll dan Anda klik sign-up / register maka di sini View memanggil Controller dan Controller memanggil Model, sedangkan Model mengecek apakah Anda sudah sesuai dengan kriteria pendaftaran seperti sudah mengisi username, password dan lain sebagainya. Kemudian Model

mengembalikan (callback) ke Controller dan Controller mengembalikan ke View, dan kita akan melihat berhasil atau tidak kita sign-up/register.

#### **f. Pengenalan View dari MVC**

MVC adalah sebuah konsep arsitektural yang membagi suatu pengembangan aplikasi menjadi 3 bagian komponen logika yakni Model, View, dan Controller. Jadi proses dasarnya, Model menghantarkan Data dari Database sehingga dapat di kelola oleh Controller kedalam View, sehingga View dapat menambikan Data yang sudah dan akan di proses.

##### **Data Tables**

Karena pada tutorial Belajar Codeigniter 3 : Controller dan Model kita sudah berhasil menampilkan data dari database sehingga ditampilkan di Controller, kita akan membuat tampilan viewnya terlebih dahulu. pada directory application/views buatlah sebuah file berekstensi php dengan nama data\_mahasiswa.php dan masukan kode berikut kedalamnya :

Kode diatas akan menampilkan data dalam bentuk tabel, namun \$data yang digunakan pada kode diatas diambil dari Model melewati Controller, karena kita belum menghubungkan data dari model ke dalam view maka jika kita menggunakan kode `$this->load->view('data_mahasiswa');` maka yang muncul adalah sebuah Error.

##### **Menghubungkan Data**

Untuk menghubungkan Kedua Model dan View yang sudah kita sambungkan kedalam Controller, kita akan mengubah beberapa kode yang terdapat dalam function index di controller helloworld menjadi seperti berikut :

`$data = $this->load->model('mymodel');` Kode ini digunakan untuk menghubungkan controller dengan model yang kita buat.

`$data = $this->mymodel->GetMahasiswa();` Kode ini sudah kita pelajari pada tutorial sebelumnya.

`$data = array('data' => $data);` Kode ini digunakan untuk mengubah data yang sudah kita panggil dari model menjadi sebuah array.

`$this->load->view('data_mahasiswa', $data);` Kode ini merupakan Kode memanggil View, namun kita menambahkan , `$data` untuk membawa data dari model ke dalam View, sehingga `$data` dalam view merupakan sebuah array yang berisi data dari model.

#### **g. Pengenalan Model dari MVC**

Model merepresentasikan knowledge (pengetahuan). Model dapat berupa single object (jarang dipakai) atau terdiri dari objek-objek yang terstruktur dimana tiap objek memiliki fungsi tersendiri sesuai dengan kebutuhan. Pada framework PHP, model biasanya digunakan sebagai penghubung antara controller dengan database untuk mengambil data pada database. Hal ini merujuk pada konsep MVC dimana model digunakan sebagai representasi dari pengetahuan (database).

#### **h. Pengenalan Autentifikasi & Autorisasi dari login hingga hak akses**

Otentikasi & Otorisasi

Kedua konsep ini perlu dipahami sebelum memulai membuat kode login sistem yang akan dijelaskan di tutorial php ini.

Otentikasi atau Authentication adalah proses mengkonfirmasi identitas dari seseorang atau suatu benda. Misalkan pada saat pulang ke rumah dan mengetuk pintu masuk, maka ibu Anda akan menanyakan siapa yang mengetuk pintu, sebelum mempersilahkan Anda masuk.

Otorisasi atau Authorization adalah proses memberi izin seseorang atau suatu benda untuk melakukan sesuatu. Contoh, setelah Anda dipersilahkan masuk, maka Anda akan

diizinkan untuk duduk di ruang tamu, menonton tv, membaca majalah di meja, masuk ke kamar tidur Anda. Namun tidak mengizinkan Anda menggunakan laptop ayah Anda atau membuka kotak penyimpanan.

Proses otentikasi & otorisasi menyangkut 4 komponen penting.

Orang atau benda yang akan melakukan sesuatu dan dikonfirmasi identitasnya, dalam contoh di atas Anda sendiri;

Aksi yang dilakukan, misal membaca, menonton;

Benda yang dikenakan Aksi, misal tv, buku, kamar tidur;

Peran atau role dari orang di dalam sistem, misal peran Anda di rumah yaitu sebagai Anak.

Perlu ditekankan di komponen terakhir peran . Pada saat Anda berdiri di depan pintu rumah, saat mengetuk pintu, maka peran Anda adalah "TAMU" atau "GUEST" . Setelah ibu Anda mengkonfirmasi diri Anda dan mengizinkan Anda masuk ke dalam rumah, maka peran Anda berubah menjadi "ANAK" . Jika yang mengetuk pintu ayah Anda maka setelah diizinkan masuk perannya berubah dari "TAMU" ke "KEPALA KELUARGA" . Peran inilah yang menentukan aksi apa yang boleh Anda lakukan terhadap sebuah benda. Aksi Aksi apa saja yang bisa Anda lakukan berdasarkan peran Anda sering disebut dengan hak akses .

Menyiapkan Tabel Database

Langkah pertama yang perlu dilakukan adalah membuat tabel yang akan digunakan pada tutorial ini. Buatlah tabel dengan nama "user" seperti di bawah:

kolom	tipe data	keterangan
user_id	varchar(40)	primary key
name	varchar(50)	
password	varchar(50)	
type	varchar(20)	mencatat tipe/peran user

```
CREATE TABLE `user` (
```

```

`user_id` VARCHAR(40) NOT NULL ,
`name` VARCHAR(50) NULL ,
`password` VARCHAR(50) NULL ,
`type` VARCHAR(20) NULL ,
PRIMARY KEY (`user_id`) )
DEFAULT CHARACTER SET = utf8
COLLATE = utf8_general_ci;

```

Perlu diperhatikan di tabel di atas ada satu kolom type, kolom type digunakan untuk mencatat/menentukan peran pengguna setelah berhasil login.

### Membatasi Hak Akses Pengguna Berdasarkan Tipe/Peranan (Otorisasi)

Pada bagian ini akan dimulai membuat kode php untuk membatasi hak akses. Seperti dijelaskan di atas otentikasi & otorisasi menyangkut 4 komponen. Jika dikaitkan dengan contoh kode di tutorial ini maka 4 komponen tersebut adalah sebagai berikut:

Orang atau benda-> pengguna atau pengunjung website;

Aksi-> mengakses/menload/mengunjungi halaman web;

Benda-> halaman web, misal: index.php, login.php, admin.php;

Peran-> guest,admin,user ditentukan dari kolom type di tabel user.

### File Konfigurasi

Pertama buat file config.php untuk menyimpan semua konfigurasi umum seperti konfigurasi koneksi database.

```

<?php
if(!defined('__NOT_DIRECT')){
    //mencegah akses langsung ke file ini
    die('Akses langsung tidak diizinkan!');
}

define('BASE_URL', '/tutorial/multi-login/');
//koneksi database

```

```
define('DB_HOST','localhost');  
define('DB_USER','root');  
define('DB_PASS', 'blah');  
define('DB_NAME' , 'test');
```

Pada baris ke-2 sampai ke-5 digunakan untuk mencegah akses langsung ke file config.php, file harus di-include oleh file lain dengan mendefinisikan konstanta `__NOT_DIRECT` terlebih dahulu.

Baris ke-7 adalah bagian terpenting, mendefinisikan konstanta `BASE_URL` jika Anda mengakses tutorial ini di browser seperti `http://localhost/tutorial/multi-login/` maka `BASE_URL` diisi dengan `/tutorial/multi-login/`. Jika `http://localhost/multi-login/` maka isi `BASE_URL` dengan `/multi-login/`. Tinggal hilangkan localhost, lebih mudahnya lagi jika Anda membuat tutorial ini di folder `htdocs/tutorial/multi-login` maka isi `BASE_URL` dengan `tutorial/multi-login`.

Baris selanjutnya adalah mendefinisikan konstanta untuk koneksi ke database.

### Konfigurasi Hak Akses Masing Masing Tipe User

Setelah selesai mendefinisikan konfigurasi global, selanjutnya adalah membuat file konfigurasi hak akses masing masing tipe user, halaman web mana saja yang bisa diakses setiap tipe user.

Pertama buat folder dengan nama "akses".

Di dalam folder akses buatlah 3 file, file `admin.php` untuk pengguna dengan peran admin, file `user.php` untuk pengguna dengan peran user (user biasa), dan file `guest.php` untuk pengguna dengan peran guest.

Masing masing file tersebut adalah sebagai berikut:

file `admin.php`

```
$__akses_config = array(
```

```
'login.php',
'logout.php',
'admin/index.php',
'admin/list-user.php',
'admin/add-user.php',
'admin/edit-user.php',
'admin/delete-user.php',
'admin/add-artikel.php',
'admin/edit-artikel.php',
'admin/list-artikel.php',
'admin/delete-artikel.php',
'index.php',
'view-artikel.php'
);
file user.php
```

```
<?php
$__akses_config = array(
    'login.php',
    'logout.php',
    'admin/index.php',
    'admin/add-artikel.php',
    'admin/edit-artikel.php',
    'admin/list-artikel.php',
    'admin/delete-artikel.php',
    'index.php',
    'view-artikel.php'
);
file guest.php
```

```
<?php
```



```

$__akses_config = array(
    'login.php',
    'logout.php',
    'register.php',
    'index.php',
    'view-artikel.php'
);

```

ketiga file di atas berisi satu variabel dengan nama \$\_\_akses\_config dengan tipe data array, variabel \$\_\_akses\_config berisi data array dari halaman halaman mana saja yang bisa diakses masing masing tipe user. Contoh admin boleh mengakses halaman admin/list-user.php sedangkan user dan guest tidak bisa, kemudian semua pengguna bisa mengakses halaman login.php.

Jumlah file di folder akses tidak terpaku hanya 3 buah, bisa ditambah sesuai dengan kebutuhan, sesuai dengan tipe user yang diinginkan.

#### File Untuk Otorisasi User

Langkah selanjutnya adalah membuat file cek-akses.php. File ini digunakan untuk menentukan apa peran pengguna dan halaman apa saja yang bisa dikunjungi.

Pertama mencegah akses langsung ke file cek-akses.php, file harus di-include oleh file lain dengan mendefinisikan konstanta \_\_NOT\_DIRECT terlebih dahulu.

```

<?php
if(!defined('__NOT_DIRECT')){
    //mencegah akses langsung ke file ini
    die('Akses langsung tidak diizinkan!');
}

```

Kedua mengecek apakah user sudah login atau belum dan menentukan tipe/peran pengguna.

```
// ....
session_start();
include_once dirname(__FILE__).DIRECTORY_SEPARATOR.'config.php';
if(!isset($_SESSION['my_user_agent'])                                     ||
($_SESSION['my_user_agent']!=md5($_SERVER['HTTP_USER_AGENT']))) {
    //user belum login
    $__tipe_user = 'guest';
} else {
    $__tipe_user = $_SESSION['tipe_user'];
}
```

Membuka session di baris ke-2, baris ke-3 meng-include file config.php yang telah dibuat sebelumnya. Baris ke-4 sampai 10 mengecek apakah pengguna sudah login, jika belum login maka tipenya adalah "guest" dan sebaliknya tipe user sesuai dengan nilai \$\_SESSION['tipe\_user'] (sesuai dengan nilai dari tabel user).

Catatan:

Berikut penjelasan beberapa fungsi yang mungkin baru Anda ketahui:

DIRECTORY\_SEPARATOR adalah konstanta yang berisi nilai 'pemisah folder', di windows nilainya \ dan di linux /.

FILE adalah konstanta yang berisi nilai nama lengkap file beserta nama foldernya misal, file cek-akses.php ada di folder C:\xampp\htdocs\multi-login maka FILE akan berisi C:\xampp\htdocs\multi-login\cek-akses.php.

dirname fungsi untuk mendapatkan nama folder dari sebuah file atau folder, dirname('C:\xampp\htdocs\multi-login\cek-akses.php'); akan menghasilkan 'C:\xampp\htdocs\multi-login'.

Ketiga meng-include salah satu file yang ada di folder akses yang dibuat sebelumnya, sesuai dengan tipe pengguna yang diperoleh pada kode di atas. Jika tipe pengguna guest maka include file akses/guest.php, jika admin maka include file akses/admin.php

```
//.....
$aksesFilename =
dirname(__FILE__).DIRECTORY_SEPARATOR.'akses'.DIRECTORY_SEPARATOR.
$__tipe_user.'.php';
if(!file_exists($aksesFilename)){
    die('Terjadi kesalahan sistem');
}
include $aksesFilename;
```

Keempat menentukan file yang sedang diakses oleh pengguna dan mengecek apakah pengguna diperbolehkan mengakses file tersebut.

```
//.....
$arrayCurrentPath = explode('?',$_SERVER['REQUEST_URI']);
$currentPath = substr($arrayCurrentPath[0], strlen(BASE_URL));
```

```
$allow = in_array($currentPath, $__akses_config);
```

Kelima jika pengguna tidak diizinkan dan pengguna belum login maka pengguna dialihkan ke halaman login (baris 28 dan 29), sebaliknya ditampilkan text "Anda tidak diizinkan mengakses halaman ini!" (baris 31).

```
//.....
if(!$allow){
    if($__tipe_user == 'guest' && $currentPath != 'login.php'){
        header("Location: ".BASE_URL.'login.php');
    }else{
        echo "Anda tidak diizinkan mengakses halaman ini!";
    }
    exit;
}
```

Setelah semua langkah di atas digabungkan maka file cek-akses.php akan seperti:

```

<?php
if(!defined('__NOT_DIRECT')){
    //mencegah akses langsung ke file ini
    die('Akses langsung tidak diizinkan!');
}

session_start();

require_once dirname(__FILE__).DIRECTORY_SEPARATOR.'config.php';

if(!isset($_SESSION['my_user_agent'])                                     ||
($_SESSION['my_user_agent']!=md5($_SERVER['HTTP_USER_AGENT']))) {
    //user belum login
    $__tipe_user = 'guest';
} else {
    $__tipe_user = $_SESSION['tipe_user'];
}

$aksesFilename                                                         =
dirname(__FILE__).DIRECTORY_SEPARATOR.'akses'.DIRECTORY_SEPARATOR.
$__tipe_user.'.php';
if(!file_exists($aksesFilename)){
    die('Terjadi kesalahan sistem');
}
include $aksesFilename;

$arrayCurrentPath = explode('?',$_SERVER['REQUEST_URI']);
$currentPath = substr($arrayCurrentPath[0], strlen(BASE_URL));

$allow = in_array($currentPath, $__akses_config);

```

```

if(!$allow){
    if($_type_user == 'guest' && $currentPath != 'login.php'){
        header("Location: ".BASE_URL.'login.php');
    }else{
        echo "Anda tidak diizinkan mengakses halaman ini!";
    }
    exit;
}

```

### Otentikasi Pengguna

Setelah membuat skrip untuk otorisasi pengguna di atas, langkah selanjutnya adalah membuat skrip untuk Otentikasi user.

### File Login.php

Berikut adalah langkah langkah untuk membuat file login.php, langkah langkah mengotentikasi pengguna.

Pertama mendefinisikan konstanta `__NOT_DIRECT` dan meng-include file cek-akses.php

```

defined('__NOT_DIRECT') || define('__NOT_DIRECT',1);
include 'cek-akses.php';

```

Kedua cek apakah pengguna sudah menginput user\_id & password dan menekan tombol login, kemudian membuka koneksi ke database

```
//....
```

```

if($_POST){
    mysql_connect(DB_HOST,DB_USER,DB_PASS);
    mysql_select_db(DB_NAME);

```

Ketiga cek kebenaran user\_id dan password yang diinput pengguna. Jika benar buat 3 session `$_SESSION['tipe_user']`, `$_SESSION['user_id']`, dan `$_SESSION['my_user_agent']` dan alihkan pengguna ke halaman admin/index.php.

Sebaliknya jika user\_id atau password salah tampilkan pesan "ID User atau password salah!"

```
//.....
$userId = mysql_real_escape_string($_POST['user_id']);
$data = mysql_fetch_array(mysql_query("select * from user where
user_id='".$userId.'""));
if($data != false && $data['password'] == md5($_POST['password'])) {
    //login berhasil
    $_SESSION['tipe_user'] = $data['type'];
    $_SESSION['user_id'] = $data['user_id'];
    $_SESSION['my_user_agent'] = md5($_SERVER['HTTP_USER_AGENT']);
    header("Location: admin/index.php");
} else {
    echo "ID User atau password salah!";
}
}??>
```

Keempat membuat form login

```
//.....
<form method="post" action="">
ID User: <input type="text" name="user_id"/>
Password: <input type="password" name="password"/>
<input type="submit" value="Login"/>
</form>
```

Jika digabungkan file login.php akan menjadi:

```
<?php
defined('__NOT_DIRECT') || define('__NOT_DIRECT',1);
include 'cek-akses.php';
if($_POST){
```

```

mysql_connect(DB_HOST,DB_USER,DB_PASS);
mysql_select_db(DB_NAME);
$userId = mysql_real_escape_string($_POST['user_id']);
$data = mysql_fetch_array(mysql_query("select * from user where
user_id='".$userId."'"));
if($data !== false && $data['password'] == md5($_POST['password'])) {
    //login berhasil
    $_SESSION['tipe_user'] = $data['type'];
    $_SESSION['user_id'] = $data['user_id'];
    $_SESSION['my_user_agent'] = md5($_SERVER['HTTP_USER_AGENT']);
    header("Location: admin/index.php");
} else {
    echo "ID User atau password salah!";
}
}
?>

```

```

<form method="post" action="">
ID User: <input type="text" name="user_id"/>
Password: <input type="password" name="password"/>
<input type="submit" value="Login"/>
</form>

```

Cara Menggunakan File cek-akses.php

Untuk menggunakan cek-akses.php sangat sederhana seperti

```

defined('__NOT_DIRECT') || define('__NOT_DIRECT',1);
include 'cek-akses.php';

```

Sekarang buatlah folder admin dan file admin/index.php, admin/add-user.php dan admin/add-artikel.php. Masing masing file ditambahkan kode di bawah di bagian paling atas.

```

<?php

```

```
defined('__NOT_DIRECT') || define('__NOT_DIRECT',1);  
include '../cek-akses.php';  
?>
```

Uji Coba

Gunakanlah sql berikut untuk menambahkan user di tabel user

```
INSERT INTO user (user_id, name, password, type) VALUES ('admin1', 'Admin',  
md5('admin'), 'admin');  
INSERT INTO user (user_id, name, password, type) VALUES ('test', 'Test',  
md5('123456'), 'user');
```

#### **i. Pengenalan JQuery EasyUI**

##### **DEFINISI JQUERY**

jQuery adalah library JavaScript multiplatform yang dirancang untuk memudahkan penyusunan client-side script pada file HTML.

Digunakan oleh 60 persen dari 10000 situs web paling banyak dikunjungi di dunia, jQuery adalah library JavaScript yang paling populer saat ini. jQuery merupakan sebuah perangkat lunak bebas sumber terbuka yang berada di bawah lisensi MIT.

Sintaks jQuery dirancang sedemikian rupa untuk memudahkan pengembang website dalam menavigasi dokumen, menyeleksi elemen-elemen DOM, menerapkan animasi, mengaplikasikan events, serta membangun aplikasi AJAX.

jQuery juga memungkinkan developer menciptakan berbagai plugin berbasis library JavaScript. Dengan plugin-plugin tersebut, pengembang situs web mampu menyusun sejumlah abstraksi untuk interaksi dan animasi sederhana, juga beberapa efek yang cukup kompleks dan berbagai widget yang dapat dikonfigurasi.

Karakter library JavaScript yang modular mendukung pengembangan laman web dinamis dengan berbagai fitur dan aplikasi berbasis web (web app).



Fitur-fitur inti jQuery – meliputi penyeleksian, traversal, dan manipulasi elemen-elemen DOM – dimampukan oleh sebuah selector engine (dinamai Sizzle mulai versi 1.3), telah mewujudkan suatu gaya pemrograman baru yang memadukan antara algoritma dan struktur data DOM. Gaya ini telah mempengaruhi arsitektur dari framework JavaScript lainnya seperti YUI v3 dan Dojo, dan di kemudian menstimulasi pengembangan Selectors API standar.

Microsoft dan Nokia membundel jQuery pada platform mereka. Microsoft memasukkannya dalam Visual Studio untuk digunakan dalam framework ASP.NET AJAX dan ASP.NET MVC, sedangkan Nokia mengintegrasikannya dalam platform pengembangan widget Web Run-Time. jQuery juga mulai dipakai pada MediaWiki sejak versi 1.16.

Situs resmi jQuery beralamat di [jquery.com](http://jquery.com). Pengembangannya dikelola oleh jQuery Foundation yang berada di [jquery.org](http://jquery.org). Library ini bisa diunduh di laman [jquery.com/download](http://jquery.com/download), atau diterapkan melalui sejumlah CDN termasuk Google jQuery CDN.

## MEMBEDAH JQUERY

jQuery, pada intinya, adalah sebuah library yang berfungsi untuk memanipulasi DOM. DOM merupakan representasi struktural dari seluruh elemen pada sebuah laman web.

Keberadaan jQuery menjadikan kerja pencarian, penyeleksian, dan manipulasi elemen-elemen DOM menjadi simpel dan mudah. Sebagai contoh, jQuery bisa dipergunakan untuk menemukan sebuah elemen dalam dokumen yang memiliki properti tertentu (misalnya: elemen dengan tag `h1`), kemudian mengubah satu atau beberapa atributnya (misalnya: warna, visibilitas), atau menjadikan elemen tersebut responsif terhadap suatu event (misalnya: klik mouse).

Selain penyeleksian dan manipulasi DOM dasar, jQuery menyajikan sebuah paradigma baru pada penanganan event oleh JavaScript. Penugasan event dan pendefinisian fungsi event callback dapat dilakukan dengan satu langkah dalam satu lokasi di dalam kode.

jQuery juga dikembangkan untuk mendayagunakan berbagai fungsionalitas JavaScript yang paling sering dipakai (misalnya: fade in atau fade out ketika menentukan visibilitas elemen, juga bermacam animasi yang dimunculkan dengan memanipulasi properti-properti CSS).

Keuntungan dari pemanfaatan jQuery antara lain:

Memisahkan JavaScript dan HTML → Tanpa menggunakan atribut-atribut HTML untuk memanggil fungsi Javascript dalam penanganan event, jQuery bisa dipergunakan untuk menangani event dengan script JS saja.

Singkat dan Jelas → jQuery mengutamakan penulisan kode yang singkat dan jelas melalui berbagi fitur seperti fungsi-fungsi yang dapat dirangkaikan (chain-able) dan nama-nama fungsi yang pendek.

Mengatasi masalah kompatibilitas antar-browser → JavaScript engine pada berbagai browser memiliki perbedaan satu sama lain, sehingga script yang berjalan pada suatu browser bisa gagal pada browser lainnya. jQuery mengatasi segala inkonsistensi antar-browser tersebut dan menyajikan antarmuka yang konsisten bekerja pada semua browser.

Ekstensibel → jQuery menjadikan pengembangan framework sangat simpel. Berbagai event, elemen, dan metode baru dapat dengan mudah ditambahkan dan digunakan ulang sebagai plugin.

## FITUR-FITUR JQUERY

jQuery memiliki fitur-fitur sebagai berikut:

Penyeleksian elemen-elemen DOM menggunakan selector engine Sizzle. Sizzle merupakan perangkat lunak multiplatform sumber terbuka yang berawal sebagai proyek sampingan jQuery.

Manipulasi DOM berdasarkan selektor CSS yang memanfaatkan nama-nama dan atribut elemen, misalnya id dan class, sebagai kriteria seleksi simpul-simpul DOM.

Events

Efek dan animasi

AJAX

Obyek-obyek deferred and promise untuk mengontrol pemrosesan asinkron

Penguraian JSON

Ekstensibilitas melalui plugin

Sejumlah utilitas, misalnya: informasi agen pengguna (user agent), deteksi fitur perangkat Metode-metode kompatibilitas yang tersedia secara bawaan pada browser-browser modern, namun membutuhkan pencadangan pada browser yang lebih tua, misalnya ``inArray()`` dan ``each``

Dukungan multi-browser

## BROWSER YANG DIDUKUNG JQUERY

Baik versi 1.x maupun 2.x dari jQuery, keduanya mendukung “current-1 versions” (artinya versi stable terkini dari sebuah browser dan satu versi yang mendahuluinya) untuk Firefox, Google Chrome, Safari, dan Opera.

Versi 1.x juga mendukung IE 6 atau di atasnya. Sedangkan versi 2.x tidak lagi mendukung Internet Explorer 6 sampai 8 (yang mewakili kurang dari 28% dari seluruh browser yang digunakan) dan hanya mendukung IE 9 dan selanjutnya.

## PENGUNAAN JQUERY

### MENAUTKAN LIBRARY

Library jQuery adalah sebuah file JavaScript yang memuat seluruh fungsi-fungsi umum DOM, event, efek, dan Ajax. File ini bisa ditautkan dalam sebuah laman web ke salinan

pada server lokal atau pada salinan yang disajikan melalui server publik (CDN) antara lain:

MaxCDN

Google

Microsoft ASP.NET

cdnJs

jsDelivr

Untuk menautkan file jQuery pada server lokal, gunakan kode berikut:

```
<script src="jquery.js"></script>
```

Untuk menggunakan CDN, sisipkan kode berikut:

```
<script src="//code.jquery.com/jquery-2.1.1.min.js"></script>
```

## GAYA PENGGUNAAN

jQuery memiliki dua gaya penggunaan:

Via fungsi `$` → adalah factory method untuk obyek jQuery. Fungsi-fungsi semacam ini, kerap disebut sebagai commands (perintah), dapat dirangkaikan (chainable) sebab masing-masing menyajikan obyek.

Via fungsi `$.-prefiks` → fungsi-fungsi utilitas yang tidak dijalankan secara langsung pada obyek.

Pada penggunaan tipikal, akses dan manipulasi atas simpul-simpul DOM diawali dengan pemanggilan fungsi `$` menggunakan string selektor CSS. Metode ini menghasilkan sebuah obyek jQuery yang merujuk pada elemen-elemen HTML yang sesuai.

Sebagai contoh, `$("div.namaclass")` akan menghasilkan obyek dengan tipe “div” dan class “namaclass”. Simpul ini dapat dimanipulasi dengan menyambungkan satu atau beberapa fungsi jQuery di belakangnya.

## MODE NO-CONFLICT

jQuery juga memuat mode `.noConflict()` yang melepaskan kontrol dari `$`. Fitur ini sangat berguna jika jQuery digunakan bersama dengan library lainnya yang juga memakai penanda `$`.

Pada mode no-conflict, kita bisa menggunakan jQuery sebagai pengganti dari `$` tanpa kehilangan fungsionalitasnya.

#### **j. Membuat Report PDF dan Excel pada Pemograman Web**

Library FPDF Untuk Membuat Laporan PDF

salah satu library yang paling banyak digunakan oleh web developer adalah FPDF, library ini punya banyak fitur seperti bisa membuat posisi text dan gambar berdasarkan koordinat, mengganti font yang akan digunakan, menambahkan garis dan masih banyak lagi.

sekarang kita akan belajar cara membuat laporan PDF dengan PHP menggunakan library FPDF ini, silahkan download library nya pada situs resmi FPDF. kemudian silahkan buat sebuah folder baru dengan nama `phpfpdf` pada `C:\xampp\htdocs\phpfpdf` dan extract file `fpdf181.zip` ke folder tersebut.

Membuat Script Laporan PDF Dengan PHP Dan MySQL

sekarang kita akan membuat sebuah file baru dengan nama `index.php`, file ini nanti yang bertugas menampilkan laporan pdf pada browser, setelah membuat file `index.php` lalu tulislah script berikut :

```
<?php
// memanggil library FPDF
require('fpdf.php');
// intance object dan memberikan pengaturan halaman PDF
$pdf = new FPDF('P','mm','A4');
```

```

// membuat halaman baru
$pdf->AddPage();
// setting jenis font yang akan digunakan
$pdf->SetFont('Arial','B',16);
// mencetak string
$pdf->Cell(40,10,'Belajar Membuat Laporan PDF Dengan FPDF');
$pdf->Output();
?>
1
2
3
4
5
6
7
8
9
10
11
12
13
<?php
// memanggil library FPDF
require('fpdf.php');
// intance object dan memberikan pengaturan halaman PDF
$pdf = new FPDF('P','mm','A4');
// membuat halaman baru
$pdf->AddPage();
// setting jenis font yang akan digunakan
$pdf->SetFont('Arial','B',16);
// mencetak string

```

```
$pdf->Cell(40,10,'Belajar Membuat Laporan PDF Dengan FPDF');  
$pdf->Output();  
?>
```

kemudian silahkan save file index.php tadi dan sekarang silahkan akses <http://localhost/phpfpdf/> menggunakan web browser anda, jika script yang anda tulis benar maka akan tampil seperti berikut ini

### Membuat Laporan PDF Dengan PHP MySQL

Sekarang kita akan membuat laporan PDF Dengan PHP Dan Data nya dari database MySQL, pertama silahkan buat sebuah database baru dengan nama tutorial dan buatlah sebuah tabel dengan struktur sebagai berikut :

Langkah selanjutnya silahkan insert beberapa data yang akan ditampilkan nantinya, kemudian silahkan buat sebuah file baru dengan nama koneksi.php untuk menyimpan konfigurasi koneksi ke database MySQL, silahkan tulis script berikut :

```
<?php  
$host    = "localhost";  
$user    = "root";  
$password = "";  
$database = "tutorial";  
$connect = mysqli_connect($host, $user, $password, $database);  
?>  
  
1  
2  
3  
4  
5
```

6

7

```
<?php
```

```
$host    = "localhost";
```

```
$user    = "root";
```

```
$password = "";
```

```
$database = "tutorial";
```

```
$connect = mysqli_connect($host, $user, $password, $database);
```

```
?>
```

lalu silahkan save untuk menyimpan perubahan, kemudian silahkan modifikasi script index.php menjadi seperti dibawah ini :

```
<?php
```

```
// memanggil library FPDF
```

```
require('fpdf.php');
```

```
// intance object dan memberikan pengaturan halaman PDF
```

```
$pdf = new FPDF('l','mm','A5');
```

```
// membuat halaman baru
```

```
$pdf->AddPage();
```

```
// setting jenis font yang akan digunakan
```

```
$pdf->SetFont('Arial','B',16);
```

```
// mencetak string
```

```
$pdf->Cell(190,7,'SEKOLAH      MENENGAH      KEJURUSAN      NEEGRI      2  
LANGSA',0,1,'C');
```

```
$pdf->SetFont('Arial','B',12);
```

```
$pdf->Cell(190,7,'DAFTAR      SISWA      KELAS      IX      JURUSAN      REKAYASA  
PERANGKAT LUNAK',0,1,'C');
```

```
// Memberikan space kebawah agar tidak terlalu rapat
```

```
$pdf->Cell(10,7,"",0,1);
```



```

$pdf->SetFont('Arial','B',10);
$pdf->Cell(20,6,'NIM',1,0);
$pdf->Cell(85,6,'NAMA MAHASISWA',1,0);
$pdf->Cell(27,6,'NO HP',1,0);
$pdf->Cell(25,6,'TANGGAL LHR',1,1);

$pdf->SetFont('Arial','',10);

include 'koneksi.php';
$mahasiswa = mysqli_query($connect, "select * from mahasiswa");
while ($row = mysqli_fetch_array($mahasiswa)){
    $pdf->Cell(20,6,$row['nim'],1,0);
    $pdf->Cell(85,6,$row['nama_lengkap'],1,0);
    $pdf->Cell(27,6,$row['no_hp'],1,0);
    $pdf->Cell(25,6,$row['tanggal_lahir'],1,1);
}

$pdf->Output();
?>
1
2
3
4
5
6
7
8
9
10
11
12

```

13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36

```
<?php
```

```
// memanggil library FPDF
```

```
require('fpdf.php');
```

```
// intance object dan memberikan pengaturan halaman PDF
```

```
$pdf = new FPDF('l','mm','A5');
```

```
// membuat halaman baru
```

```
$pdf->AddPage();
```

```

// setting jenis font yang akan digunakan
$pdf->SetFont('Arial','B',16);
// mencetak string
$pdf->Cell(190,7,'SEKOLAH      MENENGAH      KEJURUSAN      NEEGRI      2
LANGSA',0,1,'C');
$pdf->SetFont('Arial','B',12);
$pdf->Cell(190,7,'DAFTAR      SISWA      KELAS      IX      JURUSAN      REKAYASA
PERANGKAT LUNAK',0,1,'C');

// Memberikan space kebawah agar tidak terlalu rapat
$pdf->Cell(10,7,"",0,1);

$pdf->SetFont('Arial','B',10);
$pdf->Cell(20,6,'NIM',1,0);
$pdf->Cell(85,6,'NAMA MAHASISWA',1,0);
$pdf->Cell(27,6,'NO HP',1,0);
$pdf->Cell(25,6,'TANGGAL LHR',1,1);

$pdf->SetFont('Arial','',10);

include 'koneksi.php';
$mahasiswa = mysqli_query($connect, "select * from mahasiswa");
while ($row = mysqli_fetch_array($mahasiswa)){
    $pdf->Cell(20,6,$row['nim'],1,0);
    $pdf->Cell(85,6,$row['nama_lengkap'],1,0);
    $pdf->Cell(27,6,$row['no_hp'],1,0);
    $pdf->Cell(25,6,$row['tanggal_lahir'],1,1);
}

$pdf->Output();
?>

```

Jangan lupa melakukan save untuk menyimpan perubahan, kemudian silahkan akses lagi <http://localhost/phpfpdf/>

### Class untuk Membuat dan Menulis ke Excel

Untuk membuat maupun menulis file excel sebenarnya tidak terlalu sulit, karena sudah cukup banyak tersedia library atau class yang dibuat khusus untuk menangani membuat, menulis bahkan membaca file excel dan spreadsheet. Beberapa library tersebut akan diulas secara singkat di bagian akhir tutorial ini. Dalam tutorial ini, kita akan menggunakan satu class untuk membuat dan menulis file excel yang dibuat oleh Shararia Azam dan dipublikasikan di website <http://www.shahariaazam.com>. Class penanganan excel buatannya cukup singkat dan sederhana namun sudah sangat cukup untuk memberikan gambaran bagaimana penanganan file excel dengan menggunakan PHP.

Excel.class.php : Class Sederhana membuat file Excel

```
<?php
/**
 * By Shaharia Azam
 * shaharia.azam@gmail.com
 * http://www.shahariaazam.com
 */
class Excel{

    /**
     * set the header configuration
     * @param $filename the xls file name
     */
    function setHeader($filename)
    {
        header("Pragma: public");
        header("Expires: 0");
        header("Cache-Control: must-revalidate, post-check=0, pre-check=0");
```

```

header("Content-Type: application/force-download");
header("Content-Type: application/octet-stream");
header("Content-Type: application/download");;
header("Content-Disposition: attachment;filename=$filename");
header("Content-Transfer-Encoding: binary ");
}

```

```

/**
 * write the xls begin of file
 */
function BOF() {
    echo pack("sssss", 0x809, 0x8, 0x0, 0x10, 0x0, 0x0);
    return;
}

```

```

/**
 * write the xls end of file
 */
function EOF() {
    echo pack("ss", 0x0A, 0x00);
    return;
}

```

```

/**
 * write a number
 * @param $Row row to write $Value (first row is 0)
 * @param $Col column to write $Value (first column is 0)
 * @param $Value number value
 */
function writeNumber($Row, $Col, $Value) {
    echo pack("sssss", 0x203, 14, $Row, $Col, 0x0);
}

```

```

        echo pack("d", $Value);
        return;
    }

/**
 * write a string label
 * @param $Row row to write $Value (first row is 0)
 * @param $Col column to write $Value (first column is 0)
 * @param $Value string value
 */
function writeLabel($Row, $Col, $Value) {
    $L = strlen($Value);
    echo pack("sssss", 0x204, 8 + $L, $Row, $Col, 0x0, $L);
    echo $Value;
    return;
}
}

```

Pada class sederhana di atas terdapat beberapa fungsi. Fungsi pertama adalah fungsi `setHeader()` di baris 13-23 yang berperan untuk mengatur header file excel, termasuk nama file excel. Parameter dari fungsi ini adalah nama file excel. Setelah fungsi `setHeader()`, terdapat fungsi `BOF()` dan `EOF()`, masing-masing pada baris 28-31 dan 36-39. Kedua fungsi tersebut akan sebagai penanda awal dan akhir file excel. Di dalamnya terdapat fungsi `pack()` yang merupakan fungsi PHP untuk menghasilkan nilai biner berdasarkan format tertentu. Format fungsi `pack` yang digunakan dalam class diatas antara lain “s” yang berarti signed-short dan “d” yang berarti double. Selanjutnya untuk mengisikan data ke dalam cell di excel, dibuatlah fungsi `writeNumber()` dan `writeLabel()`. Fungsi `writeNumber()` digunakan untuk menulis data bertipe numerik (angka) dan fungsi `writeLabel()` untuk menulis data bertipe string (text). Kedua fungsi tersebut memiliki tiga parameter yaitu `$Row` yang menunjukkan posisi baris cell, `$Col` yang menunjukkan

posisi kolom cell dan \$Value yang merupakan nilai nilai yang akan ditulis. Perlu diingat juga bahwa urutan baris dan kolom diawali dengan 0.

#### Membuat dan Menulis ke Excel

Program PHP berikut ini akan memberikan contoh sederhana dari penggunaan class Excel sederhana di atas. Program akan membuat file excel dengan nama “contoh-1.xls”. Pada cell yang pertama (Row=0, Col=0) diisi dengan text “NIM”, pada cell kedua (Row=0, Col=1) diisi dengan text “1111500200” dan seterusnya.

contoh-1.php : Contoh sederhana pembuatan file excel

```
<?php
require_once "Excel.class.php";

$excel = new Excel();
// Send Header
$excel->setHeader('contoh-1.xls');
$excel->BOF();

$excel->writeLabel(0, 0, "NIM :");
$excel->writeLabel(0, 1, "1111500200");
$excel->writeLabel(1, 0, "NAMA :");
$excel->writeLabel(1, 1, "ACHMAD SOLICHIN");
$excel->writeLabel(2, 0, "NILAI :");
$excel->writeNumber(2, 1, 85);

$excel->EOF();
exit();
?>
```

Jika program di atas dijalankan maka akan terbuat sebuah file excel dengan nama “contoh-1.xls” yang isinya kurang lebih sebagai berikut (dibuka dengan LibreOffice 3) :

File Excel Hasil Contoh Program 1

File Excel Hasil Contoh Program 1

Hasil dari program pertama di atas dapat dilihat secara langsung disini.

Membuat Laporan Excel dari Database MySQL

Untuk keperluan contoh kali ini, seperti biasanya, kita akan menggunakan tabel sederhana berisi data mahasiswa (nama tabel=mahasiswa) dengan struktur sebagai berikut:

FIELD	TYPE	LENGTH	DESCRIPTION
-------	------	--------	-------------

nim			
-----	--	--	--

	VARCHAR		
--	---------	--	--

		10	
--	--	----	--

			NIM Mahasiswa
--	--	--	---------------

nama			
------	--	--	--

	VARCHAR		
--	---------	--	--

		30	
--	--	----	--

			NAMA Mahasiswa
--	--	--	----------------

alamat			
--------	--	--	--



## TEXT

—

### Alamat Mahasiswa

Sebelum mencoba program di bawah ini, pastikan database dan tabel di atas sudah dibuat di MySQL. Jangan lupa isikan beberapa contoh data ke dalam tabel mahasiswa tersebut.

Sementara itu, untuk menangani koneksi dan pengambilan data dari database MySQL, kita akan menggunakan fungsi mysqli yang merupakan fungsi mysql yang telah ditingkatkan kemampuan dan performanya. Kita akan menggunakan gaya penulisan mysqli dengan gaya object oriented. Saat ini, fungsi ini yang sangat disarankan untuk digunakan, dibanding menggunakan fungsi mysql\_\*.

Berikut ini contoh program kedua untuk membuat laporan excel dari tabel mahasiswa yang sudah dibuat sebelumnya.

contoh-2.php : Contoh Membuat Laporan Excel dari Database MySQL

```
<?php
require_once "Excel.class.php";

#koneksi ke mysql
$mysqli = new mysqli("localhost","root","qwerty","demo");
if ($mysqli->connect_error) {
    die('Connect Error ( ' . $mysqli->connect_error . ' )');
}
#akhir koneksi
```

```

#ambil data
$query = "SELECT nim, nama, alamat FROM mahasiswa";
$sql = $mysqli->query($query);
$arrmhs = array();
while ($row = $sql->fetch_assoc()) {
    array_push($arrmhs, $row);
}
#akhir data

$excel = new Excel();
#Send Header
$excel->setHeader('contoh-2.xls');
$excel->BOF();

#header tabel
$excel->writeLabel(0, 0, "NIM");
$excel->writeLabel(0, 1, "NAMA");
$excel->writeLabel(0, 2, "ALAMAT");

#isi data
$i = 1;
foreach ($arrmhs as $baris) {
    $j = 0;
    foreach ($baris as $value) {
        $excel->writeLabel($i, $j, $value);
        $j++;
    }
    $i++;
}

```

```
$excel->EOF();
```

```
exit();
```

```
?>
```

Berikut penjelasan singkat dari program PHP di atas:

Baris 2. Menyertakan class Excel yang disimpan dalam file Excel.class.php menggunakan fungsi `required_once`.

Baris 5. Membuat object `mysqli` dari class `mysqli` sekaligus melakukan koneksi ke server MySQL dengan mengirimkan parameter `namahost`, `user mysql`, `password mysql` dan nama database yang digunakan.

Baris 6-8. Memeriksa sekaligus menampilkan pesan error jika ternyata koneksi ke MySQL gagal dilakukan. Pesan error tersimpan dalam variabel `mysqli_error` dari class `mysqli`.

Baris 11-18. Pengambilan data dari tabel mahasiswa dan dikumpulkan ke dalam variabel array `$armhs`. Fungsi `array_push()` digunakan untuk memasukkan (push) setiap baris hasil query `$row` ke dalam array `$armhs`.

Baris 22. Memanggil fungsi `setHeader()` dan memberikan nama file excel.

Baris 25-28. Membuat header tabel di excel yang diletakkan di baris pertama.

Baris 30-39. Menuliskan data yang sudah tersimpan di array `$armhs` ke dalam cell di excel dengan menggunakan perulangan `foreach()`.

Hasil dari program kedua di atas dapat dilihat secara langsung disini dimana kurang lebih hasil file excelnya sebagai berikut:

Tampilan File Excel Hasil Program 2

Tampilan File Excel Hasil Program 2

Berbagai Library Penanganan Excel Spreadsheet

Selain dengan class sederhana yang sudah dibahas di atas, terutama untuk keperluan yang lebih kompleks, kita dapat menggunakan berbagai library PHP. Library tersebut dapat ditemukan dengan mudah di internet. Berikut ini beberapa library beserta review singkatnya:

PHP Excel (<http://phpexcel.codeplex.com>)

PHPExcel ini merupakan salah satu project open source yang didanai oleh Microsoft. PHPEXcel ini cukup tangguh dalam melakukan baca dan tulis ke file excel baik dalam format .xls maupun .xlsx. Pada situs resminya, disebutkan bahwa PHPEXcel mendukung tipe file Excel (BIFF) .xls, Excel 2007 (OfficeOpenXML) .xlsx, CSV, Libre/OpenOffice Calc .ods, Gnumeric, PDF dan HTML.

PEAR Excel Spreadsheet Writer (<http://pear.php.net/>)

Excel Spreadsheet Writer merupakan salah satu package dari PEAR (PHP Extension and Application Repository), sebuah kumpulan ekstensi dan library PHP. Package Excel Spreadsheet Writer sendiri sebenarnya memiliki kemampuan cukup lengkap dalam hal penulisan file Excel, namun sayangnya saat ini sudah tidak dikembangkan lagi. Library ini tidak bisa untuk membaca file excel.

php-excel (<https://code.google.com/p/php-excel/>)

Library ini merupakan library sederhana yang dapat digunakan untuk meng-ekspor data dari database ke file excel secara cepat. Tidak banyak fitur lain yang ditawarkan oleh library ini, namun tidak ada salahnya Anda coba.

php-excel-reader (<https://code.google.com/p/php-excel-reader/>)

Dengan library PHP ini kita dapat membaca isi file excel (.xls) beserta format-formatnya, kemudian menampilkannya menjadi format HTML. Format dan isinya dapat dibaca dengan baik oleh library ini, sehingga sangat cocok jika kita ingin membuat tampilan (preview) dari file excel yang sudah ada.

### **Daftar Pustaka**

<http://achmatim.net/2013/02/07/membuat-laporan-excel-dengan-php/>

<https://belajarphp.net/laporan-pdf-php-mysql/>

<https://ekajogja.com/arsip/jquery-definisi-fitur-penggunaan-jquery-dalam-pengembangan-website/>

<https://www.myphptutorials.com/tutorials/674/sistem-login-dengan-tipe-user-hak-akses-berbeda>