

Cheat-Sheet: CSS-Einheiten für Responsive Design

1. Relative Einheiten (Schrift & Abstände)

Diese Einheiten passen sich an die Einstellungen des Nutzers oder des Browsers an – essenziell für die Barrierefreiheit (A11y).

- **rem (Root em):** Orientiert sich an der globalen Browser-Schriftgröße (meist 16px).
 - *Best Use Case:* Fließtext & Überschriften. Garantiert, dass die Seite lesbar bleibt, wenn Nutzer ihre Browserschrift vergrößern.
- **em (Element em):** Orientiert sich an der Schriftgröße des direkten Elternelements.
 - *Best Use Case:* Padding/Margin in Buttons. Wenn du die Schrift im Button vergrößerst, wächst der Abstand automatisch proportional mit.

2. Viewport-Einheiten (Layout & Sektionen)

Diese Einheiten ignorieren die Größe von Containern und schauen direkt auf das Sichtfenster (Viewport) des Geräts.

- **vh (Viewport Height):** Entspricht exakt 1% der sichtbaren Bildschirmhöhe.
 - *Best Use Case:* Hero-Sektionen. Mit **100vh** füllt eine Sektion immer exakt den ganzen Bildschirm aus.
- **vw (Viewport Width):** Entspricht exakt 1% der sichtbaren Bildschirmbreite.
 - *Best Use Case:* Responsive Typografie. Schrift, die auf riesigen Monitoren mitwächst und auf kleinen Bildschirmen schrumpft.

3. Die Prozent-Einheit (%)

Im Gegensatz zu Viewport-Einheiten ist Prozent immer in seinem Container "gefangen".

- **Bezug:** Das direkte Elternelement. Eine Spalte mit **50%** ist immer genau halb so breit wie die Reihe, in der sie liegt – egal wie breit der gesamte Bildschirm ist.
- *Best Use Case:* Spaltenlayouts und innere Container-Breiten.

💡 **Profi-Tipp:** Pixel (px) werden im modernen Webdesign fast nur noch für winzige, fixe Details (wie **border-width: 1px**) genutzt. Wer responsive bauen will, muss "relativ" denken lernen!