**BÁO CÁO TRAINNING NGÀY 1**

**Câu 1: Function trong javascript có ý nghĩa gì ? Link tương đối là gì? Class trong CSS?**

* Trong javascript hay bất kỳ một ngôn ngữ lập trình nào cũng đều có những công việc cần lặp lại nhiều lần, mỗi lần như vậy đều phải viết lại code rất mất thời gian và làm code dài dòng. Function giúp chúng ta gom đoạn mã đó lại và tái sử dụng bằng cách gọi tên của chúng. Ngoài ra sử dụng function cũng giúp code ngắn gọn, dễ đọc và dễ bảo trì hơn.

Ví dụ: Hàm display() được gọi mỗi khi người dùng click vào thẻ input.

Text

Description automatically generated

Text

Description automatically generated

* Link tương đối là link dẫn đến một file hay một tài liệu khác trong một hệ thống dựa trên vị trí của file hiện tại.



* Class được dùng để xác định các thẻ html có chung các thuộc tính CSS. Sau khi khai báo class, chúng ta khai báo thuộc tính và giá trị trong class đó và định dạng cho các thẻ html bằng cách gán tên class đã khai báo cho thuộc tính “class”.

Ví dụ: Class button được sử dụng để định dạng cho các input

Graphical user interface, text, application

Description automatically generated

Text

Description automatically generated

**Câu 2: Cách chia cột bằng Grid trong bootstrap**

* Để chia cột trong bootstrap, đầu tiên chúng ta sẽ tạo một thẻ div có class tên là ‘row’, mỗi thẻ div như vậy sẽ là một hàng.
* Để chia cột, trong một hàng chúng ta sẽ tạo các thẻ div có class ‘col’, ví dụ muốn chia ba cột chúng ta tạo ba thẻ div có class là column.
* Chúng ta có thể điều chỉnh độ rộng cho cột bằng class col-n, trong đó n là một số tương ứng với kích thước , ví dụ như col-6 sẽ tạo ra một cột chiếm 50% độ rộng của thẻ cha.
* Chúng ta có thể giới hạn số cột trong một hàng bằng class row-cols-n, trong đó n là số cột tối đa trong hàng đó. Hoặc có thể để auto.

**Câu 3: Flexbox, position, BEM**

* Flexbox trong CSS giúp căn chỉnh các phần tử một cách linh hoạt.

Cách sử dụng:

1. Thêm cho thẻ cha thuộc tính display: flex
2. Căn chỉnh cho các thẻ con theo các thuộc tính

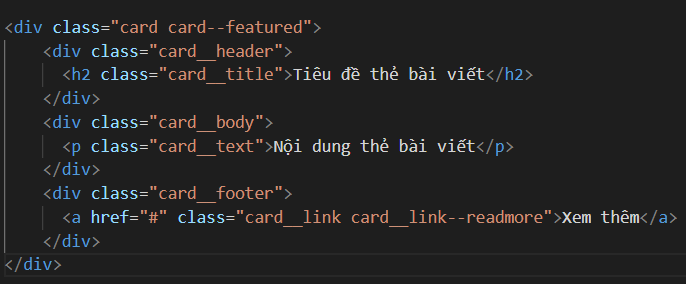
|  |  |
| --- | --- |
| Justify-content | Căn giữa theo chiều ngang của thẻ cha |
| align-items | Căn giữa theo chiều dọc của thẻ cha |
| flex-direction | Xác định hướng của các phần tử bên trong thẻ cha |
| order | Xác định thứ tự của các phần tử con |
| Flex | Phần tử con sẽ chiếm bao nhiêu diện tích của phần tử cha |

* Các loại position:

1. Relative: Di chuyển dựa trên vị trí gốc là vị trí ban đầu của phần tử
2. Absolute: Di chuyển dựa trên vị trí gốc là vị trí của thẻ cha gần nhất có thuộc tính position là relative hoặc absolute.
3. Fixed: Nằm ở một vị trí cố định trên trang web, không di chuyển khi người dùng cuộn trang web.

* BEM là viết tắt của "Block, Element, Modifier". Đây là một phương pháp đặt tên class cho các thành phần trong CSS. Trong đó Block là một thành phần lớn chứa các Elements bên trong, Modifier là một lớp đặc biệt dùng để thêm hoặc bớt các thuộc tính của Block hoặc Elements.

Ví dụ:



Ở ví dụ này, card chính là block với các elements là card\_\_header, card\_\_body, card\_\_footer, với modifier là card-featured.

**Câu 4: Cấu trúc project Laravel, các lệnh trong Laravel, Controller.**

|  |  |
| --- | --- |
| app | Chứa mã nguồn PHP của ứng dụng Laravel. Thư mục này cũng chứa các thư mục con để phân loại code, chẳng hạn như "Http" (chứa các Controllers, Middleware, Form Requests...), "Providers" (chứa các Service Provider), ... |
| bootstrap | Chứa các file khởi động cho ứng dụng, ví dụ như tạo ra instance của ứng dụng hoặc thiết lập các biến môi trường. |
| config | Chứa các file cấu hình cho ứng dụng, chẳng hạn như thông tin cơ sở dữ liệu, xác thực người dùng, mail,… |
| database | Chứa các migrations (để tạo hoặc chỉnh sửa cấu trúc CSDL), seeds (để khởi tạo dữ liệu mặc định) và các factories (để tạo data giả trong quá trình phát triển). |
| public | Chứa các file giao diện người dùng, JS, CSS, hình ảnh, fonts, ... Đây là nơi cung cấp các tài nguyên tĩnh cho ứng dụng. |
| resources | Chứa các file nguồn để phát triển giao diện người dùng như template blade, Sass/LESS, JS, hình ảnh, fonts, ... Đây là nơi để chỉnh sửa và phát triển các tài nguyên dùng cho giao diện trong ứng dụng. |
| routes | Chứa các file định tuyến của ứng dụng. File web.php quản lý các uri rút gọn và file api.php quản lý các API route. |
| storage | Chứa các file được sinh ra bởi hệ thống như session, file cache hay tệp tin log của ứng dụng. Hãy chắc chắn rằng nó có quyền write để lưu các tệp tin này. |
| tests | chứa các test case để kiểm tra tính ổn định, độ chính xác của ứng dụng Laravel. Laravel hỗ trợ nhiều loại test như unit, feature và browser testing. |

* Một số lệnh trong Laravel:

1. Tạo project mới: composer create-project laravel/laravel tên-project.
2. Tạo controller: php artisan make:controller tên-controller.
3. Tạo model: php artisan make:model tên-model.
4. Chạy project: php artisan serve.

* Controller là một lớp dùng để xử lý yêu cầu từ route hoặc view, sau đó gọi tới Model để lấy dữ liệu cần thiết rồi trả về View và hiển thị lên giao diện người dùng.

**Câu 5: Thực hành cắt giao diện từ html css vào Lavarel**

* Sau khi đã code được một trang web máy tính đơn giản bên html css, tiến hành cắt giao diện vào laravel.
* Trong folder resource/view, tạo một folder layouts để chứa phần giao diện dùng chung, folder pages để chứa các trang html.

**Text

Description automatically generated**

* File welcome: Cắt phần head + body để sử dụng chung.

**Text

Description automatically generated**

Sử dụng @yield để hiển thị nội dung của các blade view kế thừa từ trang này.

* File caculator: Cắt phần giao diện của máy tính.

Graphical user interface, text

Description automatically generated with medium confidence

Dùng @extends để kế thừa blade view welcome. Sử dụng @section để chỉ định phần nội dung sẽ hiển thị ở vùng @yield(‘main’)

* Sau khi đã cắt xong giao diện, tiến hành tạo controller với phương thức index trả về giao diện vừa cắt.

Text

Description automatically generated

* Trong file web.php, đăng ký router trả về phương thức index của AdminController.

Text

Description automatically generated