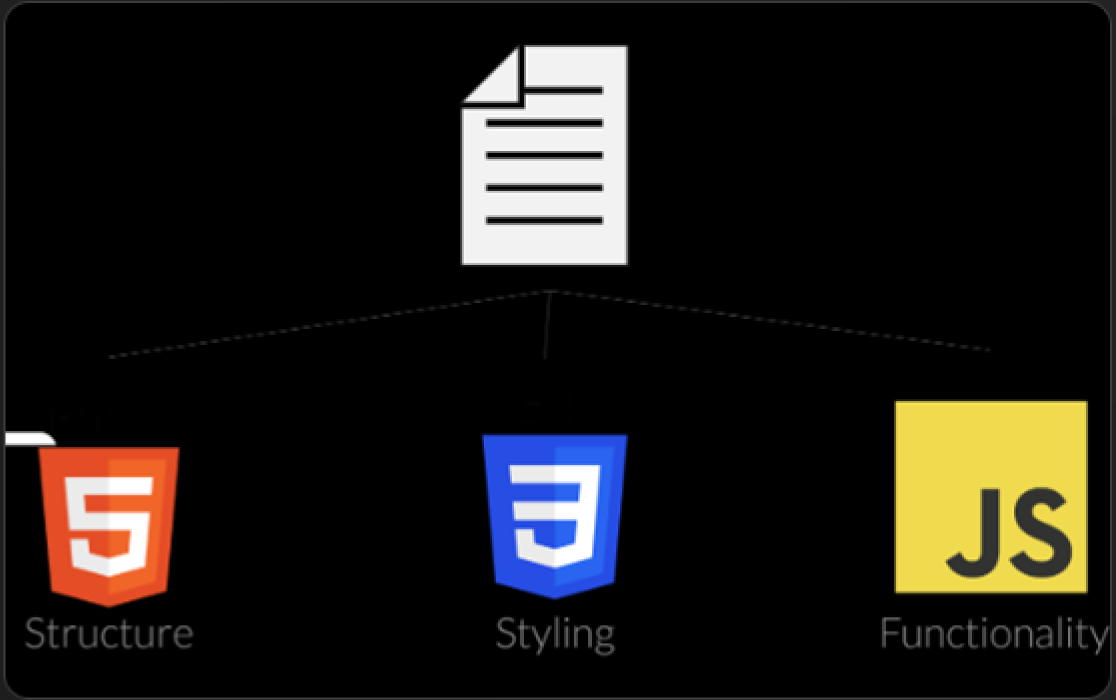
HTML Fundamentals

What is HTML?

- **H**yper**T**ext **M**arkup **L**anguage.
- It is NOT a programming language; it is a markup language used to create the structure of web pages.
- Think of it as the "skeleton" of a website.
- It works together with CSS (Style) and JavaScript (Behavior) to create modern web experiences.



The Basic Structure

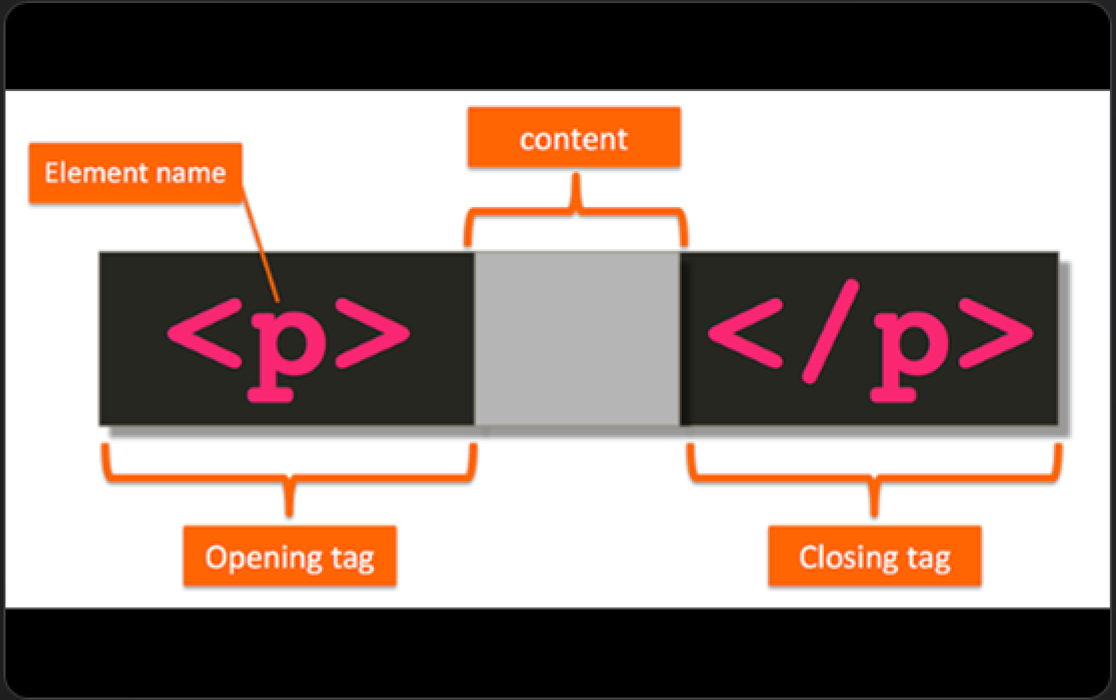
```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>My Pagetitle>
  head>
  <body>



  body>
html>
```

- : Tells the browser this is an HTML5 document.
- : The root element of the page.
- : Contains meta-information (like the title) that isn't shown on the page itself.
- : Contains the visible page content (text, images, links).

Tags & Elements

The Anatomy of a Tag

- **Opening Tag:** Starts the element (e.g.,).
- **Content:** Information between tags.
- **Closing Tag:** Ends the element, includes a forward slash (e.g.,).
- **Nesting:** Placing elements inside other elements (e.g., a bold tag inside a paragraph).



HTML Attributes

Definition

Attributes provide additional information about elements. They are always specified in the **opening tag**.

Syntax

They usually come in name/value pairs: name="value".

Example: class="btn"

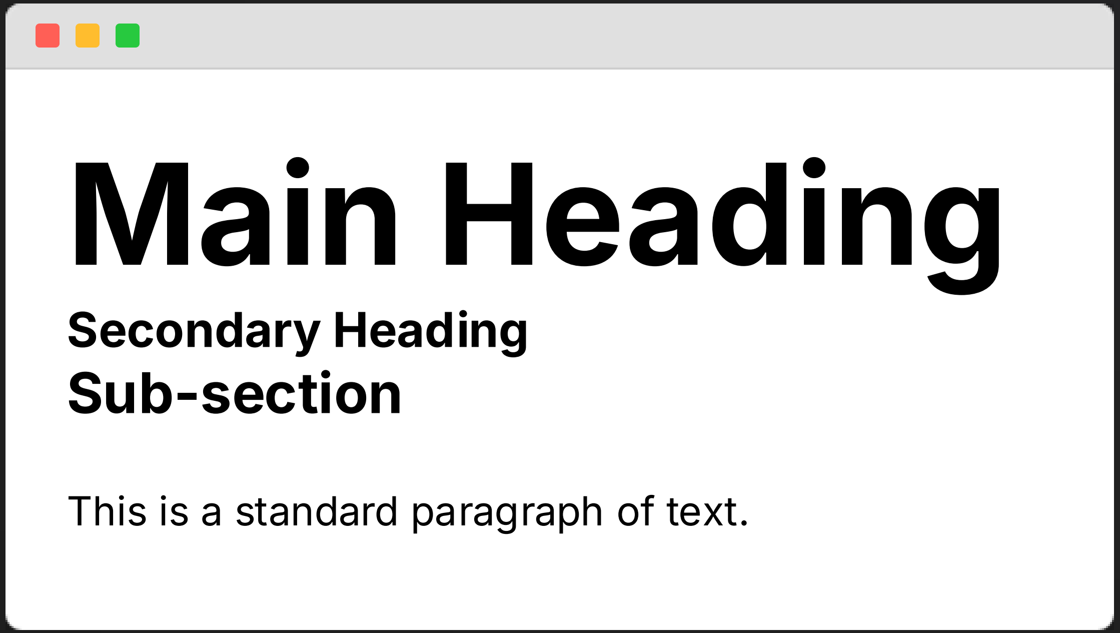
Common Types

id: Unique identifier.
class: Group identifier.
style: Inline CSS.

Text Fundamentals

```
<h1>Main Headingh1>
<h2>Secondary Headingh2>
<h3>Sub-sectionh3>
<p>This is a standard paragraph of text.p>
```

Note: Heading tags go from h1 (most important) to h6 (least important).



Main Heading

Secondary Heading

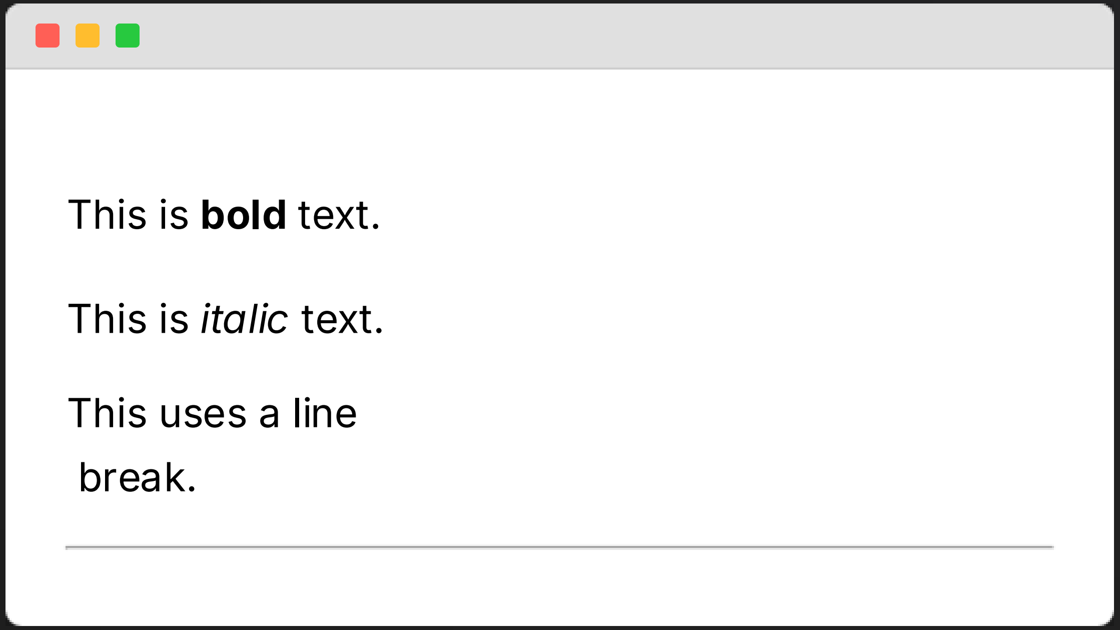
Sub-section

This is a standard paragraph of text.

Text Formatting

```
<p>
  This is <strong>boldstrong> text.
  This is <em>italicem> text.
  This uses a line<br>break.
p>
<hr>
```

Use **strong** for importance and **em** for emphasis.


This is **bold** text.

This is *italic* text.

This uses a line
 break.

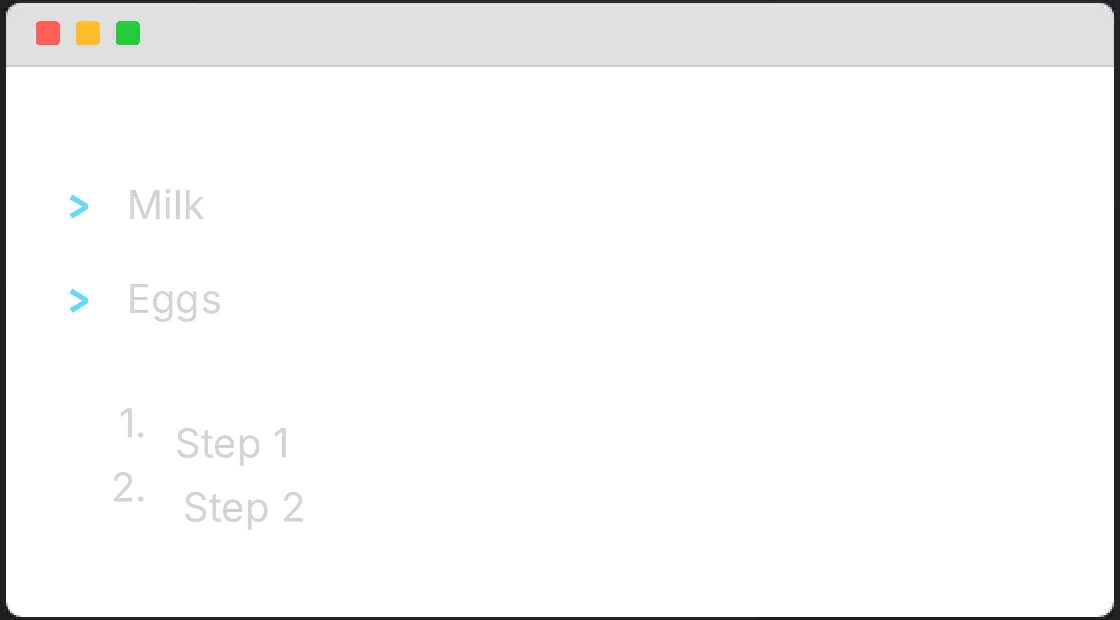
Lists: Ordered & Unordered

Unordered List (ul)

```
<ul>
  <li>Milkli>
  <li>Eggsli>
ul>
```

Ordered List (ol)

```
<ol>
  <li>Step 1li>
  <li>Step 2li>
ol>
```



Milk

Eggs

1. Step 1
2. Step 2

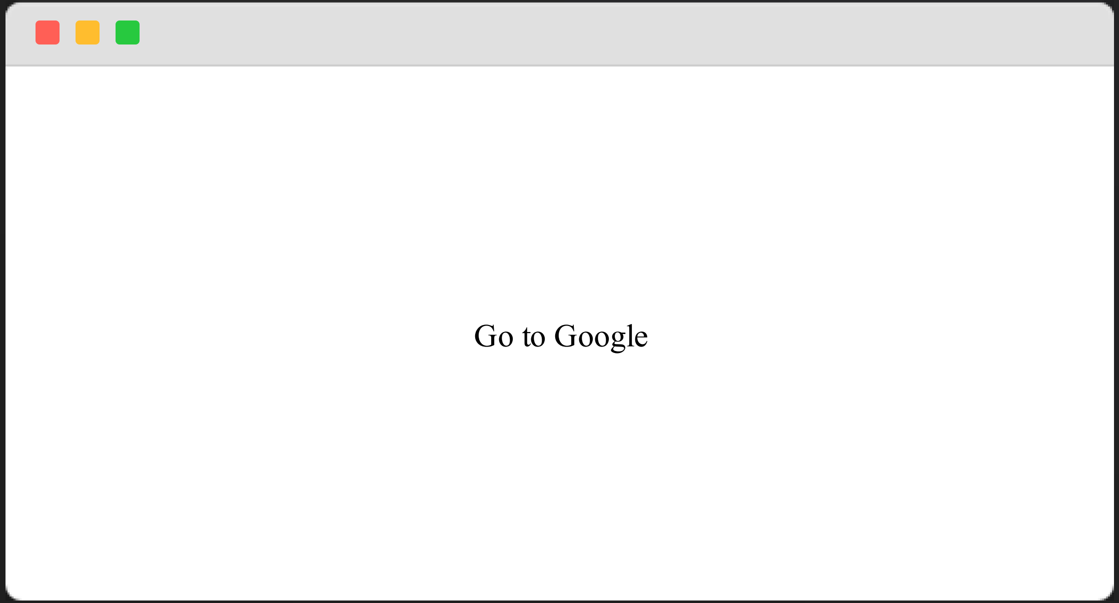
Links & Anchors

The tag defines a hyperlink.

```
<a href="https://google.com">
  Go to Google
a>
```

Open in New Tab

```
<a href="..." target="_blank">
  External Link
a>
```



Go to Google

Images

```html
<img
  src="cat.jpg"
  alt="A cute cat"
  width="300" />
```

- **src:** The source path (URL or file).
- **alt:** Alternative text for screen readers and when images fail to load.
- **Self-closing:** No closing tag needed.

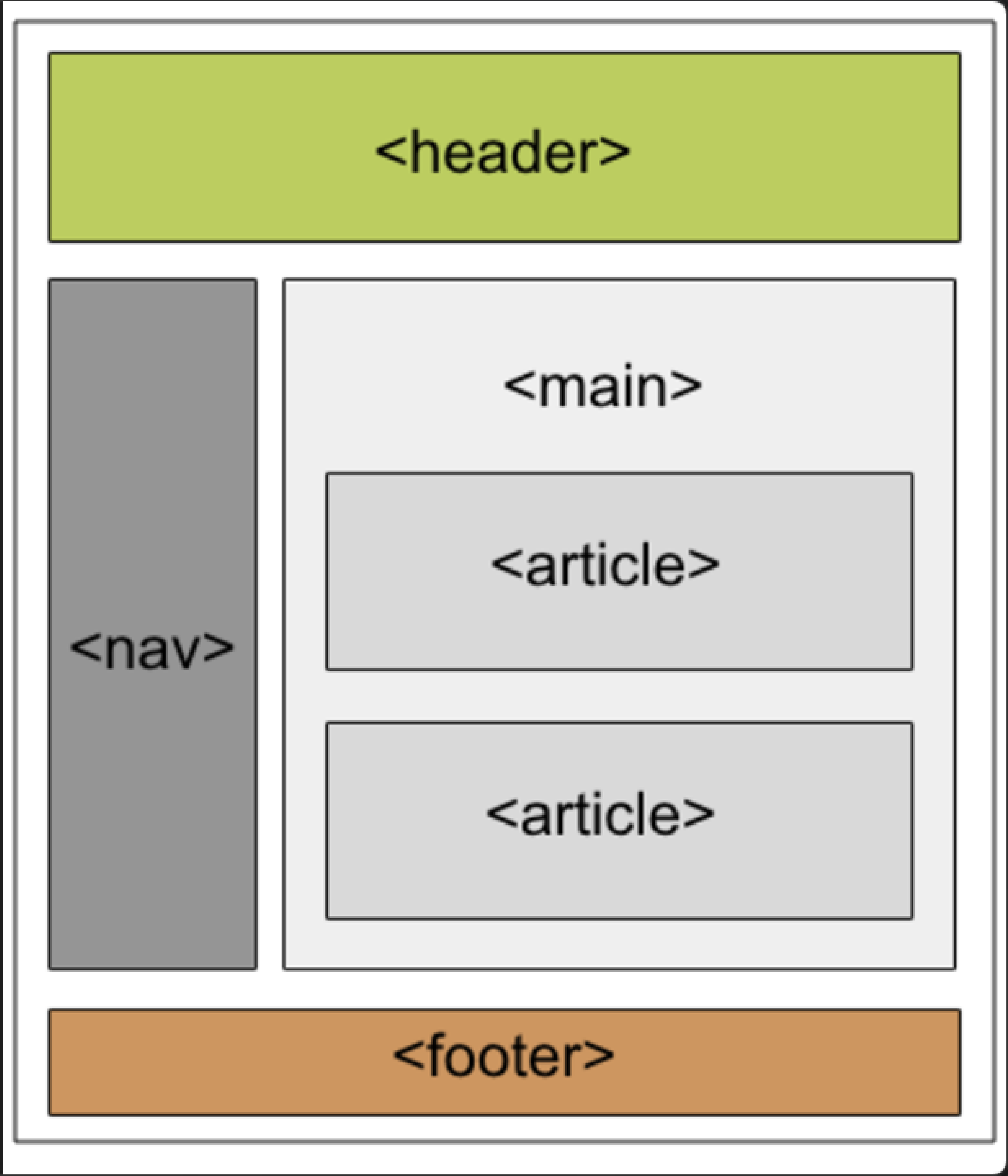
Semantic HTML

Meaningful Markup

Semantic elements clearly describe their meaning to both the browser and the developer. This improves **SEO** and **Accessibility**.

Key Tags:

-
-
-
- &
-



Tables

```
<table>
  <tr>
    <th>Nameth>
    <th>Roleth>
  tr>
  <tr>
    <td>Alicetd>
    <td>Devtd>
  tr>
table>
```

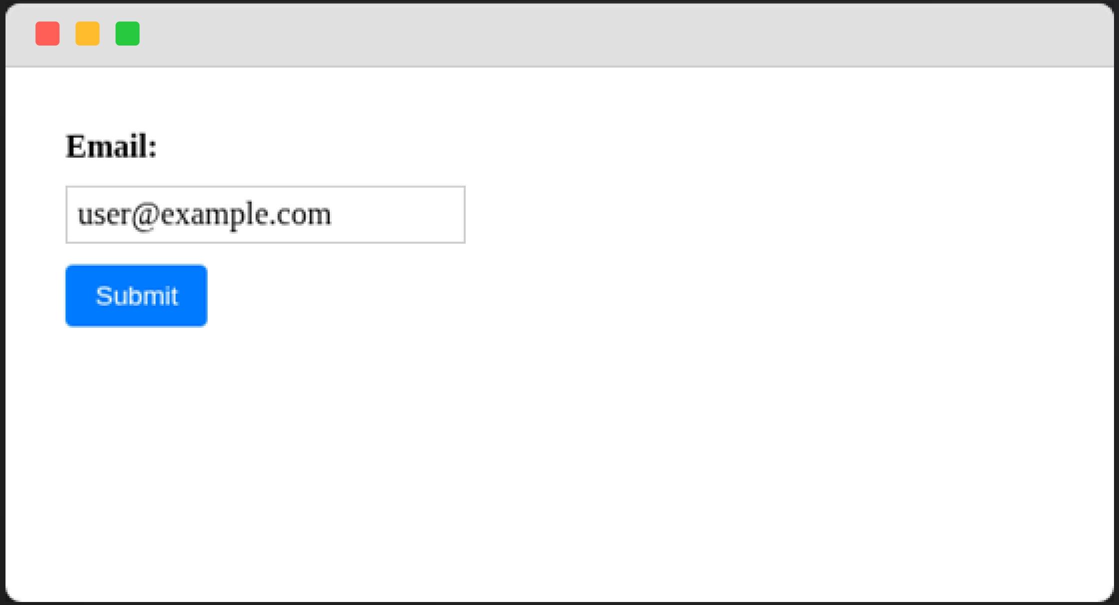
Result:

Name	Role
Alice	Developer
Bob	Designer

tr: Table Row | **th:** Table Header | **td:** Table Data

Basic Forms

```
<form>
  <label>Email:label>
  <input type="email" />

  <button>Submitbutton>
form>
```

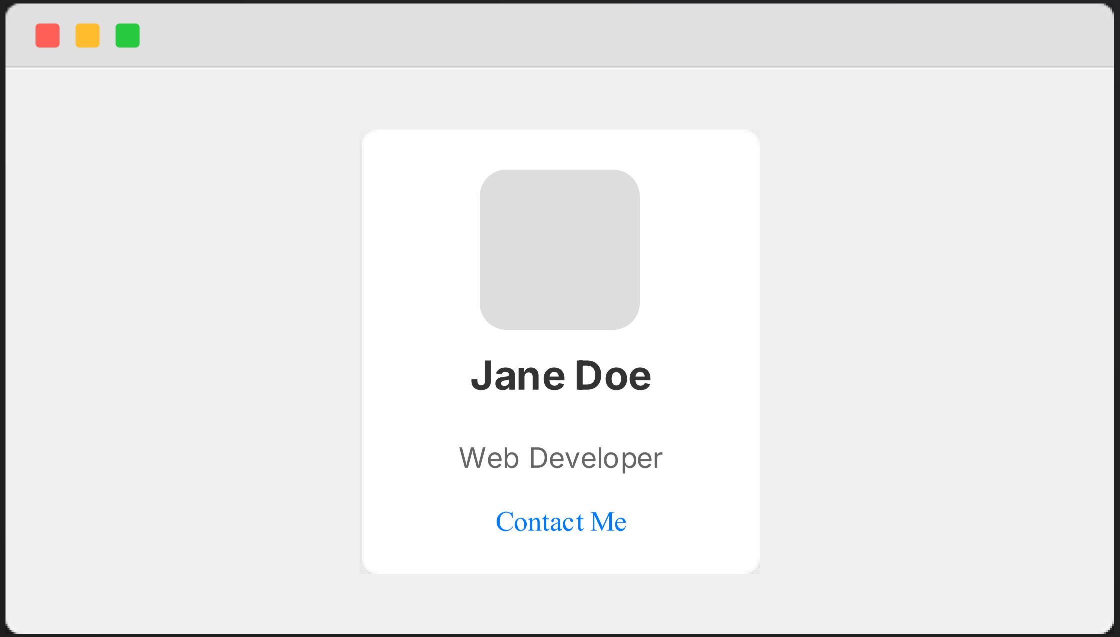


Email:

user@example.com

Practical Example: Profile Card

```
<di  clas ="card >
 v<ims sr ="pic.jpg  />
  <h >Jane Doeh >
  g  c   "
  <p>Web Developer p>
  2          2
  <a hre ="# >Contacta>
di >  f     "
v
```


Jane Doe

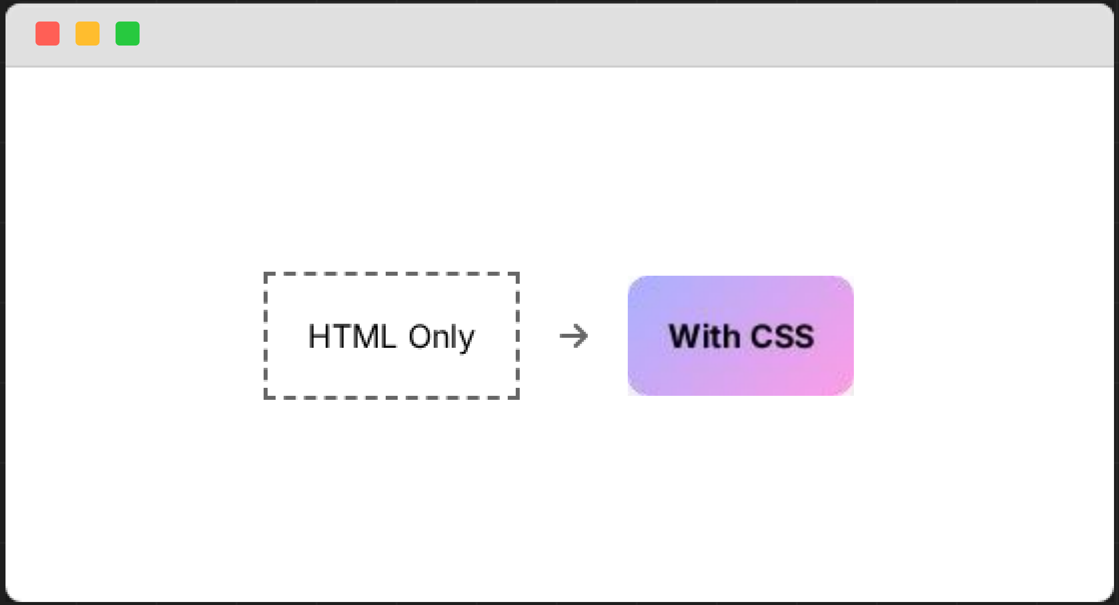
Web Developer

Contact Me

CSS Styling

What is CSS?

- **C**ascading **S**tyle **S**heets.
- While HTML provides the structure (the skeleton), CSS provides the style (the skin and clothes).
- It controls colors, fonts, layouts, and responsiveness.
- It allows you to style multiple pages with a single file.



The Syntax

```css
h1 {
  color: blue;
  font-size: 12px;
}
```

Selector: Which HTML element to target.

Property: What aspect to change.

Value: The new setting.

The Rule Set

A CSS file is simply a collection of these "rule sets". The browser reads them from top to bottom and applies the styles to the matching HTML elements.

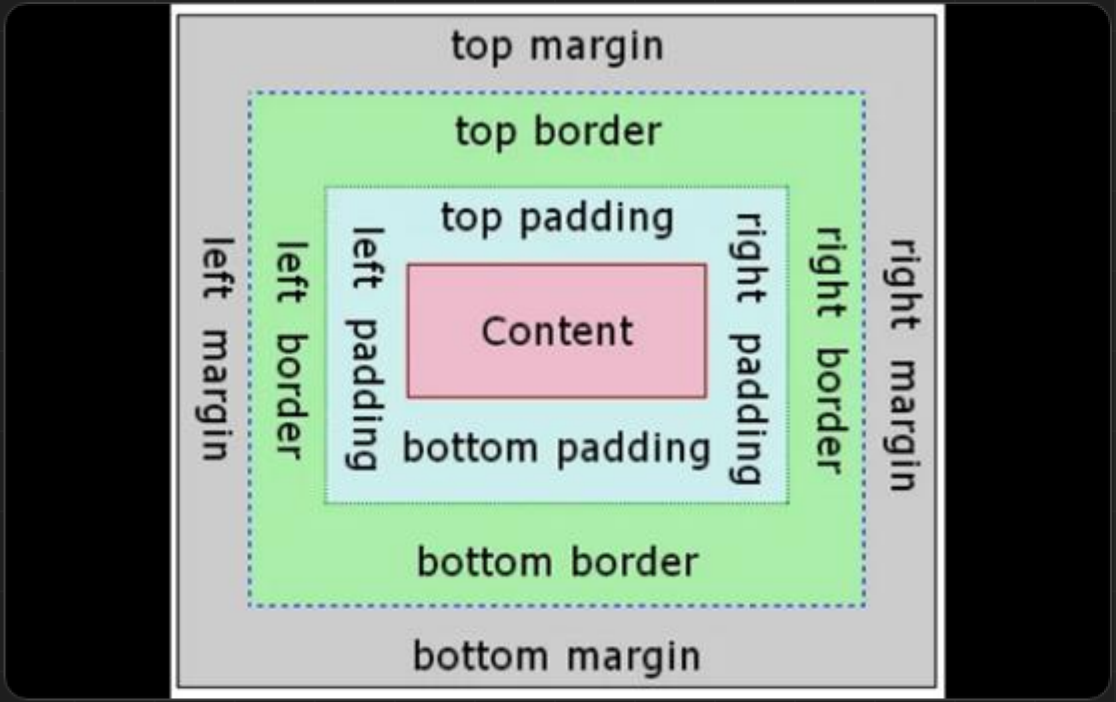
Selectors: Who are we styling?

Selector	Syntax	Description
Element	p { ... }	Targets **all** tags. Good for defaults.
Class	.btn { ... }	Targets elements with class="btn". Reusable.
ID	#header { ... }	Targets the **one** element with id="header". Unique.
Universal	* { ... }	Targets absolutely everything.

The Box Model

Every element in HTML is essentially a box.

- **Content:** The actual text or image.
- **Padding:** Space *inside* the border (clears space around content).
- **Border:** A line going around the padding.
- **Margin:** Space *outside* the border (pushes other elements away).



Colors & Units

Colors

Keywords: red, blue
Hex: #ff0000
RGB: rgb(255, 0, 0)
RGBA: rgba(0, 0, 0, 0.5) (Transparency)

Units

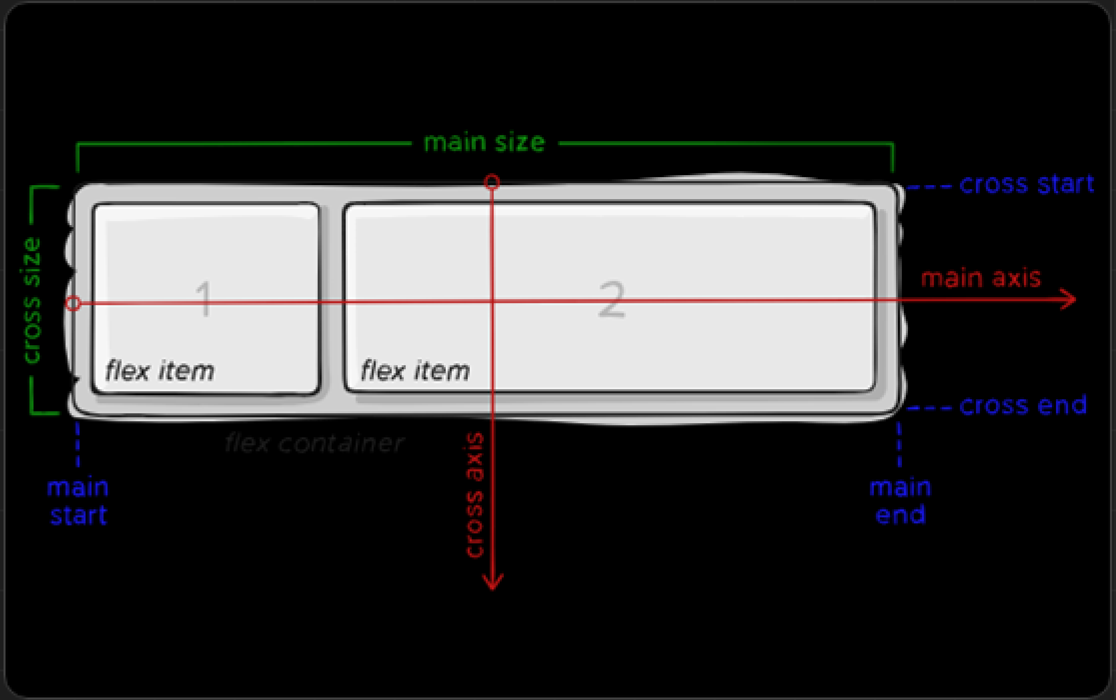
px: Pixels (Absolute)
%: Percentage (Relative to parent)
rem: Root Em (Relative to base font size, best for accessibility)

Layout: Flexbox

Flexbox is a one-dimensional layout method for laying out items in rows or columns.

```css
.container {
  display: flex;
  justify-content: center;
  align-items: center;
}
```

Perfect for centering items!

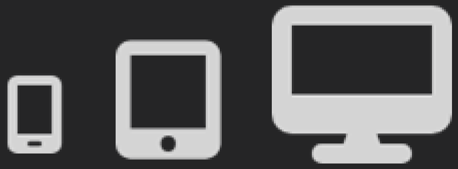


Responsive Design

Media Queries

Allow you to apply styles only when certain conditions are met, like the screen width being small (mobile).

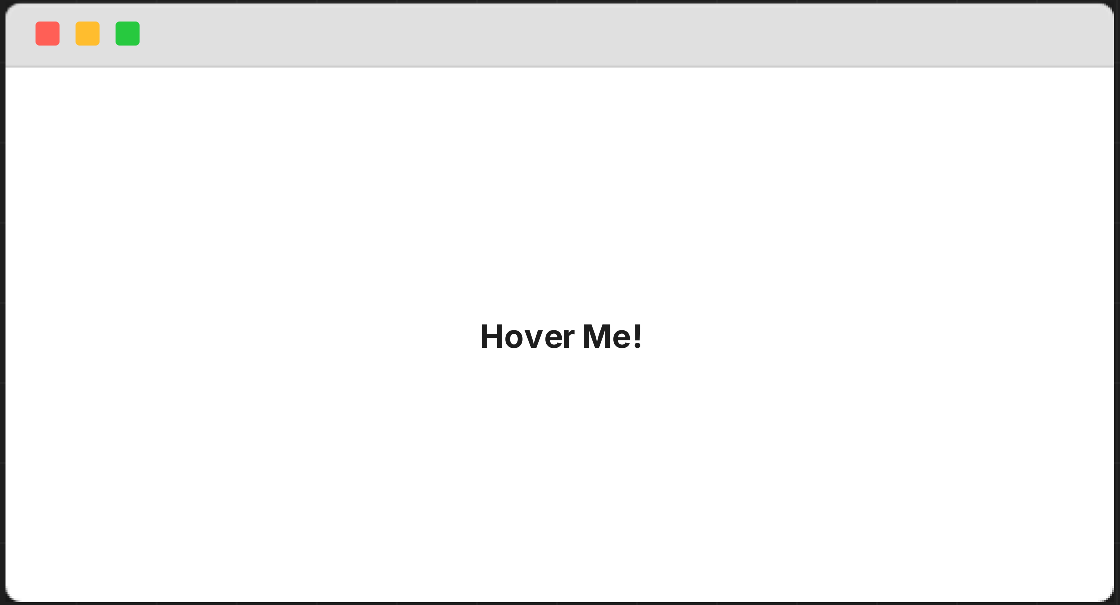
```css
.box { width: 100%; }

/* On screens larger than 768px */
@media (min-width: 768px) {
  .box { width: 50%; }
}
```



One code base, many devices.

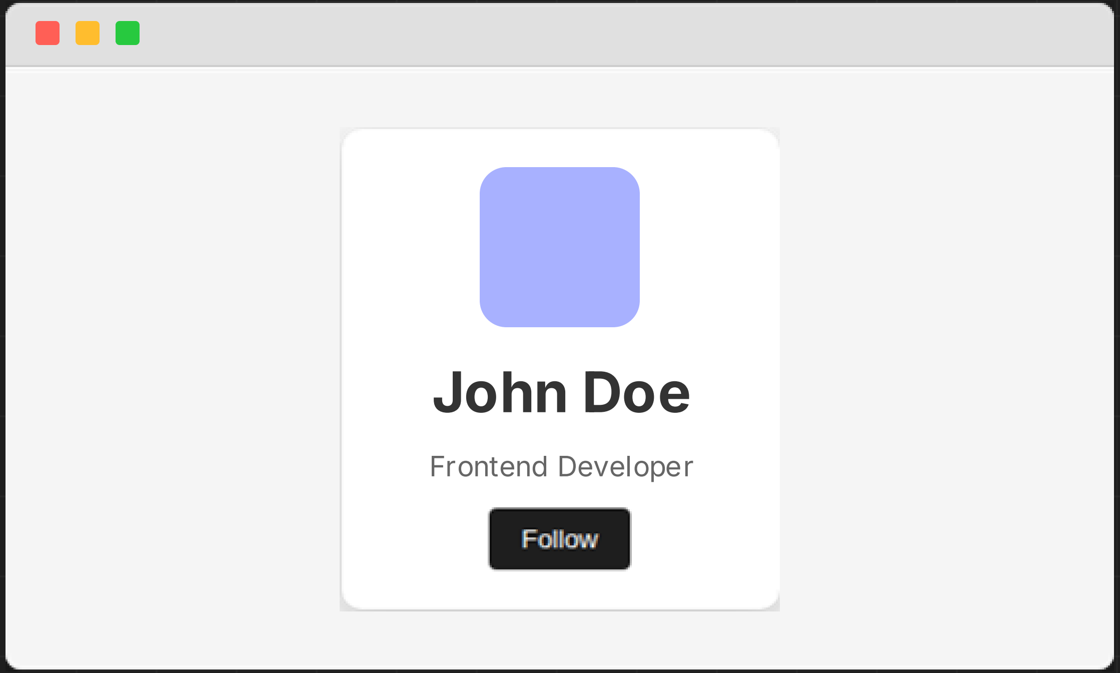
Animation: Hover Effects

```
.btn {
  background: blue;
  transition: 0.3s;
}

.btn:hover {
  transform: scale(1.1);
  background: darkblue;
}
```



Practical Example: Card Component

```
.car  {
d border-       : 1 p ;
  paddin : 2 p ;  0 x
  radius
  box-      0:x0 5px 15px              ;
  g
  text-     : cente ;
  shadow      rgba(0,0,0,0.2)
} align         r
.card     {
img border-       : 5 %;
  widt : 8 p ;    0
  radius
} h       0 x
```



JavaScript
JS

What is JavaScript?

- JavaScript is the **programming language** of the web.
- If HTML is the Skeleton and CSS is the Skin, JavaScript is the **Brain and Muscle**.
- It allows you to change content, attribute values, and styles in response to user events.
- It runs directly in the browser (Client-side).

Interactivity

Form validation, interactive maps, animated graphics, and much more.

Variables

Variables are containers for storing data values.

```
// The modern way
let score = 10; // Can change
const pi = 3.14; // Cannot change

// The old way (avoid)
var name = "John";
```

Key Concepts

- Use const by default.
- Use let only if you know the value will change (like a counter).
- Variable names are case-sensitive (myVar vs myvar).

Data Types

String

Text data.
"Hello World"

Number

Integers or decimals.
42 or 3.14

Boolean

True or False logic.
true / false

Object

Collections of data.
{ name: "John" }

Functions

A block of code designed to perform a particular task.

```
function greet(name) {
  return "Hello " + name;
}

// Calling the function
let msg = greet("Alice");
```

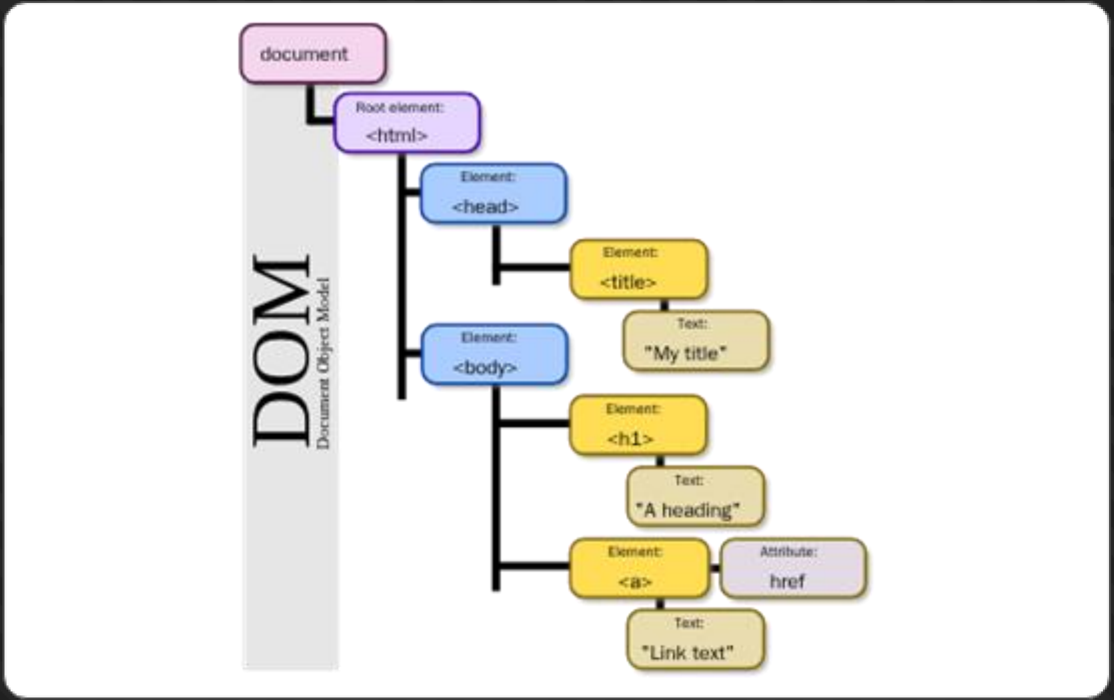
Arrow Functions (ES6)

A shorter syntax for writing functions.

```
const add = (a, b) => a + b;
```

The DOM (Document Object Model)

- The DOM is a tree-like representation of your HTML page.
- JavaScript uses the DOM to access and manipulate HTML elements.
- **document.querySelector()** is your best friend.



Handling Events

Events are things that happen to HTML elements (clicks, mouseover, keypress).

```
const btn = document.querySelector("button");

btn.addEventListener("click", () => {
  alert("Button clicked!");
});
```

Control Flow: If / Else

```
let hour = 14;

if (hour < 12) {
  console.log("Good Morning");
} else {
  console.log("Good Afternoon");
}
```

Logic Operators

- && (AND): Both must be true.
- || (OR): One must be true.
- ! (NOT): Inverses the boolean.
- ===: Strict equality check.

Loops

For Loop

Repeat code a specific number of times.

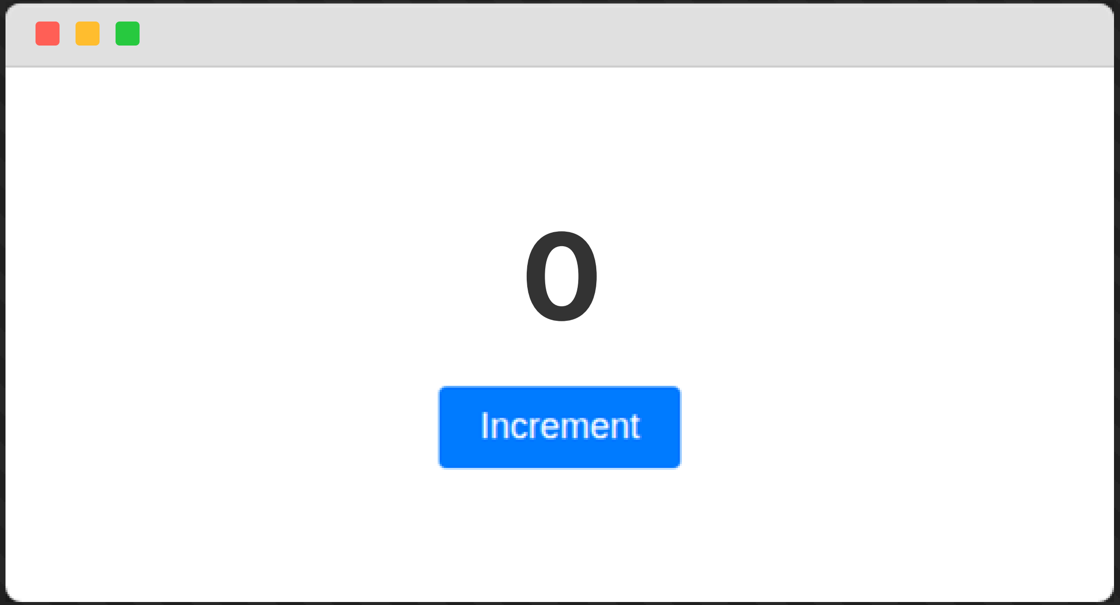
```
for (let i = 0; i < 5; i++) {
  console.log(i);
}
```

Array Loop (forEach)

```
let fruits = ["Apple", "Banana"];

fruits.forEach(item => {
  console.log(item);
});
```

Practical Example: Counter App

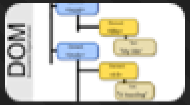
```
let count = 0;
const display = document.querySelector("#num");
const btn = document.querySelector("#btn");

btn.addEventListener("click", () => {
  count++;
  display.innerText = count;
});
```



Q & A

Now go build something interactive!

Image Sources



https://upload.wikimedia.org/wikipedia/commons/5/5a/DOM-model.svg

Source: en.wikipedia.org