Title

CSS Basics

Objectives
• To learn and apply Flexbox and Grid for creating responsive layouts.
• To use CSS positioning and media queries for responsive design.
• To understand the CSS box model (margin, padding, border, box-sizing).
• To design responsive layouts using the Bootstrap grid system.

Theory – Not Required

Procedure

1. Explain the different ways of adding CSS to an HTML document.

2. Write HTML + CSS to create a horizontal navbar using Flexbox where:

- Logo is on the left
- Menu items are centered
- Login button is on the right

3. Create a card layout using Flexbox that:

- Shows 4 cards in a row on desktop
- 2 cards per row on tablet
- 1 card per row on mobile

4. Write HTML + CSS using CSS Grid to create a 3×3 gallery layout.

5. Create a grid layout with:

- Header spanning full width
- Sidebar on the left
- Content area in the center
- Footer at the bottom

6. Write a grid program where:

- Items auto-fit based on screen size
- Minimum width of each item is 250px

7. Write CSS to place a notification badge at the top-right corner of an icon using `position`.

8. Create a webpage where:

- Header is fixed at the top

- Sidebar is sticky
- Content scrolls normally

9. Write CSS to center an element using absolute positioning.

10. Write CSS media queries for:

- Mobile (≤576px)
- Tablet (≤768px)
- Desktop (≥992px)

Create a responsive layout where: Background color changes on different screen sizes.

11. Write a media query to hide a sidebar on mobile devices.

12. Convert a 3-column layout into:

- 2 columns on tablet
- 1 column on mobile

Use media query.

13. Write a program demonstrating difference between `padding` and `margin`

14. Create a box with:

- Border of 2px solid black
- Padding of 20px
- Margin of 30px

15. Show using code how `box-sizing: border-box` changes element size.

16. Create a responsive column layout using Bootstrap where:

- Desktop → 4 columns
- Tablet → 2 columns
- Mobile → 1 column

Also use container.

Output:

Printed Screenshots after each program code or at last. Also display your name in each program output to verify its original and not friends' photocopy.

Discussion:

What you did? What problems you faced and how you solved it?
What are use cases and limitations of CSS?

Conclusion:
Conclude what you learned.

[Note: Lab report should be handwritten. You cannot use text editor with autocomplete features in board exam. So, it will be practice.]
[Note: Report won't be checked after one week of deadline]