

# HTML Tag 3: `<img>`-Element

---

Das `<img>`-Element wird in HTML verwendet, um Bilder in eine Webseite einzubinden. Es ist ein selbstschließendes Tag und unterstützt viele Attribute, um die Darstellung und Funktionalität der Bilder zu steuern.

[mdn webdocs - Element](#)

## Standard-Attribute:

- **src:** Definiert die Quelle des Bildes, entweder durch eine relative oder absolute URL.  
*Beispiel:* `src="images/logo.png"`
- **alt:** Ein alternativer Text, der angezeigt wird, falls das Bild nicht geladen werden kann. Dieser Text ist auch wichtig für Screenreader, um Webseiten barrierefrei zu gestalten.  
*Beispiel:* `alt="Logo des Unternehmens"`
- **width:** Die Breite des Bildes in Pixeln oder Prozent.  
*Beispiel:* `width="300"`
- **height:** Die Höhe des Bildes in Pixeln oder Prozent.  
*Beispiel:* `height="200"`
- **title:** Zeigt einen Tooltip an, wenn der Mauszeiger über dem Bild schwebt.  
*Beispiel:* `title="Logo des Unternehmens"`

---

## Erweiterte Attribute:

- **srcset:** Erlaubt die Angabe mehrerer Bildquellen, um abhängig von der Bildschirmauflösung oder Größe das passende Bild zu laden. Dies ist essenziell für responsive Designs.  
*Beispiel:*

```

```

Hier bedeutet `1024w`, dass die Datei `large.jpg` bei Viewports verwendet wird, die mindestens 1024px breit sind.

- **sizes:** Gibt an, welche Bildgröße vom Browser bei bestimmten Viewport-Breiten verwendet werden soll. Dies funktioniert zusammen mit `srcset`.  
*Beispiel:*

```

```

In diesem Beispiel wird das Bild mit einer Breite von 480px geladen, wenn der Viewport kleiner als 600px ist, und mit 800px, wenn er größer ist.

*Beispiel für mit srcset und sizes*

Angenommen, du hast drei Bilder in unterschiedlichen Auflösungen:

- hero-small.jpg (320px breit)
- hero-medium.jpg (640px breit)
- hero-large.jpg (1024px breit)

Dann sieht der HTML-Code so aus:

```

```

Erklärung:

- **srcset**: Gibt an, welche Datei bei welcher Mindestbreite geladen wird (1024w = 1024px Breite).
- **sizes**: Sagt dem Browser, wie groß das Bild voraussichtlich auf dem Bildschirm dargestellt wird.
- Wenn das Viewport  $\leq 600\text{px}$  breit ist  $\rightarrow$  Bild soll 100% der Breite einnehmen (100vw)
- Wenn Viewport  $\leq 1024\text{px}$   $\rightarrow$  50% der Breite
- Sonst  $\rightarrow$  33% der Breite
- **src**: Fallback-Bild, falls srcset nicht unterstützt wird.
- **loading="lazy"**: Lädt das Bild erst, wenn es im Sichtbereich erscheint (Performance-Vorteil).
- **decoding="async"**: Bild wird asynchron dekodiert (schnellere Ladezeiten).

## Responsive Images:

Bilder sollten für verschiedene Bildschirmgrößen optimiert sein, insbesondere für mobile Geräte. Das **srcset**-Attribut bietet die Möglichkeit, mehrere Bildgrößen bereitzustellen und so sicherzustellen, dass für jedes Gerät das optimale Bild geladen wird.

Beispiel für ein responsives Bild:

```

```

## Weitere Attribute:

- **loading**: Gibt an, ob das Bild sofort oder nur bei Bedarf geladen werden soll. Das lazy-Laden kann die Performance verbessern.  
*Beispiel: **loading="lazy"***
- **decoding**: Bestimmt, wie das Bild im Dokument dekodiert wird. Der Wert **async** sorgt dafür, dass die Seite schneller geladen wird, auch wenn das Bild noch nicht vollständig geladen ist.  
*Beispiel: **decoding="async"***
- **usemap**: Dieses Attribut wird verwendet, um das Bild mit einer clientseitigen Image Map zu verknüpfen, die klickbare Bereiche auf dem Bild definiert.  
*Beispiel: **usemap="#image-map"***
- **crossorigin**: Gibt an, ob CORS-Anfragen beim Laden des Bildes verwendet werden sollen. Dies ist wichtig, wenn ein Bild von einer anderen Domain geladen wird.  
*Beispiel: **crossorigin="anonymous"***

## Beispiel für ein umfassendes **<img>**-Element

```

```