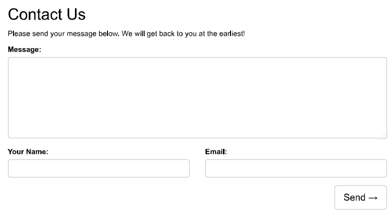
A Simple PHP Contact Form Code



Contact Us

Please send your message below. We will get back to you at the earliest!

Preview the form (http://reusableforms.com/r/e/php-contact-form-code)

Download (http://reusableforms.com/dl/e/php-contact-form-code)

Usage:

1. Press the download button above. The zip file contains all the code you need for the form.
2. Unzip the file php-contact-form-code
3. Open the file named "handler.php"
   Look for `sendEmailTo` add the email addresses to receive the form submissions.
4. Upload the whole folder to your website
5. Open the formpage.html in your browser and test

See the customization guide (http://reusableforms.com/doc/customization) . You can change the validations, edit the styles, and more

See a video demo here (https://youtu.be/jAr4ZtLnmn0)

(https://simfatic.com)

The sections below explain the code of this form

The HTML Code

```
<div class="row">
    <div class="col-md-6 col-md-offset-3" id="form_container">
        <h2>Contact Us</h2>
        <p>
           Please send your message below. We will get back to you at the earliest!
        </p>
        <form role="form" method="post" id="reused_form">

            <div class="row">
                <div class="col-sm-12 form-group">
                    <label for="message">
                        Message:</label>
                    <textarea class="form-control" type="textarea" id="message" name="me
                </div>
            </div>
            <div class="row">
                <div class="col-sm-6 form-group">
                    <label for="name">
                        Your Name:</label>
                    <input type="text" class="form-control" id="name" name="name" requir
                </div>
                <div class="col-sm-6 form-group">
                    <label for="email">
                        Email:</label>
                    <input type="email" class="form-control" id="email" name="email" rec
                </div>
            </div>


            <div class="row">
                <div class="col-sm-12 form-group">
                    <button type="submit" class="btn btn-lg btn-default pull-right" >Ser
                </div>
            </div>

        </form>
        <div id="success_message" style="width:100%; height:100%; display:none; ">
            <h3>Posted your message successfully!</h3>
        </div>
        <div id="error_message"
                style="width:100%; height:100%; display:none; ">
                    <h3>Error</h3>
                    Sorry there was an error sending your form.

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

CSS Styling

The following extra CSS styling is used to make the form look better

```
<link rel="stylesheet" href="form.css" >
```

jQuery Form submission handling

```
<script>
$(function()
{
    function after_form_submitted(data)
    {
        if(data.result == 'success')
        {
            $('form#reused_form').hide();
            $('#success_message').show();
            $('#error_message').hide();
        }
        else
        {
            $('#error_message').append('<ul></ul>');

            jQuery.each(data.errors,function(key,val)
            {
                $('#error_message ul').append('<li>'+key+':'+val+'</li>');

            });
            $('#success_message').hide();
            $('#error_message').show();

            //reverse the response on the button
            $('button[type="button"]', $form).each(function()
            {
                $btn = $(this);
                label = $btn.prop('orig_label');
                if(label)
                {
                    $btn.prop('type','submit' );
                    $btn.text(label);
                    $btn.prop('orig_label','');
                }
            });

        }//else
    }

    $('#reused_form').submit(function(e)
      {
        e.preventDefault();

        $form = $(this);
        //show some response on the button
        $('button[type="submit"]', $form).each(function()
        {
            $btn = $(this);
            $btn.prop('type','button' );
            $btn.prop('orig_label',$btn.text());
            $btn.text('Sending ...');
        });


                    $.ajax({
                type: "POST",
                url: 'handler.php',
                data: $form.serialize(),
                success: after_form_submitted,
                dataType: 'json'
            });

      });
});
</script>
```

Form Validations

HTML5 validations are used. For example, the Name and Email fields have 'required' validation and the email field is of input type 'email'.

```
<input type="email" class="form-control" id="email" name="email" required>
```

Similarly, the message field (textarea) allows a max length of 6000 characters

```
<textarea  name="Message"  maxlength="6000" required></textarea>
```

using the built-in HTML5 validations has the advantage that the browser itself shows the error message. You can customize those validations to fit your needs

Server Side Handling

Every form requires a server side script to collect the form data and do the processing like send emails, save to database etc. This form comes with a PHP script to handle the form submissions. PHP is widely supported server side scripting platform.
When the form is submitted, the javascript form submission event handler above collects the form data and sends it to the server side script.

The serverside script entry point is handler.php (see in your downloaded zip file). The script uses a library called FormHandler, which inturn, uses other libraries.

Here is the code of the handler.php

```
<?php
ini_set('display_errors', 1);
ini_set('display_startup_errors', 1);
error_reporting(E_ALL);
/*
Tested working with PHP5.4 and above (including PHP 7 )

 */
require_once './vendor/autoload.php';

use FormGuide\Handlx\FormHandler;



$pp = new FormHandler();

$validator = $pp->getValidator();
$validator->fields(['name','email'])->areRequired()->maxLength(50);
$validator->field('email')->isEmail();
$validator->field('message')->maxLength(6000);





$pp->sendEmailTo('someone@gmail.com'); // ← Your email here

echo $pp->process($_POST);
```

You have to edit the handler.php and change "someone@gmail.com" to your email address. If you want to add a second or third email address, you can do so like this:

```
$fh->sendEmailTo(['someone@gmail.com', 'someone.else@gmail.com']);
```

Server Side Form Validations

Server side form validations are essential for any form. You can't trust data sent to a form processing script.
For this form, the server side form validations are done using this PHPFormValidation library. See the documentation to add or update the validations.

Customizations

Edit the handler.php and change abc@cdef to your email address. Remove the second email address, if not required.
See the customization guide (http://reusableforms.com/doc/customization) for more customization options.

Preview the form (http://reusableforms.com/r/e/php-contact-form-code)

Download (http://reusableforms.com/dl/e/php-contact-form-code)

Forms with File Upload/Attachment (http://reusableforms.com/category/attachment-forms)
Forms with Captcha Validation (http://reusableforms.com/category/captcha-forms)
Forms with ReCaptcha Support (http://reusableforms.com/category/recaptcha-forms)
Popup Forms (http://reusableforms.com/category/popup-forms)
Contact Forms (http://reusableforms.com/category/contact-forms)
Email Forms (http://reusableforms.com/category/email-forms)
Feedback Forms (http://reusableforms.com/category/feedback-forms)

Privacy Policy (http://reusableforms.com/privacy)
Contact Us (http://reusableforms.com/contact)
About (http://reusableforms.com/about)