

## Mediale Elemente in HTML5 – Bilder

Bilddateien können für die Verwendung im World Wide Web entweder als Pixelgrafik oder als Vektorgrafik zur Verfügung stehen. Bei einer Pixelgrafik wird für jeden einzelnen Bildpunkt (Pixel) ein Farbwert in einer Datei gespeichert.

Dateiformate für Pixelgrafiken sind: GIF, TIFF, PNG und JPEG. Pixelgrafiken im TIFF-Format können nicht von HTML dargestellt werden.

Eine Vektorgrafik ist eine Computergrafik, die sich aus einfachen Formen und Flächen zusammensetzt. Diese Formen können mathematisch berechnet werden. Formate für Vektorgrafiken sind z. B.: EPS und SVG. Der Einsatz von Vektorgrafiken nicht von allen Browsern unterstützt.

### Bilder per HTML5 in eine Webseite einbinden

|                     |   |   |
|---------------------|---|---|
| <b>img-Element</b>  | <i>img</i> bedeutet <i>image</i> (Bild). Das <i>img</i> -Element benötigt keinen End-Tag.<br>Der <i>img</i> -Tag benötigt immer das Attribut <i>src</i> .   | <code>&lt;img ...&gt;</code>                          |
| <b>src-Attribut</b> | Das Attribut <i>src</i> steht für <i>source</i> (Quelle) und gibt dem Browser die Information, nach welcher Grafik (Dateinamen und -format) er an welchem Ort (Dateipfad) suchen soll.<br>Es ist sehr wichtig, dass stets der korrekte Pfad zu den Dateien angegeben ist, da sonst der Browser die Datei nicht darstellt.   | <code>src="kaffee.jpg"</code>                         |
| <b>alt-Attribut</b> | <i>alt</i> bedeutet <i>alternative text</i> . Er enthält eine kurze Beschreibung des dargestellten Bildes.<br>Dieser wird dargestellt, wenn der Browser die Bilddatei nicht finden kann oder ein Nutzer sich den Text von einem Screenreader vorlesen lässt.  | <code>alt="Tasse Kaffee"</code>                       |
| <b>width/height</b> | Mit diesen beiden Attributen kann das <i>img</i> -Element noch in der Größe beeinflusst werden.<br>Es müssen nicht immer beide Werte verwendet werden.  | <code>width="200"</code><br><code>height="300"</code> |
| <b>Aufgabe 1:</b>   | <ul style="list-style-type: none"> <li>» Erstellen Sie eine neue HTML-Datei mit dem Namen <i>medien.html</i>.</li> <li>» Verknüpfen Sie die Bilder <i>kaffee_1.jpg</i>, <i>kaffee_2.gif</i>, <i>kaffee_3.png</i>, <i>mut.tiff</i>, <i>christmaspiktogramm.svg</i> und <i>klettergriffe.eps</i>.</li> <li>» Passen Sie die Größe der Bilder an und vergeben Sie einen sinnvollen Beschreibungstext.</li> </ul> |   |
| <b>Aufgabe 2:</b>   | <ul style="list-style-type: none"> <li>» Informieren Sie sich über die HTML-Elemente <i>figure</i> und <i>figcaption</i>.</li> <li>» Verknüpfen Sie ein Bild Ihrer Wahl mit dem HTML-Element <i>figure</i> und fügen Sie eine Bildunterschrift ein.</li> </ul>  |   |

## Mediale Elemente aus einem anderen Ordner einbinden

Will man in eine HTML5-Datei mediale Elemente wie Bilder, Videos oder Audiodateien einbinden, muss angegeben werden, wo sich diese befinden. Man bezieht sich bei dieser Angabe auf den Speicherort, den sogenannten Pfad der Datei. Pfadangaben können relativ und absolut sein.

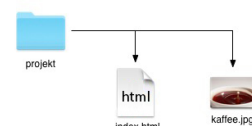
### Relative Pfadnamen verwenden

Eine Webseite besteht in der Regel aus vielen unterschiedlichen Dateien z. B.: HTML5-Dateien, Grafiken, Videos, Audiodateien usw. Diese sind normalerweise in einem Projektordner auf dem gleichen Server gespeichert. Bei umfangreichen Projekten ist sinnvoll, sich ein Datenmanagement anzueignen. Dies hat den Vorteil, dass das Projekt übersichtlich bleibt und Dateien schnell gefunden werden.

Für Bilder und Grafiken beispielsweise existiert meistens ein eigenes Verzeichnis mit dem Namen *images* oder *bilder*. Wird eine Grafik in eine HTML5-Datei mit dem *img*-Element eingebunden, muss das *src*-Attribut auf den richtigen Ordner verweisen, in dem die Grafik gespeichert ist. Dazu werden relative Pfadnamen verwendet.

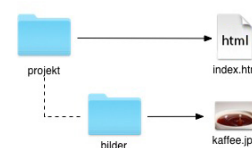
- » Das Bild ist im gleichen Ordner wie die HTML5-Datei gespeichert.

``



- » Das Bild ist im Ordner *bilder*, eine Ebene unter dem Ordner *projekt*, gespeichert.

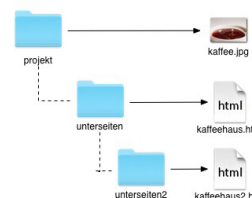
``



- » Das Bild ist im Ordner *projekte* gespeichert, die HTML5-Datei in einem Unterordner z. B.: *unterseiten*

Um eine Ebene in der Ordnerstruktur nach oben zu gelangen, müssen vor dem Dateinamen zwei Punkte und ein Schrägstrich notiert werden. Mit dieser Methode können auch mehrere Ebenen überwunden werden. Zwei Punkte und ein Schrägstrich beziehen sich immer auf eine Hierarchieebene.

`` bzw. ``

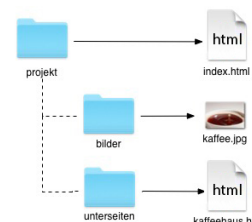


Quelle: AH

- » Das Bild ist im Ordner *bilder* gespeichert und die HTML5-Datei im Ordner *unterseiten*.

Durch die Angabe von `../` wird der nächsthöhere Ordner erreicht (hier *projekt*). Durch die Angabe des gewünschten Unterordners und des Dateinamens kann auf die gewünschte Datei zugegriffen werden.

``



Quelle: AH

### Absolute Pfadnamen verwenden

Absolute Pfadnamen werden verwendet, wenn Inhalte z. B.: Bilder von anderen Webseiten eingebettet werden sollen. Auch Verweise zu anderen Webseiten sind in der Regel absolute Pfade.

- » Verweis auf eine externe Webseite z. B.: <http://www.tagesschau.de/>
- » Einbinden von Bildern aus einer anderen Webseite: <http://www.kaffeehaus.de/bilder/kaffeehaus1.jpg>

## Mediale Elemente in HTML5 – Audio und Video

Spricht man von medialen Inhalten im Internet, meint man in der Regel Bilder, Videos, Animationen und Musik. Bisher war von diesen medialen Inhalten nur das Einbinden von Bildern problemlos möglich. Audio- und Videoinhalte konnten in Webseiten nur mit der Hilfe von Zusatztechnologien wie bspw. Adobe Flash bzw. dem Adobe Flash Player realisiert werden. Diese mussten allerdings als extra Plugin im Browser installiert werden. Erst seit HTML5 gibt es die Elemente *Video* und *Audio*, um direkt mediale Dateien in Webseiten zu integrieren.

### Audiodateien einbetten

Mit dem Element `<audio>` werden Audiodateien in HTML5 eingebunden. Dieses Element benötigt zusätzlich weitere Attribute.

Es gibt zwei verschiedene Möglichkeiten, den Pfad zu Audiodateien anzugeben:

- » Mit dem `src`-Attribut werden Audiodateien, ähnlich wie Bilddateien, verknüpft. Hier muss der Dateiname und der Pfad zu der gewünschten Datei angegeben werden.
- » Mit dem `source`-Element können verschiedene Audio-Formate angegeben werden. Diese Variante ist vorteilhaft, da noch nicht alle Formate von den Browsern unterstützt werden.

```
<audio src="musik.mp3" controls>
Alternativer Inhalt für ältere Browser
</audio>
<audio controls>
<source src="musik.mp3" type="audio/mpeg">
<source src="musik.wav" type="audio/wav">
Alternativer Inhalt für ältere Browser
</audio>
```

Innerhalb des `audio`-Elements sollte auf jeden Fall auch ein alternativer Inhalt angegeben werden, falls ein älterer Browser keines der angegebenen Formate unterstützt.

### Video

Videodateien können mit dem `video`-Element eingebunden werden. Ein Problem stellen jedoch unterschiedliche Videoformate dar, die nicht von allen Browsern unterstützt werden.

```
<video controls>
<source src="video.mp4" type="video/mp4">
<source src="video.ogv" type="video/ogg">
Alternativer Inhalt für ältere Browser
</video>
```

### Attribute für Audio und Video

Für die beiden HTML5-Elemente `<audio>` und `<video>` gibt es mehrere Attribute, mit denen das Verhalten der Wiedergabe festgelegt werden kann:

- » `loop` sorgt für eine Endlosschleife bei der Wiedergabe.
- » `autoplay` startet das Element beim Aufruf der Seite.
- » `controls` zeigt die Kontrollleiste der Wiedergabe an.
- » `preload="..."` legt fest, ob die Audio- bzw. Videodatei beim Aufrufen der Seite bereits geladen werden soll. Die möglichen Werte hierfür sind `none` (nicht automatisch laden), `metadata` (nur Metadaten wie Länge und Größe der Datei werden geladen) und `auto` (der Browser entscheidet selbst).

### Aufgabe:

- » Erstellen Sie eine neue HTML5-Datei mit dem Namen `multimedia.html`.
- » Geben Sie der Datei einen aussagekräftigen Titel und fügen Sie Blindtext ein. Diesen können Sie nach Belieben formatieren.
- » Binden Sie die zur Verfügung stehenden Video- und Audiodateien in die HTML5-Datei ein.
- » Ändern Sie die Größe der Videos mit dem Attribut `width` auf einen Wert von 720.

## Lehrerinformation und Arbeitsmaterial

Für die Aufgaben werden verschiedene mediale Elemente benötigt:

- » Bilder/Grafiken in den Dateiformaten .jpg, .gif, .png, .tiff, .eps und .svg.
- » Video- und Audiodateien in den Formaten .mp3, .wav, .mp4 und .wmv.
- » Es können eigene Video- und Audiodateien verwendet werden.

Auf den folgenden Webseiten können Audio- und Videodateien heruntergeladen werden:

- » <http://thecliparchive.com/>
- » [www.free-video-footage.com/](http://www.free-video-footage.com/)
- » [www.youtube.com/audiolibrary/music](http://www.youtube.com/audiolibrary/music)

```
1 <!doctype html>
2 <html lang="de">
3 <head>
4 <meta charset="utf-8">
5 <title>Verschiedene Medien einbinden!</title>
6
7 <style type="text/css">
8 h1 {color:#FF0000; font-size:48px;}
9 </style>
10
11 </head>
12 <body>
13
14 
15 
16 
17 
18 
19 
20
21
22 <figure>
23 
24 <figcaption>Leckerer Kaffee am Meer und nen Kecks dazu!</figcaption>
25 </figure>
26
27
28
29 <video width="720" controls autoplay>
30 <source src="Baking Cookies H264.mp4" type="video/mp4" />
31 <source src="Horizontal Match Strike WMV.wmv" type="video/wmv" />
32 <p>Dieser Browser unterstützt HTML5 noch nicht</p>
33 </video>
34
35 <video width="720" controls autoplay>
36 <source src="Horizontal Match Strike WMV.wmv" type="video/wmv" />
37 <p>Dieser Browser unterstützt HTML5 noch nicht</p>
38 </video>
39
40 <audio src="By_the_Pool.mp3" preload="auto" autoplay controls>
41 <p>Dieser Browser unterstützt HTML5 noch nicht</p>
42 </audio>
43
44 </body>
45 </html>
```

Musterlösung Aufgabe Bilder, Audio, Video | Quelle: AH

## Lehrerinformation und Arbeitsmaterial



Quelle: AH



Quelle: AH