

HTML

Allgemeines

Nun sollen dem Nutzer bei Aufruf der Webanwendung nicht nur einfache Strings angezeigt werden, sondern eine richtige Website.

Hierzu gibt es den HTML-Standard, fast ausnahmslos alle Websites im Internet basieren auf HTML.

HTML steht für "*Hypertext Markup Language*" und ist daher, wie der Name vermuten lässt, eine Auszeichnungssprache. HTML ist *keine* Programmiersprache, es können keine Funktionen o.ä. implementiert werden. Eine Auszeichnungssprache ist lediglich eine Sprache, mit der Dokumente gegliedert und strukturiert werden. Der Inhalt wird also semantisch aufbereitet, sodass der Webbrowser weiß, wie das Dokument darzustellen ist.

Die visuelle Ausgestaltung und Darstellung ist jedoch nicht Teil von HTML, sondern wird durch die Stylesheet-Sprache CSS (Cascading Stylesheet) umgesetzt. Dies folgt dem Prinzip der Trennung von Inhalt (HTML) und Darstellung (CSS).

Tags

Durch HTML-Elemente, die *Tags* genannt werden, wird dem Inhalt der Website eine Struktur gegeben.

Jedes Element besteht aus einem *Starttag* und einem *Endtag*. Alles, was sich zwischen diesem Tag-Paar befindet, wird vom Element berücksichtigt. Inhalte, die sich außerhalb des Tag-Paars befinden, sind für das Element nicht sichtbar.

Tags werden mit dem "<"-Symbol geöffnet und mit dem ">"-Symbol (bei Starttags) bzw. dem ">"-Symbol (bei Endtags) wieder geschlossen. Zwischen den Symbolen wird der Elementname notiert, z.B.

```
<h1> Das ist eine Überschrift 1. Ordnung</h1>
<p> Das ist ein Textabschnitt (Paragraph)</p>
```

Jedes Element kann außerdem eine Liste von Attributen besitzen, z.B. `id`, um das Element auf der Website später eindeutig identifizieren zu können, oder `class`, um Elemente zu einer Klasse zusammenzufassen. Dies ist insbesondere in Kombination mit CSS bei der Gestaltung der Website relevant. Diese Attribute werden jeweils im Starttag des Elements notiert, z.B.

```
<p id="main-paragraph" class="hint important">Das ist der Haupt-Textabschnitt</p>
```

Besonderheit: Manche Elemente besitzen keinen Inhalt, da die relevanten Informationen durch die Attribute angegeben werden (Bsp. 1) oder sich der Zweck des Elements lediglich auf die Struktur und nicht auf den Inhalt bezieht (Bsp. 2). Bei inhaltsleeren Elementen muss daher kein Endtag eingefügt werden:

Bsp. 1: `` Hier wird ein Bild eingebunden ("image"). Da sich der Inhalt aus dem Bild selbst ergibt, besitzt das HTML-Element keinen Inhalt. Es muss jedoch die Quelle des Bildes ("src") als Attribut angegeben werden.

Bsp. 2: `
` Hierbei handelt es sich um einen Zeilenumbruch ("break"). Da der Zeilenumbruch keinen Inhalt besitzt, ist kein Endtag nötig.

Die Struktur von HTML ist hierarchisch, d.h. HTML-Elemente können wiederum weitere HTML-Elemente beinhalten (Kind-Elemente). Es müssen immer alle Kindelemente geschlossen werden, bevor die Elternelemente geschlossen werden.

Wichtige HTML-Elemente

Im Folgenden sind die wichtigsten HTML-Elemente aufgeführt:

- `<html>` `</html>` : Äußerstes Element, umgibt das komplette Dokument. Teilt dem Browser mit, dass es sich hierbei um ein HTML-Dokument handelt
 - `<head>` `</head>` : Erstes Kindelement des `<html>` -Elements, ist notwendig. Es besitzt noch keinen Inhalt, sondern lediglich Metainformationen über das Dokument, wie z.B. den Titel (`<title>`), eingebundene Stylesheets (`<link>` bzw. `<style>`) und weitere Metainformationen (`<meta>`).
 - `<title>` `</title>` : Definiert den Titel des Dokuments. Kann nur im `<head>` -Element genutzt werden.
 - `<body>` `</body>` : Beinhaltet den eigentlichen Inhalt und die Struktur des Dokuments
 - `<h1>` `</h1>` bis `<h6>` `</h6>` : Überschriften 1. - 6. Ordnung
 - `<p>` `</p>` : Textabschnitte
 - `
` : Zeilenumbruch
 - `<!-- ... -->` : Kommentar. Alles was zwischen den `--` steht wird vom Browser ignoriert
 - `<div>` `</div>` : Definiert einen größeren Abschnitt ("division") im Dokument
 - `` `` : Definiert einen kleineren Abschnitt (inline) im Dokument
 - `` : Definiert ein Bild. "src"-Attribut notwendig
 - `<a>` `` : Definiert einen Link ("anchor"). "href"-Attribut notwendig (Ziel des Links)
 - `<form>` `</form>` : Definiert ein Formular für Inputdaten.
 - `<input>` : Generisches Inputfeld zur Dateneingabe (Text, Dateien, Datum, ...)
 - `<select>` `</select>` : Dropdownliste
 - `<option>` `</option>` : Einzelne Werte der Dropdownliste
 - `<textarea>` `</textarea>` : Texteingabe (mehrere Zeilen)
 - `<button>` `</button>` : Klickbarer Button

Aufbau

Auf oberster Ebene steht im HTML-Dokument ein spezielles Element, die Dokumenttypdeklaration, die angibt, um welchen Dokumenttyp es sich hierbei handelt.

```
<!DOCTYPE html>
```

Anschließend folgt das `<html>` -Element als "Wrapper" für das gesamte Dokument. Im `<html>` -Element befinden sich dann das `<head>` - und `<body>` -Element.

Die Grundstruktur sieht also folgendermaßen aus:

```
<!DOCTYPE html>
<html>
  <head>
    <!-- Head -->
  </head>
  <body>
    <!-- Body -->
  </body>
</html>
```

Anders als bei Python sind hier Einrückungen nicht von Bedeutungen, der gesamte HTML-Code könnte auch in einer Zeile geschrieben werden. Eine saubere Struktur durch Einrücken erhöhen die Lesbarkeit jedoch deutlich.

Aufgabe:

-
1. Erstelle ein einfaches HTML-Dokument, das im Body lediglich "Hallo Welt" als Überschrift 1. Ordnung anzeigt. Stelle dieses Dokument durch den Flask Webserver über die Route `"/html"` bereit.
 2. Erstelle ein HTML-Dokument mit folgenden Elementen und stelle dieses Dokument durch den Flask Webserver über die Route `"/form"` bereit:
 - Bild
 - Formular (mit Labels):
 - Text (Name)
 - Zahl (PLZ)
 - Datum (Geburtsdatum)
 - Radiobuttons (Lieblingsfarbe)
 - Checkbox (Hobbies)
 - Dropdown (Bundesland)
 - Submit-Button
 - Div mit Text
 - Link