

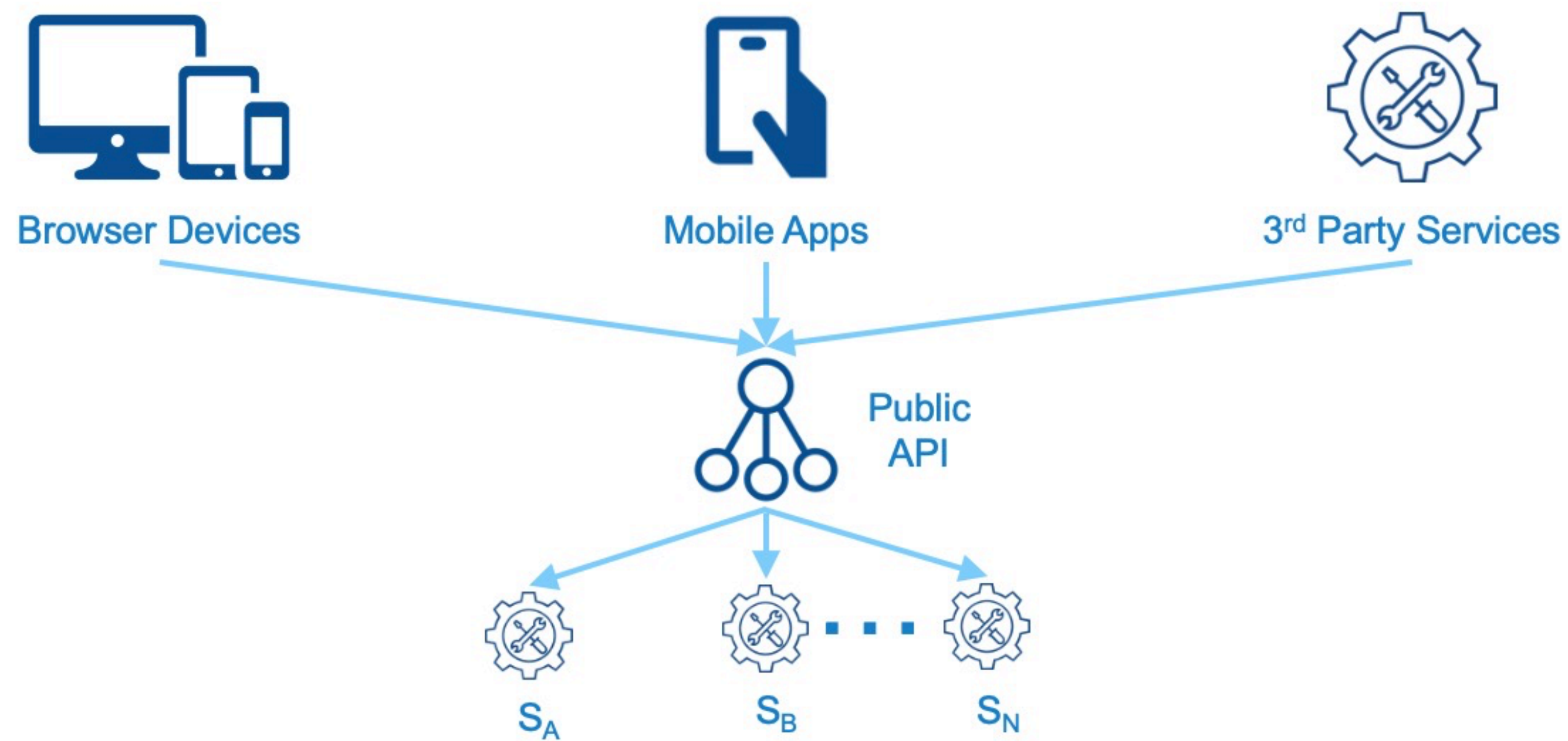
Programowanie Front-end

Wprowadzenie do HTML i CSS

Podstawy języka JavaScript

mgr inż. Jakub Gogola

Czym jest frontend?



Na czym polega tworzenie frontend'u?

- **Frontend** to tworzenie **wizualnej** i **interaktywnej** części strony/aplikacji webowej.
- Aplikacja/strona webowa powinna spełniać kryteria poprawnego **UX** (*user experience*) oraz **UI** (*user interface*).
- Podstawowymi technologiami wykorzystywanymi w programowaniu webowym są **HTML**, **CSS** oraz **JavaScript**.

Pierwsza strona internetowa

Pierwsza strona internetowa została stworzona w **CERN** i opublikowana **30. kwietnia 1993r.**

<http://info.cern.ch/>

World Wide Web

The WorldWideWeb (W3) is a wide-area [hypermedia](#) information retrieval initiative aiming to give universal access to a large universe of documents.

Everything there is online about W3 is linked directly or indirectly to this document, including an [executive summary](#) of the project, [Mailing lists](#) , [Policy](#) , November's [W3 news](#) , [Frequently Asked Questions](#) .

[What's out there?](#)

Pointers to the world's online information, [subjects](#) , [W3 servers](#), etc.

[Help](#)

on the browser you are using

[Software Products](#)

A list of W3 project components and their current state. (e.g. [Line Mode](#) ,X11 [Viola](#) , [NeXTStep](#) , [Servers](#) , [Tools](#) , [Mail robot](#) , [Library](#))

[Technical](#)

Details of protocols, formats, program internals etc

[Bibliography](#)

Paper documentation on W3 and references.

[People](#)

A list of some people involved in the project.

[History](#)

A summary of the history of the project.

[How can I help ?](#)

If you would like to support the web..

[Getting code](#)

Getting the code by [anonymous FTP](#) , etc.

Podstawowe pojęcia

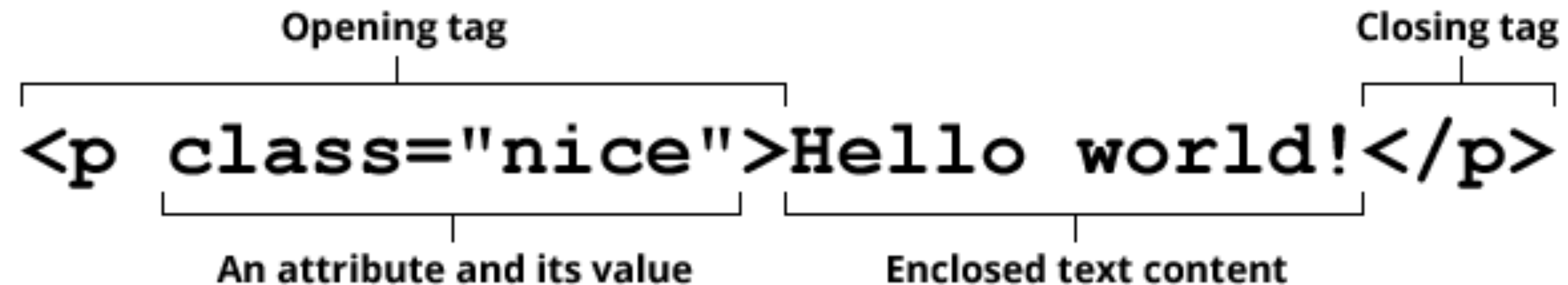
HTML

- **HTML** (*Hypertext Markup Language*) - język znaczników oparty na XML
 - **definiuje** strukturę (układ) strony internetowej - nagłówki, treść, linki itp.
 - **nie definiuje** wyglądu strony ani zachowania (interaktywność) poszczególnych jej elementów

Podstawowe pojęcia HTML

- Struktura elementu HTML

Anatomy of an HTML element



Podstawowe pojęcia HTML

- Podstawowe tagi języka HTML
 - Tekst:
 - Nagłówki: `<h1>` ... `<h6>`
 - Paragrafy: `<p>`, ``
 - Linki: `<a>`
 - Obrazy: ``
 - Formularze: `<form>`, `<input>`, `<button>`

Podstawowe pojęcia HTML

- Każdy tag języka HTML może posiadać **atrybuty** - są to dodatkowe wartości przekazywane do elementu
- Atrybuty nie są widoczne w wizualnej części strony, ale wpływają na zachowanie elementu

Podstawowe pojęcia HTML

- Przykłady atrybutów:
 - `` - źródło (link), z którego powinien zostać pobrany obraz
 - `` - link, pod który powinno nastąpić przekierowanie
 - `<button disabled />` - interakcje z przyciskiem są wyłączone

Podstawowe pojęcia HTML

- Przykład kodu HTML

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML basics</title>
  </head>
  <body>
    <h1>Welcome to My Website about HTML basics</h1>
    <p>You can find here a link to list of HTML tags</p>
    <a href="https://developer.mozilla.org/en-US/docs/Web/HTML/Content_categories">Click me!</a>
  </body>
</html>
```

Podstawowe pojęcia

HTML

- `<html>` - główny element całego dokumentu HTML, który otwiera i zamyka całą strukturę strony
- `<head>` - sekcja dokumentu, która zawiera *metadane*. Nie są one wyświetlane bezpośrednio na stronie, ale zawierają informacje takie jak tytuł strony, odwołania do arkuszy stylów, skrypty
- `<title>` - tag, który definiuje tytuł strony; pojawia się on na zakładce przeglądarki oraz jest używany przez wyszukiwarki do identyfikacji i indeksowania strony
- `<body>` - sekcja, w której znajduje się treść strony, czyli elementy widoczne bezpośrednio dla użytkownika w przeglądarce. To główna część dokumentu HTML, która zawiera wszystkie elementy interfejsu.

Podstawowe pojęcia CSS

- **CSS** (*Cascading Style Sheets*) - język używany do definiowania wizualnej części strony internetowej
 - Zapewnia separację zawartości strony (HTML) od definicji jej wyglądu
 - Składnia:

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

Podstawowe pojęcia CSS

- Przykład kodu CSS

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

```
h1 {  
    color: navy;  
    text-align: center;  
}
```

Podstawowe pojęcia CSS

- Style CSS można łączyć z dokumentem HTML na dwa różne sposoby
 - osadzić je bezpośrednio w dokumencie za pomocą tagu `<style>`
 - dodać dowiązanie za pomocą tagu `<link>`

Podstawowe pojęcia CSS

- Przykład bezpośredniego osadzenia CSS w dokumencie HTML

```
<!DOCTYPE html>
<html>
  <head>
    <title>Simple Website with CSS</title>
    <style>
      h1 {
        color: #2c3e50;
        font-size: 36px;
        text-align: center;
      }

      p {
        font-size: 18px;
        line-height: 1.6;
        color: #333;
      }
    </style>
  </head>
  <body>
    <h1>Welcome to My Website</h1>
    <p>This is a simple example of an HTML document with embedded CSS.</p>
  </body>
</html>
```

Podstawowe pojęcia CSS

- CSS jako osobny plik

```
<!DOCTYPE html>
<html>
  <head>
    <title>Simple Website with CSS</title>
    <link rel="stylesheet" href="linked-css.css">
  </head>
  <body>
    <h1>Welcome to My Website</h1>
    <p>This is a simple example of an HTML
document with an external CSS file.</p>
  </body>
</html>
```

```
h1 {
  color: #2c3e50;
  font-size: 36px;
  text-align: center;
}

p {
  font-size: 18px;
  line-height: 1.6;
  color: #333;
}
```


Podstawowe pojęcia CSS

- CSS jako osobny plik

```
<!DOCTYPE html>
<html>
  <head>
    <title>Simple Website with CSS</title>
    <link rel="stylesheet" href="linked-css.css">
  </head>
  <body>
    <h1>Welcome to My Website</h1>
    <p>This is a simple example of an HTML
document with an external CSS file.</p>
  </body>
</html>
```

```
h1 {
  color: #2c3e50;
  font-size: 36px;
  text-align: center;
}

p {
  font-size: 18px;
  line-height: 1.6;
  color: #333;
}
```

Podstawowe pojęcia CSS

- Style możemy definiować na kilka różnych sposobów:
 - dla danego typu elementu HTML
 - przez klasy
 - przez identyfikatory elementu

Podstawowe pojęcia CSS

- Tagi HTML

```
<!DOCTYPE html>
<html>
  <head>
    <title>Simple Website with CSS</title>
    <style>
      h1 {
        color: #2c3e50;
        font-size: 36px;
        text-align: center;
      }

      p {
        font-size: 18px;
        line-height: 1.6;
        color: #333;
      }
    </style>
  </head>
  <body>
    <h1>Welcome to My Website</h1>
    <p>This is a simple example of an HTML document with embedded CSS.</p>
  </body>
</html>
```

Podstawowe pojęcia CSS

- Klasy

```
<!DOCTYPE html>
<html>
  <head>
    <title>Simple Website with CSS Classes</title>
    <style>
      .header {
        color: #2c3e50;
        font-size: 36px;
        text-align: center;
      }

      .text {
        font-size: 18px;
        line-height: 1.6;
        color: #333;
      }
    </style>
  </head>
  <body>
    <h1 class="header">Welcome to My Website</h1>
    <p class="text">This is a simple example of an HTML document with CSS classes.</p>
  </body>
</html>
```

Podstawowe pojęcia CSS

- ID

```
<!DOCTYPE html>
<html>
  <head>
    <title>Simple Website with CSS IDs</title>
    <style>
      #header {
        color: #2c3e50;
        font-size: 36px;
        text-align: center;
      }

      #text {
        font-size: 18px;
        line-height: 1.6;
        color: #333;
      }
    </style>
  </head>
  <body>
    <h1 id="header">Welcome to My Website</h1>
    <p id="text">This is a simple example of an HTML document with CSS IDs.</p>
  </body>
</html>
```

Podstawowe pojęcia CSS

- Selektory

```
<!DOCTYPE html>
<html>
  <head>
    <title>Simple Website with CSS Selectors</title>
    <link rel="stylesheet" href="styles.css">
  </head>
  <body>
    <h1>Welcome to My Website</h1>
    <p id="intro">This is a simple example of an HTML document with CSS
selectors.</p>
    <p class="highlight">This paragraph has a yellow background, using a class
selector.</p>
    <div>
      <p>This paragraph is inside a div and will have blue text using a
descendant selector, and bold text because of the child combinator selector.</
p>
      <span>This span inside the div is not affected by the child combinator
selector because it only targets direct children.</span>
    </div>
    <a href="#">Hover over this link to see a pseudo-class in action!</a>
  </body>
</html>
```

```
/* Element selector */
h1 {
  color: #2c3e50;
  font-size: 36px;
  text-align: center;
}

/* Class selector */
.highlight {
  background-color: yellow;
}

/* ID selector */
#intro {
  font-size: 18px;
  color: #333;
  line-height: 1.6;
}

/* Group selector (multiple elements) */
h1, p {
  margin-bottom: 20px;
}

/* Descendant selector */
div p {
  color: blue;
}

/* Child combinator selector (>) */
div > p {
  font-weight: bold;
}

/* Pseudo-class selector */
a:hover {
  color: red;
  text-decoration: underline;
}
```

Podstawowe pojęcia CSS

- Język CSS wyróżnia się następującymi cechami:
 - kaskadowość (*cascade*)
 - specyficzność (*specificity*)
 - dziedziczność (*inheritance*)

Podstawowe pojęcia CSS

- Kaskadowość (*cascade*)

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      p {
        color: red;
      }

      .highlight {
        color: blue;
      }
    </style>
  </head>
  <body>
    <p class=„highlight">This text will be blue due to cascade rule.</p>
  </body>
</html>
```


Podstawowe pojęcia CSS

- Specyficzność (*specificity*)

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      p {
        color: red; /* specificity 1 */
      }

      .highlight {
        color: blue; /* specificity 10 */
      }

      #intro {
        color: green; /* specificity 100 */
      }
    </style>
  </head>
  <body>
    <p id="intro" class="highlight">This text will be green due to the highest specificity of ID.</p>
  </body>
</html>
```

Podstawowe pojęcia CSS

- Dziedziczność (*inheritance*)

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      body {
        color: purple;
      }
    </style>
  </head>
  <body>
    <h1>This title will be purple because it inherits from body styles.</h1>
    <p>This paragraph also will be purple.</p>
  </body>
</html>
```

Podstawowe pojęcia

DOM

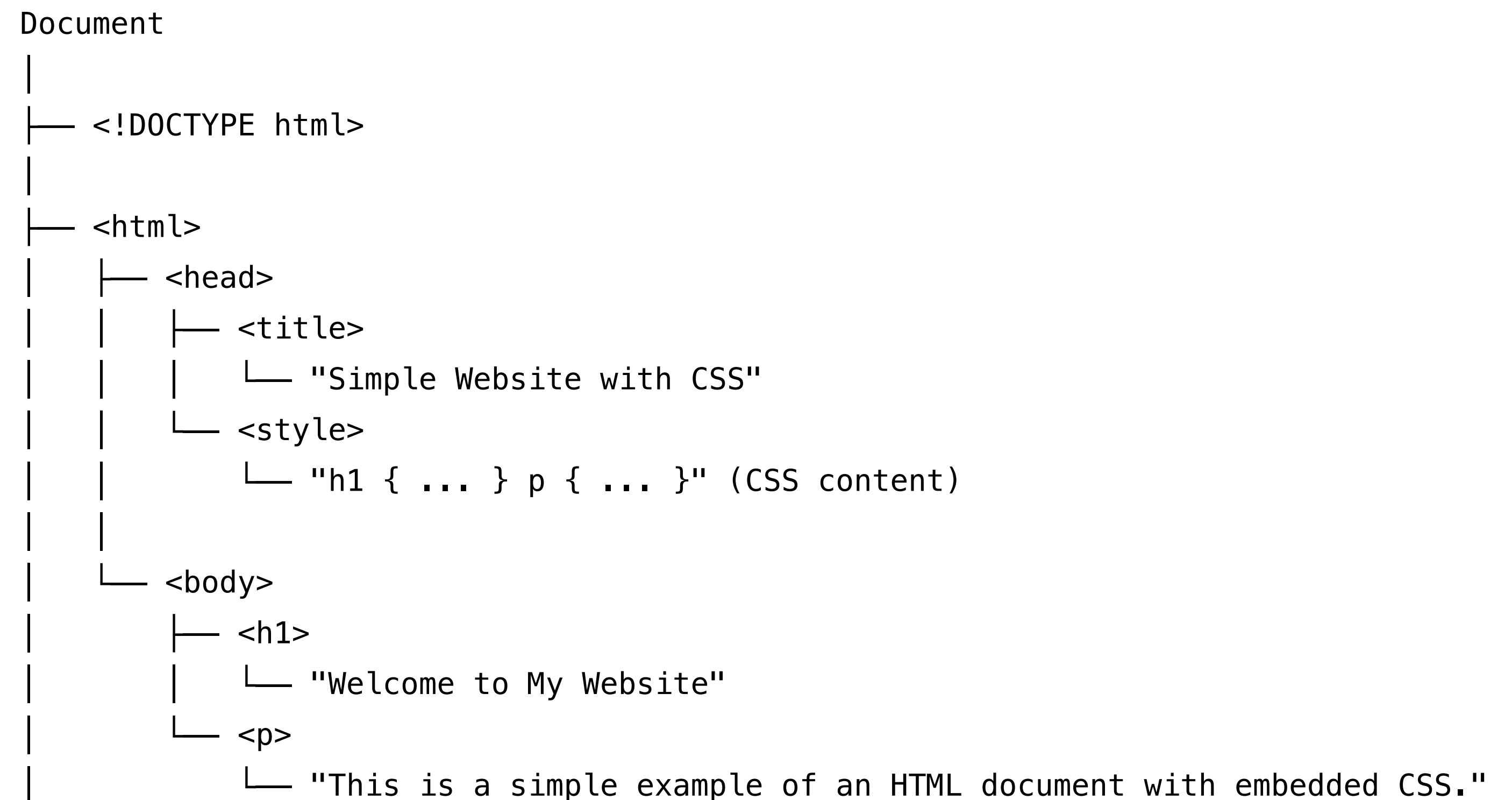
- **DOM** (*Document Object Model*) - model budowany przez przeglądarkę, który reprezentuje strukturę dokumentu HTML. Jest reprezentowany przez **drzewo**, gdzie **węzłem** jest pojedynczy element dokumentu.

Podstawowe pojęcia DOM

- Przykład wizualizacji drzewa DOM

```
<!DOCTYPE html>
<html>
  <head>
    <title>Simple Website with CSS</title>
    <style>
      h1 {
        color: #2c3e50;
        font-size: 36px;
        text-align: center;
      }

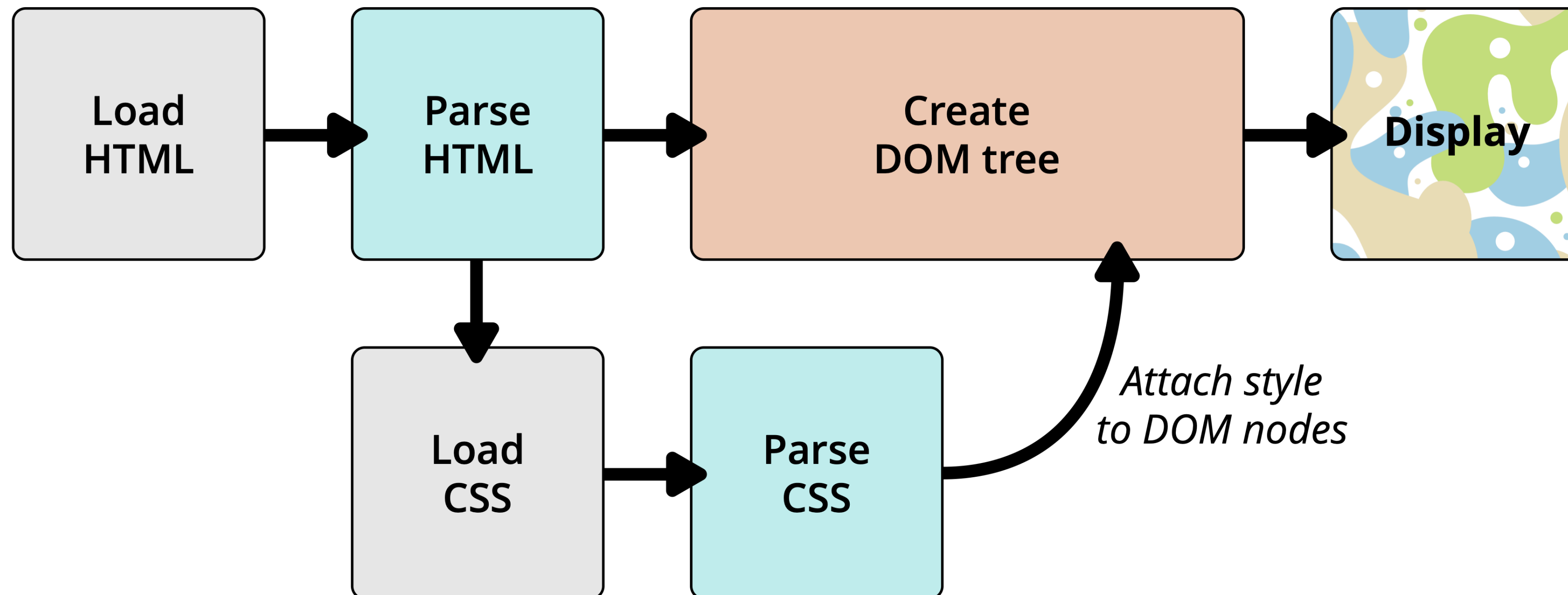
      p {
        font-size: 18px;
        line-height: 1.6;
        color: #333;
      }
    </style>
  </head>
  <body>
    <h1>Welcome to My Website</h1>
    <p>This is a simple example of an HTML document
with embedded CSS.</p>
  </body>
</html>
```



Podstawowe pojęcia

DOM + CSS

- Proces dołączania stylów do modelu dokumentu



Podstawowe pojęcia

JavaScript

- **JavaScript** - jeden z trzech głównych filarów technologii frontendowych (obok HTML i CSS) używany do tworzenia dynamicznych i interaktywnych stron internetowych. W przeciwieństwie do HTML (który określa strukturę strony) i CSS (który definiuje wygląd), JavaScript umożliwia interakcję użytkownika z elementami na stronie i reagowanie na zdarzenia, takie jak kliknięcia, wpisywanie tekstu czy przesuwanie kursora.

Podstawowe pojęcia JavaScript

- Przykład kodu w języku JavaScript

```
let counter = 0;
```

```
const step = 10;
```

```
function incrementOne() {  
    counter++;  
}
```

```
function incrementTen() {  
    counter += 10;  
}
```

```
function displayCounter() {  
    console.log(`Counter: ${counter}`);  
}
```

```
incrementOne();  
incrementOne();  
incrementOne();  
displayCounter();  
incrementTen();  
displayCounter();
```

Podstawowe pojęcia

HTML + JavaScript

- Interpreter języka JavaScript jest dostępny w każdej przeglądarce. Można za jego pomocą modyfikować zawartość dokumentu poprzez obiekt `document`, który jest *uchwytem* do elementów kodu HTML
 - pobieranie elementów dokumentu
 - modyfikowanie zawartości i atrybutów elementów
 - tworzenie i usuwanie elementów HTML
 - reagowanie na zdarzenia użytkownika

Podstawowe pojęcia

HTML + JavaScript

- Kod JavaScript, podobnie jak CSS może zostać zdefiniowany **bezpośrednio** w dokumencie HTML za pomocą tagu `<script>` lub podlinkowany jako osobny plik

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML basics</title>
    <script>
      console.log('JavaScript code embedded directly into HTML.');
    <script src="jsLinked.js"></script>
  </head>
  <body>
    <p>Use Developer Tools to check what will be printend by JavaScript.</p>
  </body>
</html>
```

```
console.log('JavaScript code in
separate file.');
```

Prezentacja przykładu

HTML + CSS + JavaScript

- Prezentacja wykorzystania HTML, CSS i JavaScript na przykładzie licznika.

Do poczytania

- Zachęcam do zapoznania się z nieco rozszerzonym wprowadzeniem do omówionych na wykładzie technologii:
 - HTML - <https://developer.mozilla.org/en-US/docs/Learn/HTML>
 - CSS - <https://developer.mozilla.org/en-US/docs/Learn/CSS>
 - JavaScript - <https://developer.mozilla.org/en-US/docs/Learn/JavaScript>
 - Przykłady z wykładu: <https://github.com/JakubGogola-IDENTT/dsw-frontend-lecture-2024/tree/main/lecture-1>

Dziękuję za uwagę!