



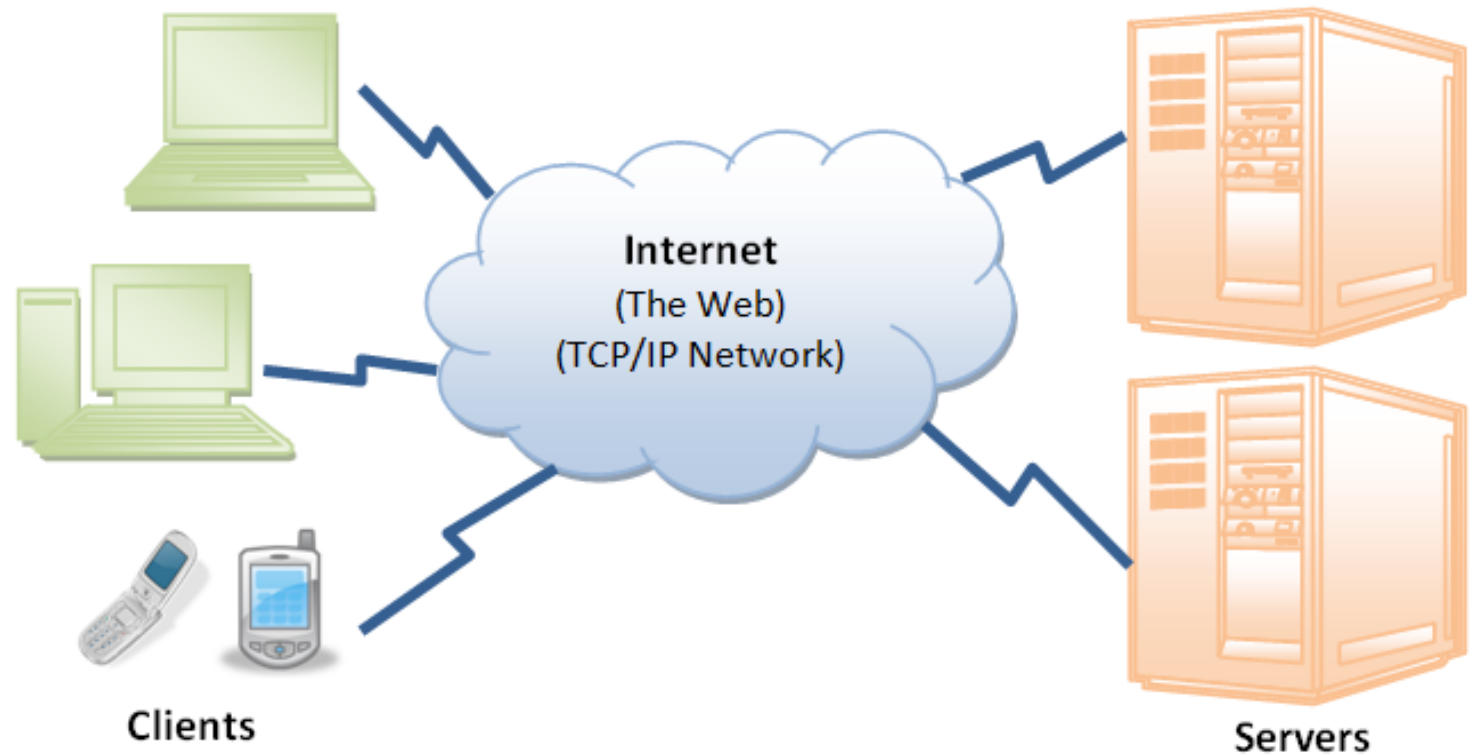
Webapplications

Design und Programmierung

Quelle: <https://images.app.goo.gl/B9g2tCoKZmqp9JPT8>

Das Internet

- Ist ein **komplexes Informationsgebilde** aus unzähligen sogenannter **Clients** (Endgeräte / Browser) **und Server** (Netzwerkrechner der seine eigenen Ressourcen anderen Clients bereitstellt)
- **Serverdienste:** Web, Email, Datenverkehr, Audio- & Video-Streaming
- **Protokolle zum Regeln dieser Dienste:** HTTP (Hyper Text Transfer Protocol), FTP (File Transfer Protocol), SMTP (Simple Mail Transfer Protocol), POP (Post Office Protocol)
 - Haben unterschiedliche Ports (z.B. FTP = 21)
`http://www.test.at:21`



Quelle: <https://goo.gl/YLRgJV>

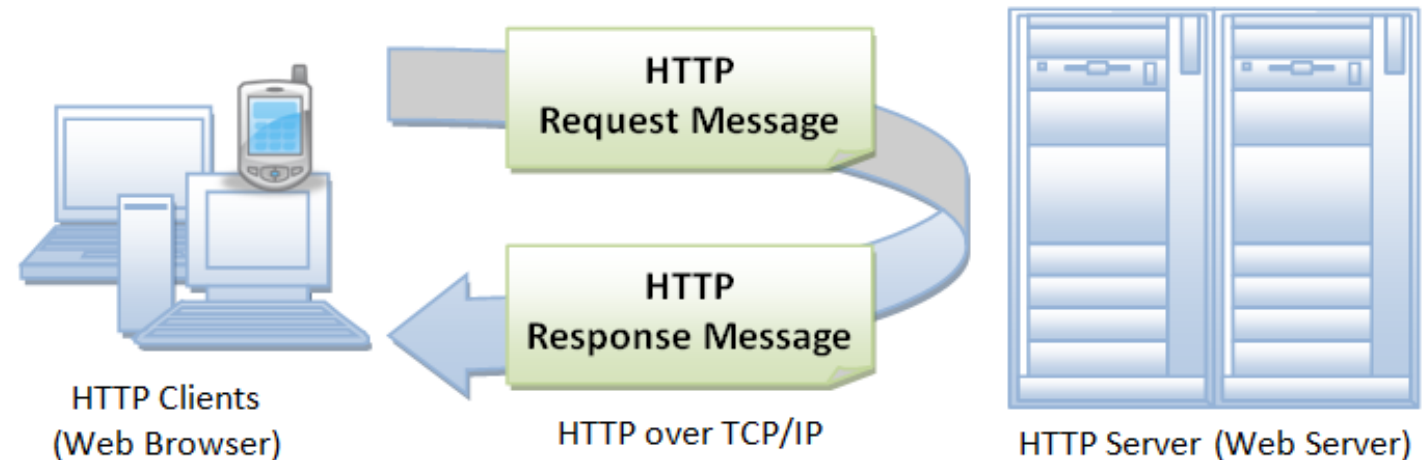
Hyper Text Transfer Protocol (HTTP)

- **Zustandsloses Protokoll**

- Alle Daten gehen bei Refresh verloren und werden neu geladen!
- Problemlösung → **Cookies**

- Ablaufplan:

1. Seitenaufruf des Clients
2. URL (Uniform Resource Locator) wird von DNS (Domain Name Server) aufgelöst (in IP-Adresse umgewandelt)
3. Mit dieser IP-Adresse (Internet Protocol) wird der Webserver kontaktiert (HTTP Request an den Webserver)
4. Webserver stellt mit entsprechendem Dienst die Webseite bereit und schickt die Websitedaten an den Client zurück (HTTP Response an den Client)



Quelle: <https://goo.gl/YLRgJV>

Die Internet- / Webseite

- Ein Dokument, welches von einem Webserver bereitgestellt werden kann
- Gängige Web-Programmiersprachen:
 - **HTML / HTML5 (Hypertext Markup Language):**
 - Textbasierte Auszeichnungssprache
 - zur Strukturierung digitaler Dokumente
 - z.B. Texte mit Hyperlinks, Bildern und anderen Inhalten
 - **CSS / CSS3 (Cascading Style Sheets):**
 - Stylesheet-Sprache für Layout und Design eines HTML-Dokuments
 - **PHP (PHP: Hypertext Preprocessor):**
 - Scriptsprache für dynamischer Webinhalte und diverse Serveranweisungen
 - **SQL (Structured Query Language):**
 - Datenbanksprache zur Definition von Datenstrukturen in relationalen Datenbanken
 - Bearbeiten (Einfügen, Verändern, Löschen) und Abfragen gespeicherter Datensätze
 - **JS (JavaScript)**
 - **AJAX (Asynchronous JavaScript and XML)**

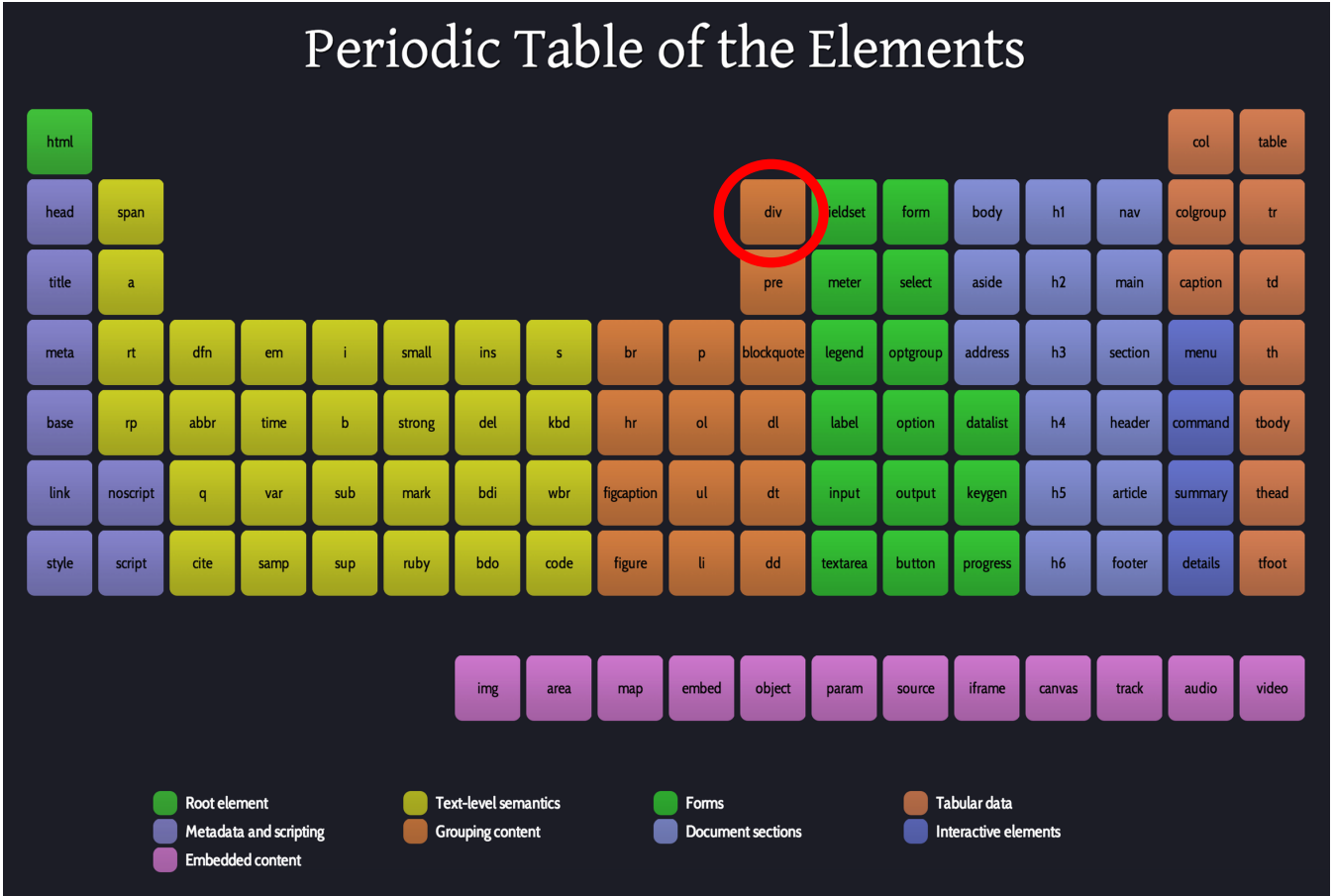
Aufbau HTML-Dokument

```
<!DOCTYPE html>
<html>
  <head>
    <title>Seitentitel</title>
    <meta charset="UTF-8">
  </head>

  <body>
    <h1>Überschrift 1</h1>
    <p>Ein Beispieltext</p>
  </body>
</html>
```

- definiert das Dokument als HTML5
 - Wurzel jedes HTML-Dokuments
 - enthält alle Meta-Informationen
 - definiert Seitentitel für Anzeige
 - definiert Zeichencodierung (ä,ö,ü,ß)
 - Endtag für das <head> Element
-
- enthält sichtbaren Seitencontent
 - definiert eine große Überschrift
 - definiert einen Platzhalter für Text
 - Endtag für das <body> Element
 - Endtag für das <html> Element

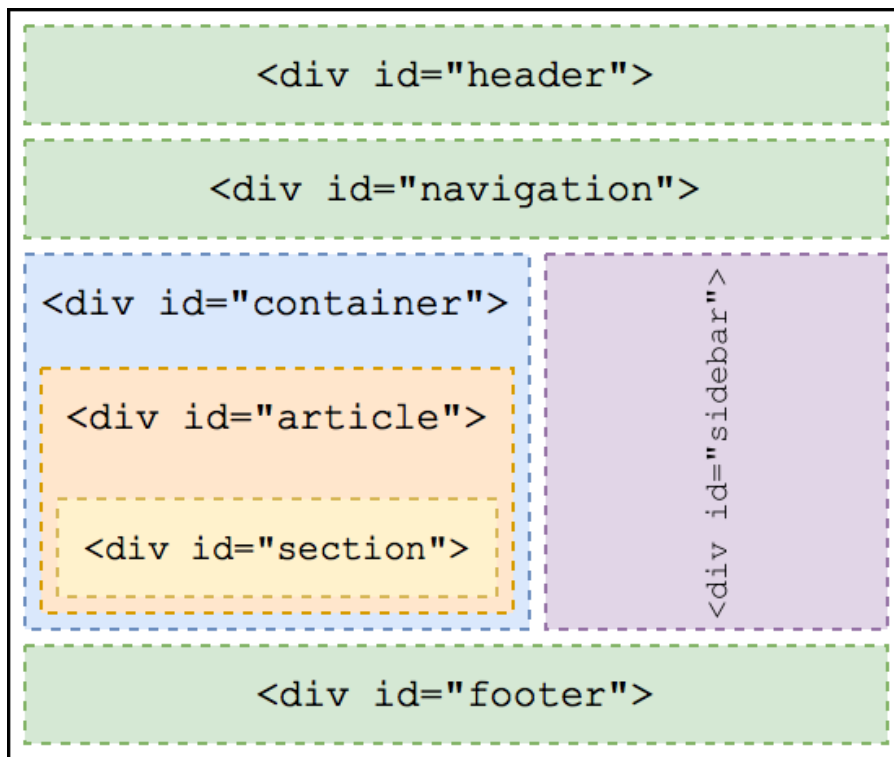
HTML5-Tags aller <Elemente>



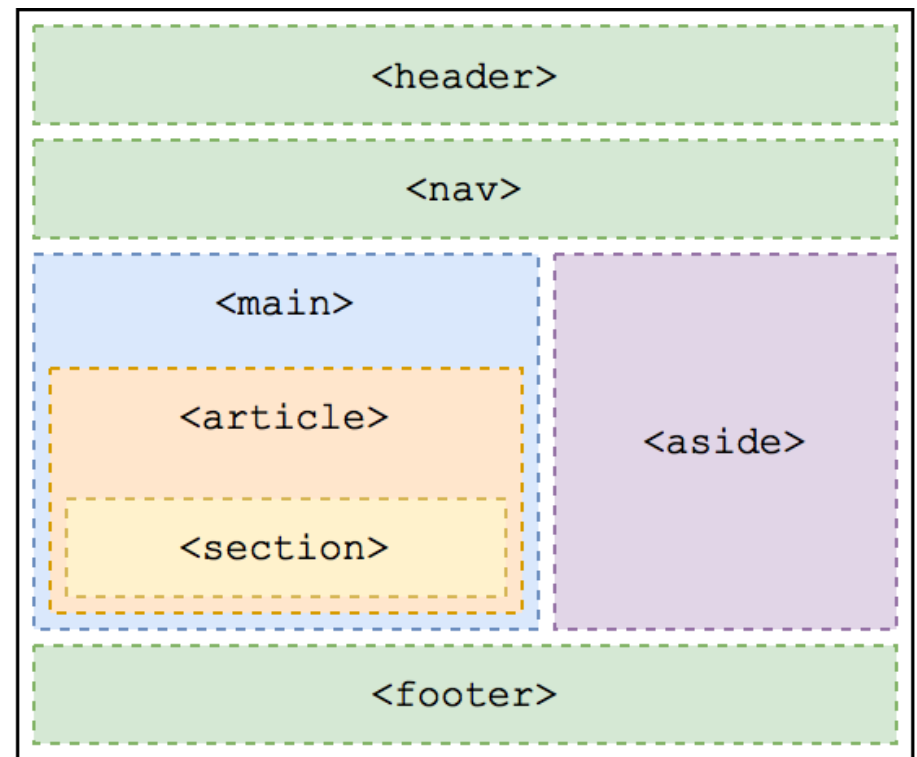
Quelle: <https://goo.gl/images/vsPLdL>

HTML → HTML5 – it's been a long way...

HTML1 – 4: (Elemente im <body>)



HTML5: (Elemente im <body>)



Quelle: Marik – Kekse ohne Salz schmecken nicht

HTML vs. CSS

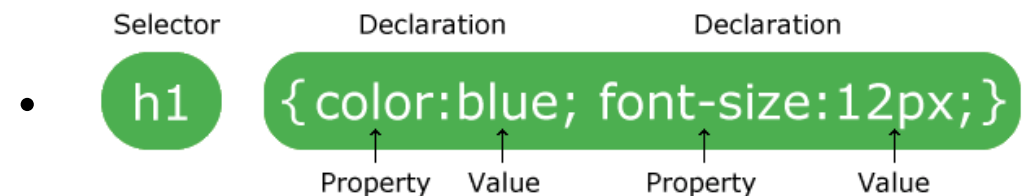
HYPertext MARKUP LANGUAGE (HTML)

- Für Inhalte (z.B. Texte, Links, Bilder, Tabellen usw.)
- Neues Element im <head>-Element:
 - `<link rel="stylesheet" type="text/css" href="layout.css">`

• **KEINE CSS BEFEHLE IM HTML-CODE!!!**

CASCADING STYLE SHEETS (CSS)

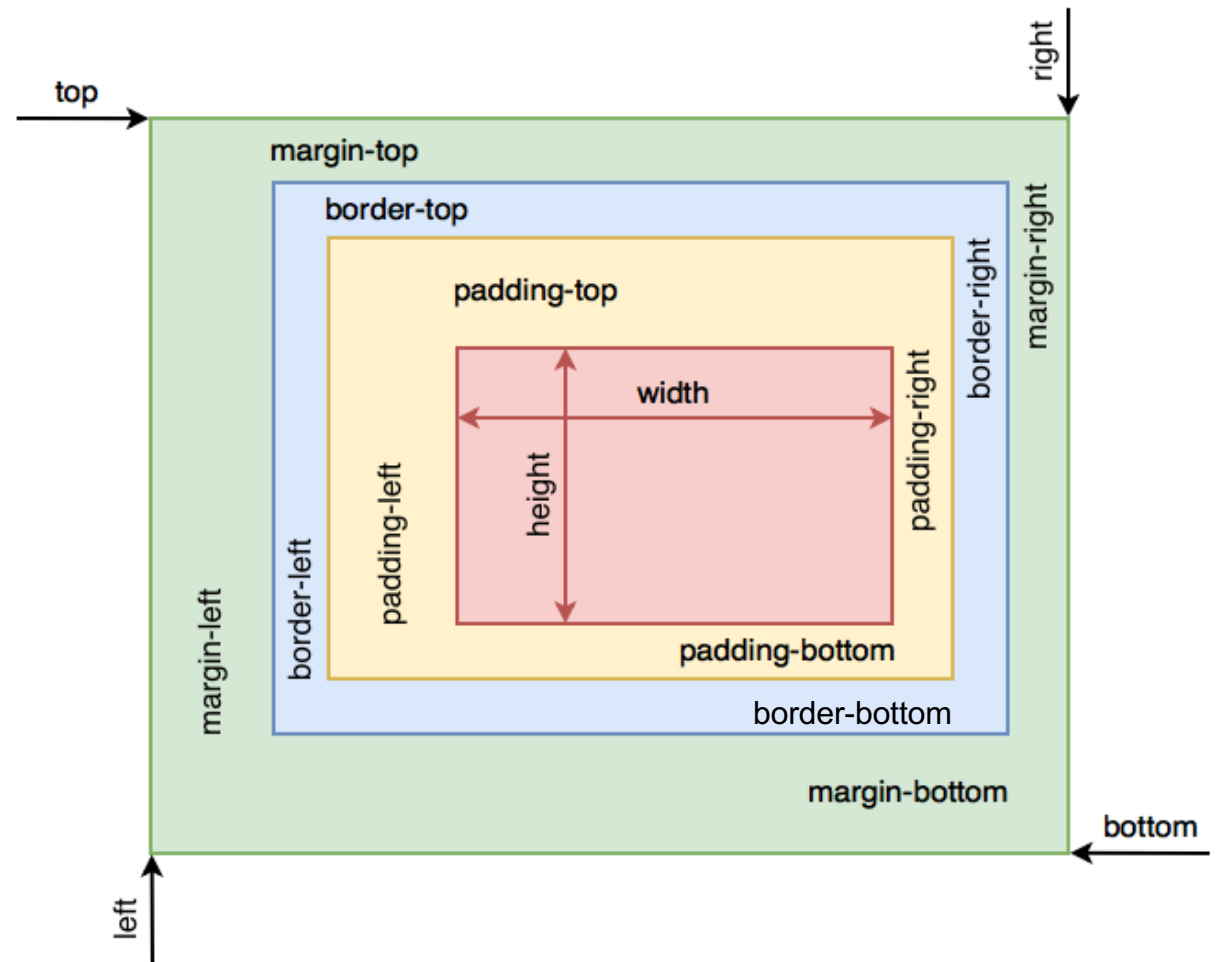
- Für Design und Layout
- Ersetzte Tabellenkonstruktionen
 - `<table>`, `<tr>`, `<td>` damals ungeheuer wichtige Tags
 - Designer besaßen bereits etliche Templates → CSS-Akzeptanz sehr gering



Quelle: <https://goo.gl/y41Xxg>

CSS3 Box Model

- top, bottom, left, right
→ Position auf der Seite
- margin, margin-top, margin-bottom, margin-left, margin-right
→ Bereich außerhalb vom Rahmen
- border, border-top, border-bottom, border-left, border-right
→ Rahmen
- padding, padding-top, padding-bottom, padding-left, padding-right
→ Bereich zwischen Inhalt und Rahmen
- width, height → Größe des Inhalts



Quelle: Marik – Kekse ohne Salz schmecken nicht

CSS Strukturmöglichkeiten (Selectors)

```
header {  
    background-image: url("image.jpg");  
}
```

- Selector für ein Element (body, header, nav, footer, usw.)
- Definiert ein Hintergrundbild im <header> (Source: image.jpg)
- Ende des Selectors

```
h1, h2 {  
    text-align: center;  
    color: #333333;  
}
```

- Selector für mehrere Elemente
- Texte mittig anordnen
- Textfarbe #333333 (Hexcode für dunkelgrau)
- Ende des Selectors

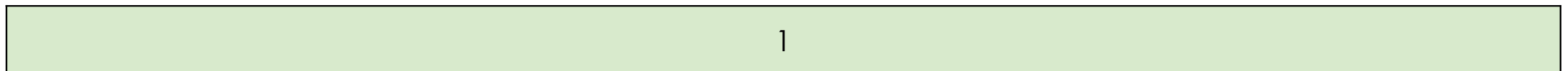
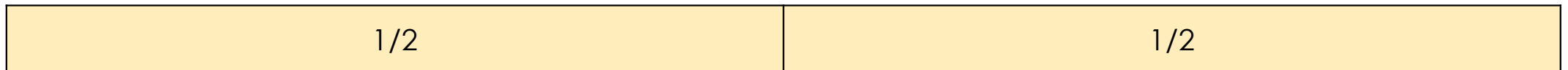
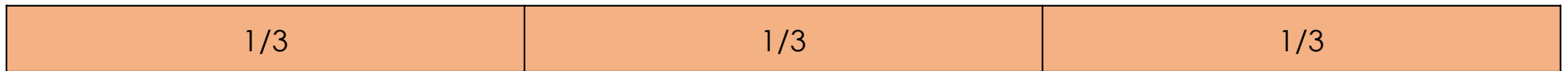
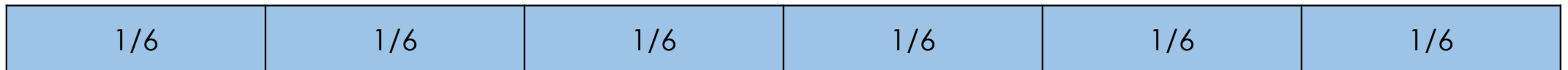
```
.profile {  
    margin-top: 1em;  
}
```

- Selector für selbstdefinierte Klassen (HTML:)
- Abstand oben auf 1em setzen (px und cm auch möglich)
- Ende des Selectors

```
#check1 {  
    padding: 0;  
    width: 100%;  
}
```

- Selector für selbstdefinierte IDs (HTML:)
- Füllbereich auf 0 setzen
- auszufüllende Breite auf 100% setzen
- Ende des Selectors

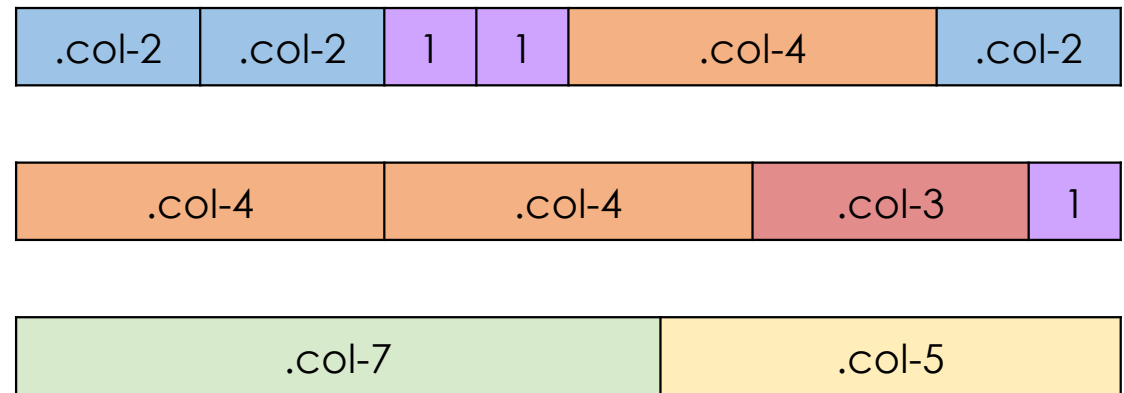
Fluid Grid



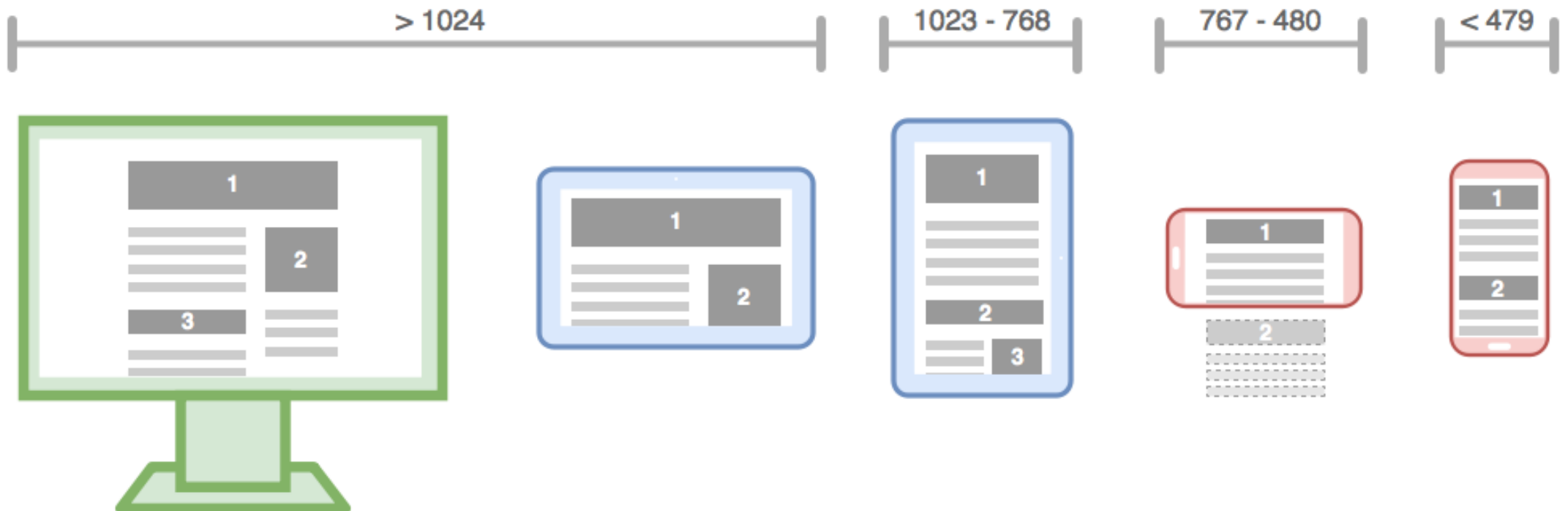
Fluid Grid

Schreibweise im CSS-Code:

```
.col-1 { width: 8.33%; }  
.col-2 { width: 16.66%; }  
.col-3 { width: 25%; }  
.col-4 { width: 33.33%; }  
.col-5 { width: 41.66%; }  
.col-6 { width: 50%; }  
.col-7 { width: 58.33%; }  
.col-8 { width: 66.66%; }  
.col-9 { width: 75%; }  
.col-10 { width: 83.33%; }  
.col-11 { width: 91.66%; }  
.col-12 { width: 100%; }
```



CSS Media Queries

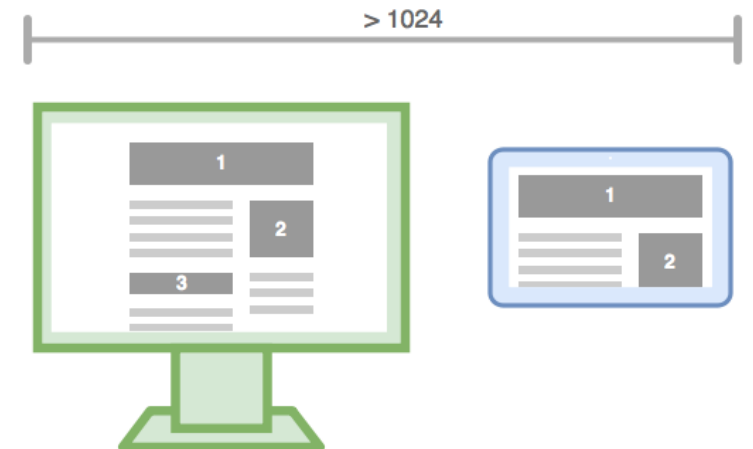
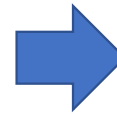


Quelle: Marik – Kekse ohne Salz schmecken nicht

CSS Media Queries

Schreibweise im CSS-Code:

```
@media screen and (min-width: 1024px) {  
  body {  
    background-color: blue;  
  }  
}
```



Quelle: Marik – Kekse ohne Salz schmecken nicht