1. Sass : who are you ?
   1. Là 1 css prepercessor giúp:
      1. Viết CSS nhanh và chuyên nghiệp
      2. Tiết kiệm thời gian và đỡ nhàm chán
      3. Dễ bảo trì và phát triển
      4. Sử dụng lại kết hợp với CSS
      5. Phù hợp với dự án lớn
      6. Sass, less
   2. Nested : là việc phân cấp cha con
   3. Nested properti : khi các thuộc tính có định dạng giống nhau thì dùng nó
      1. Font : {famili:aria}
   4. Referensing parent selector : & : là định dạng cùng 1 thẻ a … rồi thẻ a khi hover thì là &:hover hay nói cách khác dấu & đại diện cho thằng cha của nó
   5. Variable & Data Type:
      1. khai báo và sử dụng : $width : 5em
      2. global : khi khai báo biến ở đâu thì biến có có khả năng dùng ở khu vực đó
   6. DataType và operator :
      1. Khi định nghĩa 1 biến có thể là các kiểu dữ liệu sau :
         1. Number (5px; 10em; 13, 1.2)
         2. String (“../image/”)
         3. Boolean(true, false)
         4. Null
         5. List : dánh sách các function sử dụng cho list http://sass-lang.com/documentation/Sass/Script/Functions.html
         6. Map
      2. Interporation : #{$name} : dùng để nối tên thôi còn muốn tính toán thì dùng $i
      3. Set Font : vào font.google.com để lấy font . Lưu ý xem font của giao diện chính là gì và đặt biến với font
   7. Control directive và Expression
      1. If :
         1. @if
         2. @if else @if else
      2. For :
         1. $for
      3. While
      4. Each
   8. Function :
      1. Tự định nghĩa
      2. Gg sass function
   9. @-Rules and Directives:
      1. Import : trong 1 dự án khi viết nhiều code vì thế k thể đặt trong cùng 1 tập tin scss : dùng import để tách các tập tin
      2. Tập tin k muốn dịch sang css dùng \_ten
      3. Media : là việc sử dựng nó dưới các loại màn hình khác nhau
      4. Extend :
         * 1. giúp kế thừa các class được định sẵn
           2. Nếu k muốn nó chuyển sang định dạng css khi kế thừa thì %class
   10. Mixin :
       1. Các sử dụng mixin dùng : include
       2. Mixin mạnh hơn extend : và chức năng gần giốn extend
       3. Đối với dạng list hoặc dạng map thì dùng dấu … để hiển thị thay cho 3 tham số truyền vào

Bài xây dựng giao diện button : cần chú ý style-03 kiểu map lồng trong map

Bài dây dựng giao diện ảnh :

* chú ý các phân chia tập tin chính thành các tập tin nhỏ và include nó vào với nhau
* Nth-child() : định dạng các thẻ con

Bài tập grid : có thêm thuộc tính box-sizing : boder-box : thì kích thước chiều rộng gồm content , padding và border chứ k gồm margin

1. GruntJS : là 1 javaScript tab runner
   1. Muốn sử dụng được nó phải cài đặt nodeJS và npm install -g grunt-cli
   2. Các tạo ra 1 Grunjs :
      1. gồm package.json : chứa phần cấu hình project và các thông tin pubin
         1. lấy theo mẫu ở trong trang gruntjs
            1. phải tuân thủ theo cú pháp của 1 chuỗi json
            2. name : tên của project
            3. sau đó cài đặt nó
         2. các bước thực hiện :
            1. cấu hình project : để cho nó nằm ở tập tin paket.json
            2. load các plugin sử dụng
            3. đăng kí những task cần thực hiện
         3. Chạy file mà thấy syntaxError : thì liên quan tới rất nhiều lỗi cú pháp
         4. Câu lệnh chạy dự án grunt : npm install grunt --save-dev
      2. Gruntjs
         1. Css min : nhóm phát triển by Grunt Team : https://www.npmjs.com/package/grunt-contrib-cssmin
            1. npm install grunt-contrib-cssmin --save-dev
            2. nhớ vào load plugin : grunt-contrib-cssmin
            3. nhớ khai báo nó vào : grunt.registerTask('abc', ['cssmin']);
            4. thực hiện việc sử dụng nó bằng các coppy phần sử dụng vào phần 01 initConfig
            5. trong phần file của tập tin sử dụng cần khai báo đường dẫn :chú ý tập tin bên ngoài bao gồm các tập tin bên trong
            6. muốn minify nhiều tập tin css cùng 1 lúc thì phần bên dưới có nói

src : tập tin gốc \*.css : lấy tất cả các tập tin css

!\*.min.css ngoại trừ tâp tin .min.css

Cwd : phần khai báo phần chung

Dest : đường dẫn chứa tập tin output

Ext : phần chứa tên tập tin output

* + - 1. Css concat : <https://www.npmjs.com/package/grunt-contrib-concat> : nối 2 tập tin lại với nhau thành 1 tập tin duy nhất
         1. Src : tập tin nguồn
         2. Dest : tập tin output
      2. grunt-contrib-uglify : <https://www.npmjs.com/package/grunt-contrib-uglify> : mini file cac tap tin js
         1. files : output rồi mới input
      3. grunt sass : https://www.npmjs.com/package/grunt-sass
         1. phần file : file output – file input
      4. watch : <https://www.npmjs.com/package/grunt-contrib-watch>
         1. files : những tập tin nào bạn muốn quan sát hoặc theo dõi nó
         2. tasks : khai báo công việc
      5. plugin connect: <https://www.npmjs.com/package/grunt-contrib-connect>
         1. thực hiện kết nối tới 1 web server
         2. port : 3069, thực hiện kết nối tới 1 cổng
         3. hostname : ‘localhost’
         4. base : đường dẫn tới tập tin cần kết nối tới service
         5. task này phải luôn chạy cùng task watch
      6. plugin include : <https://www.npmjs.com/package/grunt-includes>
         1. Để kéo các tập tin lại với nhau
         2. Src [input]: đường dẫn đến tập tin index.html
         3. Desk [output] : đường dẫn ra
         4. Cwd : ‘.’ : thư mục hiện hành
         5. Chú ý các phần thay đổi ở các plugin nào cần quan sát thì phải thêm vào task của phần watch
      7. Các bước thực hiện
  + Tạo 2 tập tin : package.json và Gruntfile.js
  + npm install -g grunt-cli
  + coppy các thư mục
  + chạy lần lượt các câu lệnh
    - npm install grunt-contrib-cssmin --save-dev
    - npm install grunt-contrib-uglify --save-dev
    - npm install --save-dev grunt-sass
    - npm install grunt-contrib-watch --save-dev
    - npm install grunt-contrib-connect --save-dev
    - npm install grunt-includes --save-dev
    - download : font awesome lấy 2 cột css và font nhúng vào tập tin bootstrap đã được giản nén ra
    - Vị trí đặt ở phần k chỉnh sửa gì đến là phần production
    - <link href="css/font-awesome/css/font-awesome.min.css" rel="stylesheet" >
      1. Những điều cần chú ý trước khi thiết kế
         1. mở file pts lên cần chú ý file đó đã thiết kế theo chuẩn bootstrap chưa (12 cột)
         2. tính khoảng cách giữa các cột là bao nhiêu (30px)

Trong trường hợp k tuân theo chuẩn bootstrap thì phải vào phần costummize thay đổi lại số cột vs khoảng cách các cột rồi tải về

* + - * 1. Khi kéo các tập tin vào nhau trong scss thì dùng import : “đường dẫn”;
        2. Khi kéo các tập tin HTML lại với nhau sử dụng plugin thì dùng include: “đường dẫn”;
        3. Khi làm việc trên môi trường developer thì cứ chia file càng nhỏ càng dễ quản lý còn khi public sản phẩm thì phải là 1 file mà thôi
      1. Bắt tay vào thiết kế theo quy tắc từ trên xuống dưới rồi từ trái qua phải : blog\_listing.psd : đầu tiên html => CSS => reponesive
         1. Chia nhỏ foder

Devices : foder chứa các tập tin với các thiết bị khác nhau

Mixins : chứa các định nghĩa mixin

Footer : định nghĩa phần footer

Header : định nghĩa phần header

Widgets : chứa các phần định dạng cho các khối

Elements : Chứa các thành phần bài viết trong 1 trang

* + - * 1. Thực hiện thêm các tập tin

\_reset.scss : dùng để reset các định dạng

\_font.scss : dùng để khai báo các font chữ

\_variable.scss : dùng để định nghĩa các biến

* + - * 1. Thực hiện import các tập tin theo thứ tự

Font trước

Variable

Mixin

Reset

Định nghĩa font : gg font rồi tìm kiếm font đó : nhớ import và định nghĩa

Chú ý cần định nghĩa cho font dành co tiêu đề và font dành cho nội dung

* + - * 1. Xây dựng header

Thấy header có mầu tràn hết màn hình , bên trong sẽ để container

Header-logo :Sử dụng class pull-left trong bootstrap để làm nó sang bên trái

Phần menu : có class pull-right để làm nó sang bên phải

Ul > li > a

Vì định dạng thẻ a nhiều lần giống nhau lên viết 1 mixin cho nó => tập tin util.scss là mixin tự viết còn libs.scss là chứa cái downlload về

Phần đánh dấu chính là thuộc tính box-shadow

Li con thì thằng ngoài để position : ralative

Thằng con để position : absolute rồi định nghĩa top , left cho nó là ok

Chú ý : dấu mũi tên là chỉ ảnh hưởng trực tiếp đến thằng con của nó thôi , và thêm thuộc tính z-index để nó nổi lên

* + - * 1. Reponsive –phần header

Để lấy những mốc với các thiết bị khác nhau thì vào luôn bootstrap mà lấy

Thiết lập ở phần divice

Nhớ tập tin main chạy chính phải thêm phần divice vào đấy

ở màn hình medium-size : giảm padding của thẻ a là 10px và 18px;

k viết trực tiếp nó ở file từng thiết bị mà phải xây dựng các mixin để tiện cho việc nâng cấp 1 biến có thể chứa nhiều giá trị

ở size small thì thẻ a giảm padding : 8 14, header : text-aline : center , và bỏ class full-left và pull-right : bằng cách float-left : none và thêm importan

ở màn hình bé nhất : những thiết bị bé hơn 425px thì toàn bộ phần nav bị ẩn đi:

logo nằm ở vị trí chính giữa (sử dụng mixin include setCenterheader-menu)

ẩn header nav . menu và hiện thẻ a menu ra nhớ importan đấy

hiển thị các thành phần ở bên trong thẻ menu ra

* + - * 1. Xây dựng giao diện breadcump :là phần breadcump mà bootstrap đã xây dựng sẵn

Dùng section xây dựng thành từng vùng vẫn container như bình thường

Nhớ reset các phần breadcump và phần các thẻ h1 h2 h3 h4 ….

Reset cả margin và padding nữa

Reponsive thì với loại mobie thì viết 1 mixin breadcump giảm font chữ cho nó nhớ kiểm tra kĩ lại xem nếu nó k chịu thì phải impotant cho nó

* + - * 1. Xây dựng giao diện acticle:

Nhận thấy có hình ảnh , tiêu đề , phía dưới bài viết được viết bởi ai , comment , phần miêu tả và phần cuối là nút

Tiến hành phân chia từng phần : 1 phần chiếm 9 phần 1 phần chiếm 3 phần

Phần hình ảnh message và view được lấy ở bootstrap

Mỗi dòng cách nhau 1 khoảng cách vì thế đặt biến $distan : 20px mỗi thành phần cách nhau 1 khoảng cách 20px và phần tử cuối cùng k cách

Tránh trường hợp lặp đi lặp lại thì những tl còn của div article đều có phần cách nhau : > \*

Khi hover vào hình ảnh: zoom: 1;

filter: alpha(opacity=80);

opacity: 0.8;

Cấu trúc thẻ a mầu a , rồi mầu khi hover vào xảy ra rất nhiều lần thì viết cho nó 1 mixin setTagAHover();

Để nó thẳng hàng với các phần trên thì nhớ kiểm tra padding của nó

Reponsive : viết 1 mixin thiết lập marginBottom , MargiRight , distanceLine=> rồi gọi mixin trong loại màn hình tương ứng

* + - * 1. Xây dựng wedgit-tag :

Kéo vào vùng right ở index.html

Widget : là những phần định dạng chung

Widget-tag : là phần định dạng khác nhau

* + - * 1. Xây dựng giao diện footer-top

Phần company: định nghĩa dấu mũi tên chính là hình ảnh sử dụng thuộc tính &:before để thêm phía trươc thẻ a

* + - * 1. Xây dựng phần blog : sử dụng bootstrap có class media
        2. Reponsive phần footer-top :

Tạo 1 class footer-column ở tập tin footer.html

Viết 1 mixin setFooterTop() : thiết lập những tập tin có class là .footer-column : có margin-bottom : là … rồi truyền mixin này vào phần định dạng với loại màn hình tương ứng

* + - 1. Public
         1. Đưa foder : production cho phía khách hàng
         2. Kéo tập tin js vào thì nhớ chạy file ppublic trước
         3. Chỉ mini-file html sau khi bên ráp code thực hiện ok
      2. Fix cái js k thành công thì cần chuyển task
      3. Validator : kiểm tra xem thỏa chuẩn của w3 chưa : gg w3c validator để thực hiện quá trình validate
      4. Tăng tốc web :
         1. Page speed google : nhập vào url
         2. Tải trực tiếp font đó về thông qua công cụ online : từ khóa google font generator fontface : coppy toàn bộ font rồi vất vào tập tin font ở sass
         3. Nhớ download font này về vất trong tập tin font của production