

**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 : Thiết Kế Website Quản lý bán hàng**

**Nhóm 7 \_ 20CN5**

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

**Nhóm Sinh viên thực hiện:**

1. **Hà Minh Hưng:2055010132**
2. **Hoàng Tuấn Vũ:2055010293**
3. **Phạm Tuấn Anh:2055010**

**Hà Nội \_2023**

**Lớp : 20VL**

**Giảng viên giảng dạy:**

Giảng viên giảng dạy : Lương Thị Bích Phượng

HÀ NỘI, 10/2023

# 1. Giới thiệu về HTML

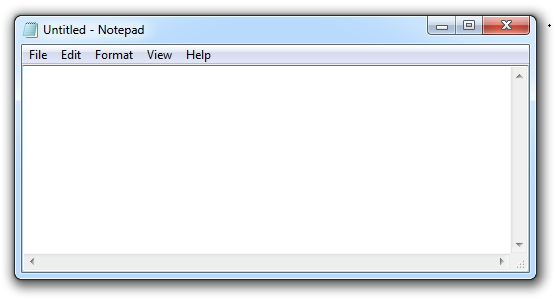
**a. Khái niệm HTML**

- HTML (Hypertext Markup Language) là ngôn ngữ đánh dấu siêu văn bản được sử dụng để tạo và cấu trúc các thành phần trong trang web hoặc ứng dụng, phân chia các đoạn văn, heading, titles, blockquotes.v.v.

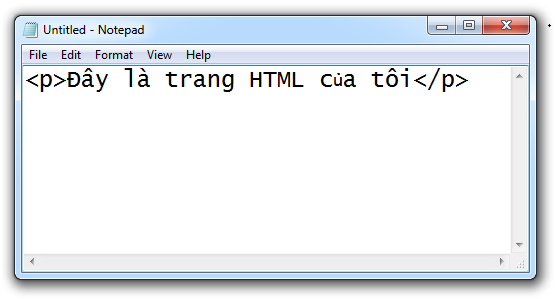
- HTML không phải là một ngôn ngữ lập trình, vì vậy không thể tạo ra các chức năng “động” được. Mục đích sử dụng tương tự như Microsoft Word, chỉ dùng để xây dựng bố cục và định dạng văn bản.

## 1.1. Hướng dẫn tạo tệp HTML cơ bản

- Bước 1:  Mở trình soạn thảo thuần văn bản Notepad lên.

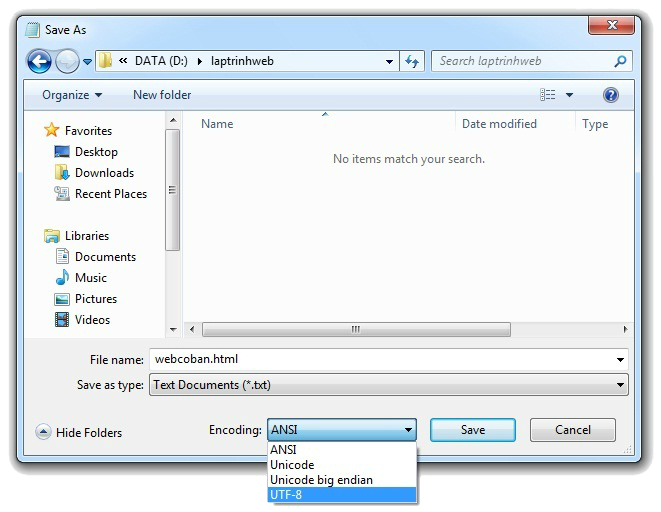


- Bước 2: Nhập những nội dung mà các bạn muốn chúng được hiển thị trên trang web.



- Bước 3: Lưu tập tin.

1. Bấm vào nút **File** rồi chọn **Save As**.
2. Chọn thư mục muốn lưu tập tin.
3. Đặt tên cho tập tin.
4. Chỗ Encoding chọn **UTF-8**.
5. Cuối cùng, bấm nút **Save** để lưu lại.



## 1.2. Tạo trang web với tiêu đề ,đoạn văn ,hình ảnh và siêu liên kết

**\* Tiêu đề:**

-Thẻ <title>: dùng để xác định tiêu đề tab trình duyệt cho trang web và được đặt bên trong phần tử <head>.

-Thẻ tiêu đề (Heading): được xác định bằng thẻ đánh dấu từ <h1> tới <h6>, trong đó <h1> là tiêu đề quan trọng nhất còn <h6> là ít quan trọng nhất.

**\* Đoạn văn bản:**

- Thẻ văn bản <p> : được dùng để thêm đoạn văn cho trang.

**\* Hình ảnh:**

- Thẻ hình ảnh <img>: là một thẻ trống và chỉ có thể chứa danh sách các thuộc tính và không có thẻ đóng.

- Xác định rõ tên file ảnh trong thuộc tính src.

- Thuộc tính alt là một thuộc tính ủy quyền, nếu hình ảnh không hiển thị, nó sẽ xác định một văn bản thay thế cho hình ảnh đó.

- Cấu trúc: <img src="đường dẫn url chứa hình ảnh" ... alt=" "/>

**\* Siêu liên kết: chứa liên kết điều hướng**

- Khái niệm:Siêu liên kết cho phép liên kết tài liệu với các tài liệu hoặc tài nguyên khác, liên kết đến các phần cụ thể của tài liệu hoặc cung cấp ứng dụng tại một địa chỉ web. Hầu như bất kỳ nội dung web nào cũng có thể được chuyển đổi thành một liên kết để khi được nhấp hoặc kích hoạt, trình duyệt web sẽ chuyển đến một địa chỉ web khác.

- Cấu trúc: <a href="duong-dan-lien-ket.html">Siêu liên kết của bạn</a>

=> Tóm lại, ví dụ trang web với tiêu đề, đoạn văn, hình ảnh và siêu liên kết:

## 1.3. Hướng dẫn về cú pháp và các phần tử HTML cơ bản

#### -Comment

**Comment** trong HTML hay bất kì ngôn ngữ lập trình nào đều rất quan trọng, giúp code dễ hiểu, dễ bảo trì và cập nhật

Cú pháp Comment trong HTML

**<!---**Nội dung comment**-->**

**Ví dụ:**

<!--Chúng ta cùng nhau code HTML -->

#### Cấu trúc của file HTML cơ bản

**<!DOCTYPE html>**

<html>

**<head>**

<title>Howkteam.com</title>

**</head>**

**<body>**

**</body>**

</html>

Cấu trúc chung của html gồm có thẻ **<!DOCTYPE html>** dùng để xác định rằng code bạn đang dùng là html

* Cặp thẻ đóng mở **<html> </html>** chứa toàn bộ các dòng lệnh html bên trong (sau này còn là code của Javascript và CSS khi chèn vào)
* Tiếp đến là 2 cặp thẻ **<head></head>**chứa thông tin của webpage và đặc trưng bởi thẻ**<title></title>**, mà mặc định sẽ không được hiển thị trên web
* Thẻ **<body></body>** chứa toàn bộ nội dung html sẽ hiển thị lên trên trang web của chúng ta.

#### -Các thẻ thường gặp trong HTML

Chúng ta tiếp tục với các thẻ thường gặp trong html

##### Thẻ tiêu đề

**<h1>** Nội dung tiêu đề **</h1>**

Tương tự với các thẻ **h2**,**h3**,**h4**,**h5,h6** với cấp độ tiêu đề giảm dần

**Ví dụ:**

<h1>Howkteam.com</h1>

<h2>Kteam </h2>

<h3>Free Education</h3>

<h4>Share To Be Better</h4>

<h5>Khóa học lập trình Front End</h5>

<h6>Project Landing Page</h6>

##### Thẻ định dạng đoạn văn bản

**<p>** Nội dung đoạn văn bản chứa bên trong **</p>**

##### Các thẻ mà tạo ra hiệu ứng với chữ

Ngoài các thẻ trên, HTML còn có các thẻ tạo hiệu ứng với chữ, đoạn văn bản như **<strong></strong>**, **<em></em>**, **<b></b>**, **<u> </u>**, **<strike></strike**>,….

Và một số thẻ khác như **<br>** để cách xuống dòng, **<hr>** để kẻ đường ngăn cách ngang,…

**Ví dụ:**

<h1>HOWKTEAM BLOG</h1>

<h2>HOWKTEAM</h2>

<!--heading tieu de cua cac phan trong website-->

<p>

<strike>Hello Kter</strike> <!--tao chu co gach ngang-->

<br> <!--break co tac dung giup xuong dong-->

<hr>

<b>HOWKTEAM</b> <!--bold tao chu in dam-->

<em>HOWKTEAM</em> <!-- Emphasize tao chu in nghieng-->

<u>HOWKTEAM</u> <!--underline tao chu co gach chan-->

Lorem, ipsum dolor sit amet consectetur adipisicing elit.

Rem debitis eos dolorem itaque tempore libero ipsa quibusdam fuga, vel,

molestiae maxime harum consequatur architecto repellat dolores cupiditate iure nisi fugiat?

Lorem ipsum, dolor sit amet consect <br>etur adipisicing elit. Nulla, culpa.

</p>

<!-- Noi ban chua van ban-->

**2. Cơ bản về CSS**

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

1. Để tạo mẫu, trong trình chỉnh sửa CSS yêu thích của bạn, hãy tạo tệp chủ đề tùy chỉnh **button\_shadow.css** và lưu lại.

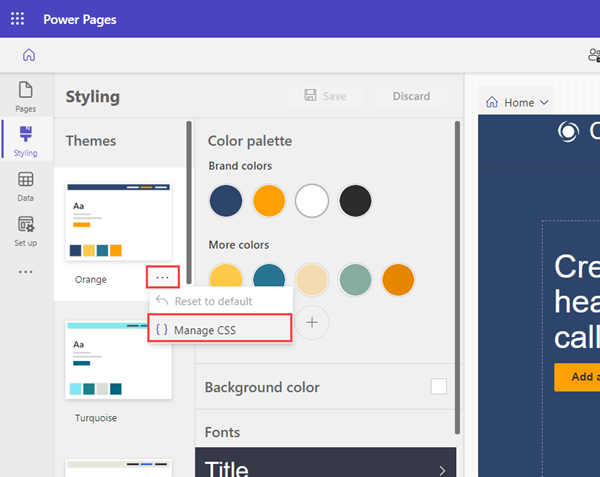
cssSao chép

.button1 {

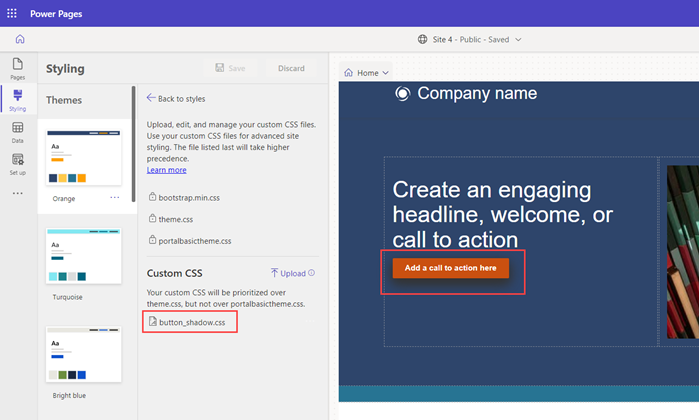
box-shadow: 0 9px 18px 0 #333333, 0 8px 24px 0 #333333;

}

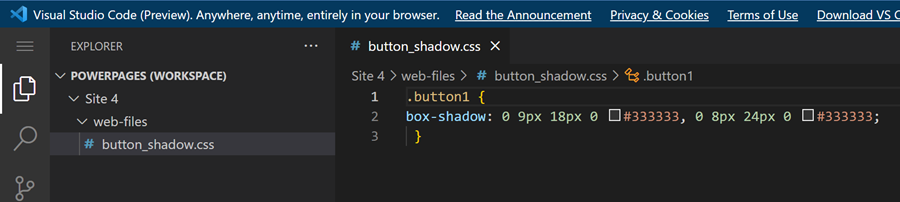
1. Đi tới [Power Pages](https://make.powerpages.microsoft.com/).
2. Chọn trang web mà bạn muốn thêm chủ đề tùy chỉnh và chọn **Chỉnh sửa**.
3. Mở không gian làm việc **Tạo kiểu** .
4. Chọn một chủ đề và chọn **...** (dấu chấm lửng), sau đó chọn **Quản lý CSS**.



1. Trong phần **Tùy chỉnh CSS** , hãy chọn **Tải lên** và chọn tệp CSS tùy chỉnh của bạn. Bạn chỉ có thể tải lên một CSS tệp mỗi lần nhưng có thể tải lên nhiều tệp. Nếu nhiều tệp CSS cập nhật cùng một thuộc tính, thì các thuộc tính trong tệp CSS ở cuối danh sách sẽ được áp dụng. Bạn có thể điều chỉnh thứ tự các tệp CSS tùy chỉnh của mình.
2. Bạn sẽ thấy ngay kết quả cập nhật trên trang canvas.



1. Bạn có thể trực tiếp chỉnh sửa tệp CSS bằng cách chọn dấu chấm lửng (**...**) rồi chọn **Chỉnh sửa mã**. Bước này sẽ mở trình chỉnh sửaMã trực quan cho Web [. Chọn](https://learn.microsoft.com/vi-VN/power-pages/configure/visual-studio-code-editor) CTRL-S **để lưu các thay đổi của bạn.**



1. **Chọn Visual Studio Đồng bộ hóa trong studio thiết kế để cập nhật và xem các thay đổi.**
2. **Chọn CSS Xem trước để xem chủ đề tùy chỉnh trên trang web của bạn.**

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

#### Ở bài hướng dẫn này, bạn sẽ được tìm hiểu một số thuộc tính định dạng chữ như sau:

* font-size - thiết lập kích cỡ chữ
* font-weight - thiết lập sự tô đậm của chữ
* font-style - thiết lập sự in nghiêng của chữ
* font-family - thiết lập kiểu chữ

## Thiết lập KÍCH CỠ CHỮ

Thuộc tính font-size dùng để thiết lập kích cỡ chữ của văn bản

Ví dụ:

p {

    font-size:40px;

}

[Xem ví dụ](https://webcoban.vn/css/demo?file=74)

Giá trị của thuộc tính font-size có thể xác định theo đơn vị:

* px [[Xem ví dụ]](https://webcoban.vn/css/demo?file=74)
* em (tỷ lệ 1em = 16px) [[Xem ví dụ]](https://webcoban.vn/css/demo?file=684)
* % (tỷ lệ phần trăm so với kích cỡ chữ của phần tử chứa nó) [[Xem ví dụ]](https://webcoban.vn/css/demo?file=685)

Lưu ý: Nếu không thiết lập kích cỡ chữ cho văn bản thì mặc định văn bản có kích cỡ chữ là 16px

## Thiết lập CHỮ TÔ ĐẬM

Thuộc tính font-weight dùng để thiết lập sự tô đậm của chữ.

Thuộc tính font-weight có hai giá trị cơ bản:

* normal: không tô đậm
* bold: tô đậm

Ví dụ:

p {

    font-weight: bold;

}

[Xem ví dụ](https://webcoban.vn/css/demo?file=73)

## Thiết lập CHỮ IN NGHIÊNG

Thuộc tính font-style dùng để thiết lập sự in nghiêng của chữ.

Thuộc tính font-style có ba giá trị cơ bản:

* normal: không in nghiêng
* italic: in nghiêng
* oblique: in nghiêng (giống italic, nhưng độ in nghiêng ít hơn)

Ví dụ:

p {

    font-style: italic;

}

[Xem ví dụ](https://webcoban.vn/css/demo?file=72)

## Thiết lập KIỂU CHỮ

### Khái niệm font chữ & nhóm font chữ

Font chữ chính là kiểu chữ, mẫu mã của chữ. Ví dụ, một số font chữ quen thuộc như: Times New Roman, Arial, Lucida Console,....

Các font chữ có cùng một vài nét tương đồng nào đó về hình dáng sẽ được xếp chung một nhóm.

Có ba nhóm font chữ cơ bản là:

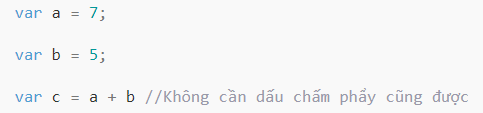
* Serif
* Sans-Serif
* Monospace

# 3. JavaScript Cơ bản

## 3.1. Giới thiệu 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.

- Các câu lệnh Javascript không cần phải đặt trong dấu ngoặc đơn, hay ngoặc nhọn và cuối mỗi câu lệnh cũng không cần dấu chấm phẩy.

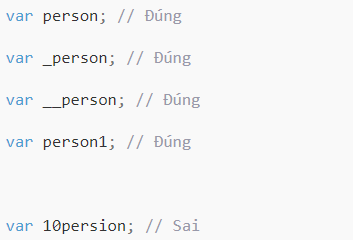


**\* Cách đặt tên biến, hàm**

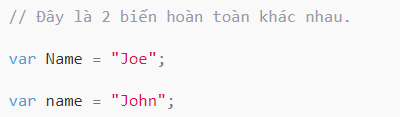
- Trong Javascript, định danh (identifiers) được sử dụng để đặt tên biến, từ khóa, hàm và labels.

- Việc định danh nói chung là theo quy tắc sau:

* Ký tự đầu tiên của định danh phải là chữ cái ( a-z, A-Z) hoặc ký tự $ hoặc dấu gạch dưới (\_)
* Không được sử dụng chữ số làm ký tự đầu tiên của định danh.



- Các định danh có phân biệt chữ hoa với chữ thường.



- Về quy tắc đặt tên các định danh điển hình có 3 loại:

* Dùng dấu gạch ngang: Ví dụ: first-name
* Dùng dấu gạch dưới. Ví dụ: first\_name
* Và trường phái **camel case**, tức là viết chữ in hoa với mỗi từ đơn. Ví dụ: FirstName

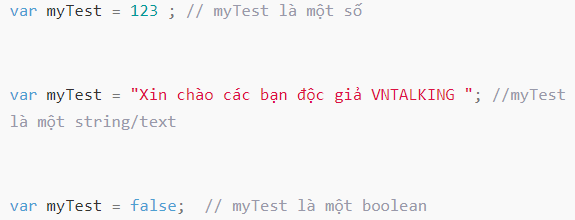
**\* Cách khai báo biến**

- 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**

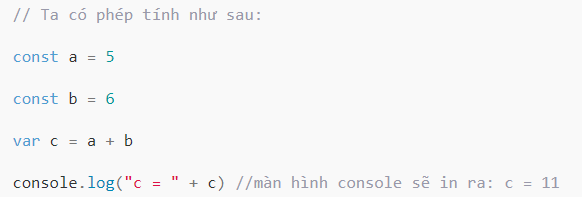
- Kiểu dữ liệu sẽ được xác định một cách tự động khi chương trình thực thi. Điều này có nghĩa là một biến có thể có nhiều kiểu dữ liệu ở những thời điểm khác nhau.



- 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**



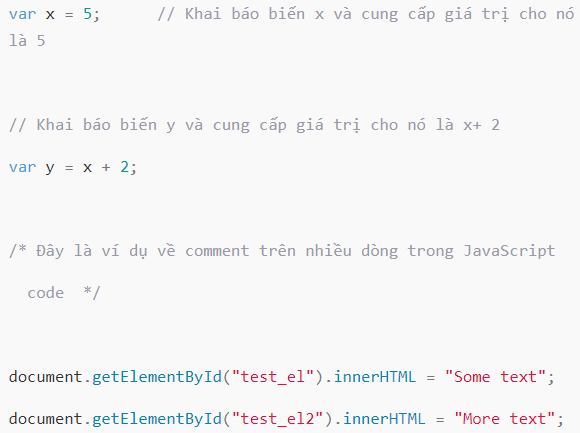
- Trong ví dụ trên, phép tính a + b, thì phép tính cộng được gọi là toán tử (operator).

- Javascript hỗ trợ đầy đủ các loại toán tử tính toán và kết hợp.

- Bảng các toán tử được hỗ trợ:

**\* Comment trong Javascript**

- Javascript hỗ trợ 2 kiểu comment: **Single Line Comments** (comment một dòng) và **Multi-line Comments** (comment nhiều dòng).





- Chú ý:

* Nên Comment để giải thích “**Tại sao**” lại viết code đó thay vì **đoạn code làm như thế nào**. Vì người đọc code là đủ hiểu code làm như nào thế nào rồi.
* Không comment những đoạn code mà đọc code đó đã quá rõ ràng rồi.
* Không comment để lấp liếm vấn đề của đoạn code. Tức là người code sau cần phải đọc comment sử dụng đúng đoạn code nếu không sẽ bị lỗi. Thay vì đó, bạn nên viết code lại cho chặt chẽ hơn.
* Xóa ngay comment khi đoạn code đi kèm đã bị xóa để tránh dư thừa comment.

## 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

**var** varName1 = 10; // khai báo và khởi tạo biến với từ khóa var

**let** varName2; // khai báo và không khởi tạo giá trị ban đầu

varName2 = 10; // gán giá trị

**const** varName3 = 10; // khai báo và khởi tạo biến với từ khóa const

**b. Tạo hàm JavaScript**

// Khai báo hàm

function isEven(num) {

return num % 2 === 0;

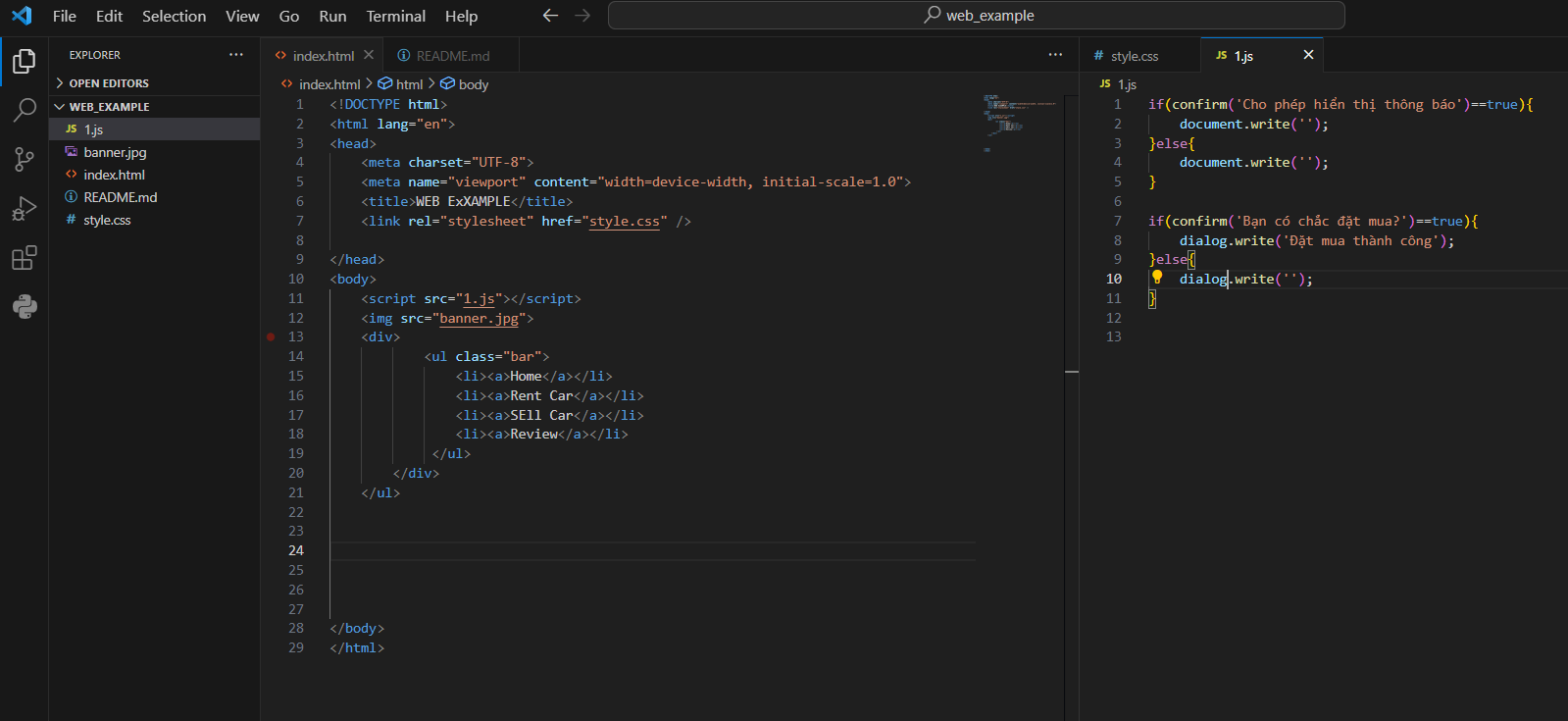
}

isEven(24); // => true

isEven(11); // => false

* Khai báo tạo một hàm trong phạm vi hiện tại với mã định nghĩa bằng hàm tên. Biến này chứa hàm object.
* Đồng thời, biến này sẽ được **nâng lên** trong phạm vi hiện tại, nhờ đó ta có thể gọi và sử dụng hàm trước khi khai báo.
* 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…

- Phần tử input có thể được hiển thị bằng nhiều cách khác nhau, tùy thuộc vào thuộc tính type.

**\* Trường nhập nút radio:**

- <input type="radio"> - Xác định một nút radio.

- Các nút radio cho phép người dùng chọn MỘT trong số các lựa chọn giới hạn.

**\* Nút Submit (Gửi đi):**

- <input type="submit"> - xác định một nút để gửi dữ liệu biểu mẫu tới bộ xử lý biểu mẫu.

- Bộ xử lý biểu mẫu thường là một trang server với script để xử lý dữ liệu đầu vào.

- Bộ xử lý biểu mẫu được xác định trong thuộc tính action của form.

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

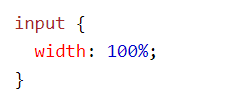
**a. Khái niệm**

- 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> :



+ Nếu chỉ muốn tạo kiểu cho một loại đầu vào cụ thể thì sử dụng bộ chọn thuộc tính sau:

* input[type=text] – Sẽ chỉ chọn các trường văn bản
* input[type=password] – Sẽ chỉ chọn các trường mật khẩu
* input[type=number] – Sẽ chỉ chọn các trường số
* v.v.v…

- 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.

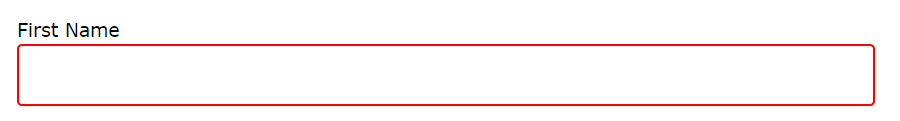
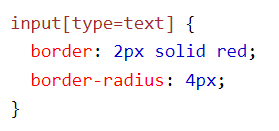
+ Đặt thuộc tính box-sizing là border-box để đảm bảo phần padding và border cuối cùng được đặt trong tổng chiều rộng và chiều cao của các phần tử.



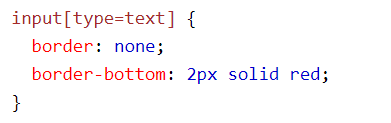
- Định dạng đường viền cho trường đầu vào:

+ 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

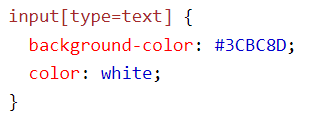


+ Nếu chỉ muốn thêm đường viền dưới cùng thì sử dụng thuộc tính *border-bottom:*



- Định dạng màu cho trường đầu vào:

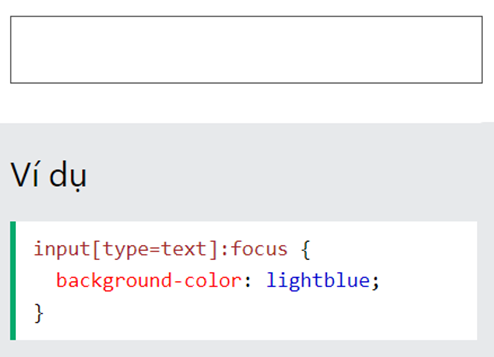
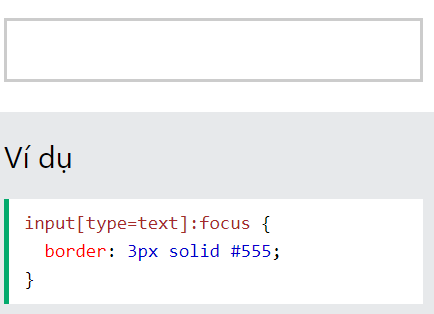
+ Sử dụng thuộc tính *background-color* để thêm màu nền cho trường đầu vào, và thuộc tính *color* thay đổi màu văn bản:



- Tạo ấn tượng cho trường đầu vào:

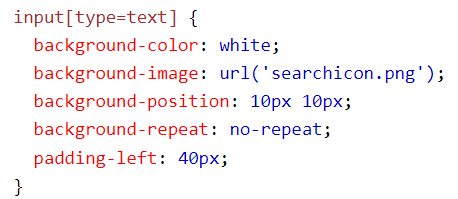
+ Một số trình duyệt mặc định sẽ thêm đường viền màu xanh xung quanh trường đầu vào khi người dùng click chuột để nhập thông tin và có thể xóa trạng thái này bằng cách nhập *outline: none;* cho trường đầu vào.

+ Sử dụng bộ chọn *:focus* để thực hiện điều gì đó với trường đầu vào khi click chuột để nhập thông tin:

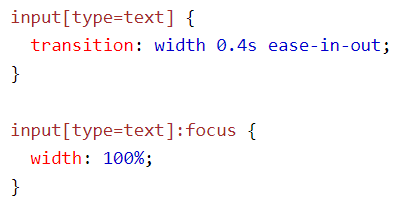
- Thêm biểu tượng/ảnh vào bên trong trường đầu vào:

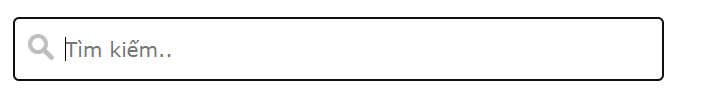
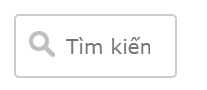
+ Sử dụng thuộc tính *background-image* và thuộc tính vị trí *background-position* để thêm icon vào trong trường đầu vào. Chú ý: thêm *padding-left* để dành khoảng trống cho biểu tượng:



- Thêm hiệu ứng tìm kiếm cho trường đầu vào:

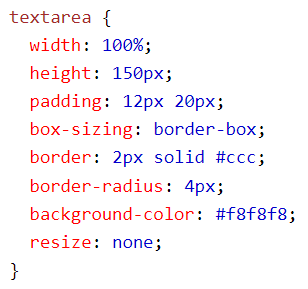
+ Sử dụng thuộc tính *transition* trong CSS để tạo hiệu ứng chiều rộng cho trường tìm kiếm khi người dùng click chuột vào trường nhập văn bản.



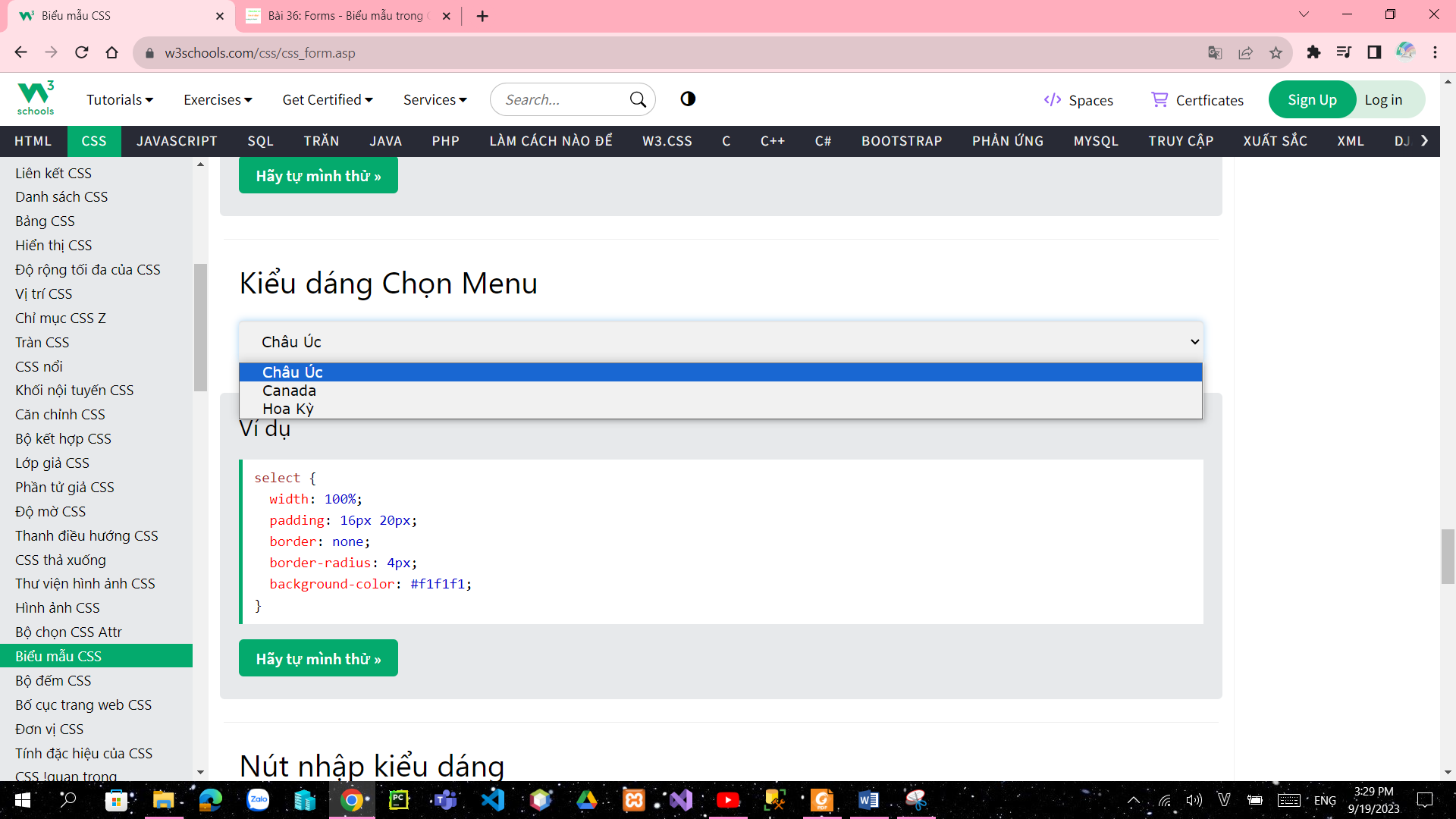
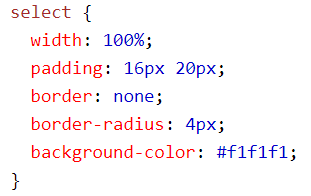


- Tạo kiểu cho Textareas:

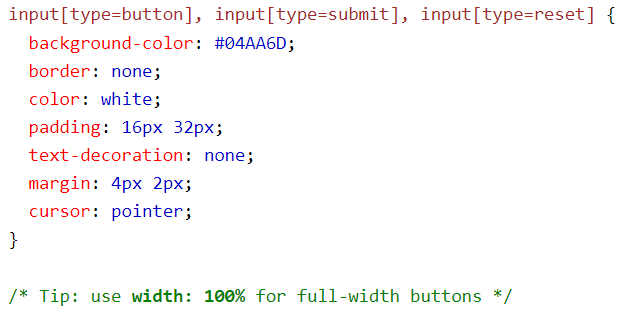
+ Sử dụng thuộc tính *resize* để ngăn các vùng văn bản bị thay đổi kích thước ( tắt “grabber” ở góc dưới cùng bên phải):

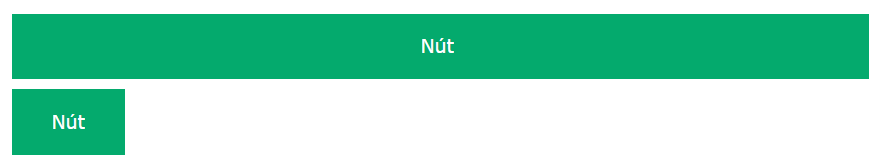


- Định dạng menu chọn:



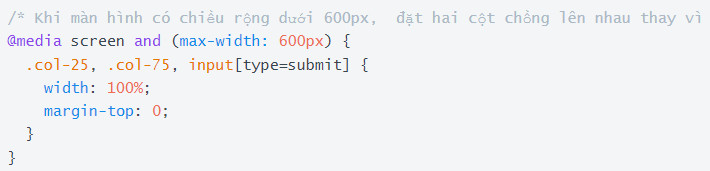
- Định dạng button:





- Responsive Form:

Thay đổi kích thước cửa sổ trình duyệt để xem hiệu ứng. Khi màn hình có chiều rộng dưới 600px, đặt hai cột chồng lên nhau thay vì cạnh nhau.



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

### ***4.3.1. 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>

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

+ Dùng hàm getElementById : documnet.getElementIdById(‘idcủaControl’).value;

+ Dùng hàm 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>

### 4.3.3. 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>

***\*DEMO KIỂM THỬ :***

<script>

function kiemtra(){

var u = document.frmlogin.u.value;

var p = document.frmlogin.p.value;

if (u=="") { *//quy tắc 1* document.getElementById("baoloi").innerHTML="Chưa nhập username mà";

**return false;** }

if (p=="") { **//quy tắc 2** document.getElementById("baoloi").innerHTML="Chưa nhập password mà";

**return false;** }

**return true;** }

</script>

# 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 :**

## 1. Tại sao phải thiết kế website?

Thiết kế website đang dần trở thành nhu cầu không thể thiếu với bất kỳ doanh nghiệp nào. Lợi ích của việc thiết kế website đối với doanh nghiệp là không thể phủ nhận. Dưới đây là một số lợi ích nổi bật mà website mang lại cho doanh nghiệp

### 2. Đưa khách hàng đến gần hơn với sản phẩm/dịch vụ của doanh nghiệp

Hiện nay, khi internet đã quá phổ biến thì website đã trở thành một**kênh quảng bá sản phẩm/dịch vụ** vô cùng hữu ích cho các doanh nghiệp. Thông qua website, thông tin chi tiết về sản phẩm/dịch vụ sẽ được cung cấp cho khách hàng một cách cụ thể, trực quan nhất.

Khi được tương tác trực tiếp với những thông tin sinh động, khách hàng sẽ có cái nhìn chính xác hơn về sản phẩm/dịch vụ mà doanh nghiệp cung cấp. Từ đó, khách hàng sẽ hình thành được ấn tượng tốt với thương hiệu của doanh nghiệp.

### 3. Tăng lượng traffic

Doanh nghiệp sẽ dễ dàng tăng thu nhập data khách hàng khi tiến hành **các bước làm website** bán hàng một cách tối ưu và hiệu quả.

### 4. Gia tăng tỷ lệ chuyển đổi, tỷ lệ mua hàng

Việc tìm hiểu **các bước xây dựng website** bán hàng trực tuyến và [thiết kế web bán hàng chuẩn seo](https://mikotech.vn/thiet-ke-website-ban-hang/#web_b%C3%A1n_h%C3%A0ng) hiệu quả sẽ góp công lớn để gia tăng tỷ lệ mua hàng cho doanh nghiệp của bạn.

Nếu website của bạn trở thành một kênh thương mại điện tử phổ biến đối với khách hàng, họ sẽ ngày càng tin tưởng và yêu mến hơn doanh nghiệp của bạn hơn. Nhờ đó doanh nghiệp có thể gia tăng tỷ lệ chuyển đổi mua hàng của người dùng truy cập trang web.

### 5.Xây dựng hình ảnh chuyên nghiệp, đáng tin cậy

Hiện nay, khách hàng thường có thói quen tìm hiểu thông tin về doanh nghiệp, sản phẩm trước khi quyết định mua hàng. Trong đó, website là kênh tra cứu thông tin phổ biến thường được khách hàng tìm đến và sử dụng.

Chính vì vậy, nếu doanh nghiệp muốn tạo được hình ảnh chuyên nghiệp đối với khách hàng thì thiết kế website là điều cần thiết. Nhất là các ngành nghề yêu cầu cao về mặt thẩm mỹ như [**thiết kế web nội thất**](https://mikotech.vn/thiet-ke-website-noi-that/)**,**[**thiết kế website du lịch chuyên nghiệp**](https://mikotech.vn/thiet-ke-website-du-lich/), bệnh viện, chuyên gia,…

Một website có giao diện chỉn chu, đẹp mắt, thân thiện sẽ làm cho khách hàng thêm tin cậy và quyết định mua sản phẩm của doanh nghiệp.

## 6.Tiêu chí khi thiết kế website

Miko Tech Agency là đơn vị **thiết kế website Hồ Chí Minh** uy tín và chuyên nghiệp hàng đầu. Sau nhiều năm hoạt động trên thị trường, chúng tôi đã hợp tác làm việc với nhiều cá nhân, tổ chức cả trong và ngoài nước.

Miko Tech hỗ trợ [**thiết kế website**](https://mikotech.vn/thiet-ke-website/#b%C3%A1o_gi%C3%A1_d%E1%BB%8Bch_v%E1%BB%A5_thi%E1%BA%BFt_k%E1%BA%BF_website)đáp ứng yêu cầu của khách hàng và phù hợp với sự phát triển của thời đại công nghệ hiện nay. Hãy cùng Miko Tech tìm hiểu 3 tiêu chí để đánh giá một thiết kế website tốt:

### 7.Giao diện website đẹp – thu hút

**Giao diện website**ảnh hưởng trực tiếp đến cảm nhận và quyết định của khách hàng, thật ra bạn chỉ cần một giao diện dễ nhìn và gọn gàng là đủ. Tuy nhiên, thiết kế một giao diện theo kịp xu hướng hiện tại cũng sẽ giúp website của bạn trở nên nổi bật và thu hút hơn.

#### Nội dung phải đảm bảo tính chính xác, hữu ích với người xem

Nội dung website cần phải được **cập nhật đầy đủ, đều đặn**, thu thập ý tưởng hay khác và biến tấu sao cho phù hợp với website của mình. Tìm hiểu và phân tích kỹ insight khách hàng theo từng giai đoạn để có những nội dung hợp lý, thu hút.

Nội dung trên website có độ chính xác càng cao, thì tỷ lệ người truy cập và tin tưởng trang web của bạn cũng càng nhiều hơn.

**Ví dụ** như khi WED THIẾT KẾ BÁN HÀNG, bạn cần thể hiện được trên website những thông tin cần thiết như thông số, công nghệ,… Thông tin mới được cập nhật nhanh nhất khi hãng có dòng xe mới ra mắt. Nội dung thu hút đánh trúng insight khách hàng, sản phẩm mang lại cảm giác sang trọng hay cổ điển,…

#### Hình thức nội dung trên website cần phải sinh động, phù hợp

Các nội dung bằng văn bản và hình ảnh trên website hiện nay rất phổ biến. Bởi vì nội dung sẽ được truyền tải thông điệp tốt hơn thông qua văn bản và sự hỗ trợ của hình ảnh.

Tuy nhiên, nếu muốn ấn tượng hơn, bạn nên biết cách xây dựng các đoạn video ngắn, có nội dung cụ thể hoặc sử dụng công nghệ đồ họa tạo ra những cách truyền tải thông điệp độc đáo hơn như animation, infographic,v.v.

#### Tính khác biệt quyết định sự lựa chọn của người dùng Internet

Có nội dung chính xác và hữu ích, hình ảnh đẹp với cách trình bày hợp lý liệu đã đủ để khiến khách hàng hay người xem lựa chọn website của bạn? Câu trả lời vẫn là chưa đủ.

## 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.

## 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.

### 5.3.1. Quản lý dự án

- Trưởng nhóm: Đóng vai trò lãnh đạo, phân phối công việc, quản lý công việc và thiết lập các mốc thời gian quan trọng cho dự án.

- Quản lý dự án: Chịu trách nhiệm chủ động lập kế hoạch, định hướng và giám sát toàn bộ quá trình dự án, bao gồm quản lý nguồn lực, theo dõi tiến độ, và đảm bảo việc hoàn thành dự án đúng theo yêu cầu và thời gian.

- Quản lý chất lượng: Đảm bảo chất lượng của công việc được thực hiện đúng theo yêu cầu khách hàng và tiêu chuẩn đã đề ra trong dự án.

### 5.3.2. Phát triển

- Kiến trúc sư: Thiết kế, phân tích và xác định kiến trúc và cấu trúc tổ chức cho dự án. Kiến trúc sư tạo ra các sơ đồ, bản thiết kế và hướng dẫn cho các thành viên khác để phát triển sản phẩm, viết đặc tả, lựa chọn công nghệ, ngôn ngữ lập trình phù hợp cho dự án.

- Nhà phát triển: Làm Bố cục trang web - một bản phác họa đồ họa hoặc một thiết kế đồ họa thực tế. Chức năng chính của bố cục là thể hiện cấu trúc thông tin, trực quan hóa nội dung và các chức năng cơ bản. Nhà phát triển là người thực hiện trực tiếp công việc lập trình và đảm bảo mã nguồn được viết theo tiêu chuẩn và đúng mục tiêu của dự án.

- Kiểm thử viên: 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. Sau khi đã kiểm thử xong, ta sẽ triển khai trang web lên máy chủ (server). Sau khi triển khai xong mã nguồn lên máy chủ, chúng ta nên chạy thêm một bài kiểm tra cuối cùng để đảm bảo rằng các mã nguồn đã được cài đặt chính xác.