# CSS (Cascading Style Sheets)

# Basics

### Q1. What is CSS?

CSS stands for Cascading Style Sheets and is used to style and layout web pages.

### Q2. What are the types of CSS?

- Inline CSS
- Internal CSS
- External CSS

#### Q3. Difference between ID and Class selectors?

- ID: Unique, #id
- Class: Reusable, .class

#### Q4. What is the Box Model?

Includes content, padding, border, and margin.

### Q5. What is specificity in CSS?

Determines which rule applies when multiple rules match.

### Q6. What are em, rem, px, %, vw, vh units?

- px: fixed
- em: relative to parent
- rem: relative to root
- %: relative
- vw/vh: relative to viewport

### Q7. What is the difference between inherit, initial, unset, revert?

- inherit: inherit from parent
- initial: default CSS spec
- unset: inherits or resets
- revert: resets to browser default

### Q8. What is z-index in CSS?

Controls stack order of elements.



### **Q9. Types of Selectors**

• Universal, Element, ID, Class, Group, Child, Sibling, Attribute

### Q10. What is a pseudo-class?

Represents the state of an element, e.g. :hover, :focus

### Q11. What is a pseudo-element?

Selects a part of the element, e.g. ::before, ::after

### Q12. Difference: :nth-child() vs :nth-of-type()

- nth-child(): every nth element
- nth-of-type(): nth of the same tag

### Q13. What is the :not() selector?

Targets all except the specified one.

### Q14. New CSS4 Selectors

• :is(), :where(), :has()

### Flexbox & Grid

### Q15. What is Flexbox?

Used for one-dimensional layout.

### **Q16.** Important Flexbox Properties

• justify-content, align-items, flex-wrap, flex

### Q17. What is CSS Grid?

Used for two-dimensional layouts.

### **Q18.** Grid Properties

• grid-template-columns, grid-template-rows, gap, place-items

### Q19. auto-fill vs auto-fit in Grid

- auto-fill: preserves empty tracks
- auto-fit: collapses tracks

Q20. What is minmax()? Used for responsive sizing in grid.

# ✓ Transitions & Animations

#### Q21. What is a transition?

Smooth change of properties.

### **Q22. Transition Timing Functions**

• ease, linear, ease-in, ease-out

### Q23. What are keyframes? Defines animation steps using @keyframes

#### **Q24.** Transition vs Animation

- Transition: on trigger
- Animation: auto or on loop

Q25. What is transform? Used for 2D/3D effects: scale, rotate, translate

# Responsive Design

#### Q26. What are media queries?

Conditionally apply styles by screen size or feature.

#### Q27. min-width vs max-width

- min-width: mobile-first
- max-width: desktop-first
- **Q28.** What is aspect-ratio? Maintains proportional dimensions.

### Q29. What are relative units for responsiveness?

- %, vw, vh, em, rem
- **Q30.** What are container queries? Apply styles based on parent size.

### Advanced CSS Features

### Q31. What are CSS Variables?

```
:root { --main: red; }
color: var(--main);
```

- Q32. What is clamp()? Sets a value between min, ideal, and max.
- Q33. What is calc()? Allows dynamic calculations in values.
- Q34. What is @supports? Feature detection block.
- Q35. What is @layer? Controls order of CSS rules.
- Q36. What is Shadow DOM styling? Use ::part, ::slotted to style custom elements.

### Q37. What are system-level color schemes?

:root { color-scheme: light dark; }

**Q38.** What is the contain property? Limits rendering scope for performance.

### Q39. Logical vs Physical properties

- Logical: margin-inline, padding-block
- Physical: margin-left, padding-top

### Q40. Shorthand vs Longhand

• Shorthand: margin: 10px 20px

• Longhand: margin-top: 10px

## **✓** Comparison & Logic Questions

### Q41. visibility: hidden vs display: none

• Hidden: takes space

• None: removed from flow

### Q42. opacity: 0 vs visibility: hidden

• Opacity: still clickable

• Hidden: not clickable

### Q43. position: relative vs absolute

Relative: from original position

• Absolute: from nearest positioned ancestor

### Q44. inline, block, inline-block

• Inline: no width/height

• Block: full-width

• Inline-block: respects size

### Q45. em vs rem

• em: relative to parent

• rem: relative to root

### Q46. inherit vs initial vs unset vs revert

• Inherit: takes from parent

• Initial: CSS default

• Unset: hybrid

• Revert: browser default

### Q47. Performance tips in CSS

Avoid deep nesting

Combine selectors

Use shorthand

Minify files

### Q48. How to debug CSS issues?

- Use DevTools
- Toggle rules
- Use outline for layout

### Q49. How to write scalable CSS?

- BEM naming
- Use variables
- Avoid !important

# **Q50.** How does the cascade work in CSS? Order of rule application:

- Importance (!important)
- Specificity
- Source order

Property	Use
color	Sets the text color of an element.
background-color	Sets the background color of an element.
background-image	Applies an image as the background.
font-size	Sets the size of the font.
font-family	Sets the typeface for text.
font-weight	Defines the thickness of the text (e.g., bold, normal, 500).
text-align	Aligns text (left, center, right, justify).
text-decoration	Adds decoration (e.g., underline, line-through, none).
line-height	Sets the height between lines of text.
letter-spacing	Sets spacing between letters.
word-spacing	Sets spacing between words.
width / height	Defines the size of elements.
max-width / min-width	Restricts the width range.
margin	Adds space around elements (outside border).
padding	Adds space inside the element (between content and border).
border	Defines the border (style, width, color).
display	Defines display type (block, inline, flex, grid, none).
position	Sets positioning (static, relative, absolute, fixed, sticky).

top, right, bottom, left	Positions element using position property.
z-index	Sets stack order of elements.
overflow	Controls content overflow (hidden, scroll, auto).
visibility	Shows or hides the element (visible, hidden).
opacity	Sets transparency (0 to 1).
box-shadow	Adds shadow to elements.
text-shadow	Adds shadow to text.
transition	Creates smooth animations between state changes.
transform	Applies 2D/3D transformation (rotate, scale, translate, skew).
animation	Animates elements using keyframes.
flex, justify-content, align-items	Controls layout in Flexbox.
grid-template-columns, grid-gap	Controls layout in Grid.
cursor	Sets the mouse cursor style (e.g., pointer, default, move).
visibility	Controls visibility while maintaining layout.
box-sizing	Controls box model (content-box, border-box).
float	Aligns elements left or right.
clear	Clears floated elements.
content	Inserts content in pseudo-elements (::before, ::after).
clip-path	Clips an element using shapes.
filter	Applies effects like blur, brightness, grayscale.