Week 1

1. **Document Object Model**

Tree like structure that describes how elements of a html page are related to one another. Root is always html element which is the parent element. Then you have child elements of head and body. Within the head element there is a title child element.

1. **Headers**

Headings tag exists within body tag in DOM. Headings are just titles

1. **Lists**

2 types of lists in html, ordered and unordered. Ordered lists will automatically have numbers attached I.E.

1. First item
2. Second Item
3. Third Item

Unordered lists will only have bullet points in front of them

* First item
* Second Item
* Third Item

Ordered lists are created with the tag <ol> while unordered lists are created with the tag <ul>. Both tags are found within the body tag. Within each lists you have a list item <li> which stands for what should actually appear in front of the number or bullet point.

1. **Images**

Html also allows us to embed other media types such as Videos or Images. To embed images we must use the img tag. Img tag takes 2 attributes (i) src – which is the path to the file to be displayed (ii) alt – words that should be displayed if image is unable to be generated for whatever reason. Img also has other optional attributes such as (i) width -> how many pixels large for width (ii) height – same as width

Img tags has no beginning and closing tag unlike the other tags as it contains nothing but a jpg file.

1. **Hyperlinks**

Links are created with the <a> tag which is the anchor tag.

1. **Forms**

Forms are how you get user input. They are many different types of forms offered by html. Create a form by having the form tag and within that form tag you must have input tag which specifies the type of input within that form.

Form types can be (i) text (ii) password (iii) radio

There is list attribute in inputs as well that indicates that you want the possible options to come from a list.

1. **CSS**

CSS can be used to style different elements. Best practice is to have a separate css file which contains all the styling for different tags and just have files link to this css file under their head tags.

1. Popular CSS properties

**Sizes**

Margin vs. Padding

Margin is the space around an element. Use margin o move element up or down a page or left and right. Margin is completely transparent and does not have any background colour. Clears the area around the element. Each side of the element has a margin size that can be changed individually i.e. margin right/left/top/bottom

Padding is the space between the element and the related contents inside. It determines how an element looks sitting inside a container. Shows the container background and element in it. Padding is affected by background colours and it clears the area around the content.

How to decide when to use which?

If width of page is fixed, centring an element horizontally is very simple just assign the margin: auto. Can use margin to set distance between nearby elements as well. Change padding if you want to create space between element and edge of container.

I.E. if you want distance between elements use margin, distance between elements and container use padding.

Note: Normally when styling html using css, the tags are normally in a tag such as div. A div is just something use to group html elements together and you can name divs. This div is therefore the entire container that is reference above.

You can also set the width and height of the div using the keywords width and height. They are measured in pixels.

**Fonts**

Some font properties are:

* Font-family -> refers to what font to use, must specify alternative in case first choice not available in browser
* Font-size -> can be specified in pixels OR small/medium/large/larger
* Font-weight -> Bold or not

**How to reference and group html elements to style them with css**

In css you can specify multiple tags share the same styling like this:

Td, tr {

…

}

The above means both table data and table row share the same styling

1. What if you want to style 2 of the same tags in different ways i.e. you have 2 h1 tags but you want to style them differently? [Using the words tag and element interchangeably]

Then you can each html element a different **id**. Html tags have this attribute where they take an id and that id can be used as a reference for CSS to style them. However, note that ID must be unique in your entire page. This means that if you want more than 1 element to share the same style but not all of the elements with the same tag then you must use a **class** instead. Classes can be used to identify elements but they are not unique. I.E. you can name 2 different tags the same class.

Note:

When you are styling a normal tag, you can just write css like this

Td {

///

}

When styling ID you must add # in front of the name like this

#name {

…

}

When styling classes you must add a . in front of the name like this

.className {

…

}

1. Precedence in html

When we use different css selectors (this means different ways of identifying html elements mentioned above) we run into the case where 1 html element might have 2 or more references and those references might have different styling instructions. I.E. 1 h1 tag with an id of “boo” and a stylesheet where h1 tags should be of the colour blue but ids of the boo should be of the colour red.

The below is how css would resolve this differences

1. Inline
2. Id
3. Class
4. Type

Inline has the most precedent while type is the least

1. Number of css selectors

There are a lot of selectors some of them are

A,b -> Multiple element selector

A b -> Descendant Selector-> This means select all descent elements of A. I.E. ul li [This means all list items that are the descendant of an uordered list]

A > b -> Child selector -> use this when you want to select child elements of A elements. For example ul > li [this means all list items that are the direct child tag of an an unordered list]

A + b -> Adjacent Sibling Selector

[a=b] -> Attribute Selector -> Use this when you want to apply css to only tags with a specific attribute. For example if your webpage has multiple links but you want to highlight the link to facebook with a specific colour you would write the below

        a[href="https://google.com"] {

            color: red

        }

A:b -> Pseudoclass Selector -> Use this selector when you want to add some other feature to the tag. For example if you want a particular css styling on a tag only when there is a cursor hovering over it.

A ::b -> Pseduoelement selector

1. **Responsive Design**

We want our webpage to look good on a variety of devices cause people can be viewing it through smartphones and desktops etc. The below are a few things to take note

1. Viewport

Viewport refers to the screen size of the device the user is on. Important to ensure that your webpage has the setting to automatically fill it according to the size of the device that is currently viewing the webpage.

1. Media Queries

Media queries control how the webpage looks deepening on how it is rendered such as on different screen sizes.

Basically media queries is you asking the device to send over information about itself and to only display certain css styling based on that response

1. Flexbox
2. **Bootstrap**

Link is [www.getbootstrap.com](http://www.getbootstrap.com)

Bootstrap is a css library that you can include in your html page. Copy the css link in their intro and write it in the head of your html page.

1. **SASS**

Sass is an extension for CSS that allow the following functions

1. Introduce basic programming features to css

Gives the ability to create variables, loops and functions to create css easily

1. Nesting

Allows you to nest items in a more maintainable method as in CSS would result in a lot of duplication

1. Inheritance

With the @extend tag you can make certain css selectors inherit css styling from another tag. This is similar to how objects can inherit from one another.

1. Interesting feature

You would need to compile a .scss file to .css file with the sass application. Before you link browser to that css file.