

10. now it is time for validation of our input of the form.

10.1 to apply validation we need to write rules in the function that handles the form.

10.2 lets apply some rule for our inputs

10.3 need to change the user_register function

10.4 need to create a function to check user name in the User Controller

10.5 need to create a function to check user name in the User_model

10.6 change the register view

10.3 lets change our User controller as highlighted

<?php

```
class User extends CI_Controller{
    function __construct(){
        parent::__construct();
        $this->load->model('User_model');
        $this->load->library('form_validation');
        if (!$this->session->userdata('language'))
            $this->lang->load('main');
        else
            $this->lang->load('main', $this->session->
userdata('language'));
    }
    function index(){
        $this->load->view('includes/langbar');
        $this->session->set_userdata('last_visited', current_url());

        echo lang('par_about');
        $session_data = $this->session->userdata();
        print_r($session_data);
    }

    /*
    * thisfunction used to loas a view called rigister located in Form
    directory of views
    */

    function register($err=''){
        $data['main_title'] = mb_convert_case(lang('title_main'),
MB_CASE_TITLE);
        $data['contents'] = 'Form/register';
        $data['error'] = $err;
        $this->load->view('main_view', $data);
    }

    // this function is to check the existance of user name in the database.
    function username_check($username)
    {
        // call a User_model function for checking the user name
        $user = $this->User_model->check_username($username);
    }
}
```

```
// if user name existed in the database than show this message
if ($user)
{
    $this->form_validation->set_message('username_check', 'The'.
$username.' Already Exist in Database');
    return FALSE;
}
else
{
    return TRUE;
}
}
```

```
/*
 * this function is to create new user and the form data is sent to this
function from register.php file
 * and send to user table of blog_data database
 */
```

```
public function register_user()
{

    $this->load->helper('date');

    if($this->input->post('submit')){
```

```
// load the form validation library
$this->load->library('form_validation');
// set rules for desired inputes
$config = array(
    array(
        'field'    => 'name',
        'label'    => 'First Name',
        'rules'    => 'required'
    ),
    array(
        'field'    => 'uname',
        'label'    => 'User Name',
        'rules'    => 'required|callback_username_check'
    ),
    array(
        'field'    => 'password',
        'label'    => 'Password',
        'rules'    => 'required'
    ),
    array(
        'field'    => 'email',
        'label'    => 'Email',
        'rules'    => 'required'
    )
);
// initialize the rules
$this->form_validation->set_rules($config);
// if the form do not be as in the rule call register function
if ($this->form_validation->run() == FALSE)
{
    $this->register();
}
```



```

/*
 * this function take login data and send to database and than retrieve the
desired field and compare
 * with these data
 */

```

```

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

    if ($this -> input -> post('remember') == 'yes') {
        // set a cookie with the username. It will expire in 269200
seconds(3 days).
        $cookie = array('name' => 'username', 'value' => $username,
'expire' => 259200);
        $this -> input -> set_cookie($cookie);
    } else {
        // delete the cookie
        $cookie = array('name' => 'username', 'value' => NULL, 'expire'
=> NULL);
        $this -> input -> set_cookie($cookie);
    }
    $uid = ($this -> User_model->login($username, $password));
    if (!$uid){
        redirect(site_url('home/get_posts'));
    }

    // if login successful than send the all data related to the user to
session array
    $this -> session -> set_userdata('uid', $uid);
    $this -> session -> set_userdata('user', $this->
User_model->get_userdata($uid));
    $this -> session -> set_userdata('login', true);
    //print_r($this->session->all_userdata());
    redirect($this -> session -> userdata('last_visited'));
}
//-----

```

```

// this function is called by log out button and delete the session array
public function logout() {
    $this -> session -> unset_userdata('login');
    $this -> session -> unset_userdata('user');
    $this -> session -> unset_userdata('uid');

    redirect($this -> session -> userdata('last_visited'));
}
//-----

```

```

public function lang($language) {
    $this -> session -> set_userdata('language',$language);
    redirect($this -> session -> userdata('last_visited'));
}
//-----

```

```

public function view($page) {
    $this -> session -> set_userdata('last_visited', current_url());
}

```

```

        // if no content to show for $page, show 404 now.
        //if (!file_exists('application/views/content/' . $page . '.php'))
        //    show_404();
        // try to translate the title, then capitalize it.
        $data['title'] = mb_convert_case(lang('title_' . $page),
MB_CASE_TITLE);
        $data['main_title'] = mb_convert_case(lang('title_main'),
MB_CASE_TITLE);
        $data['contents']='includes/' . $page;
        $this->load->view('main_view', $data);
    }
}

```

10.4 create new function in the User_model called it check_username as highlighted below

<?php

```
class User_model extends CI_Model{
```

```

    /*
    * this function is for creating new User
    */
    function get_users(){
        return $this->db->get('user')->result();
    }
    function add_user($data){

        $this->db->insert('user',$data);
    }

```

```
// this function check the database for existing record of user name
```

```

    function check_username($username){

        $this->db->where('user_name',$username);
        $this->db->select('user_name');
        $query = $this->db->get('user');
        if ($query->num_rows() > 0) {
            return true;
        } else{
            return FALSE;
        }
    }
}

```

```

    /*
    * this function is to compare user name from form coming with existing
    values in th database
    */

```

```

    function login($username, $password){

        $array = array('user_name' => $username, 'password' => $password) ;
        $this->db->where($array);
        $this->db->select('user_id');
        $query = $this->db->get('user');
        if ($query->num_rows() > 0) {
            return $query->first_row()->user_id;
        } else{
            return 0;
        }
    }
}

```

```

    }

}
/*
 * this function is to return all user data
 */
function get_userdata($uid) {
    $this->db->where('user_id', $uid);
    $query = $this->db->get('user');
    if ($query->num_rows() > 0) {
        return $query->first_row();
    } else {
        return false;
    }
}
}

```

10.6 now change our register view so go to
c/wamp/www/CI/application/views/form/register.php
<h1> Add User </h1>

```

<?php echo form_open_multipart('User/register_user');?>

<table>
    <tr>
        <td><label for="name">Name:* </label></td>
<!-- set_value is for if the error ocured the filled input remain-->
        <td><input type="text" id="name" name="name" size="30" value="<?php
echo set_value('name'); ?>" /></td>
<!-- show the error if form error exist-->
        <td><?php echo form_error('name'); ?></td>

    </tr>
    <tr>
        <td><label for="lname">Last Name</label> </td>
        <td><input type="text" id="lname" name="lname" size="30" value="<?php
echo set_value('lname'); ?>" /></td>
        <td><?php echo form_error('lname'); ?></td>
    </tr>
    <tr>
        <td><label for="uname">User Name:*</label> </td>
        <td><input type="text" id="uname" name="uname" size="30" value="<?php
echo set_value('uname'); ?>" /></td>
        <td><?php echo form_error('uname'); ?></td>
    </tr>
    <tr>
        <td><label for="password">Password:* </label> </td>
        <td><input type="password" id="password" name="password" size="30"
value="<?php echo set_value('password'); ?>" /></td>
        <td><?php echo form_error('password'); ?></td>
    </tr>
    <tr>
        <td><label for="email">Email:*</label> </td>
        <td><input type="email" id="email" name="email" size="30" value="<?php
echo set_value('email'); ?>" /></td>
        <td><?php echo form_error('email'); ?></td>
    </tr>

```

```
<tr>
    <td><label for="file">Image: *</label> </td>

    <td><input type="file" id="file" name="userfile" size="30" /><div> <?
php if (isset($error) ) { echo $error;}?></div></td>

</tr>

</table>

<input type="submit" name="submit" value="Save Changes" />
```