1. Lập danh sách

Thẻ <ol> hoặc <ul>

Thẻ <ol> sẽ tự động đánh dấu danh sách theo thứ tự 1. 2. 3. ….

Thẻ <ul> sẽ tự động đánh dấu danh sách theo các dấu chấm tròn to

Câu hỏi:

Thẻ <mark> dùng để đánh dấu đoạn văn bản mặc định sẽ bôi vàng, vậy có làm màu khác được k?

2. Khi dùng CSS nhớ thứ tự ưu tiên id > class > thẻ p

3. GITHUB

- Dùng lệnh pwd dùng để in đường dẫn của thư mục hiện tại

- Dùng lệnh mkdir (có dấu cách sau dir) để tạo thư mục

Trong khi dùng lệnh mkdir , nếu muốn tạo 1 lúc nhiều thư mục thì dùng dấu cách để gõ tên các thư mục muốn tạo

Ví dụ: mkdir folder1 folder2

Còn muốn tạo tên có dấu cách thì dùng dấu shift –

- Dùng lệnh cd để di chuyển thư mục

\*Trình tự thực hiện: dùng lệnh pwd trước để xem mình đang ở đâu, dùng lệnh ls để thống kê các thư mực, cuối cùng mới dùng lệnh cd để di chuyển vào trong thư mục đó

- Dùng lệnh touch để tạo tên file.

Ví dụ: touch file.txt file.png. Trong thư mục đang làm việc (sau khi dùng lệnh cd) sẽ tạo ra 2 file txt và png

- Dùng lệnh rm để xóa file

- Dùng lệnh rm -r để xóa folder

- Dùng lệnh cd .. để lùi lại 1 cấp, lệnh cd../.. để lùi về 2 cấp

- Dùng git init để Git quản lý thư mục

Lưu ý: Chỉ git init ở thư mục tổng, không dùng trong thư mục con

- Dùng lệnh git status kiểm tra trạng thái repo

Dùng lệnh git (dấu cách) add (dấu cách) tên thư mục để yêu cầu quản lý

- Dùng lệnh git add . để yêu cầu quản lý tất cả. (có dấu cách)

- Dùng lệnh git commit -m “ghi nội dung ghi chú” để đóng gói lại ver. đã làm

- Dùng lệnh git log để xem lịch sử đã thực hiện đã bao lần commit (ver.)

- Dùng lệnh git push -u <https://github.com/mailongvu/mailongvu.github.io.git> master để đẩy tài liệu lên Github

git push -u [https://github.com/mailongvu07061990/mailongvu.github.io.git](https://github.com/mailongvu/mailongvu.github.io.git)

- Dùng lệnh git remote add longvu <https://github.com/mailongvu/mailongvu.github.io.git> để gán địa chỉ vào giá trị origin

git remote add longvu [https://github.com/mailongvu/longvu.github.io.git](https://github.com/mailongvu/mailongvu.github.io.git)

Sau đó chỉ cần dùng lệnh Git push origin master để đẩy lại tài liệu

Lưu ý: Sau khi sửa file ở máy tính của mình, Phải git add . xong rồi mới git commit

Các nhóm thẻ

H1 đến h6

Sub sup

Ul ol

B u i

Strong em

Img a

Head body tittle

Meta

Big small

Khi git init thì mới chỉ có trên máy của mình, chưa đồng bộ trên mạng

Git clone git add git commit git push: quy trình lấy code lại từ trên giihub, trên máy chưa có

Dùng lệnh mv (link1) (link2) để di chuyển file

Sau khi dùng lệnh git add, GIT sẽ add thư mục vào 1 tầng trung gian staging area

Lệnh git restore --staged (tên thư mục) để gỡ file vừa add từ tầng trung gian quay trở về thư mục làm việc working dir

Dùng lệnh git pull origin master để tải .io trên github về thư mục mới, sau đó git push lại

Comment trong html là <!-- -->, còn trong css là /\* \*/. Phím tắt là ctrl + /

BUỔI HỌC 5: CSS LAYOUT

<head> và <header> là khác nhau

<head> nằm trong thẻ <body>

<header> thường đi với <footer>. Có thể không cần <footer>

Bản chất SideBar và navigation là giống nhau. Nhưng đã là sidebar thì phải nằm ở bên cạnh

Trong web nên có breadcrumb (điều hướng tại trang)

Nên bổ xung phân trang vào trang web, tuy chưa cần công dụng

Tất cả những gì cho phép nhập liệu vào thì nó là form, kể cả những nút bấm

Dùng lệnh float thì chỉ nên để left, các đối tượng sẽ tự chèn vào chỗ trống để đủ 100%

HỌC VỀ FLEX

|  |
| --- |
|  |

Main là theo chiều ngang

Cross là theo chiều thẳng đứng, dọc

Video học: <https://www.youtube.com/watch?v=DPwHRceskTA&feature=youtu.be>

Cả hai thuộc tính flex-direction và flex-wrap thường được sử dụng với nhau

 align-content xác định khoảng cách giữa các dòng, trong khi align-items xác định cách các hạng mục như một toàn thể được xếp trong hộp chứa. Khi chỉ có một dòng, align-content không có hiệu lực.

BUỔI HỌC 6: GITHUB BUỔI CUỐI

Lưu ý: đặt tên thư mục không nên để dấu cách

Dùng lệnh git checkout (id của dòng commit) thì git sẽ lấy nội dung của commit đó và đưa thư mục về trạng thái của lần commit đấy.

Detached head: là chế độ master đang chỉ vào 1 chỗ, còn head đang chỉ vào 1 chỗ khác. Bản chất HEAD và master đều là con trỏ.

Khi quay lại 1 commit ở dòng lịch sử, ta chỉ git log được id của từng commit phía trước nó. Khi đó muốn đưa trạng thái về mới nhất thì dung lệnh git checkout master.

Mỗi khi muốn checkout thì cần phải commit lại tất cả các thay đổi.

Dùng lệnh git branch dev để tạo con trỏ dev ngay tại vị trí con trỏ HEAD nhằm mục đích tạo nhánh mới. Và để vào nhánh đó, lại lập lại lệnh git branch dev 1 lần nữa. Sau đó phải dung lệnh git checkout dev

Khi đang ở nhánh master, muốn gộp từ nhánh dev vào thì dung lệnh git merge dev. Dùng :q để thoát khỏi màn hình merge.

Muốn merge 1 commit trước dev thì đơn giản dung lệnh git merge (id của commit). Nhưng thường chỉ merge cái mới nhất.

Mặc định body của trang web đều có margin bằng 8. Nên để margin=0 để xóa

Thuộc tính grid

display: grid

Grid-template-columns: 1fr 1 fr 1 fr 1fr

Grid-gap: 15px

Width: 100%

-> chia 4 cột, khoảng cách giữa theo cả chiều ngang lẫn dọc là 15px.

Còn nếu muốn căn theo chiều ngang và dọc khác nhau thì dung grid-gap-row và grid-gap-column

Khi chèn ảnh vào ô lưới vừa chia, phải dung maxwidth 100%

Dung float left thì thằng cha sẽ không tính toán đc kích thước của thằng con đang dung float. Khi đó dung thuộc tính clear: both

Dung display: block thì ảnh sẽ không bị các khoảng trắng ở dưới

-> Tìm hiểu lại thuộc tính này

Thuộc tính calc (tự động tính toán kích thước chiều rộng)

Width: calc ((100%-45px)/4)

BUỔI HỌC 7: HIỆU ỨNG CSS

BUỔI HỌC 8: MENU ĐA CẤP

Để định vị trí cho thằng con thì dùng absolute, còn thằng cha nó có thể dùng position gì cũng được, miễn là không phải static vì static là định dạng chung vị trí cho tất cả.

Nếu thằng cha dùng position relative thì thằng con sẽ định vị theo thằng cha.

Nếu thằng cha cũng dùng absolute thì thằng con định vị theo thằng cha, còn thằng cha sẽ phải định vị trí theo thằng to hơn. Tuy nhiên cách này sẽ tính toán phức tạp hơn.

Nếu dùng display inline thì k set padding được.

Thẻ ul hay có padding-left và có dấu chấm bên cạnh. Để bỏ thì cho padding-left bằng 0 và thuộc tính list-style-type: none

ÔN TẬP

Căn giữa

-inline theo chiều ngang: text-align: center gán cho phần tử chứa đối tượng inline

-block theo chiều ngang: margin auto cho chính phần tử cần căn giữa. Phải có chiều rộng xác định

HỌC JAVASCRIPT BUỔI 1

CÁC CÁCH KHAI BÁO CSS VÀ JAVA

Cả css và js đều có 2 cách khai báo. js thì khai báo đều phải dùng thẻ script, còn css thì dùng style khi khai báo trực tiếp trong html, còn dùng thẻ link href để trỏ từ đường dẫn khác.

Js <script>

.js <script>

Css <style>

.css <link href>

Bất kỳ đối tượng nào trong dấu nháy đơn hay kép thì đều là 1 chuỗi (string)

Dấu \” nó sẽ tương đường với dấu “

Ngoài ra có thể sử dụng cặp dấu ` ` (string template) để giới hạn chuỗi

Phép tính đối với số còn % là phép lấy dư. 5 % 2 kết quả là dư 1

Cuối 1 function thường có cụm từ return để trả kết quả về và kết thúc

Khai báo mảng (array) thì dùng dấu [ ], khai báo đối tượng (object) dùng dấu { }

DRY là gì? Don’t repeat yoursell

Single Entry Point là gì? (1 cửa) CSS là nó. Khai báo biến cũng là nó.

Syntax (cú pháp) và Coding Convention (quy ước do 1 người đặt ra)

Short hand/cut: i++ là 1 shoot hand

Implicit : ngầm chỉ định. 1 biến không khai báo, mà in luôn thì tùy ngôn ngữ lập trình sẽ ra giá trị khác nhau (0, chuỗi rỗng, do đó phải reset.css

Reset.css để khai báo tất cả thuộc tính để có thể hiển thị được tất cả trên các trình duyệt chứ không dùng mặc định theo từng trang web

gán sự kiện vào 1 thẻ

dùng onclick (dùng trên thẻ html)

dùng get...addEventListiner (dùng trên javascript)

document.querrySelector == $() == document.querrySelectorAll

document.getElementByTagName == $(“body”)

nếu muốn thay đổi nhiều thuộc tính bằng jQuerry thì $().css( {style để ở đây} )

khi dung on.submit thì phải nhớ trả về return

khi đối tượng là nút thì phải dung thẻ button

UI UX

cách tiếp cận, tìm hiểu nhu cầu của người dung như thế nào?

Xác định được mục tiêu của web/app -> xác định tệp khách hàng, hành vi thói quen người dung -> lựa chọn phong cách thiết kế

TRÁNH LỖI CHỮ XUỐNG DÒNG

Màu sắc là 1 trong 3 yếu tố quyết định việc mua hàng (2 yếu tố còn lại là gì?)

Dùng màu theo tông nhận diện thương hiệu và theo đặc tính sản phẩm (đối tượng khách hàng) -> tạo màu sắc chung, tổng thể của website

Bổ trợ màu: các màu có thể tương phản, đối lập nhưng với độ đậm nhạt tùy biến thì sẽ tạo sự cân bằng, giảm căng thẳng.

Tính tương phản: giảm sự mệt mỏi cho mắt, và tập trung sự chú ý của người dùng bằng cách phân chia rõ ràng.

Sống động: yếu tố mang lại cảm xúc cho người dùng. Chú ý tới đường đi của màu sắc tạo sự liên kết, chuyển động dọc trang web

Hãy lựa chọn màu sắc theo góc nhìn bussiness chứ không phải theo sở thích cá nhân

Trên mobile thì số chữ trên 1 dòng khoảng 30-40, còn trên máy tính, máy tính bảng thì 60-70 chữ 1 hàng

Tính phân cấp của chữ nhằm đạt hiệu quả truyền đạt nội dung. Cho phép thiết lập thứ tự tầm quan trong nội dung, cho người đọc dễ dàng tìm thấy những gì đang tìm kiếm và điều hướng nội dung.

Luôn chia làm 3 cấp độ phân cấp chữ.

Không sử dụng cùng màu hoặc các màu tương tự cho văn bản và nền

Phân chia nội dung rõ ràng theo khu vực.

Các chuyển động nhanh và các mục tiêu nhỏ dẫn tới tỷ lệ lỗi cao hơn

Đường dẫn tới hành động mong muốn càng ngắn thì trải nghiệm ng dùng càng tốt

Nên giảm khoảng cách giữa các đối tượng mà người dùng sử dụng theo thứ tự hợp lý

Đối với các hành động xóa thì khuyến khích nên dùng 2 bước confirm để hạn chế lỗi bấm bất ngờ từ người dùng

Tuy đưa ra nhiều lựa chọn có thể tạo ra sự phong phú nhưng đôi khi lại làm người dùng đưa ra quyết định chậm hơn.

Cat, house, fish, apple, door,flower,tree,square,

Link google bài giảng ui ux: <https://drive.google.com/drive/folders/1UbKSyNpTNL8mE-Gi45c217aclg3DC53e>

Nhận xét bài bảo vệ lớp 8

\* Ưu điểm\* :

+ Cố gắng trong việc cách điệu design để trang web khác biệt

\* Nhược điểm\* :

+ Menu chưa active đang ở page nào

+ Trình bày nói nhỏ, thiếu tự tin

+ Banner hơi sơ sài, chưa nêu được mục đích trang web ( Awareness )

+ Ảnh sản phẩm chưa có view nhanh ( xem tổng quá sản phẩm )

+ Chưa có view phóng to sản phẩm

+ Chưa có trường required ở filed đặt mua ( dấu \* )

+ Text thể hiện rối mắt, dày đặc

\* Ưu điểm :

+ Banner : Hình ảnh good, thu hút, có giá khuyến mãi...thể hiện được nội dung

+ Dễ đi được đến các gói

+ Dữ liệu demo, đầy đủ, đẹp

+ Thể hiện tốt mô hình AIDA

\* Nhược điểm:

+ Trình bày trung bình ( hơi nhỏ )

+ Ở blog -> thiếu suggest các bài khác ( thay vào đó là giá và đặt dịch vụ )

+ Resposive -> chưa tốt

+ Text color chưa phân biệt ra title, sub infor, thông tin tác giả..... ( đơn điệu text )

Tổng quan : Bài làm tốt, có đầu tư nghiêm túc cho đồ án

\* Ưu điểm:

+ Trình bày tốt.

+ Menu có active

+ View nhanh sản phẩm - good

+ Đăng ký thành công có thông báo - good

+ Dữ liệu demo ổn -> có thể search , fillter.. ( tất cả đã hoạt động )

UI và UX tốt

+ Có zoom sản phẩm trong trang chi tiết

+ Chi tiết design đồng nhất cho các filed , button...

+ Validate các trường -> good

+ Có validate và suggest khi điền filed

+ Mỗi action thành công dều có tông báo

+ Có giỏ hàng và thanh toán

\* Nhược điểm : Một vài nhược điểm nhỏ

Tổng quan: Bài làm tốt, tự tìm hiểu nhiều kỹ thuật phức tạp

\* Ưu điểm

+ Có giỏ hàng và thanh toán (với mình thì đây không tính là ưu điểm)

\* Nhược điểm

+ Thiếu thông tin liên hệ trên trang, ( không biết cửa hàng ở đâu, fanpage, instagram.. là gì )

+Từ sản phẩm ở trang chủ -> chưa link xem chi tiết

+ Chưa làm giao diện hoàn thành, chưa responsive

+Chưa làm nhiều thức khác ( fillter, validate )..

\* Tổng quan : Chưa đầu tư và hoàn thành đồ án

cơ bản là thế

đọc các bạn sẽ thấy điểm chung là có những bài bị những lỗi chủ yếu do không thật sự dành thời gian vào giao diện, nên bị thiếu sót về cả UX lẫn UI, ko thể hiện được bằng code

hình ảnh chiếm trung bình hơn 60% dung lượng để tải trang web. ảnh phải được linh hoạt, không được co, kéo dãn, thường được giữ chiều cao cố định và thay đổi được chiều rộng. Thủ thuật được hay dùng là che hình ảnh để hiển thị vùng tương đối, và do đó căn chỉnh hình ảnh theo tâm dọc

tùy biến top main menu thành top flow, mỗi menu thành 1 dòng hoặc thể loại drop down

trên điện thoại, mỗi item nên để thành dạng list (hàng cột). Khi nhìn vào điện thoại, người dùng chỉ nhìn được 4 đến 6 mục lưới mỗi lần. cái nào cần ng dùng biết thì phải làm nổi bật lên (như thay màu cho giá bán).

Nhược điểm của dạng lưới khiến ng dùng phải cuộn nhiều. do đó khắc phục bằng cách chuyển thành các nhóm chủ đề chung

Icon nên giống nhau (cùng là vùng hay dạng line)

Đang hơi nhiều màu xanh trên banner

Dòng giữa chính sách với banner đang hơi hẹp

Căn đều

Font chữ ở 3 mục chính to hơn. (khi nói về font chữ xem lại trích dẫn trong slide)

Nên cho 1 nút mua cho mỗi mục sản phẩm

Quy định của cửa hành thì chuyển thành text link

Lấy lại mật khẩu và quy định thì giống màu nhàu

Link trang web đến từng phần thì cho cỡ chữ to lên

Trang giỏ hàng căn ra đều dàn trang,

Pandding nhận bản tin cho nhỏ lại

Phần kiểm tra đơn hàng cho kích cỡ bằng đăng ký đăng nhập

Cỡ chữ nhỏ nhất là 14

Phân cấp cỡ chữ

Dãn cách đơn hàng của bạn

Để full các mốc thời gian

Làm thêm phần lịch sử đã mua hàng

Icon bộ lọc hơn to

Padding bộ lọc thêm và thêm icon bỏ bộ lọc

Bỏ chữ hiện có màu sắc, thêm nút mua hàng ở mỗi sản phẩm

Chỉnh lại phần giỏ hàng chi tiết, cỡ chữ to lên

Bỏ khung phần thông tin sản phẩm

Xem lại cách viết trang blog cách viết mới

Trang liên hệ 1 bên logo 1 bên text

Phần bắt buộc thì nên đánh dấu sao

Khi báo lỗi nên làm thông báo sinh động, k chỉ để mình text

Giới thiệu chủ đề, lý do chọn chủ đề, màu sắc, phông chữ. Trình bày từ trên xuống dưới, có tính năng gì thì trình bày luôn

Trong trang thì nên chia thành nhiều section

Khi gán sự kiện cho nút bằng jQuerry thì mỗi 1 lần gán thì nó sẽ lưu sự kiện đó, các sự kiện không ghi đè. Để xóa thì dung lệnh off

LINK CÁC BÀI VIẾT THAM KHẢO

1. Sử dụng Collapse ẩn hiện nội dung HTML trong Bootstrap

<https://xuanthulab.net/su-dung-collapse-an-hien-noi-dung-html-trong-bootstrap.html>

1. cách tạo custom checkbox

<https://evondev.com/cach-tao-custom-checkbox/>

1. cách tạo các hình khối

<https://css-tricks.com/the-shapes-of-css/>

1. Tìm hiểu before và after trong CSS

https://evondev.com/before-va-after-trong-css/

node\_modules bị màu xám thì lên server nó k hiện

nếu trong phương thức PUT mà không truyền vào đầy đủ các trường thì trường nào bị thiếu sẽ gán bằng rỗng

Cấu trúc ajax

$.ajax ({

url: “”,

method: “GET”

}).done (data => {})

.fail(err => {})

Sort theo asc là theo chiều tăng dần, desc là theo chiều giảm dần

-------------------------------------------------------------

**REACT**

<script crossorigin src="https://unpkg.com/react@16/umd/react.development.js"></script>

<script crossorigin src="https://unpkg.com/react-dom@16/umd/react-dom.development.js"></script>

<script src="https://unpkg.com/babel-standalone@6/babel.min.js"></script>

Class trong react thì đổi thành className, id thì vẫn giữ nguyên nhưng k dùng.

Những thuộc tính trong html phải viết lại theo kiểu campelCase

Tạo component trong react có 2 cách:

* Tạo bằng function thì nên viết hoa các chữ cái đầu tiền để phân biệt với các function khác. kết quả trả về là 1 JSX element

Ví dụ: function CartHeader(){

return(…)

}

Cách gọi: <CartHeader /> hoặc <CartHeader></ CartHeader>

Sau đó tách component thành 1 file js.

Props trong JSX cũng là 1 thuộc tính. Giá trị truyền vào trong thuộc tính có thể bất kỳ là thứ gì (khác với trong html là 1 string), và được khai báo trong 1 cặp dấu {}. Riêng chuỗi thì có thể viết luôn trong cặp dấu “”.

Chỗ function CartHeader(props) thì cái props là đại diện cho tất cả thuộc tính được khai báo khi gọi function

Khi gọi hàm trong html thì phải có dấu (), còn trong JSX chỉ truyền khai báo hàm mà không có dấu (), nếu có () thì nó sẽ thực hiện luôn hàm được gọi đó.

Use