🎨 CSS Notes

1. What is CSS?

- CSS = **Cascading Style Sheets**
- It is used to **style** HTML elements (colors, fonts, layout, etc.)

2. Ways to Use CSS

Type	Syntax Example
Inline	`<p style="color: red;">Text</p>`
Internal	Inside `<style>` in `<head>`
External	Linked with `<link rel="stylesheet" href="style.css">`

3. CSS Syntax

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

Example:

```
p {
  color: blue;
  font-size: 16px;
}
```

4. Selectors

Selector	Use
*	Universal selector
element	Targets all elements of that type
#id	Targets element with specific id
.class	Targets all elements with that class
element1, element2	Group multiple elements
element element	Descendant (nested inside)
element > element	Direct child
element + element	Immediately after sibling

5. Colors in CSS

color: red;

color: #ff0000;

color: rgb(255, 0, 0);

color: rgba(255, 0, 0, 0.5);

6. Text Styling

color: black;

text-align: center;

text-decoration: underline / none / line-through;

text-transform: uppercase / lowercase;

font-family: Arial, sans-serif;

font-size: 20px;

font-weight: bold;

letter-spacing: 2px;

line-height: 1.5;

7. Box Model

- Each element is a box made of:
    - content
    - padding
    - border
    - margin

padding: 10px;

margin: 20px;

border: 2px solid black;

8. Backgrounds

background-color: lightblue;

background-image: url("image.jpg");

background-repeat: no-repeat;

background-size: cover;

background-position: center;

9. Borders

border: 2px solid red;

border-radius: 10px; /* Rounded corners */

10. Width & Height

width: 200px;

height: 100px;

max-width: 100%;

11. Display Property

Value	Description
block	Takes full width
inline	Takes only needed width
inline-block	Like inline + allows box model
none	Hides the element
flex	Makes element a flex container

12. Positioning

Value	Description
static	Default
relative	Position relative to normal
absolute	Relative to nearest positioned parent
fixed	Fixed to screen
sticky	Sticks when scrolling

position: absolute;

top: 10px;

left: 20px;

13. Flexbox (1D Layout)

display: flex;

flex-direction: row / column;

justify-content: center / space-between / space-around;

align-items: center / flex-start / flex-end;

gap: 10px;

14. Grid (2D Layout)

display: grid;

grid-template-columns: repeat(3, 1fr);

grid-template-rows: auto;

gap: 20px;

15. Pseudo Classes

a:hover { color: red; }

input:focus { background-color: yellow; }

li:first-child { font-weight: bold; }

16. Pseudo Elements

p::first-letter { font-size: 30px; }

p::before { content: "★ "; color: gold; }

17. Transitions

transition: all 0.3s ease-in-out;

18. Transformations

transform: scale(1.2);

transform: rotate(45deg);

19. Animations

```css
@keyframes slide {
  from { left: 0; }
  to { left: 100px; }
}

.box {
  animation: slide 2s infinite;
}
```

20. Media Queries (Responsive Design)

```css
@media (max-width: 600px) {
  body {
    background-color: lightgray;
  }
}
```

21. Z-index (Layering)

```css
position: absolute;
z-index: 2;
```

22. Overflow

```css
overflow: auto / scroll / hidden / visible;
```

23. Opacity & Visibility

```css
opacity: 0.5;      /* 0 (transparent) to 1 (visible) */
visibility: hidden;   /* hides but keeps space */
```

24. Cursor Styles

cursor: pointer;

cursor: not-allowed;

25. Important Keyword

color: red !important; /* Overrides all other styles */

INTERVIEW QUESTIONS

🧠 **Basics**

1. **What is CSS?**
   → Cascading Style Sheets, used for styling HTML.

2. **Types of CSS?**
   → Inline, Internal, External.

3. **How to link external CSS?**
   → <link rel="stylesheet" href="style.css">

4. **What is a selector in CSS?**
   → It selects the HTML element to style.

5. **What is the syntax of CSS?**
   → selector { property: value; }

6. **What is the difference between class and ID?**
   → .class is reusable, #id is unique.

7. **How to write a comment in CSS?**
   → /* This is a comment */

8. **What is the Box Model?**
   → Content → Padding → Border → Margin.

9. **What is the default position in CSS?**
   → static

10. **How to apply multiple classes?**
    → <div class="box red"></div>

🎨 Colors & Fonts

11. **How to change text color?**
    → color: red;

12. **How to set background color?**
    → background-color: blue;

13. **How to use a Google Font?**
    → Link it in HTML and use font-family.

14. **Font units commonly used?**
    → px, em, rem, %

15. **How to bold text?**
    → font-weight: bold;

16. **How to italicize text?**
    → font-style: italic;

17. **Text alignment?**
    → text-align: left/center/right;

18. **Transform text to uppercase?**
    → text-transform: uppercase;

19. **Underline text?**
    → text-decoration: underline;

20. **Change font size?**
    → font-size: 20px;

📦 Box Model & Layout

21. **Padding vs Margin?**
    → Padding is inside border; margin is outside.

22. **How to center a block element?**
    → margin: auto;

23. **Make a div scrollable?**
    → overflow: auto;

24. **Hide an element?**
    → display: none; or visibility: hidden;

25. **Rounded corners?**
    → border-radius: 10px;

26. **Set border?**
    → border: 1px solid black;

27. **Set element size?**
    → width and height

28. **What is z-index?**
    → Controls stack order (layering).

29. **What is box-sizing?**
    → Controls if padding/border are included in width.

30. **Remove default margin/padding?**
    → margin: 0; padding: 0;

📐 Position & Display

31. **Different CSS positions?**
    → static, relative, absolute, fixed, sticky

32. **What is relative positioning?**
    → Moves element relative to itself.

33. **What is absolute positioning?**
    → Positions element relative to nearest positioned parent.

34. **Use of fixed?**
    → Element stays fixed when scrolling.

35. **Use of sticky?**
    → Sticks in place when scrolling past it.

36. **Difference: block vs inline?**
    → Block: full width; Inline: only content width.

37. **What is inline-block?**
   → Like inline, but respects box model.

38. **What is display: none?**
   → Hides element completely.

39. **What is visibility: hidden?**
   → Hides element but keeps space.

40. **What is float?**
   → Places element to left/right of container.

📊 **Flexbox, Grid, & Responsive**

41. **What is Flexbox?**
   → One-dimensional layout system.

42. **Enable Flexbox?**
   → display: flex;

43. **Align items in center?**
   → justify-content: center; align-items: center;

44. **Direction of Flexbox?**
   → flex-direction: row/column;

45. **What is Grid?**
   → Two-dimensional layout system.

46. **Enable Grid?**
   → display: grid;

47. **Set grid columns?**
   → grid-template-columns: repeat(3, 1fr);

48. **Make site responsive?**
   → Use @media queries.

49. **What is a media query?**
   → CSS rule for specific screen sizes.

50. **Difference: Flexbox vs Grid?**
   → Flexbox = 1D (row or column); Grid = 2D (rows + columns)