**https://www.niemvuilaptrinh.com/article/resume-va-portfolio?fbclid=IwAR24MUc42NLfZ65Tu1iDcmZ1NjrDlZhVQES5HvRkDK7rtgpw6Ng529EI-ig**

**7 steps design a website**

**1. Define your project**

* Start off by defining the goal of your project. This can be showing your portfolio to the world, selling an e-book, building a blog, etc.
* Also define your audience. Ask yourself: which is the typical user that will visit my website?
* This is important, because you should always design with your goal and audience in mind.

**2. Plan out everything**

* Once your project is defined, plan your content carefully. This includes text, images, videos, icons, etc.
* Remember what I told you about visual hierarchy. It plays an important role when you start thinking about what you want on your website and what you don't. Defining the content before actually starting the design is called the *content-first approach*. It means that you should design for the content, instead of designing a webpage and then filling it with some stuff.
* Define the navigation.
* Define the site structure. You can draw a sitemap in this step if we're talking about a bigger project.

**3. Sketch your ideas before you design**

* Now it's time to get inspired and think about your design.
* Then, get the ideas out of your head. And with that I mean that you should sketch your ideas before you start designing. It will help you explore ideas and create a concept of what you want to build. Using pencil and paper is a great way of quickly retaining your valuable ideas.
* Make as many sketches as you want, but don't spend too much time perfecting anything. Once you have an initial idea, you can concentrate on the details when designing in HTML and CSS.
* I advise you to never, ever start designing without having an idea of what you want to build. Getting inspiration is very important in this phase, and I already told you how to do that!

**4. Design and develop your website**

* After sketching, start to design your website using all the guidelines and tips you've learned in the web design section.
* You'll do that using HTML and CSS, which is called *designing in the browser*. Designing in the browser is basically designing and developing at the same time.
* There are more and more designers leaving traditional design programs such as Photoshop and start designing in the browser. The biggest reason for this is that you can't design responsive websites in photoshop. It also saves you tons of time.
* In this phase, you'll use your sketches, content and planning decisions you've made in steps 1, 2 and 3.

**5. It's not done yet: optimization**

* Before you can actually launch your beautiful masterpiece for the world to see it, we have to optimize its performance in terms of site speed.
* You also need to do some basic search engine optimization (SEO) for search engines such as google.

**6. Launch the masterpiece**

* Your optimized website is now finally ready to launch.
* All you need for launching is a webserver that will host your website and deliver it to the world.

**7. Site maintenance**

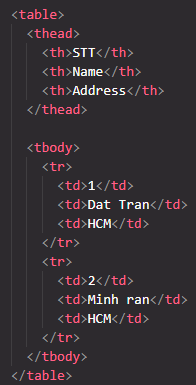
* The launch of your website is not the end of the story.
* Now it's time to monitor your users' behavior and make some changes to your website if necessary.
* You should also update your content regularly in order to show your users that your website is alive! For instance, a blog can be a great way of doing that.

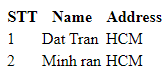
**Color Theory:**

* Red is a great color to use when power, passion, strength and excitement want to be transmitted. Brighter tones are more energetic and darker shades are more powerful and elegant.
* Orange draws attention without being as overpowering as red. It means cheerfulness and creativity. Orange can be associated with friendliness, confidence, and courage.
* Yellow is energetic and gives the feeling of happiness and liveliness. Also, it associates with curiosity, intelligence, brightness, etc.
* Green is the color of harmony, nature, life and health. Also, it is often associated with money. In design, green can have a balancing and harmonizing effect.
* Blue means patience, peace, trustworthiness, and stability. It is one of the most beloved colors, especially by men. It is associated with professionalism, trust and honor. That's actually why the biggest social networks use blue.
* Purple is traditionally associated with power, nobility and wealth. In your design, purple can give a sense of wisdom, royalty, nobility, luxury, and mystery.
* Pink expresses romance, passivity, care, peace, affection, etc.
* Brown is the color of relaxation and confidence. Brown means earthiness, nature, durability, comfort, and reliability.

phím tắt để select copy qua CSS  
  
Ctrl D : select những điểm giống  
Ctrl + Shift + L: select hết những điểm giống  
Shift + hướng mũi tên để select  
Ctrl + Alt: multiple cursor

Table:





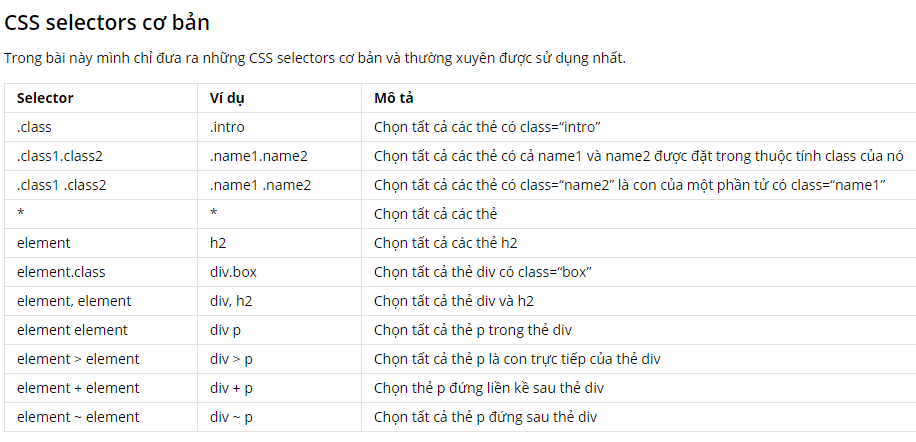
Có 3 cách để style HTML:

External (tạo file .css bên ngoài)

Internal (sử dụng thẻ style)

Inline (sử dụng attribute style)

ID thì xài #, class thì xài .



Độ ưu tiên: Inline 1000 => #id 100 => .class 10 => tag 1 => Equal specificity => Universal selector and inherit

!important là quan trọng nhất

Only use !important as a last resource => more maintainable

Inline style ưu tiên hơn style ở css

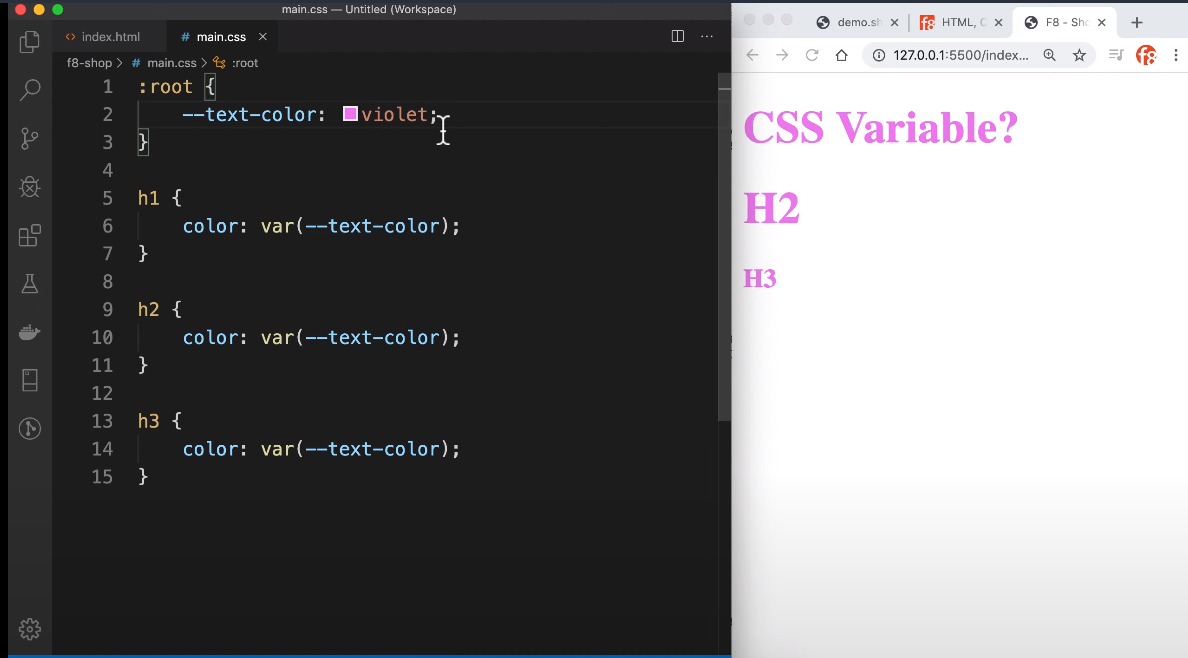
ID quan trọng hơn

Reply more on **specificity** than on **order** of other selectors

Reply on others when using 3rd party stylesheet, always put **your author stylesheet** **last in HTML**

Parent’s width là reference for percentage based calculation

Đặt biến trong CSS (dùng root), ví dụ đây là đặt biến cho 1 màu, sau này tiện thay đổi, chỉ cần đổi ở biến



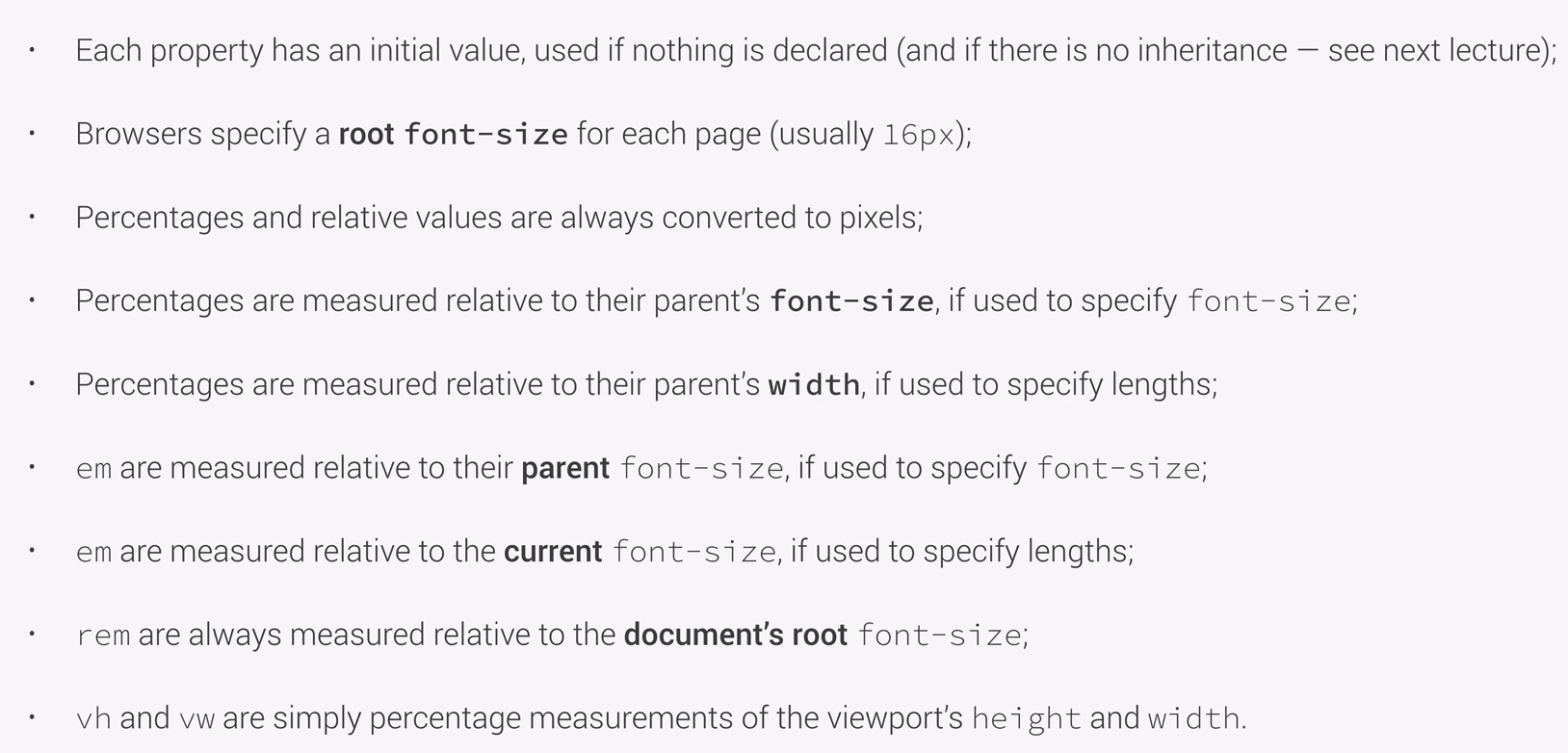
Absolute unit: đơn vị tuyệt đối

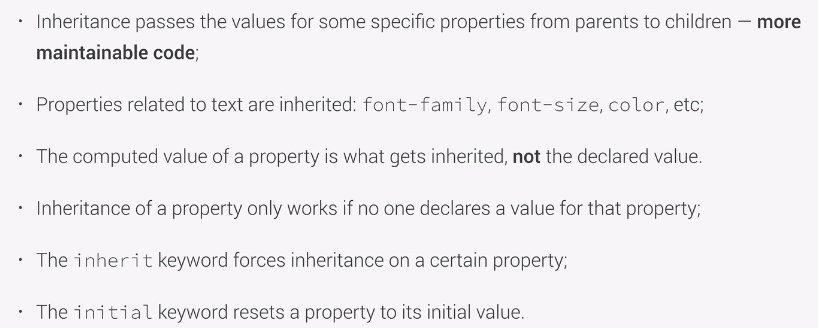
* + Kích thước cố định, ko đổi khi có yếu tố khác tác động

Relative unit: đơn vị tương đối

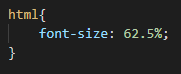
* + Giá trị phải có 1 nơi phụ thuộc, bị thay đổi kích thước
  + Rem phụ thuộc vào size trong thẻ html (dễ kiểm soát hơn em)
  + Em thì phụ thuộc vào thẻ gần nhất chứa nó mà có thuộc tính font-size
  + Html font-size : 100% = 16px
  + Vw (view width): chiều ngang trình duyệt, vh (view height): chiều rộng trình duyệt. Ex: vw=100 => chiếm hết chiều rộng trình duyệt

Thường sử dụng % để chia layout



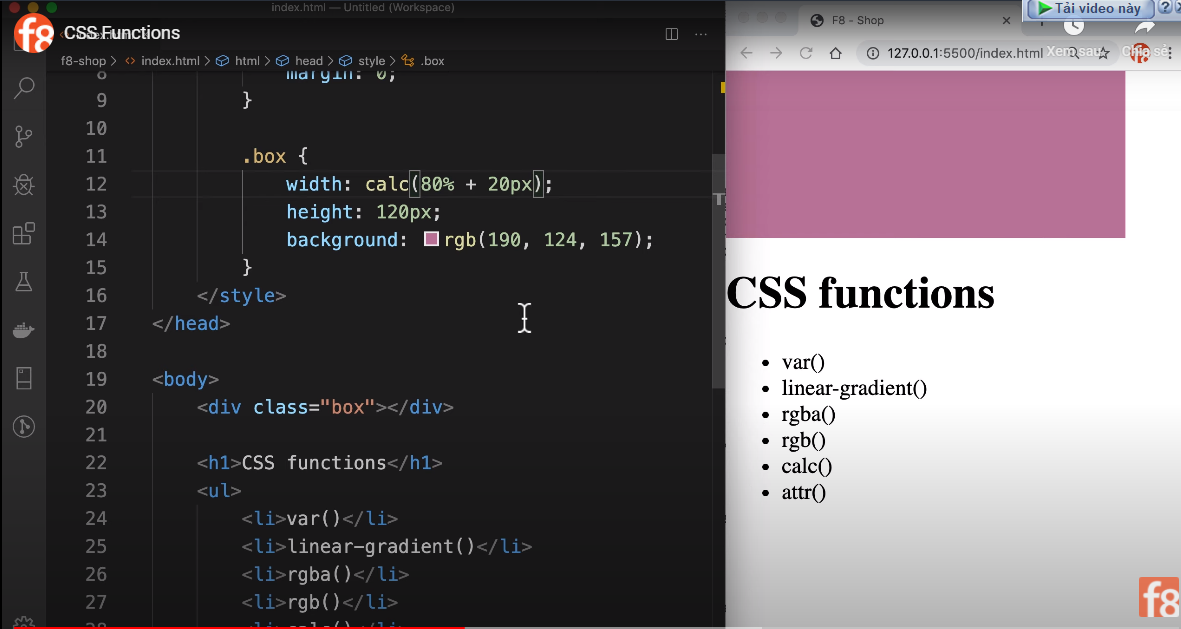


Default font-size để sử dụng rem cho dễ (1 rem = 10px) ( 16px là 100% nên 62.5% là 10px)



Size tầm 42-48px ngón tay mới bấm dc

Calc() dùng để tính toán

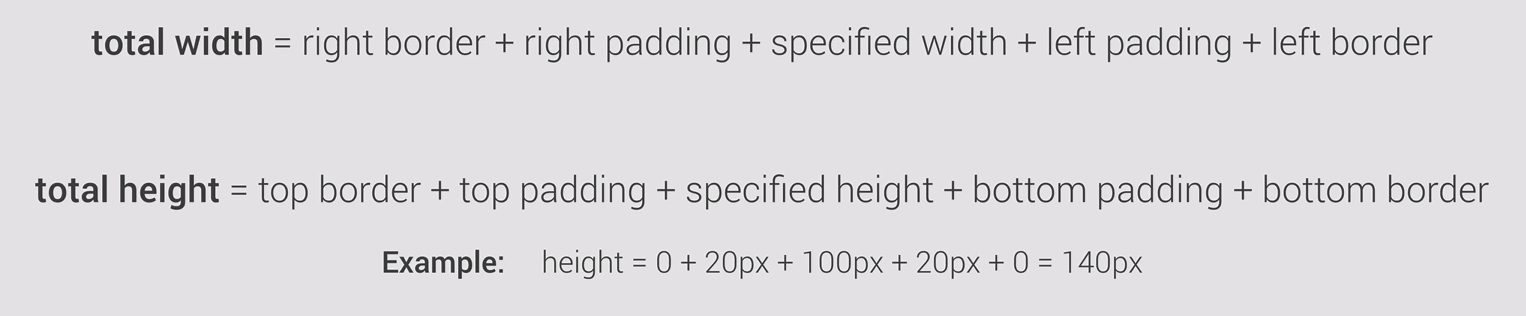


Padding: đệm them vào, đẩy background lên (still inside box)

Border: lớp vỏ, lớp viền ngoài cùng ôm 1 element

Margin: là khoảng cách đẩy ra từ element, ko làm element dày lên như padding và border (outside the box, space between boxes)

Box-sizing hoạt động chiều ngang của element = tổng border + padding + content



Position dùng để thiết lập vị trí   
Relative: tương đối ( ko phụ thuộc vào đối tượng khác, sẽ lấy chính vị trí nó đang đứng làm gốc tọa độ, nó phụ thuộc vào chính nó

Absolute: tuyệt đối (bị phụ thuộc vào thẻ parent gần nhất có thuộc tính position để lấy thẻ parent đó làm gốc tọa độ)

Fixed: vị trí phụ thuộc vào khung trình duyệt

Sticky: bám dính

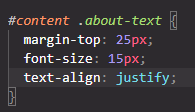
**Những thành phần thường gặp ở giao diện website:**

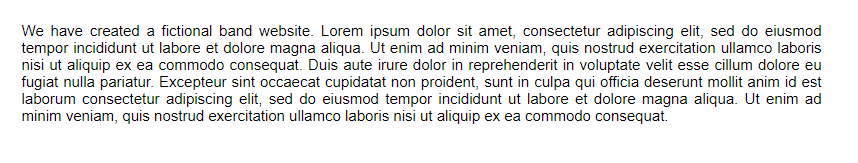
* + Header (đầu trang)
  + Navigation (điều hướng)
  + Breadcrumb (cho biết mình đang ở đâu trong website)

Ex: 

* + Sidebar
  + Slider (thanh trượt)
  + Banner
  + Content (Nội dung)
  + Footer (chân trang)

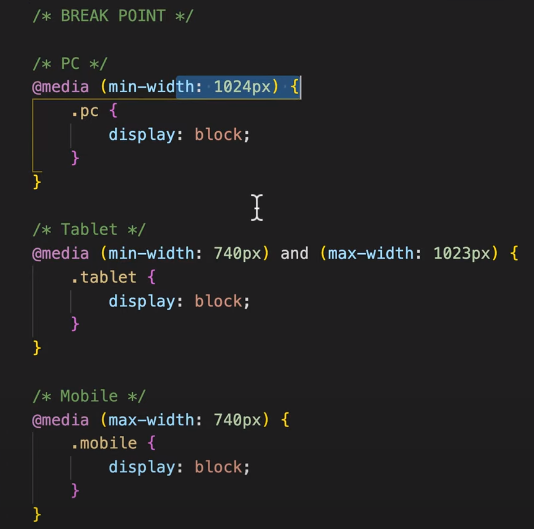
Canh 2 bên của paragraph





Line-height nếu text tiếng anh thì để 1.4, Vnese để 1.6

Breakpoint thường thấy của responsive: (file responsive.css để dưới style.css)



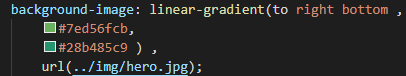
Chuyển text về uppercase (lowercase)



Cắt polygon, lên trang <https://bennettfeely.com/clippy/> sẽ có nhiều style:



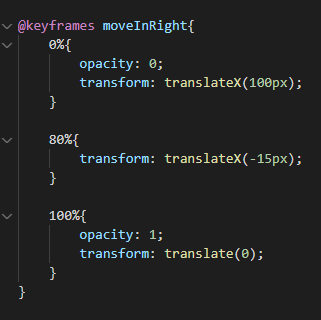
Tạo 1 lớp màu có opacity lên background:



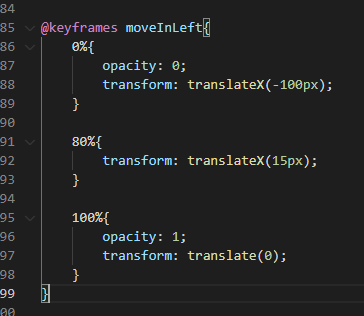
1 ví dụ của reset CSS ở \* và body



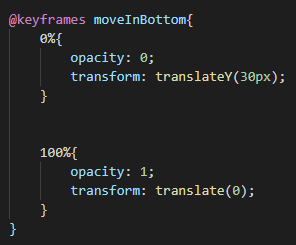
Animation từ phải qua (trục Ox), 0% là ở beginning state, 100% là end state:



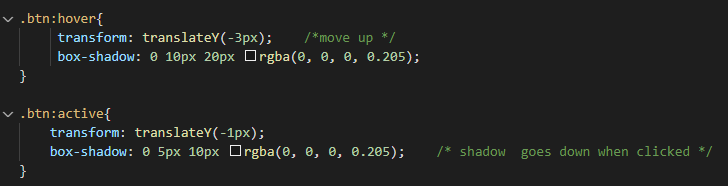
Animation từ trái qua (trục Ox)



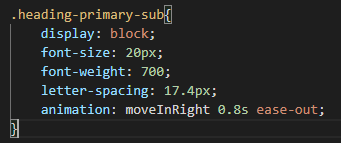
Animation từ dưới lên (trục Oy):



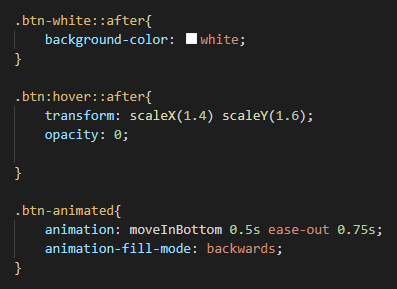
Hover vào thì nó sẽ đi lên và có shadow, click vào thì nó sẽ đi xuống xíu và shadow sẽ nhạt lại



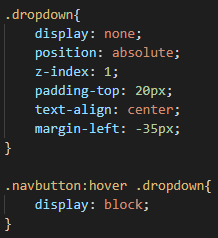
Gọi animation vô class:



Sau khi hover (:hover::after) thì nó phình ra và mờ dần lại:

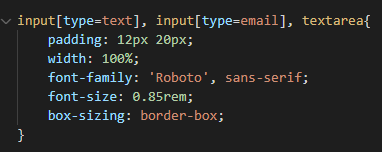


Hover tới thì menu drop down



Form:



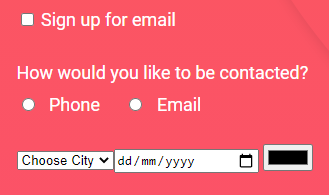


Input ra là:



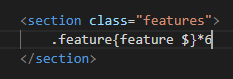
Form multiple choice





Checkbox (AND) ở Sign up thì chọn bao nhiều cái cũng dc, radio (OR) thì chỉ chọn dc 1 cái như Phone và Email

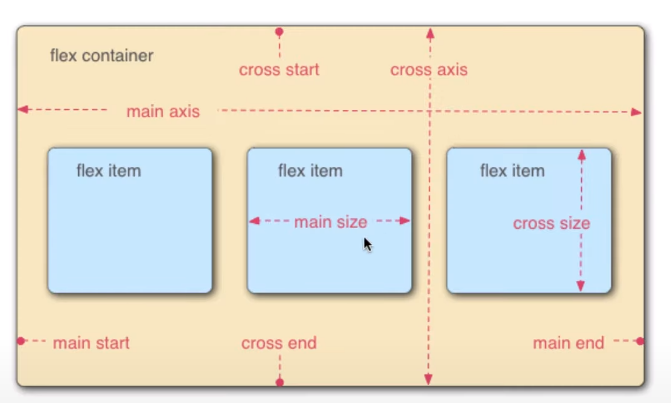
Thứ tự từ 1-> 6



RESPONSIVE

Media queries thì sử dụng đơn vị em sẽ đỡ bị lỗi, lên w3school covert px ra em

FLEX BOX



Có 2 thành phần: flex-container (thẻ chứa) và flex-item( thẻ con)

Main axis (trục chính): có main start và main end

Cross axis (luôn vuông góc với main axis): có cross start và cross end

Chiều của main axis sẽ quyết định chiều của item

Flex-direction: thay đổi hướng của main axis, value là row | column (default display: flex là row)

Flex-wrap: nowrap | wrap | wrap-reverse (default là nowrap) (wrap dung để xuống dòng, wrap-reverse thì đẩy lên

Flex-basic: xét kích thước main size

Justify-content: flex-start | flex-end | space-between | space-around : căng flex-item theo phương của main axis

Justify-self: : flex-start | flex-end | center

Align-content: cùng với hướng trục cross size

Align-self: set cho flex item (như justify-self)

Flex-grown: tăng kích thước flex-item

Flex-schrink: thu nhỏ



TIPS: Main Axis map với justify, Cross axis map với align

Kham thảo: <https://codepen.io/enxaneta/full/adLPwv/>

Quy ước BEM

BEM (Block Element Modifier) là tiêu chuẩn đặt tên class trong CSS

Block: thẻ cha

Element: phần tử trong block

Modifier: bổ sung ý nghĩa cho Block hoặc Element

Why BEM?

Vì mỗi ng đặt tên class 1 kiểu

Mỗi Member đặt tên class trùng nhau, CSS đè lên nhau

Syntax:

.block

.block\_\_element (2 dấu gạch dưới)

.block—modifier (2 dấu gạch ngang)

.block\_\_element—modifier

VD: block là card, btn là element, btn có 2 màu

Card\_\_btn—red

Card\_\_btn--green

WHEN to use?

Khi dự án có nhiều member

Dự án lớn, số lượng trang, thành phần nhiều

Trường hợp Block lồng Block

Block con là thành phần dung chung

Block con chứa nhiều element

GRID

Thành phần chính

* 1. Column: PC thường có 12/16 columns, tablet 8, mobile 4
  2. Gutter: khoảng ngăn cách giữa các column
  3. Margin: Khoảng cách lề trái/phải của bố cục chính website

Thành phần chính (CSS)

1. Grid: Lưới (thường là phần cha chứa row và column)
2. Column: Cột ( chiều dọc - chứa nội dung/ thành phần trên website)
3. Row: dòng (chiều ngang- chứa column)

Column padding bao nhiêu thì row margin âm bấy nhiêu để bù trừ cho nhau

Trong grid thì có row, trong row thì có column

