HTML

Gökçe Aydos

Lecture

Goals

- know the basic HTML, XML, CSS, Markdown syntax
- be able to use XML to structure information
- be able to create documents using HTML and Markdown

Theme

- HTML is a key part of World Wide Web
- HTML based on XML
- HTML can be read both by computers and humans (if well-formatted)

Page description languages

- content-oriented: ASCII
- structure+content-ori.: XML, HTML
- structure+content+form-ori.: Word (WYSIWYG[1]), LaTeX (after compilation)
- form: CSS

[1] what you see is what you get

Definition

- *Hypertext Markup Language*
- *markup language*
  - language for annotating a document
  - origin from marking up of manuscripts by editors with a red pencil
- *not* a programming language
- web browsers interpret HTML and render it

hypertext & hyperlink

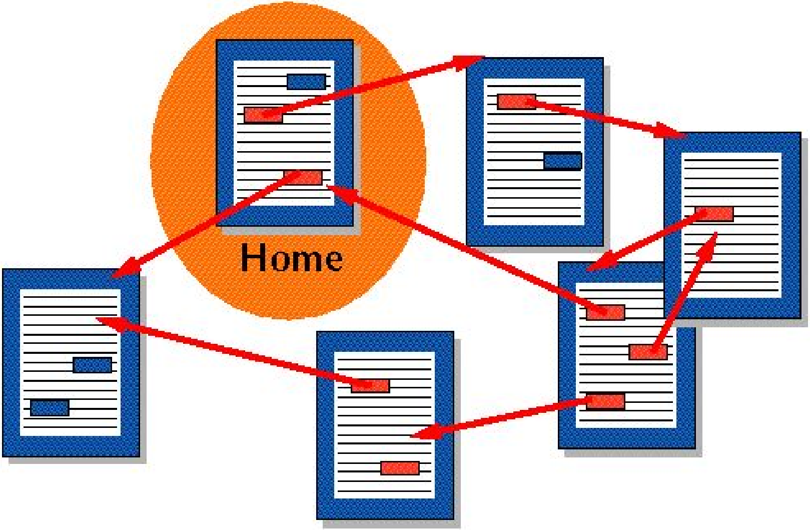

Figure 1: Andreariverac, CC BY-SA 3.0

HTML example

```
<p>
This is a <i>paragraph</i>
</p>
```

Demo on a web browser

HTML tags

e.g., `<title>`, `<img>`

- HTML consists of *elements*
- a *tag* encloses typically content:

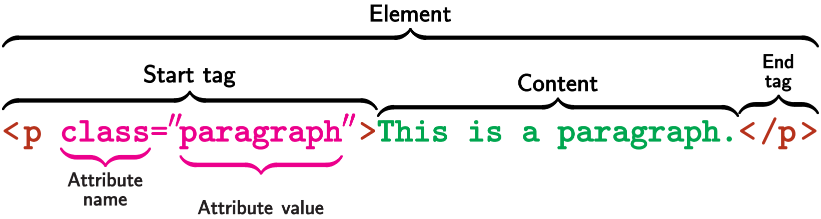

Figure 2: Wikipedia, CC BY-SA 3.0

HTML tags II

- some tags do not have to be closed, e.g., `<br>`, `<img>`.
- elements can contain other elements

```html
<p>You can force<br> line breaks with br element</p>
<p>If you close<br></br> br tag, it won't hurt</p>
<img src=05_html_element_components.svg>
```

HTML tags - exercise

How many *elements* does the following page contain?

```
<p>
  I like to
  eat <i>cheese</i>.<br>
  <ul>
    <li> Paneer </li>
    <li> Gouda </li>
  </ul>
</p>
```

HTML - page components

- `<html>` denotes whole document
- `<head>` header part containing title and metadata
- `<title>`
- `<body>` starts actual contents

Body structure tags

- `<h1>` - `<h6>` for headings of various levels
- `<p>` for paragraphs
- `<ul>`, `<ol>` for *unsorted/sorted lists*
- `<li>` for list items
- `<table>`: `<tr>` for table rows, `<td>` for table data (cells)
- `<div>` starts a division, which can be styled by CSS

Block vs inline elements

- *block* elements always starts on a new line, e.g., former body structure tags
- *inline* elements can be used in middle of a line and can end in another line

Inline elements

- `<i>` italics text
- `<b>` bold text
- `<strong>` strong importance (typically displayed bold)
- `<span>` similar to `<div>` but for spanning inline line parts

HTML bughunt - exercise

Find two mistakes:

```
<html>
  <head>
    <title>
      Example site
    </title>
  </head>
  <body>
    <h1> Main title </h1>
    <p>
      Here is an <strong>ordinary</strong> text.
    </p>
    <h2> Subsection </h1>
    <p>
      Namaste!
  </body>
</html>
```

HTML attributes

attributes are parameters for tags, e.g., a hyperlink:

```
<a href="https://wikipedia.org">Wikipedia</a> is a
helpful resource.

You can reach me via
<a href="mailto:hello@paneer.com">mail</href>
```

Further HTML tags

you cannot introduce additional HTML tags, e.g., `<cheese>`, `<friend>`. The HTML tags are defined by the HTML standard. Current tags can be found on Mozilla MDN - HTML elements reference

XML

- HTML is not extensible
- *extensible markup language* (XML)
- XML enables definition of your own tags. But you still have to define how they should be *presented* (form)
- still, XML is useful for structuring data

HTML vs XML

```
Kaiserschmarrn as main dish for one person:
<ol>
  <li> 100g flour </li>
  <li> 1/8l milk </li>
  <li> three eggs </li>
  <li> sugar and a pinch of salt </li>
</ol>
```

HTML vs XML II

```
Kaiserschmarrn as main dish for one person:
<recipe>
  <ingredient> 100g flour </ingredient>
  <ingredient> 1/8l milk </ingredient>
  <ingredient> three eggs </ingredient>
  <ingredient> sugar and a pinch of salt </ingredient>
</recipe>
```

XML - exercise

Describe the structure of an atom, e.g., helium, oxygen, in XML syntax

CSS

cascading style sheets (CSS): describes the form of a document

```
...
<head>
<style>
body {
  background-color: black
}
h1 {
  color: red;
}
</style>
...
```

CSS II

can also be put in an external file:

```
..
<head>
<link> rel="stylesheet" href="dark.css">
</head>
```

look at what CSS is capable at CSS Zen Garden

CSS - Exercise

How can we make the li elements *italic* and *grey*?

```
<p>
  I like to
  eat <i>cheese</i>.<br>
  <ul>
    <li> Paneer </li>
    <li> Gouda </li>
  </ul>
</p>
```

Why do we use CSS?

- HTML has also presentational elements

  ```
  <h1 style="color: red;"> Chapter 1. </h1>
  ```

  disadvantage: you have to change the color attribute everywhere an `<h1>` is used
- with CSS you only have to change your style

Markdown & reStructuredText

lightweight mark-up languages describing *content* and *structure*

- Markdown Cheatsheet
- A ReStructuredText Primer

Javascript adds behavior

for behavioral elements, interactivity

```
...
<button type="button"
  onclick="document.getElementById('my-p').innerHTML=
           Date()"
>Surprise me!</button>
<p id="my-p">my paragraph</p>
...
```

Javascript - Exercise

add a button in your web page which shows a surprise message

Summary

- HTML is an important part of World Wide Web
- HTML describe content and structure, XML also
- HTML and XML are based on tags
- Markdown & reStructuredText are lightweight mark-up languages
- Javascript can add *behavior* to documents