

Module 3-1

HTML Review

HTML Review: Language Introduction

- HTML (Hyper Text Markup Language) is a declarative language interpreted by internet browsers.
- Unlike Java, there is no separate step to compile the code, the instructions written in HTML are simply interpreted by the browser.

HTML Review: A Semantic Language

- HTML is a semantic language based on tags. Here are a few common ones:
 - `<html>...</html>`
 - `<head>...</head>`
 - `<body>...</body>`
 - `<nav>...</nav>`
- The rules for tags are straightforward:
 - There is a begin tag and an end tag, the end tag is denoted by the slash.
 - Some tags can be defined in a self-closing manner. For example, the tag for an image can be constructed like so:
 - `` as an alternative to ``

HTML File Structure (1/3)

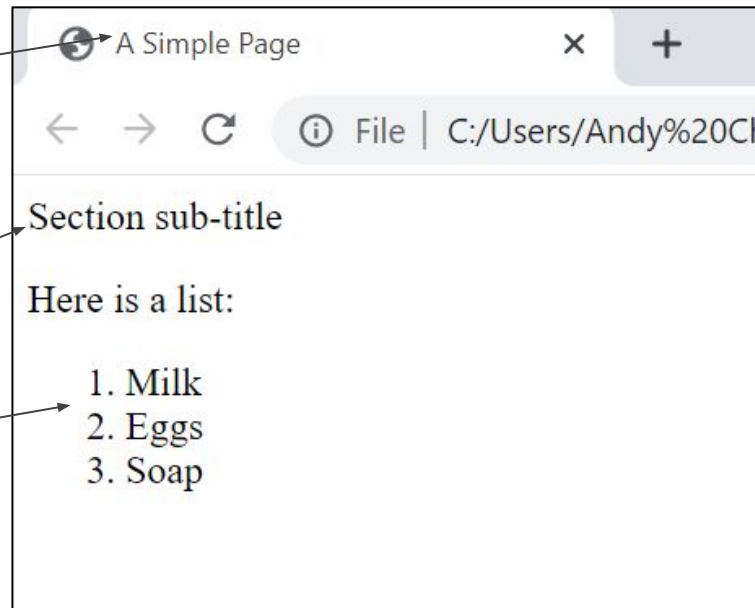
- A HTML file is named with an .html extension on the file.
- index.html is special, it's the first one a server will look at.
 - In the absence of a specific resource in the URL, the user will be shown the index's content.

HTML File Structure (2/3)

- Within the file, all content is enclosed within `<html>` tags.
 - Inside, there could be other tags
 - ... which could in turn contain other tags

HTML File Structure (3/3)

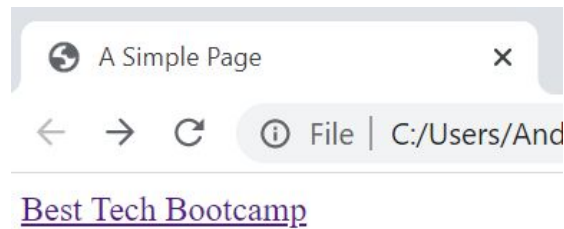
```
<html>
  <head>
    <title>A Simple Page</title>
  </head>
  <body>
    <section>
      <header> Section sub-title </header>
      <p>Here is a list: </p>
      <ol>
        <li>Milk</li>
        <li>Eggs</li>
        <li>Soap</li>
      </ol>
    </section>
  </body>
</html>
```



Creating Links

We can connect several HTML pages together with links, here is an example:

```
<html>
  <head>
    <title>A Simple Page</title>
  </head>
  <body>
    <a href='https://www.techelevator.com/'>Best Tech Bootcamp</a>
  </body>
</html>
```



Let's write some HTML

Creating Forms (1/3): The Form Tag

- HTML (Hyper Text Markup Language) is a declarative language interpreted by internet browsers.

```
<form method="POST" action="https://example.com/api/reservation">  
    <!-- By the way, this is a comment: form elements go in here -->  
</form>
```

- The form method specifies the request type (GET, POST, PUT, DELETE, etc.)
- The action specifies where the form data is being sent.

Creating Forms (2/3): Form Elements

— — —

- Within the `<form>` tag are elements:

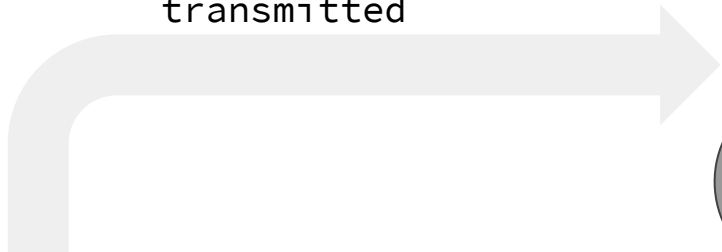
```
<div>  
<label for="id-of-field">Name:</label>  
<input id="id-of-field" name="name-of-field" type="text" value="so many choices!" />  
</div>
```

- The `label` and the `input` tags are associated via the `for` attribute.
- Various types of inputs are available (names are self explanatory):
number, date, type, checkbox, etc.
There is a special form element called an input, which generates a “send” button.

What happens when forms are sent?

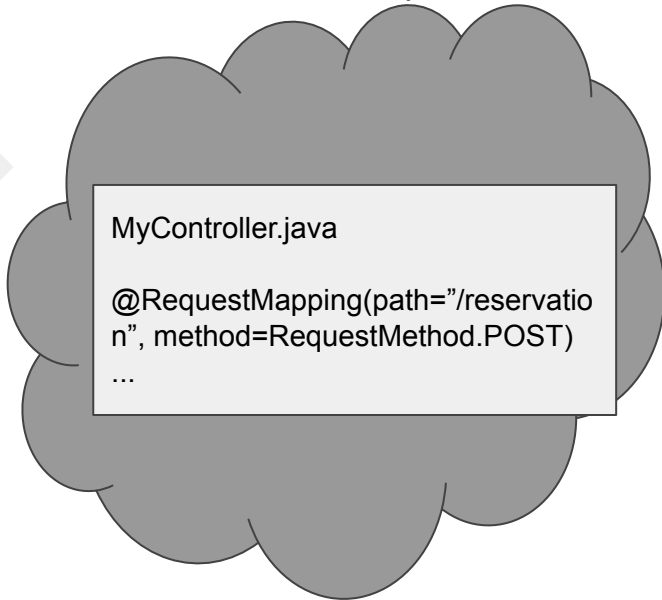
— — —

Data in JSON {...}
format is
transmitted



```
<form method="POST"  
action="https://example.com/api/reservation">  
...  
</form>
```

A server endpoint
is listening for
these requests:



Let's create a form

Formatting with CSS

— — —

- Cascading Style Sheets (CSS) define rules on how we should format the elements of the HTML page.
- It is good practice to separate “how to display (CSS)” with “what to display (HTML)”
 - Consequently, we always include the CSS code on a separate document, and reference this document in our HTML:

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Bits & Bytes Restaurant</title>
  <link rel="stylesheet" href="styles.css" />
</head>
```

What is CSS?

- CSS (Cascading Style Sheets) are used to change the default style of HTML elements.
 - Some of the things we've seen so far are considered style changes - borders, margins, making text bold, positioning, etc.
- CSS code can be included within the `<head>...</head>` section of a HTML document or it can be included in its own separate file.
 - The latter is much preferred.
 - Files names should end with a `*.css` extension.

CSS Selectors

- CSS uses selector to determine which HTML elements will be “targeted” or selected to have a specific format.
- Generally speaking, there will be a CSS block that looks like so:

```
SELECTOR {  
  
    attribute : attribute value  
  
}
```

- We will start discussion three most important types of selectors in the next section.

CSS Example

— — —

```
body {  
  background-color: lavender;  
}  
  
p {  
  font-family: Arial;  
  color: gray;  
}  
  
h1 {  
  text-align: center;  
  text-decoration: underline;  
}
```

affects

```
<html>  
  <head>  
    <title>A Simple Page</title>  
    <link rel="stylesheet" href="style.css">  
  </head>  
  <body>  
    <h1>Important Words</h1>  
    <p>  
      Lorem ipsum dolor sit amet, consectetur adipiscing elit.  
      Donec quis sem ornare lacus sodales auctor ac eu odio.  
    </p>  
  </body>  
</html>
```

What the browser
renders:

Important Words

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec quis sem ornare lacus sodales auctor ac eu odio.

Let's write some CSS