

HTML TAGS

- We create a web page by editing a text file and saving it as filename.**html**
 - (HTML stands for hypertext markup language)
 - Once you create an html file, you can view it in a browser

- Web (html) **commands** are formatted like this `<command>`

There will almost always be a start and end command: `<command>` `</command>`

We start the file with these commands to tell the browser it's a web page

```
<!DOCTYPE html>
<html lang="en">
```

- Our file will have some major sections

```
<html>
<head>

</head>

<body>

</body>
                                this is where we put the main content
<script>
                                this is where we put our JavaScript!
</script>
</html>
```

- When writing our file, it's good to indent and use blank lines

- Some commands we learned:

- `<h1>` `<h2>` `<h3>` `<h4>` `<h5>` creates different sized headers
- `<title>` text `</title>` puts text in browser tab
- `<p>` starts a paragraph
- `` displays a picture
- `` *must be in same folder*
- `<hr>` draws a horizontal line
- `<!-- some comments -->` hidden comments

- Use blank lines to separate areas of code
- Use indents to show how code lines up

JAVASCRIPT FUNCTIONS

- JavaScript Code
 - We write our JavaScript functions in the `<script> </script>` section
 - Commands always end with a semicolon: `“;”`
 - To create variables: `var name = value;`
 - Functions start with a `{` and end with a `}`
 - Use `//` to comment out lines (these are not run)
- To create code that runs **when we first load the page**
 - `<body onload="test()">` this runs a function when we load a page
 - Here's the function `test` we wrote – which just displays a message
Look at how we use the `{` and `}` to show the start and end of the function!

```
<script>
    function test(){
        alert("it does");
    }
</script>
```
- To create code that runs **when we click a button**
 - Create the button in HTML
`<button onclick="function()"> Click me </button>`
 - Then create the JavaScript function named `function()`

- Most of our code will be in *functions* in the `<script> </script>` section.
But we can create variables that are used by all functions.

Like this:

```
<script>
    var randomNumber = Math.floor(Math.random() * 100) + 1;
    alert( randomNumber );
```

This lets us use *randomNumber* in our functions

- To get User Input
 - We can get input from people by using the `<input>` tag
`<p>Enter your name <input type='text' id='name'>`
 - JS can read it by looking at the `id`
`var name = document.getElementById("name");`
- To set a Random Number from 1 to 100
 - `var randomNumber = Math.floor(Math.random() * 100) + 1;`

HOW TO CODE

- **To change what's displayed**

- In HTML we identify a paragraph with an id

```
<p id='message'>
```

- We change it with JS!

```
var message = document.getElementById("message");  
message.value = "This is new text";  
message.innerHTML = "<b> This is bold text </b>";
```

- **To use image or sound files**

- The files should be in the same folder as our code

```
src="filename.jpg"
```

- We can also use a file on the internet

```
src="https://aptokash.github.io/Home/ZHeadShot0.jpg"
```

- **To change a picture**

- In HTML we identify an image with an id

```

```

- We change it with JS!

```
var ourvariable = document.getElementById("name");  
ourvariable.src = "face happy.jpg";
```

- **To play a sound file!**

- Note: sound will only work in a function called by a button!

```
var audio = new Audio('filename.filetype');  
audio.loop = false;  
audio.play();
```

CSS – Styles!

- To format how things are displayed, we use CSS
- We put them in a section of code
`<style>`

`</style>`
- We can format different HTML objects
 - To format a tag : `h1 { color: red; fontsize 10px;}`
`p { color: blue; }`
 - To format an id: `#div1 { color: green;}`
- Using ID's
 - We specify an id to an HTML tag: `<p id="message">`
 - We add css for the tag: `#message { color: red;}`

Function Variables

- We put all our JS code in the `<script>` section – we also put functions in there
`<script>`
`function dothis() {`
 `}`
`function dothat()`
 `}`
`</script>`
- We can but VAR statements in the functionIf we have variables that are shared among functions – these are called *global* variables and these we put outside of the functions
`<script>`
`var name = " Presley";`
`function dothis() {`
 `}`
`function dothat()`
 `}`
`</script>`

This week

- I put the ANIMATION file on [aptokash.github.io/PB](https://github.com/aptokash)
- Work: add more buttons to change more formatting of the block using JS and CSS