Dhruvil Dobariya

dhruvildobariya21@gmail.com

Abstract

[Draw your reader in with an engaging abstract. It is typically a short summary of the document.   
When you’re ready to add your content, just click here and start typing.]

GUI Basics

Documentation



Index

[1 Basic of HTML 1](#_Toc125797650)

[1.1 What is HTML? 1](#_Toc125797651)

[1.2 History of Html 1](#_Toc125797652)

[1.3 Structure of HTML 1](#_Toc125797653)

[2 Basic Controls 3](#_Toc125797654)

[2.1 Form: 3](#_Toc125797655)

[2.2 File Controls 4](#_Toc125797656)

[3 Control's Attributes 7](#_Toc125797657)

[3.1 Introduction 7](#_Toc125797658)

[3.2 name 7](#_Toc125797659)

[3.3 Value: 7](#_Toc125797660)

[3.4 Id 7](#_Toc125797661)

[3.5 Class 8](#_Toc125797662)

[4 Basic Tag with its attribute 9](#_Toc125797663)

[4.1 Action Tag 9](#_Toc125797664)

[4.2 Image Tag 11](#_Toc125797665)

[4.3 Meta Tag 12](#_Toc125797666)

[4.4 Responsiveness 13](#_Toc125797667)

[4.5 Other Tags 14](#_Toc125797668)

[5 Basic of CSS 26](#_Toc125797669)

[5.1 Introduction 26](#_Toc125797670)

[5.2 Types of CSS 26](#_Toc125797671)

[5.3 CSS Syntax 27](#_Toc125797672)

[5.4 Selectors 27](#_Toc125797673)

[5.5 Properties 32](#_Toc125797674)

[6 Basic of Bootstrap 39](#_Toc125797675)

[6.1 Introduction: 39](#_Toc125797676)

[6.2 How to Use Bootstrap 39](#_Toc125797677)

[6.3 Structure of Bootstrap 39](#_Toc125797678)

[7 Basic of JavaScript 42](#_Toc125797679)

[7.1 Introduction 42](#_Toc125797680)

[7.2 Use of JavaScript 42](#_Toc125797681)

[7.3 Way to use JavaScript 42](#_Toc125797682)

[7.4 Syntax of JavaScript 43](#_Toc125797683)

[7.5 Events in JavaScript 43](#_Toc125797684)

[7.6 Eventlistener 45](#_Toc125797685)

[7.7 Validation 46](#_Toc125797686)

1. Basic of HTML

# What is HTML?

* HTML stands for Hyper Text Markup Language
* 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.

# History of Html

* The first version of HTML was written by Tim Berners-Lee in 1993.
* Since then, there have been many different versions of HTML.
* The most widely used version throughout the 2000's was HTML 4.01, which became an official standard in December 1999.
* Currently Html 5 is stable version of html.

# Structure of HTML

**Example:**

<!DOCTYPE *html*>

<html>

<head>

    <title>Page Title</title>

</head>

<body>

    <h1>My First Heading</h1>

    <p>My first paragraph.</p>

</body>

</html>

* The <!DOCTYPE html> declaration defines that this document is an HTML5 document.
* The <html> element is the root element of an HTML page.
* The <head> element contains meta information about the HTML page.
* The <title> element specifies a title for the HTML page (which is shown in the browser's title bar or in the page's tab).
* The <body> element defines the document's body, and is a container for all the visible contents, such as headings, paragraphs, images, hyperlinks, tables, lists, etc.
* The <h1> element defines a large heading.
* The <p> element defines a paragraph.

1. Basic Controls

# Form:

* The <form> tag is used to create an HTML form for user input.
* The <form> element can contain one or more of the following form elements,
  + <input>
  + <textarea>
  + <button>
  + <select>
  + <option>
  + <optgroup>
  + <fieldset>
  + <label>
  + <output>

## Attributes:

* The <form> tag also supports the Global Attributes in HTML.
* The <form> tag also supports the Event Attributes in HTML.
* The <form> tag has own attributes, like…

|  |  |  |
| --- | --- | --- |
| Attribute | Value | Description |
| [accept-charset](https://www.w3schools.com/tags/att_form_accept_charset.asp) | character\_set | Specifies the character encodings that are to be used for the form submission |
| [action](https://www.w3schools.com/tags/att_form_action.asp) | URL | Specifies where to send the form-data when a form is submitted |
| [autocomplete](https://www.w3schools.com/tags/att_form_autocomplete.asp) | on off | Specifies whether a form should have autocomplete on or off |
| [enctype](https://www.w3schools.com/tags/att_form_enctype.asp) | application/x-www-form-urlencoded multipart/form-data text/plain | 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) | get post | Specifies the HTTP method to use when sending form-data |
| [name](https://www.w3schools.com/tags/att_form_name.asp) | text | Specifies the name of a form |
| [novalidate](https://www.w3schools.com/tags/att_form_novalidate.asp) | novalidate | Specifies that the form should not be validated when submitted |
| [rel](https://www.w3schools.com/tags/att_form_rel.asp) | external help license next nofollow noopener noreferrer opener prev search | Specifies the relationship between a linked resource and the current document |
| [target](https://www.w3schools.com/tags/att_form_target.asp) | \_blank \_self \_parent \_top | Specifies where to display the response that is received after submitting the form |

# File Controls

* The <input type="file"> defines a file-select field and a "Browse" button for file uploads.
* To define a file-select field that allows multiple files to be selected, add the multiple attribute.

**Example:**

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

## Attributes:

* It contains all common attribute which have input tag.
* Like, id, name, class…

**Value:**

* A file input's value attribute contains a string that represents the path to the selected file(s).
* If no file is selected yet, the value is an empty string ("").
* When the user selected multiple files, the value represents the first file in the list of files they selected.

**Accept:**

* 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.
* Because a given file type may be identified in more than one manner, it's useful to provide a thorough set of type specifiers when you need files of a given format.
* A valid case-insensitive filename extension, starting with a period (".") character.
* For example: .jpg, .pdf, or .doc.
* A valid MIME type string, with no extensions.
* The string audio/\* meaning "any audio file".
* The string video/\* meaning "any video file".
* The string image/\* meaning "any image file".

**Example:**

<input *type*="file" *accept*="image/\*,.pdf" />

**Capture:**

* 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.
* A value of environment specifies that the outward-facing camera and/or microphone should be used.
* If this attribute is missing, the user agent is free to decide on its own what to do.
* If the requested facing mode isn't available, the user agent may fall back to its preferred default mode.

**Multiple:**

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

## Getting information on selected files

* The selected files' are returned by the element's HTMLInputElement.files property, which is a FileList object containing a list of File objects.
* The FileList behaves like an array, so you can check its length property to get the number of selected files.
* Each File object contains the following proparties,

**name:**

* The file's name.

**lastModified:**

* A number specifying the date and time at which the file was last modified, in milliseconds since the UNIX epoch (January 1, 1970 at midnight).

**size:**

* The size of the file in bytes.

**type:**

* The file's MIME type.

1. Control's Attributes

# Introduction

* HTML have some basic control attributes like,
  + name
  + id
  + class
  + value

# name

* The name attribute specifies the name of an <input> element.
* The name attribute is used to reference elements in a JavaScript, or to reference form data after a form is submitted.
* Only form elements with a name attribute will have their values passed when submitting a form.

**Example:**

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

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

# Value:

* The value attribute specifies the value of an <input> element.

**Example:**

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

<input *type*="text" *name*="name" *id*="name" *value*="Dhruvil Dobariya">

# Id

* The HTML id attribute is used to specify a unique id for an HTML element.
* You cannot give more than one element with the same id in an HTML document.
* The id attribute is used to point to a specific style declaration in a style sheet.
* It is also used by JavaScript to access and manipulate the element with the specific id.
* The syntax for id is: write a hash character (#), followed by an id name.
* Id also used for bookmark.
* To use a bookmark, you must first create it, and then add a link to it.
* Then, when the link is clicked, the page will scroll to the location with the bookmark.
* It is a global attribute, means it can be used with all HTML elements.

# Class

* The class attribute specifies one or more classnames for an element.
* The class attribute is mostly used to point to a class in a style sheet.
* It can also be used by a JavaScript (via the HTML DOM) to make changes to HTML elements with a specified class.
* It is a global attribute, means it can be used with all HTML elements.

1. Basic Tag with its attribute

# Action Tag

* The <a> tag defines a hyperlink, which is used to link from one page to another.
* 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
* But we can change these color using CSS.
* The <a> tag also supports the Global Attributes and Event Attribute in HTML.

## Attributes:

|  |  |  |
| --- | --- | --- |
| Attribute | Value | Description |
| [**download**](https://www.w3schools.com/tags/att_a_download.asp) | filename | Specifies that the target will be downloaded when a user clicks on the hyperlink |
| [**href**](https://www.w3schools.com/tags/att_a_href.asp) | URL | Specifies the URL of the page the link goes to |
| [**hreflang**](https://www.w3schools.com/tags/att_a_hreflang.asp) | language\_code | Specifies the language of the linked document |
| [**media**](https://www.w3schools.com/tags/att_a_media.asp) | media\_query | Specifies what media/device the linked document is optimized for |
| [**ping**](https://www.w3schools.com/tags/att_a_ping.asp) | list\_of\_URLs | Specifies a space-separated list of URLs to which, when the link is followed, post requests with the body ping will be sent by the browser (in the background). Typically used for tracking. |
| [**referrerpolicy**](https://www.w3schools.com/tags/att_a_referrepolicy.asp) | no-referrer no-referrer-when-downgrade origin origin-when-cross-origin same-origin strict-origin-when-cross-origin unsafe-url | Specifies which referrer information to send with the link |
| [**rel**](https://www.w3schools.com/tags/att_a_rel.asp) | alternate author bookmark external help license next nofollow noreferrer noopener prev search tag | Specifies the relationship between the current document and the linked document |
| [**target**](https://www.w3schools.com/tags/att_a_target.asp) | \_blank \_parent \_self \_top | Specifies where to open the linked document |
| [**type**](https://www.w3schools.com/tags/att_a_type.asp) | media\_type | Specifies the media type of the linked document |

# Image Tag

* The <img> tag is used to embed an image in an HTML page.
* Images are not technically inserted into a web page, images are linked to web pages.

## Attributes:

|  |  |  |
| --- | --- | --- |
| Attribute | Value | Description |
| [**alt**](https://www.w3schools.com/tags/att_img_alt.asp) | text | Specifies an alternate text for an image |
| **crossorigin** | anonymous use-credentials | Allow images from third-party sites that allow cross-origin access to be used with canvas |
| [**height**](https://www.w3schools.com/tags/att_img_height.asp) | pixels | Specifies the height of an image |
| [**ismap**](https://www.w3schools.com/tags/att_img_ismap.asp) | ismap | Specifies an image as a server-side image map |
| [**loading**](https://www.w3schools.com/tags/att_img_loading.asp) | eager lazy | Specifies whether a browser should load an image immediately or to defer loading of images until some conditions are met |
| [**longdesc**](https://www.w3schools.com/tags/att_img_longdesc.asp) | URL | Specifies a URL to a detailed description of an image |
| [**referrerpolicy**](https://www.w3schools.com/tags/att_img_referrepolicy.asp) | no-referrer no-referrer-when-downgrade origin origin-when-cross-origin unsafe-url | Specifies which referrer information to use when fetching an image |
| **sizes** | sizes | Specifies image sizes for different page layouts |
| [**src**](https://www.w3schools.com/tags/att_img_src.asp) | URL | Specifies the path to the image |
| **srcset** | URL-list | Specifies a list of image files to use in different situations |
| [**usemap**](https://www.w3schools.com/tags/att_img_usemap.asp) | #mapname | Specifies an image as a client-side image map |
| [**width**](https://www.w3schools.com/tags/att_img_width.asp) | pixels | Specifies the width of an image |

# Meta Tag

* The <meta> tag defines metadata about an HTML document. Metadata is data (information) about data.
* <meta> tags always go inside the <head> element.
* It is used to specify,
  + specify character set
  + page description
  + keywords
  + author of the document
  + viewport settings
* It will not be displayed on the page, but is machine parsable.
* It is used by browsers (how to display content or reload page), search engines (keywords), and other web services.
* There is a method to let web designers take control over the viewport (the user's visible area of a web page), using the <meta> tag.

## Attributes:

|  |  |  |
| --- | --- | --- |
| Attribute | Value | Description |
| [**charset**](https://www.w3schools.com/tags/att_meta_charset.asp) | character\_set | Specifies the character encoding for the HTML document |
| [**content**](https://www.w3schools.com/tags/att_meta_content.asp) | text | Specifies the value associated with the http-equiv or name attribute |
| [**http-equiv**](https://www.w3schools.com/tags/att_meta_http_equiv.asp) | content-security-policy content-type default-style refresh | Provides an HTTP header for the information/value of the content attribute |
| [**name**](https://www.w3schools.com/tags/att_meta_name.asp) | application-name author description generator keywords viewport | Specifies a name for the metadata |

**Example:**

<head>

*<!-- Define keywords for search engines: -->*

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

*<!-- Define a description of your web page: -->*

    <meta *name*="description" *content*="About meta tag">

*<!-- Define the author of a page: -->*

    <meta *name*="author" *content*="Dhruvil A. Dobariya">

*<!-- 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">

    <title>Document</title>

</head>

# Responsiveness

* It is a web development approach that creates dynamic changes to the appearance of a website, depending on the screen size and orientation of the device being used to view it.
* For archive responsiveness we have three approach,
  + Media Query
  + Web Browser
  + Website Interface(HTML, CSS, Javascript)

**Media Query:**

* Media queries are supported by the latest web browsers and work by creating media queries.
* They are called “media queries” and can be used in various ways, such as in connection with the layout of a page or as part of the content of an application.
* They also permit designers to fabricate various formats utilizing similar HTML archives by specifically serving templates dependent on the client specialist’s highlights, like the browser’s window’s size.

**Web Browser:**

* Since websites can contain many images, it is essential to keep these images fluid too.
* The thought behind liquid pictures is that you convey concepts at the greatest size they will utilize.
* In responsive web pages, designers don’t announce the stature and width in your code instead, they let the browsers resize the pictures on a case by case basis while utilizing CSS to manage their relative size.
* It’s an extraordinary and straightforward procedure to resize images correctly.

**Website Interface(HTML, CSS, Javascript):**

* Many mobile devices are touchscreen nowadays, which requires mindfulness of the size of the interactive elements within interfaces.
* Aside from the flexibility of images, usage of native controls such as keyboards and drop-out menus should be optimized so it can provide a great experience to its users, whether it’s on mobile or desktop**.**

# Other Tags

## Basic Tags:

|  |  |
| --- | --- |
| Tag | Description |
| [<!DOCTYPE>](https://www.w3schools.com/tags/tag_doctype.asp) | Defines the document type |
| [<html>](https://www.w3schools.com/tags/tag_html.asp) | Defines an HTML document |
| [<head>](https://www.w3schools.com/tags/tag_head.asp) | Contains metadata/information for the document |
| [<title>](https://www.w3schools.com/tags/tag_title.asp) | Defines a title for the document |
| [<body>](https://www.w3schools.com/tags/tag_body.asp) | Defines the document's body |
| [<h1> to <h6>](https://www.w3schools.com/tags/tag_hn.asp) | Defines HTML headings |
| [<p>](https://www.w3schools.com/tags/tag_p.asp) | Defines a paragraph |
| [<br>](https://www.w3schools.com/tags/tag_br.asp) | Inserts a single line break |
| [<hr>](https://www.w3schools.com/tags/tag_hr.asp) | Defines a thematic change in the content |
| [<!--...-->](https://www.w3schools.com/tags/tag_comment.asp) | Defines a comment |

## Formatting Tags:

|  |  |
| --- | --- |
| Tag | Description |
| [<acronym>](https://www.w3schools.com/tags/tag_acronym.asp) | Not supported in HTML5. Use <abbr> instead. Defines an acronym |
| [<abbr>](https://www.w3schools.com/tags/tag_abbr.asp) | Defines an abbreviation or an acronym |
| [<address>](https://www.w3schools.com/tags/tag_address.asp) | Defines contact information for the author/owner of a document/article |
| [<b>](https://www.w3schools.com/tags/tag_b.asp) | Defines bold text |
| [<bdi>](https://www.w3schools.com/tags/tag_bdi.asp) | Isolates a part of text that might be formatted in a different direction from other text outside it |
| [<bdo>](https://www.w3schools.com/tags/tag_bdo.asp) | Overrides the current text direction |
| [<big>](https://www.w3schools.com/tags/tag_big.asp) | Not supported in HTML5. Use CSS instead. Defines big text |
| [<blockquote>](https://www.w3schools.com/tags/tag_blockquote.asp) | Defines a section that is quoted from another source |
| [<center>](https://www.w3schools.com/tags/tag_center.asp) | Not supported in HTML5. Use CSS instead. Defines centered text |
| [<cite>](https://www.w3schools.com/tags/tag_cite.asp) | Defines the title of a work |
| [<code>](https://www.w3schools.com/tags/tag_code.asp) | Defines a piece of computer code |
| [<del>](https://www.w3schools.com/tags/tag_del.asp) | Defines text that has been deleted from a document |
| [<dfn>](https://www.w3schools.com/tags/tag_dfn.asp) | Specifies a term that is going to be defined within the content |
| [<em>](https://www.w3schools.com/tags/tag_em.asp) | Defines emphasized text |
| [<font>](https://www.w3schools.com/tags/tag_font.asp) | Not supported in HTML5. Use CSS instead. Defines font, color, and size for text |
| [<i>](https://www.w3schools.com/tags/tag_i.asp) | Defines a part of text in an alternate voice or mood |
| [<ins>](https://www.w3schools.com/tags/tag_ins.asp) | Defines a text that has been inserted into a document |
| [<kbd>](https://www.w3schools.com/tags/tag_kbd.asp) | Defines keyboard input |
| [<mark>](https://www.w3schools.com/tags/tag_mark.asp) | Defines marked/highlighted text |
| [<meter>](https://www.w3schools.com/tags/tag_meter.asp) | Defines a scalar measurement within a known range (a gauge) |
| [<pre>](https://www.w3schools.com/tags/tag_pre.asp) | Defines preformatted text |
| [<progress>](https://www.w3schools.com/tags/tag_progress.asp) | Represents the progress of a task |
| [<q>](https://www.w3schools.com/tags/tag_q.asp) | Defines a short quotation |
| [<rp>](https://www.w3schools.com/tags/tag_rp.asp) | Defines what to show in browsers that do not support ruby annotations |
| [<rt>](https://www.w3schools.com/tags/tag_rt.asp) | Defines an explanation/pronunciation of characters (for East Asian typography) |
| [<ruby>](https://www.w3schools.com/tags/tag_ruby.asp) | Defines a ruby annotation (for East Asian typography) |
| [<s>](https://www.w3schools.com/tags/tag_s.asp) | Defines text that is no longer correct |
| [<samp>](https://www.w3schools.com/tags/tag_samp.asp) | Defines sample output from a computer program |
| [<small>](https://www.w3schools.com/tags/tag_small.asp) | Defines smaller text |
| [<strike>](https://www.w3schools.com/tags/tag_strike.asp) | Not supported in HTML5. Use <del> or <s> instead. Defines strikethrough text |
| [<strong>](https://www.w3schools.com/tags/tag_strong.asp) | Defines important text |
| [<sub>](https://www.w3schools.com/tags/tag_sub.asp) | Defines subscripted text |
| [<sup>](https://www.w3schools.com/tags/tag_sup.asp) | Defines superscripted text |
| [<template>](https://www.w3schools.com/tags/tag_template.asp) | Defines a container for content that should be hidden when the page loads |
| [<time>](https://www.w3schools.com/tags/tag_time.asp) | Defines a specific time (or datetime) |
| [<tt>](https://www.w3schools.com/tags/tag_tt.asp) | Not supported in HTML5. Use CSS instead. Defines teletype text |
| [<u>](https://www.w3schools.com/tags/tag_u.asp) | Defines some text that is unarticulated and styled differently from normal text |
| [<var>](https://www.w3schools.com/tags/tag_var.asp) | Defines a variable |
| [<wbr>](https://www.w3schools.com/tags/tag_wbr.asp) | Defines a possible line-break |

## Forms and Inputs Tags:

|  |  |
| --- | --- |
| Tag | Description |
| [<form>](https://www.w3schools.com/tags/tag_form.asp) | Defines an HTML form for user input |
| [<input>](https://www.w3schools.com/tags/tag_input.asp) | Defines an input control |
| [<textarea>](https://www.w3schools.com/tags/tag_textarea.asp) | Defines a multiline input control (text area) |
| [<button>](https://www.w3schools.com/tags/tag_button.asp) | Defines a clickable button |
| [<select>](https://www.w3schools.com/tags/tag_select.asp) | Defines a drop-down list |
| [<optgroup>](https://www.w3schools.com/tags/tag_optgroup.asp) | Defines a group of related options in a drop-down list |
| [<option>](https://www.w3schools.com/tags/tag_option.asp) | Defines an option in a drop-down list |
| [<label>](https://www.w3schools.com/tags/tag_label.asp) | Defines a label for an <input> element |
| [<fieldset>](https://www.w3schools.com/tags/tag_fieldset.asp) | Groups related elements in a form |
| [<legend>](https://www.w3schools.com/tags/tag_legend.asp) | Defines a caption for a <fieldset> element |
| [<datalist>](https://www.w3schools.com/tags/tag_datalist.asp) | Specifies a list of pre-defined options for input controls |
| [<output>](https://www.w3schools.com/tags/tag_output.asp) | Defines the result of a calculation |

## Frames Tags:

|  |  |
| --- | --- |
| Tag | Description |
| [<frame>](https://www.w3schools.com/tags/tag_frame.asp) | Not supported in HTML5. Defines a window (a frame) in a frameset |
| [<frameset>](https://www.w3schools.com/tags/tag_frameset.asp) | Not supported in HTML5. Defines a set of frames |
| [<noframes>](https://www.w3schools.com/tags/tag_noframes.asp) | Not supported in HTML5. Defines an alternate content for users that do not support frames |
| [<iframe>](https://www.w3schools.com/tags/tag_iframe.asp) | Defines an inline frame |

## Images Tags:

|  |  |
| --- | --- |
| Tag | Description |
| [<img>](https://www.w3schools.com/tags/tag_img.asp) | Defines an image |
| [<map>](https://www.w3schools.com/tags/tag_map.asp) | Defines a client-side image map |
| [<area>](https://www.w3schools.com/tags/tag_area.asp) | Defines an area inside an image map |
| [<canvas>](https://www.w3schools.com/tags/tag_canvas.asp) | Used to draw graphics, on the fly, via scripting (usually JavaScript) |
| [<figcaption>](https://www.w3schools.com/tags/tag_figcaption.asp) | Defines a caption for a <figure> element |
| [<figure>](https://www.w3schools.com/tags/tag_figure.asp) | Specifies self-contained content |
| [<picture>](https://www.w3schools.com/tags/tag_picture.asp) | Defines a container for multiple image resources |
| [<svg>](https://www.w3schools.com/tags/tag_svg.asp) | Defines a container for SVG graphics |

## Audio / Video Tags:

|  |  |
| --- | --- |
| Tag | Description |
| [<audio>](https://www.w3schools.com/tags/tag_audio.asp) | Defines sound content |
| [<source>](https://www.w3schools.com/tags/tag_source.asp) | Defines multiple media resources for media elements (<video>, <audio> and <picture>) |
| [<track>](https://www.w3schools.com/tags/tag_track.asp) | Defines text tracks for media elements (<video> and <audio>) |
| [<video>](https://www.w3schools.com/tags/tag_video.asp) | Defines a video or movie |

## Links Tags:

|  |  |
| --- | --- |
| Tag | Description |
| [<a>](https://www.w3schools.com/tags/tag_a.asp) | Defines a hyperlink |
| [<link>](https://www.w3schools.com/tags/tag_link.asp) | Defines the relationship between a document and an external resource (most used to link to style sheets) |
| [<nav>](https://www.w3schools.com/tags/tag_nav.asp) | Defines navigation links |

## List Tags:

|  |  |
| --- | --- |
| Tag | Description |
| [<ul>](https://www.w3schools.com/tags/tag_ul.asp) | Defines an unordered list |
| [<ol>](https://www.w3schools.com/tags/tag_ol.asp) | Defines an ordered list |
| [<li>](https://www.w3schools.com/tags/tag_li.asp) | Defines a list item |
| [<dir>](https://www.w3schools.com/tags/tag_dir.asp) | Not supported in HTML5. Use <ul> instead. Defines a directory list |
| [<dl>](https://www.w3schools.com/tags/tag_dl.asp) | Defines a description list |
| [<dt>](https://www.w3schools.com/tags/tag_dt.asp) | Defines a term/name in a description list |
| [<dd>](https://www.w3schools.com/tags/tag_dd.asp) | Defines a description of a term/name in a description list |

## Table Tags:

|  |  |
| --- | --- |
| Tag | Description |
| [<table>](https://www.w3schools.com/tags/tag_table.asp) | Defines a table |
| [<caption>](https://www.w3schools.com/tags/tag_caption.asp) | Defines a table caption |
| [<th>](https://www.w3schools.com/tags/tag_th.asp) | Defines a header cell in a table |
| [<tr>](https://www.w3schools.com/tags/tag_tr.asp) | Defines a row in a table |
| [<td>](https://www.w3schools.com/tags/tag_td.asp) | Defines a cell in a table |
| [<thead>](https://www.w3schools.com/tags/tag_thead.asp) | Groups the header content in a table |
| [<tbody>](https://www.w3schools.com/tags/tag_tbody.asp) | Groups the body content in a table |
| [<tfoot>](https://www.w3schools.com/tags/tag_tfoot.asp) | Groups the footer content in a table |
| [<col>](https://www.w3schools.com/tags/tag_col.asp) | Specifies column properties for each column within a <colgroup> element |
| [<colgroup>](https://www.w3schools.com/tags/tag_colgroup.asp) | Specifies a group of one or more columns in a table for formatting |

## Styles and Semantics:

|  |  |
| --- | --- |
| Tag | Description |
| [<style>](https://www.w3schools.com/tags/tag_style.asp) | Defines style information for a document |
| [<div>](https://www.w3schools.com/tags/tag_div.asp) | Defines a section in a document |
| [<span>](https://www.w3schools.com/tags/tag_span.asp) | Defines a section in a document |
| [<header>](https://www.w3schools.com/tags/tag_header.asp) | Defines a header for a document or section |
| [<footer>](https://www.w3schools.com/tags/tag_footer.asp) | Defines a footer for a document or section |
| [<main>](https://www.w3schools.com/tags/tag_main.asp) | Specifies the main content of a document |
| [<section>](https://www.w3schools.com/tags/tag_section.asp) | Defines a section in a document |
| [<article>](https://www.w3schools.com/tags/tag_article.asp) | Defines an article |
| [<aside>](https://www.w3schools.com/tags/tag_aside.asp) | Defines content aside from the page content |
| [<details>](https://www.w3schools.com/tags/tag_details.asp) | Defines additional details that the user can view or hide |
| [<dialog>](https://www.w3schools.com/tags/tag_dialog.asp) | Defines a dialog box or window |
| [<summary>](https://www.w3schools.com/tags/tag_summary.asp) | Defines a visible heading for a <details> element |
| [<data>](https://www.w3schools.com/tags/tag_data.asp) | Adds a machine-readable translation of a given content |

## Meta Info Tags:

|  |  |
| --- | --- |
| Tag | Description |
| [<head>](https://www.w3schools.com/tags/tag_head.asp) | Defines information about the document |
| [<meta>](https://www.w3schools.com/tags/tag_meta.asp) | Defines metadata about an HTML document |
| [<base>](https://www.w3schools.com/tags/tag_base.asp) | Specifies the base URL/target for all relative URLs in a document |
| [<basefont>](https://www.w3schools.com/tags/tag_basefont.asp) | Not supported in HTML5. Use CSS instead. Specifies a default color, size, and font for all text in a document |

## Programming Tags:

|  |  |
| --- | --- |
| Tag | Description |
| [<script>](https://www.w3schools.com/tags/tag_script.asp) | Defines a client-side script |
| [<noscript>](https://www.w3schools.com/tags/tag_noscript.asp) | Defines an alternate content for users that do not support client-side scripts |
| [<applet>](https://www.w3schools.com/tags/tag_applet.asp) | Not supported in HTML5. Use <embed> or <object> instead. Defines an embedded applet |
| [<embed>](https://www.w3schools.com/tags/tag_embed.asp) | Defines a container for an external (non-HTML) application |
| [<object>](https://www.w3schools.com/tags/tag_object.asp) | Defines an embedded object |
| [<param>](https://www.w3schools.com/tags/tag_param.asp) | Defines a parameter for an object |

1. Basic of CSS

# Introduction

* CSS stands for Cascading Style Sheets.
* CSS describes how HTML elements are to be displayed on screen, paper, or in other media.
* It can control the layout of multiple web pages all at once.

# Types of CSS

* We have three types of CSS,
  + Inline CSS
  + Internal or Embedded CSS
  + External CSS

## Inline CSS:

* Inline CSS contains the CSS property in the body section attached with element is known as inline CSS.
* This kind of style is specified within an HTML tag using the style attribute.

## Internal or Embedded CSS:

* This can be used when a single HTML document must be styled uniquely.
* The CSS rule set should be within the HTML file in the head section.

## External CSS:

* External CSS contains separate CSS file which contains only style property with the help of tag attributes.
* CSS property written in a separate file with .css extension and should be linked to the HTML document using link tag.
* This means we can use CSS file in multiple HTML documents.

## Priority of CSS:

* Inline CSS has the highest priority, then comes Internal/Embedded followed by External CSS which has the least priority.
* Multiple style sheets can be defined on one page.
* If for an HTML tag, styles are defined in multiple style sheets then the below order will be followed.

# CSS Syntax

* The selector points to the HTML element you want to style.
* The declaration block contains one or more declarations separated by semicolons.
* Each declaration includes a CSS property name and a value, separated by a colon.
* Multiple CSS declarations are separated with semicolons, and declaration blocks are surrounded by curly braces.

**Example:**

p {

    color: red;

    text-align: center;

}

* Here, p is a selector in CSS.
* color is a property, and red is the property value.
* text-align is a property, and center is the property value.

# Selectors

* CSS have different types of selectors like,
  + Basic Selector
  + Grouping Selector
  + Combinator Selector
  + Pseudo Selector

## Basic Selector:

* Basic selector contains five different types like,
  + Universal selector
  + Class selector
  + ID selector
  + Attribute selector
  + Type selector

**Universal Selector:**

* Selects all elements.
* Optionally, it may be restricted to a specific namespace or to all namespaces.

**Syntax:** \* ns|\* \*|\*

**Example:**

\*{

    background-color: rgb(209, 209, 209);

    font-size: 25*px*;

}

**Class Selector:**

* Selects all elements that have the given class attribute.

**Syntax:** .classname

**Example:**

*.class-selector*{

    color: white;

}

**Id Selector:**

* Selects an element based on the value of its id attribute.

**Syntax:** #idname

**Example:**

*#Id-Selector*{

    color: red;

}

**Type(Tag) Selector:**

* Selects all elements that have the given node name.

**Syntax:** elementname

**Example:**

span{

    color: green;

}

**Attribute Selector:**

* Selects all elements that have the given attribute.

**Syntax:** [attr], [attr=value], [attr~=value], [attr|=value], [attr^=value], [attr$=value], [attr\*=value]

**Example:**

[*type*=text]{

    color: aqua;

}

## Grouping Selector:

* Grouping selector contains five different types like,
  + Selector list

**Select list Selector:**

* Selects all the matching nodes.
* All node separated by “,”.

**Syntax:** elementname1, elementname2

**Example:**

i,*.select-list-with-class*{

    color: blue;

}

## Combinator Selector

* Combinator selector contains five different types like,
  + Descendant combinator
  + Child combinator
  + General sibling combinator
  + Adjacent sibling combinator

**Descendant Combinator:**

* It select elements which is exist inside the particular element.
* First we write element which contain our selector node and give space and we write second element which is our selector node.

**Syntax:** div span

* So here, “div span” will match all <span> elements that are exist a <div> element.

**Example:**

div b{

    color: blueviolet;

}

**Child Combinator:**

* It select elements which is directly children of the particular element.
* First we write element which is **directly parent node** of our selector node and write “>” and we write second element which is our selector node.

**Syntax:** ui > li

* So here, “ui > li” will match all <li> elements that are directly child of the <ui> element.

**Example:**

p>b{

    color: brown;

}

**General Sibling Combinator:**

* It select elements which is sibling of the particular element.
* First we write element which is sibling of our selector node and write “~” and we write second element which is our selector node.

**Syntax:** p ~ span

* So here, “p ~ span” will match all <span> elements that are sibling of the <p> element.

**Example:**

h3~h4,h3~*.sibling-combinator-with-class*{

    color: cadetblue;

}

**Adjacent Combinator:**

* It select elements which **first and immediately follows** the particular element.
* First we write element which is followed by our selector node and write “+” and we write second element which is our selector node.

**Syntax:** h2 + p

* So here, “h2 + p” will match all <p> element that is first and followed the <h2> element.

**Example:**

header+footer{

    color: yellowgreen;

}

## Pseudo Selector

* Pseudo selector contains five different types like,
  + Pseudo classes selector
  + Pseudo elements selector

**Pseudo Classes Selector:**

* It select element based on their state.
* First we write element which is our selector node and write “:” and we write state of that element when we want to select it in that state.

**Syntax:** a:visited

* So, here <a> is selected when it visited.

**Example:**

a*:visited*{

    color: coral;

}

input*:invalid*{

    background-color: red;

}

**Pseudo Element Selector:**

* It is used to select particular parts of selector.
* First we write element which is our selector node and write “::” and we write part of that element which we want to select it.

**Syntax:** p::first-line

* So, here first line of <p> is selected.

**Example:**

p*::first-line*{

    color: rgb(73, 82, 27);

}

# Properties

* CSS have many properties, like background, text, box-model…

## Background Properties:

|  |  |
| --- | --- |
| Property | Description |
| [background](https://www.tutorialrepublic.com/css-reference/css-background-property.php) | Defines a variety of background properties within one declaration. |
| [background-attachment](https://www.tutorialrepublic.com/css-reference/css-background-attachment-property.php) | Specify whether the background image is fixed in the viewport or scrolls. |
| [background-clip](https://www.tutorialrepublic.com/css-reference/css3-background-clip-property.php) | Specifies the painting area of the background. |
| [background-color](https://www.tutorialrepublic.com/css-reference/css-background-color-property.php) | Defines an element's background color. |
| [background-image](https://www.tutorialrepublic.com/css-reference/css-background-image-property.php) | Defines an element's background image. |
| [background-origin](https://www.tutorialrepublic.com/css-reference/css3-background-origin-property.php) | Specifies the positioning area of the background images. |
| [background-position](https://www.tutorialrepublic.com/css-reference/css-background-position-property.php) | Defines the origin of a background image. |
| [background-repeat](https://www.tutorialrepublic.com/css-reference/css-background-repeat-property.php) | Specify whether/how the background image is tiled. |
| [background-size](https://www.tutorialrepublic.com/css-reference/css3-background-size-property.php) | Specifies the size of the background images. |

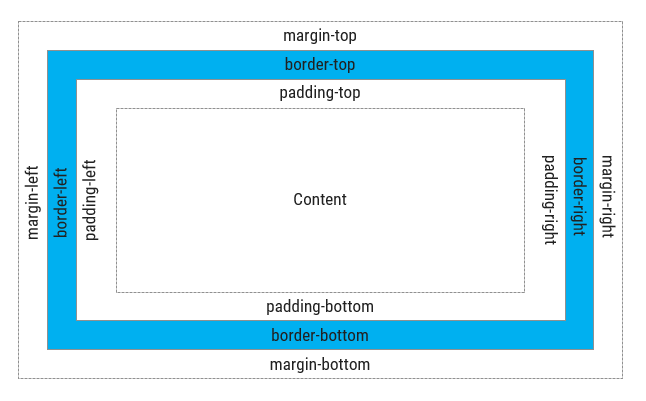
## Font Properties:

|  |  |
| --- | --- |
| Property | Description |
| [font](https://www.tutorialrepublic.com/css-reference/css-font-property.php) | Defines a variety of font properties within one declaration. |
| [font-family](https://www.tutorialrepublic.com/css-reference/css-font-family-property.php) | Defines a list of fonts for element. |
| [font-size](https://www.tutorialrepublic.com/css-reference/css-font-size-property.php) | Defines the font size for the text. |
| [font-size-adjust](https://www.tutorialrepublic.com/css-reference/css3-font-size-adjust-property.php) | Preserves the readability of text when font fallback occurs. |
| [font-stretch](https://www.tutorialrepublic.com/css-reference/css3-font-stretch-property.php) | Selects a normal, condensed, or expanded face from a font. |
| [font-style](https://www.tutorialrepublic.com/css-reference/css-font-style-property.php) | Defines the font style for the text. |
| [font-variant](https://www.tutorialrepublic.com/css-reference/css-font-variant-property.php) | Specify the font variant. |
| [font-weight](https://www.tutorialrepublic.com/css-reference/css-font-weight-property.php) | Specify the font weight of the text. |
| [color](https://www.tutorialrepublic.com/css-reference/css-color-property.php) | Specify the color of the text of an element. |

## Text Properties:

|  |  |
| --- | --- |
| Property | Description |
| [text-align](https://www.tutorialrepublic.com/css-reference/css-text-align-property.php) | Sets the horizontal alignment of inline content. |
| [text-align-last](https://www.tutorialrepublic.com/css-reference/css3-text-align-last-property.php) | Specifies how the last line of a block or a line right before a forced line break is aligned when [text-align](https://www.tutorialrepublic.com/css-reference/css-text-align-property.php) is justify. |
| [text-decoration](https://www.tutorialrepublic.com/css-reference/css-text-decoration-property.php) | Specifies the decoration added to text. |
| [text-decoration-color](https://www.tutorialrepublic.com/css-reference/css3-text-decoration-color-property.php) | Specifies the color of the [text-decoration-line](https://www.tutorialrepublic.com/css-reference/css3-text-decoration-line-property.php). |
| [text-decoration-line](https://www.tutorialrepublic.com/css-reference/css3-text-decoration-line-property.php) | Specifies what kind of line decorations are added to the element. |
| [text-decoration-style](https://www.tutorialrepublic.com/css-reference/css3-text-decoration-style-property.php) | Specifies the style of the lines specified by the [text-decoration-line](https://www.tutorialrepublic.com/css-reference/css3-text-decoration-line-property.php) property |
| [text-indent](https://www.tutorialrepublic.com/css-reference/css-text-indent-property.php) | Indent the first line of text. |
| [text-justify](https://www.tutorialrepublic.com/css-reference/css3-text-justify-property.php) | Specifies the justification method to use when the [text-align](https://www.tutorialrepublic.com/css-reference/css-text-align-property.php) property is set to justify. |
| [text-overflow](https://www.tutorialrepublic.com/css-reference/css3-text-overflow-property.php) | Specifies how the text content will be displayed, when it overflows the block containers. |
| [text-shadow](https://www.tutorialrepublic.com/css-reference/css3-text-shadow-property.php) | Applies one or more shadows to the text content of an element. |
| [text-transform](https://www.tutorialrepublic.com/css-reference/css-text-transform-property.php) | Transforms the case of the text. |

## Box Model Properties:

* All HTML elements can be considered as boxes.
* In CSS, the term "box model" is used when talking about design and layout.
* The CSS box model is essentially a box that wraps around HTML elements, and it consists of margins, borders, padding, and the actual content.
* The box model allows us to place a border around elements and space elements in relation to other elements.

|  |  |
| --- | --- |
| Property | Description |
| [padding-bottom](https://www.tutorialrepublic.com/css-reference/css-padding-bottom-property.php) | Sets the padding to the bottom side of an element. |
| [padding-left](https://www.tutorialrepublic.com/css-reference/css-padding-left-property.php) | Sets the padding to the left side of an element. |
| [padding-right](https://www.tutorialrepublic.com/css-reference/css-padding-right-property.php) | Sets the padding to the right side of an element. |
| [padding-top](https://www.tutorialrepublic.com/css-reference/css-padding-top-property.php) |  |
| [margin](https://www.tutorialrepublic.com/css-reference/css-margin-property.php) | Sets the margin on all four sides of the element. |
| [margin-bottom](https://www.tutorialrepublic.com/css-reference/css-margin-bottom-property.php) | Sets the bottom margin of the element. |
| [margin-left](https://www.tutorialrepublic.com/css-reference/css-margin-left-property.php) | Sets the left margin of the element. |
| [margin-right](https://www.tutorialrepublic.com/css-reference/css-margin-right-property.php) | Sets the right margin of the element. |
| [margin-top](https://www.tutorialrepublic.com/css-reference/css-margin-top-property.php) | Sets the top margin of the element. |
| [border](https://www.tutorialrepublic.com/css-reference/css-border-property.php) | Sets the width, style, and color for all four sides of an element's border. |
| [border-bottom](https://www.tutorialrepublic.com/css-reference/css-border-bottom-property.php) | Sets the width, style, and color of the bottom border of an element. |
| [border-bottom-color](https://www.tutorialrepublic.com/css-reference/css-border-bottom-color-property.php) | Sets the color of the bottom border of an element. |
| [border-bottom-left-radius](https://www.tutorialrepublic.com/css-reference/css3-border-bottom-left-radius-property.php) | Defines the shape of the bottom-left border corner of an element. |
| [border-bottom-right-radius](https://www.tutorialrepublic.com/css-reference/css3-border-bottom-right-radius-property.php) | Defines the shape of the bottom-right border corner of an element. |
| [border-bottom-style](https://www.tutorialrepublic.com/css-reference/css-border-bottom-style-property.php) | Sets the style of the bottom border of an element. |
| [border-bottom-width](https://www.tutorialrepublic.com/css-reference/css-border-bottom-width-property.php) | Sets the width of the bottom border of an element. |
| [border-collapse](https://www.tutorialrepublic.com/css-reference/css-border-collapse-property.php) | Specifies whether table cell borders are connected or separated. |
| [border-color](https://www.tutorialrepublic.com/css-reference/css-border-color-property.php) | Sets the color of the border on all the four sides of an element. |
| [border-image](https://www.tutorialrepublic.com/css-reference/css3-border-image-property.php) | Specifies how an image is to be used in place of the border styles. |
| [border-image-outset](https://www.tutorialrepublic.com/css-reference/css3-border-image-outset-property.php) | Specifies the amount by which the border image area extends beyond the border box. |
| [border-image-repeat](https://www.tutorialrepublic.com/css-reference/css3-border-image-repeat-property.php) | Specifies whether the image-border should be repeated, rounded or stretched. |
| [border-image-slice](https://www.tutorialrepublic.com/css-reference/css3-border-image-slice-property.php) | Specifies the inward offsets of the image-border. |
| [border-image-source](https://www.tutorialrepublic.com/css-reference/css3-border-image-source-property.php) | Specifies the location of the image to be used as a border. |
| [border-image-width](https://www.tutorialrepublic.com/css-reference/css3-border-image-width-property.php) | Specifies the width of the image-border. |
| [border-left](https://www.tutorialrepublic.com/css-reference/css-border-left-property.php) | Sets the width, style, and color of the left border of an element. |
| [border-left-color](https://www.tutorialrepublic.com/css-reference/css-border-left-color-property.php) | Sets the color of the left border of an element. |
| [border-left-style](https://www.tutorialrepublic.com/css-reference/css-border-left-style-property.php) | Sets the style of the left border of an element. |
| [border-left-width](https://www.tutorialrepublic.com/css-reference/css-border-left-width-property.php) | Sets the width of the left border of an element. |
| [border-radius](https://www.tutorialrepublic.com/css-reference/css3-border-radius-property.php) | Defines the shape of the border corners of an element. |
| [border-right](https://www.tutorialrepublic.com/css-reference/css-border-right-property.php) | Sets the width, style, and color of the right border of an element. |
| [border-right-color](https://www.tutorialrepublic.com/css-reference/css-border-right-color-property.php) | Sets the color of the right border of an element. |
| [border-right-style](https://www.tutorialrepublic.com/css-reference/css-border-right-style-property.php) | Sets the style of the right border of an element. |
| [border-right-width](https://www.tutorialrepublic.com/css-reference/css-border-right-width-property.php) | Sets the width of the right border of an element. |
| [border-spacing](https://www.tutorialrepublic.com/css-reference/css-border-spacing-property.php) | Sets the spacing between the borders of adjacent table cells. |
| [border-style](https://www.tutorialrepublic.com/css-reference/css-border-style-property.php) | Sets the style of the border on all the four sides of an element. |
| [border-top](https://www.tutorialrepublic.com/css-reference/css-border-top-property.php) | Sets the width, style, and color of the top border of an element. |
| [border-top-color](https://www.tutorialrepublic.com/css-reference/css-border-top-color-property.php) | Sets the color of the top border of an element. |
| [border-top-left-radius](https://www.tutorialrepublic.com/css-reference/css3-border-top-left-radius-property.php) | Defines the shape of the top-left border corner of an element. |
| [border-top-right-radius](https://www.tutorialrepublic.com/css-reference/css3-border-top-right-radius-property.php) | Defines the shape of the top-right border corner of an element. |
| [border-top-style](https://www.tutorialrepublic.com/css-reference/css-border-top-style-property.php) | Sets the style of the top border of an element. |
| [border-top-width](https://www.tutorialrepublic.com/css-reference/css-border-top-width-property.php) | Sets the width of the top border of an element. |
| [border-width](https://www.tutorialrepublic.com/css-reference/css-border-width-property.php) | Sets the width of the border on all the four sides of an element. |

## List Properties:

|  |  |
| --- | --- |
| Property | Description |
| [list-style](https://www.tutorialrepublic.com/css-reference/css-list-style-property.php) | Defines the display style for a list and list elements. |
| [list-style-image](https://www.tutorialrepublic.com/css-reference/css-list-style-image-property.php) | Specifies the image to be used as a list-item marker. |
| [list-style-position](https://www.tutorialrepublic.com/css-reference/css-list-style-position-property.php) | Specifies the position of the list-item marker. |
| [list-style-type](https://www.tutorialrepublic.com/css-reference/css-list-style-type-property.php) | Specifies the marker style for a list-item. |

## Overflow Properties:

|  |  |
| --- | --- |
| Property | Description |
| [overflow](https://www.tutorialrepublic.com/css-reference/css-overflow-property.php) | Specifies the treatment of content that overflows the element's box. |
| [overflow-x](https://www.tutorialrepublic.com/css-reference/css3-overflow-x-property.php) | Specifies the treatment of content that overflows the element's box horizontally. |
| [overflow-y](https://www.tutorialrepublic.com/css-reference/css3-overflow-y-property.php) | Specifies the treatment of content that overflows the element's box vertically. |

## Flexbox Properties:

|  |  |
| --- | --- |
| Property | Description |
| [flex](https://www.tutorialrepublic.com/css-reference/css3-flex-property.php) | Specifies the components of a flexible length. |
| [flex-basis](https://www.tutorialrepublic.com/css-reference/css3-flex-basis-property.php) | Specifies the initial main size of the flex item. |
| [flex-direction](https://www.tutorialrepublic.com/css-reference/css3-flex-direction-property.php) | Specifies the direction of the flexible items. |
| [flex-flow](https://www.tutorialrepublic.com/css-reference/css3-flex-flow-property.php) | A shorthand property for the [flex-direction](https://www.tutorialrepublic.com/css-reference/css3-flex-direction-property.php) and the [flex-wrap](https://www.tutorialrepublic.com/css-reference/css3-flex-wrap-property.php) properties. |
| [flex-grow](https://www.tutorialrepublic.com/css-reference/css3-flex-grow-property.php) | Specifies how the flex item will grow relative to the other items inside the flex container. |
| [flex-shrink](https://www.tutorialrepublic.com/css-reference/css3-flex-shrink-property.php) | Specifies how the flex item will shrink relative to the other items inside the flex container. |
| [flex-wrap](https://www.tutorialrepublic.com/css-reference/css3-flex-wrap-property.php) | Specifies whether the flexible items should wrap or not. |
| [order](https://www.tutorialrepublic.com/css-reference/css3-order-property.php) | Specifies the order in which a flex items are displayed and laid out within a flex container. |

## Grid Properties:

|  |  |
| --- | --- |
| Property | Description |
| [column-gap](https://www.w3schools.com/cssref/css3_pr_column-gap.asp) | Specifies the gap between the columns |
| [gap](https://www.w3schools.com/cssref/css3_pr_gap.asp) | A shorthand property for the row-gap and the column-gap properties |
| [grid](https://www.w3schools.com/cssref/pr_grid.asp) | A shorthand property for the grid-template-rows, grid-template-columns, grid-template-areas, grid-auto-rows, grid-auto-columns, and the grid-auto-flow properties |
| [grid-area](https://www.w3schools.com/cssref/pr_grid-area.asp) | Either specifies a name for the grid item, or this property is a shorthand property for the grid-row-start, grid-column-start, grid-row-end, and grid-column-end properties |
| [grid-auto-columns](https://www.w3schools.com/cssref/pr_grid-auto-columns.asp) | Specifies a default column size |
| [grid-auto-flow](https://www.w3schools.com/cssref/pr_grid-auto-flow.asp) | Specifies how auto-placed items are inserted in the grid |
| [grid-auto-rows](https://www.w3schools.com/cssref/pr_grid-auto-rows.asp) | Specifies a default row size |
| [grid-column](https://www.w3schools.com/cssref/pr_grid-column.asp) | A shorthand property for the grid-column-start and the grid-column-end properties |
| [grid-column-end](https://www.w3schools.com/cssref/pr_grid-column-end.asp) | Specifies where to end the grid item |
| [grid-column-gap](https://www.w3schools.com/cssref/pr_grid-column-gap.asp) | Specifies the size of the gap between columns |
| [grid-column-start](https://www.w3schools.com/cssref/pr_grid-column-start.asp) | Specifies where to start the grid item |
| [grid-gap](https://www.w3schools.com/cssref/pr_grid-gap.asp) | A shorthand property for the grid-row-gap and grid-column-gap properties |
| [grid-row](https://www.w3schools.com/cssref/pr_grid-row.asp) | A shorthand property for the grid-row-start and the grid-row-end properties |
| [grid-row-end](https://www.w3schools.com/cssref/pr_grid-row-end.asp) | Specifies where to end the grid item |
| [grid-row-gap](https://www.w3schools.com/cssref/pr_grid-row-gap.asp) | Specifies the size of the gap between rows |
| [grid-row-start](https://www.w3schools.com/cssref/pr_grid-row-start.asp) | Specifies where to start the grid item |
| [grid-template](https://www.w3schools.com/cssref/pr_grid-template.asp) | A shorthand property for the grid-template-rows, grid-template-columns and grid-areas properties |
| [grid-template-areas](https://www.w3schools.com/cssref/pr_grid-template-areas.asp) | Specifies how to display columns and rows, using named grid items |
| [grid-template-columns](https://www.w3schools.com/cssref/pr_grid-template-columns.asp) | Specifies the size of the columns, and how many columns in a grid layout |
| [grid-template-rows](https://www.w3schools.com/cssref/pr_grid-template-rows.asp) | Specifies the size of the rows in a grid layout |
| [row-gap](https://www.w3schools.com/cssref/css3_pr_row-gap.asp) | Specifies the gap between the grid rows |

## Animation Properties:

|  |  |
| --- | --- |
| Property | Description |
| [animation](https://www.tutorialrepublic.com/css-reference/css3-animation-property.php) | Specifies the keyframe-based animations. |
| [animation-delay](https://www.tutorialrepublic.com/css-reference/css3-animation-delay-property.php) | Specifies when the animation will start. |
| [animation-direction](https://www.tutorialrepublic.com/css-reference/css3-animation-direction-property.php) | Specifies whether the animation should play in reverse on alternate cycles or not. |
| [animation-duration](https://www.tutorialrepublic.com/css-reference/css3-animation-duration-property.php) | Specifies the number of seconds or milliseconds an animation should take to complete one cycle. |
| [animation-fill-mode](https://www.tutorialrepublic.com/css-reference/css3-animation-fill-mode-property.php) | Specifies how a CSS animation should apply styles to its target before and after it is executing. |
| [animation-iteration-count](https://www.tutorialrepublic.com/css-reference/css3-animation-iteration-count-property.php) | Specifies the number of times an animation cycle should be played before stopping. |
| [animation-name](https://www.tutorialrepublic.com/css-reference/css3-animation-name-property.php) | Specifies the name of [@keyframes](https://www.tutorialrepublic.com/css-reference/css3-keyframes-rule.php) defined animations that should be applied to the selected element. |
| [animation-play-state](https://www.tutorialrepublic.com/css-reference/css3-animation-play-state-property.php) | Specifies whether the animation is running or paused. |
| [animation-timing-function](https://www.tutorialrepublic.com/css-reference/css3-animation-timing-function-property.php) | Specifies how a CSS animation should progress over the duration of each cycle. |

## Transition Properties:

|  |  |
| --- | --- |
| Property | Description |
| [transition](https://www.tutorialrepublic.com/css-reference/css3-transition-property.php) | Defines the transition between two states of an element. |
| [transition-delay](https://www.tutorialrepublic.com/css-reference/css3-transition-delay-property.php) | Specifies when the transition effect will start. |
| [transition-duration](https://www.tutorialrepublic.com/css-reference/css3-transition-duration-property.php) | Specifies the number of seconds or milliseconds a transition effect should take to complete. |
| [transition-property](https://www.tutorialrepublic.com/css-reference/css3-transition-property-property.php) | Specifies the names of the CSS properties to which a transition effect should be applied. |
| [transition-timing-function](https://www.tutorialrepublic.com/css-reference/css3-transition-timing-function-property.php) | Specifies the speed curve of the transition effect. |

## Transform Properties:

|  |  |
| --- | --- |
| Property | Description |
| [transform](https://www.tutorialrepublic.com/css-reference/css3-transform-property.php) | Applies a 2D or 3D transformation to an element. |
| [transform-origin](https://www.tutorialrepublic.com/css-reference/css3-transform-origin-property.php) | Defines the origin of transformation for an element. |
| [transform-style](https://www.tutorialrepublic.com/css-reference/css3-transform-style-property.php) | Specifies how nested elements are rendered in 3D space. |

## Other Properties:

|  |  |
| --- | --- |
| Property | Description |
| [cursor](https://www.tutorialrepublic.com/css-reference/css-cursor-property.php) | Specify the type of cursor. |
| [direction](https://www.tutorialrepublic.com/css-reference/css-direction-property.php) | Define the text direction/writing direction. |
| [display](https://www.tutorialrepublic.com/css-reference/css-display-property.php) | Specifies how an element is displayed onscreen. |
| [position](https://www.tutorialrepublic.com/css-reference/css-position-property.php) | Specifies how an element is positioned. |
| [opacity](https://www.tutorialrepublic.com/css-reference/css3-opacity-property.php) | Specifies the transparency of an element. |
| [float](https://www.tutorialrepublic.com/css-reference/css-float-property.php) | Specifies whether or not a box should float. |
| [height](https://www.tutorialrepublic.com/css-reference/css-height-property.php) | Specify the height of an element. |
| [width](https://www.tutorialrepublic.com/css-reference/css-width-property.php) | Specify the width of an element. |
| [left](https://www.tutorialrepublic.com/css-reference/css-left-property.php) | Specify the location of the left edge of the positioned element. |
| [right](https://www.tutorialrepublic.com/css-reference/css-right-property.php) | Specify the location of the right edge of the positioned element. |
| [top](https://www.tutorialrepublic.com/css-reference/css-top-property.php) | Specify the location of the top edge of the positioned element. |
| [bottom](https://www.tutorialrepublic.com/css-reference/css-bottom-property.php) | Specify the location of the bottom edge of the positioned element. |
| [z-index](https://www.tutorialrepublic.com/css-reference/css-z-index-property.php) | Specifies a layering or stacking order for positioned elements. |
| [max-height](https://www.tutorialrepublic.com/css-reference/css-max-height-property.php) | Specify the maximum height of an element. |
| [max-width](https://www.tutorialrepublic.com/css-reference/css-max-width-property.php) | Specify the maximum width of an element. |
| [min-height](https://www.tutorialrepublic.com/css-reference/css-min-height-property.php) | Specify the minimum height of an element. |
| [min-width](https://www.tutorialrepublic.com/css-reference/css-min-width-property.php) | Specify the minimum width of an element. |
| [word-break](https://www.tutorialrepublic.com/css-reference/css3-word-break-property.php) | Specifies how to break lines within words. |
| [word-spacing](https://www.tutorialrepublic.com/css-reference/css-word-spacing-property.php) | Sets the spacing between words. |
| [word-wrap](https://www.tutorialrepublic.com/css-reference/css3-word-wrap-property.php) | Specifies whether to break words when the content overflows the boundaries of its container. |

1. Basic of Bootstrap

# Introduction:

* Bootstrap is the most popular HTML, CSS, and JavaScript framework for developing responsive, mobile-first websites.
* Bootstrap is completely free to download and use!
* It provides faster and easier web development.
* It contains HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plugins.
* It also gives you the ability to easily create responsive designs.
* Bootstrap was developed by Mark Otto and Jacob Thornton at Twitter, and released as an open source product in August 2011 on GitHub.
* Bootstrap 5 is a latest version of Bootstrap.
* Until Bootstrap 4, it use jQuery with Javascript.
* But in Bootstrap 5, it removed jQuery and it totally use vanilla Javascript for DOM manipulation.

# How to Use Bootstrap

* We have two different way to use Bootstrap,
  + Install Bootstrap
  + Using CDN
* We should include four file in HTML file for use Bootstrap.
  + bootstrap.css or bootstrap.min.css
  + bootstrap.js or bootstrap.min.js
  + popper.js or popper.min.js
  + jQuery (if use Bootstrap 5 then doesn’t need to include)
* jQuery must include above the all JS files.
* We can also include the bundle.js or it’s minified file.
* bundle.js is a combination of bootstrap.js and popper.js file, So these file don’t need to include when we include bundle.js or it’s minified file.

# Structure of Bootstrap

* A web page using twitter bootstrap has a basic html structure which should start from type of document declaration, navigation and container in body, adding style sheets, java scripts declaration and Meta tags in header.
* For uniform rendering of its components and controls across all browsers, a HTML 5 doctype is needed by twitter bootstrap.
* This is due to the fact that HTML 5 elements are understood by the bootstrap which makes it necessary to include an appropriate doctype tag to the web page.

## HTML Header Structure:

* In order for a responsive working of the bootstrap with mobile devices and latest browsers, the minimum requirements in the html header part should be to include a bootstrap style sheet and declare a viewport meta tag, which is crucial for a mobile device and bootstrap to be able to respond in accordance with the zoom level and the width of display.
* Adjusting viewport’s width enables browsers to appropriately respond for the display size.
* During the earlier versions of bootstrap, a separate style sheet referred to as bootstrap-responsive existed.
* But latter, an inbuilt responsive in nature base style sheet known as bootstrap.css which is a single style sheet that handles web page responsiveness was developed.
* This bootstrap.css proves to be very useful for debugging during development.

## HTML Body Structure:

* Before you close the body tag in the html body, the bootstrap java script file and jQuery file must be included.
* The bootstrap java script file should be rendered after the jQuery file.

## Layout:

**Fluid layout:**

* No extra style sheet or additional step is required to obtain a responsive fluid layout in the bootstrap.
* It is designed to implement the responsive layout by default.
* This layout is highly recommended for public websites.

**Fixed layout:**

* Creating a website with a fixed layout is not generally recommended.
* This is mainly due to the increasing amount of internet website traffic that is arising from small devices such as smart phones and tablets.
* A fixed layout is however needed in some cases such as intranet applications.
* If this is the case, steps followed to disable the fluid responsiveness of the layout like,
  + Removing viewport mega tag
  + Override width property to a fixed size
  + If you are using NavBar, get rid of expanding and collapsing behavior
  + Instead of .col-md-\* and .col-lg-\*., use col-xs-\* class in grid layouts

## Components:

**Menu section:**

* In bootstrap, to design a menu is the easiest thing to do in the web designing world.
* This is because it is designed to be responsive by default and even has the ability to get new appearances in smaller devices.

**Header area:**

* Jumbotron, which can display large headers and contents is a highly usable class offered by bootstrap.
* It is largely utilized by product based websites.

**The content area:**

* This content should be divided into three equal parts placed side by side.
* With the bootstrap’s flex box-based grid, doing this is one easy and fast task.
* A12-column grid system is provided, so dividing the screen into equal parts and all one needs to do is to specify which HTML is occupied by each part.

**Footer area:**

* Footer area uses the same principle as the content area.

1. Basic of JavaScript

# Introduction

* JavaScript (JS) is a lightweight, interpreted(implementations execute instructions directly without earlier compiling a program into machine language), or just-in-time(run time compilation) compiled programming language with first-class functions(function treat like variable).
* is a prototype-based(classes are not explicitly defined), multi-paradigm, single-threaded(run on main thread), dynamic language(interpreter assigns variables a type at runtime based on the variable's value).
* It is used for client-side scripting as well as server side using Node.js.

# Use of JavaScript

* JavaScript helps the users to build modern web applications to interact directly without reloading the page every time.
* JavaScript is commonly used to dynamically modify HTML and CSS to update a user interface by the DOM API.
* It is mainly used in web applications.
  + Web Development
  + Mobile Development
  + Game Development
  + Presentation

# Way to use JavaScript

* We should use JavaScript two different way,
  + Internal scripting
  + External scripting

## Internal Scripting:

* We wrap script inside <script>.
* We use <script> inside <head> or <body>.

## External Scripting:

* We write script in separate JavaScript file, which have “.js” extension.
* We use <script> inside <head> or <body>.

# Syntax of JavaScript

**Basic Syntax:**

* JavaScript is case-sensitive.
* Statements should end in a semicolon (;), but it is not compulsory.

**Variable Syntax:**

* Must be defined before being used.
* The variable name can contain A – Z, a – z, underscore or digits and must start with a letter or an underscore (“\_”).
* The data type does not have to be explicitly defined.

**Comment Syntax:**

* For single line comment we use “//”.
* For multiline comment we use “\*/…/\*”

# Events in JavaScript

* Events are very important part of DOM.
* Using JavaScript we should manipulate element of DOM based on various events.
* We have four types of events in JavaScript,
  + Mouse events
  + Keyboard events
  + Frame or Object events
  + Form events

## Mouse events:

* These events base on mouse.
* Here we should manipulate DOM on mouse events.

|  |  |  |
| --- | --- | --- |
| Event | Attribute | Description |
| click | onclick | The event occurs when the user clicks on an element |
| dblclick | ondblclick | The event occurs when the user double-clicks on an element |
| mousedown | onmousedown | The event occurs when a user presses a mouse button over an element |
| mousemove | onmousemove | The event occurs when a user moves the mouse pointer over an element |
| mouseover | onmouseover | The event occurs when a user mouse over an element |
| mouseout | onmouseout | The event occurs when a user moves the mouse pointer out of an element |
| mouseup | onmouseup | The event occurs when a user releases a mouse button over an element |

## Keyboard Events:

* These events based on keyboard.
* Here, we should manipulate DOM based on keyboard event.

|  |  |  |
| --- | --- | --- |
| Event | Attribute | Description |
| keydown | onkeydown | The event occurs when the user is pressing a key or holding down a key |
| keypress | onkeypress | The event occurs when the user is pressing a key or holding down a key |
| keyup | onkeyup | The event occurs when a keyboard key is released |

## Frame or Object Events:

* These events based on object.
* Here, we should manipulate DOM based on object event.

|  |  |  |
| --- | --- | --- |
| Event | Attribute | Description |
| abort | onabort | The event occurs when an image is stopped from loading before completely loaded (for <object>) |
| error | onerror | The event occurs when an image does not load properly (for <object>, <body> and <frameset>) |
| load | onload | The event occurs when a document, frameset, or <object> has been loaded |
| resize | onresize | The event occurs when a document view is resized |
| scroll | onscroll | The event occurs when a document view is scrolled |

## Form Events:

* These events based on form.
* Here, we should manipulate DOM based on form event.

|  |  |  |
| --- | --- | --- |
| Event | Attribute | Description |
| blur | onblur | The event occurs when a form element loses focus |
| change | onchange | The event occurs when the content of a form element, the selection, or the checked state have changed (for <input>, <select>, and <textarea>) |
| focus | onfocus | The event occurs when an element gets focus (for <label>, <input>, <select>, textarea>, and <button>) |
| reset | onreset | The event occurs when a form is reset |
| select | onselect | The event occurs when a user selects some  text (for <input> and <textarea>) |
| submit | onsubmit | The event occurs when a form is submitted |

# Eventlistener

* Event listener is a method which used to attached event handler with particular element.
* We can attach many event handler on a element.
* Here, we are attach event handler with the DOM object not HTML element.
* It provide more reliability and flexibility, because we are not deal with HTML markup, we can directly control any element without HTML markup control.
* We have two method in event listener.

**Syntax:**

element.addEventListener(event, *function*, useCapture); *// Add event listener*

element.removeEventListener(); *// remove elevent listener*

* “event” parameter is use to specify event.
* "function” parameter is used to specify function that execute when event triggered.
* “useCapture” is optional parameter which by default is false.
* “useCapture” is used for specify propagation mode.
* We have two type of propagation,
  + Bubbling
  + Capturing
* Bubbling is use when we want that child element handle event before parent element when event of child element triggered.
* Let say, we have one <div> and inside that we have <p>, now we bind event listener on both element for same event, So when event of **<p>** triggered at that time <p> handle event first and after <div> handle event second.
* Bubbling is use when we want that parent element handle event before child element when event of child element triggered.
* Let say, we have one <div> and inside that we have <p>, now we bind event listener on both element for same event, So when event of **<p>** triggered at that time <div> handle event first and after <p> handle event second.

# Validation

* It is a technique that used to ensure that user fill data correctly.
* In traditional way validation, we perform server-side validation.
* If the data entered by a client was incorrect or was simply missing, the server would have to send all the data back to the client and request that the form be resubmitted with correct information.
* This was really a lengthy process which used to put a lot of burden on the server.
* JavaScript provides a way to validate form's data on the client's computer before sending it to the web server.
* We have two type of validation which we should do for archive better validation,
  + Basic Validation
    - Emptiness
    - Confirm Password
    - Length Validation…
  + Data Format Validation
    - Email Validation
    - Mobile Number Validation
    - Enrollment Number Validation…
* We are use regular expression for validate data.

## RegExp:

* A regular expression is an object that describes a pattern of characters.
* Regular expressions are used to perform pattern-matching and "search-and-replace" functions on text.

**Syntax:**

*var* pattern = "^[\\w]+$";   *// will allow only words in the string*

*var* regex = new RegExp(pattern);

*if* (regex.test(testString)){

*//Valid*

} *else* {

*//Invalid*

}

**Regular Expression Syntax:**

* To find word characters in the string we can use \w
* We can also use [a-zA-Z0-9\_] for the same
* To find non-word characters in the string we can use \W
* to find digit characters in the string we can use \d
* We can also use [0-9] for the same
* To find non-digit characters in the string we can use \D
* We can use \n for new line and \t for tab