**Class Activity 9 – Web Design & Development – Y2Sem1**

This week we are going to implement the SASS features we discussed in our lecture namely

Variable definition and usage

Nesting

@import

Mixins

**Task A – Installing a preprocessor**

Q1 - Install CSS preprocessor for VSCode (Live Sass Compiler – Glenn Marks)

**Task B – Get some source CSS**

*Rather than generate some SASS from scratch – we could use existing CSS you have from the ITS Sports & Social – and turn it into SASS;*

*Remember SASS uses CSS syntax already!*

Q2 - Copy and paste the CSS you have into a new file called styles.scss – make sure this exists in a /lab9 directory or similar.

**Task C – implement SASS functionality**

Q3 – Implement some variables in your .scss file

Isolate CSS values that are used more than once – example font-size, border/margin pixels margin values – replace these with variables

e.g.

$body-color:#999;

body {

color:$body-color;

}

Q4 – Nesting

Identify opportunities in your code so that you can introduce nesting

Q5 - @import

Cut and paste your media queries from styles.scss to a separate file called mediaqueries.scss.

Use the @import statement at the end of styles.scss

Q6 – @mixins

Are there any opportunities to use @mixins in your file – is there any potential repeatable code in your file? If there is – replace with @mixin

**Task B – Implementing the form**

**Q2 – Implement the form - applying your planned sketch to your site**

You now have your sketch with a layout for your form.

* Apply this layout to a new page soccerclub.html
* Add all of the relevant content components (text boxes, select boxes, check boxes, etc)
* Make sure the form is responsive – does it use the full width on mobile device?
* Conversely on the desktop – is the form an appropriate width? By appropriate – we do not mean the full width of the screen!!!

**Q3 – When form is submitted, write the form values to console**

Using the sample code in formdemo.zip (available in Moodle Week 5) – ensure that the form values are written to the console

**Q4 – Use localstorage object to store the form values to a session so that they are available from another page.**

Using sample code again from formdemo.zip – write the form values to a JSON string save by the localstorage object.

Access the localstorage object in another page – and display the values entered on the form.