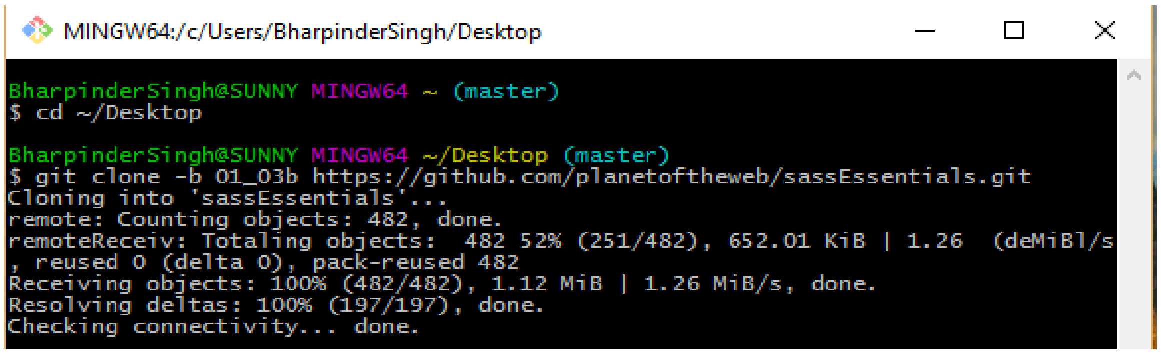
Learning SASS

The first step was to install SASS and other additional software command such as Ruby, Gitbash, etc. Once this was done, I had a look at SASS essentials tutorials on Lynda.com and accessed the SASS essentials file the tutorial provided so that I can learn along the tutorial. To do this, I firstly opened up Gitbash and accessed my Desktop by writing the cd ~/Desktop command as illustrated below. Once this was done, it was important that the directory I was going to access had a gulp file as this was required to learn SASS.
So the next step was to access the drive from github.com and to do this as illustrated in the tutorial a clone command had to be used to access the right tutorial that could be used with the web address link. As you can see below that I have used the git clone –b 01_03b https://github.com/planetoftheweb/sassEssentials.git to access this SASS Essentials tutorial. To ensure that I had successfully accessed the drive, it shows connectivity as done to ensure the directory accessed is the correct one and it create a sassEssentials folder on Desktop.
Once I had accessed the directory from github.com and a folder was created, I used the cd sassEssentials command to get into the directory and access the full folder.



```
BharpinderSingh@SUNNY MINGW64 ~ (master)
$ cd ~/Desktop

BharpinderSingh@SUNNY MINGW64 ~/Desktop (master)
$ git clone -b 01_03b https://github.com/planetoftheweb/sassEssentials.git
Cloning into 'sassEssentials'...
remote: Counting objects: 482, done.
remoteReceiv: Totaling objects:  482 52% (251/482), 652.01 KiB | 1.26  (deMiBl/s
, reused 0 (delta 0), pack-reused 482
Receiving objects: 100% (482/482), 1.12 MiB | 1.26 MiB/s, done.
Resolving deltas: 100% (197/197), done.
Checking connectivity... done.
```

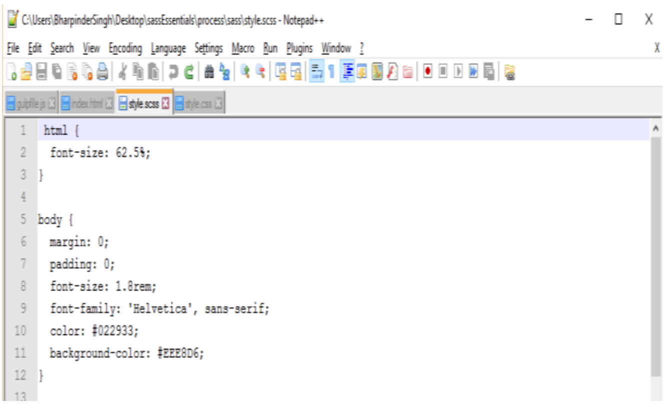
Once this was done it was now time to use the gulp command however you need to ensure this is installed by typing in npm install. The gulp command opens up the index.html folder. Since I had to demonstrate the learning on my site, I replaced the directories index.html with my index.html file.

Working with variables

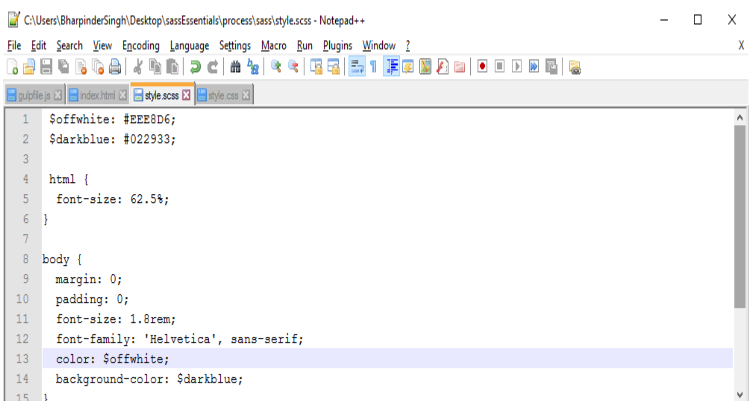
The next step was working with variable to change some of the content on my old portfolio. So to ensure that sass variables were working, I started to use some of the variables such and checking it on my style.css as this is where the changes would show up. So anything that I was changing on my style.scss was automatically updated on style.css. As you can see below the before and after screenshot of my index.html page it shows the difference.

No Variable used to show the changes

Variable used to show changes below.

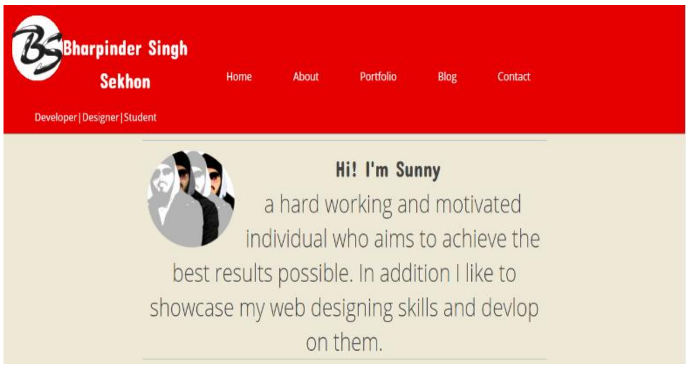

C:\Users\BharpinderSingh\Desktop\sassEssentials\process\sass\style.scss - Notepad++
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
gulpfile.js
index.html
style.scss
style.css

```
html {
  font-size: 62.5%;
}

body {
  margin: 0;
  padding: 0;
  font-size: 1.8rem;
  font-family: 'Helvetica', sans-serif;
  color: #022933;
  background-color: #EEE8D6;
}
```


C:\Users\BharpinderSingh\Desktop\sassEssentials\process\sass\style.scss - Notepad++
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
gulpfile.js
index.html
style.scss
style.css

```
$offwhite: #EEE8D6;
$darkblue: #022933;

html {
  font-size: 62.5%;
}

body {
  margin: 0;
  padding: 0;
  font-size: 1.8rem;
  font-family: 'Helvetica', sans-serif;
  color: $offwhite;
  background-color: $darkblue;
}
```

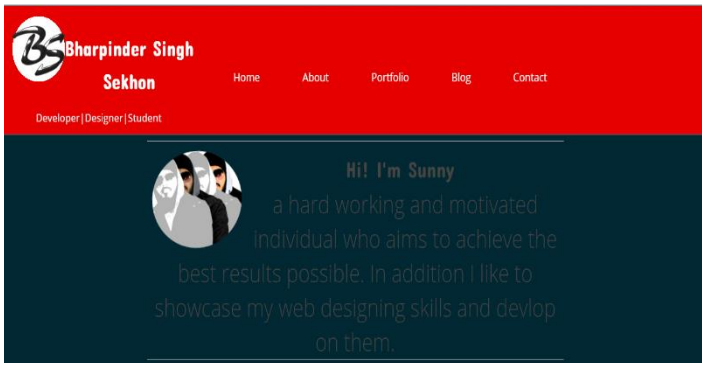
Before variables were used.



Hi! I'm Sunny

a hard working and motivated individual who aims to achieve the best results possible. In addition I like to showcase my web designing skills and devlop on them.

After variable were used.

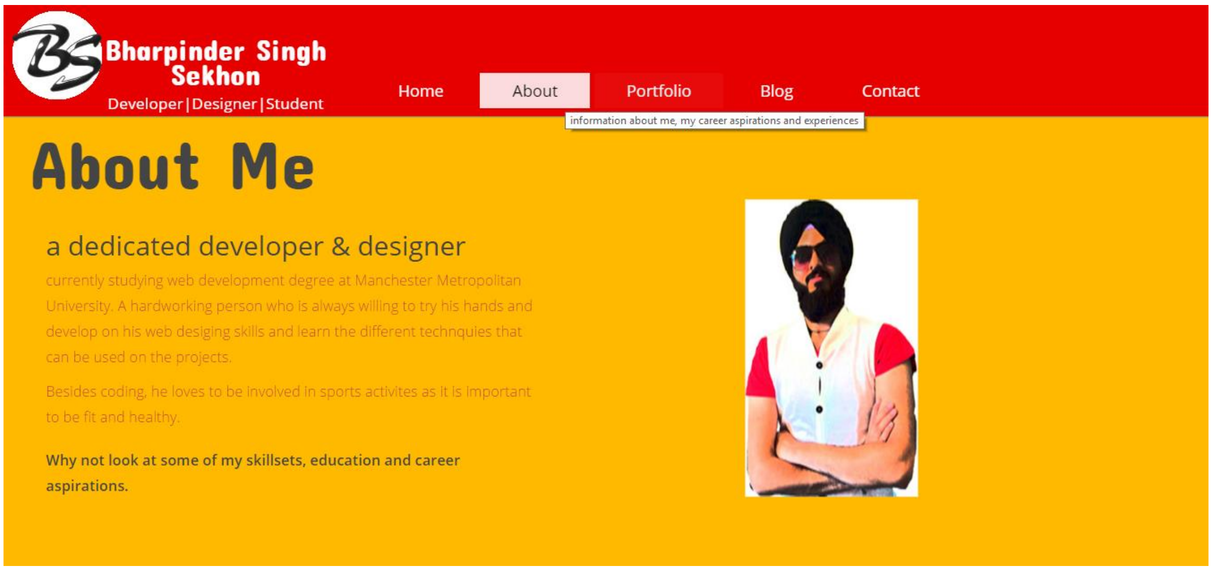


Hi! I'm Sunny

a hard working and motivated individual who aims to achieve the best results possible. In addition I like to showcase my web designing skills and devlop on them.

Another way of just changing colours is by using the different sections if I had a lot of coding on my documents, different sections such as adding $color-main variable would make it easy for me to change just the background color if required. Sections would also help me to manage my code more efficiently rather than having to put up different color variable, I could just change this on the main sections and it would update the entire stylesheet.

With different variables such as for font I just had to include a main variable and the fonts that I want to use. After this I had to just include the main variable next to my font-family code rather than typing up all fonts that I required. So by just adding a few variables and matching it up to ensure the code is working I have successfully learnt how to use variables as shown below that using different variables changed a lot of font and colors on my about me page.


Bharpinder Singh
Sekhon
Developer|Designer|Student
Home
About
Portfolio
Blog
Contact
information about me, my career aspirations and experiences

About Me

a dedicated developer & designer

currently studying web development degree at Manchester Metropolitan University. A hardworking person who is always willing to try his hands and develop on his web desiging skills and learn the different technquies that can be used on the projects.

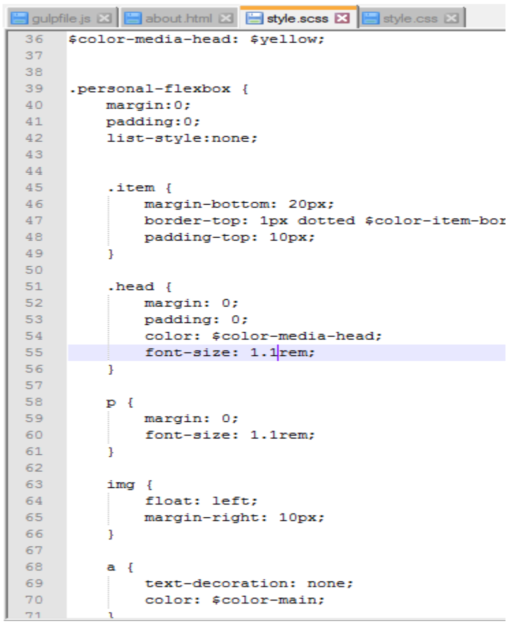
Besides coding, he loves to be involved in sports activites as it is important to be fit and healthy.

Why not look at some of my skillsets, education and career aspirations.

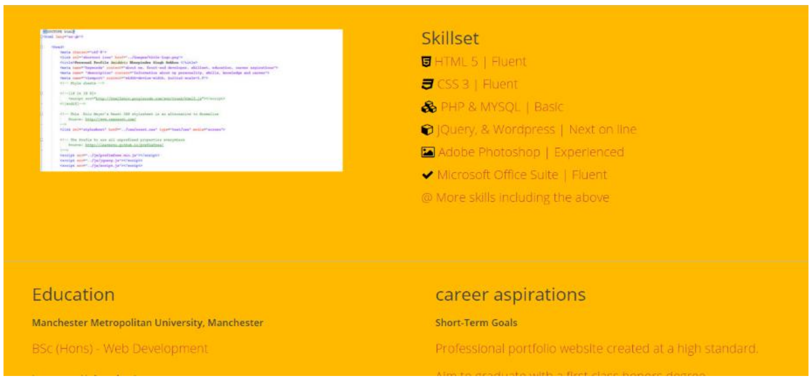
Nesting Styles

For this section I thought of using the about me page to use nesting styles. The first thing was to find a relevant section on the about me page so I used the personal-flexbox as my class and cleared all the margin and padding by changing these to 0 on my style.scss stylesheet. Nesting style was just having rules inside other rules which would become subrules, I included .item in my personal-flexbox rule.

As you can see that I have added different rules which show the changes that have been done my about me page. Nesting styles just lets you add new sections on style.scss and once you have chosen the content that needs to be changed, you just have to find your class from the stylesheet.



```
$color-media-head: $yellow;


.personal-flexbox {
    margin:0;
    padding:0;
    list-style:none;


    .item {
        margin-bottom: 20px;
        border-top: 1px dotted $color-item-bor
        padding-top: 10px;
    }

    .head {
        margin: 0;
        padding: 0;
        color: $color-media-head;
        font-size: 1.1rem;
    }

    p {
        margin: 0;
        font-size: 1.1rem;
    }

    img {
        float: left;
        margin-right: 10px;
    }

    a {
        text-decoration: none;
        color: $color-main;
    }
```



Skillset
HTML 5 | Fluent
CSS 3 | Fluent
PHP & MYSQL | Basic
JQuery, & Wordpress | Next on line
Adobe Photoshop | Experienced
✓ Microsoft Office Suite | Fluent
@ More skills including the above

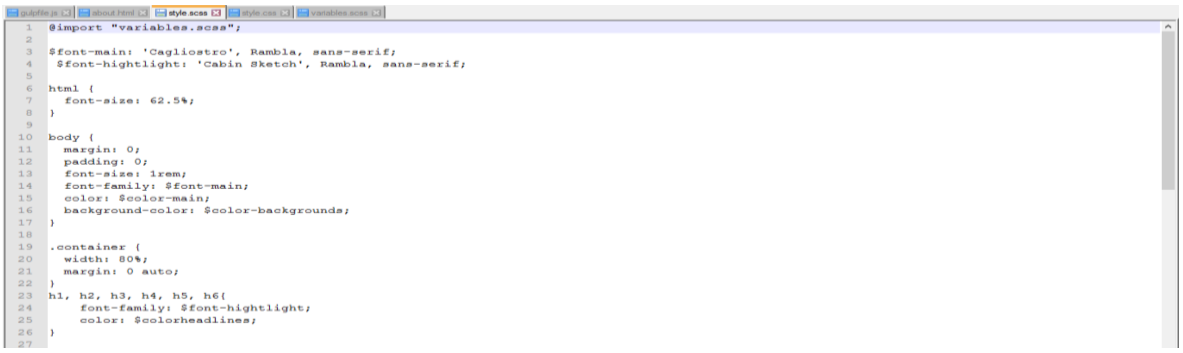
Education
Manchester Metropolitan University, Manchester
BSc (Hons) - Web Development

career aspirations
Short-Term Goals
Professional portfolio website created at a high standard.
Aim to graduate with a first class honors degree.

Using Partials

The next process in my learning process was partials which helped me to organize my code into logical groups. For this all I had to do was add @import command to my stylesheet and the only difference was that the CSS got combined into a single file. The first thing that I did was create a new file called variables.scss and imported all the variables from the style.scss file into the variables file. Once I copied the variables, all I did on my style.scss file was add a @import “variables.scss”; command to ensure that the partial was working fine on my site. As you can see below that after adding a partial in my main scss file, the site was displaying the same contents with colours as it was before creating a new variable.scss file.

Removing variables from style.scss file and adding it on variable.scss, then adding the @import command.



```
@import "variables.scss";

$font-main: 'Cagliostro', Rambla, sans-serif;
 $font-hightlight: 'Cabin Sketch', Rambla, sans-serif;

html {
  font-size: 62.5%;
}

body {
  margin: 0;
  padding: 0;
  font-size: 1rem;
  font-family: $font-main;
  color: $color-main;
  background-color: $color-backgrounds;
}

.container {
  width: 80%;
  margin: 0 auto;
}
h1, h2, h3, h4, h5, h6{
    font-family: $font-hightlight;
    color: $colorheadlines;
}
```

After adding the variables in the file variables.scss, I added the html, body and container coding in a file called base.scss. The results that I was getting were successful as there were no errors and the files were more organised in terms of having the codes in one whole css file.

Conclusion

Since I am still learning SASS and all the elements that I have learnt so far make good sense to me as I have been able to work along the tutorial provided by Lynda.com about SASS. I now believe that using other elements will help me to increase my knowledge about sass even further.