# Kapitel 6 - Teil 2

## *Bilder und Videos*

Inhalte in diesem Kapitel

Das richtige Bildformat wählen

Bilder auf Webseiten einbinden

Videos und Audio einfügen

# Bilder

**Damit Bilder gut funktionieren** -> zwei wesentliche Voraussetzungen

* + Müssen auf allen Betriebssystemen funktionieren
  + Möglichst klein sein
  + Je mehr KB desto länger die Ladezeit

**Raster versus Vektor**

* + Rastergrafiken (Pixelbilder oder Bitmapgrafiken)
    - Rasterförmige Anordnung von farbingen Pixel (Bildpunkten)
    - Bei Vergrößerung (Skalierung) -> Verpixelung
    - Formate: JPEG, GIF, PNG, WebP
  + Vektorgrafiken
    - Sind durch Pfade erstellt, die mathematisch definiert sind
    - Beliebige Skalierung, bleiben immer scharf
    - Formate für Web: SVG   
      weitere Formate (nicht für Web): TIFF, BMP oder EPS

**Eine Frage des Bildes**

* + JPG
    - Abk.: Joint Photographic Experts Group nach dem Gremium, dass das Format entwickelt hat
    - JPG/JPEG gut bei feinen Farbverläufen und Schattierungen
    - Weniger gut: scharfen Kanten, Farbflächen
    - Unscharf bei zu starker Komprimierung
    - -> tinypng.com/
  + PNG
    - PNG 24 und PNG 8
      * PNG 24: verlustfreie Komprimierung und unterstützt True Color (16 Mio Farben)
      * PNG 8: 8-Bit-Grafiken (256 Farben)
    - Transparente Bereiche möglich
  + GIF
    - Graphic Interchange Format
    - War das erste Format im Web
    - 256 Farben
    - Komprimierung verlustfrei
    - Können animiert werden
  + WebP
    - Neues Format
    - Von Google definiert
    - Fast alle Vorteile der anderen Formate bei kleinerer Dateigröße
  + SVG
    - Scalable Vector Graphics
    - Format für Cektorgrafiken
    - Problemlos skalieren
    - Logos & Icons geeignet

**Das richtige Format in der richtigen Situation**

* + Fotos: JPG
  + Icons und flächige Grafiken: PNG oder SVG
  + Kombi von Grafik u Foto: PNG oder WebP
  + Bei Transparenz: PNG

**Bilder einbinden**

<img src=“bild.jpg“ alt=“Landschaft“>

* + Img-Element = leeres Element
  + Beinhaltet weder Text noch andere Elemente
  + src = Source/Quelle  
    alt = kurze Beschreibung des Bildes
    - wenn Bild nicht geladen wird
    - Barrierefreiheit wichtig
    - Suchmaschinenoptimierung wichtig
  + Optional: title:
    - Tooltip bei Bild

<img src=“bild.jpg“ alt=“Landschaft“ title=“blabla“>

**Beschriftung**

* + figcation-Element
  + Zusammenhang zum Bild bleibt gewahrt

<figure>

<img src=“bild.jpg“ alt=““>

<figcaption>Das ist die Landschaft von</figcaption>

</figure>

* + Kein alt da Information in der Beschriftung steht

**Breiten und Höhen**

<img src=“…“ alt=“…“ width=“400“ heigth=“200“>

* + Unbedingt mit CSS machen
  + Dient dazu, den Platz für das Bild frei zu halten bis das Bild geladen ist = Seite springt sonst

**SVG Bilder**

* + Ist ein XML-basiertes Format für Bilder  
    blauer Kreis:

<svg xmlns=“<http://www.w3.org/2000/svg>“ viewBox=“0 0 120 120“>

<circle r=“60“ cx=“60“ cy=“60“ fill=“blue“/>

</svg>

* + svg ist Wurzelelement mit Namensraumangabe xmlns und viewBox für Koordinatensystem
  + circle erzeugt Kreis  
    r legt Radius  
    cx und cy bestimmten Position des Kreismittelpunkts  
    fill ist Füllfarbe

**SVG einbinden**

<img src=“kreis.svg“ height=“200“ width=“200“ alt=“blauer Kreis“>

**Favicon**

* + Im Browsertab angezeigt
  + Wenn auf oberster Ebene in Webordner abgespeichert und favicon.ico heißt = automatisch eingebunden  
    ansonsten:

<link rel=“shortcut icon“ type=“image/x-icon“ href=“favicon.ico“>

* + Für Touch-Icons

<link rel=“shortcut icon“ type=“image/x-icon“ href=“favicon.ico“>

<link rel=“icon“ type=“image/png“ href=“favicon.png“ size=“32x32“>

<link rel=“icon“ type=“image/png“ href=“favicon.png“ size=“96x96“>

<link rel=“apple-touch-icon“ type=“image/png“ href=“apple-touch-icon.png“ size=“180x180“>

* + Für Erstellung gibt es Generatoren

# Videos

**Kodierung und Container**

* + Man unterscheidet bei Videos die Kodierung und das Containerformat
    - Video-Codec  
      * wie das Video kodiert ist
      * Typische Formate: mp3, h.264, Vorbis, Theora, AAC und VP8
    - Container-Format
      * Legt eine Datenstruktur fest, mit der einzelne Datenströme zu einem Datenstrom zusammengeführt werden
      * Bei Audio-/Video können Containerformate zumindest einen Audio- und Videostream enthalten
      * Manche auch Untertitel und Menüstruktur. Etc
  + Kombinationen im Web
    - MPEG-4(H.264)  
      gute Unterstützung bis auf Opera Mini
    - WebM (VP8 oder VP9)  
      gute Unterstützung bis auf Safari und IE
    - Ogg (Theora)  
      unterstützt von Edge ab 17, Firefox und Chrome  
      Safari und Android-Browser nicht

**Video** **einbinden**

<video src=“beispiel.mp4“ width=“400“ controls>

Leider unterstützt ihr Browser das video-Element nicht. Sie können das <a href=“beispiel.mp4“>Video auch herunterladen</a>

</video>

* + Pfadangabe über src-Attribut
  + controls: Standardschaltfläche zur Bedieung des Videos
  + autoplay: Video beginnt direkt abzuspielen
    - normalfall nicht benutzen
  + poster: Bild angeben, das angezeigt wird, bis das Video abgespielt wird
  + loop: Endlosschleife
  + muted: stummschalten
  + preload: welcher Inhalt soll geladen werden, bevor das Video startet

<video src=“beispiel.mp4“ width=“400“ controls muted loop poster=“katze.jpg“>

Leider unterstützt ihr Browser das video-Element nicht. Sie können das <a href=“beispiel.mp4“>Video auch herunterladen</a>

</video>

* + Video in weiteren Formaten zur Verfügung stellen

<video width=“400“ controls>

<source src=“beispiel.webm“ type=“video/webm“>

<source src=“beispiel.mp4“ type=“video/mp4“>

</video>

**Videos von YouTube und anderen Diensten einbinden**

* + YouTube: Link Teilen -> Einbetten = iFrame Code

**iFrames**

* + Inline-Frame
  + HTML-Datei in eine andere Webseite einzubinden
  + Wird bei YouTube Videos benutzt

**Audio** **Element**

* + Funktioniert prinzipiell genauso wie das video-Element

<audio src=“audiotest.ogg“ autoplay> Ihr Browser unterstützt das <code>Audio</code>-Element nicht.</audio>

* + Oder

<audio controls=“controls“>

Ihr Browser unterstützt das <code>Audio</code>-Element nicht.

<source src=“beispiel.wav“ type=“audio/wav“>

</audio>

**Mehrere Objekte**

* + Weitere Elemente für Integration externer Inhalte
    - object
      * allgemeines Element um externe Inhalte einzubinden
      * früher für Flash-Videos genutzt
      * SVG-Dateien
    - embed
      * Einbetten von externen Inhalten zB Plugins
      * Wirst es im Normalfall nicht benötigen
    - canvas
      * Zeichenfläche erzeugt auf die mit JavaScript Bilder gezeichnet und animiert werden können