Introduction to HTML

LAB

HTML

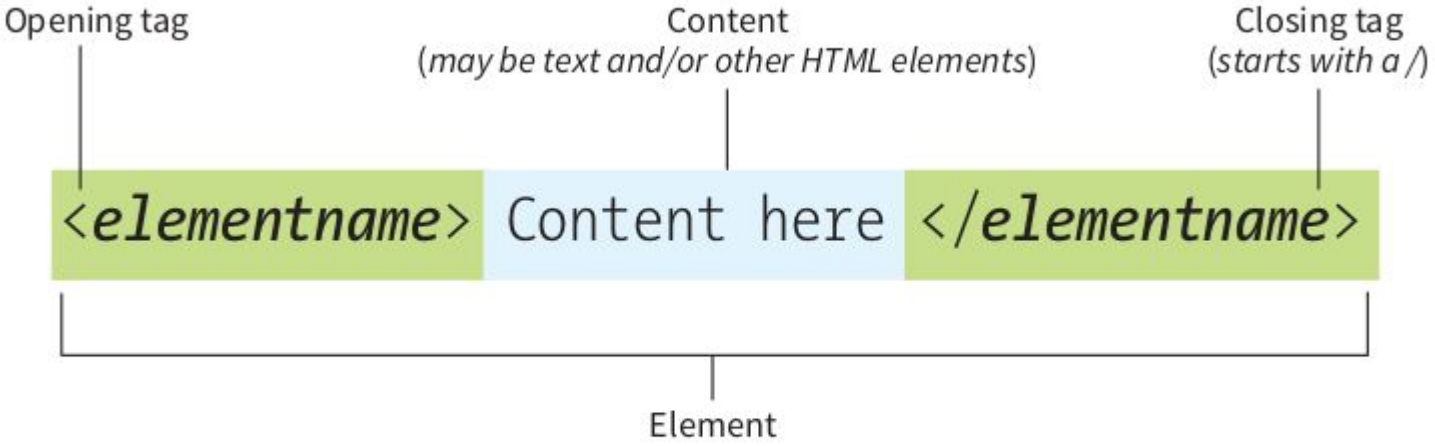
HTML code is made up of characters that live inside **angled brackets** called **HTML elements**

HTML Elements are usually made up of two **tags**:

an opening tag and a closing tag

Each HTML element tells the browser something about the information that sits between its opening and closing tags

HTML Tag



Example: `<h1>Black Goose Bistro</h1>`

HTML Tag Attributes

Attributes provide additional information about the contents of an element

They appear on the opening tag of the element and are made up of two parts: a name and a value, separated by an equals sign

Basic HTML document structure

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8">
        <title>Document</title>
    </head>
    <body>


    </body
</html>
```

Basic HTML document structure

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8">
        <title>Document</title>
    </head>
    <body>

    </body
</html>
```

DOCTYPE identifies the document as written in **HTML5**

Basic HTML document structure

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8">
        <title>Document</title>
    </head>
    <body>

    </body
</html>
```

The **`html`** element is called the **root element** because it contains all the elements in the document

Basic HTML document structure

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8">
        <title>Document</title>
    </head>
    <body>

    </body
</html>
```

The **head** element contains elements that pertain to the document that are not rendered as part of the content, such as its **title**, **style sheets**, **scripts**, and **metadata**

Basic HTML document structure

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8">
        <title>Document</title>
    </head>
    <body>

    </body
</html>
```

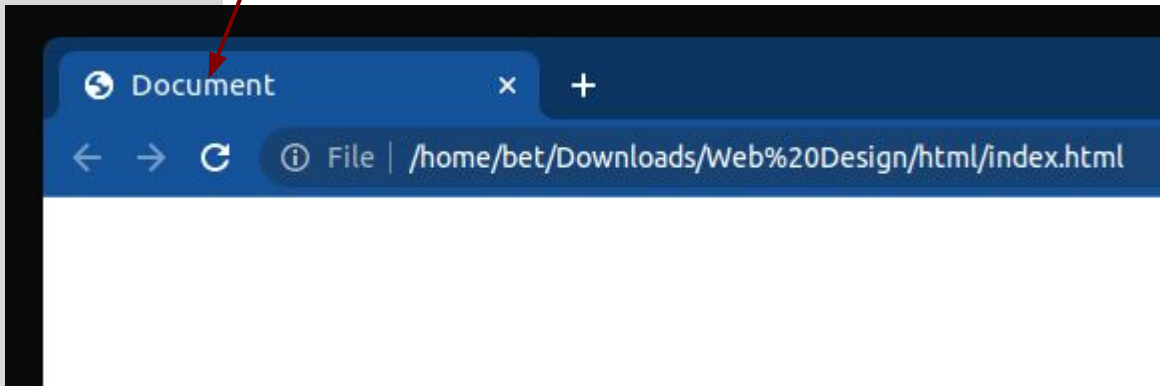
`meta` elements provide document metadata, information about the document

Basic HTML document structure

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8">
        <title>Document</title>
    </head>
    <body>


    </body
</html>
```

`title` mandatory element



Basic HTML document structure

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8">
        <title>Document</title>
    </head>
    <body>

    </body
</html>
```

body element contains everything that we want to show up in the browser window

Exercise

Create a folder named “**Web Dev Course**”

Open the folder in **VS Code** editor

Create a file named **index.html**

Create the basic HTML5 Structure

Basic HTML Tags

Paragraph Tag:

```
<p> Paragraph Content </p>
```

Exercise

Go to the https://www.lipsum.com/ website

Create one paragraph

Inside the body tag add a paragraph tag and put the content you got from the above website inside the opening and closing paragraph tag

Header Tag: <h1></h1> … <h6></h6>

```
<h1>I am the title of the story.</h1>
```

Exercise

Add six paragraphs

Above each paragraph add a header starting from `<h1></h1>` up to `<h6></h6>`

Unordered List:

```
<ul>
  <li>item t</li>
  <li>item a</li>
  <li>item x</li>
  <li>item e</li>
</ul>
```

Exercise

Add a list of courses inside the body tag

Ordered Lists:

```
<ol>
  <li>item 1</li>
  <li>item 2</li>
  <li>item 3</li>
  <li>item 4</li>
</ol>
```

Exercise

Add ordered course lists inside the body tag

Exercise

Create a nested list

- Show three ethiopian regions with three Cities/Towns under each region

Emphasis and importance

What is the difference between the following statements, in meaning

I am glad you weren't late.

I am *glad* you weren't *late*.

Emphasis:

When we want to add emphasis in spoken language, we stress certain words, subtly altering the meaning of what we are saying

Similarly, **in written language we tend to stress words by putting them in italics**

```
<p>I am <em>glad</em> you weren't <em>late</em>.</p>
```

Strong Importance:

To emphasize important words, we tend to stress them in spoken language and bold them in written language.

```
<p>This liquid is <strong>highly toxic</strong>.</p>

<p>I am counting on you. <strong>Do not</strong> be late!</p>
```

Exercise

Add a paragraph and use `em` and `strong` tags to emphasis or show importance of words

Creating hyperlinks

Hyperlinks allow us to **link** documents to other documents or resources, link to specific parts of documents, or make apps available at a web address

Link: <a>

A basic link is created by wrapping the text or other content inside an **<a>** element and using the **href** attribute, also known as a **Hypertext Reference**, or **target**, that contains the web address.

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

Adding supporting information with the title attribute

```
<a
    href="https://www.google.com"
    title="Search any thing">
    Go to Google
</a>
```

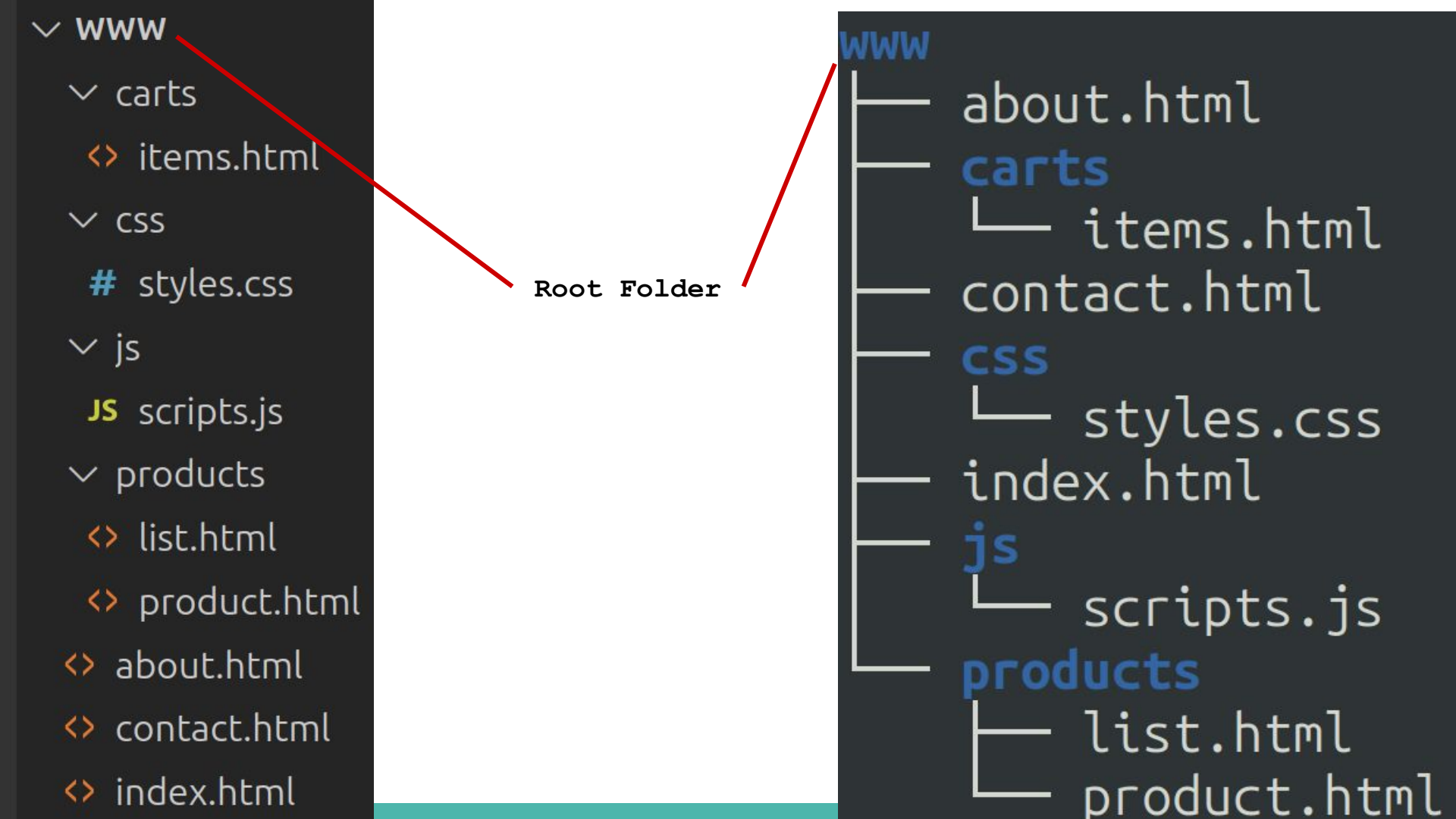
URLs and Paths

A **`URL`**, or **`Uniform Resource Locator`** is a string of text that defines where something is located on the Web

For example Google's home page is located at `https://www.google.com`

`URL`s use paths to find files. Paths specify where the file you're interested in is located in the filesystem

The following slide shows an example file hierarchy


www
carts
items.html
css
styles.css
js
scripts.js
products
list.html
product.html
about.html
contact.html
index.html
Root Folder
www
about.html
carts
items.html
contact.html
css
styles.css
index.html
js
scripts.js
products
list.html
product.html

Same Directory File Reference

If you wanted to include a hyperlink inside **`index.html`** (the top level **`index.html`**) pointing to **`contact.html`** or **`about.html`**, you would specify the filename that you want to link to

```
<a href="contacts.html">contacts page</a>
```

Sub Directory File Reference

If you wanted to include a hyperlink inside **`index.html`** (the top level **`index.html`**) pointing to **`products/list.html`**, you would need to go down into the projects directory before indicating the file you want to link to

```
<a href="products/list.html">Products</a>
```

Parent Directory File Reference

If you wanted to include a hyperlink inside **`products/list.html`** pointing to **`index.html`**, you'd have to go up a directory level

```
<a href="../index.html">Home</a>
```

Parent Directory File Reference

If you wanted to include a hyperlink inside `products/list.html` pointing to `carts/items.html`, you'd have to go up a directory level, then back down into the `carts` directory

```
<a href="../carts/items.html">Carts</a>
```

Parent Directory File Reference

To go up a directory, use two dots `..`

You can combine multiple two dots to go up further

```
../../data/config.csv
```

Exercise

Create the directories and files shown on the right and create links to move from one document to another as shown below

index.html -> products/list.html

index.html -> about.html

products/list.html -> index.html

carts/items.html -> product.html

product.html -> index.html

Absolute versus relative URLs

Absolute URL

Points to a location defined by its absolute location on the web, including protocol and domain name.

For example, if an `list.html` page is uploaded to a directory called `products` that sits **inside the root of a web server**, and the website's domain is `https://www.example.com`, the page would be available at `https://www.example.com/products/list.html`

Absolute versus relative URLs

Relative URL

Points to a location that is relative to the file you are linking from

Exercise

Use Absolute URLs to move from page to page as shown below

```
index.html -> products/list.html

index.html -> about.html

products/list.html -> index.html

carts/items.html -> product.html

product.html -> index.html
```

Document fragments

It's possible to link to a specific part of an HTML document, known as a document fragment, rather than just to the top of the document

To do this you first have to assign an id attribute to the element you want to link to

It normally makes sense to link to a specific heading, so this would look something like the following:

```
<h2 id="intro">Introduction</h2>

<a href="#intro">Go to Introduction</a>
```

Exercise

Create the an HTML document with the following structure

- Title (use h1 tag here)
- Topic 1 (use h2 tag here)
- Topic 2 (use h2 tag here)
- Topic 3 (use h2 tag here)
- Conclusion (use h1 tag here)

Under each topic and conclusion add lorem ipsum paragraphs

Exercise

Give **id**s to each tag

Create links at the top of the page that takes the user to the specific section when clicked

Create another page and create a link on the other page that takes the user to a specific section of the page you have created before

Exercise

Give **id**s to each tag

Create links at the top of the page that takes the user to the specific section when clicked

Linking to Non-HTML resources

```
<a href="https://www.example.com/report.pdf">
  Open Report
</a>
```

Linking to Non-HTML resources

Open in new tab

```
<a href="https://www.example.com/videos/"
target="_blank">

    Watch the video (opens in separate tab)

</a>
```

Linking to Non-HTML resources

Use the **`download`** attribute when linking to a download

The download attribute allows you to provide a default save filename

```
<a
  href="https://example.com/report.pdf"
  download="annual-report.pdf">
  Download Annual Report
</a>
```

Exercise

Link to a video, audio, and file resources/documents

- Open the document on same page
- Open the document on new tab
- Download the document with a default name

Email links

```
<a href="mailto:admin@example.com">Contact Admin</a>
```

Email with details

```
<a
    href="mailto:admin@example.com?cc=support@example.com&bcc
    =info@example.com&subject=The%20could%20not%20login&body=
    I%20got%20invalid%20password%20message">

    Send mail with cc, bcc, subject and body

</a>
```

Exercise

Add email link and see what happens when you click the link

Description lists

The purpose of these lists is to **mark up a set of items and their associated descriptions**, such as **terms** and **definitions**, or **questions** and **answers**

```
<dl>
   <dt>Term</dt>
      <dd>
      Definition
      </dd>
   <dt>Question</dt>
      <dd>
      answer
      </dd>
</dl>
```

Description list

Can have multiple descriptions for a given term

```
<dl>
   <dt>Term</dt>
      <dd>
      Definition 1
      </dd>
      <dd>
      Definition 2
      </dd>
</dl>
```

Exercise

Create a term and definition list for the following words

abject, conduit, demagogue

Create a list of question and answer for basic country information such as area, population, continent

Quotations

Blockquotes

Inline quotations

Blockquotes

If a section of block level content (a paragraph, multiple paragraphs, a list, etc.) is quoted from somewhere else, you should wrap it inside a **<blockquote>** element to signify this, and include a **URL** pointing to the source of the quote inside a **cite** attribute

```
<blockquote cite="https://en.wikipedia.org/wiki/HTML">
    <p>
    HTML is the standard markup language for documents
    designed to be displayed in a web browser.
    </p>
</blockquote>
```

Exercise

Take some text about something from wikipedia and use `<blockquote>` to display the content with citation

Inline quotations

```
<q cite="https://en.wikipedia.org/wiki/HTML">
    HTML is the standard markup language for documents
    designed to be displayed in a web browser.
</q>
```

Exercise

Add inline quotes

Abbreviations

`<abbr>`

is used to wrap around an **abbreviation** or **acronym**

```
<p>
We use <abbr>HTML</abbr>, Hypertext Markup Language, to
structure our web documents.
</p>
```

Abbreviations

`<abbr>`

is used to wrap around an **abbreviation** or **acronym**

```
<p>
We use <abbr title=”Hypertext Markup Language”>HTML</abbr>,
to structure our web documents.
</p>
```

Exercise

Address

<address>Addis Ababa Institute of Technology, 5 Kilo</address>

Address

```
<address>
  <p>
    Mr. Student<br />
    Software Student<br />
    5 Kilo
  </p>
  <ul>
    <li>Tel: +2510911111111</li>
    <li>Email:student@aait.edu.et</li>
  </ul>
</address>
```

Superscript and subscript

```
<p>
  Caffeine's chemical formula is
    C<sub>8</sub>H<sub>10</sub>N<sub>4</sub>O<sub>2</sub>.
</p>
<p>If x<sup>2</sup> is 9, x must equal 3 or -3.</p>
```

Exercise

Add the following formulas

$C_5H_{11}NO_2S$

n^3-2n^2

Representing computer code

`<code>`

For marking up generic pieces of computer code

`<pre>`

For retaining whitespace (generally code blocks)

Your whitespace will be rendered identically to how you see it in your text editor

Representing computer code

`<var>`

For specifically marking up variable names

`<kbd>`

For marking up keyboard (and other types of) input entered into the computer

`<samp>`

For marking up the output of a computer program

Representing computer code

```
<pre>
    <code>
        const para = document.querySelector('p');
        para.onclick = function() {
          alert('Paragraph clicked');
        }
</code>
</pre>
```

Representing computer code

```
<p>

    Select all the text with <kbd>Ctrl</kbd>/<kbd>Cmd</kbd> +
    <kbd>A</kbd>

</p>
```

Representing computer code

```
<pre>
    $ <kbd>ping 127.0.0.1</kbd>
    <samp>
     PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
     64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.027 ms
     64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.074 ms
    </samp>
</pre>
```

Times and Dates

Allows you to attach machine-readable time/date

```
<time datetime="2023-10-27">27 October 2023</time>
```

Images

```
<img src="cat.jpg" alt="Cat" />

<img src="images/cat.jpg" alt="Cat" />

<img src="https://www.cats.com/images/cat.jpg" alt="Cat" />

<img src="images/cat.jpg" alt="A Cat image" />
```

Images Width and Height

```
<img src="images/cat.jpg" alt="A Cat image" width="400"
height="341" />
```

The height and width are given as integers without a unit, and represent the image's width and height in pixels

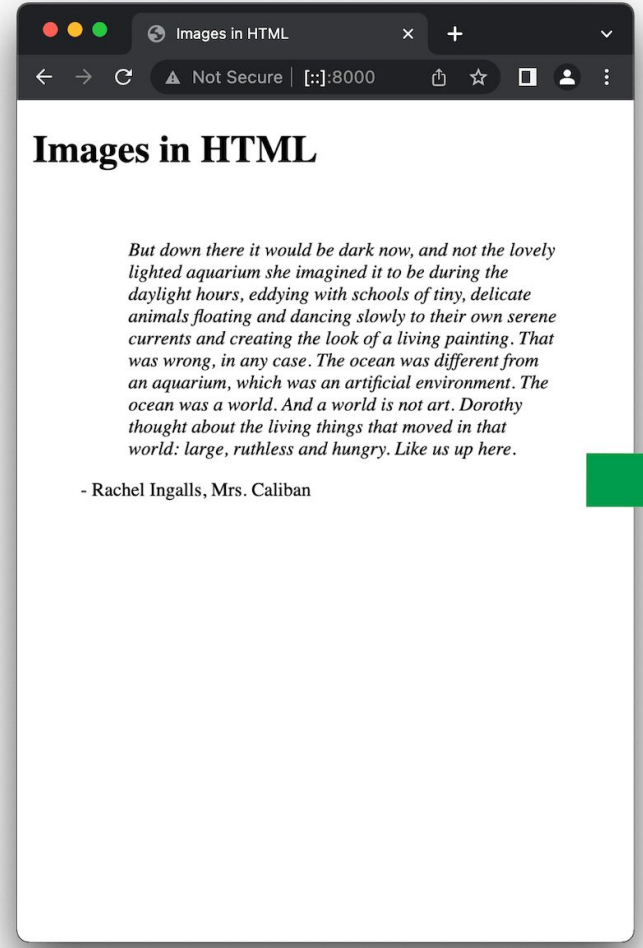
Why do you need to provide image size

The HTML for your page and the image are separate resources, fetched by the browser as separate HTTP(S) requests

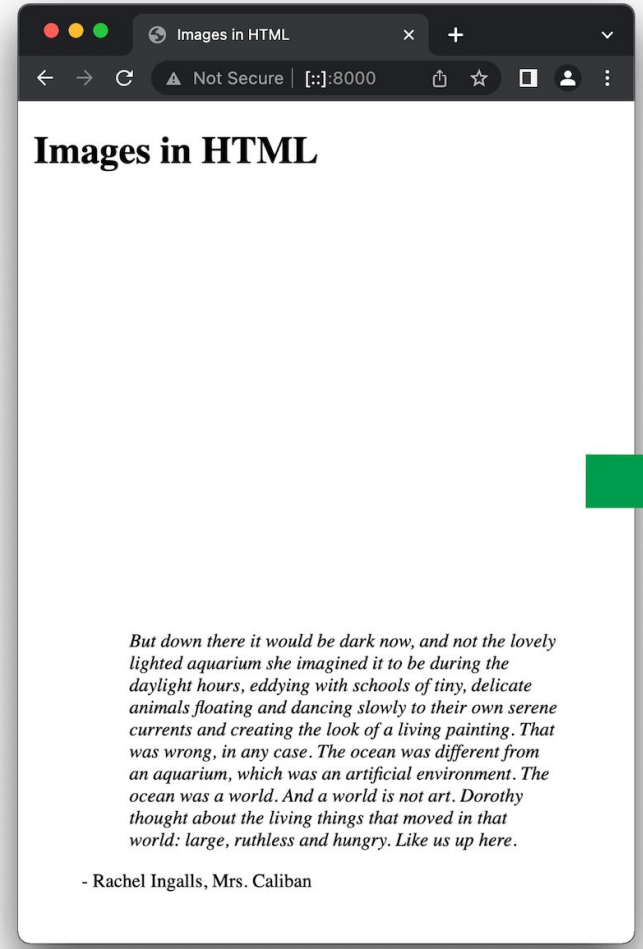
As soon as the browser has received the HTML, it will start to display it to the user

If the images haven't yet been received, as image file sizes are often much larger than HTML files, then the browser will render only the HTML, and will update the page with the image as soon as it is received

The following two slides show the browser behavior with or without image size specified


Images in HTML
Not Secure | [::]:8000
Images in HTML
But down there it would be dark now, and not the lovely lighted aquarium she imagined it to be during the daylight hours, eddying with schools of tiny, delicate animals floating and dancing slowly to their own serene currents and creating the look of a living painting. That was wrong, in any case. The ocean was different from an aquarium, which was an artificial environment. The ocean was a world. And a world is not art. Dorothy thought about the living things that moved in that world: large, ruthless and hungry. Like us up here.
- Rachel Ingalls, Mrs. Caliban


Images in HTML
Not Secure | [::]:8000
Images in HTML
But down there it would be dark now, and not the lovely lighted aquarium she imagined it to be during the daylight hours, eddying with schools of tiny, delicate animals floating and dancing slowly to their own serene currents and creating the look of a living painting. That was wrong, in any case. The ocean was different from an aquarium, which was an artificial environment. The ocean was a world. And a world is not art. Dorothy thought about the living things that moved in that world: large, ruthless and hungry. Like us up here.
- Rachel Ingalls, Mrs. Caliban


Images in HTML
Not Secure | [::]:8000
Images in HTML
But down there it would be dark now, and not the lovely lighted aquarium she imagined it to be during the daylight hours, eddying with schools of tiny, delicate animals floating and dancing slowly to their own serene currents and creating the look of a living painting. That was wrong, in any case. The ocean was different from an aquarium, which was an artificial environment. The ocean was a world. And a world is not art. Dorothy thought about the living things that moved in that world: large, ruthless and hungry. Like us up here.
- Rachel Ingalls, Mrs. Caliban


Images in HTML
Not Secure | [::]:8000
Images in HTML
But down there it would be dark now, and not the lovely lighted aquarium she imagined it to be during the daylight hours, eddying with schools of tiny, delicate animals floating and dancing slowly to their own serene currents and creating the look of a living painting. That was wrong, in any case. The ocean was different from an aquarium, which was an artificial environment. The ocean was a world. And a world is not art. Dorothy thought about the living things that moved in that world: large, ruthless and hungry. Like us up here.
- Rachel Ingalls, Mrs. Caliban

figure and figure figcaption

```
<figure>
  <img src="cat.jpg" alt="A Cat" width="400" height="341"/>
  <figcaption>
    A Cat sleeping.
  </figcaption>
</figure>
```

`figure` and figure `figcaption`

A figure doesn't have to be an image

It could be **several images**, a **code snippet**, **audio**, **video**, **equations**, a **table**, or something else.

The <video> element

```
<video src="river-flow.webm" controls>
  <p>
    Your browser doesn't support HTML video. Here is a
    <a href="river-flow.webm">link to the video</a> instead.
  </p>
</video>
```

Multiple source formats for compatibility

```
<video controls>
  <source src="river-flow.mp4" type="video/mp4" />
  <source src="river-flow.webm" type="video/webm" />
  <p>
    Your browser doesn't support this video. Here is a
    <a href="river-flow.mp4">link to the video</a> instead.
  </p>
</video>
```

Multiple source formats for compatibility

https://developer.mozilla.org/en-US/docs/Web/Media/Formats/Containers#choosing_the_right_container

https://developer.mozilla.org/en-US/docs/Web/Media/Formats/Video_codecs#choosing_a_video_codec

The <video> element

```
<video width="400" height="400"
  control autoplay loop muted preload="auto"
  poster="poster.png">
  <source src="river.mp4" type="video/mp4" />
  <source src="river.webm" type="video/webm" />
</video>
```

The <audio> element

```
<audio controls>
  <source src="song.mp3" type="audio/mp3" />
  <source src="song.ogg" type="audio/ogg" />
  <p>
    Your browser doesn't support this audio file. Here is a
    <a href="song.mp3">link to the audio</a> instead.
  </p>
</audio>
```

Displaying video text tracks

You can use the `WebVTT` file format and the `<track>` element to add transcripts

Example WebVTT file

```
WEBVTT

1
00:00:22.230 --> 00:00:24.606
This is the first subtitle.

2
00:00:30.739 --> 00:00:34.074
This is the second.

…
```

How to add transcript

1. Save the **WebVTT** file as a **.vtt** file
2. Link to the **.vtt** file with the **<track>** element.

   <track> should be placed within <audio> or <video>, but after all <source> elements.

3. Use the kind attribute to specify whether the cues are subtitles, captions, or descriptions.
4. Use srclang to tell the browser what language you have written the subtitles in.
5. Add label to help readers identify the language they are searching for.

Displaying video text tracks

```
<video controls>
 <source src="song.mp4" type="video/mp4" />
 <source src="song.webm" type="video/webm" />
 <track kind="subtitles" src="subtitles_am.vtt" srclang="am"
label="Amharic" />
</video>
```

Vector Graphics

On the web, you'll work with **two types of images**

raster images, and

vector images

Raster Images

are defined using a **grid of pixels**

a raster image file contains information showing exactly where each pixel is to be placed, and exactly what color it should be

popular web raster formats include

Bitmap (`.bmp`), PNG (`.png`), JPEG (`.jpg`), and GIF (`.gif`.)

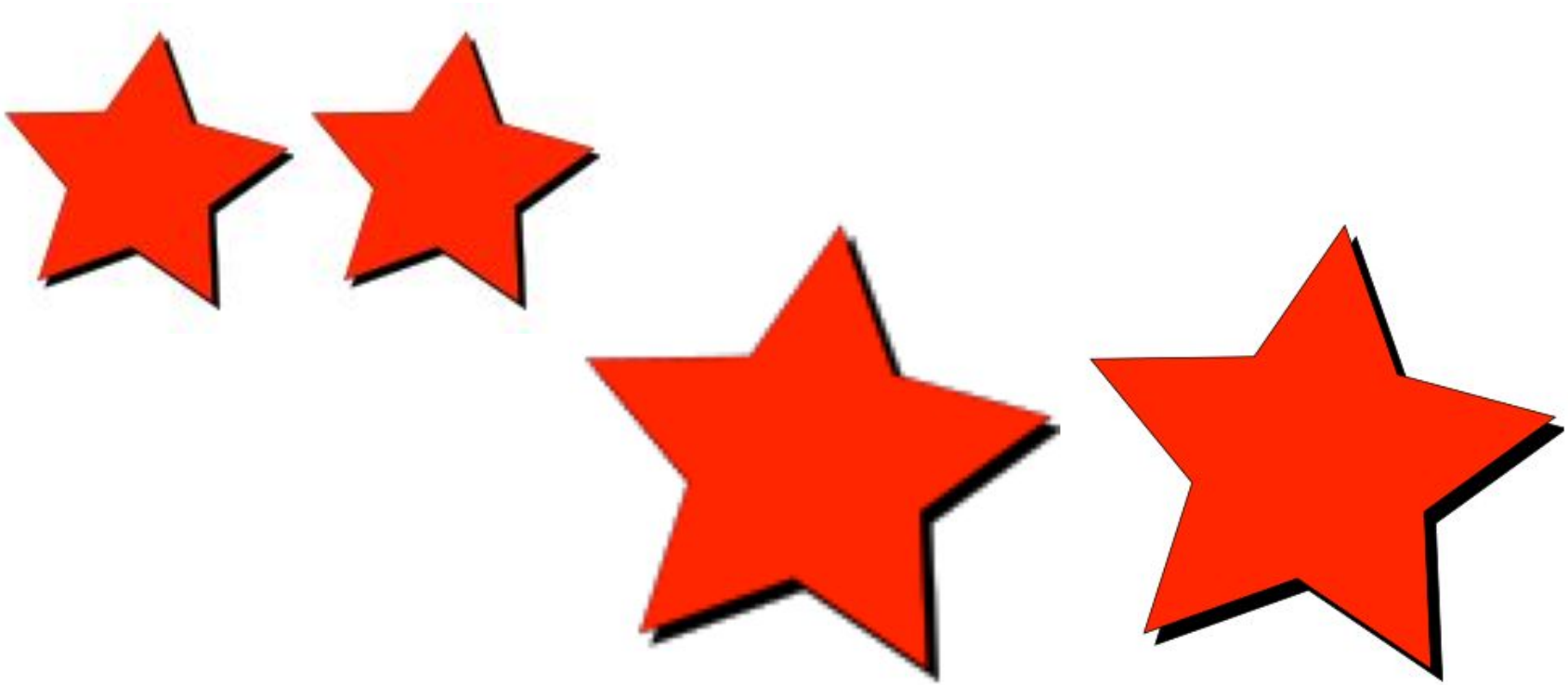
Vector Image

are defined using algorithms

a vector image file contains shape and path definitions that the computer can use to work out what the image should look like when rendered on the screen.

the SVG format allows us to create powerful vector graphics for use on the Web

Raster vs Vector Image



What is SVG?

SVG is an XML-based language for describing vector images

```
<svg
  version="1.1"
  baseProfile="full"
  width="300"
  height="200"
  xmlns="http://www.w3.org/2000/svg">
  <rect width="100%" height="100%" fill="black" />
  <circle cx="150" cy="100" r="90" fill="blue" />
</svg>
```

Adding SVG to your pages

```
<img
  src="equilateral.svg"
  alt="triangle with all three sides equal"
  height="87"
  width="100" />
```

Adding SVG to your pages

```
<svg width="300" height="200">
  <rect width="100%" height="100%" fill="green" />
</svg>
```

Responsive images

```
<picture>
  <source media="(max-width: 799px)" srcset="cat-480w.jpg" />
  <source media="(min-width: 800px)" srcset="cat-800w.jpg" />
  <img src="cat-800w.jpg" alt="A cat" />
</picture>
```

Table

```
<table>
     <tr>
       <td>1.1</td>
       <td>1.2</td>
     </tr>
     <tr>
       <td>2.1</td>
       <td>2.2</td>
     </tr>
</table>
```

Table with header

```
<table>
     <tr>
       <th>Col 1</th>
       <th>Col 2</th>
     </tr>
     <tr>
       <td>1.1</td>
       <td>1.2</td>
     </tr>
</table>
```

`Colspan` and `Rowspan`

To make a cell span over multiple columns, use the **`colspan`** attribute:

- The value of the **`colspan`** attribute represents the number of columns to span

To make a cell span over multiple rows, use the **`rowspan`** attribute

- The value of the **`rowspan`** attribute represents the number of rows to span

Exercise

Time Table					
Hours	**Mon**	**Tue**	**Wed**	**Thu**	**Fri**
	Science	Maths	Science	Maths	Arts
	Science	Maths	Science	Maths	Arts
	LUNCH				
	Science	Maths	Science	Maths	Project
	Science	Maths	Science	Maths	

Exercise

Go to https://validator.w3.org/ and write a valid HTML document structure and modify until all errors and warnings disappear

Check if the any of the elements in basic HTML structure can be left out or can have empty content