Lesson 7: Intro to Front-End and HTML

[Overview](#h.vqbhj2q8jt98)

[Intro to Front-End](#h.lgvd9kxkapo2)

[HTML](#h.6mpd7j6emevx)

[We Do: Hello World](#h.4a180y24viz0)

[Basic Tags](#h.2jud9c4489ej)

[Links](#h.zhtv0ukyd2yj)

[Lists](#h.jy59yu1hfx9n)

[Comments](#h.wqxcu9cmmg0z)

[Style Teaser](#h.ofzqgi8f5f7v)

[Tables](#h.rd6c7izdspo)

[Div](#h.7bo910fz4c7j)

[Span](#h.61kky7dyne6q)

# Overview

1. What are the major elements of front-end pages?
2. Document structure (file type, head, body)
3. Tags (open, close, self-closing)
4. Style tags, links, images, tables, comments
5. Inline style (CSS teaser)
6. Divs and spans

# Intro to Front-End

Front-end development is the development of those elements of a website that the user sees and interacts with directly. It is a combination of programming skills and aesthetics (understanding element arrangements on the screen, the color and font choices).

Front-end development uses 3 key languages or tools: HTML, CSS, and JavaScript.

The browser:

* parses the **HTML** to build a hierarchical model of the document;
* applies the **CSS** rules to the elements to render the model as a page on-screen; and
* runs the **Javascript** in the page to animate or manipulate both the model and the CSS on-screen in reaction to the user's clicks and keypresses.

One way to think of this is with a grammar analogy:

* HTML does the nouns: <p></p> means "paragraph";
* CSS does the adjectives: p {color:#000;} means that paragraph text is black;
* Javascript does verbs: window.open() means "open a window"

Today, we are going to focus on HTML.

# HTML

HTML stands for Hyper Text Markup Language.

What is a **markup language**? It’s a system for marking or tagging a document that indicates its logical structure (like headers and paragraphs) and gives instructions for its layout on the page.

Go to <http://www.w3schools.com/html/html_intro.asp> to show the HTML example and talk through it:

* The **DOCTYPE** declaration defines the document type to be HTML
* **<html>** describes an **HTML document**
* **<head>** provides **information** about the document such as the title for the document, scripts, styles, meta information, and more.
* **<title>** provides a **title** for the document - as seen in the top of your browser tab
* **<body>** describes the **visible page content**
* <h1> describes a heading
* <p> describes a paragraph

## We Do: Hello World

Create a new folder in TTS called front-end. Using Sublime, create a new file in there called hello.html. Type “html” then hit tab. Inside the body, type “Hello World!” Save and open in Chrome.

## Basic Tags

We use **carets** a lot in HTML! We use an open caret < and close carets > to denote tags <>.

Anything inside of the carets is considered a tag, or an attribute of the tag.

Tags almost always come in **pairs**. With an open tag like <p> and close tag </p>. We SHOULD ALWAYS CLOSE a tag with its closing tag which is denoted with a /. Think of tags as parenthesis.

Add h1, h2, h3 tags and a p tag to hello.html.

While most tags need an open and a close, some tags are **self-closing**. These are called self closing tags, here are a few that don't need a partner:

* <br>
* <img>
* <hr>
* <meta>

Add br and img (image from the internet).

About images… it’s very important to use the **image alt** attribute of an image tag for those people who are vision impaired. Always, always add a text description of the image inside the alt attribute whenever you add an image. alt=”description of image”

The **image source** can be a file within your folder tree or a location on the internet.

We can also **nest** tags. Which means we can have certain tags living within other tags.

<p><strong> Example </strong></p>

Make sure to close them in order! Add strong and em tags. Mention that we don’t use b and i anymore.

## Links

We create links with an <a> tag which stands for **anchor**.

We include the **href="url"** attribute, which stands for hyperlink reference and it points to the specific url that we want, either a file path or an entire url.

<a href="/another-page.html">This is the link text</a>  
<a href="http://techtalentsouth.com/">This is the link text</a>

Everything inside of the actual tags is called **anchor text**. i.e. the blue text of a link. We can also set the link to open up in a new tab by including **target**=”\_blank” inside the a tag.

Classroom challenge: In hello.html, add a link to your github profile page that opens up in a new tab.

We can make images links by nesting them inside the a tag. We can actually make anything a link by putting it inside the a tag.

Classroom challenge: Change your github link content to one of the Octocat images: <https://octodex.github.com/>

## Lists

There are two types of lists - ordered and unordered - denoted as <ol> and <ul>. Each individual bullet or number is denoted as a list item, or the <li> tag.

Classroom challenge: Add an unordered list of your favorite foods, and an ordered list of steps to do something.

## Comments

Comments are made by using:

<!-- This is a comment -->

Make a comment in your file.

## Style Teaser

We can style elements with inline css like so:

<h1 style="font-size: 75px; color: red;">This is huge and red!</h1>

Here are some basic styling attributes:

* font-size
* font-family
* color
* background or background-color
* text-align

Add at least 3 different style elements to your page.

## Tables

Stock Rails uses tables a lot - they are automatically created whenever we “scaffold” a database table.

First, we start with a “table” tag, then we add thead and tbody to section off our header and body. Next, we use “th” for each table header and “tr” for each table row. Inside table rows, we use “td” for table data.

<table border="1px">

<th>Name</th>

<th>Spirit Animal</th>

<tr>

<td>Sia</td>

<td>Dolphin</td>

</tr>

<tr>

<td>Gant</td>

<td>Taco</td>

</tr>

</table>

We can also make a 2-row header and merge columns:

<thead>

<tr>

<th colspan="2">Class Spirit Animals</th>

</tr>

<tr>

<th>Name</th>

<th>Spirit Animal</th>

</tr>

</thead>

## Div

The div tag is one of the more interesting tags. Div is short for division. It allows us to divide up our page into containers or different pieces. These don’t really do much without css, but they are super versatile with css.

First, add 3 div tags:

<div>A div</div>

<div>Another Div</div>

<div>Also a Div</div>

What happened? Not much. Now add style:

<div style="width: 100px; height: 50px; background: red;">A div</div>

<div style="width: 100px; height: 50px; background: purple; color: white;">Another Div</div>

<div style="width: 100px; height: 50px; background: blue; color: white; font-weight: bold;">Also a Div</div>

## Span

And finally we come to a <span> tag which is very similar to a div, but is for smaller parts of your page, like specific text.

Wrap a word on your page in a:

<h1>I'm playing with <span style="color: blue;">span</span> tags!</h1>