# **H**yper**t**ext **M**arkup **L**anguage

Introduction to networking

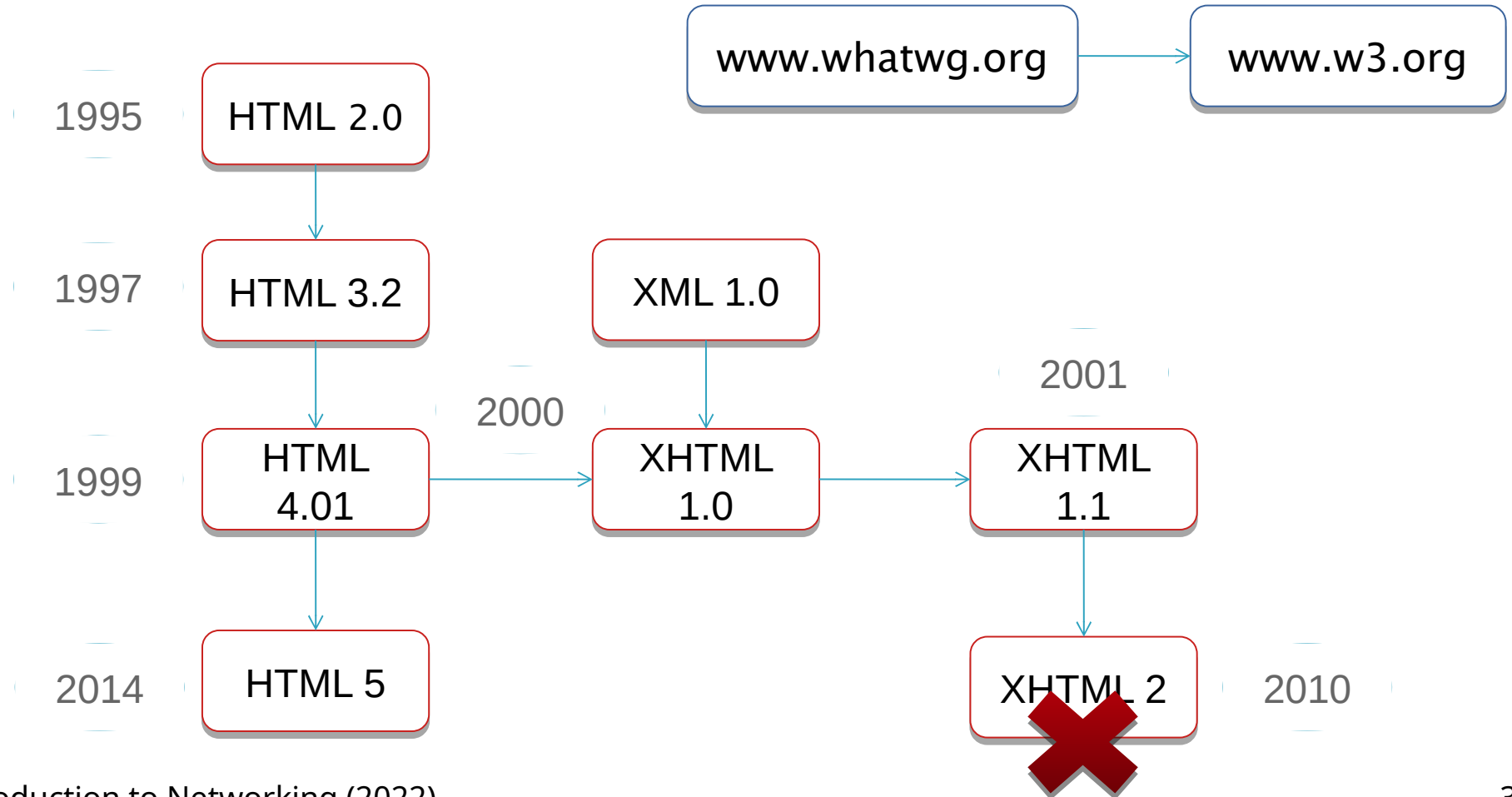Dr. Klára Pešková,  Klara.Peskova@mff.cuni.cz
Department of Software and Computer Science Education

# HTML – **H**yper**t**ext **M**arkup **L**anguage

- World Wide Web's markup language

- Language for describing the structure of Web pages


- Web page

  - Plain text marked by HTML tags that create the document structure

  - Tags add meaning to their contents, e.g. *heading, paragraph, link*

# Evolution of HTML

| www.whatwg.org | → | www.w3.org |

1995 — HTML 2.0

1997 — HTML 3.2

XML 1.0

2000

2001

1999 — HTML 4.01 → XHTML 1.0 → XHTML 1.1

2014 — HTML 5

XHTML 2 — 2010

# Evolution of HTML

- W3C

  - Director – Tim Berners Lee

  - Formal standardization process

- WhatWG (Web Hypertext Application Technology Working Group)

  - Html standard:

    - https://html.spec.whatwg.org/

# HTML source code

```html
<!DOCTYPE html>
<html>
    <head>
        <title>Iris</title>
    </head>

    <body>
        <h1>Iris versicolor (Blue flag)</h1>

        <p>
            Iris takes its name from the Greek word for a rainbow.
        </p>

        <img src="iris.jpg" alt="Blue flag">
    </body>
</html>
```

# HTML source code

```
<!DOCTYPE html>
<html>
    <head>
        <title>Iris</title>
    </head>

    <body>
        <h1>Iris versicolor ... </h1>

        <p>
            Iris takes its name ...
        </p>

        <img src="iris.jpg" alt="Blue flag">
    </body>
</html>
```
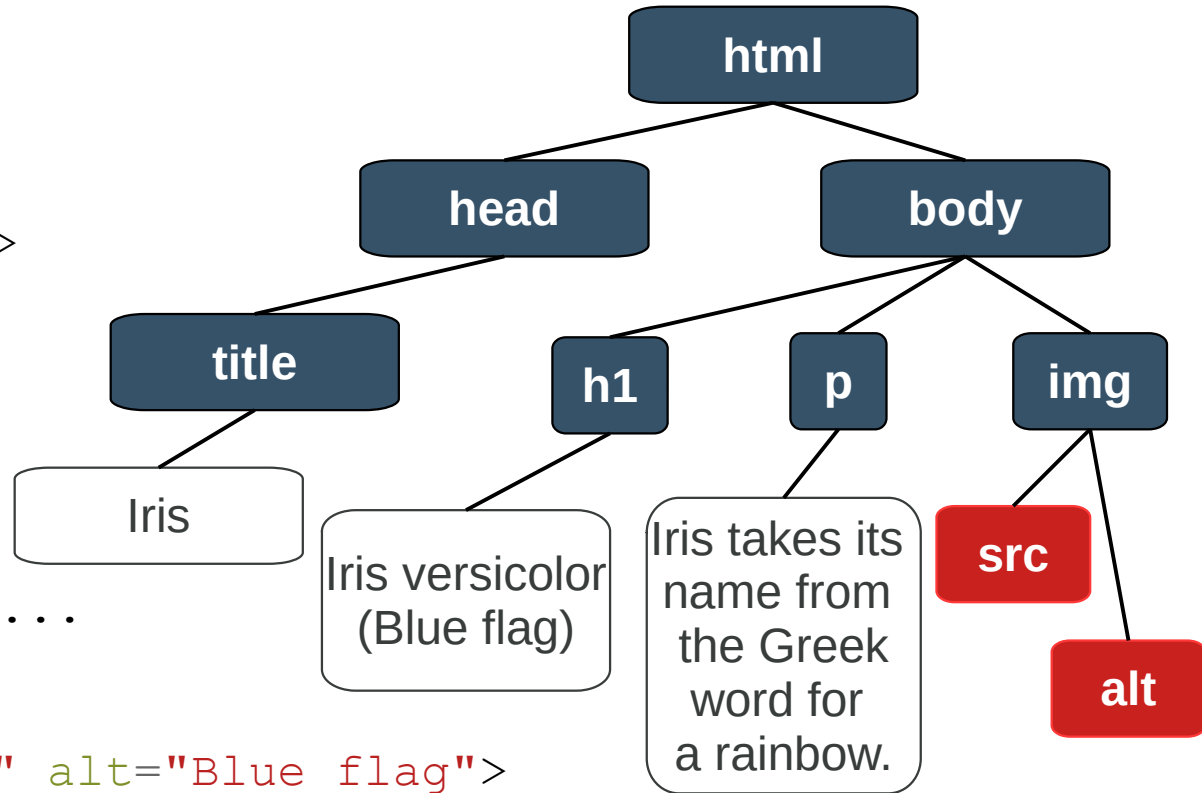
## Iris versicolor (Blue flag)

Iris takes its name from the Greek word for a rainbow.

# HTML syntax

- HTML document is structured as a tree

- Various types of nodes

  - Elements, text, attributes, comments, …

- HTML syntax is an infix serialization of the tree

- Tree is represented as Document Object Model (DOM) inside the browser

  - The DOM can be manipulated with JavaScript and plays role in CSS selectors evaluation

# DOM (Document Object Model) tree

```html
<!DOCTYPE html>
<html>
    <head>
        <title>Iris</title>
    </head>

    <body>
        <h1>Iris ... </h1>

        <p>
            Iris takes its ...
        </p>

        <img src="iris.jpg" alt="Blue flag">
    </body>
</html>
```

# HTML Body markup

- Inside element `<body>`

- Tags mark parts of the HTML document and add a meaning, e.g.

    - Text-level semantics elements

    - Sectioning the document

    - Grouping content

    - Tables, forms

    - External sources (images)

    - Hyperlinks

    - ...

# HTML syntax – HTML element

- A HTML element represents a fragment of a web page

  - Gives semantic meaning to its content

- Opening and closing tags work as parentheses for the content (i.e. boundaries of the element)

  - Content-less elements may omit closing tag → paired / singular or empty tags

  - Elements may not cross-overlap

- Attributes

  - Name-value pairs specified in an opening tag

  - Values are optional and they are optionally enclosed in single or double quotes (recommended)

    - Quotes are mandatory if the value contains ",', <, or >

```
<h1 id="about">Iris</h1>
<p>Iris takes its name from the Greek word for a rainbow.</p>
```

# HTML syntax

- Tags

  - Tag names in angle brackets **<>**

  - Closing tag name starts with **/**

- Whitespace characters do not play any role

- Comments

  - Enclosed in **<!--** and **-->**

  - Not displayed when page is rendered

- HTML entities: **&entity-name;**

  - Provide a way to encode special characters

```
<h1>Heading</h1>
```

```
<!-- comment -->
```

```
<  … &lt;
>  … &gt;
&  … &amp;
"  … &qout;
©  … &copy;
```

```
non-breaking space …  
numerically represented characters ü … &#252;
```

# Basic HTML element

- Heading

- Paragraph

- Link

- List

- Table

- Image

# Heading elements – `<h1>` – `<h6>`

- Before HTML5

  - Six levels of importance (rank)

  - `<h1>` most important (highest), `<h6>` least important

- Headings in HTML5

  - Combined with sectioning `<section>`, `<article>`, …

  - Attempt to keep some backwards compatibility

  - Quite difficult to do though

  - Each section has its own heading hierarchy

  - First heading element in a section is the main heading of that section (no matter its rank)

# Paragraphs

```
<p>A paragraph in HTML.</p>
```

# HTML lists – `<ul>`, `<ol>` elements

- Ordered or un-ordered

- Can be nested

  - day
  - month
    1. January
    2. February
    3. March
    4. ...
  - year

```
<ul>
    <li>day</li>
    <li>month
      <ol>
          <li>January</li>
          <li>February</li>
          <li>March</li>
          <li>...</li>
      </ol>
    </li>
    <li>year</li>
</ul>
```

# HTML tables – `<table>` element

`<table>` defines a table

`<caption>` defines a table caption (must be inserted immediately after the `<table>` tag)

`<thead>` groups the header content in a table (optional)

`<tbody>` groups the body content in a table (optional)

`<tfoot>` groups the footer content in a table (optional)

`<tr>` table row

`<td>` table cell

- a cell can span more than one row/column (attributes `colspan`, `rowspan`)

`<th>` header cell

# HTML table – example

Users

| E-mail | Name |
|---|---|
| smith@gmail.com | Adam Smith |
| joe@black.ml | Joe Black |
| me@black.ml | |

```
<table>
  <caption>Users</caption>
  <thead>
    <tr>
      <th>E-mail</th><th>Name</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>smith@gmail.com</td>
      <td>Adam Smith</td>
    </tr>
    <tr>
      <td>joe@black.ml</td>
      <td rowspan="2">Joe Black</td>
    </tr>
    <tr>
      <td>me@black.ml</td>
    </tr>
  </tbody>
</table>
```
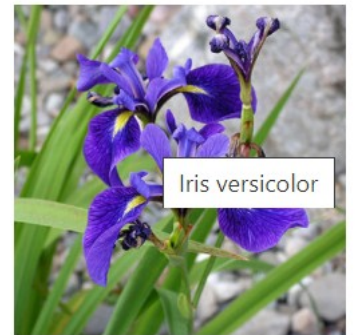
# HTML images – `<img>` element

- Singular element (has no content → closing tag can be omitted)

- Attributes

  - `src` – the URL of an image to be loaded

  - `alt` – alternative textual representation

    - Describes, what the image shows

    - Used, when the image loading fails, or for "non-visual" outputs

  - `title` – shows on hover

```
<img src="iris.jpg"
 alt="Blue flag"
 title="Iris versicolor">
```

# HTML hyperlinks – <a> element

- Links to external resources

- Attributes

  - `href` – specifies URL of linked resource

  - `target = "_blank"` – opens the link in a new window

```
Go to <a href="http://www.google.com">Google</a>
```

Go to <u>Google</u>

# HTML hyperlinks continued...

- Link to a web page fragment

    ```
    <a href="#net">Go to Networking section</a>
    <section id="net">Networking ...</section>
    ```

- Special links

    ```
    <a href="tel:+420603123456">603 123 456</a>
    <a href="mailto:klara@pisecko.cz">send e-mail</a>
    ```

- e.g. clickable image (put image into a link)

    ```
    <a href="http://www.seznam.cz">
        <img src="logo.gif">
    </a>
    ```

# Block vs inline elements

- Block-level element always takes up the full width available (stretches out to the left and right as far as it can)

  - It always starts on a new line

- Inline element only takes up as much width as necessary

  - It does not start on a new line

# HTML element types

- Semantic elements
  - describe the meaning of the HTML contents – to web browsers and web developers
  - e.g. `<nav>`, `<header>` ...

- Document structure
  - e.g. `<h1>`, `<p>` ...

- Text-level elements
  - e.g. `<em>`, `<strong>` ...

- General containers
  - e.g. `<div>`, `<span>`...
  - Tell nothing about their contents

# Text-level semantic elements

`<em>` represents stress emphasis of its content. The level of stress is given by the level of nesting of particular `em` elements

`<strong>` represents strong importance of its content

`<small>` represents a side comment

`<cite>` represents a title of a work (book, game, software, song, …)

`<code>` defines a piece of computer code

`<abbr>` an abbreviation or acronym, optionally with its expansion in `title` attribute

`<i>` indicates a different quality of text

`<b>` a content to which attention is being drawn

`<sub>` `<sup>` subscript and superscript

`<br>` a line break

# Sectioning content – Semantic HTML5 elements

`<section>` defines a section in a document (with a heading)

`<article>` independent, self-contained content; can be shared independently (newspaper article, blog post)

`<header>` a header for a document or section (intro content, set of navigational links, logo, …)

`<footer>` a footer for a document or section, copyright, contact info, back to top links, …

`<nav>` defines navigation links

`<aside>` content aside from the page content; its contents should relate to adjacent content

`<figure>` specifies self-contained content (illustrations, diagrams, photos, code listings, etc.)

`<figcaption>` a caption for a `<figure>` element
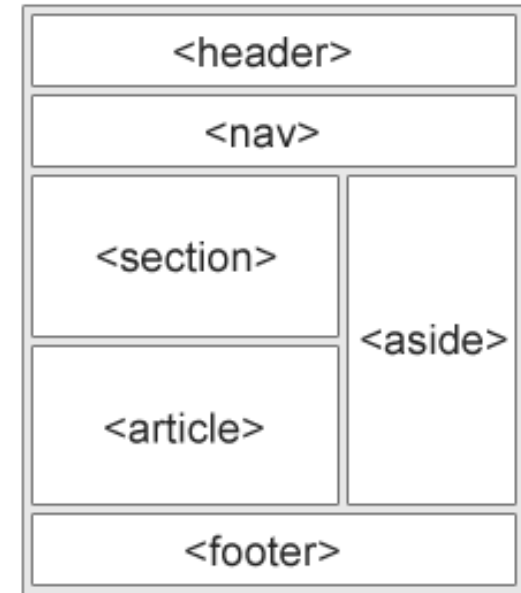
`<details>` additional details that the user can view or hide

`<summary>` a visible heading for a `<details>` element

`<main>` the main content of a document

`<mark>` marked/highlighted text

`<time>` defines a date/time

```
<header>
<nav>
<section>          <aside>
<article>
<footer>
```

# More HTML elements...

General containers:

`<div>` block element with no special meaning (generic container)

`<span>` inline element with no special meaning (generic container)

`<pre>` a block of preformatted text

`<dl>` a definition list

`<dt>`, `<dd>` definition term, definition description

# HTML forms

- A component composed of UI controls

```
<form action="script.php">

    <label for="name">Name:</label>
    <input type="text" id="name">

    <label for="pwd">Password:</label>
    <input type="password" id="pwd">

    <input type="submit" value="OK">

</form>
```

Name: [                    ]

Password: [                    ]

[ OK ]

Name: [ Klara             ]

Password: [ •••••••••••        ]
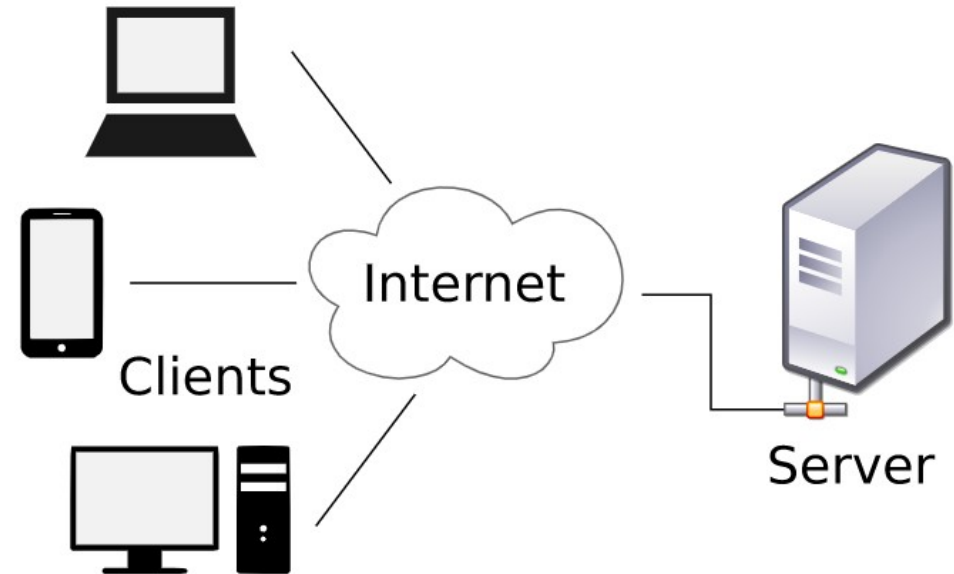
[ OK ]

# Submitting forms

- User fills in the data

- Form data are validated (optional)

- Data are sent to a server to be processes

- Data are pre-processed and validated on the server

- Server processes the data

- Response is sent to a client (web browser) as a HTML file



Internet

Clients

Server

# Form controls

- `<input>`

  - Basic input controls
  - Various types based on input attributes

- `<textarea>`

  - Input for longer (multi row) texts

- `<select>`

  - Selection list with `<options>`

- `<button>`

  - Submit or reset button

# Basic form components – `<input>`

```
<form>

    <input type="text">
    <input type="radio" name="group" checked>
    <input type="checkbox" checked>
    <input type="submit" value="OK">

    <input type="hidden">
    <input type="file">

    <input type="password">

    <input type="reset">
    <input type="button">

</form>
```

```
checked, disabled, required, maxwidth, size="num of chars"
```

Name: [                    ]

Passw  ⚠ Please fill out this field.

- ● male
- ○ female
- ○ other

I came here to study:

- ☑ HTML
- ☑ CSS
- ☐ JavaScript
- ☐ PHP
- ☐ Flash

Choose File  No file chosen

OK  Reset

# HTML5 inputs

```html
<form>
    <input type="color">
    <input type="date">
    <input type="datetime-local">
    <input type="month">
    <input type="email">
    <input type="number">
    <input type="range">
</form>
```

# HTML5 inputs – examples

Date picker

December 2020 ▾                    ↑    ↓

| Mo | Tu | We | Th | Fr | Sa | Su |
|----|----|----|----|----|----|----|
| 30 | 1  | 2  | 3  | 4  | 5  | 6  |
| 7  | 8  | 9  | 10 | 11 | 12 | 13 |
| **14** | 15 | 16 | 17 | 18 | 19 | 20 |
| 21 | 22 | 23 | 24 | 25 | 26 | 27 |
| 28 | 29 | 30 | 31 | 1  | 2  | 3  |
| 4  | 5  | 6  | 7  | 8  | 9  | 10 |

Today

Date and tim

Month picke

E-mail  klara.cz

⚠ Please include an '@' in the email address. 'klara.cz' is missing an '@'.

Color picker

| 255 | 226 | 36 |
|-----|-----|----|
| R   | G   | B  |

Number  6

Range

OK

# More form components

```
<select>
  <option>Ford</option>
  <option selected>Škoda</option>
  <option>Mercedes</option>
<select>

<textarea rows="3" cols="10">
</textarea>

<input list="cars">
<datalist id="cars">
  <option>Ford</option>
  <option>Škoda</option>
  <option>Mercedes</option>
</datalist>
```

# More on forms

```html
<label for="element_id">
    Element description
</label>

<fieldset>
  <legend>Fieldset heading</legend>
</fieldset>
```

Personal
Name [                    ]

Password [                    ]

○ male
● female
○ other

OK

- Form elements attributes:

`autocomplete="on"/"off"` **– can be set for the form or for single form controls**

`placeholder`

`title` - **a hint for the field ("bubble" with text)**

# HTML forms – `<form>` element

- Attributes

    - `method` – HTTP method used for data transfer (GET/POST)
        - `post` – data are transferred in HTTP request body
        - `get` – data are encoded in URL (query part) – visible to everyone
    - `action` – URL where the data are sent to

```
<form action="script.php" method="get">
    ...
</form>
```

# Submitting forms

- When a form is submitted, data are encoded into HTTP request and sent to the selected URL using the given method (GET/POST)

- Submit button works as a link to a new URL created based on filled form data

```
<form action="script.php" method="get">

    <input type="text" name="name">
    <input type="password" name="pwd">

</form>

=> URL: script.php?name=John&pwd=45ak!
```

# HTML Head – document metadata

- Additional information about the document

- Inside element `<head>`

  - `<title>` - document title or name (browser window caption)

  - `<link>` - links document to other resources, e.g. CSS files

  - `<style>` - embeds style (CSS) information inside the document

    - Contents is written in CSS stylesheet syntax

  - `<meta>` - additional metadata and HTTP supplements

    - Description, keywords, document author, …

# HTML Head – metadata - example

```html
<html lang="cs">
   <head>
     <title>Iris species</title>

     <link rel="stylesheet" type="text/css" href="styles.css">

     <style>
         body { color: red }
     </style>

     <meta name="author" content="Joe Black">
     <meta charset="utf-8">

   </head>
</html>
```

# More on links `<a>`, `<link>`

- `href` attribute

    - URL of resource linked by relationship

- `rel` attribute

    - Type of the relationship

- `media` attribute

    - Specified the media the linked resource applies to

    - E.g., `print`, `screen`, `all`

- `type` attribute

    - MIME type of linked resource

    - `text/html`, `application/xhtml+xml`, `text/css`, `application/pdf`

# More on links – examples

```html
<html>
  <head>
    <title>NSWI142 - Materials</title>
    <link rel="stylesheet" href="default.css"
          type="text/css" media="screen">

    <link rel="stylesheet" href="default-print.css"
          type="text/css" media="print">

    <link rel="icon" href="http://example.com/myicon.png">

  </head>

...
```

# Sources

- **W3c validator**
  - https://validator.w3.org/

- **HTML Tutorial at w3schools**
  - https://www.w3schools.com/html/