



## **TECHBOOST WORKSHOP**

***Peningkatan Kemampuan Pelajar Melalui  
Pengembangan Sistem Teknologi***



**Disusun Oleh :**

1. *Fabio Junior, S.Kom., M.Kom. (Dosen Universitas Tangerang Raya)*
2. *Hendra Kurniawan*
3. *Dany Kurniawan*
4. *Zidane Baihaqi*

**HIMPUNAN MAHASISWA TEKNOLOGI INFORMASI  
UNIVERSITAS TANGERANG RAYA**

Di tahap ini kita akan memulai membuat sebuah project aplikasi berbasis web yang akan kita buat dengan menggunakan CodeIgniter. Akan dijelaskan secara berurutan secara bertahap. Teman-teman bisa mengikuti urutan yang harus dilakukan. Yang dimulai dari instalasi codeigniter 3 untuk memulai pembuatan aplikasi rental mobil berbasis web

Dengan mengucap bismillah langsung saja kita mulai dari tahap instalasi codeigniter ke XAMPP.

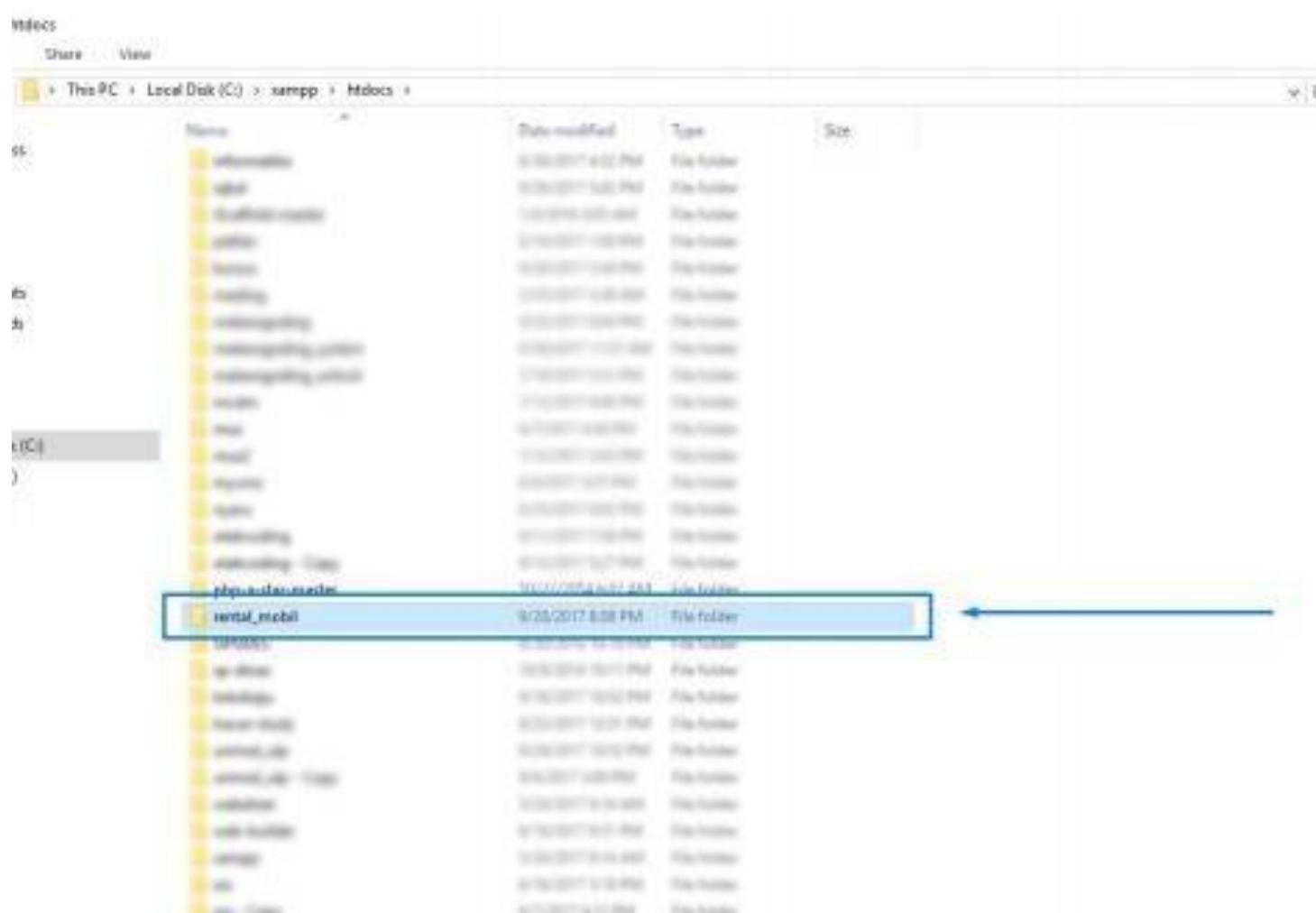
Adapun beberapa aplikasi pendukung yang digunakan adalah sebagai berikut :

1. Text Editor Sublime
2. XAMPP v3.2.4
3. Web Browser Google Chrome

## Instalasi Codeigniter 3

Silahkan teman-teman download codeigniter dari situs resminya (<https://codeigniter.com/>) . codeigniter 3 juga telah di sertakan di dalam ebook ini.

kemudian copy project codeigniter nya ke direktori localhost. ( C:\XAMPP\htdocs ) dan ubah nama folder nya menjadi “rental\_mobil”. Seperti pada gambar berikut.



Kemudian buka web browser dan jalankan.

***http://localhost/rental\_mobil***



Sampai di sini maka instalasi Codeigniter pun selesai.

## **Persiapan Membuat Aplikasi Rental Mobil**

Diperlukan beberapa penyesuaian atau pengaturan pada codeigniter sebelum mulai kita gunakan untuk membuat aplikasi rental mobil. Semua pengaturan atau penyesuaian yang kita lakukan di sini sesuai dengan kebutuhan kita untuk membuat aplikasi rental mobil dengan codeigniter. Di antaranya adalah :

### ***Setting Base URL***

Pengaturan base url adalah pengaturan URL dasar dalam sebuah project aplikasi yang dibuat dengan menggunakan codeigniter. Pengaturan base url di codeigniter terdapat di dalam file config.php di dalam folder project codeigniter.

***rental\_mobil/application/config/config.php***

The screenshot shows a Sublime Text window with the file path: C:\xampp\htdocs\rental\_mobil\application\config\config.php (rental\_mobil) - Sublime Text (UNREGISTERED). The left sidebar shows the project structure with folders like rental\_mobil, application, cache, config, controllers, core, helpers, hooks, language, libraries, logs, models, third\_party, and views. The right pane displays the config.php code:

```
<?php  
defined('BASEPATH') OR exit('No direct script access allowed');  
  
/*  
 * Base Site URL.  
 */  
  
URL to your CodeIgniter root. Typically this will be your base URL,  
WITH a trailing slash:  
http://example.com/  
  
WARNING: You MUST set this value!  
  
If it is not set, then CodeIgniter will try guess the protocol and path  
of your installation, but due to security concerns the hostname will be set  
to $_SERVER['SERVER_NAME'] if available, or localhost otherwise.  
The auto-detection mechanism exists only for convenience during  
development and MUST NOT be used in production!  
  
If you need to allow multiple domains, remember that this file is still  
a PHP script and you can easily do that on your own.  
  
$config['base_url'] = 'http://localhost/rental_mobil/';  
  
Index File  
Typically this will be your index.php file, unless you've renamed it to
```

Sesuaikan dengan nama directory project kita. Dimana tadi folder project aplikasi kita beri nama “rental\_mobil”. Maka sesuaikan pengaturan base\_url nya menjadi.

```
$config['base_url'] = 'http://localhost/rental_mobil/' ;
```

## INGAT !

Base url adalah fitur dari codeigniter untuk menetapkan url dasar. Dan base\_url akan sering kita gunakan dalam pengembangan aplikasi menggunakan codeigniter.

# **Load Library, Helper, dan segala yang dibutuhkan**

Helper dan Library bisa di bilang seperti kita memiliki perpusataan coding dengan fungsi-fungsi yang berbeda dan memiliki kegunaan masing-masing. Kita tinggal memanggil dan menggunakannya saja. Kita juga bisa membuat helper atau library kita sendiri di codeigniter untuk memudahkan pekerjaan kita.

Di tahap ini kita akan men-load atau membuka beberapa library dan helper codeigniter yang kita butuhkan dalam pembuatan aplikasi rental mobil ini. Untuk menjalankan atau membuka library dan helper secara otomatis dicodeiginter, kita bisa mengaturnya dalam file autoload.php dalam folder application/config/autoload.php

## ***Rental\_mobil/application/config/autoload.php***

Library yang akan kita load secara otomatis adalah *database,session* dan *form\_validation* .

### **Cari syntax**

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

Kemudian tambahkan database, session dan form\_validation. Sehingga menjadi seperti berikut.

```
$autoload['libraries']= array('database','session','form','validation');
```

Dan helper yang akan kita load secara otomatis adalah url :

### **Cari syntax**

```
$autoload['helper'] = array('');
```

Kemudian tambahkan “url” (karena kita hanya membutuhkan helper url disini). Maka akan menjadi seperti berikut

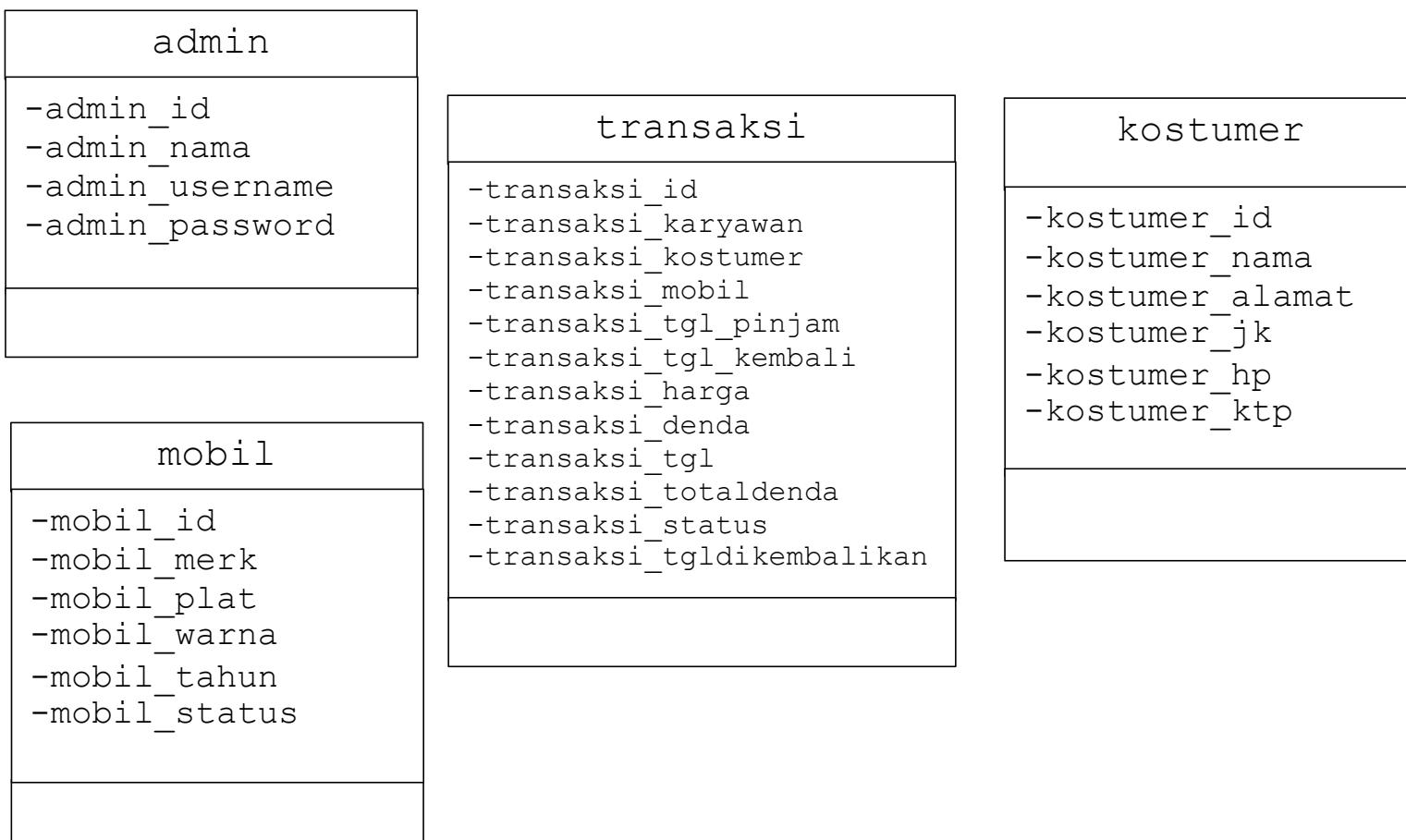
```
$autoload['helper'] = array('url');
```

Proses autoload library dan helper selesai . . .

# Merancang & Membuat Database Aplikasi Rental Mobil

## dan Konfigurasi database di Codeigniter

Berikut adalah gambaran struktur database yang akan kita buat untuk aplikasi rental mobil ini



Buat sebuah database baru dengan nama “**rental\_mobil**”.

Ada 4 buah tabel yang akan kita buat dalam database. Yaitu :

- admin
- mobil
- transaksi
- kostumer

## Tabel admin

tabel admin akan kita jadikan sebagai tempat untuk menyimpan data admin seperti nama, username dan password

Nama	Tipe	Auto_increment	Primary Key
admin_id	Int(11)	✓	✓
admin_nama	Varchar (255)		
admin_username	Varchar (255)		
admin_password	Varchar (255)		

## Tabel mobil

tabel mobil akan kita jadikan sebagai tempat untuk menyimpan data mobil seperti merk mobil , no.plat, warna, tahun dan status ketersediaan mobil

Nama	Tipe	Auto_increment	Primary Key
mobil_id	Int(11)	✓	✓
mobil_merk	Varchar (30)		
mobil_plat	Varchar (20)		
mobil_warna	Varchar (30)		
mobil_tahun	Int(11)		
mobil_status	Int(11)		

## Tabel kostumer

Tabel kostumer akan kita jadikan sebagai tempat menyimpan data kostumer atau data orang yang menyewa mobil. seperti nama, alamat, jenis kelamin, no. hp dan no.ktp.

Nama	Tipe	Auto_increment	Primary Key
kostumer_id	Int(11)	✓	✓
kostumer_nama	Varchar (255)		
kostumer_alamat	Text		
kostumer_jk	Varchar (10)		
kostumer_hp	Varchar (20)		
kostumer_ktp	Varchar (50)		

## Tabel transaksi

Tabel transaksi kita gunakan untuk menyimpan data transaksi, seperti karyawan/admin yg melakukan atau melayani kostumer saat melakukan transaksi, kostumer, mobil yang di sewa, tanggal mulai di rental, tanggal batas pengembalian, harga rental, harga denda per hari, tanggal transaksi, total denda jika terlambat di kembalikan dan tanggal saat mobil di kembalikan oleh kostumer.

Nama	Tipe	Auto_increment	Primary Key
transaksi_id	Int(11)	✓	✓
transaksi_karyawan	Int(11)		
transaksi_kostumer	Int(11)		
transaksi_mobil	Int(11)		
transaksi_tgl_pinjam	date		
transaksi_tgl_kembali	date		
transaksi_harga	Int(11)		
transaksi_denda	Int(11)		
transaksi_tgl	date		
transaksi_totaldenda	Int(11)		
transaksi_status	Int(11)		
transaksi_tgldikembalikan	date		

## Konfigurasi database pada codeigniter

Untuk melakukan konfigurasi database pada codeigniter, silahkan buka file database.php pada application/config/database.php.

Lihat pada gambar berikut:

```
File Edit Selection Find View Dtoz Tools Project Preferences Help

FOLDERS
+-- rental_mobil
+-- application
+-- cache
+-- config
    +-- autoloader.php
    +-- config.php
    +-- constants.php
    +-- database.php
    +-- doctype.php
    +-- foreign_chars.php
    +-- hooks.php
    +-- index.html
    +-- i18n.php
    +-- migration.php
    +-- minify.php
    +-- profiler.php
    +-- routes.php
    +-- smiley.php
    +-- user_agents.php
+-- controllers
+-- core
+-- helpers
+-- hooks
+-- language
+-- libraries
+-- logs
+-- models

database.php
67  * The $active_group variable lets you choose which connection group to
68  * make active. By default there is only one group (the 'default' group).
69  *
70  * $active_group = 'default';
71  */
72 /**
73 * $active_group = 'default';
74 */
75 /**
76 * $db['default'] = array(
77 *     'dsn' => '',
78 *     'hostname' => 'localhost',
79 *     'username' => 'root',
80 *     'password' => '',
81 *     'database' => 'rental_mobil',
82 *     'dbdriver' => 'mysqli',
83 *     'dbprefix' => '',
84 *     'pconnect' => FALSE,
85 *     'db_debug' => (ENVIRONMENT === 'production'),
86 *     'cache_on' => FALSE,
87 *     'cachedir' => '',
88 *     'char_set' => 'utf8',
89 *     'dbcollat' => 'utf8_general_ci',
90 *     'swap_pre' => '',
91 *     'encrypt' => FALSE,
92 *     'compress' => FALSE,
93 *     'stricton' => FALSE,
94 *     'failover' => array(),
95 *     'save_queries' => TRUE
96 );
97 */
```

# **Membuat Model Yang Terstruktur dan Efisien Untuk CRUD**

Ini merupakan cara yang biasa saya lakukan setiap membuat aplikasi dengan codeigniter. Saya lebih senang membuat fungsi-fungsi yang bisa saya gunakan berulang kali, tanpa harus repot banyak-banyak membuat fungsi pada model. contohnya untuk insert data ke database, kita buat satu fungsi untuk insert data ke database. Dan bisa kita gunakan secara berulang-ulang kali. Begitujuga dengan fungsi edit, hapus dan update data pada database. Yuk simak caranya

Buat sebuah file model dengan nama M\_rental.php.

Ingat! Untuk penamaan file model di codeigniter, harus di awali dengan huruf besar. Contoh : M\_rental.php

## **Ingat!**

Penamaan Controller dan Model harus di awali dengan huruf besar di codeigniter 3

Memang tidak masalah jika dibuat dengan nama kecil, tetapi akan menyalahi aturan penggunaan codeigniter yang benar dan akan terjadi trouble saat di upload di hosting.

## ***rental\_mobil/application/models/M\_rental.php***

```
<?php

// WWW.AKADEMIKITA.COM === Author : Fabio Junior
// Model yang terstruktur. agar bisa digunakan berulang kali untuk membuat CRUD.
// Sehingga proses pembuatan CRUD menjadi lebih cepat dan efisien.

class M_rental extends CI_Model{

    function edit_data($where,$table) {
        return $this->db->get_where($table,$where);
    }

    function get_data($table) {
        return $this->db->get($table);
    }

    function insert_data($data,$table) {
        $this->db->insert($table,$data);
    }

    function update_data($where,$data,$table) {
        $this->db->where($where);
        $this->db->update($table,$data);
    }

    function delete_data($where,$table) {
        $this->db->where($where);
        $this->db->delete($table);
    }
}
?>
```

## **PENJELASAN !!!**

Dengan model seperti di atas, kita bisa menggunakannya secara berulang-ulang tanpa membuat 1 function model untuk 1 function pengelolaandatabase. Untuk menginput data ke database kita bisa menggunakan function insert\_data(), untuk menampilkan data atau mengambil data dari database kita bisa menggunakan function get\_data(). Untuk mengambil data tertentu bisa menggunakan edit\_data() , untuk menghapus data dari database kita bisa menggunakan function delete\_data(), dan untuk mengupdate data pada database kita bisa menggunakan function update\_data()

Kemudian kita harus mengatur autoload model di application/config/autoload.php. agar model m\_rental di load secara otomatis. Jadi tidak perlu lagi kita load dari controller.

### ***Application/config/autoload.php***

Cari syntax barikut

```
$autoload['model'] = array('');
```

Dan Ubah Menjadi

```
$autoload['model'] = array('m_rental');
```

Nah, sampai disini kita telah selesai melakukan semua autoload library, helper dan model yang kita butuhkan dalam pembuatan aplikasi rental mobil.

Jika teman-teman mengikuti panduan ini dengan benar, maka file ***autoload.php*** nya akan seperti berikut :

```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

/*
| -----
| AUTO-LOADER
|
| This file specifies which systems should be loaded by default.
|
| In order to keep the framework as light-weight as possible only the
| absolute minimal resources are loaded by default. For example,
| the database is not connected to automatically since no assumption
| is made regarding whether you intend to use it. This file lets
| you globally define which systems you would like loaded with every
| request.
|
| -----
| Instructions
|
| These are the things you can load automatically:
|
| 1. Packages
| 2. Libraries
| 3. Drivers
| 4. Helper files
| 5. Custom config files
| 6. Language files
| 7. Models
|
*/
/*
| -----
| Auto-load Packages
| -----
| Prototype:
|
| $autoload['packages'] = array(APPPATH.'third_party',
'usr/local/shared');
|
*/
$autoload['packages'] = array();

/*
| -----
| Auto-load Libraries
| -----
| These are the classes located in system/libraries/ or your
| application/libraries/ directory, with the addition of the
| 'database' library, which is somewhat of a special case.
|
| Prototype:
|
| $autoload['libraries'] = array('database', 'email', 'session');
|
| You can also supply an alternative library name to be assigned
| in the controller:
|
| $autoload['libraries'] = array('user_agent' => 'ua');
*/
$autoload['libraries'] = array('database','session','form_validation');

/*
| -----
| Auto-load Drivers
|
| These classes are located in system/libraries/ or in your
| application/libraries/ directory, but are also placed inside their
| own subdirectory and they extend the CI_Driver_Library class. They
| offer multiple interchangeable driver options.
|
| Prototype:
|
| $autoload['drivers'] = array('cache');
|
| You can also supply an alternative property name to be assigned in
| the controller:
|
| $autoload['drivers'] = array('cache' => 'cch');
*/

```

```

$autoload['drivers'] = array();

/*
| -----
| Auto-load Helper Files
|
| Prototype:
|
| $autoload['helper'] = array('url', 'file');
*/
$autoload['helper'] = array('url');

/*
| -----
| Auto-load Config files
|
| Prototype:
|
| $autoload['config'] = array('config1', 'config2');
|
| NOTE: This item is intended for use ONLY if you have created custom
| config files. Otherwise, leave it blank.
|
*/
$autoload['config'] = array();

/*
| -----
| Auto-load Language files
|
| Prototype:
|
| $autoload['language'] = array('lang1', 'lang2');
|
| NOTE: Do not include the " lang" part of your file. For example
| "codeigniter_lang.php" would be referenced as array('codeigniter');
|
*/
$autoload['language'] = array();

/*
| -----
| Auto-load Models
|
| Prototype:
|
| $autoload['model'] = array('first_model', 'second_model');
|
| You can also supply an alternative model name to be assigned
| in the controller:
|
| $autoload['model'] = array('first_model' => 'first');
*/
$autoload['model'] = array('m_rental');

```

# Menghilangkan index.php dari URL Codeigniter

## *Application/config/config.php*

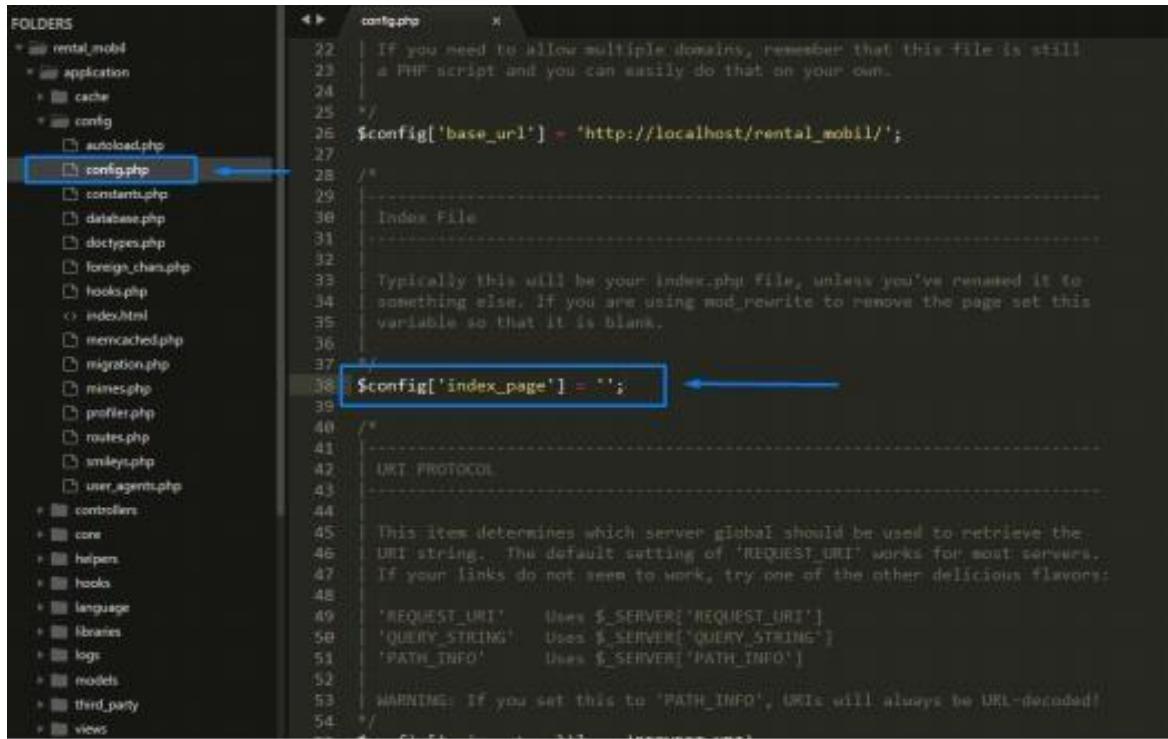
Cari syntax

```
$config['index_page'] = 'index.php';
```

Lalu hapus index.php, sehingga menjadi

```
$config['index_page'] = '';
```

Perhatikan pada gambar berikut



```
FOLDERS
+ rental_mobil
  + application
    + cache
    + config
      - unencoded.php
      - config.php
      - constants.php
      - database.php
      - doctypes.php
      - foreign_chars.php
      - hooks.php
      - index.html
      - memcached.php
      - migration.php
      - mimes.php
      - profiler.php
      - routes.php
      - smileys.php
      - user_agents.php
    + controllers
    + core
    + helpers
    + hooks
    + language
    + libraries
    + logs
    + models
    + third_party
    + views

config.php
22 | If you need to allow multiple domains, remember that this file is still
23 | a PHP script and you can easily do that on your own.
24 |
25 */
26 $config['base_url'] = "http://localhost/rental_mobil/";
27 /**
28 Index File
29
30 Typically this will be your index.php file, unless you've renamed it to
31 something else. If you are using mod_rewrite to remove the page set this
32 variable so that it is blank.
33
34
35
36
37 */
38 $config['index_page'] = '';
39 /**
40 URI PROTOCOL
41
42 This item determines which server global should be used to retrieve the
43 URI string. The default setting of 'REQUEST_URI' works for most servers.
44 If your links do not seem to work, try one of the other delicious flavors:
45
46 'REQUEST_URI'    Uses ${_SERVER['REQUEST_URI']}
47 'QUERY_STRING'   Uses ${_SERVER['QUERY_STRING']}
48 'PATH_INFO'      Uses ${_SERVER['PATH_INFO']}
49
50
51
52
53 WARNING: If you set this to 'PATH_INFO', URIs will always be URL-decoded!
54 */
```

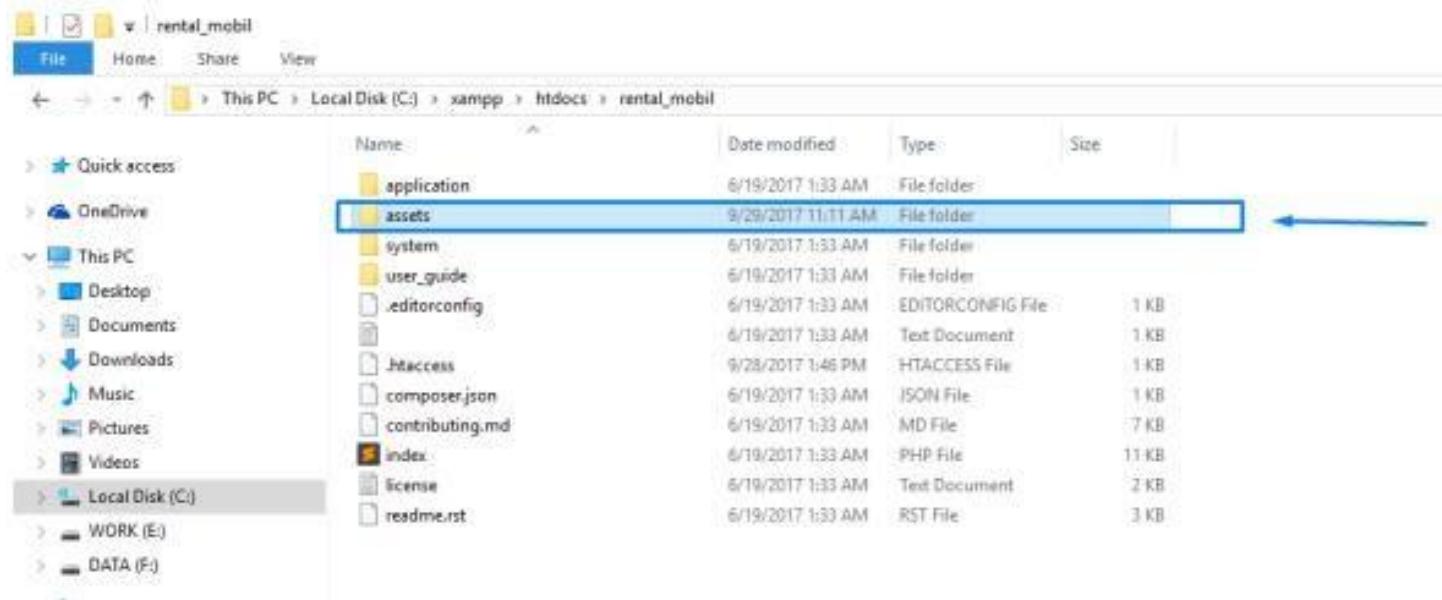
Kedua, buatlah sebuah file .htaccess pada directory root project rental\_mobil.

.htaccess

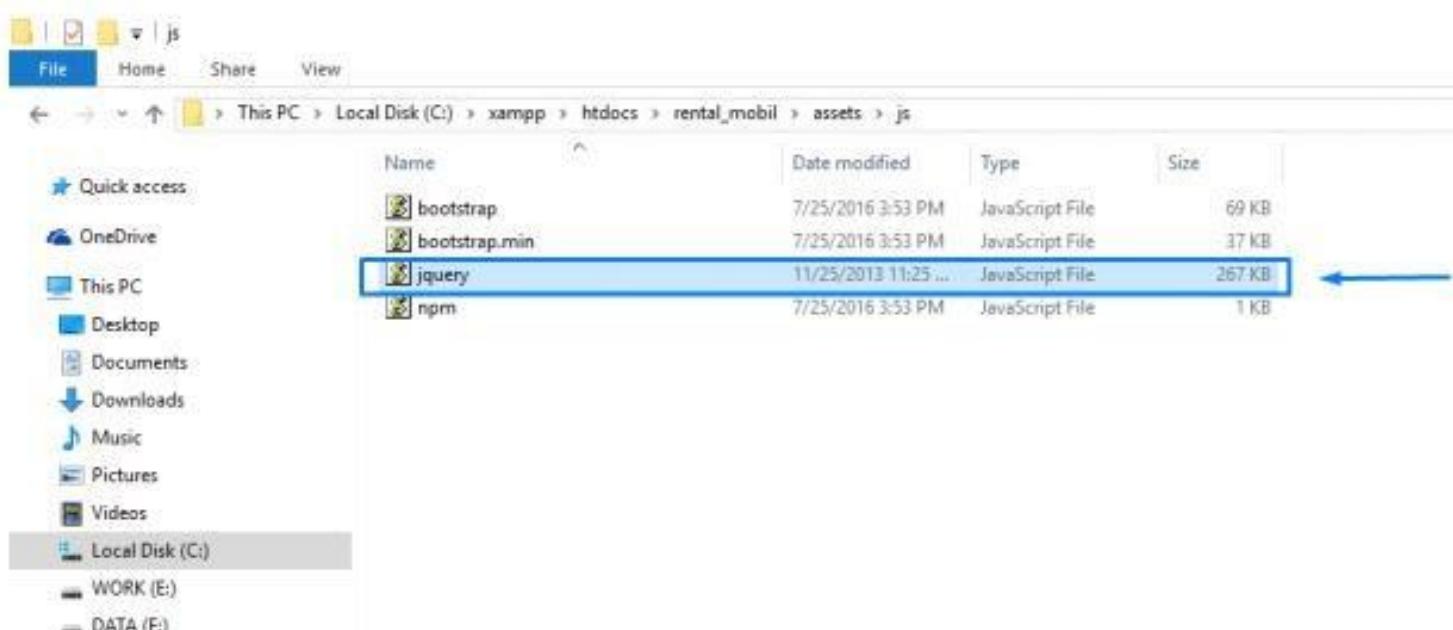
```
RewriteEngine On
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule ^(.*)$ index.php/$1 [L]
```

# Integrasi Codeigniter dan Bootstrap

Teman-teman bisa mendownload bootstrap dari situs resminya di <http://getbootstrap.com/>, atau bisa menggunakan bootstrap yang telah disertakan di dalam folder ebook ini. ekstrak bootstrap ke directory root project rental\_mobil ini. Lalu rename nama folder menjadi “assets”



Download juga file jquery.js dan letakkan dalam folder “assets/js”. File jquery.js juga telah disertakan di dalam folder ebook ini.



## Membuat login dengan codeigniter, md5 & Pesan

### Notifikasi

Setelah segala persiapan selesai, kita akan mulai dengan membuat login untuk admin. Halaman login akan kita buat di halaman utama pada saat aplikasi rental mobil ini di akses. Oh ya, untuk percobaan, kita harus menginput data username dan password admin terlebih dulu di tabel admin. Caranya teman-teman bisa klik tambahkan pada tabel admin. Kemudian isi admin\_nama dengan “nama kamu”, admin\_username dengan “admin”, dan admin\_password dengan “admin123”. Jangan lupa pilih fungsi md5 pada admin\_password. Seperti contoh pada gambar berikut ini

Server: 127.0.0.1 » Database: rental_mobil » Table: admin					
Column	Type	Function	Null	Value	
admin_id	int(11)		▼	1	
admin_nama	varchar(255)		▼	Fabio Junior	
admin_username	varchar(255)		▼	admin	
admin_password	varchar(255)	MD5	▼	admin123	

Kemudian klik kirim untuk menyimpan data. Oke selesai, akun admin sudah kita buat, selanjutnya kita buat sistem loginnya. Login akan kita buat pada controller Welcome.php. karena controller Welcome.php adalah controller yang default di gunakan pada codeigniter. Untuk mengubah default controllernya teman-teman bisa mengubahnya di **application/config/routes.php**. tapi di sini kita tidak akan mengubahnya, kita akan langsung menggunakan controller Welcome.php sebagai controller default. Halaman login pun akan kita buat di controller ini.

Buka Controller welcome.php dan tuliskan syntax berikut.

### **Application/controller/Welcome.php**

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');class Welcome
extends CI_Controller {

    function construct() {
        parent::__construct();
        $this->load->model('m_rental');
    }

    public function index() {
        $this->load->view('login');
    }

    function login() {
        $username = $this->input->post('username');
        $password = $this->input->post('password');

        $this->form_validation->set_rules('username','Username','trim|required');
        $this->form_validation->set_rules('password','Password','trim|required');
        if($this->form_validation->run() != false) {
            $where = array(
                'admin_username' => $username,
                'admin_password' => md5($password)
            );
            $data = $this->m_rental->edit_data($where,'admin');
            $d = $this->m_rental->edit_data($where,'admin')->row();
            $cek = $data->num_rows();
            if($cek > 0) {
                $session = array( 'id'=>
                    $d->admin_id,
                    'nama'=> $d->admin_nama,
                    'status' => 'login'
                );
                $this->session->set_userdata($session);
                redirect(base_url().'admin');
            } else{
                redirect(base_url().'welcome?pesan=gagal');
            }
        } else{
            $this->load->view('login');
        }
    }
}
```

## PENJELASAN !!!

Setelah teman-teman selesai menulis syntax di atas pada controller Welcome.php, akan sedikit saya jelaskan dulu, teman-teman pasti sudah tau apa itu function construct() jika teman-teman pernah belajar OOP, function construct seperti contoh di atas adalah function yang pertama kali akan di jalankan pada saat sebuah class di jalankan. Setelah function construct() di jalankan, selanjutnya yang di jalankan adalah function index(). Ini merupakan ketetapan dari codeigniter.

Coba teman-teman perhatikan pada function construct() di atas, pada function construct() kita memanggil model ' m\_rental '. kenapa kita perlumemanggil model ini ? karena kita akan berhubungan dengan model m\_rental di dalam controller Welcome ini.

Selanjutnya perhatikan juga pada function index() di atas.

```
public function index(){
    $this->load->view('login');
}
```

Pada function index() kita membuat perintah untuk membuka view login. Syntax \$this->load->view() adalah perintah untuk menampilkan view (tampilan). Di sini kita memerintahkan untuk menampilkan view login (login.php). ekstensi .php nya tidak perlu lagi kita tuliskan saat memanggil view. Cukup seperti perintah di atas.

Pada saat controller welcome di jalankan, otomatis function index juga di jalankan, dan akan menampilkan halaman login (view login.php).

### Ingin !

Function index() adalah function yang pertama kali di jalankan setelah function construct() pada sebuah controller.

Karena pada function index kita ingin menampilkan view login.php, maka sekarang kita buat dulu view login.php nya.

## Application/view/login.php

```
<!DOCTYPE html>
<html>
<head>
    <title>Login - Aplikasi Rental Mobil Berbasis WEB - www.akademikitita.com</title>
    <link rel="stylesheet" type="text/css" href="<?php echo base_url().'assets/css/bootstrap.css' ?>">
        <script type="text/javascript" src="<?php echo base_url().'assets/js/jquery.js'; ?>"></script>
        <script type="text/javascript" src="<?php echo base_url().'assets/js/bootstrap.js'; ?>"></script>
</head>
<body>
    <div class="col-md-4 col-md-offset-4" style="margin-top: 50px">
        <center>
            <h2>APLIKASI RENTAL MOBIL</h2>
            <h3>LOGIN</h3>
        </center>
        <br/>
        <?php
            if(isset($_GET['pesan'])) {
                if($_GET['pesan'] == "gagal") {
                    echo "<div class='alert alert-danger'>Login gagal! Username dan password salah.</div>";
                } else if($_GET['pesan'] == "logout") {
                    echo "<div class='alert alert-danger'>Anda telah logout.</div>";
                } else if($_GET['pesan'] == "belumlogin") {
                    echo "<div class='alert alert-success'>Silahkan login dulu.</div>";
                }
            }
        ?>
        <br/>
        <div class="panel panel-default">
            <div class="panel-body">
                <br/>
                <br/>
                <form method="post" action="<?php echo base_url().'welcome/login' ?>">
                    <div class="form-group">
                        <input type="text" name="username" placeholder="username" class="form-control">
                            <?php echo form_error('username'); ?>
                    </div>
                    <div class="form-group">
                        <input type="password" name="password" placeholder="password" class="form-control">
                            <?php echo form_error('password'); ?>
                    </div>
                    <div class="form-group">
                        <input type="submit" value="Login" class="btn btn-primary">
                    </div>
                </form>
                <br/>
                <br/>
            </div>
        </div>
    </body>
</html>
```

## PENJELASAN !!!

Perhatikan Pada tag <head>..</head> diatas, disinilah kegunaan dari base\_url() yang sudah dijelaskan diatas tadi

```
<title>Login - Aplikasi Rental Mobil Berbasis WEB - www.malasngoding.com</title>
<link rel="stylesheet" type="text/css" href="php echo base_url().'assets/css/bootstrap.css' ?&gt;"&gt;
&lt;script type="text/javascript" src="<?php echo base_url().'assets/js/jquery.js'; ?&gt;"&gt;&lt;/script&gt;
&lt;script type="text/javascript" src="<?php echo base_url().'assets/js/bootstrap.js'; ?&gt;"&gt;&lt;/script&gt;</pre
```

Pada pengaturan config['base\_url'] sebelumnya kita telah menetapkannya dengan http://localhost/rental\_mobil/. Jadi saat kita gunakan seperti contoh menghubungkan template dengan bootstrap seperti di atas, sama seperti

```
href="http://localhost/rental_mobil/assets/css/bootstrap.css' ?>">
```

Karena base\_url() sudah sama dengan http://localhost/rental\_mobil/. Kemudian lagi perhatikan pada form yang sudah kita buat pada view login.php di atas.

```
<form method="post" action="php echo base_url().'welcome/login' ?&gt;"&gt;
    &lt;div class="form-group"&gt;
        &lt;input type="text" name="username" placeholder="username" class="form-control"&gt;
        &lt;?php echo form_error('username'); ?&gt;
    &lt;/div&gt;
    &lt;div class="form-group"&gt;
        &lt;input type="password" name="password" placeholder="password" class="form-control"&gt;
        &lt;?php echo form_error('password'); ?&gt;
    &lt;/div&gt;
    &lt;div class="form-group"&gt;
        &lt;input type="submit" value="Login" class="btn btn-primary"&gt;
    &lt;/div&gt;
&lt;/form&gt;</pre
```

Action dari form ini telah kita atur ke function/method login di controller welcome.

```
<?php echo base_url().'welcome/login' ?>">
```

Oke kembali lagi perhatikan method/function login pada controller welcome yang telah kita buat. Karena function ini yang akan di eksekusi sebagai form handler dari form login kita

## PENJELASAN !!!

```
function login(){
    $username = $this->input->post('username');
    $password = $this->input->post('password');
    $this->form_validation->set_rules('username','Username','trim|required');
    $this->form_validation->set_rules('password','Password','trim|required');
    if($this->form_validation->run() != false){
        $where = array(
            'admin_username' => $username,
            'admin_password' => md5($password)
        );
        $data = $this->m_rental->edit_data($where,'admin');
        $d = $this->m_rental->edit_data($where,'admin')->row();
        $cek = $data->num_rows();
        if($cek > 0){
            $session = array(
                'id'=> $d->admin_id,
                'nama'=> $d->admin_nama,
                'status' => 'login'
            );
            $this->session->set_userdata($session);
            redirect(base_url().'admin');
        }else{
            redirect(base_url().'welcome?pesan=gagal');
        }
    }else{
        $this->load->view('login');
    }
}
```

Pertama kita menangkap data yang di kirim dari form. Dan kita tetapkan validasinya, bahwa username dan password harus wajib di isi (required)

```
$username = $this->input->post('username');
$password = $this->input->post('password');
$this->form_validation->set_rules('username','Username','trim|required');
$this->form_validation->set_rules('password','Password','trim|required');
```

## PENJELASAN !!!

Kemudian kita cek validasinya

```
if($this->form_validation->run() != false){  
    $where = array(  
        'admin_username' => $username,  
        'admin_password' => md5($password)  
    );  
    $data = $this->m_rental->edit_data($where, 'admin');  
    $d = $this->m_rental->edit_data($where, 'admin')->row();  
    $cek = $data->num_rows();  
    if($cek > 0){  
        $session = array(  
            'id'=> $d->admin_id,  
            'nama'=> $d->admin_nama,  
            'status' => 'login'  
        );  
        $this->session->set_userdata($session);  
        redirect(base_url().'admin');  
    }else{  
        redirect(base_url().'welcome?pesan=gagal');  
    }  
}else{  
    $this->load->view('login');  
}
```

Jika tidak ada masalah, kita cek username dan password yang di input sesuai atau tidak dengan username dan password pada tabel admin. Jika sesuai maka kita buat session id, nama dan status. Yang mana session id kita simpan id admin yang melakukan login, session nama kita mengimpaunama admin yang login, dan status nya kita set dengan 'login'.

Setelah kita buat sessionnya, maka kita alihkan halamannya ke controller admin. Karena controller admin ini yang akan kita jadikan sebagai halaman admin.

```
redirect(base_url().'admin')
```

Jika login gagal maka kita alihkan ke controller welcome kembali. Yang manalahalaman login ada di controller welcome. Sambil mengirimkan pesan melaluiurluntuk membuat pesan notifikasi bahwa login gagal.

```
redirect(base_url().'welcome?pesan=gagal');
```

## PENJELASAN !!!

Perhatikan pada view login.php yang sudah kita buat. Di sana kita telah membuat pesan notifikasi. Jika terdapat data pesan yang di kirim dari url, dan jika pesannya adalah “gagal”, maka akan di tampilkan pesan notifikasi “ login gagal! Username dan password salah ”.

```
<?php
if(isset($_GET['pesan'])){
    if($_GET['pesan'] == "gagal"){
        echo "<div class='alert alert-danger'>Login gagal! Username dan password salah.</div>";
    }else if($_GET['pesan'] == "logout"){
        echo "<div class='alert alert-danger'>Anda telah logout.</div>";
    }else if($_GET['pesan'] == "belumlogin"){
        echo "<div class='alert alert-success'>Silahkan login dulu.</div>";
    }
}
?>
```

Jika pesannya adalah logout, maka akan di tampilkan “ anda telah logout ”. Jika pesannya adalah “belumlogin”. Maka akan di tampilkan notifikasi “silahkan login dulu”. Sampai di sini, sistem login kita belum selesai, tapi belum rampung 100%, karena kita belum menyiapkan halaman admin. Oleh sebab itu kita akan membuat dan menyiapkan halaman admin dulu. Baru setelah itu kita coba apakah login sudah berjalan sebagai mana yang kita perintahkan atau belum.

# Menyiapkan Halaman Dashboard Admin

Jika login berhasil, halaman akan di alihkan ke controller admin. Maka mari kita siapkan dulu controller admin nya. Buat sebuah controller baru lagi dengan nama Admin.php

## *Application/controller/Admin.php*

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

class Admin extends CI_Controller {

    function __construct() {
        parent::__construct();

        // cek login
        if($this->session->userdata('status') != "login"){
            redirect(base_url().'welcome?pesan=belumlogin');
        }
    }

    function index() {
        $data['transaksi'] = $this->db->query("select * from transaksi order by transaksi_id desc limit 10")->result();
        $data['kostumer'] = $this->db->query("select * from kostumer order by kostumer_id desc limit 10")->result();
        $data['mobil'] = $this->db->query("select * from mobil order by mobil_id desc limit 10")->result();
        $this->load->view('admin/header');
        $this->load->view('admin/index', $data);
        $this->load->view('admin/footer');
    }
}
```

## PENJELASAN !!!

Pada function construct() kita membuat pengecekan jika si admin belum login, maka akan di alihkan ke halaman login. Ini bertujuan untuk melindungi controller admin dari pengguna yang belum melakukan login

```
if($this->session->userdata('status') != "login"){
    redirect(base_url().'welcome?pesan=belumlogin');
}
```

Dan pada function index() di controller admin, kita akan membuat halaman dashboard

```
function index(){
    $data['transaksi'] = $this->db->query("select * from transaksi order by transaksi_id desc limit 10")->result();
    $data['kostumer'] = $this->db->query("select * from kostumer order by kostumer_id desc limit 10")->result();
    $data['mobil'] = $this->db->query("select * from mobil order by mobil_id desc limit 10")->result();
    $this->load->view('admin/header');
    $this->load->view('admin/index', $data);
    $this->load->view('admin/footer');
}
```

## PENJELASAN !!!

Pada function index() kita menampilkan view header, index dan footer. Yang mana pada view index kita memarsing data yang kita ambil dari database.

```
$data['transaksi'] = $this->db->query("select * from transaksi order by transaksi_id desc limit 10")->result();
$data['kostumer'] = $this->db->query("select * from kostumer order by kostumer_id desc limit 10")->result();
$data['mobil'] = $this->db->query("select * from mobil order by mobil_id desc limit 10")->result();
```

\$this->db->query() adalah fungsi untuk menjalankan query database di codeigniter. Teman-teman bisa membaca panduannya pada user\_guide codeigniter.

[http://localhost/rental\\_mobil/user\\_guide/database/queries.html](http://localhost/rental_mobil/user_guide/database/queries.html)

Di sarankan untuk membaca tutorial membuat template web sederhana dengan codeigniter dan tutorial cara memarsing data ke view codeigniter terlebih dulu sebelum melanjutkan ke depan. Tutorial membuat tampilan atau template web sederhana dengan codeigniter dan cara memarsing data ke view sudah di jelaskan pada tutorial codeigniter dasar sebelumnya.

Pada variabel array transaksi di atas bertujuan untuk mengambil data transaksi dari tabel transaksi sebanyak 10 record. Untuk di tampilkan di halaman admin. Variabel array kostumer kita buat untuk mengambil 10 record data kostumer terakhir, dan variabel array mobil kita buat untuk mengambil 10 record terakhir data mobil dari tabel mobil. Semua data yang kita ambil ini akan kita tampilkan pada view index (halaman dashboard admin).

Buat sebuah folder dengan nama ‘admin’ di dalam folder view.kemudian buat 3 buah view baru dengan nama index.php, header.php dan footer.php di dalam folder admin. pada view index.php ini kita akan membuat halaman dashboard admin.

## Application/view/admin/header.php

```
<!DOCTYPE html>
<html>
<head>
    <title>Dashboard - Aplikasi Rental Mobil</title>
    <link rel="stylesheet" type="text/css" href="<?php echo base_url().'assets/css/bootstrap.css' ?>">
    <link rel="stylesheet" type="text/css" href="<?php echo base_url().'assets/datatables/datatables.css' ?>">
    <script type="text/javascript" src="<?php echo base_url().'assets/js/jquery.js'; ?>"></script>
    <script type="text/javascript" src="<?php echo base_url().'assets/js/bootstrap.js'; ?>"></script>
    <script type="text/javascript" src="<?php echo base_url().'assets/dataTables/jquery.dataTables.js'; ?>"></script>
    <script type="text/javascript" src="<?php echo base_url().'assets/dataTables/datatables.js'; ?>"></script>
?>"></script>
</head>
<body>
    <nav class="navbar navbar-default">
        <div class="container">
            <!-- Brand and toggle get grouped for better mobile display -->
            <div class="navbar-header">
                <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1" aria-expanded="false">
                    <span class="sr-only">Toggle navigation</span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                </button>
                <a class="navbar-brand" href="<?php echo base_url().'admin'; ?>">Rental Mobil</a>
            </div>

            <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
                <ul class="nav navbar-nav">
                    <li class="active"><a href="<?php echo base_url().'admin'; ?>"><span class="glyphicon glyphicon-home"></span> Dashboard <span class="sr-only">(current)</span></a></li>
                    <li><a href="<?php echo base_url().'admin/mobil'; ?>"><span class="glyphicon glyphicon-folder-open"></span> Data Mobil</a></li>
                    <li><a href="<?php echo base_url().'admin/kostumer'; ?>"><span class="glyphicon glyphicon-user"></span> Data Kostumer</a></li>
                    <li><a href="<?php echo base_url().'admin/transaksi'; ?>"><span class="glyphicon glyphicon-sort"></span> Transaksi Rental</a></li>
                    <li><a href="<?php echo base_url().'admin/laporan'; ?>"><span class="glyphicon glyphicon-list-alt"></span> Laporan</a></li>
                </ul>

                <ul class="nav navbar-nav navbar-right">
                    <li><a href="<?php echo base_url().'admin/logout'; ?>"><span class="glyphicon glyphicon-log-out"></span> Logout</a></li>
                    <li class="dropdown">
                        <a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true" aria-expanded="false"><?php echo "Halo, "<br>". $this->session->userdata('nama'); ?></a> <span class="caret"></span></a>
                        <ul class="dropdown-menu">
                            <li>
                                <a href="<?php echo base_url().'admin/ganti_password' ?>"><i class="glyphicon glyphicon-lock"></i> Ganti Password</a>
                            </li>
                        </ul>
                    </li>
                </ul>
            </div><!-- /.navbar-collapse -->
        </div><!-- /.container-fluid -->
    </nav>

    <div class="container">
```

# Application/view/admin/index.php

```
<div class="page-header">
    <h3>Dashboard</h3>
</div>

<div class="row">
    <div class="col-lg-3 col-md-6">
        <div class="panel panel-primary">
            <div class="panel-heading">
                <div class="row">
                    <div class="col-xs-3">
                        <i class="glyphicon glyphicon-folder-open"></i>
                    </div>
                    <div class="col-xs-9 text-right">
                        <div class="huge">
                            <?php echo
$this->m_rental->get_data('mobil')->num_rows(); ?>
                        </div>
                        <div>Jumlah Mobil</div>
                    </div>
                </div>
            </div>
            <a href="php echo base_url().'admin/mobil' ?"&gt;"&gt;
                &lt;div class="panel-footer"&gt;
                    &lt;span class="pull-left"&gt;View Details&lt;/span&gt;
                    &lt;span class="pull-right"&gt;&lt;i class="glyphicon glyphicon-arrow-right"&gt;&lt;/i&gt;&lt;/span&gt;
                    &lt;div class="clearfix"&gt;&lt;/div&gt;
                &lt;/div&gt;
            &lt;/a&gt;
        &lt;/div&gt;
    &lt;/div&gt;
    &lt;div class="col-lg-3 col-md-6"&gt;
        &lt;div class="panel panel-success"&gt;
            &lt;div class="panel-heading"&gt;
                &lt;div class="row"&gt;
                    &lt;div class="col-xs-3"&gt;
                        &lt;i class="glyphicon glyphicon-user"&gt;&lt;/i&gt;
                    &lt;/div&gt;
                    &lt;div class="col-xs-9 text-right"&gt;
                        &lt;div class="huge"&gt;
                            &lt;?php echo
$this-&gt;m_rental-&gt;get_data('kostumer')-&gt;num_rows(); ?&gt;
                        &lt;/div&gt;
                        &lt;div&gt;Jumlah Kostumer&lt;/div&gt;
                    &lt;/div&gt;
                &lt;/div&gt;
            &lt;/div&gt;
            &lt;a href="<?php echo base_url().'admin/kostumer' ?"&gt;"&gt;
                &lt;div class="panel-footer"&gt;
                    &lt;span class="pull-left"&gt;View Details&lt;/span&gt;
                    &lt;span class="pull-right"&gt;&lt;i class="glyphicon glyphicon-arrow-right"&gt;&lt;/i&gt;&lt;/span&gt;
                    &lt;div class="clearfix"&gt;&lt;/div&gt;
                &lt;/div&gt;
            &lt;/a&gt;
        &lt;/div&gt;
    &lt;/div&gt;
&lt;/div&gt;</pre
```

## **Application/view/admin/index.php (code lanjutan)**

```
<div class="col-lg-3 col-md-6">
    <div class="panel panel-warning">
        <div class="panel-heading">
            <div class="row">
                <div class="col-xs-3">
                    <i class="glyphicon glyphicon-sort"></i>
                </div>
                <div class="col-xs-9 text-right">
                    <div class="huge">
                        <?php echo
$this->m_rental->get_data('transaksi')->num_rows(); ?>
                        </div>
                    <div>Jumlah Transaksi</div>
                </div>
            </div>
        </div>
        <a href="php echo base_url().'admin/transaksi' ?&gt;"&gt;
            &lt;div class="panel-footer"&gt;
                &lt;span class="pull-left"&gt;View Details&lt;/span&gt;
                &lt;span class="pull-right"&gt;&lt;i class="glyphicon glyphicon-arrow-right"&gt;&lt;/i&gt;&lt;/span&gt;
                &lt;div class="clearfix"&gt;&lt;/div&gt;
            &lt;/div&gt;
        &lt;/a&gt;
    &lt;/div&gt;
&lt;/div&gt;
&lt;div class="col-lg-3 col-md-6"&gt;
    &lt;div class="panel panel-danger"&gt;
        &lt;div class="panel-heading"&gt;
            &lt;div class="row"&gt;
                &lt;div class="col-xs-3"&gt;
                    &lt;i class="glyphicon glyphicon-ok"&gt;&lt;/i&gt;
                &lt;/div&gt;
                &lt;div class="col-xs-9 text-right"&gt;
                    &lt;div class="huge"&gt;
                        &lt;?php echo
$this-&gt;m_rental-&gt;edit_data(array('transaksi_status'=&gt;1), 'transaksi')-&gt;num_rows(); ?&gt;
                        &lt;/div&gt;
                    &lt;div&gt;Rental Selesai&lt;/div&gt;
                &lt;/div&gt;
            &lt;/div&gt;
        &lt;/div&gt;
        &lt;a href="<?php echo base_url().'admin/transaksi' ?&gt;"&gt;
            &lt;div class="panel-footer"&gt;
                &lt;span class="pull-left"&gt;View Details&lt;/span&gt;
                &lt;span class="pull-right"&gt;&lt;i class="glyphicon glyphicon-arrow-right"&gt;&lt;/i&gt;&lt;/span&gt;
                &lt;div class="clearfix"&gt;&lt;/div&gt;
            &lt;/div&gt;
        &lt;/a&gt;
    &lt;/div&gt;
&lt;/div&gt;</pre
```

## **Application/view/admin/index.php (code lanjutan)**

```
<hr>
<div class="row">
    <div class="col-lg-4">
        <div class="panel panel-default">
            <div class="panel-heading">
                <h3 class="panel-title"><i class="glyphicon glyphicon-random arrow-right"></i> Mobil</h3>
            </div>
            <div class="panel-body">
                <div class="list-group">
                    <?php foreach($mobil as $m) { ?>
                    <a href="#" class="list-group-item">
                        <span class="badge"><?php if($m->mobil_status == 1){echo "Tersedia";}else{echo "D dirental"; } ?></span>
                        <i class="glyphicon glyphicon-user"></i> <?php echo $m->mobil_merk; ?>
                    </a>
                    <?php } ?>
                </div>
                <div class="text-right">
                    <a href="<?php echo base_url().'admin/mobil' ?>">Lihat Semua Mobil <i class="glyphicon glyphicon-arrow-right"></i></a>
                </div>
            </div>
        </div>
    <div class="col-lg-3">
        <div class="panel panel-default">
            <div class="panel-heading">
                <h3 class="panel-title"><i class="glyphicon glyphicon-user-o"></i> Kostumer Terbaru</h3>
            </div>
            <div class="panel-body">
                <div class="list-group">
                    <?php foreach($kostumer as $k) { ?>
                    <a href="#" class="list-group-item">
                        <span class="badge"><?php echo $k->kostumer_jk ?></span>
                        <i class="glyphicon glyphicon-user"></i> <?php echo $k->kostumer_nama; ?>
                    </a>
                    <?php } ?>
                </div>
                <div class="text-right">
                    <a href="<?php echo base_url().'admin/kostumer' ?>">Lihat Semua Kostumer <i class="glyphicon glyphicon-arrow-right"></i></a>
                </div>
            </div>
        </div>
    </div>
</div>
```

## **Application/view/admin/index.php (code lanjutan)**

```
<div class="col-lg-5">
    <div class="panel panel-default">
        <div class="panel-heading">
            <h3 class="panel-title"><i class="glyphicon glyphicon-sort"></i>
Peminjaman Terakhir</h3>
        </div>
        <div class="panel-body">
            <div class="table-responsive">
                <table class="table table-bordered table-hover table-striped">
                    <thead>
                        <tr>
                            <th>Tgl. Transaksi</th>
                            <th>Tgl. Pinjam</th>
                            <th>Tgl. Kembali</th>
                            <th>Total</th>
                        </tr>
                    </thead>
                    <tbody>
                        <?php
                        foreach($transaksi as $t) {
                            ?>
                            <tr>
                                <td><?php echo
date('d/m/Y', strtotime($t->transaksi_tgl)); ?></td>
                                <td><?php echo
date('d/m/Y', strtotime($t->transaksi_tgl_pinjam)); ?></td>
                                <td><?php echo
date('d/m/Y', strtotime($t->transaksi_tgl_kembali)); ?></td>

                                <td><?php echo "Rp.
".number_format($t->transaksi_harga).",-"; ?></td>
                            </tr>
                        <?php } ?>
                    </tbody>
                </table>
            </div>
            <div class="text-right">
                <a href=<?php echo base_url().'admin/transaksi' ?>">Lihat Semua
Transaksi <i class="glyphicon glyphicon-arrow-right"></i></a>
            </div>
        </div>
    </div>
</div>
<!-- /.row -->
```

## Application/view/footer.php

```
</div>
<script type="text/javascript">
$(document).ready(function() {
    $("#table-datable").dataTable();
});
</script>
</body>
</html>
```

The screenshot shows the Sublime Text interface with two tabs open: 'header.php' and 'footer.php'. The left sidebar displays the project's directory structure:

```
File Edit Selection Find View Goto Tools Project Preferences Help
FOLDERS
+ rental_mobil
  + application
    - cache
    - config
      + controllers
        - Admin.php
        - index.html
        - Welcome.php
      - core
      - helpers
      - hooks
      - language
      - Models
      - logs
      - models
      - third_party
      - views
    + view
      - admin
        - footer.php
        - header.php
        - index.php
      - errors
        - index.html
        - login.php
        - welcome_message.php
      - Macros
        - index.html
      - assets
      - system
```

The 'header.php' tab contains the following code:

```
<!DOCTYPE html>
<html>
<head>
    <title>Dashboard - Aplikasi Rental Mobil</title>
    <link rel="stylesheet" type="text/css" href="php echo base_url(); 'assets/css/bootstrap.css'; ?&gt;"&gt;
    &lt;link rel="stylesheet" type="text/css" href="<?php echo base_url(); 'assets/datable/datatables.css'; ?&gt;"&gt;
    &lt;script type="text/javascript" src="<?php echo base_url(); 'assets/js/jquery.js'; ?&gt;"&gt;&lt;/script&gt;
    &lt;script type="text/javascript" src="<?php echo base_url(); 'assets/js/bootstrap.js'; ?&gt;"&gt;&lt;/script&gt;
    &lt;script type="text/javascript" src="<?php echo base_url(); 'assets/datable/jquery.dataTables.js'; ?&gt;"&gt;&lt;/script&gt;
    &lt;script type="text/javascript" src="<?php echo base_url(); 'assets/datable/dataTables.js'; ?&gt;"&gt;&lt;/script&gt;
&lt;/head&gt;
&lt;body&gt;
    &lt;nav class="navbar navbar-default"&gt;
        &lt;div class="container"&gt;
            &lt;!-- Brand and toggle get grouped for better mobile display --&gt;
            &lt;div class="navbar-header"&gt;
                &lt;button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1"&gt;
                    &lt;span class="sr-only"&gt;Toggle navigation&lt;/span&gt;
                    &lt;span class="icon-bar"&gt;&lt;/span&gt;
                    &lt;span class="icon-bar"&gt;&lt;/span&gt;
                    &lt;span class="icon-bar"&gt;&lt;/span&gt;
                &lt;/button&gt;
                &lt;a href="#" class="navbar-brand" style="color: #333; font-weight: bold;"Rental Mobil</a>
            </div>
            <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
                <ul class="nav navbar-nav">
                    <li class="active"><a href="php echo base_url(); 'admin'; ?&gt;"&gt;&lt;span class="glyphicon glyphicon-home"&gt;&lt;/span&gt; Home&lt;/a&gt;&lt;/li&gt;
                    &lt;li&gt;&lt;a href="<?php echo base_url(); 'admin/mobil'; ?&gt;"&gt;&lt;span class="glyphicon glyphicon-list-alt"&gt;&lt;/span&gt; Mobil&lt;/a&gt;&lt;/li&gt;
                    &lt;li&gt;&lt;a href="<?php echo base_url(); 'admin/kostumer'; ?&gt;"&gt;&lt;span class="glyphicon glyphicon-user"&gt;&lt;/span&gt; Kostumer&lt;/a&gt;&lt;/li&gt;
                    &lt;li&gt;&lt;a href="<?php echo base_url(); 'admin/transaksi'; ?&gt;"&gt;&lt;span class="glyphicon glyphicon-tasks"&gt;&lt;/span&gt; Transaksi&lt;/a&gt;&lt;/li&gt;
                    &lt;li&gt;&lt;a href="<?php echo base_url(); 'admin/laporan'; ?&gt;"&gt;&lt;span class="glyphicon glyphicon-bar-chart"&gt;&lt;/span&gt; Laporan&lt;/a&gt;&lt;/li&gt;
                &lt;/ul&gt;
            &lt;/div&gt;
        &lt;/div&gt;
    &lt;/nav&gt;
    &lt;div class="container"&gt;
        &lt;div class="row"&gt;
            &lt;div class="col-md-12" style="text-align: center; margin-top: 20px;"&gt;
                &lt;h1&gt;Selamat Datang di Aplikasi Rental Mobil&lt;/h1&gt;
                &lt;p&gt;Silahkan login untuk melanjutkan...&lt;/p&gt;
                &lt;div style="margin-top: 10px;"&gt;
                    &lt;button type="button" class="btn btn-primary" style="width: 150px; height: 40px; border-radius: 20px; padding: 10px; font-size: 12px; font-weight: bold; border: none; background-color: #337AB7; color: white; text-decoration: none; transition: all 0.3s ease-in-out; "&gt;Login&lt;/button&gt;
                &lt;/div&gt;
            &lt;/div&gt;
        &lt;/div&gt;
    &lt;/div&gt;
&lt;/body&gt;</pre

The 'footer.php' tab contains the following code:



```
</div>
<script type="text/javascript">
$(document).ready(function() {
    $("#table-datable").dataTable();
});
</script>
</body>
</html>
```


```

Sampai di sini, login yang kita buat telah selesai. Silahkan buka controller welcome pada browser untuk melihat hasilnya.

[http://localhost/rental\\_mobil/](http://localhost/rental_mobil/)

perlu di perhatikan di sini, saat kita mengakses [http://localhost/rental\\_mobil/](http://localhost/rental_mobil/) sama saja seperti kita mengakses [http://localhost/rental\\_mobil/welcome](http://localhost/rental_mobil/welcome). karena controller welcome telah kita set sebagai controller default.

The screenshot shows a web browser window with the URL 'localhost/rental\_mobil/' in the address bar. The page title is 'APLIKASI RENTAL MOBIL' and the heading is 'LOGIN'. Below the heading is a login form with two input fields labeled 'username' and 'password', and a blue 'Login' button. A blue arrow points from the text in the preceding paragraph to this login form.

Jika teman-teman mengikuti panduan sebelumnya tentang membuat akun admin/menginsert data username dan password admin ke tabel admin di atas, maka akun adminnya adalah :

Username : admin

Password : admin123

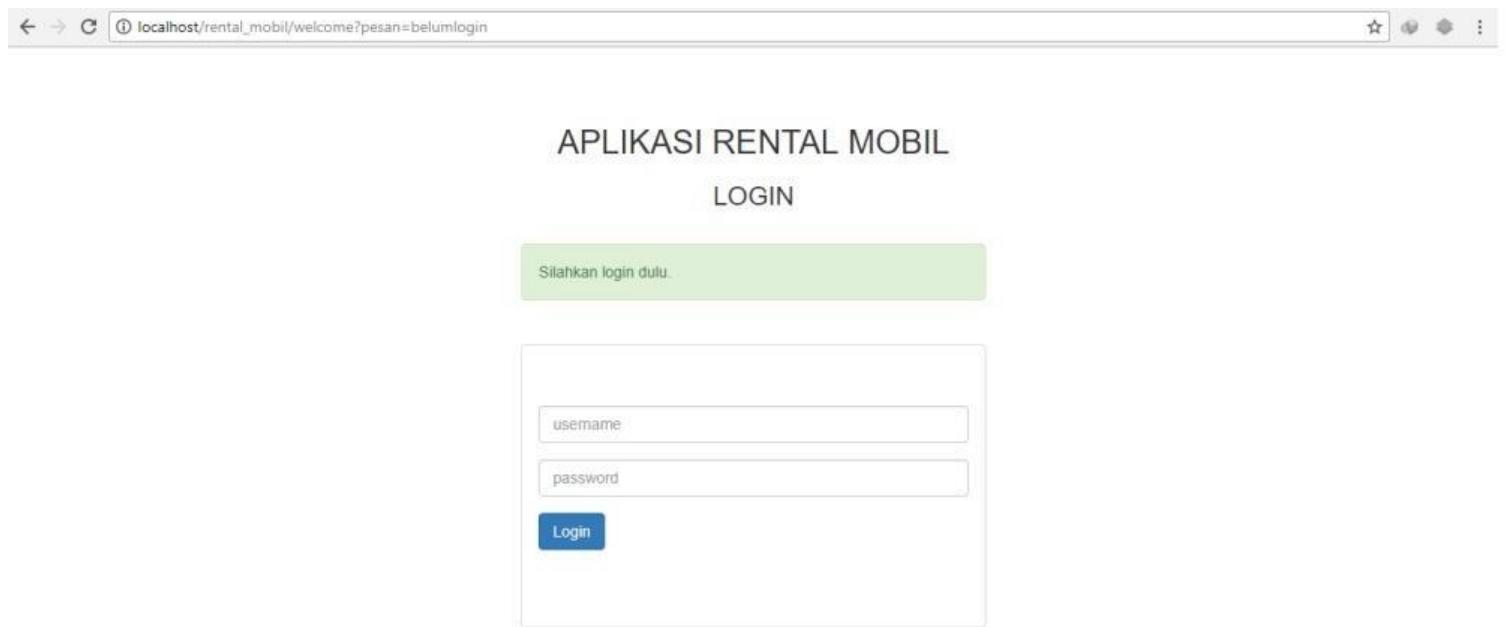
Pertama saya akan mencoba memasukkan username dan password yang salah. Untuk memastikan sistem login yang kita buat sudah berjalan sebagaimana mestinya. Di sini saya mencoba memasukkan username : fabio, dan passwordnya fabio123.

The screenshot shows a web browser window with the URL 'localhost/rental\_mobil/welcome?pesan=gagal' in the address bar. The page title is 'APLIKASI RENTAL MOBIL' and the heading is 'LOGIN'. Above the login form, there is a pink message box containing the text 'Login gagal Username dan password salah.' A blue arrow points from the text in the preceding paragraph to this error message.

Dan coba untuk akses halaman admin tanpa login. Dengan langsung mengakses controller admin. Seharusnya jika belum login, maka tidak bisa mengakses halaman admin (controller admin). Coba kita akses langsung halaman admin dengan mengakses url berikut.

***http://localhost/rental\_mobil/admin***

maka akan muncul pesan notifikasi yang mengharuskan kita untuk melakukann login terlebih dulu.



Setelah kita mencoba masuk langsung ke halaman admin tanpa login, dan mencoba memasukkan username password yang salah, sekarang kita akan mencoba memasukkan username dan password yang benar. Coba kita masukkan username = admin, dan password = admin123.

Maka hasilnya kita berhasil login dan masuk ke halaman dashboard admin.

## Dashboard

 0  
Jumlah Mobil  
[View Details ➔](#)

 0  
Jumlah Kostumer  
[View Details ➔](#)

 0  
Jumlah Transaksi  
[View Details ➔](#)

 0  
Rental Selesai  
[View Details ➔](#)

 Mobil  
[Lihat Semua Mobil ➔](#)

 Kostumer Terbaru  
[Lihat Semua Kostumer ➔](#)

 Peminjaman Terakhir  

| Tgl. Transaksi                          | Tgl. Pinjam | Tgl. Kembali | Total |
|---|-------------|--------------|-------|
| <a href="#">Lihat Semua Transaksi ➔</a> |             |              |       |

**Login selesai**

# Membuat Logout

Setelah berhasil membuat login, selanjutnya kita akan membuat logout. Atau link keluar dari sistem. Perhatikan pada menu logout pada view header.php. kita telah membuat link menu logout.

```
<li><a href="php echo base_url().'admin/logout'; ?&gt;"&gt;&lt;span class="glyphicon glyphicon-log-out"&gt;&lt;/span&gt; Logout&lt;/a&gt;&lt;/li&gt;</pre
```

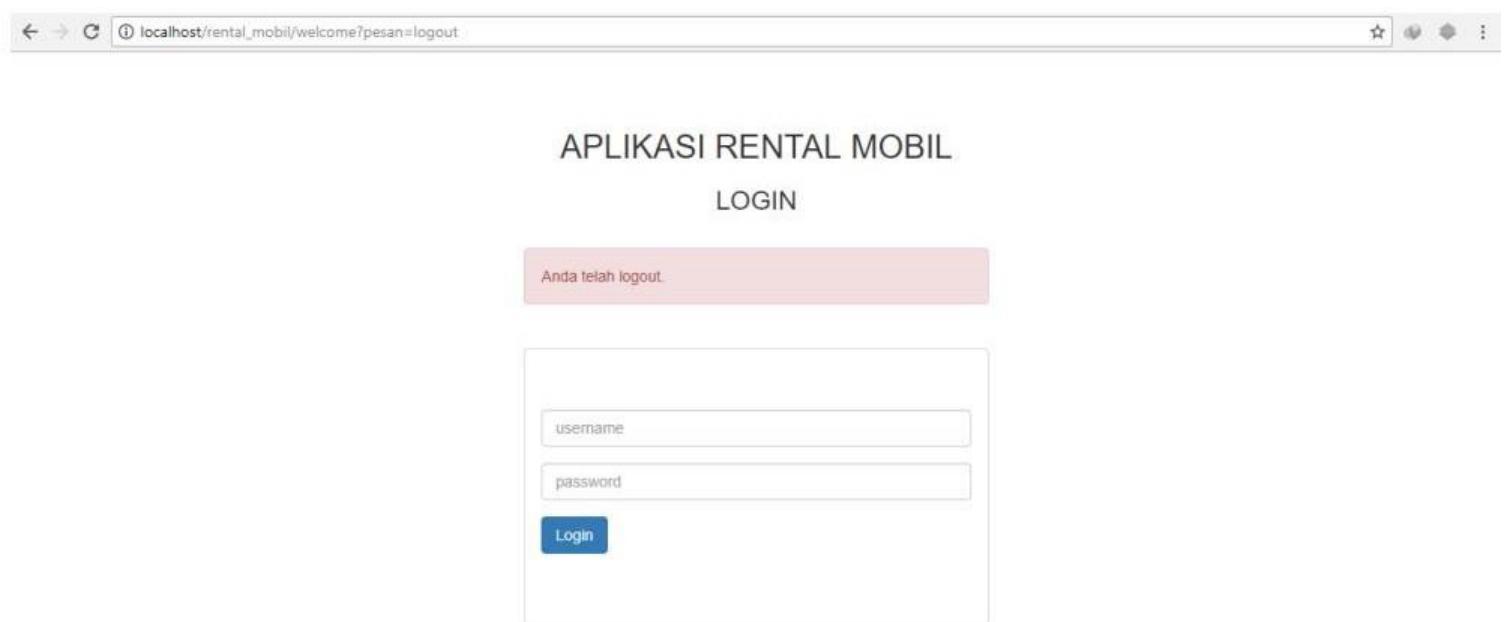
Link logout di atas, mengarahkan ke function/method logout di controller admin. Oleh karena itu kita akan membuat function logout pada controlleradmin.

**Tambahkan function logout berikut ini ke controller admin.**

```
function logout() {  
    $this->session->sess_destroy();  
    redirect(base_url().'welcome?pesan=logout');  
}
```

## PENJELASAN !!!

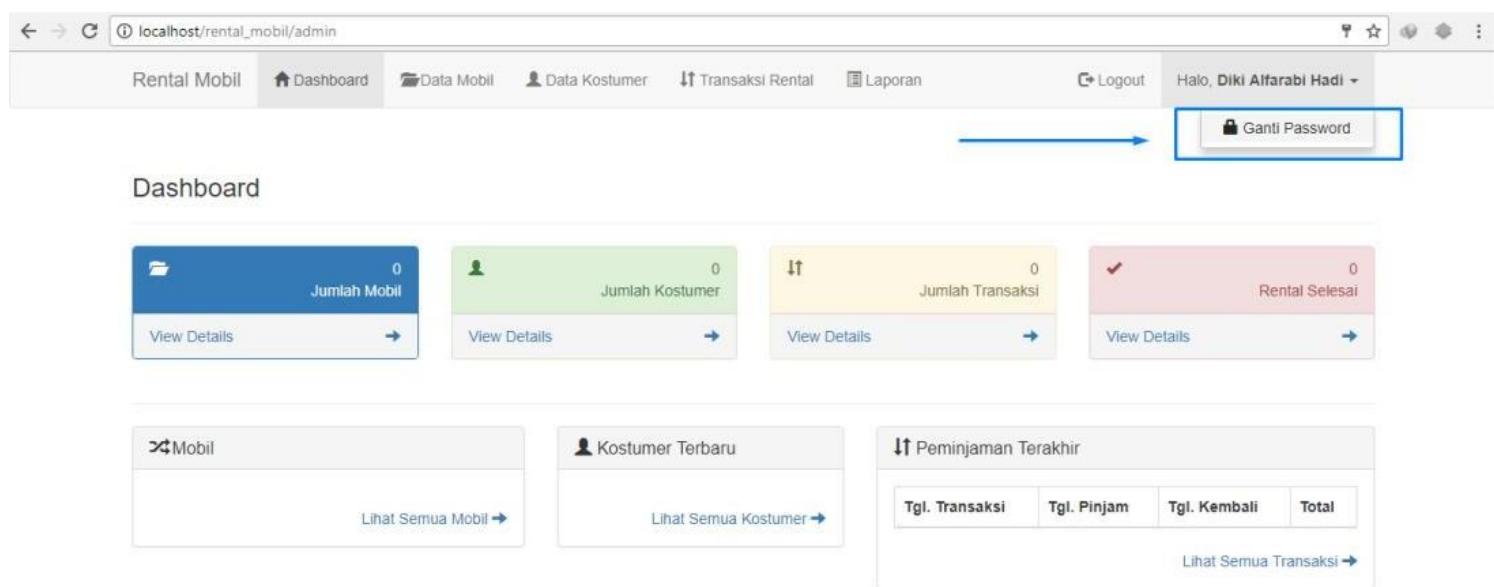
Syntax \$this->session->sess\_destroy(); merupakan fungsi codeigniter untuk menghapus semua session. Setelah menghapus session, kita alihkan halaman ke halaman login. Sekarang coba klik pada menu logtout. Maka halaman akan di alihkan kembali ke halaman login.



# Membuat Fitur Ganti Password Admin

Sampai pada tahap ini, kita akan membuat sebuah fitur untuk admin, yaitu fitur untuk mengganti password. Perhatikan pada menu ganti password di header admin. Kita telah membuatkan link menu untuk mengganti password. Dimana link tersebut sudah kita arahkan ke function ganti\_password di controller admin

```
<li>
    <a href="php echo base_url().'admin/ganti_password' ?&gt;"&gt;&lt;i class="glyphicon glyphicon-lock"&gt;&lt;/i&gt; Ganti Password&lt;/a&gt;
&lt;/li&gt;</pre
```



Jadi kita akan membuat sebuah function baru lagi di controller admin, yaitu function ganti\_password(). Tambahkan function/syntax berikut pada controller admin.php.

## Application/controller/Admin.php

```
function ganti_password(){
    $this->load->view('admin/header');
    $this->load->view('admin/ganti_password');
    $this->load->view('admin/footer');
}
```

Seperti yang teman-teman lihat pada function di atas, kita menampilkan view header.php, view ganti\_password.php dan view footer.php. pada view ganti\_password.php akan kita buat form untuk mengganti password. Yuk kita buat sebuah file baru untuk view. Buat file baru dengan nama ganti\_password.php. dalam folder admin dalam view.

### ***Application/view/admin/ganti\_password.php***

```
<div class="page-header">
    <h3>Ganti Password</h3>
</div>

<div class="row">
    <div class="col-md-6 col-md-offset-3">
        <?php
            if(isset($_GET['pesan'])){
                if($_GET['pesan'] == "berhasil"){
                    echo "<div class='alert alert-success'>Password berhasil
di ganti.</div>";
                }
            }
        ?>
        <form action="<?php echo
base_url().'admin/ganti_password_act' ?>" method="post">
            <div class="form-group">
                <label>Password Baru</label>
                <input class="form-control" type="password"
name="pass_baru">
                    <?php echo form_error('pass_baru'); ?>
            </div>
            <div class="form-group">
                <label>Ulangi Password Baru</label>
                <input class="form-control" type="password"
name="ulang_pass">
                    <?php echo form_error('ulang_pass'); ?>
            </div>
            <div class="form-group">
                <input class="btn btn-primary btn-sm" type="submit"
value="Simpan">
            </div>
        </form>
    </div>
</div>
```

Aksi dari form ini akan di jalankan pada function ganti\_password\_act pada controller admin.

```
<form action="php echo base_url().'admin/ganti_password_act' ?&gt;" method="post"&gt;</pre
```

Jadi sekarang buat function ganti\_password\_act() pada controller admin.

### ***Application/controllers/Admin.php***

```
function ganti_password_act() {
    $pass_baru = $this->input->post('pass_baru');
    $ulang_pass = $this->input->post('ulang_pass');
    $this->form_validation->set_rules('pass_baru', 'Password Baru', 'required|matches[ulang_pass]');
    $this->form_validation->set_rules('ulang_pass', 'Ulangi Password Baru', 'required');

    if($this->form_validation->run() != false) {
        $data = array(
            'admin_password' => md5($pass_baru)
        );
        $w = array(
            'admin_id' => $this->session->userdata('id')
        );
        $this->m_rental->update_data($w, $data, 'admin');
        redirect(base_url().'admin/ganti_password?pesan=berhasil');
    } else{
        $this->load->view('admin/header');
        $this->load->view('admin/ganti_password');
        $this->load->view('admin/footer');
    }
}
```

### **PENJELASAN !!!**

Syntax di atas adalah function yang kita buat untuk menangani form ganti password. Perhatikan pada validation yang kita buat, kita menggunakan validation required dan matches. Seperti yang pernah di singgung sebelumnya tentang form validation required, required berarti kita menetapkan form tersebut menjadi wajib di isi atau tidak boleh kosong. Dan matches[] adalah form validation untuk memvalidasi kesamaan value. Di sini kita mewajibkan password baru harus sama dengan form ulangi password.

```
$pass_baru = $this->input->post('pass_baru');
$ulang_pass = $this->input->post('ulang_pass');
$this->form_validation->set_rules('pass_baru', 'Password Baru', 'required|matches[ulang_pass]');
$this->form_validation->set_rules('ulang_pass', 'Ulangi Password Baru', 'required');
```

## PENJELASAN !!!

Kemudian kita cek validasinya, jika sesuai dan tidak ada masalah, maka kita update data password si admin.

```
if($this->form_validation->run() != false){  
    $data = array(  
        'admin_password' => md5($pass_baru)  
    );  
    $w = array(  
        'admin_id' => $this->session->userdata('id')  
    );  
    $this->m_rental->update_data($w, $data, 'admin');  
    redirect(base_url().'admin/ganti_password?pesan=berhasil');  
} else{  
    $this->load->view('admin/header');  
    $this->load->view('admin/ganti_password');  
    $this->load->view('admin/footer');  
}
```

Syntax `$this->session->userdata('id')` merupakan perintah untuk menampilkan session 'id' admin yang sedang login. Setelah data password admin di update, maka kemudian halaman akan di alihkan lagi ke function `ganti_password`.

Sekarang kita akan mencoba mengganti password admin. Seperti yang di jelaskan tadi sebelumnya, kita membuat validasi password dimana password baru yang di ketikkan pada form password baru harus sama dengan yang di ketikkan pada form ulang password. Ini berguna untuk memvalidasi dan mencegah admin salah memasukkan password barunya (salah ketik).

Disini kita akan mencoba mengganti password admin dengan "akademikita123". Tapi kita coba dulu memasukkan password yang berbeda. Maka hasilnya akan muncul pesan notifikasi bahwa "password baru" dan "ulangi password baru" yang di input harus sama.

The screenshot shows a web application interface for changing a password. At the top, there's a navigation bar with links like 'Rental Mobil', 'Dashboard', 'Data Mobil', 'Data Kostumer', 'Transaksi Rental', 'Laporan', 'Logout', and a user greeting 'Halo, Diki Alfarabi Hadi'. Below the navigation is a form titled 'Ganti Password'. It has two input fields: 'Password Baru' and 'Ulangi Password Baru'. A red error message 'The Password Baru field does not match the Ulangi Password Baru field.' is displayed between them. A blue arrow points from this message to the 'Ulangi Password Baru' field. A 'Simpan' button is located at the bottom right of the form.

Coba kita masukkan password baru yang sesuai, pada form password baru kitaketik “akademikita123”. Dan pada form ulangi password baru kita ketik “akademikita123”.

This screenshot shows the same 'Ganti Password' page after the password has been successfully changed. The validation error message is gone, and instead, there's a green success message box containing the text 'Password berhasil di ganti.' (Password has been successfully changed). The rest of the page remains the same, with the 'Simpan' button at the bottom.

Maka password admin pun berhasil di ganti. Dan muncul pesan notifikasi “password berhasil di ganti.”.

This screenshot shows the final step of the password change process. The 'Ganti Password' page displays a green success message box with the text 'Password berhasil di ganti.' (Password has been successfully changed). The 'Simpan' button is visible at the bottom of the form.

**Sampai sini kita telah selesai membuat fitur ganti password**

# Membuat CRUD Data Mobil

Cara membuat CRUD sudah di jelaskan pada tutorial codeigniter bagian dasar sebelumnya. Pada tahapan ini kita akan membuat CRUD untuk data-data mobil. Seperti perancangan database khusus nya pada tabel mobil yang sudah kita buat, sesuai pada gambar perancangan database di atas, data-data mobil yang akan kita simpan adalah data id mobil (mobil\_id), merk mobil(mobil\_merk), nomor plat mobil(mobil\_plat), warna mobil (mobil\_warna), tahun mobil (mobil\_tahun) dan status ketersediaan mobil (mobil\_status).

## Menampilkan Data Mobil dari database

Data mobil akan kita tampilkan pada function/method mobil(). Sesuai dengan link yang sudah kita tentukan di menu “Data Mobil” pada menu navigasi

```
<li><a href="<?php echo base_url().'admin/mobil'; ?>"><span class="glyphicon glyphicon-folder-open"></span> Data Mobil</a></li>
```

Tambahkan sebuah function dengan mobil() pada controller admin.php.

### ***Application/controllers/Admin.php***

```
// CRUD MOBIL
function mobil(){
    $data['mobil'] = $this->m_rental->get_data('mobil')->result();
    $this->load->view('admin/header');
    $this->load->view('admin/mobil',$data);
    $this->load->view('admin/footer');
}
```

### **PENJELASAN !!!**

Pertama, kita akan mengambil data mobil dari database dengan menggunakan function get\_data() di model m\_rental.

```
function edit_data($where,$table){
    return $this->db->get_where($table,$where);
}
```

Dari data yang di dapatkan dari hasil query di function edit\_data() di atas kita return kembali untuk kita masukkan ke variabel array mobil.

## PENJELASAN !!!

```
$data['mobil'] = $this->m_rental->get_data('mobil')->result();
$this->load->view('admin/header');
$this->load->view('admin/mobil', $data);
$this->load->view('admin/footer');
```

Lalu kita parsing datanya ke view mobil.php seperti pada syntax di atas. jadi di sini pada view mobil.php nah kita akan menampilkan data mobilnya.

Buat view baru lagi di dalam folder admin dengan nama mobil.php

### Application/view/admin/mobil.php

```
<div class="page-header">
    <h3>Data Mobil</h3>
</div>

<a href="=base_url().'admin/mobil_add'; ?" class="btn btn-primary btn-sm"><span class="glyphicon glyphicon-plus"></span> Mobil Baru</a>
<br/><br/>
<div class="table-responsive">
<table class="table table-bordered table-striped table-hover" id="table-datable">
    <thead>
        <tr>
            <th>No</th>
            <th>Merk Mobil</th>
            <th>Plat</th>
            <th>Warna</th>
            <th>Tahun Pembuatan</th>
            <th>Status</th>
            <th></th>
        </tr>
    </thead>
    <tbody>
        <?php
        $no = 1; foreach($mobil as
        $m){
        ?>
        <tr>
            <td><?php echo $no++;></td>
            <td><?php echo $m->mobil_merk ?></td>
            <td><?php echo $m->mobil_plat ?></td>
            <td><?php echo $m->mobil_warna ?></td>
            <td><?php echo $m->mobil_tahun ?></td>
            <td>
                <?php
                if($m->mobil_status == "1"){
                    echo "Tersedia";
                }else if($m->mobil_status == "2"){echo
                    "Sedang Di Rental";
                }
                ?>
            </td>
            <td>
                <a class="btn btn-warning btn-sm" href="=base_url().'admin/mobil_edit/'.$m-&gt;mobil_id; ?&gt;"&gt;&lt;span class="glyphicon glyphicon-plus"&gt;&lt;/span&gt; Edit&lt;/a&gt;
                &lt;a class="btn btn-danger btn-sm" href="<?=base_url().'admin/mobil_hapus/'.$m-&gt;mobil_id; ?&gt;"&gt;&lt;span class="glyphicon glyphicon-trash"&gt;&lt;/span&gt; Hapus&lt;/a&gt;
            &lt;/td&gt;
        &lt;/tr&gt;
        &lt;?php
    }
    ?&gt;
    &lt;/tbody&gt;
&lt;/table&gt;
&lt;/div&gt;</pre
```

Sekarang coba kita klik pada menu data mobil. Maka halaman akan di arahkan ke method/function mobil(). Dan data mobil pun berhasil di tampilkan.

Data Mobil

+ Mobil Baru

10 records per page

Search:

No	Merk Mobil	Plat	Warna	Tahun Pembuatan	Status
No data available in table					

Showing 0 to 0 of 0 entries

← Previous Next →

Sampai di sini kita telah selesai menampilkan data mobil. Tapi karena kita masih belum memiliki data di tabel mobil, jadi masih kosong. Selanjutnya kita akan membuat form input data mobil.

# Input Data Mobil ke Database

Form input data mobil akan kita buat pada method/function mobil\_add() dalam controller Admin.php. seperti yang sudah kita arahkan pada tombol “Mobil Baru”.

```
<a href=<?php echo base_url().'admin/mobil_add'; ?><span class="glyphicon glyphicon-plus"></span> Mobil Baru</a>
```

Buat function mobil\_add() baru pada controller Admin.php.

## ***Application/controller/Admin.php***

```
function mobil_add() {  
    $this->load->view('admin/header');  
    $this->load->view('admin/mobil_add');  
    $this->load->view('admin/footer');  
}
```

Pada function mobil\_add() ini kita akan membuat form penginputan data mobil baru. Yaitu akan kita buat pada view mobil\_add.php. seperti yang sudah kita tetapkan pada function mobil\_add.php.

## **Application/view/admin/mobil\_add.php**

```
<div class="page-header">
    <h3>Mobil Baru</h3>
</div>

<form action="<?php echo base_url().'admin/mobil_add_act' ?>" method="post">
    <div class="form-group">
        <label>Merk Mobil</label>
        <input type="text" name="merk" class="form-control">
        <?php echo form_error('merk'); ?>
    </div>
    <div class="form-group">
        <label>No. Plat Kendaraan</label>
        <input type="text" name="plat" class="form-control">
    </div>
    <div class="form-group">
        <label>Warna</label>
        <input type="text" name="warna" class="form-control">
    </div>
    <div class="form-group">
        <label>Tahun Kendaraan</label>
        <input type="text" name="tahun" class="form-control">
    </div>
    <div class="form-group">
        <label>Status Mobil</label>
        <select name="status" class="form-control">
            <option value="1">Tersedia</option>
            <option value="2">Sedang Di Rental</option>
        </select>
        <?php echo form_error('status'); ?>
    </div>
    <div class="form-group">
        <input type="submit" value="Simpan" class="btn btn-primary">
    </div>
</div>
</form>
```

## **PENJELASAN !!!**

Form tambah mobil ini akan di proses pada function/method mobil\_add\_act() di controller admin.php.

```
<form action="<?php echo base_url().'admin/mobil_add_act' ?>" method="post">
```

Jadi kita buat lagi function baru di controller admin.php. yaitu function mobil\_add\_act()

## **Application/controller/Admin.php**

```
function mobil_add_act(){
    $merk = $this->input->post('merk');
    $plat = $this->input->post('plat');
    $warna = $this->input->post('warna');
    $tahun = $this->input->post('tahun');
    $status = $this->input->post('status');
    $this->form_validation->set_rules('merk','Merk Mobil','required');
    $this->form_validation->set_rules('status','Status Mobil','required');

    if($this->form_validation->run() != false) {
        $data = array(
            'mobil_merk' => $merk,
            'mobil_plat' => $plat,
            'mobil_warna' => $warna,
            'mobil_tahun' => $tahun,
            'mobil_status' => $status
        );
        $this->m_rental->insert_data($data,'mobil');
        redirect(base_url().'admin/mobil');
    }else{
        $this->load->view('admin/header');
        $this->load->view('admin/mobil_add');
        $this->load->view('admin/footer');
    }
}
```

## **PENJELASAN !!!**

Untuk penjelasannya sendiri sama saja seperti tutorial cara membuat CRUD dengan codeigniter yang telah di jelaskan pada tutorial codeigniter bagian dasar sebelumnya.

Sekarang kita akan mencoba menginput data mobil baru dengan klik pada tombol “mobil baru”.

A screenshot of a web application's admin panel. The URL is `localhost/rental_mobil/admin/mobil`. The top navigation bar includes links for Rental Mobil, Dashboard, Data Mobil (which is active), Data Kostumer, Transaksi Rental, Laporan, Logout, and a user profile for Diki Alfarabi Hadi.

The main content area is titled "Data Mobil". It features a table with columns: No, Merk Mobil, Plat, Warna, Tahun Pembuatan, and Status. A blue-bordered button labeled "+ Mobil Baru" is positioned above the table. Below the table, there are dropdowns for "records per page" (set to 10) and a search bar. The message "No data available in table" is displayed. At the bottom, it shows "Showing 0 to 0 of 0 entries" and navigation buttons for "Previous" and "Next".

Kemudian akan di alihkan ke halaman tambah mobil baru. Lalu isikan datamobil baru.

A screenshot of the "Mobil Baru" (New Car) form page. The URL is `localhost/rental_mobil/admin/mobil_add`. The top navigation bar is identical to the previous screen.

The form fields are as follows:

- Merk Mobil: Toyota Avanza
- No. Plat Kendaraan: B 7777 XX
- Warna: Hitam Metalik
- Tahun Kendaraan: 2015
- Status Mobil: Tersedia

A blue-bordered button labeled "Simpan" is at the bottom of the form.

Dan klik simpan. Maka data mobil akan di simpan ke database

A screenshot of the "Data Mobil" page after saving the new car. The URL is `localhost/rental_mobil/admin/mobil`. The table now contains one row:

No	Merk Mobil	Plat	Warna	Tahun Pembuatan	Status	Action
1	Toyota Avanza	B 7777 XX	Hitam Metalik	2015	Tersedia	

Below the table, it says "Showing 1 to 1 of 1 entries".

## Data Mobil

+ Mobil Baru

10 records per page

Search:

No	Merk Mobil	Plat	Warna	Tahun Pembuatan	Status	Action
1	Toyota Avanza	B 7777 XX	Hitam Metalik	2015	Tersedia	

Showing 1 to 1 of 1 entries

← Previous 1 Next →

Oke, data mobil pun berhasil di tambahkan. Selanjutnya kita akan membuat fitur

# Update Data Mobil di Database

Untuk megredit data mobil, akan kita buat sebuah form baru lagi pada function mobil\_edit() di controller admin.php. seperti yang telah kita arahkan pada tombol edit di data mobil.

```
<a class="btn btn-warning btn-sm" href="php echo base_url().'admin/mobil_edit/'.$m-&gt;mobil_id; ?&gt;"&gt;&lt;span class="glyphicon glyphicon-plus"&gt;
&lt;/span&gt; Edit&lt;/a&gt;</pre
```

Buat function baru lagi dengan nama mobil\_edit() dalam controller admin.php.

## ***Application/controller/Admin.php***

```
function mobil_edit($id) {
    $where = array(
        'mobil_id' => $id
    );
    $data['mobil'] = $this->m_rental->edit_data($where, 'mobil')->result();

    $this->load->view('admin/header');
    $this->load->view('admin/mobil_edit', $data);
    $this->load->view('admin/footer');
}
```

Di sini kita menangkap data id mobil yang ingin di edit di dalam parameter function mobil\_edit(). Kemudian kita ambil data mobil yang memiliki id tersebut, kemudian kita parsing ke view mobil\_edit.php. karena pada view mobil\_edit.php inilah akan kita buat form untuk mengedit data mobil.

## **Application/view/admin/mobil\_edit.php**

```
<div class="page-header">
    <h3>Edit Mobil</h3>
</div>

<?php foreach($mobil as $m) { ?>
<form action="<?php echo base_url().'admin/mobil_update' ?>" method="post">
    <div class="form-group">
        <label>Merk Mobil</label>
        <input type="hidden" name="id" value="<?php echo $m->mobil_id; ?>">
        <input class="form-control" type="text" name="merk" value="<?php echo
$m->mobil_merk; ?>">
        <?php echo form_error('merk'); ?>
    </div>

    <div class="form-group">
        <label>No. Plat Kendaraan</label>
        <input class="form-control" type="text" name="plat" value="<?php echo
$m->mobil_plat; ?>">
    </div>

    <div class="form-group">
        <label>Warna</label>
        <input class="form-control" type="text" name="warna" value="<?php echo
$m->mobil_warna; ?>">
    </div>

    <div class="form-group">
        <label>Tahun Kendaraan</label>
        <input class="form-control" type="text" name="tahun" value="<?php echo
$m->mobil_tahun; ?>">
    </div>

    <div class="form-group">
        <label>Status Mobil</label>
        <select name="status" class="form-control">
            <option <?php if($m->mobil_status == "1") {echo "selected='selected'"; }>
echo $m->mobil_tahun; ?> value="1">Tersedia</option>
            <option <?php if($m->mobil_status == "2") {echo "selected='selected'"; }>
echo $m->mobil_tahun; ?> value="2">Sedang Di Rental</option>
        </select>
        <?php echo form_error('status'); ?>
    </div>

    <div class="form-group">
        <input type="submit" value="Simpan" class="btn btn-primary">
    </div>
</form>
<?php } ?>
```

Form edit data mobil ini akan di proses pada function mobil\_update() dalam controller admin.php pada saat di submit.

```
<form action="<?php echo base_url().'admin/mobil_update' ?>" method="post">
```

Jadi buat lagi sebauh function baru dengan nama mobil\_update() pada controller Admin.php

## ***Application/controller/Admin.php***

```
function mobil_update(){
    $id = $this->input->post('id');
    $merk = $this->input->post('merk');
    $plat = $this->input->post('plat');
    $warna = $this->input->post('warna');
    $tahun = $this->input->post('tahun');
    $status = $this->input->post('status');
    $this->form_validation->set_rules('merk','Merk Mobil','required');
    $this->form_validation->set_rules('status','Status Mobil','required');
    if($this->form_validation->run() != false){
        $where = array(
            'mobil_id' => $id
        );
        $data = array(
            'mobil_merk' => $merk,
            'mobil_plat' => $plat,
            'mobil_warna' => $warna,
            'mobil_tahun' => $tahun,
            'mobil_status' => $status
        );
        $this->m_rental->update_data($where,$data,'mobil');
        redirect(base_url().'admin/mobil');
    }else{
        $where = array(
            'mobil_id' => $id
        );
        $data['mobil'] = $this->m_rental->edit_data($where,'mobil')->result();
        $this->load->view('admin/header');
        $this->load->view('admin/mobil_edit',$data);
        $this->load->view('admin/footer');
    }
}
```

## PENJELASAN !!!

Penjelasan tentang update data di database dengan codeigniter, bisa teman-teman baca kembali pada pembahasan membuat crud di tutorial codeigniter bagian dasar sebelumnya. Karena intinya sama saja.

Sekarang kita akan mencoba mengubah/mengedit data mobil “toyota avanza” ke “honda jazz”. Klik pada tombol edit

The screenshot shows a web application interface for managing vehicle data. At the top, there's a navigation bar with links for 'Rental Mobil', 'Dashboard', 'Data Mobil', 'Data Kostumer', 'Transaksi Rental', 'Laporan', 'Logout', and a greeting 'Halo, Diki Alfarabi Hadi'. Below the navigation is a search bar and a dropdown for 'records per page' set to 10. The main area is titled 'Data Mobil' and contains a table with one row. The table columns are labeled 'No', 'Merk Mobil', 'Plat', 'Warna', 'Tahun Pembuatan', and 'Status'. The first row shows data for a Toyota Avanza with license plate B 7777 XX, black color, and year 2015. The status is 'Tersedia'. To the right of the table are two buttons: a blue 'Edit' button and a red 'Hapus' (Delete) button. A blue arrow points from the text 'Klik pada tombol edit' to the 'Edit' button. Below the table, it says 'Showing 1 to 1 of 1 entries' and includes navigation links for 'Previous', '1', and 'Next'.

Maka akan di alihkan ke halaman edit mobil.

The screenshot shows the 'Edit Mobil' page. The title 'Edit Mobil' is at the top. The page contains several input fields: 'Merk Mobil' with value 'Toyota Avanza', 'No. Plat Kendaraan' with value 'B 7777 XX', 'Warna' with value 'Hitam Metalik', 'Tahun Kendaraan' with value '2015', and 'Status Mobil' with value 'Tersedia'. There is also a dropdown menu next to 'Status Mobil'. At the bottom left is a blue 'Simpan' (Save) button.

Lalu ubah merk mobil dengan “honda jazz”. Dan klik simpan.

# PENJELASAN !!!

The screenshot shows a web-based administration interface for a car rental system. The top navigation bar includes links for 'Rental Mobil', 'Dashboard', 'Data Mobil', 'Data Kostumer', 'Transaksi Rental', 'Laporan', 'Logout', and a welcome message for 'Diki Alfarabi Hadi'. The main content area is titled 'Data Mobil' and contains a table with the following data:

No	Merk Mobil	Plat	Warna	Tahun Pembuatan	Status	Actions
1	Honda Jazz	B 7777 XX	Hitam Metalik	2015	Tersedia	<a href="#">Edit</a> <a href="#">Hapus</a>

Below the table, a message indicates 'Showing 1 to 1 of 1 entries'. The 'Edit' button for the first row is highlighted with a yellow box, and a blue arrow points to the 'Honda Jazz' entry in the 'Merk Mobil' column.

Dan data pun berhasil di edit. Dan yang terakhir kita akan membuat hapus data dari database ketika tombol hapus di klik.

# Menghapus Data Mobil dari Database

Sesuai dengan link yang sudah kita tentukan pada tombol hapus di table data mobil seperti di atas.

```
<a class="btn btn-danger btn-sm" href="php echo base_url().'admin/mobil_hapus/'.$m-&gt;mobil_id; ?&gt;"></span> Hapus</a>
```

Untuk aksi dari penghasan data mobil kita akan membuatnya di function/method mobil\_hapus() dalam controller admin.

Buat sebuah method baru dengan nama mobil\_hapus() di controller admin.php.

## ***Application/controllers/admin.php***

```
function mobil_hapus($id){  
    $where = array(  
        'mobil_id' => $id  
    );  
    $this->m_rental->delete_data($where, 'mobil');  
    redirect(base_url().'admin/mobil');  
}  
// AKHIR CRUD MOBIL
```

## **PENJELASAN !!!**

Untuk lebih detail tentang menghapus data dari database menggunakan codeigniter, silahkan teman-teman baca kembali pada tutorial CRUD sebelumnya di buku ini. Yang sudah di jelaskan secara detail.

Sekarang coba klik tombol hapus pada mobil yang ingin di hapus.

The screenshot shows a web browser displaying a 'Data Mobil' (Car Data) page. The URL in the address bar is 'localhost/rental\_mobil/admin/mobil'. The page has a navigation menu at the top with items: Rental Mobil, Dashboard, Data Mobil, Data Kostumer, Transaksi Rental, Laporan, Logout, and a welcome message 'Halo, Diki Alfarabi Hadi'. Below the menu is a table titled 'Data Mobil' with the following columns: No, Merk Mobil, Plat, Warna, Tahun Pembuatan, Status, Edit, and Hapus. There is one entry in the table: '1' (No), 'Honda Jazz' (Merk Mobil), 'B 7777 XX' (Plat), 'Hitam Metalik' (Warna), '2015' (Tahun Pembuatan), 'Tersedia' (Status), '+ Edit' (Edit button), and a red-bordered 'Hapus' (Delete button). A blue arrow points to the 'Hapus' button. At the bottom of the table, there is a pagination bar with links for 'Previous', '1', and 'Next'.

## PENJELASAN !!!

Dan mobil pun berhasil di hapus.

Data Mobil

+ Mobil Baru

10 records per page

Search:

No	Merk Mobil	Plat	Warna	Tahun Pembuatan	Status
No data available in table					

Showing 0 to 0 of 0 entries

← Previous Next →

proses hapus data mobil pun berhasil dan selesai

### Ingat !

CRUD adalah singkatan dari Create, Read, Update and Delete. Istilah CRUD seringdi gunakan untuk gambaran proses input data ke database, menampilkan datake database, mengedit data ke database dan menghapus data dari database.

# Membuat CRUD Data Kostumer

Setelah selesai membuat CRUD data mobil, selanjutnya kita akan membuat crud data kostumer, tujuan kita membuat crud kostumer adalah untuk menyimpan data kostumer atau pelanggan. Sesuai dengan perancangan relasi database yang telah kita buat pada gambar xxxx , proses transaksi nantinya akan kita buat dengan memilih pelanggan yang ingin menyewa mobil, kemudian memilih mobil yang ingin di sewa, baru kemudian transaksi akan di simpan.

## Menampilkan Data Kostumer dari database

Untuk menampilkan data kostumer dari database, kita akan membuat function atau method baru pada controller admin.php. function yang akan kita buat adalah kostumer().

### *Application/controller/Admin.php*

```
// CRUD KOSTUMER
function kostumer(){
    $data['kostumer'] = $this->m_rental->get_data('kostumer')->result();
    $this->load->view('admin/header');
    $this->load->view('admin/kostumer', $data);
    $this->load->view('admin/footer');
}
```

### PENJELASAN !!!

Sama saja seperti cara menampilkan data mobil dari database yang telah kita bahas sebelumnya.

Disini pertama kita mengambil data kostumer dari table ‘kostumer’.

```
$data['kostumer'] = $this->m_rental->get_data('kostumer')->result();
```

Kemudian kita masukkan ke variabel array data. Dan kemudian kita parsing ke view kostumer.php.

Selanjutnya kita buat view baru lagi dengan nama kostumer.php pada folder viewadmin.

## **Application/view/admin/kostumer.php**

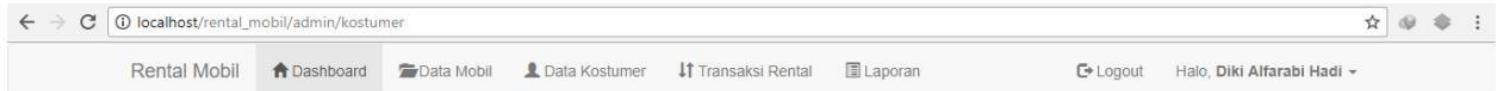
```
<h3>Data Kostumer</h3>
</div>

<a href=<?php echo base_url().'admin/kostumer_add'; ?>" class="btn btn-sm
btn-primary"><span class='glyphicon glyphicon-plus'></span> Kostumer Baru</a>
<br/>
<br/>
<div class="table-responsive">
    <table class="table table-bordered table-hover table-striped"
id="table-datable">
        <thead>
            <tr>
                <th>No</th>
                <th>>Nama</th>
                <th>Jenis Kelamin</th>
                <th>Alamat</th>
                <th>HP</th>
                <th>No. KTP</th>
                <th></th>
            </tr>
        </thead>
        <tbody>
            <?php
                $no = 1;
                foreach($kostumer as $k) {
                    ?>
                    <tr>
                        <td><?php echo $no++; ?></td>
                        <td><?php echo $k->kostumer_nama ?></td>
                        <td><?php echo $k->kostumer_jk ?></td>
                        <td><?php echo $k->kostumer_alamat ?></td>
                        <td><?php echo $k->kostumer_hp ?></td>
                        <td><?php echo $k->kostumer_ktp ?></td>
                        <td>
                            <a class="btn btn-sm btn-warning" href=<?php echo
base_url().'admin/kostumer_edit/'.$k->kostumer_id; ?>"><span class="glyphicon
glyphicon-wrench"></span> Edit</a>
                            <a class="btn btn-sm btn-danger" href=<?php echo
base_url().'admin/kostumer_hapus/'.$k->kostumer_id; ?>"><span class="glyphicon
glyphicon-trash"></span> Hapus</a>
                        </td>
                    </tr>
                    <?php
                }
                ?>
            </tbody>
        </table>
    </div>
```

## PENJELASAN !!!

Data kostumer yang kita parsing dari controller tadi kita tangkap dan menggunakan function foreach() untuk di tampilkan dengan perulangan.

```
<table class="table table-bordered table-hover table-striped" id="table-datable">
  <thead>
    <tr>
      <th>No</th>
      <th>Nama</th>
      <th>Jenis Kelamin</th>
      <th>Alamat</th>
      <th>HP</th>
      <th>No. KTP</th>
      <th></th>
    </tr>
  </thead>
  <tbody>
    <?php
      $no = 1;
      foreach($kostumer as $k){
        ?>
        <tr>
          <td><?php echo $no++; ?></td>
          <td><?php echo $k->kostumer_nama ?></td>
          <td><?php echo $k->kostumer_jl ?></td>
          <td><?php echo $k->kostumer_alamat ?></td>
          <td><?php echo $k->kostumer_hp ?></td>
          <td><?php echo $k->kostumer_ktp ?></td>
          <td>
            <a class="btn btn-sm btn-warning" href="<?php echo base_url().'admin/kostumer_edit/'.$k->kostumer_id; ?>"><span class="glyphicon glyphicon-wrench"></span> Edit</a>
            <a class="btn btn-sm btn-danger" href="<?php echo base_url().'admin/kostumer_hapus/'.$k->kostumer_id; ?>"><span class="glyphicon glyphicon-trash"></span> Hapus</a>
          </td>
        </tr>
      <?php
    }
    ?>
  </tbody>
</table>
```



### Data Kostumer



No	Nama	Jenis Kelamin	Alamat	HP	No. KTP
No data available in table					
Showing 0 to 0 of 0 entries					

Seperti yang terlihat pada gambar xx di atas, kita telah berhasil membuat tabel yang menampilkan data kostumer. Karena kita belum memiliki data kostumer di table kostumer, maka data masih kosong.

# Input Data Kostumer Ke Database

Dan seperti yang telah teman-teman lihat pada syntax view kostumer.php di atas. kita telah membuat juga tombol untuk mengalihkan halaman ke method kostumer\_add() di controller Admin.php, karena pada method nantinya akan kita buat halaman untuk input data kostumer ke database. Yaitu tombol “kostumer baru”.

```
<a href="<?php echo base_url().'admin/kostumer_add'; ?>" class="btn btn-sm btn-primary"><span class='glyphicon glyphicon-plus'></span> Kostumer Baru</a>
```

Maka dari itu, akan kita buat method baru yaitu kostumer\_add() pada controller Admin.php

## ***Application/controller/Admin.php***

```
function kostumer_add() {
    $this->load->view('admin/header');
    $this->load->view('admin/kostumer_add');
    $this->load->view('admin/footer');
}
```

Pada method atau function kostumer\_add() ini kita hanya menampilkan view kostumer\_add.php. pada view kostumer\_add.php ini kita akan membuat form input data kostumer.

Buat view baru dalam folder admin. Dengan nama kostumer\_add.php

## **Application/view/admin/kostumer\_add.php**

```
<div class="page-header">
    <h3>Kostumer Baru</h3>
</div>

<form action=<?php echo base_url().'admin/kostumer_add_act' ?>" method="post">
    <div class="form-group">
        <label>Nama Kostumer</label>
        <input type="text" name="nama" class="form-control">
        <?php echo form_error('nama'); ?>
    </div>
    <div class="form-group">
        <label>Alamat</label>
        <textarea name="alamat" class="form-control"></textarea>
    </div>
    <div class="form-group">
        <label>Jenis Kelamin</label>
        <select name="jk" class="form-control">
            <option value="L">Laki-laki</option>
            <option value="P">Perempuan</option>
        </select>
    </div>
    <div class="form-group">
        <label>HP</label>
        <input type="number" name="hp" class="form-control">
    </div>
    <div class="form-group">
        <label>No. KTP</label>
        <input type="text" name="ktp" class="form-control">
    </div>
    <div class="form-group">
        <input type="submit" value="Simpan" class="btn btn-primary">
    </div>
</form>
```

## **PENJELASAN !!!**

The screenshot shows a web application interface for adding a new customer. The URL in the address bar is `localhost/rental_mobil/admin/kostumer_add`. The page title is "Kostumer Baru". The form consists of several input fields:

- Nama Kostumer**: A text input field.
- Alamat**: A textarea for address input.
- Jenis Kelamin**: A dropdown menu with options "Laki-laki" and "Perempuan", currently showing "Laki-laki".
- HP**: A text input field for phone number.
- No. KTP**: A text input field for ID card number.

At the bottom right of the form is a blue "Simpan" (Save) button.

## PENJELASAN !!!

Form input data kostumer ini akan di proses pada method kostumer\_add\_act() sesuai dengan yang sudah kita setting pada form di atas.

```
<form action="php echo base_url().'admin/kostumer_add_act' ?&gt;" method="post"&gt;</pre
```

Maka sekarang buat sebuah method baru lagi dengan nama costumer\_add\_act() pada controller Admin.php.

### *Application/controllers/Admin.php*

```
function kostumer_add_act(){
    $nama = $this->input->post('nama');
    $alamat = $this->input->post('alamat');
    $jk = $this->input->post('jk');
    $hp = $this->input->post('hp');
    $ktp = $this->input->post('ktp');
    $this->form_validation->set_rules('nama','nama','required');

    if($this->form_validation->run() != false){
        $data = array(
            'kostumer_nama' => $nama,
            'kostumer_alamat' => $alamat,
            'kostumer_jk' => $jk,
            'kostumer_hp' => $hp,
            'kostumer_ktp' => $ktp
        );
        $this->m_rental->insert_data($data,'kostumer');
        redirect(base_url().'admin/kostumer');
    }else{
        $this->load->view('admin/header');
        $this->load->view('admin/kostumer_add');
        $this->load->view('admin/footer');
    }
}
```

## PENJELASAN !!!

Untuk proses aksi dari penginputan data kostumer ini sama saja intinya seperti proses aksi penginputan data mobil. Bedanya data mobil kita simpan ke table ‘mobil’. Dan data kostumer kita simpan ke table ‘kostumer’. Setelah proses input data selesai, halaman akan langsung kita alihkan ke function kostumer() di controller admin.php.

## PENJELASAN !!!

```
redirect(base_url().'admin/kostumer');
```

Sekarang akan kita coba menginput data kostumer. Silahkan klik pada tombol “kostumer baru”. Lalu isi data kostumer dan klik “simpan”.

Kostumer Baru

Nama Kostumer  
Diki Alfarabi Hadi

Alamat  
Jl. planet antariksa sejati. no. 123, bumi, indonesia.

Jenis Kelamin  
Laki-laki

HP  
082277125900

No. KTP  
3294729834792

**Simpan**

Setelah di klik pada tombol “simpan”. Maka data akan tersimpan. Di sini saya mencoba mengisi nama kostumer dengan nama saya sendiri. “Diki AlfarabiHadi”. Alamat “jl. Planet antariksa sejati no.123, bumi, indonesia”. Jenis kelamin “laki-laki”, hp “082277125900”. No.KTP “3294729834792”

Data Kostumer

+ Kostumer Baru

10 records per page

No	Nama	Jenis Kelamin	Alamat	HP	No. KTP	
1	Diki Alfarabi Hadi	L	Jl. planet antariksa sejati. no. 123, bumi, indonesia.	082277125900	3294729834792	<b>Edit</b> <b>Hapus</b>

Showing 1 to 1 of 1 entries

← Previous 1 Next →

Dan data kostumer berhasil di simpan.

# Update Data Kostumer Di Database

Untuk membuat edit dan update data kostumer, kita akan membuat function kostumer\_edit() di controller Admin.php. seperti yang sudah kita tetapkan pada tombol edit di tabel yang menampilkan data kostumer seperti di atas.

```
<a class="btn btn-sm btn-warning" href="php echo base_url().'admin/kostumer_edit/'.$k-&gt;kostumer_id; ?&gt;"&gt;&lt;span class="glyphicon glyphicon-wrench"&gt;&lt;/span&gt; Edit&lt;/a&gt;</pre
```

Buat sebuah method baru dengan nama kostumer\_edit().

## *Application/controller/Admin.php*

```
function kostumer_edit($id) {
    $where = array(
        'kostumer_id' => $id
    );
    $data['kostumer'] =
$this->m_rental->edit_data($where, 'kostumer')->result();
    $this->load->view('admin/header');
    $this->load->view('admin/kostumer_edit', $data);
    $this->load->view('admin/footer');
}
```

Di sini kita menangkap data id yang dikirimkan dari url. Kita menangkapnya dalam function kostumer\_id(). Kemudian kita ambil data kostumer yang memiliki id sesuai dengan id yang kita tangkap. Kemudian kita parsing data nya ke view kostumer\_edit.php di folder admin.

Buat view baru dengan nama kostumer\_edit.php di dalam folder admin.

## **Application/view/admin/kostumer\_edit.php**

```
<div class="page-header">
    <h3>Edit Kostumer</h3>
</div>

<?php foreach($kostumer as $k) { ?>
<form action=<?php echo base_url().'admin/kostumer_update' ?>" method="post">
    <div class="form-group">
        <label>Nama Kostumer</label>
        <input type="text" name="nama" value=<?php echo $k->kostumer_nama; ?>" class="form-control">
        <input type="hidden" name="id" value=<?php echo $k->kostumer_id; ?>" class="form-control">
        <?php echo form_error('nama'); ?>
    </div>
    <div class="form-group">
        <label>Alamat</label>
        <textarea name="alamat" class="form-control"><?php echo $k->kostumer_alamat; ?></textarea>
    </div>
    <div class="form-group">
        <label>Jenis Kelamin</label>
        <select name="jk" class="form-control">
            <option value="L">Laki-laki</option>
            <option value="P">Perempuan</option>
        </select>
    </div>
    <div class="form-group">
        <label>HP</label>
        <input type="number" name="hp" value=<?php echo $k->kostumer_hp; ?>" class="form-control">
    </div>
    <div class="form-group">
        <label>No. KTP</label>
        <input type="text" name="ktp" value=<?php echo $k->kostumer_ktp; ?>" class="form-control">
    </div>
    <div class="form-group">
        <input type="submit" value="Simpan" class="btn btn-primary">
    </div>
</form>
<?php } ?>
```

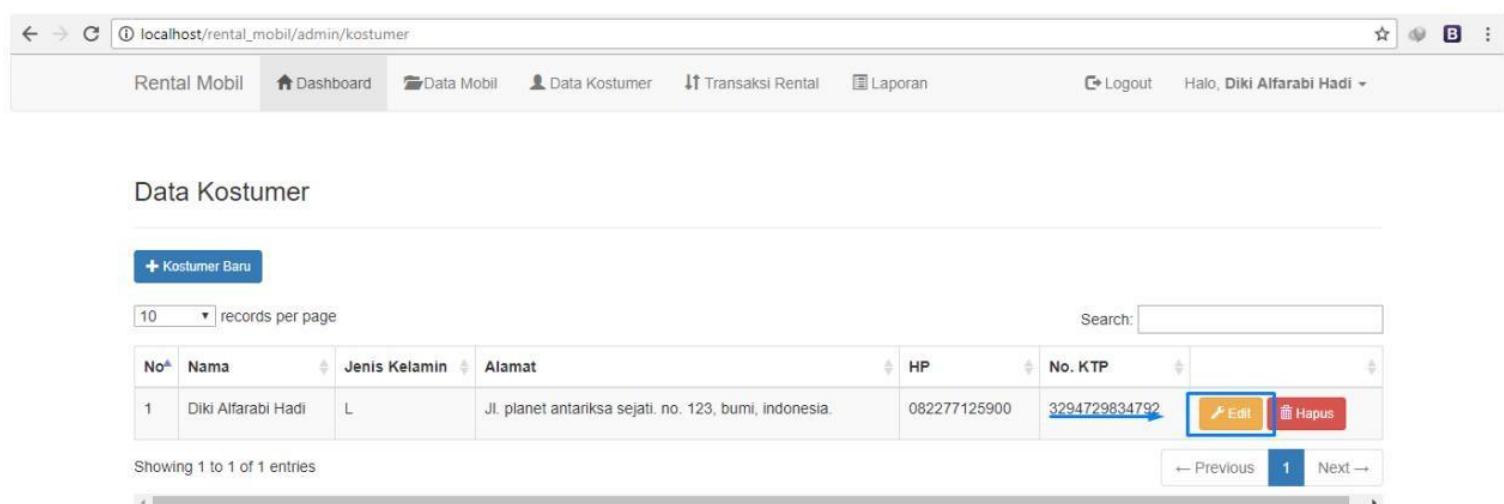
## **Application/controller/Admin.php**

```
function kostumer_update(){
    $id = $this->input->post('id');
    $nama = $this->input->post('nama');
    $alamat = $this->input->post('alamat');
    $jk = $this->input->post('jk');
    $hp = $this->input->post('hp');
    $ktp = $this->input->post('ktp');
    $this->form_validation->set_rules('nama','nama','required');

    if($this->form_validation->run() != false) {
        $where = array(
            'kostumer_id' => $id
        );
        $data = array(
            'kostumer_nama' => $nama,
            'kostumer_alamat' => $alamat,
            'kostumer_jk' => $jk,
            'kostumer_hp' => $hp,
            'kostumer_ktp' => $ktp
        );
        $this->m_rental->update_data($where,$data,'kostumer');
        redirect(base_url().'admin/kostumer');
    }else{
        $where = array(
            'kostumer_id' => $id
        );
        $data['kostumer'] =
    }
}
```

## **PENJELASAN !!!**

Dan coba sekarang kita klik pada tombol edit untuk mengubah data kostumer.



The screenshot shows a web browser window with the URL `localhost/rental_mobil/admin/kostumer`. The page title is "Data Kostumer". At the top, there is a navigation bar with links: "Rental Mobil", "Dashboard", "Data Mobil", "Data Kostumer", "Transaksi Rental", "Laporan", "Logout", and a user profile "Halo, Diki Alfarabi Hadi". Below the navigation bar, there is a search bar labeled "Search: [ ]".  
  
The main content area displays a table titled "Data Kostumer" with the following columns: No, Nama, Jenis Kelamin, Alamat, HP, and No. KTP. There is one record listed:  

No	Nama	Jenis Kelamin	Alamat	HP	No. KTP
1	Diki Alfarabi Hadi	L	Jl. planet antarika sejati. no. 123, bumi, indonesia.	082277125900	3294729834792

  
At the bottom of the table, there are buttons for "Edit" and "Hapus". The "Edit" button is highlighted with a yellow box. Below the table, there is a message "Showing 1 to 1 of 1 entries" and a set of navigation buttons for "Previous", "Next", and page number "1".  
  
A legend at the bottom right of the table indicates that the "Edit" button is represented by a pencil icon and the "Hapus" button is represented by a trash bin icon.

Maka kita akan di alihkan ke form edit kostumer.

# PENJELASAN !!!

A screenshot of a web application interface titled "Edit Kostumer". The page has a header with navigation links: "Rental Mobil", "Dashboard", "Data Mobil", "Data Kostumer", "Transaksi Rental", "Laporan", "Logout", and a user greeting "Halo, Diki Alfarabi Hadi". The main content area contains fields for "Nama Kostumer" (Diki Alfarabi Hadi), "Alamat" (Jl. planet antarika sejati. no. 123, bumi, indonesia.), "Jenis Kelamin" (Laki-laki), "HP" (082277125900), and "No. KTP" (3294729834792). A blue "Simpan" button is at the bottom.

Di sini kita akan mencoba mengubah alamat dan nomor ktp dari kostumer.

A screenshot of the same web application interface, showing the "Edit Kostumer" form after changes have been made. The "Alamat" field now contains "Jl. bumi alami. no. 123, indonesia.". The "No. KTP" field now contains "123456789". All other fields remain the same as in the first screenshot. The "Simpan" button is visible at the bottom.

Di sini saya mencoba merubah alamat kostumer menjadi “Jl. bumi alami. no. 123, indonesia.”, dan mengubah nomor KTP kostumer menjadi “123456789”. Lalu klik simpan.

# PENJELASAN !!!

The screenshot shows a web-based administration interface for a car rental system. The URL in the browser is `localhost/rental_mobil/admin/kostumer`. The top navigation bar includes links for Rental Mobil, Dashboard, Data Mobil, Data Kostumer (which is currently selected), Transaksi Rental, Laporan, Logout, and a user greeting "Halo, Diki Alfarabi Hadi".

The main content area is titled "Data Kostumer" and displays a table of customer records. A blue button labeled "+ Kostumer Baru" is visible at the top left of the table area. The table has columns: No, Nama, Jenis Kelamin, Alamat, HP, and No. KTP. The first record in the table is highlighted with a blue border and contains the following data:

No	Nama	Jenis Kelamin	Alamat	HP	No. KTP
1	Diki Alfarabi Hadi	L	Jl. bumi alami. no. 123, indonesia.	082277125900	123456789

Below the table, there are buttons for "Edit" (yellow) and "Hapus" (red). At the bottom of the page, there is a footer with links for "← Previous", the page number "1", and "Next →".

Dan data kostumer pun berhasil di update.

# Menghapus Data Kostumer dari Database

Dan terakhir untuk crud data kostumer, kita akan membuat fungsi untuk hapus data kostumer.

Coba teman-teman perhatikan pada tombol hapus pada table data kostumer. Saat tombol hapus di klik maka akan di alihkan ke function kostumer\_hapus() di controller Admin sambil membawa id kostumer yang akan di hapus.

```
<a class="btn btn-sm btn-danger" href="#"><?php echo base_url().'admin/kostumer_hapus/'.$k->kostumer_id; ?>><span class="glyphicon glyphicon-trash"></span> Hapus</a>
```

Jadi aksi penghapusan data kostumer ini akan kita buat di function/method kostumer\_hapus() di controller admin.

Maka kita harus membuat method baru di controller admin. Yaitu method kostumer\_hapus().

## ***Application/controllers/admin.php***

```
function kostumer_hapus($id) {
    $where = array(
        'kostumer_id' => $id
    );
    $this->m_rental->delete_data($where, 'kostumer');
    redirect(base_url().'admin/kostumer');
}
// AKHIR CRUD MOBIL
```

## **PENJELASAN !!!**

Di sini kita tanggap id kostumer yang akan di hapus, kemudian kita masukkan ke variabel array where. Untuk kita seleksi data yang akan kita hapus. Setelah data di hapus maka halaman akan di alihkan ke method kostumer().

Sebenarnya cara menghapus data kostumer masih sama saja dengan cara menghapus data mobil seperti yang sudah kita buat sebelumnya.

Coba klik tombol hapus.

# PENJELASAN !!!

The screenshot shows a table with one row of data. The columns are labeled: No, Nama, Jenis Kelamin, Alamat, HP, and No. KTP. The data in the first row is: 1, Diki Alfarabi Hadi, L, Jl. bumi alami. no. 123, indonesia., 082277125900, 123456789. To the right of the table is a red rectangular button with a white icon and the word "Hapus". A blue arrow points from the text "dan data kostumer pun berhasil dihapus." to this button.

No	Nama	Jenis Kelamin	Alamat	HP	No. KTP
1	Diki Alfarabi Hadi	L	Jl. bumi alami. no. 123, indonesia.	082277125900	123456789

Showing 1 to 1 of 1 entries

← Previous 1 Next →

dan data kostumer pun berhasil dihapus.

The screenshot shows an empty table with the same columns as the previous one. The message "No data available in table" is displayed below the table. The "Hapus" button is still present on the right.

No	Nama	Jenis Kelamin	Alamat	HP	No. KTP
No data available in table					

Showing 0 to 0 of 0 entries

← Previous Next →

Oke selesai sudah proses CRUD untuk kostumer. Selanjutnya kita akan lanjut ke modul transaksi rental mobil.

# Membuat Proses Transaksi Rental Mobil

Setelah selesai membuat CRUD data mobil dan CRUD untuk data kostumer. Selanjutnya kita akan lanjut membuat proses transaksi rental mobil. Silahkan perhatikan kembali perancangan table transaksi yang telah kita buat.

transaksi	
-transaksi_id	*
-transaksi_karyawan	*
-transaksi_kostumer	*
-transaksi_mobil	*
-transaksi_tgl_pinjam	*
-transaksi_tgl_kembali	*
-transaksi_harga	*
-transaksi_denda	*
-transaksi_tgl	*
-transaksi_totaldenda	*
-transaksi_status	*
-transaksi_tgldikembalikan	*

Seperti yang sudah di singgung sebelumnya. Pada table transaksi ini kita akan menyimpan data id karyawan yang menangani transaksi di kolumn **transaksi\_karyawan**, menyimpan data id kostumer yang ingin menyewa mobil di kolumn **transaksi\_kostumer**. Selengkapnya silahkan perhatikan pada table berikut.

<b>transaksi_karyawan</b>	Menyimpan data id karyawan yang menangani transaksi sewa/rental mobil.
<b>Transaksi_kostumer</b>	Menyimpan id kostumer yang melakukan rental mobil.
<b>transaksi_mobil</b>	Menyimpan Id mobil yang di rental.
<b>transaksi_tgl_pinjam</b>	Menyimpan data tanggal mobil mulai di sewa.
<b>transaksi_thl_kembali</b>	Menyimpan data tanggal mobil akan di kembalikan
<b>transaksi_harga</b>	Menyimpan harga rental mobil.
<b>transaksi_denda</b>	Menyimpan harga denda mobil perhari.
<b>transaksi_tgl</b>	Menyimpan data tanggal transaksi di lakukan.
<b>transaksi_totaldenda</b>	Menyimpan total denda untuk kostumer yang telat mengembalikan mobil.
<b>transaksi_status</b>	Menyimpan status transaksi. 0 = belum selesai (dalam masa rental). 1 = selesai
<b>transaksi_tgl_dikembalikan</b>	Menyimpan tanggal di kembalikannya mobil oleh kostumer.

## Menampilkan Data Transaksi Rental

Pertama kita akan menampilkan dulu data transaksi. Kita akan menampilkan data transaksi di method transaksi(). Dalam controller Admin.php. Buat sebuah method baru pada controller Admin.php. yaitu method transaksi().

### *Application/controllers/Admin.php*

```
// CRUD transaksi
function transaksi() {
    $data['transaksi'] = $this->db->query("select * from
transaksi,mobil,kostumer where transaksi_mobil=mobil_id and
transaksi_kostumer=kostumer_id")->result();
    $this->load->view('admin/header');
    $this->load->view('admin/transaksi', $data);
    $this->load->view('admin/footer');
}
```

Query mysql yang kita jalankan di sini adalah “select \* from transaksi,mobil,kostumer where transaksi\_mobil=mobil\_id and transaksi\_kostumer=kostumer\_id ”. yaitu untuk menampilkan atau mengambil data dari table transaksi, table mobil dan table kostumer yang mobil\_id sama dengan transaksi\_mobil. Dan kostumer\_id sama dengan transaksi\_kostumer. Setelah kita mendapatkan data kemudian kita parsing ke view transaksi.php untuk di tampilkan.

Buat view baru dalam folder admin dengan nama transksi.php

## Application/view/admin/transaksi.php

```
<div class="page-header">
    <h3>Data Transaksi</h3>
</div>

<a href=<?php echo base_url().'admin/transaksi_add'; ?>" class="btn btn-primary btn-sm"><span class="glyphicon glyphicon-plus"></span> Transaksi Baru</a>
<br/><br/>
<div class="table-responsive">
    <table class="table table-bordered table-striped table-hover" id="table-datable">
        <thead>
            <tr>
                <th>No</th>
                <th>Tanggal</th>
                <th>Kostumer</th>
                <th>Mobil</th>
                <th>Tgl. Pinjam</th>
                <th>Tgl. Kembali</th>
                <th>Harga</th>
                <th>Denda / Hari</th>
                <th>Tgl. Dikembalikan</th>
                <th>Total Denda</th>
                <th>Status</th>
                <th></th>
            </tr>
        </thead>
        <tbody>
            <?php
            $no = 1; foreach($transaksi
            as $t){
                ?>
                <tr>
                    <td><?php echo $no++; ?></td>
                    <td><?php echo date('d/m/Y',strtotime($t->transaksi_tgl)); ?></td>
                    <td><?php echo $t->kostumer_nama; ?></td>
                    <td><?php echo $t->mobil_merk; ?></td>
                    <td><?php echo date('d/m/Y',strtotime($t->transaksi_tgl_pinjam)); ?></td>
                    <td><?php echo date('d/m/Y',strtotime($t->transaksi_tgl_kembali)); ?></td>
                    <td><?php echo "Rp. ".number_format($t->transaksi_harga); ?></td>
                    <td><?php echo "Rp. ".number_format($t->transaksi_denda); ?></td>
                    <td>
                        <?php
                        if($t->transaksi_tgldikembalikan == "0000-00-00"){echo "-";
                        ";
                        }else{
                            echo date('d/m/Y',strtotime($t->transaksi_tgldikembalikan));
                        }
                        ?>
                    </td>
                    <td><?php echo "Rp. ". number_format($t->transaksi_totaldenda).", -"; ?></td>
                    <td>
                        <?php
                        if($t->transaksi_status == "1"){echo
                            "Selesai";
                        }else{
                            echo "-";
                        }
                        ?>
                    </td>
                    <td>
                        <?php
                        if($t->transaksi_status == "1"){echo
                            "-";
                        }else{
                            ?>
                            <a class="btn btn-sm btn-success" href=<?php echo base_url().'admin/transaksi_selesai/'.$t-
                        >transaksi_id; ?>><span class="glyphicon glyphicon-ok"></span> TransaksiSelesai</a>
                            <br/>
                            <a class="btn btn-sm btn-danger" href=<?php echo
                        base_url().'admin/transaksi_hapus/'.$t->transaksi_id; ?>><span class="glyphicon glyphicon-remove"></span> BatalTransaksi</a>
                            <?php
                        }
                        ?>
                    </td>
                </tr>
            <?php
            }
            ?>
        </tbody>
    </table>
</div>
```

# PENJELASAN !!!

Hasilnya seperti gambar berikut ini.

Pada table data transaksi di atas, kita telah berhasil menampilkan data transaksi. Namun masih kosong karena belum ada data transaksi yang kita input.

Oh ya, kita juga membuat kondisi, jika status transaksi nantinya sama dengan 1. Maka kita tampilkan “selesai” pada kolumn status di tabel transaksi di atas. dan pada kolumn terakhir kita membuat kondisi juga, jika status transaksi belum selesai, maka kita tampilkan tombol “batalkan transaksi” dan tombol“transaksi selesai”.

```
<td>
<?php
if($t->transaksi_status == "1"){
    echo "Selesai";
} else{
    echo "-";
}
?>
</td>
<td>
<?php
if($t->transaksi_status == "1"){
    echo "-";
} else{
    ?>
        <a class="btn btn-sm btn-success" href="<?php echo base_url().'admin/transaksi_selesai/'.$t->transaksi_id; ?>"><span class="glyphicon glyphicon-ok"></span> Transaksi Selesai</a>
        <br/>
        <a class="btn btn-sm btn-danger" href="<?php echo base_url().'admin/transaksi_hapus/'.$t->transaksi_id; ?>"><span class="glyphicon glyphicon-remove"></span> Batalkan Transaksi</a>
    <?php
}
?>
</td>
```

# Menginput Data Transaksi Rental Mobil

Sekarang saya telah menginput data mobil dan beberapa data kostumer. Sebagai percobaan kita membuat proses transaksi. Silahkan teman-teman input beberapa data mobil dan data kostumer. Untuk data percobaan kita membuat proses transaksi rental mobil.

+ Mobil Baru

10 records per page

Search:

No	Merk Mobil	Plat	Warna	Tahun Pembuatan	Status	
1	Toyota Avanza	B 7777 xx	Hitam Metalik	2018	Tersedia	<button>+ Edit</button> <button>Hapus</button>
2	Honda Brio	B 1111 AA	Putih Mutiara	2016	Tersedia	<button>+ Edit</button> <button>Hapus</button>
3	Toyota Fortuner	BK 2222 XX	Hitam Metalik	2017	Tersedia	<button>+ Edit</button> <button>Hapus</button>
4	Suzuki Swift	B 2333 B	Putih	2016	Tersedia	<button>+ Edit</button> <button>Hapus</button>
5	Toyota Innova	B 1222 BB	Hitam	2015	Tersedia	<button>+ Edit</button> <button>Hapus</button>
6	Toyota Agya	B 6666 B	Putih	2017	Tersedia	<button>+ Edit</button> <button>Hapus</button>
7	Daihatsu Xenia	B 6666 GG	Hitam	2017	Tersedia	<button>+ Edit</button> <button>Hapus</button>
8	Honda HR-V	B 123 NN	Hitam Metalik	2017	Tersedia	<button>+ Edit</button> <button>Hapus</button>
9	Suzuki Ertiga	BG 7777 HG	Putih	2017	Tersedia	<button>+ Edit</button> <button>Hapus</button>
10	Daihatsu Sigra	B 7767 B	Biru Muda	2017	Tersedia	<button>+ Edit</button> <button>Hapus</button>

Showing 1 to 10 of 10 entries

← Previous 1 Next →

## Data Kostumer

+ Kostumer Baru

10 records per page

Search:

No*	Nama	Jenis Kelamin	Alamat	HP	No. KTP	
1	Zainal Abidin	L	Jl. gagak putih no 627, medan , Indonesia	082737373787	998798798798	<button>Edit</button> <button>Hapus</button>
2	Marlon Suwangan	L	Jl. supapa no.23 , bekasi, Indonesia	081234554443	434534343434	<button>Edit</button> <button>Hapus</button>
3	Muhammad Zainal	L	Jl. merpati putih 2- no.66 , Indonesia	081212121212	232323232330	<button>Edit</button> <button>Hapus</button>
4	Ratna Dewi	L	Jl. ampera selatan. no. 67 , jawa barat, indonesia	081112122219	232231131313	<button>Edit</button> <button>Hapus</button>
5	Samsul Bahri	L	Jl. peletok. jawa timur. Indonesia	081212221221	876876876786	<button>Edit</button> <button>Hapus</button>
6	Diki Alfarabi Hadi	L	Jl. badak utara. no.123. aceh, indonesia	081221222212	876876876876	<button>Edit</button> <button>Hapus</button>

Showing 1 to 6 of 6 entries

← Previous 1 Next →

Kita akan melanjutkan proses selanjutnya setelah memiliki beberapa data mobil dan data kostumer.

Selanjutnya kita akan membuat penginputan transaksi rental mobil. Penginputan transaksi akan kita buat pada method transaksi\_add(). Langsung saja kita buat method transaksi\_add() di controller Admin.php.

### ***Application/controllers/Admin.php***

```
function transaksi_add(){
    $w = array('mobil_status'=>'1');
    $data['mobil'] = $this->m_rental->edit_data($w, 'mobil')->result();
    $data['kostumer'] = $this->m_rental->get_data('kostumer')->result();
    $this->load->view('admin/header');
    $this->load->view('admin/transaksi_add', $data);
    $this->load->view('admin/footer');
}
```

### **PENJELASAN !!!**

Seperti yang teman-teman lihat pada method transaksi\_add di atas. di sini kita akan membuat form input transaksi nya di view transaksi\_add.php. dan juga mengirimkan(parsing) 2 data ke view transaksi\_add.php. yaitu data mobil dan data kostumer.

```
$data['mobil'] = $this->m_rental->edit_data($w, 'mobil')->result();
$data['kostumer'] = $this->m_rental->get_data('kostumer')->result();
```

Kemudian buat view nya dengan nama transaksi\_add.php. seperti yang telah kita tetapkan pada method transaksi\_add.php.

## **Application/views/admin/transaksi\_add.php**

```
<div class="page-header">
    <h3>Transaksi Baru</h3>
</div>

<form action="=base_url('admin/transaksi_add_act') ?" method="post">
    <div class="form-group">
        <label>Kostumer</label>
        <select name="kostumer" class="form-control">
            <option value="">-Pilih Kostumer-</option>
            <?php foreach($kostumer as $k){ ?>
                <option value="=echo $k-&gt;kostumer_id; ?&gt;"&gt;&lt;?php echo
$ k-&gt;kostumer_nama; ?&gt;&lt;/option&gt;
            &lt;?php } ?&gt;
        &lt;/select&gt;
        &lt;?php echo form_error('kostumer'); ?&gt;
    &lt;/div&gt;

    &lt;div class="form-group"&gt;
        &lt;label&gt;Mobil&lt;/label&gt;
        &lt;select name="mobil" class="form-control"&gt;
            &lt;option value=""&gt;-Pilih Mobil-&lt;/option&gt;
            &lt;?php foreach($mobil as $m){ ?&gt;
                &lt;option value="<?=echo $m-&gt;mobil_id; ?&gt;"&gt;&lt;?php echo
$ m-&gt;mobil_merk; ?&gt;&lt;/option&gt;
            &lt;?php } ?&gt;
        &lt;/select&gt;
        &lt;?php echo form_error('mobil'); ?&gt;
    &lt;/div&gt;

    &lt;div class="form-group"&gt;
        &lt;label&gt;Tanggal Pinjam&lt;/label&gt;
        &lt;input type="date" name="tgl_pinjam" class="form-control"&gt;
        &lt;?php echo form_error('tgl_pinjam'); ?&gt;
    &lt;/div&gt;

    &lt;div class="form-group"&gt;
        &lt;label&gt;Tanggal Kembali&lt;/label&gt;
        &lt;input type="date" name="tgl_kembali" class="form-control"&gt;
        &lt;?php echo form_error('tgl_kembali'); ?&gt;
    &lt;/div&gt;

    &lt;div class="form-group"&gt;
        &lt;label&gt;Harga&lt;/label&gt;
        &lt;input type="number" name="harga" class="form-control"&gt;
        &lt;?php echo form_error('harga'); ?&gt;
    &lt;/div&gt;

    &lt;div class="form-group"&gt;
        &lt;label&gt;Harga Denda / Hari&lt;/label&gt;
        &lt;input type="text" name="denda" class="form-control"&gt;
        &lt;?php echo form_error('denda'); ?&gt;
    &lt;/div&gt;
    &lt;div class="form-group"&gt;
        &lt;input type="submit" value="Simpan" class="btn btn-primary btn-sm"&gt;
    &lt;/div&gt;
&lt;/form&gt;</pre
```

## PENJELASAN !!!

Tadi pada method transaksi\_add, kita mengirimkan data mobil dan data kostumer ke view transaksi\_add.php.

Kemudian pada view transaksi\_add.php kita tampilkan data mobil pada form mobil. Dan tampilkan data kostumer pada form kostumer.

```
<div class="form-group">
    <label>Kostumer</label>
    <select name="kostumer" class="form-control">
        <option value="">-Pilih Kostumer-</option>
        <?php foreach($kostumer as $k){ ?>
            <option value="php echo $k-&gt;kostumer_id; ?&gt;"&gt;&lt;?php echo $k-&gt;kostumer_nama; ?&gt;&lt;/option&gt;
        &lt;?php } ?&gt;
    &lt;/select&gt;
    &lt;?php echo form_error('kostumer'); ?&gt;
&lt;/div&gt;

&lt;div class="form-group"&gt;
    &lt;label&gt;Mobil&lt;/label&gt;
    &lt;select name="mobil" class="form-control"&gt;
        &lt;option value=""&gt;-Pilih Mobil-&lt;/option&gt;
        &lt;?php foreach($mobil as $m){ ?&gt;
            &lt;option value="<?php echo $m-&gt;mobil_id; ?&gt;"&gt;&lt;?php echo $m-&gt;mobil_merk; ?&gt;&lt;/option&gt;
        &lt;?php } ?&gt;
    &lt;/select&gt;
    &lt;?php echo form_error('mobil'); ?&gt;
&lt;/div&gt;</pre
```

Maka hasilnya akan seperti berikut.

The screenshot shows a web application interface for adding a new transaction. The title bar indicates the URL is `localhost/rental_mobil/admin/transaksi_add`. The top navigation menu includes links for Rental Mobil, Dashboard, Data Mobil, Data Kostumer, Transaksi Rental, Laporan, Logout, and a user profile. The main content area is titled "Transaksi Baru". It contains several form fields:

- Kostumer:** A dropdown menu with the placeholder "-Pilih Kostumer-".
- Mobil:** A dropdown menu with the placeholder "-Pilih Mobil-".
- Tanggal Pinjam:** An input field with the placeholder "hh/bb/tttt".
- Tanggal Kembali:** An input field with the placeholder "hh/bb/tttt".
- Harga:** An input field with a numeric keypad overlay.
- Harga Denda / Hari:** An input field with a numeric keypad overlay.

At the bottom of the form is a blue "Simpan" button.

# PENJELASAN !!!

Data kostumer tampil pada form kostumer seperti berikut.

Kostumer

- Pilih Kostumer-
- Pilih Kostumer-
- Zainal Abidin
- Marlion Suwanggal
- Muhammad Zainal
- Ratna Dewi
- Samsul Bahri
- Diki Alfarabi Hadi
- hh/bb/ttt

Tanggal Kembali

hh/bb/ttt

Harga

Harga Denda / Hari

Simpan

Data mobil tampil pada form mobil.

Kostumer

- Pilih Kostumer-

Mobil

- Pilih Mobil-
- Pilih Mobil-
- Toyota Avanza
- Honda Brio
- Toyota Fortuner
- Suzuki Swift
- Toyota Innova
- Toyota Agya
- Daihatsu Xenia
- Honda HR-V
- Suzuki Ertiga
- Daihatsu Sigra

Harga Denda / Hari

Simpan

Selanjutnya kita akan membuat aksi dari form input transaksi ini. Aksi dari form ini akan kita buat di method `transaksi_add_act` di controller Admin.php. seperti yang sudah kita buat pada form input transaksi ini.

```
<form action="php echo base_url().'admin/transaksi_add_act' ?&gt;" method="post"&gt;</pre
```

Buat method baru dengan nama transaksi\_add\_act di controller admin.php.

## **Application/controllers/admin.php**

```
function transaksi_add_act() {
    $kostumer = $this->input->post('kostumer');
    $mobil = $this->input->post('mobil');
    $tgl_pinjam = $this->input->post('tgl_pinjam');
    $tgl_kembali = $this->input->post('tgl_kembali');
    $harga = $this->input->post('harga');
    $denda = $this->input->post('denda');

    $this->form_validation->set_rules('kostumer', 'Kostumer', 'required');

    $this->form_validation->set_rules('mobil', 'Mobil', 'required');
    $this->form_validation->set_rules('tgl_pinjam', 'Tanggal Pinjam', 'required');
    $this->form_validation->set_rules('tgl_kembali', 'Tanggal Kembali', 'required');
    $this->form_validation->set_rules('harga', 'Harga', 'required');
    $this->form_validation->set_rules('denda', 'Denda', 'required');

    if($this->form_validation->run() != false) {
        $data = array(
            'transaksi_karyawan' => $this->session->userdata('id'),
            'transaksi_kostumer' => $kostumer,
            'transaksi_mobil' => $mobil,
            'transaksi_tgl_pinjam' => $tgl_pinjam,
            'transaksi_tgl_kembali' => $tgl_kembali,
            'transaksi_harga' => $harga,
            'transaksi_denda' => $denda,
            'transaksi_tgl' => date('Y-m-d')
        );
        $this->m_rental->insert_data($data, 'transaksi');

        // update status mobil yg di pinjam
        $d = array(
            'mobil_status' => '2'
        );
        $w = array(
            'mobil_id' => $mobil
        );
        $this->m_rental->update_data($w, $d, 'mobil');
        redirect(base_url().'admin/transaksi');

    }else{
        $w = array('mobil_status'=>'1');
        $data['mobil'] = $this->m_rental->edit_data($w, 'mobil')->result();
        $data['kostumer'] = $this->m_rental->get_data('kostumer')->result();
        $this->load->view('admin/header');
        $this->load->view('admin/transaksi_add', $data);
        $this->load->view('admin/footer');
    }
}
```

## PENJELASAN !!!

Perhatikan pada action/aksi dari form transaksi di atas. pertama kita menangkap data-data yang dikirim dari form ke dalam variabel masing-masing.

```
$kostumer = $this->input->post('kostumer');
$mobil = $this->input->post('mobil');
$tgl_pinjam = $this->input->post('tgl_pinjam');
$tgl_kembali = $this->input->post('tgl_kembali');
$harga = $this->input->post('harga');
$denda = $this->input->post('denda');
```

Kemudian kita setting form validation nya. Karena kita ingin form kostumer, mobil, tanggal pinjam, tanggal kembali, harga dan denda wajib di isi(required).

```
$this->form_validation->set_rules('kostumer','Kostumer','required');
$this->form_validation->set_rules('mobil','Mobil','required');
$this->form_validation->set_rules('tgl_pinjam','Tanggal Pinjam','required');
$this->form_validation->set_rules('tgl_kembali','Tanggal Kembali','required');
$this->form_validation->set_rules('harga','Harga','required');
$this->form_validation->set_rules('denda','Denda','required');
```

Dan jika validasi sesuai, maka kita input datanya ke table transaksi. Dengan menyesuaikan data yang di input sebagai berikut

```
$data = array(
    'transaksi_karyawan' => $this->session->userdata('id'),
    'transaksi_kostumer' => $kostumer,
    'transaksi_mobil' => $mobil,
    'transaksi_tgl_pinjam' => $tgl_pinjam,
    'transaksi_tgl_kembali' => $tgl_kembali,
    'transaksi_harga' => $harga,
    'transaksi_denda' => $denda,
    'transaksi_tgl' => date('Y-m-d')
);
$this->m_rental->insert_data($data,'transaksi');
```

Tabel berikut akan menjelaskan data apa saja yang kita input di table transaksi.

<b>transaksi_karyawan</b>	Menyimpan data id karyawan yang menangani transaksi sewa/rental mobil.
<b>Transaksi_kostumer</b>	Menyimpan id kostumer yang melakukan rental mobil.
<b>transaksi_mobil</b>	Menyimpan Id mobil yang di rental.
<b>transaksi_tgl_pinjam</b>	Menyimpan data tanggal mobil mulai di sewa.
<b>transaksi_thl_kembali</b>	Menyimpan data tanggal mobil akan di kembalikan
<b>transaksi_harga</b>	Menyimpan harga rental mobil.
<b>transaksi_denda</b>	Menyimpan harga denda mobil perhari.
<b>transaksi_tgl</b>	Menyimpan data tanggal transaksi di lakukan.

Setelah menginput data transaksi, kita juga akan mengubah status ketersediaan mobil menjadi “2”. Seperti yang kita rancang sebelumnya. Jika status nya 1, berarti mobil tersedia. dan mobil yang berstatus 2 berarti sedang di sewa atau sedang di rental.

Jadi di sini kita mengubah status mobil yang ingin di sewa menjadi ‘2’.

```
// update status mobil yg di pinjam
$d = array(
    'mobil_status' => '2'
);
$w = array(
    'mobil_id' => $mobil
);
$this->m_rental->update_data($w, $d, 'mobil');
redirect(base_url().'admin/transaksi');
```

Sekarang yuk kita coba jalankan pada browser dan lihat bagaimana hasilnya.

Transaksi Baru

Kostumer  
Zainal Abidin

Mobil  
Suzuki Swift

Tanggal Pinjam  
29/10/2017

Tanggal Kembali  
31/10/2017

Harga  
500000

Harga Denda / Hari  
100000

## PENJELASAN !!!

Di sini saya menginput transaksi baru, dengan mengisi kostumer yang ingin melakukan rental mobil adalah zainal abidin, mobil yang ingin di sewa adalah suzuki swift, tanggal peminjamannya di mulai tanggal 29/10/2017 sampai dengan tanggal 31/10/2017 (selama 3 hari). Dengan biaya atau harga sebesar 500.000, dan denda perhari jika terlambat sebesar 100.000. dan klik simpan.

The screenshot shows a web-based administration interface for a car rental system. The URL in the address bar is `localhost/rental_mobil/admin/transaksi`. The page title is "Data Transaksi". The top navigation menu includes links for "Rental Mobil", "Dashboard", "Data Mobil", "Data Kostumer", "Transaksi Rental", "Laporan", "Logout", and a user profile "Halo, Diki Alfarabi Hadi".

The main content area displays a table titled "Data Transaksi" with the following columns: No, Tanggal, Kostumer, Mobil, Tgl. Pinjam, Tgl. Kembali, Harga, Denda / Hari, Tgl. Dikembalikan, Total Denda, and Status. There is one entry in the table:

No	Tanggal	Kostumer	Mobil	Tgl. Pinjam	Tgl. Kembali	Harga	Denda / Hari	Tgl. Dikembalikan	Total Denda	Status	
1	29/10/2017	Zainal Abidin	Suzuki Swift	29/10/2017	31/10/2017	Rp. 500,000	Rp. 100,000	-	Rp. 0,-	-	<span style="background-color: green; color: white; padding: 2px;">✓ Transaksi Selesai</span>

Below the table, there are two buttons: a green button labeled "✓ Transaksi Selesai" and a red button labeled "✗ Batalkan Transaksi". At the bottom of the table, it says "Showing 1 to 1 of 1 entries".

Sampai di sini kita telah berhasil menginput data transaksi ke table transaksi. Dan pada table di atas terdapat 2 buah tombol. Yaitu tombol batalkan transaksi, dan tombol transaksi selesai untuk menyelesaikan status transaksi. Yang ingin kita buat di sini adalah jika kostumer telah selesai menyewa mobil, admin/karywan bisa klik pada transaksi selesai. Dan akan di alihkan ke sebuah halaman baru untuk mengkonfirmasi bahwa mobil telah di kembalikan.

# Membuat Batalkan Transaksi

Pada table transaksi di atas, kita telah membuat tombol/link yang mengarah ke method transaksi\_hapus.

```
<a class="btn btn-sm btn-danger" href="php echo base_url().'admin/transaksi_hapus/'.$t-&gt;transaksi_id; ?&gt;"</span> Batalkan Transaksi</a>
```

Pada method transaksi\_hapus ini kita akan membuat aksi untuk menghapus transaksi. Buat method baru lagi dengan nama transaksi\_hapus() pada controller Admin.php.

## *Application/controllers/Admin.php*

```
function transaksi_hapus($id) {
    $w = array(
        'transaksi_id' => $id
    );
    $data = $this->m_rental->edit_data($w, 'transaksi')->row();

    $ww = array(
        'mobil_id' => $data->transaksi_mobil
    );
    $data2 = array(
        'mobil_status' => '1'
    );
    $this->m_rental->update_data($ww, $data2, 'mobil');

    $this->m_rental->delete_data($w, 'transaksi');
    redirect(base_url().'admin/transaksi');
}
```

## **PENJELASAN !!!**

Pada method transaksi\_hapus(), kita membuat aksi untuk menghapus mengubah kembali status mobil ke “1”. Dan menghapus data transaksi dari table transaksi.

# Membuat Transaksi Selesai (Konfirmasi)

Pada saat tombol “transaksi\_selesai” di klik, maka halaman akan di alihkan ke method transaksi\_selesai().

Pada method transaksi\_selesai() ini, kita akan menampilkan sebuah form baru, yaitu form untuk konfirmasi transaksi selesai.

Buat method baru dalam controllers Admin.php. yaitu method transaksi\_selesai().

## ***Application/controllers/Admin.php***

```
function transaksi_selesai($id){  
    $data['mobil'] = $this->m_rental->get_data('mobil')->result();  
    $data['kostumer'] = $this->m_rental->get_data('kostumer')->result();  
  
    $data['transaksi'] = $this->db->query("select * from  
transaksi,mobil,kostumer where transaksi_mobil=mobil_id and  
transaksi_kostumer=kostumer_id and transaksi_id='$id'")->result();  
    $this->load->view('admin/header');  
    $this->load->view('admin/transaksi_selesai',$data);  
    $this->load->view('admin/footer');  
}
```

## **PENJELASAN !!!**

Di sini kita memarsing data mobil, data kostumer ke view transaksi\_selesai.php. kita juga memarsing data transaksi yang ingin di proses. Dan query untuk mendapatkan data transaksi yang ingin di proses/di konfirmasi adalah seperti berikut.

```
$data['transaksi'] = $this->db->query("select * from transaksi,mobil,kostumer where transaksi_mobil=mobil_id and transaksi_kostumer=kostumer_id and  
transaksi_id='$id'")->result();
```

Ketiga data yang disebutkan di atas tadi kita parsing ke view transaksi\_selesai.php.

Buat view baru dalam folder admin dengan nama transaksi\_selesai.php.

# Application/views/admin/transaksi\_selesai.php

```
<div class="page-header">
    <h3>Transaksi Selesai</h3>
</div>

<?php foreach($transaksi as $t){ ?>
<form action="<?php echo base_url().'admin/transaksi_selesai_act' ?>" method="post">
    <input type="hidden" name="id" value="<?php echo $t->transaksi_id ?>">
    <input type="hidden" name="mobil" value="<?php echo $t->transaksi_mobil ?>">
    <input type="hidden" name="tgl_kembali" value="<?php echo $t->transaksi_tgl_kembali ?>">
    <input type="hidden" name="denda" value="<?php echo $t->transaksi_denda ?>">

    <div class="form-group">
        <label>Kostumer</label>
        <select class="form-control" name="kostumer" disabled>
            <option value="">-Pilih Kostumer-</option>
            <?php foreach($kostumer as $k){ ?>
                <option <?php if($t->transaksi_kostumer == $k->kostumer_id) {echo "selected='selected'"; } ?>
value="<?php echo $k->kostumer_id; ?><?php echo $k->kostumer_nama; ?></option>
            <?php } ?>
        </select>
    </div>

    <div class="form-group">
        <label>Mobil</label>
        <select class="form-control" name="mobil" disabled>
            <option value="">-Pilih Mobil-</option>
            <?php foreach($mobil as $m){ ?>
                <option <?php if($t->transaksi_mobil == $m->mobil_id) {echo "selected='selected'"; } ?>value="<?php echo $m->mobil_id; ?><?php echo $m->mobil_merk; ?></option>
            <?php } ?>
        </select>
    </div>

    <div class="form-group">
        <label>Tanggal Pinjam</label>
        <input class="form-control" type="date" name="tgl_pinjam" value="<?php echo
$t->transaksi_tgl_pinjam ?>" disabled>
    </div>

    <div class="form-group">
        <label>Tanggal Kembali</label>
        <input class="form-control" type="date" name="tgl_kembali" value="<?php echo
$t->transaksi_tgl_kembali ?>" disabled>
    </div>

    <div class="form-group">
        <label>Harga</label>
        <input class="form-control" type="number" name="harga" value="<?php echo
$t->transaksi_harga ?>" disabled>
    </div>

    <div class="form-group">
        <label>Harga Denda / Hari</label>
        <input class="form-control" type="text" name="denda" value="<?php echo
$t->transaksi_denda ?>" disabled>
    </div>

    <div class="form-group">
        <label>Tanggal Dikembalikan Oleh Kostumer</label>
        <input class="form-control" type="date" name="tgl_dikembalikan">
        <?php echo form_error('tgl_dikembalikan'); ?>
    </div>

    <div class="form-group">
        <input type="submit" value="Simpan" class="btn btn-primary btn-sm">
    </div>
</form>
<?php } ?>
```

# PENJELASAN !!!

Maka hasilnya seperti berikut.

The screenshot shows a web browser window with the URL `localhost/rental_mobil/admin/transaksi_selesai/1`. The page title is "Transaksi Selesai". The form fields are as follows:

- Kostumer: Zainal Abidin
- Mobil: Suzuki Swift
- Tanggal Pinjam: 29/10/2017
- Tanggal Kembali: 31/10/2017
- Harga: 500000
- Harga Denda / Hari: 100000
- Tanggal Dikembalikan Oleh Kostumer: hh/bb/tttt

The screenshot shows a web browser window with the URL `localhost/rental_mobil/admin/transaksi_selesai/1`. The page title is "Transaksi Selesai". The form fields are identical to the one above, but the "Tanggal Dikembalikan Oleh Kostumer" field is highlighted with a blue border. A blue arrow points from the text "PILIH TANGGAL MOBIL DI KEMBALIKAN OLEH PENYEWA/KOSTUMER" towards this field.

Jadi di sini data yang kita dapatkan dari query

```
$data['transaksi'] = $this->db->query("select * from transaksi,mobil,kostumer where transaksi_mobil=mobil_id and transaksi_kostumer=kostumer_id and transaksi_id='$id'")->result();
```

## PENJELASAN !!!

Kita tampilkan pada form nya masing-masing. Saya yakin teman-teman bisa memahaminya karena pada dasarnya sama saja seperti kita membuat edit data pada proses CRUD. Bedanya di sini pada semua form kita berikan atribut “disabled=’disabled’” kecuali pada form “tanggal di kembalikan oleh kostumer”.

Disinilah letak logika dari aplikasi rental mobil ini. Pada saat tanggal mobil di kembalikan oleh kostumer,”tanggal dikembalikan oleh kostumer” akan kita cek dengan data “tanggal kembali”. Apakah pengembalian mobil terlambat atau tidak. Jika terlambat, berapa hari keterlambatannya. Maka harga denda akan kita kalikan dengan berapa hari keterlambatan mobil di kembalikan. Jadi kita bisa mendapatkan berapa jumlah denda yang harus di bayar oleh kostumer pada saat pengembalian mobil.

Dan jika tanggal pengembalian mobil oleh kostumer sesuai dengan tanggal kembali yang sudah di tentukan pada saat transaksi, maka denda tidak ada. Tentu mudah bukan.

Mari kita simak penjelasan selanjutnya untuk membuat aksi dari form transaksi selesai (konfirmasi transaksi) ini.

Seperti yang sudah kita tetapkan pada form transaksi selesai di atas, kita menetapkan untuk aksi dari form ini akan di proses pada method transaksi\_selesai\_act() pada controller Admin.php.

Buat method baru dengan nama transaksi\_selesai\_act() dalam controller Admin.php.

## **Application/controllers/Admin.php**

```
function transaksi_selesai_act() {
    $id = $this->input->post('id');
    $tgl_dikembalikan = $this->input->post('tgl_dikembalikan');
    $tgl_kembali = $this->input->post('tgl_kembali');
    $mobil = $this->input->post('mobil');
    $denda = $this->input->post('denda');

    $this->form_validation->set_rules('tgl_dikembalikan', 'Tanggal Di Kembalikan', 'required');

    if($this->form_validation->run() != false) {
        // menghitung selisih hari
        $batas_kembali = strtotime($tgl_kembali);
        $dikembalikan = strtotime($tgl_dikembalikan);
        $selisih = abs(( $batas_kembali - $dikembalikan ) / (60*60*24));
        $total_denda = $denda * $selisih;

        // update status transaksi
        $data = array(
            'transaksi_tgldikembalikan' => $tgl_dikembalikan,
            'transaksi_status' => '1',
            'transaksi_totaldenda' => $total_denda
        );
        $w = array(
            'transaksi_id' => $id
        );

        $this->m_rental->update_data($w, $data, 'transaksi');

        // update status mobil
        $data2 = array(
            'mobil_status' => '1'
        );
        $w2 = array(
            'mobil_id' => $mobil
        );

        $this->m_rental->update_data($w2, $data2, 'mobil');
        redirect(base_url().'admin/transaksi');
    }else{
        $data['mobil'] = $this->m_rental->get_data('mobil')->result();
        $data['kostumer'] = $this->m_rental->get_data('kostumer')->result();

        $data['transaksi'] = $this->db->query("select * from transaksi,mobil,kostumer where transaksi_mobil=mobil_id and transaksi_kostumer=kostumer_id and transaksi_id='$id'")->result();
        $this->load->view('admin/header');
        $this->load->view('admin/transaksi_selesai', $data);
        $this->load->view('admin/footer');
    }
}
// AKHIR TRANSAKSI RENTAL
```

## PENJELASAN !!!

Perhatikan, pertama setelah kita menangkap data dan membuat validasinya.

```
$id = $this->input->post('id');
$tgl_dikembalikan = $this->input->post('tgl_dikembalikan');
$tgl_kembali = $this->input->post('tgl_kembali');
$mobil = $this->input->post('mobil');
$denda = $this->input->post('denda');

$this->form_validation->set_rules('tgl_dikembalikan','Tanggal Di Kembalikan','required');
```

Selenjutnya kita mengubah format tanggal kembali dan tanggal mobil di kembalikan. Tanggal kembali kita masukkan ke variabel \$batas\_kembali. Dan tanggal mobil di kembalikan kita masukkan ke variabel \$dikembalikan. Selanjutnya kita menghitung selisih hari dari batas pengembalian mobil dan tanggal mobil di kembalikan oleh kostumer. Selanjutnya kita menggunakan fungsi abs() berguna untuk membulatkan bilangan. Baru terakhir kita kalikan jumlah denda perhari dengan selisih hari pengembalian mobil. Hasil dari perkalian ini adalah total denda yang harus di bayar oleh kostumer (totaldenda = denda perhari X berapa hari terlambat). Dan selanjutnya kita update datanya yang ada di table “transaksi”.

```
// menghitung selisih hari
$batas_kembali = strtotime($tgl_kembali);
$dikembalikan = strtotime($tgl_dikembalikan);
$selisih = abs(( $batas_kembali - $dikembalikan) / (60*60*24));
$total_denda = $denda*$selisih;

// update status transaksi
$data = array(
    'transaksi_tgldikembalikan' => $tgl_dikembalikan,
    'transaksi_status' => '1',
    'transaksi_totaldenda' => $total_denda
);
$w = array(
    'transaksi_id' => $id
);

$this->m_rental->update_data($w,$data,'transaksi');
```

## PENJELASAN !!!

Terakhir kita juga mengupdate atau mengubah status mobil yang di kembalikan menjadi kembali tersedia (1). Tentu teman-teman masih ingat bukan, kalau kolumn mobil\_status yang kita buat pada table mobil untuk membuat status mobil. Yang mana kita telah menentukan jika nilai nya 1 = tersedia dan 2 = sedang dalam rental.

```
// update status mobil
$data2 = array(
    'mobil_status' => '1'
);
$w2 = array(
    'mobil_id' => $mobil
);

$this->m_rental->update_data($w2, $data2, 'mobil');
redirect(base_url() . 'admin/transaksi');
```

Selesai. Yuk kita coba.

The screenshot shows a web browser window with the URL `localhost/rental_mobil/admin/transaksi_selesai/1`. The page title is "Transaksi Selesai". It contains several input fields and dropdown menus:

- Kostumer:** Zainal Abidin
- Mobil:** Suzuki Swift
- Tanggal Pinjam:** 29/10/2017
- Tanggal Kembali:** 31/10/2017 (highlighted with a blue border)
- Harga:** 500000
- Harga Denda / Hari:** 100000
- Tanggal Dikembalikan Oleh Kostumer:** 02/11/2017

Below the form is a blue "Simpan" button.

## PENJELASAN !!!

Pada transaksi ini, sebelumnya batas pengembalian mobil adalah tanggal 31/10/2017. Dan di sini pada saat proses transaksi selesai kita masukkan tanggal pengembalian mobil oleh kostumer nya adalah tanggal 02/11/2017. Jadi otomatis kan si penyewa/kostumer sudah telat 2 hari pengembalian mobil nya dari batas yang sudah di tentukan. Karena denda perharinya adalah 100.000, maka total denda nya adalah 200.000, karena  $100.000 \times 2$ . Dan coba langsung kita klik simpan.

No	Tanggal	Kostumer	Mobil	Tgl. Pinjam	Tgl. Kembali	Harga	Denda / Hari	Tgl. Dikembalikan	Total Denda	Status
1	29/10/2017	Zainal Abidin	Suzuki Swift	29/10/2017	31/10/2017	Rp. 500.000	Rp. 100.000	02/11/2017	Rp. 200.000,-	Selesai

Hasilnya sesuai dengan perkiraan kita dan sesuai dengan yang seharusnya. Oh ya perlu teman-teman perhatikan di sini, pada column paling ujung sudah tidak ada lagi tombol “transaksi selesai” dan “batalkan transaksi”. Karena kita hanya menampilkan pada transaksi yang berstatus 2 (belum selesai). Jadi pada transaksi yang sudah selesai, data tidak bisa lagi di ganggu gugat. Coba kita input transaksi baru lagi. Dan input transaksi selesai nya sesuai jadwal pengembalian mobil. Apakah hasilnya sesuai atau tidak.

## PENJELASAN !!!

The screenshot shows a web-based application for managing car rentals. The URL is [localhost/rental\\_mobil/admin/transaksi\\_add](http://localhost/rental_mobil/admin/transaksi_add). The top navigation bar includes links for Rental Mobil, Dashboard, Data Mobil, Data Kostumer, Transaksi Rental, Laporan, Logout, and a user greeting "Halo, Diki Alfarabi Hadi". The main content area is titled "Transaksi Baru" (New Transaction). The form fields are as follows:

- Kostumer: Marlon Suwanganai (selected from a dropdown menu)
- Mobil: Toyota Avanza (selected from a dropdown menu)
- Tanggal Pinjam: 29/10/2017
- Tanggal Kembali: 30/10/2017
- Harga: 100000
- Harga Denda / Hari: 200000

A blue arrow points to the "Simpan" (Save) button at the bottom left of the form.

Untuk percobaan kedua ini. Kostumer “marlon suwanganai” yang ingin menyewa mobil. Dia ingin menyewa atau merental mobil “toyota avanza”. Mulai di sewa pada tanggal “29/10/2017” dan harus di kembalikan pada tanggal “30/10/2017”. Dan harganya sebesar Rp.100.000,-. Dengan harga denda perhari sebesar Rp.200.000,-. Dan klik simpan.

The screenshot shows a list of transactions. The URL is [localhost/rental\\_mobil/admin/transaksi](http://localhost/rental_mobil/admin/transaksi). The top navigation bar is identical to the previous screenshot. The main content area is titled "Data Transaksi". It displays two entries in a table:

No	Tanggal	Kostumer	Mobil	Tgl. Pinjam	Tgl. Kembali	Harga	Denda / Hari	Tgl. Dikembalikan	Total Denda	Status
1	29/10/2017	Zainal Abidin	Suzuki Swift	29/10/2017	31/10/2017	Rp. 500,000	Rp. 100,000	02/11/2017	Rp. 200.000	Selesai -
2	30/10/2017	Marlon Suwanganai	Toyota Avanza	29/10/2017	30/10/2017	Rp. 100,000	Rp. 200,000	-	Rp. 0,-	<span style="background-color: green; color: white; padding: 2px;">✓ Transaksi Selesai</span> <span style="background-color: red; color: white; padding: 2px;">✗ Batalkan Transaksi</span>

At the bottom, it says "Showing 1 to 2 of 2 entries" and has navigation buttons for "Previous" and "Next". A blue arrow points to the "Transaksi Selesai" button for the second entry.

## PENJELASAN !!!

Data berhasil di simpan. Dan klik transaksi baru. Karena pura-pura hari ini sudah tanggal 30/10/2017. Dan si kostumer sudah datang untuk mengembalikan mobilnya.

Kostumer  
Marlon Suwangan

Mobil  
Toyota Avanza

Tanggal Pinjam  
29/10/2017

Tanggal Kembali  
30/10/2017

Harga  
100000

Harga Denda / Hari  
200000

Tanggal Dikembalikan Oleh Kostumer  
30/10/2017

Simpan

Dan klik simpan.

Rental Mobil    Dashboard    Data Mobil    Data Kostumer    Transaksi Rental    Laporan    Logout    Halo, Diki Alfarabi Hadi

Data Transaksi

Data Transaksi										
<a href="#">+ Transaksi Baru</a>										
records per page: 10   <a href="#">Search</a>										
No	Tanggal	Kostumer	Mobil	Tgl. Pinjam	Tgl. Kembali	Harga	Denda / Hari	Tgl. Dikembalikan	Total Denda	Status
1	29/10/2017	Zainal Abidin	Suzuki Swift	29/10/2017	31/10/2017	Rp. 500,000	Rp. 100,000	02/11/2017	Rp. 200,000,-	Selesai
2	30/10/2017	Marlon Suwangan	Toyota Avanza	29/10/2017	30/10/2017	Rp. 100,000	Rp. 200,000	30/10/2017	Rp. 0,-	Selesai

Showing 1 to 2 of 2 entries

## **PENJELASAN !!!**

Karena si “marlon suwanggai” mengembalikan mobil nya tepat pada tanggal 30/10/2017 sesuai dengan batas pengembalian, maka tidak ada harga denda. Selesai sudah aplikasi rental mobil yang kita buat. Tapi terakhir untuk menambah kelengkapannya, kita akan membuat sebuah fitur laporan. Untuk memfilter dan mencetak laporan rental mobil.

# Membuat Filter dan Cetak Laporan Rental Mobil

Yang terakhir yang akan kita buat adalah fitur laporan. Dimana nanti kita bisa memfilter data berdasarkan tanggal. Membuat cetak laporan ke pdf dan membuat fitur untuk print laporan transaksi rental mobil.

Coba teman-teman perhatikan pada menu laporan. Kita telah membuat link nya menuju ke method laporan() di controller Admin.php.

```
<li><a href="<?php echo base_url().'admin/laporan'; ?>"><span class="glyphicon glyphicon-list-alt"></span> Laporan</a></li>
```

Oleh karena itu kita akan membuat method baru lagi di controller Admin.php. yaitu method/function laporan().

## ***Application/controllers/Admin.php***

```
// LAPORAN
function laporan(){
    $dari = $this->input->post('dari');
    $sampai = $this->input->post('sampai');
    $this->form_validation->set_rules('dari','Dari Tanggal','required');

    $this->form_validation->set_rules('sampai','Sampai Tanggal','required');

    if($this->form_validation->run() != false){
        $data['laporan'] = $this->db->query("select * from
transaksi,mobil,kostumer where transaksi_mobil=mobil_id and
transaksi_kostumer=kostumer_id and date(transaksi_tgl) >= '$dari'")->result();

        $this->load->view('admin/header');
        $this->load->view('admin/laporan_filter',$data);
        $this->load->view('admin/footer');
    }else{
        $this->load->view('admin/header');
        $this->load->view('admin/laporan');
        $this->load->view('admin/footer');
    }
}
```

## **PENJELASAN !!!**

Pada method ini kita membuat kondisi. Jika terdapat data tanggal “dari” dan tanggal “sampai”, maka view yang kita tampilkan adalah view laporan\_filter.php. Tapi jika tidak ada data tanggal “dari” dan “sampai”, maka view yang kita tampilkan adalah view laporan.php

## PENJELASAN !!!

Pada view laporan.php ini kita hanya akan membuat form input penginputan tanggal. Untuk di filter data laporan yang mau di tampilkan dari tanggal berapa dan sampai tanggal berapa.

Sedangkan pada view laporan\_filter.php, kita menampilkan data yang kita ambil dari table transaksi dengan detail nya data transaksi dari tanggal yang di input,sampai dengan tanggal yang di input. Fitur ini sering di sebut dengan menampilkan data dengan filter tanggal.

Sekarang kita akan membuat 2 buah view baru dalam folder admin. Dengan nama laporan.php dan laporan\_filter.php

### ***Application/views/admin/laporan.php***

```
<div class="page-header">
    <h3>Laporan</h3>
</div>

<form method="post" action="<?php echo base_url() . 'admin/laporan' ?>">
    <div class="form-group">
        <label>Dari Tanggal</label>
        <input type="date" name="dari" class="form-control">
        <?php echo form_error('dari'); ?>
    </div>
    <div class="form-group">
        <label>Sampai Tanggal</label>
        <input type="date" name="sampai" class="form-control">
        <?php echo form_error('sampai'); ?>
    </div>
    <div class="form-group">
        <input type="submit" value="CARI" name="cari" class="btn btn-sm
btn-primary">
    </div>
</form>
```

# **Application/views/admin/laporan\_filter.php**

```
<div class="page-header">
    <h3>Laporan</h3>
</div>

<form method="post" action="php echo base_url().'admin/laporan' ?&gt;"&gt;
    &lt;div class="form-group"&gt;
        &lt;label&gt;Dari Tanggal&lt;/label&gt;
        &lt;input type="date" name="dari" value="<?php echo set_value('dari'); ?&gt;" class="form-control"&gt;
        &lt;?php echo form_error('dari'); ?&gt;
    &lt;/div&gt;
    &lt;div class="form-group"&gt;
        &lt;label&gt;Sampai Tanggal&lt;/label&gt;
        &lt;input type="date" name="sampai" value="<?php echo set_value('sampai'); ?&gt;" class="form-control"&gt;
        &lt;?php echo form_error('sampai'); ?&gt;
    &lt;/div&gt;
    &lt;div class="form-group"&gt;
        &lt;input type="submit" value="CARI" name="cari" class="btn btn-sm btn-primary"&gt;
    &lt;/div&gt;
&lt;/form&gt;

&lt;div class="btn-group"&gt;
    &lt;a class="btn btn-warning btn-sm" href="<?php echo base_url().'admin/laporan_pdf/?dari='.$set_value('dari'). '&amp;sampai='.$set_value('sampai') ?&gt;" Cetak PDF</a>
    <a class="btn btn-default btn-sm" href="php echo base_url().'admin/laporan_print/?dari='.$set_value('dari'). '&amp;sampai='.$set_value('sampai') ?&gt;" Print</a>
</div>
<br/>
<br/>

<div class="table-responsive">
    <table border="1" class="table table-striped table-hover table-bordered" id="table-datatable">
        <thead>
            <tr>
                <th>No</th>
                <th>Tanggal</th>
                <th>Kostumer</th>
                <th>Mobil</th>
                <th>Tgl. Pinjam</th>
                <th>Tgl. Kembali</th>
                <th>Harga</th>
                <th>Denda / Hari</th>
                <th>Tgl. Dikembalikan</th>
                <th>Total Denda</th>
                <th>Status</th>
            </tr>
        </thead>
        <tbody>
            <?php
            $no = 1; foreach($laporan as
            $l) {
                ?>
                <tr>
                    <td><?php echo $no++; ?></td>
                    <td><?php echo date('d/m/Y',strtotime($l->transaksi_tgl)); ?></td>
                    <td><?php echo $l->kostumer_nama; ?></td>
                    <td><?php echo $l->mobil_merk; ?></td>
                    <td><?php echo date('d/m/Y',strtotime($l->transaksi_tgl_pinjam)); ?></td>
                    <td><?php echo date('d/m/Y',strtotime($l->transaksi_tgl_kembali)); ?></td>
                    <td><?php echo "Rp. ".number_format($l->transaksi_harga); ?></td>
                    <td><?php echo "Rp. ".number_format($l->transaksi_denda); ?></td>
                    <td>
                        <?php
                            if($l->transaksi_tgldikembalikan == "0000-00-00") {echo "-";
                            }else{
                                echo date('d/m/Y',strtotime($l->transaksi_tgldikembalikan));
                            }
                        ?>
                    </td>
                    <td><?php echo "Rp. ". number_format($l->transaksi_totaldenda)."-"; ?></td>
                    <td>
                        <?php
                            if($l->transaksi_status == "1") {echo
                                "Selesai";
                            }else{
                                echo "-";
                            }
                        ?>
                    </td>
                </tr>
            }
            <?php
            ?>
        </tbody>
    </table>
</div>
```

## PENJELASAN !!!

Sekarang kita coba klik pada menu “laporan”.

The screenshot shows a web-based application interface. At the top, there is a navigation bar with links: Rental Mobil, Dashboard, Data Mobil, Data Kostumer, Transaksi Rental, Laporan (which is highlighted with a blue border), Logout, and a user profile. Below the navigation bar is a section titled "Dashboard" containing four cards: "Jumlah Mobil" (10), "Jumlah Kostumer" (6), "Jumlah Transaksi" (2), and "Rental Selesai" (2). Each card has a "View Details" button. Below these cards are three sections: "Mobil" (listing car models like Daihatsu Sigra, Suzuki Ertiga, etc.), "Kostumer Terbaru" (listing customer names like Diki Alfarabi Hadi, Samsul Bahri, etc.), and "Peminjaman Terakhir" (listing recent rentals with details like Tgl. Transaksi, Tgl. Pinjam, Tgl. Kembali, Total). A blue arrow points upwards from the "Laporan" link in the navigation bar towards the "Laporan" section in the main content area.

Maka akan diarahkan ke method laporan() yang menampilkan form untuk filter tanggal.

The screenshot shows the "Laporan" page. The navigation bar at the top is identical to the one in the previous screenshot. The main content area is titled "Laporan" and contains two input fields: "Dari Tanggal" (with placeholder "hh/bb/tttt") and "Sampai Tanggal" (with placeholder "hh/bb/tttt"). Below these fields is a blue "CARI" button. This represents the state where the user has clicked on the "Laporan" menu item, which has triggered the display of a search form for filtering by date.

# Menampilkan Data Transaksi Berdasarkan Filter Tanggal

Pada form penginputan filter tanggal di atas, kita telah menetapkan aksi dari form tersebut ke method laporan () di controller Admin.php.

```
<form method="post" action="php echo base_url().'admin/laporan' ?&gt;&gt;</pre
```

tapi karena ada data tanggal dari dan tanggal sampai yang dikirim, maka view yang ditampilkan adalah view laporan\_filter.php. bukan lagi view laporan.php.



C:\xampp\htdocs\rental\_mobil\application\controllers\Admin.php (UNREGISTERED)

File Edit Selection Find Goto Tools Project Preferences Help

FOLDERS

- rental\_mobil
- application
  - cache
  - config
  - controllers
    - Admin.php
    - index.html
    - Welcome.php
  - core
  - helpers
  - hooks
  - language
  - libraries
  - logs
  - models
  - third\_party
- views
- admin
  - footer.php
  - ganti\_password.php
  - header.php
  - index.php
  - kostumer.php
  - kostumer\_add.php
  - kostumer\_edit.php
  - laporan.php
  - laporan\_filter.php
  - laporan\_pdf.php
  - laporan\_print.php

Dan view laporan\_filter.php sudah kita buat pada penjelasan sebelumnya. Jadi langsung saja kita coba jalankan pada browser dengan langsung menentukan data yang ingin ditampilkan dari tanggal berapa sampai tanggal berapa. Disini saya mencoba menampilkan data transaksi dari tanggal 1/10/2017 sampai dengan 31/10/2017.

## Laporan

Dari Tanggal  
→

Sampai Tanggal  
→  x ▾ ▾

**CARI**

Dan klik tombol cari. Maka akan di tampilkan semua data transaksi sesuai tanggal yang kita tentukan.

## Laporan

Dari Tanggal

Sampai Tanggal

**CARI**[Cetak PDF](#)[Print](#)

10 records per page Search:

No	Tanggal	Kostumer	Mobil	Tgl. Pinjam	Tgl. Kembali	Harga	Denda / Hari	Tgl. Dikembalikan	Total Denda	Status
1	29/10/2017	Zainal Abidin	Suzuki Swift	29/10/2017	31/10/2017	Rp. 500,000	Rp. 100,000	02/11/2017	Rp. 200,000,-	Selesai
2	30/10/2017	Marlon Suwanganai	Toyota Avanza	29/10/2017	30/10/2017	Rp. 100,000	Rp. 200,000	30/10/2017	Rp. 0,-	Selesai

Showing 1 to 2 of 2 entries ← Previous 1 Next →

Dan data transaksi pun berhasil tampil sesuai tanggal yang di filter.

# Membuat Print Laporan

Perhatikan pada tombol print yang terdapat di atas tabel data transaksi.

```
<a class="btn btn-default btn-sm" href="php echo base_url().'admin/laporan_print/?dari='.$_set_value('dari'). '&amp;sampai='.$_set_value('sampai') ?&gt;&lt;span class="glyphicon glyphicon-print"&gt;&lt;/span&gt; Print&lt;/a&gt;</pre
```

Pada tombol tersebut kita telah membuat url

```
href="php echo base_url().'admin/laporan_print/?dari='.$_set_value('dari'). '&amp;sampai='.$_set_value('sampai') ?&gt;</pre
```

Pada tombol print tersebut, kita mengarahkan halaman ke method laporan\_print() dengan mengirimkan data tanggal dari. Dan data tanggal sampai. set\_value('dari') berisi data tanggal dari. Dan set\_value('sampai') berisi data tanggal sampai. set\_value() sudah dijelaskan pada penggunaan form\_validation codeigniter pada “user\_guide”.

[http://localhost/rental\\_mobil/user\\_guide/libraries/form\\_validation.html?highlight=form%20validation](http://localhost/rental_mobil/user_guide/libraries/form_validation.html?highlight=form%20validation)

sehingga url yang dihasilkan adalah seperti pada gambar berikut

The screenshot shows a web application interface for managing vehicle rentals. At the top, there is a navigation bar with links for 'Rental Mobil', 'Dashboard', 'Data Mobil', 'Data Kostumer', 'Transaksi Rental', 'Laporan', 'Logout', and a user profile. Below the navigation bar, there is a search form titled 'Laporan' with fields for 'Dari Tanggal' (01/10/2017) and 'Sampai Tanggal' (31/10/2017). There are buttons for 'CARI' (Search), 'Cetak PDF', and 'Print'. A blue arrow points from the 'Print' button to the URL in the browser's address bar. Below the search form is a table displaying transaction data. The table has columns: No, Tanggal, Kostumer, Mobil, Tgl. Pinjam, Tgl. Kembali, Harga, Denda / Hari, Tgl. Dikembalikan, Total Denda, and Status. Two entries are shown:

No	Tanggal	Kostumer	Mobil	Tgl. Pinjam	Tgl. Kembali	Harga	Denda / Hari	Tgl. Dikembalikan	Total Denda	Status
1	29/10/2017	Zainal Abidin	Suzuki Swift	29/10/2017	31/10/2017	Rp. 500,000	Rp. 100,000	02/11/2017	Rp. 200,000,-	Selesai
2	30/10/2017	Marlon Suwanggai	Toyota Avanza	29/10/2017	30/10/2017	Rp. 100,000	Rp. 200,000	30/10/2017	Rp. 0,-	Selesai

At the bottom of the page, there is a footer with a link to 'localhost/rental\_mobil/admin/laporan\_print/?dari=2017-10-01&sampai=2017-10-31' and a timestamp '4:27 PM'.

Perhatikan pada gambar di atas. terlihat url tujuan dari link/tombol print saat cursor di letakkan pada tombol print. Seperti yang terlihat pada gambar. Halaman akan diarahkan ke method laporan\_print(). Dengan mengirim data melalui url (GET).

Selanjutnya buat method baru dengan nama transaksi\_print() pada controller Admin.php.

## **Application/controllers/Admin.php**

```
function laporan_print(){
    $dari = $this->input->get('dari');
    $sampai = $this->input->get('sampai');

    if($dari != "" && $sampai != "") {
        $data['laporan'] = $this->db->query("select * from
transaksi,mobil,kostumer where transaksi_mobil=mobil_id and
transaksi_kostumer=kostumer_id and date(transaksi_tgl) >= '$dari'")->result();

        $this->load->view('admin/laporan_print', $data);
    }else{
        redirect("admin/laporan");
    }
}
```

Seperti yang terlihat pada method laporan\_print() di atas, query berikut adalah query mysql untuk mendapatkan data transaksi sesuai dengan tanggal dari dan tanggal sampai. sama seperti cara menampilkan data transaksi yang sudah di bahas sebelumnya.

```
$data['laporan'] = $this->db->query("select * from transaksi,mobil,kostumer where transaksi_mobil=mobil_id and transaksi_kostumer=kostumer_id and
date(transaksi_tgl) >= '$dari'")->result();
```

Hasil dari query di atas kita tampilkan pada view laporan\_print.php dalam folder admin. Buat view baru dengan nama laporan\_print.php.

# Application/views/admin/laporan\_print.php

```
<!DOCTYPE html>
<html>
<head>
    <title></title>
</head>
<body>
    <style type="text/css">
        .table-data{
            width: 100%;
            border-collapse: collapse;
        }
        .table-data tr th,
        .table-data tr td{
            border:1px solid black;font-size: 10pt;
        }
    </style>
    <h3>Laporan Transaksi Rental Mobil</h3>

    <table>
        <tr>
            <td>Dari Tgl</td>
            <td><?php echo date('d/m/Y',strtotime($_GET['dari'])); ?></td>
        </tr>
        <tr>
            <td>Sampai Tgl</td>
            <td><?php echo date('d/m/Y',strtotime($_GET['sampai'])); ?></td>
        </tr>
    </table>

    <br/>
    <table class="table-data">
        <thead>
            <tr>
                <th>No</th>
                <th>Tanggal</th>
                <th>Kostumer</th>
                <th>Mobil</th>
                <th>Tgl. <RP></RP>ental</th>
                <th>Tgl. Kembali</th>
                <th>Harga</th>
                <th>Denda / Hari</th>
                <th>Tgl. Dikembalikan</th>
                <th>Total Denda</th>
                <th>Status</th>
            </tr>
        </thead>
        <tbody>
            <?php
                $no = 1; foreach($laporan as $l){
                    ?>
                    <tr>
                        <td><?php echo $no++; ?></td>
                        <td><?php echo date('d/m/Y',strtotime($l->transaksi_tgl)); ?></td>
                        <td><?php echo $l->kostumer_nama; ?></td>
                        <td><?php echo $l->mobil_merk; ?></td>
                        <td><?php echo date('d/m/Y',strtotime($l->transaksi_tgl_pinjam)); ?></td>
                        <td><?php echo date('d/m/Y',strtotime($l->transaksi_tgl_kembali)); ?></td>
                        <td><?php echo "Rp. ".number_format($l->transaksi_harga); ?></td>
                        <td><?php echo "Rp. ".number_format($l->transaksi_denda); ?></td>
                        <td>
                            <?php
                                if($l->transaksi_tgldikembalikan == "0000-00-00"){echo "-";
                                }else{
                                    echo date('d/m/Y',strtotime($l->transaksi_tgldikembalikan));
                                }
                            ?>
                        </td>
                        <td><?php echo "Rp. ". number_format($l->transaksi_totaldenda)."-"; ?></td>
                        <td>
                            <?php
                                if($l->transaksi_status == "1"){echo
                                    "Selesai";
                                }else{
                                    echo "-";
                                }
                            ?>
                        </td>
                    </tr>
                <?php
                }
            ?>
        </tbody>
    </table>

    <script type="text/javascript">
        window.print();
    </script>
</body>
</html>
```

# PENJELASAN !!!

Cara menampilkan datanya sama saja seperti cara menampilkan data dari database yang sudah di jelaskan sebelumnya. Bedanya di sini kita menambahkan syntax javascript untuk membuat print data.

```
<script type="text/javascript">
    window.print();
</script>
```

Sekarang kita akan mencoba klik tombol print. Dan lihat hasilnya.

The screenshot shows a web-based application for managing car rentals. At the top, there's a navigation bar with links for 'Rental Mobil', 'Dashboard', 'Data Mobil', 'Data Kostumer', 'Transaksi Rental', 'Laporan', 'Logout', and a user greeting 'Halo, Diki Alfarabi Hadi'. Below the navigation is a search form titled 'Laporan' with fields for 'Dari Tanggal' (01/10/2017) and 'Sampai Tanggal' (31/10/2017). It includes a 'CARI' button, a 'Cetak PDF' button, and a 'Print' button which is highlighted with a blue border. A blue arrow points from the 'Print' button towards the report table. Below the form is a table displaying rental transactions:

No	Tanggal	Kostumer	Mobil	Tgl. Pinjam	Tgl. Kembali	Harga	Denda / Hari	Tgl. Dikembalikan	Total Denda	Status
1	29/10/2017	Zainal Abidin	Suzuki Swift	29/10/2017	31/10/2017	Rp. 500,000	Rp. 100,000	02/11/2017	Rp. 200,000,-	Selesai
2	30/10/2017	Marlon Suwanggai	Toyota Avanza	29/10/2017	30/10/2017	Rp. 100,000	Rp. 200,000	30/10/2017	Rp. 0,-	Selesai

Below the table, it says 'Showing 1 to 2 of 2 entries.' and has 'Previous' and 'Next' buttons. The URL in the address bar is 'localhost/rental\_mobil/admin/laporan\_print/?dari=2017-10-01&sampai=2017-10-31'.

Maka akan muncul halaman print seperti berikut.

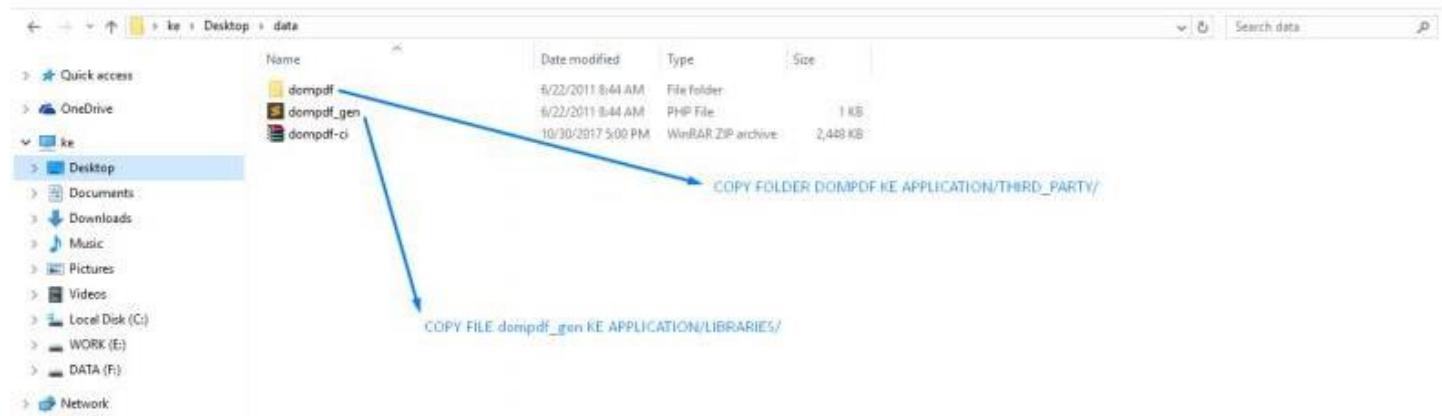
The screenshot shows a 'Cetak' (Print) dialog box. On the left, there are settings for 'Tujuan' (Destination) including 'Send To OneNote 2007' and 'Ubah...', 'Laman' (Pages) set to 'Semua' (All), 'Tata letak' (Orientation) set to 'Portrait', and 'Warna' (Color) set to 'Warna'. On the right, the print preview shows the title 'Laporan Transaksi Rental Mobil' with the date '30/10/2017'. It also displays the same report table as the previous screenshot. The URL in the address bar is 'localhost/rental\_mobil/admin/laporan\_print/?dari=2017-10-01&sampai=2017-10-31'.

Membuat cetak laporan selesai. Dan selanjutnya kita akan membuat cetak laporan

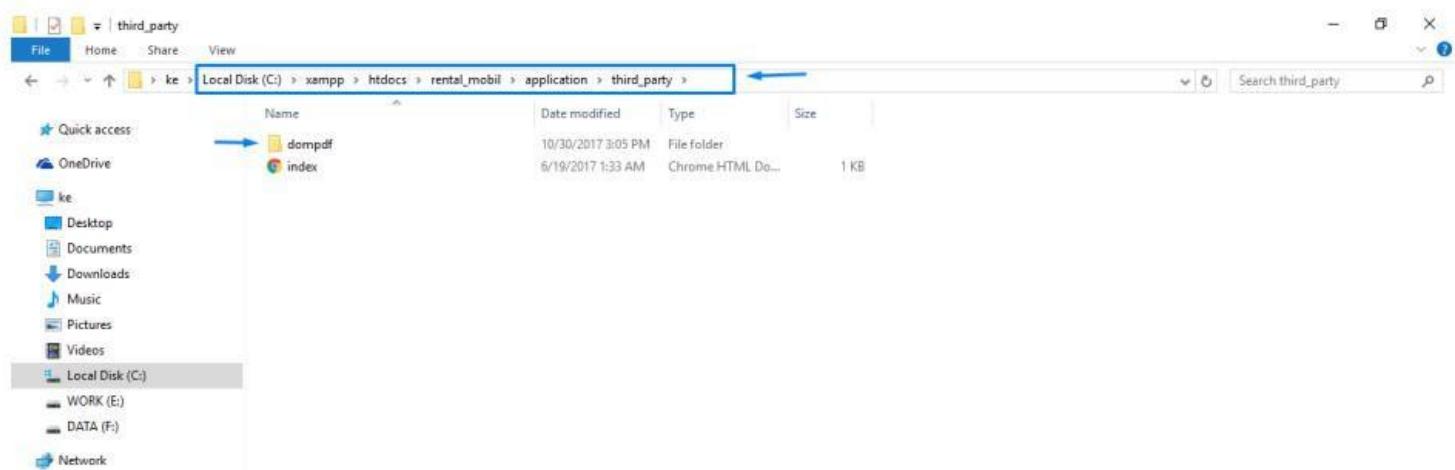
# Membuat Cetak Laporan Ke PDF Menggunakan DOMPDF

Untuk membuat cetak laporan PDF pada PHP, kita bisa menggunakan library DOMPDF. DOMPDF salah satu library PHP yang bisa di gunakan untuk membuat cetak laporan PDF dengan sangat mudah.

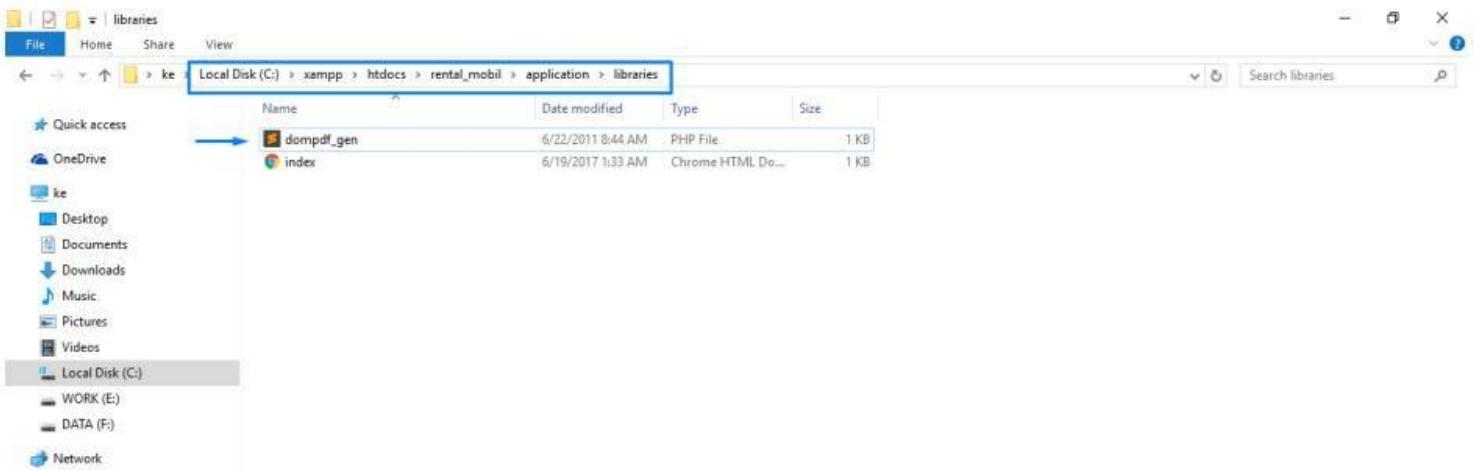
Untuk library DOMPDF nya sendiri sudah saya sertakan 1 paket dalam buku ini. Silahkan teman-teman ekstrak file dompdf-ci.zip. kemudian akan terdapat sebuah folder dompdf, dan satu buat file dompdf\_gen.php.



Selanjutnya copy folder dompdf ke application/third\_party/.



Dan copy file dompdf\_gen.php ke application/libraries/.



Sampai di sini instalasi dompdf telah selesai.

Jika pada tombol print kita membuat link menuju method laporan\_print(). Di tombol cetak kita membuat link menuju method laporan\_pdf().

```
<a class="btn btn-warning btn-sm" href="php echo base_url().'admin/laporan_pdf/?dari='.$_POST['dari'].'&amp;sampai='.$_POST['sampai']?"><span class="glyphicon glyphicon-print"></span> Cetak PDF</a>
```

A screenshot of a web browser displaying a page titled 'Rental Mobil'. The URL in the address bar is 'localhost/rental\_mobil/admin/laporan'. The page contains a navigation bar with links: 'Rental Mobil', 'Dashboard', 'Data Mobil', 'Data Kostumer', 'Transaksi Rental', 'Laporan', 'Logout', and a user greeting 'Halo, Diki Alfarabi Hadi'. Below the navigation is a search form with fields for 'Dari Tanggal' (01/10/2017) and 'Sampai Tanggal' (31/10/2017), a 'CARI' button, and two buttons: 'Cetak PDF' (highlighted with a blue arrow) and 'Print'. There is also a dropdown for 'records per page' set to 10 and a 'Search' input field. A table below lists rental transactions with columns: No#, Tanggal, Kostumer, Mobil, Tgl. Pinjam, Tgl. Kembali, Harga, Denda / Hari, Tgl. Dikembalikan, Total Denda, and Status. The table shows two entries: one for Zainal Abidin (Suzuki Swift) and another for Marlion Suwanggai (Toyota Avanza). At the bottom, it says 'Showing 1 to 2 of 2 entries' and has 'Previous' and 'Next' buttons. A blue double-headed arrow at the bottom indicates a comparison between the current screenshot and the previous one.

Jadi sekarang buat method baru lagi pada controller Admin.php dengan nama laporan\_pdf().

## **Application/controllers/Admin.php**

```
function laporan_pdf(){
    $this->load->library('dompdf_gen');
    $dari = $this->input->get('dari');
    $sampai = $this->input->get('sampai');

    $data['laporan'] = $this->db->query("select * from transaksi,mobil,kostumer
where transaksi_mobil=mobil_id and transaksi_kostumer=kostumer_id and
date(transaksi_tgl) >= '$dari'")->result();
    $this->load->view('admin/laporan_pdf', $data);

    $paper_size = 'A4'; // ukuran kertas
$orientation = 'landscape'; //tipe format kertas potrait atau landscape
$html = $this->output->get_output();

$this->dompdf->set_paper($paper_size, $orientation);
//Convert to PDF
$this->dompdf->load_html($html);
$this->dompdf->render();
$this->dompdf->stream("laporan.pdf", array('Attachment'=>0)); // nama file
pdf yang di hasilkan

}

}
```

## **PENJELASAN !!!**

Seperti biasa, kita menangkap data tanggal dari, dan tanggal sampai. kemudian kita ambil data transaksi rental dari table transaksi dengan memfilter data yang di ambil yaitu dari tanggal yang sudah di tentukan, dan sampai tanggal yang sudah di tentukan.

Pertama, kita panggil dulu library “dompdf\_gen” yang sudah kita copy tadi untuk mengaktifkannya.

**\$this->load->library('dompdf\_gen');**

Dan kemudian data yang kita dapat dari query tersebut kita tampilkan pada view laporan\_pdf.php.

## PENJELASAN !!!

```
$dari = $this->input->get('dari');
$ sampai = $this->input->get('sampai');

$data['laporan'] = $this->db->query("select * from transaksi,mobil,kostumer where transaksi_mobil=mobil_id and transaksi_kostumer=kostumer_id and date(transaksi_tgl) >= '$dari'" )->result();
```

Dan untuk mengkonversinya ke file pdf,

```
$paper_size = 'A4'; // ukuran kertas
$orientation = 'landscape'; //tipe format kertas potrait atau landscape
$html = $this->output->get_output();

$this->dompdf->set_paper($paper_size, $orientation);
//Convert to PDF
$this->dompdf->load_html($html);
$this->dompdf->render();
$this->dompdf->stream("laporan.pdf", array('Attachment'=>0)); // nama file pdf yang di hasilkan
```

Teman-teman bisa mengatur ukuran kertas, format kertas serta nama file pdf yang akan di hasilkan nantinya.

Selanjutnya ..

Buat view baru dengan nama laporan\_pdf.php.

# Application/views/admin/laporan\_pdf.php

```
<!DOCTYPE html>
<html>
<head>
    <title></title>
</head>
<body>
    <style type="text/css">
        .table-data{
            width: 100%;
            border-collapse: collapse;
        }

        .table-data tr th,
        .table-data tr td{
            border:1px solid black;font-size: 10pt;
        }
    </style>
</body>

<h3>Laporan Transaksi Rental Mobil</h3>

<table>
    <tr>
        <td>Dari Tgl</td>
        <td>:</td>
        <td><?php echo date('d/m/Y',strtotime($_GET['dari'])); ?></td>
    </tr>
    <tr>
        <td>Sampai Tgl</td>
        <td>:</td>
        <td><?php echo date('d/m/Y',strtotime($_GET['sampai'])); ?></td>
    </tr>
</table>

<br/>
<table class="table-data">
    <thead>
        <tr>
            <th>No</th>
            <th>Tanggal</th>
            <th>Kostumer</th>
            <th>Mobil</th>
            <th>Tgl. <RP>ental</th>
            <th>Tgl. Kembali</th>
            <th>Harga</th>
            <th>Denda / Hari</th>
            <th>Tgl. Dikembalikan</th>
            <th>Total Denda</th>
            <th>Status</th>
        </tr>
    </thead>
    <tbody>
        <?php
        $no = 1; foreach($laporan
        as $l){
        ?>
        <tr>
            <td><?php echo $no++; ?></td>
            <td><?php echo date('d/m/Y',strtotime($l->transaksi_tgl)); ?></td>
            <td><?php echo $l->kostumer_nama; ?></td>
            <td><?php echo $l->mobil_merk; ?></td>
            <td><?php echo date('d/m/Y',strtotime($l->transaksi_tgl_pinjam)); ?></td>
            <td><?php echo date('d/m/Y',strtotime($l->transaksi_tgl_kembali)); ?></td>
            <td><?php echo "Rp. ".number_format($l->transaksi_harga); ?></td>
            <td><?php echo "Rp. ".number_format($l->transaksi_denda); ?></td>
            <td>
                <?php
                if($l->transaksi_tgldikembalikan == "0000-00-00"){echo "-";
                };
                else{
                    echo date('d/m/Y',strtotime($l->transaksi_tgldikembalikan));
                }
                ?>
            </td>
            <td><?php echo "Rp. ". number_format($l->transaksi_totaldenda)." , -"; ?></td>
            <td>
                <?php
                if($l->transaksi_status == "1") (echo
                    "Selesai";
                )else{
                    echo "-";
                }
                ?>
            </td>
        </tr>
        <?php
        }
        ?>
    </tbody>
</table>
```

# PENJELASAN !!!

Yuk kita lihat hasil pembuatan cetak laporan PDF berdasarkan tanggalnya. Klik pada tombol “Cetak PDF”

The screenshot shows a web-based application interface for managing vehicle rentals. At the top, there is a navigation bar with links for 'Rental Mobil', 'Dashboard', 'Data Mobil', 'Data Kostumer', 'Transaksi Rental', 'Laporan', 'Logout', and a user greeting 'Halo, Diki Alfarabi Hadi'. Below the navigation bar, the page title is 'Laporan'. There are two input fields for date ranges: 'Dari Tanggal' (01/10/2017) and 'Sampai Tanggal' (31/10/2017). A blue arrow points to the 'Cetak PDF' button, which is highlighted with a yellow border. Below the search form is a table displaying rental transactions. The table has columns for No., Tanggal, Kostumer, Mobil, Tgl. Pinjam, Tgl. Kembali, Harga, Denda / Hari, Tgl. Dikembalikan, Total Denda, and Status. Two entries are listed:

No.	Tanggal	Kostumer	Mobil	Tgl. Pinjam	Tgl. Kembali	Harga	Denda / Hari	Tgl. Dikembalikan	Total Denda	Status
1	29/10/2017	Zainal Abidin	Suzuki Swift	29/10/2017	31/10/2017	Rp. 500,000	Rp. 100,000	02/11/2017	Rp. 200,000,-	Selesai
2	30/10/2017	Marlon Suwanggai	Toyota Avanza	29/10/2017	30/10/2017	Rp. 100,000	Rp. 200,000	30/10/2017	Rp. 0,-	Selesai

Below the table, it says 'Showing 1 to 2 of 2 entries'. On the right side of the table, there are buttons for 'Previous', 'Next', and a page number '1'. There are also buttons for 'Print' and a magnifying glass icon labeled 'Search'.

Dan hasilnya.

The screenshot shows a PDF document titled 'laporan.pdf' being viewed in a browser. The PDF contains the heading 'Laporan Transaksi Rental Mobil', the search criteria 'Dari Tgl : 01/10/2017' and 'Sampai Tgl : 31/10/2017', and a table of rental transactions identical to the one shown in the previous screenshot. The PDF viewer interface includes a toolbar with various icons and zoom controls.

No	Tanggal	Kostumer	Mobil	Tgl. ental	Tgl. kembali	Harga	Denda / Hari	Tgl. Dikembalikan	Total Denda	Status
1	29/10/2017	Zainal Abidin	Suzuki Swift	29/10/2017	31/10/2017	Rp. 500,000	Rp. 100,000	02/11/2017	Rp. 200,000,-	Selesai
2	30/10/2017	Marlon Suwanggai	Toyota Avanza	29/10/2017	30/10/2017	Rp. 100,000	Rp. 200,000	30/10/2017	Rp. 0,-	Selesai

Cetak laporan ke format PDF pun selesai.

**SELESAI . . .**