**SASS – CSS (STYLESHEETS WORK FROM BOTTOM TO TOP)**

VARIABLES

$primaryBtn : green; // All the buttons will be green

$textColor: blue; // All the text will be blue

This is a variable is SASS

$primaryBtn : green; // All the buttons will be green

$textColor: blue; // All the text will be blue

header {

background: lightblue;

display: flex;

justify-content: center;

color: $textColor;

}

header button {

background: $primaryBtn;

;

}

.contact button {

background: $primaryBtn;

;

}

Graphical user interface

Description automatically generated

**NESTING**

header {

background: lightblue;

display: flex;

justify-content: center;

color: $textColor;

button {

background: $primaryBtn;

}

p {

color: yellowgreen;

}

.mainText {

font-weight: strong;

}

}

This is the code nested – anything to do with header goes in the header {}

button {

background: $primaryBtn;

&:hover {

background: greenyellow;

}

}

Adding a hover/focus etc in the button just put the **&**

Graphical user interface, application

Description automatically generated

Can add different styles sheets and import it into one style sheet

Save the additional stylesheet as \_nameOfSheet.scss

Then import it into the style.scss file

@import "./header.scss";

@import "./header";

Any two will be recognised.

**MIXIN – SIMILAR TO A FUNCTION IN JAVASCRIPT**

Allows you to have more than one instruction – which can be used.

@mixin flexCenter {

display: flex;

justify-content: center;

align-items: center;

}

header {

@include flexCenter();

background: lightblue;

height: 100vh;

color: $textColor;

button {

background: $primaryBtn;

&:hover {

background: greenyellow;

}

}

}

@mixin flexCenter($direction) {

height: 100vh;

display: flex;

justify-content: center;

align-items: center;

flex-direction: $direction;

}

header {

@include flexCenter(column);

background: lightblue;

height: 100vh;

color: $textColor;

button {

background: $primaryBtn;

&:hover {

background: greenyellow;

}

}

If I want the direction to change I can add a variable $direction in the mixin

Then when I include I add the direction for variable

@include flexCenter(column);

@include flexCenter(column);

@mixin flexCenter($direction, $background) {

height: 100vh;

display: flex;

justify-content: center;

align-items: center;

flex-direction: $direction;

background: $background;

}

header {

@include flexCenter(column);

Can also do more than one variable

.contact {

@extend header;

}

This extend element basically inherits all the styles in header and puts it in the contact.

.contact {

@extend header;

background: lightgrey;

}

When you add things after it overrides it but will still keep the same styles.