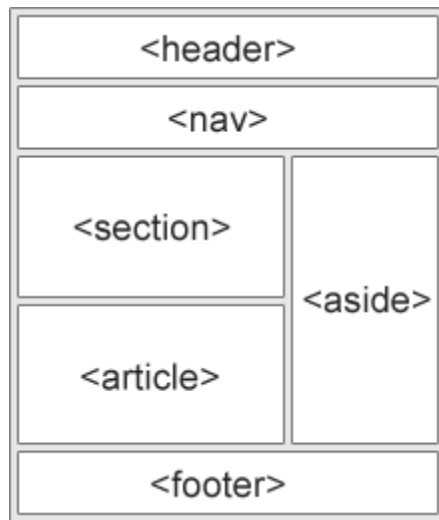


HTML Layout Elements and Techniques

HTML Layout Elements

HTML has several semantic elements that define the different parts of a web page:



- `<header>` - Defines a header for a document or a section
- `<nav>` - Defines a set of navigation links
- `<section>` - Defines a section in a document
- `<article>` - Defines an independent, self-contained content
- `<aside>` - Defines content aside from the content (like a sidebar)
- `<footer>` - Defines a footer for a document or a section
- `<details>` - Defines additional details that the user can open and close on demand
- `<summary>` - Defines a heading for the `<details>` element

You can read more about semantic elements in our [HTML Semantics](#) chapter.

Semantic Elements

Tag	Description	Example
header	Defines the header for a document or a section.	<header><h1>My Website</h1></header>
Nav	Defines a section containing navigation links.	<nav>HomeAbout Us</nav>
section	Defines a section of a document with a common theme	<section><h2>Products</h2><p>This is a description of our products.</p></section>
Article	Defines an independent, self-contained piece of content	<article><h3>Blog Post Title</h3><p>This is the content of the blog post.</p></article>
Aside	Defines content related to the main content, but placed separately. (e.g., sidebar)	<aside><h2>Contact Information</h2><p>Phone: 123-456-7890</p></aside>
Footer	Defines the footer for a document or a section. Typically contains copyright information, contact details, etc.	<footer><p>© 2024 My Company</p></footer>
details	Defines a disclosure widget that contains additional details the user can reveal/hide	<details><summary>Click to expand details</summary><p>This is the detailed content.</p></details>
summary	Defines a summary (heading) for the details element.	

Introduction in CSS

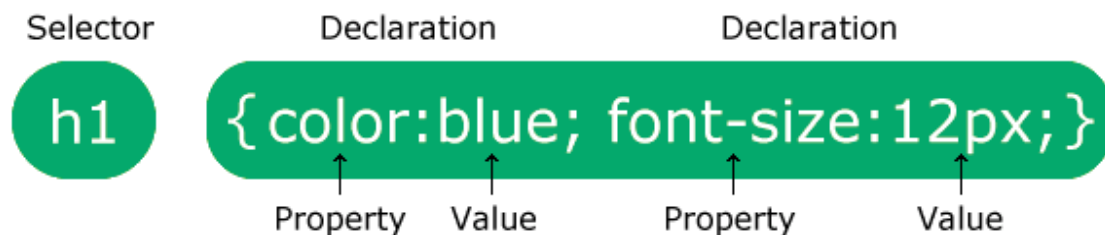
- CSS stands for Cascading Style Sheets
- CSS describes how HTML elements are to be displayed on screen, paper, or in other media
- CSS saves a lot of work. It can control the layout of multiple web pages all at once
- External stylesheets are stored in CSS files

Using CSS

CSS can be added to HTML documents in 3 ways:

- **Inline** - by using the `style` attribute inside HTML elements
- **Internal** - by using a `<style>` element in the `<head>` section
- **External** - by using a `<link>` element to link to an external CSS file

CSS Syntax



The selector points to the HTML element you want to style.

▪ CSS Colors

- Colors are specified using predefined color names, or RGB, HEX, HSL, RGBA, HSLA values.
- CSS/HTML support [140 standard color names](#).

▪ CSS Background Color

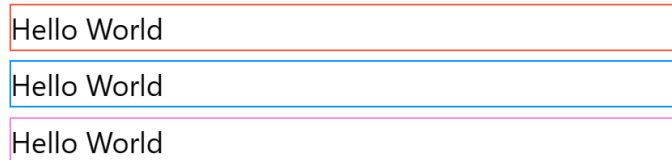
Example

```
<h1 style="background-color:DodgerBlue;">Hello World</h1>  
<p style="background-color:Tomato;">Lorem ipsum...</p>
```

- **CSS Text Color**

```
<h1 style="color:Tomato;">Hello World</h1>
<p style="color:DodgerBlue;">Lorem ipsum...</p>
<p style="color:MediumSeaGreen;">Ut wisi enim...</p>
```

- **CSS Border Color**



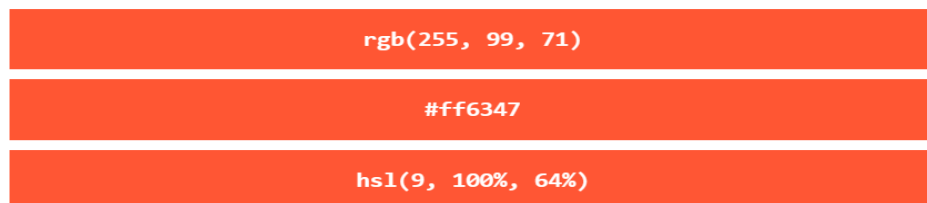
Example

```
<h1 style="border:2px solid Tomato;">Hello World</h1>
<h1 style="border:2px solid DodgerBlue;">Hello World</h1>
<h1 style="border:2px solid Violet;">Hello World</h1>
```

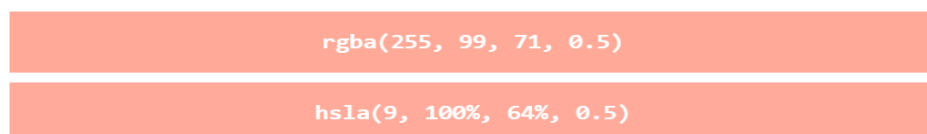
- **CSS Color Values**

In CSS, colors can also be specified using RGB values, HEX values, HSL values, RGBA values, and HSLA values:

Same as color name "Tomato":



Same as color name "Tomato", but 50% transparent:



```
<h1 style="background-color:rgb(255, 99, 71);">...</h1>
<h1 style="background-color:#ff6347;">...</h1>
<h1 style="background-color:hsl(9, 100%, 64%);">...</h1>

<h1 style="background-color:rgba(255, 99, 71, 0.5);">...</h1>
<h1 style="background-color:hsla(9, 100%, 64%, 0.5);">...</h1>
```

▪ CSS Backgrounds

The CSS background properties are used to add background effects for elements.

CSS background properties:

- `background-color`
- `background-image`
- `background-repeat`
- `background-attachment`
- `background-position`
- `background` (shorthand property)

▪ CSS background-color

```
body {  
  background-color: lightblue;  
}
```

▪ CSS Background Image

Example

Set the background image for a page:

```
body {  
  background-image: url("paper.gif");  
}
```

▪ CSS Background Image Repeat :

If the image above is repeated only horizontally (`background-repeat: repeat-x;`), the background will look better:

Example

```
body {  
  background-image: url("gradient_bg.png");  
  background-repeat: repeat-x;  
}
```

- **CSS background-repeat: no-repeat**

Showing the background image only once is also specified by the **background-repeat** property:

Example

Show the background image only once:

```
body {  
  background-image: url("img_tree.png");  
  background-repeat: no-repeat;  
}
```

- **CSS background-position:** The **background-position** property is used to specify the position of the background image.

Example

Position the background image in the top-right corner:

```
body {  
  background-image: url("img_tree.png");  
  background-repeat: no-repeat;  
  background-position: right top;  
}
```

- **CSS Background Attachment:** The **background-attachment** property specifies whether the background image should scroll or be fixed (will not scroll with the rest of the page):

Example: Specify that the background image should be fixed:

```
body {  
  background-image: url("img_tree.png");  
  background-repeat: no-repeat;  
  background-position: right top;  
  background-attachment: fixed;  
}
```

Example: Specify that the background image should scroll with the rest of the page:

```
body {  
  background-image: url("img_tree.png");  
  background-repeat: no-repeat;  
  background-position: right top;  
  background-attachment: scroll;  
}
```

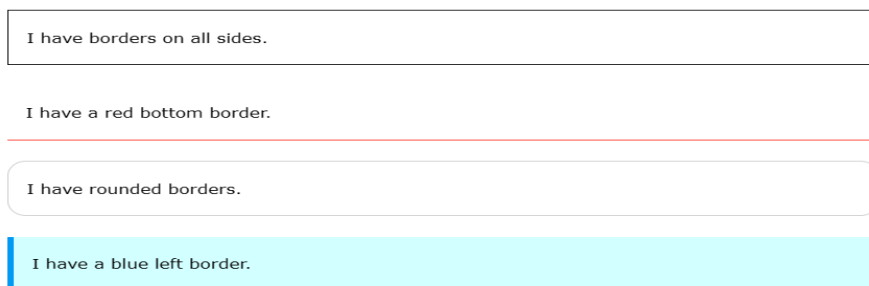
- **CSS Background Shorthand**

Example: Use the shorthand property to set the background properties in one declaration:

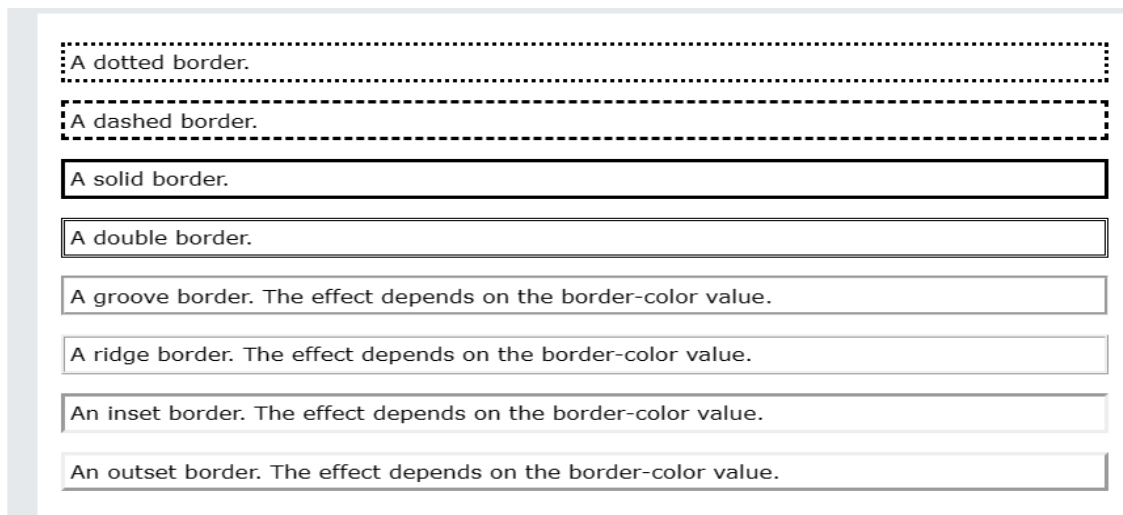
```
body {  
  background: #ffffff url("img_tree.png") no-repeat right top;  
}
```

- **CSS Borders**

The CSS border properties allow you to specify the style, width, and color of an element's border.



- **CSS Border Style:** The **border-style** property specifies what kind of border to display.



<style>

```
p.dotted {border-style: dotted;}  
p.dashed {border-style: dashed;}  
p.solid {border-style: solid;}  
p.double {border-style: double;}  
p.groove {border-style: groove;}  
p.ridge {border-style: ridge;}  
p.inset {border-style: inset;}
```

```
p.outset {border-style: outset;}
p.none {border-style: none;}
p.hidden {border-style: hidden;}
p.mix {border-style: dotted dashed solid double;}
```

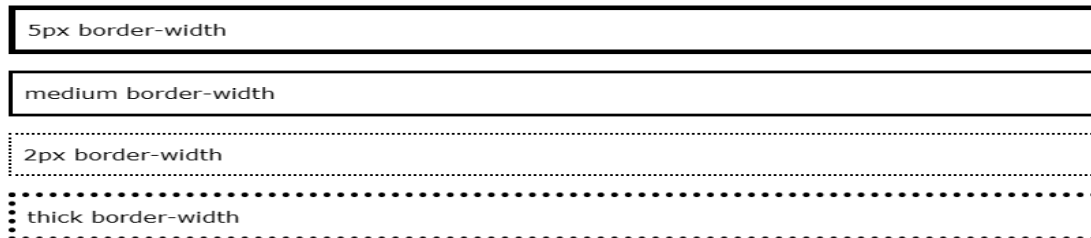
</style>

<body>

```
<h2>The border-style Property</h2>
<p>This property specifies what kind of border to display:</p>
<p class="dotted">A dotted border.</p>
<p class="dashed">A dashed border.</p>
<p class="solid">A solid border.</p>
<p class="double">A double border.</p>
<p class="groove">A groove border.</p>
<p class="ridge">A ridge border.</p>
<p class="inset">An inset border.</p>
<p class="outset">An outset border.</p>
<p class="none">No border.</p>
<p class="hidden">A hidden border.</p>
<p class="mix">A mixed border.</p>
```

</body>

- **CSS Border Width:** The **border-width** property specifies the width of the four borders. set as a specific size (in px, pt, cm, em, etc)



```
p.one {
  border-style: solid;
  border-width: 5px;
}
p.two {
  border-style: solid;
  border-width: medium;
}
p.three {
  border-style: dotted;
  border-width: 2px;
}
p.four {
  border-style: dotted;
```



```
border-width: thick;
}
```

Example

```
p.one {
  border-style: solid;
  border-width: 5px 20px; /* 5px top and bottom, 20px on the
sides */
}

p.two {
  border-style: solid;
  border-width: 20px 5px; /* 20px top and bottom, 5px on the
sides */
}

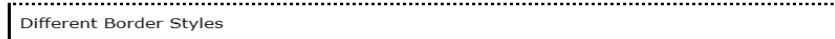
p.three {
  border-style: solid;
  border-width: 25px 10px 4px 35px; /* 25px top, 10px right, 4px
bottom and 35px left */
}
```

- **CSS Border Color**



CSS Border Sides

- **CSS Border - Individual Sides**



In CSS, there are also properties for specifying each of the borders (top, right, bottom, and left):

Example

```
p {
  border-top-style: dotted;
  border-right-style: solid;
  border-bottom-style: dotted;
  border-left-style: solid;
}
```

- **CSS Rounded Borders:** The `border-radius` property is used to add rounded borders to an element:

Normal border

Round border

Rounder border

Roudest border

Example

```
p {  
  border: 2px solid red;  
  border-radius: 5px;  
}
```

CSS Margins Margins are used to create space around elements, outside of any defined borders.

- Margin - Individual Sides

CSS has properties for specifying the margin for each side of an element:

- `margin-top`
- `margin-right`
- `margin-bottom`
- `margin-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
- `inherit` - specifies that the margin should be inherited from the parent element

Tip: Negative values are allowed.

Example

Set different margins for all four sides of a `<p>` element:

```
p {  
  margin-top: 100px;  
  margin-bottom: 100px;
```

```
margin-right: 150px;
margin-left: 80px;
}
```

- Margin - Shorthand Property

To shorten the code, it is possible to specify all the margin properties in one property.

The `margin` property is a shorthand property for the following individual margin properties:

- `margin-top`
- `margin-right`
- `margin-bottom`
- `margin-left`

So, here is how it works:

If the `margin` property has four values:

- **`margin: 25px 50px 75px 100px;`**
 - top margin is 25px
 - right margin is 50px
 - bottom margin is 75px
 - left margin is 100px

Example

Use the margin shorthand property with four values:

```
p {
  margin: 25px 50px 75px 100px;
}
```

- The auto Value

You can set the margin property to `auto` to horizontally center the element within its container.

Example

Use `margin: auto;`

```
div {
  width: 300px;
  margin: auto;
  border: 1px solid red;
}
```

- **CSS Margin Collapse**

Example

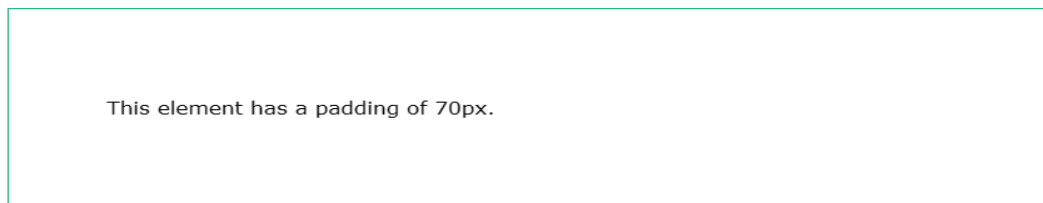
Demonstration of margin collapse:

```
h1 {  
  margin: 0 0 50px 0;  
}
```

```
h2 {  
  margin: 20px 0 0 0;  
}
```

❖ **CSS Padding:**

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



- **Padding - Individual Sides**

CSS has properties for specifying the padding for each side of an element:

- padding-top
- padding-right
- padding-bottom
- padding-left

Example

Set different padding for all four sides of a <div> element:

```
div {  
  padding-top: 50px;  
  padding-right: 30px;  
  padding-bottom: 50px;  
  padding-left: 80px;  
}
```

- Padding - Shorthand Property

To shorten the code, it is possible to specify all the padding properties in one property.

The `padding` property is a shorthand property for the following individual padding properties:

- `padding-top`
- `padding-right`
- `padding-bottom`
- `padding-left`

Example

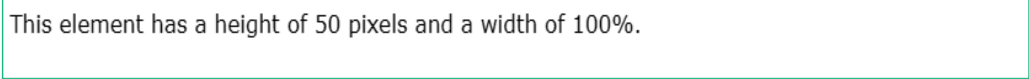
Use the padding shorthand property with four values:

```
div {  
  padding: 25px 50px 75px 100px;  
}
```

❖ **CSS** Height, Width and Max-width

The CSS `height` and `width` properties are used to set the height and width of an element.

The CSS `max-width` property is used to set the maximum width of an element.



This element has a height of 50 pixels and a width of 100%.

```
div {  
  
  height: 50px;  
  
  width: 100%;  
  
  border: 1px solid #4CAF50;  
  
}
```

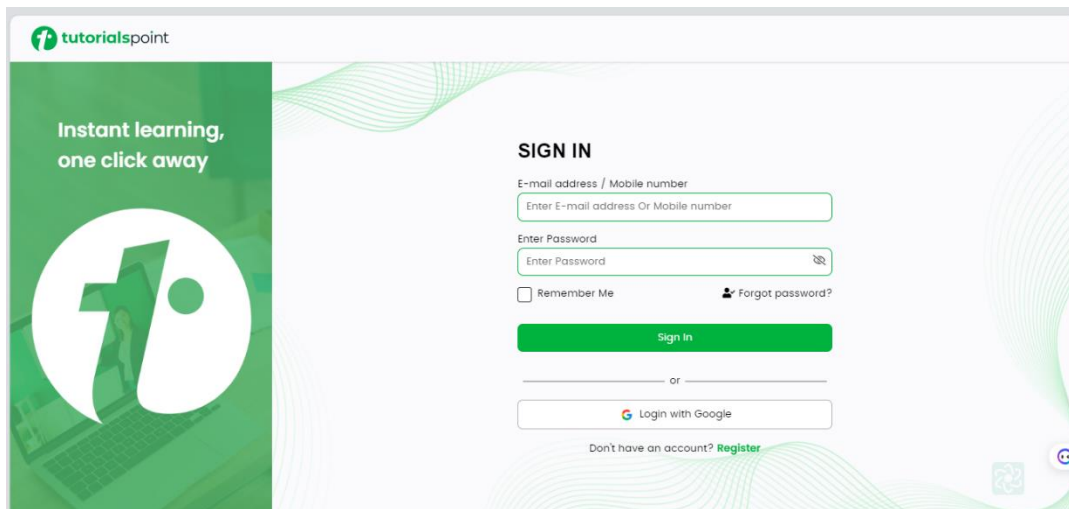
Example

This `<div>` element has a height of 100 pixels and a max-width of 500 pixels:

```
div {  
  max-width: 500px;  
  height: 100px;
```

```
background-color: powderblue;  
}
```

Design the following Page by HTML and CSS



```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport"
content="width=device-width, initial-
scale=1.0">
<title>Sign In</title>
<style>
body {
  font-family: Arial, sans-serif;
  margin: 0;
  padding: 0;
  background-color: #f2f2f2;
}
.container {
  width: 400px;
  margin: 50px auto;
  background-color: #fff;
  border-radius: 5px;
```

```
padding: 30px;
box-shadow: 0 0 10px rgba(0, 0, 0,
0.1);
}
.h1 {
  text-align: center;
  margin-bottom: 20px;
}
.input-field {
  margin-bottom: 15px;
}
.input-field label {
  display: block;
  margin-bottom: 5px;
}
.input-field input {
  width: 100%;
  padding: 10px;
  border: 1px solid #ccc;
  border-radius: 3px;
}
```

```

.checkbox {
    display: block;
    margin-bottom: 15px;
}
.checkbox label {
    margin-right: 5px;
}
.checkbox input {
    vertical-align: middle;
}
.button {
    display: block;
    width: 100%;
    padding: 10px 20px;
    background-color: #007bff;
    color: #fff;
    border: none;
    border-radius: 3px;
    cursor: pointer;
}
.button:hover {
    background-color: #0067cc;
}
.or {
    text-align: center;
    margin: 20px 0;
}
.social-login {
    text-align: center;
    margin-bottom: 20px;
}
.social-login a {

```

```

    margin-right: 10px;
    color: #007bff;
}
.social-login a:hover {
    text-decoration: underline;
}
.register {
    text-align: center;
}
.register a {
    color: #007bff;
}
</style>
</head>
<body>
<div class="container">
    <h1 class="h1">SIGN IN</h1>
    <form action="/action_page.php"
    method="post">
        <div class="input-field">
            <label for="username">E-mail
            address / Mobile number</label>
            <input type="text" id="username"
            name="username" placeholder="Enter
            E-mail address Or Mobile number">
        </div>
        <div class="input-field">
            <label
            for="password">Password</label>
            <input type="password"
            id="password" name="password"
            placeholder="Enter Password">
        </div>
        <div class="checkbox">
            <label for="remember_me">

```



```
<input type="checkbox"
id="remember_me"
name="remember_me">

Remember Me

</label>

<a href="#" class="forgot-
password">Forgot password?</a>

</div>

<button type="submit"
class="button">Sign In</button>

<div class="or">

<span>or</span>

</div>

<div class="social-login">

<a href="#" class="social-link"><i
class="fab fa-google"></i> Google
Login</a>

</div>

<p class="register">Don't have an
account? <a
href="#">Register</a></p>

</form>

</div>

</body>

</html>
```