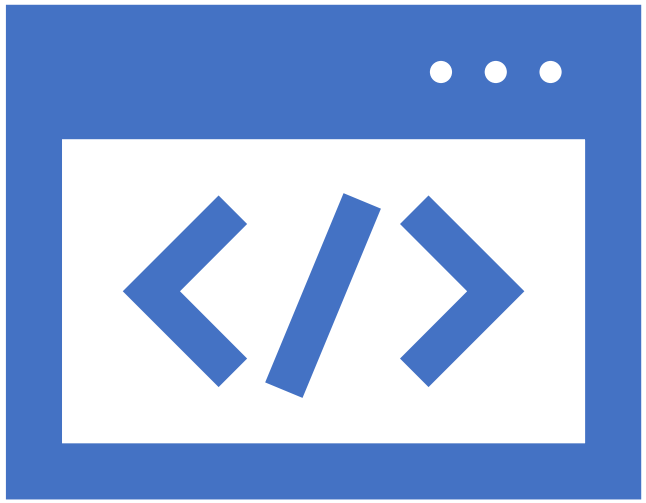
Lecture 2 Html and Css advanced

- Write by / Dev.Youssef Khaled



HTML Overview

- HTML stands for HyperText Markup Language. It is the standard markup language for creating web pages and web applications.

CSS Overview

- CSS stands for Cascading Style Sheets. It is a style sheet language used for describing the look and formatting of a document written in HTML.

Semantic HTML

- Semantic HTML refers to the practice of using HTML tags to convey the meaning and structure of content, making it more accessible to both browsers and developers. Utilizing elements like `<header>`, `<article>`, and `<nav>` enhances document clarity, fostering improved search engine optimization and overall user experience.

CSS Flexbox

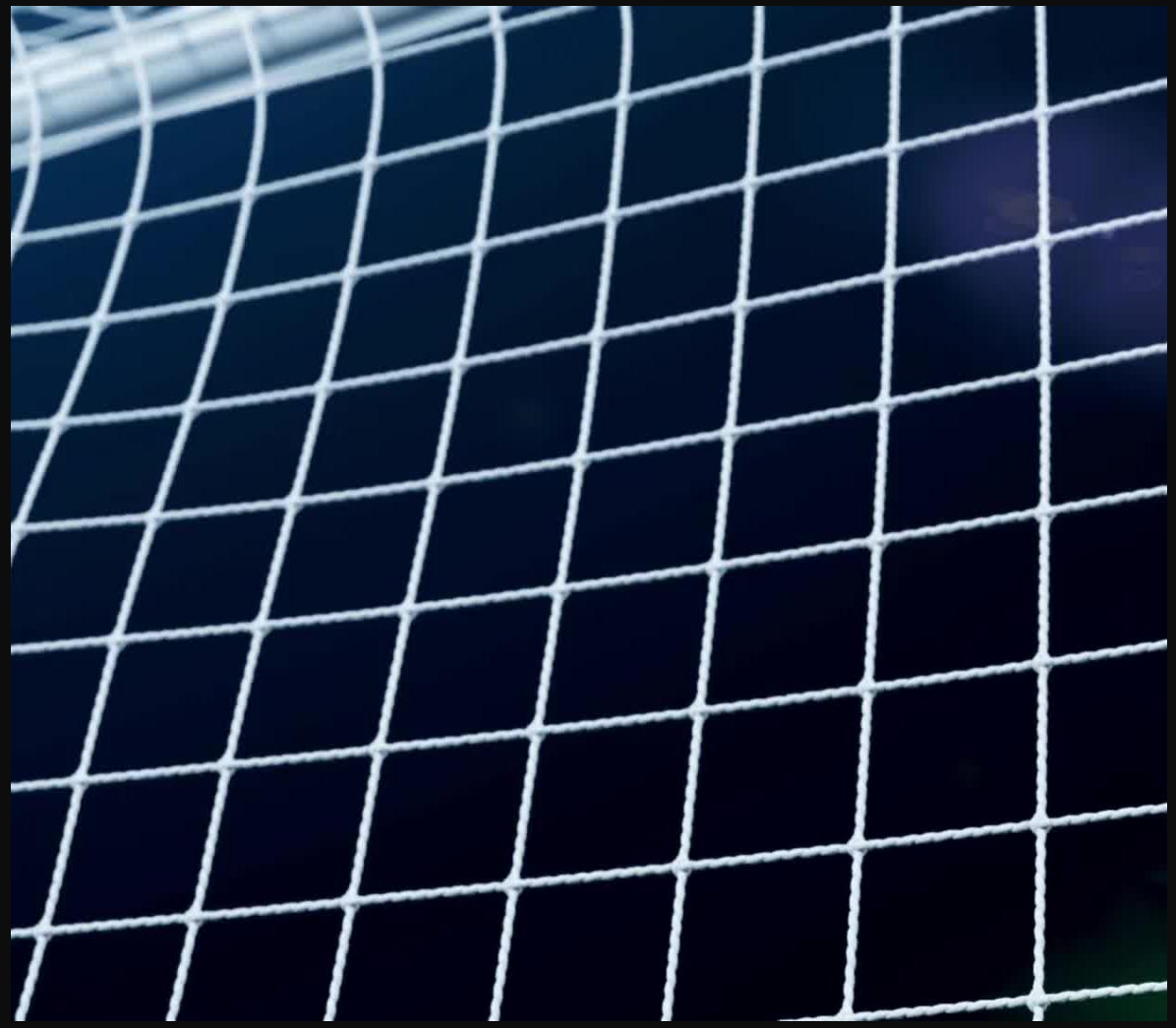
- CSS Flexbox simplifies layout design.
- With `display: flex`, it creates flexible containers where child items can easily adjust. For example, in a navigation bar, setting

`justify-content: space-between`

evenly spaces items without fixed widths, streamlining responsive design. Flexbox is a powerful tool for efficient and dynamic web layouts.

CSS Grid

- CSS Grid is a layout system that allows you to design web page layouts in a more powerful and flexible way than traditional methods. It's like a virtual grid or table that helps you organize content into rows and columns, making it easier to create complex and responsive layouts.



Example

- .grid-container { display: grid;

grid-template-rows: auto 1fr auto;

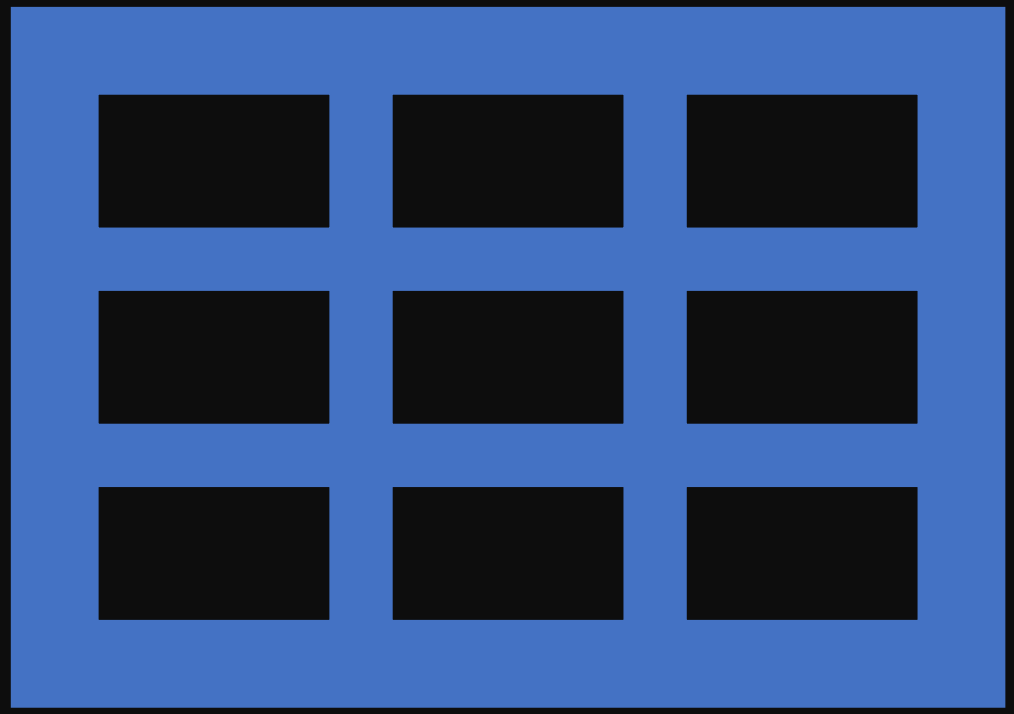
grid-template-columns: 20% 1fr;

gap: 20px;

height: 100vh;}

.grid-item { padding: 20px;

border: 1px solid #ddd; }



Responsive Web Design

- Responsive Web Design (RWD) is an approach to designing and coding websites to provide an optimal viewing experience across a wide range of devices and screen sizes. The goal is to ensure that users can easily access and navigate a website regardless of whether they are using a desktop computer, tablet, or smartphone.
- Key principles of Responsive Web Design include:
- **Fluid Grid Layout:** Instead of fixed-width layouts, RWD uses relative units like percentages for width and flexible grids to adapt the layout to different screen sizes.
- **Flexible Images and Media:** Images and other media elements are sized in relative units as well, ensuring they scale proportionally and don't overflow or shrink disproportionately on different devices.
- **Media Queries:** CSS3 media queries are used to apply styles based on the characteristics of the device, such as screen width, height, or orientation. This allows for tailored styles for different devices.

Example

- /* Tablet Styles */
- @media (max-width: 768px) {

.container {

width: 100%;

padding: 0 20px }

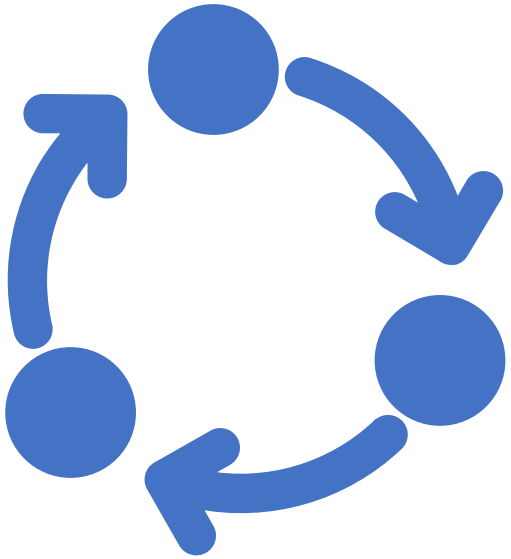
- /* Mobile Styles */
- @media (max-width: 480px)

.container{

padding: 0 10px ; }

CSS Transitions:

CSS transitions enable the gradual change of property values over a specified duration. Here's a simple example using a button that changes its background color on hover:



Example :

- button {
- background-color: #3498db;
- color: #fff;
- transition: background-color 0.3s ease;}
- button:hover {
- background-color: #e74c3c; }

- <button>Hover me</button>

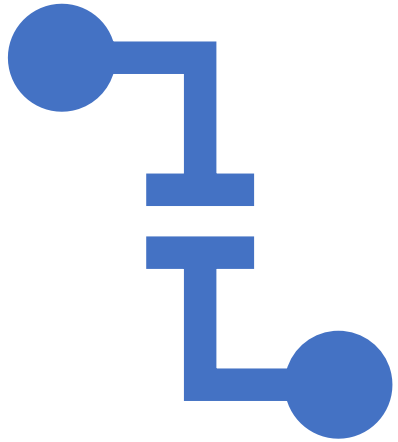
HTML Forms

- HTML forms are a crucial part of web development, allowing users to input and submit data to a web server. They consist of various form elements, such as text fields, checkboxes, radio buttons, and buttons. Here's a short explanation with a simple

Example

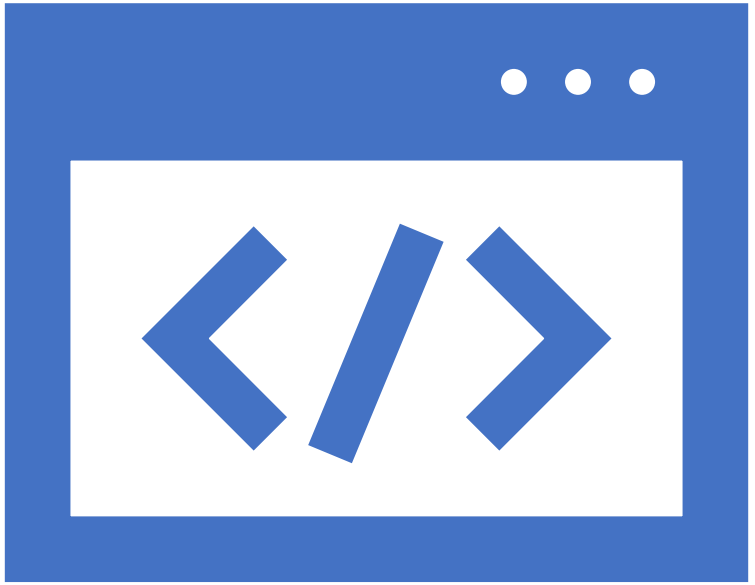
- <h2>Contact Us</h2>
- <!-- HTML form starts here -->
- <form action="/submit_form" method="post">
- <!-- Text Input -->
- <label for="name">Name:</label>
- <input type="text" id="name" name="name" required>
-
- <!-- Email Input -->
- <label for="email">Email:</label>
- <input type="email" id="email" name="email" required>

```
    <!-- Radio Buttons -->
    <label>Gender:</label>
    <input type="radio" id="male" name="gender" value="male" checked>
    <label for="male">Male</label>
    <input type="radio" id="female" name="gender" value="female">
    <label for="female">Female</label>
    <br>
    <!-- Checkbox -->
    <input type="checkbox" id="subscribe" name="subscribe" checked>
    <label for="subscribe">Subscribe to newsletter</label>
    <br>
    <!-- Textarea for longer text -->
    <label for="message">Message:</label>
    <textarea id="message" name="message" rows="4" required></textarea>
    <br>
    <!-- Submit Button -->
    <input type="submit" value="Submit">
</form>
```



CSS Frameworks (e.g., Bootstrap)

- CSS frameworks, such as Bootstrap, are pre-prepared libraries of CSS code and sometimes JavaScript, designed to simplify and speed up the process of web development. They provide a set of ready-to-use styles, components, and layout grids, allowing developers to create visually appealing and responsive websites with less effort.



Example

```
<div class="container">
  <div class="row">
    <div class="col-sm-6">
      <!-- Content for the first column -->
    </div>
    <div class="col-sm-6">
      <!-- Content for the second column -->
    </div>
  </div>
</div>
```