Day 1

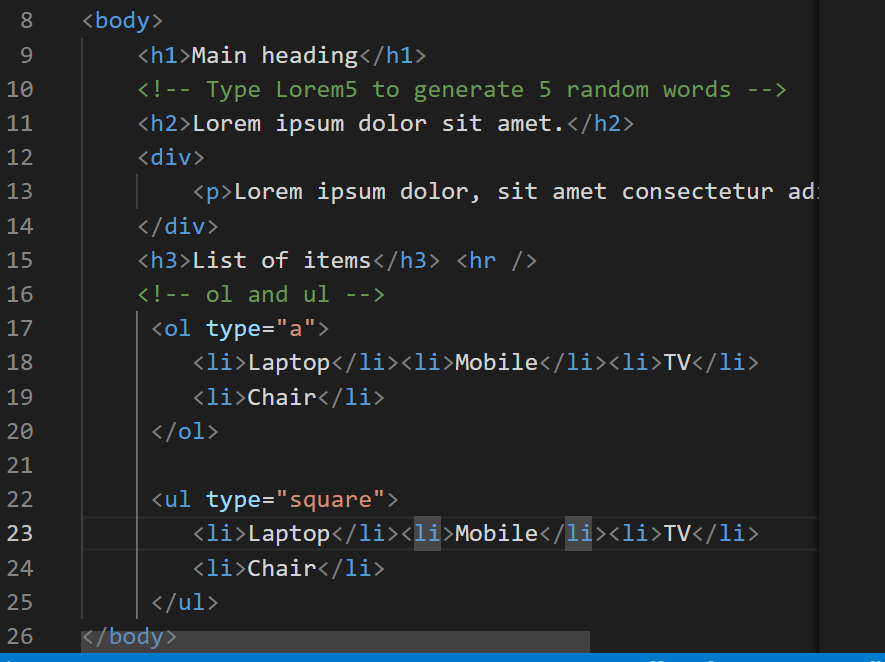
HTML, CSS & SASS/SCSS

HTML: Hyper Text Markup Language - Display the content

CSS - Cascading Style Sheet - Style

HTML file

Lists & headings



Output:



You can create HTML tables, forms and other elements

CSS:

It is used mainly to add the styles to HTML elements

RWD

It helps to fit your web page to all types of devices it could be desktop, mobiles, printing devices, tablets

Same HTML content to render differently in different devices

Medi Query: it applies the styles based on the conditions.

Grid: It arranges the elements in row & column format

Flex: It automatically arranges the elements based on the size like horizontally or vertically.

{display: flex }

SASS

Syntactically Awesome Style sheets

It avoids lot of boiler plate code which you write in CSS

SASS -> SCSS (this resembles css syntax), but sass follows indentation syntax

SASS syntax  
$font-style: arial;  
$color: blue

h1   
 color: $color;  
 font-family: $font-style

SCSS syntax:

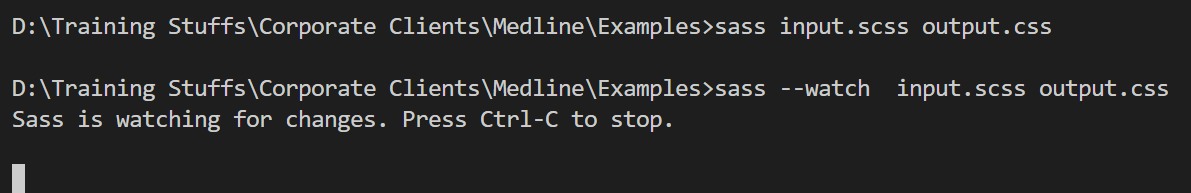
$color: blue;  
$font-style: arial;  
h1 {   
 color: $color;  
 font-family: $font-style;  
}

Install sass

npm install -g sass

To compile scss to css

sass input.scss output.css



--watch lets you to automatically compile scss to css

Mixins

These allow you to reuse the group of CSS declarations.

@mixin shape($color, $width, $height) {   
 width: $width;  
 height: $height;  
 background-color: $color;  
}   
.square {  
 @include shape(blue, 400px, 400px);   
}  
.rectangle {  
 @include shape(red, 400px, 200px);  
}

Parent selector

It is used in nested selector to refer the outer selector

.alert {  
 &:hover { color: red }  
}

In CSS you write

alert : hover { color: red; }

Placeholder

It is to reuse some selectors using inheritance

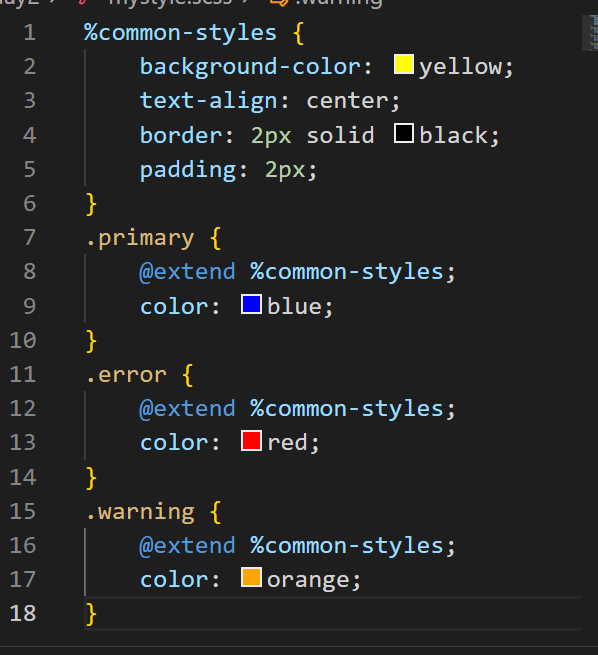
%name {  
 // css selectors  
}

.primary {  
 @extend %name;  
 color: blue;  
}  
.secondary {  
 @extend %name;  
 color: grey;  
}

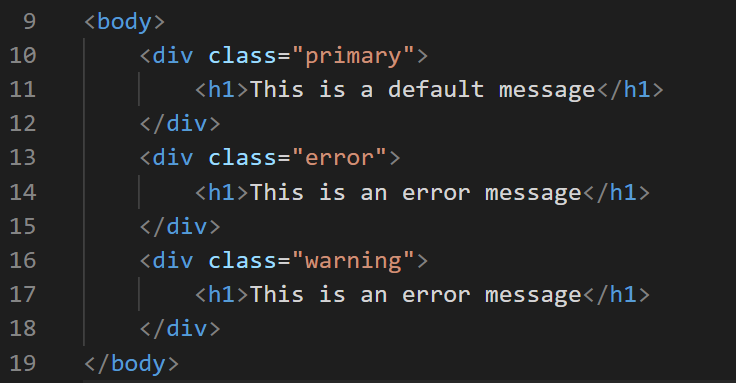
How to compile multiple SASS files

sass --watch file1.scss : file1.css file2.scss : file2.css

day2/mystyle.scss



index.html



Output:



Special functions:

SCSS provides various functions to perform complex operations

$colors: red, green, blue;  
.error { color : nth($colors, 1) }

.warning { color: nth($colors, 2) }

Mixing colors using mix()

.mixed-color { color: mix(red, green, 60%) }

Here 60% red & 40% green

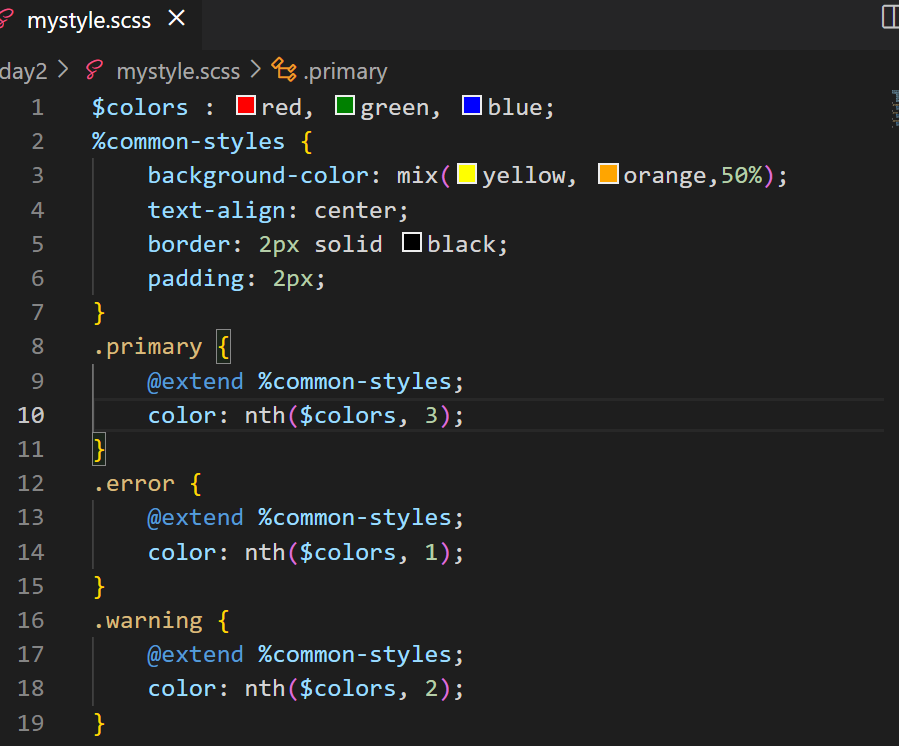
Note: You can only mix two colors, however you can get the reference of two colors and pass the color again to the mix

mix(mix(red, green, 60%), blue, 50%)

Round up the numbers

font-size: round(19.6px); // font-size: 20px

scss



Output:

