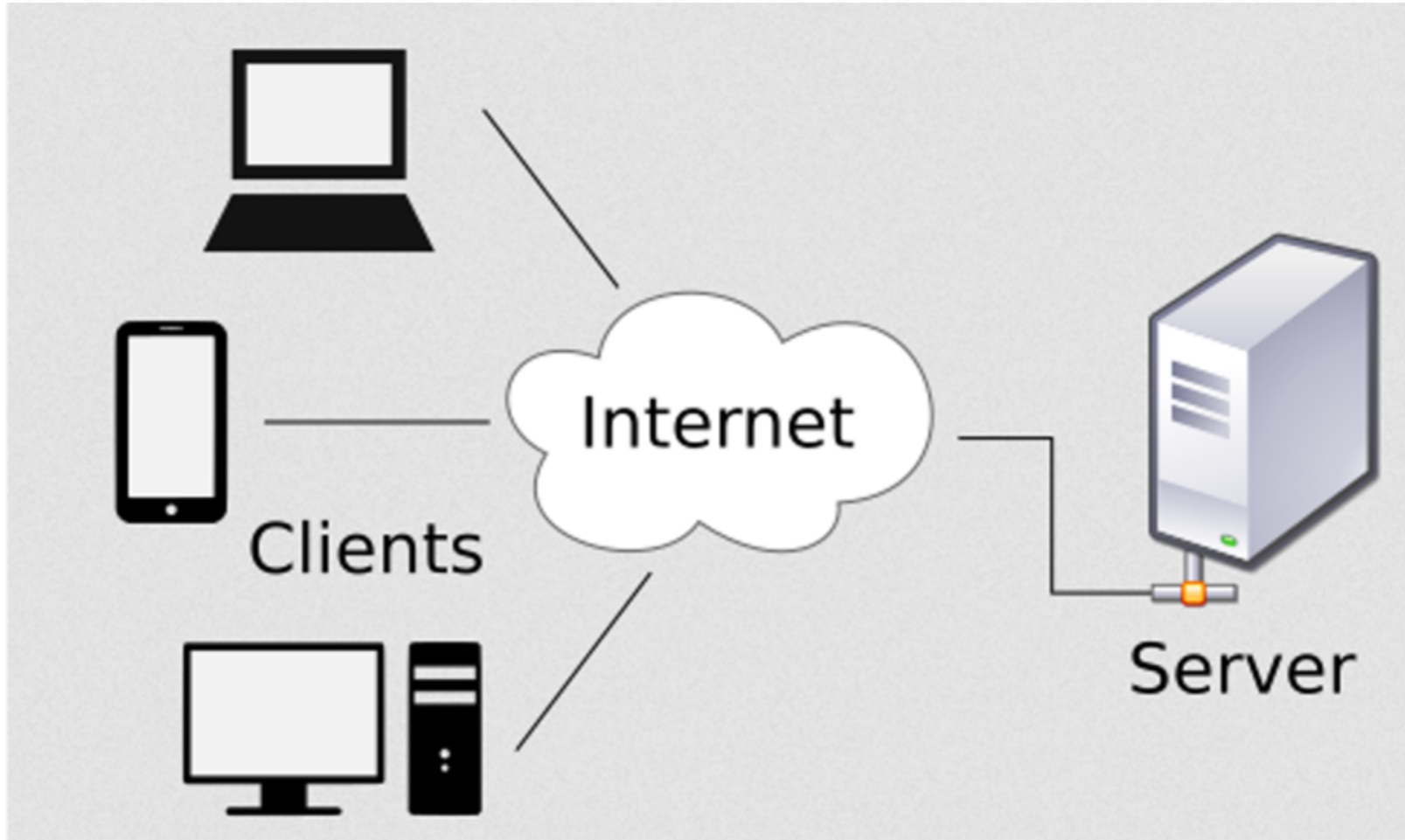


Grundlagen

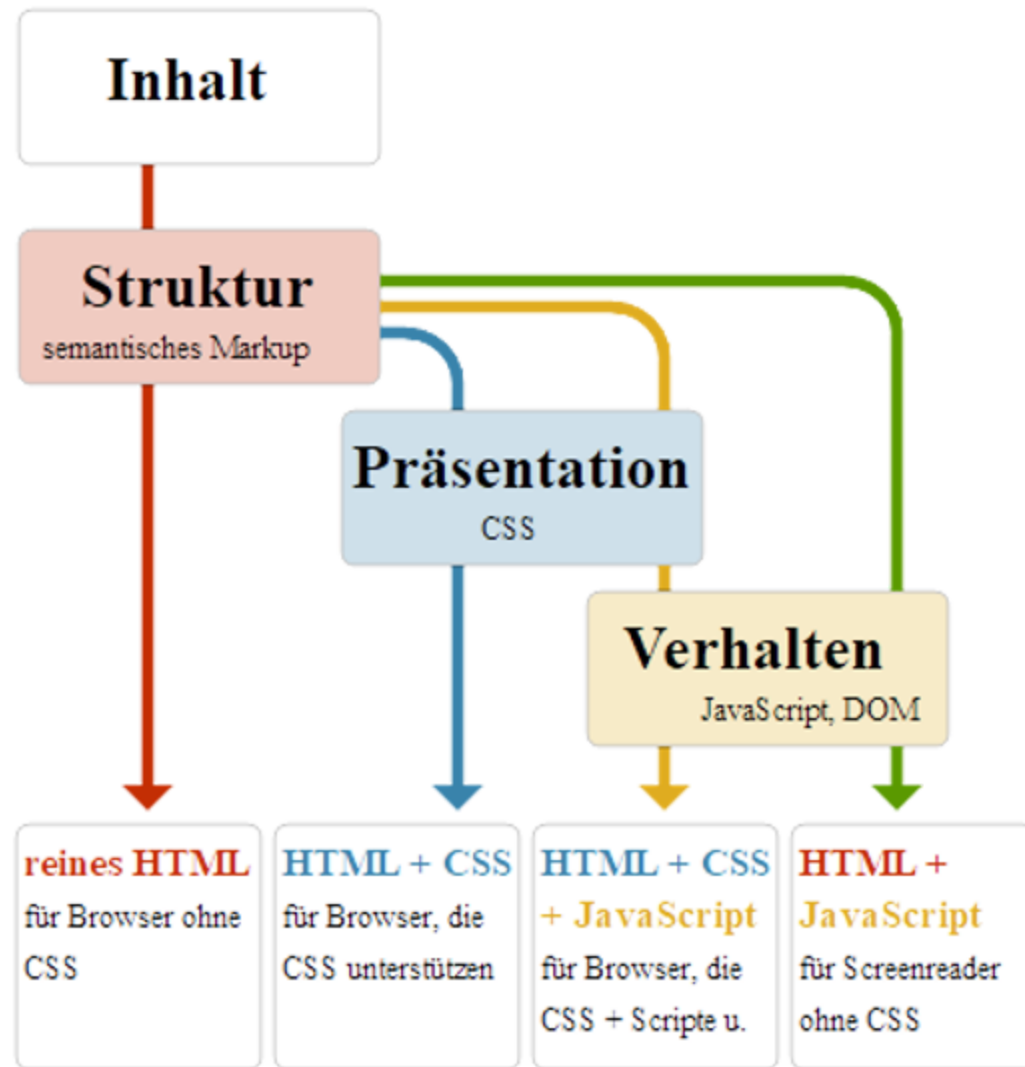
Client - Server



URL

<https://www.philipackermann.de:80/books/web.html?language=de#chapter7>

Aufbau von Webapplikationen





Linux



Apache



MySQL



PHP

MERN



Werkzeuge

Inspektor

Konsole

Debugger

Netzwerkanalyse

Stilbearbeitung

Laufzeitanalyse

Speicher

>>

...

X

HTML durchsuch

+

✎

Stile filtern

:hov .cls

+

📄

Layout

Berechnet

Änderungen

Schriftarten

▼

<!DOCTYPE html>

<html lang="de">

>

</html>

<body>

>

</body>

</html>

Element {

body, h1, dark.css:7

h2, h3, h4,

a, p, blockquote cite {

color: #ddd;

}

body { dark.css:2

background-color: #333;

}

body { style.css:8

font-family: sans-serif;

}

body { normalize.css:23

margin: 0;

}

Geerbt von html

html { normalize.css:11

line-height: 1.15;

webkit-text-size-adjust: 100%;

}

▼ Flexbox

Flex-Behälter oder -Element auswählen, um fortzufahren.

▼ Raster

Raster hervorheben

☐ ul.wp-block-gallery.columns-2.is-cropped

☐ ul.wp-block-gallery.columns-3.is-cropped

☐ ul.wp-block-gallery.columns-3.is-cropped

☐ ul.wp-block-gallery.columns-3.is-cropped

Anzeigeeinstellungen für Raster

☒ Zeilennummern anzeigen

☐ Bereichsnamen anzeigen

☐ Unendliche Linien

▼ Box-Modell

margin

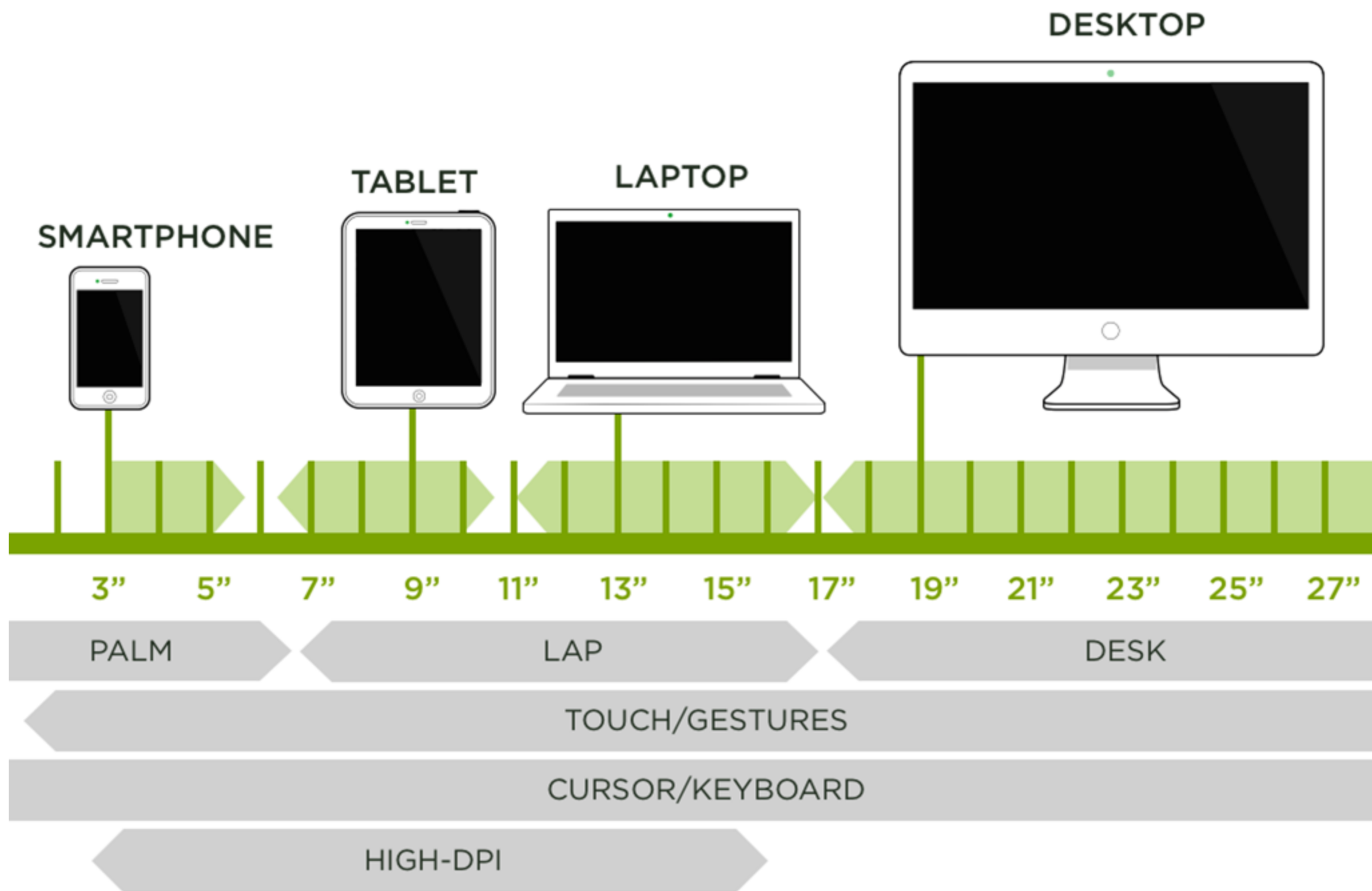
border

padding

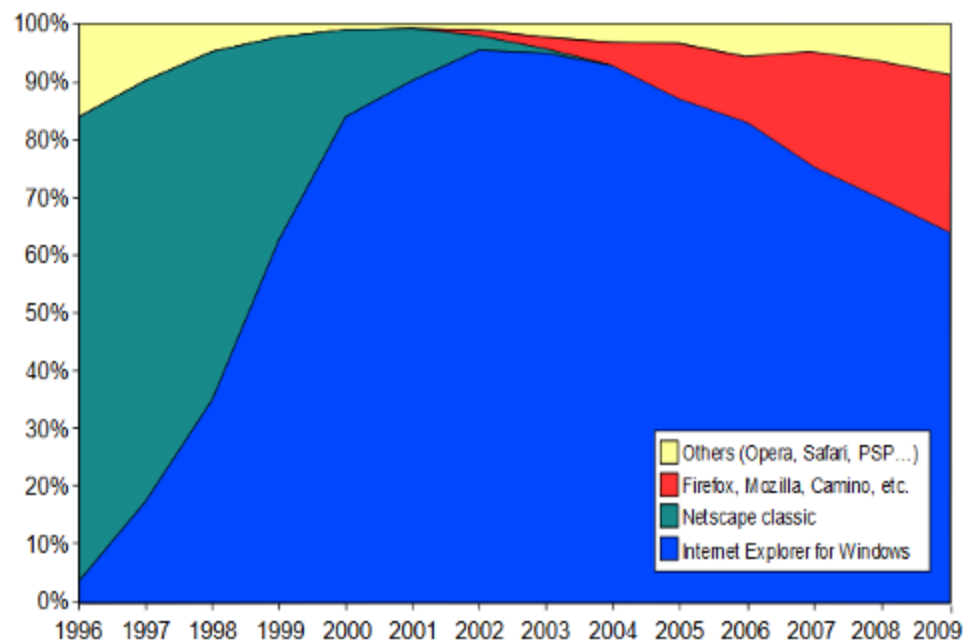
823 × 2397.2

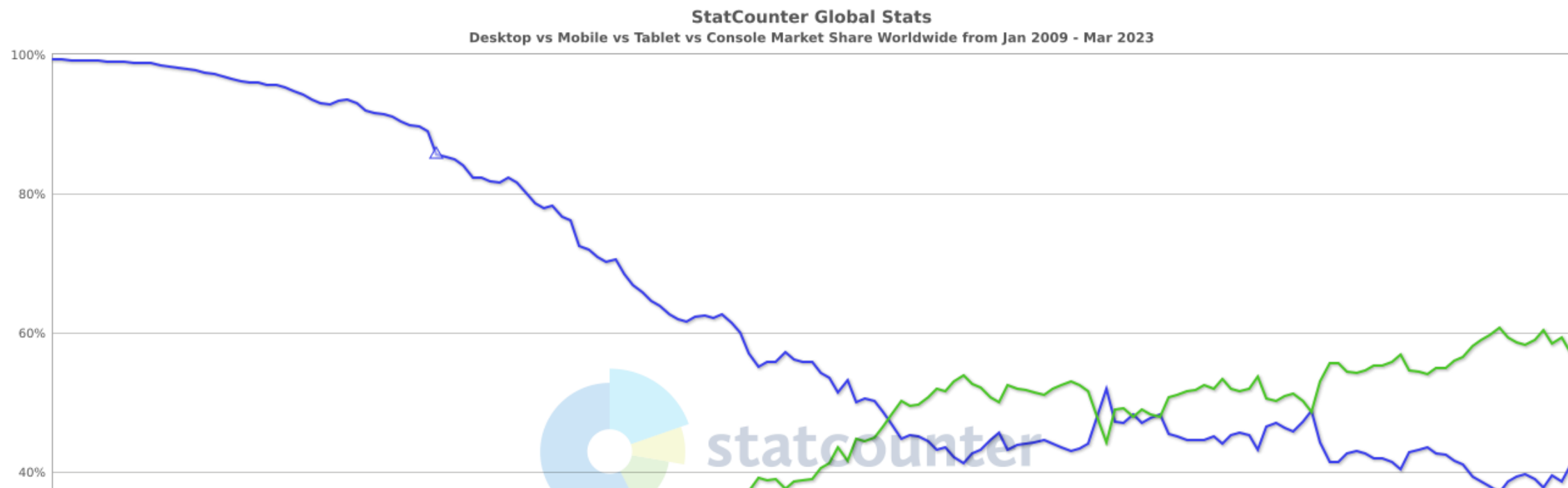
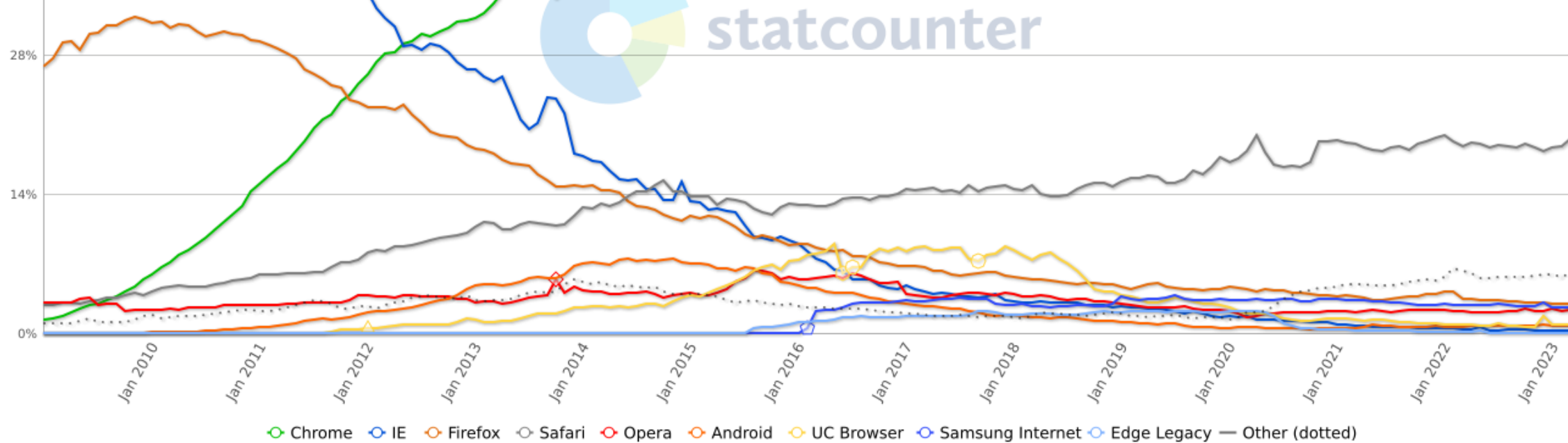
html > body

Webseiten strukturieren mit HTML



Browser Wars





Layoutkonzepte

- <http://info.cern.ch/hypertext/WWW/TheProject.html>
- Framesets
- Tabellen
- Cascading Style Sheets (CSS)
- Fixed vs. Liquid Layout
- Responsive Webdesign
- Device Agnostic
- Mobile First

Grundstruktur

```
<!DOCTYPE html>
<html lang="de">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Titel</title>
  </head>
  <body>

  </body>
</html>
```

Webseiten gestalten mit CSS

Box Model

Box Model

The diagram illustrates the Box Model with the following dimensions:

- margin:** 40px (yellow dashed border)
- border:** 5px (dark grey solid border)
- padding:** 40px (light purple area)
- content:** 300x150 (light blue dashed border)

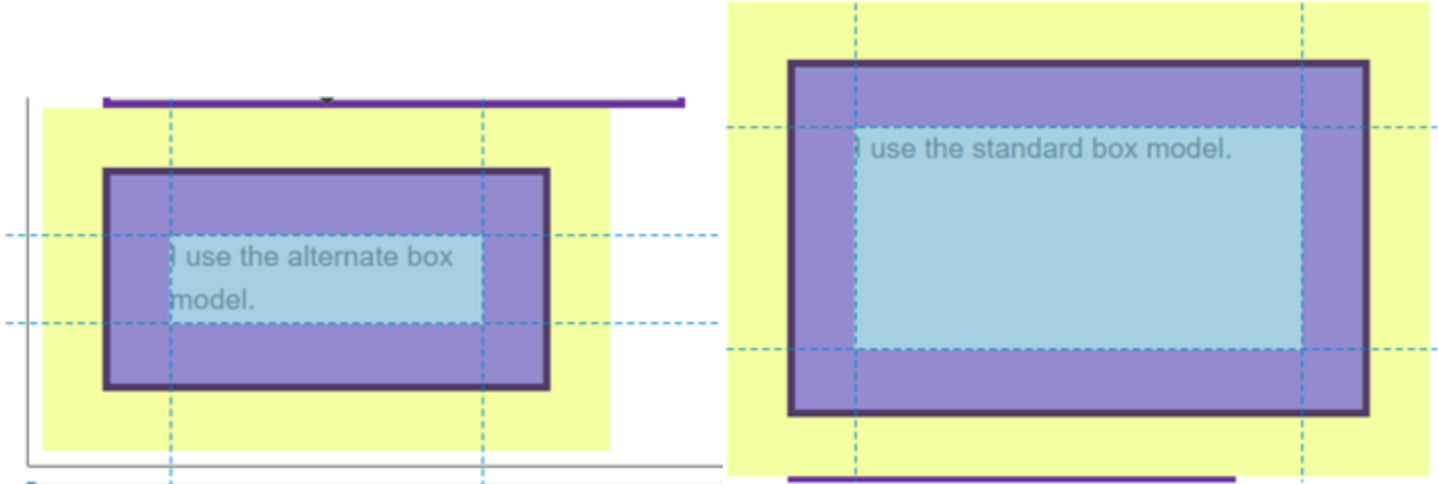
Labels 'margin', 'border', and 'padding' are placed near their respective layers. The content area is labeled '300x150'.

390x240

static

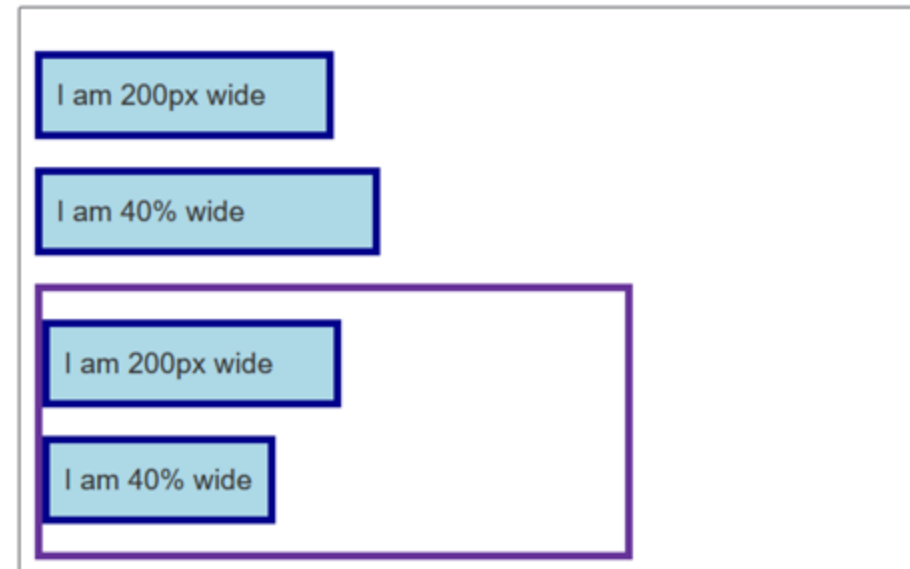
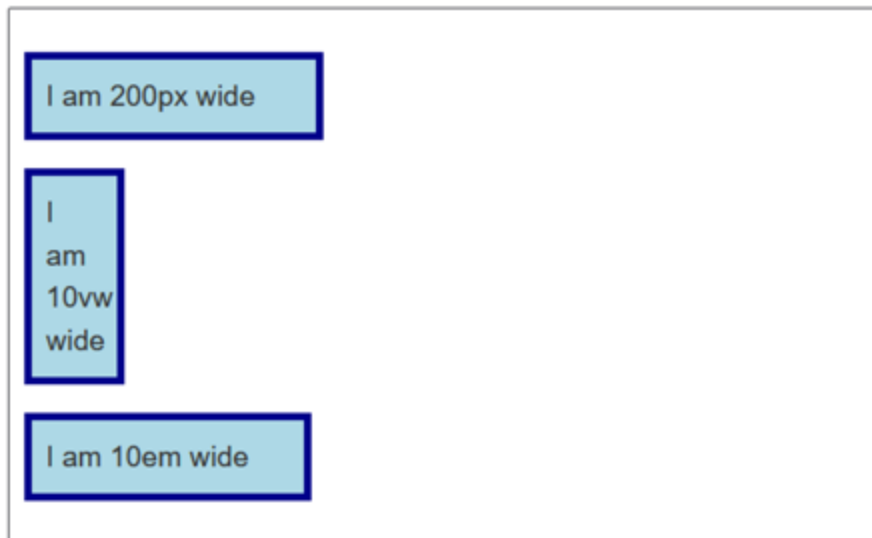
Box Model Properties

box-sizing	content-box	line-height	28.8px
display	block	position	static
float	none	z-index	auto

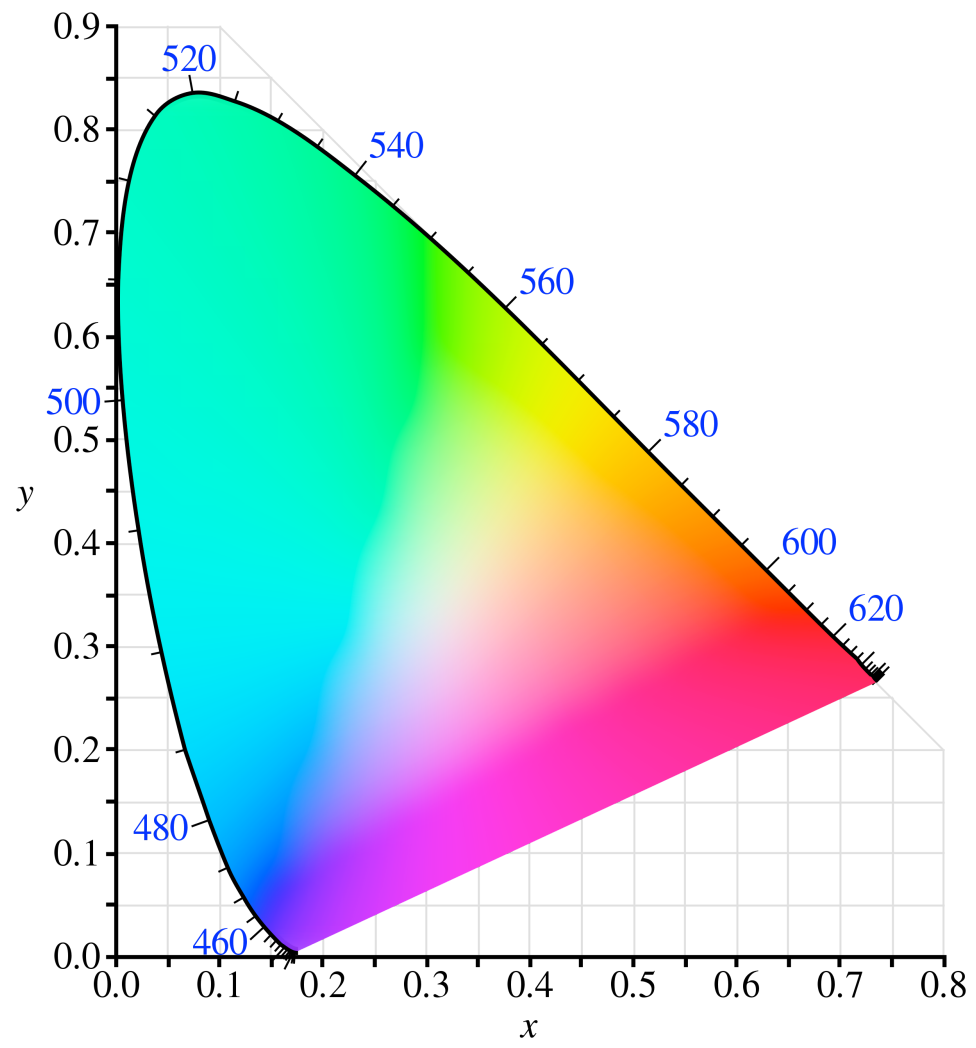


Einheiten

- Absolute Grössen: px (cm , mm , ...) -> sparsam verwenden
- Relative Grössen
 - em : Schriftgrösse des Elternelements
 - rem : Schriftgrösse des Wurzelements
 - vw , vh : viewport breite, viewport höhe

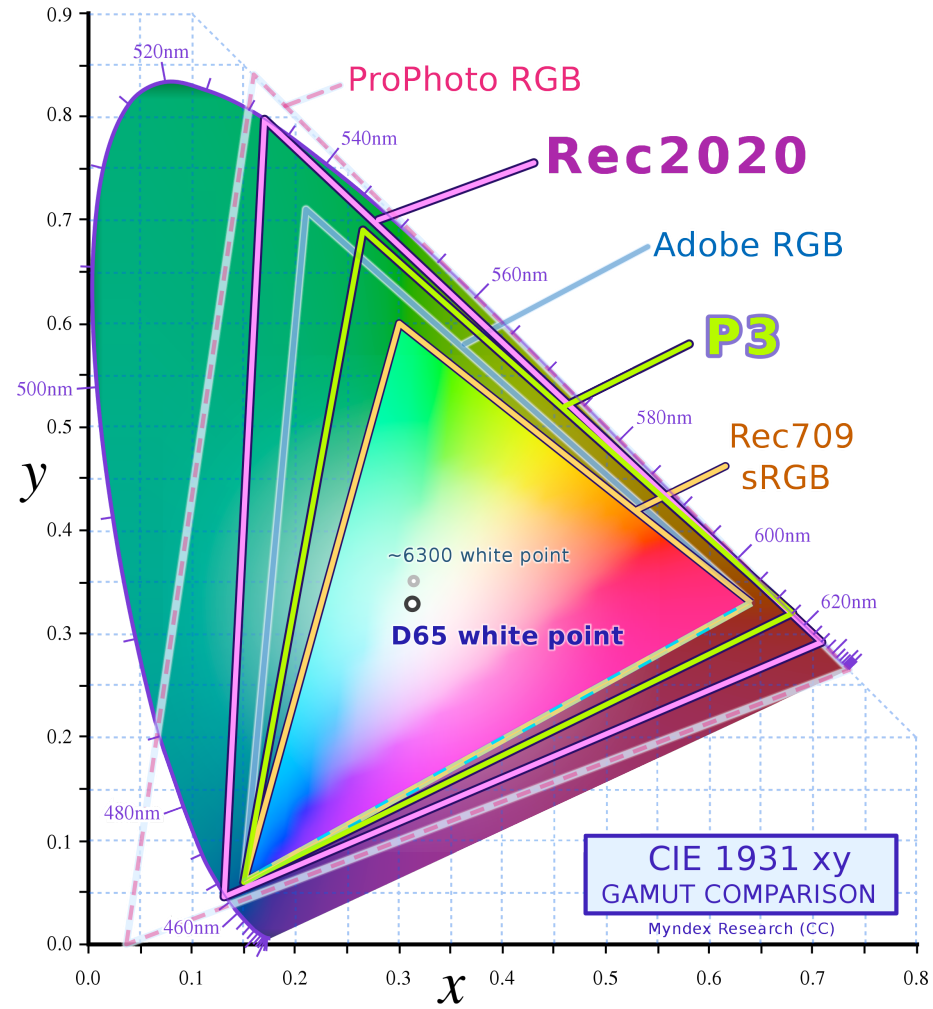


Farben



CIE 1931 Farbraum

Vergleich Farbräume



Farben in CSS

sRGB Farbraum

- Farbnamen: `color: darkblue;`
- Hex-Werte: `color: #ffa500;`
- RGBA-Werte (mit Deckkraft): `color: rgba(169, 169, 169, 0.5)`
- HSL-Werte (Hue, Saturation, Lightness): `color: hsl(60, 100, 50)`

Alle sichtbaren Farben

- LCH (Lightness Chroma Hue / Opacity): `color: lch(29.2345% 44.2 27 / 0.5)`
- Oklch: `color: oklch(40.1% 0.123 21.57)`
- CIELAB (Lightness, red-green, blue-yellow): `color: lab(29.2345% 39.3825 20.0664);`
- Oklab: `color: oklab(40.1% 0.1143 0.045);`

HSL					LCH
	50%	54%	50%	50%	
	50%	68%	50%	50%	
	50%	97%	50%	50%	
	50%	90%	50%	50%	
	50%	89%	50%	50%	
	50%	88%	50%	50%	
	50%	91%	50%	50%	
	50%	53%	50%	50%	
	50%	30%	50%	50%	
	50%	39%	50%	50%	
	50%	60%	50%	50%	
	50%	56%	50%	50%	

<https://codepen.io/web-dot-dev/pen/poZgXxy>

Webanwendungen deployen und hosten

Continuous Integration

- Kein Branching, alle Änderungen werden von allen Teammitgliedern mehrmals täglich in den Master Branch eingecheckt.
- Dieser Branch ist jederzeit lauffähig
- Dadurch werden die Releases vereinfachen
- Eine sehr hohe, automatische Testabdeckung ist zwingend

Continuous Delivery

- Ziel: Releases werden vereinfacht
- Time to market ist kürzer, neue Features sind sofort verfügbar
- Durch automatisierte deployments ist der Aufwand initial höher, anschliessend jedoch sehr klein

Begriffe

GitOps: Git als Single Source of Truth für alles was für die Applikation relevant ist.

Webanwendungen organisieren und verwalten

Webseiten interaktiv machen mit JavaScript

Web-APIs verwenden

Webanwendungen testen

Webprotokolle verwenden

HTTP

Websockets

Webformate verwenden

Rastergrafiken

Vektorgrafiken

Single-Page-Applikationen implementieren

JavaScript auf der Serverseite verwenden

Webservices implementieren

REST

Daten in Datenbanken speichern

Webanwendungen absichern

Die Performance von Webanwendungen optimieren