***CSS***

CSS is the language we use to style a Web page.

**What is CSS?**

CSS stands for Cascading Style Sheets

CSS describes how HTML elements are to be displayed on screen, paper, or in other media

CSS saves a lot of work. It can control the layout of multiple web pages all at once

External stylesheets are stored in CSS files

CSS is the language we use to style a Web page.

# **What is CSS**

CSS stands for Cascading Style Sheets. It is a style sheet language which is used to describe the look and formatting of a document written in markup language. It provides an additional feature to HTML. It is generally used with HTML to change the style of web pages and user interfaces. It can also be used with any kind of XML documents including plain XML, SVG and XUL.

CSS is used along with HTML and JavaScript in most websites to create user interfaces for web applications and user interfaces for many mobile applications.

## What does CSS do

You can add new looks to your old HTML documents.

You can completely change the look of your website with only a few changes in CSS code.

## Why use CSS

These are the three major benefits of CSS:

## Solves a big problem

Before CSS, tags like font, color, background style, element alignments, border and size had to be repeated on every web page. This was a very long process. For example: If you are developing a large website where fonts and color information are added on every single page, it will be become a long and expensive process. CSS was created to solve this problem. It was a W3C recommendation.

## 2) Saves a lot of time

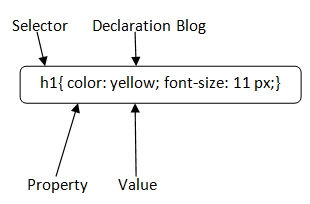
CSS style definitions are saved in external CSS files so it is possible to change the entire website by changing just one file.

## 3) Provide more attributes

CSS provides more detailed attributes than plain HTML to define the look and feel of the website.

# **CSS Syntax**

A CSS rule set contains a selector and a declaration block.



**Selector:** Selector indicates the HTML element you want to style. It could be any tag like <h1>, <title> etc.

**Declaration Block:** The declaration block can contain one or more declarations separated by a semicolon. For the above example, there are two declarations:

1. color: yellow;
2. font-size: 11 px;

Each declaration contains a property name and value, separated by a colon.

**Property:** A Property is a type of attribute of HTML element. It could be color, border etc.

**Value:** Values are assigned to CSS properties. In the above example, value "yellow" is assigned to color property.

Selector{Property1: value1; Property2: value2; ..........;}

# **CSS Selector**

**CSS selectors** are used to select the content you want to style. Selectors are the part of CSS rule set. CSS selectors select HTML elements according to its id, class, type, attribute etc.

There are several different types of selectors in CSS.

1. CSS Element Selector
2. CSS Id Selector
3. CSS Class Selector
4. CSS Universal Selector
5. CSS Group Selector

## CSS Element Selector

The element selector selects the HTML element by name.

<!DOCTYPE html**>**

**<html>**

**<head>**

**<style>**

p{

    text-align: center;

    color: blue;

}

**</style>**

**</head>**

**<body>**

**<p>**This style will be applied on every paragraph.**</p>**

**<p** id="para1"**>**Me too!**</p>**

**<p>**And me!**</p>**

**</body>**

**</html>**

**Output:**

This style will be applied on every paragraph.

Me too!

And me!

## 2) CSS Id Selector

The id selector selects the id attribute of an HTML element to select a specific element. An id is always unique within the page so it is chosen to select a single, unique element.

It is written with the hash character (#), followed by the id of the element.

Let?s take an example with the id "para1".

<!DOCTYPE html**>**

**<html>**

**<head>**

**<style>**

#para1 {

    text-align: center;

    color: blue;

}

**</style>**

**</head>**

**<body>**

**<p** id="para1"**>**Hello Javatpoint.com**</p>**

**<p>**This paragraph will not be affected.**</p>**

**</body>**

**</html>**

**Output:**

Hello Javatpoint.com

This paragraph will not be affected.

## 3) CSS Class Selector

The class selector selects HTML elements with a specific class attribute. It is used with a period character . (full stop symbol) followed by the class name.

#### **Note: A class name should not be started with a number.**

Let's take an example with a class "center".

<!DOCTYPE html**>**

**<html>**

**<head>**

**<style>**

.center {

    text-align: center;

    color: blue;

}

**</style>**

**</head>**

**<body>**

**<h1** class="center"**>**This heading is blue and center-aligned.**</h1>**

**<p** class="center"**>**This paragraph is blue and center-aligned.**</p>**

**</body>**

**</html>**

**Output:**

## This heading is blue and center-aligned.

This paragraph is blue and center-aligned.

## CSS Class Selector for specific element

If you want to specify that only one specific HTML element should be affected then you should use the element name with class selector.

Let's see an example.

<!DOCTYPE html**>**

**<html>**

**<head>**

**<style>**

p.center {

    text-align: center;

    color: blue;

}

**</style>**

**</head>**

**<body>**

**<h1** class="center"**>**This heading is not affected**</h1>**

**<p** class="center"**>**This paragraph is blue and center-aligned.**</p>**

**</body>**

**</html>**

**OUTPUT**

# This heading is not affected

This paragraph is blue and center-aligned.

## 4) CSS Universal Selector

The universal selector is used as a wildcard character. It selects all the elements on the pages.

<!DOCTYPE html**>**

**<html>**

**<head>**

**<style>**

\* {

   color: green;

   font-size: 20px;

}

**</style>**

**</head>**

**<body>**

**<h2>**This is heading**</h2>**

**<p>**This style will be applied on every paragraph.**</p>**

**<p** id="para1"**>**Me too!**</p>**

**<p>**And me!**</p>**

**</body>**

**</html>**

Output:

## This is heading

This style will be applied on every paragraph.

Me too!

And me!

## 5) CSS Group Selector

The grouping selector is used to select all the elements with the same style definitions.

Grouping selector is used to minimize the code. Commas are used to separate each selector in grouping.

Let's see the CSS code without group selector.

h1 {

    text-align: center;

    color: blue;

}

h2 {

    text-align: center;

    color: blue;

}

p {

    text-align: center;

    color: blue;

}

As you can see, you need to define CSS properties for all the elements. It can be grouped in following ways:

h1,h2,p {

    text-align: center;

    color: blue;

}

<!DOCTYPE html**>**

**<html>**

**<head>**

**<style>**

h1, h2, p {

    text-align: center;

    color: blue;

}

**</style>**

**</head>**

**<body>**

**<h1>**Hello Javatpoint.com**</h1>**

**<h2>**Hello Javatpoint.com (In smaller font)**</h2>**

**<p>**This is a paragraph.**</p>**

**</body>**

**</html>**

**Output:**

## Hello Javatpoint.com

### Hello Javatpoint.com (In smaller font)

This is a paragraph.

# **How to add CSS**

CSS is added to HTML pages to format the document according to information in the style sheet. There are three ways to insert CSS in HTML documents.

Inline CSS

Internal CSS

External CSS

## Inline CSS

Inline CSS is used to apply CSS on a single line or element.

For example:

**<p** style="color:blue"**>**Hello CSS**</p>**

For more visit here: [Inline CSS](https://www.javatpoint.com/inline-css)

**2) Internal CSS**

Internal CSS is used to apply CSS on a single document or page. It can affect all the elements of the page. It is written inside the style tag within head section of html.

For example:

**<style>**

p{color:blue}

**</style>**

## 3) External CSS

External CSS is used to apply CSS on multiple pages or all pages. Here, we write all the CSS code in a css file. Its extension must be .css for example style.css.

For example:

p{color:blue}

You need to link this style.css file to your html pages like this:

**<link** rel="stylesheet" type="text/css" href="style.css"**>**

The link tag must be used inside head section of html.

# **Inline CSS**

We can apply CSS in a single element by inline CSS technique.

The inline CSS is also a method to insert style sheets in HTML document. This method mitigates some advantages of style sheets so it is advised to use this method sparingly.

If you want to use inline CSS, you should use the style attribute to the relevant tag.

Syntax:

**<htmltag** style="cssproperty1:value; cssproperty2:value;"**>** **</htmltag>**

Example:

**<h2** style="color:red;margin-left:40px;"**>**Inline CSS is applied on this heading.**</h2>**

**<p>**This paragraph is not affected.**</p>**

**Output:**

## Inline CSS is applied on this heading.

This paragraph is not affected.

## Disadvantages of Inline CSS

You cannot use quotations within inline CSS. If you use quotations the browser will interpret this as an end of your style value.

These styles cannot be reused anywhere else.

These styles are tough to be edited because they are not stored at a single place.

It is not possible to style pseudo-codes and pseudo-classes with inline CSS.

Inline CSS does not provide browser cache advantages.

# **Internal CSS**

The internal style sheet is used to add a unique style for a single document. It is defined in <head> section of the HTML page inside the <style> tag.

Example:

<!DOCTYPE html**>**

**<html>**

**<head>**

**<style>**

body {

    background-color: linen;

}

h1 {

    color: red;

    margin-left: 80px;

}

**</style>**

**</head>**

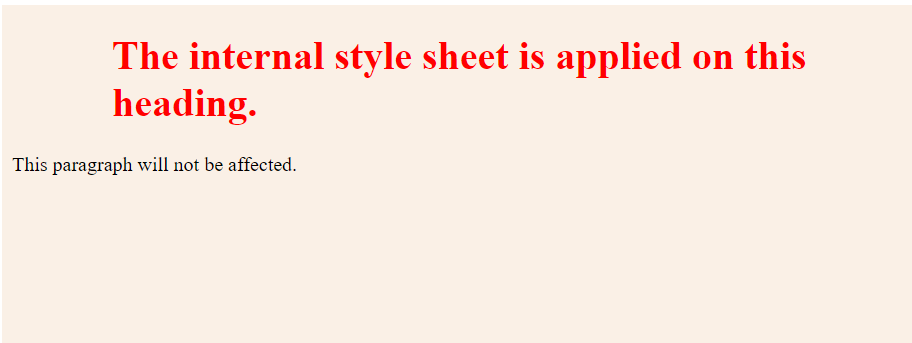
**<body>**

**<h1>**The internal style sheet is applied on this heading.**</h1>**

**<p>**This paragraph will not be affected.**</p>**

**</body>**

**</html>**



# **External CSS**

The external style sheet is generally used when you want to make changes on multiple pages. It is ideal for this condition because it facilitates you to change the look of the entire web site by changing just one file.

It uses the <link> tag on every pages and the <link> tag should be put inside the head section.

Example:

**<head>**

**<link** rel="stylesheet" type="text/css" href="mystyle.css"**>**

**</head>**

The external style sheet may be written in any text editor but must be saved with a .css extension. This file should not contain HTML elements.

Let's take an example of a style sheet file named "mystyle.css".

***File: mystyle.css***

body {

    background-color: lightblue;

}

h1 {

    color: navy;

    margin-left: 20px;

}

Note: You should not use a space between the property value and the unit. For example: It should be margin-left:20px not margin-left:20 px.

# **CSS Comments**

CSS comments are generally written to explain your code. It is very helpful for the users who reads your code so that they can easily understand the code.

Comments are ignored by browsers.

Comments are single or multiple lines statement and written within /\*............\*/ .

<!DOCTYPE html**>**

**<html>**

**<head>**

**<style>**

p {

    color: blue;

    /\* This is a single-line comment \*/

    text-align: center;

}

/\* This is

a multi-line

comment \*/

**</style>**

**</head>**

**<body>**

**<p>**Hello Javatpoint.com**</p>**

**<p>**This statement is styled with CSS.**</p>**

**<p>**CSS comments are ignored by the browsers and not shown in the output.**</p>**

**</body>**

**</html>**

Output:

Hello Javatpoint.com

This statement is styled with CSS.

CSS comments are ignored by the browsers and not shown in the output.

**NOTE:**

**Difference between id and class attribute:** The only difference between them is that “id” is unique in a page and can only apply to at most one element, while “class” selector can apply to multiple elements. 

## HTML Links - Hyperlinks

HTML links are hyperlinks.

You can click on a link and jump to another document.

When you move the mouse over a link, the mouse arrow will turn into a little hand.

**Note:** A link does not have to be text. A link can be an image or any other HTML element!

## HTML Links - Syntax

The HTML <a> tag defines a hyperlink. It has the following syntax:

<a href="*url*">*link text*</a>

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

The link text is the part that will be visible to the reader.

Clicking on the link text, will send the reader to the specified URL address.

<a href="https://www.w3schools.com/">Visit W3Schools.com!</a>

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

**Tip:** Links can of course be styled with CSS, to get another look!

HTML Links - The target Attribute

By default, the linked page will be displayed in the current browser window. To change this, you must specify another target for the link.

The target attribute specifies where to open the linked document.

The target attribute can have one of the following values:

* \_self - Default. Opens the document in the same window/tab as it was clicked
* \_blank - Opens the document in a new window or tab
* \_parent - Opens the document in the parent frame
* \_top - Opens the document in the full body of the window

### **Example**

Use target="\_blank" to open the linked document in a new browser window or tab:

<a href="https://www.w3schools.com/" target="\_blank">Visit W3Schools!</a>

## Absolute URLs vs. Relative URLs

Both examples above are using an **absolute URL** (a full web address) in the href attribute.

A local link (a link to a page within the same website) is specified with a **relative URL** (without the "https://www" part):

### **Example**

<h2>Absolute URLs</h2>  
<p><a href="https://www.w3.org/">W3C</a></p>  
<p><a href="https://www.google.com/">Google</a></p>  
  
<h2>Relative URLs</h2>  
<p><a href="html\_images.asp">HTML Images</a></p>  
<p><a href="/css/default.asp">CSS Tutorial</a></p>

CSS Background

CSS background property is used to define the background effects on element. There are 5 CSS background properties that affects the HTML elements:

background-color

background-image

background-repeat

background-attachment

background-position

|  |  |  |
| --- | --- | --- |
| **Value** | **Description** |  |
| Repeat | The background image is repeated both vertically and horizontally.  The last image will be clipped if it does not fit. This is default |
| repeat-x | The background image is repeated only horizontally |
| repeat-y | The background image is repeated only vertically |
| no-repeat | The background-image is not repeated. The image will only be shown once |

### **Property Values**

**scroll:** It is the default value that prevents the element from scrolling with the contents, but scrolls with the page.

**fixed:** Using this value, the background image doesn't move with the element, even the element has a scrolling mechanism. It causes the image to be locked inone place, even the rest of the document scrolls.

**local:** Using this value, if the element has a scrolling mechanism, the background image scrolls with the content of the element.

**initial:** It sets the property to its default value.

# CSS Font

CSS Font property is used to control the look of texts. By the use of CSS font property you can change the text size, color, style and more. You have already studied how to make text bold or underlined. Here, you will also know how to resize your font using percentage.

These are some important font attributes:

1. **CSS Font color**: This property is used to change the color of the text. (standalone attribute)
2. **CSS Font family**: This property is used to change the face of the font.
3. **CSS Font size**: This property is used to increase or decrease the size of the font.
4. **CSS Font style**: This property is used to make the font bold, italic or oblique.
5. **CSS Font variant**: This property creates a small-caps effect.
6. **CSS Font weight**: This property is used to increase or decrease the boldness and lightness of the font.

CSS font family can be divided in two types:

* Generic family: It includes Serif, Sans-serif, and Monospace.
* Font family: It specifies the font family name like Arial, New Times Roman etc.

**Serif**: Serif fonts include small lines at the end of characters. Example of serif: Times new roman, Georgia etc.

**Sans-serif**: A sans-serif font doesn't include the small lines at the end of characters. Example of Sans-serif: Arial, Verdana etc.

**monospaced font**, also called a **fixed-pitch**, **fixed-width**, or **non-proportional font**, is a [font](https://en.wikipedia.org/wiki/Font) whose letters and characters each occupy the same amount of horizontal space. This contrasts with [variable-width fonts](https://en.wikipedia.org/wiki/Typeface#Proportion), where the letters and spacings have different widths.

Oblique type is **a form of type that slants slightly to the right, used for the same purposes as italic type**.

|  |  |
| --- | --- |
| **Font Size Value** | **Description** |
| xx-small | used to display the extremely small text size. |
| x-small | used to display the extra small text size. |
| small | used to display small text size. |
| medium | used to display medium text size. |
| large | used to display large text size. |
| x-large | used to display extra large text size. |
| xx-large | used to display extremely large text size. |
| smaller | used to display comparatively smaller text size. |
| larger | used to display comparatively larger text size. |
| size in pixels or % | used to set value in percentage or in pixels. |

# CSS Border

The CSS border is a shorthand property used to set the border on an element.

The [CSS](https://www.javatpoint.com/css-tutorial) border properties are use to specify the style, color and size of the border of an element. The CSS border properties are given below

* border-style
* border-color
* border-width

## CSS Margins

The CSS margin properties are used to create space around elements, outside of any defined borders.

With CSS, you have full control over the margins. There are properties for setting the margin for each side of an element (top, right, bottom, and left).

All the margin properties can have the following values:

* auto - the browser calculates the margin
* *length* - specifies a margin in px, pt, cm, etc.
* *%* - specifies a margin in % of the width of the containing element

CSS Padding

Padding is used to create space around an element's content, inside of any defined borders.

## Different List Item Markers

The list-style-type property specifies the type of list item marker.

ul.a {  
  list-style-type: circle;  
}  
  
ul.b {  
  list-style-type: square;  
}  
  
ol.c {  
  list-style-type: upper-roman;  
}  
  
ol.d {  
  list-style-type: lower-alpha;  
}

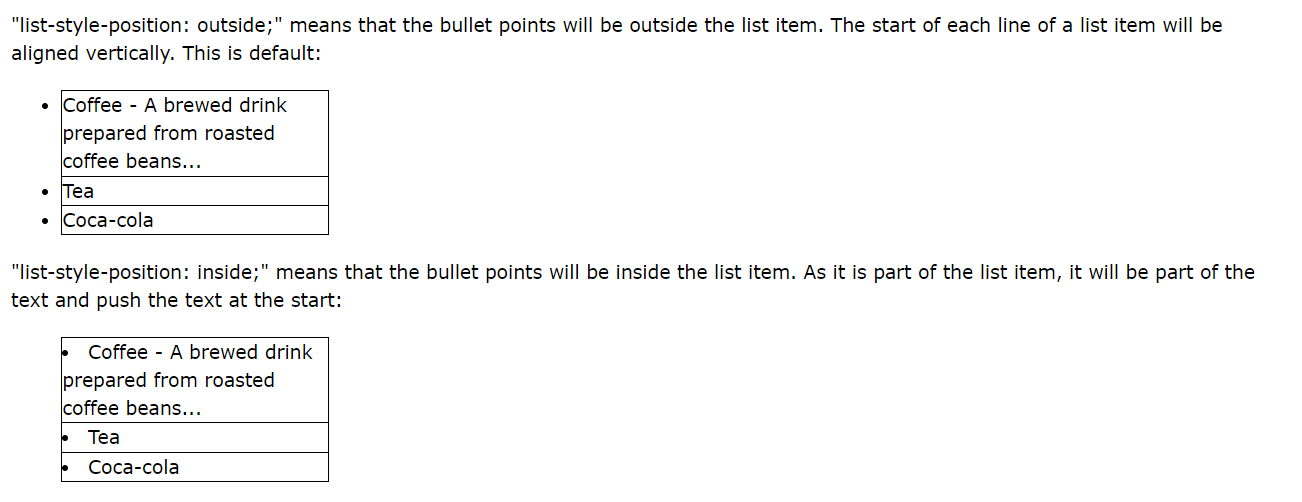
## An Image as The List Item Marker

The list-style-image property specifies an image as the list item marker:

ul {  
  list-style-image: url('sqpurple.gif');  
}

## position The List Item Markers

The list-style-position property specifies the position of the list-item markers (bullet points).



ul.a {  
  list-style-position: outside;  
}

ul.b {  
  list-style-position: inside;  
}

# CSS Text

body {  
  color: blue;  
}  
  
h1 {  
  color: green;  
}

h1 {  
  text-align: center;  
}  
  
h2 {  
  text-align: left;  
}  
  
h3 {  
  text-align: right;  
}

h1 {

text-decoration: overline;

}

h2 {

text-decoration: line-through;

}

h3 {

text-decoration: underline;

}

p.ex {

text-decoration: overline underline;

}

p{

text-transform: uppercase;

text-transform: lowercase;

text-transform: capitalize;}

# CSS Text Spacing

|  |  |
| --- | --- |
| **Property** | **Description** |
| [letter-spacing](https://www.w3schools.com/cssref/pr_text_letter-spacing.asp) | Specifies the space between characters in a text |
| [line-height](https://www.w3schools.com/cssref/pr_dim_line-height.asp) | Specifies the line height |
| [text-indent](https://www.w3schools.com/cssref/pr_text_text-indent.asp) | Specifies the indentation of the first line in a text-block |
| [white-space](https://www.w3schools.com/cssref/pr_text_white-space.asp) | Specifies how to handle white-space inside an element |
| [word-spacing](https://www.w3schools.com/cssref/pr_text_word-spacing.asp) | Specifies the space between words in a text |

The text-indent property is used to specify the indentation of the first line of a text:

p {  
  text-indent: 50px;  
}

## Letter Spacing

The letter-spacing property is used to specify the space between the characters in a text.

h1 {  
  letter-spacing: 5px;  
}  
h2 {  
  letter-spacing: -2px;  
}

## Line Height

The line-height property is used to specify the space between lines:

p.small {  
  line-height: 0.8;  
}  
  
p.big {  
  line-height: 1.8;  
}

## Word Spacing

The word-spacing property is used to specify the space between the words in a text.

p.one {  
  word-spacing: 10px;  
}  
  
p.two {  
  word-spacing: -2px;  
}

## White Space

The white-space property specifies how white-space inside an element is handled.

p {  
  white-space: nowrap;  
}

**Differences Between <datalist> and <select> tag**

|  |  |
| --- | --- |
| **<select> tag** | **<datalist> tag** |
| The user can choose only one option from the given list. | The user can choose any option from the given list but can also use its own input. |
| This tag is a form input type. | This tag is not a form input type. |
| The user has to scan a long list so as to select an option. | The user can easily input the option and get the hints and then can be chosen by the user. |
| The user can be restricted to a list of options. | The user is not restricted by the list of options. |
| It doesn’t provide the auto-complete feature. | It provides the auto-complete feature. |

# **HTML Div Tag**

The **HTML <div> tag** is used *to group the large section of HTML elements together*.

We know that every tag has a specific purpose e.g. p tag is used to specify paragraph, <h1> to <h6> tag are used to specify headings but the <div> tag is just like a container unit which is used to encapsulate other page elements and divides the HTML documents into sections.

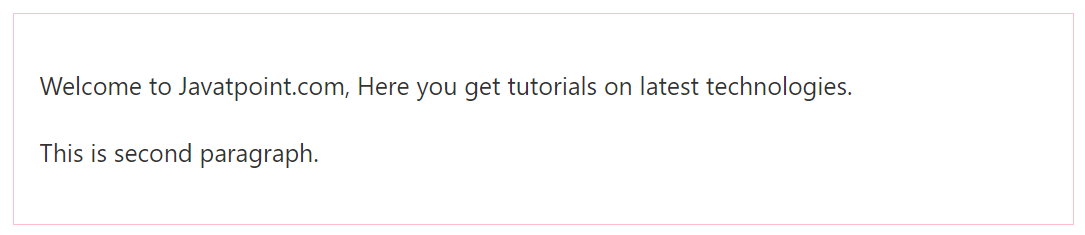
The div tag is generally used by web developers to group HTML elements together and apply CSS styles to many elements at once. For example: If you wrap a set of paragraph elements into a div element so you can take the advantage of CSS styles and apply font style to all paragraphs at once instead of coding the same style for each paragraph element.

**<div** style="border:1px solid pink;padding:20px;font-size:20px"**>**

**<p>**Welcome to Javatpoint.com, Here you get tutorials on latest technologies.**</p>**

**<p>**This is second paragraph**</p>**

**</div>**



## Difference between HTML div tag and span tag

|  |  |
| --- | --- |
| **div tag** | **span tag** |
| HTML div is a **block** element. | HTML span is an **inline** element |
| HTML div element is used to **wrap large sections of elements**. | HTML span element is used to **wrap small portion of texts, image** etc. |

# **HTML <span> tag**

HTML <span> tag is used as a generic container of inline elements. It is used for styling purpose to the grouped inline elements (using class and id attribute or inline style).

The <span> tag does not have any default meaning or rendering.

The <span> tag can be useful for the following task:

* To change the language of a part of the text.
* To change the color, font, background of a part of text using CSS
* To apply the scripts to the particular part of the text.

#### **Note: HTML <span> is much similar as <div> tag, but <div> is used for block-level elements and <span> tag is used for inline elements.**

### **Syntax**

**<span>**Write your content here......**</span>**

### **Example**

<!DOCTYPE html**>**

**<html>**

**<head>**

**<title>**Span Tag**</title>**

**</head>**

**<body>**

**<h2>**Example of span tag**</h2>**

**<p>**I have choosen only

**<span** style="color: red;"**>**red**</span>**,

**<span** style="color: blue;"**>**blue**</span>**, and

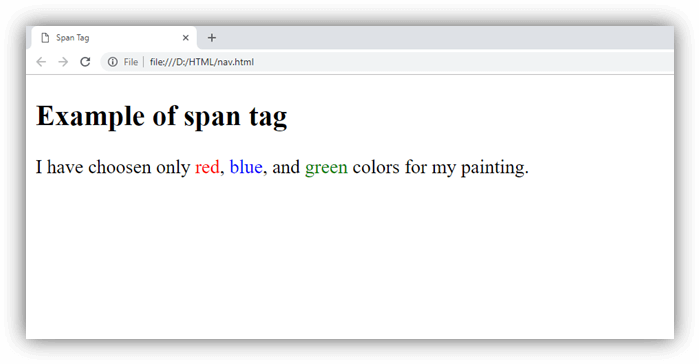
**<span** style="color: green;"**>**green**</span>** colors for my painting.

**</p>**

**</body>**

**</html>**

**Output:**



# **HTML Section Tag**

The HTML <section> tag is used to define sections in a document. When you put your content on a web page, it may contains many chapters, headers, footers, or other sections on a web page that is why HTML <section> tag is used.

HTML <section> is a new tag introduced in HTML5.

## HTML section tag example

**CSS code:**

section{

border:1px solid pink;

padding:15px;

margin:10px;

}

**HTML code:**

**<h2>** Indian Leader**</h2>**

**<section>**

**<h3>** Jawaharlal Nehru **</h3>**

**<p>** Jawaharlal Nehru was the first Prime Minister of India and a central figure in

Indian politics for much of the 20th century. He emerged as the paramount leader of the Indian

independence movement under the tutelage of Mahatma Gandhi. -Source Wikipedia **</p>**

**</section>**

**<section>**

**<h3>**Subhas Chandra Bose **</h3>**

**<p>**Subhas Chandra Bose was an Indian nationalist whose attempt during World War II to rid India of

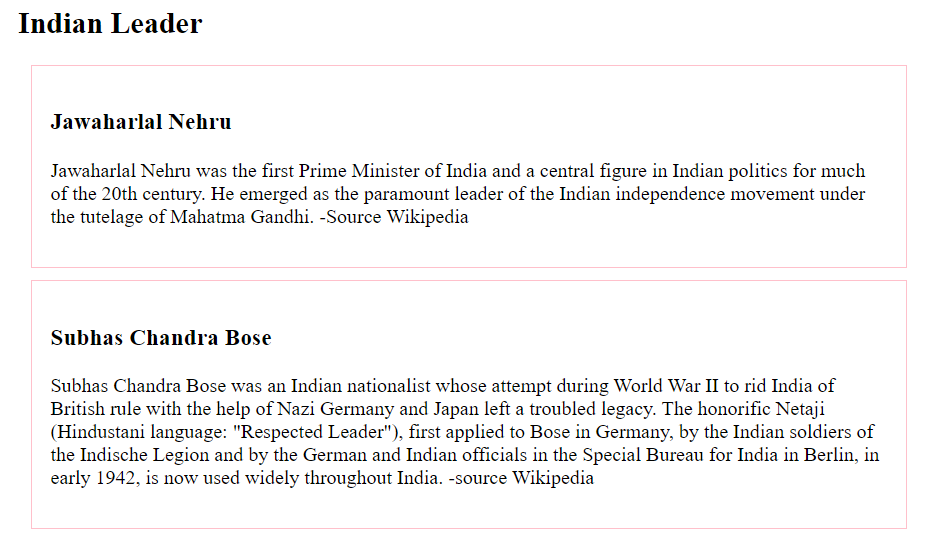
British rule with the help of Nazi Germany and Japan left a troubled legacy.

The honorific Netaji (Hindustani language: "Respected Leader"), first applied to Bose in Germany,

 by the Indian soldiers of the Indische Legion and by the German and Indian officials

 in the Special Bureau for India in Berlin, in early 1942, is now used widely throughout India. -source Wikipedia**</p>**

**</section>**



# **HTML Header Tag**

**HTML <header> tag** is used as a container of introductory content or navigation links. Generally a <header> element contains one or more heading elements, logo or icons or author's information.

You can use several <header> elements in one document, but a <header> element cannot be placed within a <footer>, <address> or another <header> element.

## HTML Header Tag Example

**<header>**

**<h2>**ABCOnline.com**</h2>**

**<p>** World's no.1 shopping website**</p>**

**</header>**

## ABCOnline.com

World's no.1 shopping website

CSS Code:

header{

border: 1px solid pink;

background-color:pink;

padding:10px;

border-radius:5px;

}

HTML Code:

**<header>**

**<h2>**ABCOnline.com**</h2>**

**<p>** World's no.1 shopping website**</p>**

**</header>**



# **HTML Footer Tag**

**HTML <footer> tag** is used to define a footer for a document or a section. It is generally used in the last of the section (bottom of the page).

The footer tag is included in HTML5.

HTML <footer> tag contains information about its containing elements for example:

* author information
* contact information
* copyright information
* sitemap
* back to top links
* related documents etc.

#### **Note: You can have one or many footer elements in one document.**

If you want to put information like address, e-mail etc. about the author on your web page, all the relevant elements should be included into the footer element.

<!DOCTYPE>

<html>

<body>

<footer>

<p>Posted by: Sonoo Jaiswal</p>

<p>

<address> JavaTpoint, plot no. 6, near MMX Mall,Mohan Nagar, Ghaziabad Pin no. 201007

</address>

</p>

<p>Contact information:

<a href="mailto:sonoojaiswal1987@gmail.com">sonoojaiswal1987@gmail.com</a>.

</p>

</footer>

</body>

</html>



# **HTML Audio Tag**

**HTML audio tag** is used to define sounds such as music and other audio clips. Currently there are three supported file format for HTML 5 audio tag.

1. mp3
2. wav
3. ogg

HTML5 supports <video> and <audio> controls. The Flash, Silverlight and similar technologies are used to play the multimedia items.

This table defines that which web browser supports which audio file format.

|  |  |  |  |
| --- | --- | --- | --- |
| **Browser** | **mp3** | **wav** | **ogg** |
| ie browser Internet Explorer | yes | no | no |
| chrome browser Google Chrome | yes | yes | yes |
| firefox browser Mozilla Firefox | yes\* | yes | yes |
| opera browser Opera | no | yes | yes |
| safari browser Apple Safari | yes | yes | no |

## HTML Audio Tag Example

Let's see the code to play mp3 file using HTML audio tag.

<!DOCTYPE>

<html>

<body>

<audio controls>

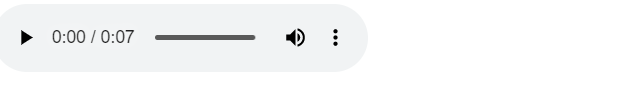
<source src="koyal.mp3" type="audio/mpeg">

Your browser does not support the html audio tag.

</audio>

</body>

</html>



## Attributes of HTML Audio Tag

There is given a list of HTML audio tag.

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| controls | It defines the audio controls which is displayed with play/pause buttons. |
| autoplay | It specifies that the audio will start playing as soon as it is ready. |
| loop | It specifies that the audio file will start over again, every time when it is completed. |
| muted | It is used to mute the audio output. |
| preload | It specifies the author view to upload audio file when the page loads. |
| src | It specifies the source URL of the audio file. |

## HTML Audio Tag Attribute Example

Here we are going to use controls, autoplay, loop and src attributes of HTML audio tag.

**<audio** controls autoplay loop**>**

**<source** src="koyal.mp3" type="audio/mpeg"**></audio>**

## MIME Types for HTML Audio format

The available MIME type HTML audio tag is given below.

|  |  |
| --- | --- |
| **Audio Format** | **MIME Type** |
| mp3 | audio/mpeg |
| ogg | audio/ogg |
| wav | audio/wav |

# **HTML Video Tag**

HTML 5 supports <video> tag also. The HTML video tag is used for streaming video files such as a movie clip, song clip on the web page.

Currently, there are three video formats supported for HTML video tag:

1. mp4
2. webM
3. ogg

Let's see the table that defines which web browser supports video file format.

|  |  |  |  |
| --- | --- | --- | --- |
| **Browser** | **mp4** | **webM** | **ogg** |
| ie browser Internet Explorer | yes | no | no |
| chrome browser Google Chrome | yes | yes | yes |
| firefox browser Mozilla Firefox | yes | yes | yes |
| opera browser Opera | no | yes | yes |
| safari browser Apple Safari | yes | no | no |

#### **Android also supports mp4 format.**

## HTML Video Tag Example

Let's see the code to play mp4 file using HTML video tag.

**<video** controls**>**

**<source** src="movie.mp4" type="video/mp4"**>**

  Your browser does not support the html video tag.

**</video>**

## Attributes of HTML Video Tag

Let's see the list of HTML 5 video tag attributes.

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| controls | It defines the video controls which is displayed with play/pause buttons. |
| height | It is used to set the height of the video player. |
| width | It is used to set the width of the video player. |
| poster | It specifies the image which is displayed on the screen when the video is not played. |
| autoplay | It specifies that the video will start playing as soon as it is ready. |
| loop | It specifies that the video file will start over again, every time when it is completed. |
| muted | It is used to mute the video output. |
| preload | It specifies the author view to upload video file when the page loads. |
| src | It specifies the source URL of the video file. |

## HTML Video Tag Attribute Example

Let's see the example of video tag in HTML where are using height, width, autoplay, controls and loop attributes.

1. **<video** width="320" height="240" controls autoplay loop**>**
2. **<source** src="movie.mp4" type="video/mp4"**>**
3. Your browser does not support the html video tag.
4. **</video>**

## MIME Types for HTML Video format

The available MIME type HTML video tag is given below.

|  |  |
| --- | --- |
| **Video Format** | **MIME Type** |
| mp4 | video/mp4 |
| Ogg | video/ogg |
| webM | video/webM |

# **HTML Form Attributes**

## The Action Attribute

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

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

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

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

**Tip:** If the action attribute is omitted, the action is set to the current page.

## The Target Attribute

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

The target attribute can have one of the following values:

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

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

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

## The Method Attribute

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

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

The default HTTP method when submitting form data is GET.

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

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

**Notes on GET:**

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

**Notes on POST:**

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

**Tip:** Always use POST if the form data contains sensitive or personal information!

## The Autocomplete Attribute

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

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

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

## The Novalidate Attribute

The novalidate attribute is a boolean attribute.

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

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

## List of All <form> Attributes

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