Week 1 Workshop

Web Development Fundamentals
HTML, CSS, and JavaScript

Agenda

Activity	Time
Get Prepared: Log in to Nucamp Learning Portal • Slack • Screenshare	10 minutes
Introductions	20 minutes
Course Setup	30 minutes
Week 1 Recap - Introduction to HTML	40 minutes
Workshop Assignment	20 minutes
BREAK	15 minutes
Workshop Assignment	75 minutes
Show and Tell	30 minutes

Introductions & Course Setup

Instructor Introduction

Student Introductions

- Students, please introduce yourselves. If your course is online, the instructor can call out the name of who should go next.
- Share:
  - Your name
  - If you will be going on to the Front End or Full Stack bootcamp after this one
  - What motivates you to take this bootcamp?
- Take no more than 2 minutes per person. You will have a chance to share more about yourself and get to know each other better through the Workshop Assignment.

Course Setup Checklist

Make sure that all students can say yes to the following:

Have you…	
Confirmed that you receive emails from Nucamp?	Installed VS Code on your laptop?
Installed or updated to the latest version of Chrome or Firefox?	Installed the Moodle app on your phone and connected to Nucamp with it?
Downloaded and extracted the NucampFolder to your Desktop?	Joined the Nucamp Slack workspace?
Discussed your time commitment pledge with your friends and family?	Read and agreed with the Code of Conduct?

Using Slack

- <u>Check:</u>
- Has everyone joined the three course & community channels in the Nucamp's Slack workspace? (1) course channel, 2) community channel, 3) community-course channel)
- Does everyone know how to send a Direct Message to the instructor?
- Does everyone know how to use message threads in channels?
- Does everyone know how to attach files?

Asking For Help in Slack Channels

- Remember these key points when asking for help in a Slack channel:
  - Ask, don't ask to ask (Instead of saying "Can anyone help me with a problem?", ask a direct question about your problem.)
  - Give detailed information about your issue, including what you have tried, and which specific exercise/task you are working on.
  - Attach your relevant code file(s).
  - If you are seeing error messages, attach a screenshot of the error.
  - Use message threads to prevent channel clutter, otherwise a conversation about one person's question can drown out other students' questions!

Your Learning Experience for this Bootcamp

- Next week's content is unlocked at the end of the week (Fridays)
- Daily tasks **every day (2+ hours)**:
  - View videos and follow along with the exercises
  - Complete any Challenges
  - Quiz at end of week, before workshop
  - Additional exploration
  - Prepare for workshop assignment
  - Help other classmates via class forums or Slack

- **4-hour workshop every weekend**
  - Read the assignment instructions beforehand
  - Bring your laptop (for in-person workshops)

IT WILL BE HARD
IT WILL IMPACT
YOUR ROUTINE

The 20-Minutes Rule

If you ask for help too soon:

- You will not learn how to tackle problems or remove obstacles, which is core to coding.
- You will likely not remember the process or path that resolved the issue, which is more important than the solution.

If you ask for help too late:

- You will get frustrated and tired.
- You will miss an opportunity to go deeper on the same topic (time is limited).

10 minutes rule during workshops

- During the week, go by the 20-minutes rule.
- During workshops, go by a 10-minutes rule – try to solve the issue yourself (or with your classmate, if working together) for 10 minutes before asking your instructor for help.

20 minutes, then ask for help!

Next 4 Weeks

- Week 1: This week! Course Setup & Introduction to HTML
- Week 2: More HTML and Introduction to CSS
- Week 3: Introduction to JavaScript
- Week 4: JavaScript and the DOM (Document Object Model)

Week 1 Recap

Introduction to HTML

Week 1 Recap - Overview

New Concepts This Week

- HTML vs CSS vs JavaScript
- History of HTML & WWW
- HTML Elements, Tags, Attributes
- Using VS Code & Extensions
- The DOCTYPE Declaration
- <html> <head> <body>
- <meta> <title>
- <p> <section>
-

- Heading Elements
- Ordered & Unordered Lists
- Semantic HTML
- Block-level vs Inline Elements
- Comments
- Anchors
- Relative vs Absolute Paths
- Embedding Images and Video

Next slides will review these concepts

HTML vs CSS vs JavaScript

- HTML (HyperText Markup Language)
  - First language of the World Wide Web, used to structure and give meaning to content
  - Markup language
- CSS (Cascading Style Sheets)
  - Used for style and positioning – colors, fonts, etc
  - Stylesheet language
- JavaScript
  - Used to create dynamic, interactive websites
  - The only one of the three that is a programming language

History of HTML and World Wide Web

- HTTP (HyperText Protocol), the underlying protocol of the Web, was invented in 1990 by scientist Tim Berners-Lee at CERN research lab
- HTML invented in 1991 by same person
- World Wide Web is ***not*** the same as the Internet
- The Internet includes many other protocols for communication beyond just the Web – FTP (file transfer protocol), SMTP (simple mail transfer protocol), etc
- HTML5 is the current version of HTML
- WHAT-WG and W3C are the organizations that govern HTML standards

Web Designers vs Web Developers

Discuss (ask for a student to answer):

- What do you understand to be the difference between web design and web development?
- Which you will be learning with us at Nucamp?

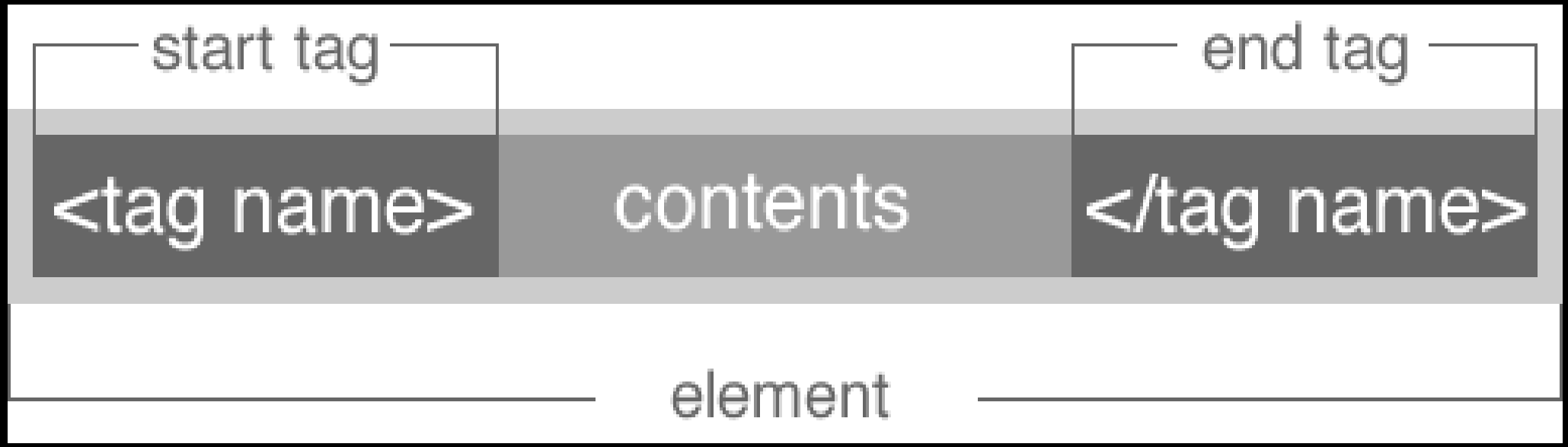
Front End vs Back End

- **Discuss** (ask for a student to answer):
- What do you understand to be the difference between front end and back end web development? What does full stack mean?
- Which you will be learning with us in *this* course?

Using Visual Studio Code

- Check:
  - Did everyone turn on AutoSave?
  - Did everyone get Live Server to work, or did anyone run into an issue?

Elements



The start and end tags are also called the opening and closing tags.

Void Elements

Void elements have only one tag, called "void" or "empty" because it doesn't hold content between start and end tags

Example: `<img src="myImg.jpg" />`

- The tag for a void element is called a self-closing tag because it's both an opening and closing tag in one.
- They can optionally end with `/>` but this is not required:
  - `<img src="myImg.jpg">` is correct as well.
- Void elements can have attributes.

Attributes

- All elements have a set of applicable attributes.
- Attributes are written in the start tag.
- They can be written as a key-value pair
  - e.g. attributeName="value"
  - The value can be in single or double quotes, be consistent
- Some attributes are Boolean - use only the name of the atttribute, such as controls
- If an element has multiple attributes, use spaces between each; the order does not technically matter
  - e.g. <video src="myVideo.mp4" loop controls> is the same as <video src="myVideo.mp4" controls loop>

Basic HTML5 Document Structure

```
<!DOCTYPE html>
<html>
<head>
   <meta charset="utf-8" />
   <title>My Webpage</title>
</head>
<body>
   <!-- Your webpage content goes here  -->
</body>
</html>
```

DOCTYPE Declaration

Not an element but a special declaration.

All HTML5 documents must start with <!DOCTYPE html>. This tells the browser that you are using HTML5, and it will render your document according to the HTML5 specifications.

If you don’t start with this declaration, modern browsers will automatically use a different rendering mode used for legacy HTML documents, and this may cause issues.

<html>

- Called the "root element", one per HTML document
- Place immediately after the DOCTYPE declaration
- Use the attribute lang="en" to indicate that language is in English

<head> and <body>

- Both are set inside the <html> element
- <head> should be first element inside <html>, contains metadata about the document
- <meta charset="utf-8" /> should be first element inside <head>, informs browser to use the standard UTF-8 character set
- <title> element should be set in <head>
- <body> element contains the webpage content that will actually be rendered by the browser
- DISCUSS: What are two reasons to always set a title?

Block-level vs Inline Elements

- Block-level elements will always begin on a new line.
- Inline elements continue on the same line.

Division

- Non-semantic element – has no inherent meaning
- Block-level
- Container that can hold almost anything, even other s and other block-level elements
- Ideally, use as last resort when a more semantic element doesn't make sense to use

<section>

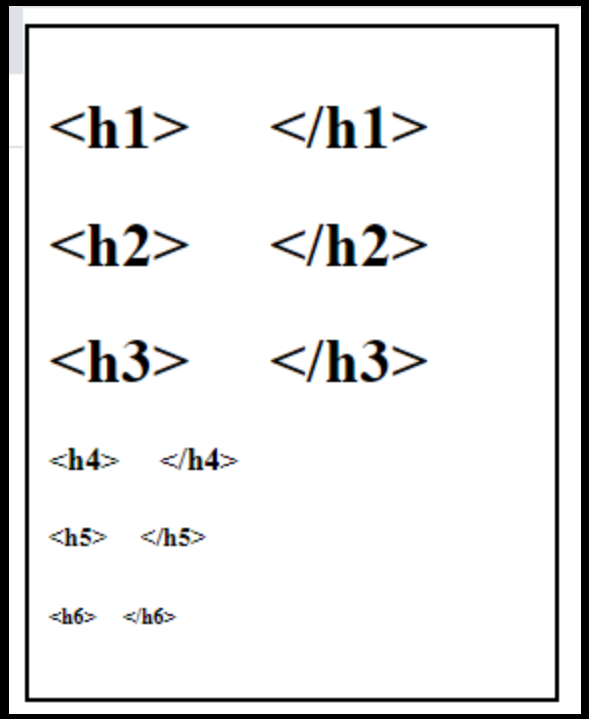
- Semantic element used to identify a thematic group of content
- Generally but not always includes a heading
- Default styles and behavior are same as : block-level, can hold almost anything including block-level elements and other <section>s.

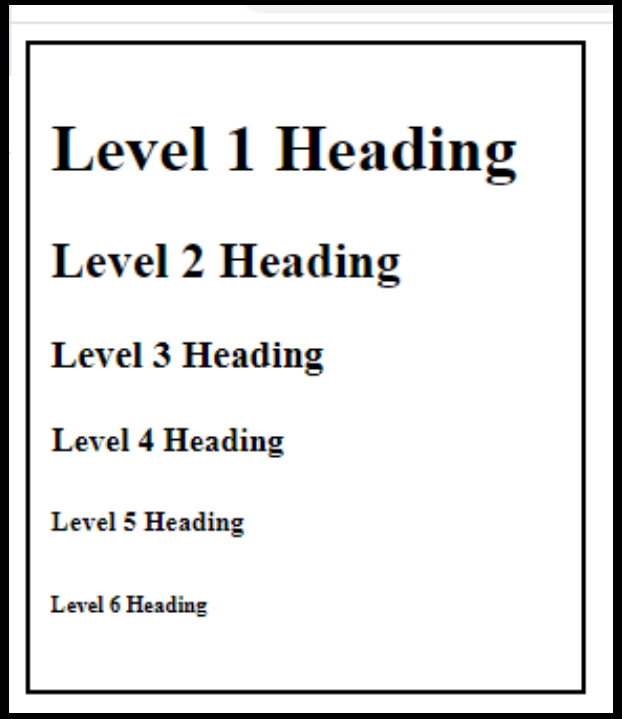
Source: https://learn.shayhowe.com/html-css/getting-to-know-html/

<p> Paragraph

- Use to contain text
- Block-level element
- Cannot hold other block-level elements, only inline elements
- Default style includes extra top and bottom margins

<h1>-<h6> Headings





- Use to indicate the document structure – ask yourself when you use one, would this text fit into an outline/table of contents for my document?
- *Do not use* just for styling the size.
- Block-level

   Ordered List, Unordered List, Listitem

 and <ul> elements both use the <li> element:

```
<ol>
    <li>Coconuts</li>
    <li>Mangos</li>
    <li>Bananas</li>
</ol>
```

1. Coconuts
2. Mangos
3. Bananas

```
<ul>
    <li>Coconuts</li>
    <li>Mangos</li>
    <li>Bananas</li>
</ul>
```

- Coconuts
- Mangos
- Bananas

- can also use reversed attribute to reverse order & start attribute to set start value

<!-- Comments in HTML -->

- Comments are not visible in the browser, ignored when it renders the webpage
- Three reasons to use comments:
  1. Leave a helpful note for yourself
  2. Leave notes for other developers
  3. "Comment out" code that you don't want to be rendered, temporarily

<a> Anchor (1 of 2)

- Use to create hyperlinks
- Required **href** attribute:
  - Link via absolute path:
    `<a href="https://www.nucamp.co">Nucamp</a>`
  - Link via relative path:
    `<a href="../otherPage.html">Some Other Page</a>`
- **Discuss:** Any questions about absolute vs relative path? Does everyone understand what **../** means and how it is used?

<a> (2 of 2)

- Use # and the **id** attribute to link to another element in the same page. Example:

```
<a href="#someId">Link to another element</a>
<p id="someId">  (notice no # in the id)
```

- Use tel: and mailto: to call a number or write to an email using default phone and email apps:

```
<a href="tel:+19995551212">Call Me</a>
<a href="mailto:hello@somewhere.com">Email Me</a>
```

Images

```
<img src="images/someImg.png" alt="Descriptive text" />
```

- Void element
- Required src attribute, give absolute or relative path to an image
- GIF, PNG, JPG, SVG are some of the most common image filetypes that browsers recognize, there are others – typically use PNG for digitally created images and JPG for photographs
- Always provide a descriptive alt attribute text, useful for screenreaders and search engines, also as backup for if the image fails to load

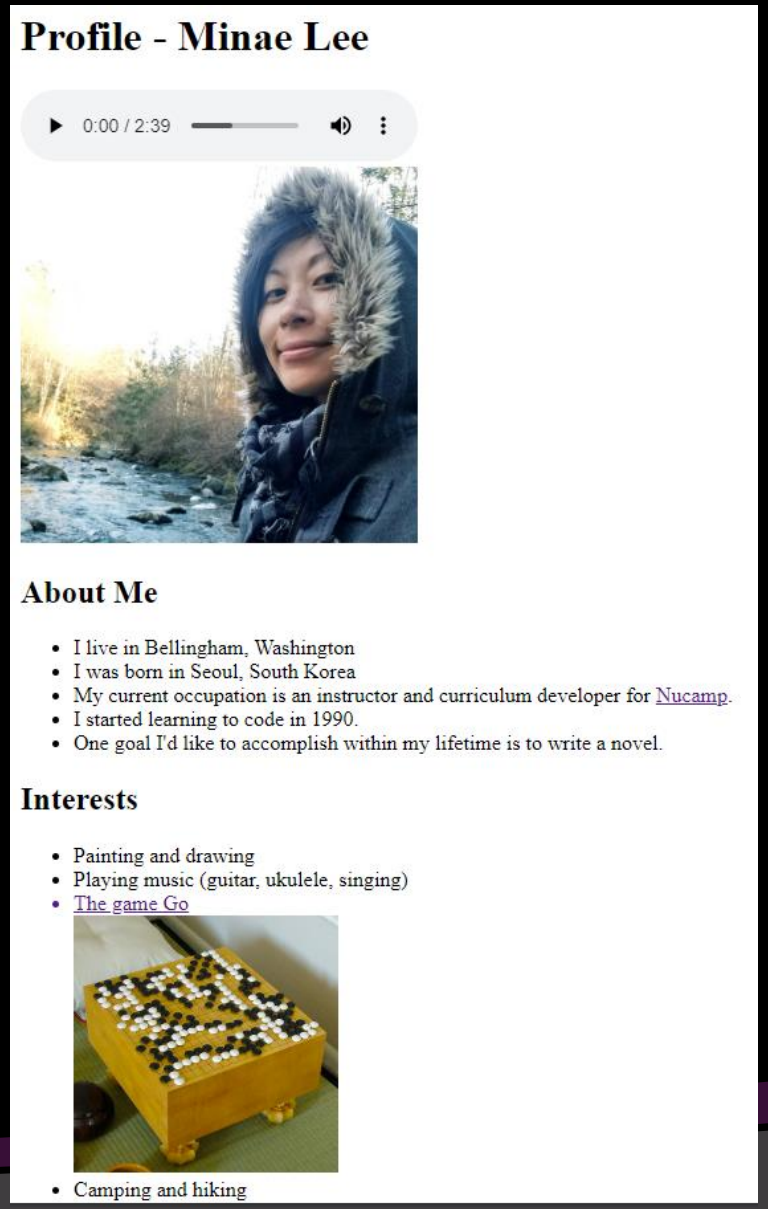
<video>

```
<video src="videos/someVideo.mp4" controls />
```

- *Not* a void element
- New in HTML5, supports MP4/OGG/WEBM video formats
- Use fallback content between start and end tags, e.g.: Your browser does not support HTML5 video. Then provide a link to the video.
- Can use **src** attribute to provide absolute or relative path to a video
- Boolean **controls** attribute will show video player controls, **loop** will automatically loop the video

Workshop Assignment

- Create a profile.html page with information about yourself
- Include at least two photos, two anchor elements, an audio element, two unordered lists, and the other elements described in the instructions



Workshop Assignment

- It's time to start the workshop assignment!
- The final 30 minutes of your workshop will be used to share your profile pages with each other.
- Break out into groups of 2-3. Sit near your workshop partner(s).
  - Your instructor may assign partners, or have you choose.
- Work closely with each other.
  - 10-minute rule does *not* apply to talking to your partner(s). Work together throughout. This will be useful practice for working with teams in real life.
- Follow the workshop instructions very closely.
  - Talk to your instructor if any of the instructions are unclear to you.

Assignment Submission

- Create a screenshot of your profile.html page as shown in Chrome or Firefox. See the "Submission" section at the bottom of the written instructions to find out how to create a full page screenshot.
- Submit the screenshot and your profile.html page at the bottom of the assignment page in the learning portal.