

HTML, CSS, and JavaScript



Rainforest

Frontend Lead @ Hashgreen

- Overview
 - Frontend
 - Backend
 - DevOps
- HTML
- CSS
- JavaScript

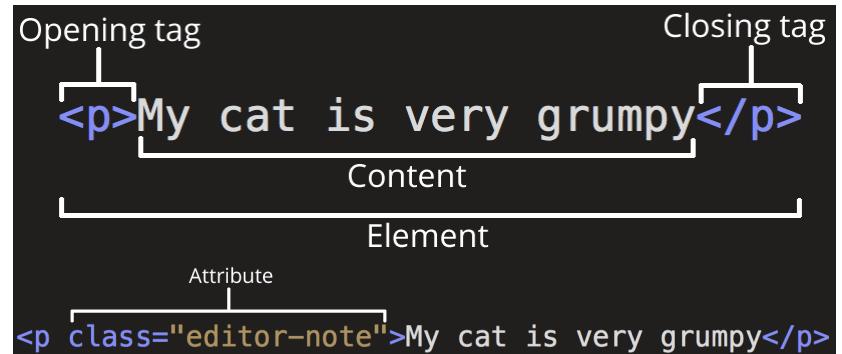
Overview

HTML

- Hypertext Markup Language,
- markup language
-
- nesting element

```
<ul id="todo__items" class="todo__items">...</ul>
```

- `<ul` : opening tag
- `` : closing tag
- `...` : content
- `id`, `class`, ...: attributes



HTML

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>My test page</title>
  </head>
  <body>
    <nav></nav>
    <aside>
      <ul>
        <li>Item 01</li>
        <li>Item 02</li>
        ...
      </ul>
    </aside>
    <main>
      <p>
        This is demo text!
      </p>
    </main>
    <footer>Rainforest@2022</footer>
  </body>
</html>
```

- head
 - CSS
 - JavaScript
- - title:
 - favicon
 - description
 - ...
- body
 - nav
 - aside
 - main
 - footer
 - ...

HTML elements

`div`

- the generic container for flow content
- no effect on the content or layout until styled in some way using CSS

- heading: h1, h2, ..., h6

- paragraph: p

- link: a

- span

- generic inline container for phrasing content

{ Block-level elements v.s. Inline elements

- Unordered lists: `ul -> li`

- Ordered lists: `ol -> li`

Heading 1

Heading 2

Heading 3

Heading 4

Heading 5

HEADING 6

paragraph paragraph paragraph paragraph paragraph
paragraph paragraph

Rainforest

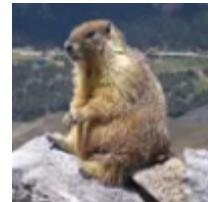
paragraph paragraph paragraph paragraph **span with red text** paragraph paragraph

- Unordered List Item 1 1. Ordered List Item 1
- Unordered List Item 2 2. Ordered List Item 2
- Unordered List Item 3 3. Ordered List Item 3
- ... 4. ...

HTML elements(Cont.)

- `img`
- `table`
- `button`
- `form`
- `label`
- `input`
- `select`, `option`
- `textarea`

{ Further Reading



Column 1	Column 2	...
Cell 1	Cell 2	...
...
Button		
text input		
number input		
password input		
Label for Select		
Option 1		▼
textarea		〃

ARIA

- Accessible Rich Internet Applications
- a set of roles and attributes that define ways to make web content and web applications (especially those developed with JavaScript) more accessible to people with disabilities.

``data-*``

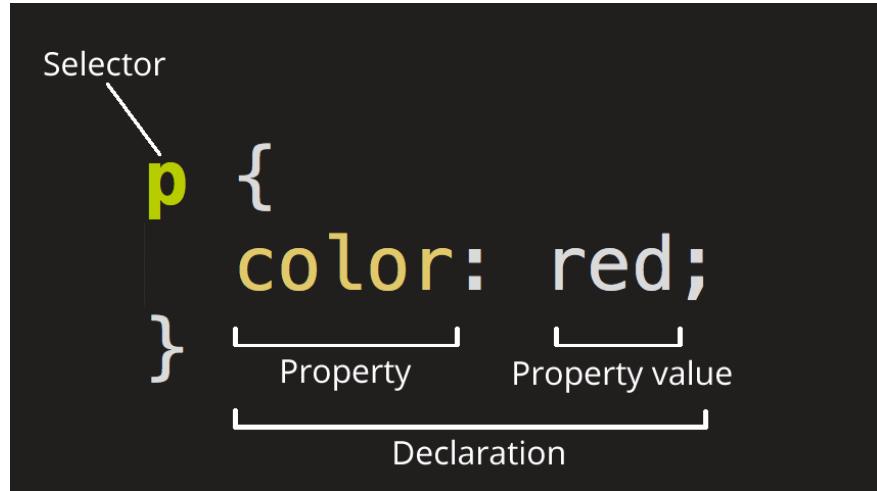
allow us to store extra information on standard, semantic HTML elements without other hacks such as non-standard attributes, or extra properties on DOM.

[Further Reading](#)

[Further Reading](#)

CSS

- Cascading Stylesheets,
- style sheet language
- Selector
- Declaration = (Properties) + (Property value)



Selector

```
<img id='root' class='class1' src='/ha.png' />
```



- ID

```
#root {  
    border-radius: 50px;  
}
```

- Class

```
.class1 {  
    border: 10px solid blue;  
}
```

- Pseudo-class selector

```
img:hover {  
    cursor: pointer;  
}
```

- Pseudo-elements selector

```
elements ::first-line {  
    color: blue;  
}
```

paragraph paragraph paragraph paragraph paragraph
paragraph paragraph paragraph paragraph paragraph

```
[src] {  
    padding: 10px;
```

Selector(Cont.)

- Grouping selectors

```
p, #root, .class1, [src] {  
    padding: 0;  
}
```

- Combinators

- Descendant: " "

```
p span {  
    color: red;  
}
```

- Child: ">"

```
p > span {  
    color: red;  
}
```

- ...

Further Reading

paragraph paragraph paragraph paragraph paragraph
paragraph paragraph paragraph paragraph paragraph
span subspan

paragraph paragraph paragraph paragraph paragraph
paragraph paragraph paragraph paragraph paragraph
span subspan

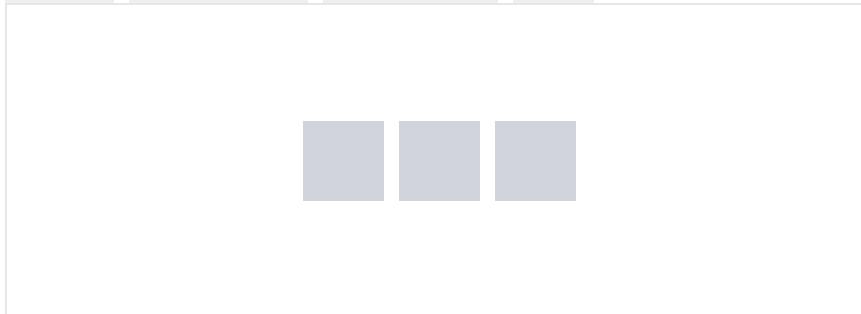
Position

- relative
- absolute
 - removed from the normal document flow
 - no space is created for the element in the page layout
 - positioned relative to its **closest positioned ancestor**
- fixed
 - removed from the normal document flow
 - no space is created for the element in the page layout
 - positioned relative to **the initial containing block** established by the viewport
- sticky
 - offset relative to its nearest scrolling ancestor and containing block (nearest block-level ancestor)

Display

- block: generating line breaks both before and after the element when in the normal flow.
- inline-block
- flex
 - CSS Flexible Box Layout

Row ▾ Center ▾ Center ▾ 2 ▾



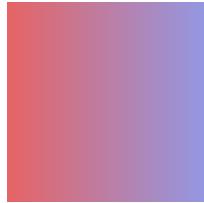
- grid

- Grid Layout

Colors

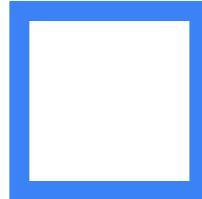
- background

```
background: red;  
background: url('/ha.png');  
background-image: linear-gradient(#e66465, #9198e5);
```



- border

```
border: 10px solid blue;
```



- text

```
color: red;
```

Green Text

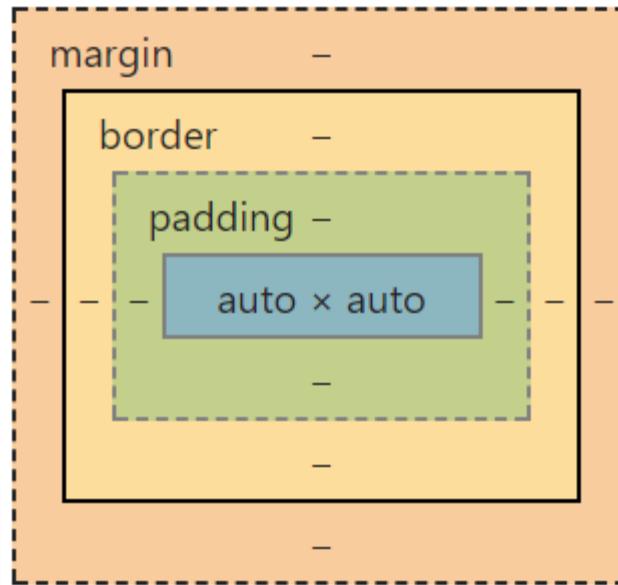
Text

- font-family
- font-size
- font-weight

Sizing

- width
- height
- padding
- margin
- box-sizing
 - border-box
 - content-box

```
width: 100px;  
height: 100px;  
padding: 8px;  
margin: 4px;  
box-sizing: border-box | content-box;
```



Pre or Post-processor



- Pre-processor

- , , , 

- Take a language that compiles to CSS

```
$font-stack: Helvetica, sans-serif;  
$primary-color: #333;
```

```
body {  
  font: 100% $font-stack;  
  color: $primary-color;  
}
```

- Post-processor

- 

- Align and refurbish CSS to have the best possible outcome for today's browsers.

Deconfusing Pre- and Post-processing

→

```
body {  
  font: 100% Helvetica, sans-serif;  
  color: #333;  
}
```

JavaScript

- JavaScript is bananas

```
+ 'a'  
// NaN, Not a number  
'b' + 'a'  
// 'ba'  
'b' + 'a' + + 'a' + 'a'  
// 'baNaNa'  
('b' + 'a' + + 'a' + 'a').toLowerCase()  
// 'banana'
```

- == v.s. ===

- ==: operator attempts to convert them to the same type before comparing.
- ===: strict equality operators

- Shallow Copy v.s. Deep Copy

- shallow copy is copy by reference
- deep copy

```
const data = { a: 'test' }  
const b = Object.assign({}, data)  
const c = {...a}
```

- JS HTML DOM

- JS Web APIs

- TypeScript: strongly typed programming language that builds on JavaScript

- Wappalyzer
- Inspect Element
 - Elements
 - Console
 - Storage
 - (Sources)
- Visual Studio Code + Web Server
 - Live Server
- Codepen
- Stackblitz

Demo



Starting dev server