



# Web Development in 5 weeks

---

Tamas Piros

[tpiros.dev](http://tpiros.dev)



# Agenda

---

- **HTML & CSS**
- JavaScript (part 1)
- JavaScript (part 2)
- React (part 1)
- React (part 2)



# Agenda

---

- **HTML & CSS**
  - HTML5 for Modern Browsers
  - Introduction to CSS
  - CSS Preprocessors
  - CSS Frameworks



# HTML



# Internet History 101

---

- Tim Berners-Lee, 1993
  - “HTML Tags” - 18 elements, links based on SGML
- HTML 2 - 1995, HTML 3 - 1997 (Jan), HTML 4 - 1997 (Dec)
- HTML 5 - 2014, HTML 5.1 - 2016, HTML 5.2 - 2017
- XHTML



# Anatomy of an HTML file

```
1  <!doctype html>
2
3  <html lang="en">
4  <head>
5      <meta charset="utf-8">
6      <title></title>
7      <meta name="description" content="">
8      <meta name="author" content="Tamas Piros">
9      <link rel="stylesheet" href="">
10     <script src=""></script>
11 </head>
12
13 <body>
14 |
15 </body>
16 </html>
```

1: HTML declaration

3-11: “head” (metadata, CSS, scripts)

13: body



# Anatomy of an HTML file

---

<meta>: HTML element

charset: HTML attribute

```
<meta charset="utf-8">
```

“utf-8”: attribute value



# HTML5 Prominent Elements



# HTML5 Prominent Elements

---

- article
- aside
- audio
- bdi
- canvas
- data
- datalist
- details
- dialog
- embed
- fig caption
- figure
- footer
- header
- hgroup
- keygen
- main
- mark
- menuitem
- meter
- nav
- output
- picture
- progress
- rp
- ruby
- section
- source
- summary
- svg
- template
- time
- track
- video
- wbr



# <nav>

---

- Major block for navigation
- Multiple <nav> elements are OK to have
- Common to have it for page-wide navigation as well as in page navigation



# <header>

---

- Represents introductory content to a piece of content
- “Thematic grouping of content”
  - heading elements, intros, profile pic, logo, search form, author name, date(s), metadata, social links
- Can contain an image (e.g. logo), author information or any other element (except another header or a footer)
- Not to be confused with <**head**>



## <main>

---

- Represents the main corpus of text, the most important content within a page
- One document should only have one <main> element



## <footer>

---

- Should contain further author information, copyright data, links, footnotes, metadata, legalities, *low priority site nav*
- Can be overall footer (for the entire page) or for a given section



# <article>

---

- Self contained entity - article, page, forum document, blog entry, widget etc
  - Independently distributable
  - Makes sense even when taken out of context



# HTML 5 Multimedia



# <audio>

---

- Embed sound to HTML documents
- Vast number of *attributes* and event listeners are available
  - autoplay, controls, preload\*, src
- Note browser support for audio files and code types
  - mp3 vs ogg vs mp4
- Volume attribute is not supported (changeable via JS)
- \*preload values can be ‘none’, ‘metadata’ or ‘auto’. Note that the ‘autoplay’ attribute has precedence over ‘preload’.



# <video>

---

- Embed video in an HTML document
- Vast number of *attributes* and event listeners are available
- Note browser support for video files and code types
  - mp4 vs webm
  - Single or multiple source elements



# HTML 5 Semantic Elements



# <figure> and <figcaption>

---

- A figure should represent a content that consists of an image (illustration, diagram or other “snippet”) and a caption text
  - Caption text is optional
- Difference between <img> and <figure>?
  - Semantics!



# <progress>

---

- Indicates progress as a progress bar
  - Ideal for showing upload progress for example
  - Two available attributes: **value** and **max**
- Can have two states: determined and indeterminate
- Default styling (display) will be different based on browser / OS



# HTML 5 Canvas Element



# <canvas>

---

- Allows for (scriptable) rendering of shapes (2D, 3D - WebGL) as well as bitmap images
- JavaScript is used as scripting for the canvas
- circles, squares, arcs, lines, rectangles, different shapes, transparency, gradients, text
- Transformations (skew, rotate)
- Animations



# HTML 5 Input Type



# HTML 5 Input Type

---

- Client side validation and usability (+accessibility)
- Email, number, search, phone number, url, datetime, colour



# HTML 5 Web APIs



# Web APIs

---

- HTML5 really opened up the Web API space
- Geolocation, History, Audio, Video, Network Information API, Drag-n-Drop, Fetch, Web Speech API and others



# Introduction to CSS



# Introduction to CSS

---

- Cascading Style Sheets
- Apply style to HTML
- Set rules via “selectors”
  - Modular system (e.g. border styling, background styling)



# How to select an element?

---

- Single, multi line grouping
- Elements: single elements, multiple elements, all elements
- ID based selection
- Attribute / attribute value based selection
- “+”, “>”, “~” - single preceding element, direct child selector, select any child
- Pseudo selectors (“:visited”, “:hover”)



# CSS pre-processors



# CSS pre-processors

---

- CSS compiler - write CSS programatically, with less syntax
  - Sass (Syntactically Awesome Style Sheet) / SCSS (Sassy CSS)
  - LESS (Leaner Style Sheets)
  - Stylus
  - PostCSS



# CSS Frameworks



# CSS Frameworks

---

- Bootstrap (<https://getbootstrap.com>)
- Foundation (<https://get.foundation>)
- Bulma (<https://bulma.io>)
- Materialize (<https://materializecss.com>)
- TailwindCSS (<https://tailwindcss.com>)
- And many more ...