<form> Tag for HTML Forms

#form# creates an HTML form. #form# description, syntax, usage, attributes and examples. HTML-5.com is a great guide for web developers. TV Series & Actors and Actresses. Follow TV Series and HTML 5 on Google+.

HTML-5.com ➤ itemscopehttp://data-vocabulary.org/BreadcrumbHTML 5 ➤ itemscopehttp://data-vocabulary.org/BreadcrumbHTML Tags ➤ itemscopehttp://data-vocabulary.org/Breadcrumb **<form> Tag for HTML Forms**

➤ itemscopehttp://data-vocabulary.org/BreadcrumbExamples

The <form> Tag in HTML 5

The <form> tag is used to create an input form that the user can interact with. The interaction with the user can be within the browser client only, using client-side scripts, or with a web server. For example, a <dfn>fill-in form</dfn> can consist of a set of labeled text input fields along with a <dfn>submit button</dfn> that transmits the data to the web server.

<form> <fieldset> <legend>Login credentials</legend> <label for="ex1user">Username:</label> <input id="ex1user" name="username" required="required" autofocus="autofocus" size="30" maxlength="15" placeholder="Your username"></input>
<label for="ex1pass">Password: </label> <input id="ex1pass" name="password" required="required" type="password" size="30" maxlength="15" placeholder="Your password"></input> </fieldset> <fieldset> <legend>Contact information</legend> <label for="ex1addr">Email address:</label> <input id="ex1addr" name="emailaddr" required="required" type="email" size="60" maxlength="80" placeholder="Your e-mail address"></input>
<label for="ex1site">Web address: </label> <input id="ex1site" name="siteurl" required="required" type="url" size="60" maxlength="80" placeholder="Your web site URL"></input> </fieldset> <label for="ex1date">Date:</label> <input id="ex1date" name="date" type="date"></input>   <label for="ex1cmnts">Comments:</label> <textarea id="ex1cmnts" placeholder="Your comments" cols="40" rows="3"></textarea>
<label for="ex1age1">Age:</label> <input id="ex1age1" type="range" min="18" max="99" value="25" onchange="document.getElementById('ex1age2').value = this.value"></input> <input id="ex1age2" size="3" value="25" onchange="if (this.value < 18) this.value = 18; if (this.value > 99) this.value = 99; document.getElementById('ex1age1').value = this.value"></input> </form>
This is an actual working demo of the <form> example code below.

Since `autofocus` automatically positions the cursor to the input field, the `placeholder` text for the Username field will not appear unless you tab to the masked password field or click elsewhere. If your browser does not support the HTML 5 `placeholder` attribute, none of the placeholders will appear.

Note: The first browser where placeholder text in form input fields actually works is Chrome. It may not work yet in other browsers.

back to top

‹form› Tag Syntax

```
<a mode="pre" href="../body-tag/index.html#syntax"><body></a>
    ...
    <a mode="pre" href="../html-tag-list.html#flow-content-model">... flow content expected ...</a>
    <b mode="pre"><form id="form-id" method="GET|POST|etc." action="target-URL"></b>
        <a mode="pre" href="../../definitions/index.html#flow-content">... flow content ...</a>
    <b mode="pre"></form></b>
    ...
  </body>
```

Rules for coding HTML form elements

Make sure you understand the difference between a tag and element and are familiar with the definitions of namespace and other HTML terms.

1. Include a **form** element where flow content is expected.
2. Begin the **form** element with a starting ‹form› tag. The element name uses lower case letters and should be in the HTML namespace, which it will pick up automatically from the `xmlns` attribute on the ‹html› tag.
3. Inside the **form** element, between the starting `<form>` tag and the ending `</form>` tag, code the inner HTML flow content.
4. End the **form** element with a matching `</form>` closing tag.

Content Model

The content of the **form** element can include HTML comments, text content and any tags that can be used in flow content.

back to top

‹form› Tag Attributes

Attributes of the ‹form› tag

global attributes	In addition to the personal attributes of the ‹form› tag below, any of the common HTML attributes can also be coded.
`accept-charset`	
`action`	a URI reference (or IRI reference depending on the `encoding`) for the destination that is to process the submitted form data
	Use percent escape codes as explained in the URL Encoding Tutorial for any special characters in the URI reference.
	If the value of the `action` attribute resolves to an HTTP URI, it may contain any of the following components:
	1. protocol scheme, typically `http:` or else `https:` 2. username, followed by an "@" 3. host name or IP address

	4. port number, which defaults to 80 for the `http` scheme and 443 for the `https` scheme 5. absolute or relative path 6. search query, indicated by "?" 7. fragment identifier, indicated by "#" if the response from the form submission has placemarks If the protocol scheme, username, host name/IP address and port number are omitted the default is the current host - the same server as the base of the current document. If the path starts with a slash `/...`, it is an absolute path from the document root directory ([AKA](#) "\<dfn\>web root\</dfn\>") on the server. A relative path will be resolved relative to the base of the current document. A URI with a "#" but no fragment identifier points to the top of the response document.
`autocomplete="on"` `autocomplete="off"`	

`enctype="application/x-www-form-urlencoded"`
`enctype="multipart/form-data"`
`enctype="text/plain"`	

`method="GET"`
`method="POST"`
`method="PUT"`
`method="DELETE"`	

`name`	
`novalidate="novalidate"`	
`target`	

[back to top](#)

\<form\> Tag Examples

Examples of the `form` tag in HTML 5

Input field with placeholder text and autofocus on input field in HTML form

```
<form>
   <style scoped="scoped">
      input::-webkit-input-placeholder, textarea::-webkit-input-placeholder
         { font-family: Verdana; color: #006633; font-style: italic }
      input:-moz-placeholder, textarea:-moz-placeholder
         { font-family: Verdana; color: #006633; font-style: italic }
   </style>
```

```
    <fieldset>
        <legend>Login credentials</legend>
        <label for="ex1user">Username:</label>
        <input id="ex1user" name="username" required="required"
autofocus="autofocus"
            size="30" maxlength="15"
            placeholder="Your username"
        /><br/>
        <label for="ex1pass">Password: </label>
        <input id="ex1pass" name="password" required="required"
            type="password" size="30" maxlength="15"
            placeholder="Your password"
        />
    </fieldset>
    <fieldset>
        <legend>Contact information</legend>
        <label for="ex1addr">Email address:</label>
        <input id="ex1addr" name="emailaddr" required="required"
            type="email" size="60" maxlength="80"
            placeholder="Your e-mail address"
        /><br/>
        <label for="ex1site">Web address: </label>
        <input id="ex1site" name="siteurl" required="required"
            type="url" size="60" maxlength="80"
            placeholder="Your web site URL"
        />
    </fieldset>
    <label for="ex1date">Date:</label>
    <input id="ex1date" name="date" type="date"/>
     
    <label for="ex1cmnts">Comments:</label>
    <textarea id="ex1cmnts" placeholder="Your comments" cols="40"
rows="3"></textarea><br/>
    <label for="ex1age1">Age:</label>
    <input id="ex1age1" type="range" min="18" max="99" value="25"
        onchange="document.getElementById('ex1age2').value = this.value"
    />
    <input id="ex1age2" size="3" value="25"
        onchange="if (this.value &lt; 18) this.value = 18;
            if (this.value &gt; 99) this.value = 99;
            document.getElementById('ex1age1').value = this.value"
    />
</form>
```

The "username" field is a `type="text"` input field by default.

WebKit (which is used by Chrome) ignores CSS styles which have a selector that it does not recognize, even if the other CSS selectors are valid. Therefore, the CSS style for the placeholder attributes must be duplicated for Chrome and Firefox because following CSS code does *not* work:

```
<style scoped="scoped">
    input::-webkit-input-placeholder, textarea::-webkit-input-placeholder,
    input:-moz-placeholder, textarea:-moz-placeholder
        { font-family: Verdana; color: #006633; font-style: italic }
</style>
```

back to top

Changes in HTML 5 - <form> Tag

What's new in HTML 5

The novalidate="novalidate" attribute has been added.

Differences between HTML 5 and earlier versions of HTML

The following attributes should *not* be coded on the <form> tag because they either have been deprecated or were never officially supported:

- `target`

The 2000-2010 Recommendations from the W3C HTML Working Group defined the HTML namespace for the **form** element type name along with the names of all HTML element types. In older (pre-2000) versions of HTML, element type names were not associated with a namespace.

back to top

THE END