

Responsive Design – CSS Sizing Notes

Responsive design focuses on building layouts that adapt to different screen sizes using flexible units, layouts, and constraints.

Core Concept

- Responsive design avoids fixed pixel-based layouts.
- Relative units allow layouts to adapt naturally.

CSS Units for Responsiveness

- rem – relative to root font size; best for typography and spacing.
- % – relative to parent element; best for widths.
- vw / vh – relative to viewport; useful for hero sections.
- px – fixed unit; avoid for responsive layouts.

Typography Strategy

- Define base font size on the html element.
- Use rem units for headings and text.
- Improves accessibility and consistency.

Layout Width Control

- Use percentage widths for fluid layouts.
- Use max-width to limit excessive stretching.
- Common pattern: width: 90% and max-width: 1200px.

Responsive Images

- Use max-width: 100% on images.
- Set height to auto to preserve aspect ratio.

Spacing Guidelines

- Use rem units for padding and margin.
- Spacing scales proportionally with text.

Flexbox and Grid Usage

- Flexbox is ideal for one-dimensional layouts.
- Grid is ideal for two-dimensional layouts.
- Avoid fixed positioning for main layouts.

Media Queries

- Use media queries only when necessary.
- Follow mobile-first approach with min-width.
- Adjust layout progressively.

Advanced Responsive Technique

- Use clamp() for fluid typography.
- Example: font-size: clamp(1.8rem, 5vw, 3rem).

Golden Rules

- Fonts → rem.
- Layout widths → % with max-width.
- Images → max-width: 100%.
- Heights → min-height instead of fixed height.
- Layouts → Flexbox or Grid.