

100 Practical CSS Code Snippets



1) Modern CSS Reset (minimal)	4
2) Fluid Container	5
3) Responsive Typography (clamp)	5
4) Better Line Length	5
5) Center Anything (Grid)	5
6) Sticky Header	5
7) Smooth Scrolling	6
8) Scroll Margin for Anchors (fix sticky header overlap)	6
9) Nice Focus Ring (accessible)	6
10) Hide Visually, Keep for Screen Readers	7
Layout & Positioning	7
11) Flex Row with Gap	7
12) Wrap Items Responsively	7
13) Equal Columns (Flex)	7
14) Auto-fit Grid Cards	8
15) Sidebar Layout (Grid)	8
16) Full-Height App Shell	8
17) Truncate Single Line	8

Get more Resources from Laurence Svekis <https://basescripts.com/>

18) Clamp to 3 Lines	9
19) Absolute Center (classic)	9
20) Safe Area Padding (iOS notch)	9
Spacing, Sizing, and Units	10
21) Consistent Section Spacing	10
22) Fluid Spacing Scale	10
23) Logical Properties (better for RTL)	10
24) Square Avatar	10
25) Responsive Images Crop	10
Color, Backgrounds, and Effects	11
26) Subtle Gradient Background	11
27) Background Pattern (CSS-only dots)	11
28) Glassmorphism Card (simple)	11
29) Soft Shadow	12
30) Hover Lift	12
Borders & Shapes	12
31) Nice Rounded Corners	12
32) Gradient Border	12
33) Dashed Divider	13
34) Fancy Corner Ribbon (badge)	13
Typography	13
35) Font Smoothing + Base Type	13
36) Better Headline Spacing	13
37) Balanced Text Wrap	14
38) Numerals Align Nicely	14
Buttons & UI Components	14
39) Solid Button	14
40) Outline Button	15
41) Disabled Style	15
42) Loading Spinner (CSS-only)	15
43) Pill Badge	15
44) Card Component	16
45) Modal Backdrop	16
Forms	16
46) Clean Input	16
47) Input with Icon Layout	17
48) Checkbox Accent Color	17
49) Validation States	17
50) Form Row Spacing	17
Images & Media	18
51) Cover Image in Card	18

52) Prevent Layout Shift (reserve space)	18
Lists, Tables, and Content	18
53) Nice List Spacing	18
54) Zebra Table	18
55) Sticky Table Header	19
Animation & Motion	19
56) Fade In	19
57) Slide Up	19
58) Respect Reduced Motion	20
59) Pulse Attention (subtle)	20
60) Skeleton Loading	20
Responsive Design Helpers	20
61) Mobile-first Breakpoint	20
62) Desktop-only	21
63) Responsive Padding	21
64) Container Queries (modern)	21
Interactions, Hover, and State	21
65) Hover Underline Animation (links)	21
66) Clickable Card	22
67) Disabled Pointer Events	22
68) Prevent Text Selection	22
69) Scrollbar Styling (WebKit)	22
Advanced UI Patterns	23
70) Toast Position	23
71) Tooltip (CSS-only)	23
72) Dropdown Panel	24
73) Aspect Ratio Utility	24
74) Blended Overlay on Image	24
75) Frosted Overlay for Text	25
CSS Variables & Theming	25
76) Theme Tokens	25
77) Light/Dark Theme Switch	25
78) Automatic Dark Mode	26
Performance & Safety	26
79) Content Visibility (perf boost)	26
80) Prevent Image Drag (UX)	26
More Components & Patterns	26
81) Breadcrumbs	26
82) Tabs (base styles)	27
83) Progress Bar	27
84) Notification Banner	27

85) “Chip” Input Tags	27
Tricks & Utilities	28
86) Prevent Overscroll Bounce (mobile)	28
87) Snap Scrolling Carousel	28
88) Hide Scrollbar (still scrollable)	28
89) CSS Triangle (tooltip arrow)	28
90) Checkerboard Transparency BG	29
Accessibility & UX Polish	29
91) Bigger Tap Targets on Mobile	29
92) Prevent Long Words Breaking Layout	30
93) Better Selection Color	30
94) Cursor for Disabled Links	30
Modern CSS Features	30
95) :has() Parent Selector (modern)	30
96) Color Mix (dynamic)	30
97) Scroll-Driven Sticky Section Title	31
98) Reduce Layout Jank for Fonts	31
99) Print Styles (basic)	31
100) Debug Layout (outline everything)	31

GitHub <https://github.com/lsvakis/100-CSS-Snippets>

1) Modern CSS Reset (minimal)

```
*,
*::before,
*::after { box-sizing: border-box; }

html, body { height: 100%; }

body { margin: 0; line-height: 1.5; -webkit-font-smoothing:
antialiased; }

img, picture, video, canvas, svg { display: block; max-width: 100%; }

input, button, textarea, select { font: inherit; }
```

Does: Prevents common layout surprises (box sizing, default margins, media sizing).

Use it: Put at the top of your stylesheet for consistent baseline behavior.

2) Fluid Container

```
.container { width: min(1100px, 92%); margin-inline: auto; }
```

Does: Keeps content readable on large screens and padded on small screens.

Use it: Wrap page sections in `.container`.

3) Responsive Typography (clamp)

```
h1 { font-size: clamp(1.8rem, 3vw + 1rem, 3.2rem); }
```

Does: Automatically scales font size between min and max based on viewport.

Use it: Great for headings on responsive pages.

4) Better Line Length

```
.prose { max-width: 65ch; }
```

Does: Limits text width to a comfortable reading length.

Use it: Apply to blog/article text containers.

5) Center Anything (Grid)

```
.center { display: grid; place-items: center; }
```

Does: Perfect vertical + horizontal centering.

Use it: For modals, empty states, hero sections.

6) Sticky Header

```
.header {
  position: sticky;
  top: 0;
  z-index: 10;
  background: white;
}
```

Does: Keeps header visible while scrolling.

Use it: Add to nav bars; ensure background isn't transparent.

7) Smooth Scrolling

```
html { scroll-behavior: smooth; }
```

Does: Smoothly scrolls to anchor links.

Use it: Great for one-page sites with section links.

8) Scroll Margin for Anchors (fix sticky header overlap)

```
section { scroll-margin-top: 80px; }
```

Does: Adds space when jumping to anchors so content isn't hidden behind sticky header.

Use it: Set to your header height.

9) Nice Focus Ring (accessible)

```
:focus-visible {
  outline: 3px solid #3b82f6;
  outline-offset: 3px;
}
```

Does: Shows focus only for keyboard users (better UX).

Use it: Improves accessibility on forms/buttons/links.

10) Hide Visually, Keep for Screen Readers

```
.sr-only {  
  position: absolute; width: 1px; height: 1px;  
  padding: 0; margin: -1px; overflow: hidden;  
  clip: rect(0,0,0,0); white-space: nowrap; border: 0;  
}
```

Does: Hides text visually while keeping it accessible.

Use it: Labels, helper text, “Skip to content”.

Layout & Positioning

11) Flex Row with Gap

```
.row { display: flex; gap: 1rem; align-items: center; }
```

Does: Row layout with consistent spacing.

Use it: Toolbars, nav items, button groups.

12) Wrap Items Responsively

```
.wrap { display: flex; flex-wrap: wrap; gap: 12px; }
```

Does: Items wrap onto new lines automatically.

Use it: Tag lists, chips, card grids.

13) Equal Columns (Flex)

```
.cols > * { flex: 1; }  
.cols { display: flex; gap: 1rem; }
```

Does: All children become equal-width columns.

Use it: Simple two/three-column layouts.

Get more Resources from Laurence Svekis <https://basescripts.com/>

14) Auto-fit Grid Cards

```
.grid {
  display: grid;
  gap: 1rem;
  grid-template-columns: repeat(auto-fit, minmax(220px, 1fr));
}
```

Does: Responsive grid that adapts number of columns automatically.

Use it: Card layouts without media queries.

15) Sidebar Layout (Grid)

```
.layout {
  display: grid;
  grid-template-columns: 280px 1fr;
  gap: 1rem;
}
@media (max-width: 900px) {
  .layout { grid-template-columns: 1fr; }
}
```

Does: Sidebar + main content; collapses on small screens.

Use it: Dashboards, docs sites.

16) Full-Height App Shell

```
.app { min-height: 100vh; display: grid; grid-template-rows: auto 1fr auto; }
```

Does: Header + main + footer, with main filling remaining height.

Use it: Pages where footer should sit at bottom.

17) Truncate Single Line

```
.truncate { overflow: hidden; text-overflow: ellipsis; white-space: nowrap; }
```

Does: Adds “...” when text overflows one line.

Use it: Card titles, table cells.

18) Clamp to 3 Lines

```
.clamp-3 {
  display: -webkit-box;
  -webkit-line-clamp: 3;
  -webkit-box-orient: vertical;
  overflow: hidden;
}
```

Does: Limits text to 3 lines with truncation.

Use it: Previews, summaries.

19) Absolute Center (classic)

```
.abs-center {
  position: absolute;
  inset: 50% auto auto 50%;
  transform: translate(-50%, -50%);
}
```

Does: Centers an element inside its positioned parent.

Use it: Badges, overlays.

20) Safe Area Padding (iOS notch)

```
.safe {
  padding: max(16px, env(safe-area-inset-top))
           max(16px, env(safe-area-inset-right))
           max(16px, env(safe-area-inset-bottom))
           max(16px, env(safe-area-inset-left));
}
```

Does: Prevents content from hiding behind notches/rounded corners.

Use it: Full-screen mobile layouts.

Spacing, Sizing, and Units

21) Consistent Section Spacing

```
section { padding-block: 4rem; }
```

Does: Adds vertical breathing room.

Use it: Marketing pages.

22) Fluid Spacing Scale

```
:root {
  --space: clamp(1rem, 2vw, 2rem);
}
.card { padding: var(--space); }
```

Does: Padding grows slightly with screen size.

Use it: Cards/containers.

23) Logical Properties (better for RTL)

```
.box { margin-inline: auto; padding-inline: 1rem; }
```

Does: Works for left-to-right and right-to-left layouts automatically.

Use it: International-friendly layouts.

24) Square Avatar

```
.avatar { width: 48px; aspect-ratio: 1; border-radius: 999px;
object-fit: cover; }
```

Does: Perfect circles for images.

Use it: Profile pictures.

25) Responsive Images Crop

```
.hero-img { width: 100%; height: 50vh; object-fit: cover; }
```

Does: Crops image to fill area without distortion.

Use it: Hero banners.

Color, Backgrounds, and Effects

26) Subtle Gradient Background

```
.bg {
  background: linear-gradient(135deg, #0f172a, #111827 50%, #1f2937);
}
```

Does: Adds depth without images.

Use it: Headers, hero sections.

27) Background Pattern (CSS-only dots)

```
.dots {
  background-image: radial-gradient(#e5e7eb 1px, transparent 1px);
  background-size: 16px 16px;
}
```

Does: Light dotted pattern.

Use it: Sections or cards (keep subtle).

28) Glassmorphism Card (simple)

```
.glass {
  background: rgba(255,255,255,0.08);
  backdrop-filter: blur(10px);
  border: 1px solid rgba(255,255,255,0.15);
  border-radius: 16px;
}
```

Does: “Frosted” glass effect on supported browsers.

Use it: Over background images/gradients.

29) Soft Shadow

```
.shadow {
  box-shadow: 0 10px 25px rgba(0,0,0,0.12);
}
```

Does: Elevates elements without harsh edges.

Use it: Cards, modals.

30) Hover Lift

```
.lift { transition: transform 160ms ease, box-shadow 160ms ease; }
.lift:hover { transform: translateY(-4px); box-shadow: 0 14px 28px
  rgba(0,0,0,0.14); }
```

Does: Makes cards feel clickable.

Use it: Feature cards, product tiles.

Borders & Shapes

31) Nice Rounded Corners

```
.round { border-radius: 16px; }
```

Does: Modern rounded look.

Use it: Buttons/cards/images.

32) Gradient Border

```
.grad-border {
  border: 2px solid transparent;
  border-radius: 16px;
  background:
    linear-gradient(#fff, #fff) padding-box,
    linear-gradient(135deg, #60a5fa, #a78bfa) border-box;
}
```

Does: Creates a gradient outline without extra wrappers.

Use it: Highlighted cards.

33) Dashed Divider

```
.divider { border-top: 1px dashed #d1d5db; }
```

Does: Gentle separation.

Use it: Between sections or list items.

34) Fancy Corner Ribbon (badge)

```
.ribbon {
  position: absolute; top: 12px; right: -40px;
  transform: rotate(45deg);
  padding: 6px 48px;
  background: #10b981; color: white;
}
```

Does: Diagonal “NEW”/“SALE” badge.

Use it: On cards with `position: relative;`

Typography

35) Font Smoothing + Base Type

```
body { font-family: system-ui, -apple-system, Segoe UI, Roboto, Arial,
sans-serif; }
```

Does: Uses fast, clean system fonts.

Use it: Default typography for apps.

36) Better Headline Spacing

```
h1,h2,h3 { line-height: 1.15; letter-spacing: -0.02em; }
```

Does: Makes headings feel tighter and more modern.

Use it: Marketing pages.

37) Balanced Text Wrap

```
h1, h2 { text-wrap: balance; }
```

Does: Balances line breaks (supported in modern browsers).

Use it: Headlines so they don't look awkward.

38) Numerals Align Nicely

```
.nums { font-variant-numeric: tabular-nums; }
```

Does: Makes digits equal width (great for tables/counters).

Use it: Prices, dashboards.

Buttons & UI Components

39) Solid Button

```
.btn {
  padding: 10px 14px;
  border-radius: 12px;
  border: 1px solid #111827;
  background: #111827;
  color: white;
  cursor: pointer;
}
.btn:hover { opacity: 0.92; }
```

Does: Clean, modern button style.

Use it: Primary actions.

40) Outline Button

```
.btn-outline {
  padding: 10px 14px;
  border-radius: 12px;
  border: 1px solid #cbd5e1;
  background: transparent;
}
.btn-outline:hover { background: #f1f5f9; }
```

Does: Secondary button look.

Use it: Less prominent actions.

41) Disabled Style

```
button:disabled { opacity: 0.5; cursor: not-allowed; }
```

Does: Clear disabled state.

Use it: Forms and loading states.

42) Loading Spinner (CSS-only)

```
.spinner {
  width: 18px; aspect-ratio: 1;
  border: 2px solid #cbd5e1;
  border-top-color: #111827;
  border-radius: 50%;
  animation: spin 700ms linear infinite;
}
@keyframes spin { to { transform: rotate(360deg); } }
```

Does: Animated spinner without images.

Use it: Inside buttons or loading screens.

43) Pill Badge

```
.badge {
  display: inline-block;
  padding: 4px 10px;
```

```
border-radius: 999px;
background: #eef2ff;
}
```

Does: Small pill tag.

Use it: Status labels, categories.

44) Card Component

```
.card {
  border: 1px solid #e5e7eb;
  border-radius: 16px;
  padding: 16px;
  background: white;
}
```

Does: Base card styling.

Use it: Reusable UI layout.

45) Modal Backdrop

```
.backdrop {
  position: fixed; inset: 0;
  background: rgba(0,0,0,0.5);
}
```

Does: Dim background behind modal.

Use it: Pair with a centered modal panel.

Forms

46) Clean Input

```
.input {
  width: 100%;
  padding: 10px 12px;
```

Get more Resources from Laurence Svekis <https://basescripts.com/>

```
border: 1px solid #cbd5e1;
border-radius: 12px;
}
.input:focus { border-color: #3b82f6; outline: none; }
```

Does: Modern input styling with focus feedback.

Use it: Text fields, search bars.

47) Input with Icon Layout

```
.field { position: relative; }
.field svg { position: absolute; left: 10px; top: 50%; transform:
translateY(-50%); }
.field input { padding-left: 36px; }
```

Does: Adds room for icon inside input.

Use it: Search inputs.

48) Checkbox Accent Color

```
input[type="checkbox"] { accent-color: #3b82f6; }
```

Does: Styles native checkboxes easily.

Use it: Modern browsers.

49) Validation States

```
.input.ok { border-color: #10b981; }
.input.bad { border-color: #ef4444; }
```

Does: Visual feedback for valid/invalid fields.

Use it: Toggle classes via JS.

50) Form Row Spacing

```
.form { display: grid; gap: 12px; }
```

Does: Consistent spacing across form controls.

Use it: Wrap inputs in `.form`.

Images & Media

51) Cover Image in Card

```
.cover { width: 100%; aspect-ratio: 16/9; object-fit: cover;
border-radius: 12px; }
```

Does: Keeps consistent thumbnail shape.

Use it: Blog cards, video thumbnails.

52) Prevent Layout Shift (reserve space)

```
.media { aspect-ratio: 4/3; background: #f1f5f9; }
```

Does: Holds space while image loads.

Use it: Wrap images inside `.media`.

Lists, Tables, and Content

53) Nice List Spacing

```
ul { padding-left: 1.2rem; }
li { margin-block: 0.35rem; }
```

Does: Improves readability.

Use it: Articles/docs.

54) Zebra Table

```
table { width: 100%; border-collapse: collapse; }
```

```
tr:nth-child(even) { background: #f8fafc; }
td, th { padding: 10px; border-bottom: 1px solid #e5e7eb; }
```

Does: Easy-to-scan tables.

Use it: Data displays.

55) Sticky Table Header

```
thead th {
  position: sticky;
  top: 0;
  background: white;
}
```

Does: Keeps column headers visible while scrolling.

Use it: Tables inside a scroll container.

Animation & Motion

56) Fade In

```
.fade-in { animation: fade 300ms ease both; }
@keyframes fade { from { opacity: 0; } to { opacity: 1; } }
```

Does: Smooth entrance.

Use it: Modals, toast notifications.

57) Slide Up

```
.slide-up { animation: up 260ms ease both; }
@keyframes up { from { transform: translateY(10px); opacity: 0; } to {
  transform: translateY(0); opacity: 1; } }
```

Does: Subtle slide + fade.

Use it: Dropdowns, panels.

58) Respect Reduced Motion

```
@media (prefers-reduced-motion: reduce) {
  * { animation-duration: 0.001ms !important; transition-duration:
0.001ms !important; }
}
```

Does: Helps users who prefer less motion.

Use it: Global accessibility improvement.

59) Pulse Attention (subtle)

```
.pulse { animation: pulse 1.6s ease-in-out infinite; }
@keyframes pulse { 50% { transform: scale(1.03); } }
```

Does: Gentle attention cue.

Use it: “New” indicators (don’t overuse).

60) Skeleton Loading

```
.skeleton {
  background: linear-gradient(90deg, #e5e7eb, #f3f4f6, #e5e7eb);
  background-size: 200% 100%;
  animation: shimmer 1.2s linear infinite;
}
@keyframes shimmer { to { background-position: -200% 0; } }
```

Does: Shimmer placeholder while content loads.

Use it: Loading cards and lists.

Responsive Design Helpers

61) Mobile-first Breakpoint

```
@media (min-width: 768px) { .hide-md { display: none; } }
```

Does: Utility to hide elements on larger screens.

Use it: Pair with `.show-md` style.

62) Desktop-only

```
.desktop-only { display: none; }
@media (min-width: 1024px) { .desktop-only { display: block; } }
```

Does: Shows content only on bigger screens.

Use it: Sidebars, extra navigation.

63) Responsive Padding

```
.section { padding: 24px; }
@media (min-width: 900px) { .section { padding: 48px; } }
```

Does: More spacing on larger screens.

Use it: Layout polish.

64) Container Queries (modern)

```
.card-wrap { container-type: inline-size; }
@container (min-width: 420px) {
  .card { display: flex; gap: 12px; }
}
```

Does: Styles based on container size, not viewport.

Use it: Reusable components in different layouts.

Interactions, Hover, and State

65) Hover Underline Animation (links)

```
a {
  text-decoration: none;
```

```
background: linear-gradient(currentColor, currentColor) 0 100% / 0
2px no-repeat;
transition: background-size 180ms ease;
}
a:hover { background-size: 100% 2px; }
```

Does: Smooth underline grow effect.

Use it: Nav links, inline links.

66) Clickable Card

```
.card { cursor: pointer; }
.card:active { transform: scale(0.99); }
```

Does: Makes cards feel interactive.

Use it: When the whole card is clickable.

67) Disabled Pointer Events

```
.is-disabled { pointer-events: none; opacity: 0.6; }
```

Does: Prevents interaction with UI.

Use it: While loading or locked.

68) Prevent Text Selection

```
.no-select { user-select: none; }
```

Does: Stops accidental text highlight.

Use it: Buttons, draggable items.

69) Scrollbar Styling (WebKit)

```
::-webkit-scrollbar { width: 10px; }
::-webkit-scrollbar-thumb { background: #cbd5e1; border-radius: 10px;
}
```

Does: Makes scrollbar match UI (Chrome/Safari).

Use it: Light theming.

Advanced UI Patterns

70) Toast Position

```
.toast-area {
  position: fixed;
  right: 16px;
  bottom: 16px;
  display: grid;
  gap: 10px;
}
```

Does: Stack notifications neatly.

Use it: Render multiple toasts in this container.

71) Tooltip (CSS-only)

```
.tip { position: relative; }
.tip::after {
  content: attr(data-tip);
  position: absolute;
  left: 50%; bottom: 120%;
  transform: translateX(-50%);
  padding: 6px 10px;
  background: #111827;
  color: white;
  border-radius: 10px;
  opacity: 0;
  pointer-events: none;
  transition: opacity 150ms ease;
  white-space: nowrap;
}
.tip:hover::after { opacity: 1; }
```

Does: Tooltip from a data attribute.

Use it: Add `data-tip="Hello"` to any element with class `tip`.

72) Dropdown Panel

```
.menu { position: relative; }
.panel {
  position: absolute; top: 110%; right: 0;
  min-width: 220px;
  border: 1px solid #e5e7eb;
  background: white;
  border-radius: 14px;
  padding: 10px;
}
```

Does: Anchors a dropdown to a trigger.

Use it: Toggle `.panel` with JS.

73) Aspect Ratio Utility

```
.ratio-16-9 { aspect-ratio: 16 / 9; }
```

Does: Keeps fixed proportions.

Use it: Video embeds, thumbnails.

74) Blended Overlay on Image

```
.banner {
  background: url(hero.jpg) center/cover;
  background-blend-mode: multiply;
  background-color: rgba(0,0,0,0.45);
}
```

Does: Darkens image for readable text.

Use it: Hero sections.

75) Frosted Overlay for Text

```
.overlay {
  background: rgba(255,255,255,0.65);
  backdrop-filter: blur(8px);
  border-radius: 14px;
  padding: 16px;
}
```

Does: Keeps text readable over busy backgrounds.

Use it: Cards on photos.

CSS Variables & Theming

76) Theme Tokens

```
:root{
  --bg: #0b1220;
  --card: #111827;
  --text: #e5e7eb;
  --muted: #94a3b8;
  --accent: #60a5fa;
}
body { background: var(--bg); color: var(--text); }
```

Does: Centralizes colors for consistent theme.

Use it: Change one variable to re-skin the whole site.

77) Light/Dark Theme Switch

```
:root[data-theme="light"] { --bg: #ffffff; --text: #0f172a; }
:root[data-theme="dark"] { --bg: #0b1220; --text: #e5e7eb; }
```

Does: Theme based on attribute toggle.

Use it: Set `document.documentElement.dataset.theme = "dark"`.

78) Automatic Dark Mode

```
@media (prefers-color-scheme: dark) {
  :root { --bg: #0b1220; --text: #e5e7eb; }
}
```

Does: Adjusts colors based on user preference.

Use it: Default theming without JS.

Performance & Safety

79) Content Visibility (perf boost)

```
.card-list > * { content-visibility: auto; contain-intrinsic-size:
300px; }
```

Does: Skips rendering off-screen elements for faster scroll.

Use it: Long lists/grids (modern browsers).

80) Prevent Image Drag (UX)

```
img { -webkit-user-drag: none; }
```

Does: Avoids awkward dragging on some browsers.

Use it: UI icons, logos.

More Components & Patterns

81) Breadcrumbs

```
.breadcrumbs { display: flex; gap: 8px; flex-wrap: wrap; }
.breadcrumbs a { color: #3b82f6; }
```

Does: Clean breadcrumb layout.

Use it: Navigation trails.

82) Tabs (base styles)

```
.tabs { display: flex; gap: 8px; border-bottom: 1px solid #e5e7eb; }
.tab { padding: 10px 12px; border-radius: 10px 10px 0 0; }
.tab[aria-selected="true"] { background: #f1f5f9; }
```

Does: Visual tab UI.

Use it: With JS for switching content.

83) Progress Bar

```
.progress { height: 10px; background: #e5e7eb; border-radius: 999px;
overflow: hidden; }
.progress > span { display: block; height: 100%; width: 60%;
background: #3b82f6; }
```

Does: Simple progress indicator.

Use it: Set width dynamically.

84) Notification Banner

```
.banner {
  padding: 12px 14px;
  border: 1px solid #fde68a;
  background: #fffbeb;
  border-radius: 14px;
}
```

Does: Highlight important message.

Use it: Warnings/info callouts.

85) “Chip” Input Tags

```
.chips { display: flex; flex-wrap: wrap; gap: 8px; }
.chip { padding: 6px 10px; border-radius: 999px; background: #f1f5f9;
}
```

Does: Compact tag UI.

Use it: Categories, filters.

Tricks & Utilities

86) Prevent Overscroll Bounce (mobile)

```
body { overscroll-behavior: none; }
```

Does: Reduces scroll chaining/bounce.

Use it: Web apps; be careful with usability.

87) Snap Scrolling Carousel

```
.carousel {  
  display: flex;  
  overflow-x: auto;  
  gap: 12px;  
  scroll-snap-type: x mandatory;  
}  
.carousel > * { scroll-snap-align: start; }
```

Does: Scroll snaps to items.

Use it: Horizontal card scrollers.

88) Hide Scrollbar (still scrollable)

```
.hide-scrollbar { scrollbar-width: none; }  
.hide-scrollbar::-webkit-scrollbar { display: none; }
```

Does: Cleaner look for carousels.

Use it: Only when obvious that it scrolls.

89) CSS Triangle (tooltip arrow)

Get more Resources from Laurence Svekis <https://basescripts.com/>

```
.arrow {
  width: 0; height: 0;
  border-left: 8px solid transparent;
  border-right: 8px solid transparent;
  border-top: 8px solid #111827;
}
```

Does: Creates a triangle shape.

Use it: Tooltip pointers, speech bubbles.

90) Checkerboard Transparency BG

```
.checker {
  background:
    linear-gradient(45deg, #e5e7eb 25%, transparent 25%) 0 0/20px
    20px,
    linear-gradient(45deg, transparent 75%, #e5e7eb 75%) 0 0/20px
    20px,
    linear-gradient(45deg, #e5e7eb 25%, transparent 25%) 10px
    10px/20px 20px,
    linear-gradient(45deg, transparent 75%, #e5e7eb 75%) 10px
    10px/20px 20px;
}
```

Does: Fake transparency background.

Use it: Image editors/previews.

Accessibility & UX Polish

91) Bigger Tap Targets on Mobile

```
button, a { min-height: 44px; }
```

Does: Easier tapping on touch screens.

Use it: Especially for nav bars.

92) Prevent Long Words Breaking Layout

```
.prose { overflow-wrap: anywhere; }
```

Does: Stops long URLs/words from blowing up the layout.

Use it: Comments, user-generated content.

93) Better Selection Color

```
::selection { background: #bfdbfe; }
```

Does: Makes text selection match theme.

Use it: Branding polish.

94) Cursor for Disabled Links

```
a[aria-disabled="true"] { pointer-events: none; opacity: 0.5; cursor: not-allowed; }
```

Does: Clear disabled link behavior.

Use it: Menus with locked items.

Modern CSS Features

95) :has() Parent Selector (modern)

```
.card:has(a:hover) { transform: translateY(-3px); }
```

Does: Styles parent when a child matches a condition.

Use it: Card hover when inner link is hovered (supported in modern browsers).

96) Color Mix (dynamic)

```
.btn {  
  background: color-mix(in srgb, #3b82f6 85%, black);  
}
```

Does: Creates derived colors without manual tweaking.

Use it: Theme systems.

97) Scroll-Driven Sticky Section Title

```
.section-title {
  position: sticky;
  top: 70px;
  background: white;
}
```

Does: Makes section headers stick while that section scrolls.

Use it: Docs pages, long forms.

98) Reduce Layout Jank for Fonts

```
:root { font-synthesis: none; }
```

Does: Avoids synthetic bold/italic in some browsers.

Use it: When using custom fonts and consistent typography matters.

99) Print Styles (basic)

```
@media print {
  nav, footer { display: none; }
  body { color: black; background: white; }
}
```

Does: Cleaner printing.

Use it: Docs pages, invoices.

100) Debug Layout (outline everything)

```
* { outline: 1px solid rgba(255,0,0,0.12); }
```

Does: Visualizes box boundaries to debug spacing issues.

Use it: Temporarily while building layouts.

