Introduction to CSS3

Lokesh Kumar

Introduction

CSS is a style sheet language used for informing the browser how to present a document.

It uses markup language for describing the presentation semantics of a document. In other words, an HTML document defines the content of the file whereas the CSS file defines how HTML elements are to be displayed.

CSS3

CSS is a mechanism used for adding style such as fonts, colors, and spacing to Web documents. CSS has multiple levels and profiles. Each level of CSS is updated from the earlier version, by adding new features. CSS version are denoted as CSS1, CSS2, CSS3, and CSS4, where the numbers are different for each version or level.

Note - The drafting of CSS4 was started by W3C on Sep 29, 2009. However, it is currently not supported by any Web browser.

CSS3 is divided into multiple documents called "modules". Each of these modules have new capabilities or extends the features present in CSS2. Drafting of CSS3 started when publication of the original CSS2 recommendation was released. The first CSS3 drafts were released on June 1999. CSS3 extends variety of new ways to create an impact with any designs, with quite a few important changes. The CSS3 logo is displayed in figure 5.1.

Modules

Since CSS3 is available as modules and is still evolving, there are many modules having different stability and status. Out of the fifty modules published by the CSS working group, only three modules are released as recommendations and they are as follows:

- CSS Color Level 3
- CSS Namespaces
- Selectors Level 3

The following modules are stable and in recommendation stage:

- Media Queries
- CSS style Attributes

Modules (contd.)

The following modules are in testing phase and in recommendation stage:

- CSS Backgrounds and Borders Level 3
- CSS Image Values and Replaced Content Level 3
- CSS Marquee
- CSS Multi-column Layout
- CSS Speech
- CSS Mobile Profile 2.0
- CSS TV Profile 1.0

Modules

The following modules are in refining phase and in working draft stage:

- CSS Transforms
- CSS Transitions
- CSS Values and Units Level 3
- CSS Print Profile

Modules (contd.)

The following modules are in revising phase and in working draft and recommendation stage:

- CSS Animations
- CSS Flexible Box Layout
- CSS Fonts Level 3
- CSS Paged Media Level 3
- CSS Text Level 3
- CSS Basic User Interface Level 3
- CSS Writing Modes Level 3
- CSSOM View

Modules (contd.)

The following modules are in exploring phase and in working draft stage:

- CSS Cascading and Inheritance Level 3
- CSS Conditional Rules Level 3
- CSS Grid Layout
- CSS Grid Template Layout
- CSS Line Grid
- CSS Lists Level 3
- CSS Tables Level 3
- Selectors Level 4
- CSS Object Model

Modules (contd.)

The following modules are in rewriting phase and in working draft stage:

- CSS Line Layout Level 3
- CSS Ruby
- CSS Syntax Level 3

The following modules are in abandoned phase and in working draft stage:

- Behavioral Extensions to CSS
- CSS Hyperlink Presentation

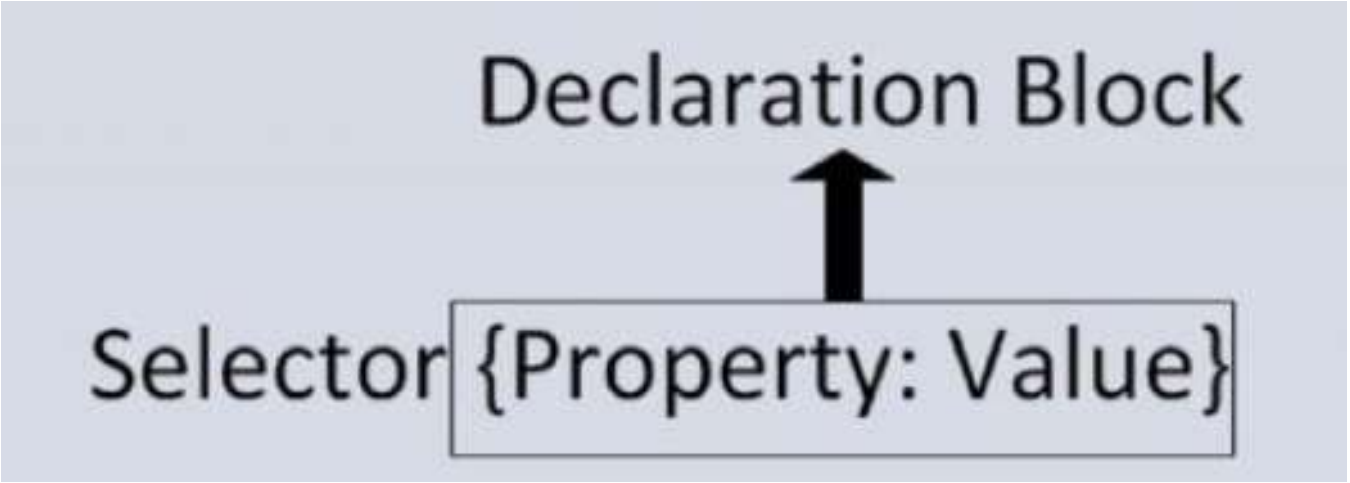
CSS Syntax

The general syntax of CSS consists of three parts namely, selector, property, and value.

A selector is an HTML element for which you want to specify the style or the formatting instruction. A property of a selected element is a CSS property that specifies the type of the style to be applied to the selector. CSS allows controlling the appearance of the content by providing various properties. These properties include text properties, positioning properties, font properties, color properties, and so on. A value refers to the value of the CSS property. A CSS property can have multiple values. For example, the values of the color property include red, green, yellow, and so on.

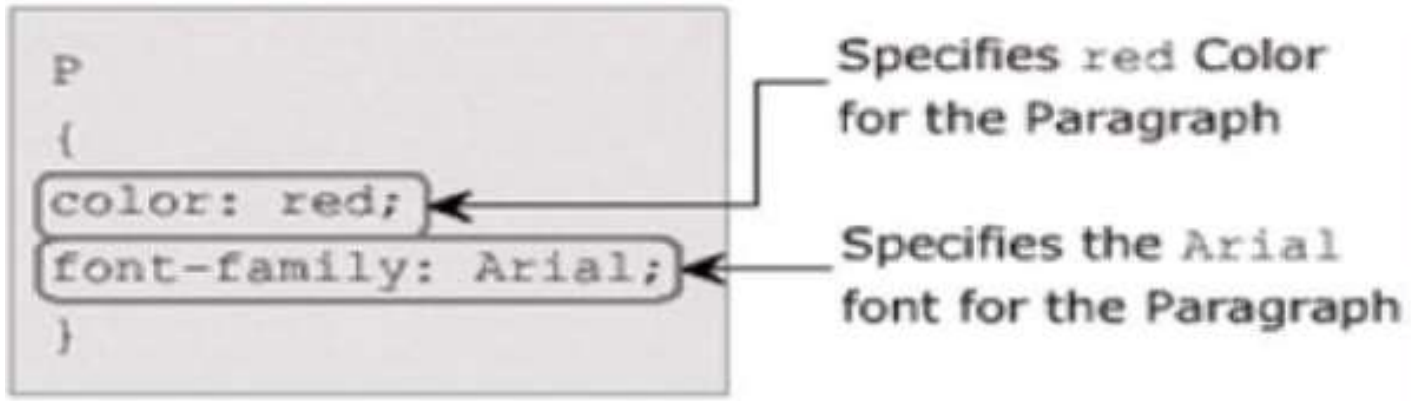
Syntax (contd.)

The property and the value for a selector are separated with a colon (:). They are enclosed within the curly brackets ({ }) that is known as the declaration block. Figure 5.2 shows a CSS syntax.



Syntax (contd.)

You can have various combinations to specify rules for HTML elements. First, you can specify multiple property-value pairs for a selector, which are separated by a semicolon (;) within the declaration block. Second, you can specify multiple selectors for a single property by grouping the selectors. To group the selectors, the selectors are separated by commas followed by a declaration block of properties and values. Third, you can specify properties for multiple selectors. Here, the comma-separated selectors are followed with multiple property-value pairs.



Multiple Declarations for a Single Selector

Example

```
H1,H3,H4
{
font-family: Times
New Roman
}
```

Single Declaration for Multiple Selectors

```
P,BODY
{
color: blue;
font-family: Arial;
}
```

Multiple Declarations for Multiple Selectors

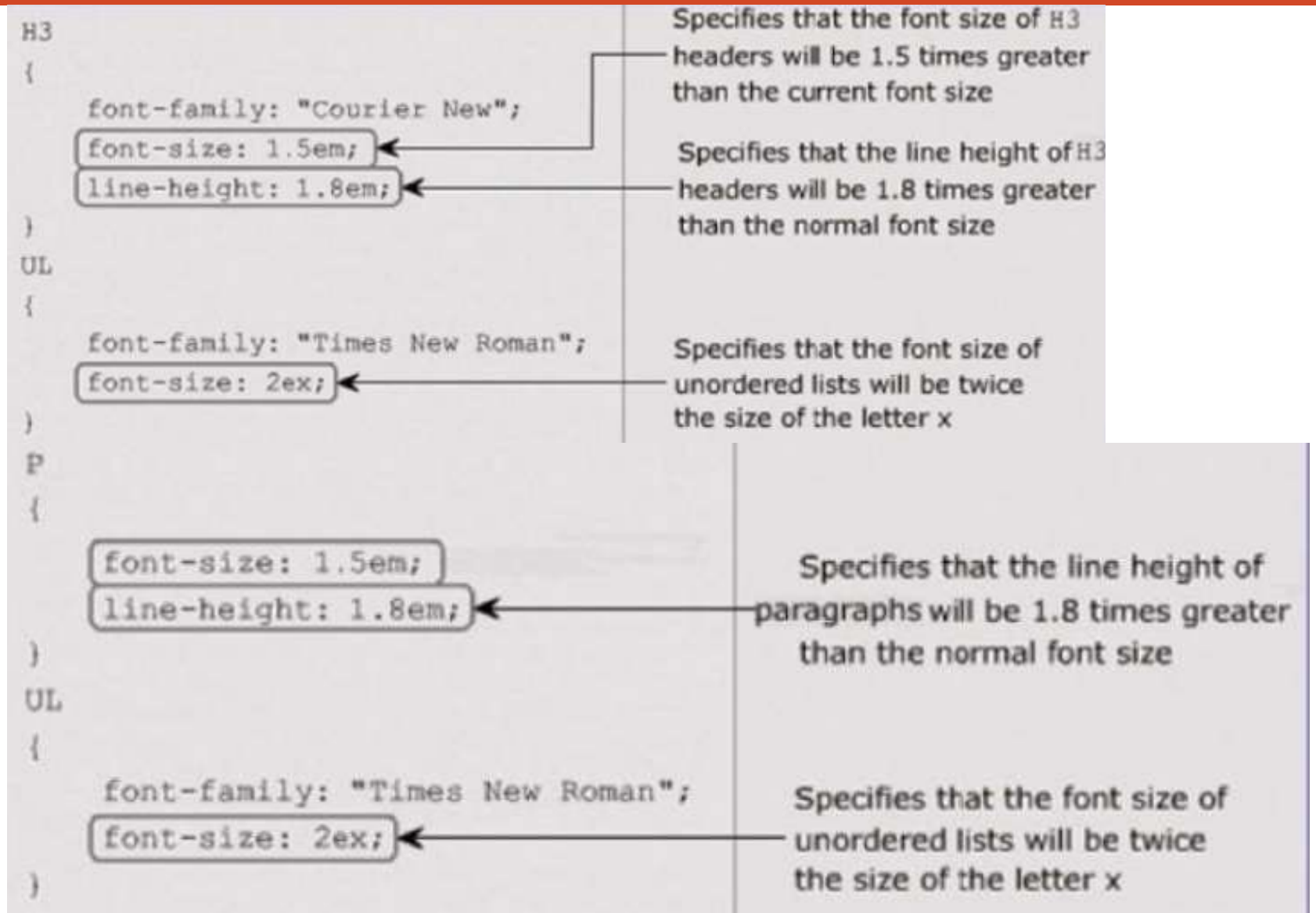
Length Measurement Values

CSS uses various units of measurements for specifying size of the font, width and height of margins, and so on. These units measure the horizontal and vertical length of the content. CSS supports two types of length measurement units namely, relative and absolute.

➔ **Relative**: Relative length specifies the length units related to other length property that are calculated in comparison to a current value.

Relative Length	Description
`em`	Specifies the font size (height) of a particular font. The `em` unit is relative to the value of the `font-size` property of the selector.
`ex`	Specifies the 'x-height' of a particular font. The 'x-height' value is approximately half the font size or the height of the lowercase letter 'x'.
`px`	Specifies the size in pixels, which is relative to the screen of the device.

Example


H3
{
font-family: "Courier New";
font-size: 1.5em;
line-height: 1.8em;
}
UL
{
font-family: "Times New Roman";
font-size: 2ex;
}
P
{
font-size: 1.5em;
line-height: 1.8em;
}
UL
{
font-family: "Times New Roman";
font-size: 2ex;
}
Specifies that the font size of H3 headers will be 1.5 times greater than the current font size
Specifies that the line height of H3 headers will be 1.8 times greater than the normal font size
Specifies that the font size of unordered lists will be twice the size of the letter x
Specifies that the line height of paragraphs will be 1.8 times greater than the normal font size
Specifies that the font size of unordered lists will be twice the size of the letter x

Absolute Lengths

➔ **Absolute**: Absolute lengths are specified when the Web page designer is aware of the physical properties of the output device. These are specific and fixed values.

Absolute Length	Description
in	Specifies the size in inches, where 1 inch = 2.54 centimeters.
cm	Specifies the size in centimeters.
mm	Specifies the size in millimeters.
pt	Specifies the size in points, where 1 point = 1/72th of an inch.
pc	Specifies the size in picas, where 1 pica = 12 points.

Example

```
OL
{
    font-family: "Times New Roman";
    font-size: 0.5cm;
}
TD
{
    font-size: 0.2in;
}
CAPTION
{
    font-size: 3mm;
}
```

Percentage

➔ **Percentages**: Percentage allows specifying the length of the content, which is relative to another value.

```
H1
 {
     font-size: 120%;
     line-height: 200%;
 }
```

In the figure, the CSS code specifies the styles for the `H1` element. The `font-size` property is set to a value of `120%`. This means that the size of the header will appear `20%` greater than its current size. The `line-height` property is set to the value `200%`. This means that the height of the line will be double the value of the `font-size` property.

Types of style sheets

There are three types of `style` sheets namely, inline, internal or embedded, and external style sheets. An inline style sheet uses the style attribute within an HTML element to specify the style for HTML elements.

An internal style sheet is also included within the HTML document. However, it is defined using the `style` element within the `style` element. The style rules appear in a declaration block for each HTML element under the `style` element. The type attribute of the `style` element specifies the content `type`, which is `text/css`. This means that the content under the `style` element is CSS code. You can specify any combinations of specifying style rules. The style rules specified for an element will be applied to all the sub-elements. Internal style sheets are useful when styles are to be applied to a specific Web page.

Internal/Embedded Styles

Internal styles are placed inside the <head> section of a particular Web page source code. These styles can be re-used in the same Web page in which they are placed.

```
<head>
  <meta charset="utf-8">
  <title>Sample HTML5 Structure</title>
  <style>
  h1,h2{
  margin:0px;
  font-size:1.5em;
  }
  footer{
  background-color:#999;
  text-align:center;
  }
  </style>
</head>
```

In figure 5.7, inside the <style> tag, CSS styles for <h1>, <h2>, and <footer> tags are defined. This can be re-used in the same Web page multiple times.

Inline styles

Inline styles are placed directly inside an HTML element. A Web designer cannot use the style builder to create an inline style. Inline style cannot be reused at any point of time in a Web page.

```
<p style="font-size: 14px; color: purple;"></p>
```

Note - An inline style sheet holds the highest priority, which means that the style specified for the same element in any other style sheet will be ignored. The browser-specific styles are applied to a Web page when there are no styles specified for a Web page.

External Style Sheet

An external CSS is defined in a separate file and is saved with the `.css` extension. It provides the benefit of reusability by implementing common style rules for multiple HTML pages. Hence, external CSS are widely used to provide a consistent look across the Web pages of a Web site.

```
BODY
{
        background-color: gray;
        font-family: arial;
        font-style: italic;
}
```

Example

```
<!DOCTYPE HTML>
<HTML>
<HEAD>
<LINK rel="stylesheet" type="text/css" href="body.css"/>
<TITLE>Webex e-Server</TITLE>
</HEAD>
<BODY>
This is the fastest web server..!!
</BODY>
</HTML>
```

`<LINK`

Specifies that the HTML page is linked to another object.

`rel="stylesheet"`

Specifies that the linked object is a style sheet.

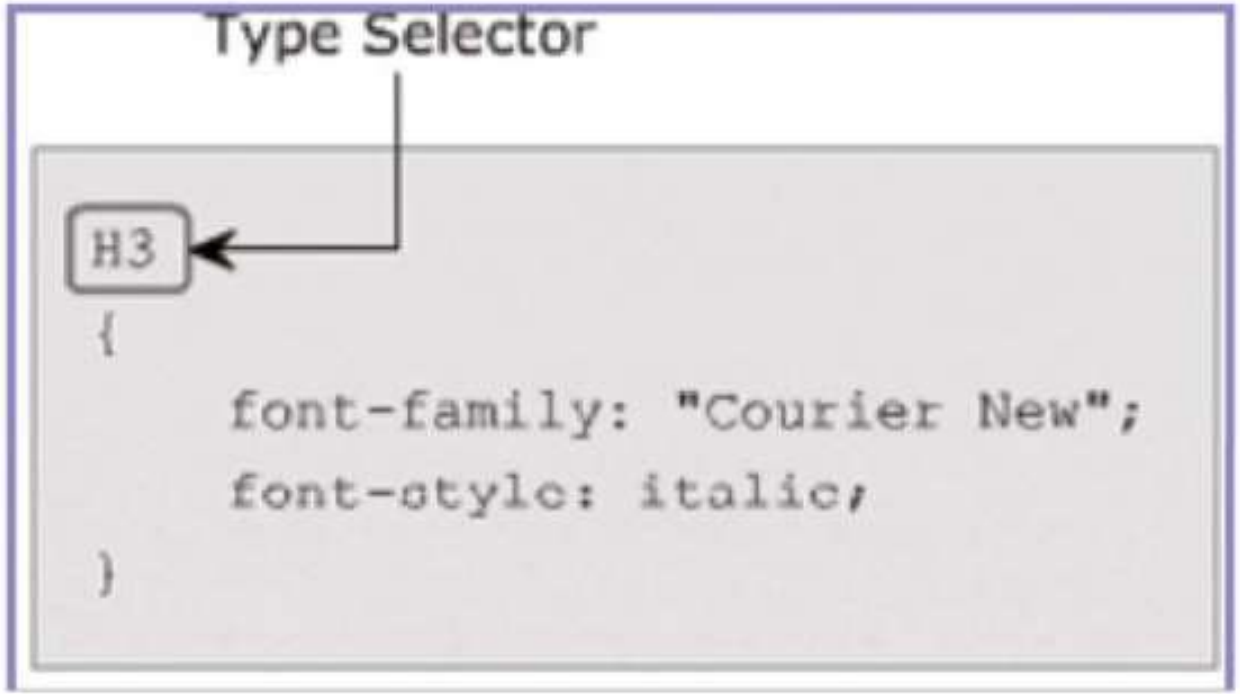
Selectors

Selectors refer to the HTML elements with the styles the users want to apply to them. The three different types of CSS selectors are as follows:

- ➔ Type selector
- ➔ Class selector
- ➔ ID selector
- ➔ Universal selector

Type Selector

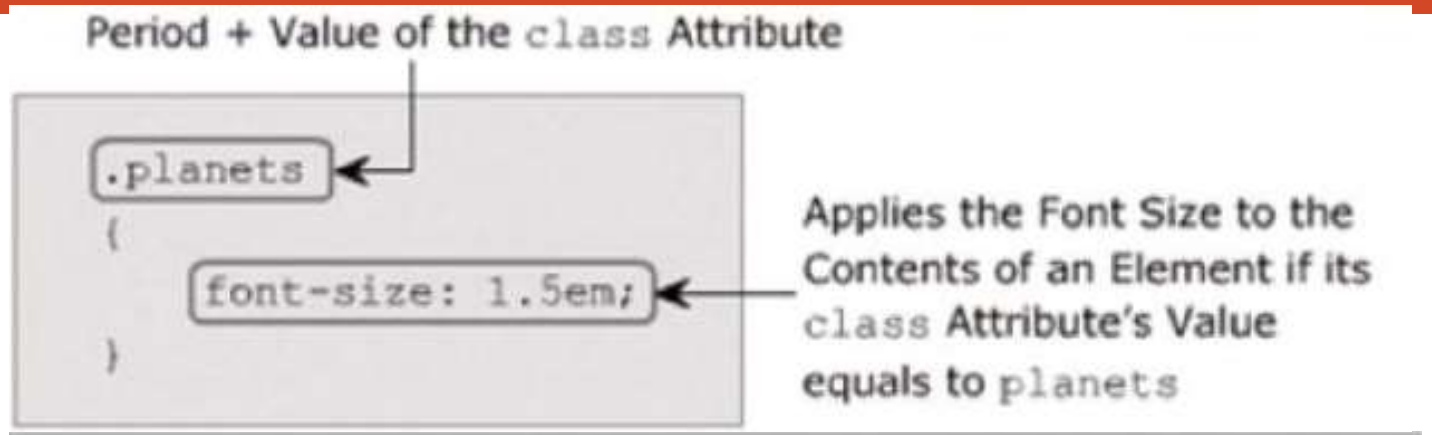
A type selector simply specifies the element name along with the styles to be applied to that element. This results in applying the specified styles to all the occurrence of that element in a Web page. Here, the styles are specified only once for an HTML element and are applied to all the occurrences of that element. Figure 5.10 shows an example of type selector.



Class Selector

A class selector matches elements, whose `class` attribute is set in an HTML page and applies styles to the content of all those elements. For example, if there are `span` and `div` elements in a Web page with their `class` attributes set, the style specified for the class selector will be applied to both the elements. A class selector starts with a period followed by the value of the class attribute. Figure 5.11 shows an example of class selector.

Example




```
<!DOCTYPE HTML>
<HTML>
<HEAD>
<LINK rel="stylesheet" type="text/css" href="Class.css"/>
…
</HEAD>
<BODY>
<DIV class="planets">
This is my first HTML <span class="planets">page</SPAN>.
</DIV>
</BODY>
</HTML>
```

In figure 5.11, the style will be applied to both the places where the values of the class attribute have been set to **planets**. If the style is required to be applied to only the element then class element would be used as shown in Code Snippet 2.

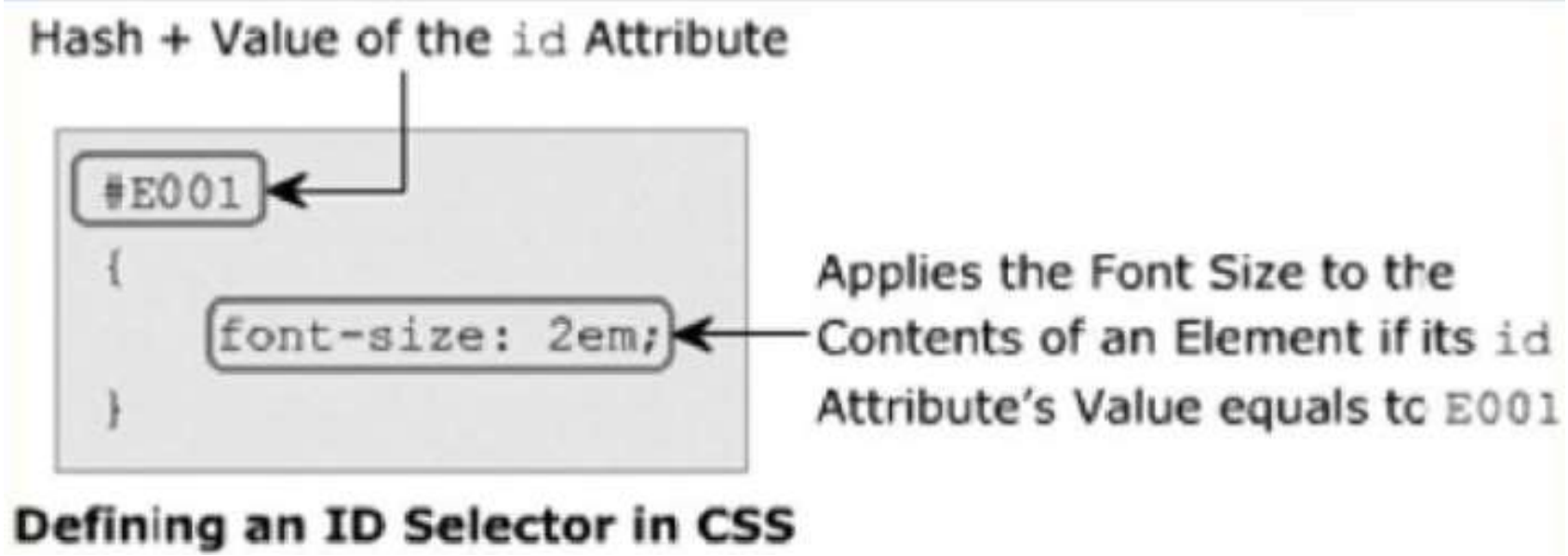
Code Snippet 2:

```
div.planets
     {
       font-size:1.5em;
     }
```

ID Selector

An ID selector matches an element whose id attribute is set in an HTML page and applies styles to the content of that element. The ID selector specifies styles for an element whose `id` attribute is set to a unique value.

An ID selector starts with the hash symbol (#) followed by the id attribute's value and the declaration block. Figure 5.12 shows an example of ID selector.



Defining an ID Selector in CSS

Universal Selector

The universal selector can be applied to all elements in the document. This means that it applies the specified styles to the content of all the elements. It is represented by an asterisk (*) sign. For example, universal selector is used define the font family for all the elements as shown in Code Snippet 3.

Code Snippet 3:

```
* {
   font-family: Verdana, Calibri, sans-serif;
  }
```

Generic Cascading Order

Consider a scenario where you have multiple style sheets defined for an HTML page. These style sheets might have various selectors and multiple styles defined for an HTML element. Therefore, W3C has defined some rules for applying styles to an HTML element. These rules are as follows:

- ➔ Gather all the styles that are to be applied to an element.

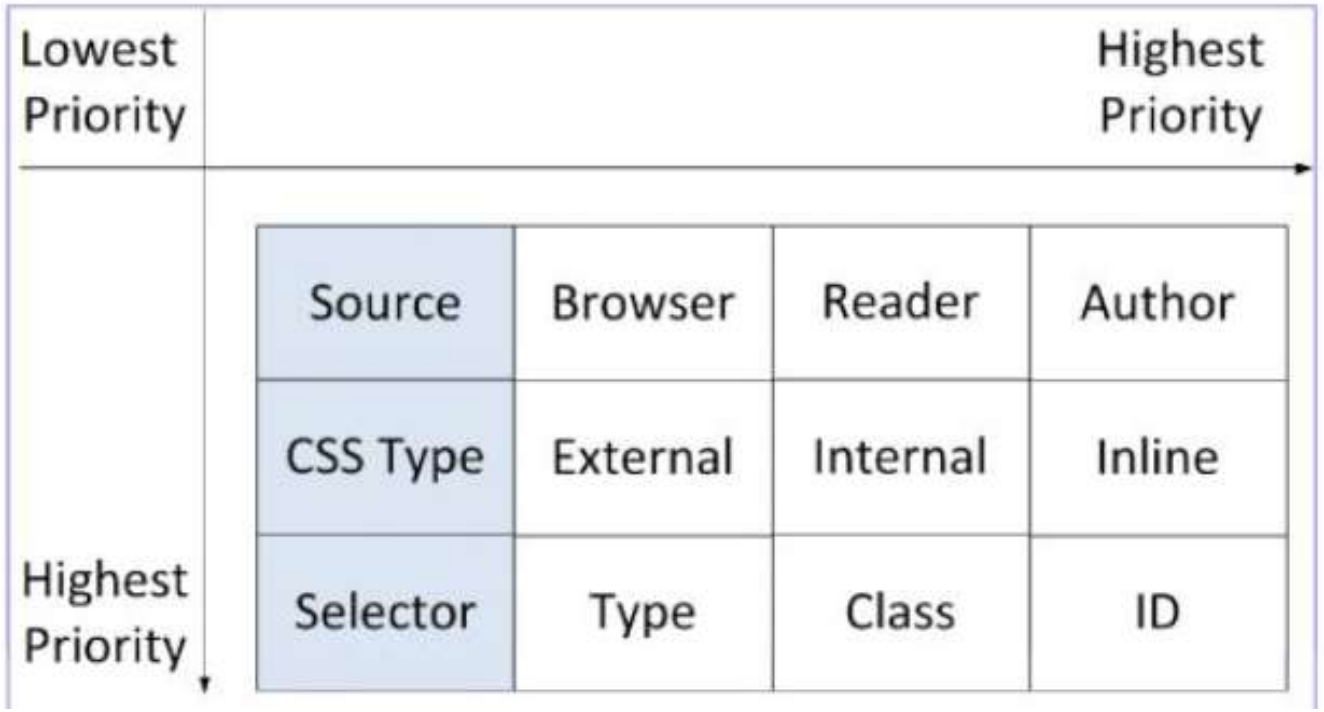
- ➔ Sort the declarations by the source and type of style sheet. The source specifies the origin from where the styles are rendered.

The highest priority is given to the external style sheet defined by an author. The next priority is of the reader, which can be a software that reads the content (screen reader software), and the last priority is of the browser.

Generic Cascading Order

- Sort the declarations by the priority of a selector, where the ID selector has the highest priority.

- Sort the declaration according to the specified order.

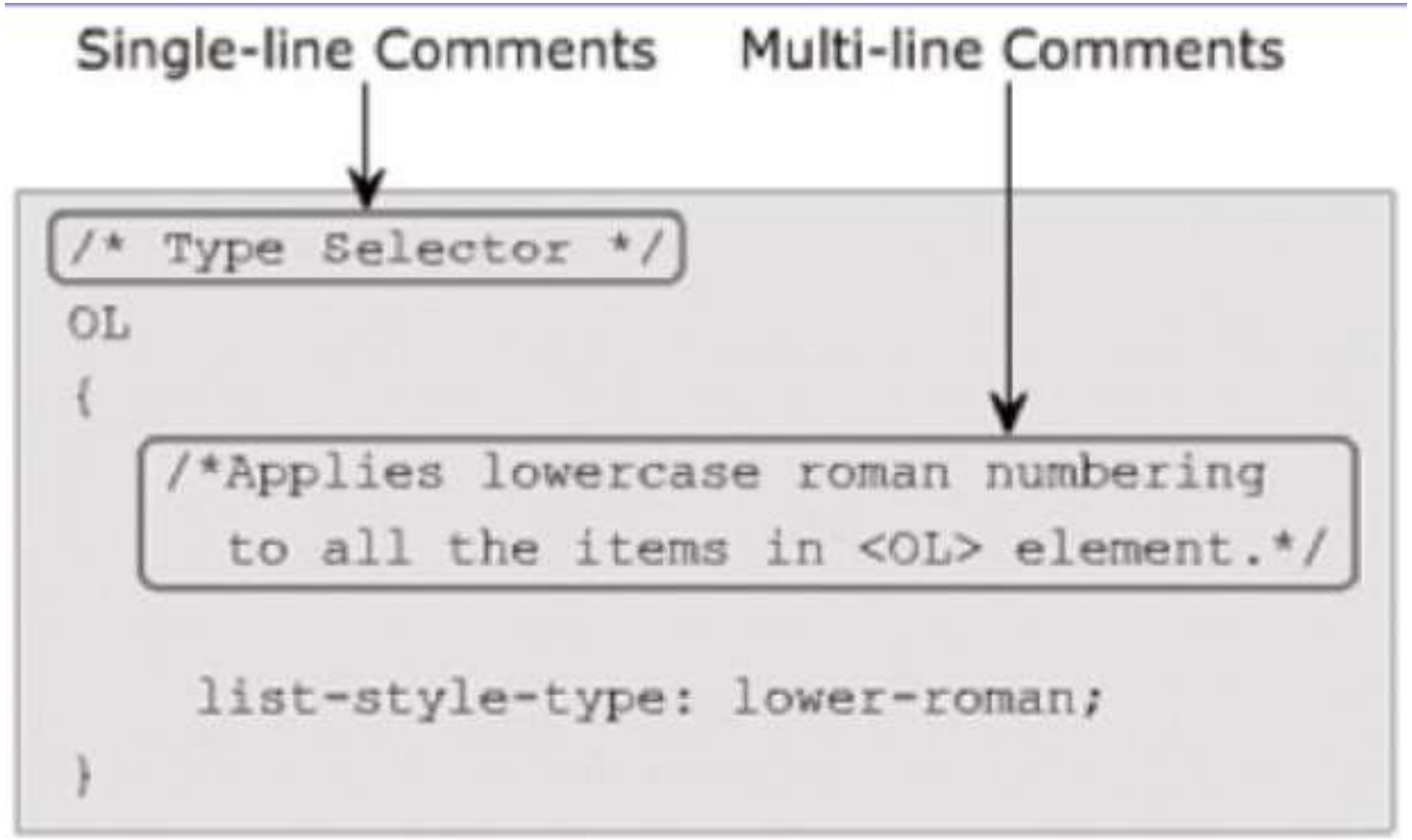


Comments

A comment refers to the descriptive text that allows a Web page designer to provide information about the CSS code. Comments make the program more readable and help the designer to explain the styles specified for elements. This is helpful when other Web designers analyze the CSS code.

The browser can identify comments as they are marked with special characters, which are '/*' and '*/'. When the browser encounters these symbols, the text within them are ignored and are not displayed in the browser. You can have single-line and multi-line comments in the CSS file.

Example


Single-line Comments
Multi-line Comments
/* Type Selector */
OL
{
/*Applies lowercase roman numbering
to all the items in <OL> element.*/
list-style-type: lower-roman;
}

Grouping and nesting of selectors

In style sheets there are often elements with the same style.

```
h1
{
 color:green;
}
h2
{
 color:green;
}
p
{
 color:green;
}
```

To reduce the code, developers can group selectors. Separate each selector with a comma.

```
h1,h2,p
{
 color:green;
}
```

Grouping and Nesting

It is also possible to apply a style for a selector within a selector.

```
p
{
 color:green;
}
 .changed
{
 background-color:red
}
```

In the code, a style is specified for all the paragraphs and another style is specified for all elements whose class attribute has the value set to changed.

Pseudo Classes

Consider a scenario where a Web site consists of multiple Web pages linked through hyperlinks. Browse through various Web pages by randomly clicking the links within the main page. At times, it might happen that unknowingly the same Web page get open that you have already visited. In such a case, you might feel the need for a mechanism that could differentiate the already visited links from the remaining ones. In CSS, this is possible by using pseudo classes.

Pseudo classes allow the users to apply different styles to the elements such as buttons, hyperlinks, and so on.

Pseudo Class (contd.)

State	Description
active	Defines a different style to an element that is activated by the user.
hover	Defines a different style to an element when the mouse pointer is moved over it.
link	Defines a different style to an unvisited hyperlink.
visited	Defines a different style to the visited hyperlink.

Syntax:

`selector_name:state_name {property: value}` where,

`selector_name`: Is an element name.

`state_name`: Is one of the states of an element.

`property`: Is any CSS property such as color, border, and font.

Selector Name

Selector Name	Description
:link	Is used for selecting all unvisited links
:visited	Is used for selecting all visited links
:active	Is used for selecting the active link
:hover	Is used for selecting links on mouse over
:focus	Is used for selecting the input element which has focus
:first-letter	Is used for selecting the first letter of every <p> element
:first-line	Is used for selecting the first line of every <p> element
:first-child	Is used for selecting every <p> elements that is the first child of its parent
:before	Is used for inserting content before every <p> element
:after	Is used for inserting content after every <p> element
:lang (language)	Is used for selecting every <p> element with a lang attribute value

Example

Pseudo classes specify the styles to be applied on an element depending on its state. In CSS3, a selector can contain multiple pseudo-classes. These pseudo-classes should not be mutually exclusive. For example, the selectors `a:visited:hover` and `a:link:hover` are applicable, but `a:link:visited` is not applicable because `:link` and `:visited` are mutually exclusive selectors. The HTML code creates a form that accepts the customer details and provides a link that allows the user to view the bill as shown in figure 5.15.

Example

```
<!DOCTYPE HTML>
<HTML>
<HEAD>
<TITLE>Bill Payment Form</TITLE>
<LINK rel="Stylesheet" type="text/css" href="Payment.css" />
</HEAD>
<BODY>
<H2>Payment Details</H2>
<FORM method="POST" action="">
<TABLE>
<TR>
<TD>Name:</TD>
<TD><INPUT type="text" /></TD>
</TR>
<TR>
<TD>Payment Mode:</TD>
<TD>
<SELECT>
<OPTION>Select...</OPTION>
<OPTION>Credit Card</OPTION>
<OPTION>Cash</OPTION>
</SELECT>
</TD>
</TR>
<TR>
<TD>Total Amount:</TD>
<TD><INPUT type="text" /></TD>
</TR>
</TABLE>
</FORM>
<A href="printform.html">Click here to view the bill.</A>
</BODY>
</HTML>
```

Example Explained

Specifies the styles for an unvisited link.

```
a:link
{
  color: white;
  background-color: black;
  border: 2px solid white;
}
```

Specifies the styles for a visited link.

```
a:visited
{
  color: white;
  background-color: brown;
  border: 2px solid white;
}
```

Contd.

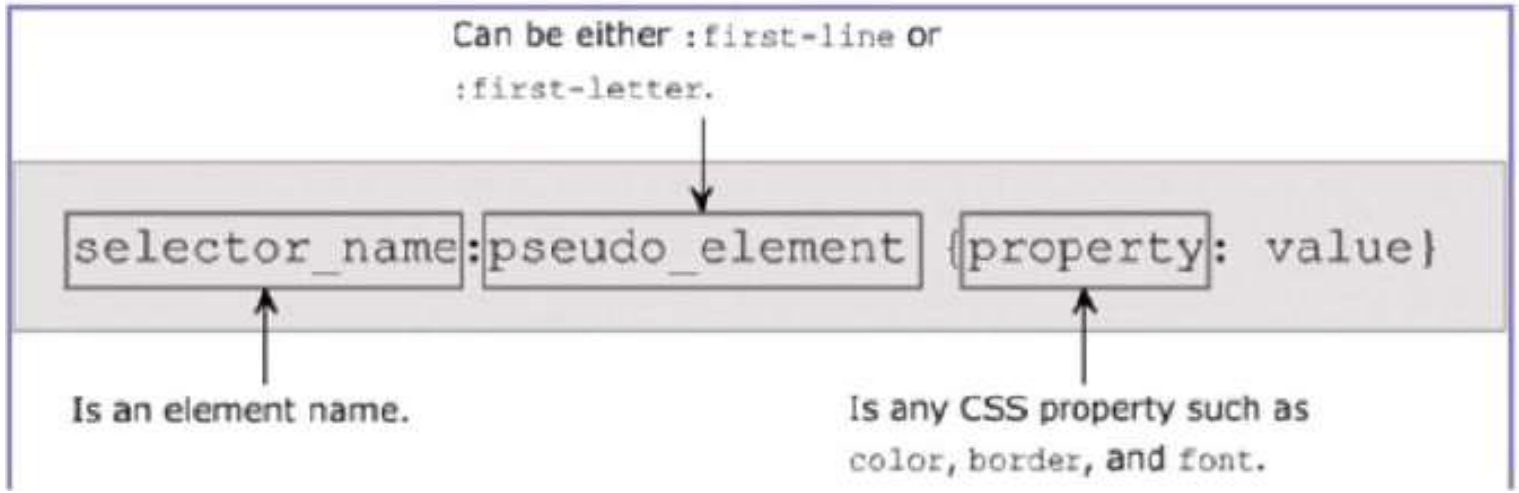
Specifies the styles for a link when a mouse hovers over it.

```
a:hover
{
  color: black;
  background-color: white;
  border: 2px solid black;
}
```

Pseudo Elements

Consider a scenario where you are designing a Web site that explains the important technical terms. While defining such terms, you might feel the need to emphasize more on the first letter by applying different styles. It becomes difficult if you try to apply styles only on the first letter of a line or paragraph. This can be achieved by using the pseudo elements.

Pseudo elements provide you with a flexibility to apply styles to a specific part of the content such as a first letter or first line. This allows you to control the appearance of that specific content without affecting the rest of the content.



Pseudo Elements (contd.)

The :first-line and :first-letter pseudo elements allow you to apply styles to the first line and first letter respectively.

➔ **:first-line**

The :first-line pseudo element allows you to apply styles to the first line.

```
<!DOCTYPE HTML>
<HTML>
<HEAD>
<TITLE>E-Commerce</TITLE>
<LINK rel="Stylesheet" type="text/css" href="E-commerce.css" />
</HEAD>
<BODY>
<H2>E-Commerce</H2>
<P>E-commerce (Electronic commerce) is defined as the sale and
purchase of products over the Internet. E-mail, accounting,
shipment information, and enterprise information reporting are the
some common applications of e-commerce.</P>
</BODY>
</HTML>
```

Pseudo

```
p:first-line
{
font-family: "Tahoma";
font-weight: bolder;
background-color: #FFFFCC;
}
```

Specifies the styles to be applied to the first line of the paragraph content.

→ **:first-letter**

The `:first-letter` pseudo element allows you to apply styles to the first letter.

Pseudo

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Physical Chemistry</title>
    <link rel="stylesheet" type="text/css" href="Cefine.css"></link>
  </head>
  <body>
<h2>Physical Chemistry</h2>
<p>Physical chemistry is a branch of chemistry that analyzes the physical
chemicals.</p>
```

```
p:first-letter
{
 font-family:fantasy;
 font-size:xx-large;
 font-weight:bold;
 }
```

Physical Chemistry

Physical chemistry is a branch of chemistry that analyzes the physical properties of chemicals.

Styles to Hyperlink

CSS can be used to change the appearance and behavior of hyperlinks. To do this, use the following selectors/pseudo-classes:

- `a`
- `a:link`
- `a:visited`
- `a:hover`
- `a:active`

Styles (contd.)

This selectors/pseudo classes represent the 'anchor' element (specified using the HTML 'a' tag) and its various states.

There are two other ways to assign hyperlink styles. They are as follows:

1. Div specific
2. Link specific

ID Specific

A hyperlink styles can be created and assigned to a specific `div`. This will have all the hyperlinks present within the div to follow the specified rules. It is irrelevant if the div is an (#) `id` or (.) `class`.

```
#navone a:link {
      text-decoration: underline;
      color: #005;
      background: transparent;
}
#navone a:visited {
text-decoration: none;
      color: #FFA500;
      background: transparent;
}
```

Class Specific

Specific styling can be assigned to a specific type of hyperlink. This is achieved by creating the style rules in the CSS. For this type of hyperlink styling, a class is used generally than an id. A point to note that an id can only be used once on a page whereas a class can be used multiple times as required.

```
<!DOCTYPE html>
<html lang="en">
 <head>
  <meta charset="utf-8">
<style>
.navone:link {
      text-decoration: underline;
      color: #FFF;
      background: #008;
      font-size:30px;
}
.navone:visited {
      text-decoration: none;
      color: #FFF;
      background: #06a;
}
.navone:hover {
text-decoration: none;
      color: #FFF;
      background: #000;
}
</style>
 </head>
 <body>
<a href="6.html" class="navone">LinkText</a>
 </body>
</html>
```

Quiz

1. Which of the following statements are true for class selector?

(A)	It starts with a period followed by the value of the class attribute
(B)	It matches elements and applies the style to the content of those elements whose class attribute is same
(C)	It specifies the element name along with the style
(D)	It starts with a hash symbol followed by the value of the class attribute

2. Absolute lengths are specified when the Web page designer is aware of the _____ of the output device.

(A)	Text properties	(C)	Image properties
(B)	Physical properties	(D)	Positioning properties

3. Which of these options represent valid style sheets?

(A)	Vertical	(C)	Horizontal
(B)	Inline	(D)	Embedded

4. Which of these options are valid selectors?

(A)	ID	(C)	External
(B)	Inline	(D)	Class

Problem to try

- Adding Inline and Internal Styles
- Adding External Styles
- Adding ID, Class, and Universal Selectors