１．xampp のバージョンを変更（5.6 -> 7.1）

1) <http://windows.php.net/download/> からphp のThread Safeのzip版をダウンロードする。

※ Windows が 64bit 環境であっても、XAMPP は 32bit なので、x64 ではなく x86 じゃないとダメ。

2)Apache の停止と PHP ファイルのコピー。

XAMPP内のphpフォルダの名前を変えておく。例えば、php-70

　　　zipを展開したフォルダをphpというフォルダ名にして、xamppのインストールフォルダに移す。

　　　3)古いPHPフォルダのphp.iniを新しいPHPフォルダにコピーする。

mysql.dll のextension を無効する。(php.ini のextension=php\_mysql.dllを無効化)

　　　4)apacheのphp設定をいじる。

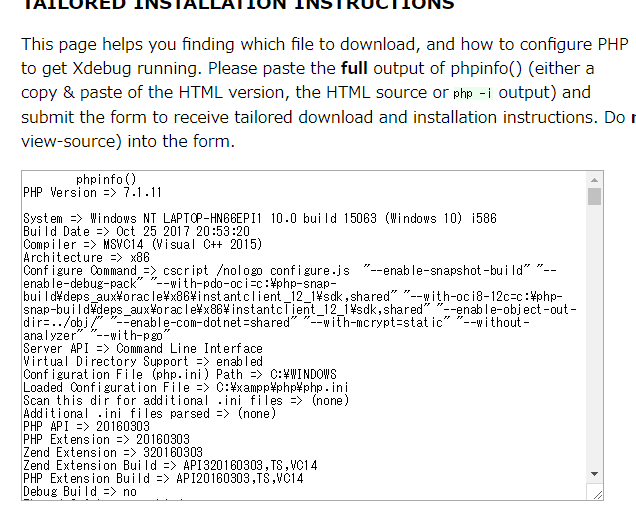
　　　デフォルトだと、apache/conf/extra/httpd-xampp.confのやつ。php5の設定になっているから、7に変えていく。5を7に置換していけばOK。

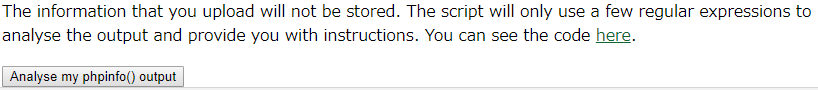
　　　5)apache起動させる

　　　※もし何のファイルを見えないというエーラがあれば、古いphpファイル中に探して、新しいのphpファイルにコピーしてください。

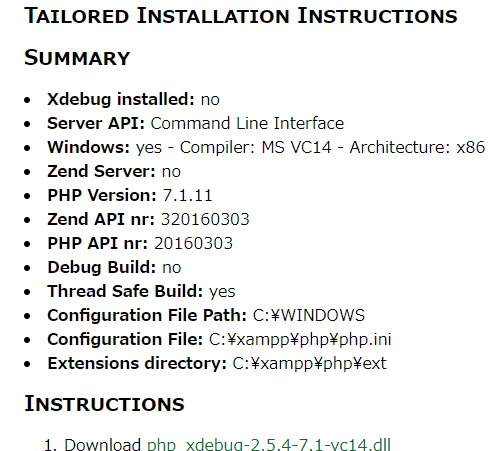
　　　以上の手順で、php7うまくいく。でも、php\_xdebug動きかない。

6) php\_xdebugの問題を解決ために。Xamppのshell を起動して、php -i でphpinfo()を獲得できる。後、全部のphpinfo()をコピーして、<https://xdebug.org/wizard.php>　にコピーする。





このボタンをクリックする。下の画面を見える。https://xdebug.org/wizard.php



このphp\_xdebugをダウンロードして、xampp/php/ext/の下にコピーする。名前をphp\_xdebug.dllに変更する。apacheを再起動する。okです。

２．CodeIgniter 環境を作成：

1. <https://codeigniter.com/download> からCodeIgniter 3.xをダウンロードする
2. ファイルを展開する。C:\xampp\apache\conf\extra\httpd-vhosts.confの中に下のコードを記入する。赤い部分は自分のcodeIgniterのindex.php のpath.(つまり上でダウンロードしたコードイグナイターのファイルの展開先は、下の書き方に則ると、C:\Users\staff\Documents\Duanの下に展開する)

　　　<VirtualHost \*:80>

##ServerAdmin webmaster@dummy-host2.example.com

DocumentRoot "C:\Users\staff\Documents\Duan\codeIgniter"

ServerName duan-codeIgniter.localhost

ErrorLog "logs/duan-codeIgniter.localhost-error.log"

CustomLog "logs/duan-codeIgniter.localhost-access.log" common

<Directory "C:\Users\staff\Documents\Duan\codeIgniter">

Options Indexes FollowSymLinks

AllowOverride All

Require all granted

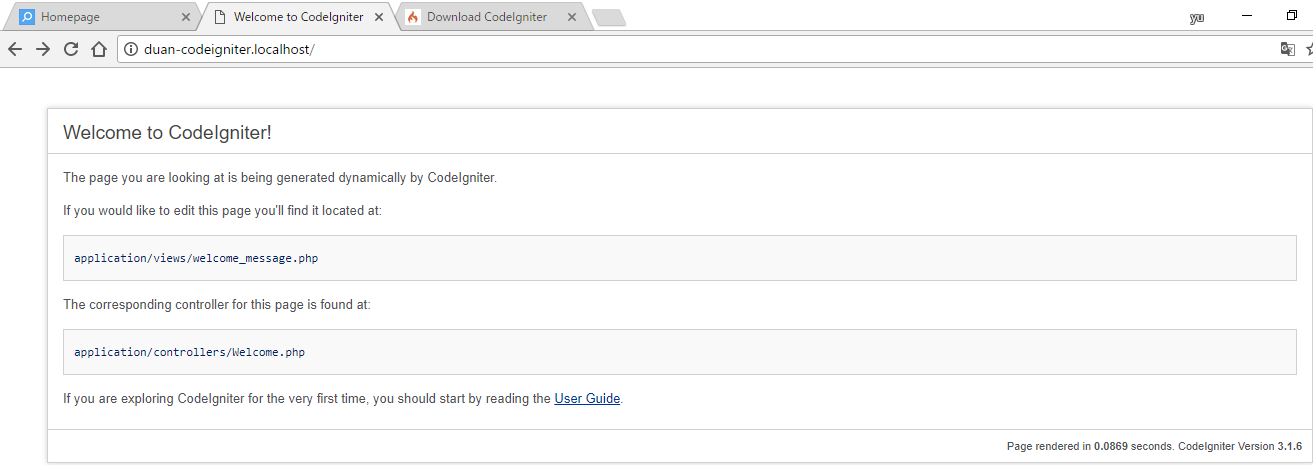
</Directory>

</VirtualHost>

3) C:\Windows\System32\drivers\etc\hosts の中にlocal URLを設定する。

　　例えば、下のコードを記入する。

　　127.0.0.1 duan-codeIgniter.localhost

　　後、Apacheを再起動して、[http://duan-codeigniter.localhost](about:blank) をchrome に訪問する。下の画面が見える。　　

　　/application/config/config.php　の下に、base\_urlを設定した。

　　$config['base\_url'] = '<http://duan-codeigniter.localhost>';　// 自分のbase\_urlを設定してください。

　 4) データベースを作成： データベース名前が　duan-sample　のデータベースを作成する。テーブルduanusersを作成する。構造は下のような表示される。/application/config/database.php の中に、データベースを設定する。例えば：(赤い部分は自分のデータベースの設定)

$active\_group = 'default';

$query\_builder = TRUE;

$db['default'] = array(

'dsn' => '',

'hostname' => 'localhost',

'username' => 'root',

'password' => '',

'database' => 'duan-sample',

'dbdriver' => 'mysqli',

'dbprefix' => '',

'pconnect' => FALSE,

'db\_debug' => (ENVIRONMENT !== 'production'),

'cache\_on' => FALSE,

'cachedir' => '',

'char\_set' => 'utf8',

'dbcollat' => 'utf8\_general\_ci',

'swap\_pre' => '',

'encrypt' => FALSE,

'compress' => FALSE,

'stricton' => FALSE,

'failover' => array(),

'save\_queries' => TRUE

);



5)controllerの作り方

　\application\controllers の下に、自分のコントロールを新規作成する。例えば：私はTop.php, Users.php　二つコントロールを作成した。コントロール基本な形は下のように表示される。

class Users extends CI\_Controller {

public function \_\_construct() {

parent::\_\_construct();

}

public function index() { //action

}

}

<http://duan-codeigniter.localhost/index.php/users> で、usersコントロールを訪問できる。

6)modelの作り方：

\application\models の下に、自分のmodelを新規作成する。例えば：Duanusers\_model.phpを作成した。テーブルduanusersと対応する。

<?php

class Duanusers\_model extends CI\_Model {

public function \_\_construct() {

$this->load->database();

}

}

7)view の作り方：

\application\views の下に、コントロールを対応のviewのフォルダを新規作成して、各action 対応の .php ファイルを作成する。例えば：usersフォルダを新規作成して、中に、index.php を新規作成。

便利のため、\application\views の下に、templatesフォルダを新規作成した。templatesフォルダの中に、footer.php と　header.phpを新規作成した。html の共用footerとheaderを設定した。

このview を訪問ために、controller の中に、下のような方法で訪問する。

$this->load->view('templates/header', $data);

$this->load->view('users/index', $data);　 // controller/action; $dataはviewに渡すのデータ

$this->load->view('templates/footer');

8) URL 、routeを設定：

<http://duan-codeigniter.localhost/index.php/users> で、usersコントロールを訪問できる。

URLのindex.php 部分を消したいので、

ソース・ファイルの下に、.htaccess を作成した。内容は

RewriteEngine On

RewriteBase /

RewriteCond %{REQUEST\_FILENAME} -s [OR]

RewriteCond %{REQUEST\_FILENAME} -l [OR]

RewriteCond %{REQUEST\_FILENAME} -d

RewriteRule ^.\*$ - [NC,L]

RewriteRule ^.\*$ index.php [NC,L]

[http://codeigniter.jp/user\_guide/3/general/routing.html](http://codeigniter.jp/user_guide/3/general/routing.html%E3%81%ABroutes.php)　[にroutes.php](http://codeigniter.jp/user_guide/3/general/routing.html%E3%81%ABroutes.php)　の設定原因と設定方法がある。私はただ　default\_controller　を設定した。

/application/config/routes.php下に、設定した。

$route['default\_controller'] = 'top/index';

３．CodeIgniter form を利用する：

　　１) controller の下に、actionの中に、下のコードを記入する。

　　　　$this->load->helper('form');

　　$this->load->library('form\_validation');

　　　　例えば： users controller のadd action　のコードは：

　　　　public function add() {

　　　$this->load->helper('form');

　　　$this->load->library('form\_validation');

$data['title'] = 'ユーザー新規作成';

$this->form\_validation->set\_rules('name', '名前', 'required');//form\_validation設定した。

if ($this->form\_validation->run() === FALSE) {

$this->load->view('templates/header', $data);

$this->load->view('users/add');

$this->load->view('templates/footer');

}

else {

$data = array(

'name' => $this->input->post('name'),

'gender' => $this->input->post('gender'),

'identity' => $this->input->post('identity')

);

$is\_insert = $this->db->insert('duanusers', $data); //DB insert操作

if($is\_insert){

redirect('/users/index');

}

}

}

view/add.php　のコードは：(赤い部分はform のkeywordである)

　　　各関数の説明：　<https://www.codeigniter.com/userguide3/helpers/form_helper.html#>

<?php echo validation\_errors(); ?>

<!--form→

<?php echo form\_open('users/add'); ?>

<?php echo form\_label('名前：', 'name');?>

<!--text-->

<?php echo form\_input('name');?>

<?php echo br(1)?>

<?php echo form\_label('性別：', 'gender');?>

<!--radio-->

男<?php echo form\_radio('gender', '0', TRUE);?>

女<?php echo form\_radio('gender', '1');?>

<?php echo br(1)?>

<?php echo form\_label('身分：', 'identity');?>

<!--select-->

<?php echo form\_dropdown('identity', array(

'0' => '学生',

'1' => '教師',

'2' => '係員',

'3' => '訪問者',

));?>

<?php echo br(1)?>

<?php echo form\_submit('submit', '新規作成');?>

</form>

4．TCPDF を利用する(PDFを生成する)：

下のURLの手順で、行ける。

<https://github.com/bcit-ci/CodeIgniter/wiki/TCPDF-Integration>

でも、私はダウンロードしたファイル違い場所(赤い部分)に置くので、下に、説明する。

1)Download TCPDF - <http://sourceforge.net/projects/tcpdf/files/>

2) Unzip the above download into /application/third\_party/tcpdf　 Make sure that the main tcpdf.php file is in this directory

3)Create a new file: /application/libraries/Pdf.php:

<?php

if ( ! defined('BASEPATH')) {

exit('No direct script access allowed');

}

require\_once APPPATH."/third\_party/tcpdf/tcpdf.php";

class Pdf extends TCPDF {

public function \_\_construct() {

parent::\_\_construct();

$this->setFont('cid0jp');

}

}

日本語を表示ために、したの設定を必要とする。

　　 $this->setFont('cid0jp');

　　今までの手順で、TCPDFの配置が終わった。私はサンプルUsersPdf.php を作成した。

５．PHPExcel を利用する(Excelを生成する)：

下のURLの手順で、行ける。　<http://www.ahowto.net/php/easily-integrateload-phpexcel-into-codeigniter-framework/>

でも、私はダウンロードしたファイル違い場所(赤い部分)に置くので、下に、説明する。

1)download PHPExcel from <https://github.com/PHPOffice/PHPExcel>

2)中のClasses folder　中の内容を抽出する。/application/third\_party/PHPExcel フォルダを作成して、抽出した内容をapplication/third\_party/PHPExcel の下にコピーする。

3) Create new PHP file inside /application/libraries/ name it Excel.php :

<?php

if ( ! defined('BASEPATH')) {

exit('No direct script access allowed');

}

require\_once APPPATH."/third\_party/PHPExcel/PHPExcel.php";

class Excel extends PHPExcel {

public function \_\_construct() {

parent::\_\_construct();

}

}

今までの手順で、PHPExcelの配置が終わった。私はサンプルUsersExcel.php を作成した。

6．ユーザー認証機能：

1. <https://github.com/benedmunds/CodeIgniter-Ion-Auth> からlon-Authをダウンロードする。中に、配置手順がある。
2. 下のようにダウンロードしたファイルを配置する。:



　　3) コントロールの中に、下のコードを記入する。

$this->load->add\_package\_path(APPPATH.'third\_party/ion\_auth/');  
 $this->load->library('ion\_auth’);

例えば：Auth.phpの下に：(赤い部分)

public function \_\_construct()

{

parent::\_\_construct();

$this->load->add\_package\_path(APPPATH.'third\_party/ion\_auth/');

$this->load->database();

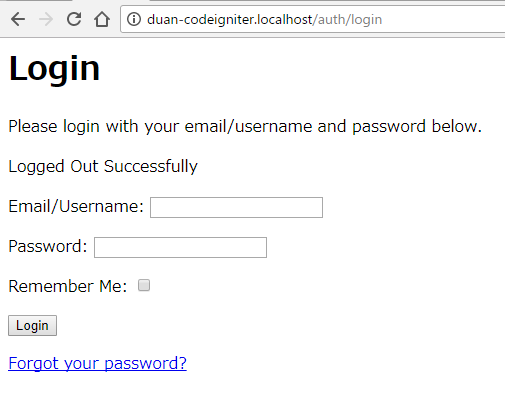
$this->load->library(array('ion\_auth', 'form\_validation'));

$this->load->helper(array('url', 'language'));

$this->form\_validation->set\_error\_delimiters($this->config->item('error\_start\_delimiter', 'ion\_auth'), $this->config->item('error\_end\_delimiter', 'ion\_auth'));

$this->lang->load('auth');

}



admin の判断：/application/config/ion\_auth.phpに設定する。

　$config['admin\_group'] = 　array('admin');

　実際に動きの関数は：　　　C:\Users\staff\Documents\Duan\codeIgniter\application\third\_party\ion\_auth\libraries\Ion\_auth.php　中のis\_admin

public function is\_admin($id = FALSE)

{

$this->ion\_auth\_model->trigger\_events('is\_admin');

$admin\_group = $this->config->item('admin\_group', 'ion\_auth');

return $this->in\_group($admin\_group, $id);

}

　もしadmin 以外のgroup認証したいれば、自分でis\_xxx() 関数を作成する。データのgroupsとusers-groups　を記入必要。

controller の中にユーザー一権限区分する時、

if (!$ion\_auth->is\_admin()) {

}