Learning Outcome

## Create Styles of web pages using CSS

## 

# Cascaded Style Sheet (CSS)

## Introduction to CSS

CSS stands for Cascading Style Sheets. It is the language for describing the presentation of Web pages, including colours, layout, and fonts, thus making our web pages presentable to the users.

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

It is used to change the style (color, height, width, padding, margin, etc.) of the HTML element. By using CSS, we can design a very good user interface for your website or web application.

Styling your web page to make it look appealing is also an essential element of web designing and development. CSS helps in giving styles for describing the presentation of the markup-based documents.

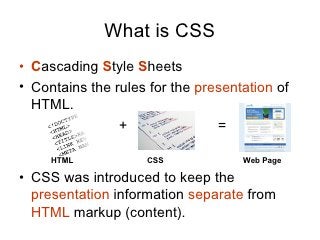


Image 1: CSS

Reference: <https://www.slideshare.net/webdevninja/introduction-to-css-13603389>

**World Wide Web Consortium(W3C)**

CSS is developed, updated, and maintained by a faction of people of the W3C, and the group is named "CSS Working Group". This group produces documents called specifications. When these specifications in the form of documents are released and approved by the developers' community, they are set officially so that other users can use it as a standard mechanism of using CSS.

### History of CSS​

CSS was first proposed by **Hakon Wium** Lie on October 10, 1994. At the time, Lie was working with Tim Berners-Lee (father of Html) at CERN. The European Organization for Nuclear Research is known as CERN. Hakon wium lie is known as father of css.​

CSS was proposed in 1994 as a web styling language, to solve some of the problems of Html 4. ​There were other styling languages proposed at this time, such as Style Sheets for Html and JSSS but CSS won.



Image 2: Founder of CSS

Reference: : <https://www.shecodes.io/workshops/shecodes-online-workshop-46-0/projects/197698>

**Include properties in CSS2**​

CSS level 2 specification was developed by the W3C and published as a recommendation in May 1998. CSS 2 includes a number of new capabilities like below;​

* absolute​
* relative​
* fixed​
* positioning​
* z-index​
* concept of media type​
* bidirectional text​
* new font properties such as shadows.​

CSS3 was started in 1998 but it has never been completed. Some parts are still being developed and some components work on some browsers. It published in June 1999. CSS 3 is divided into several separate documents called "modules". Each module adds new capabilities or extends features defined in CSS 2.​

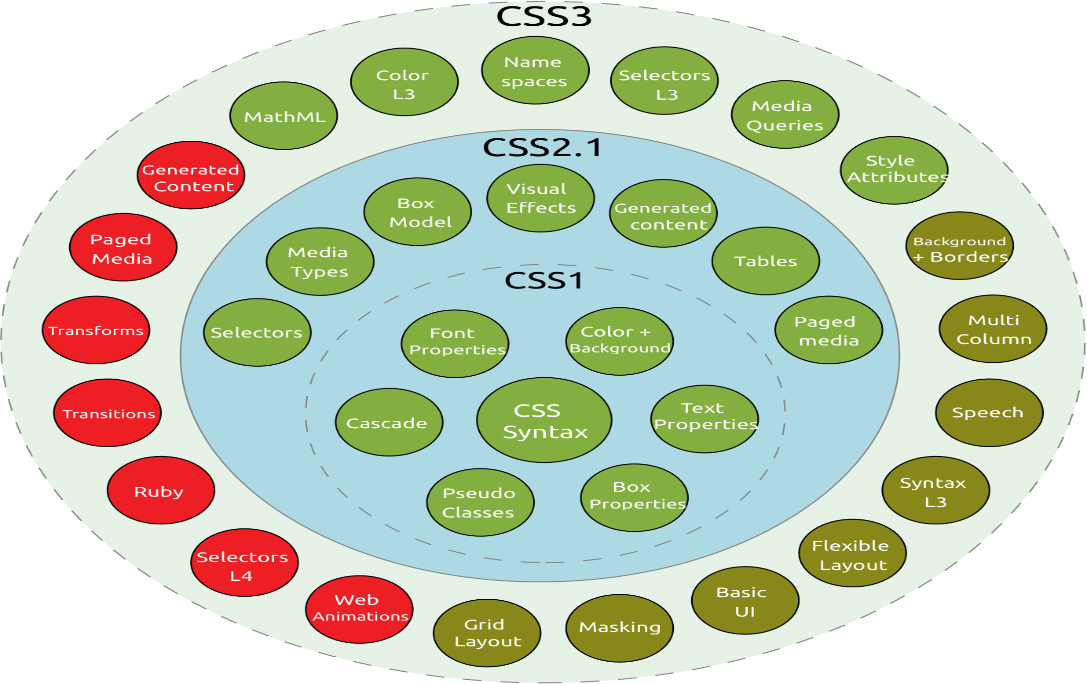


Image 3: History of CSS

Reference: <https://www.shecodes.io/workshops/shecodes-online-workshop-46-0/projects/197698>

### Why CSS?

**CSS saves time:**

You can write CSS once and reuse same sheet in multiple HTML pages.

**Easy Maintenance:**

To make a global change simply change the style, and all elements in all the webpages will be updated automatically.

**Search Engines:**

CSS is considered as clean coding technique, which means search engines won’t have to struggle to “read” its content.

**Superior styles to HTML:**

CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.

**Offline Browsing:**

CSS can store web applications locally with the help of offline cache. Using of this we can view offline websites.

## Limitations of CSS

**Confusion due to many CSS Versions:**

Beginners are more vulnerable to this issue. They might get confused while opting to learn CSS as there are many levels of CSS such as CSS2, CSS3, etc.

**Cross-Browser Issues:**

Different browsers work differently. So, you have to check that changes implemented in the website via CSS codes are reflected properly among all the browsers.

**Security Issues:**

Security is important in today’s world driven by technology and data. One of the major disadvantages of CSS is that it has limited security.

**Extra Work for Developers:**

Design services are required to consider and test all CSS codes across different browsers for compatibility. Due to developers testing compatibility for different browsers, their workload increases.

## Advantages of CSS

**CSS saves time**

Once you write a CSS code you can use it in more than one HTML page. User can define a style for each HTML element and it can apply as many web pages. By doing this we save a lot of time.

**Pages load faster**

You don't need to write HTML tag attributes every time if you are using CSS. Just write a CSS Rule of a tag and apply it to every web pages. By doing this, the download time of your website will be greatly reduced.

Maintenance of a website becomes very easy using Easy maintenance −CSS. If you want to change the style of your website completely, then you have to change the simple style code, all the other page elements will be changed automatically.

**Superior styles to HTML**

CSS has a lot more detailed features than HTML, allowing you to give your HTML page a much better look than HTML features.

**Multiple Device Compatibility**

Style sheets allow the material to be optimized for more than one type of device. Using the same HTML document, different versions of a website can be presented for handheld devices such as PDAs and cell phones or for printing.

**Global web standards**

Now recommended to use CSS as required by HTML attributes. So, it is a good idea to start using CSS in all HTML pages to make them compatible with future browsers.

**Offline Browsing**

On using CSS, we can view offline websites. The cache is also very helpful in making a website load faster and ensures better performance of the website.

**Platform Independence**

CSS Script provides consistent platform independence and also supports every latest Web Browser.

## CSS Syntax

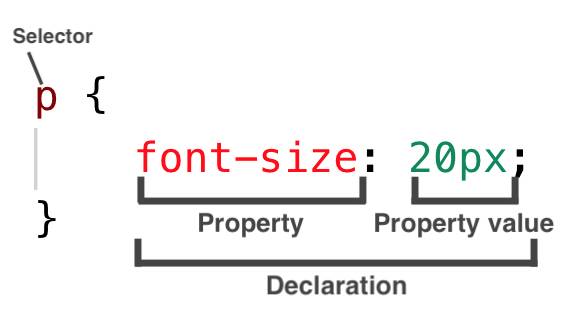
****

Image 4: CSS Syntax

**Selector**: selects the element you want to target​

**Keys**: properties(attributes) like color, font-size, background, width, height, etc​

**Value**: values associated with these properties​

There are few basic selectors like tags, id’s, and classes​All forms this key-value pair​.

### CSS Comments

Comments don’t render on the browser​

* Helps to understand our code better and makes it readable.​
* Helps to debug our code​

Comments are used to explain the code, and may help when you edit the source code at a later date.​

Comments are ignored by browsers.​

​**Two ways to comment:​**

**Single line**​

/\*<h6> This represents the most/ least important line of the doc. </h6>\*/​

​

**Multiple lines:**​​

/\*​

         h1​

     {​

     color: red;​

     text-align: center;​

      } ​

\*/​

### White Spaces in CSS

White spaces are special characters that can be an actual space, tab, or newline (carriage return). These whitespaces are used to construct your stylesheets extra readable. Like that of HTML, the browser usually ignores almost all of the whitespaces within your CSS code; and is meant to make the code human-readable.

## Three ways to integrate CSS

There are three ways of inserting a style sheet in any Html documents.

* Inline style sheet​
* Internal style sheet​ (Embedded)
* External style sheet​

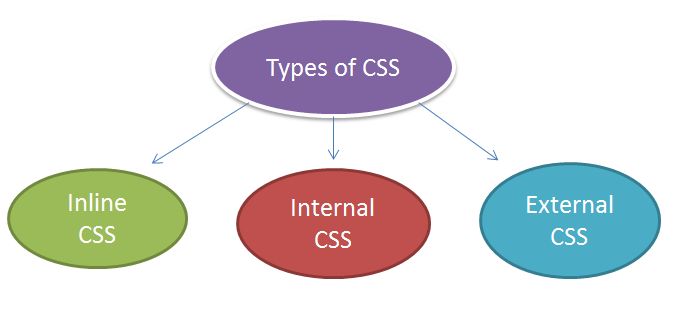


Image 5: Types of CSS

Reference: <https://www.bitdegree.org/learn/inline-css>​

**Inline Styles:**

* Inline styles are placed within an HTML element in the code.
* When you use inline style, your styling will only affect the element you selected.
* Inline styles do not have selectors because its written inside the html element.

Example:

<p style="color: red; margin-left:20px">This is a paragraph. </p>​

**Internal Styles:**

* are placed in head head section of the web page you are writing via style tag <style type=”text/css></style>.
* The styles you have written will only be used for the web page you used it in.
* Internal Styles are also called “Embedded styles”

Example:

<style>

hr {color:red;}

p {margin-left:20px;}

</style>

**External Styles:**

External Styles can be reused to apply on more than one page by only linking the style sheet to the web page.

Write your CSS codes via any of the code editors and then save it as a .css file. then link the style sheet to the HTML page by adding this code in the head section:

Example:

<head>​

<link rel="stylesheet" type="text/css" href="name of the Css file">​

</head>​

            p{​color: red; }​      //.css file​

## Merits and demerits of - external Style Sheets, Embedded Style Sheets

**Embedded Style Sheets**

**Merits**

* No need to upload multiple files as the CSS code is added to the same HTML file corresponding to the web page.
* Class and ID selectors can be used.

**Demerits**

* Adding CSS code to the HTML file results in increasing the page size and therefore, reducing the loading speed.
* Using it for multiple web pages is ineffective as it is required to add the same CSS rules for every web page

**External Style Sheets**

**Merits**

* A single external CSS file can be used for styling several web pages.
* HTML files leveraging external CSS have a cleaner structure and are smaller in size.

**Demerits**

* Linking to or uploading several external CSS files might decrease a website’s download speed and affect its performance.
* Web pages requiring the external CSS file might not be rendered accurately until the same is fully loaded.



Image 6: Types of CSS

References

1. <https://www.w3schools.in/css3/introduction-to-css/>

# CSS Units and Values

CSS has several different units for expressing a length. Many CSS properties take "length" values, such as width, margin, padding, font-size, etc.

**Length** is a number followed by a length unit, such as 10px, 2em, etc.

**Example**

Set different length values, using px (pixels):

h1 {  
  font-size: 60px;  
}  
  
p {  
  font-size: 25px;  
  line-height: 50px;  
}

There are two types of length units:

* **absolute**
* **relative**

### Absolute Lengths

The absolute length units are fixed and a length expressed in any of these will appear as exactly that size.

Absolute length units are not recommended for use on screen, because screen sizes vary so much. However, they can be used if the output medium is known, such as for print layout.

|  |  |
| --- | --- |
| **Unit** | **Description** |
| cm | centimeters |
| mm | millimeters |
| in | inches (1in = 96px = 2.54cm) |
| px \* | pixels (1px = 1/96th of 1in) |
| pt | points (1pt = 1/72 of 1in) |
| pc | picas (1pc = 12 pt) |

\* Pixels (px) are relative to the viewing device. For low-dpi devices, 1px is one device pixel (dot) of the display. For printers and high-resolution screens 1px implies multiple device pixels.

### Relative Lengths

Relative length units specify a length relative to another length property. Relative length units scale better between different rendering medium.

|  |  |
| --- | --- |
| **Unit** | **Description** |
| em | Relative to the font-size of the element (2em means 2 times the size of the current font) |
| ex | Relative to the x-height of the current font (rarely used) |
| ch | Relative to the width of the "0" (zero) |
| rem | Relative to font-size of the root element |
| vw | Relative to 1% of the width of the viewport\* |
| vh | Relative to 1% of the height of the viewport\* |
| vmin | Relative to 1% of viewport's\* smaller dimension |
| vmax | Relative to 1% of viewport's\* larger dimension |
| % | Relative to the parent element |

# CSS Styling Text

CSS has a lot of properties for formatting text.

**Text Formatting**

This text is styled with some of the text formatting properties. The heading uses the text-align, text-transform, and color properties. The paragraph is indented, aligned, and the space between characters is specified.

**Text Color**

The color property is used to set the color of the text.

The color is specified by:

* a color name - like "red"
* a HEX value - like "#ff0000"
* an RGB value - like "rgb(255,0,0)"

Look at [CSS Color Values](https://www.w3schools.com/cssref/css_colors_legal.asp) for a complete list of possible color values.

The default text color for a page is defined in the body selector.

**Example**

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

**Text Color and Background Color**

In this example, we define both the background-color property and the color property:

**Example**

body {  
  background-color: lightgrey;  
  color: blue;  
}  
  
h1 {  
  background-color: black;  
  color: white;  
}  
div {  background-color: blue; color: white;}

**The CSS Text Color Property**

|  |  |
| --- | --- |
| **Property** | **Description** |
| [color](https://www.w3schools.com/cssref/pr_text_color.asp) | Specifies the color of text |

**CSS Text Alignment and Text Direction**

In this chapter you will learn about the following properties:

* text-align
* text-align-last
* direction
* Unicode-bidi
* vertical-align

**Text Alignment**

The text-align property is used to set the horizontal alignment of a text.

A text can be left or right aligned, centered, or justified.

The following example shows center aligned, and left and right aligned text (left alignment is default if text direction is left-to-right, and right alignment is default if text direction is right-to-left):

**Example**

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

When the text-align, property is set to "justify", each line is stretched so that every line has equal width, and the left and right margins are straight (like in magazines and newspapers):

**Example**

div {  
  text-align: justify;  
}

**Text Align Last**

The text-align-last property specifies how to align the last line of a text.

**Example**

Align the last line of text in three <p> elements:

p.a {  
  text-align-last: right;  
}  
  
p.b {  
  text-align-last: center;  
}  
  
p.c {  
  text-align-last: justify;  
}

**Text Direction**

The direction and unicode-bidi properties can be used to change the text direction of an element:

**Example**

p {  
  direction: rtl;  
  unicode-bidi: bidi-override;  
}

**Vertical Alignment**

The vertical-align property sets the vertical alignment of an element.

**Example**

Set the vertical alignment of an image in a text:

img.a {  
  vertical-align: baseline;  
}  
  
img.b {  
  vertical-align: text-top;  
}  
  
img.c {  
  vertical-align: text-bottom;  
}  
  
img.d {  
  vertical-align: sub;  
}  
  
img.e {  
  vertical-align: super;  
}

**The CSS Text Alignment/Direction Properties**

|  |  |
| --- | --- |
| Property | Description |
| [direction](https://www.w3schools.com/cssref/pr_text_direction.asp) | Specifies the text direction/writing direction |
| [text-align](https://www.w3schools.com/cssref/pr_text_text-align.asp) | Specifies the horizontal alignment of text |
| [text-align-last](https://www.w3schools.com/cssref/css3_pr_text-align-last.asp) | Specifies how to align the last line of a text |
| [unicode-bidi](https://www.w3schools.com/cssref/pr_text_unicode-bidi.asp) | Used together with the [direction](https://www.w3schools.com/cssref/pr_text_direction.asp) property to set or return whether the text should be overridden to support multiple languages in the same document |
| [vertical-align](https://www.w3schools.com/cssref/pr_pos_vertical-align.asp) | Sets the vertical alignment of an element |

**CSS Text Decoration**

In this chapter you will learn about the following properties:

* text-decoration-line
* text-decoration-color
* text-decoration-style
* text-decoration-thickness
* text-decoration

**Add a Decoration Line to Text**

The text-decoration-line property is used to add a decoration line to text.

**Tip:** You can combine more than one value, like overline and underline to display lines both over and under a text.

**Example**

h1 {  
  text-decoration-line: overline;  
}  
  
h2 {  
  text-decoration-line: line-through;  
}  
  
h3 {  
  text-decoration-line: underline;  
}  
  
p {  
  text-decoration-line: overline underline;  
}

**Specify a Color for the Decoration Line**

The text-decoration-color property is used to set the color of the decoration line.

**Example**

h1 {  
  text-decoration-line: overline;  
  text-decoration-color: red;  
}  
  
h2 {  
  text-decoration-line: line-through;  
  text-decoration-color: blue;  
}  
  
h3 {  
  text-decoration-line: underline;  
  text-decoration-color: green;  
}  
  
p {  
  text-decoration-line: overline underline;  
  text-decoration-color: purple;  
}

**Specify a Style for the Decoration Line**

The text-decoration-style property is used to set the style of the decoration line.

**Example**

h1 {  
  text-decoration-line: underline;  
  text-decoration-style: solid;  
}  
  
h2 {  
  text-decoration-line: underline;  
  text-decoration-style: double;  
}  
  
h3 {  
  text-decoration-line: underline;  
  text-decoration-style: dotted;  
}  
  
p.ex1 {  
  text-decoration-line: underline;  
  text-decoration-style: dashed;  
}  
  
p.ex2 {  
  text-decoration-line: underline;  
  text-decoration-style: wavy;  
}  
  
p.ex3 {  
  text-decoration-line: underline;  
  text-decoration-color: red;  
  text-decoration-style: wavy;  
}

**Specify the Thickness for the Decoration Line**

The text-decoration-thickness property is used to set the thickness of the decoration line.

**Example**

h1 {  
  text-decoration-line: underline;  
  text-decoration-thickness: auto;  
}  
  
h2 {  
  text-decoration-line: underline;  
  text-decoration-thickness: 5px;  
}  
  
h3 {  
  text-decoration-line: underline;  
  text-decoration-thickness: 25%;  
}  
  
p {  
  text-decoration-line: underline;  
  text-decoration-color: red;  
  text-decoration-style: double;  
  text-decoration-thickness: 5px;  
}

**The Shorthand Property**

The text-decoration property is a shorthand property for:

* text-decoration-line (required)
* text-decoration-color (optional)
* text-decoration-style (optional)
* text-decoration-thickness (optional)

**Example**

h1 {  
  text-decoration: underline;  
}  
  
h2 {  
  text-decoration: underline red;  
}  
  
h3 {  
  text-decoration: underline red double;  
}  
p {  
  text-decoration: underline red double 5px;  
}

**All CSS text-decoration Properties**

|  |  |
| --- | --- |
| **Property** | **Description** |
| [text-decoration](https://www.w3schools.com/cssref/pr_text_text-decoration.asp) | Sets all the text-decoration properties in one declaration |
| [text-decoration-color](https://www.w3schools.com/cssref/css3_pr_text-decoration-color.asp) | Specifies the color of the text-decoration |
| [text-decoration-line](https://www.w3schools.com/cssref/css3_pr_text-decoration-line.asp) | Specifies the kind of text decoration to be used (underline, overline, etc.) |
| [text-decoration-style](https://www.w3schools.com/cssref/css3_pr_text-decoration-style.asp) | Specifies the style of the text decoration (solid, dotted, etc.) |
| [text-decoration-thickness](https://www.w3schools.com/cssref/pr_text_text-decoration-thickness.asp) | Specifies the thickness of the text decoration line |

**Text Transformation**

The text-transform property is used to specify uppercase and lowercase letters in a text.

It can be used to turn everything into uppercase or lowercase letters, or capitalize the first letter of each word:

**Example**

p.uppercase {  
  text-transform: uppercase;  
}  
  
p.lowercase {  
  text-transform: lowercase;  
}  
  
p.capitalize {  
  text-transform: capitalize;  
}

**The CSS Text Transformation Property**

|  |  |
| --- | --- |
| **Property** | **Description** |
| [text-transform](https://www.w3schools.com/cssref/pr_text_text-transform.asp) | Controls the capitalization of text |

**CSS Text Indentation, Letter Spacing, Line Height, Word Spacing, and White Space**

In this chapter you will learn about the following properties:

* text-indent
* letter-spacing
* line-height
* word-spacing
* white-space

**Text Indentation**

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

**Example**

p {  
  text-indent: 50px;  
}

**Letter Spacing**

The letter-spacing property is used to specify the space between the characters in a text. The following example demonstrates how to increase or decrease the space between characters:

**Example**

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

**Line Height**

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

**Example**

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.The following example demonstrates how to increase or decrease the space between words:

**Example**

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.This example demonstrates how to disable text wrapping inside an element:

**Example**

p {  
  white-space: nowrap;  
}

**The CSS Text Spacing Properties**

|  |  |
| --- | --- |
| **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 |

**Text Shadow**

The text-shadow property adds shadow to text. In its simplest use, you only specify the horizontal shadow (2px) and the vertical shadow (2px):

Text shadow effect!

**Example**

h1 {  
  text-shadow: 2px 2px;  
}

Next, add a color (red) to the shadow:

Text shadow effect!

**Example**

h1 {  
  text-shadow: 2px 2px red;  
}

Then, add a blur effect (5px) to the shadow:

**Text shadow effect**

**Example**

h1 {  
  text-shadow: 2px 2px 5px red;  
}

# Styling Box

### CSS Box Model

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 every HTML element. It consists of: margins, borders, padding, and the actual content. The image below illustrates the box model:

Explanation of the different parts:

Graphical user interface

Description automatically generated

* **Content -** The content of the box, where text and images appear
* **Padding -** Clears an area around the content. The padding is transparent
* **Border**- A border that goes around the padding and content
* **Margin -** Clears an area outside the border. The margin is transparent

The box model allows us to add a border around elements, and to define space between elements.

**Example**

Demonstration of the box model:

div

 {  
  width: 300px;  
  border: 15px solid green;  
  padding: 50px;  
  margin: 20px;  
}

**Width and Height of an Element**

In order to set the width and height of an element correctly in all browsers, you need to know how the box model works.

Important: When you set the width and height properties of an element with CSS, you just set the width and height of the content area. To calculate the full size of an element, you must also add padding, borders and margins.

**Example**

This <div> element will have a total width of 350px:

div {  
  width: 320px;  
  padding: 10px;  
  border: 5px solid gray;  
  margin: 0;  
}

**Here is the calculation:**

320px (width)  
+ 20px (left + right padding)  
+ 10px (left + right border)  
+ 0px (left + right margin)  
**= 350px**

The total width of an element should be calculated like this:

Total element width = width + left padding + right padding + left border + right border + left margin + right margin

The total height of an element should be calculated like this:

Total element height = height + top padding + bottom padding + top border + bottom border + top margin + bottom margin

# Website Layout

A website can be divided into various sections comprising of header, menus, content and footer based on which there are many different layout designs available for developer. Different layouts can be created by using div tag and use CSS property to style it.   
The most common structure of website layout is given below: 

Diagram

Description automatically generated with medium confidence

Notice: Header section contains a website logo, a search bar and profile of user. The navigation menu contains link to various categories of articles available and content section is divided into 3 parts(columns) with left and right sidebar containing links to other articles and advertisements whereas the main content section is the one containing this article, then at the bottom there is a footer section which contains address, links, contacts etc.

**Header Section:** The header section is generally placed either at the top of the Website or just below a top navigation menu. It often comprises of the name of the Website or the logo of the Website.

**Example:**

|  |
| --- |
| <!-- This code describes the header section  of website layout -->  <!DOCTYPE html>  <html>      <head>          <title>              Website Layouts          </title>            <style>              .header {                  background-color: green;                  padding: 15px;                  text-align: center;              }          </style>      </head>        <body>          <div class = "header">              <h2 style = "color:white;">                  EduNet              </h2>          </div>          <br>            <center style="font-size:200%;">              Remaining Section          </center>      </body>  </html> |

**Output:**

Graphical user interface, text, application, chat or text message

Description automatically generated

**EduNet**

**Navigation Menu:** A Navigation Bar/Menu is basically a list of links that allows visitor to navigate through the website comfortably with easy access.

**Example:**

|  |
| --- |
| <!DOCTYPE html>  <html>      <head>          <title>              Website Layout          </title>            <style>                /\* CSS property for header section \*/              .header {                  background-color: green;                  padding: 15px;                  text-align: center;              }                /\* CSS property for navigation menu \*/              .nav\_menu {                  overflow: hidden;                  background-color: #333;              }              .nav\_menu a {                  float: left;                  display: block;                  color: white;                  text-align: center;                  padding: 14px 16px;                  text-decoration: none;              }              .nav\_menu a:hover {                  background-color: white;                  color: green;              }          </style>      </head>        <body>            <!-- header of website layout -->          <div class = "header">              <h2 style = "color:white;font-size:200%;">                  EduNet              </h2>          </div>            <!-- navigation menu for website layout -->          <div class = "nav\_menu">              <a href = "#">Algo</a>              <a href = "#">DS</a>              <a href = "#">Language</a>          </div><br>            <center style = "font-size:200%;">              Remaining Section          </center>      </body>  </html> |

**Output:** 

Graphical user interface, text, application, chat or text message

Description automatically generated

**EduNet**

**Content Section:** The content section is the main body of the website.

The user can divide content section in n-column layout.   
The most common layouts are: 

**1-Column Layout:** It is mostly used for mobile layout. 

Graphical user interface, text, application

Description automatically generated

**2-Column Layout:** This website layout is mostly used for tablets or laptops. 

Timeline

Description automatically generated with low confidence

**3-Column Layout:** This website layout is mostly used for desktops. 

A picture containing table

Description automatically generated

The user can also create a responsive layout where the layout will get changed as per screen size. Consider the below example where if width of screen is more than 600px then there will be 3-column layout and if width of screen is between 400px to 600px then there will be 2-column layout and if screen size less than 400px then 1-column layout will display.

**Example:**

|  |
| --- |
| <!DOCTYPE html>  <html>      <head>          <title>              Website Layout          </title>          <style>              \* {                  box-sizing: border-box;              }                /\* CSS property for header section \*/              .header {                  background-color: green;                  padding: 15px;                  text-align: center;              }                /\* CSS property for navigation menu \*/              .nav\_menu {                  overflow: hidden;                  background-color: #333;              }              .nav\_menu a {                  float: left;                  display: block;                  color: white;                  text-align: center;                  padding: 14px 16px;                  text-decoration: none;              }              .nav\_menu a:hover {                  background-color: white;                  color: green;              }                /\* CSS property for content section \*/              .columnA, .columnB, .columnC {                  float: left;                  width: 31%;                  padding: 15px;                  text-align:justify;              }              h2 {                  color:green;                  text-align:center;              }                /\* Media query to set website layout              according to screen size \*/              @media screen and (max-width:600px) {                  .columnA, .columnB, .columnC {                      width: 50%;                  }              }              @media screen and (max-width:400px) {                  .columnA, .columnB, .columnC {                      width: 100%;                  }              }          </style>      </head>        <body>          <!-- header of website layout -->          <div class = "header">              <h2 style = "color:white;font-size:200%">  EduNet              </h2>          </div>            <!-- navigation menu of website layout -->          <div class = "nav\_menu">              <a href = "#">Algo</a>              <a href = "#">DS</a>              <a href = "#">Language</a>          </div>            <!-- Content section of website layout -->          <div class = "row">                <div class = "columnA">                  <h2>Column A</h2>      <p>Prepare for the Recruitment drive of product                  based companies like Microsoft, Amazon, Adobe                  etc with a free online placement preparation                  course. The course focuses on various MCQ's                  & Coding question likely to be asked in the                  interviews & make your upcoming placement                  season efficient and successful.</p>                  </div>                <div class = "columnB">                  <h2>Column B</h2>    <p>Prepare for the Recruitment drive of product                  based companies like Microsoft, Amazon, Adobe                  etc with a free online placement preparation                  course. The course focuses on various MCQ's                  & Coding question likely to be asked in the                  interviews & make your upcoming placement                  season efficient and successful.</p>                  </div>                <div class = "columnC">                  <h2>Column C</h2>      <p>Prepare for the Recruitment drive of product                  based companies like Microsoft, Amazon, Adobe                  etc with a free online placement preparation                  course. The course focuses on various MCQ's                  & Coding question likely to be asked in the                  interviews & make your upcoming placement                  season efficient and successful.</p>                  </div>          </div>      </body>  </html> |
|  |

**Output:   
The width of screen size greater then 700px:**

Graphical user interface

Description automatically generated

**EduNet**

**The width of screen size greater then 400px and less then 600px:** 

Text

Description automatically generated with medium confidence

**EduNet**

**The width of screen size less then 400px:**

Text

Description automatically generated

**EduNet**

**Footer Section:** A footer section is placed at the bottom of the webpage and it generally consists of information like contact info, copyrights, About us etc.

**Example:**

|  |
| --- |
| <!DOCTYPE html>  <html>      <head>          <title>              CSS Website Layout          </title>            <style>                /\* Style for footer section \*/              .footer {                  background-color: green;                  padding: 15px;                  text-align: center;              }          </style>      </head>        <body>          <center style = "font-size:200%;">              Upper section          </center>            <!-- footer Section -->          <div class = "footer">              <a href = "#">About</a><br>              <a href = "#">Career</a><br>              <a href = "#">Contact Us</a>          </div>      </body>  </html> |

**Output:** 

Table

Description automatically generated with medium confidence