

RWD | Responsive Webdesign

Responsive Webdesign bedeutet, dass sich das **Aussehen der Webseite an das Endgerät des Nutzers anpasst**. Die **Webseite reagiert** (engl., response) auf die **Rahmenbedingungen des benutzten Geräts (Größe, Auflösung, usw.)** und zeigt eine **darauf abgestimmte Seite an!**



Das **Ziel von responsive Design** ist, dass die Webseite **überall gut aussieht** und **gut zu bedienen** ist!

Webseitenbeispiele

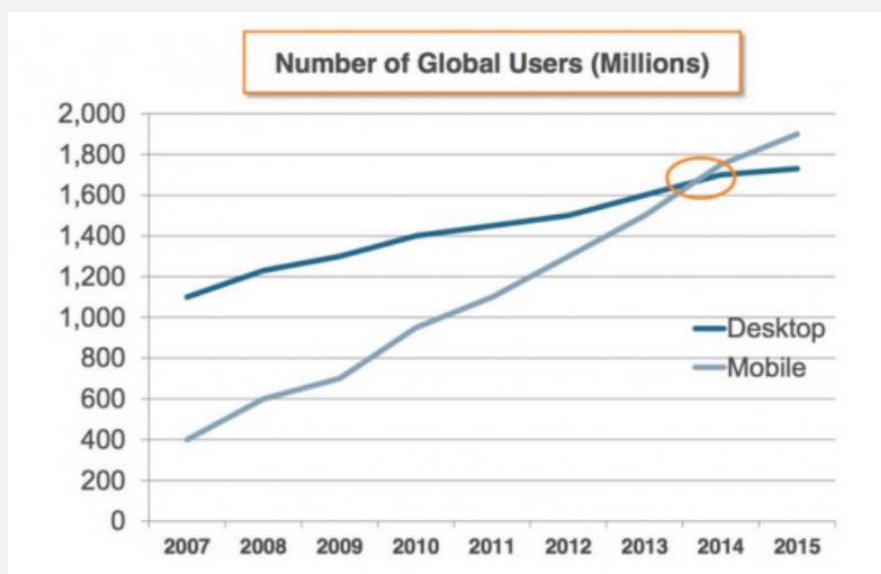
W3Schools	https://www.w3schools.com/
Saturn	http://www.saturn.at/de/category/_smartphone-381496.html
AllesAuto.at	https://www.allesauto.at/

Statistiken

Browserverteilung

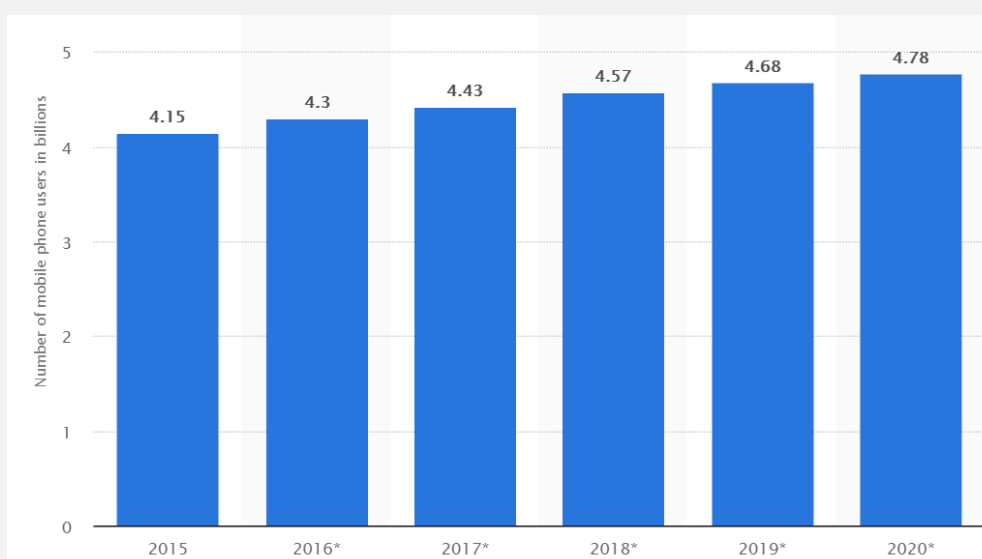
2018	<u>Chrome</u>	<u>Edge/IE</u>	<u>Firefox</u>	<u>Safari</u>	<u>Opera</u>
December	79.6 %	3.8 %	10.1 %	3.4 %	1.7 %

W3Schools.com



comScore: Morgan Stanley Research

Number of Mobile Users: Trend



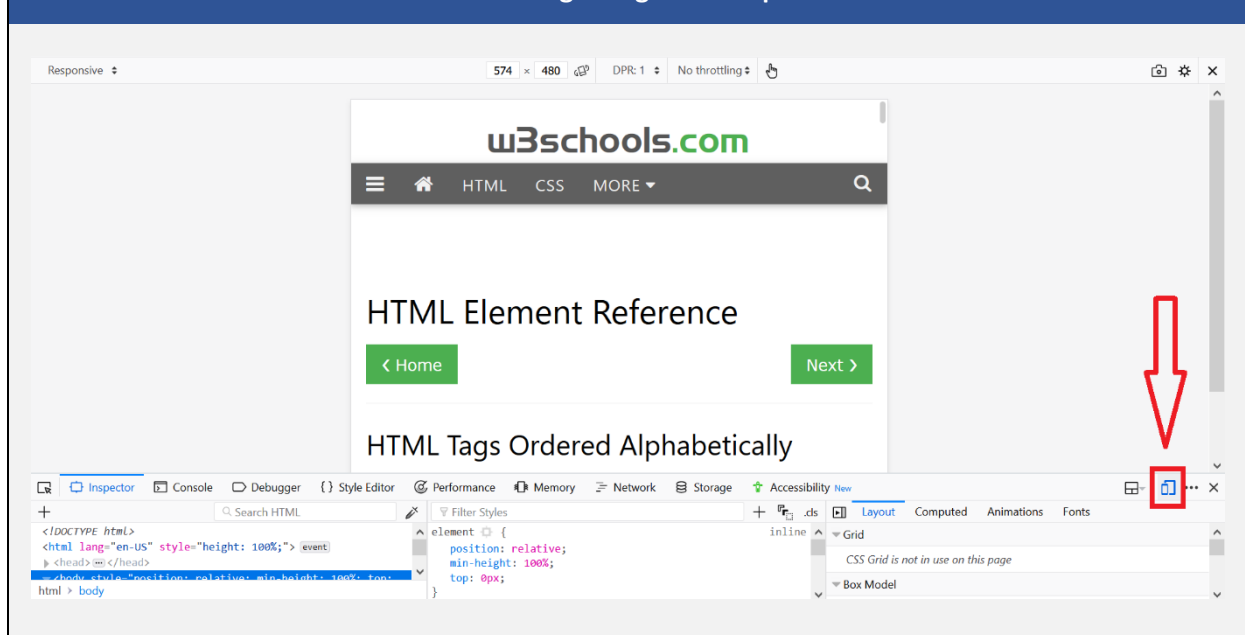
Statista.com

Technische Umsetzung

- **JavaScript** Auslesen der Clientinformationen → Seite passend formatieren

könnte vom User deaktiviert sein!
- **CSS3** Media Queries stellen → passenden Code einbinden

Entwicklerumgebung Chrome | Firefox



Viewport

Für mobile Endgeräte ist es wichtig, den Viewport anzugeben (er teilt dem Browser die Dimensionen und das Skalierverhalten mit). Für responsive Design ist in der Regel nur folgender Code sinnvoll:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

CSS3 Media Queries | @media

Seit CSS3 kann man in einem Stylesheet **Abfragen** (engl., query) definieren. Nur wenn deren Bedingungen erfüllt sind, wird der entsprechende CSS-Code angezeigt.

Eine CSS Media Query entspricht daher im Grunde einer **IF-Anweisung**.

Tutorial: https://www.w3schools.com/css/css_rwd_mediaqueries.asp

Referenz: https://www.w3schools.com/cssref/css3_pr_mediaquery.asp

Wichtige Aspekte:

- **Breakpoint(s)** bei welcher Bedingung (z.B. Auflösung) soll sich das Aussehen ändern?
es kann auch mehrere Breakpoints geben!
- **Mobile First** zuerst wird das Layout für mobile Endgeräte programmiert,
über Media Queries wird dann der passende Code für größere Geräte
dynamisch eingebunden
- **Desktop First** zuerst wird das Layout für größere Geräte programmiert,
über Media Queries wird der passende Code für mobile Geräte eingebunden

Syntax:

```
@media not|only mediatype and (mediafeature and|or|not mediafeature) {  
    CSS-Code;  
}
```

mediatype:

- all Query ist unabhängig vom Medientyp
- print Nur Printmedien (z.B. drucken)
- screen Bildschirme (Computer, Tablet, Smartphone)
- speech Screenreaders (Geräte, die Text vorlesen)

mediafeature (nur ein Auszug, vollständige Liste siehe Referenz!)

- (min-width: 600px) minimale erforderliche Größe
- (max-width: 800px) maximal erlaubte Größe
- (orientation: landscape) Handy im Hoch- oder Querformat?

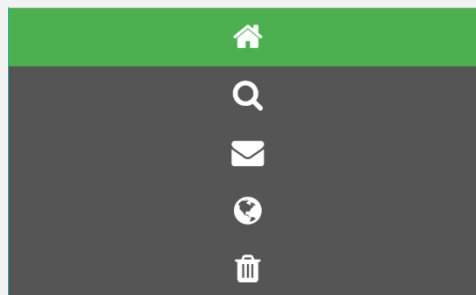
Übung 1

Responsive Navigation

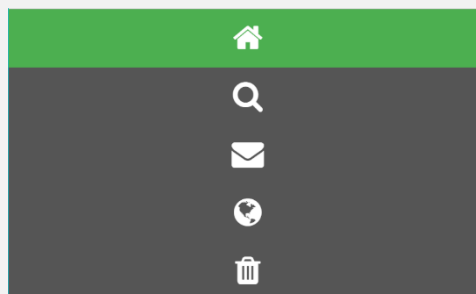
Erstelle eine **responsive Navigation**: stehen **weniger als 800 Pixel Breite** zur Verfügung, soll die Navigation von einer horizontalen Anordnung in ein bildschirmfüllendes vertikales Layout wechseln!



weniger als 800 Pixel Breite?



1. Verwende die **Vorlage** und erstelle ein responsives Design nach **Desktop First**!
2. Erstelle eine **zweite Version**, bei der du den Code als **Mobile First** umsetzt!



mehr als 800 Pixel Breite?



3. Erstelle eine **dritte Version**, bei der du folgendes **mobiles Design** umsetzt:



Übung 2 Responsive Galerie

Erstelle eine **responsive Bildergalerie**!

- Die Galerie wird innerhalb eines Wrappers zentriert.
- Das Standarddesign für **Desktop-Computer** ist eine Galerie mit **4 Spalten**!
- Responsive Layout für **Tablet-Geräte**:
unter 1080 Pixel Breite soll das Layout auf ein **2 Spalten Layout** wechseln!
- **Unter 780 Pixel** Breite soll das Layout auf ein **1 Spalten Layout** wechseln!



Übung 3 Responsive Homepage

Du wirst beauftragt für die HTL Leonding eine **responsive Spoiler-Seite** umzusetzen.

Dein Beitrag besteht aus der Umsetzung des responsiven Teils, das bedeutet

- **Implementierung des Grid-Systems im HTML-Code**
- **Codierung der nötigen CSS-Mediaqueries**

Gefordert sind folgende responsiven Gestaltungen:

- **Home-Navigation Desktop** **2-Spalten Layout**



- **Home-Navigation Mobile** **1-Spalten Layout (alles untereinander)**

- **Unterseiten-Navigation Desktop** **4-Spalten Layout**



- **Unterseiten-Navigation Mobile** **1-Spalten Layout (alles untereinander)**

Tipp fürs Grid-System:

Suche im HTML Quellcode nach den Klassen *row* und *column* – das sind jene Elemente, die eine Zuweisung im Grid benötigen.

Hinweis:

Die Seiten sind als Theme aufgebaut. Die Hauptdateien inkludieren noch zusätzlich andere PHP Files wie Header, Navigation und Footer!