

Grundlagen der HyperText Markup Language (HTML)

Sommersemester 2022

Prof. Dr.-Ing. Thomas Barth

Anwendungsentwicklung und IT-Systemarchitektur

Studiengang Wirtschaftsinformatik

RFH Köln

barth@rfh-koeln.de

- ✓ Motivation und Zielsetzung von HTML kennen
- ✓ HTML gegenüber anderen Sprachen abgrenzen können
- ✓ Grundstruktur eines HTML-Dokuments („Web-Seite“) kennen
- ✓ Ausgewählte HTML-Elemente kennen und anwenden können
- ✓ HTML-Dokument manuell erstellen können
- ✓ HTML-Dokument mit Werkzeug erstellen können
- ✓ Unterschied statisches und dynamisches HTML kennen
- ✓ Defizite von HTML kennen und Lösungswege kennen

✓ In welche Bestandteile läßt sich eine „typische Webseite“ zerlegen?

➤ Inhalt

- Text
- Grafik
- Verknüpfung
- Interaktionselemente
 - Knöpfe
 - Texteingabe

➤ Struktur

- Seiten
- Abschnitte
- Überschriften

➤ Darstellung

- Vordergrund-/Hintergrundfarbe
- Schriftgröße
- Stil

- ✓ HTML bietet Möglichkeiten zur Nutzung bzw. Steuerung aller dieser Aspekte einer Webseite
- ✓ Ein HTML-Dokument enthält also ...
 - ... den eigentlichen (hoffentlich für den Besucher nützlichen) **Inhalt**, der auf der Webseite dargestellt werden soll
 - ... Informationen zur **Struktur** der Seite, in der die Inhalte verankert sind
 - ... Informationen zur **Darstellung** der Inhalte in der Struktur gegenüber dem Betrachter
- ✓ Die Darstellung des Inhalts kann zum Einen **durch HTML** bestimmt werden
 - Überschriften
 - Hervorhebung (Fettdruck, hoch-/tiefgestellte Zeichen, Schriftfarbe, Font)
- ✓ Um die Darstellung vom HTML-Dokument zu trennen (und damit die einheitliche Pflege einer größeren Menge von Seiten zu vereinfachen), kann die Bestimmung der Darstellung zum Anderen in **Cascading Style Sheets (CSS)** ausgelagert werden
- ✓ CSS-Dateien können zentral gepflegt werden und damit die Darstellung aller HTML-Dokumente ändern, die sich auf diese CSS-Dateien beziehen

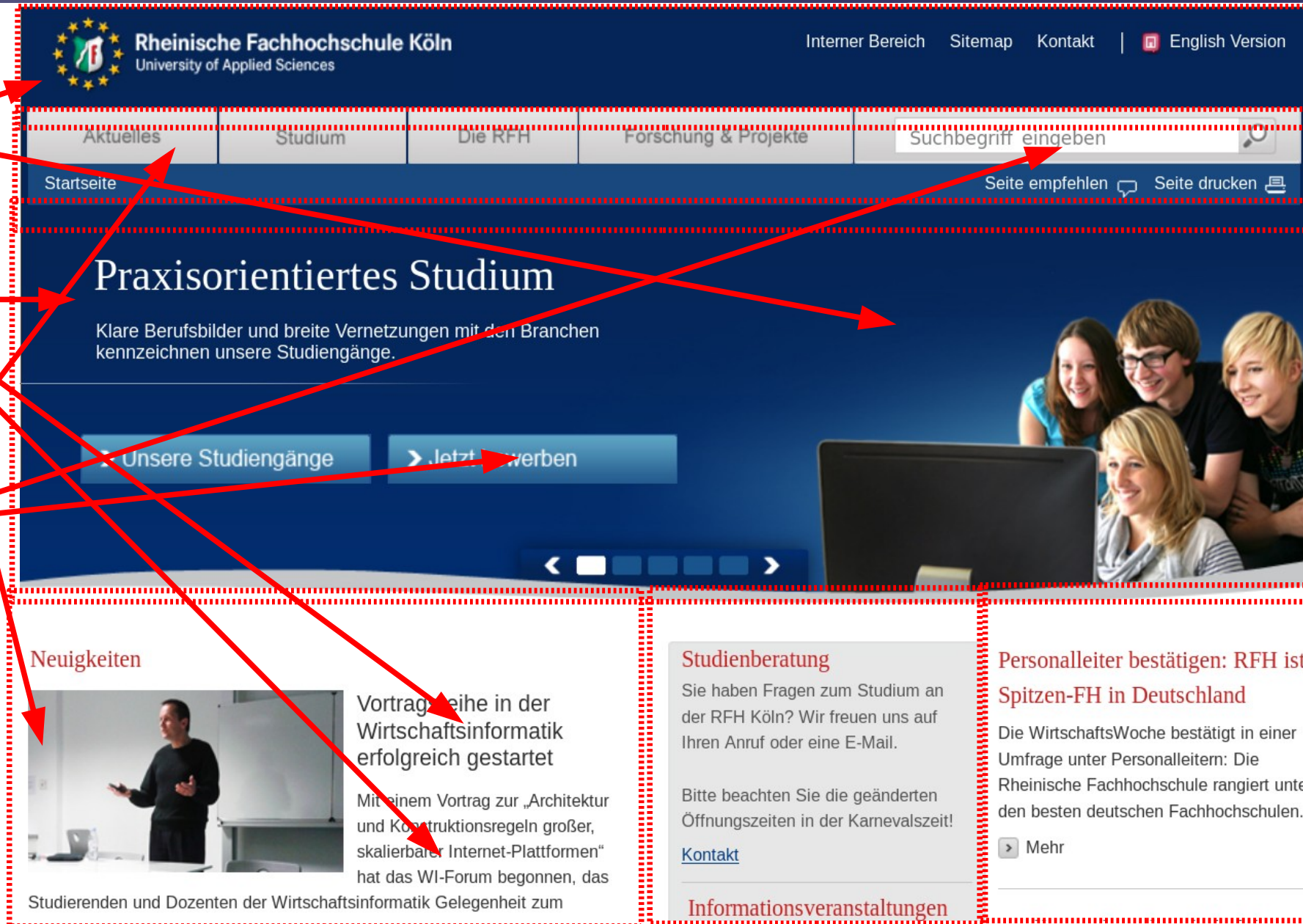
Inhalte einer Webseite

Grafik

Text

Interaktions-
elemente
und
Eingabefelder

Struktur



The screenshot shows the homepage of the Rheinische Fachhochschule Köln. Red arrows from the labels on the left point to specific elements on the page:

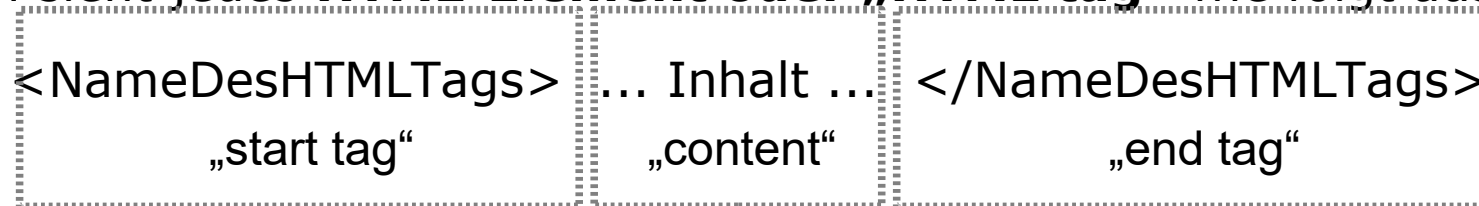
- Grafik** points to the university logo and the navigation bar.
- Text** points to the main heading "Praxisorientiertes Studium" and the subtext "Klare Berufsbilder und breite Vernetzungen mit den Branchen kennzeichnen unsere Studiengänge."
- Interaktions-elemente und Eingabefelder** points to the search bar, the "Suchbegriff eingeben" button, and the "Jetzt bewerben" button.
- Struktur** points to the "Neuigkeiten" section, specifically to the article about the "Vortragreihe in der Wirtschaftsinformatik".

Other visible elements include the top navigation bar with links like "Aktuelles", "Studium", "Die RFH", "Forschung & Projekte", and "Suchbegriff eingeben". The right side features a photo of students and a sidebar with sections like "Studienberatung", "Personalleiter bestätigen: RFH ist Spitzen-FH in Deutschland", and "Informationsveranstaltungen".

- ✓ HTML bietet Möglichkeiten, **Inhalt**, **Aufbau** und **Aussehen** einer Webseite zu definieren:

Daher ist HTML eine **Auszeichnungssprache** („markup language“), **KEINE Programmiersprache**

- ✓ Eine Auszeichnungssprache besteht aus Markierungen („**tags**“), die den jeweiligen Inhalt beschreiben (aber selbst im Ergebnis nicht „sichtbar“ sind)
- ✓ HTML-Dokumente bestehen ausschließlich aus **Text (Inhalt)** und **tags (Beschreibung des Inhalts)**
- ✓ Allgemein sieht jedes **HTML-Element oder „HTML tag“** wie folgt aus:



- ✓ Einfache Beispiele:
 - Absatz: <p>Ein kompletter Absatz</p>
 - Fettdruck: fettgedruckt
 - Wichtigste Überschrift der Seite: <h1>Überschrift</h1>

- ✓ HTML-Elemente **können** in einander geschachtelt werden, d.h. der Inhalt(!) eines tags kann andere tags enthalten
 - Mehrere Absätze innerhalb einer Seite
 - Fettdruck innerhalb der Überschrift
 - Kompletter Absatz fettgedruckt
- ✓ HTML-Elemente **sollten** jeweils aus Start- und End-Tag bestehen, obwohl das Weglassen des End-Tags oftmals nicht zum Fehler durch den Browser führt
 - `<p>erster Absatz<p>zweiter Absatz ...` funktioniert **wahrscheinlich**
 - `<p>erster Absatz</p><p>zweiter Absatz</p> ...` funktioniert **immer!**
- ✓ HTML-Elemente **können** leer sein
 - Zeilenumbruch („linebreak“) durch `
` statt `
</br>`
 - Kurzform in XHTML: `
`
- ✓ HTML-Elemente **sollten** in Kleinbuchstaben notiert werden, z.B. `<p>` vs. `<P>`
- ✓ Jegliche Formatierung innerhalb eines HTML-Dokuments (Zeilenumbrüche, Leerzeilen, Leerzeichen) ist für die Darstellung im Browser (fast) irrelevant
 - Ausnahme: ein Zeilenumbruch bedeutet den Beginn eines neuen Wortes

- ✓ Ein Web Browser hat die Aufgabe, die HTML-Elemente einer Web-Seite ...
 - ... syntaktisch zu **erkennen** (ist HTML tag bekannt?),
 - ... semantisch zu **interpretieren** (ist Schrifteffekt zu ändern?) und
 - ... **darzustellen**. „HTML-Interpreter“ im Web-Browser
 - Beispiel:
im HTML-Dokument: `Überschrift` wird dargestellt als **Überschrift**
- ✓ Ein HTML-Dokument besteht aus:
 - Der **Deklaration des Typs** des Dokuments `<!DOCTYPE html>`
 - Der Beschreibung der gesamten **Webseite** `<html> ... </html>`
 - Dem **sichtbaren Teil (Rumpf)** der Web-Seite `<body> ... </body>`
- ✓ Jedes HTML-Dokument enthält also bereits verschachtelte HTML-Elemente (mindestens html und body)
- ✓ **Kommentare** (`<!-- ... -->`) können in HTML-Dokumenten eingefügt werden und werden **nicht** im Web Browser angezeigt
 - `<!--` Kommentar beginnt hier ... und endet hier `-->`

Beispiel: Einfaches HTML-Dokument

```
<html>
```

```
<body>
```

```
<h1>Überschrift</h1>
```

```
<p>Ein kompletter Absatz.</p>
```

```
<b>fettgedruckt</b>
```

```
</body>
```

```
</html>
```

Beispiel: Einfaches HTML-Dokument

```
<html>
```

```
<body>
```

```
<h1>Überschrift</h1>
```

```
<b>
```

```
<p>Ein kompletter Absatz.</p>
```

```
</b>
```

```
</body>
```

```
</html>
```

✓ Q1:

Realisieren Sie eine der beiden (oder beide) HTML-Dokumente in einem beliebigen Texteditor (notepad, notepad++, gedit, editor) und nennen Sie die Datei bspw. `mypage.html` (im Dateinamen Leerzeichen eher vermeiden).

- Legen Sie im Verzeichnis, das von `phpinfo` in `DOCUMENT_ROOT` genannt ist (siehe vorheriger Aufgabenblock zu `localhost`), einen Unterordner (z.B. `webdev/`) an.
- Kopieren/verschieben Sie die HTML-Datei in das neu angelegte Verzeichnis.
- Öffnen Sie diese Datei in Ihrem Web-Browser unter <http://localhost/neuesVerzeichnis/neueDatei.html>

✓ Q2:

Ändern Sie das HTML-Dokument, bauen Sie Tippfehler ein und beobachten Sie, wie sich die Anzeige der Seite im Web-Browser ändert.

Ändern Sie HTML tags (z.B. `<H1>` statt `<h1>` oder `<bdy>` statt `<body>`) und fügen Sie Leerzeilen/-zeichen hinzu.

Wie reagiert der Web-Browser? Wie ändert sich die Darstellung?

✓ Q3

Besuchen Sie die Web-Seite

http://w3schools.com/html/tryit.asp?filename=tryhtml_intro und nutzen Sie den Editor für die Beispiele aus Q1.