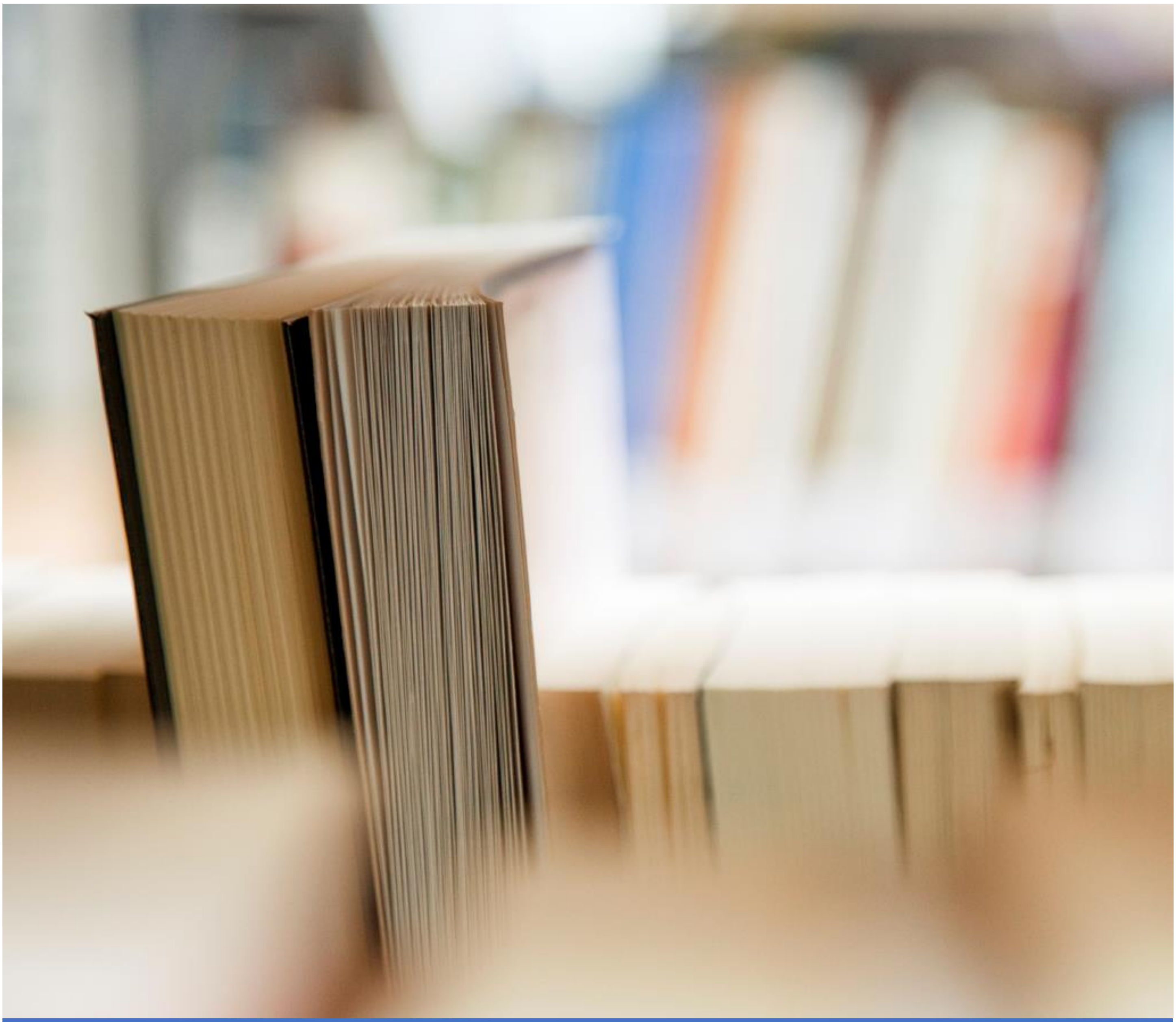

Css Notes

Santosh Kumar 12/14/22 MCA and PGDCA

What is CSS?

Cascading Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.

CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, as well as a variety of other effects.

CSS is easy to learn and understand but it provides a powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML or XHTML.

Advantages of CSS

1. **CSS saves time** - You can write CSS once and then reuse the same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many web pages as you want.
2. **Pages load faster** - If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So, less code means faster download times.
3. **Easy maintenance** - To make a global change, simply change the style, and all the elements in all the web pages will be updated automatically.
4. **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.
5. **Multiple Device Compatibility** - Style sheets allow content to be optimized for more than one type of device. By using the same HTML document, different versions of a website can be presented for handheld devices such as PDAs and cellphones or for printing.
6. **Global web standards** – Now HTML attributes are being deprecated and it is being recommended to use CSS. So it’s a good idea to start using CSS in all the HTML pages to make them compatible with future browsers.

CSS Syntax

A CSS comprises of style rules that are interpreted by the browser and then applied to the corresponding elements in your document. A style rule is made of three parts:

- ✓ **Selector**: A selector is an HTML tag at which a style will be applied. This could be any tag like <h1> or <table> etc.
- ✓ **Property**: A property is a type of attribute of HTML tag. Put simply, all the HTML attributes are converted into CSS properties. They could be color, border, etc.
- ✓ **Value**: Values are assigned to properties. For example, color property can have the value either red or #F1F1F1 etc.

You can put CSS Style Rule Syntax as follows:

```
selector { property : value }
```

CSS Selector

Selectors are the names given to styles in internal and external style sheets. The style rules associated with that selector will be applied to the elements that match the selector pattern.

Selectors are one of the most important aspects of CSS as they allow you to target specific elements on your web page in various ways so that they can be styled.

Types of selectors:

1. Universal Selector
2. Element Type Selectors
3. Id Selectors
4. Class Selectors
5. Descendant Selectors
6. Child Selectors
7. Grouping Selectors

1) Universal Selector

Universal selector, denoted by an asterisk (*), matches every single element on the page.

The universal selector may be omitted if other conditions exist on the element. This selector is often used to remove the default margins and paddings from the elements for quick testing purpose.

Example:

```
        * {
                margin: 0;
                padding: 0;
          }
```

2) Element Type Selectors

Element type selector matches all instance of the element in the document with the corresponding element type name.

Example:

```
        p {
                color: blue;
          }
```

The style rules inside the p selector will be applied on every <p> element (or paragraph) in the document and color it blue, regardless of their position in the document tree.

3) Id Selectors

The id selector is used to define style rules for a single or unique element.

The id selector is defined with a hash sign (#) immediately followed by the id value.

Example:

```
                        #mystyle {
                        color: red;
                        }
```

This style rule renders the text of an element in red, whose id attribute is set to mystyle.

4) Class Selectors

The class selectors can be used to select any HTML element that has a class attribute. All the elements having that class will be formatted according to the defined rule.

The class selector is defined with a period sign (.) immediately followed by the class value.

Example:

```
.blue {
        color: blue;
}
```

The above style rules renders the text in blue of every element in the document that has class attribute set to blue.

You can make it a bit more particular. For example:

```
p.blue {
        color: blue;
}
```

The style rule inside the selector p.blue renders the text in blue of only those <p> elements that has class attribute set to blue, and has no effect on other paragraphs.

5) Descendant Selectors

You can use these selectors when you need to select an element that is the descendant of another element.

For example, if you want to target only those anchors that are contained within an unordered list, rather than targeting all anchor elements.

Example:

```
ul li a {
        text-decoration : none;
        color : green;
}
```

6) Child Selectors

A child selector is used to select only those elements that are the direct children of some element.

A child selector is made up of two or more selectors separated by a greater than symbol (>). You can use this selector, for instance, to select the first level of list elements inside a nested list that has more than one level.

For Example

```
ul > li {
        list-style: square;
}
ul > li ol {
        list-style: none;
}
```

The style rule inside the selector ul > li applied to only those <li> elements that are direct children of the <ul> elements, and has no effect on other list elements.

7) Grouping Selectors

Often several selectors in a style sheet share the same style rules declarations. You can group them into a comma-separated list to minimize the code in your style sheet. It also prevents you from repeating the same style rules over and over again.

h1 {
  font-size: 36px;
  font-weight: normal;
}
h2 {
  font-size: 28px;
  font-weight: normal;
}
h3 {
  font-size: 22px;
  font-weight: normal;
}
	h1, h2, h3 {
	  font-weight: normal;
	}
	h1 {
	  font-size: 36px;
	}
	h2 {
	  font-size: 28px;
	}
	h3 {
	  font-size: 22px;
	}

Applying CSS

There are three ways to associate styles with your HTML document.

1) In-line CSS
2) Internal CSS
3) External CSS

In-line CSS

In-line styles are plonked straight into the HTML tags using the style attribute.

Syntax:

```
<element style=" …..style rules….. ">
```

Example

```
<p style="color : red">This is my CSS Tag</p>
```

This will show the text of paragraph in red.

Internal CSS

Embedded, or internal styles are used for the whole page. Inside the head tags, the style tags surround all of the styles for the page.

Syntax:

```
<style type="text/css" media="...">
        Style Rules
        ............
</style>
```

Attributes associated with <style> elements are:

Attribute	Value	Description
Type	Text/css	Specifies the style sheet language as a content-type (MIME type). This is a required attribute.
media	screen tv print all	Specifies the device, the document will be displayed on. Default value is all. This is an optional attribute.

External CSS

External styles are used for the whole, multiple-page website.

An external style sheet is a separate text file with .css extension. You define all the Style rules within this text file and then you can include this file in any HTML document using <link> element.

Here is the generic syntax of including external CSS file:

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

Attributes associated with <style> elements are:

Attributes	**Value**	**Description**
type	text/css	Specifies the style sheet language as a content-type (MIME type). This attribute is required.
href	URL	Specifies the style sheet file having Style rules. This attribute is a required.
Rel	stylesheet	Specifies the relationship between the current document and the linked document

CSS Comments

Many times, you may need to put additional comments in your style sheet blocks. So, it is very easy to comment any part in the style sheet. You can simply put your comments inside /*.....this is a comment in style sheet.....*/.

Example:

```
/* This is an external style sheet file */
h1, h2, h3 {
color: #36C;
font-weight: normal;
text-transform: lowercase;
}
/* end of style rules. */
```

CSS Properties

CSS Text Properties

CSS provides several properties that allows you to define various text styles such as color, alignment, spacing, decoration, transformation, etc. very easily and effectively.

The commonly used text properties are: text-align, text-decoration, text-transform, text-indent, line-height, letter-spacing, word-spacing, and more. These properties give you precise control over the visual appearance of the characters, words, spaces, and so on.

Property	**Description**	**Values**
color	Sets the color of a text	RGB, hex, keyword
line-height	Sets the distance between lines	normal, *number, length, %*
letter-spacing	Increase or decrease the space between characters	normal, *length*
text-align	Aligns the text in an element	left, right, center, justify
text-decoration	Adds decoration to text	none, underline, overline, line-through
word-spacing	Increase or decrease the space between words	normal, *length*
text-transform	Controls the letters in an element	none, capitalize, uppercase, lowercase
text-shadow	apply the shadow effects on text	offset-x offset-y blur-radius color

CSS Background Properties

Background plays an important role in the visual presentation of a web page.

CSS provide several properties for styling the background of an element, including coloring the background, placing images in the background and managing their positioning, etc.

Property	Description	Values
background-color	used to set the background color of an element.	RGB, HEX and color name
background-image	set an image as a background of an HTML element.	url(‘image-url’)
background-repeat	allows you to control how a background image is repeated or tiled in the background of an element.	repeat, repeat-x, repeat-y, no-repeat
background-size	Sets the width and height of the background image	Auto, length, cover, percentage

CSS Font Properties

CSS provide several properties for styling the font of the text, including changing their face, controlling their size and boldness, managing variant, and so on.

Property	Description	Values
font-family	specify the font to be used to render the text.	serif, sans-serif, monospace, cursive and fantasy
font-style	set the font face style for the text content of an element	normal, italic or oblique
font-weight	specifies the weight or boldness of the font	normal, bold
font-size	set the size of font for the text content of an element.	xx-small, x-small, small, medium, large, x-large, xx-large or px
font-variant	allows the text to be displayed in a special small-caps variation	Normal, small-caps

CSS width Properties

The width property in CSS is used to set the width to the text, images. The width can be assign to the text and images in the form of pixels(px), percentage(%), centimetre(cm) etc.

The width property does not contain padding, borders, or margins. The width property is overridden by the *min-width* & *max-width* properties.

Syntax:

```
width: auto | value | initial | inherit;
```

CSS Height Properties

The height property is used to set the height of an element. The height property does not contain padding and margin and border of element.

Syntax:

```
height: auto | length | initial | inherit;
```

CSS Border Properties

CSS border properties allow you to define the border area of an element's box.

Borders appear directly between the margin and padding of an element. The border can either be a predefined style like, solid line, dotted line, double line, etc.

Property	Description	Values
border-style	sets the style of a box's border	none, hidden, solid, dashed, dotted, double, inset, outset, groove, and ridge
border-width	specifies the width of the border area	pixcel
border-color	specifies the color of the border area	Color-name, Hax, RGB
border	shorthand property for setting one or more of the individual border properties border-width, border-style and border-color in a single rule.	Pixcel
border-radius	used to create rounded corners	Pixcel

CSS Padding Properties

The CSS padding properties allow you to set the spacing between the content of an element and its border.

Property	Description	Values
padding	Specified padding for all sides	auto, length, percentage
padding-left	specifies the padding from left side only	auto, length, percentage
padding-right	specifies the padding from right side only	auto, length, percentage
padding-top	specifies the padding from top side only	auto, length, percentage
padding-bottom	specifies the padding from bottom side only	auto, length, percentage

CSS Margin Properties

The CSS margin properties allow you to set the spacing around the border of an element's box.

Property	Description	Values
margin	Specified margin for all sides	auto, length, percentage
margin-left	specifies the margin on left side only	auto, length, percentage
margin-right	specifies the margin on right side only	auto, length, percentage
margin-top	specifies the margin on top side only	auto, length, percentage
margin-bottom	specifies the margin on bottom side only	auto, length, percentage

CSS Display Property

Display property specifies the display behavior (the type of rendering box) of an element.

Syntax:

display: value;

Property Values:

Value	Description
inline	Displays an element as an inline element (like <span>). Any height and width properties will have no effect
block	Displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width
contents	Makes the container disappear, making the child elements children of the element the next level up in the DOM
grid	Displays an element as a block-level grid container
inline-block	Displays an element as an inline-level block container. The element itself is formatted as an inline element, but you can apply height and width values
list-item	Let the element behave like a <li> element
none	The element is completely removed

CSS float property

float property specifies how an element should float on webpage.

Syntax:

float: none | left | right | initial | inherit;

CSS Position Property

The CSS position property is used to set position for an element. it is also used to place an element behind another and also useful for scripted animation effect.

You can position an element using the top, bottom, left and right properties. These properties can be used only after position property is set first. A position element's computed position property is relative, absolute, fixed or sticky.

Let's have a look at following CSS positioning:

1) CSS Static Positioning
2) CSS Fixed Positioning
3) CSS Relative Positioning
4) CSS Absolute Positioning

CSS Static Positioning

A static positioned element is always positioned according to the normal flow of the page. HTML elements are positioned static by default. Static positioned elements are not affected by the top, bottom, left, right, and z-index properties.

CSS Fixed Positioning

The fixed positioning property helps to put the text fixed on the browser. This fixed test is positioned relative to the browser window, and doesn't move even you scroll the window.

Example:

```
.box { position: fixed; top: 200px; left: 100px; }
```

CSS Relative Positioning

A relative positioned element is positioned relative to its normal position.

In the relative positioning scheme the element's box position is calculated according to the normal flow. Then the box is shifted from this normal position according to the properties — top or bottom and/or left or right.

Example:

```
.box { position: relative; left: 100px; }
```

CSS Absolute Positioning

The absolute positioning is used to position an element relative to the first parent element that has a position other than static. If no such element is found, the containing block is HTML.

With the absolute positioning, you can place an element anywhere on a page.

Example:

```
.box { position: absolute; top: 200px; left: 100px; }
```