

# Tailwind CSS Roadmap with Subtopics

## 1. Basics (Getting Started)

- What is Tailwind CSS? Utility-first approach
- Pros & cons vs traditional CSS / Bootstrap
- Installation methods: CDN link, NPM + PostCSS, Framework integration
- Understanding `tailwind.config.js`
- Build process & purge (removing unused CSS)
- Project setup & folder structure

## 2. Core Concepts

- Typography utilities: font size, weight, style, letter spacing, text alignment, decoration, transform
- Spacing: margin, padding, gap in flex/grid layouts
- Colors: text, background, border, opacity & gradients
- Sizing: width, height, min/max
- Borders: radius, style, width
- Layout: flexbox & grid utilities
- Positioning: relative, absolute, fixed, sticky, z-index
- Effects: shadows, opacity, cursor, object fit & position
- Background utilities: solid colors, gradients, images

## 3. Responsive Design

- Breakpoints (sm, md, lg, xl, 2xl)
- Mobile-first approach
- Applying utilities conditionally
- Responsive typography & layouts
- Hiding/showing elements at breakpoints

## 4. Advanced Styling

- Pseudo-classes: hover, focus, active, disabled, checked
- Pseudo-elements: `before`, `after`
- Dark mode (dark: variant, class vs media)
- State variants: group-hover, peer, peer-focus
- Animations & transitions: duration, delay, keyframes
- Transform utilities: scale, rotate, translate, skew

## 5. Customization

- Extending theme in `tailwind.config.js`: colors, fonts, spacing
- Extracting components using `@apply`
- Adding plugins: forms, typography, aspect-ratio, line-clamp
- Custom container settings
- Adding brand-specific themes

## 6. Integration with Frameworks

- Tailwind with React (className, reusable components)
- Tailwind with Next.js (SSR, optimization)
- Tailwind with Vue & Angular
- Tailwind with Vite
- Using UI libraries: DaisyUI, Flowbite, Headless UI, shadcn/ui

## 7. Production & Best Practices

- Purge & optimization
- Theming (light/dark, brand colors)
- Accessibility best practices
- Reusability (extracting components)
- Organizing config for large projects
- Performance tips

## 8. Pro-Level Topics

- Tailwind JIT (Just-In-Time compiler, dynamic class names)
- Building your own Tailwind plugin
- Tailwind + CSS-in-JS (twin.macro)
- Creating a design system with Tailwind
- Migrating projects from CSS/Bootstrap
- Enterprise-scale usage