

CSS (Ch#18)

What is CSS?

- CSS is a language that applies to a HTML document and its elements to change the look and feel and is usually stored in separate .css style which can be reused for all web pages.
- A website is made up of HTML plus CSS for appearance.

HTML + CSS = web page
(Content) + (Presentation)

Advantages

- CSS saves time.
- Pages load faster.
- Easy maintenance.
- Superior styles to HTML.
- Multiple device capability.
- Global web standards.

Disadvantages:

- Browser compatibility.

Basic CSS syntax:

- The CSS syntax consists of a set of rules.
- These rules have 3 parts:

Selector & Property : value ;

Selector: A selector is an HTML tag at which style will be applied. This could be any tag like `<h1>`, `<table>` etc.

Property: A property is a type of attribute of HTML tag.

- o All the HTML attributes are converted into CSS properties. They could be color, border, background-color etc.

value:

- o Values are assigned to properties. For example, color property can have value either red.

Example:

```
table { border: 1px; }
```

Comments:

⇒ Comment can be within a pair of `/* ... */` tag like this:

```
/* This is CSS comment */
```

Style Types

In this case a number of different styles are defined in each type have hierarchy of precedence, from low to high.

Default Styles:

⇒ The lowest level of style precedence is the default styling applied by a web browser.

⇒ These styles are created as a fallback for when a web page does not have any styles and they are intended to any generic set of styles.

User styles:

They are supported by modern browsers but implemented differently by each.

⇒ If a user style is assigned that has already been defined as a browser by default, it will override the browser's default setting.

Levels of Stylesheet (less inclusions)

These are three levels of style sheet:

(1) Inline:

- ⇒ Inline style sheet rules will be applied to the content of a single element.
- ⇒ Inline style sheet rules are specified as the values of the style attribute.

⇒ These is the generic syntax:

<Element style="... style rules ...>

⇒ Style rule: is a list of property value pairs

Example:

<h1 style="color: red;> This is inline

</h1>

(2) Document level CSS - The <style> element:

Element:

- ⇒ Document level style sheet rules will be applied to all the elements available in the document.
- ⇒ Document level style sheet rules are specified as the content of <style> element.
- ⇒ This tag is placed inside <head> tags.

Here is the generic syntax:

<head>

<style type="text/css" media="projection, screen">

Style Rules

</style>

</head>

<link href="style.css" type="text/css" media="all" />

⇒ An external style sheet is a separate file with .css extension.

- ⇒ we can define all the style rules within this text file and can be include this file in any HTML document using <link> element.
- ⇒ External style sheets can apply to the bodies of multiple HTML documents.
- ⇒ it is recommended method.
- ⇒ Here is the generic syntax:

<head>

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

Where

- o `rel="stylesheet"` Specifies the relationship of the linking document to the current document.

- o `type = "text/css"` Specifies the stylesheet language as a content-type.
(This is required attribute).

- o `href = "filename.css"` Specifies the style sheet file having style rules.
(This is required attribute).

Example:

⇒ Consider a separate stylesheet file with a name `mystyle.css` having the following rules:

```
h1 { color:blue ; font-size : 34pt; }
```

⇒ Now this file `mystyle.css` can be included in any HTML document as follows:

```
<head>
```

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

Q. What is Selector? Describe different types of selector?

Def:

In CSS, a Selector is the target element to which each CSS rule is applied.

The means by which we can access one or more elements is called selection and the part of a CSS rule that does this is known as Selector.

Example:

The different types of Selectors are follows:

- o Type Selectors
- o Class Selectors
- o ID Selectors
- o Descendent Selectors
- o Child Selectors
- o Universal Selectors
- o Adjacent Selectors
- o Sibling

o Attribute Selectors

o Pseudo classes & Pseudo-elements

(i) The type selector:

- o it is single element name.
- o it works on types of HTML elements such as `<P>` or `<i>` etc.
- o In this case the property values in the rule apply to all occurrences of the named elements.

Example:

```
h1 { font-size: 24pt; }  
p { text-align: justify; }
```

(ii) The universal selector:

- ⇒ The universal selector, denoted by `<all>`
- an asterisk (*), applies its style to all elements in the document.

Ex:

```
* { color: red; }
```

- o makes all elements in the document red.

ID Selector:

```
o ul li b { color: blue; }
```

- o ID Selector is an individually identified selector to which a specific style is declared.

The Descendant Selector:

The Descendant Selectors are used to select elements that are descendants of another element in the document tree. Descendant Selectors let us apply styles to elements that are contained within other elements. Descendant Selectors can continue nesting indefinitely.

Example:-

The following rule sets all text within `...` tags to red, but only if they occur within `<P>...</P>` tags alike. The `<P>Hello</P>`

```
o p b { color: red; }
```

- o ID Selectors are created by a character '#' followed by the selector name.

Example:

```
<div id='myid'> Hello </div>
#myid { text-decoration: underline; }
```

if

```
<span id='myid'> Today </span>
```

Because id must be unique in this case under-line rule will be apply to only the first occurrence of 'myid'.
for both

```
span#myid { text-decoration: underline; }
div#myid { text-decoration: underline; }
```

- o Not preferred method.

(5) The class Selector:

- o Class Selectors are used to allow different occurrences of the same tag to use different style specifications
- o Used to select any XML element that has a class attribute.
- o A class Selector is indicated by Period(.) .

when there are a number of elements on a page that you want to share the same styling, we can assign them all the same class name (like ``) then create a single rule to modify all these elements at once.

Example:

```
.myclass { margin-left: 10px; }
```

```
<head>
```

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

```
</head>
```

```
...
```

```
<body>
```

```
<p class="normal"> text </p>
```

```
</body>
```

(6) Child Selectors:

- o A child Selector is used to select an element that is a direct child of another element.
- o It is similar to the descendant selector but is more restrictive about when the style will be applied, by selecting

only those elements that are directly child of complex elements

Example:

The following code uses a descendant selector that will change any bold text within a paragraph to red, even if the bold text itself is within italics

```
<P><i> <b>Hello </b> there </i></P></P>
```

```
P b { color: red; }
```

- o The word Hello will display red.
- o When this general behavior is not required, a child selector can be used to narrow the scope of the selector.
- o For example:

The following child selector will set bold text to red only if the element is a direct child of a paragraph and it is not contained within another element.

```
P>b { color: red; }
```

Now, hello world will not change.

example 2:

```
<style>
```

```
ol>li { font-size: 10pt; }
```

```
</style>
```

```
/
```

The attribute Selectors:

Attribute selectors are used to select elements based on their attribute and attribute values.

Using this type of selector can save time having to use IDs and classes to refer to them, means we can directly reference attribute.

[attribute = value]

Example: [attribute = value]

```
[type = "submit"] { width: 100px; }
```

[attribute \$ = value]

```
img [src = "small.gif"] { border-width: 1px; }
```

[attribute A = value]

```
border-style: dotted; }
```

[attribute A = value]

Adjacent sibling Selectors:

It will select siblings immediately following an element.

e.g.

```
em + strong { font-style: italic; }
```

Pseudo classes:

Pseudo classes are used to add special effects to some selectors.

Pseudo class styles apply when something happens earlier than because

the target element simply exists.

Syntax:

Selectors: Pseudo-class, Property: value;}

=> The commonly used Pseudo classes are:

:first-child, :link, :visited, :active
, hover:, focus:

Example:

a:visited {color: red;}

a:active {color: green;}

a:link {color: blue;}

input: hover {color: red;}

input: focus {color: green;}

10) Selecting by Group:

O A rule apply to more than one element, class or any other type of selector at the same time by separating the selectors with

Commas

Ex:

P, #idname, .classname {border-bottom:
1px dotted orange;}

Measurement

Every CSS property has a value and we can include some measurements like cm, mm etc.

Absolute size

Sets the text to a Specified size.

e.g. inches, centimeters, points and so on.

Relative size:

Sets the size relative to surrounding elements.

e.g. percentages and em units.

Unit	Description	Example
em	1em is equal to the current font size.	PI font-size: 1.5em;
2em	2em means 2-times the size of current font. Used to describe relative dimensions.	
ex	<ul style="list-style-type: none">o x-height (height of the lower case letters)o Also related to the current font sizeo Help to set a width of a box.	PI width: 2ex;

Fonts & Typography

Px	<ul style="list-style-type: none"> o Pixel (a dot on computer screen) o The size of a pixel varies according to the dimensions and pixel depth of the user's monitor. 	PE font-size: 1px;
in	<ul style="list-style-type: none"> o Define measurements in inches. o An inch is the equivalent of 72 points. 	PE font-size: 0.15in;
cm	<ul style="list-style-type: none"> o Define measurements in cm. o One cm is little over 28 points. 	PE font-size: 0.15cm;
mm	<ul style="list-style-type: none"> o Define measurements in mm. o A mm is 1/10 of a cm. 	PE font-size: 0.15mm;
PC	<ul style="list-style-type: none"> o Picas (1pica = 12 pt, 1 inch = 6 picas) - it is typographic measurement 	PE font-size: 12pt;
Pt	<ul style="list-style-type: none"> o Points o 1 pt = 1/72 inch. 	PE font-size: 14pt;
%	<ul style="list-style-type: none"> o Relative to the values o 100% take the current values. 	PE font-size: 200%;

There are four main properties that can style using CSS:

- 1) family
- 2) style
- 3) size
- 4) weight

font-family:

font-family is the type of font used rendering text similar to the font used in the select in ms word.

To change the fonts we can use font-family property with font names comma separated values.

Ex:

P { font-family: verdana, Arial, sans-serif; }

when a font name is made up

two or more words, you must enclose the name in quotation marks like

Ex:

P { font-family: "Times new Roman", serif; }

In CSS there are two types of font names:

family names:

- o generic family
 - ↳ a group of font families with a similar look (like serif or "monospace").
- o font family
 - ↳ a specific font family like "Times New Roman" or "Arial".

generic family	Description	Example
serif	Some lines cut the ends on some chars.	Times new roman, Georgia, Arial, Verdana
sans-serif	do not tiny lines at the end of letters.	Geneva, Corsica new, Andale mono
monospace	All characters have the same width	
cursive	Emulates handwritten appearance.	Comic sans, Apple Chancery, Impact, Western
fantasy	Purely decorative & appropriate for headings.	

a) Font-Style:-

⇒ The font-style property is mostly used to specify whether the letters shape are normal or slanted.

e.g.

```
P { font-style: italic; }
```

Description

- The text is shown normally.
- The text is shown italic.
- The text is slanted.

b) Font Size:

The font-size property specifies the size of text.

```
h1 { font-size: 240%; }  
h2 { font-size: 200%; }  
h3 { font-size: 160%; }  
h4 { font-size: 120%; }
```

c) Font weights:

it specifies the thickness of the font.

normal
Defines normal characters. This is elegant.

bold
Defines thick characters.

oblique
Defines thicker characters.

light
Defines lighter characters.

```
P { font-weight: light; }
```

Text-Properties

(1) Text-align:-

⇒ The text-align property specifies the horizontal alignment of text in an element.

Value	Description
Right	Text is justified to the right.
Left	Text is justified to the left.
Center	Text is centered.
Justified	Text is both right & left justified.

e.g.

```
h2 {text-align: left;}
```

(2) Text-decoration:

⇒ The text-decoration property specifies the decoration added to text.

Value	Description
underline	Adds an underline to the text.
overline	Adds a line on top of the text.
blink	Adds a line through the middle of the text.
line-through	Causes the text to blink.

Ex:

```
h2 {text-decoration: line-through;}
```

text-indent:

The text-indent property specifies the indentation of the first line in a text-block.

e.g. {text-indent: 50px;}

text-transform:

The text-transform property controls capitalization of text.

Value	Description
capitalize	Capitalizes the first letter in a word.
uppercase	Makes the entire word uppercase.
lowercase	Makes the entire word lowercase.
none	No transformation is performed.

e.g. {text-transform: capitalize;}

text-spacing:

We can modify line, word and letter spacing.

- Line-height property > 25%
- Word-spacing property > 30 pixels
- Letter-spacing = 3px

e.g. {

```
line-height: 125%;  
word-spacing: 30px;  
letter-spacing: 3px;
```

CSS Colors

=> Colors can be applied to foreground and background of text and objects by using the color and background properties.

=> The colors specified can be:

- named colors
- hexadecimal RGB triplets
- rgb CSS function.

Format	Syntax	Example
Hex code	#rrrgbb	P {color: #ffff00;}
Short hex code	#rgb	P {color: #667;}
rgb%	rgb(rrr%; ggg%; bbb%)	P {color: rgb(50%, 50%, 50%);}
Rgb absolute	Rgb (rrr, ggg, bbb)	P {color: rgb(0, 0, 255);}
Keyword	black, red, black	P {color: orange;}

Examples

```
body {color: red;}
```

```
h1 {color: #00ffff;}
```

```
P {color: rgb(0, 0, 255);}
```

background-color:

- → color property is used to set the background color.

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

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

gradients: → display smooth transitions between more specified colors.
Instead of using solid background color, we apply a gradient, which will automatically flow from a given initial color to a final color of your choice.
To create a gradient:

- 1) choose where it will begin out, top, bottom, left, right and center.
 - 2) either the start and end color require
 - 3) Apply either linear-gradient or radial gradient rule
 - 4) Also supply rules for all browsers you are targeting.
- Note:

Ex:	browser specific prefixes
background: orange;	=> -moz-
<style>	=> -ms-
!important;	=> -o-
background: linear-gradient(to top, #ff00, #ff50);	=> -webkit-
background: -o-linear-gradient(to top, #ff00, #ff50);	=> -o-

background: orange;

background: linear-gradient(to top, #ff00, #ff50);

background: -o-linear-gradient(to top, #ff00, #ff50);

}

</style>

Syntax: background-image: linear-gradient(
 direction, color-stop-1, color-stop-2, ...);
⇒ radial-gradient (shape size at posn, start-color
 stop-color)

Positioning Elements

⇒ Elements within a web page fall where they are placed in the document, but you can move them about by changing an element's position property from the default of static to one of absolute, relative, or fixed.

Absolute Positioning:

⇒ An element with absolute positioning is removed from the document, and any other elements that are capable will flow into its released space.

⇒ Then element can be positioned anywhere within the document by using top, left, right, bottom properties.

Example: (id)

#Object {

 Position: absolute;

 top: 100px;

 left: 200px;

}

Relative Positioning:

⇒ Likewise, relative object can be moved to the location, it would occupy in the normal

document flow.

e.g:-

#Object {

 Position: relative;

 top: 10px;

 left: 10px;

}

Fixed Positioning:

⇒ Allows to move an object to an absolute location, but only within the current browser's viewport.

⇒ Then, when the document is scrolled the object remains exactly where it has been placed, with the main document scrolling beneath it.

e.g:- #Object {

 Position: fixed;

 top: 0px;

 left: 0px;

}

CSS Comments:

Syntax:

/* */

CSS Box Model

⇒ The term "box model" is about

layout and design.

⇒ The box model is a term used when

referring to the rectangular box placed around every element in the web-page.

⇒ An XHTML element can be considered a box, and so the box model applies to all HTML elements.

⇒ The box model is the specification that defines how a box and its attributes relate to each other.

Margin

Border

Padding

Content

Padding

The padding defines the space between the content and the border.

⇒ The top, right, bottom and left padding can be changed independently using border properties.

Ex:

padding-top: 20px;

padding-bottom: 20px;

Content:

The innermost part of the box is the content, such as "`h1`", "`ul`", "`p`" ... etc. The width and height property defines the width and height of the element.

Ex:
p {
width: 100px;
height: 50px;

These are four different parts of the model:
Content • padding • Borders • margin

padding-right: 50px;

padding-left: 50px;

}

⇒ A shorthand property of padding can also be used to change all padding at once.

Ex:

P{ Padding: 10px 5px 15px 20px; }

(3) Borders:

⇒ The middle layer in the box model is the element border.

⇒ The space used by the border in the box model is the thickness of the border.

⇒ CSS border has following 4-properties:

o border-style o border-color

o border-width o border-individual sides.

⇒ The border style property sets the style of an element's four borders.

Ex:

(1)

P{

border-top-style: dotted;

border-right-style: solid;

border-bottom-style: dotted;

border-left-style: solid;

}

border-style: solid;

border-color: red;

border-width: 5px;

Margin:

The space just outside the border is margin.

The margin is completely invisible.

The top, right, bottom, and left margin can be changed independently using separate properties.

Ex:

P{

margin-top: 10px;

margin-bottom: 10px;

margin-right: 5px;

margin-left: 5px;

}

Short hand property: P{ margin: 8px 5px 15px 10px; }

Questions

1) Which directive would you use to import one style sheet into another?

⇒ To import one style sheet into another we use @import directive, like this:
@import url('style.css');

2) What HTML tag can we use to import a style sheet into a document?

⇒ To import style sheet into a document, we use html link tag, like this:

<link rel='stylesheet' type='text/css' href='style.css'/>

3) Which HTML tag attribute is used to directly embed a style into an element?

⇒ By using style attribute, like this

<div style='color:blue;'>

4) What is difference b/w CSS ID and CSS class?

⇒ CSS ID can be applied to only a single element.

Whereas a class can be applied to multiple elements.

Which characters are used
prefix
(a) ID's (b) class names

(a) characters for ID
(b) characters for class name.

In CSS rule what is the purpose of semicolon?

Used as a separator between declaration.

11) How can you add a comment to style sheet?

/* */

12) Which character is used to CSS to represent an element?

⇒ In CSS, you can match any element by the *universal selector.

13) How can you select a group of different elements in CSS?

⇒ place a comma b/w each element id or class.

14) Given a pair of CSS rules with equal precedence, how can

you make one have greater precedence over the other?

⇒ By appling the !important declaration to it like this:

P{ color: #ff0000 !important; }

→

- o the following four style sheet entries are to control hyperlinks in CSS:
 - o a:link { color: blue; text-decoration none; }
 - o a:active { color: red; }
 - o a:hover { color: green; }
 - o a:visited { color: magenta; }

LESS & Hyperlinks

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

Background - color:

- o background-color property specifies the background color of an element.
- o The background color of a page is set like this:

c.g

body {

background-color: lightblue;

}

LESS & Backgrounds

- ⇒ with CSS, a color is most often specified by
- a valid color name
 - a Hex value
 - an RGB-value

⇒ Background Color of any HTML elements:

e.g.

h1 {

background-color: green;

}

div {

background-color: lightblue;

}

p {

background-color: yellow;

}

CSS background-image:

⇒ The background-image property specifies an image to use as the background of an element.

⇒ By default, the image is repeated so it covers the entire element.

Example:

- Set the background image for a page.

background-image: url("paper.jpg");

The background image can also be set for elements like the `<P>` element:

p {

background-image: url("paper.jpg");

}

CSS background-repeat:

By default, the background-image property sets an image both horizontally and vertically.

⇒ if the image is repeated only horizontally

e.g.
body {

background-repeat: repeat-x;

}

⇒ if the image is repeated only vertically.

e.g.
body {

background-image: url("gradient.png");

background-repeat: repeat-y;

}

⇒ Background image is also specified only once:

e.g.

body {

background-image: url("img-tree.png");

background-repeat: no-repeat;

}

LESS background-Position:-

⇒ The background-position property is used to specify the position of the background image.

⇒ e.g.

following 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;

}

LESS background attachment:-

⇒ The background-attachment property specifies whether the background image should scroll or be fixed.

0.

Example:

body {

background-image: url("img-tree.png");

background-position: right top;

background-repeat: no-repeat;

background-attachment: fixed;

or scroll;

What is Shorthand property?

To shorten the code, it is also possible to unify all the background properties in one property.

This is called a shorthand property.

e.g.

body {

background: #000000 url("img-tree.png")

no-repeat right top;

}

⇒ When using the shorthand property the order of the values is:

- o background-color

- o background-image

- o background-repeat

- o background-attachment

- o background-position

Background-clip property:

⇒ The background-clip property specifies whether the background should be ignored if it appears within either the border or padding area of an element.

⇒ The following declaration states that the background may display in all parts of an element, all the way to the outer edge of borders:

background-clip: border-box;

⇒ To keep the background from appearing within the borders area of an element;

background-clip: padding-box;

⇒ To restrict the background to display only the content area of an element:

background-clip: content-box;

Background-Size property:-

⇒ width and height of an image, when it is specified the `` tag, can also be used for background images in the latest version of browsers.

e.g;

background-size: 100px 100px;

Using the auto value:-

if we scale only one dimension of background image, and then have the browser calculate automatically to retain the same aspect ratio.

e.g.

background-size: 100px auto;

Example: To display multiple background images in a single declaration:

<!DOCTYPE html>

<html> <!-- backgroundimages.html -->
<head>

<title> CSS3 multiple Background
</title>

<style>

.border {

font-family: 'Times New Roman';

font-style: italic;

font-size: 100%;

text-align: center;

padding: 60px;

width: 350px;

`height : 50px;`

`background : url('b1.gif') top left no-repeat;`

`url('b2.gif') top right no-repeat;`

`url('b3.gif') bottom left no-repeat;`

`url('b4.gif') bottom right no-repeat;`

`url('b5.gif') top repeat-y;`

`url('b6.gif') left repeat-y;`

`url('b7.gif') right repeat-y;`

`url('b8.gif') bottom repeat-x;`

`}`

`</style>`

`</head>`

`<body>`

`<div class = 'bordered'>`

`<h1> Employee of the month </h1>`

`<h2> Awarded to : </h2>`

`<h3> </h3>`

`<hr> Date : </hr>`

`<hr> /<hr>`

`</body>`

`</html>`

css and Borders

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

Important properties that are used to change the borders are as follows:

- o border-width
- o border-color
- o border-style

These properties can be used together by border attributes.

Parts of Border

These are four parts to each border:

o border-top

o border-left

o border-bottom

o border-right

Border - Width:

⇒ The border-width property specifies the width of the borders.

⇒ The width can be set as a specific size (in px, pt, cm, em etc)

⇒ or by using the three predefined values:

- o thin
- o medium
- o thick

Example:

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: dashed;

border-width: thick;

}

Specific Side widths:-

O The border-width property can have from 0 to four values.

O C for the top border, right border bottom border and the left border.

Example:

Specific side colors:

P. one {

border-color: red green blue yellow;

border-color: red green blue yellow;

Border Colors:-

The border-color property is used to set the color of the borders. It can be specified as color-name or its hexadecimal value.

⇒ The colors can be set by:

O name - "red"

O Hex - "#ff0000"

O HSL - "hsl(0, 100%, 50%)"

O transparent

O RGB - "rgb(255,0,0)"

O

border-color: blue;

border-color: red green blue yellow;

border-width: 5px 10px 15px 20px;
5px for top & 20px for bottom */
border-width: 5px 10px 15px 20px;

Border Style:

⇒ There are eight possible styles for borders.

- dotted
- groove
- dashed
- ridge
- solid
- inset
- double
- outset
- none
- hidden

⇒ There are also properties for specifying each of the borders (top, right, bottom, left)

Example:

P1

border-top-style: dotted;

border-bottom-style: dotted;

border-right-style: solid;

border-left-style: solid;

Border - Shorthand property:

Border will be:

- o border-width
- o border-style (required)
- o border-color

e.g P1 border: 5px solid red

Left Border

P1 border-left: 6px solid red

Bottom border

P1 border-bottom: 6px solid red

css rounded borders

⇒ The border-radius property is used to add rounded borders to an element.

e.g

P1

border: 8px solid black

border-radius: 5px

3

e.g

border-top-left-radius: 5px

border-top-right-radius: 5px

border-bottom-left-radius: 5px

border-bottom-right-radius: 5px



What is box shadows?

⇒ To apply a box shadow, specify a horizontal and vertical offset from the object, the amount of blurring to add to shadow, and the color to use. Like this:

```
box-shadow: 15px 15px 10px #888;  
           ^  
           |  
           specify the  
           amount of blury
```

CSS Layout - Overflow

⇒ The CSS overflow property controls what happens to content that is too big to fit into an area.

⇒ Overflow property specifies whether to clip the content or to add scrollbars if the content of an element is too big to fit in the specified area.

⇒ The overflow property has the following values:

- o visible (default)
- o scroll
- o hidden
- o auto

Example

```
div {  
    overflow: hidden;
```

overflow-x & overflow-y:-

The overflow-x and overflow-y properties specifies whether to change the overflow of content just horizontally or vertically or both.

Example

```
div {  
    background-color: coral;  
    width: 40px;  
    height: 65px;  
    border: 1px solid black;  
    overflow-x: hidden;  
    overflow-y: scroll;  
}
```

CSS - Outline

⇒ An outline is a line drawn outside the element's border.

⇒ CSS has the following outline properties:

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

CSS Opacity / Transparency

⇒ The opacity property specifies the opacity / transparency of an element.

⇒ The opacity property can take a value from 0.0 - 1.0.

⇒ The lower the value, the more transparent.

Examples

```
img {  
    opacity: 0.5;  
}
```

Transparent hover effect

⇒ The opacity property is often used together with the :hover selector to change the opacity on mouse-over.

Example

```
img {  
    opacity: 0.5;  
}  
  
img:hover {  
    opacity: 1.0;  
}
```

Transparent box

When using the opacity property to add transparency to the background of an element, all of its child elements inherit the same transparency.

This can make the text inside a fully transparent element hard to read.

```
div {  
    opacity: 0.3;  
}
```

```
div {  
    background: rgba(76, 175, 80, 0.3);
```

Navigation Bar

Navigation bar = List of links

A navigation bar needs standardised HTML as a base.

Navigation bar can be build from a standard HTML list.

A navigation bar is basically a list of links, so using the `ul` and `li` elements.

Examples

 Home

 News

o bullet margins & padding family

list:

e.g. ul {
list-style-type: none;
margin: 0;
padding: 0;}

a li

display: block;

padding: 8px;

background-color: #dd0000;

Vertical Navigation bar:

CSS Transitions

o CSS Transitions allows to change property

smoothly over a given duration.

o To create a transition effect, we must specify:

- the CSS property that we want to add an effect to

- the duration of the effect.

o If the duration is not specified, the transition has no effect, because the default

value is 0.

Example:

 o display: inline;

Horizontal Navigation:

o There are two ways to create a horizontal navigation bar:

o inline

o floating list items

Example: `div {`
width: 100px;
height: 100px;
background-color:
transition: width 2s;

}

\Rightarrow div has

width: 300px;

}

\Rightarrow transition can also be add several property
values.

e.g. div{

transition: width 2s, height 4s;

}

Transition-Duration:

- o it specifies how many seconds or milliseconds a transition effect takes to complete.

e.g.: transition-duration: 1.2s;

Transition-delay:

- o it specifies a delay (in seconds) for the transition effect.

e.g.

div{

transition-delay: 1s;

}

Transition-timing-function:

- o specifies the speed curve of the transition effect.

The transition-timing-function property

specifies the

- o The transition-timing-function property has the following properties:

linear:

- Start slowly, get faster, and then end slowly

linear

- Transition at constant speed.

ease-in:

- Start slowly, and then go quickly until finished.

ease-in:

- Start quickly, stay fast until near the end, and then end slowly.

ease-in-out:

- Start slowly, go fast, and then end slowly.

Example:

#div{

transition-timing-function: linear;

}

css transformation

- Transformation allow you to move, rotate, scale and skew elements.

- `transform` methods:

matrix:

- Transforms an object by applying a matrix of values to it.

translate:

- move an element's origin

scale:

- Scales an object.

rotate:

- Rotates an object

skew:

- Skew an object

Example:

1) div {

`transform: rotate(90deg);`

}

2) div {

`transform: scale(2,3);`

}

3) div {

`transform: matrix(1, -0.3, 0, 1, 6, 0);`

}

css Text Effects

text-justify

- Specifies how justified text should be aligned and spaced.

text-overflow

- Specifies how overflowed content that is not displayed should be signaled to the user.

Example:

`oliv-test: none;`

`over-flow: visible;`

`text-overflow: ellipsis;`

3

word-wrapping

- It allows long words to be able to be broken and wrap onto the next line.

e.g. p {

`word-wrap: break-word;`

}

Word-Break

- Specifies words breaking rules or line breaking rules.

e.g. p {

word-break: keep-all;

}

p {

word-break: break-all;

}

Writing mode:

- it specifies whether lines of text are laid out horizontally or vertically.

EX:

p {

writing-mode: horizontal-rl;

}

span {

writing-mode: vertical-rl;

}

Shadow Effects

- Shadow can be added to text and to elements.

- Shadows can be:

- Text-Shadow

- Box-Shadow

Text-Shadow:

- text-shadow property applies shadow to text.

- to apply this, specify the horizontal shadow and the vertical shadow.

Example:

h1 {

text-shadow: 2px 2px 8px red;

}

- To add a blur effect:-

h1 {

{

text-shadow: 2px 2px 5px red;

- more than one shadows can also be applied.

Box-Shadow:

- All rules same like Text-Shadow.

e.g.

div {

box-shadow: 10px 10px 5px lightblue;

}

LESS Web fonts

Web fonts allow a web designer to use fonts that are not installed on the user's computer.

→ When the font is not installed on the user's computer, just include the font file on your server and it will be automatically downloaded to be used when needed.

⇒ To declare own fonts use @font-face, like this:

```
@font-face
```

```
{
```

```
    font-family: FontName;  
    src: url('FontName.otf');
```

```
}
```

⇒ font-family and src are combined.

Google web-fonts

⇒ It is one of the best way to use web fonts is to load them in from Google's servers.

⇒ To use this, just add a special sheet link in the <head> section and refers to the font in the CSS.

Example:

```
<head>
```

```
<link rel="stylesheet"
```

```
href="https://fonts.googleapis.com/css2?
```

```
family=Sofia">
```

```
<style>
```

```
body {
```

```
    font-family: "Sofia", sans-serif;
```

```
</style>
```

```
</head>
```

How to add icons in CSS?

⇒ The simple way to add icon is to add icon library such as Font Awesome.

⇒ Add the name of the specified icon class to any inline HTML element like (<i> or)

Example:

```
<head>
```

```
<link rel="stylesheet"
```

```
href="https://cdnjs.cloudflare.com/ajax/libs/
```

```
font-awesome/4.7.0/font-awesome.min.css?z
```

</head>

</body>

<i class="fa fa-car"></i>

<i class="fa fa-car" style="font-size: 48px; color: red;"></i>

<i class="fa fa-car" style="font-size: 60px; color: red;"></i>

</body>

</html>

CSS image Gallery

<style>

div.gallery {

margin: 5px;

border: 1px solid #ccc;

float: left;

width: 180px;

}

div.gallery : hover {

border: 1px solid #777;

}

div.gallery img {

width: 100%;

height: auto;

}

</style>

</body>

<div class="gallery">

</div>

Ch #19

Exercise Questions

(1) What do the CSS3 attribute Selector operators A= , $\$=$, and $*=$ do?

(1) A= : This operator matches all the start of any string.

\Rightarrow it can be used to make an attribute selector matches any element whose attribute value starts with a specified value.

\Rightarrow it does not have to be a whole word.

Example:

```
a [ href A= "http://"] {  
background: url ("outline.png") 100% 50%;  
padding-right: 15px; no-repeat;  
}
```

*= : To match any substring anywhere in the attribute *= selector is used.

o it can be used to make an attribute selector matches all elements whose attribute value contains a specified value.

properties of their browser-specific vs.
like this:

Column-Count: 3;

Column-gap: 1em;

Column-rule: 1px solid black;

(5) Name the four functions with
which you can specify CSS color.

⇒ The four functions:

◦ hsl

◦ hsla

◦ rgb

◦ rgba

e.g

Color: rgba(0%, 60%, 40%, 0.4);

(7) How can you indicate with any you use to rotate an object by ellipsis that text is truncated? 90°?

⇒ Text is truncated with an

ellipsis by the following declaration:

text-overflow: ellipsis;

(8) How can you include a Google web font in a web page?

⇒ Google web font in a web page
can be included by first selecting it
from <http://google.com/fonts>:

Then assuming, for example, chose
Lobster; include it in a <link> tag,
like this:

<link href="http://fonts.googleapis.com/css?
family=Lobster"

and also refers to the font in a CSS
declaration such as this:

hi { font-family: 'Lobster', arial, serif;

9) What CSS declaration would

⇒ transform: rotate(90deg);

b) How would you create a
gray text shadow under some
text, offset diagonally to the
bottom right by 5 pixels, with
a blurring of 3 pixels?

⇒

(10) How do you set up a transition ⇒ An image sprite is a collection of images put into a single image.

on an object so that when its properties are changed, the change will transition instead of a linear fashion over multiple seconds? ⇒ It takes a long time to load and generate a web page with many images can cause a long time to load and generate multiple second requests.

⇒ transition: all .5s linear;

Example:

```
#home {  
    width: 40px;  
    height: 40px;  
    background: url('img-newsletter.gif') 0 0;
```

Q. External stylesheets are stored in which file?

⇒ The style definitions are normally saved in "external.css" files.

⇒ it is a separate file. If we have a reference in the <link> tag made the head > in html.

```
<link rel='stylesheet' href='mystylesheet.css'>  
type='text/css'>
```

Q. What is PSEUDO class in CSS?

- ⇒ A Pseudo class is used to style a special state of element.
- ⇒ used to add styles to selector, but only when those selectors meet certain conditions.

⇒ A Pseudo class is represented by adding a colon (:) after a selector in CSS, followed by a Pseudo class such as "hover", "focus", or "active".

like this:

a:hover {

/* ... */

}

for example:

- o Style an element when a user moves cursor over it

- o Style visited and unvisited links differently

- o Style an element when it gets focus.

Syntax:

- o Selector : Pseudo-class {

 property : value;

Example:

a:hover {

 color: #FF0000;

}

a:visited {

 color: #00FFFF;

}

a: hover {

 color: #FF00FF;

}

a: active {

 color: #0000FF;

}

Q. Why external CSS better than others?

- ⇒ CSS code is in a separate document, HTML files will have cleaner structure and are smaller in size.

- ⇒ External CSS is used to add styling to multiple HTML pages at a time.

- ⇒ <link> tag is used for external style sheet.

- ⇒ Downloadable both HTML and CSS file.

Q. What is the purpose for CSS code?

- ⇒ CSS is markup language responsible and controls the look how your web pages will look like.
- ⇒ used for describing the presentation of web pages including colors, layout and fonts of website.
- ⇒ This style sheet language also allows you to add effects or animations to website.
- ⇒ CSS is independent of HTML and can be used with any XML-based markup language.

Q. Write CSS code for "Time" new Roman", kept bold font

```
h1 {  
    font-family: "Time New Roman", Times,  
    font-size: 16pt;  
    font-weight: bold;  
}
```

" " are missing

- Q. How can you select a group of different elements such as elements type in the CSS2
- Q. What are background styles in CSS
- Q. Identity the class body
- background-image: url('logo.jpg');
background-position: bottom right;
- 3

Ch #8

P

Q. what is the difference between relative and absolute position?

=> Position: relative => Places an element relative to its current position without changing the layout around it.

=> Position: absolute => Places an element relative to its parent's position and changing the layout around it.

div. relative {

Position: relative;

width: 400px

height: 200px

div. absolute {

Position: absolute;
top: 80px;

right: 0;

width: 200px;

height: 100px;

border: 3px solid black;

Q. what is difference } between block

and inline element?
=> these are two display values?

Block

=> it always starts on a new line, and the browser automatically add some space before and after the element.

=> it always takes up the full available width.

=> `<div>` and `<p>`

are commonly used block elements.

inline

=> it does not start on a new line:

=> it only takes up as much width as necessary

=> `` is commonly used inline element
`<i>`, `<a>` etc.