

Web Programming

HTML

Leander Jehl | University of Stavanger

Thanks & Credits to

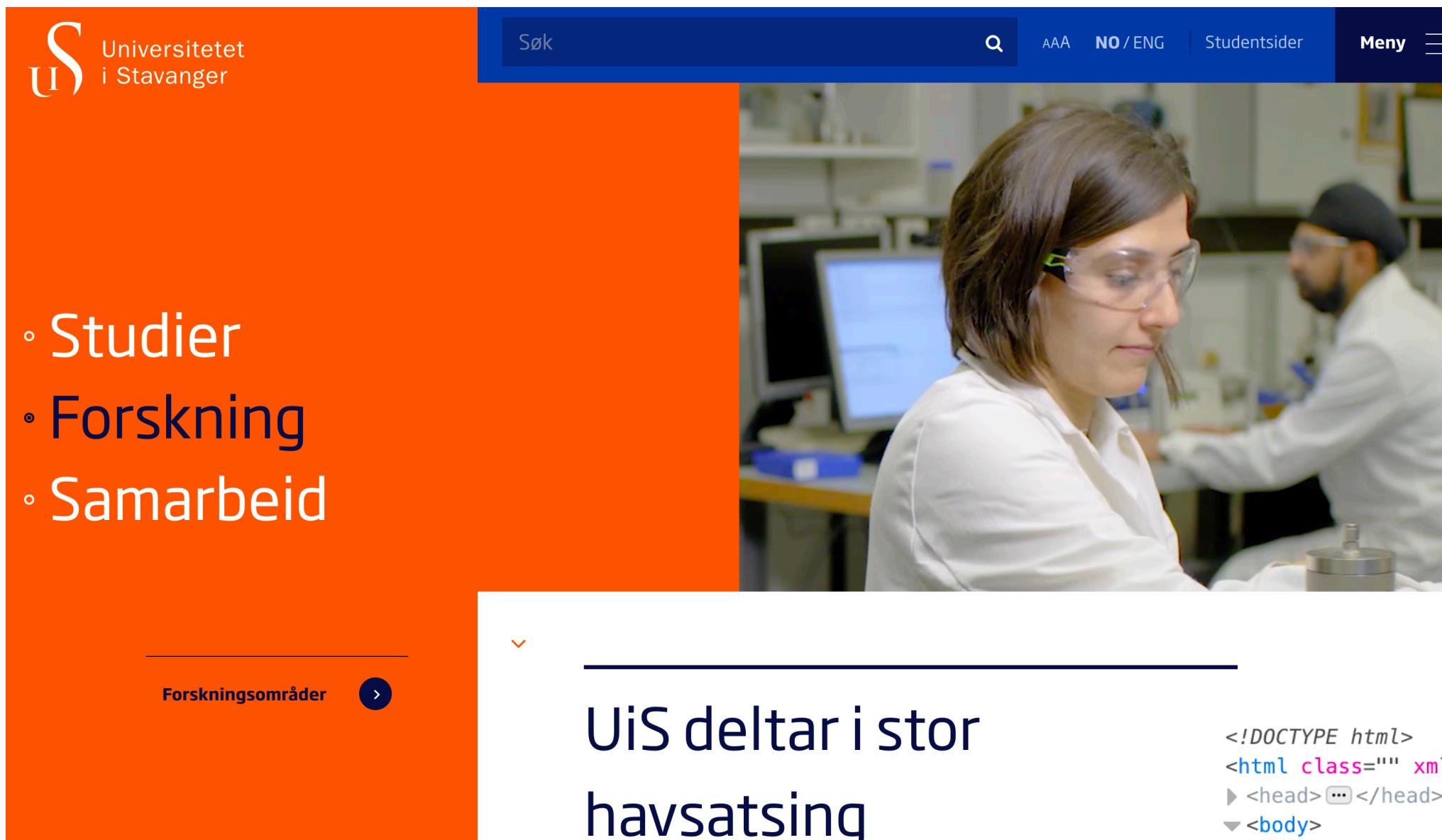
Krisztian Balog

For creating these slides and other materials for this lecture.

What is it?

- **HyperText Markup Language**
 - Language for creating web pages
 - Text surrounded by markings that specify the meaning of the content
 - Make the document displayable on different systems (different browsers)
- **Division of responsibilities for a web document**
 - HTML describes the content and structure
 - Style sheets (CSS) describe the appearance
 - Scripts (e.g., JavaScript) describe the behavior

HTML page rendered in the browser



HTML source

```
<!DOCTYPE html>
<html class="" xml:lang="no" lang="no"> event
  > <head>[...]</head>
  > <body>
    <!--Page Header-->
    > <header class="pageHeader pageHeader--page-home grid-container">[...]</header>
    <!--END: Page Header-->
    <!--Page Main-->
    > <main id="content" class="pageHome">
      > <div class="pageHomeSwipeNav" style="touch-action: pan-y; -moz-user-select: none;">[...]</div> event
      > <i id="FrontpageSwipeArrowRight" class="FrontpageSwipeIndicator">[...]</i>
        <span>
          <i id="FrontpageSwipeArrowLeft" class="FrontpageSwipeIndicator">[...]</i>
          <section id="Studier" class="pageHomeSection pageHomeSection--blue" data-block-theme="blue" data-block-title="Studier" data-cta-title="Se studietilbud" data-cta-href="http://www.uis.no/studier/" data-video-src="https://www.uis.no/getfile.php/13454231/Forside%20-%20video/20180823%20UiS%20Fadderbarn2_v4.mp4" data-video-image="https://www.uis.no/getfile.php/13454232/Forside%20-%20video/...til%20ny%20forside%202018/20180824%20-%20Campus.Still001.jpg" data-video-mobile-src="https://www.uis.no/getfile.php/13454231/Forside%20-%20video/20180823%20UiS%20Fadderbarn2_v4.mp4" data-video-mobile-image="https://www.uis.no/getfile.php/13454232/Forside%20-%20video/...til%20ny%20forside%202018/20180824%20-%20Campus.Still001.jpg" data-pos="1">[...]</span>
        <!--Section Studier-->
        > <section id="Forskning" class="pageHomeSection pageHomeSection--orange selected" data-block-theme="orange" data-block-title="Forskning" data-cta-title="Forskningsområder" data-cta-href="http://www.uis.no/forskning-og-ph-d/vi-forsker-paa/" data-video-src="https://www.uis.no/getfile.php/13402371/Forside%20-%20video/06%20Lab%20-%20kvinnene%20i%20forgrunn%20-%2015%20sek-.mp4" data-video-image="https://www.uis.no/getfile.php/13414087/Forside%20-%20video/...%20kvinnene%20i%20forgrunn%20-%2015%20sek-.mp4_00_00_00.Still001.jpg" data-video-mobile-src="https://www.uis.no/getfile.php/13402371/Forside%20-%20video/06%20Lab%20-%20kvinnene%20i%20forgrunn%20-%2015%20sek-.mp4" data-video-mobile-image="https://www.uis.no/getfile.php/13413375/Forside%20-%20video/...%20Stills_06%20Lab%20-%20kvinnene%20i%20forgrunn%20720x800.jpg" data-pos="2">[...]
        > <a class="pageHomeSection__down" href="#hovedartikkkel" title="Se ledige semesteremner høsten 2017">[...]</a> event
      > <div class="grid-x"> flex
        > <article id="hovedartikkkel" class="pageHomeSection__article cell small-12 large-10 grid-x"> flex
          <!--Tittel-->
          <!--h1 class="cell small-12 space-small-m-bottom-4">UiS deltar i stor havsatsing</h1-->
          <h1 class="cell small-12">UiS deltar i stor havsatsing</h1>
        </article>
      </div>
    </main>
  </body>
</html>
```

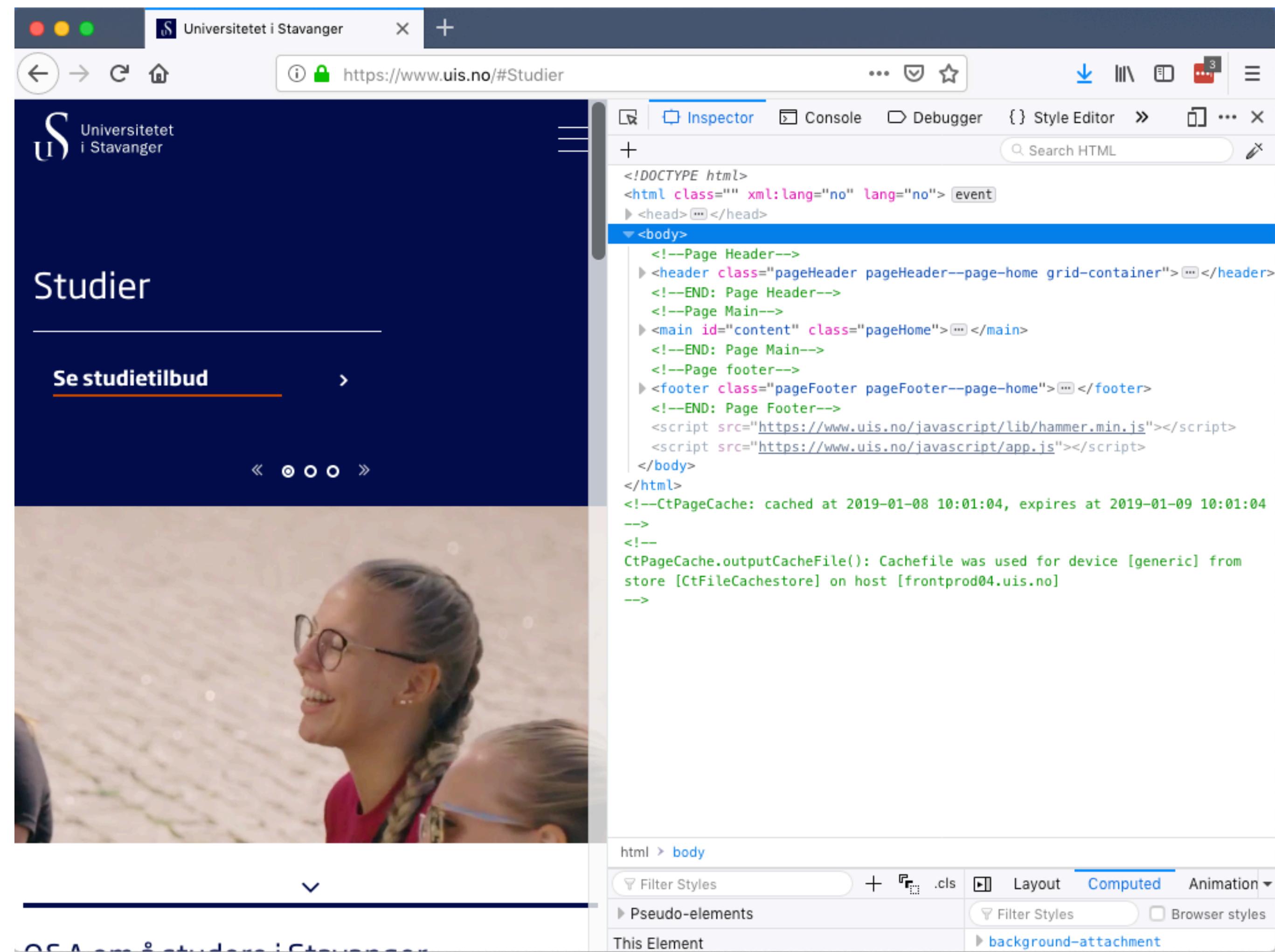
View HTML document in developer tools

Mac:

⌘ + ⌘ + i

Windows/Linux:

Ctrl + ⌘ + I



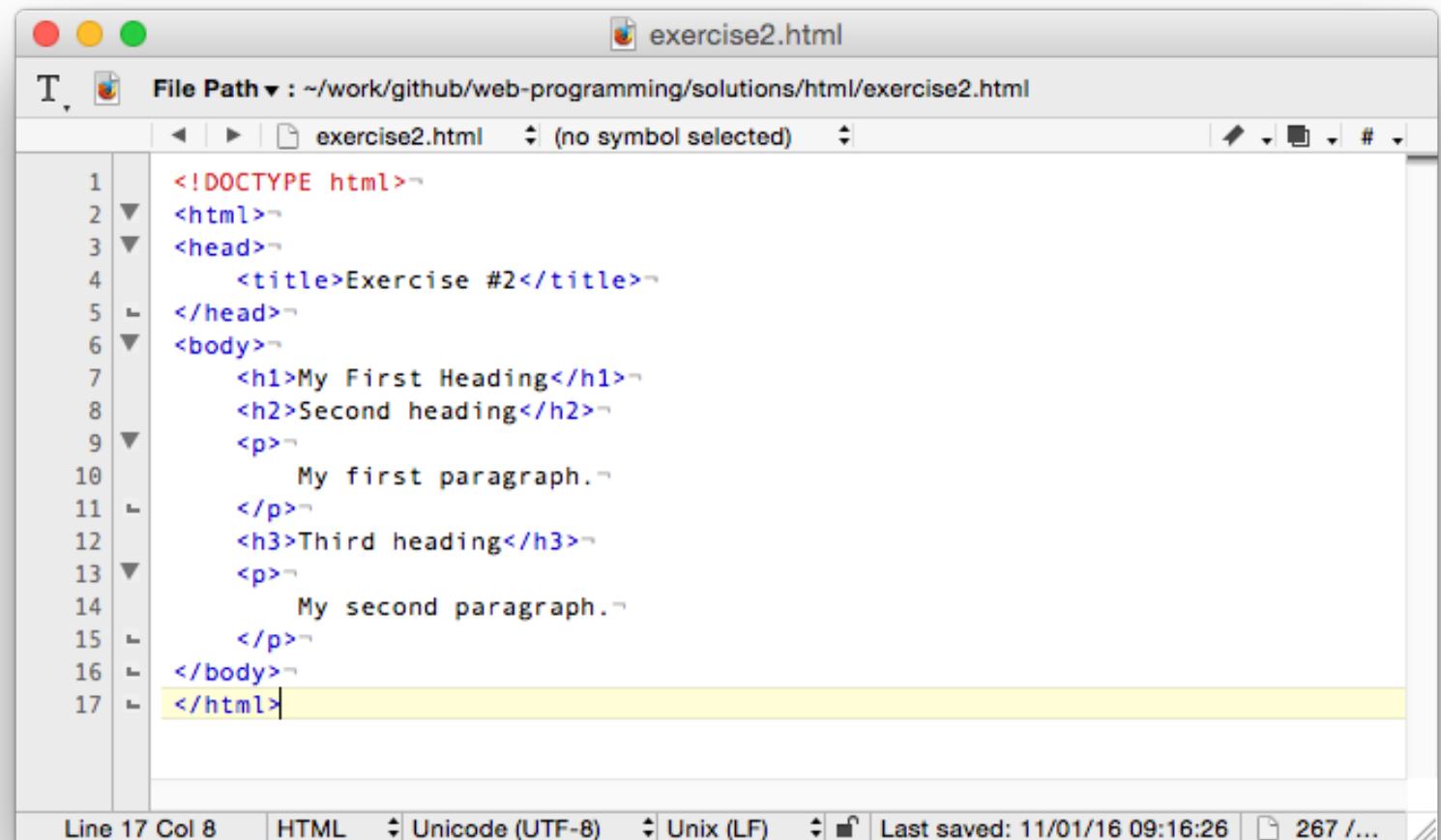
History

- 1991: HTML created by Tim Berners-Lee
- 1993: HTML standardized
- 1997: HTML 3.2
- 1999: HTML 4.01
- 2000-01: XHTML
 - Reformulation of HTML 4.01 using XML
 - No longer being developed
- 2014: HTML5



HTML "programming"

Programming environment
any text editor

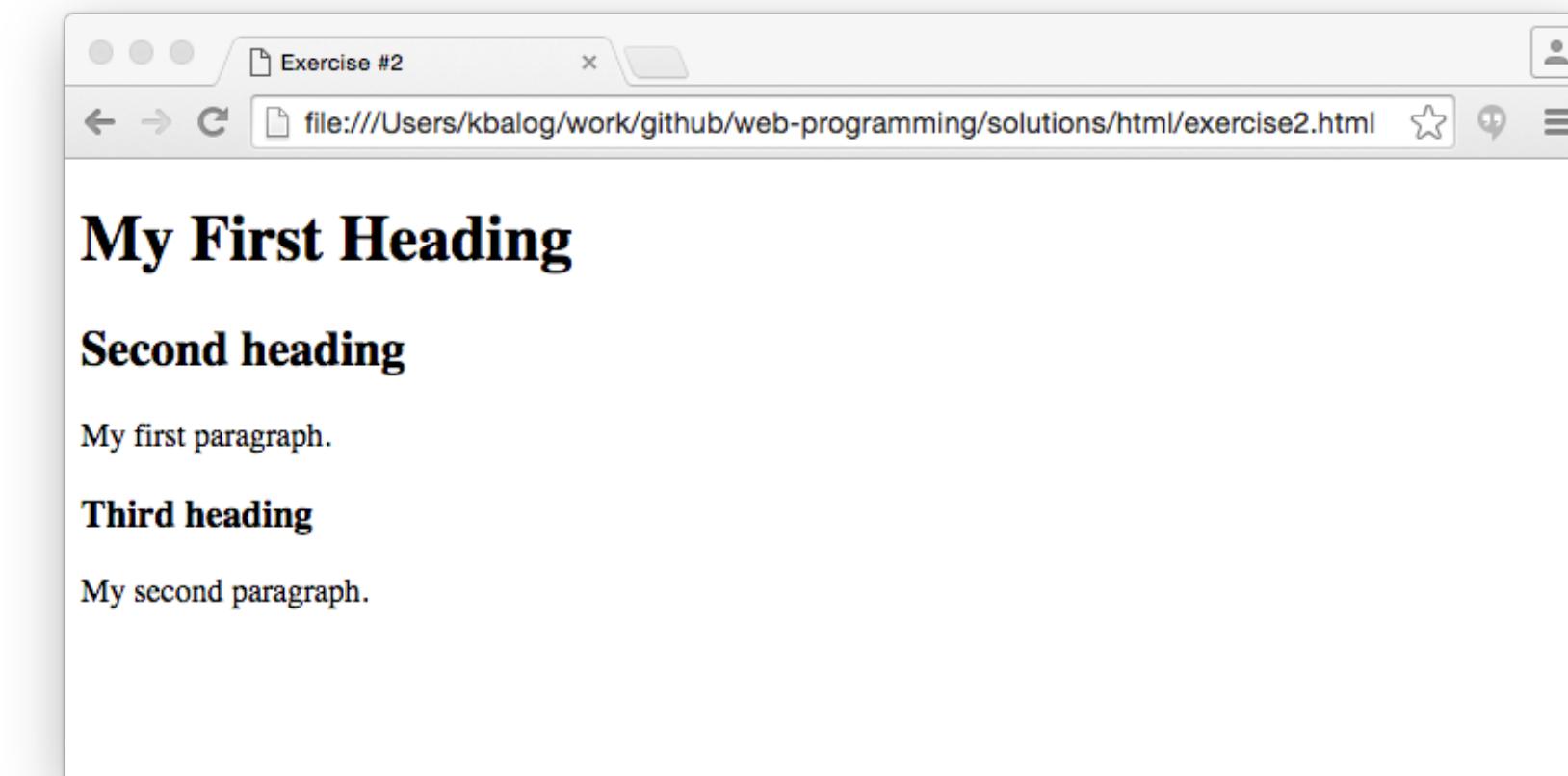


A screenshot of a Mac OS X-style text editor window titled "exercise2.html". The file path is listed as "/work/github/web-programming/solutions/html/exercise2.html". The code editor displays the following HTML structure:

```
<!DOCTYPE html>
<html>
<head>
    <title>Exercise #2</title>
</head>
<body>
    <h1>My First Heading</h1>
    <h2>Second heading</h2>
    <p>
        My first paragraph.
    </p>
    <h3>Third heading</h3>
    <p>
        My second paragraph.
    </p>
</body>
</html>
```

The status bar at the bottom shows "Line 17 Col 8" and "Last saved: 11/01/16 09:16:26".

Execution environment
any web browser



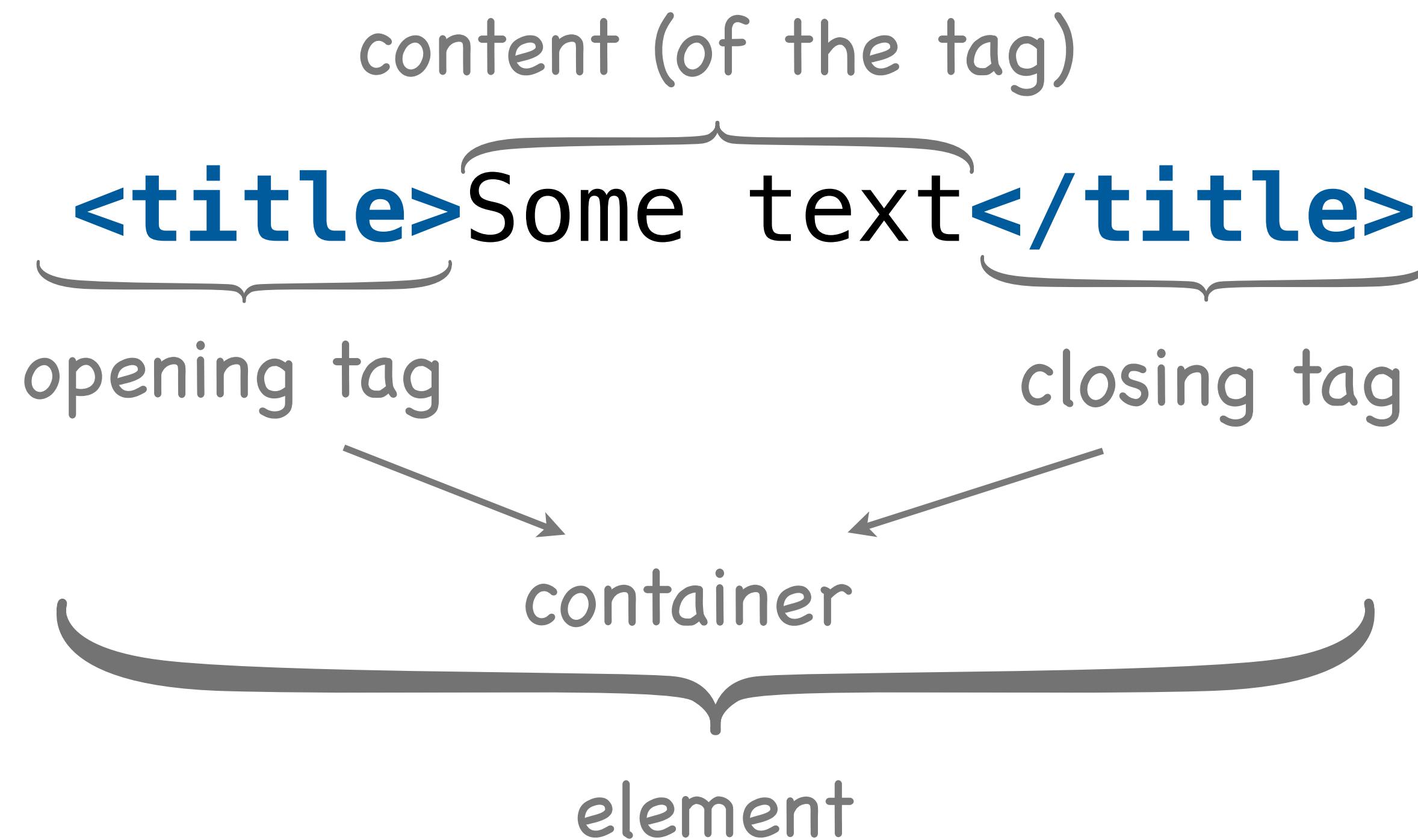
Syntax

- Fundamental syntactic units are **<tags>**
 - Specify type of content
 - Presentation is specific to the type of the tag
 - Most tags appear in pairs **<tag>...</tag>**
 - A few tags are self-closing **<tag />**
- Case insensitive but lowercase is recommended
- “Error handling” is left to the browser

Syntax (2)

- Tags without a closing pair
`<tag>` or `<tag />` (preferred)
- Comments are ignored by the browser
`<!-- comment text -->`
- White spaces are collapsed
- Indentation makes the source more readable
`<tag1>`
`<tag2>`
`<tag3>...</tag3>`
`</tag2>`
`</tag1>`

Terminology



Terminology

```

```

attribute
name

attribute
value

HTML Reference

www.w3schools.com/html

HTML Tag

[« Previous](#)

[Complete HTML Reference](#)

[Next »](#)

Definition and Usage

The tag defines an ordered list. An ordered list can be numerical or alphabetical.

Use the [](#) tag to define list items.

Browser Support

Element						
	Yes	Yes	Yes	Yes	Yes	Yes

HTML Reference

www.w3schools.com/html

HTML Tag

[« Previous](#)

[Complete HTML Reference](#)

[Next »](#)

Attributes

 = New in HTML5.

Do not use attributes that are no longer supported in HTML5!

Attribute	Value	Description
<u>compact</u>	compact	Not supported in HTML5. Specifies that the list should render smaller than normal
<u>reversed</u>	 reversed	Specifies that the list order should be descending (9,8,7...)
<u>start</u>	<i>number</i>	Specifies the start value of an ordered list
<u>type</u>	1 A a I i	Specifies the kind of marker to use in the list

HTML Reference

www.w3schools.com/html

HTML Tag

[« Previous](#)

[Complete HTML Reference](#)

[Next »](#)

Example

2 different ordered lists:

```
<ol>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>

<ol start="50">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

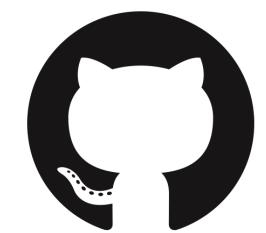
[Try it Yourself »](#)

Play with the examples!

Standard HTML document structure

```
<!DOCTYPE html>
<html>
  <head>
    ...
  </head> } document metadata
  <body>
    ...
  </body> } document content
</html>
```

Exercise #1

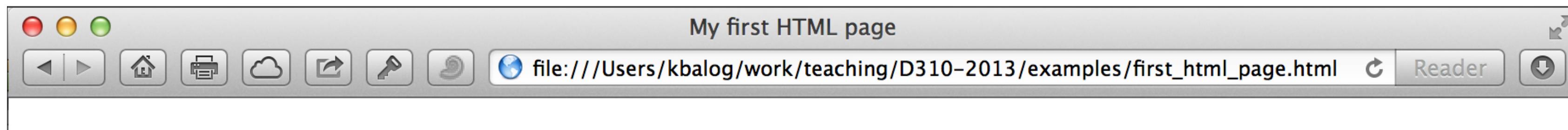


[github.com/dat310-spring20/course-info/tree/master/
exercises/html/basics](https://github.com/dat310-spring20/course-info/tree/master/exercises/html/basics)

<head> elements

- <title> is required
- <meta> tags are optional
 - Keywords, description, author

```
<head>
  <title>My first HTML page</title>
  <meta name="keywords" content="example, html">
  <meta name="description" content="Just a simple example">
  <meta name="author" content="Krisztian Balog">
</head>
```



Character encoding

- Indicate the character encoding used to write the page
- Use UTF-8 by default
- Best is to always include this line in the HTML head section

```
<head>
  <meta charset="UTF-8">
  <title>Title</title>
</head>
```

<body> elements

- Paragraphs
- Headings
- Text formatting
- Lists
- Tables
- Images
- Links

Paragraphs

<p>

- Documents are divided into paragraphs
 - Browsers automatically add an empty line before and after a paragraph
 - Use
 for a line break without starting a new paragraph

```
<p>This is a paragraph</p>
<p>Second paragraph. We force a linebreak here<br />then
continue.</p>
```

This is a paragraph

Second paragraph. We force a linebreak here
then continue.

Headings

`<h1>, <h2>, . . . , <h6>`

- Headings reflect document structure
 - `<h1>` is most important, `<h6>` is least important
 - Important for search engines

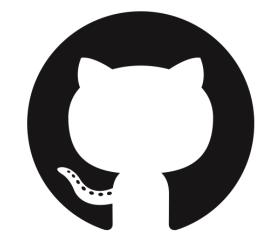
```
<h1>Heading 1</h1>
<h2>Heading 2</h2>
<h3>Heading 3</h3>
```

Heading 1

Heading 2

Heading 3

Exercise #2



[github.com/dat310-spring20/course-info/tree/master/
exercises/html/basics](https://github.com/dat310-spring20/course-info/tree/master/exercises/html/basics)

Phrase elements

,

- <**strong**> defines important text and is displayed in **bold** by default
- <**em**> defines emphasized text and is displayed in *italic* by default

```
<p>This is a paragraph. And we can make the text here  
<strong>bold</strong> or <em>emphasized</em>. <br />  
It is also possible to <strong><em>combine these</em>  
</strong> by nesting tags.</p>
```

This is a paragraph. And we can make the text here **bold** or *emphasized*. It is also possible to *combine these* by nesting the tags.

More text markup

<s>	Strikethrough
<small>	Smaller text
<sub>	Subscripted text
<sup>	Superscripted text
<mark>	Highlighted text

Character entities

 	Non-breaking space
<	<
>	>
€	€
©	©
®	®

Lines

<hr>

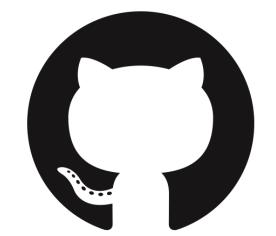
- Draw a horizontal rule
- No closing tag <hr> and <hr /> are both fine

```
<p>There is some text</p>
<hr />
<p>There is some more text</p>
```

There is some text

There is some more text

Exercise #3



[github.com/dat310-spring20/course-info/tree/master/
exercises/html/basics](https://github.com/dat310-spring20/course-info/tree/master/exercises/html/basics)

Lists

, ,

Ordered list

```
<ol>
    <li>Item 1</li>
    <li>Item 2</li>
    <li>Item 3</li>
</ol>
```

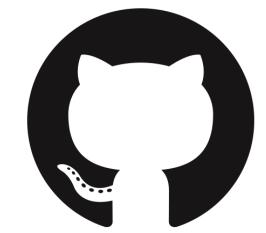
1. Item 1
2. Item 2
3. Item 3

Unordered list

```
<ul>
    <li>Item 1</li>
    <li>Item 2</li>
    <li>Item 3</li>
</ul>
```

- Item 1
- Item 2
- Item 3

Exercise #4 (#4b)



[github.com/dat310-spring20/course-info/tree/master/
exercises/html/basics](https://github.com/dat310-spring20/course-info/tree/master/exercises/html/basics)

Tables

<table>, <thead>, <tbody>, <tr>, <th>, <td>

```
<table border="1">
  <thead>
    <tr>
      <th>First name</th>
      <th>Last name</th>
      <th>Points</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>John</td>
      <td>Smith</td>
      <td>100</td>
    </tr>
    [...]
  </tbody>
</table>
```

First name	Last name	Points
John	Smith	100
Jack	Jackson	90
John	Doe	75

Merging cells

- Merging columns: **colspan="x"**

```
<td colspan="2">Content in the cell</td>
```

- Merging rows: **rowspan="x"**

```
<td rowspan="2">Content in the cell</td>
```

example

```
<table border="1">
  <tr>
    <td>Row 1, Col 1</td>
    <td>Row 1, Col 2</td>
    <td>Row 1, Col 3</td>
  </tr>
  <tr>
    <td>Row 2, Col 1</td>
    <td>Row 2, Col 2</td>
    <td>Row 2, Col 3</td>
  </tr>
  <tr>
    <td>Row 3, Col 1</td>
    <td>Row 3, Col 2</td>
    <td>Row 3, Col 3</td>
  </tr>
</table>
```

Row 1, Col 1	Row 1, Col 2	Row 1, Col 3
Row 2, Col 1	Row 2, Col 2	Row 2, Col 3
Row 3, Col 1	Row 3, Col 2	Row 3, Col 3

example (colspan)

```
<table border="1">
  <tr>
    <td colspan="2">Row 1,
Col 1-2</td>
    <td>Row 1, Col 2</td>
    <td>Row 1, Col 3</td>
  </tr>
  <tr>
    <td>Row 2, Col 1</td>
    <td>Row 2, Col 2</td>
    <td>Row 2, Col 3</td>
  </tr>
  <tr>
    <td colspan="3">Row 3,
Col 1-3</td>
    <td>Row 3, Col 2</td>
    <td>Row 3, Col 3</td>
  </tr>
</table>
```

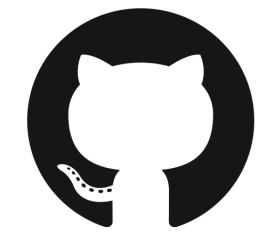
Row 1, Col 1-2	Row 1, Col 3
Row 2, Col 1	Row 2, Col 2
Row 3, Col 1-3	

example (rowspan)

```
<table border="1">
  <tr>
    <td rowspan="2">Row 1-2,  
Col 1</td>
    <td>Row 1, Col 2</td>
    <td rowspan="3">Row 1-3,  
Col 3</td>
  </tr>
  <tr>
    <td>Row 2, Col 1</td>
    <td>Row 2, Col 2</td>
    <td>Row 2, Col 3</td>
  </tr>
  <tr>
    <td>Row 3, Col 1</td>
    <td>Row 3, Col 2</td>
    <td>Row 3, Col 3</td>
  </tr>
</table>
```

Row 1-2, Col 1	Row 1, Col 2	Row 1-3, Col 3
	Row 2, Col 2	
Row 3, Col 1	Row 3, Col 2	

Exercise #5



[github.com/dat310-spring20/course-info/tree/master/
exercises/html/basics](https://github.com/dat310-spring20/course-info/tree/master/exercises/html/basics)

Links

<a>

- Clickable text to jump to another page
 - Absolute URL **href="http://www.uis.no"**
 - Relative to current page **href="page2.html"**

```
<p>There is some text with a <a href="http://www.uis.no">  
clickable link</a> in it</p>
```

There is some text with a clickable link in it

Links (2)

<a>

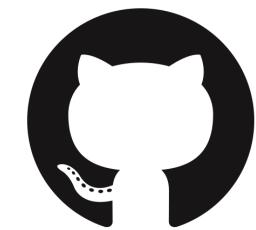
- Link to email address

```
<a href="mailto:info@uis.no">email us</a>
```

- **target="_blank"** makes the browser open the link in a new window

```
<a href="http://www.uis.no" target="_blank">UiS</a>
```

Exercises #6 and #7



[github.com/dat310-spring20/course-info/tree/master/
exercises/html/basics](https://github.com/dat310-spring20/course-info/tree/master/exercises/html/basics)

Links within a page

<a>

- **id** is a global attribute that can be defined for any HTML element
 - It needs to be unique, i.e., no two HTML elements can have the same id value
- Any element with an id can be linked within the page, e.g.,
 - Element to be linked

```
<h2 id="sec2">Section 2</h2>
```
 - Link to the element

```
<a href="#sec2">Section 2</a>
```

Example

⌚ examples/html/link_within_page.html

The screenshot shows a Mac OS X desktop environment with a web browser window titled "Example: Link and element". The browser's address bar displays the local file path: "file:///Users/kbalog/work/github/web-programming/examples/html/link_within_p...". The main content area of the browser shows two sections of text. The first section is titled "Section 1" and contains a paragraph of placeholder text (Lorem ipsum). The second section is titled "Section 2" and also contains a paragraph of placeholder text. A blue highlight box surrounds the entire "Section 2" title and its associated paragraph. To the right of this highlighted area, the corresponding HTML code is displayed in green and blue: "Section 2" above the section title, and "<h2 id="sec2">Section 2</h2>" above the paragraph. The browser interface includes standard controls like back, forward, and search.

[Section 2](#)

Section 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin lacinia lorem eu porttitor luctus. Sed ac mauris vitae arcu consectetur tempor sed congue felis. Vivamus posuere scelerisque dapibus. Nam ultricies laoreet augue id semper. Sed eu lacus ac dolor volutpat ullamcorper at sed risus. Nunc sollicitudin bibendum varius. Donec laoreet gravida metus. Fusce porta consequat augue, eget fermentum lectus molestie ultrices. Etiam sit amet sem sed massa maximus eleifend. Etiam eu maximus ipsum. Morbi ut venenatis mauris. Etiam vehicula nisl neque, fringilla pellentesque dui consequat quis. Nunc feugiat, massa non dapibus blandit, tellus diam gravida nisl, eu varius leo quam nec leo. Nunc fermentum mauris viverra purus elementum, at bibendum dui fringilla. Nam ac lacus a leo iaculis pellentesque rhoncus sed quam.

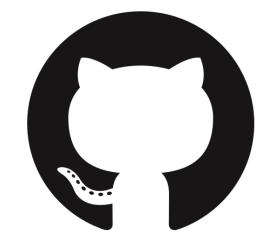
Section 2

[Etiam in ex quis eros hendrerit sollicitudin vitae iaculis neque. Vestibulum ornare nibh lectus, nec blandit justo](#)

```
<a href="#sec2">Section 2</a>
```

```
<h2 id="sec2">Section 2</h2>
```

Exercise #8



[github.com/dat310-spring20/course-info/tree/master/
exercises/html/basics](https://github.com/dat310-spring20/course-info/tree/master/exercises/html/basics)

Images

- Only attributes, no closing tag
- **src** can point to local or remote file (`http://...)`
- **alt** is alternative text if the image cannot be displayed
- **width** and **height** are optional
 - value in default is given in pixels
 - percentages can also be used (e.g., 50%)

```

```

Images - best practices

- Keep all images in a separate folder (e.g., images)
 - You can use further subfolders based on type or size
- Always use an alt tag
 - Screen readers read the alt tags for visually impaired
- Always specify width and height
 - To avoid the page flickering while the image loads

Images - best practices (2)

- Resize image files to the required size (e.g., for thumbnails and previews)
 - This makes your site load faster and reduces bandwidth

original jpg file (1000x1500px), 1MB



```

```

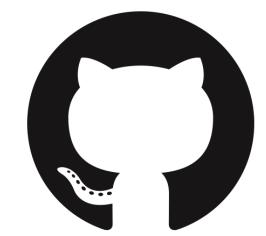
resized jpg file (300x200px), 55KB



```

```

Exercises #9 and #10



[github.com/dat310-spring20/course-info/tree/master/
exercises/html/basics](https://github.com/dat310-spring20/course-info/tree/master/exercises/html/basics)

Inline vs. block-level elements

- A **block-level element** always starts on a new line and takes up the full width available
 - Examples: <h1>,...<h6>, <p>, <div>, <form>
- An **inline element** does not start on a new line and only takes up as much width as necessary
 - Examples: <a>, , ,



block level



inline

Code and preformatted text

`<code>`, `<pre>`

- `<code>` is an *inline* element that designates a short piece of text as being source code

The `<code>System.out.println</code>` method produces console output in a Java program.

The `System.out.println` method produces console output in a Java program.

- `<pre>` is a *block-level* element that preserves all the whitespaces in its content exactly as written
 - By default rendered in a fixed-width font, but the content is still interpreted as HTML

Example

<pre>

```
<pre>
    public static void main(String[] args) {
        System.out.println("<b>Hello world</b>");
    }
</pre>
```

```
public static void main(String[ ] args) {
    System.out.println("Hello world");
}
```

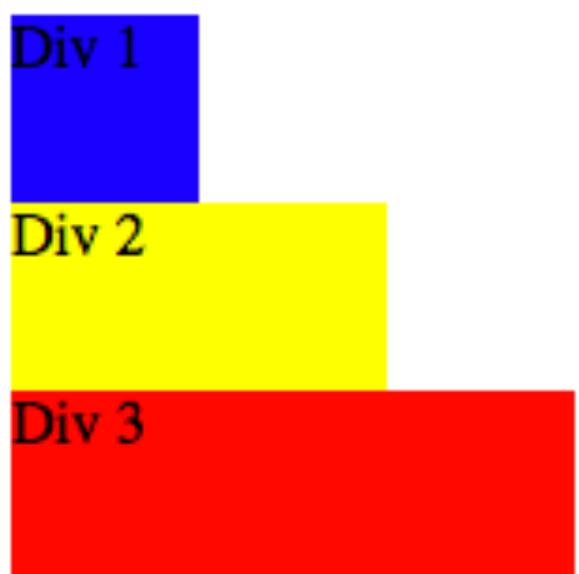
Content inside is still interpreted as HTML!

Container elements

- **<div>** defines a division or section in a page
 - Often used as a container for other HTML elements, to style blocks of content with CSS
- **** groups inline elements in a document
 - Often used as a container for some text

div example

```
<div style="width:50px; height:50px; background-color:blue">  
Div 1  
</div>  
  
<div style="width:100px; height:50px; background-color:yellow">  
Div 2  
</div>  
  
<div style="width:150px; height:50px; background-color:red">  
Div 3  
</div>
```



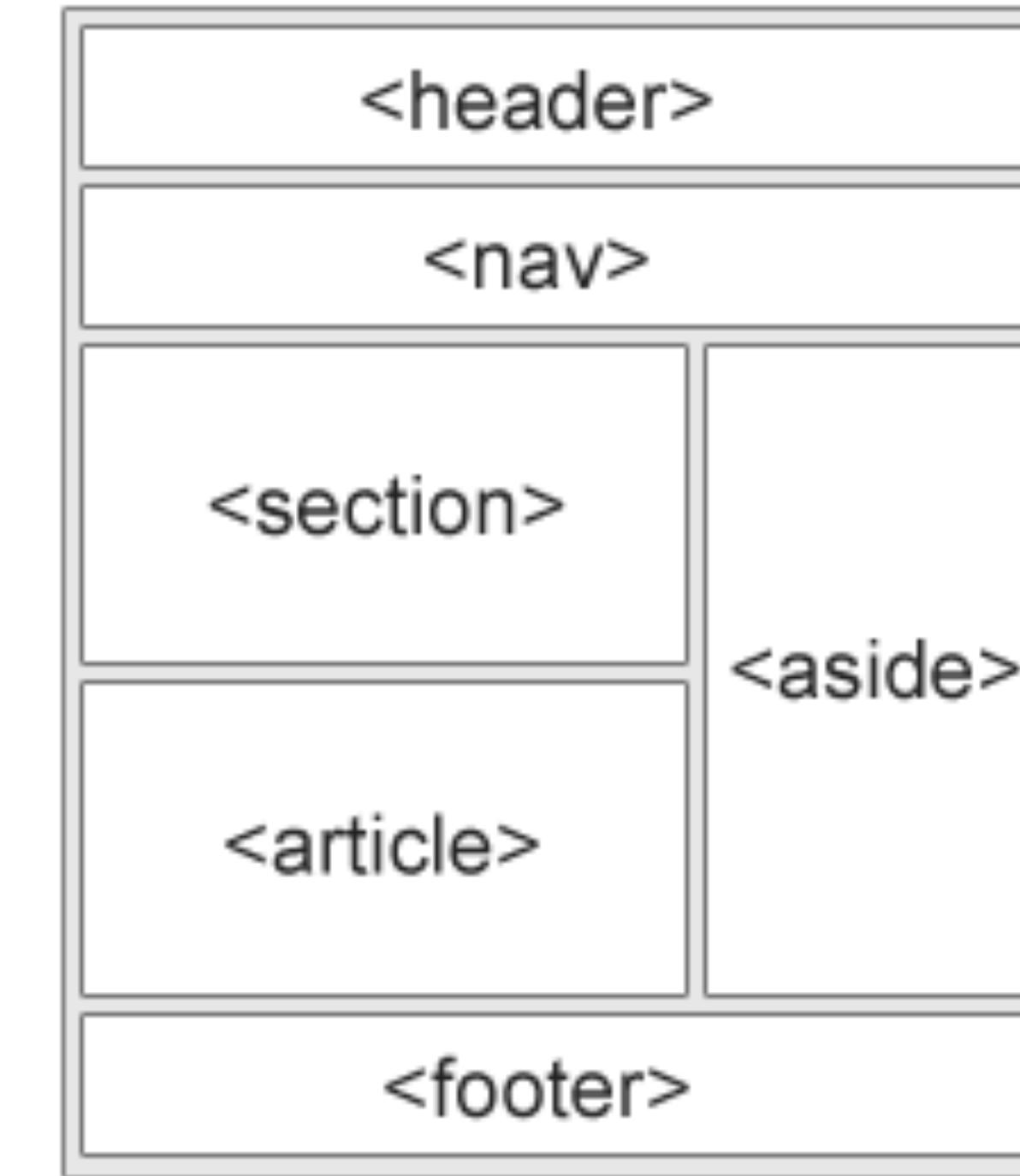
span example

```
<p style="font-family:Verdana;font-size:14px">
    Lorem Ipsum is <span style="font-family:Impact;font-size:
    18px">simply dummy text</span> of the printing and typesetting
    industry.
</p>
```

Lorem Ipsum is **simply dummy text** of the printing and typesetting industry.

HTML5 elements for document structure/semantics

- <article>
- <aside>
- <details>
- <figure>
- <footer>
- <header>
- <main>
- <menuitem>
- <nav>
- <section>
- <summary>



Iframe

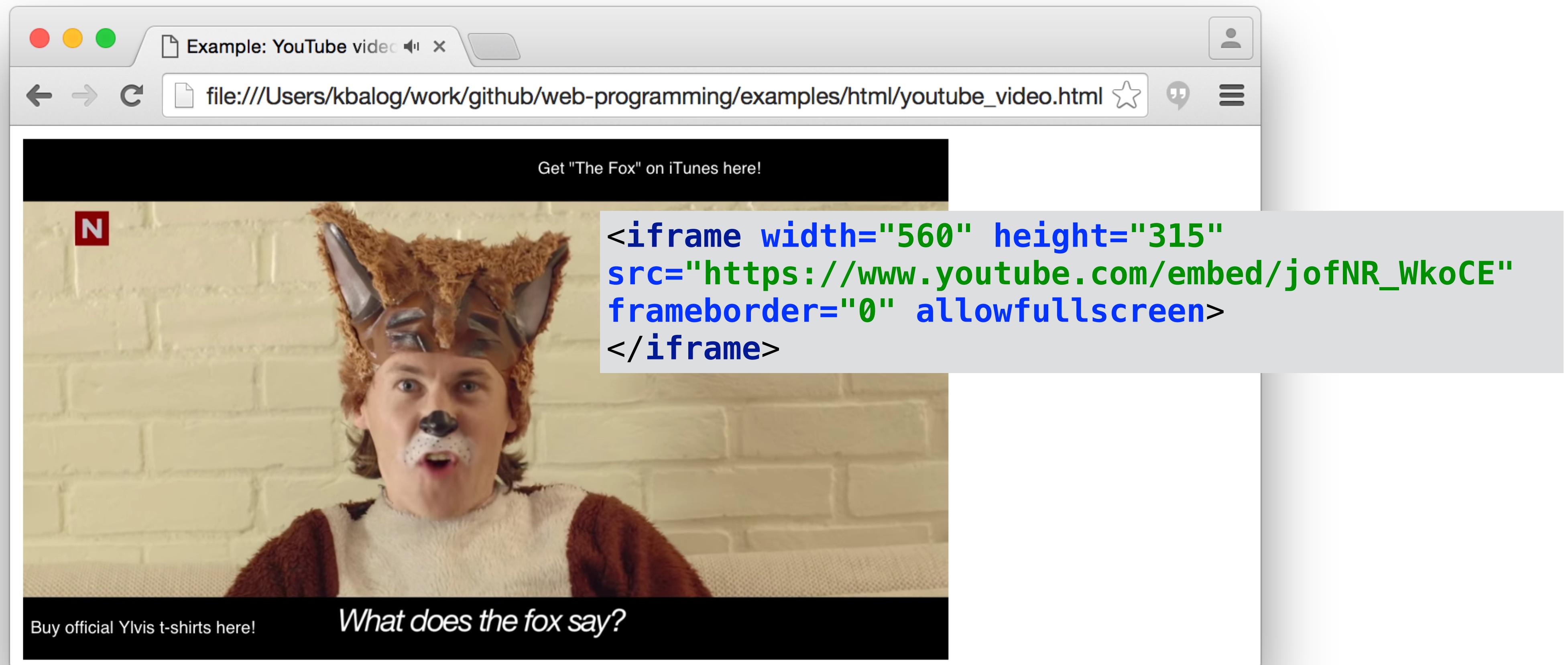
- **<iframe>** is used to display a webpage within a webpage

```
<iframe src="http://address.com"></iframe>
```

- **src** attribute (mandatory) specifies the address of the document to embed
- **height** and **width** attributes specify the size
- Recommended method for embedding YouTube videos

Example

examples/html/youtube_video.html



HTML5 audio and video

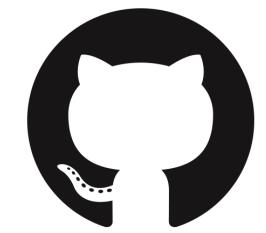
<audio>, <video>

- HTML5 provides standard ways to embed audio and video in web pages
 - Use .mp3 format for audios (supported by all browsers)

```
<audio controls>
  <source src="horse.mp3" type="audio/mpeg">
    Your browser does not support the audio element.
</audio>
```



Exercises #11 and #12



[github.com/dat310-spring20/course-info/tree/master/
exercises/html/basics](https://github.com/dat310-spring20/course-info/tree/master/exercises/html/basics)