

Creating forms in CodeIgniter

Topics : [Codeigniter](#)

Written on [March 01, 2024](#)

Creating forms in CodeIgniter involves generating HTML form elements and handling form submissions. Here's a step-by-step guide to creating forms in CodeIgniter:

1. Create a View File for the Form:

- Create a new PHP file in the `application/views` directory to represent your form.
- Define the HTML markup for the form elements inside the view file.

Example view file (`my_form.php`):

```
<!DOCTYPE html>
<html>
<head>
    <title>My Form</title>
</head>
<body>
    <h2>My Form</h2>
    <?php echo form_open('form/process'); ?>

    <label for="username">Username:</label>
    <input type="text" name="username" id="username" required>

    <label for="email">Email:</label>
    <input type="email" name="email" id="email" required>

    <input type="submit" value="Submit">

    <?php echo form_close(); ?>
</body>
</html>
```

2. Load the Form View in a Controller:

- Create a controller method to load the form view.
- Load the form view using the `$this->load->view()` method.

Example controller method:

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

3. Process Form Submissions:

- Create another controller method to handle form submissions.
- Use CodeIgniter's form validation library to validate form input.
- Process the form data accordingly (e.g., save to database, send email).

Example controller method to process form submissions:

```
public function process() {  
    // Load form validation library  
    $this->load->library('form_validation');  
  
    // Set validation rules  
    $this->form_validation->set_rules('username', 'Username', 'required');  
    $this->form_validation->set_rules('email', 'Email',  
'required|valid_email');  
  
    if ($this->form_validation->run() == FALSE) {  
        // Form validation failed, reload the form  
        $this->load->view('my_form');  
    } else {  
        // Form validation passed, process the form data  
        $username = $this->input->post('username');  
        $email = $this->input->post('email');  
  
        // Process the form data (e.g., save to database, send email)  
  
        // Redirect or show success message  
    }  
}
```

4. Display Validation Errors (Optional):

- If form validation fails, display validation errors next to the form fields.

Example of displaying validation errors in the view file:

```
<?php echo validation_errors(); ?>
```