

2 Was ist HTML?

Worum es geht

HTML ist die «Sprache des Webs». Wie sieht es aus? Was ist die Idee dahinter? Wie ist es aufgebaut? Weshalb benötigen wir HTML, um Webseiten zu erstellen? Fragen, die der folgende Text klären soll.

Du wirst nach dem Durcharbeiten dieses Kapitels folgende Techniken beherrschen:

- Konzept und Funktionsweise von HTML
- Das Tag-Konzept
- Logik und Aufbau von HTML-Dokumenten
- Der Einsatz von Attributen
- Faustregeln für die stilistisch saubere Programmierung

Material und Referenzen

Solltest du an einem Punkt nicht mehr weiterkommen, benutze das Mozilla Developer Network (<https://developer.mozilla.org/de/docs/Web/HTML/HTML5>) oder Wikipedia (<http://de.wikipedia.org>) als weiterführende Referenzen!

HTML: HyperTextMarkupLanguage - Zwei Konzepte in einem

Die Abkürzung HTML steht für HyperTextMarkupLanguage und verbindet die beiden folgenden Konzepte:

Hypertext: Ein Text, der sich aus mehreren Texteinheiten zusammensetzt, die über Verweise (so genannte Hyperlinks) mit anderen Texten verknüpft sind. Das daraus entstehende Textmodell ist nicht mehr linear wie ein Buch, sondern netzartig.

Markup Language: Heisst auf deutsch «Auszeichnungssprache» und meint einen Sprachcode, der es ermöglicht in einem Klartextdokument für jedes Textelement festzuhalten, um was für eine Art von Textelement (Titel, Liste, Abschnitt, Abbildung etc.) es sich handelt, ohne es dafür speziell formatieren zu müssen. Der Trick besteht darin, in den Text entsprechende Markierungen (sogenannte Tags) einzufügen, die Auskunft darüber geben, um was für eine Art von Textelement es sich handelt. Die markierenden Tags werden einfach zu Beginn und am Ende des betreffenden Textelements hinzu geschrieben. Tags sind immer in spitzen Klammern geschrieben: <TAG-NAME>. Will man etwa eine Passage als Titel markieren, geht das so:

```
<TITLE>Titel des Dokuments</TITLE>
```

HTML: HTML kombiniert diese beiden Konzepte, um die logische und inhaltliche Struktur eines komplexen Textes in Klartext abspeichern zu können und eine einfache Navigation zwischen verknüpften Dokumenten zu ermöglichen. HTML ist somit eine besondere Auszeichnungssprache, weil es zum Erstellen von Hypertexten (sprich: Websites) dient.

Homepage (Website): Ist im technischen Sinn eine Sammlung von HTML-Dokumenten, die untereinander über Hyperlinks verknüpft sind und dabei auch andere Dateien wie Bilder, Ton- und Filmdateien etc. einbinden.

Weil HTML-Dokumente nur aus schwer zu entzifferndem Klartext bestehen, gibt es besondere Programme, nämlich die Browser (Firefox, Safari, Chrome, Edge etc.), die diese Dokumente in einer für Menschen ansprechenden Weise darstellen. Ein Browser liest die Tags als Anweisung, um die einzelnen Textelemente am Bildschirm darzustellen (also einen Titel dick hervorgehoben, eine Liste leicht eingerückt etc.). Der Browser nimmt also die Stellung eines Übersetzers oder eines Filters ein, indem er die reinen Textinformationen, die er aus dem Internet erhält, grafisch so aufbereitet, dass sie für die Benutzerin leicht aufzufassen und einfach zu bedienen sind.

In einem HTML-Dokument sind also zwei Ebenen von Informationen enthalten:

1. Die **eigentlichen Inhalte, die in der Webseite dargestellt werden sollen**. Diese Informationen sind für die Benutzerin des Browsers gedacht. (Abbildung 2.1: grau hinterlegte Bereiche)

2. Die **Anweisungen für den Browser, um was für eine Art von Information es sich handelt und wie er sie darzustellen hat**. (Abbildung 2.1: alle nicht grau hinterlegten Bereiche) Hierbei handelt es sich um die HTML-Tags. Die Verwendungsweise dieser Tags ist standardmässig definiert (aktuell HTML5), damit die Browser auch alle HTML-Dokumente korrekt dekodieren und darstellen können. Es gibt nur eine bestimmte Anzahl von Tags bzw. Elementtypen in HTML. Wichtige Beispiele sind: `<A>` für Anker (Abbildung 2.1, Zeile 11, für Hyperlinks, siehe auch Kapitel 13.2.1), oder `<P>` für Abschnitt (Abbildung 2.1, Zeile 9, siehe auch Kapitel 13.2.12).

Beide Informationsebenen werden in formatierungsfreiem Klartext in das HTML-Dokument geschrieben und bilden dessen Quellcode.

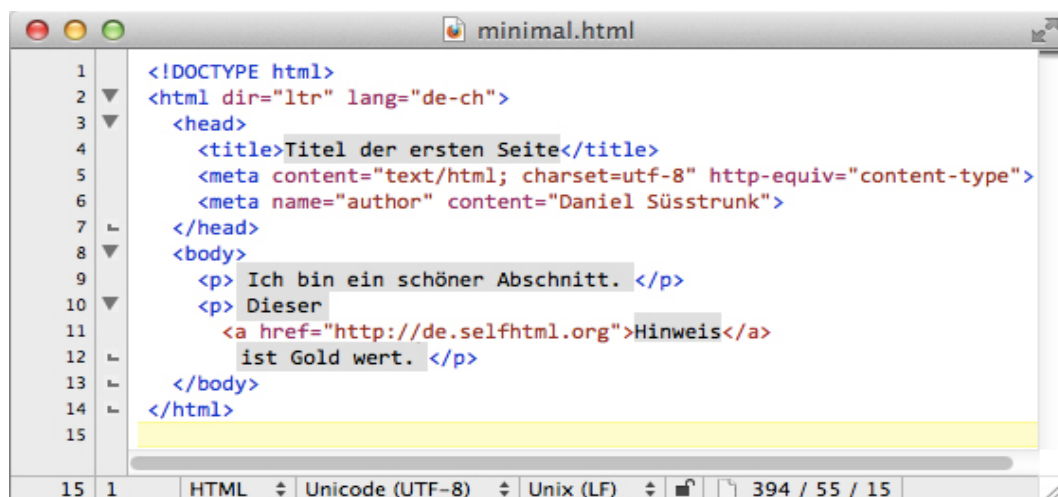


Abbildung 2.1: Der Quellcode der einfachsten Homepage: minimal.html

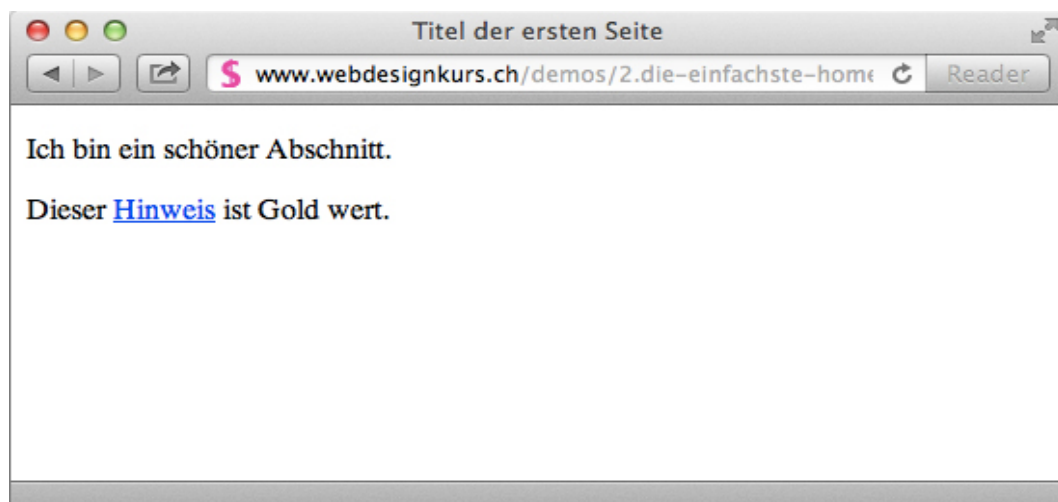


Abbildung 2.2: Die Browseransicht der einfachsten Homepage.

Markierende Tags und hierarchischer Aufbau

Jedes Textelement in einem HTML-Dokument muss mit einem Tag markiert werden, der Auskunft darüber gibt, um was für eine Art von Textelement es sich handelt. Weil der Geltungsbereich eines jeden Tags klar definiert sein muss, muss er immer mit einem Anfangs-Tag eröffnet und mit einem End-Tag wieder abgeschlossen werden (
, <META>, bilden die Ausnahme). End-Tags unterscheiden sich durch den dem Anfangs-Tag vorangestellten Schrägstrich. Also: </TAG-NAME>. Fehlt der End-Tag, weiss der Browser nicht, wann der Geltungsbereich wieder aufhört. Sehr anschaulich lässt sich dies am <A>-Tag illustrieren (Abbildung 2.3, Zeile 11), der nur einen kurzen Ausschnitt - im Beispiel das Wort «Hinweis» - innerhalb eines längeren Textes als Link kennzeichnet:

Dieser [A HREF="http://de.selfhtml.org"](http://de.selfhtml.org)
Hinweis ist Gold wert.

Tags können kombiniert und verschachtelt werden, wobei nicht alle Verschachtelungen erlaubt sind (z.B. darf innerhalb eines `<A>`-Tags kein weiterer `<A>`-Tag enthalten sein). Dies führt zu einem hierarchischen Aufbau des HTML-Dokuments. Die äusserste Position in der Hierarchie nimmt immer der `<HTML>`-Tag ein (Abbildung 2.3, Zeile 2 und 14), der das gesamte Dokument umfasst. Dann müssen die beiden Tags `<HEAD>` (Abbildung 2.3, Zeile 3 und 7) und `<BODY>` (Abbildung 2.3, Zeile 8 und 13) in dieser Reihenfolge folgen. Daher besteht das minimale HTML-Gerüst eines jeden HTML-Dokuments aus den Tags `<HTML>`, `<HEAD>` und `<BODY>`. Innerhalb des `<HEAD>`-Tags werden alle Informationen festgehalten, die nicht im Browserfenster dargestellt werden sollen, z.B. der Titel des Dokuments. Im Geltungsbereich des `<BODY>`-Tags werden all jene Anweisungen an den Browser geschrieben, die im Browserfenster dargestellt werden.

Weitere quasi immer vorkommende Tags sind der `<TITLE>`-Tag (Abbildung 2.3, Zeile 4) für die Titelangabe in der Kopfleiste des Browserfensters, sowie der `<META>`-Tag (Abbildung 2.3, Zeile 5 und 6, siehe auch Kapitel 13.2.9), der Zusatzinformationen über das HTML-Dokument enthält, die von Suchmaschinen ausgelesen werden. Beide Tags werden innerhalb des Heads eingefügt.



Abbildung 2.3: Der Quellcode der einfachsten Homepage: minimal.html

Weitere Tags zur Textstrukturierung

Um Texte mit HTML gut auszeichnen zu können sind neben dem <P>-Tag, der einfach einen generellen Text-Abschnitt beschreibt, weitere Tags vorhanden. Einige wichtige sind dabei <HEADER> und <FOOTER>, <NAV> für die Navigation (das Menu der Seite, <ASIDE> für eine Seitenleiste und die Überschriftentags <H1> (Überschrift 1. Ordnung), <H2> ... bis <H6> (Überschrift 6. Ordnung) (siehe Abbildung 2.4).

```

1 <!DOCTYPE html>
2 <html dir="ltr" lang="de-ch">
3   <head>
4     <title>Webseite der Kanti Baden</title>
5     <meta content="text/html; charset=utf-8" http-equiv="content-type">
6     <meta content="Daniel Süsstrunk" name="author">
7   </head>
8   <body>
9     <header>
10      <h1>Übersicht</h1>
11    </header>
12    <nav>
13      <a href="home.html">Home</a><br>
14      <a href="projekte.html">Projekte</a>
15    </nav>
16    Lorem ipsum dolor ...
17    <footer>
18      Sie finden mich per
19      <a href="mailto:daniel.suesstrunk@kanti-baden.ch">Email</a>.
20    </footer>
21  </body>
22 </html>

```

Abbildung 2.4: Einsatz von <HEADER>, <NAV> und <FOOTER> in einer Webseite.

Zur logischen Strukturierung eines Textdokuments ist es zudem möglich, innerhalb des <BODY>-Tags Abschnitte mit dem Tag <SECTION> zu definieren. Solche Abschnitte beinhalten typischerweise eine Überschrift, gefolgt von Inhalt.

Ähnlich wie das <SECTION>-Tag, aber vor allem im Kontext von Blogs oder Nachrichtenseiten verwendet, gibt es auch ein Tag für einen Artikel <ARTICLE> und zeichnet zu einem Artikel oder Blog-Post gehörende Inhalte.

Attribute und guter Stil

Die meisten HTML-Tags lassen noch eine genauere Bestimmung ihrer Eigenschaften zu, indem man ihnen Attribute hinzufügt. Jedes Attribut setzt sich aus dem Attributnamen, und dem in Anführungszeichen zugewiesenen Wert zusammen. Ein Beispiel soll dies veranschaulichen. Der Tag, um einen Link zu kennzeichnen, ist `<A>`. Damit der Browser weiss, wo sich das Zieldokument befindet (also das Dokument, das nach einem Klick auf den Link geladen werden soll) , muss ich das Attribut `HREF` hinzufügen und diesem den Wert mit der Zieladresse zuweisen. Konkret sieht das dann so aus:

Dieser `<A HREF="http://de.selfhtml.org">`
Hinweis`</A>` ist Gold wert.

Im Endtag braucht das Attribut nicht mehr wiederholt zu werden (Abbildung 2.3, Zeile 11).

Für den Browser spielt es keine Rolle, wie der Quellcode einer HTML-Datei dargestellt ist, solange er sich korrekt an die Konventionen des HTML-Standards hält. Theoretisch liesse sich daher auch die ganze Datei in eine einzige Zeile hinein schreiben. Da der Quellcode aber bisweilen auch von Menschen und ProgrammiererInnen betrachtet wird, so ist es ratsam, auch im Quellcode darauf zu achten, dass man sich an gewisse stilistische Konventionen hält, die das spätere Weiterarbeiten erleichtern.

Die elementarste dieser stilistischen Regeln besteht darin, die hierarchische Verschachtelung der Elemente im Quellcode durch Einrückungen abzubilden. Das heisst, der Tag eines Elements, das sich innerhalb eines anderen Elements befindet, wird gegenüber dessen Tag nach rechts eingertückt.