**HTML Tutorial: Forms & Input Tags | Web Development Tutorials #8**

This HTML tutorial involves the learning of forms and Input tags used in HTML. We are going to start by making a new HTML file as “*tut8.html*” and adding a boilerplate through Emmet abbreviation. Forms are a very important part of HTML. It represents the document section containing interactive controls for submitting information.

Whenever we add a [<form>](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/form) tag in the HTML, it is going to ask for some action for submitting that particular form in the backend for future reference. So, for now, we will write it as *backend.php.*All the data submitted in a form will be stored automatically in the backend “backend.php” after submitting it.

The template will look like-

<form action= “backend.php”>

## **The Action Attribute**

The action attribute defines the action to be performed when the form is submitted.

Usually, the form data is sent to a file on the server when the user clicks on the submit button.

## **The Target Attribute**

The target attribute specifies where to display the response that is received after submitting the form.

The target attribute can have one of the following values:

|  |  |
| --- | --- |
| **Value** | **Description** |
| \_blank | The response is displayed in a new window or tab |
| \_self | The response is displayed in the current window |
| \_parent | The response is displayed in the parent frame |
| \_top | The response is displayed in the full body of the window |
| *framename* | The response is displayed in a named iframe |

The default value is \_self which means that the response will open in the current window.

### **Example**

Here, the submitted result will open in a new browser tab:

<form action="/action\_page.php" target="\_blank">

## **The Method Attribute**

The method attribute specifies the HTTP method to be used when submitting the form data.

The form-data can be sent as URL variables (with method="get") or as HTTP post transaction (with method="post").

The default HTTP method when submitting form data is GET.

### **Example**

This example uses the GET method when submitting the form data:

<form action="/action\_page.php" method="get">

## **The Autocomplete Attribute**

The autocomplete attribute specifies whether a form should have autocomplete on or off.

When autocomplete is on, the browser automatically complete values based on values that the user has entered before.

### **Example**

A form with autocomplete on:

<form action="/action\_page.php" autocomplete="on">

## **The Novalidate Attribute**

The novalidate attribute is a boolean attribute.

When present, it specifies that the form-data (input) should not be validated when submitted.

### **Example**

A form with a novalidate attribute:

<form action="/action\_page.php" novalidate>

## **List of All <form> Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| [accept-charset](https://www.w3schools.com/tags/att_form_accept_charset.asp) | Specifies the character encodings used for form submission |
| [action](https://www.w3schools.com/tags/att_form_action.asp) | Specifies where to send the form-data when a form is submitted |
| [autocomplete](https://www.w3schools.com/tags/att_form_autocomplete.asp) | Specifies whether a form should have autocomplete on or off |
| [enctype](https://www.w3schools.com/tags/att_form_enctype.asp) | Specifies how the form-data should be encoded when submitting it to the server (only for method="post") |
| [method](https://www.w3schools.com/tags/att_form_method.asp) | Specifies the HTTP method to use when sending form-data |
| [name](https://www.w3schools.com/tags/att_form_name.asp) | Specifies the name of the form |
| [novalidate](https://www.w3schools.com/tags/att_form_novalidate.asp) | Specifies that the form should not be validated when submitted |
| [rel](https://www.w3schools.com/tags/att_form_rel.asp) | Specifies the relationship between a linked resource and the current document |
| [target](https://www.w3schools.com/tags/att_form_target.asp) | Specifies where to display the response that is received after submitting the form |

## **The <input> Element**

The HTML <input> element is the most used form element.

An <input> element can be displayed in many ways, depending on the type attribute.

Here are some examples:

|  |  |
| --- | --- |
| **Type** | **Description** |
| <input type="text"> | Displays a single-line text input field |
| <input type="radio"> | Displays a radio button (for selecting one of many choices) |
| <input type="checkbox"> | Displays a checkbox (for selecting zero or more of many choices) |
| <input type="submit"> | Displays a submit button (for submitting the form) |
| <input type="button"> | Displays a clickable button |

Then comes the [<input>](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/input)tags which are present inside the form, where the user provides the input. These inputs can be of any type whether text, button, checkbox, date, time, etc. Input Tags are used to create interactive controls for web-based forms in order to accept data from the user.

The **<span>** is an in-line element and **<div>** is a block element. Which means, if we use two separate div tags for different inputs, then all the inputs will come on different lines. We will learn about the span and div in detail in the upcoming tutorial. Till then we will use the <br> tags for breaking a line.

* To get the input as*text*, the syntax is-

<input type= “text”>

Copy

* To get the input type as an *email* in the form, the syntax is-

<div>

Email: <input type="email" name="myEmail">

</div>

Copy

The *name*here is used so that the backend can recognize the tag that we are using.

* To get the *submit*button in the form, the syntax is-

<div>

<input type= “submit” value= “submit now”>

</div>

Copy

* We can also add date and time in the form. To add these, the general syntax is-

<div>

<input type= “date” name= ”myDate” id= “”>

</div>

Copy

It will give the complete date form in the format of “dd/mm/yyyy”.\

* To add any numeric text in the HTML form, the syntax is-

<div>

Number: <input type= “number” name “myNumber”>

</div>

Copy

While filling several online forms, you must have seen the [radio](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/input/radio) buttons and [checkboxes](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/input/checkbox) in the form. Radio buttons are such buttons that allows to select any one of the following options amongst all. For example, while selecting the gender, we can only select either male or female. Whereas the checkbox allows selecting the multiple options available. The example of both the formats are as follows-

* For checkbox-

<div>

Are you eligible?: <input type="checkbox" name="myEligibility" checked>

</div>

Copy

* For Radio buttons-

<div>

Gender: Male <input type="radio" name="myGender"> Female <input type="radio" name="myGender">

Other <input type="radio" name="myGender">

</div>

Copy

* To reset all the information, entered in the form, we take the help of a ***reset*** button. To get the reset button, we have to write-

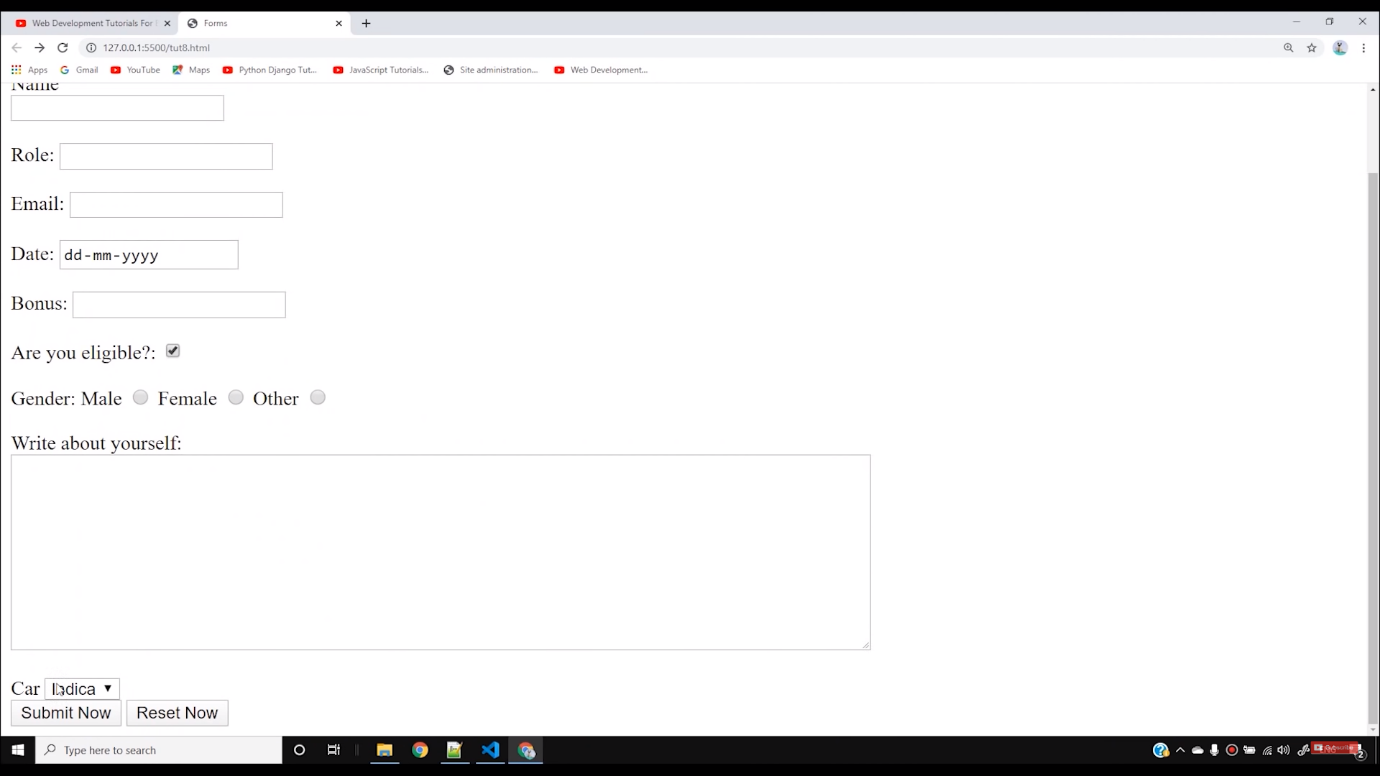
<div>

Input type= “reset” value= “Reset Now”

</div>

Copy

After inserting all these input tags, our form will look like-



These are some of the examples of basic input tags used inside the form in the HTML. Apart from these, there are numerous more input tags available, but you need not to learn all these at once. You can always take help of references available. And one more advice, never try to grasp all things at once. Practice makes a man perfect.

**Code as described/written in the video**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<title>Forms</title>

</head>

<body>

<h2>This is HTML forms tutorial</h2>

<form action="backend.php">

<label for="name"> Name</label>

<div>

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

</div>

<br>

<div>

Role: <input type="text" name="myRole">

</div>

<br>

<div>

Email: <input type="email" name="myEmail">

</div>

<br>

<div>

Date: <input type="date" name="myDate">

</div>

<br>

<div>

Bonus: <input type="number" name="myBonus">

</div>

<br>

<div>

Are you eligible?: <input type="checkbox" name="myEligibility" checked>

</div>

<br>

<div>

Gender: Male <input type="radio" name="myGender"> Female <input type="radio" name="myGender">

Other <input type="radio" name="myGender">

</div>

<br>

<div>

Write about yourself: <br><textarea name="myText" cols="90" rows="10"></textarea>

</div>

<br>

<div>

<label for="car">Car</label>

<select name="myCar" id="car">

<option value="ind">Indica</option>

<option value="swf" selected>Swift</option>

</select>

</div>

<br>

<div>

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

<input type="reset" value="Reset Now">

</div>

</form>

</body>

</html>

## **Text Fields**

The <input type="text"> defines a single-line input field for text input.

### **Example**

A form with input fields for text:

<form>  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname"><br>  
  <label for="lname">Last name:</label><br>  
  <input type="text" id="lname" name="lname">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_form_text)

This is how the HTML code above will be displayed in a browser:

Top of Form

First name:  
  
Last name:  


Bottom of Form

**Note:** The form itself is not visible. Also note that the default width of an input field is 20 characters.

## **The <label> Element**

Notice the use of the <label> element in the example above.

The <label> tag defines a label for many form elements.

The <label> element is useful for screen-reader users, because the screen-reader will read out loud the label when the user focus on the input element.

The <label> element also help users who have difficulty clicking on very small regions (such as radio buttons or checkboxes) - because when the user clicks the text within the <label> element, it toggles the radio button/checkbox.

The for attribute of the <label> tag should be equal to the id attribute of the <input> element to bind them together.

## **Radio Buttons**

The <input type="radio"> defines a radio button.

Radio buttons let a user select ONE of a limited number of choices.

### **Example**

A form with radio buttons:

<p>Choose your favorite Web language:</p>  
  
<form>  
  <input type="radio" id="html" name="fav\_language" value="HTML">  
  <label for="html">HTML</label><br>  
  <input type="radio" id="css" name="fav\_language" value="CSS">  
  <label for="css">CSS</label><br>  
  <input type="radio" id="javascript" name="fav\_language" value="JavaScript">  
  <label for="javascript">JavaScript</label>  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_form_radio)

This is how the HTML code above will be displayed in a browser:

Choose your favorite Web language:

 HTML  
 CSS  
 JavaScript

## **Checkboxes**

The <input type="checkbox"> defines a **checkbox**.

Checkboxes let a user select ZERO or MORE options of a limited number of choices.

### **Example**

A form with checkboxes:

<form>  
  <input type="checkbox" id="vehicle1" name="vehicle1" value="Bike">  
  <label for="vehicle1"> I have a bike</label><br>  
  <input type="checkbox" id="vehicle2" name="vehicle2" value="Car">  
  <label for="vehicle2"> I have a car</label><br>  
  <input type="checkbox" id="vehicle3" name="vehicle3" value="Boat">  
  <label for="vehicle3"> I have a boat</label>  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_input_checkbox)

This is how the HTML code above will be displayed in a browser:

 I have a bike  
 I have a car  
 I have a boat

## **The Submit Button**

The <input type="submit"> defines a button for submitting the form data to a form-handler.

The form-handler is typically a file on the server with a script for processing input data.

The form-handler is specified in the form's action attribute.

### **Example**

A form with a submit button:

<form action="/action\_page.php">  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname" value="John"><br>  
  <label for="lname">Last name:</label><br>  
  <input type="text" id="lname" name="lname" value="Doe"><br><br>  
  <input type="submit" value="Submit">  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_form_submit)

This is how the HTML code above will be displayed in a browser:

Top of Form

First name:  
  
Last name:  
  
  


Bottom of Form

## **The Name Attribute for <input>**

Notice that each input field must have a name attribute to be submitted.

If the name attribute is omitted, the value of the input field will not be sent at all.

### **Example**

This example will not submit the value of the "First name" input field:

<form action="/action\_page.php">  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" value="John"><br><br>  
  <input type="submit" value="Submit">

## **The HTML <form> Elements**

The HTML <form> element can contain one or more of the following form elements:

* <input>
* <label>
* <select>
* <textarea>
* <button>
* <fieldset>
* <legend>
* <datalist>
* <output>
* <option>
* <optgroup>

## **The <select> Element**

The <select> element defines a drop-down list:

### **Example**

<label for="cars">Choose a car:</label>  
<select id="cars" name="cars">  
  <option value="volvo">Volvo</option>  
  <option value="saab">Saab</option>  
  <option value="fiat">Fiat</option>  
  <option value="audi">Audi</option>  
</select>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_elem_select)

The <option> elements defines an option that can be selected.

By default, the first item in the drop-down list is selected.

To define a pre-selected option, add the selected attribute to the option:

### **Example**

<option value="fiat" selected>Fiat</option>

### **Visible Values:**

Use the size attribute to specify the number of visible values:

### **Example**

<label for="cars">Choose a car:</label>  
<select id="cars" name="cars" size="3">  
  <option value="volvo">Volvo</option>  
  <option value="saab">Saab</option>  
  <option value="fiat">Fiat</option>  
  <option value="audi">Audi</option>  
</select>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_elem_select_size)

### **Allow Multiple Selections:**

Use the multiple attribute to allow the user to select more than one value:

### **Example**

<label for="cars">Choose a car:</label>  
<select id="cars" name="cars" size="4"multiple>  
  <option value="volvo">Volvo</option>  
  <option value="saab">Saab</option>  
  <option value="fiat">Fiat</option>  
  <option value="audi">Audi</option>  
</select>

## **The <button> Element**

The <button> element defines a clickable button:

### **Example**

<button type="button" onclick="alert('Hello World!')">Click Me!</button>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_elem_button)

This is how the HTML code above will be displayed in a browser:

Click Me!

## **The <fieldset> and <legend> Elements**

The <fieldset> element is used to group related data in a form.

The <legend> element defines a caption for the <fieldset> element.

### **Example**

<form action="/action\_page.php">  
  <fieldset>  
    <legend>Personalia:</legend>  
    <label for="fname">First name:</label><br>  
    <input type="text" id="fname" name="fname" value="John"><br>  
    <label for="lname">Last name:</label><br>  
    <input type="text" id="lname" name="lname" value="Doe"><br><br>  
    <input type="submit" value="Submit">  
  </fieldset>  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_form_legend)

This is how the HTML code above will be displayed in a browser:

Top of Form

Personalia:First name:  
  
Last name:  
  
  


Bottom of Form

## **The <datalist> Element**

The <datalist> element specifies a list of pre-defined options for an <input> element.

Users will see a drop-down list of the pre-defined options as they input data.

The list attribute of the <input> element, must refer to the id attribute of the <datalist> element.

### **Example**

<form action="/action\_page.php">  
  <input list="browsers">  
  <datalist id="browsers">  
    <option value="Internet Explorer">  
    <option value="Firefox">  
    <option value="Chrome">  
    <option value="Opera">  
    <option value="Safari">  
  </datalist>  
</form>

[Try it Yourself »](https://www.w3schools.com/html/tryit.asp?filename=tryhtml_elem_datalist)

## **The <output> Element**

The <output> element represents the result of a calculation (like one performed by a script).

### **Example**

Perform a calculation and show the result in an <output> element:

<form action="/action\_page.php"  
  oninput="x.value=parseInt(a.value)+parseInt(b.value)">  
  0  
  <input type="range"  id="a" name="a" value="50">  
  100 +  
  <input type="number" id="b" name="b" value="50">  
  =  
  <output name="x" for="a b"></output>  
  <br><br>  
  <input type="submit">  
</form>

## **HTML Input Types**

Here are the different input types you can use in HTML:

* <input type="button">
* <input type="checkbox">
* <input type="color">
* <input type="date">
* <input type="datetime-local">
* <input type="email">
* <input type="file">
* <input type="hidden">
* <input type="image">
* <input type="month">
* <input type="number">
* <input type="password">
* <input type="radio">
* <input type="range">
* <input type="reset">
* <input type="search">
* <input type="submit">
* <input type="tel">
* <input type="text">
* <input type="time">
* <input type="url">
* <input type="week">

## **Input Type Hidden**

The <input type="hidden"> defines a hidden input field (not visible to a user).

A hidden field lets web developers include data that cannot be seen or modified by users when a form is submitted.

A hidden field often stores what database record that needs to be updated when the form is submitted.

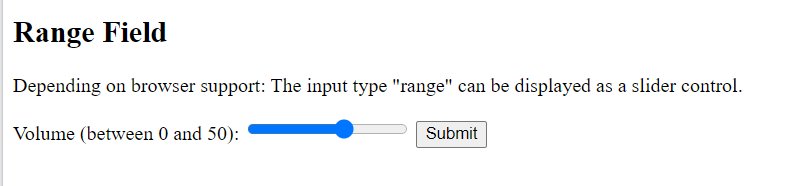
**Note:** While the value is not displayed to the user in the page's content, it is visible (and can be edited) using any browser's developer tools or "View Source" functionality. Do not use hidden inputs as a form of security!

## **Input Type Range**

The <input type="range"> defines a control for entering a number whose exact value is not important (like a slider control). Default range is 0 to 100. However, you can set restrictions on what numbers are accepted with the min, max, and step attributes:

### **Example**

<form>  
  <label for="vol">Volume (between 0 and 50):</label>  
  <input type="range" id="vol" name="vol" min="0" max="50">  
</form>



## **Input Type Search**

The <input type="search"> is used for search fields (a search field behaves like a regular text field).

### **Example**

<form>  
  <label for="gsearch">Search Google:</label>  
  <input type="search" id="gsearch" name="gsearch">  
</form>

## **The value Attribute**

The input value attribute specifies an initial value for an input field:

### **Example**

Input fields with initial (default) values:

<form>  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname" value="John"><br>  
  <label for="lname">Last name:</label><br>  
  <input type="text" id="lname" name="lname" value="Doe">

## **The size Attribute**

The input size attribute specifies the visible width, in characters, of an input field.

The default value for size is 20.

**Note:** The size attribute works with the following input types: text, search, tel, url, email, and password.

### **Example**

Set a width for an input field:

<form>  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname" size="50"><br>  
  <label for="pin">PIN:</label><br>  
  <input type="text" id="pin" name="pin" size="4">  
</form>

## **The min and max Attributes**

The input min and max attributes specify the minimum and maximum values for an input field.

The min and max attributes work with the following input types: number, range, date, datetime-local, month, time and week.

**Tip:** Use the max and min attributes together to create a range of legal values.

### **Example**

Set a max date, a min date, and a range of legal values:

<form>  
  <label for="datemax">Enter a date before 1980-01-01:</label>  
  <input type="date" id="datemax" name="datemax" max="1979-12-31"><br><br>  
  
  <label for="datemin">Enter a date after 2000-01-01:</label>  
  <input type="date" id="datemin" name="datemin" min="2000-01-02"><br><br>  
  
  <label for="quantity">Quantity (between 1 and 5):</label>  
  <input type="number" id="quantity" name="quantity" min="1" max="5">  
</form>

## **The required Attribute**

The input required attribute specifies that an input field must be filled out before submitting the form.

The required attribute works with the following input types: text, search, url, tel, email, password, date pickers, number, checkbox, radio, and file.

### **Example**

A required input field:

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