**GUI Basic Training**

**Prince Goswami**

**Table of Contents**

[1 HTML 1](#_Toc142284959)

[1.1 Basic of HTML 1](#_Toc142284960)

[1.1.1 what is HTML , use of html , about different web browser 1](#_Toc142284961)

[1.1.2 HTML Version : 1](#_Toc142284962)

[1.1.3 Structure of HTML 2](#_Toc142284963)

[1.2 Basic Controls 3](#_Toc142284964)

[1.2.1 form (method type , action...) : 3](#_Toc142284965)

[1.2.2 Input 4](#_Toc142284966)

[1.2.3 Text Area 4](#_Toc142284967)

[1.2.4 Select Box 4](#_Toc142284968)

[1.2.5 Checkbox 5](#_Toc142284969)

[1.2.6 Radio Button 5](#_Toc142284970)

[1.2.7 Button 6](#_Toc142284971)

[1.2.8 Submit Input 7](#_Toc142284972)

[1.2.9 File Control with its attribute 7](#_Toc142284973)

[1.3 Control Attribute 8](#_Toc142284974)

[1.4 Control's Attributes 9](#_Toc142284975)

[1.4.1 img tag, a tag 9](#_Toc142284976)

[1.4.2 What is meta tag , use of meta tag 9](#_Toc142284977)

[1.4.3 What is responsive website , how user can do it? 10](#_Toc142284978)

# HTML

## Basic of HTML

### what is HTML , use of html , about different web browser

[HTML](https://www.codecademy.com/catalog/language/html-css?utm_source=ccblog&utm_medium=ccblog&utm_campaign=ccblog&utm_content=what_is_html_blog) is an acronym that stands for Hyper Text Markup Language. Markup languages are different from programming languages. Whereas programming languages help us modify data, we use markup languages to determine how elements are displayed on a webpage

.

* Structuring web pages.

With tags and elements, we can define the headings, paragraphs, and other contents of a web page.

* Navigating the Internet

Using hyperlink we can easily navigate through internet. Imagine entering URL for every webpage you wanted to visit.

* Improving client side data storage and offline capabilities.

Website can use cookies to store data about users that can be retrieved later when they revisit the website.

Web Browsers are software installed on your PC. To access the Web, you need a web browser, such as Netscape Navigator, Microsoft Internet Explorer or Mozilla Firefox.

* Internet Explorer – Product from Microsoft, The most commonly used web browser in world
* Google Chrome – Developed by Google, The most popular browser.
* Safari – Developed by Apple.inc and included Mac OS X. Very good support for new technologies like XHTML,CSS2 etc.

### HTML Version :

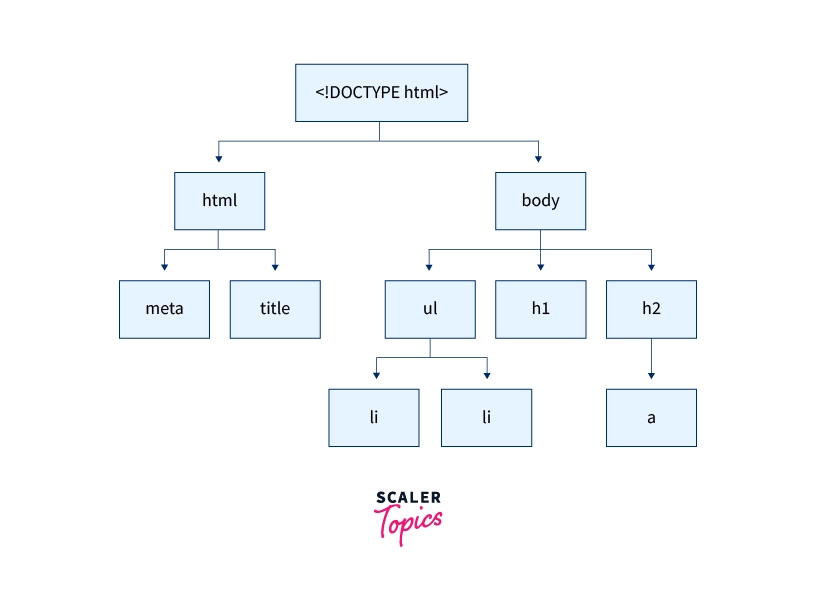
Over the years, there have been many developments in HTML.

Different versions of HTML :

* HTML 1.0 – 1993
  + Structuring Elements
  + Simpler
  + Font support
* HTML 2.0 – 1995
  + Standardization of HTML
  + Forms
  + Tables
  + Formation of W3C(World Wide Web Consortium)
* HTML 3.2 – 1997
  + Upgraded form elements
  + CSS support
  + Image features
* HTML 4.01 – 1999
  + CSS linking
  + New Tags (header, footer, legend)
  + Table enhancement (Colspan, Rowspan)
* XHTML 1.0 – 2000
  + Strict standards and compatibility
  + Compatibility with XML
  + Future adaptability
* HTML5 – 2014
  + New form elements (email, password)
  + Audio Tag
  + Section Tag

### Structure of HTML

An HTML document's basic structure consists of 5 elements:

* <!DOCTYPE>
  + The Tag in HTML is used to inform the browser about the HTML version used in webpage. It is referred as **document type declaration (DTD)**.
* <html>
  + The <html> tag specifies root of HTML or XHTML pages. It is second outer container for everything in HTML pages.  The <html> tag requires a beginning and ending tag.
* <head>
  + The head of an HTML document is a section of the document whose content is not displayed in the browser when the page loads. It only contains HTML document metadata, which specifies information about the HTML document.
  + The document title, character set, styles, links, scripts, and other meta information are defined by metadata.
* <title>
  + This <title> tag in HTML displays the title of a web page and can help in higher rankings in search results if appropriate keywords are included.
  + It can be found in all HTML/XHTML documents. The <title> element must be positioned between the <head> element, and there can only be one title element per document.
* <body>
  + The <body> tag in HTML specifies the main content of an HTML document that appears on the browser. It can contain headings, text, paragraphs, photos, tables, links, videos, etc.
  + The <body> tag must come after the <head> tag, or it must be inserted between the </head> and </html> tags. This tag is essential for and should only be used once throughout the document
  + 

## Basic Controls

### form (method type , action...) :

An HTML Form is a section of the document that collects input from the user. The input from the user is generally sent to a server (Web servers, Mail clients, etc). We use the HTML <form> element to create forms in HTML.

HTML form method

The method attribute of the <form> element specifies how the form data is to be sent to the server. The two most common values for the method attribute are get and post.

* get - The form data is sent as URL variables. This means that the form data will be appended to the URL of the page that the form is submitted to.
* post - The form data is sent as an HTTP POST transaction. This means that the form data will be sent to the server in the body of the HTTP request.

HTML form type

The type attribute of the <input> element specifies the type of input field. The most common values for the type attribute are text, password, checkbox, and radio.

* text - A text input field.
* password - A password input field.
* checkbox - A checkbox input field.
* radio - A radio button input field.

HTML form action

The action attribute of the <form> element specifies the URL of the page that the form data is to be sent to. The action attribute is required for all forms.

For example, the following code creates a form with a method of post and an action of /submit.php:

<form method="post" action="/submit.php">

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

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

</form>

When the user submits this form, the form data will be sent to the /submit.php page as an HTTP POST transaction.

### Input

The <input> tag is an HTML element that is used to create interactive controls for web-based forms. The <input> tag can be used to create a variety of input fields, such as text input fields, password input fields, checkbox input fields, radio button input fields, and submit buttons.

The <input> tag has a number of attributes that can be used to customize the appearance and behaviour of the input field. Some of the most common attributes include:

* type - The type of input field.
* name - The name of the input field.
* value - The initial value of the input field.
* placeholder - A hint that describes the expected value of the input field.
* required - Specifies that the input field is required.
* maxlength - The maximum length of the input field.
* size - The width of the input field.

### Text Area

The <textarea> tag is used to create a multi-line text input field. It is often used in forms to collect user feedback or comments. The <textarea> tag has the following attributes:

* name - The name of the text area. This is used to identify the text area in the form data.
* cols - The number of columns in the text area.
* rows - The number of rows in the text area.
* wrap - The way in which the text is wrapped in the text area. The possible values are soft and hard.
* readonly - This attribute makes the text area read-only.
* disabled - This attribute disables the text area.

For example, the following code creates a text area with 20 columns and 10 rows:

<textarea name="comments" cols="20" rows="10"></textarea>

### Select Box

A select box is a form element that allows the user to select one or more options from a list. The select box is created using the <select> element.

The <select> element has the following attributes:

* name - The name of the select box. This attribute is used to identify the select box when the form is submitted.
* id - The ID of the select box. This attribute is used to reference the select box in JavaScript.
* multiple - A Boolean attribute that specifies whether the user can select multiple options from the select box.
* size - The number of options that are displayed in the select box at once.
* disabled - A Boolean attribute that specifies whether the select box is disabled.

The <select> element contains <option> elements that define the options in the select box.

The <option> element has the following attributes:

* value - The value of the option. This value is sent to the server when the form is submitted.
* selected - A Boolean attribute that specifies whether the option is selected by default.
* disabled - A Boolean attribute that specifies whether the option is disabled.

When the user selects an option from the select box, the value of the option is sent to the server.

For example, the following code creates a select box with two options:

<select name="country">

<option value="USA">United States</option>

<option value="Canada">Canada</option>

</select>

### Checkbox

A checkbox is a form element that allows the user to select one or more options from a limited number of choices. The checkbox is created using the <input> element with the type attribute set to checkbox.

The <input> element has the following attributes:

* name - The name of the checkbox. This attribute is used to identify the checkbox when the form is submitted.
* id - The ID of the checkbox. This attribute is used to reference the checkbox in JavaScript.
* value - The value of the checkbox. This value is sent to the server when the form is submitted.
* checked - A Boolean attribute that specifies whether the checkbox is checked by default.
* disabled - A Boolean attribute that specifies whether the checkbox is disabled.

For example, the following code creates a checkbox with the value "agree":

<input type="checkbox" name="agree" value="yes">

### Radio Button

A radio button is a form element that allows the user to select one option from a limited number of choices. Radio buttons are typically used when there is a single correct answer or when the user needs to choose one option from a group of related options.

Radio buttons are created using the <input> element with the type attribute set to radio.

The <input> element has the following attributes:

* name - The name of the radio button group. This attribute is used to identify the radio button group when the form is submitted.
* id - The ID of the radio button. This attribute is used to reference the radio button in JavaScript.
* value - The value of the radio button. This value is sent to the server when the form is submitted.
* checked - A Boolean attribute that specifies whether the radio button is checked by default.
* disabled - A Boolean attribute that specifies whether the radio button is disabled.

For example, the following code creates a radio button group with three options:

<input type="radio" name="gender" value="male"> Male

<input type="radio" name="gender" value="female"> Female

<input type="radio" name="gender" value="other"> Other

### Button

A button is a form element that allows the user to interact with the page. Buttons can be used to submit forms, reset forms, or perform other actions.

Buttons are created using the <button> element. The <button> element has the following attributes:

* type - The type of button. The most common value for the type attribute is submit.
* name - The name of the button. This attribute is used to identify the button when the form is submitted.
* value - The value of the button. This value is sent to the server when the form is submitted.
* id - The ID of the button. This attribute is used to reference the button in JavaScript.
* disabled - A Boolean attribute that specifies whether the button is disabled.

For example, the following code creates a button with the value "Submit":

<button type="submit" value="Submit">Submit</button>

### Submit Input

The <input type="submit"> element creates a submit button that submits all form values to a form-handler. The submit button is typically used to send form data to the server.

The <input type="submit"> element has the following attributes:

* name - The name of the submit button. This attribute is used to identify the submit button when the form is submitted.
* value - The value of the submit button. This value is displayed on the button.
* id - The ID of the submit button. This attribute is used to reference the submit button in JavaScript.
* disabled - A Boolean attribute that specifies whether the submit button is disabled.

For example, the following code creates a submit button with the value "Submit":

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

### File Control with its attribute

The <input type="file"> element creates a file input field that allows the user to select a file from their computer. The file input field is typically used to upload files to a server.

The <input type="file"> element has the following attributes:

* name - The name of the file input field. This attribute is used to identify the file input field when the form is submitted.
* accept - A comma-separated list of file extensions or MIME types that the user can select.
* multiple - A Boolean attribute that specifies whether the user can select multiple files.
* id - The ID of the file input field. This attribute is used to reference the file input field in JavaScript.
* disabled - A Boolean attribute that specifies whether the file input field is disabled.

For example, the following code creates a file input field with the name "file":

<input type="file" name="file">

## Control Attribute

HTML control attributes are used to identify and style form elements. The most commonly used control attributes are:

1. name - The name of the control. This attribute is used to identify the control when the form is submitted.
2. id - The ID of the control. This attribute is used to reference the control in JavaScript.
3. class - The class of the control. This attribute is used to style the control with CSS.
4. value - The value of the control. This attribute is used to store the value of the control when the form is submitted.

For example, the following code creates a text input field with the name "name", the ID "my-input", and the class "my-class":

<input type="text" name="name" id="my-input" class="my-class">

## Basic Tag with it`s Attribute

### img tag, a tag

The img tag and a tag are both HTML tags used to insert content into a web page. However, they have different purposes.

The img tag is used to insert images into a web page. The a tag is used to create hyperlinks.

The img tag has the following attributes:

* src - The URL of the image.
* alt - The text that is displayed if the image cannot be loaded.
* width - The width of the image.
* height - The height of the image.

For example, the following code inserts an image into a web page:

<img src="image.jpg" alt="My image" width="100" height="100">

The a tag has the following attributes:

* href - The URL of the link.
* target - The target window or frame in which the link should be opened.
* title - The title of the link.

For example, the following code creates a hyperlink to a web page:

<a href="https://www.example.com">My link</a>

### What is meta tag , use of meta tag

Meta tags are hidden text elements in HTML documents that provide information about the document to search engines and other web services. Meta tags are not visible to users, but they are very important for SEO (search engine optimization).

There are many different types of meta tags, but some of the most common include:

* <title> - This meta tag specifies the title of the web page. The title is displayed in the browser tab and in search engine results pages (SERPs).
* <meta name="description"> - This meta tag provides a brief description of the web page. The description is displayed in SERPs, so it is important to make it clear and concise.
* <meta name="keywords"> - This meta tag specifies the keywords that the web page is about. The keywords are used by search engines to index the web page, so it is important to choose keywords that are relevant to the content of the page.
* <meta name="viewport"> - This meta tag specifies the viewport, which is the dimensions of the web page in the browser. The viewport is important for mobile devices, as it ensures that the web page is displayed correctly on different screen sizes.

Meta tags are a valuable tool for SEO and can help your web pages to rank higher in search engine results pages. However, it is important to note that meta tags are not the only factor that affects search engine ranking. Other factors, such as the quality of the content and the number of backlinks, also play a role.

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

Responsive website is a website that automatically adjusts its layout and content to fit the screen size of the device it is being viewed on. This means that your website will look good and be easy to use on any device, whether it is a desktop computer, laptop, tablet, or smartphone.

There are two main ways to create a responsive website:

* Using a responsive web design framework: A responsive web design framework is a set of pre-built code that you can use to create a responsive website. This is the easiest way to create a responsive website, but it can be limiting if you want to have a lot of control over the layout of your website.
* Using CSS media queries: CSS media queries allow you to specify how your website should be displayed on different screen sizes. This is the most flexible way to create a responsive website, but it can be more difficult to learn.

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

1. Start with a mobile-first design: This means that you should design your website for mobile devices first and then scale it up for larger screens. This will help you to ensure that your website looks good and is easy to use on all devices.
2. Use a responsive web design framework: If you are not comfortable with CSS media queries, then you can use a responsive web design framework to create your website. There are many different frameworks available, so you can choose one that fits your needs.
3. Use CSS media queries: If you want to have more control over the layout of your website, then you can use CSS media queries. Media queries allow you to specify how your website should be displayed on different screen sizes.
4. Test your website on different devices: Once you have created your website, it is important to test it on different devices to make sure that it looks good and is easy to use. You can use a variety of tools to test your website, such as Google's Mobile-Friendly Test.

Responsive websites are becoming increasingly important as more and more people use mobile devices to access the internet. By creating a responsive website, you can ensure that your website is accessible to everyone, regardless of the device they are using.