

**TRƯỜNG ĐẠI HỌC KIẾN TRÚC HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

A blue and white logo

Description automatically generated with low confidence

**CÔNG NGHỆ WEB**

**Đề tài : Xây Dựng Thiết Kế Website Nghe Nhạc**

**Nhóm 4 \_ 20CN5**

**Giảng viên hướng dẫn : Nguyễn Thị Hạnh**

**Nhóm Sinh viên thực hiện:**

1. **Nguyễn Đức Thắng**
2. **Nguyễn Thành Trung**

**Hà Nội \_2023**

**Lớp : 20VL**

**Giảng viên giảng dạy:**

Giảng viên giảng dạy : Lương Thị Bích Phượng

HÀ NỘI, 10/2023

MỤC LỤC

[1. Giới thiệu về HTML 3](#_Toc146468296)

[1.1 Hướng dẫn về cú pháp và các phần tử HTML cơ bản. 3](#_Toc146468297)

[1.2 Hướng dẫn tạo một tệp HTML cơ bản 5](#_Toc146468298)

[1.3 Tạo trang web với tiêu đề, đoạn văn bản, hình ảnh và siêu liên kết 5](#_Toc146468299)

[2. Cơ bản về CSS 6](#_Toc146468300)

[2.1 Hướng dẫn cách thêm kiểu cho trang web bằng CSS. 6](#_Toc146468301)

[2.2 Tạo các lớp và áp dụng kiểu bằng cách sử dụng lựa chọn. 7](#_Toc146468302)

[2.3. Thử nghiệm với các thuộc tính CSS cơ bản như font chữ, màu sắc, độ rộng và chiều cao. 11](#_Toc146468303)

[3. JavaScript Cơ bản 13](#_Toc146468304)

[3.1. Giới thiệu cú pháp JavaScript cơ bản 13](#_Toc146468305)

[3.2. Hướng dẫn tạo biến , hàm và sử dụng sự kiện 15](#_Toc146468306)

[3.3. Tạo các ví dụ đơn giản như hiển thị thông báo khi nút được nhấn. 17](#_Toc146468307)

[4. HTML Forms và Input Elements 18](#_Toc146468308)

[4.1. Hướng dẫn cách tạo biểu mẫu HTML đơn giản với các phần tử nhập liệu như ô văn bản, ô chọn, và nút gửi. 18](#_Toc146468309)

[4.2. Tạo kiểu cho biểu mẫu bằng CSS. 23](#_Toc146468310)

[4.3. Sử dụng JavaScript để xử lý dữ liệu từ biểu mẫu (vd: kiểm tra dữ liệu hợp lệ). 25](#_Toc146468311)

[a. Truy cập đến form dùng javascript 25](#_Toc146468312)

[b.Truy cập đến các control trong form để lấy value 26](#_Toc146468313)

[c. Kiểm tra dữ liệu trong Form với JavaScript 26](#_Toc146468314)

[5. Quy trình phát triển web 27](#_Toc146468315)

[5.1. Giới thiệu về quy trình phát triển web từ lập kế hoạch đến triển khai trang web. 27](#_Toc146468316)

[5.2. Xác định các bước quan trọng như phân tích yêu cầu, thiết kế giao diện, phát triển, kiểm thử, và triển khai. 29](#_Toc146468317)

[5.3 : Hướng dẫn về vai trò và trách nhiệm của từng thành viên trong nhóm, bao gồm quản lý dự án và phát triển. 30](#_Toc146468318)

[TÀI LIỆU THAM KHẢO 31](#_Toc146468319)

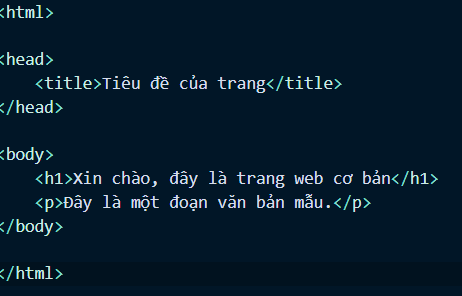
# 

# 1. Giới thiệu về HTML

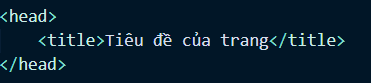
* 1. **Hướng dẫn về cú pháp và các phần tử HTML cơ bản.**
* Khai báo tài liệu HTML: Tài liệu HTML bắt đầu bằng dòng khai báo sau:



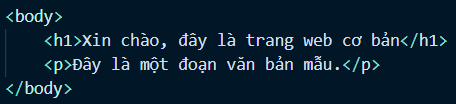
* Khai báo tài liệu HTML: Tài liệu HTML bắt đầu bằng dòng khai báo sau:



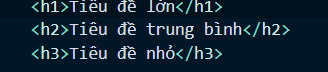
* Phần đầu (Head): Thẻ <head>...</head> chứa thông tin về tài liệu như tiêu đề, tập tin CSS, tập tin JavaScript, và các siêu liên kết.



* Phần thân (Body): Thẻ <body>...</body> chứa nội dung chính của trang web. html



* Tiêu đề (Headings): Thẻ từ <h1>...</h1> đến <h6>...</h6> dùng để định dạng tiêu đề với <h1> là tiêu đề lớn nhất và <h6> là tiêu đề nhỏ nhất.



* Đoạn văn bản (Paragraphs): Thẻ <p>...</p> dùng để tạo đoạn văn bản.



* Hình ảnh (Images): Để chèn hình ảnh, bạn sử dụng thẻ <img>. Thuộc tính src là đường dẫn đến hình ảnh và alt là văn bản mô tả hình ảnh (được sử dụng khi hình ảnh không thể hiển thị).



* Siêu liên kết (Hyperlinks): Để tạo liên kết đến trang web khác, bạn sử dụng thẻ <a>. Thuộc tính href chứa đường dẫn đến trang web mục tiêu.



* Bình luận (Comments): Bạn có thể thêm bình luận trong mã nguồn HTML bằng <!-- và -->.



* 1. **Hướng dẫn tạo một tệp HTML cơ bản**

Bước 1: Mở trình soạn thảo văn bản: Bạn có thể sử dụng các trình soạn thảo văn bản phổ biến như Notepad (trên Windows), Visual Studio Code, Sublime Text, hoặc bất kỳ trình soạn thảo nào bạn ưa thích.

Bước 2: Tạo file với đuôi .html

Bước 3: Trong file vừa tạo, thêm phần nội dung, văn bản, hình ảnh và các phần khác của trang.



Bước 4: Lưu tệp.

Bước 5: Sau khi bạn đã lưu tệp HTML, hãy mở trình duyệt web và kéo và thả tệp HTML vào trình duyệt hoặc sử dụng một trình duyệt để mở tệp.

* 1. **Tạo trang web với tiêu đề, đoạn văn bản, hình ảnh và siêu liên kết**

Để tạo một trang web với tiêu đề, đoạn văn bản, hình ảnh và siêu liên kết, bạn có thể sử dụng các thẻ HTML cơ bản như <html>, <head>, <body>, <h1>, <p>, <img>, và <a>.

Ví dụ:



# 2. Cơ bản về CSS

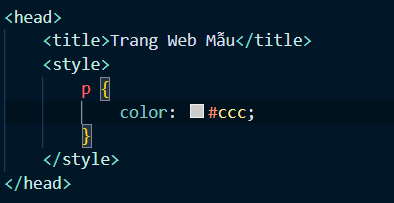
## **2.1 Hướng dẫn cách thêm kiểu cho trang web bằng CSS.**

Có 3 cách kết nối style CSS với trang web:

Cách 1: Kết nối bằng cách sử dụng thẻ <link>



Cách 2: Kết nối bằng cách sử dụng thẻ <style> trong phần <head>



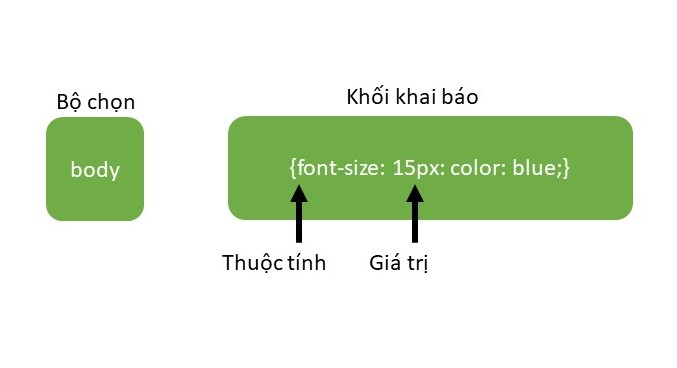
Cách 3: Kết nối bằng cách sử dụng thuộc tính style trực tiếp trên phần tử HTML



## **2.2 Tạo các lớp và áp dụng kiểu bằng cách sử dụng lựa chọn.**

**\* Cú pháp CSS**

Quy tắc CSS được thiết lập bao gồm một bộ chọn (selector) và một khối khai báo (declaration).. Bộ chọn trỏ tới phần tử HTML mà bạn muốn định kiểu. Khối khai báo chứa một hoặc nhiều khai báo ngăn cách nhau bởi dấu chấm phảy. Mỗi khai báo bao gồm một tên thuộc tính và một giá trị, ngăn cách nhau bởi dấu hai chấm.



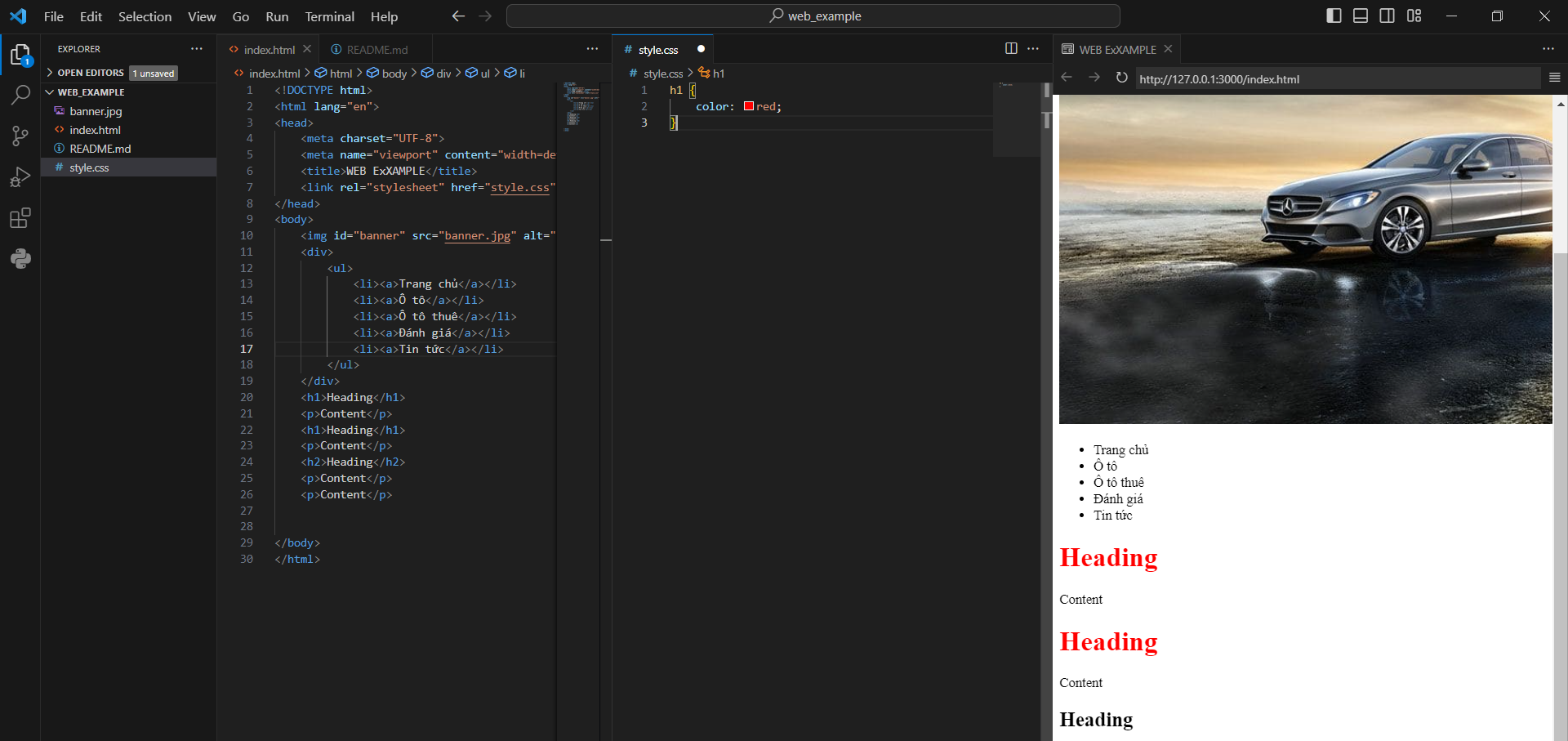
**\* Bộ chọn của CSS – CSS Selectors**

**a. Bộ chọn dựa vào tên thẻ – Element Selector**

Ví dụ .body{}, .h1{},…

Kiểu vùng chọn này là đơn giản nhất, đó là nó sẽ chọn toàn bộ các phần tử trên tài liệu HTML dựa vào tên thẻ có trong tài liệu rồi áp dụng CSS. Với kiểu sử dụng vùng chọn này thì toàn bộ các thẻ HTML trong website được chọn trong CSS đều biến đổi theo, tức là bạn không thể sử dụng kiểu chọn tên thẻ để viết CSS cho một khu vực độc lập.

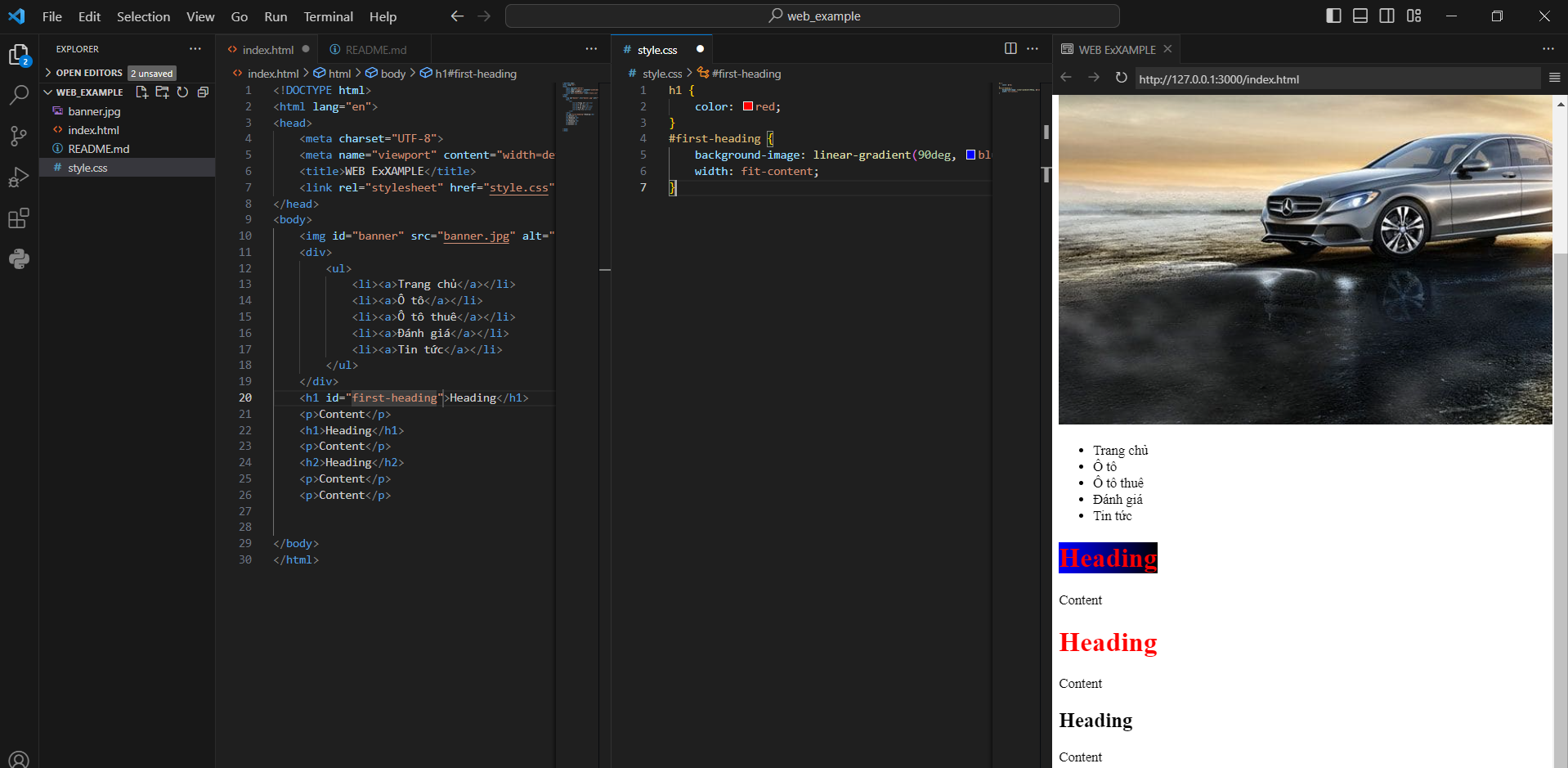
Ví dụ ta muốn thay đối style cho toàn bộ thẻ h1 trong website thì sẽ có đoạn CSS sau với vùng chọn h1.



**b. Bộ chọn dựa vào ID**

Vùng chọn dựa vào ID (tên định danh) nghĩa là bạn có thể chọn một phần tử cụ thể dựa vào giá trị của thuộc tính id trong thẻ HTML. Sở dĩ vùng chọn id được sử dụng để chọn một phần tử cụ thể là vì trên một trang tài liệu HTML thì mỗi phần tử phải mang một id riêng biệt không trùng nhau.

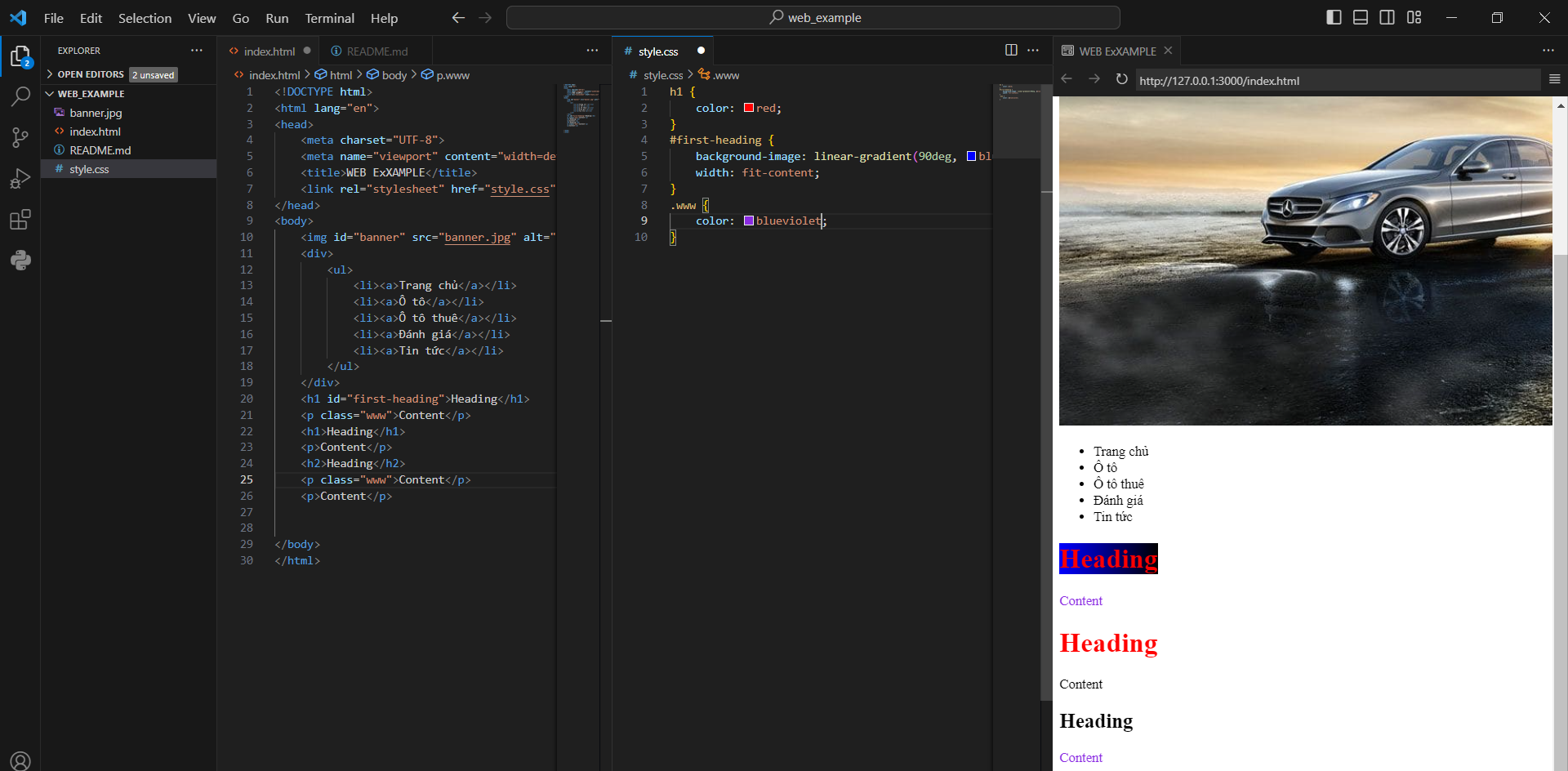
Id được thiết lập dựa vào thuộc tính id trong thẻ HTML và bất cứ thẻ nào cũng có thể sử dụng id. Khi viết tên id vào CSS thì nó phải có dấu thăng (#tên-id) đặt trước tên id để phân biệt với các loại vùng chọn khác. Ví dụ:



**c. Bộ chọn dựa vào Class**

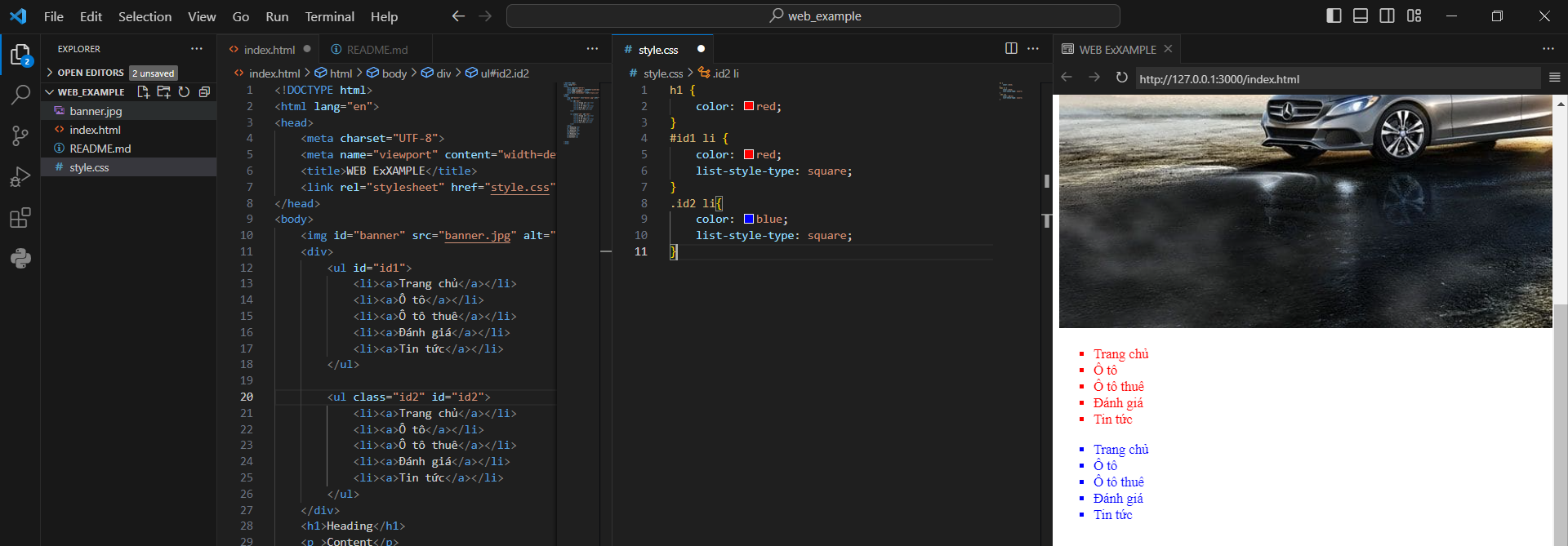
Class (lớp) cũng rất được sử dụng phổ biến như id nhưng một điểm khác biệt của class là một class có thể được sử dụng cho nhiều phần tử trên một trang tài liệu HTML, còn id thì chỉ được sử dụng một lần duy nhất cho một phần tử.

Class được khai báo trong một phần tử HTML bởi thuộc tính class như <h1 class="tên-class">. Khi khai báo vùng chọn class trong CSS, thì tên class phải được đặt sau dấu chấm (.tên-class). Ví dụ



**d. Bộ chọn theo thứ cấp**

Kiểu vùng chọn này bạn cũng sẽ sử dụng rất thường xuyên, đặc biệt là khi tiến hành viết CSS cho website đó là chọn phần tử theo thứ cấp. Nghĩa là với vùng chọn này, bạn có thể chọn một phần tử con trong một phần tử mẹ nào đó.





## **2.3. Thử nghiệm với các thuộc tính CSS cơ bản như font chữ, màu sắc, độ rộng và chiều cao.**

**\* Các thuộc tính màu sắc:**

***- Color:*** dùng để thiết lập màu sắc cho văn bản.

***- Opacity:*** dùng để thiết lập mức độ trong suốt của 1 phần tử.

**\* Các thuộc tính màu nền và đường biên:**

- Background

- Background-attachment

- Background-color: thiết lập màu nền của 1 phần tử.

- Background-image

- Background-position

**\* Các thuộc tính text:**

***- Thuộc tính text-align:*** được sử dụng để căn chỉnh văn bản có thể căn chỉnh văn bản sang trái, phải hoặc chính giữa.

Nếu thuộc tính text-align được thiết lập giá trị là "justify", các dòng sẽ được kéo dài sao cho chiều rộng bằng nhau.

***- Thuộc tính text-decoration:*** được sử dụng để tạo hoặc xóa các dấu gạch ngang khỏi văn bản.

Giá trị text-decoration: none thường được sử dụng để xóa các dấu gạch chân khỏi các liên kết.

***- Thuộc tính text-transform:*** được sử dụng để chỉ định, thiết lập chữ hoa hoặc chữ thường cho text và còn được sử dụng để chuyển đổi text thành chữ in hoa hoặc chữ thường, hoặc viết hoa chữ cái đầu tiên của mỗi từ.

***- Thuộc tính text-indent:*** được sử dụng để chỉ định thụt dòng đầu tiên trong đoạn văn bản.

***- Thuộc tính letter-spacing:*** được sử dụng để chỉ định khoảng cách giữa các ký tự trong một từ.

***- Thuộc tính line-height:*** được sử dụng để chỉ định khoảng cách giữa các dòng trong đoạn văn bản.

***- Thuộc tính direction:*** được sử dụng để thay đổi hướng văn bản của một phần tử.

***- Thuộc tính word-spacing:*** được sử dụng để chỉ định khoảng cách giữa các từ trong một câu.

***- Thuộc tính text-shadow:*** được sử dụng để tạo hiệu ứng chữ đổ bóng cho text.

# 3. JavaScript Cơ bản

## 3.1. Giới thiệu cú pháp JavaScript cơ bản

**a. Khái niệm JavaScript**

- Javascript là ngôn ngữ lập trình web và có thể thêm mới và update cả HTML và CSS.

- Javascript không chỉ gói gọn trong các ứng dụng front end trên trình duyệt mà còn có thể phát triển cả ứng dụng cho desktop, mobile hay server.

- Javascript được hỗ trợ trên tất cả các trình duyệt như Firefox, Chrome, IE… Thậm chí hỗ trợ các trình duyệt trên thiết bị di động.

* jQuery – thư viện Javascript để đơn giản hóa việc xử lý DOM.
* ReactJS – thư viện phát triển ứng dụng web hiện đại.
* NodeJS – là nền tảng phát triển ứng dụng phía server bằng javascript.

**b. Cú pháp JavaScript cơ bản**

- Một chương trình máy tính là một tập danh sách các hướng dẫn (statements) để máy tính thực thi. Và Javascript cũng vậy nhưng khác một điều là các tập lệnh này cho trình duyệt thực hiện thay vì máy tính.

**\* Cách khai báo biến**

- Javascript hỗ trợ 3 cách để khai báo một biến, đó là sử dụng các từ khóa: let, var và const.

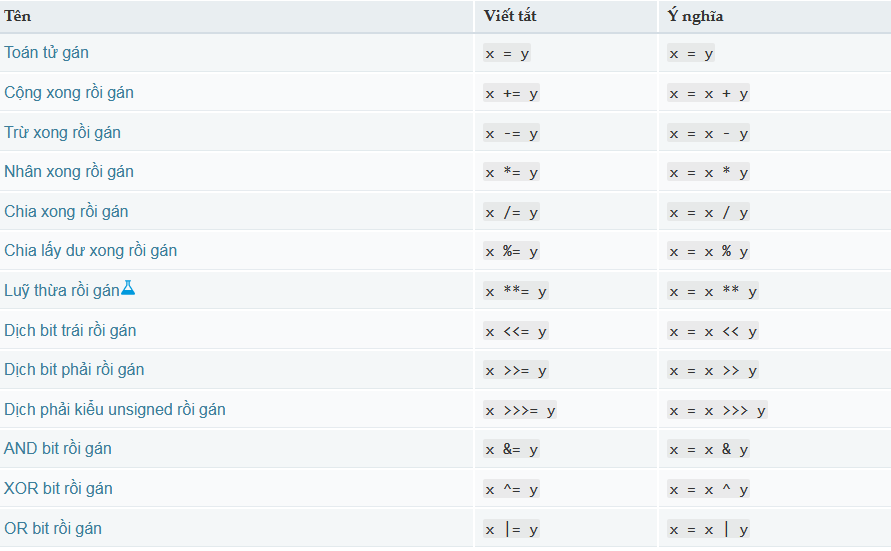
* Const: được dùng để khai báo hằng số, giá trị của nó không thay đổi trong suốt chương trình. Nó giống với từ khóa *final* của Java vậy.
* var: biến được khai báo với từ khóa *var* sẽ được truy cập ở bất kể đâu trong chương trình.
* let: từ khóa này giúp cho biến chỉ có thể được truy xuất bên trong phạm vi block quanh nó (block được định nghĩa bởi dấu ngoặc nhọn {} ).

**\* Kiểu dữ liệu của biến**

- Theo tiêu chuẩn ES7, có 7 kiểu dữ liệu:

* Kiểu boolean
* Kiểu null
* Kiểu undefined
* Kiểu số (int)
* Kiểu chuỗi (String)
* Kiểu Symbol (mới trong ECMAScript 6)
* Kiểu đối tượng (Object)

**\* Toán tử trong JavaScript**



## **3.2. Hướng dẫn tạo biến , hàm và sử dụng sự kiện**

**a. Tạo biến JavaScript**

- Khai báo và sử dụng biến trong JavaScript với var và let ,quy tắc đặt tên biến và phép toán gán ,khai báo hằng số với const trong js

-Biến dùng để lưu trữ giá trị các dữ liệu hay các đối tượng

-Giá trị của biến có thể được thay đổi cập nhật trong quá trình ứng dụng hoạt động (tên biến,khai báo và khởi tạo biến …)

- Quy tắc đặt tên biến :

+ Tên biến được tạo ra bởi các ký tự chữ , số , \_ và $

+ Tên không được phép bắt đầu bằng số ( chỉ bắt đầu bằng ký tự chữ , $ , \_ )

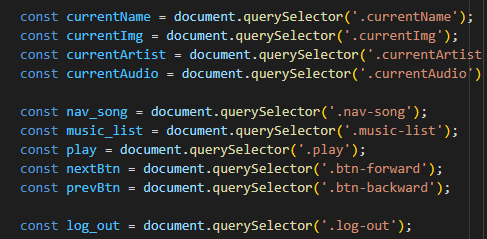
+ Không được chứa các ký hiệu đặc biệt như toán học , logic (như +,-,\*,>,<,…)

+ Không chứa khoảng trắng

+ Không được đặt tên biến trùng với các từ khóa dành riêng cho ngôn ngữ JavaScript liệt kê ở bảng sau

- Khái báo và khởi tạo biến

Lấy ví dụ về một phần bài tập nhóm làm ví dụ ta có :



**b. Tạo hàm JavaScript**

function renderContent(music) {

    let htmls = music.map((song) => {

        return `

                <div class="m-4" style="width: 200px;">

                  <div style="position: relative;">

                      <img src="${song.image}" class="rounded-3 img-fluid" style="object-fit: cover; width: 200px; height: 200px;">

                        <a href="#">

                          <div class="song-hover">

                            <i class="h1 bi bi-play" style="left: 50%;top: 50%; position: absolute; transform: translate(-50%, -50%);"></i>

                            </div>

                        </a>

                    </div>

                <div>

                    <div class="lead text-white">

                        ${song.title}

                    </div>

                    <small class="text-white">${song.artist}</small>

                    </div>

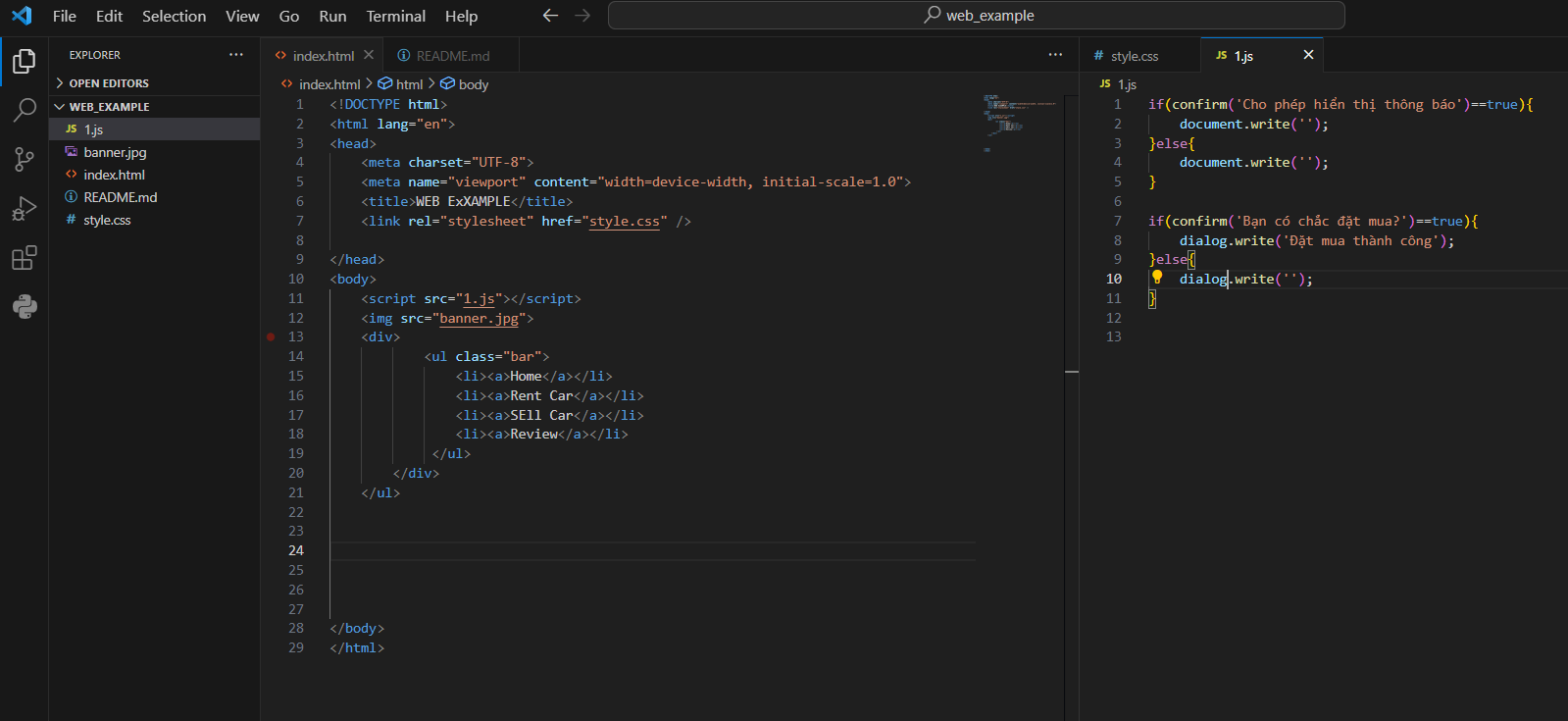
                </div>

        `

    })

Hàm được tạo ra có nghĩa là tên thuộc tính của hàm chứa tên đối tượng. Nó rất hữu ích khi xem cuộc gọi: trong quá trình gỡ lỗi hoặc đọc thông báo lỗi.

## **3.3. Tạo các ví dụ đơn giản như hiển thị thông báo khi nút được nhấn.**

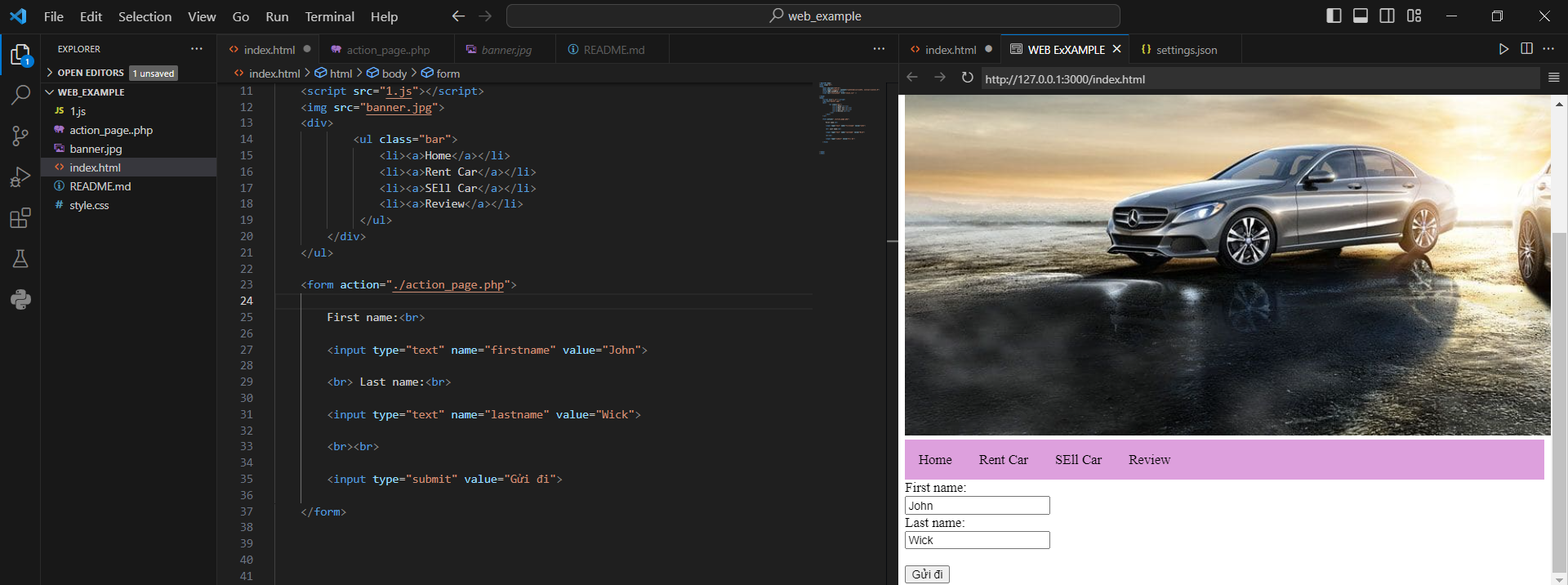


- Ví dụ: sử dụng js để kiểm tra form và hiện thông báo



# 4. HTML Forms và Input Elements

## **4.1. Hướng dẫn cách tạo biểu mẫu HTML đơn giản với các phần tử nhập liệu như ô văn bản, ô chọn, và nút gửi.**



- Phần tử HTML <form> xác định một biểu mẫu dùng để thu thập thông tin người dùng nhập vào. Các phần tử form là các kiểu phần tử nhập khác nhau như trường nhập văn bản, hộp kiểm, nút radio, nút gửi,v.v… Dưới đây là các thanh phần :

**Cấu trúc form**

<form action="process.php" method="get">

<!--Các phần tử form-->

</form>

Trong đó:

* <form> ... </form>: Thẻ khai báo một form.
* action: Thuộc tính khai báo đường dẫn xử lý dữ liệu sau khi người dùng submit form.
* method: Thuộc tính khai báo phương thức xử lý dữ liệu bao gồm POST và GET.

Lưu ý: Phần action và method qua nội dung php bạn sẽ được nắm rõ hơn để khai báo. Còn trong quá trình xây dựng html tạm thời bạn cứ để trống.

**Các phần tử form thường dùng**

**Input text – nhập đoạn text ngắn**

Đây là phần tử form dùng nhập text có số lượng ký tự ngắn như tên người dùng, địa chỉ, tiêu đề bài viết – sản phẩm…

Ví dụ bên dưới chúng ta tạo ra input nhập username

<form action="" method="">

Username: <input type="text" name="username">

</form>

**Input password – nhập mật khẩu**

Để lập những thông tin cần bảo mật như mật khẩu trong form có type = "password" giúp chúng ta nhập thông tin một cách an toàn.

<form action="" method="">

Password: <input type="password" name="password">

</form>

**Input submit – gửi dữ liệu lên server**

Khi đã nhập dữ liệu vào form để gửi nó lên server chúng ta cần khai báo input type = "submit". Khi người dùng nhấn vào nút submit dữ liệu sẽ được gửi lên server để xử lý.

Bên dưới là demo form đăng nhập bao gồm username, password, và một nút bấm đăng nhập.

<form action="" method="">

Username: <input type="text" name="username"><br>

Password: <input type="password" name="password"><br>

<input type="submit" name="btn\_login" value="Đăng nhập">

</form>

**Input email – nhập địa chỉ email**

Trong quá trình đăng ký tài khoản hoặc các form đăng ký nhận bản tin, giảm giá thì email là một phần không thể thiếu.

Để nhập liệu email chúng ta sử dụng input type = "email"

<form action="" method="">

<input type="email" name="email">

</form>

**Input date – nhập ngày tháng**

<form action="" method="">

<input type="date" name="birthday">

</form>

**Textarea – nhập đoạn văn**

Textarea thường dùng để nhập liệu đoạn văn dài như mô tả bài viết, nội dung bài viết, nội dung sản phẩm…

<form action="" method="">

<textarea name="post\_content" cols="30" rows="10"></textarea>

</form>

**Trong đó**

* cols: Độ rộng của ô nhập liệu
* rows: Độ cao của ô nhập liệu

**Input number – nhập số lượng**

Trong một số trường hợp dữ liệu nhập vào ở dạng số nguyên bạn có thể sử dụng type = "number". Phần này thường dùng trong chọn số lượng sản phẩm trong giỏ hàng.

<form action="" method="">

<input type="number" name="num\_order" min="1" value="1" max="10">

</form>

Trong đó

* min: Thuộc tính khai báo giá trị bé nhất của dữ liệu.
* max: Thuộc tính khai báo gía trị lớn nhất của dữ liệu.
* value: Thuộc tính khai báo giá trị mặc định.

**Checkbox – hộp chọn nhiều**

Trong trường hợp muốn khảo sát thông tin từ người dùng về một chủ đề có nhiều lựa chọn. Người dùng có thể chọn một hoặc nhiều option thì chúng ta sử dụng input = "checkbox".

Ví dụ bên dưới giúp người dùng khai báo kỹ năng của họ bao gồm html, css, php.

Lưu ý: Các checkbox có cùng một chủ đề bạn cần sử dụng có cùng giá trị của thuộc tính name

<form action="" method="">

<input type="checkbox" name="skill" value="html"> Html<br>

<input type="checkbox" name="skill" value="css"> Css<br>

<input type="checkbox" name="skill" value="php"> Php<br>

</form>

**Radiobox – hộp chọn một trong số**

Khi muốn lấy thông tin của người dùng chỉ được chọn một trong số rất nhiều lựa chọn chúng ta sử dụng input type="radio".

Ví dụ bên dưới giúp người dùng chọn giới tính của mình.

<form action="" method="">

<input type="radio" name="gender" value="male">Nam<br>

<input type="radio" name="gender" value="female">Nữ<br>

</form>

**Selectbox – chọn qua thanh xổ**

Một chức năng thường dùng select chính là thanh chọn vị trí tỉnh/thành phố đang ở.

<form action="" method="">

<select name="city">

<option value="">Hà Nội</option>

<option value="">Hải Phòng</option>

<option value="">Đà Nẵng</option>

<option value="">TP.HCM</option>

<option value="">Tỉnh khác</option>

</select>

</form>

## **4.2. Tạo kiểu cho biểu mẫu bằng CSS.**

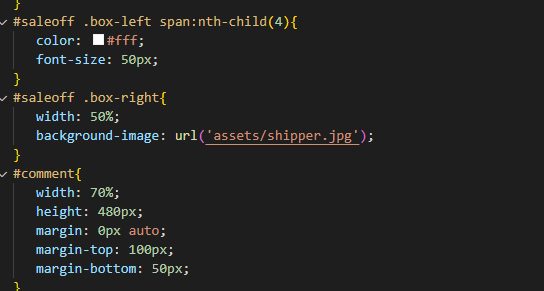
**a. Khái niệm**

- Biểu mẫu (Form) là đối tượng giúp ích cho việc nhập và hiển thị dữ liệu, thông tin một cách trực quan, có hệ thống, có thể thống kê dễ thuận tiện cho việc điều khiển một ứng dụng, hoặc thực hiện một chức năng nào đó.

**b. Tạo kiểu cho biểu mẫu HTML bằng CSS**

- Sử dụng thuộc tính *width* để xác định rõ chiều rộng của trường đầu vào:

+ Ví dụ: áp dụng cho tất cả các phần tử <input> :



- Định dạng *padding* cho trường đầu vào:

+ Sử dụng thuộc tính padding để thêm khoảng trống bên trong trường văn bản.

+ Khi có nhiều trường đầu vào khác nhau có thể thêm margin, để thêm khoảng trống bên ngoài cho trường đầu vào.

+ Đặt thuộc tính box-sizing là border-box để đảm bảo phần padding và border cuối cùng được đặt trong tổng chiều rộng và chiều cao của các phần tử.

- Định dạng đường viền cho trường đầu vào:

+ Sử dụng thuộc tính *border* để thay đổi kích thước và màu sắc đường viền

+ Sử dụng thuộc tính *border-radius* để bo góc tròn

+ Nếu chỉ muốn thêm đường viền dưới cùng thì sử dụng thuộc tính *border-bottom:*

- Định dạng màu cho trường đầu vào:

+ Sử dụng thuộc tính *background-color* để thêm màu nền cho trường đầu vào, và thuộc tính *color* thay đổi màu văn bản:

- Thêm biểu tượng/ảnh vào bên trong trường đầu vào:

+ Sử dụng thuộc tính *background-image* và thuộc tính vị trí *background-position* để thêm icon vào trong trường đầu vào. Chú ý: thêm *padding-left* để dành khoảng trống cho biểu tượng:

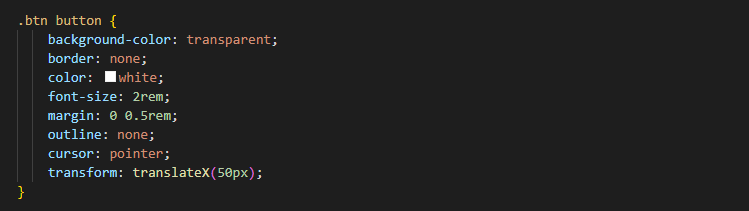
- Thêm hiệu ứng tìm kiếm cho trường đầu vào:

+ Sử dụng thuộc tính *transition* trong CSS để tạo hiệu ứng chiều rộng cho trường tìm kiếm khi người dùng click chuột vào trường nhập văn bản.

- Tạo kiểu cho Textareas:

+ Sử dụng thuộc tính *resize* để ngăn các vùng văn bản bị thay đổi kích thước ( tắt “grabber” ở góc dưới cùng bên phải):

- Định dạng button:

.

## **4.3. Sử dụng JavaScript để xử lý dữ liệu từ biểu mẫu (vd: kiểm tra dữ liệu hợp lệ).**

### **a. Truy cập đến form dùng javascript**

Bằng cách sau :

+ Truy xuất đến form theo chỉ số : document.form[ chỉ số]

+ Truy xuất đến form theo tên : document.form[‘tênform’]

+ Truy xuất đến form theo cú pháp ngắn gọn : document.tênform

+ Ví dụ :

<form name="timkiem">

<input name="tukhoa" value="Từ khóa" />

</form>

<form name="dangnhap">

<input name="un" value="teonv" />

<input name="pw" value="123" />

</form>

<script>

console.log(document.timkiem.tukhoa.value); //Từ khóa

console.log(document.forms[1].un.value); //teonv

console.log(document.forms['dangnhap'].pw.value); //123

</script>

### **b.Truy cập đến các control trong form để lấy value**

+getElementsByName: documnet.getElementsByName(‘namecủaControl’)[chỉ số].value;

+ Truy cập thông qua form :

Document.tênForm.tênControl.value;

+ Ví dụ :

<form name="tk">

<input type="text" name="tukhoa" id="kw" value="Từ khóa">

<button type="submit">Tìm kiếm</button>

</form> <hr>

<script>

t = document.tk.tukhoa.value; //Từ khóa

t = document.forms[0].tukhoa.value; //Từ khóa

t = document.getElementById("kw").value; //Từ khóa

</script>

### **c. Kiểm tra dữ liệu trong Form với JavaScript**

- Là kiểm tra xem thông tin có được điền vào form đúng yêu cầu hay chưa. Ví dụ tuổi hay số lượng sản phẩm phải là kiểu số nguyên, ngàysinh được được là chữ, mã sinh viên phải dài 7 ký tự …. Nhờ kiểm tra và báo lỗi, user được thông báo và nhập lại cho đúng, khi đó thông tin được xử lý và lưu trữ mới mới có giá trị

- Thực hiện kiểm tra dữ liệu trong form với javascript : Hàm gọi trong sự kiện onSubmit của tag form để kiểm tra dữ liệu nếu không hợp lệ thì trả về false ,còn nếu hợp lệ thì trả về true.

- Ví dụ :

<form name="frmlogin" action="xuly.php" **onsubmit="return kiemtra()"**>

<div>

<label>Username</label> <input type="text" name="u" id="un">

</div>

<div>

<label>Password</label> <input type="password" name="p" id="pw">

</div>

<div>

<button type="submit">Đăng nhập</button>

</div>

<h4 id="baoloi"></h4>

</form>

# 5. Quy trình phát triển web

## **5.1. Giới thiệu về quy trình phát triển web từ lập kế hoạch đến triển khai trang web.**

**\* Quy trình phát triển web :**

**- B1: Thu thập thông tin**

Giai đoạn này tập trung tìm hiểu và nghiên cứu về yêu cầu khách hàng để quyết định các bước tiếp theo sẽ thực hiện như thế nào. Nhiệm vụ quan trọng nhất trong bước này là hiểu rõ về mục đích của website, các chức năng website sẽ cung cấp, nhóm người dùng mục tiêu của website và thu thập đầy đủ các yêu cầu từ phía khách hàng, người dùng. Trả lời những câu hỏi trên sẽ giúp chúng ta lựa chọn được phương án tốt nhất để triển khai dự án

**- B2: Lập kế hoạch**

Ở bước này, lập trình viên sẽ đưa cho khách hàng cái nhìn tổng quan về website bằng các sitemap, wireframe hoặc bảng biểu dựa trên các thông tin đã thu thập được ở bước 1.

**- B3: Thiết kế giao diện**

Trong giai đoạn thiết kế, trang web sẽ trực quan hơn với các nút, hình ảnh, video,... sẽ được tạo , tất cả thông tin được thu thập trong giai đoạn đầu tiên là rất quan trọng. Các yêu cầu của khách hàng và trải nghiệm người dùng phải luôn được ghi nhớ trong khi thực hiện thiết kế.

**- B4 : Viết tài liệu đặc tả SRS**

SRS là tài liệu được sử dụng để mô tả chi tiết các yêu cầu chức năng và phi chức năng của hệ thống. Tài liệu này sẽ hỗ trợ đưa ra các tính năng của hệ thống hoặc dùng cho việc đọc hiểu hệ thống của phía khách hàng.

Đây là một tài liệu quan trọng cho đội phát triển (system analyst, business, analyst, code) và kiểm thử (tester). Ngoài ra, dựa vào các yêu cầu trong SRS, ta có thể đánh giá được số lượng scope, thời gian hoàn thành và các chi phí trong quá trình phát triển sản phẩm một cách nhanh chóng và dễ dàng hơn.

**- B5: Viết mã nguồn**

Bắt tay vào tạo trang web. Trước tiên khi bắt tay vào viết mã nguồn, cần lựa chọn công nghệ, ngôn ngữ lập trình phù hợp cho dự án.

**- B6: Kiểm thử , đánh giá và ra mắt sản phẩm**

Quá trình kiểm thử là quá trình thường xuyên nhất của một quy trình. Mọi liên kết nên được kiểm tra để đảm bảo rằng không có liên kết nào bị lỗi hay gián đoạn. Nên kiểm tra mọi biểu mẫu, mọi câu lệnh và kiểm tra lỗi chính tả của toàn bộ trang web. Sử dụng các tiêu chuẩn web chung để kiểm tra xem mã nguồn có đảm bảo sự tương thích giữa các trình duyệt hay không.

**- B7 : Bảo trì**

Điều quan trọng cần lưu ý, hay coi trang web là một dịch vụ hơn là một sản phẩm. Nên đảm bảo mọi thứ đều hoạt động tốt, mọi người đều hài lòng và luôn chuẩn bị sẵn sàng để thực hiện các thay đổi trong trường hợp khác.

## **5.2. Xác định các bước quan trọng như phân tích yêu cầu, thiết kế giao diện, phát triển, kiểm thử, và triển khai.**

- Công nghệ web thường tuân theo quy trình phát triển phần mềm chung và các bước quan trọng trong quy trình phát triển công nghệ web là:

+ Phân tích yêu cầu: Đây là giai đoạn đầu tiên và quan trọng nhất trong quy trình phát triển. Tại giai đoạn này, cần hiểu rõ yêu cầu của khách hàng, xác định tính năng và chức năng cần có trong ứng dụng web.

+ Thiết kế giao diện: Sau khi có được yêu cầu của khách hàng cần thiết kế giao diện người dùng (UI/UX) cho ứng dụng web. Điều này bao gồm việc xác định các trang và luồng điều hướng giữa chúng.

+ Phát triển: Ở bước này cần triển khai mã nguồn để xây dựng chức năng của ứng dụng web. Có thể sử dụng các ngôn ngữ lập trình phù hợp như HTML, CSS, JavaScript, PHP, Ruby, Python, Java,… để phát triển ứng dụng.

+ Kiểm thử: Sau khi hoàn thành phát triển, chúng ta cần tiến hành kiểm thử ứng dụng. Kiểm thử bao gồm kiểm tra tính năng, tương thích, hiệu suất và bảo mật của ứng dụng web.

+ Triển khai: Khi kiểm thử thành công, triển khai ứng dụng web lên môi trường hoạt động thực tế. Điều này có thể là máy chủ chạy trên internet hoặc một môi trường đám mây như AWS, Google Cloud...

+ Bảo trì và nâng cấp: Sau khi triển khai, cần duy trì và cải tiến ứng dụng web theo yêu cầu của khách hàng. Bảo trì và nâng cấp bao gồm khắc phục lỗi, bảo mật và cập nhật tính năng mới cho ứng dụng.

- Ngoài ra, quy trình phát triển công nghệ web còn có thể bao gồm các bước khác như thu thập phản hồi từ người dùng, tối ưu hóa hiệu suất và tăng cường bảo mật. Các bước này có thể thay đổi phụ thuộc vào quy mô và phạm vi của dự án quản lý dự án và phát triển.

## **5.3 : Hướng dẫn về vai trò và trách nhiệm của từng thành viên trong nhóm, bao gồm quản lý dự án và phát triển.**

**Quản lý dự án:**

Trưởng nhóm dự án: Chịu trách nhiệm về lập kế hoạch, giám sát và điều phối công việc của toàn bộ nhóm. Đảm bảo dự án được triển khai đúng tiến độ và chất lượng.

**Phân tích:**

Nhân viên phân tích dự án: Nghiên cứu, phân tích và đặc tả yêu cầu của dự án. Liên hệ và làm việc với khách hàng để hiểu rõ các yêu cầu và mong muốn.

**Thiết kế:**

* Nhân viên thiết kế giao diện: Tạo ra thiết kế giao diện dự án dựa trên yêu cầu và sự hỗ trợ từ phân tích dự án.
* Nhân viên thiết kế hệ thống: Thiết kế kiến trúc hệ thống và quyết định về cấu trúc, giao thức và công nghệ phù hợp.

**Phát triển:**

* Nhân viên lập trình: Xây dựng và triển khai phần mềm, sáng tạo giải pháp kỹ thuật dựa trên yêu cầu và thiết kế.
* Nhân viên kiểm thử: Thực hiện các kiểm thử phần mềm, đảm bảo rằng phần mềm hoạt động đúng và đáp ứng yêu cầu.

**Quản lý chất lượng:**

Nhân viên quản lý chất lượng: Theo dõi chất lượng và hiệu suất của phần mềm, thực hiện các kiểm tra và đánh giá kỹ thuật.

**Quản lý tiến độ:**

Nhân viên quản lý dự án: Theo dõi tiến độ dự án, quản lý nguồn lực và giải quyết các vấn đề phát sinh trong quá trình thực hiện dự án.

**Hỗ trợ và bảo trì:**

* Nhân viên hỗ trợ: Cung cấp hỗ trợ kỹ thuật, giải quyết sự cố và đào tạo người dùng cuối.
* Nhân viên bảo trì: Đảm bảo phần mềm hoạt động ổn định sau khi triển khai, thực hiện bảo trì và cập nhật.

# TÀI LIỆU THAM KHẢO

- Link tài liệu:

<https://timoday.edu.vn/bai-3-css-selectors/>

https://thachpham.com/web-development/html-css/vung-chon-css-co-ban.html

- Link tài liệu: [Tổng hợp các thuộc tính của CSS | Tìm ở đây (timoday.edu.vn)](https://timoday.edu.vn/tong-hop-cac-thuoc-tinh-cua-css/#text)

**-** Link tài liệu: [Javascript là gì? Cú pháp Javascript cơ bản - VNTALKING](https://vntalking.com/cu-phap-javascript-co-ban-nhung-buoc-di-dau-tien.html#javascript-la-gi)

Tài liệu tham khảo: https://www.w3schools.com/

<https://www.codecademy.com/catalog>

Tài liệu tham khảo: A Complete Guide to Flexbox (https://csstricks.com/snippets/css/a-guide-to-flexbox/), A Complete Guide to Grid

(<https://css-tricks.com/snippets/css/complete-guide-grid/>)

Tài liệu tham khảo: http://html5doctor.com/

https://css-tricks.com/

https://www.pluralsight.com/courses/html5-css3-

fundamentals

Tài liệu tham khảo: https://moz.com/beginners-guide-to-seo

https://yoast.com/seo-blog/

<https://yoast.com/seo-blog/>

Tài liệu tham khảo: https://git-scm.com/book/en/v2

<https://www.atlassian.com/git>