

# Codelgniter



# Introduction

- ▶ PHP Framework
- ▶ MVC Design Pattern
  - ▶ Model
  - ▶ View
  - ▶ Controller

# Setup CodeIgniter

- ▶ Download CodeIgniter จาก <https://www.codeigniter.com/download>
- ▶ หลังจาก Download เสร็จให้ทำการแตกไฟล์ จะได้ Folder CodeIgniter-3.0.4

## **CodeIgniter 3.0**

CodeIgniter 3.0.4 is the current version of the framework.

There have been a number of refinements since version 2.x, notably with the database, session handling and encryption. Development of this version is ongoing.

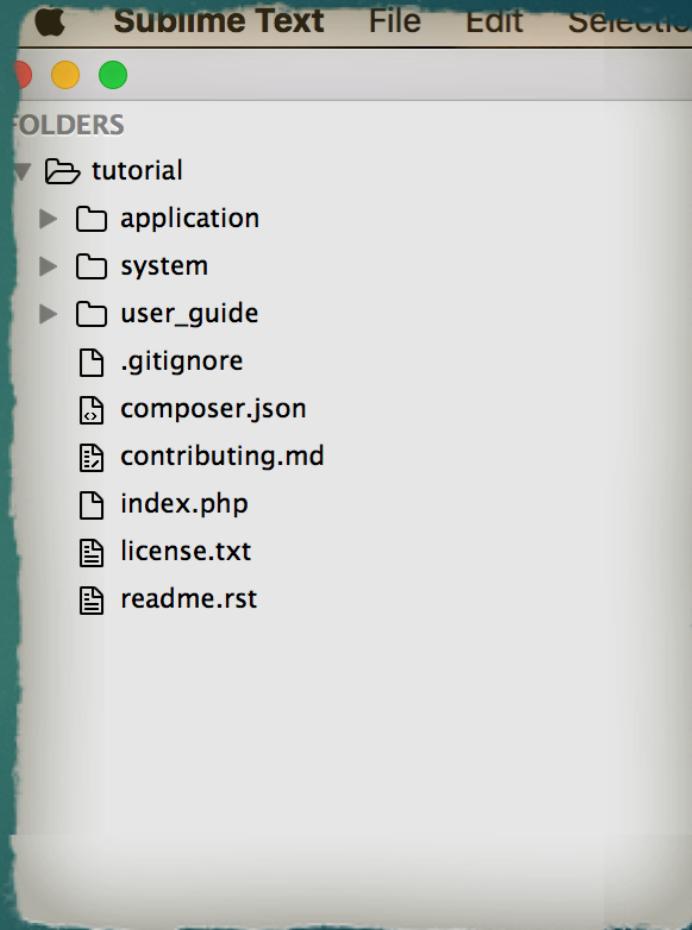
[View CodeIgniter 3 on Github](#)

 [Download CodeIgniter 3](#)

 [Download System Message Translations](#)

# Setup CodeIgniter

- ▶ ทำการย้าย Folder ไปไว้ใน Root ของ PHP Server และทำการเปลี่ยนชื่อ Folder ตามต้องการโดยในที่นี่เป็น tutorial
- ▶ ทำการเปิด Folder Project ด้วยโปรแกรม Sublime โดยจะเป็นโครงสร้างของโปรเจคดังภาพ

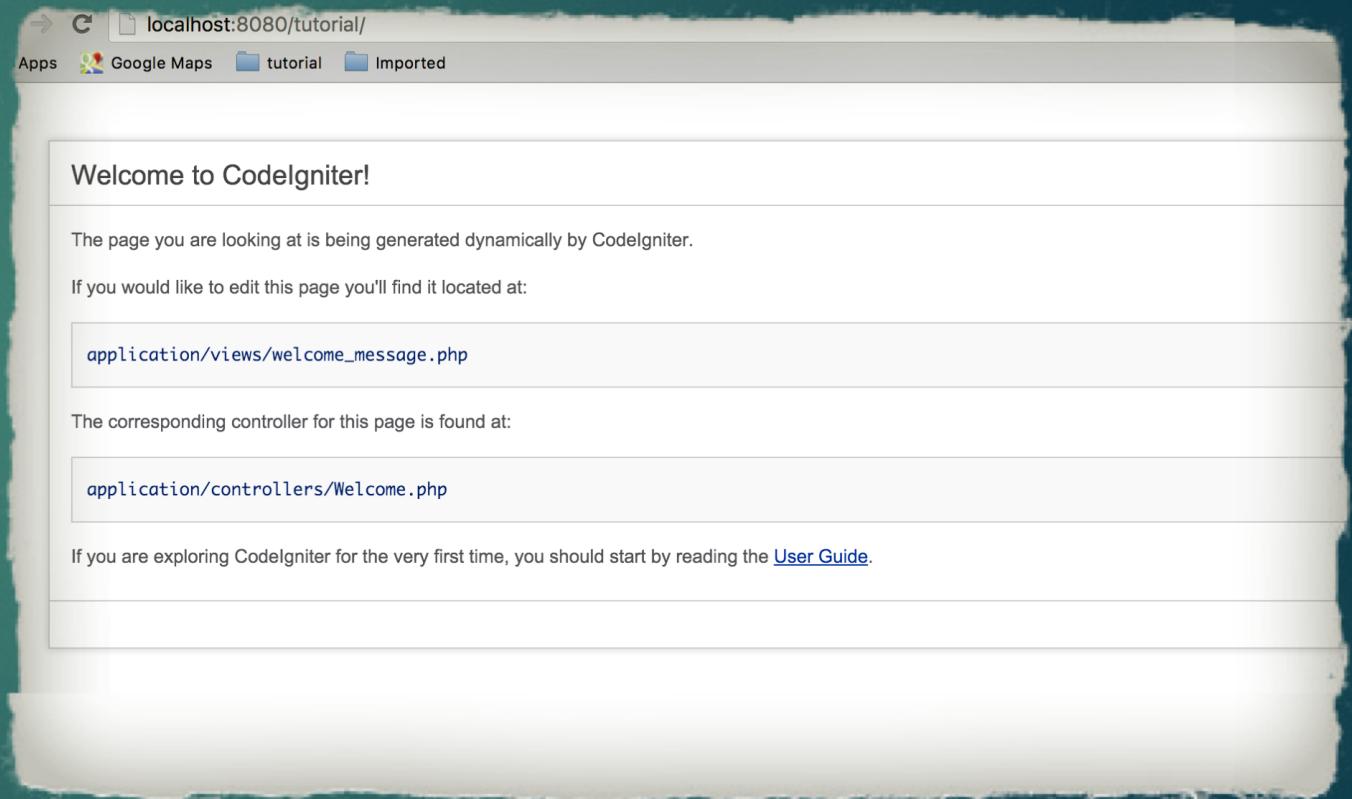


# Setup CodeIgniter

- ▶ ทดสอบ Run โปรแกรมโดยเข้าไปที่

<http://127.0.0.1:8080/tutorial>

จะปรากฏหน้าจอดังภาพ



# Config Project



# Remove index.php

- ▶ ไปที่ Path tutorial จากโปรแกรม Sublime  
ทำการ New File ชื่อว่า .htaccess  
เพื่อทำหน้าที่ในการตัดindex.php ใน URL
- ▶ ทำการพิมพ์ข้อความตามภาพลงในFile  
.htaccess

```
1 RewriteEngine On
2 RewriteCond %{REQUEST_FILENAME} !-f
3 RewriteCond %{REQUEST_FILENAME} !-d
4 RewriteRule ^(.*)$ index.php/$1 [L]
5
6
```

# Auto load library and helper

- ▶ ทำการกด short key (command + p) และพิมพ์คำว่า autoload.php จากนั้นทำการกด Enter
- ▶ ทำการใส่ค่า 'database' ใน \$autoload['libraries'] เพื่อทำการ load database library
- ▶ ทำการใส่ค่า 'form', 'url' ใน \$autoload['helper'] เพื่อทำการ load helper

# Auto load library and helper

```
57 |     in the controller.
58 |
59 |     $autoload['libraries'] = array('user_agent' => 'ua');
60 */
61 $autoload['libraries'] = array('database');
62
63 /*
64 | -----
65 |   Auto-load Drivers
66 | -----
67 | These classes are located in system/libraries/ or in your
68 | application/libraries/ directory, but are also placed inside their
69 | own subdirectory and they extend the CI_Driver_Library class. They
70 | offer multiple interchangeable driver options.
71 |
72 | Prototype:
73 |
74 |     $autoload['drivers'] = array('cache');
75 */
76 $autoload['drivers'] = array();
77
78 /*
79 | -----
80 |   Auto-load Helper Files
81 | -----
82 | Prototype:
83 |
84 |     $autoload['helper'] = array('url', 'file');
85 */
86 $autoload['helper'] = array('form', 'url');
87
```

# Setup config file

- ▶ ทำการเปิด File config.php
- ▶ ทำการใส่ค่า 'http://127.0.0.1:8080/tutorial' ลงใน \$config['base\_url'] เพื่อทำการตั้งค่า base\_url
- ▶ ทำการลบค่า index.php ใน \$config['index\_page'] เพื่อทำการลบ default index.php ใน url กรณีที่ใช้การ function ในการ generate link ของ codeigniter

# Setup config file

```
1  |
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://127.0.0.1:8080/tutorial';
27 |
28 /*
29 | -----
30 | Index File
31 | -----
32 |
33 | Typically this will be your index.php file, unless you've renamed it to
34 | something else. If you are using mod_rewrite to remove the page set this
35 | variable so that it is blank.
36 |
37 */
38 $config['index_page'] = '';
39 |
40 /*
41 | -----
42 | URI PROTOCOL
43 | -----
44 |
45 | This item determines which server global should be used to retrieve the
46 | URI string. The default setting of 'REQUEST_URI' works for most servers.
```

# Setup database connection

- ▶ ทำการเปิด File database.php
- ▶ ทำการใส่ค่า ใน \$db['default'] ดังนี้
  - 'hostname' => 'localhost'
  - 'username' => 'root'
  - 'password' => 'root'
  - 'database' => 'tutorial\_db'

```
2  */
3 $active_group = 'default';
4 $query_builder = TRUE;
5
6 $db['default'] = array(
7     'dsn' => '',
8     'hostname' => 'localhost',
9     'username' => 'root',
10    'password' => 'root',
11    'database' => 'tutorial_db',
12    'dbdriver' => 'mysqli',
13    'dbprefix' => '',
14    'pconnect' => FALSE,
15    'db_debug' => (ENVIRONMENT !== 'production'),
16    'cache_on' => FALSE,
17    'cachedir' => '',
18    'char_set' => 'utf8',
19    'dbcollat' => 'utf8_general_ci',
20    'swap_pre' => '',
21    'encrypt' => FALSE,
22    'compress' => FALSE,
23    'stricton' => FALSE,
24    'failover' => array(),
25    'save_queries' => TRUE
26 );
27
```

# Create database

- ▶ ทำการเปิด phpMyAdmin และทำการสร้าง database ชื่อว่า tutorial\_db
- ▶ ทำการสร้าง table ชื่อ customers โดยมีรายละเอียดดังนี้

Field	Type	NULL	Default	Auto increment
custId	int	N	-	Y
firstname	varchar(50)	N	-	N
lastname	varchar(50)	N	-	N
email	varchar(50)	Y	-	N
c_date	timestamp	N	current_timestamp	N

# Display Customer

BY CODEIGNITER

# Create Layout View

- ▶ ทำการสร้างไฟล์ layout.php ที่ (tutorial/application/views)
- ▶ ทำการใส่ html โดยประกอบด้วย nav menu และ container เพื่อสร้าง template หลัก โดยในส่วนของ content ของ container ให้ใส่ Code ดังนี้

```
<?php (isset($yield) && !empty($yield))? print $yield:""; ?>
```

- ▶ ในส่วนที่เป็น Hyperlink ให้ใช้ function anchor ของ codeIgniter แทน ดังนี้  

```
<?php echo anchor('link url', 'label', array('class'=> 'classe name')); ?>
```

# Create Layout View

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Codeigniter Tutorial</title>
5      <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css"
6          <style>
7              .txt-not-found{
8                  text-align: center;
9                  color: #585858;
10             }
11         </style>
12     </head>
13     <body>
14         <nav class="navbar navbar-inverse">
15             <div class="container">
16                 <div class="navbar-header">
17                     <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1" expanded="false">
18                         <span class="sr-only">Toggle navigation</span>
19                         <span class="icon-bar"></span>
20                         <span class="icon-bar"></span>
21                         <span class="icon-bar"></span>
22                     </button>
23                     <?php
24                         echo anchor('customer', 'Codeigniter Tutorial', array('class'=>'navbar-brand'));
25                     ?>
26                 </div>
27                 <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
28                     <ul class="nav navbar-nav navbar-right">
29                         <li>
30                             <?php
31                             echo anchor('customer/create', 'Create Customer');
32                         ?>
33                         </li>
34                     </ul>
35                 </div><!-- /.navbar-collapse -->
36             </div><!-- /.container-fluid -->
37         </nav>
38         <div class="container">
39             <div class="row">
40                 <div class="col-md-offset-3 col-md-6">
41                     <!-- Content -->
42                     <?php if(isset($yeild) && !empty($yeild)) { print $yeild; } ?>
43                 </div>
44             </div>
45         </div>
46         <script src="http://code.jquery.com/jquery-2.2.1.min.js"></script>
47         <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js"></script>
48     </body>
```

# Create View Customer

- ▶ ทำการสร้างไฟล์ view\_customer.php ที่ (tutorial/application/views)
- ▶ ทำการใส่ Code ตามภาพ เพื่อทำการแสดงตาราง Customer จาก database

```
database.php          layout.php           view_customer.php
1 <div class="adv-table">
2   <div class="space15"></div>
3   <table class="table table-striped table-hover table-bordered" id="editable-sample">
4     <thead>
5       <tr>
6         <th>No.</th>
7         <th>Firstname</th>
8         <th>Lastname</th>
9         <th>Email</th>
10        <th>Edit</th>
11        <th>Delete</th>
12      </tr>
13    </thead>
14    <tbody>
15      <?php
16        if(!isset($list_data)|| empty($list_data)) {
17          echo "<tr class='txt-not-found'><td colspan=6>Data Not Found</td></tr>";
18        } else{
19          $i=1;
20          foreach ($list_data as $rows){
21            echo "<tr class='>";
22            echo "<td>$i</td>";
23            echo "<td>$rows->firstname</td>";
24            echo "<td>$rows->lastname</td>";
25            echo "<td>$rows->email</td>";
26            echo "<td>";
27            echo anchor("customer/edit/$rows->custId", 'Edit');
28            echo "</td>";
29            echo "<td>";
30            echo anchor("customer/delete/$rows->custId", 'Delete');
31            echo "</td>";
32            echo "</tr>";
33            $i++;
34          }
35        }
36      ?>
37    </tbody>
38  </table>
39 </div>
40 </div>
```

# Create Model

- ▶ ทำการสร้าง File M\_Customer.php ที่ Path (tutorial/application/models)
- ▶ เปิด File M\_Customer.php ทำการสร้าง Class M\_Customer โดยทำการ Extends CI\_Model
- ▶ ทำการสร้าง function \_\_construct เพื่อเป็น constructer method และทำการเรียก construct method ของ class CI\_Model
- ▶ ทำการประกาศตัวแปรชื่อ table และใส่ค่า 'tbl\_customer' ดังภาพ

```
<?php  
  
class M_Customer extends CI_Model{  
  
    private $table = 'tbl_customer';  
  
    public function __construct()  
    {  
        parent::__construct();  
    }  
}
```

# Create Model

- ▶ ทำการสร้าง File M\_Customer.php ที่ Path (tutorial/application/models)
- ▶ เปิด File M\_Customer.php ทำการสร้าง Class M\_Customer โดยทำการ Extends CI\_Model
- ▶ ทำการสร้าง function \_\_construct เพื่อเป็น constructer method และทำการเรียก construct method ของ class CI\_Model
- ▶ ทำการประกาศตัวแปรชื่อ table และใส่ค่า 'tbl\_customer' ดังภาพ

```
<?php  
  
class M_Customer extends CI_Model{  
  
    private $table = 'tbl_customer';  
  
    public function __construct()  
    {  
        parent::__construct();  
    }  
}
```

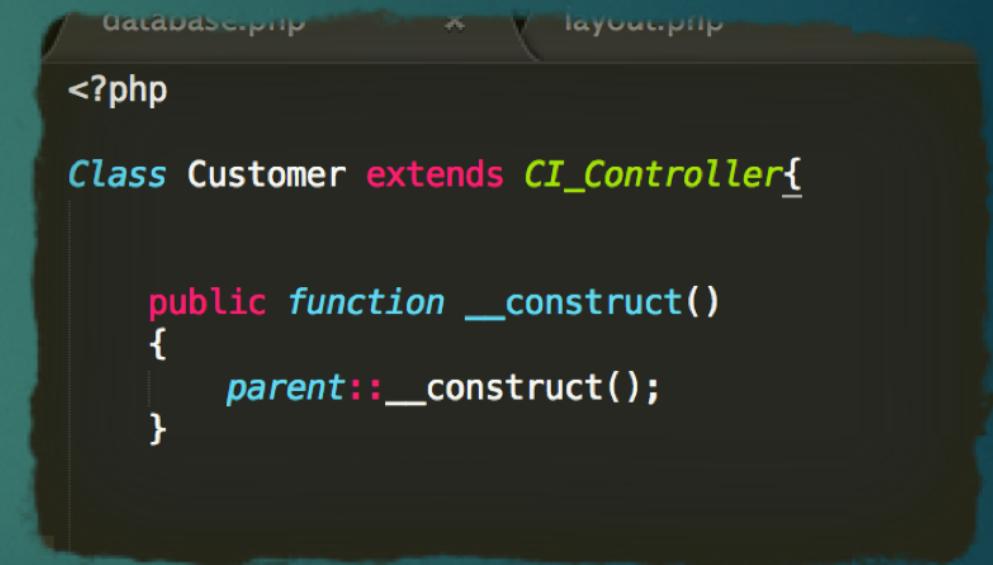
# Create Model

- ▶ ทำการสร้าง function getAllData เพื่อทำการ Select ข้อมูลจาก table customer
- ▶ ทำการเรียก function get จาก database library โดยทำการส่ง parameter เป็นชื่อ table name เพื่อทำการสร้าง query สำหรับ select customer
- ▶ ใช้ function result สำหรับ run query และทำการ return result ที่ได้ดังภาพ

```
1 <?php
2
3 class M_Customer extends CI_Model{
4
5     private $table = 'tbl_customer';
6
7     public function __construct()
8     {
9         parent::__construct();
10    }
11
12     public function getAllData(){
13         $query = $this->db->get($this->table);
14         return $query->result();
15     }
16 }
```

# Create Controller

- ▶ ทำการสร้าง File Customer.php ที่ Path (tutorial/application/controllers)
- ▶ เปิด File Customer.php ทำการสร้าง class Customer และทำการ Extends CI\_Controller
- ▶ ทำการสร้าง function \_\_construct ภายใน function ให้ทำการเรียก construct method ของ class CI\_Controller โดยใช้คำสั่ง parent::\_\_construct ดังภาพ



```
<?php
Class Customer extends CI_Controller{
    public function __construct()
    {
        parent::__construct();
    }
}
```

# Create Controller

- ▶ ที่ construct ทำการ load model M\_Customer และทำการใส่ alias เป็น cust ดังนี้  
    \$this->load->model(array('M\_customer'=>'cust'));

```
Class Customer extends CI_Controller{  
  
    public function __construct()  
    {  
        parent::__construct();  
        $this->load->model(array('M_Customer'=>'cust'));  
    }  
}
```

# Create Controller

- ▶ ทำการสร้าง function index และทำการเรียก function getAllData จาก model M\_Customer โดยใช้ \$this->cust->getAllData(); และทำการรับด้วย array \$listData[“list\_data”] ดังภาพ

```
2
3 Class Customer extends CI_Controller{
4
5     public function __construct()
6     {
7         parent::__construct();
8         $this->load->model(array('M_Customer'=>'cust'));
9     }
10
11
12    public function index(){
13        $listData["list_data"] = $this->cust->getAllData();
14    }
15
16}
```

# Create Controller

- ▶ ทำการ load view\_customer และใส่ค่าในลงใน array \$html['yeild'] ดังภาพ

```
3 Class Customer extends CI_Controller{
4
5     public function __construct()
6     {
7         parent::__construct();
8         $this->load->model(array('M_Customer'=>'cust'));
9     }
10
11
12     public function index(){
13         $listData["list_data"] = $this->cust->getAllData();
14         $html['yeild'] = $this->load->view('view_customer',$listData,true);
15     }
16 }
```

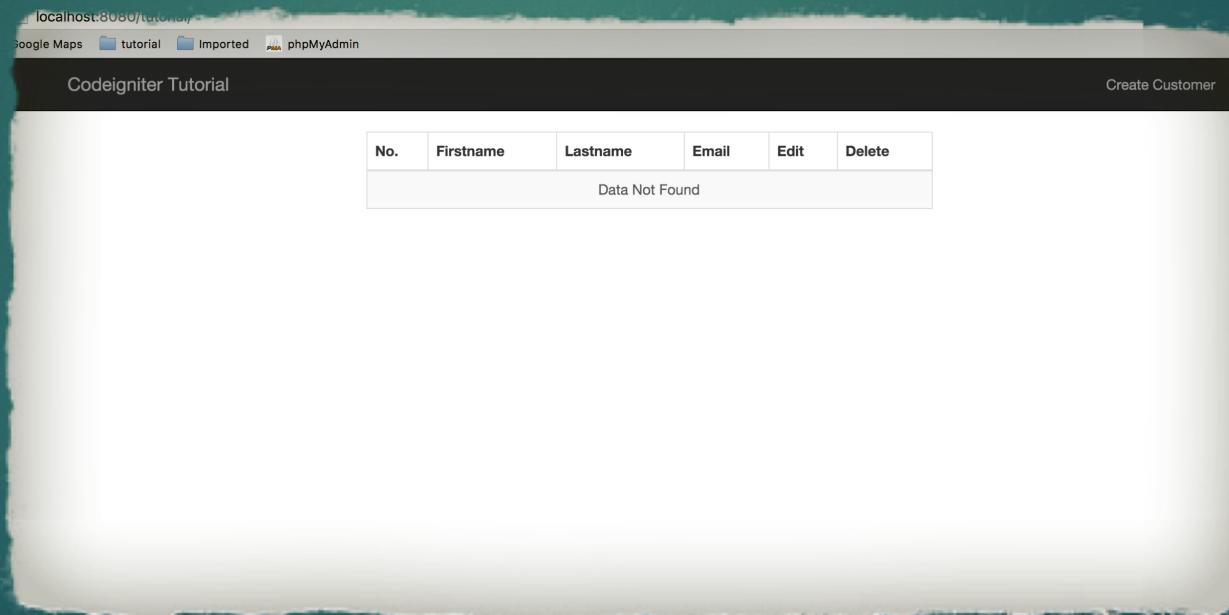
# Create Controller

- ▶ ทำการ load view layout และทำการสร้างสิ่งค่า \$html เพื่อทำการ render หน้าจอดังภาพ

```
3 Class Customer extends CI_Controller{
4
5     public function __construct()
6     {
7         parent::__construct();
8         $this->load->model(array('M_Customer'=>'cust'));
9     }
10
11    public function index(){
12        $listData["list_data"] = $this->cust->getAllData();
13        $html['yeild'] = $this->load->view('view_customer',$listData,true);
14        $this->load->view('layout.php',$html);
15    }
16}
```

# Test Display Customer

- ▶ ทำการ Run Web โดยพิมพ์ <http://127.0.0.1:8080/tutorial> จะปรากฏหน้าจอดังภาพ



# Insert Customer

BY CODEIGNITER

# Create View

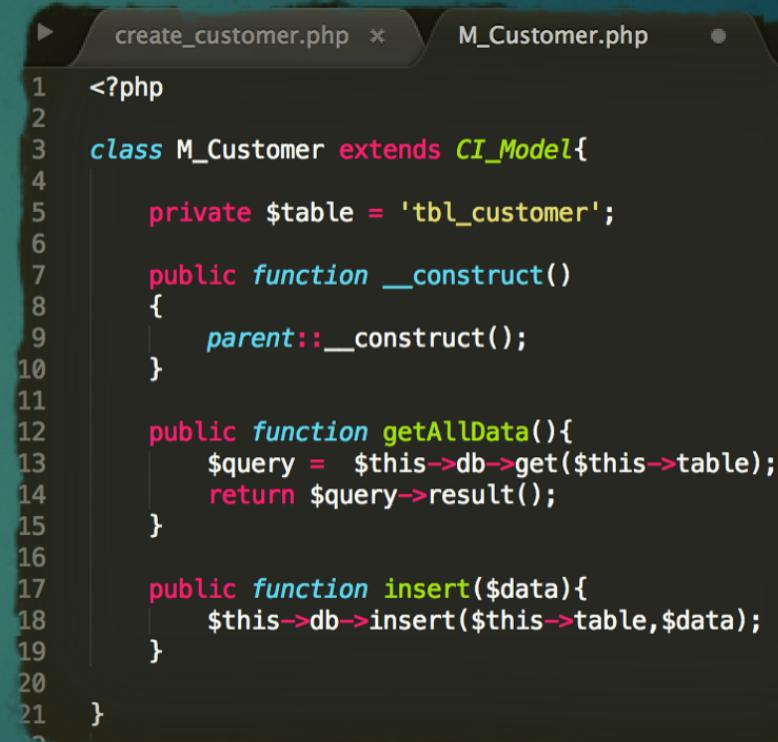
- ▶ สร้าง File create\_customer.php ที่ path (tutorial/application/views)
- ▶ ทำการสร้าง form ในการรับข้อมูล customer โดยใช้ form helper ของ CodeIgniter เข้ามาช่วยประกอบด้วยคำสั่งดังนี้
  - ▶ form\_open ใช้สำหรับสร้าง form element โดยทำการรับ parameter action
  - ▶ form\_input ใช้สำหรับสร้าง input element
  - ▶ form\_label ใช้สำหรับสร้าง label element
  - ▶ form\_button ใช้สำหรับสร้าง button
  - ▶ form\_close ใช้สำหรับสร้าง tag ปิด form

# Create View

```
create_customer.php * Customer.php * layout.php * M_Customer.php *
1 <div class="panel panel-default">
2   <div class="panel-heading">Create Customer</div>
3   <div class="panel-body">
4     <?php
5       $firstname = array('name'=>'firstname', 'id'=>'firstname', 'class'=>'form-control',
6                         'required'=>');
7       $lastname= array('name'=>'lastname', 'id'=>'lastname', 'class'=>'form-control', 'required'=>');
8       $email = array('type'=>'email', 'name'=>'email', 'id'=>'email', 'class'=>'form-control');
9       $action = 'customer/save';
0
11      echo form_open($action, array('id'=>'formCustomer', 'role'=>'form'));
12      echo '<div class="form-group">';
13      echo form_label('First Name', 'firstname');
14      echo form_input($firstname);
15      echo '</div>';
16      echo '<div class="form-group">';
17      echo form_label('Lastname ', 'lastname');
18      echo form_input($lastname);
19      echo '</div>';
20      echo '<div class="form-group">';
21      echo form_label('Email ', 'email');
22      echo form_input($email);
23      echo '</div>';
24      echo form_button(array('type'=>'submit', 'class'=>'btn btn-info', 'content'=>'Save'));
25      echo form_button(array('type'=>'reset', 'class'=>'btn btn-info', 'content'=>'Cancel'));
26      echo form_close();
27
28    ?>
29    </div>
30  </div>
```

# Create Insert Function

- ▶ ทำการสร้าง function insert เพื่อใช้ในการ insert ข้อมูลลง table customer
- ▶ โดยทำการรับ 1 parameter คือ \$data สำหรับข้อมูลของ customer
- ▶ ทำการเรียก method insert จาก library database โดยทำการส่ง 2 parameter table name และ ข้อมูลสำหรับ insert



```
<?php  
class M_Customer extends CI_Model{  
    private $table = 'tbl_customer';  
    public function __construct()  
    {  
        parent::__construct();  
    }  
    public function getAllData(){  
        $query = $this->db->get($this->table);  
        return $query->result();  
    }  
    public function insert($data){  
        $this->db->insert($this->table,$data);  
    }  
}
```

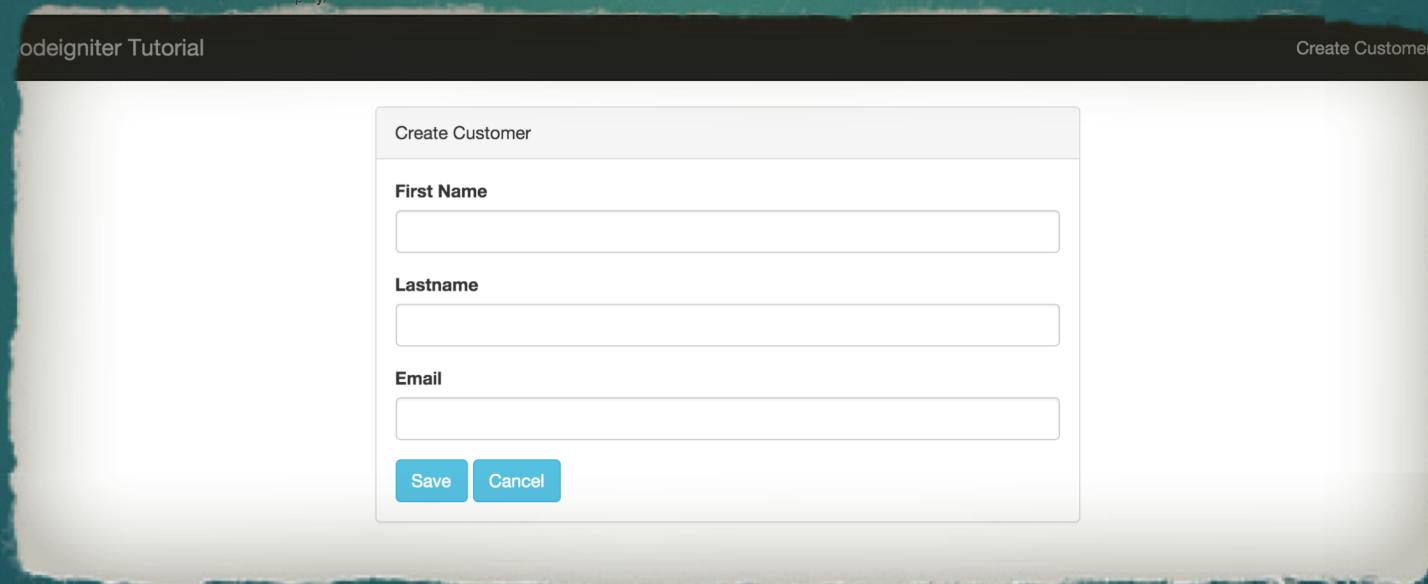
# Create function controller

- ▶ ทำการสร้าง function เพื่อทำการแสดงหน้าในการ create customer โดยให้ชื่อว่า create ทำการ load view create\_customer โดยทำการ set parameter ตำแหน่งที่ 3 เป็น true เพื่อให้ทำการ return ค่าเป็น string และทำการเป็นไว้ใน array html['yeild']
- ▶ ทำการ load view layout โดยทำการส่ง html ไปเพื่อ แสดงหน้าจอ create customer

```
<?php  
  
Class Customer extends CI_Controller{  
  
    public function __construct()  
    {  
        parent::__construct();  
        $this->load->model(array('M_Customer'=>'cust'));  
    }  
  
    public function index(){  
        $listData["list_data"] = $this->cust->getAllData();  
        $html['yeild'] = $this->load->view('view_customer',$listData,true);  
    }  
  
    public function create(){  
        $html['yeild'] = $this->load->view('create_customer','','true');  
        $this->load->view('layout.php',$html);  
    }  
}
```

# Test create customer

- ▶ ทำการ run โปรแกรมโดยพิมพ์ URL =  
<http://127.0.0.1:8080/tutorial/customer/create> จะปรากฏหน้าจอดังภาพ



# Update Customer

BY CODEIGNITER

# Update Add Form

- ▶ ทำการเพิ่มตัวแปร action เพื่อใช้ในการตรวจสอบว่าเป็นการ New หรือเป็นการ Edit
- ▶ กรณี Action New Customer
  - ▶ กำหนด action path เป็น 'customer/save'
- ▶ กรณี Action Edit
  - ▶ กำหนด action path เป็น 'customer/update/#id' โดย #id จะเป็น id ของ customer ที่ส่งมาจาก controller โดยส่งมาใน Object \$editData
  - ▶ ทำการ Set Value ให้กับ field ต่างๆ ด้วย function set\_value และนำค่าที่ได้จาก \$editData มาใส่กรอบสีแดงดังภาพ

# Update View Customer Form

```
<div class="panel panel-default">
    <div class="panel-heading">Create Customer</div>
    <div class="panel-body">
        <?php
            $firstname = array('name'=>'firstname', 'id'=>'firstname', 'class'=>'form-control',
                'required'=>');
            $lastname= array('name'=>'lastname', 'id'=>'lastname', 'class'=>'form-control', 'required'=>');
            $email = array('type'=>'email', 'name'=>'email', 'id'=>'email', 'class'=>'form-control');

            if(isset($action) && $action == 'edit'){
                $action = "customer/update/$editData->custId";
                $firstname['value'] = set_value('firstname', $editData->firstname);
                $lastname['value'] = set_value('lastname', $editData->lastname);
                $email['value'] = set_value('email', $editData->email);

            }else{
                $action = 'customer/save';
            }

            echo form_open($action, array('id'=>'formCustomer', 'role'=>'form'));
            echo '<div class="form-group">';
            echo form_label('First Name', 'firstname');
            echo form_input($firstname);
            echo '</div>';
            echo '<div class="form-group">';
            echo form_label('Lastname ', 'lastname');
            echo form_input($lastname);
            echo '</div>';
            echo '<div class="form-group">';
            echo form_label('Email ', 'email');
            echo form_input($email);
            echo '</div>';
            echo form_button(array('type'=>'submit', 'class'=>'btn btn-info', 'content'=>'Save'));
            echo form_button(array('type'=>'reset', 'class'=>'btn btn-info', 'content'=>'Cancel'));
            echo form_close();

        ?>
    </div>
</div>
```

# Update Customer Model

- ▶ ทำการสร้าง function getDataByCriteria เพื่อทำการดึงค่า customer จาก database ตาม criteria ที่ส่งมา
- ▶ ทำการสร้าง function update เพื่อทำการ update ข้อมูล customer ดังภาพ

```
view_customer.php * Customer.php M_Customer.php
1 <?php
2
3 class M_Customer extends CI_Model{
4
5     private $table = 'tbl_customer';
6
7     public function __construct()
8     {
9         parent::__construct();
10    }
11
12    public function insert($data){
13        $this->db->insert($this->table,$data);
14    }
15
16    public function getAllData(){
17        $query = $this->db->get($this->table);
18        return $query->result();
19    }
20
21
22    public function getDataByCriteria($criteria,$limit=""){
23        $query = $this->db->get_where($this->table,$criteria);
24        return $query->result();
25    }
26
27    public function update($data,$criteria){
28        $this->db->update($this->table,$data,$criteria);
29    }
30
```

# Update Customer Controller

- ▶ เพิ่ม function ในการดึงข้อมูล Customer ชื่อว่า edit
- ▶ ทำการรับค่า customer id จาก url query string  
ด้วย url helper (`$this->uri->segment`)
- ▶ ทำการส่ง customer id ไปดึงข้อมูลโดยผ่าน function `getDataByCriteria`  
โดยส่ง criteria เป็น customer id
- ▶ ส่งผลลัพธ์ที่ได้ไปยัง view `create_customer` ดังภาพ

# Update Customer Controller

```
view_customer.php * Customer.php M_Customer.php
8     $this->load->model(array('M_Customer'=>'cust'));
9 }
10
11 public function index(){
12     $listData["list_data"] = $this->cust->getAllData();
13     $html['yeild'] = $this->load->view('view_customer',$listData,true);
14 }
15
16
17
18
19 public function create(){
20     $html['yeild'] = $this->load->view('create_customer','','true');
21     $this->load->view('layout.php',$html);
22 }
23
24 public function save(){
25     unset($this->input->post()['action']);
26     $this->cust->insert($this->input->post());
27     redirect('customer');
28 }
29
30 public function edit(){
31     $customerId = $this->uri->segment(3);
32     if($customerId){
33         $data = $this->cust->getDataByCriteria(array('custId'=>$customerId));
34         $editData['editData'] = $data[0];
35         $editData['action'] = 'edit';
36         $html['yeild'] = $this->load->view('create_customer',$editData,true);
37         $this->load->view('layout.php',$html);
38     }
39 }
40
41
42
```

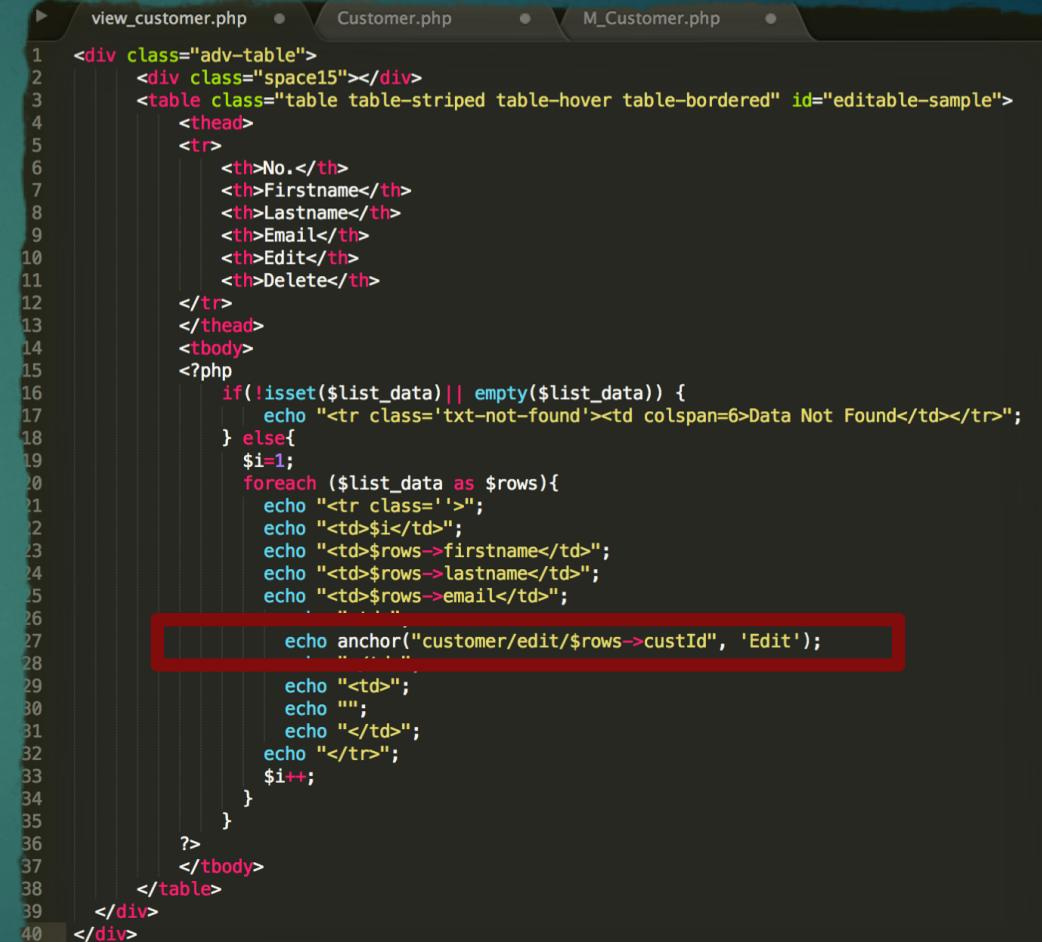
# Update Customer Controller

- ▶ ทำการสร้าง function update ในการส่งค่าข้อมูล customer ที่ทำการแก้ไขให้กับ function update ของ model ดังภาพ

```
view_customer.php * Customer.php * M_Customer.php
5  public function __construct()
6  {
7      parent::__construct();
8      $this->load->model(array('M_Customer'=>'cust'));
9
10
11
12  public function index(){
13      $listData["list_data"] = $this->cust->getAllData();
14      $html['yeild'] = $this->load->view('view_customer',$listData,true);
15  }
16
17
18  public function create(){
19      $html['yeild'] = $this->load->view('create_customer','','true');
20      $this->load->view('layout.php',$html);
21  }
22
23
24  public function save(){
25      unset($this->input->post()['action']);
26      $this->cust->insert($this->input->post());
27      redirect('customer');
28  }
29
30  public function edit(){
31      $customerId = $this->uri->segment(3);
32      if($customerId){
33          $data = $this->cust->getDataByCriteria(array('custId'=>$customerId));
34          $editData['editData'] = $data[0];
35          $editData['action'] = 'edit';
36          $html['yeild'] = $this->load->view('create_customer',$editData,true);
37          $this->load->view('layout.php',$html);
38      }
39  }
40
41
42  public function update(){
43
44      $customerId = $this->uri->segment(3);
45      $this->cust->update($this->input->post(),array('custId'=>$customerId));
46      redirect('customer');
47 }
```

# Update View Customer

- ▶ ทำการเพิ่ม link ในการ edit ในแต่ละ record โดยผ่าน function anchor โดย ส่ง URL = 'customer/edit/#id'
- ▶ #id เป็น customer id ที่ได้จาก Object \$list\_data กรอบแดง ดังภาพ



```
view_customer.php • Customer.php • M_Customer.php •
1 <div class="adv-table">
2   <div class="space15"></div>
3   <table class="table table-striped table-hover table-bordered" id="editable-sample">
4     <thead>
5       <tr>
6         <th>No.</th>
7         <th>Firstname</th>
8         <th>Lastname</th>
9         <th>Email</th>
10        <th>Edit</th>
11        <th>Delete</th>
12      </tr>
13    </thead>
14    <tbody>
15    <?php
16      if(!isset($list_data)|| empty($list_data)) {
17        echo "<tr class='txt-not-found'><td colspan=6>Data Not Found</td></tr>";
18      } else{
19        $i=1;
20        foreach ($list_data as $rows){
21          echo "<tr class='>";
22          echo "<td>$i</td>";
23          echo "<td>$rows->firstname</td>";
24          echo "<td>$rows->lastname</td>";
25          echo "<td>$rows->email</td>";
26          echo " <td><a href='customer/edit/$rows->custId'>Edit</a></td>";
27          echo " <td><a href='customer/delete/$rows->custId'>Delete</a></td>";
28        }
29      }
30    </tbody>
31  </table>
32 </div>
33 </div>
```

# Delete Customer

BY CODEIGNITER

# Update Customer Model

- ▶ ทำการเพิ่ม function ในการ delete customer ชื่อ delete โดยทำการรับค่า criteria
- ▶ ทำการเรียก function delete ของ database library ดังภาพ

```
<?php  
  
class M_Customer extends CI_Model{  
  
    private $table = 'tbl_customer';  
  
    public function __construct()  
    {  
        parent::__construct();  
    }  
  
    public function insert($data){  
        $this->db->insert($this->table,$data);  
    }  
  
    public function getAllData(){  
        $query = $this->db->get($this->table);  
        return $query->result();  
    }  
  
    public function getDataByCriteria($criteria,$limit=""){  
        $query = $this->db->get_where($this->table,$criteria);  
        return $query->result();  
    }  
  
    public function update($data,$criteria){  
        $this->db->update($this->table,$data,$criteria);  
    }  
  
    public function delete($criteria){  
        $this->db->delete($this->table,$criteria);  
    }  
}
```

# Update Customer Controller

- ▶ ทำการเพิ่ม function delete ทำการเรียก function delete model โดยทำการส่งค่า customer id ไปใน criteria ดังภาพ

```
0  public function edit(){
1      $customerId = $this->uri->segment(3);
2      if($customerId){
3          $data = $this->cust->getDataByCriteria(array('custId'=>$customerId));
4          $editData['editData'] = $data[0];
5          $editData['action'] = 'edit';
6          $html['yeild'] = $this->load->view('create_customer',$editData,true);
7          $this->load->view('layout.php',$html);
8      }
9  }
10
11  public function update(){
12
13      $customerId = $this->uri->segment(3);
14      $this->cust->update($this->input->post(),array('custId'=>$customerId));
15      redirect('customer');
16
17
18  public function delete(){
19      $this->cust->delete(array('custId'=>$this->uri->segment(3)));
20      redirect('customer');
21  }
22
23
24
```

# Update View Customer

- ▶ ทำการเพิ่ม link ในการ edit ในแต่ละ record โดยผ่าน function anchor โดยส่ง URL = 'customer/edit/#id'
- ▶ #id เป็น customer id ที่ได้จาก Object \$list\_data กรอบแดงดังภาพ

```
<?php
if(!isset($list_data) || empty($list_data)) {
    echo "<tr class='txt-not-found'><td colspan=6>Data Not Found</td></tr>";
} else{
    $i=1;
    foreach ($list_data as $rows){
        echo "<tr class='>";
        echo "<td>$i</td>";
        echo "<td>$rows->firstname</td>";
        echo "<td>$rows->lastname</td>";
        echo "<td>$rows->email</td>";
        echo "<td>";
        echo anchor("customer/edit/$rows->custId", 'Edit');
        echo "</td>";
        echo "<td>";
        echo anchor("customer/delete/$rows->custId", 'Delete');
        echo "</td>";
        echo "</tr>";
        $i++;
    }
}
?>
```

# Close Database Connection

- ▶ ทำการเปิด Customer Model  
(M\_Customer.php)
- ▶ ทำการสร้าง function \_\_deconstruct
- ▶ เรียก function \$this->db->close()  
เพื่อทำการ close connection  
database ดังภาพ

```
1  public function getDataByCriteria($criteria,$limit=""){
2      $query = $this->db->get_where($this->table,$criteria);
3      return $query->result();
4  }
5
6
7  public function update($data,$criteria){
8      $this->db->update($this->table,$data,$criteria);
9  }
10
11 public function delete($criteria){
12     $this->db->delete($this->table,$criteria);
13 }
14
15 public function __destruct() {
16     $this->db->close();
17 }
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39 }
```



Thank You

THANK YOU THANK YOU THANK YOU