

# Index

|  |           |
|--|-----------|
| <b>What Is HTML.....</b>                                     | <b>2</b>  |
| <b>Use Of HTML.....</b>                                      | <b>2</b>  |
| <b>About Different Browser.....</b>                          | <b>2</b>  |
| <b>HTML Version.....</b>                                     | <b>3</b>  |
| <b>Structure of HTML.....</b>                                | <b>4</b>  |
| <b>Forms.....</b>  | <b>5</b>  |
| <b>Action Attribute.....</b>                                 | <b>5</b>  |
| <b>Target attribute.....</b>                                 | <b>6</b>  |
| <b>Method Attribute.....</b>                                 | <b>6</b>  |
| <b>Autocomplete Attribute.....</b>                           | <b>7</b>  |
| <b>Novalidate Attribute.....</b>                             | <b>7</b>  |
| <b>Input Element.....</b>                                    | <b>8</b>  |
| <b>textarea Element.....</b>                                 | <b>9</b>  |
| <b>Selectbox.....</b>  | <b>9</b>  |
| <b>Checkbox.....</b>   | <b>10</b> |
| <b>radio button.....</b>                                     | <b>10</b> |
| <b>Button.....</b>   | <b>10</b> |
| <b>Submit Input.....</b>                                     | <b>11</b> |
| <b>File Control with its attribute.....</b>                  | <b>11</b> |
| Accept attribute.....  | 11        |
| Capture attribute.....                                       | 11        |
| Multiple attribute.....                                      | 11        |
| Non-standard attributes.....                                 | 12        |
| <b>Name.....</b>   | <b>12</b> |
| <b>Id.....</b>   | <b>12</b> |
| <b>Value.....</b>  | <b>13</b> |
| <b>Class.....</b>  | <b>13</b> |
| <b>Img tag.....</b>  | <b>14</b> |
| <b>A Tag.....</b>  | <b>14</b> |
| <b>What is meta tag , use of meta tag.....</b>               | <b>15</b> |
| Use of meta data.....  | 16        |
| <b>What is responsive website , how user can do it?.....</b> | <b>16</b> |

# What Is HTML

HTML (HyperText Markup Language) is the most basic building block of the Web. HTML are generally used to describe a web page's appearance

"Hypertext" refers to links that connect web pages to one another, either within a single website or between websites.

HTML uses "markup" to annotate text, images, and other content for display in a Web browser.

- HTML is the standard markup language for creating Web pages
- HTML describes the structure of a Web page
- HTML consists of a series of elements
- HTML elements tell the browser how to display the content
- HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

## Use Of HTML

- HTML is used to create the structure and content of web pages
- HTML can be used to add images and videos to a web page
- HTML can be used to create lists on a web page. This includes ordered lists, unordered lists, and definition lists.
- HTML can be used to create hyperlinks on a web page.

## About Different Browser

Basically Browser used for display html content, browser not showing us to html tags its know meanings of all html tags so it's display content accordingly to the tags we used.

There are many different web browsers available, but some of the most popular include:

- Google Chrome: Google Chrome is the most popular web browser in the world. It is fast, secure, and has a wide range of features.
- Mozilla Firefox: Mozilla Firefox is a popular open-source web browser. It is known for its privacy features and customization options
- Microsoft Edge: Microsoft Edge is the successor to Internet Explorer. It is based on the Chromium web engine, and it offers a number of new features, such as a built-in ad blocker and a reading mode
- Apple Safari: Apple Safari is the default web browser on Apple devices. It is fast, secure, and integrates well with other Apple products
- Opera: Opera is a popular web browser that is known for its speed and customization options. It also offers a built-in VPN and ad blocker.
- Brav: Brave is a privacy-focused web browser. It blocks ads and trackers by default, and it also offers a number of other privacy features.

## HTML Version

Html 1st version HTML1.0 released at 1991 and currently latest version of html is HTML 5 which released at 2014.

List of all HTML version :

- HTML 1.0

The first version of HTML was released in 1991. It was a very simple language, and it only supported a few basic elements, such as <p> for paragraphs and <a> for links.

- HTML 2.0

It added a few new elements and attributes, and it also made some changes to the syntax. Some of the new elements included <img> for images, <table> for tables, and <form> for forms.

- HTML 3.2

It added a lot of new elements and attributes, and it also made some changes to the structure of HTML documents. Some of the new elements included `<b>` for bold text, `<i>` for italic text, and `<u>` for underlined text.

- HTML 4.01

It was a major revision of HTML, and it added a lot of new features, such as CSS support and tables.

- XHTML 1.0

It is a reformulation of HTML 4.01 using XML syntax. XML is a more strict markup language than HTML, and it is designed to be more machine-readable.

- HTML5:

HTML5 is the current version of HTML. It was released in 2014, and it adds a lot of new features, such as video and audio support, canvas, and web storage. Some of the new features included `<video>` for embedded videos, `<audio>` for embedded audio, and `<canvas>` for drawing on the web page.

## Structure of HTML

```
<html>
  <head>
    <title>Page title</title>
  </head>
  <body>
    <h1>This is a heading</h1>
    <p>This is a paragraph.</p>
    <p>This is another paragraph.</p>
  </body>
</html>
```

This is structure of HTML two main sections `<head>` and `<body>`

**<head>** : in this section include meta tags and other useful information it not display to user on browser.

**<body>** : in this section include all tags that used for display and format content of website.

# Forms

- HTML forms is used for get input from user.
- **<form>** this tag is used for create form.
- The <form> element is a container for different types of input elements, such as: text fields, checkboxes, radio buttons, submit buttons, etc.

- Layout of forms :

<form>.

form elements.

</form>

## 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.

In the example below, the form data is sent to a file called "action\_page.php". This file contains a server-side script that handles the form data.

### Example

On submit, send form data to "action\_page.php":

```
<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>
```

**Note : if we dont give action attribute it send action to current page.**

## Target attribute

This is also attribute for form tag.

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

There are some different target attribute values :

- `_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

Note : 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">
```

## 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">
```

Example

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

```
<form action="/action_page.php" method="post">
```

GET :

- Appends the form data to the URL, in name/value pairs
- NEVER use GET to send sensitive data! (the submitted form data is visible in the URL!)
- The length of a URL is limited (2048 characters)
- Useful for form submissions where a user wants to bookmark the result
- GET is good for non-secure data, like query strings in Google

POST :

- Appends the form data inside the body of the HTTP request (the submitted form data is not shown in the URL)
- POST has no size limitations, and can be used to send large amounts of data.
- Form submissions with POST cannot be bookmarked

## 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">
```

## 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>
```

# Input Element

One of the most used form element is the `<input>` element.

The `<input>` element can be displayed in several ways, depending on the type attribute.

Example

```
<label for="fname">First name:</label>
```

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

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">`

**Note:** The default value of the type attribute is "text".



# textarea Element

The <textarea> element defines a multi-line input field (a text area):

Example

```
<textarea name="message" rows="10" cols="30">  
The cat was playing in the garden.  
</textarea>
```

Note : The rows attribute specifies the visible number of lines in a text area.  
The cols attribute specifies the visible width of a text area.

# Selectbox

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>
```

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

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

**selected attribute:** use for a pre-selected option.:

**size attribute:** use for how many options we want to display at a time.

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

# Checkbox

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

Example

```
<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>
```

# radio button

Radio buttons let a user select ONLY ONE of a limited number of choices:

Example:

```
<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>
```

# Button

Used for create button

Example

```
<input type="button" value="submit">
```

**Note :** The `<input type="button">` and `<button>` tag are two different approach to use Button in HTML document.

## Submit Input

The `<input type="submit">` defines a submit button which submits all form values to a form-handler.

Define a submit button:

```
<input type="submit">
```

## File Control with its attribute

It let the user choose one or more files from their device storage. Once chosen, the files can be uploaded to a server

Example :

```
<input type="file" id="avatar" name="avatar" accept="image/png, image/jpeg" />
```

### Accept attribute

The accept attribute value is a string that defines the file types the file input should accept. This string is a comma-separated list of unique file type specifiers.

### Capture attribute

The capture attribute value is a string that specifies which camera to use for capture of image or video data, if the accept attribute indicates that the input should be of one of those types. A value of user indicates that the user-facing camera and/or microphone should be used.

### Multiple attribute

When the multiple Boolean attribute is specified, the file input allows the user to select more than one file.

## Non-standard attributes

**webkitdirectory** : HTML attribute and indicates that the <input> element should let the user select directories instead of files. When a directory is selected, the directory and its entire hierarchy of contents are included in the set of selected items. The selected file system entries can be obtained using the webkitEntries property.

## Name

The name attribute specifies a name for an HTML element. It is often used with form elements, such as input and select elements, to identify the element when the form is submitted. The name attribute can also be used to reference elements in JavaScript.

**Note : In HTML5, the name attribute is deprecated for the a element. Instead, the id attribute should be used.**

The syntax for the name attribute is as follows:

```
<element name="value">
```

For example, the following code would create an input element with the name my\_input:

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

When the form is submitted, the value of the my\_input element will be submitted along with the other form data.

**The name attribute must be unique within a form.**

## Id

The id attribute specifies a unique id for an HTML element. The value of the id attribute must be unique within the HTML document.

Id element used to give specific style in any element or for javascript access that element and manipulate it.

**Note :**

- The id name is case sensitive!
- The id name must contain at least one character, cannot start with a number, and must not contain whitespaces (spaces, tabs, etc.).

### Example

```
<!DOCTYPE html>
<html>
<head>
<style>
#id1 {
  background-color: blue;}
</style>
</head>
<body>

<h1 id="id1">Hello RKIT</h1>

</body>
</html>
```

## Value

- The value attribute used for specifies the value of an <input> element.
- 
- The value attribute is used differently for different input types
- 
- Note: The value attribute cannot be used with <input type="file">.

## Class

This is used for give one or more classname to an element.

When we have to give some common style sheets into multiple elements we give that all element to a one class name.

Also it is used in javascript for a access element and modify it.

### Example

Use of the class attribute in an HTML document:

```
<html>
<head>
<style>
.intro {
    color: blue;
}

</style>
</head>
<body>
<h1 class="intro">Hello RKIT</h1>
</body>
</html>
```

## Img tag

It is used for display image in website.

The <img> tag has two required attributes:

- src - Specifies the path to the image
- alt - Specifies an alternate text for the image, if the image for some reason cannot be displayed

Note: Also, always specify the width and height of an image. If width and height are not specified, the page might flicker while the image loads.

## A Tag

This tag is used for give hyperlink to other pages.

The most important attribute of the <a> element is the href attribute, which indicates the link's destination.

By default, links will appear as follows in all browsers:

- An unvisited link is underlined and blue
- A visited link is underlined and purple
- An active link is underlined and red

# What is meta tag , use of meta tag

Meta tag is used for write metadata.

Meta data means data about data.

<meta> tags always go inside the <head> element, and are typically used to specify character set, page description, keywords, author of the document, and viewport settings.

Some attribute of meta tag

Charset : Specifies the character encoding for the HTML document.

Content : Specifies the value associated with the http-equiv or name attribute.

Http-equiv : Provides an HTTP header for the information/value of the content attribute.

Name : Specifies a name for the metadata.

More Examples

Define keywords for search engines:

```
<meta name="keywords" content="HTML, CSS, JavaScript">
```

Define a description of your web page:

```
<meta name="description" content="Free Web tutorials for HTML and CSS">
```

Define the author of a page:

```
<meta name="author" content="John Doe">
```

Refresh document every 30 seconds:

```
<meta http-equiv="refresh" content="30">
```

Setting the viewport to make your website look good on all devices:

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

## Use of meta data

Meta tag helps browser's crawler to understand what type of content are present in website and according to that browser understand what type of user are visit this website.

Meta tag is important for SEO.

## What is responsive website , how user can do it?

When www is just started in 19's that time website are make according to look good in Computer. After time smartphone take many user base so now website look good in phone first. So now we have to make website responsive for all screen size that call responsive website.

There are many benefits to using responsive web design, including:

- Improved user experience: RWD websites provide a better user experience for users on all devices. This is because the website will automatically adjust its layout and content to fit the size of the screen, so users don't have to zoom or scroll to see the content.
- Increased mobile traffic: More and more people are using their mobile devices to access the internet. By making your website responsive, you can ensure that you are reaching a wider audience.
- Improved SEO: Google and other search engines now favor responsive websites in their search results. This means that your website is more likely to be seen by potential users if it is responsive.

Here are some steps on how to create a responsive website:

- Start with a mobile-first approach. This means designing your website for mobile devices first, and then scaling it up for larger screens.
- Use flexible grids and layouts. This will allow your website to adapt to different screen sizes without breaking.
- Use media queries. Media queries are used to control how your website is displayed on different devices.
- Optimize your images. Make sure your images are optimized for the web so they load quickly on all devices.



- Test your website on different devices. Make sure your website looks good and works properly on all devices before you launch it.