COMP519 Web Programming
Lecture 3: HTML (HTLM5 Elements: Part 1)
Handouts

Ullrich Hustadt

Department of Computer Science
School of Electrical Engineering, Electronics, and Computer Science
University of Liverpool

Contents

HTML5 Documents

- An HTML5 document has a very simple form:
  It consists of a DOCTYPE-declaration and an `html`-element

```
<!DOCTYPE html>
html-element
```

- An `html`-element has the form

```
<html>
head-element
body-element
</html>
```

- It is recommended that the start tag of an `html`-element specifies the language used in the document

```
<html lang="en-GB">
```

Head

- The head-element should include a title-element
  (typically appears in the (tab) title bar of a browser)
- The head-element should also include meta data
  such as the author of the page, a description of its content, keywords
- The head-element can also include Cascading Style Sheet (CSS) definitions or links to external style sheets
- The head-element can also include JavaScript code or links to files containing such code

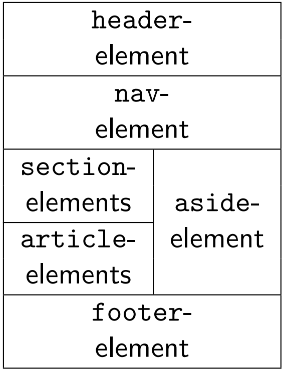
```
<head>
  <title>The Highway Code</title>
  <meta charset="UTF-8">
  <meta name="author" content="John Doe">
  <meta name="description" content="Rules of the UK Highway Code">
  <meta name="keywords" content="british,highway,highways,car,pedestrian">
  <link rel="stylesheet" href="default.css">
  <script src="code.js"></script>
</head>
```

Body

- The body-element contains the content that is to be displayed by a web browser including
  - Articles, sections, footers, and navs
  - Headings
  - Paragraphs
  - Lists and tables
  - Images
- The body-element may contain PHP code that is executed by the web server, producing HTML markup, that is then merged with the other content before being send to a web browser
- The body-element may contain JavaScript code that reacts to events in the web browser and can dynamically change the content

Structuring the Body

- The `main-element` contains the main content
- An `article-element` contains text that makes sense on its own
- A `section-element` contains text on the same theme
- A `header-element` contains introductory text for a document, article, or section
- A `footer-element` typically contains the author of the document, copyright information, links to terms of use, contact information, etc
- A `nav-element` contains a set of navigation hyperlinks
- An `aside-element` contains related but independent content to the articles/sections



Several of these could be in one body-element

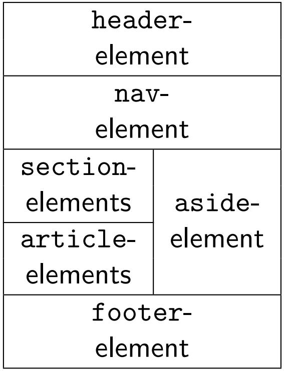
The elements are semantic, not layout related

Structuring the Body

article-elements and section-elements are typically nested inside each other:

- In an HTML document corresponding to a scientific paper one expects several section-elements (for introduction, conclusion, etc) inside one article-element
- In an HTML document corresponding to a newspaper one expects several article-elements (one for each report/story) inside one section-element

  The whole newspaper would consist of several section-elements (sport, business, etc), possibly inside a main-element



Headings

- Sections are meant to be organised into a hierarchy (not necessarily using nested `section`-elements)
- The hierarchy can be up to six levels deep
- The heading elements `h1` to `h6` allow to specify a heading for a section at the corresponding level, with `h1` being the highest level and `h6` the lowest
- Web browsers typically use font-size and font-weight to distinguish between headings at different levels

```
<h1>Fruit</h1>
    <h2>Apples</h2>
        <h3>Colour</h3>
        <h3>Taste</h3>
    <h2>Oranges</h2>
        <h3>Colour</h3>
        <h3>Taste</h3>
```

`http://cgi.csc.liv.ac.uk/~ullrich/COMP519/examples/headings.html`

Structure and Headings

```
<body>
  <main>
    <article>
      <header>
        <h1>Temporal Logic Reasoning</h1>
        <address>Ullrich Hustadt, University of Liverpool, UK</address>
      </header>
      <nav>
        <ul><li><a href="#Intro">Introduction</li>
            <li><a href="#Exp">Experiments</li></ul>
      <nav>
      <section id="Intro">
        <h2>Introduction</h2>
      </section>
      <section id="Exp">
        <h2>Experiments</h2>
        <section>
          <h3>Experimental Setup</h3>
        </section>
        <section>
          <h3>Observations</h3>
        </section>
      </section>
      <footer> &copy;2019 Ullrich Hustadt </footer>
    </article>
  </main>
</body>
```

Structure and Headings

```
<body>
  <header>
    <h1>Daily Newspaper</h1>
  </header>
  <nav>
    <ul><li><a href="#News">News</li>
        <li><a href="#Sport">Sport</li></ul>
  <nav>
  <main>
    <section id="News">
      <h2>News</h2>
      <article>
        <h3>First News Item</h3>
      </article>
      <article>
        <h3>Second News Item</h3>
      </article>
    </section>
    <section id="Sport">
      <h2>Sport</h2>
      <article>
        <h3>Third News Item</h3>
      </article>
    </section>
  </main>
  <footer> &copy;2019 Ullrich Hustadt </footer>
</body>
```

Lists

There are three different types of lists:

- Ordered list: `ol`-element with `li`-elements as content

```
<ol>
<li>Item 1</li>
<li>Item 2</li>
</ol>
```

  Typically uses numbers or letters to label each item in the list

- Unordered list: `ul`-element with `li`-elements as content

```
<ul>
<li>Item 1</li>
<li>Item 2</li>
</ul>
```

  Typically uses bullet points to label each item in the list

- Definition list: `dl`-element typically with pairs of `dt`-elements and `dd`-elements as content

`http://cgi.csc.liv.ac.uk/~ullrich/COMP519/examples/lists.html`

Lists

There are three different types of lists:

- Ordered list: `ol`-element with `li`-elements as content

  Typically uses numbers or letters to label each item in the list
- Unordered list: `ul`-element with `li`-elements as content

  Typically uses bullet points to label each item in the list
- Definition list: `dl`-element typically with pairs of `dt`-elements and `dd`-elements as content

```
<dl>
<dt>Internet</dt>
<dd>is a physical network of networks<dd>
<dt>World Wide Web</dt>
<dd>is a collection of interlinked multimedia documents</dd>
</dl>
```

`http://cgi.csc.liv.ac.uk/~ullrich/COMP519/examples/lists.html`

Paragraphs

- A paragraph is a group of sentences that is centred on a single idea
- HTML5 provides the p-element for paragraphs

```
<p>This Web site provides clients, customers, interested parties and
    our staff with all of the information that they could want on
        our   products,   services,   success   and   failures.
</p>
```

- Several spaces within a paragraph will always be rendered as just one
- Line breaks will not be preserved
- The void element `br` can be used to force a line break
- Alignment will be determined by the style that applies
  (typically, by default, paragraphs are only left-aligned)
- The p-element should not be used when a more specific element is more appropriate

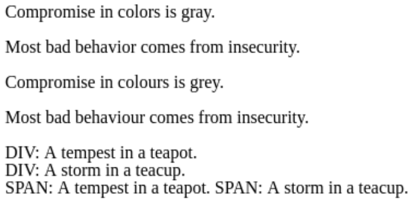
Div and Span

- The div-element and the span-element are used as containers for a group of consecutive elements
- A common semantics or a common style can then be applied to all elements of that container

```
<div lang="en-US">
<p>Compromise in colors is gray.</p>
<p>Most bad behavior comes from insecurity.</p>
</div>
<div lang="en-GB">
<p>Compromise in colours is grey.</p>
<p>Most bad behaviour comes from insecurity.</p>
</div>
<div lang="en-US">DIV: A tempest in a teapot.</div>
<div lang="en-GB">DIV: A storm in a teacup.</div>
<span lang="en-US">SPAN: A tempest in a teapot.</span>
<span lang="en-GB">SPAN: A storm in a teacup.</span>
```

Div and Span

```
<div lang="en-US">
<p>Compromise in colors is gray.</p>
<p>Most bad behavior comes from insecurity.</p>
</div>
<div lang="en-GB">
<p>Compromise in colours is grey.</p>
<p>Most bad behaviour comes from insecurity.</p>
</div>
<div lang="en-US">A tempest in a teapot.</div>
<div lang="en-GB">A storm in a teacup.</div>
<span lang="en-US">A tempest in a teapot.</span>
<span lang="en-GB">A storm in a teacup.</span>
```



```
Compromise in colors is gray.

Most bad behavior comes from insecurity.

Compromise in colours is grey.

Most bad behaviour comes from insecurity.

DIV: A tempest in a teapot.
DIV: A storm in a teacup.
SPAN: A tempest in a teapot. SPAN: A storm in a teacup.
```

The difference between `div` and `span` is that by default:

- `span`-elements are phrasing content (HTML4: inline content)
  - $\leadsto$ Two consecutive `span`-elements are placed side-by-side
  - $\leadsto$ `span`-elements have neither width nor height
- `div`-elements are floating content (HTML4: block content)
  - $\leadsto$ Each `div`-element starts on a new line and ends a line
  - $\leadsto$ `div`-elements have width and height

Paragraphs, Divs and Lists

- List elements cannot be children of p-elements

Wrong:

```
<p>The body-element of an HTML document may include
<ul>
  <li>headings and
  <li>paragraphs
</ul>
as well as many other things.</p>
```

Better (maybe only slightly):

```
<p>The body-element of an HTML document may include</p>
<ul>
  <li>headings, and
  <li>paragraphs
</ul>
<p>as well as many other things.</p>
```

Best:

```
<div>The body-element of an HTML document may include
<ul>
  <li>headings, and
  <li>paragraphs
</ul>
as well as many other things.</div>
```

Address

- The address element represents contact information for a person organization
- It is one of the few elements in which the use of a br element makes sense though paragraph, span or div could also be used

```
<address>
Dr Ullrich Hustadt<br>
Department of Computer Science<br>
University of Liverpool<br>
Email: U.Hustadt@liverpool.ac.uk
</address>
```

Hyperlinks

- Hyperlinks are created using

  ```
  <a href="url">text</a>
  ```

  where *text* is what the web browser will show to the user and *url* is the URL of a web page / resource that the web browser would visit if the user clicks on *text*

- The a-element has an optional attribute `target`
  Possible values include
  - `_blank`:
    Opens the linked web page in a new window or tab
    With HTML5 alone it is not possible to force whether a window or a tab is opened
  - `_self`:
    Opens the linked web page in the same window or tab (default)

```
<a href="http://cgi.csc.liv.ac.uk/" target="_blank">CS Website</a>
```

Hyperlinks

- Instead of a whole document, a URL can also refer to a particular element within a document, provided that element has an id
- In HTML5 any element can be given an id via the `id` attribute:

```
<tagName id="ID"> ... </tagName>
```

  where *ID* is non-empty sequence of characters without spaces, unique within the document
- It is then possible to internally link to that element using

```
<a href="#ID">text</a>
```

- It is also possible to externally link to that element using

```
<a href="url#ID">text</a>
```

  assuming *url* is the URL of the document containing the element with id *ID*

Hyperlinks

`http://w3.f.org/f.html`

```
<!DOCTYPE html>
<html lang="en-GB">
<head>
 <title>Document A</title>
</head>
<body>
  <h1>Fruit</h1>
  <h2 id="a">Apples</h2>
  <h3>Colour</h3>
  <h3>Taste</h3>
  <h2 id="o">Oranges</h2>
  <h3>Colour</h3>
  <h3>Taste</h3>
</body>
</html>
```

`http://www.cb.com/b.html`

```
<!DOCTYPE html>
<html lang="en-GB">
 <head>
  <title>Document B</title>
 </head>
 <body>
  <h1>Fruit</h1>
   <h2 id="p">Peaches</h2>
    <h3>Colour</h3>
    <h3>Taste</h3>
   <h2 id="o">Other</h2>
    <a href="http://w3.f.org/f.html#a">
    Apples</a>,
    <a href="http://w3.f.org/f.html#o">
    Oranges</a>.
    <a href="#p">Peaches</a>
    were covered above.
 </body>
</html>
```

See `http://cgi.csc.liv.ac.uk/~ullrich/COMP519/examples/fruit.html`
and `http://cgi.csc.liv.ac.uk/~ullrich/COMP519/examples/links.html`

Revision and Further Reading

Read

- Chapter 4: Creating a Simple Web Page
- Chapter 5: Marking Up Text
- Chapter 6: Adding Links

of

J. Niederst Robbins: Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics (5th ed).
O'Reilly, 2018.
E-book `https://library.liv.ac.uk/record=b5647021`