CSS ROADMAP (BASIC TO ADVANCED)

🎯 Ultimate CSS Roadmap (Beginner to Advanced)

💡 *Includes All Topics + Questions You Should Be Able to Answer*

□ Beginner Level – Foundations of CSS

📓 Topics:

1. What is CSS? Why do we use it?
2. CSS Syntax (selectors, properties, values)
3. Types of CSS: Inline, Internal, External
4. Colors and Units
    - Hex, RGB, HSL, px, em, rem, %
5. Text and Font Styling
    - font-family, font-size, text-align, line-height
6. Box Model
    - margin, border, padding, content
7. Backgrounds
    - background-color, background-image, repeat, position

8. Borders & Outlines
9. Display Property
    - block, inline, inline-block, none
10. Basic Positioning
    - static, relative, absolute, fixed
11. Overflow, Visibility, Z-index

❓ Questions:

- What are the types of CSS and which one has the highest priority?
- How does the CSS box model work?
- What's the difference between em and rem?
- How to center a div using margin or padding?
- Difference between class and id selectors?

□ Intermediate Level – Layouts, Styling & Responsiveness

📓 Topics:

1. Advanced Selectors
    - Attribute selectors, Pseudo-classes (:hover, :nth-child()), Pseudo-elements (::before, ::after)
2. Flexbox
    - flex-direction, justify-content, align-items, flex-wrap, gap

3. CSS Grid
    - grid-template-rows, grid-template-columns, place-items, grid-gap
4. Responsive Design
    - Media Queries, Mobile-first design
5. Dimensions and Units (Advanced)
    - vw, vh, %, calc()
6. Transitions & Animations
    - transition, animation, @keyframes
7. Positioning (Advanced)
    - sticky, z-index stacking
8. Custom Properties (CSS Variables)
9. Combinators
    - Descendant , Child >, Adjacent +, Sibling ~

❓ Questions:

- How is Flexbox different from Grid?
- What are pseudo-classes vs pseudo-elements?
- How do you make a responsive layout with media queries?
- Explain how transitions and animations differ.
- What does position: sticky do?

🔴 Advanced Level – Structure, Tools & Performance

📓 Topics:

1. CSS Architecture
    - BEM (Block Element Modifier), OOCSS, SMACSS
2. Preprocessors
    - SASS/SCSS basics: variables, nesting, mixins
3. CSS Functions
    - calc(), clamp(), min(), max()
4. Advanced Animations
    - Animation chaining, timing functions, delays
5. Responsive Techniques
    - Fluid typography, Responsive grids, Container queries (new)
6. Theming & Dark Mode
    - prefers-color-scheme, CSS variables for themes
7. Accessibility (A11y) with CSS
    - Focus management, outline control
8. Compositing & Effects
    - mix-blend-mode, isolation, backdrop-filter
9. Performance Optimization
    - Critical CSS, avoiding reflows/repaints
10. Modern CSS Tools

- Tailwind CSS, PostCSS, PurgeCSS

11. Browser Compatibility
    - Feature queries @supports, fallbacks
12. Print Styling
    - @media print, hiding elements for print

❓ Questions:

- What is BEM and why is it used in large projects?
- What's the difference between SASS and CSS?
- How does clamp() work for responsive text?
- How to apply dark mode using only CSS?
- How to optimize CSS for performance and reduce load time?

□ Optional Mastery Topics (Extra Edge)

- CSS Logical Properties: margin-inline, padding-block
- Writing Modes (RTL, vertical)
- CSS Houdini (paint worklets, layout API – very advanced)
- Custom Scrollbars (::-webkit-scrollbar)
- Using CSS with JavaScript (toggle themes, class manipulation)

❒ Bonus: Practice Ideas

- Build a personal portfolio using Flexbox and Grid.
- Create a responsive card UI using media queries.
- Implement a dark/light mode toggle with CSS variables.
- Animate a button or loader using keyframes.
- Style a form with focus/hover states and validation cues.
