

National University of Computer and Emerging Sciences



Laboratory Manual

for

Web Engineering (SL3003)

Lab Instructor	Sana Ejaz
Section	6C 1,2
Semester	Spring 2025

Department of Computer Science

FAST-NU, Lahore, Pakistan

Lab 2: Introduction to CSS

Objectives

- Apply advanced CSS styling techniques to enhance web pages.
- Understand and implement the Box Model, Flexbox, and Grid for layout structuring.
- Develop responsive designs using media queries and modern CSS practices.

Instructions:

Setting up the VS Code Environment

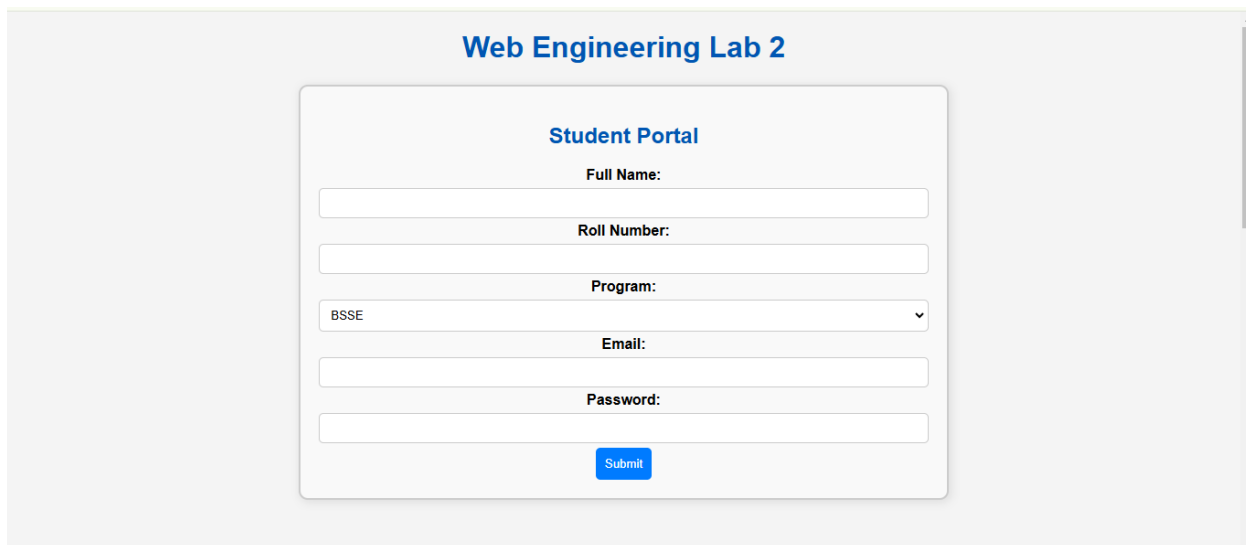
Step 1: Create Required Files

- `index.html` (for structure)
- `styles.css` (for styling)

Step 2: Link CSS in HTML: Add the following line inside the <head> tag of index.html:

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

Task 1: Add CSS to Your Student Portal Form



The screenshot displays a web browser window with a light gray background. At the top center, the text 'Web Engineering Lab 2' is written in a bold, blue font. Below this, centered, is a white rectangular box with a thin gray border and rounded corners. Inside this box, the title 'Student Portal' is centered in a bold, blue font. Below the title, there are six form fields arranged vertically, each with a label above it: 'Full Name:', 'Roll Number:', 'Program:', 'Email:', and 'Password:'. The 'Program:' field is a dropdown menu with 'BSSE' selected. At the bottom of the form box is a blue button with the word 'Submit' in white text.

For the body, we apply a `font-family` for the text and a `background-color` to set the overall page appearance. Additionally, we use `margin-top` to create spacing for the content at the top.

CSS Suggestions:

- Use `background-color` to change the page background.
- Apply `margin-top` to give the page some spacing from the top.

Form Container Styling

We style the form container to give it a clean, modern look. `width`, `padding`, `border-radius`, and `box-shadow` are commonly used properties for forms.

CSS Suggestions:

- Use `width` to control the size of the container.
- `box-shadow` can give depth and make your form stand out.

Styling Inputs and Select Elements

We use padding, margin, and border-radius to make the input fields look clean and accessible. By setting the `width` property, we can make sure the inputs take up the full width of the container.

CSS Suggestions:

- Set `width: 100%` to make the input fields fill the container.
- `padding` makes the form fields more comfortable for users to interact with.

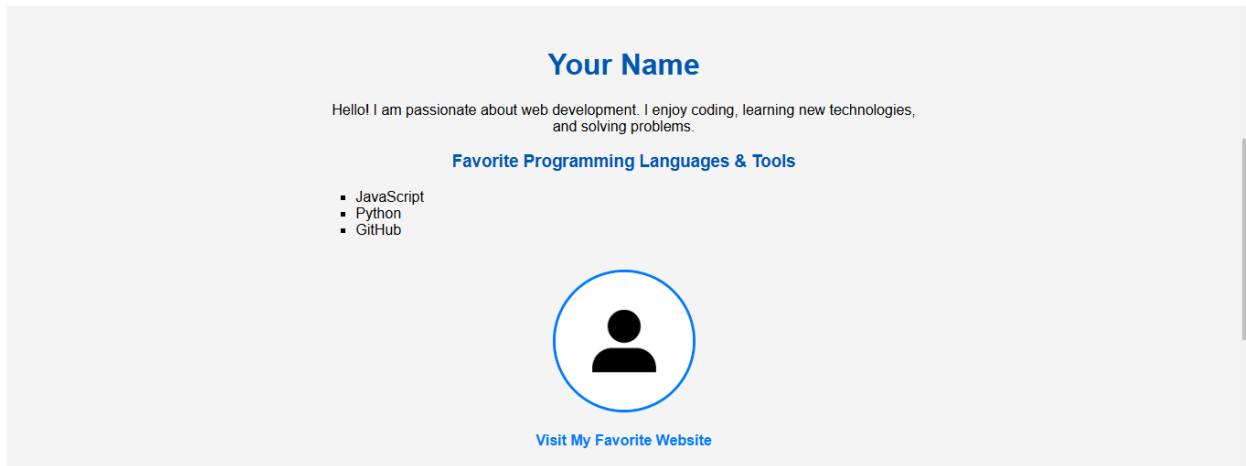
Button Styling

Buttons need to be visually appealing and provide feedback when interacted with. Use `background-color`, `border-radius`, and `hover` effects to enhance the user experience.

CSS Suggestions:

- `background-color` controls the button's appearance.
- Add a `hover` effect to change the button's color when hovered.

Task 2: Styling the Profile Section



For the profile image, `border-radius` and `border` are applied to create a circular avatar. The profile description is aligned using standard text styling.

CSS Suggestions:

- `border-radius: 50%` for circular images.
- Apply `border` to add an outline around images.

Task 3: Styling Course Registration Page

Course Registration

Student Name:

Gender:

☐ Male ☐ Female ☐ Other

Courses Offered:

☐ Web Development ☐ AI ☐ Data Science

Available Courses

Course Code	Course Name	Credits
CS101	Web Engineering	3
CS102	Data Science	4
CS103	Artificial Intelligence	4
CS104	Mobile App Development	3

Styling Radio Buttons and Checkboxes

By default, radio buttons and checkboxes stack vertically. To align them horizontally, we use `display: inline-block`.

CSS Suggestions:

- `display: inline-block`; allows the radio buttons to be aligned horizontally.
- Adjust the size of radio buttons and checkboxes using `width` and `height`.

```
input[type="radio"], input[type="checkbox"] {  
  
    margin-right: 5px;  
  
    display: inline-block;  
  
    margin-top: 2px;  
  
    width: 12px !important; /* Adjust size */  
  
    height: 12px !important;  
  
}
```

Using Flexbox for Layout

Flexbox is a powerful CSS tool for creating flexible layouts. You can align items both horizontally and vertically using `justify-content` and `align-items`.

CSS Suggestions:

- `display: flex`; activates Flexbox layout.
- Use `justify-content: center`; to horizontally center the items.

Table Styling

Tables are styled with `border-collapse` to make the borders look cleaner, and `overflow-x: auto` is used to make the table scrollable on small screens.

CSS Suggestions:

- `border-collapse: collapse;` merges borders between table cells.
- Use `overflow-x: auto` for responsiveness on smaller screens.

Additional CSS Tips

- **Flexbox:** Use Flexbox for aligning items either horizontally or vertically. You can center a form or group of items easily with Flexbox.
- **Responsive Design:** Use `width: 100%` for elements like inputs and tables to ensure they resize automatically on smaller screens.
- **Use media queries to create Mobile responsive designs.**