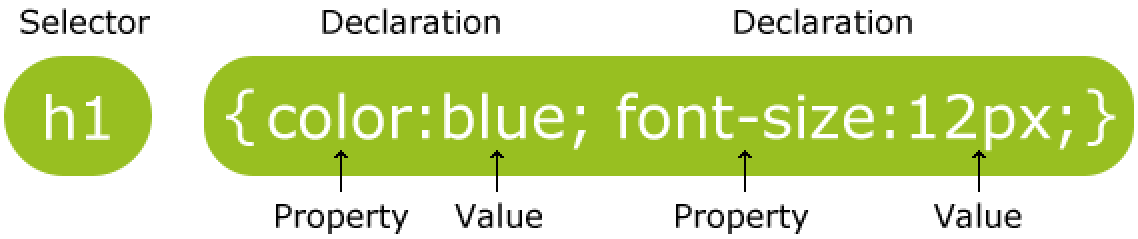
Cascading Style Sheets™ (CSS)

- *Cascading Style Sheets* (*CSS*) allow you to specify the style of your page elements (spacing,
- margins, etc.) separately from the structure of your document (section headers, body text,
- links, etc.). This *separation of structure from presentation* allows greater manageability
- and makes changing the style of your document easier.

What is CSS?

- **CSS** stands for **C**ascading **S**tyle **S**heets
- 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 Syntax
- A CSS rule-set consists of a selector and a declaration block:



- 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.
- A CSS declaration always ends with a semicolon, and declaration blocks are surrounded by curly braces.

The element Selector

<p> elements will be center-aligned, with a red text color:

```
p {
    color: red;
    text-align: center;
}
```

The id Selector

- The id selector uses the id attribute of an HTML element to select a specific element.
- The id of an element should be unique within a page, so the id selector is used to select one unique element!
- To select an element with a specific id, write a hash (#) character, followed by the id of the element.
- The style rule below will be applied to the HTML element with id="para1":

```
#para1 {
   text-align: center;
   color: red;
}
```

Note: An id name cannot start with a number!

```html
<!DOCTYPE html>
<html>
<head>
<style>
#para1 {
    text-align: center;
    color: red;
}
</style>
</head>
<body>

<p id="para1">Hello World!</p>
<p>This paragraph is not affected by the style.</p>

</body>
</html>
```

The class Selector

- The class selector selects elements with a specific class attribute.
- To select elements with a specific class, write a period (.) character, followed by the name of the class.
- In the example below, all HTML elements with class="center" will be red and center-aligned:

```html
<!DOCTYPE html>
<html>
<head>
<style>
.center {
    text-align: center;
    color: red;
}
</style>
</head>
<body>

<h1 class="center">Red and center-aligned heading</h1>
<p class="center">Red and center-aligned paragraph.</p>

</body>
</html>
```

```
<!DOCTYPE html>
<html>
<head>
<style>
.colortext {
    text-align: center;
    color: red;

</style>
</head>
<body>

<h1>This is a Heading</h1>
<p>This is a paragraph.</p>
<p class="colortext">This is another paragraph.</p>
<p class="colortext">This is also a paragraph.</p>

</body>
</html>
```

- you can also specify that only specific HTML elements should be affected by a class.
- In the example below, only <p> elements with class="center" will be center-aligned:

```
p.center {
  text-align: center;
  color: red;
}
```

HTML elements can also refer to more than one class.

In the example below, the <p> element will be styled according to class="center" and to class="large":

```
<!DOCTYPE html>
<html>
<head>
<style>
p.center {
    text-align: center;
    color: red;
}
p.large {
    font-size: 300%;
}
</style>
</head>
<body>
<h1 class="center">This heading will not be affected</h1>
<p class="center">This paragraph will be red and center-aligned.</p>
<p class="center large">This paragraph will be red, center-aligned, and in a large font-size.</p>
</body>
</html>
```

Note: A class name cannot start with a number!

Grouping Selectors

- If you have elements with the same style definitions, like this:

```
h1 {
   text-align: center;
   color: red;
}

h2 {
   text-align: center;
   color: red;
}

p {
   text-align: center;
   color: red;
}
```

- It will be better to group the selectors, to minimize the code.
- To group selectors, separate each selector with a comma.

```
h1, h2, p {
   text-align: center;
   color: red;
}
```

Exercise:

Change the color of all <p> and <h1> elements, to "red". Group the selectors to minimize code.

```
<!DOCTYPE html>
<html>
<head>
<style>

</style>
</head>
<body>

<h1>This is a heading</h1>
<h2>This is a smaller heading</h2>
<p>This is a paragraph.</p>
<p>This is another paragraph.</p>

</body>
</html>
```

```html
<!DOCTYPE html>
<html>
<head>
<style>
h1,h2,p{color:red;}

</style>
</head>
<body>

<h1>This is a heading</h1>
<h2>This is a smaller heading</h2>
<p>This is a paragraph.</p>
<p>This is another paragraph.</p>

</body>
</html>
```

- CSS Comments
- 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.
- A CSS comment starts with /* and ends with */. Comments can also span multiple lines:

```
p {
   color: red;
   /* This is a single-line comment */
   text-align: center;
}

/* This is
a multi-line
comment */
```

Exercise:

Change the color of all elements with the class "colortext", to "red".

```
<!DOCTYPE html>
<html>
<head>
<style>
</style>
</head>
<body>
<h1>This is a Heading</h1>
<p>This is a paragraph.</p>
<p class="colortext">This is another paragraph.</p>
<p class="colortext">This is also a paragraph.</p>
</body>
</html>
```

Three Ways to Insert CSS

There are three ways of inserting a style sheet:

- External style sheet
- Internal style sheet
- Inline style

Inline Styles

- An inline style may be used to apply a unique style for a single element.
- To use inline styles, add the style attribute to the relevant element. The style attribute can contain any CSS property.
- The example below shows how to change the color and the left margin of a <h1> element:

```
<!DOCTYPE html>
<html>
<body>

<h1 style="color:blue;margin-left:30px;">This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

Internal Style Sheet

- An internal style sheet may be used if one single page has a unique style.
- Internal styles are defined within the <style> element, inside the <head> section of an HTML page:

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
   background-color: linen;
}
h1 {
   color: maroon;
   margin-left: 40px;
}
</style>
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

External Style Sheet

- With an external style sheet, you can change the look of an entire website by changing just one file!
- Each page must include a reference to the external style sheet file inside the <link> element. The <link> element goes inside the <head> section:

Example

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

"mystyle.css" looks:

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

h1 {
   color: navy;
   margin-left: 20px;
}
```

Multiple Style Sheets

If some properties have been defined for the same selector (element) in different style sheets, the value from the last read style sheet will be used.

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css">
<style>
h1 {
    color: orange;
}
</style>
</head>
<body>

<h1>This is a heading</h1>
<p>The style of this document is a combination of an external stylesheet, and
internal style</p>

</body>
</html>
```

```
<head>
<style>
h1 {
    color: orange;
}
</style>
<link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
```

Cascading Order

- Inline style (inside an HTML element)
- External and internal style sheets (in the head section)
- Browser default
- So, an inline style (inside a specific HTML element) has the highest priority, which means that it will override a style defined inside the <head> tag, or in an external style sheet, or a browser default value.

CSS Colors

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

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

Border color

```
<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>
```

RGB Value

- In HTML, a color can be specified as an RGB value, using this formula:

rgb(*red, green*, *blue*)

HEX Value

- In HTML, a color can be specified using a hexadecimal value in the form:

#rrggbb

HSL Value

- In HTML, a color can be specified using hue, saturation, and lightness (HSL) in the form:

hsl(*hue*, *saturation*, *lightness*)

Eg. hsl(147, 50%, 47%)

RGBA Value

- RGBA color values are an extension of RGB color values with an alpha channel - which specifies the opacity for a color.
- An RGBA color value is specified with:

  rgba(*red, green*, *blue, alpha*)

The alpha parameter is a number between 0.0 (fully transparent) and 1.0 (not transparent at all)

HSLA Value

- HSLA color values are an extension of HSL color values with an alpha channel - which specifies the opacity for a color.
- An HSLA color value is specified with:
- **hsla(*hue, saturation*, *lightness, alpha*)**
- The alpha parameter is a number between 0.0 (fully transparent) and 1.0 (not transparent at all):

CSS Backgrounds

The CSS background properties are used to define the background effects for elements.

- CSS background properties:
- background-color
- background-image
- background-repeat
- background-attachment
- background-position

```
h1 {
   background-color: green;
}
```

image is repeated only horizontally (background-repeat: repeat-x;),

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

Tip: To repeat an image vertically, set background-repeat: repeat-y;

Background Image - Set position and no-repeat

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

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

The position of the image is specified by the background-position property:

```
background-position: right top;
```

Border Style

The border-style property specifies what kind of border to display.

The following values are allowed:

- dotted - Defines a dotted border
- dashed - Defines a dashed border
- solid - Defines a solid border
- double - Defines a double border
- groove - Defines a 3D grooved border. The effect depends on the border-color value
- ridge - Defines a 3D ridged border. The effect depends on the border-color value
- inset - Defines a 3D inset border. The effect depends on the border-color value
- outset - Defines a 3D outset border. The effect depends on the border-color value
- none - Defines no border
- hidden - Defines a hidden border

Border Width

- The border-width property specifies the width of the four borders.
- The width can be set as a specific size (in px, pt, cm, em, etc) or by using one of the three pre-defined values: thin, medium, or thick.

- **Example**

-
  ```
  p.one {
      border-style: solid;
      border-width: 5px;
  }

  p.two {
      border-style: solid;
      border-width: medium;
  }

  p.three {
      border-style: solid;
      border-width: 2px 10px 4px 20px;
  }
  ```

Border Color

- The border-color property is used to set the color of the four borders.
- The color can be set by:
- name - specify a color name, like "red"
- Hex - specify a hex value, like "#ff0000"
- RGB - specify a RGB value, like "rgb(255,0,0)"
- transparent

```
p.two {
  border-style: solid;
  border-color: green;
}

p.three {
  border-style: solid;
  border-color: red green blue yellow;
}
```

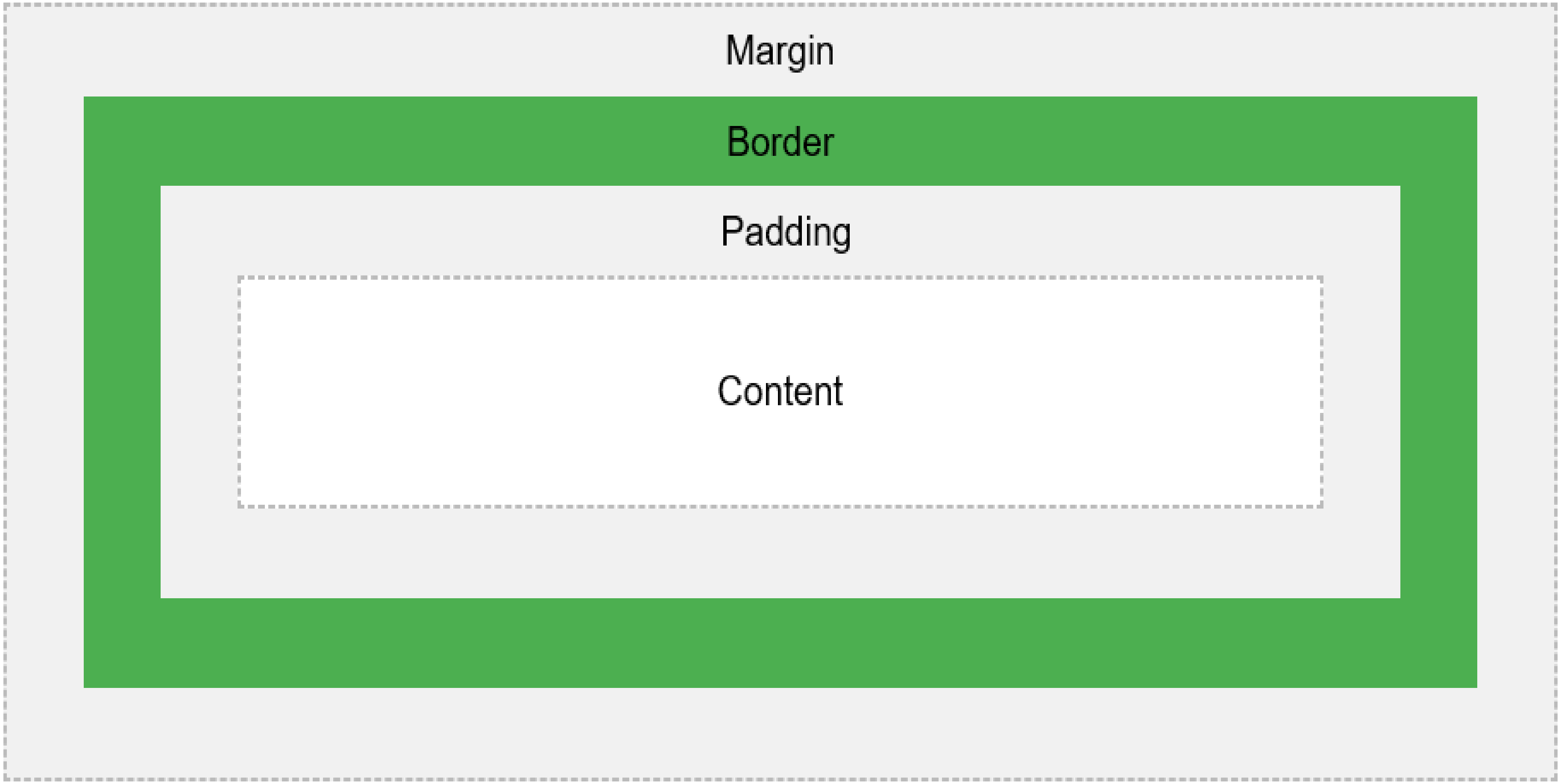
Border - Individual Sides

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

CSS Box Model

- The CSS Box Model
- 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 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:
- **Content** - The content of the box, where text



```css
div {
    width: 300px;
    border: 25px solid green;
    padding: 25px;
    margin: 25px;
}
```

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.

```
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:

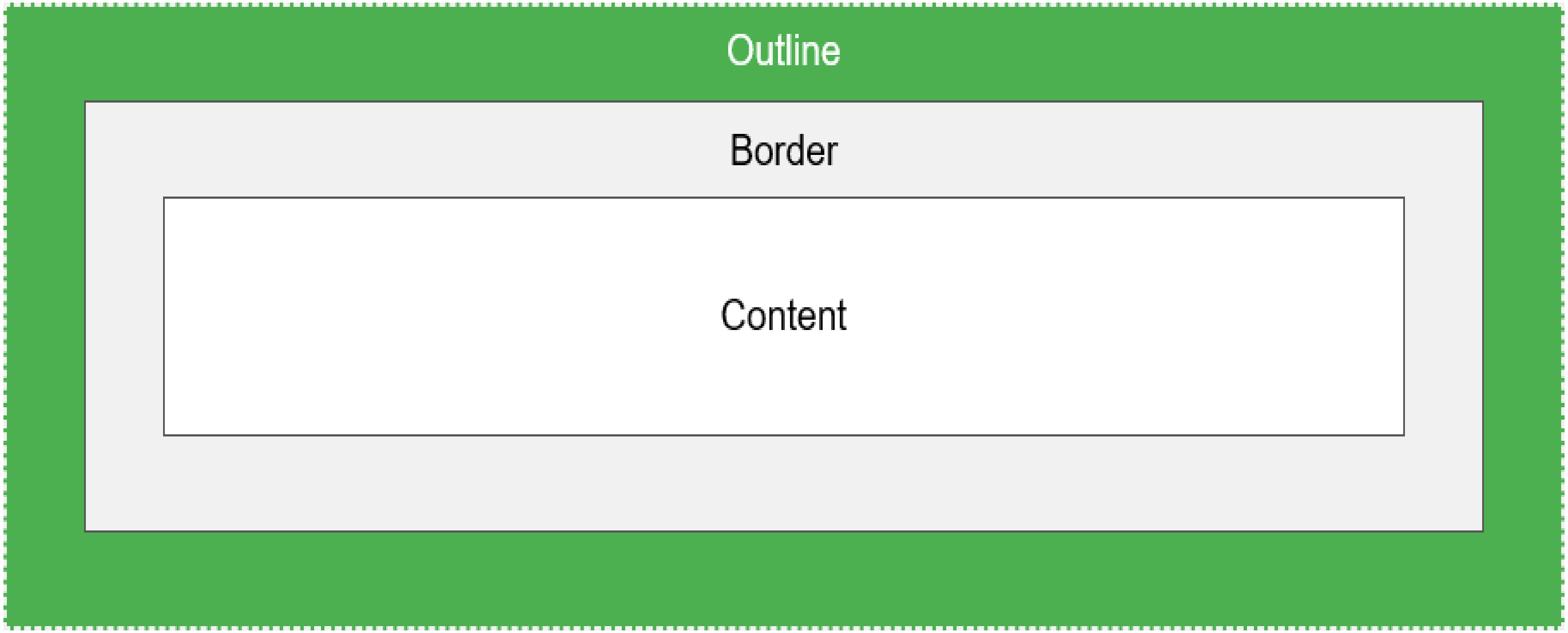
- Total element width = width + left padding + right padding + left border + right border + left margin + right margin

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

CSS Outline

- An outline is a line that is drawn around elements, OUTSIDE the borders, to make the element "stand out".



CSS has the following outline properties:

- outline-style
- outline-color
- outline-width
- outline-offset
- outline

Note: Outline differs from borders! Unlike border, the outline is drawn outside the element's border, and may overlap other content. Also, the outline is NOT a part of the element's dimensions; the element's total width and height is not affected by the width of the outline.

Outline Style

- The outline-style property specifies the style of the outline, and can have one of the following values:
- dotted - Defines a dotted outline
- dashed - Defines a dashed outline
- solid - Defines a solid outline
- double - Defines a double outline
- groove - Defines a 3D grooved outline
- ridge - Defines a 3D ridged outline
- inset - Defines a 3D inset outline
- outset - Defines a 3D outset outline
- none - Defines no outline
- hidden - Defines a hidden outline

Outline Color

- The outline-color property is used to set the color of the outline.
- The color can be set by:
- name - specify a color name, like "red"
- RGB - specify a RGB value, like "rgb(255,0,0)"
- Hex - specify a hex value, like "#ff0000"
- invert - performs a color inversion (which ensures that the outline is visible, regardless of color background)
-
  ```
  p.ex1 {
     border: 1px solid yellow;
     outline-style: solid;
     outline-color: invert;
  }
  ```

Outline Width

- The outline-width property specifies the width of the outline, and can have one of the following values:
- thin (typically 1px)
- medium (typically 3px)
- thick (typically 5px)
- A specific size (in px, pt, cm, em, etc)

Outline Offset

- The outline-offset property adds space between an outline and the edge/border of an element. The space between an element and its outline is transparent.
- The following example specifies an outline 15px outside the border edge:

```
p {
  margin: 30px;
  border: 1px solid black;
  outline: 1px solid red;
  outline-offset: 15px;
}
```

CSS Pseudo-classes

- A pseudo-class is used to define a special state of an element.

For example, it can be used to:

- Style an element when a user mouses over it
- Style visited and unvisited links differently
- Style an element when it gets focus

```
selector:pseudo-class {
   property:value;
}
```

Anchor Pseudo-classes

Links can be displayed in different ways:

```
<!DOCTYPE html>
<html>
<head>
<style>
/* unvisited link */
a:link {
   color: red;
}
/* visited link */
a:visited {
   color: green;
}
/* mouse over link */
a:hover {
   color: hotpink;
}
/* selected link */
a:active {
   color: blue;
}
</style>
</head>
```

```
<body>


<p><b><a href="default.asp" target="_blank">This is
a link</a></b></p>
<p><b>Note:</b> a:hover MUST come after a:link
and a:visited in the CSS definition in order to be
effective.</p>
<p><b>Note:</b> a:active MUST come after a:hover
in the CSS definition in order to be effective.</p>


</body>
</html>
```

Pseudo-classes and CSS Classes

```
<html>
<head>
<style>
a.highlight:hover {
   color: #ff0000;
}
</style>
</head>
<body>

<p><a class="highlight" href="css_syntax.asp">CSS Syntax</a></p>
<p><a href="default.asp">CSS Tutorial</a></p>

</body>
</html>
```

CSS - The :first-child Pseudo-class

- The :first-child pseudo-class matches a specified element that is the first child of another element.
- Match the first <p> element
- In the following example, the selector matches any <p> element that is the first child of any element:
- Example

-
  ```
  p:first-child {
      color: blue;
  }
  ```

Match the first <i> element in all <p> elements

-
  ```
  p i:first-child {
      color: blue;
  }
  ```

CSS - The :lang Pseudo-class

- The :lang pseudo-class allows you to define special rules for different languages.
- In the example below, :lang defines the quotation marks for <q> elements with lang="no":

```
<!DOCTYPE html>
<html>
<head>
<style>
q:lang(no) {
    quotes: "~" "~";
}
</style>
</head>
<body>
<p>Some text <q lang="no">A quote in a paragraph</q> Some text.</p>
<p>IE8 supports the :lang pseudo class only if a !DOCTYPE is specified.</p>
</body>
</html>
```

Output:

- Some text ~A quote in a paragraph~ Some text.
- IE8 supports the :lang pseudo class only if a !DOCTYPE is specified.

All CSS Pseudo Elements

Selector	Example	Example description
::after	p::after	Insert content after every <p> element
::before	p::before	Insert content before every <p> element
::first-letter	p::first-letter	Selects the first letter of every <p> element
::first-line	p::first-line	Selects the first line of every <p> element
::selection	p::selection	Selects the portion of an element that is selected by a user

All CSS Pseudo Classes

:active	a:active	Selects the active link
:checked	input:checked	Selects every checked <input> element
:disabled	input:disabled	Selects every disabled <input> element
:empty	p:empty	Selects every <p> element that has no children
:enabled	input:enabled	Selects every enabled <input> element
:first-child	p:first-child	Selects every <p> elements that is the first child of its parent
:first-of-type	p:first-of-type	Selects every <p> element that is the first <p> element of its parent
:focus	input:focus	Selects the <input> element that has focus
:hover	a:hover	Selects links on mouse over

Examples see from w3schools.com

<u>Css with xml:</u>

```xml
<?xml version="1.0"?>

<!--CSS stylesheet declaration.-->
<?xml:stylesheet type="text/css" href="Library.css"?>

<library>
        <name>Northmead Local Library</name>
        <book status="available">
                <publisher>Wiley International Edition</publisher>
                <title>Circuits, Devices and Systems</title>
                <edition>5th</edition>
                <author>
                        <first-name>Ralph J</first-name>
                        <last-name>Smith</last-name>
                </author>
                <author>
                        <first-name>Richard C</first-name>
                        <last-name>Dorf</last-name>
                </author>
                <isbn>0471552216</isbn>
                <callno>005.133/C</callno>
                <online_url>http://library/online_books/005133C.html</online_url>
        </book>
```

```
<book status="on_loan">
                <publisher>Prentice Hall PTR</publisher>
                <title>The C Programming Language</title>
                <edition>2nd</edition>
                <author>
                        <first-name>Brian W</first-name>
                        <last-name>Kernighan</last-name>
                </author>
                <author>
                        <first-name>Dennis M</first-name>
                        <last-name>Ritchie</last-name>
                </author>
                <isbn>0131103628</isbn>
                <callno>005.133/C2</callno>

        <online_url>http://library/online_books/005133C2.html</online_url>
        </book>
```

```
<journal series="XML Users Journal">
        <title>XML Users Journal August 1999</title>
        <date>1999-08-01</date>
        <callno>005.133/C</callno>
    </journal>
    <video status="available">
        <title>Lord of the Rings: The Fellowship of the Ring</title>
        <year>2001</year>
        <director>
            <name>Peter Jackson</name>
        </director>
        <genre>fantasy</genre>
        <genre>adventure</genre>
        <callno>643.11/L</callno>
    </video>
</library>
```

```
/* See LibraryCss.xml for details on how to use this CSS stylesheet with an XML file for display */

name
        {
        font-family: verdana, arial;
        font-size: 15pt;
        font-weight:bold;
        display: block;
        margin-bottom: 5pt;
}

media
        {
        font-family: verdana, arial;
        font-size: 12pt;
        font-weight:bold;
        color:green;
        display:block;
        margin-bottom: 20pt;
        margin-top: 40pt;
}
```

```
title
            {
            font-family: verdana, arial;
            font-size: 10pt;
            font-weight:bold;
            display:block;
            margin-top: 5pt;
}

author,date,director
            {           font-family: verdana, arial;
            font-size: 10pt;
            font-weight:normal;
            display:block;
            margin-left: 20pt;
}

online_url
            {display: none;}

callno
            {
            font-family: verdana, arial;
            font-size: 10pt;
            color:blue;
            display:block;
            margin-left: 20pt;
}
```