1. CSS Introduction

What is CSS?

CSS (Cascading Style Sheets) is a styling language used to control the **appearance, layout, spacing, and design** of web pages.

HTML gives structure,
CSS gives style.
Together they make a complete webpage.

How CSS Works

CSS selects **HTML elements** and applies styling rules to them.

There are 3 ways to apply CSS:

---

1. Inline CSS

Applied directly to an element.

```
<p style="color: red;">Hello</p>
```

✔ Quick
✘ Not reusable
✘ Hard to maintain

---

2. Internal CSS

Defined inside the `<style>` tag.

```
<style>
  p {
    color: red;
  }
</style>
```

✔ Good for small projects

---

3. External CSS (Recommended)

CSS stored in a separate file.

index.html

```
<link rel="stylesheet" href="style.css">
```

style.css

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

✔ Industry standard
✔ Reusable
✔ Clean and organized

□ 2. CSS Syntax

CSS syntax follows this structure:

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

Selector

Chooses which HTML element(s) to style.

Property

The feature of styling you want to change (color, font-size, margin, etc.).

Value

The setting for that property.

Example

```
h1 {
  color: blue;
  font-size: 30px;
  text-align: center;
}
```

This means:

- Select all `<h1>` elements
- Make their text blue
- Increase font size
- Align text to center

□ 3. CSS Selectors (Basic to Advanced)

Selectors tell the browser *which* HTML element to style.

✔ Element Selector

Targets all elements of a specific type.

```
p {
  color: green;
}
```

✔ Class Selector

Used for multiple elements, starts with a dot `.`

```
.card {
  background: lightgray;
}
```

HTML:

```
<div class="card"></div>
```

💡 **Classes are reusable**
Use them when many elements share the same style.

✔ ID Selector

Unique selector, starts with `#`

```
#header {
  background: black;
  color: white;
}
```

HTML:

```
<h1 id="header">Welcome</h1>
```

💡 **ID must be used only once per page**

✔ Attribute Selector

Select elements based on attribute value.

```
input[type="email"] {
  border: 2px solid blue;
}
```

Meaning:

- Find all `<input>` elements
- Check if `type="email"`
- Apply border

✔ Descendant Selector

Selects elements inside another element.

```
ul li {
  color: purple;
}
```

HTML:

```
<ul>
  <li>Item</li>
</ul>
```

✔ Child Selector (>)

Direct children only.

```
.container > p {
  color: orange;
}
```

✔ Adjacent Sibling Selector (+)

Selects the element right after another.

```
h2 + p {
  color: red;
}
```

✔ General Sibling Selector (~)

Selects all siblings after an element.

```
h2 ~ p {
  font-style: italic;
}
```

✔ Pseudo-class

Represents element **states** (hover, focus, active, visited...).

```
button:hover {
  background: navy;
  color: white;
}
```

✔ Pseudo-element

Targets specific **parts** of an element.

```
p::first-letter {
  font-size: 40px;
  font-weight: bold;
}
```

Meaning:

- Style only the **first letter** of every `<p>`

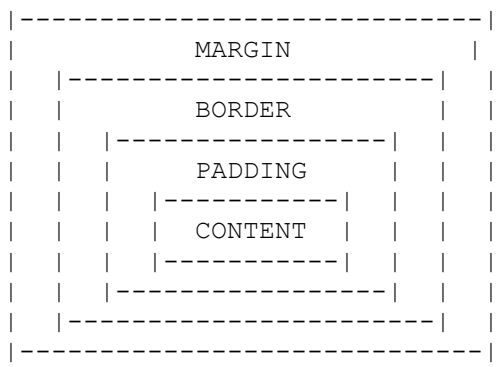
□ 4. CSS Box Model (Important Concept!)

Every HTML element is treated like a **box**.

The box has 4 parts:

1. **Content** – text or image inside
2. **Padding** – space *inside* the box
3. **Border** – line surrounding the padding
4. **Margin** – space *outside* the box

Box Model Diagram



Example

```
.box {
  width: 300px;
  padding: 20px;
  border: 3px solid black;
  margin: 30px;
}
```

This means:

- `width` = content width
- `padding` = adds inside space
- `border` = adds line
- `margin` = adds outer space

☆ IMPORTANT: Actual Width Calculation

```
width + left padding + right padding + left border + right border
```

✔ box-sizing (Professional Use)

```
* {
  box-sizing: border-box;
}
```

Meaning:

- Padding and border are included *inside* the width
- Keeps layout clean
- Used in every modern website

---

5. Margin

Margin adds **space outside** an element.

Single-value margin

```
margin: 20px;
```

Different sides

```
margin-top: 10px;
margin-bottom: 20px;
margin-left: 15px;
margin-right: 5px;
```

Shorthand

```
margin: 10px 20px 30px 40px;
/* top right bottom left */
```

☆ Center an element horizontally

Very common:

```
margin: 0 auto;
width: 300px;
```

☆ Margin Collapse (Important concept)

When two vertical margins meet, they combine into the **largest** one.

---

6. Padding

Padding adds **space inside** the element between content and border.

Single value

```
padding: 20px;
```

Different sides

```
padding-top: 10px;
padding-left: 20px;
```

Shorthand

```
padding: 10px 15px 20px 25px;
```

💡 Padding does **not** collapse like margin.
Padding always adds extra space.

□ 7. Border

Border is the line around an element.

Basic border

```
border: 2px solid black;
```

Border styles

```
border-style: dotted;
border-width: 4px;
border-color: red;
```

Rounded corners

```
border-radius: 10px;
```

Make a circle

```
.circle {
  width: 120px;
  height: 120px;
  background: orange;
  border-radius: 50%;
}
```

□ 8. Real-World Practice Example

HTML

```
<div class="profile-card">
  <img src="https://via.placeholder.com/100" alt="">
  <h2>John Doe</h2>
  <p>Front-End Developer</p>
</div>
```

CSS

```
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}

.profile-card {
  width: 260px;
  margin: 40px auto;
  padding: 25px;
  border: 2px solid #333;
  border-radius: 12px;
  text-align: center;
  background: #fafafa;
}

.profile-card img {
  width: 100px;
  border-radius: 50%;
  margin-bottom: 15px;
}

.profile-card h2 {
  font-size: 22px;
  margin-bottom: 8px;
}

.profile-card p {
  color: #555;
  font-size: 16px;
}
```

- ✔ Uses selectors
- ✔ Uses margin, padding
- ✔ Uses border
- ✔ Uses box model
- ✔ Looks like a real UI component