Day 2 – CSS Fundamentals and Responsive Design

Objective: Learn how to style HTML pages using CSS, understand the box model, apply selectors effectively, and use Flexbox and Bootstrap to create responsive layouts.

Date: 24-10-2025

1. INTRODUCTION TO CSS

CSS (Cascading Style Sheets) is used to style and visually format HTML elements. It controls layout, color, spacing, fonts, and responsiveness.

Ways to Include CSS

- 1. Inline CSS: Inside the HTML element
- 2. Hello World
- 3. **Internal CSS:** Inside <style> tag in the <head> section
- 4. <style>
- 5. p { color: red; }
- 6. </style>
- 7. External CSS: Using a .css file linked to HTML
- 8. link rel="stylesheet" href="style.css">

2. CSS SELECTORS

Selectors define which HTML elements the styles will apply to.

Types of Selectors:

Selector Type	Example	Description	
Element	p { }	Targets all elements	
Class	.container { }	Targets elements with class="container"	
ID	#header { }	Targets a specific element with id="header"	
Grouping	h1, h2, h3 { }	Targets multiple tags at once	
Universal	* { }	Targets all elements	

Example:

```
body {
  background-color: #f5f5f5;
  font-family: Arial, sans-serif;
}
```

```
h1 {
  color: darkblue;
  text-align: center;
}
.container {
  margin: 20px;
  padding: 15px;
  border: 1px solid #ccc;
}
```

3. CSS COLORS, FONTS, AND TEXT

Color Formats:

```
• Named colors: color: red;
```

• HEX: color: #ff0000;

• RGB: color: rgb(255, 0, 0);

Font Styling:

```
p {
  font-size: 16px;
  font-family: 'Verdana', sans-serif;
  font-weight: bold;
  text-align: justify;
}
```

4. BOX MODEL

Every HTML element is treated as a rectangular box consisting of:

- 1. **Content** The actual text or image
- 2. **Padding** Space between content and border
- 3. **Border** Line around the element

4. **Margin** – Space between elements

Visualization:

```
| Margin |
| Border |
| Padding |
| Content |

Example:
div {
 width: 300px;
 padding: 10px;
 border: 2px solid black;
 margin: 20px;
}
```

5. LAYOUTS USING FLEXBOX

Flexbox is a powerful CSS layout system that makes it easy to arrange items responsively.

Example:

```
.container {
    display: flex;
    justify-content: space-around; /* Align items horizontally */
    align-items: center; /* Align items vertically */
    flex-wrap: wrap; /* Wrap items if too many */
}

.box {
    background-color: #007bff;
    color: white;
    padding: 20px;
    width: 200px;
```

```
margin: 10px;

text-align: center;

}

HTML Example:

<div class="container">

<div class="box">Box 1</div>

<div class="box">Box 2</div>

<div class="box">Box 3</div>
</div>
```

6. RESPONSIVE DESIGN

Responsive design ensures that web pages look good on all screen sizes (mobile, tablet, desktop).

Media Queries:

```
@media (max-width: 768px) {
    .container {
     flex-direction: column;
    }
}
```

This changes layout for smaller screens (less than 768px wide).

7. BOOTSTRAP OVERVIEW

Bootstrap is a CSS framework used for rapid and responsive web development. It provides pre-designed components like buttons, grids, forms, and navigation bars.

Adding Bootstrap:

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">

Bootstrap Grid Example:

```
<div class="container">
    <div class="row">
    <div class="col-md-4 bg-primary text-white p-3">Column 1</div>
```

```
<div class="col-md-4 bg-success text-white p-3">Column 2</div>
<div class="col-md-4 bg-warning text-white p-3">Column 3</div>
</div>
</div>
```

Bootstrap Button Example:

```
<button class="btn btn-primary">Submit</button>
<button class="btn btn-danger">Cancel</button>
```

8. BEST PRACTICES

- Keep styling in an external .css file.
- Use class selectors for reusability.
- Maintain consistent spacing and color schemes.
- Use Flexbox or Grid instead of old layout methods (like tables).
- Always test responsiveness on different devices.

Mini Practice Task (Day 2)

Objective: Style your Day 1 HTML page using CSS and Bootstrap. **Tasks:**

- 1. Create a separate file style.css.
- 2. Apply colors, padding, and margins to sections.
- 3. Convert your student registration form into a centered, responsive layout.
- 4. Use Bootstrap buttons and table styling.

Snapshoots:

```
| Time |
```

Code: Day2_CSS_Styling.html

Code: style.css

		npusoft Training - Day 2: CSS, Flexbox & Bootstrap Styling	Portal			
Training Portal			Register Students About			
	Stude	ent Registration Form				
	Full Name:					
	Enter your name					
Enter your email						
Select Course:						
	Select					
Comments: Any additional information						
		Submit Clear	, s			
	Regi	istered Students List				
Student Name	Email	Course	Comments			
Udaya Kumar	udaya@example.com	.NET Development	Interested in backend development			
Rahul	rahul@example.com	Java Development	Wants to learn REST APIs			
About Dhruv Compusoft Dhruv Compusoft Consultancy Pvt. Ltd. is a software solutions company specializing in web development, software testing, and IT consulting. This training program aims to transform fresh graduates into full-stack .NET developers skilled in both front-end and back-end technologies.						
	© 2025	Contact: hr@dhruvts.com Dhruv Compusoft Consultancy Pvt. Ltd.				

Output: Day2_CSS_Styling.html and style.css