

LU08.L01 - Zusammenfassung

Live Preview

index.html

```
<!DOCTYPE html>
<html lang="de">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <meta name="author" content="Illic">
    <title>Zusammenfassung Modul 293</title>
    <style>
      body {
        padding-top: 70px; /* Höhe der Navbar */
      }
    </style>
  </head>
  <body>
    <header>
      <nav class="navbar navbar-expand-md navbar-dark bg-dark fixed-
top">
        <a class="navbar-brand" href="#">Modul 293</a>
        <button class="navbar-toggler" type="button" data-
toggle="collapse" data-target="#navbarNav"
               aria-controls="navbarNav" aria-expanded="false" aria-
label="Toggle navigation">
          <span class="navbar-toggler-icon"></span>
        </button>
        <div class="collapse navbar-collapse" id="navbarNav">
          <ul class="navbar-nav">
            <li class="nav-item"><a class="nav-link" href="#lu01">LU
01</a></li>
            <li class="nav-item"><a class="nav-link" href="#lu02">LU
02</a></li>
            <li class="nav-item"><a class="nav-link" href="#lu03">LU
03</a></li>
            <li class="nav-item"><a class="nav-link" href="#lu04">LU
04</a></li>
          </ul>
        </div>
      </nav>
    </header>

    <div class="container">
      <h1>Zusammenfassung Modul 293</h1>
    </div>
  </body>
</html>
```

```
<div id="lu01">

<br>
<br>
<br>

<!-- Inhalt von LU01 -->
<p>
<h3> In der Learningunit 01 gibt es folgende Themen:</h3> <br>
- HTML <br>
- Werkzeuge <br>
- Einstellungen von WebStorm
</p>

<h4>HTML Grundlagen</h4> <br>
<p><b>H</b>yper<b>T</b>ext <b>M</b>arkup <b>L</b>anguage kurz<br>
HTML ist einer der verbreitesten Auszeichnungssprachen.
Mit HTML werden Websites erstellt.<br>
Zu HTML geh&ouml;rt noch das Element, ein Tag, die Attribute und CSS dazu. Die Elemente sind die Bausteine von HTML und wird durch einen Tag markiert. Ein Element kann noch weitere verschachtelte Elemente enthalten.
Ein Tag, deutsch Markierung, ist ein Schl&uuml;sselwort von HTML. Es wird zum markieren von Elementen gebraucht und wird zwischen spitzen Klammern geschrieben, hinzu kommt, dass sie meistens geschlossen werden m&uuml;ssen, Bsp. <body> </body>
Attribute kommen nur innerhalb eines Tags und erhalten Angaben zur Verarbeitung eines Tags.
Um den Inhalt zu gestalten braucht man CSS (<b>C</b>ascading <b>S</b>tyle <b>S</b>heets) f&uuml;r die Darstellung.

<br><br><br>
</p>

</div>

<div id="lu02">

<br>
<br>
<br>

<!-- Inhalt von LU02 -->
<p>
```

<h3>In der Learningunit 02 geht es um den Aufbau einer HTML-Datei</h3>

<!DOCTYPE html>

Auf der ersten Zeile legen wir die Dokumentenart (engl. Document Type Definition (DTD) fest. Durch den <!DOCTYPE html> weiss der Browser, wie er diese Webseite darstellen soll.

DOCTYPE sieht einem HTML-Tag ähnlich, ist aber kein HTML-Tag. Es gibt zum Beispiel kein schliessendes Tag.

< html> ... </html>

Das ganze HTML-Dokument muss zwischen diesen beiden Tags stehen. Für die Benutzbarkeit der Seite, muss das Attribut lang=? angegeben werden.

< head> ... </head>

Der head-Block enthält Informationen über den Inhalt der Seite. Diese Angaben werden im Webbrowser nicht angezeigt.

Die Angabe eines Titels (< title>...</title>) ist zwingend.
Durch die Angabe der Zeichencodierung (< meta charset="utf-8">) vermeiden wir Fehler bei der Darstellung von Umlauten und Sonderzeichen.

< body> ... </body>

Der Inhalt unserer Webseite wird im body-Block geschrieben. Dieser Bereich des HTML-Dokuments wird vom Browser angezeigt.

< header> und < footer>

Wie ein Worddokument haben auch Webseiten in der Regel eine Kopf- und Fusszeile. Diese enthalten Logos, Name der Seite, Copyright, ... Besonders bei einem Webauftritt mit mehreren Seiten helfen einheitliche Kopf- und Fusszeilen dem Betrachter bei der Orientierung.

< nav>

Siehe auch selfhtml - nav

Das Element < nav> umschliesst die Navigation unseres Webauftritts. Es wird häufig innerhalb von < head> platziert.

< main>

Der Inhalt unserer Webseite wird innerhalb von < main>

geschrieben. Dieser Bereich kann durch < article> und < section> weiter unterteilt werden.

< article>

Ein Artikel stellt einen in sich geschlossenen Teil des Inhalts dar. Dieser kann aus mehreren Abschnitten (< section>) bestehen.

< section>

Das Element < section> bezeichnet einen Abschnitt innerhalb eines Artikels.

Dabei gibt es noch eine Vorlage, die man Personalisieren kann.

</p>

</div>

<div id="lu03">
 <!-- Inhalt von LU03 --></p>

<p>

<h3> In der Learningunit 03 geht es um Blöcke und Inhalte, sowie die Struktur</h3>

<h5>Die Struktur</h5>

Sie werden hauptsächlich auf 2 Arten von HTML-Elementen stoßen:

Block-Elemente wie:

Absätze < p>

Listen: ungeordnete (mit Aufzählungspunkten) < ul> oder geordnete Listen (mit Zahlen) < ol>

Überschriften: von der 1. Ebene < h1> bis zur 6. Ebene < h6>

artikel < artikel>

abschnitte < section>

Inline-Elemente wie:

Links < a>

hervorgehobene Wörter < em>

wichtige Wörter < strong>

kurze Zitate < q>

Block-Elemente dienen dazu, die wichtigsten Teile Ihrer Seite zu strukturieren, indem sie Ihren Inhalt in zusammenhängende Blöcke unterteilen.

Inline-Elemente dienen dazu, einen Teil eines Textes zu differenzieren, um ihm eine bestimmte Funktion oder Bedeutung zu geben.
Inline-Elemente bestehen in der Regel aus einem einzigen oder wenigen Wörtern.

<h5>Beispiele von Elementen</h5>

h1 bis h6-Element

Die Tags < h1> bis < h6> werden zur Definition von HTML-Überschriften verwendet. < h1> definiert die wichtigste Überschrift.

< h6> definiert die unwichtigste Überschrift.

Hinweis: Verwenden Sie nur eine < h1> pro Seite - diese sollte die Hauptüberschrift/das Hauptthema für die gesamte Seite darstellen.

Überspringen Sie auch keine Überschriftenebenen - beginnen Sie mit < h1>, verwenden Sie dann < h2> und so weiter.

div-Element

Es funktioniert wie ein Blockelement, das die gesamte Breite des Bildschirms einnimmt und immer in einer neuen Zeile beginnt.

Der div-Tag definiert eine Unterteilung oder einen Abschnitt in einem HTML-Dokument. Das < div>-Tag wird als Container für HTML-Elemente verwendet, die dann mit CSS gestaltet oder mit JavaScript manipuliert werden. Jede Art von Inhalt kann in das < div>-Tag eingefügt werden!

p-Element

Normale Texte werden auf Webseiten in Absätzen geschrieben. Für jeden Textabschnitt wird ein Absatz erzeugt und große Texte können so in sinnvolle Abschnitte unterteilt werden. Um mit HTML einen Absatz zu erzeugen, verwendet man den < p>-Tag.
Der Buchstabe P ist von Paragraph abgeleitet und das bedeutet Absatz.

table-Element

Eine Tabelle wird verwendet, um Daten in einem Tabellenformat darzustellen, das einen Tabellenkopf, einen Tabellenkörper mit einer Tabellenzeile und eine Tabellenspalte umfasst.

Der < table>-Tag definiert dazu eine HTML-Tabelle. Darin können eines oder mehrere < tr>- , < th>- und < td>-Elemente vorkommen.

Das < tr>-Element definiert eine Tabellenzeile, das < th>-Element definiert eine Tabellenüberschrift, und das < td>-Element definiert eine Tabellenzelle.

Weiter kann eine HTML-Tabelle kann auch < caption>- , < colgroup>- , < thead>- , < tfoot>- und < tbody>-Elemente enthalten, mehr dazu bei w3school.

<h5> Beispiele von Inline-Elementen</h5>

span-Element

Der < span>-Tag ist ein Inline-Container, der zur Kennzeichnung eines Textteils oder eines Teils eines Dokuments verwendet wird.

Er lässt sich leicht mit CSS gestalten oder mit JavaScript unter Verwendung des class- oder id-Attributs manipulieren.

Der < span>-Tag ist dem < div>-Element sehr ähnlich, aber < div> ist ein Block-Level-Element und < span> ein Inline-Element.

**a-Element
**

Der `<a>`-Tag definiert einen Hyperlink, der verwendet wird, um von einer Seite auf eine andere zu verlinken.

Das wichtigste Attribut des `<a>`-Elements ist das `href`-Attribut, das das Ziel des Links angibt.

**
**

Standardmäßig werden Links in allen Browsern wie folgt dargestellt:

**

Ein nicht besuchter Link ist unterstrichen und blau

Ein besuchter Link ist unterstrichen und lila

Ein aktiver Link ist unterstrichen und rot.

Live Preview

****img-Element
**

Der ``-Tag wird verwendet, um ein Bild in eine HTML-Seite einzubetten.

Bilder werden technisch gesehen nicht in eine Webseite eingefügt; Bilder werden mit Webseiten verknüpft.

Der ``-Tag schafft einen Bereich für das referenzierte Bild.

Der ``-Tag hat zwei erforderliche Attribute:

`src` - Gibt den Pfad zu dem Bild an
`alt` - Gibt einen alternativen Text für das Bild an, falls das Bild aus irgendeinem Grund nicht angezeigt werden kann. Hinweis: Geben Sie auch immer die Breite (`width`) und Höhe

(`height`) eines Bildes an. Wenn Breite und Höhe nicht angegeben werden, kann die Seite beim Laden des Bildes flackern.

**
**

Tipp: Um ein Bild mit einem anderen Dokument zu verknüpfen, verschachteln Sie einfach den ``-Tag in einen `<a>`-Tag.

**

****</p>****</div>****<div id="lu04">**

`<!-- Inhalt von LU04 -->`

**

****<p>**

`<h3> In der Learningunit 04 geht es um Codingstyle und Validator</h3>`

<h5>Elementnamen in Kleinbuchstaben schreiben </h5>
HTML erlaubt die Mischung von Groß- und Kleinbuchstaben in Elementnamen.

Wir empfehlen jedoch, Elementnamen in Kleinbuchstaben zu verwenden, denn:

Die Vermischung von Groß- und Kleinbuchstaben sieht schlecht aus
Entwickler verwenden normalerweise Kleinbuchstaben
Kleinbuchstaben sehen sauberer aus
Kleinschreibung ist einfacher zu schreiben

<h5>Kleinbuchstaben für Attributnamen verwenden</h5>
HTML erlaubt die Mischung von Groß- und Kleinbuchstaben in Attributnamen.

Wir empfehlen jedoch die Verwendung von Attributnamen in Kleinbuchstaben, denn:

Die Vermischung von Groß- und Kleinbuchstaben sieht schlecht aus
Entwickler verwenden normalerweise Kleinbuchstaben
Kleinbuchstaben sehen sauberer aus
Kleinschreibung ist einfacher zu schreiben

<h5>Attributwerte immer in Hochkomma setzen</h5>
HTML erlaubt Attributwerte ohne Hochkommal) oder Anführungszeichen.

setzen
Entwickler sind es sich gewohnt Attributwerte in Hochkomma zu setzen
in Hochkomma gesetzte Werte sind leichter zu lesen
Sie MÜSSEN sowieso Hochkomma verwenden, wenn der Wert Leerzeichen enthält

<h5> Alle HTML-Elemente schließen</h5>
In HTML müssen Sie nicht alle Elemente schließen (zum Beispiel das < p>-Element).

<h5>alt, width und height für Bilder angeben</h5>
Geben Sie immer das alt-Attribut für Bilder an. Dieses Attribut

ist wichtig, wenn das Bild aus irgendeinem Grund nicht angezeigt werden kann.

Geben Sie außerdem immer die Breite width und Höhe height von Bildern an. Dies verringert das Flackern, da der Browser vor dem Laden Platz für das Bild reservieren kann.

<h5>Leerzeichen und Gleichheitszeichen</h5>

HTML erlaubt Leerzeichen um Gleichheitszeichen. Ein Text ohne Leerzeichen ist jedoch leichter zu lesen und gruppiert Einheiten besser zusammen.

<h5>Vermeiden Sie lange Codezeilen</h5>

Wenn Sie einen HTML-Editor verwenden, ist es NICHT bequem, nach rechts und links zu scrollen, um den HTML-Code zu lesen.

Versuchen Sie, zu lange Codezeilen zu vermeiden.

<h5>Leerzeilen und Einrückungen</h5>

Fügen Sie keine Leerzeilen, Leerzeichen oder Einrückungen ohne Grund ein. Fügen Sie zur besseren Lesbarkeit

Leerzeilen ein, um große oder logische Codeblöcke zu trennen.

Fügen Sie zur besseren Lesbarkeit zwei Leerzeichen als Einrückung ein.

Hinweis: Webstorm ersetzt einen Tabulator automatisch mit Leerzeichen

<h5>< title>-Element immer angeben</h5>

Das < title>-Element ist in HTML erforderlich.

Der Inhalt eines Seitentitels ist sehr wichtig für die Suchmaschinenoptimierung (SEO)! Der Seitentitel wird von den Algorithmen der Suchmaschinen verwendet, um die Reihenfolge bei der Auflistung der Seiten in den Suchergebnissen zu bestimmen.

Das < title>-Element:

definiert einen Titel in der Browser-Symbolleiste

liefert einen Titel für die Seite, wenn sie zu den Favoriten hinzugefügt wird

zeigt einen Titel für die Seite in den Suchergebnissen der Suchmaschinen an

Versuchen Sie also, den Titel so genau und aussagekräftig wie

möglich zu gestalten:


```
< title>HTML Style Guide and Coding Conventions</title>
```

<h5>HTML-Kommentare</h5>

Kurze Kommentare sollten in einer Zeile geschrieben werden, etwa so:

```
<!-- This is a comment -->
</p>
```

</div>

```
<div id="lu05">
```


<!-- Inhalt von LU05 -->

```
<p>Kommt noch</p>
```

</div>

</div>

```
<footer class="fixed-bottom bg-light">
```

<div class="container">

<p>Footer</p>

</div>

</footer>

<script

```
src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
```

<script

```
src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.5.4/dist/umd/popper.min.js"></script>
```

<script

```
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/js/bootstrap.min.js"></script>
```

</body>

</html>



Kevin Maurizi, Marcel Suter

From:

<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:

<https://wiki.bzz.ch/modul/m293/learningunits/lu08/loesungen/zusammenfassung?rev=1716790763>

Last update: **2024/05/27 08:19**

