

Form Validation using jQuery

Forms aren't much use if the person submitting the form either forgets to put information in or it is formatted incorrectly.

Form validation requires checking the form fields ***before*** the form is submitted to make sure that required information is formatted correctly.

If everything is fine, the form info travels to the web server.
If there's a problem, the script stops the submission process and displays the errors on the page.

Twitter / Create an Account

Twitter, Inc. Reader Google

Twitter / Create an Account

Have an account? [Sign in](#)

It's your turn.

Join Twitter.

me

✓ Name looks great.

dkjasd

✗ Doesn't look like a valid email.

••••••

✓ Password could be more secure.

me

✗ This username is already taken!

Suggestions: dkjasd · DkjasdMe · MeDkjasd · dkjasd_me

☒ Keep me logged-in on this computer.

By clicking the button, you agree to the terms below:

These Terms of Service ("Terms") govern your access to and use of the services and Twitter's websites (the "Services"), and any information, text, graphics, photos or other materials

Create my account

Printable version:

[Terms of Service](#) · [Privacy Policy](#)

Note: Others will be able to find you by name, username or email. Your email will not be shown publicly. You can change your privacy settings at any time.

Basic Form Validation

(using jQuery)

using jquery form validation

(1)

download and attach the **jquery.js** file
to the page containing the form that
you wish to validate

attach just like any other external js file

```
<script type="text/javascript"  
src="js/jquery.js">  
</script>
```

using jquery form validation

(2)

download and attach the validation plugin

<http://bassistance.de/jquery-plugins/jquery-plugin-validation/>

you really only need the **jquery.validate.js** file.

attach just like any other external js file.

```
<script type="text/javascript"  
src="js/jquery.validate.js">  
</script>
```

using jquery form validation

(3)

add validation rules

these are the instructions that assign
which fields get validated and how

jQuery validation methods

The validation plugin includes methods that cover the most common validation needs:

<i>required</i>	form won't be submitted unless the field is completed
<i>date</i>	information must be in the format MM/DD/YYYY
<i>url</i>	must be a full valid web address
<i>email</i>	must be formatted like an email address but doesn't check to make sure the address is real
<i>number</i>	must be a number, however the input cannot include symbols so \$45 would be invalid
<i>digits</i>	can only include positive integers.... 20 is valid but 10.73 or -12 are valid
<i>credit card</i>	must be a validly formatted credit card number

the simplest way to validate an input field is to assign one (or more) of the class names to the form element.

original code would look like this...

```
<input name= "name" id="name" type="text">
```

code with 'required' class attribute would look like this...

```
<input name= "name" id="name"  
type="text" class= "required">
```

you might also want to make a field required
AND contain a specific type of information

(note the space separating the classes)

code with 'required' class attribute:

```
<input name= "name" id="name"  
type="text" class= "required">
```

code with 'required' and 'date' class attributes:

```
<input name= "name" id="name"  
type="text" class= "required date">
```

using jquery form validation

(4)

add error messages
(optional)

The validation plugin supplies generic error messages...however if you may customize the error messages to give further instruction

To customize error messages, add a *title* attribute to the form field and supply the error message as the title's value.

original code would look like this...

```
<input name= "name" type= "text" class= "required">
```

code with error message would look like this...

```
<input name= "name" type= "text" class= "required"  
      title= "please give us your name." >
```

using jquery form validation

(5)

apply the *validate()* function to the form

the Validation plug-in includes a function
that makes all the magic happen.

To apply it, use jQuery to select the form
then attach the function to that selection.

example:

if you had a form with an ID of signup your entire script for basic form validation could look like this...

```
<script type= "text/javascript" src= "js/jquery.js"></script>
<script type= "text/javascript" src= "js/jquery.validate.js"></script>

<script type= "text/javascript">
$(document).ready(function() {
$('#signup').validate();
});
</script>
```

Preview your form in the browser.

Don't fill in any values and click submit.

You should get error messages on your form fields. If you did not, something is wrong... :-/
go back and check...

Styling error messages

use CSS :) of course

in your external css file, create a class that will define the look for the error message

```
label.error {color:#F00; font-style: italic;}
```

you can also create a class that will define the look for the error fields

```
input.error, select.error { border: 1px red solid;}
```


jQuery form selectors

selector	example	what it does
:input	\$(':input')	selects all input areas
:text	\$(':text')	selects all text fields
:password	\$(':password')	selects all password fields
:radio	\$(':radio')	selects all radio buttons
:checkbox	\$(':checkbox')	selects all checkboxes
:submit	\$(':submit')	selects all submit buttons
:image	\$(':image')	selects all image buttons
:reset	\$(':reset')	selects all reset buttons
:file	\$(':file')	selects all file fields
:hidden	\$(':hidden')	selects all hidden fields

Advanced Form Validation

(using jQuery)

but why?
when basic validation is so easy :)

advanced validation allows you to assign
different error messages for different
validation problems or require a specific
number of characters for input.

(this could be useful for creating passwords)

find the \$('form1').validate(); and add

```
$('#form1').validate({
```

```
    rules: {
```

```
    }, //end rules
```

```
    messages: {
```

```
    } //end messages
```

```
});<!--end validate-->
```

advanced validation

minlength	minlength:6
maxlength	maxlength:100
rangelength	rangelength:[6,100]
min	min:10
max	max:1000
range	range:[10,1000]
equalTo	equalTo: '#name'

```
rules: {  
    password: {  
        required: true,  
        rangelenh: [8, 16]  
    },  
    password2: {  
        equalTo: '#password'  
    }  
}, //end rules
```

```
messages: {  
    password: {  
        required: 'please type a password',  
        rangelength: 'password must be 8 to 16  
characters long'  
    },  
    password2: {  
        equalTo: 'the two passwords do not match'  
    }  
}, //end messages
```

