

# CSS Cheatsheet

## Essential properties and techniques for web styling and layout

This cheatsheet provides a quick reference to fundamental CSS operations, syntax, and modern features, ideal for both beginners and experienced web developers for efficient web styling and layout design.

<b>CSS Basics</b> Syntax, selectors, and properties	<b>Styling Text</b> Typography and text formatting	<b>Box Model</b> Margins, padding, and borders
<b>Layout &amp; Positioning</b> Flexbox, Grid, and positioning		<b>Responsive Design</b> Media queries and mobile-first

## CSS Syntax & Selectors

### Basic Syntax

CSS consists of selectors and declarations. The selector targets HTML elements, and declarations set property values.

```
/* Basic syntax */
selector {
  property: value;
  property: value;
}

/* Example */
p {
  color: red;
  font-size: 16px;
}
```

### Element Selectors

Target HTML elements by their tag name.

```
/* Select all paragraphs */
p { color: blue; }

/* Select all headings */
h1 { font-size: 2em; }

/* Select all links */
a { text-decoration: none; }
```

### Class Selectors

Target elements with specific class attributes.

```
/* Select elements with class="highlight" */
.highlight { background-color: yellow; }

/* Select paragraphs with class="intro" */
p.intro { font-weight: bold; }

/* Multiple classes */
large.bold { font-size: 20px; font-weight: bold; }
```

## Box Model & Layout

Understanding the CSS Box Model is key to creating layouts and aligning items with other elements.

<b>01</b> <b>Content: `width` / `height`</b> The actual content area of the element.	<b>02</b> <b>Padding: `padding`</b> Space between content and border, inside the element.	<b>03</b> <b>Border: `border`</b> Borders provide a frame for elements with customizable size, style and color.
--	---	---

```
/* Set dimensions */
div { width: 300px; height: 200px; }

/* Responsive sizing */
.container { width: 100%; max-width: 1200px; }

/* Min/max constraints */
.box { min-height: 100px; max-width: 500px; }
```

```
/* All sides */
div { padding: 20px; }

/* Individual sides */
div { padding-top: 10px; padding-right: 15px; }

/* Shorthand: top right bottom left */
div { padding: 10px 15px 20px 5px; }
```

```
/* Border shorthand */
div { border: 2px solid #333; }

/* Individual properties */
div {
  border-width: 1px;
  border-style: dashed;
  border-color: red;
}
```

```
/* Individual sides */
div { border-bottom: 3px solid blue; }
```

<b>04</b> <b>Margin: `margin`</b> Space outside the border, between elements.
---

```
/* All sides */
div { margin: 20px; }

/* Center horizontally */
div { margin: 0 auto; }

/* Individual sides */
div { margin-top: 30px; margin-bottom: 10px; }

/* Negative margins */
div { margin-left: -20px; }
```

## Text & Typography

### Font Properties

Control font family, size, weight, and style.

```
/* Font family */
body { font-family: Arial, sans-serif; }

/* Google Fonts */
@import url('https://fonts.googleapis.com/css?family=Roboto:wght@300;400;700&display=swap');
body { font-family: 'Roboto', sans-serif; }

/* Font size */
h1 { font-size: 2rem; }
p { font-size: 16px; }

/* Font weight */
h2 { font-weight: bold; }
span { font-weight: 300; }
```

### Text Alignment

Control text positioning and spacing.

```
/* Horizontal alignment */
h1 { text-align: center; }
p { text-align: justify; }

/* Line height */
p { line-height: 1.6; }

/* Letter and word spacing */
h1 { letter-spacing: 2px; }
p { word-spacing: 4px; }
```

## Flexbox Layout

Flexbox is a layout mechanism designed for laying out groups of items in one dimension.

### Flex Container Properties

Properties applied to the parent container.

```
/* Enable flexbox */
.container {
  display: flex;
}

/* Flex direction */
.container {
  flex-direction: row; /* default */
  flex-direction: column;
  flex-direction: row-reverse;
}

/* Justify content (main axis) */
.container {
  justify-content: flex-start; /* default */
  justify-content: center;
  justify-content: space-around;
  justify-content: space-between;
}

/* Align items (cross axis) */
.container {
  align-items: stretch; /* default */
  align-items: center;
  align-items: flex-start;
}

/* Flex grow/shrink */
.item {
  flex-grow: 1; /* grow to fill space */
  flex-shrink: 1; /* shrink if needed */
  flex-basis: auto; /* initial size */
}

/* Shorthand */
.item {
  flex: 1; /* flex: 1 1 0% */
  flex: 0 0 200px; /* fixed width */
}

/* Individual alignment */
.item {
  align-self: center;
  align-self: flex-end;
}

/* Order */
.item {
  order: 2; /* change visual order */
}
```

### Flex Item Properties

Properties applied to child elements.

```
/* Flex grow/shrink */
.item {
  flex-grow: 1; /* grow to fill space */
  flex-shrink: 1; /* shrink if needed */
  flex-basis: auto; /* initial size */
}

/* Shorthand */
.item {
  flex: 1; /* flex: 1 1 0% */
  flex: 0 0 200px; /* fixed width */
}

/* Individual alignment */
.item {
  align-self: center;
  align-self: flex-end;
}

/* Order */
.item {
  order: 2; /* change visual order */
}
```

### Grid Container

Define grid structure and properties.

```
/* Enable grid */
.grid-container {
  display: grid;
}

/* Define columns and rows */
.grid-container {
  grid-template-columns: 1fr 2fr 1fr;
  grid-template-rows: 100px auto 50px;
}

/* Grid gaps */
.grid-container {
  gap: 20px;
  column-gap: 30px;
  row-gap: 10px;
}

/* Named grid areas */
.grid-container {
  grid-template-areas:
    "header header header"
    "sidebar main main"
    "footer footer footer";
}
```

CSS Grid Layout provides a two dimensional layout system, controlling layout in rows and columns.

### Grid Items

Position and size grid items.

```
/* Grid positioning */
.grid-item {
  grid-column: 1 / 3; /* span columns 1-2 */
  grid-row: 2 / 4; /* span rows 2-3 */
}

/* Shorthand */
.grid-item {
  grid-area: 2 / 1 / 4 / 3; /* row-start / col-start / row-end / col-end */
}

/* Named areas */
.header { grid-area: header; }
.sidebar { grid-area: sidebar; }
.main { grid-area: main; }

/* Auto placement */
.grid-item {
  grid-column: span 2; /* span 2 columns */
  grid-row: span 3; /* span 3 rows */
}
```

## Positioning

### Position Property

Control element positioning behavior.

```
/* Static (default) */
.element { position: static; }

/* Relative positioning */
.element {
  position: relative;
  top: 20px;
  left: 10px;
}

/* Absolute positioning */
.element {
  position: absolute;
  top: 0;
  right: 0;
}

/* Fixed positioning */
.navbar {
  position: fixed;
  top: 0;
  width: 100%;
}

/* Sticky positioning */
.sidebar {
  position: sticky;
  top: 20px;
}
```

## Responsive Design

Create responsive designs that adapt seamlessly to different screen sizes and devices, which is critical in today's mobile-centric world.

### Media Queries

Apply styles based on device characteristics.

```
/* Mobile first approach */
.container { width: 100%; }

/* Tablet styles */
@media (min-width: 768px) {
  .container { width: 750px; }
}

/* Desktop styles */
@media (min-width: 1024px) {
  .container { width: 960px; }
}

/* Print styles */
@media print {
  body { font-size: 12pt; }
  .no-print { display: none; }
}

/* Orientation */
@media (orientation: landscape) {
  .sidebar { display: block; }
}
```

### Media Queries

Apply styles based on device characteristics.

```
/* Mobile first approach */
.container { width: 100%; }

/* Tablet styles */
@media (min-width: 768px) {
  .container { width: 750px; }
}

/* Desktop styles */
@media (min-width: 1024px) {
  .container { width: 960px; }
}

/* Print styles */
@media print {
  body { font-size: 12pt; }
  .no-print { display: none; }
}

/* Orientation */
@media (orientation: landscape) {
  .sidebar { display: block; }
}
```

CSS custom properties (variables) help organize and reuse values, making styles more flexible and easier to understand.

### CSS Variables

Define and use custom properties for consistent theming.

```
/* Define variables */
:root {
  --primary-color: #3498db;
  --secondary-color: #2ecc71;
  --font-size-large: 24px;
  --border-radius: 8px;
}

/* Use variables */
button {
  background-color: var(--primary-color);
  border-radius: var(--border-radius);
}

/* Fallback values */
.text {
  color: var(--text-color, #333);
}

/* Local variables */
.card {
  --card-padding: 20px;
  padding: var(--card-padding);
}
```

CSS has a range of built-in functions for calculations and dynamic values.

```
/* Calc function */
.container {
  width: calc(100% - 40px);
  height: calc(100vh - 60px);
}

/* Min/Max functions */
.responsive {
  width: min(90%, 1200px);
  font-size: max(16px, 1.2vw);
  height: clamp(200px, 50vh, 400px);
}

/* Color functions */
.element {
  background-color: hsl(200, 50%, 50%);
  color: rgb(255, 87, 51);
}

/* Transform functions */
.rotate { transform: rotate(45deg); }
.scale { transform: scale(1.5); }
.translate { transform: translate(20px, 30px); }
```

## Best Practices & Organization

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization

Structure your CSS for maintainability.

```
/* Use meaningful class names */
.hero-section {}
.primary-button {}
.navigation-menu {}

/* BEM methodology */
.block {}
.block__element {}

/* Example */
.card_header {}
.card_body {}
.card_featured {}

/* Group related styles */
/* ===== LAYOUT ===== */
.container {}
.grid {}

/* ===== COMPONENTS ===== */
.button {}
.card {}
```

### CSS Organization