

# Web-basierte Systeme

## 02: Hypertext Markup Language

---

Wintersemester 2024

Rüdiger Kapitza



Lehrstuhl für Informatik 4  
Systemsoftware



Friedrich-Alexander-Universität  
Technische Fakultät

# HTML & CSS

---

# Vorläufiger Vorlesungsplan

<b>16. Oktober</b>	Einführung und Darstellung von Webseiten (HTML und CSS)
<b>23. Oktober</b>	Hypertext Transfer Protocol
<b>30. Oktober</b>	Browser Schnittstellen
<b>6. November</b>	Kommunikationsschnittstellen im Browser
<b>13. November</b>	WebAssembly
<b>20. November</b>	Architektur moderner Browser
<b>27. November</b>	Clientseitige Architekturmuster und serverseitige Implementierung von Web-basierten Systemen
<b>4. Dezember</b>	Clientseitige Architekturmuster und serverseitige Implementierung von Web-basierten Systemen
<b>11. Dezember</b>	Vorbereitung Papieranalyse
<b>8. Januar</b>	Papieranalyse
<b>15. Januar</b>	Lastverteilung durch Zwischenspeicher
<b>22. Januar</b>	Web3
<b>29. Januar</b>	Aspekte von Web Sicherheit
<b>5. Februar</b>	Zusammenfassung und Ausblick

## Zielsetzung der Lerneinheit

- Basisverständnis von *Hypertext Markup Language (HTML)*
- Gängigste Elemente der Sprache und wie man sie anwendet
- *Cascading Style Sheets (CSS)* als Gestaltungsanweisungen für HTML

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title></title>
5   </head>
6   <body>
7
8   </body>
9 </html>
```

# Verwendung und Unterschiede zwischen HTML & CSS

- Webbrowser visualisieren in HTML beschriebene Dokumente
- Bei purem HTML wird die genaue Darstellung weitgehend vom Browser bestimmt
  - Seit HTML 5 bildet das Canvas-Element hier eine Ausnahme
- Mittels Cascading Style Sheets (CSS) kann die Darstellung von HTML-Elementen und Inhalten genauer definiert werden
- Anwendungen *aus* Dokumenten
  - **Ursprünglich:** Dokumente die durch Formularfelder Informationen gesammelt und an Webserver vermittelt haben
  - **Aktuell** werden HTML-Dokumente bei der Auslieferung oft dynamisch generiert und im Browser modifiziert

HyperText **Markup Language** – eine Auszeichnungssprache

- HTML ist eine textbasierte Auszeichnungssprache zur Strukturierung und Darstellung digitaler Dokumente
- Auszeichnungssprachen sind maschinenlesbare Sprachen für die Gliederung und Darstellung:

1 `<i>italics word</i>` und `<title>Title words</title>`

- HTML nutzt `<` und `>` zur Kennzeichnung von *Tags*

Arbeitsablauf

1. Erarbeitung des Inhalts
2. Strukturierung und Formatierung durch *Tags*

Tags übernehmen verschiedene Funktionen

- **Formatierungshinweise** wie im Fall von `<i>` für *kursiv*
- **Strukturhinweise**
  - `<h1>` Überschrift
  - `<p>` Textparagraph
  - `<ul><li>` Liste
  - Tags können aber auch weitere Informationen einbinden (z.B. `<img>`) und verschachtelt mit anderen Tags auftreten

- 1 Vorlesung
- 2 Die Vorlesung vermittelt grundsätzliches Verständnis von Web-basierten Systemen. Im Einzelnen:
  - 3
  - 4 Technische Grundlagen des Internets
  - 5 HTTP als Transportprotokoll
  - 6 HTML und CSS
  - 7 Paradigmenwechsel zu Web-basierten Systemen

```
1 <h2>Vorlesung</h2>
2 <p>Die Vorlesung vermittelt grunds&auml;tztliches Verst&
   auml;ndnis von <i>Web-basierten Systemen</i>. Im
   Einzelnen:</p>
3 <ul>
4 <li>Technische Grundlagen des Internets
5 <li>HTTP als Transportprotokoll
6 <li>HTML und CSS
7 <li>Paradigmenwechsel zu Web-basierten Systemen
8 </ul>
```

```
1 <h2>Vorlesung</h2>
2
3 <p>Die Vorlesung vermittelt grunds&uuml;tztliches Verst&
  auml;ndnis von <i>Web-basierten Systemen</i>.
4   Im Einzelnen:
5 </p>
6
7 <ul>
8   <li>Technische Grundlagen des Internets
9   <li>HTTP als Transportprotokoll
10  <li>HTML und CSS
11  <li>Paradigmenwechsel zu Web-basierten Systemen
12 </ul>
```

## Vorlesung

Die Vorlesung vermittelt grundsätzliches Verständnis von *Web-basierten Systemen*. Im Einzelnen:

- Technische Grundlagen des Internets
- HTTP als Transportprotokoll
- HTML und CSS
- Paradigmenwechsel zu Web-basierten Systemen

- HTML gibt es als Auszeichnungssprache seit mehr als 20 Jahren
- Im Folgenden wird *ein Teil* der Version 5 vorgestellt
  - Aktuellste Version: HTML5 (<https://html.spec.whatwg.org>)
- In den Jahren davor wurde auch XHTML neben HTML verwendet
  - XHTML ist XML konform und ist strikter in seiner Struktur
  - Im Rahmen der Veranstaltung wird aber HTML5 genutzt

## Zeichensatz

- HTML-Dateien bestehen aus reinem Text
- Oft wird das Format ISO 8859-1 (Latin-1) oder utf-8 genutzt
- Mit utf-8 kann man den *gesamten* Zeichensatz abbilden
- Beispiel: `<meta charset="UTF-8">`

## Sonderzeichen

- Sonderzeichen in HTML nennt man *character entities*
- Sie haben die Form von `&entityname;` angegeben
- Beispiele: `< → &lt;`; oder `& → &amp;`;

## Etwas mehr zu Tags

- Tags sind zwischen spitzen Klammern eingeschlossen
- Zu (fast) jedem Anfangs-Tag gibt es einen End-Tag
  - Z. B. `<p>hier der Absatz</p>`.
- Nur in XHTML müssen alleinstehende Tags mit einem Schrägstrich am Ende geschrieben werden: `<br/>`  
In HTML5 schreibt man diesen Tag (wieder) als `<br>`
- In XHTML werden Tags immer klein geschrieben, bei HTML ist die Groß- und Kleinschreibung egal

## Formatierung

- Jegliche Formatierungshinweise wie Leerzeichen, Tabulatoren und Zeilenumbrüche in der HTML-Datei werden ignoriert.
- Zeilenumbruch – wenn sinnvoll – mit `<br>` erzwingen

## Attribute

- Manche Tags können Attribute enthalten
  - Beispiel: ``
  - `src` – Quelle der Grafik
  - `alt` – Alternative Darstellung
- Reihenfolge neutral!
- Anführungszeichen für XHTML zwingend erforderlich – für HTML optional und implizites Ende beim nächsten Leerzeichen!

## Unbekannte Attribute und Tags

- Neue Attribute und Tags, die z. B. ein Browser nicht erkennt, werden (typischerweise) ignoriert!
- Bei Neuerungen ist zu überlegen, wer diese bereits kennt und welche Konsequenzen es hat, wenn nicht unterstützte Tags und Attribute ignoriert werden.
- Stichwort: **Graceful degradation**

## Text formatieren

- Man unterscheidet zwischen HTML-Tags, die Blöcke definieren, und solchen, die dies nicht tun, den so genannten *inline*-Tags
- Blockbildende Tags beanspruchen immer einen rechteckigen Bereich bei der Ausgabe, inline-Tags tun das nicht.
- Bsp.: **Paragraph** `<p>` als blockbildende Tag – versus – **kursiv** `<i>`

## Blockbildende Tags für Text

- Inhalt eines Absatzes (Paragraph) `<p>`
  - Zentrierung des Textes & weitere Darstellung ist Aufgabe von CSS
- `<h1>` bis `<h3>` - Überschriften
- `<nav>`-Tag zur Auszeichnung von Navigations-Menüs

```
1 <nav>
2   <a href="index.html">Home</a>
3   ....
4 </nav>
```

- ... dieser Tag hat erst mal keine sichtbare Wirkung.
- Weitere Tags zur Strukturierung einer Seite:
  - `<header>`, `<footer>` und `<aside>`
- Block-Zitate via `<blockquote>`

## Nicht-Blockbildende (inline) Tags für Text

- Diese Tags müssen innerhalb eines Blocks verwendet werden
- Es werden logische und physische Elemente unterschieden
- Logische Tags geben dabei die genaue Darstellung nicht vor:

```
1 <strong>(stark betont, meist fett)</strong>  
2 <em>(betont, meist kursiv)</em>
```

- Physische Tags geben die genaue Darstellung vor, werden aber als veraltet betrachtet
  - Beispiel: <b>, <i> oder auch <font> Tag

## Bilder

- Bilder werden in separaten Dateien gespeichert und über einen (absoluten oder relativen) Verweis referenziert

```
1 
2 
3 (mit Ersatztext, da das Bild wichtige Information enthält)
4 
5 (ohne Ersatztext, weil das Bild zur Dekoration dient)
```

- Es werden ein Vektor-Format (SVG) und drei Pixel-Formate von vielen Browsern unterstützt: GIF, JPG und PNG
- *Neuere* Formate: APNG, WebP und AVIF
- Alternativ können Bilder auch programmiert werden, vgl. canvas-Tag

## Links

- Über das href-Attribut können absolute und relative URLs angegeben werden:

```
1 <a href="http://cnn.com">zu CNN</a>  
2 <a href="seite2.htm">mehr</a>  
3 <a href="mailto:aa@bb.cc">Mail</a>
```

## Listen

- Listen werden mit zwei verschachtelten Tags gebaut:
  - der <li>-Tag für den einzelnen Listen-Punkt,
  - und der <ol>- oder <ul>-Tag für die gesamte Liste

## Tabellen

- Eine Tabelle setzt sich aus einem `<table>`-tag und
  - `<tr>`s (Table Row = Tabellenzeile) sowie `<td>`s (Table Data) zusammen

```
1 <table style="border-collapse:collapse;">
2   <tr>
3     <td>Samstag</td>
4     <td>Sonntag</td>
5   </tr>
6   <tr>
7     <td>lernen</td>
8     <td>lange schlafen</td>
9   </tr>
10 </table>
```

- Tabellen wurden früher oft für das Layout der Seite verwendet
  - Obsolet und über CSS zu lösen

## ■ DOCTYPE, html, head und body

```
1 <!DOCTYPE HTML>
2 <html>
3 <head>
4   <meta charset="utf-8">
5   <title>Übersicht Web-basierten Systemen</title>
6 </head>
7 <body>
8   <h2>Vorlesung</h2>
9   <p>Die Vorlesung vermittelt grundsätzliche Verstandnis von <i>Web-basierten Systemen</i>. Im Einzelnen:</p>
10  <ul>
11    ...
12 </body>
13 </html>
```