Chapter 4

CSS
Style Sheets

Need Of CSS

- CSS is a language that describes the style of an HTML document.
- CSS describes how HTML elements should be displayed.
- **CSS** to define how HTML elements should appear.

CSS Selectors

1. Element Selector
2. Id Selector
3. Class Selector
4. Group selector

Element Selector

- The element selector select the element based on element name

```
p
{
Background-color:red;
}
```

Id selector

- Id selector use the id attribute of the HTML element to select specific element.

```
#abc
{
background-color:red;
}
```

Class selector

- The class selector element based on class attribute.

```
.abc
{
Background-color:red;
}
```

Group selector

- If we have elements with same style.

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

we can write above code as

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

What is CSS?

- **CSS** stands for **C**ascading **S**tyle **S**heets
- CSS describes **how HTML elements are to be displayed on screen.**
- CSS **saves a lot of work**. It can control the layout of multiple web pages all at once.[With an external stylesheet file, you can change the look of an entire website by changing just one file!]
- External style sheets are stored in **CSS files.**

CSS Backgrounds

- CSS background properties:

1. background-color
2. background-image
3. background-repeat
4. background-attachment
5. background-position

Background Color

- The background-color property specifies the background color of an element.

Example

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

Background Color Example 2

- Example

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

div {
        background-color: lightblue;
    }

p {
        background-color: yellow;
    }
```

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.

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

Repeat Horizontally or Vertically

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

Background-repeat values:

repeat-x

repeat-y

no-repeat

Background-position

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

- It has three different types of values:

Length values (e.g. 100px 5px)

Percentages (e.g. 100% 5%)

Keywords (e.g. Right top)

Background-position

- The default values are 0 0. This places your background image at the top left of the container.
- The first value is the horizontal position, second value is the vertical position.
- So 100px 5px will move the image 100px to the right and five pixels down.

Background-position

- Keywords are just shortcuts for percentages. It's easier to remember and write top right than 0 100%, and that's why keywords are a thing. Here is a list of all five keywords and their equivalent values:

  top: 0% vertically

  right: 100% horizontally

  bottom: 100% vertically

  left: 0% horizontally

  center: 50% horizontally if horizontal isn't already defined. If it is then this is applied vertically.

Background Image - Fixed position

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

Shorthand Property
Background

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

Sequence:

background-color

background-image

background-repeat

background-attachment

background-position

CSS
Borders

Different border Properties

- border-width:medium/thick/thin
- border-style (required)
- border-color
- Border-collapse: initial/seperate/collapse

Border Width

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

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

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

Border Style

- 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 Color

The border-color property can have from one to four values (for the top border, right border, bottom border, and the left border).

```
.one {
    border-style: solid;
    border-color: red;
}
```

Border - Individual Sides

- From the examples above you have seen that it is possible to specify a different border for each side.

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

Border - Shorthand Property

```
p {
    border: 5px solid red;
}
```

Sequence

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

CSS Margins

- The CSS margin properties are used to generate space around elements.
- The margin properties set the size of the white space outside the border.
- CSS has properties for specifying the margin for each side of an element:

1. margin-top
2. margin-right
3. margin-bottom
4. margin-left

Example

```
p {
    margin-top: 100px;
    margin-bottom: 100px;
    margin-right: 150px;
    margin-left: 80px;
  }
```

Margin - Shorthand Property

```
p {
    margin: 100px 150px 100px 80px;
  }
```

Sequence

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

Margin - Shorthand Property

margin: 100px 150px 100px 80px;

Top Right Bottom Left

margin: 100px 150px 100px;

Top Right/Left Bottom

margin: 100px 150px;

Top/bottom Right/left

Marging: 100px;

All

CSS Padding

- The CSS padding properties are used to generate space around content.
- The padding clears an area around the content (inside the border) of an element

1. padding-top
2. padding-right
3. padding-bottom
4. padding-left

Example

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

Padding - Shorthand Property

```
p {
    padding: 50px 30px 50px 80px;
  }
```

Sequence

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

Padding - Shorthand Property

padding: 100px 150px 100px 80px;
Top Right Bottom Left

padding: 100px 150px 100px;
Top Right/Left Bottom

padding: 100px 150px;
Top/bottom Right/left

padding: 100px;
All

CSS
Text

TEXT CSS Properties

- color
- Text-alignment: center/inherit/justify/left/right
- text-decoration:blink/inherit/line-through/none/overline/underline
- Text-transform:capitalize/inherit/lowercase/uppercase
- Text-indent
- Letter-spacing
- Line-Height
- Direction:rtl/ltr
- Word-spacing
- Text-shadow: *h-shadow v-shadow color*|none|initial|inherit;

Text Color and Text Align

```
h1 {
    color: green;
}

h1 {
    text-align: center/left/right;
}
```

Text Decoration

```
a {
    text-decoration: none;
  }
```

Other Values for Text Decoration

overline

line-through

underline

Text Transformation

The text-transform property is used to specify uppercase and lowercase letters in a text.

```
p {
    text-transform: uppercase;
}
```

- Other values should be

uppercase

lowercase

capitalize

Text Indentation

- The text-indent property is used to specify the indentation of the first line of a text:

```
p {
    text-indent: 50px;
}
```

Output

In my younger and more vulnerable years my father gave me some advice that I've been turning over in my mind ever since. 'Whenever you feel.

Letter Spacing & Line Height

```
h1 {
        letter-spacing: 3px;
}


p {
        line-height: 0.8;
    }
```

Text Direction and Word Spacing

- The direction property is used to change the text direction of an element:

```
div {
    direction: rtl;
}
```

The word-spacing property is used to specify the space between the words in a text.

```
h1 {
    word-spacing: 10px;
}
```

Text Shadow

- The text-shadow property adds shadow to text.

The following example specifies the position of the horizontal shadow (3px), the position of the vertical shadow (2px) and the color of the shadow (red):

```
h1 {
    text-shadow: 3px 2px red;
}
```

CSS
FONT

CSS Font Family

- The font family of a text is set with the font-family property
- Example

```
p {
    font-family: Times New Roman, Times, serif;
}
```

Font Style

- The font-style property is mostly used to specify italic text.
- This property has three values:
- normal - The text is shown normally
- italic - The text is shown in italics
- oblique - The text is "leaning" (oblique is very similar to italic, but less supported)

Font Style

- Example

```
p {
    font-style: normal;
}
```

Font Size

```
h1 {
      font-size: 40px;
   }
```

Set Font Size With Em [emphemeral unit]

pixels/16=*em*

e.g. 16pixels is 1em.

Font Weight

- The font-weight property specifies the weight of a font:

```
p {
    font-weight: normal/bold;
}
```

Font Variant

- The font-variant property specifies whether or not a text should be displayed in a small-caps font.
- In a small-caps font, all lowercase letters are converted to uppercase letters. However, the converted uppercase letters appears in a smaller font size than the original uppercase letters in the text.

Font Variant: Example

```
p{
   font-variant: normal;
}

h1 {
   font-variant: small-caps;
}
```

CSS
Links

Styling Links

The four links states are:

- a:link - a normal, unvisited link
- a:visited - a link the user has visited
- a:hover - a link when the user mouse over it.
- a:active - a link the moment it is clicked

- Example

```
a:link {
    color: red;
}
/* visited link */
a:visited {
    color: green;
}
/* mouse over link */
a:hover {
    color: hotpink;
}
/* selected link */
a:active {
    color: blue;
}
```

Text Decoration by Link

- a:link {
  text-decoration: none;
  }

  a:visited {
  text-decoration: none;
  }

  a:hover {
  text-decoration: underline;
  }

  a:active {
  text-decoration: underline;
  }

Background Color by link

- a:link {
  background-color: yellow;
  }

  a:visited {
  background-color: cyan;
  }

  a:hover {
  background-color: lightgreen;
  }

  a:active {
  background-color: hotpink;
  }

Advanced - Link

```
a:link, a:visited {
background-color: #f44336;
color: white;
padding: 14px 25px;
text-align: center;
text-decoration: none;
display: inline-block;
}

a:hover, a:active {
    background-color: yellow;
    color:black;
      text-decoration: underline;
}
```

Different List Item Markers

- list-style-type:circle/square/upper-roman/lower-alpha
- list-style-image:url(‘imagename')
- list-style-position:inside/outside

Example

- ul{
  list-style-type: circle;
  }
- ul {
  list-style-image: url('sqpurple.gif');
  }
- ul {
  list-style-position: inside;
  }

List - Shorthand property

- ul {
  list-style: square inside url("sqpurple.gif");
  }

1. list-style-type
2. list-style-position
3. list-style-image

CSS Layout - The position Property

- The position Property

  The position property specifies the type ofpositioning method used for an element.

There are four different position values:

1. static
2. relative
3. fixed
4. absolute

position: static;

- Static positioned elements are not affected by the top, bottom, left, and right properties.

- div.static {
  position: static;
  border: 3px solid #73AD21;
  }

position: relative;

- An element with position: relative; is positioned relative to its normal position.

- div.relative {
  position: relative;
  left: 30px;
  border: 3px solid #73AD21;
  }

position: fixed;

- An element with position: fixed; is positioned relative to the viewport, which means it always stays in the same place even if the page is scrolled. The top, right, bottom, and left properties are used to position the element.
- div.fixed {
  position: fixed;
  bottom: 0;
  right: 0;
  width: 300px;
  border: 3px solid #73AD21;
  }

position: absolute;

- An element with position: absolute; is positioned relative to the nearest positioned ancestor (instead of positioned relative to the viewport, like fixed).

Example

- ```
  div.relative {
      position: relative;
      width: 400px;
      height: 200px;
      border: 3px solid #73AD21;
  }

  div.absolute {
      position: absolute;
      top: 80px;
      right: 0;
      width: 200px;
      height: 100px;
      border: 3px solid #73AD21;
  }
  ```
  ```

CSS Combinators

- There are four different combinators in CSS3:

1. descendant selector (space)
2. child selector (>)
3. adjacent sibling selector (+)
4. general sibling selector (~)

Descendant Selector

- The descendant selector matches all elements that are descendants of a specified element.
- div p {
  background-color: yellow;
  }
- In above example,Apply properties of all paragraph of div

Example

```
<div>
  <p>Paragraph 1 in the div.</p>
  <p>Paragraph 2 in the div.</p>
  <span><p>Paragraph 3 in the div.</p></span>
</div>
```

Child Selector

- The child selector selects all elements that are the immediate children of a specified element.

- ```
  div > p {
      background-color: yellow;
  }
  ```
  ```

Example

```
<div>
  <p>Paragraph 1 in the div.</p>
  <p>Paragraph 2 in the div.</p>
  <span><p>Paragraph 3 in the div.</p></span>
</div>
```

Adjacent Sibling Selector

- The adjacent sibling selector selects all elements that are the adjacent siblings of a specified element.

- div + p {
  background-color: yellow;
  }
- Apply to next immediate element after completion of div element

Example

```
<div>
  <p>Paragraph 1 in the div.</p>
  <p>Paragraph 2 in the div.</p>
</div>

<p>Paragraph 3. Not in a div.</p>
<p>Paragraph 4. Not in a div.</p>
```

General Sibling Selector

- The general sibling selector selects all elements that are siblings of a specified element.

- ```
  div ~ p {
      background-color: yellow;
  }
  ```
  ```

Example

```
<p>Paragraph 1.</p>

<div>
  <code>Some code.</code>
  <p>Paragraph 2.</p>
</div>

<p>Paragraph 3.</p>
<code>Some code.</code>
<p>Paragraph 4.</p>
```

CSS Pseudo-classes

- What are 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

Syntax

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

}
```

Anchor Pseudo-classes

- ```
  a:link {
     color: #FF0000;
  }

  /* visited link */
  a:visited {
     color: #00FF00;
  }

  /* mouse over link */
  a:hover {
     color: #FF00FF;
  }

  /* selected link */
  a:active {
     color: #0000FF;
  }
  ```
  ```

Pseudo-classes and CSS Classes

- Pseudo-classes can be combined with CSS classes:

```
a.highlight:hover
 {
      color: red;
   }
```

Hover on

```
div:hover
 {
      background-color: blue;
   }
```

```
<style
    >
p {
    display: none;
    background-color: yellow;
    padding: 20px;
}

div:hover p {
    display: block;
}
</style>
</head>
<body>

<div>Hover over me to show the p element
  <p>Tada! Here I am!</p>
</div>
```

CSS - The :first-child Pseudo-class

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

Apply to first paragraph of body part

Element within first child

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

Apply to first child italic of paragraph element.

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

Apply to all italic elements of first paragraph element.

CSS Pseudo-elements

- What are Pseudo-Elements?
  a) A CSS pseudo-element is used to style specified parts of an element.
  b) For example, it can be used to:
  – Style the first letter, or line, of an element
  – Insert content before, or after, the content
    - of an element

Syntax

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

The ::first-line Pseudo-element

The ::first-line pseudo-element is used to add a special style to the first line of a text.

```
p::first-line
 {
      color: #ff0000;
      font-variant: small-caps;
   }
```

Apply css to first line of the paragaraph.

The ::first-letter Pseudo-element

The ::first-letter pseudo-element is used to add a special style to the first letter of a text.

```
p::first-letter
{
    color: #ff0000;
    font-size: xx-large;
}
```

Pseudo-elements and CSS Classes

- Pseudo-elements can be combined with CSS classes:

```
p.intro::first-letter
 {
      color: #ff0000;
      font-size:200%;
   }
```

Multiple Pseudo-elements

- Several pseudo-elements can also be combined.

```
p::first-letter {
    color: #ff0000;
    font-size: xx-large;
}

p::first-line {
    color: #0000ff;
    font-variant: small-caps;
}
```

CSS - The ::before Pseudo-element

- The ::before pseudo-element can be used to insert some content before the content of an element.

```
h1::before
 {
      content: url(smiley.gif);
  }
```

CSS - The ::after Pseudo-element

- The ::after pseudo-element can be used to insert some content after the content of an element.

```
h1::after
{
    content: url(smiley.gif);
}
```

CSS3 Modules

- CSS3 has been split into "modules". It contains the "old CSS specification" (which has been split into smaller pieces). In addition, new modules are added.

CSS3 Modules

- Some of the most important CSS3 modules are:
  1. Selectors
  2. Box Model
  3. Backgrounds and Borders
  4. Image Values and Replaced Content
  5. Text Effects
  6. 2D/3D Transformations
  7. Animations
  8. Multiple Column Layout
  9. User Interface

CSS3 Rounded Corners

- With the CSS3 border-radius property, you can give any element "rounded corners".
- #a {

  border-radius: 25px;

  background: #73AD21;

  padding: 20px;

  width: 200px;

  height: 150px;

  }

CSS3 border-radius - Specify Each Corner

- **Four values:** first value applies to top-left, second value applies to top-right, third value applies to bottom-right, and fourth value applies to bottom-left corner
- **Three values:** first value applies to top-left, second value applies to top-right and bottom-left, and third value applies to bottom-right
- **Two values:** first value applies to top-left and bottom-right corner, and the second value applies to top-right and bottom-left corner
- **One value:** all four corners are rounded equally

CSS3 Border Images

- #borderimg1 {
  border: 10px solid transparent;
  padding: 15px;
  -webkit-border-image: url(border.png) 50 round; /* Safari 3.1-5 */
  -o-border-image: url(border.png) 50 round/stretch; /* Opera 11-12.1 */
  border-image: url(border.png) 50 round/stretch;
  }

CSS3 Backgrounds

- background-size: contain/cover/values h v
- background-origin: border-box/padding-box/content-box
- background-clip

CSS3 background-origin Property

- The CSS3 background-origin property specifies where the background image is positioned.
- The property takes three different values:

1. border-box - the background image starts from the upper left corner of the border
2. padding-box - (default) the background image starts from the upper left corner of the padding edge
3. content-box - the background image starts from the upper left corner of the content

Example

```
#abc {
    border: 10px solid black;
    padding: 35px;
    background: url(img_flwr.gif);
    background-repeat: no-repeat;
    background-origin: content-box;
}
```

CSS3 background-clip Property

- The CSS3 background-clip property specifies the painting area of the background.
- The property takes three different values:

1. border-box - (default) the background is painted to the outside edge of the border
2. padding-box - the background is painted to the outside edge of the padding
3. content-box - the background is painted within the content box

Example

- #abc1 {
  border: 10px dotted black;
  padding: 35px;
  background: yellow;
  background-clip: content-box;
  }

- Thank you So muchhh.....