

Front-end Workshop

Damiano Shushku
Marco Coppola

Introduction

Brief overview of HTML and CSS

- HTML : (HyperText Markup Language) structures web content using tags, defining elements like headings, paragraphs, and links.
- CSS : (Cascading Style Sheets) enhances HTML by styling web pages, controlling layout, colors, fonts, and visual aspects for a better user experience. Together, HTML and CSS form the foundation of web development.

HTML5 Fundamentals

Document Structure

HTML as other document structures, has to follows some structure rule.

- <!DOCTYPE> :

Defines the “language” used on the document, in this case HTML5

- <html> :

Defines the beginning of the html content. (</html> closes it)

- <head> :

Contains all the elements non visible to the user :

<title> = to store website name or content to be displayed.

<link> = to link to external style sheets or to add a favicon to your website.

<meta> = 1. to store data about website, organisation , creator/ owner

2. for responsive website via attributes

3. to tell compatibility of html with browser

<script> = to add javascript file.

- <body> contains all the content that will displayed by the user.

Tag behaviours :

Most of the tags would need a closing one, for example <html>...</html>

In others like <input /> they have to be self closing one.

Body elements

Semantic Elements

Some elements do not have practical usage, they do not have a visual impact but they need to be used to have a well structured HTML code.

<header>, <nav>, <article>, <main>, <aside>, <section>, <footer>, etc.

Following these elements will allow you to have a better SEO indexing.

What is the SEO (search engine optimization)

The SEO it is a set of activities that allows you to improve the position of a website among the results of a search engine.

Other tags.

Alcuni tag hanno invece delle funzionalità predefinite:

- We can handle some Multimedia using this tags : , <audio>, <video> tags.
- We can handle Forms using this tags : <form>, <input>, <textarea>, <select>, <button>.

Exercise 1

Let's code

```
<html lang="it"><head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Titolo della Scheda</title>
</head>
<body>
  <h1>Titolo della Pagina</h1>
  <p>Paragrafo</p>
  
  <input type="text" name="name" id="name">
  <button>Cliccami</button>
  <a href="https://www.google.com">Link interno alla pagina</a>
  <a href="https://www.google.com" target="_blank">Link Esterno alla pagina</a>
</body>
</html>
```

CSS3 Basics

Selectors and Styles

CSS syntax consists of selectors, properties, and values.

Selectors target HTML elements, while properties define the styling aspects such as color or font. Values, assigned to properties, specify the desired characteristics.

The structure involves a selector followed by a block of property-value pairs enclosed in curly braces, creating a rule like

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

Element, class, and ID selectors.

#id {...} : id selector

.class {...} : class selector,

tag {...} : tag selector

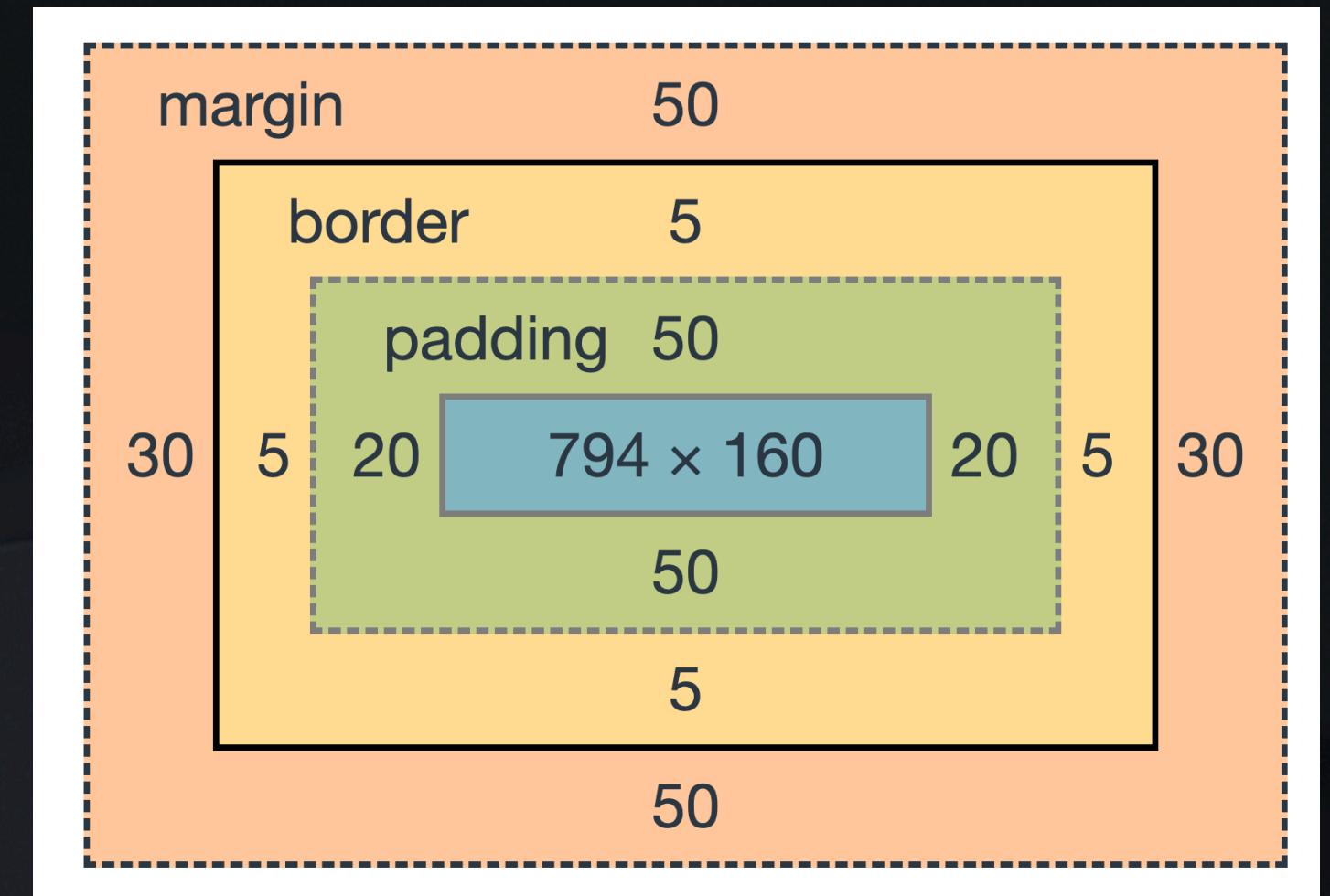
- Basic styling properties :

color, font-size, margin, padding, etc.

Box Model

The box model in CSS defines how elements are structured in terms of content, padding, border, and margin.

1. Content: Represents the actual content of the element.
2. Padding: Clears space between the content and the border.
3. Border: A border surrounding the padding (if any).
4. Margin: Clears space outside the border, affecting the spacing between elements.



Media queries

Media queries in CSS allow developers to apply different styles based on various characteristics of the user's device or viewport

@media (dimensions) {...}

```
@media only screen and (min-width: 768px) {  
    body {  
        background-color: "red";  
    }  
}
```

Advanced CSS3

Flexbox

Flexbox, or the Flexible Box Layout module in CSS, is a powerful layout model that provides a more efficient way to design, align, and distribute space among items in a container, even when their size is unknown or dynamic.

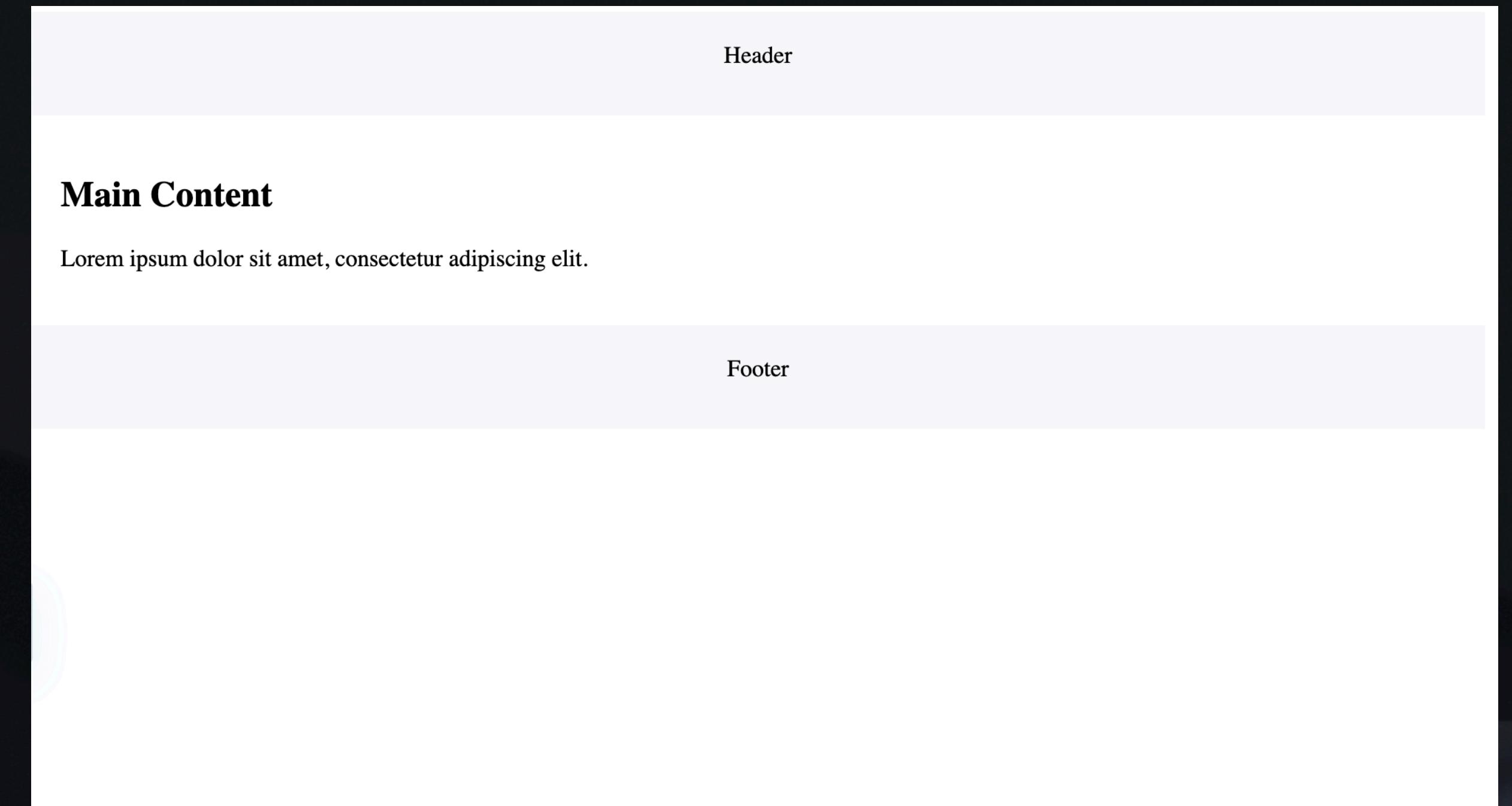
```
.container {  
  display: flex;  
  justify-content: center;  
}
```

Click Me

CSS Grid

CSS Grid is a layout system in CSS that allows to create two-dimensional grid-based layouts.

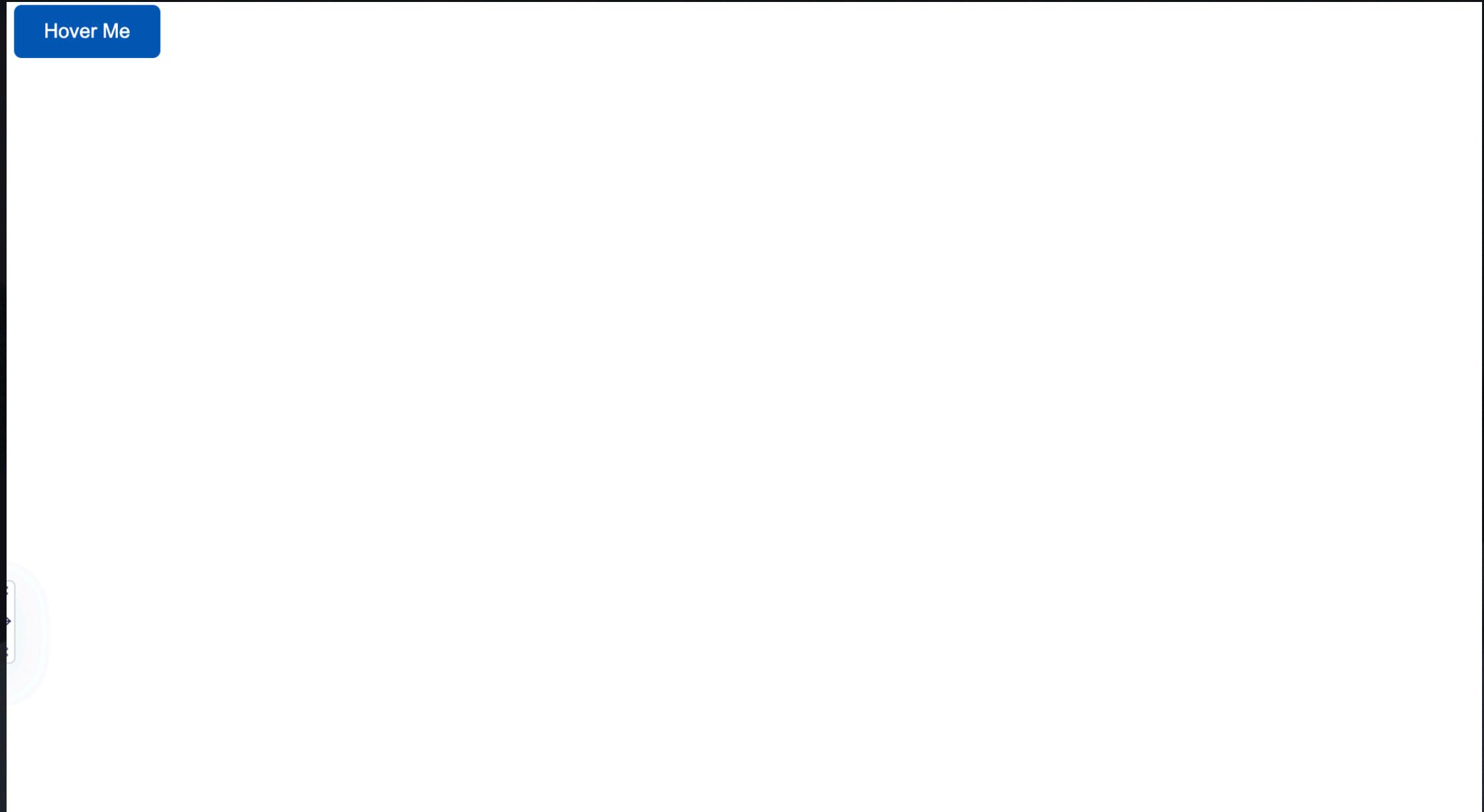
```
.container {  
    display: grid;  
    grid-template-rows: 70px 1fr 70px;  
    grid-template-columns: 1fr;  
}
```



Transitions and Animations

Transitions and animations in CSS add dynamic and engaging elements to web interfaces. Here's the scope of using transitions and animations:

```
.button {  
background-color: #007bff;  
color: white;  
padding: 10px 20px;  
border: none;  
border-radius: 5px;  
cursor: pointer;  
transition: background-color 0.5s ease-in-out;  
}  
  
.button:hover {  
background-color: #0056b3;  
}
```



Animations:

Keyframe Animations: CSS animations use keyframes to define a series of style changes over time.

```
<!DOCTYPE html>
<html>
<head>
<title>CSS Animation Example</title>
<style>
.button {
background-color: #007bff;
color: white;
padding: 10px 20px;
border: none;
border-radius: 5px;
cursor: pointer;
animation: pulsate 2s infinite ease-in-out;
}
@keyframes pulsate {
0% {
transform: scale(1);
}
50% {
transform: scale(0.9);
}
100% {
transform: scale(1);
}
}
</style>
</head>
<body>
<button class="button">Pulsating Button</button>
</body>
</html>
```

Custom Properties (Variables)

CSS ci permette di dichiarare ed utilizzare delle variabili, per esempio possiamo creare delle palette di colori principali secondari e via dicendo per poi utilizzarle sempre.

```
:root {  
  --nome: #fffff;  
}  
  
body {  
  color: var(--nome);  
}
```

Final Exercise

Let's go golfin'

