Week 1 Workshop

Web Development Fundamentals
HTML, CSS, and JavaScript

Agenda

Activity	Time	~Start
Set-Up / Prep: Log into Nucamp Learning Portal • Slack • Screenshare	10 mins	9:00am
Instructor & Student Introductions / Ice Breakers	20 mins	9:10am
Your Bootcamp Checklist	30 mins	9:30am
Review	40 mins	10:00am
Workshop Assignment	20 mins	10:40am
Break	15 mins	11:00am
Workshop Assignment	75 mins	11:15am
Show and Tell + Checkout (Feedback & Wrap-Up)	30 mins	12:30pm

Introductions

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
  - Any Code Challenges and Quiz toward end of week
  - Additional exploration
  - Prepare for workshop assignment
  - Help other classmates via class forums or Slack

- **4-hour workshop every weekend**
  - Read the assignment instructions prior to Saturday

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.
- Remember the process or path that resolved the issue 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!

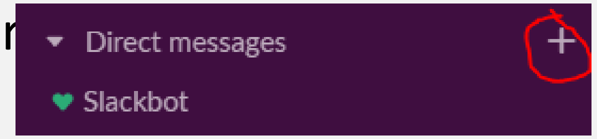
Bootcamp Checklist

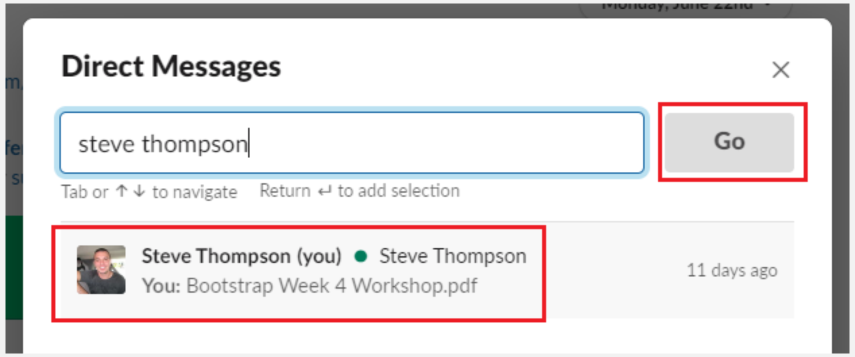
- Receiving Daily Emails
- VS Code installed
- Chrome / Firefox (updated)
- Installed Moodle App on phone
- NucampFolder
- Slack (introduced yourself)
- Time Commitment Pledge
- Read and agree to the Code of Conduct

Using Slack

- Join the following Slack channels
  - #1-htmlcssjavascript
  - Your cohort's course channel, e.g. #online-central-htmlcssjavascript-03

- Does everyone know how to send a Direct Message to the instructor?
  - Click the + to the right of the "Direct messages"
  - Search for your instructor
  - Click their image
  - Select "Go"




Direct Messages

steve thompson

Go

Tab or ↑ ↓ to navigate   Return ↵ to add selection

Steve Thompson (you) ● Steve Thompson
You: Bootstrap Week 4 Workshop.pdf
11 days ago

- Does everyone know how to use message threads in channels? Demonstrate

- Does everyone know how to attach files? Demonstrate

Asking For Help in Slack Channels

- Remember these key points when asking for help in a Slack channel
  - Instead of saying "Can anyone help me with a problem?", ask a direct question about your problem/issue,
    - Include what you have tried, and which 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!

Next 4 Weeks

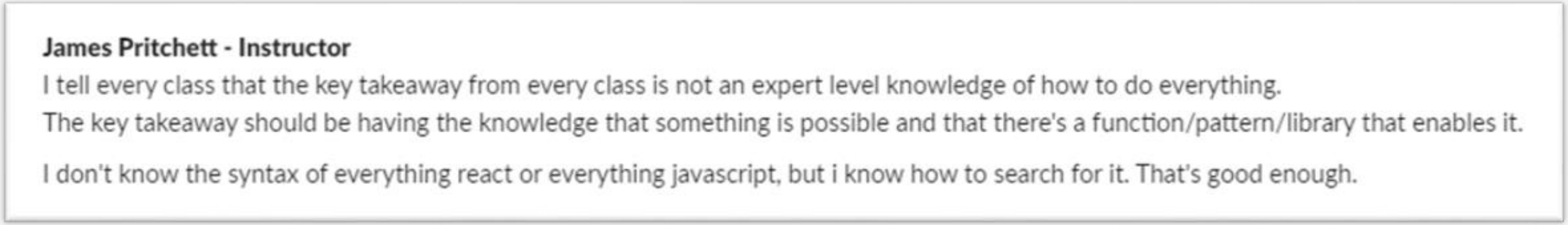
Week 1: This week! Course Setup & Introduction to HTML

Week 2: More HTML and Introduction to CSS

Week 3: Introduction to JavaScript

Week 4: JavaScript and the DOM (Document Object Model)

Words of Wisdom



James Pritchett - Instructor
I tell every class that the key takeaway from every class is not an expert level knowledge of how to do everything.
The key takeaway should be having the knowledge that something is possible and that there's a function/pattern/library that enables it.

I don't know the syntax of everything react or everything javascript, but i know how to search for it. That's good enough.

Week 1 - Review

Introduction to HTML

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
  - Structure
    - Think of framing and layout of a house

- CSS (Cascading Style Sheets)
  - Used for style and positioning – colors, fonts, etc.
  - Stylesheet language
  - Style & Appearance
    - Think of painted walls, veneered fireplace, texture, and position of decor

- JavaScript
  - Used to create dynamic, interactive websites
  - The only one of the three that is a **programming** language
  - Interactivity
    - Think of light switches; push a switch (button) and turns on a light

HTML – Created the button/switch components

CSS - Change the appearance of a button

JS would make the lights turn on when triggered

Web Designers vs Web Developers

Discuss:

- What do you understand to be the difference between web design and web development?

Web Design – Concerned more about the look and feel of the site (fonts, colors, layout)

Web Development – Turns the design into code to create a functional site

- Which you will be learning with us at Nucamp?

Web Development

Front End vs Back End

Discuss:

- What do you understand to be the difference between front end and back end web development? What does full stack mean?

Front End – Concerned with client side code and functionality (HTML, CSS, JavaScript)

Back End - Concerned more with the server-side code that responds to requests from the client (JavaScript, PHP, Ruby, Python, C#, etc)

- Which you will be learning with us in *this* course?

Front End

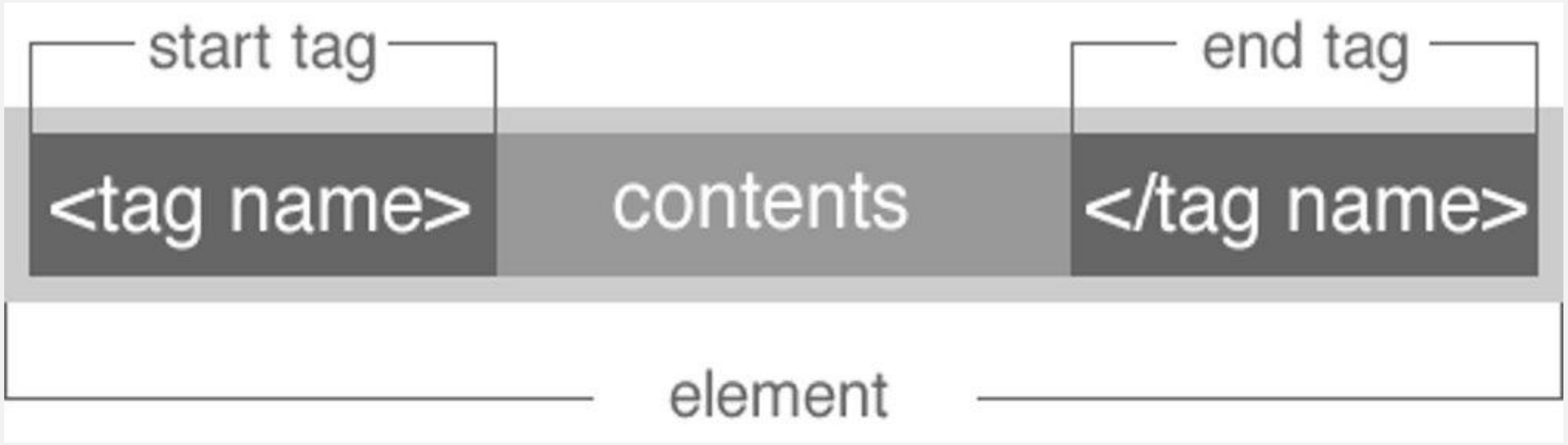
Using Visual Studio Code

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

- 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:

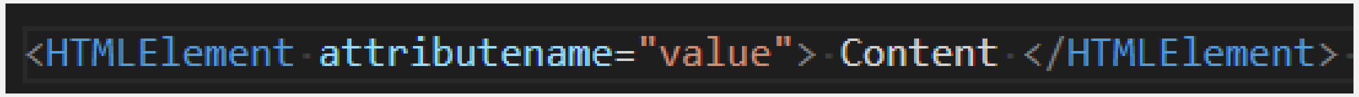
- Void elements can have attributes.

Examples: **<img src=**"myImg.jpg" **/>**

<img src="myImg.jpg"**>**

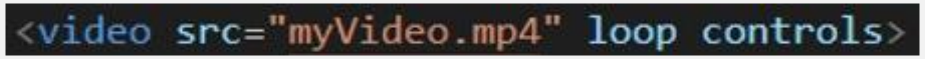
Attributes

- All elements have a set of applicable attributes.
- Attributes are written in the start tag.
- They can be written as a key-value pair (**key**="value")
  - The value can be in single or double quotes, be consistent



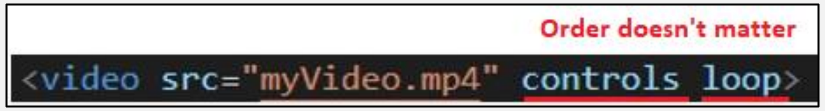
```
<HTMLElement attributename="value"> Content </HTMLElement>
```

- Some attributes are **Boolean** - only the attribute name, no value, such as **controls**
- If an element has multiple attributes, use spaces between each; the order does not technically matter



```
<video src="myVideo.mp4" loop controls>
```

is the same as



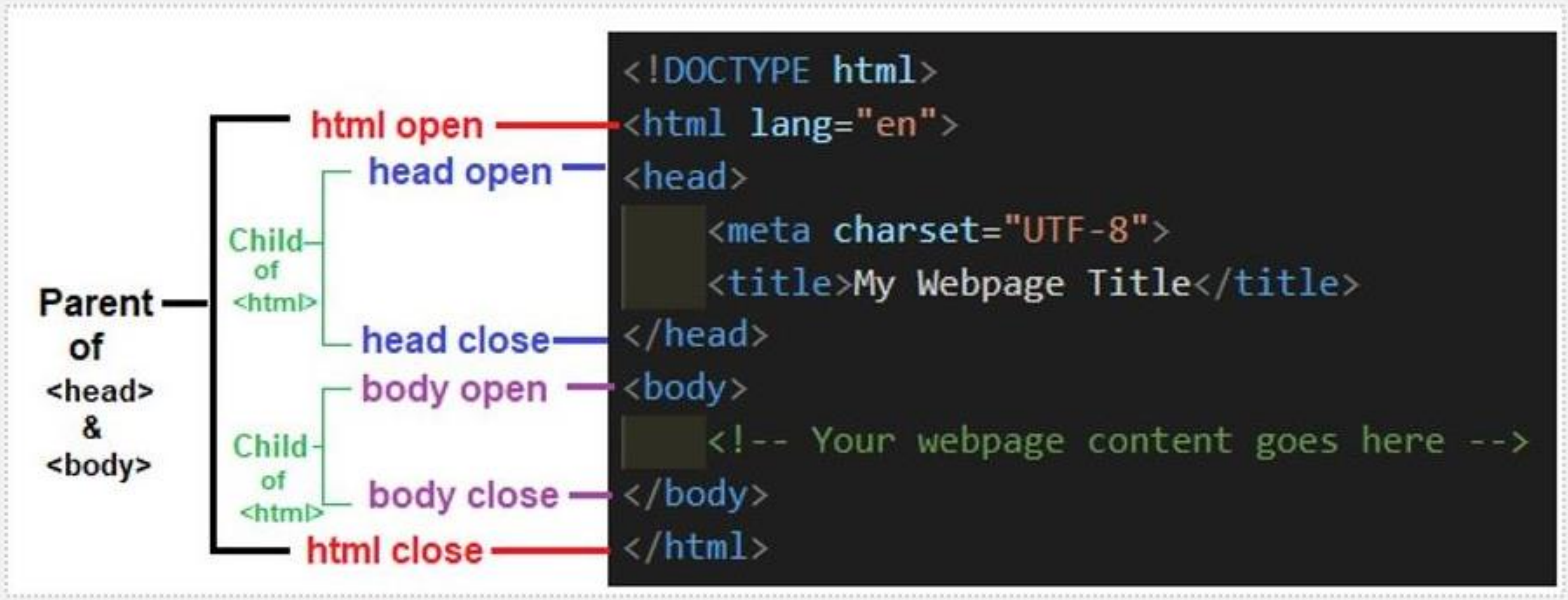
Order doesn't matter

```
<video src="myVideo.mp4" controls loop>
```

History of HTML and World Wide Web

- HTTP (HyperText Transfer 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

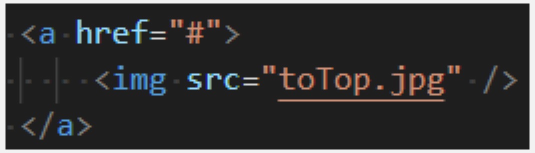
Basic HTML5 Document Structure



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

- When an HTML element "opens" it is considered the **parent** of the elements directly inside of it prior to its "closing"
  - e.g. In the above code, the **head** element is the parent of the **meta** and **title** elements

- You'll sometimes hear the term "container" or "wrapper", this just means the parent element
  - e.g. "*wrap an **<a>** element around an **<img>** element*"



```html
<a href="#">
    <img src="toTop.jpg" />
</a>
```

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
  - This is ***critical*** for accessibility compliance. Every single page should have a "lang" attribute included with the html start tag, as in:
    - <html lang="en">
  - Feel free to change the language as needed throughout the document. For example, a particular paragraph element might contain text in French, so for that paragraph element's start tag, you could specify:
    - <p lang="fr">Parlez-vous francais?</p>

<head> and <body>

- Both are set inside the <html> element
- <head> should be first element inside <html> and contains metadata (sets of data that describes other data) about the document
- <meta charset="utf-8" /> should be first element inside <head>, informs browser to use the standard UTF-8 character set

Unicode Transformation Format (8-bit) - Character encoding (converting characters to binary code)

- <title> element should be set in <head>

Text displayed in your brower tab

- <body> element contains the webpage content that will actually be rendered by the browser

First impression people have of your page

<u>DISCUSS</u>: What are two reasons to always set a title?

Help Search Engine understand what your page is about

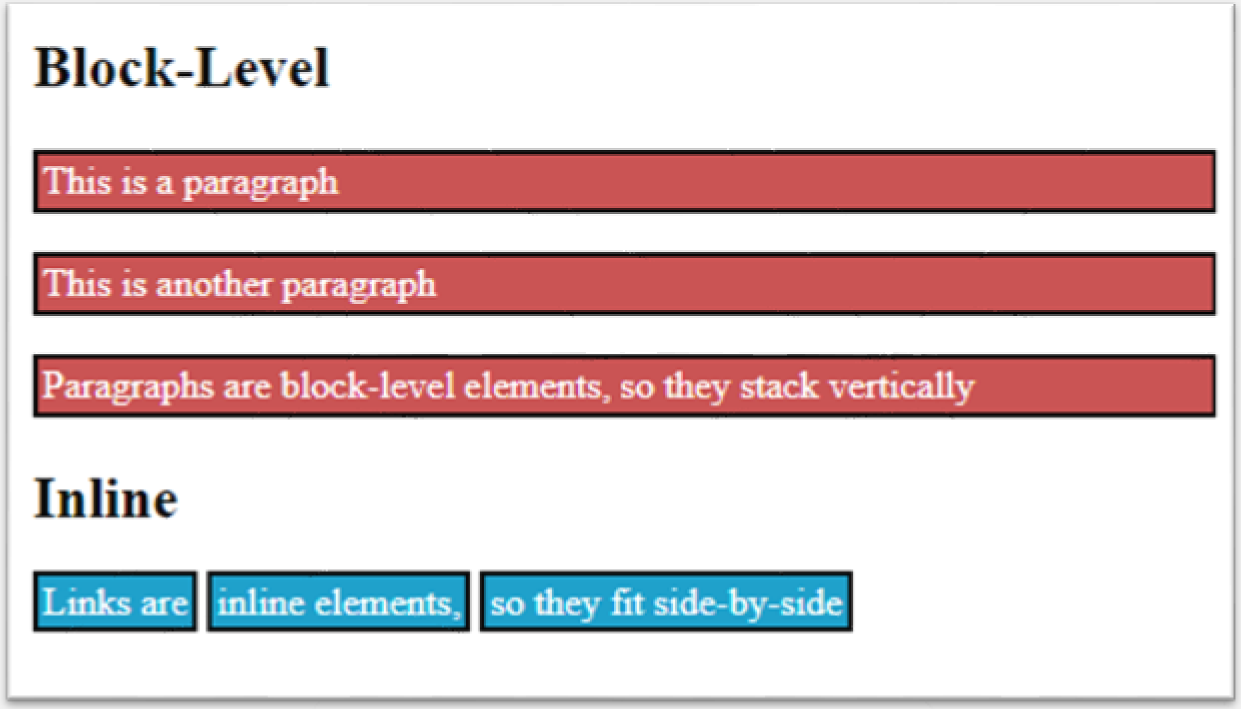
Block-level vs Inline Elements

- **Block-level** elements will always begin on a new line.

  Common Examples: <div>, <p>, <ul>, <li>, <h1>

- **Inline** elements can start anywhere and continue on the same line.

  Common Examples: <img>,<a>, <strong>, <input>, <label>, <select>, <video>



Block-Level

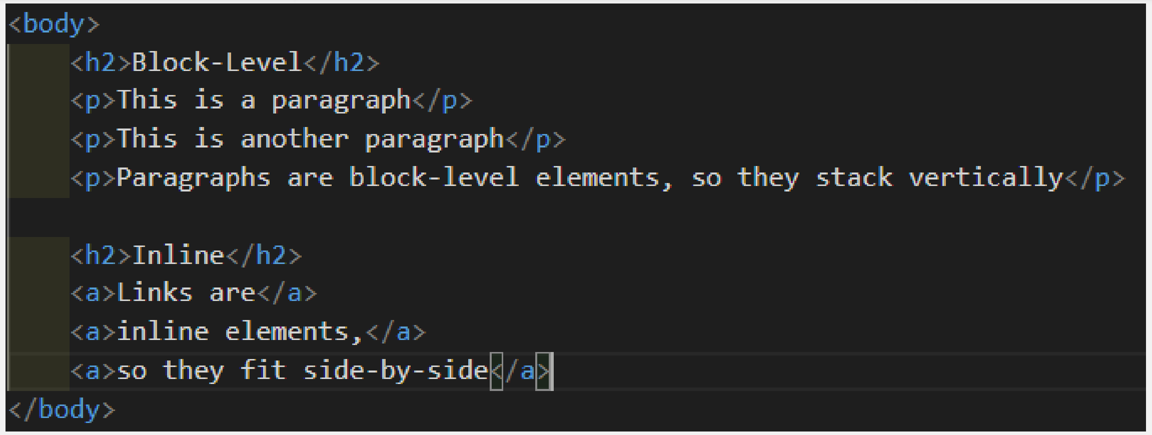
This is a paragraph

This is another paragraph

Paragraphs are block-level elements, so they stack vertically

Inline

Links are inline elements, so they fit side-by-side



```
<body>
    <h2>Block-Level</h2>
    <p>This is a paragraph</p>
    <p>This is another paragraph</p>
    <p>Paragraphs are block-level elements, so they stack vertically</p>

    <h2>Inline</h2>
    <a>Links are</a>
    <a>inline elements,</a>
    <a>so they fit side-by-side</a>
</body>
```

<div> Division

- Non-semantic element – has no inherent meaning
    - For this reason, it should never hold content, in order to be WCAG accessibility compliant (Web Content Accessibility Guidelines)

- Block-level

- Container that can hold almost anything, even other **<div>**s and other block-level elements
    - Ideally, use this only when a more semantic element doesn't make sense to use. If you have something that already works, use it. If you need a button use a <button> don't recreate it with <span>. Use CSS to style. HTML is not for styling.

<section>

- Semantic element used to identify a related group of content

- Generally, but not always, includes a heading

- Default styles and behavior are same as **<div>**: block-level, can hold almost anything including other block-level elements and other **<section>**s.

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

<p> Paragraph

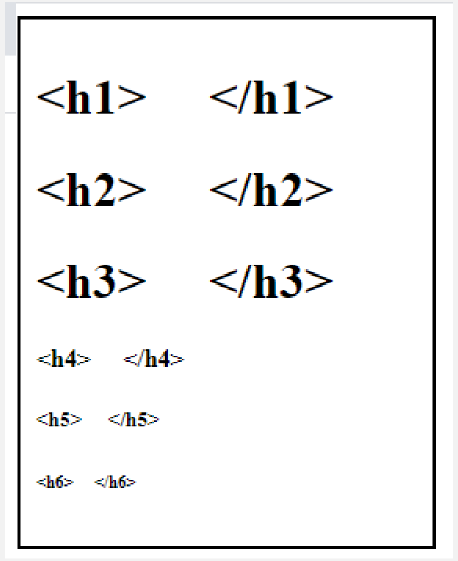
- Use to contain text
- Block-level element
- Cannot hold other block-level elements, only inline elements
- Default style includes extra top and bottom margins

<h1>-<h6> Headings

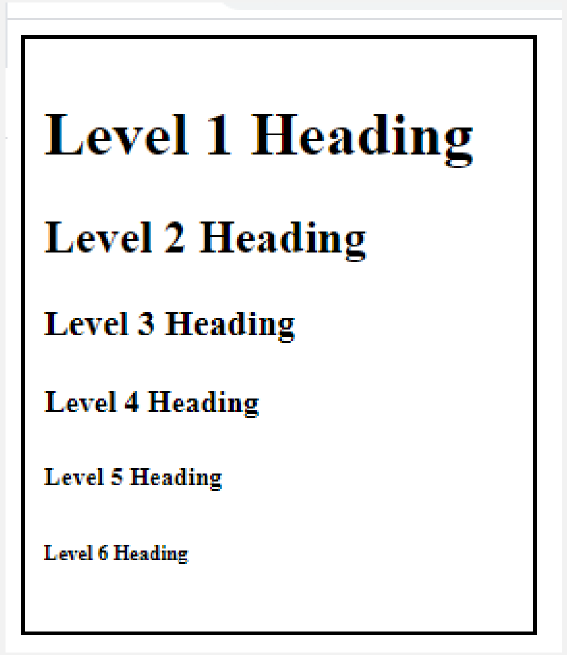
- Use to indicate the document structure – ask yourself when you use one, would this text fit into an outline/table of contents for my document?
  - Structure is critical for SEOs and Accessibility. Structure announces to Google and screen readers what order is most important.

Do not use just for styling the size.

- Block-level



```
<h1>    </h1>
<h2>    </h2>
<h3>    </h3>
<h4>    </h4>
<h5>    </h5>
<h6>    </h6>
```



Level 1 Heading

Level 2 Heading

Level 3 Heading

Level 4 Heading

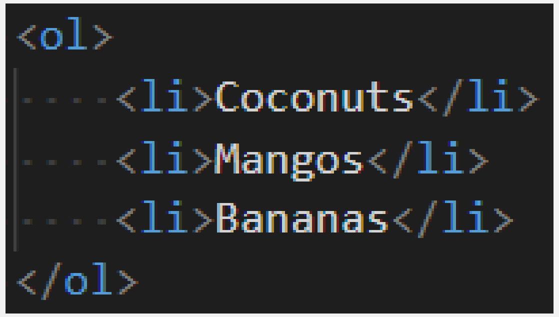
Level 5 Heading

Level 6 Heading

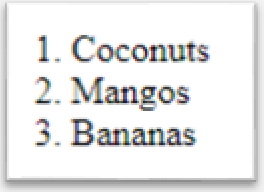
<ol> <ul> <li> Ordered List, Unordered List, List-item

- **<ol>** and **<ul>** elements both use the **<li>** element:
- **<ol>** can also use **reversed** attribute to reverse order & **start** attribute to set start value

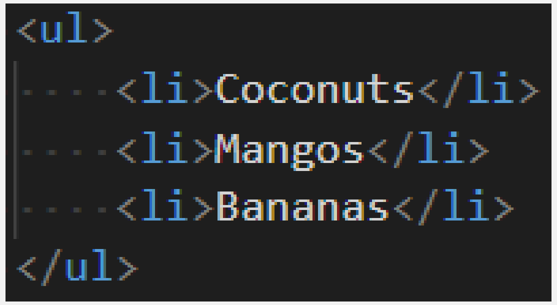
Accessibility note:
Be careful with the CSS property:value "list-style: none;"
This may remove key components the screen reader relies on to provide feedback to the user.



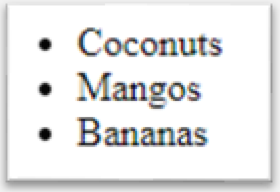
```
<ol>
    <li>Coconuts</li>
    <li>Mangos</li>
    <li>Bananas</li>
</ol>
```



1. Coconuts
2. Mangos
3. Bananas

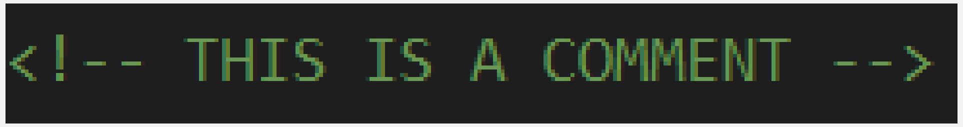


```
<ul>
    <li>Coconuts</li>
    <li>Mangos</li>
    <li>Bananas</li>
</ul>
```



- Coconuts
- Mangos
- Bananas

<!-- Comments in HTML -->



```
<!-- THIS IS A COMMENT -->
```

- 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>
```

Think of your home address
(123 Mockingbird Lane, New York, NY)

  - Link via relative path:

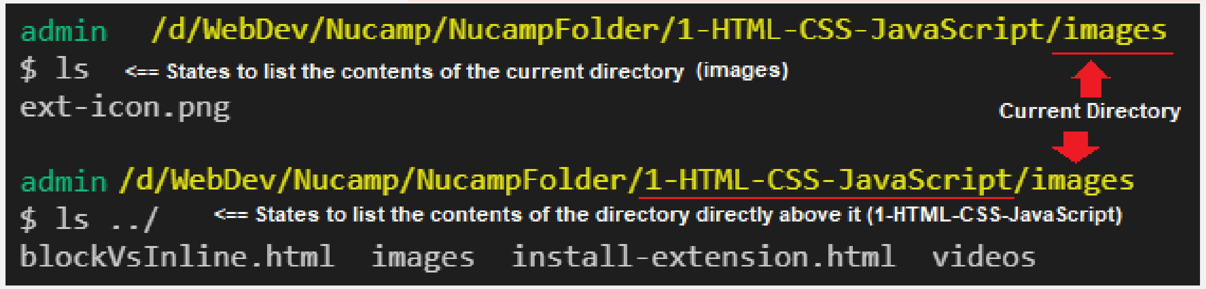
```
<a href="../otherPage.html">Some Other Page</a>
```

Think of direction to your house based on your current position
(make a left on Cherry St. and it's the 3rd house on the left)

- **Discuss:** Any questions about absolute vs relative path? Does everyone understand what **../** means and how it is used?

../ states to go UP one directory

Example below from a Bash terminal

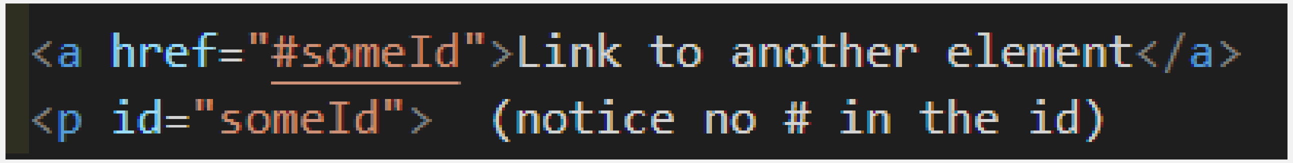


```
admin  /d/WebDev/Nucamp/NucampFolder/1-HTML-CSS-JavaScript/images
$ ls   <== States to list the contents of the current directory (images)
ext-icon.png

admin /d/WebDev/Nucamp/NucampFolder/1-HTML-CSS-JavaScript/images
$ ls ../     <== States to list the contents of the directory directly above it (1-HTML-CSS-JavaScript)
blockVsInline.html  images  install-extension.html  videos
```

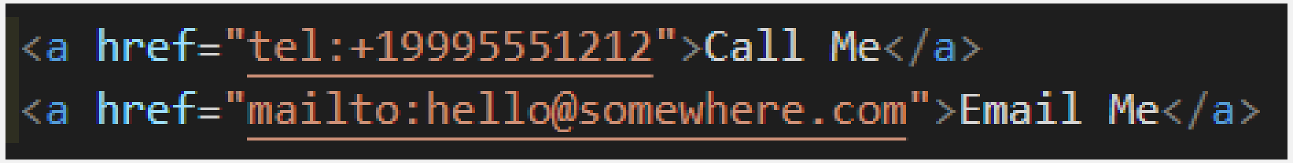
<a> Anchor (2 of 2)

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



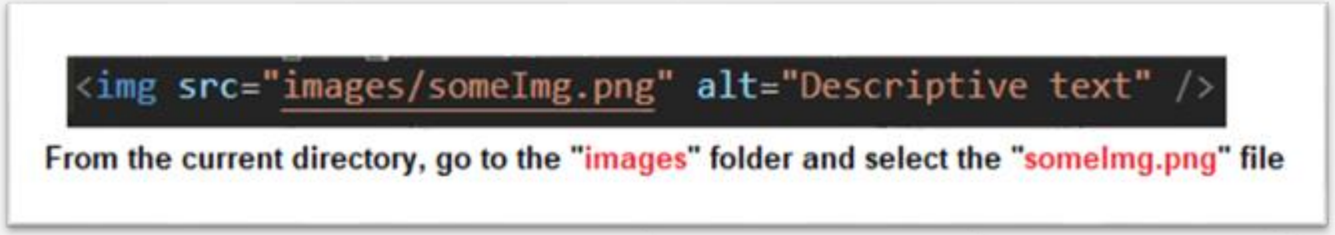
```
<a href="#someId">Link to another element</a>
<p id="someId">  (notice no # in the id)
```

- 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

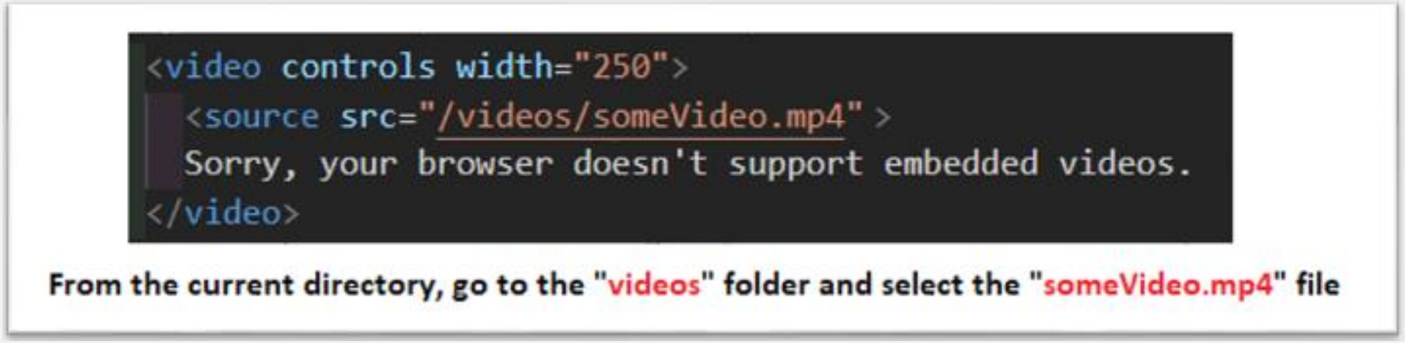


```
<img src="images/someImg.png" alt="Descriptive text" />
```

From the current directory, go to the "images" folder and select the "someImg.png" file

- 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 PNG is used for digitally created images and JPG for photographs
- Always provide a descriptive alt attribute text, useful for screen readers and search engines, also as backup for if the image fails to load

<video>



```
<video controls width="250">
  <source src="/videos/someVideo.mp4" >
  Sorry, your browser doesn't support embedded videos.
</video>
```

From the current directory, go to the "videos" folder and select the "someVideo.mp4" file

- *Not* a void element
- New in HTML5, supports MP4/OGG/WEBM video formats
- Use fallback text 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 if present, Boolean loop attribute will automatically loop the video if present

Week 1 - Workshop

Introduction to HTML

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.

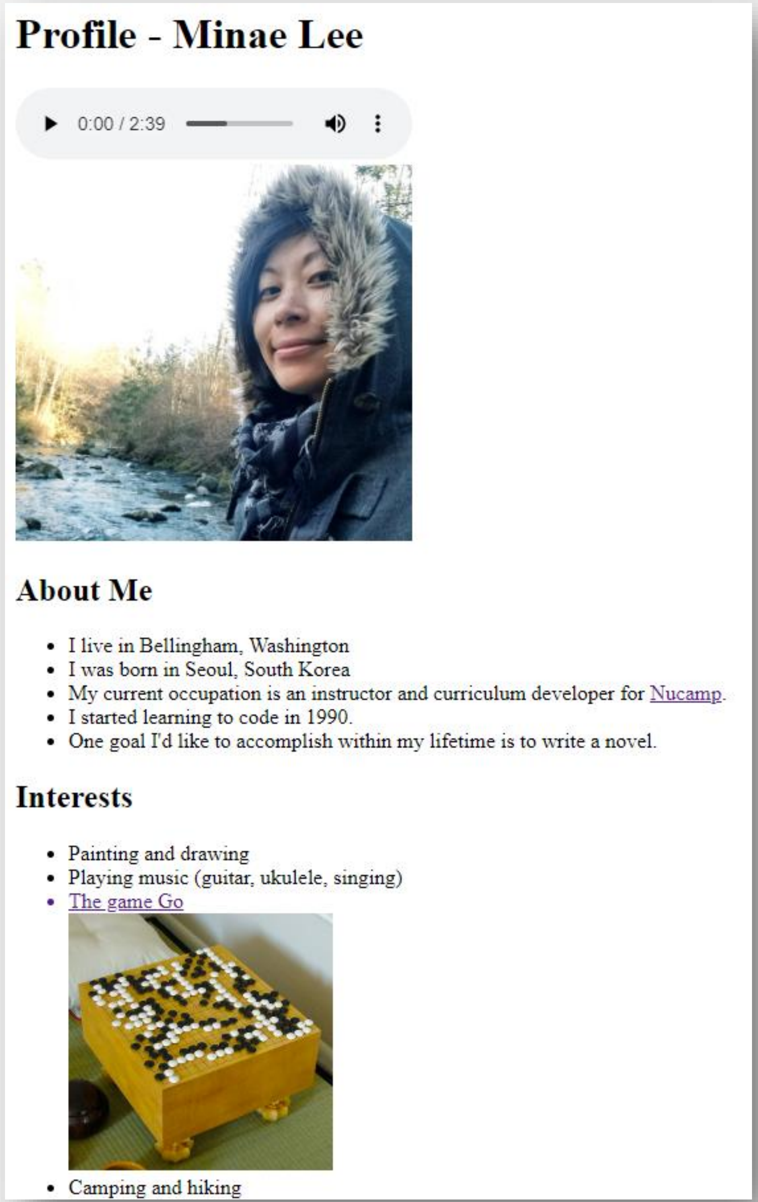
- Break out into groups of 2-3
  - Sit near your workshop partner(s) if your workshop is in person
  - For online workshops, your instructor may break you out into different virtual rooms

- 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.

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
  - Other elements described in the instructions



Profile - Minae Lee

About Me

- I live in Bellingham, Washington
- I was born in Seoul, South Korea
- My current occupation is an instructor and curriculum developer for Nucamp.
- I started learning to code in 1990.
- One goal I'd like to accomplish within my lifetime is to write a novel.

Interests

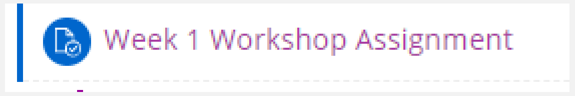
- Painting and drawing
- Playing music (guitar, ukulele, singing)
- The game Go
- Camping and hiking

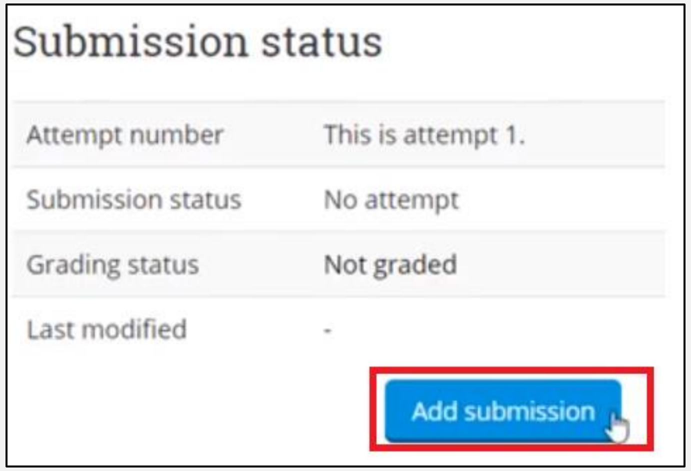
Assignment Submission

- 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.
- Example instruction on the next slide

Submitting Your Assignment

- Go to https://learn.nucamp.co
  - Click "**Workshop Assignment: Students' Work**"
  - Upload your work by clicking "**Add Submission**", select the file, and then click "save"

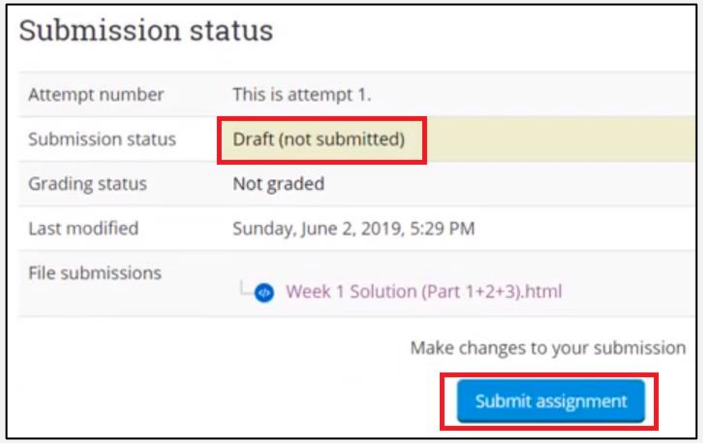




Submission status

Attempt number	This is attempt 1.
Submission status	No attempt
Grading status	Not graded
Last modified	-

- Note that your work is in Draft status
  - Click "**Submit assignment**" to submit it

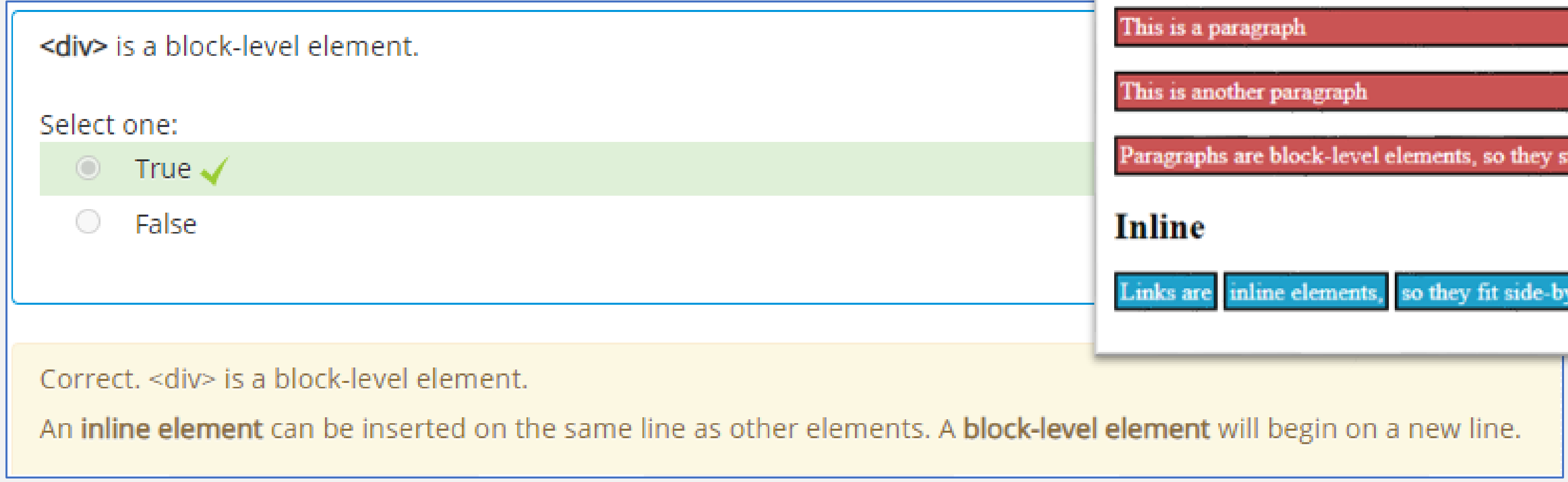


Submission status

Attempt number	This is attempt 1.
Submission status	Draft (not submitted)
Grading status	Not graded
Last modified	Sunday, June 2, 2019, 5:29 PM
File submissions	Week 1 Solution (Part 1+2+3).html

Make changes to your submission

Review: Week 1 Quiz



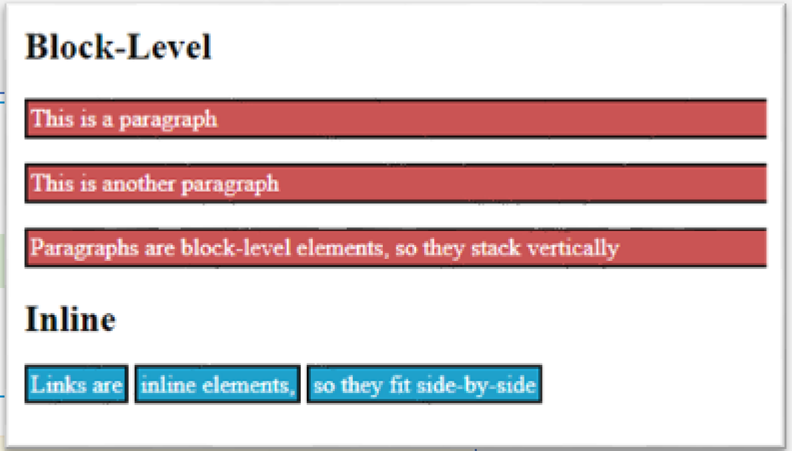
<div> is a block-level element.

Select one:

- [✓ True] ✓
- False

Correct. <div> is a block-level element.

An **inline element** can be inserted on the same line as other elements. A **block-level element** will begin on a new line.



Block-Level

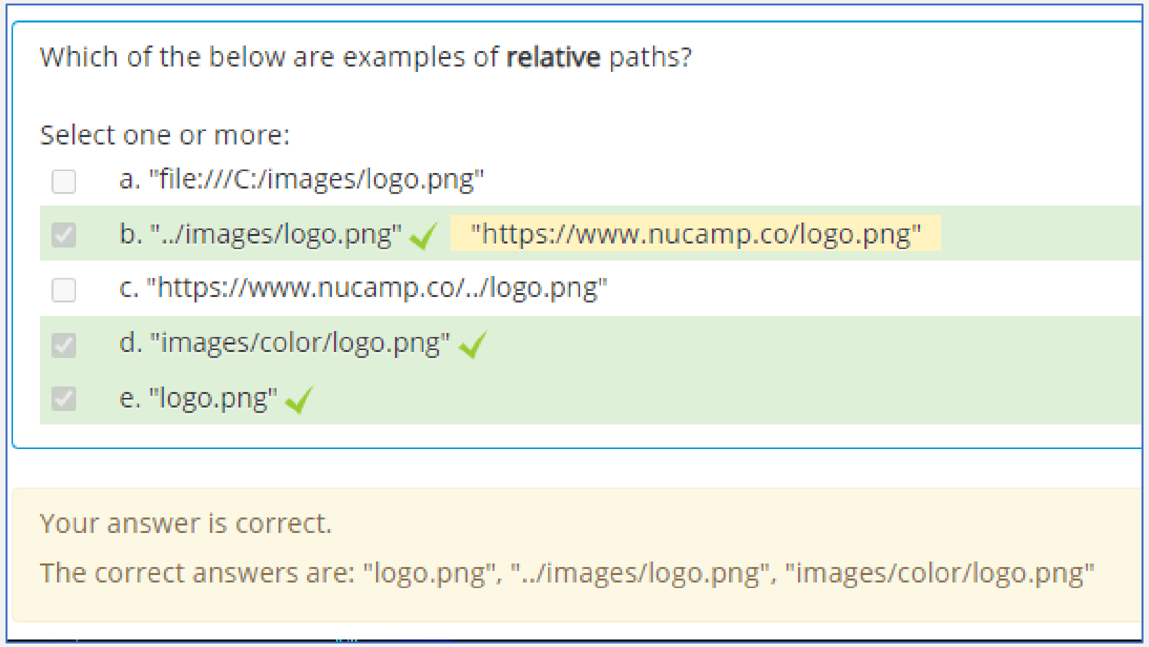
This is a paragraph

This is another paragraph

Paragraphs are block-level elements, so they stack vertically

Inline

Links are inline elements, so they fit side-by-side



Which of the below are examples of **relative** paths?

Select one or more:

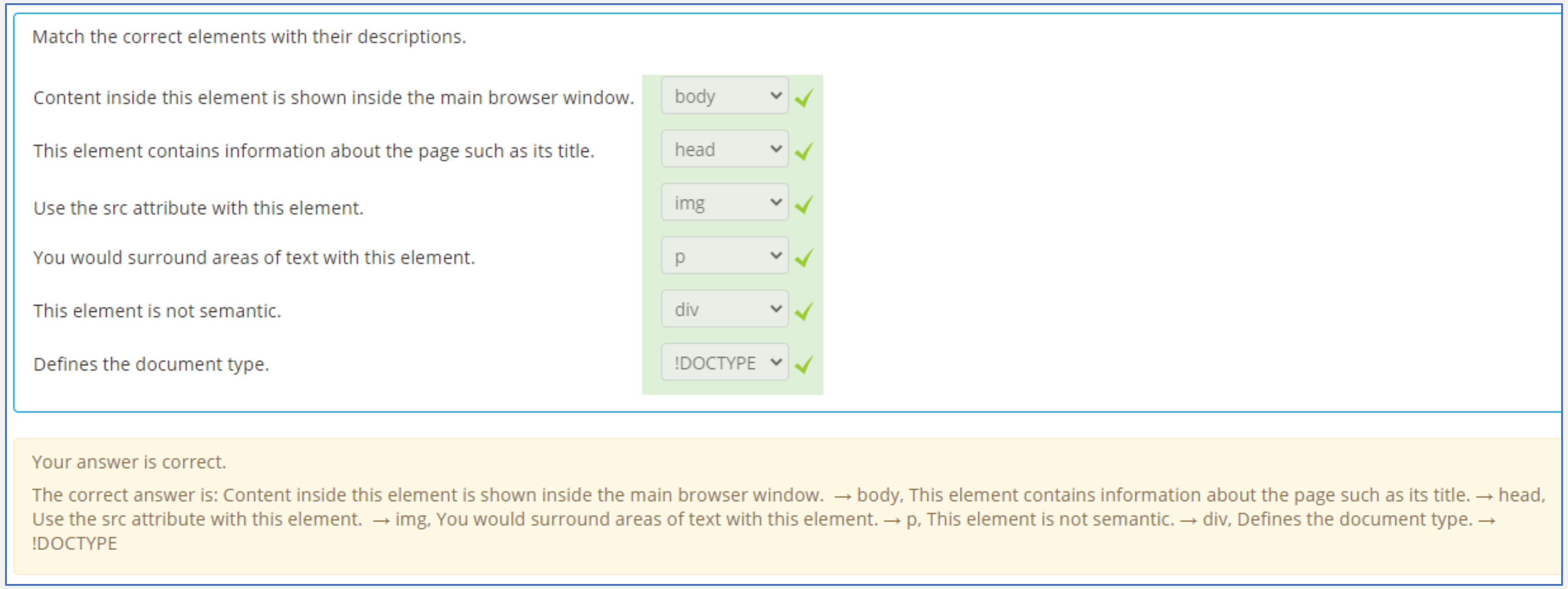
- a. "file:///C:/images/logo.png"
- [✓ b. "../images/logo.png"] ✓ "https://www.nucamp.co/logo.png"
- c. "https://www.nucamp.co/../logo.png"
- [✓ d. "images/color/logo.png"] ✓
- [✓ e. "logo.png"] ✓

Your answer is correct.

The correct answers are: "logo.png", "../images/logo.png", "images/color/logo.png"

Relative path – based on the HTML document's current location in your project to the desired file

Review: Week 1 Quiz

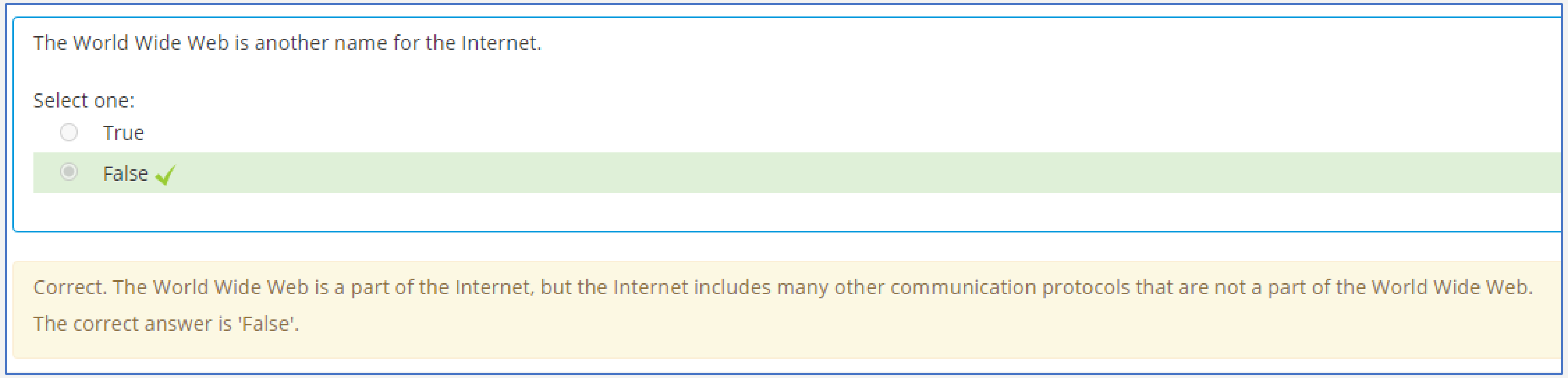


Match the correct elements with their descriptions.

Content inside this element is shown inside the main browser window.	body ✓
This element contains information about the page such as its title.	head ✓
Use the src attribute with this element.	img ✓
You would surround areas of text with this element.	p ✓
This element is not semantic.	div ✓
Defines the document type.	!DOCTYPE ✓

Your answer is correct.

The correct answer is: Content inside this element is shown inside the main browser window. → body, This element contains information about the page such as its title. → head, Use the src attribute with this element. → img, You would surround areas of text with this element. → p, This element is not semantic. → div, Defines the document type. → !DOCTYPE



The World Wide Web is another name for the Internet.

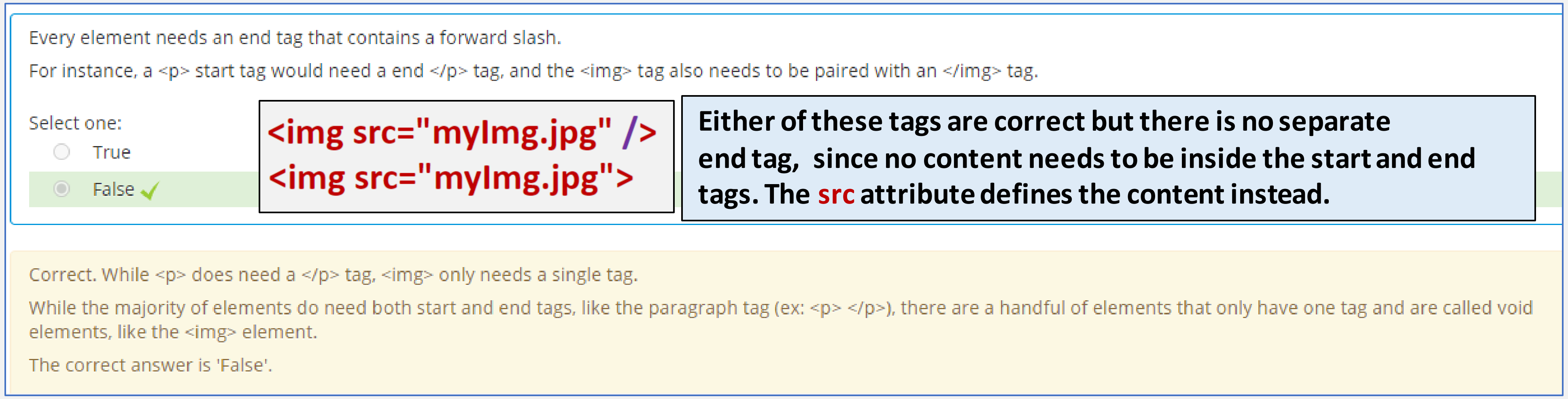
Select one:

True [✓ False]

Correct. The World Wide Web is a part of the Internet, but the Internet includes many other communication protocols that are not a part of the World Wide Web.

The correct answer is 'False'.

Review: Week 1 Quiz



Every element needs an end tag that contains a forward slash.

For instance, a <p> start tag would need a end </p> tag, and the <img> tag also needs to be paired with an </img> tag.

Select one:

True [✓ False ✓]

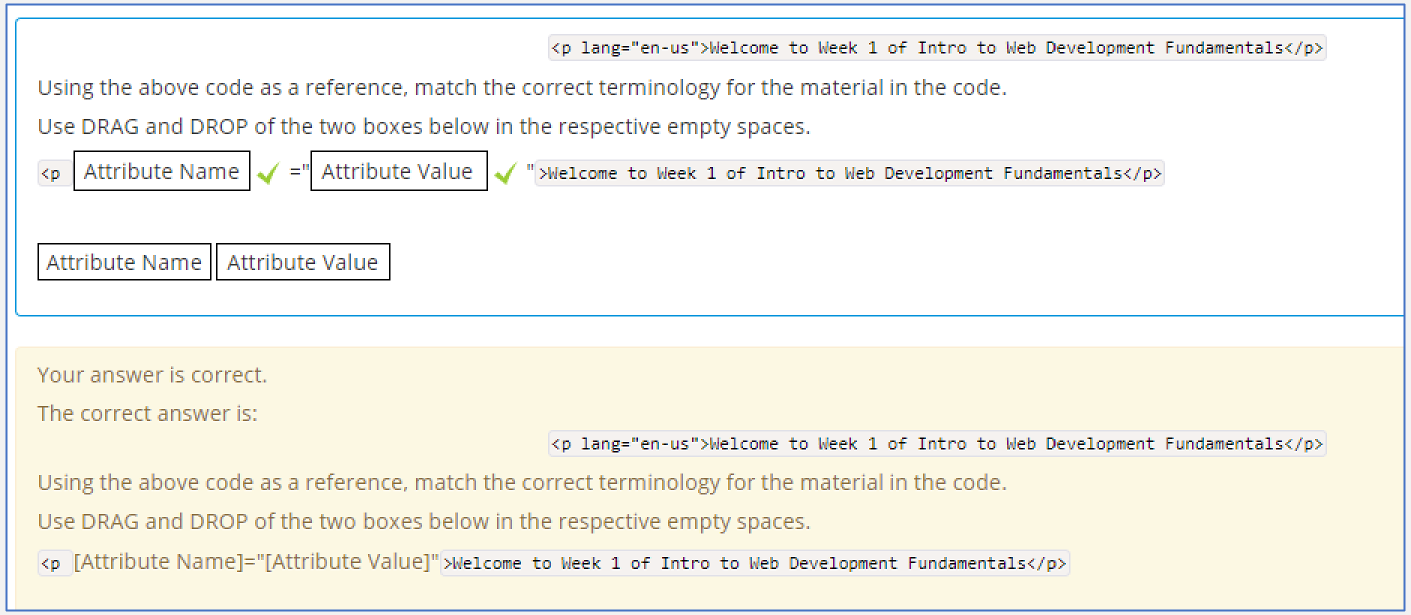
<img src="myImg.jpg" />
<img src="myImg.jpg">

Either of these tags are correct but there is no separate end tag, since no content needs to be inside the start and end tags. The src attribute defines the content instead.

Correct. While <p> does need a </p> tag, <img> only needs a single tag.

While the majority of elements do need both start and end tags, like the paragraph tag (ex: <p> </p>), there are a handful of elements that only have one tag and are called void elements, like the <img> element.

The correct answer is 'False'.



```
<p lang="en-us">Welcome to Week 1 of Intro to Web Development Fundamentals</p>
```

Using the above code as a reference, match the correct terminology for the material in the code.

Use DRAG and DROP of the two boxes below in the respective empty spaces.

<p [Attribute Name] ✓ =" [Attribute Value] ✓ ">Welcome to Week 1 of Intro to Web Development Fundamentals</p>

[Attribute Name] [Attribute Value]

Your answer is correct.

The correct answer is:

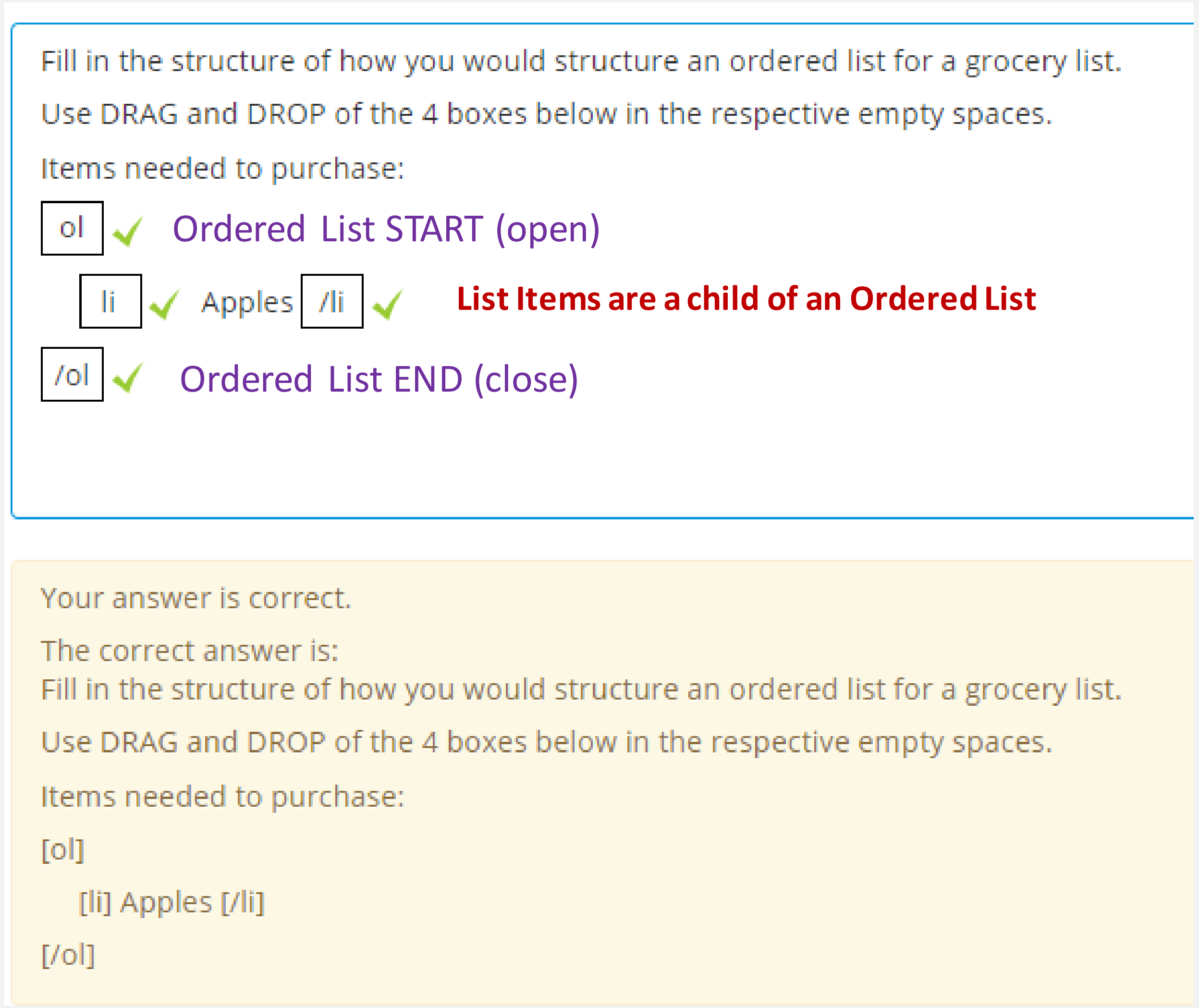
```
<p lang="en-us">Welcome to Week 1 of Intro to Web Development Fundamentals</p>
```

Using the above code as a reference, match the correct terminology for the material in the code.

Use DRAG and DROP of the two boxes below in the respective empty spaces.

<p [Attribute Name]="[Attribute Value]">Welcome to Week 1 of Intro to Web Development Fundamentals</p>

Review: Week 1 Quiz



Fill in the structure of how you would structure an ordered list for a grocery list.

Use DRAG and DROP of the 4 boxes below in the respective empty spaces.

Items needed to purchase:

ol ✓ Ordered List START (open)

li ✓ Apples /li ✓ **List Items are a child of an Ordered List**

/ol ✓ Ordered List END (close)

Your answer is correct.

The correct answer is:
Fill in the structure of how you would structure an ordered list for a grocery list.

Use DRAG and DROP of the 4 boxes below in the respective empty spaces.

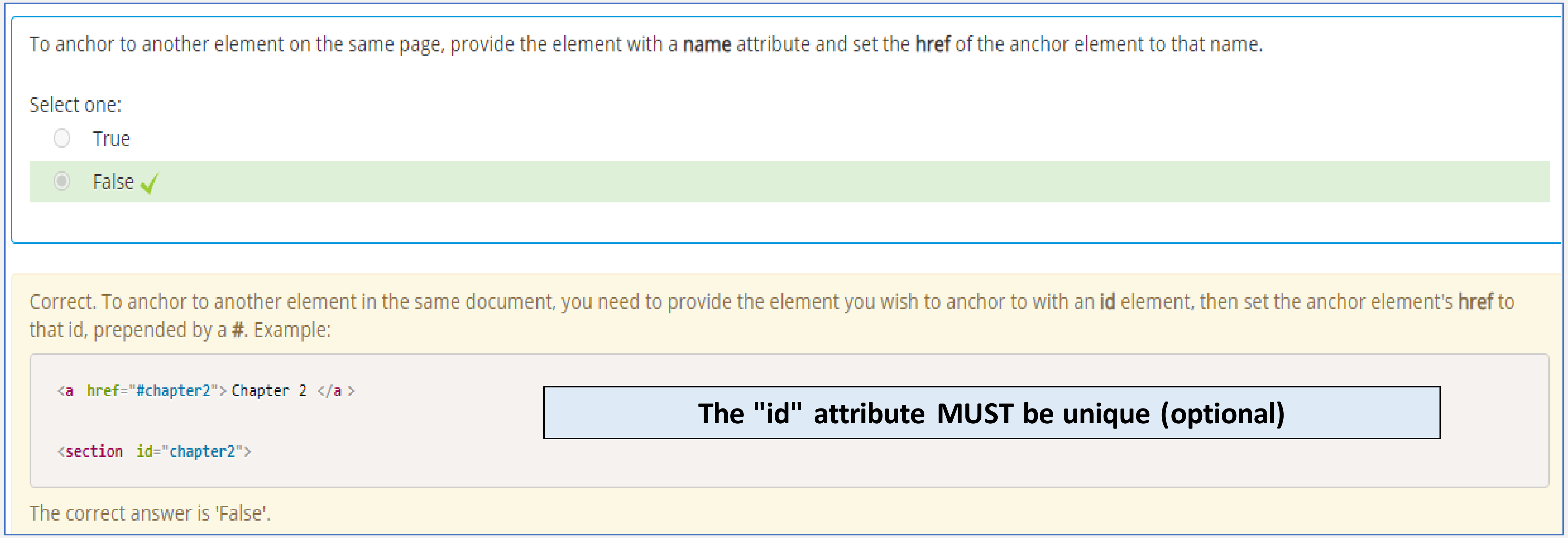
Items needed to purchase:

[ol]

    [li] Apples [/li]

[/ol]

Review: Week 1 Quiz



To anchor to another element on the same page, provide the element with a **name** attribute and set the **href** of the anchor element to that name.

Select one:

True

[✓ False] ✓

Correct. To anchor to another element in the same document, you need to provide the element you wish to anchor to with an **id** element, then set the anchor element's **href** to that id, prepended by a **#**. Example:

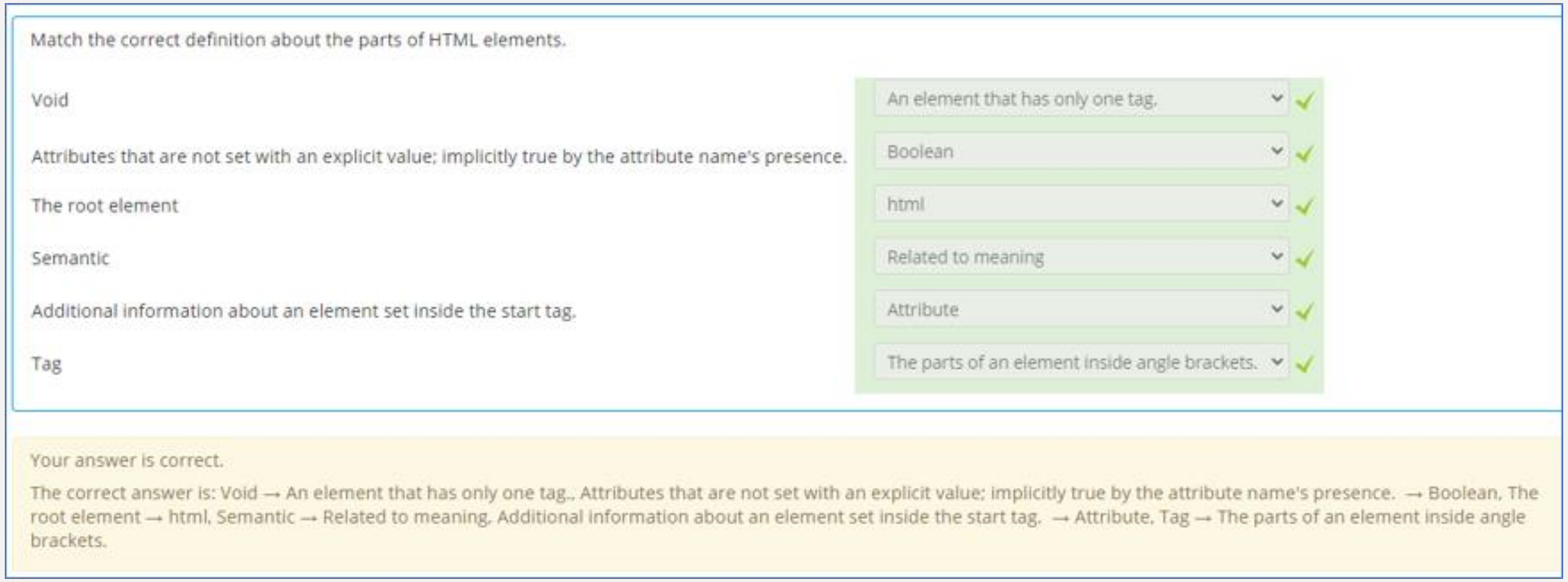
```
<a  href="#chapter2"> Chapter 2 </a >


<section  id="chapter2">
```

The "id" attribute MUST be unique (optional)

The correct answer is 'False'.

Review: Week 1 Quiz

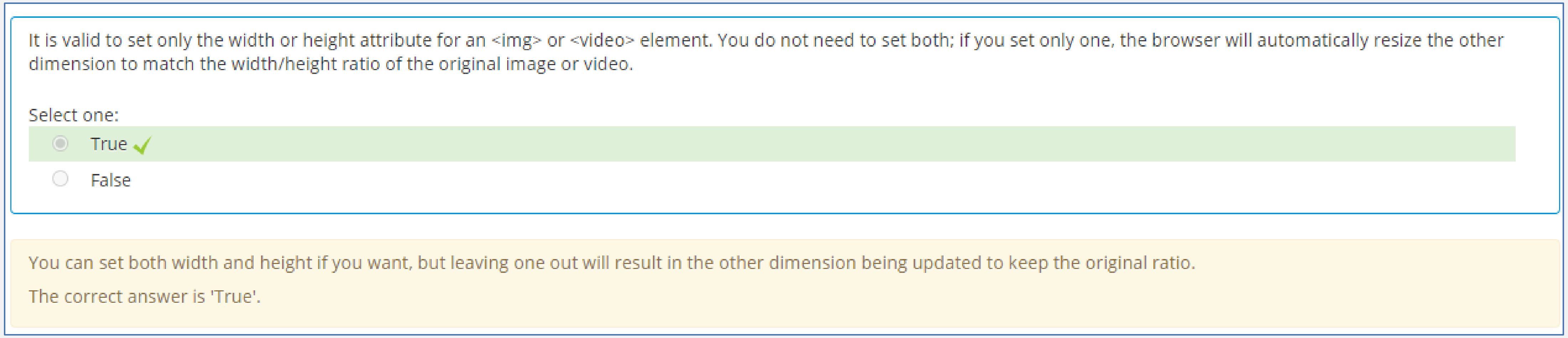


Match the correct definition about the parts of HTML elements.

Void	An element that has only one tag. ✓
Attributes that are not set with an explicit value; implicitly true by the attribute name's presence.	Boolean ✓
The root element	html ✓
Semantic	Related to meaning ✓
Additional information about an element set inside the start tag.	Attribute ✓
Tag	The parts of an element inside angle brackets. ✓

Your answer is correct.

The correct answer is: Void → An element that has only one tag., Attributes that are not set with an explicit value; implicitly true by the attribute name's presence. → Boolean, The root element → html, Semantic → Related to meaning, Additional information about an element set inside the start tag. → Attribute, Tag → The parts of an element inside angle brackets.



It is valid to set only the width or height attribute for an <img> or <video> element. You do not need to set both; if you set only one, the browser will automatically resize the other dimension to match the width/height ratio of the original image or video.

Select one:

[✓ True] ✓

False

You can set both width and height if you want, but leaving one out will result in the other dimension being updated to keep the original ratio.

The correct answer is 'True'.

Happy learning!