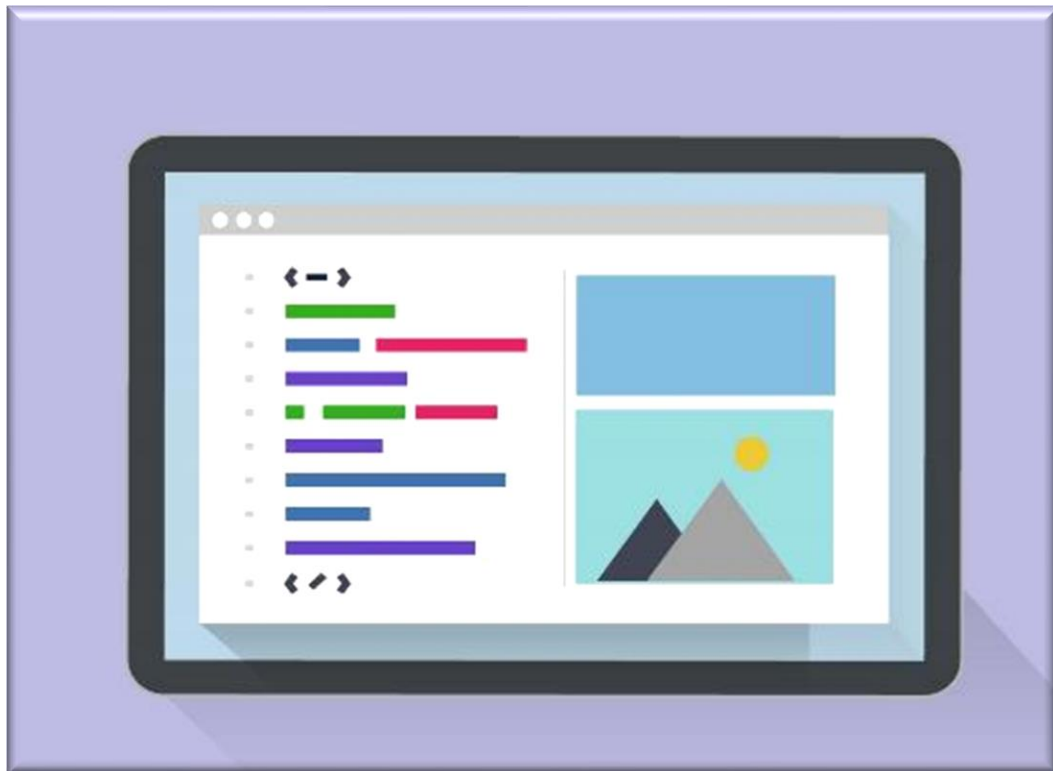




# CSS: Concepts, Architecture, Nutshell Examples, Outlook

Marina Wustinger

# Topics



---

What is CSS? What does it stand for?

---

History

---

Syntax

---

Implementation of Styles

---

CSS Architecture

---

Nutshell Examples

---

Outlook

# CSS

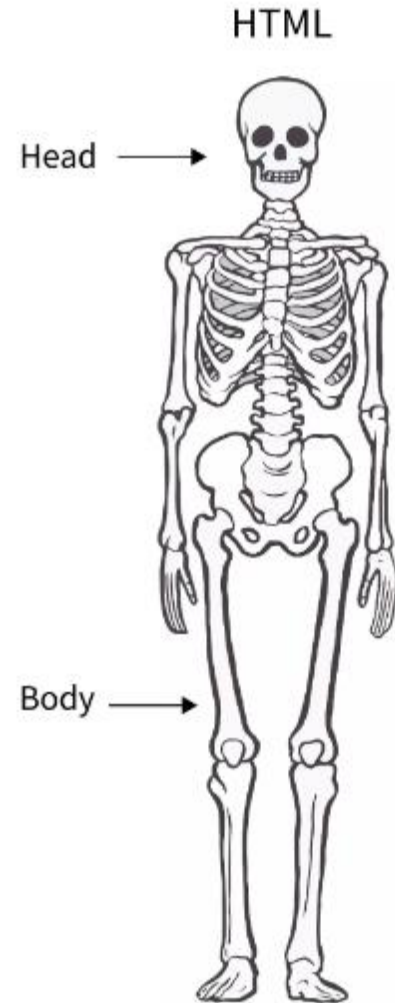
---

## What is CSS?

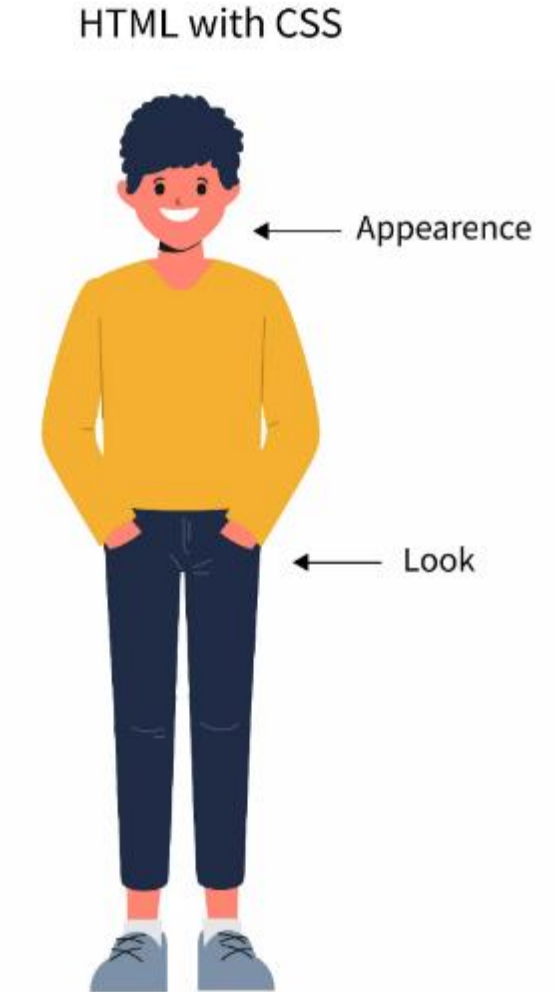
CSS is a stylesheet language that specifies how a document is visually presented to users. - style, layout, etc.

## What does CSS stand for?

Cascading Style Sheets



Structural Layer



Presentation Layer

# History

- Invented by Hakon Wium Lie in 1990
- First released in 1996
- Standardized by W3C – control and develop new features
- CSS4 ??

# Syntax

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

# How are styles implemented?

---

There are three methods to implement styles in a document.

# Inline styles

---

- Bad practice
- Hard to make changes
- Only really useful when specifying a unique style to only one single element.

```
<> demo.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8      <title>Demo Document</title>
9  </head>
10
11 <body>
12
13     <h1 style="color: ■red; font-size: 30px;">Hello World!</h1>
14
15 </body>
16
17 </html>
```

# Embedded Styles

---

- Also not recommended
- Makes it impossible to share styles

```
<> demo.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8      <title>Demo Document</title>
9      <style>
10         h1 {
11             color: red;
12             font-size: 30px;
13         }
14     </style>
15 </head>
16
17 <body>
18
19     <h1>Hello World!</h1>
20
21 </body>
22
23 </html>
```



# External Stylesheets

---

- Best practice
- Great reusability
- Can use it to style multiple pages
- Can link several stylesheets to one document
- @import

```
<> demo.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8      <title>Demo Document</title>
9      <link rel="stylesheet" href="demo_app.css">
10
11     <!-- or using the import rule -->
12     <style>
13         @import url("demo_app.css");
14     </style>
15
16 </head>
17
18 <body>
19
20     <h1>Hello World!</h1>
21
22 </body>
23
24 </html>
```

# CSS Architecture

What is CSS Architecture?

Can be regarded as a set of guidelines and is essential for complex projects.

Gives CSS code a clear structure and enforces consistent styling.

Provides code consistency, reusability and easy maintenance.

# SMACSS – Scalable and Modular CSS

- Focuses on file/folder organization
- Break CSS down based on five rules:
  1. Base
  2. Layout
  3. Modules
  4. State
  5. Theme
- Simple and easy to incorporate into workflow

# BEM – Block Element Modifier

- Simple naming convention
- Defining classes in HTML to be able to write independent CSS blocks which can be reused
- No complications with code
- Works well with parts of SMACSS
- `[block]__[element]--[modifier]`

<> demo.html > ...

```
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8      <title>Demo Document</title>
9      <link rel="stylesheet" href="demo_app.css">
10 </head>
11
12 <body>
13     <figure class="photo">
14         
15         <figcaption class="photo__caption photo__caption--large">Look at me!</figcaption>
16     </figure>
17 </body>
18
19 </html>
```

# CSS Architecture

Which Architecture should you use?

Really does not matter. Use the one you are the most comfortable with, the one the team prefers working with.

Highly recommended to include it in the documentation of the project. This gives newcomers the opportunity to catch up easily and quickly.

# Nutshell Examples

# Travel Bucket List

```
14 <h1>My Travel Bucket List</h1>
15 <div>
16 |   <p> These are some cities I would love to visit one day!</p>
17 </div>
18 <hr>
19 <ul>
20 |   <li>Barcelona, Spain</li>
21 |   <li class="must done">London, UK</li>
22 |   <li>Tokyo, Japan</li>
23 |   <li class="must">Honolulu, Hawaii</li>
24 |   <li>Helsinki, Finland</li>
25 |   <li class="done">Paris, France</li>
26 |   <li class="must done">New York City, USA</li>
27 |   <li>Toronto, Canada</li>
28 |   <li class="must">Edinburgh, Scotland</li>
29 |   <li class="must">Auckland, New Zealand</li>
30 |   <li class="done">Rome, Italy</li>
31 |   <li>Seoul, South Korea</li>
32 |   <li>Oslo, Norway</li>
33 |   <li class="done">Galway, Ireland</li>
34 |   <li class="must">Amsterdam, Netherlands</li>
35 </ul>
36 <form>
37 |   <input type="text" name="city" id="name" placeholder="City, Country">
38 |   <input type="submit" value="Add City" id="add">
39 </form>
```

## My Travel Bucket List

These are some cities I would love to visit one day!

- Barcelona, Spain
- London, UK
- Tokyo, Japan
- Honolulu, Hawaii
- Helsinki, Finland
- Paris, France
- New York City, USA
- Toronto, Canada
- Edinburgh, Scotland
- Auckland, New Zealand
- Rome, Italy
- Seoul, South Korea
- Oslo, Norway
- Galway, Ireland
- Amsterdam, Netherlands



# How to Apply and Change Colours

```
# demo_app.css > ...
```

```
1  h1 {  
2  |   color:  teal;  
3  | }  
4  
5  html {  
6  |   background-color:  antiquewhite;  
7  | }
```

- Element Selector: h1
- Property: color
- Value: green
  
- Element Selector: html
- Property: background-color
- Value: antiquewhite

# Colours in CSS

- Human-readable
- RGB/RGBA
  - (red, green, blue, alpha channel)
  - 0-255
  - Red: `rgba(255, 0, 0, 0)`
- Hexadecimals
  - # followed by six hex numbers
  - 0-1
  - a-f
  - White: `#ffffff`
  - Black: `#000000`

```
# demo_app.css > ...
1  html {
2  |     background-color:  rgb(237, 231, 227);
3  | }
4  |
5  h1 {
6  |     color:  rgba(72, 159, 181, 0.925);
7  | }
8  |
9  .must {
10 |     color:  #82c0cc;
11 | }
```

# My Travel Bucket List

These are some cities I would love to visit one day!

- Barcelona, Spain
- London, UK
- Tokyo, Japan
- Honolulu, Hawaii
- Helsinki, Finland
- Paris, France
- New York City, USA
- Toronto, Canada
- Edinburgh, Scotland
- Auckland, New Zealand
- Rome, Italy
- Seoul, South Korea
- Oslo, Norway
- Galway, Ireland
- Amsterdam, Netherlands

# How to Style a Document Using Common Text Properties

`text-align`: sets horizontal alignment of a content

`font-weight`: sets weight/boldness of a text element

`font-size`: sets size of font, does not affect thickness of characters

`letter-spacing`: sets space between characters

`line-height`: sets distance between lines, here: height of line boxes

`text-decoration`: sets appearance of decorative lines on text

```
# demo_app.css > ...
```

```
1  p {
2      text-align: center;
3      font-weight: bold;
4      font-size: 20px;
5      letter-spacing: 3px;
6      line-height: 2.5;
7      text-decoration: underline solid ■ black;
8  }
```

# My Travel Bucket List

These are some cities I would love to visit one day!

Barcelona, Spain

~~London, UK~~

Tokyo, Japan

Honolulu, Hawaii

Helsinki, Finland

~~Paris, France~~

~~New York City, USA~~

Toronto, Canada

Edinburgh, Scotland

```
# demo_app.css > ...
```

```
1  * {
2  |   text-align: center;
3  | }
4
5  ul {
6  |   list-style-type: none;
7  |   font-weight: bolder;
8  |   line-height: 2.5;
9  |   font-size: 180%;
10 | }
11
12 h1 {
13 |   letter-spacing: 5px;
14 |   font-size: 370%;
15 | }
16
17 .done {
18 |   text-decoration: line-through ■ black;
19 | }
```

# How to Change the Font Style

- Styles depend on the ones build into browser
- `font-family`: sets the font style
- Font stack: acts as a back-up if browser can't display a font

```
# demo_app.css > ...  
1  * {  
2  |   font-family: 'Courier New', Courier, monospace;  
3  | }  
-
```

# My Travel Bucket List

These are some cities I would love to visit one day!

Barcelona, Spain

~~London, UK~~

Tokyo, Japan

Honolulu, Hawaii

Helsinki, Finland

~~Paris, France~~

~~New York City, USA~~

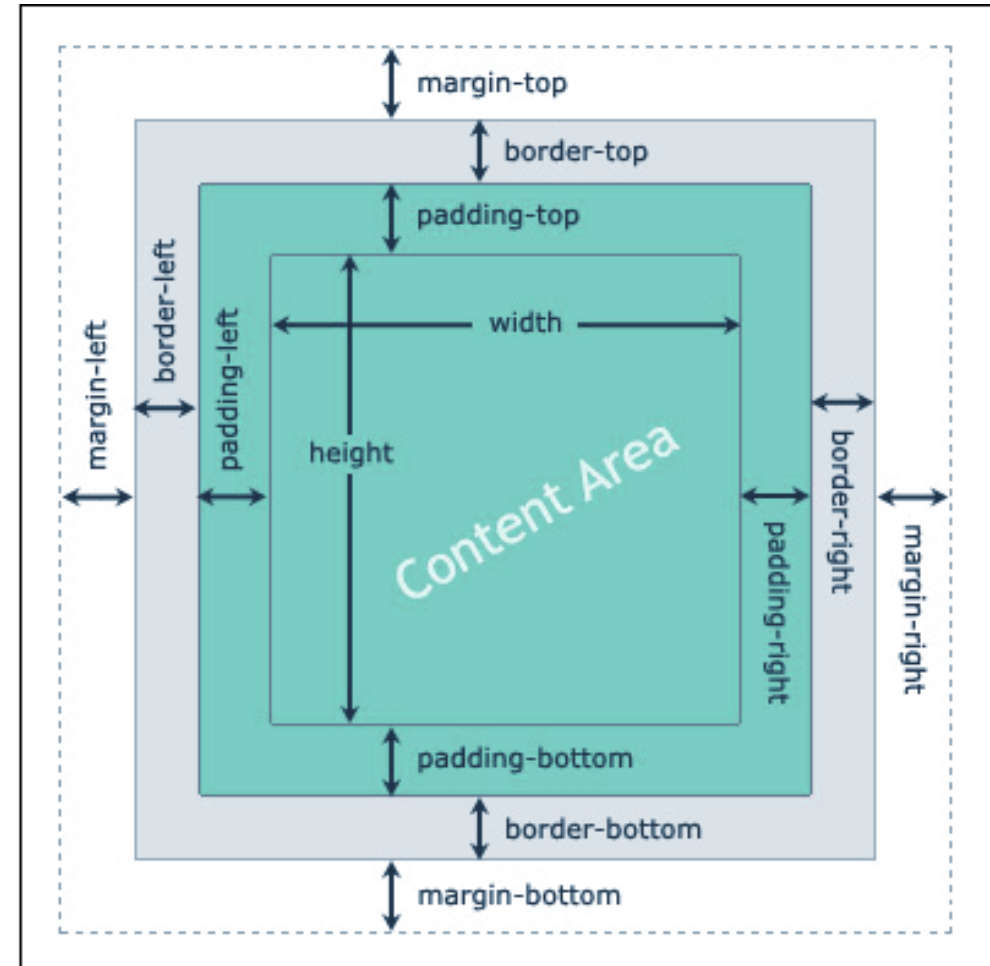
Toronto, Canada

Edinburgh, Scotland

# The Box Model

---

- Each element is presented as a box by browser's rendering engine
- Content area: displays content
- Padding: creates space around content
- Border: runs around element
- Margin: separates element from its neighbour





# How to Use Box Model Properties

- Attribute selector
- `border-box`: any padding and border will be regarded
- `margin`: to increase the distance between elements
- `border`: to set width, colour and style of the border
- `padding`: not necessarily needed for this example
- `outline: none` will stop it from changing when selected
- `border-radius`: to round off boxes

```
# demo_app.css > ...
1  * {
2  |   box-sizing: border-box;
3  | }
4
5  input[type="text"] {
6  |   font-size: 120%;
7  |   width: 300px;
8  |   height: 50px;
9  |   margin: 5px;
10 |   border: 2px solid #16697a;
11 |   outline: none;
12 |   border-radius: 10px;
13 | }
14
15 input[type="submit"] {
16 |   font-size: 120%;
17 |   font-weight: bold;
18 |   width: 120px;
19 |   height: 50px;
20 |   margin: 5px;
21 |   border: 2px solid #16697a;
22 |   background-color: #16697a;
23 |   border-radius: 10px;
24 | }
```

City, Country

Add City

City, Country

Add City

# How to Create a Hover Effect

- Pseudo-classes: act like a class, specify a certain state of an element
- State when hovering on element:
  - Background colour changes
  - Cursor changes to hand pointer

```
# demo_app.css > ...  
1   #add:hover {  
2       background-color: #82c0cc;  
3       cursor: pointer;  
4   }
```

# How to stick an element to one position

- Different values to position an element
- Here: `sticky`
- Sticks element to bottom of the page, making it overlap with others when scrolling up

```
# demo_app.css > ...  
1  ∨ form {  
2    |     position: sticky;  
3    |     bottom: 10px;  
4    | }
```


# Advent Calendar

---


```
13 <h1>Advent Calendar</h1>
14 <section id="container1">
15   <div>
16     <h4 class="day">1</h4>
17     
19   </div>
20   <div>
21     <h4 class="day">2</h4>
22     
24   </div>
25   <div>
26     <h4 class="day">3</h4>
27     
29   </div>
30   <div>
31     <h4 class="day">4</h4>
32     
34   </div>
35   <div>
36     <h4 class="day">5</h4>
37     
39   </div>
40   <div>
41     <h4 class="day">6</h4>
42     
44   </div>
45 </section>
```

## Advent Calendar


1

 A Dog Wearing a Santa Hat


2

 Christmas Tree with Popcorn

3

 Red and Green Elf Hat


4

 Owl Wearing a Santa Hat

5

 Christmas Gift with Big Bow

6

 Boy Holding a Christmas Wish List

```
demo_app.css > ...
```

```
1  #container1,  
2  #container2,  
3  #container3,  
4  #container4 {  
5      display: flex;  
6      width: 90%;  
7      height: 100%;  
8      margin: 3% auto;  
9      flex-direction: row;  
10     justify-content: space-evenly;  
11     align-items: center;  
12 }  
13  
14 section div {  
15     display: flex;  
16     justify-content: center;  
17     align-items: center;  
18 }
```

# How to Create a Layout Using Flexbox

- Display: Containers are set to flex
- Flex-direction: specifies direction of main axis
- Justify-content: specifies how items are distributed along main axis
- Align-items: distributes space between items along cross axis
- align-self: sets the behaviour of one element inside a container

# ADVENT CALENDAR

2

3

4

5

8

9

10

11

```
# demo_app.css > ...
```

```
1  section div {
2  |   transition: all 3s;
3  | }
4
5  img {
6  |   transition: opacity 4s;
7  | }
8
9  section div:hover {
10 |   color: transparent;
11 |   box-shadow: none;
12 |   background-color: □#ffffff;
13 |   transform: rotate(360deg);
14 | }
15
16 img:hover {
17 |   opacity: 1;
18 | }
```

# How to Create a Transition

- Change the state of an element by specifying transition
- Pseudo-classes are used to set the “new” state
- transition: property, duration, timing function, delay
- transform: gives you the ability to rotate a transformable element



# How to Make Design Responsive

# demo\_app.css > ...

```
1  @media only screen and (min-width:481px) and (max-width: 768px) {
2
3      section div {
4          width: 100px;
5          height: 100px;
6          font-size: 2em;
7          margin-left: 2px;
8          margin-right: 2px;
9      }
10
11     h1 {
12         font-size: 4em;
13     }
14
15     img {
16         width: 1em;
17         height: 1.5em;
18     }
19 }
```

- Media Queries: used to target media features, make web page more responsive
- At-rule: @media
- only screen: keeps older browsers from applying styles
- Conditions: must be true for styles to be applied



# Outlook

---

- Adding scripting language
  - Making pages dynamic
  - JavaScript
- Working with frameworks
  - Bootstrap
- Web Accessibility
  - ARIA
  - Automatic.CSS



The End

# Image Sources

**Title Page:**

<https://cdn.windowsreport.com/wp-content/uploads/2020/11/10-Best-desktops-for-programming-and-coding.jpg>

**Topics:** <https://www.levelaccess.com/wp-content/uploads/2011/06/css.jpg>

**CSS:**

<https://scaler.com/topics/images/html-and-css-meme.webp>

**The Box Model:**

<https://static.javatpoint.com/csspages/images/css-box-model.png>

**Outlook:** [https://external-](https://external-preview.redd.it/XKWf2weCD6EGL6HzI28kQXX7SAv70T3kLOWQh9AMsZA.jpg?auto=webp&s=e9e270390d2e9d40454422e8938e15d43fcdbee9)

[preview.redd.it/XKWf2weCD6EGL6HzI28kQXX7SAv70T3kLOWQh9AMsZA.jpg?auto=webp&s=e9e270390d2e9d40454422e8938e15d43fcdbee9](https://external-preview.redd.it/XKWf2weCD6EGL6HzI28kQXX7SAv70T3kLOWQh9AMsZA.jpg?auto=webp&s=e9e270390d2e9d40454422e8938e15d43fcdbee9)