

HTML Tag 4: Styling und globale Attribute – Grundlagen

1. Inline-Styling mit dem style-Attribut

Was ist Inline-Styling?

Inline-Styling bedeutet, dass CSS-Stile direkt in einem HTML-Tag definiert werden. Das style-Attribut wird dazu innerhalb des öffnenden Tags verwendet.

Syntax:

```
<element style="property: value;">
```

Beispiel:

```
<p style="color: red; font-weight: bold;">Dies ist ein rot gefärbter, fatter Text.</p>
```

Weitere Beispiele:

```
<h2 style="text-align: center; color: darkgreen;">Willkommen auf meiner Webseite!</h2>
```

```
<a href="#" style="text-decoration: none; color: orange;">Link ohne Unterstreichung</a>
```

Vorteile:

- Schnell & einfach für einzelne Elemente.
- Gut zum Testen von Stilen.

Nachteile:

- Schwer wartbar bei größeren Projekten.
- Stil und Inhalt sind nicht sauber getrennt.

2. Internes Stylesheet mit dem <style>-Element

Was ist ein internes Stylesheet?

CSS-Regeln werden im <head>-Bereich der HTML-Datei im <style>-Tag eingefügt.

Beispiel:

```
<head>
  <style>
    h1 {
      color: orange;
    }
    p {
      font-size: 18px;
      color: blue;
    }
  </style>
</head>
```

Vorteile:

- Trennung von Stil und Inhalt.
- Einfaches Ändern mehrerer Elemente gleichzeitig.

Weitere Beispiele:

```
<style>
  body {
    background-color: #f0f0f0;
    font-family: Arial, sans-serif;
  }
</style>
```

```
<style>
  .info-box {
    border: 2px dashed gray;
    padding: 10px;
  }
</style>
```

3. Farben und Farbformate in HTML

Farbdefinitionen:

1. Farbnamen: red, blue, green
2. Hexadezimal: #FF0000 (rot)
3. RGB: rgb(255, 0, 0) (rot)

Beispiel:

```
<h1 style="color: #FFFFFF; background-color: green;">Weißer Text auf grünem Hintergrund</h1>
<p style="color: rgb(0,0,255);">Blauer Text</p>
```

Farbcode-Beispiele:

- Rot: #FF0000
- Grün: #00FF00
- Blau: #0000FF
- Schwarz: #000000
- Weiß: #FFFFFF

Farbtools:

Nutze Online-Tools wie den HTML Color Picker von W3Schools zur Farbauswahl.

4. Schriftarten und Textgestaltung

Wichtige Eigenschaften:

- **font-family:** Schriftart (z. B. Arial, Verdana)
- **font-size:** Größe (z. B. 16px, 1.2em)
- **font-weight:** Fettgrad (z. B. normal, bold)
- **font-style:** Stil (normal, italic)

Beispiel (Inline):

```
<p style="font-family: monospace; font-size: 20px; font-weight: 900; font-style: normal;">
Das ist ein Absatz mit Stil.
</p>
```

Beispiel (Internes Stylesheet):

```
<style>
p {
  font-family: Georgia;
  font-size: 18px;
  font-style: italic;
  font-weight: lighter;
}
</style>
```

5. Die ``- und `<div>`-Elemente

`` – Inline-Container für Textteile

Verwendet für das Styling einzelner Wörter oder kurzer Textabschnitte.

Beispiel:

```
<p>Dieses Wort ist <span style="color: red;">rot</span>.</p>
```

`<div>` – Block-Container für Layout-Gruppierung

Ideal zum Gruppieren ganzer Abschnitte oder Elementgruppen.

Beispiel:

```
<div style="border: 1px solid black; padding: 10px;">
  <p>Text innerhalb eines Rahmens</p>
</div>
```

6. Globale Attribute: id, class, title

id – Einzigartige Kennung pro Element

```
<p id="intro">Willkommen!</p>
```

CSS:

```
#intro {
  color: red;
}
```

class – Gemeinsames Styling mehrerer Elemente

```
<p class="highlight">Text 1</p>
<p class="highlight">Text 2</p>
```

CSS:

```
.highlight {  
  background-color: yellow;  
  font-weight: bold;  
}
```

title – Tooltip bei Mouseover

```
<button title="Hier klicken!">Mehr erfahren</button>
```

7. Weitere globale Attribute

Attribut	Funktion
accesskey	Tastenkombination für Zugriff
lang	Sprache des Inhalts
hidden	Element verstecken
dir	Textausrichtung (ltr, rtl)
translate	Automatische Übersetzung erlauben
onclick	Skript beim Klicken ausführen

8. Event-Attribute

Ermöglichen Interaktivität durch Benutzeraktionen.

Beispiele:

```
<button onclick="alert('Hallo Welt!')">Klick mich</button>
```

Weitere Event-Attribute:

- onmouseover
- onmouseout
- onkeydown
- onkeyup

9. Data-Attribute (data-*)

Speichern von benutzerdefinierten Daten in HTML.

Beispiel:

```
<li data-rezept-id="001">  
  <h2>Lasagne</h2>  
  <button class="save">Speichern</button>  
</li>
```

JavaScript (später):

```
const id = document.querySelector('li').dataset.rezeptId;
```