

HTML - Forms

HTML Forms are required, when you want to collect some data from the site visitor. For example, during user registration you would like to collect information such as name, email address, credit card, etc.

A form will take input from the site visitor and then will post it to a back-end application such as CGI, ASP Script or PHP script etc. The back-end application will perform required processing on the passed data based on defined business logic inside the application.

There are various form elements available like text fields, textarea fields, drop-down menus, radio buttons, checkboxes, etc.

The HTML **<form>** tag is used to create an HTML form and it has following syntax –

```
<form action = "Script URL" method = "GET|POST">  
    form elements like input, textarea etc.  
</form>
```

Form Attributes

Apart from common attributes, following is a list of the most frequently used form attributes –

Sr.No	Attribute & Description
1	action Backend script ready to process your passed data.
2	method Method to be used to upload data. The most frequently used are GET and POST methods.
3	target Specify the target window or frame where the result of the script will be displayed. It takes values like _blank, _self, _parent etc.
4	enctype You can use the enctype attribute to specify how the browser encodes the data before it sends it to the server. Possible values are – application/x-www-form-urlencoded – This is the standard method most forms use in simple scenarios. multipart/form-data – This is used when you want to upload binary data in the form of files like image, word file etc.

Note – You can refer to Perl & CGI for a detail on how form data upload works.

AD



HTML Form Controls

There are different types of form controls that you can use to collect data using HTML form –

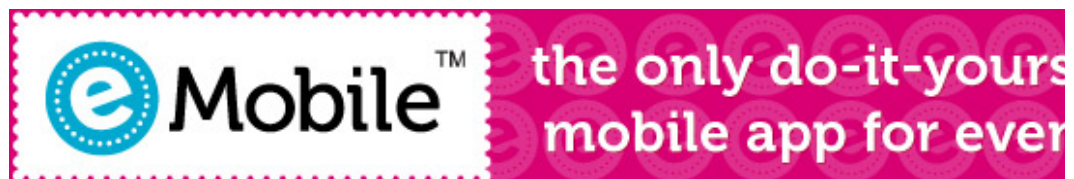
- Text Input Controls
- Checkboxes Controls
- Radio Box Controls
- Select Box Controls
- File Select boxes
- Hidden Controls
- Clickable Buttons
- Submit and Reset Button

Text Input Controls

There are three types of text input used on forms –

- **Single-line text input controls** – This control is used for items that require only one line of user input, such as search boxes or names. They are created using HTML `<input>` tag.
- **Password input controls** – This is also a single-line text input but it masks the character as soon as a user enters it. They are also created using HTML `<input>` tag.
- **Multi-line text input controls** – This is used when the user is required to give details that may be longer than a single sentence. Multi-line input controls are created using HTML `<textarea>` tag.

AD



Single-line text input controls

This control is used for items that require only one line of user input, such as search boxes or names. They are created using HTML `<input>` tag.

Example

Here is a basic example of a single-line text input used to take first name and last name –

[Live Demo](#)

```
<!DOCTYPE html>
<html>

  <head>
    <title>Text Input Control</title>
  </head>

  <body>
    <form >
      First name: <input type = "text" name = "first_name" />
      <br>
      Last name: <input type = "text" name = "last_name" />
    </form>
  </body>

</html>
```

This will produce the following result –

First name:
Last name:

Attributes

Following is the list of attributes for <input> tag for creating text field.

Sr.No	Attribute & Description
1	type Indicates the type of input control and for text input control it will be set to text .
2	name Used to give a name to the control which is sent to the server to be recognized and get the value.
3	value This can be used to provide an initial value inside the control.
4	size Allows to specify the width of the text-input control in terms of characters.
5	maxlength Allows to specify the maximum number of characters a user can enter into the text box.

Password input controls

This is also a single-line text input but it masks the character as soon as a user enters it. They are also created using HTML <input>tag but type attribute is set to **password**.

Example

Here is a basic example of a single-line password input used to take user password –

```
<!DOCTYPE html>
<html>

  <head>
    <title>Password Input Control</title>
  </head>

  <body>
    <form >
      User ID : <input type = "text" name = "user_id" />
      <br>
      Password: <input type = "password" name = "password" />
    </form>
  </body>

</html>
```

This will produce the following result –

User ID :

Password:

Attributes

Following is the list of attributes for <input> tag for creating password field.

Sr.No	Attribute & Description
1	type Indicates the type of input control and for password input control it will be set to password .
2	name Used to give a name to the control which is sent to the server to be recognized and get the value.
3	value This can be used to provide an initial value inside the control.
4	size Allows to specify the width of the text-input control in terms of characters.
5	maxlength Allows to specify the maximum number of characters a user can enter into the text box.

Multiple-Line Text Input Controls

This is used when the user is required to give details that may be longer than a single sentence. Multi-line input controls are created using HTML `<textarea>` tag.

Example

Here is a basic example of a multi-line text input used to take item description –

[Live Demo](#)

```
<!DOCTYPE html>
<html>

  <head>
    <title>Multiple-Line Input Control</title>
  </head>

  <body>
    <form>
      Description : <br />
      <textarea rows = "5" cols = "50" name = "description">
        Enter description here...
      </textarea>
    </form>
  </body>
</html>
```

```
</form>
</body>

</html>
```

This will produce the following result –

Description:

Enter description here...

Attributes

Following is the list of attributes for <textarea> tag.

Sr.No	Attribute & Description
1	name Used to give a name to the control which is sent to the server to be recognized and get the value.
2	rows Indicates the number of rows of text area box.
3	cols Indicates the number of columns of text area box

Checkbox Control

Checkboxes are used when more than one option is required to be selected. They are also created using HTML <input> tag but type attribute is set to **checkbox**..

Example

Here is an example HTML code for a form with two checkboxes –

[Live Demo](#)

```
<!DOCTYPE html>
<html>

  <head>
    <title>Checkbox Control</title>
  </head>

  <body>
    <form>
      <input type = "checkbox" name = "maths" value = "on"> Maths
      <input type = "checkbox" name = "physics" value = "on"> Physics
    </form>
  </body>

</html>
```

This will produce the following result –

☐ Maths ☐ Physics

Attributes

Following is the list of attributes for <checkbox> tag.

Sr.No	Attribute & Description
1	type Indicates the type of input control and for checkbox input control it will be set to checkbox ..
2	name Used to give a name to the control which is sent to the server to be recognized and get the value.
3	value The value that will be used if the checkbox is selected.
4	checked Set to <i>checked</i> if you want to select it by default.

Radio Button Control

Radio buttons are used when out of many options, just one option is required to be selected. They are also created using HTML `<input>` tag but type attribute is set to **radio**.

Example

Here is example HTML code for a form with two radio buttons –

[Live Demo](#)

```
<!DOCTYPE html>
<html>

  <head>
    <title>Radio Box Control</title>
  </head>

  <body>
    <form>
      <input type = "radio" name = "subject" value = "maths"> Maths
      <input type = "radio" name = "subject" value = "physics"> Physics
    </form>
  </body>

</html>
```

This will produce the following result –

☐ Maths ☐ Physics

Attributes

Following is the list of attributes for radio button.

Sr.No	Attribute & Description
1	type Indicates the type of input control and for checkbox input control it will be set to radio.
2	name Used to give a name to the control which is sent to the server to be recognized and get the value.
3	value The value that will be used if the radio box is selected.
4	checked Set to <i>checked</i> if you want to select it by default.

Select Box Control

A select box, also called drop down box which provides option to list down various options in the form of drop down list, from where a user can select one or more options.

Example

Here is example HTML code for a form with one drop down box

[Live Demo](#)

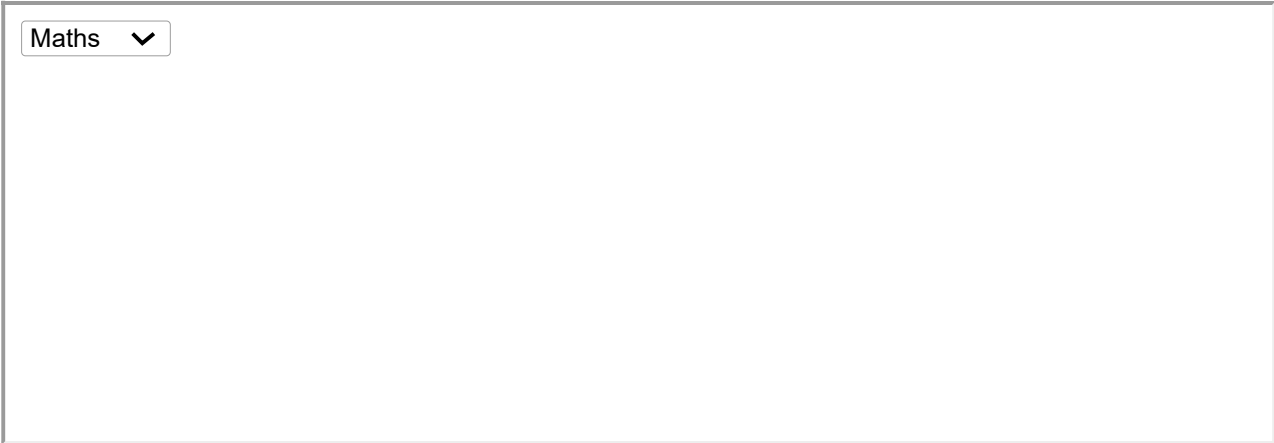
```
<!DOCTYPE html>  
<html>
```

```
<head>
  <title>Select Box Control</title>
</head>

<body>
  <form>
    <select name = "dropdown">
      <option value = "Maths" selected>Maths</option>
      <option value = "Physics">Physics</option>
    </select>
  </form>
</body>

</html>
```

This will produce the following result –



Attributes

Following is the list of important attributes of <select> tag –

Sr.No	Attribute & Description
1	name Used to give a name to the control which is sent to the server to be recognized and get the value.
2	size This can be used to present a scrolling list box.
3	multiple If set to "multiple" then allows a user to select multiple items from the menu.

Following is the list of important attributes of <option> tag –

Sr.No	Attribute & Description
1	value The value that will be used if an option in the select box box is selected.
2	selected Specifies that this option should be the initially selected value when the page loads.
3	label An alternative way of labeling options

File Upload Box

If you want to allow a user to upload a file to your web site, you will need to use a file upload box, also known as a file select box. This is also created using the <input> element but type attribute is set to **file**.

Example

Here is example HTML code for a form with one file upload box –

[Live Demo](#)

```
<!DOCTYPE html>
<html>

  <head>
    <title>File Upload Box</title>
  </head>

  <body>
    <form>
      <input type = "file" name = "fileupload" accept = "image/*" />
    </form>
  </body>

</html>
```

This will produce the following result –

No file chosen

Attributes

Following is the list of important attributes of file upload box –

Sr.No	Attribute & Description
1	name Used to give a name to the control which is sent to the server to be recognized and get the value.
2	accept Specifies the types of files that the server accepts.

Button Controls

There are various ways in HTML to create clickable buttons. You can also create a clickable button using `<input>` tag by setting its type attribute to **button**. The type attribute can take the following values –

Sr.No	Type & Description
1	submit This creates a button that automatically submits a form.
2	reset This creates a button that automatically resets form controls to their initial values.
3	button This creates a button that is used to trigger a client-side script when the user clicks that button.
4	image This creates a clickable button but we can use an image as background of the button.

Example

Here is example HTML code for a form with three types of buttons –

[Live Demo](#)

```
<!DOCTYPE html>
<html>

  <head>
    <title>File Upload Box</title>
  </head>

  <body>
    <form>
      <input type = "submit" name = "submit" value = "Submit" />
      <input type = "reset" name = "reset" value = "Reset" />
      <input type = "button" name = "ok" value = "OK" />
      <input type = "image" name = "imagebutton" src = "/html/images/logo.png" />
    </form>
  </body>

</html>
```

This will produce the following result –



Hidden Form Controls

Hidden form controls are used to hide data inside the page which later on can be pushed to the server. This control hides inside the code and does not appear on the actual page. For example, following hidden form is being used to keep current page number. When a user will click next page then the value of hidden control will be sent to the web server and there it will decide which page will be displayed next based on the passed current page.

Example

Here is example HTML code to show the usage of hidden control –

[Live Demo](#)

```
<!DOCTYPE html>
<html>

  <head>
    <title>File Upload Box</title>
  </head>

  <body>
    <form>
      <p>This is page 10</p>
      <input type = "hidden" name = "pagename" value = "10" />
      <input type = "submit" name = "submit" value = "Submit" />
      <input type = "reset" name = "reset" value = "Reset" />
    </form>
  </body>

</html>
```

This will produce the following result –

This is page 10

Submit

Reset