**TRƯỜNG ĐẠI HỌC ĐẠI NAM**

**KHOA CÔNG NGHỆ THÔNG TIN**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

**TÊN MÔN HỌC: THIẾT KẾ WEB VÀ TRIỂN KHAI HỆ THỐNG PHẦN MỀM**

**TÊN ĐỀ TÀI: Thiết Kế Wbsite giới Thiệu Địa Điểm Du lịch**

**Giảng viên hướng dẫn: ThS. Trần Thị Huệ**

**Sinh viên thực hiện: Ngô Khánh Tùng**

**Hà Nội, 2024**

**TRƯỜNG ĐẠI HỌC ĐẠI NAM**

**KHOA CÔNG NGHỆ THÔNG TIN**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

**TÊN MÔN HỌC: THIẾT KẾ WEB VÀ TRIỂN KHAI HỆ THỐNG PHẦN MỀM**

**TÊN ĐỀ TÀI: Thiết Kế wedsite giới thiệu địa điểm du lịch**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Mã Sinh Viên | Họ và Tên | Ngày Sinh | Điểm | |
| Bằng Số | Bằng Chữ |
| 01 | 1771020725 | Ngô Khánh Tùng | 04/12/2005 |  |  |

### 

|  |  |
| --- | --- |
| CÁN BỘ CHẤM THI 1 | CÁN BỘ CHẤM THI 2 |
|
| Trần Thị Huệ |  |

**Hà Nội, 2024**

**LỜI NÓI ĐẦU**

Internet ngày nay đóng vai trò vô cùng quan trọng trong mọi lĩnh vực của đời sống, bao gồm cả việc quảng bá và thu hút khách hàng. Thiết kế web là một phần không thể thiếu trong việc tạo dựng hình ảnh chuyên nghiệp và thu hút khách hàng tiềm năng cho doanh nghiệp của bạn.

Bài viết này sẽ giới thiệu về:

Thiết kế web: Khái niệm, tầm quan trọng và các nguyên tắc cơ bản trong thiết kế web.

HTML và CSS: Hai ngôn ngữ lập trình cơ bản để xây dựng website.

Cách áp dụng HTML và CSS vào thiết kế web: Ví dụ minh họa và hướng dẫn thực hành.

Mục tiêu của bài viết này là:

Giúp người đọc hiểu rõ hơn về tầm quan trọng và các yếu tố cần thiết để thiết kế một website hiệu quả.

Cung cấp cho người đọc kiến thức cơ bản về HTML và CSS để có thể tự tạo hoặc chỉnh sửa website của mình.

Hướng dẫn người đọc cách áp dụng HTML và CSS vào thiết kế web một cách thực tế.

Bài viết này được viết bởi một đội ngũ giàu kinh nghiệm trong lĩnh vực thiết kế web. Chúng tôi hy vọng bài viết này sẽ cung cấp cho bạn những thông tin hữu ích và cần thiết để bắt đầu hành trình thiết kế web của riêng bạn.

**Vẻ đẹp hùng vĩ của Hà Giang** luôn là điểm đến mơ ước của biết bao du khách. Để quảng bá hình ảnh du lịch Hà Giang đến với du khách trong và ngoài nước, việc thiết kế một website du lịch Hà Giang chuyên nghiệp và thu hút là vô cùng quan trọng.

MỤC LỤC

Contents

[**CHƯƠNG I: GIỚI THIỆU VỀ THIẾT KẾ WEB VÀ TRIỂN KHAI HỆ THỐNG PHẦN MỀM** 11](#_Toc160958820)

[**1.1.** **Các khái niệm cơ bản** 11](#_Toc160958821)

[**1.2.Phân biệt Web tĩnh, Web động** 14](#_Toc160958822)

[**1.3. Một số thuật ngữ (Hosting, Tên miền-Domain Name-Web Server Name, HomePage,** 15](#_Toc160958823)

[**Web Site, WebServer, URL-Uniform Resource Location, Browser, …)** 15](#_Toc160958824)

[**1.4. Một số công cụ dùng thiết kế Web** 17](#_Toc160958825)

[**Kết luận chương 1** 20](#_Toc160958826)

[**CHƯƠNG II: HTML & HTML5 (Hyper Text Markup Language)** 21](#_Toc160958827)

[**2.1. Tổng quan về HTML** 21](#_Toc160958828)

[**2.2. Cấu trúc tổng quát trang HTML** 22](#_Toc160958829)

[**2.3. Các thẻ HTML thông dụng** 23](#_Toc160958830)

[***-Các thẻ xử lý văn bản*** 23](#_Toc160958831)

[***-Các thẻ tạo bảng*** 23](#_Toc160958832)

[***-Thẻ liên kết*** 24](#_Toc160958833)

[***Các thẻ đa phương tiện*** 24](#_Toc160958834)

[***-Thẻ tạo khung*** 25](#_Toc160958835)

[**2.4.Các thẻ tạo biểu mẫu (form)** 25](#_Toc160958836)

[**2.5.Một số thẻ HTML đặc biệt** 26](#_Toc160958837)

[**2.6. HTML5** 28](#_Toc160958839)

[***-HTML5 là gì?*** 28](#_Toc160958840)

[***-Cú pháp HTML5*** 29](#_Toc160958841)

[***-Khai báo DOCTYPE trong HTML5*** 30](#_Toc160958842)

[***-Thẻ ngữ nghĩa*** Thẻ ngữ nghĩa trong HTML5 32](#_Toc160958843)

[***-Webform*** 33](#_Toc160958845)

[**Kết luận chương 2** 34](#_Toc160958846)

[**CHƯƠNG III: CSS và CSS3Cascading Style Sheets)** 35](#_Toc160958847)

[**3.1. CSS là gì?** 35](#_Toc160958848)

[**3.2. Cú pháp CSS** 35](#_Toc160958849)

[**3.3. Áp dụng CSS vào trang HTML** 37](#_Toc160958850)

[**3.4. Selectors** 37](#_Toc160958851)

[***3.4.1.Universal selector*** 39](#_Toc160958852)

[***3.4.2. Type selector*** 40](#_Toc160958854)

[***3.4.3. Identity selector*** 42](#_Toc160958855)

[***3.4.4. Class selector*** 44](#_Toc160958856)

[***3.4.5 Descendant selector*** Descendant selector là gì? 46](#_Toc160958857)

[**3.4.6. Child selector** 48](#_Toc160958858)

[***3.4.7 Adjacent sibling selector*** 50](#_Toc160958859)

[***3.4.8. Attribute selector*** 52](#_Toc160958860)

[***3.4.9. Pseudo class selector*** 55](#_Toc160958861)

[***3.4.10. Group selector*** 57](#_Toc160958862)

[**3.5. Đơn vị đo lường CSS** 59](#_Toc160958863)

[**3.6. Kế thừa thuộc tính** 60](#_Toc160958864)

[**3.7. Các nhóm thuộc tính trong CSS** 61](#_Toc160958865)

[***-Thuộc tính định dạng font chữ, văn bản (font, text)*** 61](#_Toc160958866)

[**-*Thuộc tính định dạng nền (background)*** 62](#_Toc160958867)

[**-*Mô hình hộp (box model)*** 62](#_Toc160958868)

[***-Các thuộc tính định margin, padding, border*** 62](#_Toc160958869)

[***-Thuộc tính định dạng cách hiển thị (display)*** 63](#_Toc160958870)

[***-Thuộc tính xác định vị trí (position)*** 63](#_Toc160958871)

[**3.8. Float & Clear** Float & Clear là gì? 63](#_Toc160958872)

[**3.9.Flex** 65](#_Toc160958873)

[**3.10. Grid** 67](#_Toc160958874)

[**3.11. CSS3** 70](#_Toc160958875)

[***-CSS3 là gì?*** 70](#_Toc160958876)

[***-CSS3 Selectors*** 71](#_Toc160958877)

[***-Universal Selector:*** 72](#_Toc160958878)

[***-CSS3 Rounded Corners*** 73](#_Toc160958879)

[**-*CSS3 Border Images*** 75](#_Toc160958880)

[**-*CSS3 Backgrounds*** 77](#_Toc160958881)

[***- CSS Gradients*** 80](#_Toc160958883)

[***-CSS3 Shadows*** 82](#_Toc160958885)

[**- *CSS3 Fonts*** 84](#_Toc160958886)

[***- CSS3 2D Transforms*** 85](#_Toc160958888)

[**-*CSS3 3D Transforms*** 87](#_Toc160958889)

[***-CSS3 Transitions*** 89](#_Toc160958890)

[***- CSS3 Multiple Columns*** 90](#_Toc160958891)

[***-CSS3 Animations*** 91](#_Toc160958892)

[**3.12. SCSS** 94](#_Toc160958894)

[**3.13. SASS** 95](#_Toc160958895)

[**Kết luận chương 3** 96](#_Toc160958896)

[**CHƯƠNG 4: THIẾT KẾ WEBISTE THEO ĐỀ TÀI BẠN CHỌN** 97](#_Toc160958897)

[**4.1. Ý tưởng của Website** 97](#_Toc160958898)

[**4.2. Xây dựng bố cục của trang Web** 97](#_Toc160958899)

[**4.3. Thiết kế trang Web bằng HTML và CSS** 97](#_Toc160958900)

[***4.3.1.Wed trang chủ bằng html và css*** 97](#_Toc160958901)

[***4.3.2. HTML CSS TP Hà Giang*** 102](#_Toc160958902)

[***4.3.3. HTML CSS Huyện Quản Bạ*** 107](#_Toc160958903)

[***4.3.4. HTML Đồng Văn*** 112](#_Toc160958904)

[4.3.5. HTML CSS Yên Minh 117](#_Toc160958906)

[***4.3.6.HTML Mèo Vạc*** 122](#_Toc160958907)

[***4.3.7.HTML CSS Bắc Mê*** 127](#_Toc160958908)

[**4.5. Kết quả đạt được** 131](#_Toc160958909)

[**Kết luận chương 4** 132](#_Toc160958910)

**BẢNG CÁC TỪ VIẾT TẮT**

**(Nếu có)**

|  |  |  |
| --- | --- | --- |
| **STT** | **TỪ VIẾT TẮT** | **VIẾT ĐẦY ĐỦ** |
| 1 | HTML | Hyper Text Markup Language |
| 2 | **CSS** | Cascading Style Sheets |

**CHƯƠNG I: GIỚI THIỆU VỀ THIẾT KẾ WEB VÀ TRIỂN KHAI HỆ THỐNG PHẦN MỀM**

* 1. **Các khái niệm cơ bản**

Thiết kế web: Là quá trình tạo ra giao diện và cấu trúc của trang web, bao gồm bố cục, hình ảnh, màu sắc, typography, và chức năng.



***Thiết kế web***

Triển khai hệ thống phần mềm: Là quá trình cài đặt, cấu hình và đưa hệ thống phần mềm vào hoạt động.

Triển khai hệ thống phần mềm

Website: Là tập hợp các trang web được liên kết với nhau, có chung tên miền và được lưu trữ trên một hoặc nhiều máy chủ web.

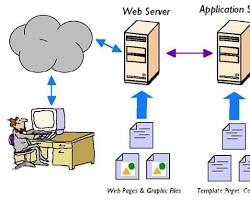


***Website***

Trang web: Là một tài liệu được lưu trữ trên máy chủ web, có thể truy cập được qua mạng internet bằng trình duyệt web.

Trang web

Máy chủ web: Là máy tính lưu trữ các trang web và cung cấp chúng cho người dùng khi có yêu cầu.



***Máy chủ web***

Tên miền: Là tên dễ nhớ được sử dụng để truy cập website thay cho địa chỉ IP của máy chủ web.

Tên miền

Hosting: Là dịch vụ cung cấp không gian lưu trữ cho website trên máy chủ web.

***Hosting***

Giao diện người dùng (UI): Là phần mà người dùng trực tiếp tương tác với hệ thống, bao gồm các yếu tố như bố cục, hình ảnh, màu sắc, typography, và chức năng.



***Giao diện người dùng (UI)***

Trải nghiệm người dùng (UX): Là cảm nhận của người dùng khi sử dụng hệ thống, bao gồm sự dễ sử dụng, hiệu quả và mức độ hài lòng.

Trải nghiệm người dùng (UX)

Một số thuật ngữ khác:

HTML: Ngôn ngữ lập trình được sử dụng để tạo ra cấu trúc của trang web.

CSS: Ngôn ngữ lập trình được sử dụng để định kiểu cho trang web, bao gồm màu sắc, phông chữ, và bố cục.

JavaScript: Ngôn ngữ lập trình được sử dụng để thêm chức năng cho trang web, chẳng hạn như tương tác với người dùng và xử lý dữ liệu.

PHP: Ngôn ngữ lập trình được sử dụng để tạo ra các trang web động.

ASP.NET: Ngôn ngữ lập trình được sử dụng để tạo ra các trang web động trên nền tảng Microsoft.

Ruby on Rails: Khung ứng dụng web được sử dụng để tạo ra các trang web động.

Kết luận

Thiết kế web và triển khai hệ thống phần mềm là hai lĩnh vực quan trọng trong công nghệ thông tin. Việc hiểu rõ các khái niệm cơ bản về hai lĩnh vực này sẽ giúp bạn có thể bắt đầu học tập và làm việc trong lĩnh vực này.

**1.2.Phân biệt Web tĩnh, Web động**

Web tĩnh:

Là website có nội dung được lưu trữ dưới dạng các tập tin HTML, CSS và JavaScript tĩnh.

Khi người dùng truy cập website, máy chủ web sẽ gửi các tập tin này đến trình duyệt web của họ.

Nội dung của website không thay đổi thường xuyên.

Ví dụ: website giới thiệu công ty, website portfolio cá nhân.

Web động:

Là website có nội dung được tạo ra động khi người dùng truy cập.

Máy chủ web sẽ sử dụng các ngôn ngữ lập trình như PHP, ASP.NET, hoặc Ruby on Rails để tạo ra nội dung dựa trên yêu cầu của người dùng.

Nội dung của website có thể thay đổi thường xuyên.

Ví dụ: website bán hàng trực tuyến, website tin tức, website mạng xã hội.

Bảng so sánh Web tĩnh và Web động:

|  |  |  |
| --- | --- | --- |
| Đặc điểm | Web tĩnh | Web động |
| Nội dung | Tĩnh, được lưu trữ dưới dạng tập tin | Động, được tạo ra khi có yêu cầu |
| Thay đổi nội dung | Khó khăn, cần chỉnh sửa tập tin | Dễ dàng, thông qua hệ thống quản trị nội dung |
| Ví dụ | Website giới thiệu công ty, portfolio cá nhân | Website bán hàng trực tuyến, website tin tức |
| Ưu điểm | Chi phí thấp, dễ triển khai, bảo mật cao | Tính tương tác cao, dễ dàng cập nhật nội dung |
| Nhược điểm | Khó thay đổi nội dung, không có tính tương tác | Chi phí cao hơn, bảo mật thấp hơn |

Lựa chọn Web tĩnh hay Web động:

Lựa chọn Web tĩnh hay Web động phụ thuộc vào nhu cầu sử dụng của bạn:

Web tĩnh: Phù hợp cho website có nội dung ít thay đổi, chi phí thấp và yêu cầu bảo mật cao.

Web động: Phù hợp cho website có nội dung thay đổi thường xuyên, cần tính tương tác cao và dễ dàng cập nhật nội dung.

Kết luận:

Web tĩnh và Web động là hai loại website khác nhau với những ưu điểm và nhược điểm riêng.

**1.3. Một số thuật ngữ (Hosting, Tên miền-Domain Name-Web Server Name, HomePage,**

**Web Site, WebServer, URL-Uniform Resource Location, Browser, …)**

**Hosting:**

Dịch vụ cung cấp không gian lưu trữ cho website trên máy chủ web.

Giống như thuê một mảnh đất trên internet để chứa các tập tin của website của bạn.

Các nhà cung cấp dịch vụ hosting thường cung cấp nhiều gói hosting khác nhau với các mức giá và dung lượng lưu trữ khác nhau.

**Tên miền (Domain Name):**

Tên dễ nhớ được sử dụng để truy cập website thay cho địa chỉ IP của máy chủ web.

Giống như địa chỉ nhà của bạn trên internet, giúp người dùng dễ dàng tìm thấy website của bạn.

Tên miền cần được đăng ký với một nhà cung cấp dịch vụ tên miền để có thể sử dụng.

**Web Server Name:**

Tên của máy chủ web lưu trữ website.

Thường trùng với tên miền của website, nhưng không phải lúc nào cũng vậy.

Ít được sử dụng thông thường bởi người dùng cuối vì họ thường truy cập website thông qua tên miền.

**HomePage:**

Trang chủ của website, thường là trang đầu tiên mà người dùng nhìn thấy khi truy cập website.

Giống như trang chào đón của website, thường chứa các thông tin chính, menu điều hướng và liên kết đến các trang khác trong website.

**Website:**

Tập hợp các trang web được liên kết với nhau, có chung tên miền và được lưu trữ trên một hoặc nhiều máy chủ web.

Giống như một bộ sưu tập các tài liệu điện tử được tổ chức theo một cấu trúc nhất định.

Website có thể được sử dụng cho nhiều mục đích khác nhau, như giới thiệu thông tin, bán hàng trực tuyến, cung cấp dịch vụ hoặc giao tiếp.

**WebServer:**

Phần mềm chạy trên máy chủ web để xử lý các yêu cầu truy cập website.

Giống như một người trung gian giữa người dùng và website, nhận yêu cầu từ người dùng và gửi lại nội dung của website.

Các web server phổ biến bao gồm Apache, Nginx, IIS.

**URL (Uniform Resource Locator):**

Địa chỉ của trang web trên mạng internet.

Giống như địa chỉ đường dẫn đến một trang cụ thể trên website.

URL bao gồm các thành phần như giao thức (http, https), tên miền, đường dẫn thư mục và tên file.

**Browser (Trình duyệt web):**

Phần mềm được sử dụng để truy cập website.

Giống như một công cụ để hiển thị nội dung của website trên màn hình máy tính của bạn.

Các trình duyệt web phổ biến bao gồm Google Chrome, Mozilla Firefox, Safari, Microsoft Edge.

**Ngoài ra, còn một số thuật ngữ khác thường gặp:**

**HTML:** Ngôn ngữ lập trình được sử dụng để tạo ra cấu trúc của trang web.

**CSS:** Ngôn ngữ lập trình được sử dụng để định kiểu cho trang web, bao gồm màu sắc, phông chữ, và bố cục.

**JavaScript:** Ngôn ngữ lập trình được sử dụng để thêm chức năng cho trang web, chẳng hạn như tương tác với người dùng và xử lý dữ liệu.

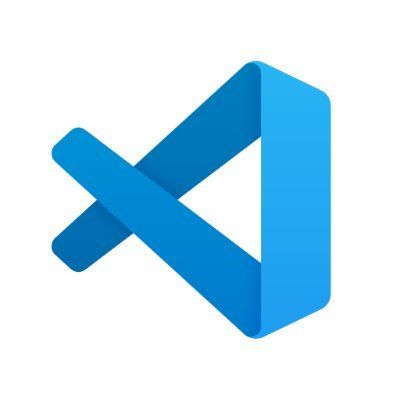
**Database:** Cơ sở dữ liệu lưu trữ thông tin của website, thường được sử dụng trong các website động.

Một số công cụ dùng thiết kế Web (Visual Studio code, Adobe Dreamweaver, …)

**1.4. Một số công cụ dùng thiết kế Web**

Có rất nhiều công cụ khác nhau được sử dụng để thiết kế web, mỗi công cụ có ưu điểm và nhược điểm riêng. Dưới đây là một số công cụ phổ biến:

**Visual Studio Code:**



***Trình soạn thảo mã miễn phí và mã nguồn mở được phát triển bởi Microsoft.***

Hỗ trợ nhiều ngôn ngữ lập trình, bao gồm HTML, CSS, JavaScript, PHP, Python, Java, C++, C#,...

Có nhiều tính năng hữu ích cho việc thiết kế web, như tô sáng cú pháp, tự động hoàn thành mã, gỡ lỗi,...

Phù hợp cho cả người mới bắt đầu và lập trình viên chuyên nghiệp.

**Adobe Dreamweaver:**



***Phần mềm thiết kế web chuyên nghiệp được phát triển bởi Adobe.***

Cung cấp giao diện trực quan WYSIWYG (What You See Is What You Get) giúp người dùng dễ dàng thiết kế web mà không cần biết nhiều về mã.

Hỗ trợ nhiều tính năng nâng cao, như quản lý nội dung, tích hợp với các hệ thống quản trị nội dung (CMS),...

Phù hợp cho người dùng muốn thiết kế web một cách chuyên nghiệp và hiệu quả.

**3. Sublime Text:**

Trình soạn thảo mã trả phí với nhiều tính năng nâng cao.

Hỗ trợ nhiều ngôn ngữ lập trình, bao gồm HTML, CSS, JavaScript, PHP, Python, Java, C++, C#,...

Giao diện đơn giản và dễ sử dụng.

Tốc độ xử lý nhanh chóng.

Phù hợp cho lập trình viên có kinh nghiệm.

**Notepad++:**



***Trình soạn thảo mã miễn phí và nhẹ dành cho hệ điều hành Windows.***

Hỗ trợ nhiều ngôn ngữ lập trình, bao gồm HTML, CSS, JavaScript, PHP, Python, Java, C++, C#,...

Giao diện đơn giản và dễ sử dụng.

Phù hợp cho người mới bắt đầu học lập trình web.

**Ngoài ra, còn có một số công cụ khác như:**

**WYSIWYG editors:** Các trình soạn thảo web trực quan cho phép người dùng thiết kế web mà không cần biết mã.

**CMS (Content Management System):** Hệ thống quản trị nội dung giúp người dùng dễ dàng tạo và quản lý website.

**Web frameworks:** Khung ứng dụng web giúp người dùng phát triển website một cách nhanh chóng và hiệu quả.

**Kết luận chương 1**

Thiết kế web và triển khai hệ thống phần mềm là hai lĩnh vực quan trọng trong công nghệ thông tin.

Việc hiểu rõ các khái niệm cơ bản về hai lĩnh vực này sẽ giúp bạn có thể bắt đầu học tập và làm việc trong lĩnh vực này.

Lựa chọn loại Web và công cụ thiết kế Web phù hợp phụ thuộc vào nhu cầu và kỹ năng và ứng dụng nó tốt hơn

**CHƯƠNG II: HTML & HTML5 (Hyper Text Markup Language)**

**2.1. Tổng quan về HTML**

**HTML** là viết tắt của **Hyper Text Markup Language**, là ngôn ngữ đánh dấu siêu văn bản được sử dụng để tạo ra cấu trúc và định dạng cho trang web. HTML sử dụng các thẻ để xác định các phần tử khác nhau của trang web, như tiêu đề, đoạn văn, hình ảnh, bảng, liên kết,.

Chức năng chính của HTML:

Xác định cấu trúc của trang web.

Định dạng nội dung trang web.

Tạo liên kết giữa các trang web.

Chèn hình ảnh, video, âm thanh vào trang web.

Tạo biểu mẫu để thu thập dữ liệu từ người dùng.

Ưu điểm của HTML:

Dễ học và sử dụng.

Miễn phí và mã nguồn mở.

Hỗ trợ bởi tất cả các trình duyệt web.

Có thể kết hợp với các ngôn ngữ lập trình khác để tạo ra các trang web phức tạp.

Nhược điểm của HTML:

Không có khả năng xử lý dữ liệu động.

Khó tạo ra các trang web có giao diện phức tạp.

Để học HTML, bạn có thể tham khảo các tài liệu sau:

Sách giáo khoa, tài liệu hướng dẫn.

Các khóa học trực tuyến.

Các cộng đồng online về thiết kế web và lập trình.

Kết luận:

HTML là ngôn ngữ cơ bản để tạo trang web. Việc học HTML là bước đầu tiên để bạn bước vào lĩnh vực thiết kế web và lập trình web.

**2.2. Cấu trúc tổng quát trang HTML**

Cấu trúc tổng quát của trang HTML bao gồm:

**Dòng khai báo DOCTYPE:**

Khai báo phiên bản HTML được sử dụng

**Thẻ** <html>**:**

Khai báo bắt đầu trang HTML.

**Thẻ** <head>**:**

Chứa thông tin về trang web, như tiêu đề, metadata.

**Thẻ** <title>**:**

Chứa tiêu đề của trang web.

**Thẻ** <body>**:**

Chứa nội dung chính của trang web.

**Thẻ** </html>**:**

Khai báo kết thúc trang HTML.

Dưới đây là ví dụ về cấu trúc tổng quát của trang HTML:

Giải thích các phần:

Dòng khai báo DOCTYPE: Khai báo phiên bản HTML5 được sử dụng.

Thẻ <html>: Khai báo bắt đầu trang HTML.

Thẻ <head>: Chứa thông tin về trang web, như tiêu đề.

Thẻ <title>: Chứa tiêu đề của trang web.

Thẻ <body>: Chứa nội dung chính của trang web, bao gồm:

Thẻ <h1>: Tạo tiêu đề cấp 1.

Thẻ <p>: Tạo đoạn văn.

Thẻ </html>: Khai báo kết thúc trang HTML.

Lưu ý:

Thẻ <html> và </html> là thẻ bắt đầu và kết thúc trang HTML.

Thẻ <head> và </head> là thẻ bắt đầu và kết thúc phần đầu trang HTML.

Thẻ <body> và </body> là thẻ bắt đầu và kết thúc phần thân trang HTML.

Thẻ <title> là thẻ khai báo tiêu đề của trang web.

Nội dung trang web được đặt trong thẻ <body>.

**2.3. Các thẻ HTML thông dụng**

***-Các thẻ xử lý văn bản***

<h1> đến <h6>: Tạo tiêu đề với các cấp độ khác nhau, <h1> là cấp độ lớn nhất, <h6> là cấp độ nhỏ nhất.

<p>: Tạo đoạn văn.

<strong>: Làm đậm văn bản.

<em>: In nghiêng văn bản.

<br>: Dòng xuống.

<code>: Định dạng văn bản như mã code.

<pre>: Định dạng văn bản với định dạng cố định.

<kbd>: Định dạng văn bản như phím bấm.

***-Các thẻ tạo bảng***

**Thẻ** <table>**:**

Khai báo bắt đầu bảng.

**Thẻ** <tr>**:**

Khai báo bắt đầu hàng trong bảng.

**Thẻ** <td>**:**

Khai báo bắt đầu ô trong bảng.

**Thẻ** </th>**:**

Khai báo bắt đầu ô tiêu đề trong bảng (thường dùng cho hàng đầu tiên của bảng).

**Thẻ** </table>**:**

Khai báo kết thúc bảng.

***-Thẻ liên kết***

**Thẻ** <a> được sử dụng để tạo liên kết trong HTML

***Các thẻ đa phương tiện***

**Chèn hình ảnh:**

Thẻ <img>:

src: Thuộc tính bắt buộc, chỉ ra đường dẫn đến hình ảnh.

alt: Thuộc tính mô tả nội dung của hình ảnh (hiển thị khi không tải được hình ảnh hoặc người dùng di chuột qua hình ảnh).

Chèn video:

Thẻ <video>:

controls: Thuộc tính hiển thị thanh điều khiển video.

source: Thuộc tính chỉ ra đường dẫn đến video.

type: Thuộc tính chỉ ra định dạng video

Chèn âm thanh:

Thẻ <audio>:

controls: Thuộc tính hiển thị thanh điều khiển âm thanh.

source: Thuộc tính chỉ ra đường dẫn đến âm thanh.

type: Thuộc tính chỉ ra định dạng âm thanh.

**Kết luận:**

Các thẻ đa phương tiện giúp chèn hình ảnh, video, âm thanh vào trang web, tạo ra nội dung phong phú và thu hút người dùng. Việc sử dụng các thẻ đa phương tiện một cách hợp lý sẽ giúp bạn tạo ra các trang web đẹp mắt và hiệu quả.

***-Thẻ tạo khung***

**Thẻ** <frameset>**:**

Khai báo bắt đầu bộ khung.

**Thẻ** <frame>**:**

Khai báo mỗi khung trong bộ khung.

rows: Thuộc tính chỉ ra kích thước các hàng trong bộ khung.

cols: Thuộc tính chỉ ra kích thước các cột trong bộ khung.

src: Thuộc tính chỉ ra trang web được hiển thị trong mỗi khung.

**2.4.Các thẻ tạo biểu mẫu (form)**

**Thẻ** <form>**:**

Khai báo bắt đầu biểu mẫu.

**Thẻ** <input>**:**

Tạo các trường nhập liệu trong biểu mẫu.

**Thẻ** <label>**:**

Nhãn cho các trường nhập liệu.

**Thẻ** <button>**:**

Tạo nút submit hoặc reset cho biểu mẫu.

Ngoài ra, bạn có thể sử dụng các thẻ khác để tạo biểu mẫu:

textarea: Tạo trường nhập liệu đa dòng.

select: Tạo danh sách lựa chọn.

option: Tạo các lựa chọn trong danh sách.

fieldset: Nhóm các trường nhập liệu liên quan.

legend: Tiêu đề cho nhóm trường nhập liệu.

**2.5.Một số thẻ HTML đặc biệt**

Một số thẻ HTML đặc biệt

Ngoài các thẻ HTML cơ bản được sử dụng phổ biến, còn có một số thẻ đặc biệt có chức năng riêng biệt và hữu ích trong việc thiết kế và phát triển trang web. Dưới đây là một số ví dụ:

**Thẻ** DOCTYPE**:**

Khai báo phiên bản HTML của trang web.

Giúp trình duyệt web hiểu và hiển thị trang web chính xác.

Ví dụ: <!DOCTYPE html>

**Thẻ** html**:**

Khai báo bắt đầu trang web HTML.

Chứa tất cả các nội dung của trang web.

Ví dụ: <html>

**Thẻ** head**:**

Chứa thông tin meta về trang web, như tiêu đề, mô tả, charset, favicon.

Không hiển thị trực tiếp trên trang web.

Ví dụ: <head>

**Thẻ** title**:**

Xác định tiêu đề của trang web.

Hiển thị trên thanh tiêu đề của trình duyệt web.

Ví dụ: <title>Trang web của tôi</title>

**Thẻ** meta**:**

Cung cấp thông tin meta về trang web, như charset, author, keywords, description.

Không hiển thị trực tiếp trên trang web.

Ví dụ: <meta charset="UTF-8">

**Thẻ** body**:**

Chứa nội dung chính của trang web, như văn bản, hình ảnh, video, biểu mẫu.

Hiển thị trực tiếp trên trang web.

Ví dụ: <body>

**Thẻ** script**:**

Chèn mã JavaScript vào trang web.

Cho phép thêm chức năng tương tác cho trang web.

Ví dụ: <script src="script.js"></script>

**Thẻ** link**:**

Tạo liên kết đến các tài nguyên bên ngoài, như stylesheet, favicon.

Ví dụ: <link rel="stylesheet" href="style.css">

**Thẻ** style**:**

Định dạng nội dung trang web bằng CSS.

Ví dụ: <style>body {color: red;}</style>

**Thẻ** img**:**

Chèn hình ảnh vào trang web.

Ví dụ: <img src="image.jpg" alt="Đây là một bức ảnh">

**Thẻ** video**:**

Chèn video vào trang web.

Ví dụ: <video controls><source src="video.mp4" type="video/mp4"></video>

**Thẻ** audio**:**

Chèn âm thanh vào trang web.

Ví dụ: <audio controls><source src="audio.mp3" type="audio/mp3"></audio>

**Thẻ** iframe**:**

Chèn một trang web khác vào trang web hiện tại.

Ví dụ: <iframe src="https://www.google.com"></iframe>

**2.6. HTML5**

***-HTML5 là gì?***

HTML5 là phiên bản thứ năm của ngôn ngữ lập trình HTML (HyperText Markup Language) được sử dụng để tạo cấu trúc và nội dung cho các trang web. HTML5 được phát triển bởi World Wide Web Consortium (W3C) với mục tiêu tạo ra một ngôn ngữ web mạnh mẽ, linh hoạt và dễ sử dụng hơn so với các phiên bản trước đây.

Một số tính năng chính của HTML5 bao gồm:

Cấu trúc ngữ nghĩa: HTML5 cung cấp các thẻ mới để xác định rõ ràng ý nghĩa của nội dung trên trang web, ví dụ như header, article, section, aside và footer.

Đa phương tiện: HTML5 hỗ trợ tích hợp các nội dung đa phương tiện như video và âm thanh trực tiếp vào trang web mà không cần đến plugin hay phần mềm bổ sung.

Vẽ đồ họa: HTML5 cung cấp API Canvas cho phép vẽ đồ họa và hình ảnh trực tiếp trên trang web.

Lưu trữ dữ liệu cục bộ: HTML5 cung cấp API Web Storage cho phép lưu trữ dữ liệu cục bộ trên trình duyệt web.

Kết nối WebSocket: HTML5 cho phép tạo kết nối hai chiều giữa trình duyệt web và máy chủ, giúp tạo ra các ứng dụng web thời gian thực.

HTML5 mang lại nhiều lợi ích cho việc phát triển web:

Dễ sử dụng: HTML5 có cú pháp đơn giản và dễ học hơn so với các phiên bản trước đây.

Hiệu quả: HTML5 giúp tối ưu hóa hiệu suất của trang web.

Khả năng tương thích: HTML5 được hỗ trợ bởi hầu hết các trình duyệt web hiện đại.

Tính linh hoạt: HTML5 có thể được sử dụng để phát triển nhiều loại ứng dụng web khác nhau.

HTML5 là một công nghệ quan trọng trong việc phát triển web hiện đại. Việc sử dụng HTML5 giúp bạn tạo ra các trang web đẹp mắt, hiệu quả và dễ sử dụng.

***-Cú pháp HTML5***

Cú pháp HTML5 dựa trên cú pháp XML, với một số điểm khác biệt để đơn giản hóa việc sử dụng. Dưới đây là một số điểm chính về cú pháp HTML5:

Cấu trúc cơ bản:

Một trang HTML5 bắt đầu vớiDOCTYPE <!DOCTYPE html>.

Tiếp theo là thẻ <html> bao gồm toàn bộ nội dung của trang web.

Bên trong thẻ <html> có thẻ <head> chứa thông tin meta về trang web, như tiêu đề, mô tả, charset, favicon.

Sau thẻ <head> là thẻ <body> chứa nội dung chính của trang web, như văn bản, hình ảnh, video, biểu mẫu.

Thẻ HTML:

Thẻ HTML được viết dưới dạng cặp thẻ mở và thẻ đóng.

Ví dụ: <p> là thẻ mở cho đoạn văn bản, </p> là thẻ đóng cho đoạn văn bản.

Một số thẻ HTML có thể được sử dụng tự đóng, ví dụ: <img> là thẻ tự đóng cho hình ảnh.

Thuộc tính:

Thẻ HTML có thể có các thuộc tính để định dạng hoặc cung cấp thêm thông tin về nội dung.

Ví dụ: thẻ <img> có thể có thuộc tính src để chỉ ra đường dẫn đến hình ảnh.

Giá trị thuộc tính:

Giá trị img thuộc tính được đặt trong ngoặc kép.

Ví dụ: < src="image.jpg">

Nhận xét:

Nhận xét được sử dụng để ghi chú trong mã HTML.

Nhận xét bắt đầu với ``.

Dấu cách:

Dấu cách được sử dụng để định dạng mã HTML cho dễ đọc.

Tuy nhiên, HTML5 không yêu cầu bắt buộc sử dụng dấu cách.

Chữ hoa và chữ thường:

Cú pháp HTML5 không phân biệt chữ hoa và chữ thường.

***-Khai báo DOCTYPE trong HTML5***

Khai báo DOCTYPE là dòng đầu tiên trong một tài liệu HTML5. Nó thông báo cho trình duyệt web về phiên bản HTML được sử dụng trong tài liệu. Khai báo DOCTYPE cho HTML5 là:

HTML

<!DOCTYPE html>

**Lý do cần khai báo DOCTYPE:**

Giúp trình duyệt web hiển thị trang web chính xác theo tiêu chuẩn HTML5.

Kích hoạt các tính năng mới của HTML5.

Tránh các lỗi hiển thị và tương thích.

**Cách khai báo DOCTYPE:**

Khai báo DOCTYPE phải là dòng đầu tiên trong tài liệu HTML.

Khai báo DOCTYPE phải được viết đúng chính tả và cú pháp.

Không được thêm bất kỳ ký tự nào khác vào dòng khai báo DOCTYPE.

**Ví dụ:**

HTML

<!DOCTYPE html>

<html lang="vi">

<head>

<meta charset="UTF-8">

<title>Tiêu đề trang web</title>

</head>

<body>

Nội dung trang web

</body>

</html>

**Lưu ý:**

Khai báo DOCTYPE không bắt buộc, nhưng nó được khuyến nghị để đảm bảo tính tương thích và hiển thị chính xác.

Nếu bạn không khai báo DOCTYPE, trình duyệt web có thể hiển thị trang web theo chế độ quirks mode, có thể dẫn đến các lỗi hiển thị và tương thích.

**Kết luận:**

Khai báo DOCTYPE là một phần quan trọng trong HTML5. Khai báo DOCTYPE giúp đảm bảo tính tương thích, hiển thị chính xác và kích hoạt các tính năng mới của HTML5.

***-Thẻ ngữ nghĩa***  
Thẻ ngữ nghĩa trong HTML5

Thẻ ngữ nghĩa trong HTML5 là những thẻ được sử dụng để mô tả ý nghĩa và chức năng của các phần nội dung trên trang web. Việc sử dụng thẻ ngữ nghĩa giúp trình duyệt và các công cụ tìm kiếm hiểu rõ hơn về cấu trúc và nội dung của trang web, từ đó cải thiện khả năng truy cập, khả năng sử dụng và SEO.

**-Dưới đây là một số thẻ ngữ nghĩa phổ biến trong HTML5:**

article**:** Dùng để định nghĩa một bài viết độc lập, có thể bao gồm tiêu đề, nội dung, tác giả, ngày xuất bản, v.v.

aside**:** Dùng để định nghĩa nội dung bổ sung cho nội dung chính của trang web, như thanh bên, quảng cáo, v.v.

details**:** Dùng để định nghĩa nội dung chi tiết có thể thu gọn/mở rộng.

figure**:** Dùng để định nghĩa một hình ảnh cùng với chú thích.

footer**:** Dùng để định nghĩa phần chân trang của trang web, thường bao gồm thông tin liên hệ, bản quyền, v.v.

header**:** Dùng để định nghĩa phần đầu trang của trang web, thường bao gồm logo, tiêu đề, menu, v.v.

main**:** Dùng để định nghĩa nội dung chính của trang web.

nav**:** Dùng để định nghĩa một hệ thống điều hướng cho trang web.

section**:** Dùng để định nghĩa một phần nội dung riêng biệt trong trang web.

**Lợi ích của việc sử dụng thẻ ngữ nghĩa:**

**Cải thiện khả năng truy cập:** Thẻ ngữ nghĩa giúp trình đọc màn hình và các công cụ hỗ trợ khác hiểu rõ hơn về nội dung trang web, giúp người dùng khuyết tật có thể truy cập và sử dụng trang web dễ dàng hơn.

**Cải thiện khả năng sử dụng:** Thẻ ngữ nghĩa giúp người dùng dễ dàng xác định và hiểu nội dung của trang web.

**Cải thiện SEO:** Thẻ ngữ nghĩa giúp các công cụ tìm kiếm hiểu rõ hơn về nội dung trang web, từ đó cải thiện thứ hạng của trang web trong kết quả tìm kiếm.

**Lưu ý:**

Nên sử dụng thẻ ngữ nghĩa một cách chính xác và phù hợp với nội dung của trang web.

Tránh sử dụng thẻ ngữ nghĩa để thay đổi định dạng của trang web.

**Kết luận:**

Thẻ ngữ nghĩa là một phần quan trọng trong HTML5. Việc sử dụng thẻ ngữ nghĩa giúp cải thiện khả năng truy cập, khả năng sử dụng và SEO cho trang web của bạn.

***-Webform***

Web form là một phần tử HTML được sử dụng để thu thập dữ liệu từ người dùng. Web form bao gồm các thành phần như:

Nhãn (label): Dùng để mô tả mục đích của từng trường dữ liệu.

Trường dữ liệu (input): Nơi người dùng nhập dữ liệu. Có nhiều loại trường dữ liệu khác nhau như text, password, checkbox, radio, select, v.v.

Nút submit (button): Dùng để gửi dữ liệu đến máy chủ.

HTML5 cung cấp một số tính năng mới cho web form:

Kiểu dữ liệu (data type): Giúp trình duyệt xác định loại dữ liệu được nhập vào từng trường.

Placeholder: Giúp hiển thị nội dung gợi ý trong trường dữ liệu.

Autofocus: Tự động đặt con trỏ chuột vào trường dữ liệu đầu tiên khi trang web được tải.

Required: Yêu cầu người dùng phải nhập dữ liệu vào trường dữ liệu.

Validation: Giúp kiểm tra tính hợp lệ của dữ liệu được nhập.

Lợi ích của việc sử dụng web form:

Thu thập dữ liệu từ người dùng: Web form giúp bạn thu thập thông tin từ người dùng như tên, email, số điện thoại, v.v.

Tạo các ứng dụng web tương tác: Web form có thể được sử dụng để tạo các ứng dụng web tương tác như khảo sát, bình chọn, đăng ký, v.v.

Lưu ý:

Nên thiết kế web form đơn giản và dễ sử dụng.

Sử dụng các thông báo lỗi rõ ràng để giúp người dùng sửa lỗi khi nhập dữ liệu.

Bảo vệ dữ liệu người dùng bằng các biện pháp an ninh phù hợp.

Kết luận:

Web form là một phần quan trọng trong HTML5. Việc sử dụng web form giúp bạn thu thập dữ liệu từ người dùng và tạo các ứng dụng web tương tác.

**Kết luận chương 2**

-giúp ta xác định được HTML có lịch sử và lợi ích như nào .cho ta thấy cách nó hoạt động các cú pháp có thể tạo ra 1 trang wed hoản trình (từ âm thanh đến hình ảnh ) và có thể tạo ra đa phương tiện HTML5 là ngôn ngữ web hiện đại với nhiều tính năng mạnh mẽ và linh hoạt.

Cú pháp HTML5 đơn giản và dễ học.

Việc sử dụng HTML5 giúp bạn tạo ra các trang web đẹp mắt, hiệu quả và dễ sử dụng.

**CHƯƠNG III: CSS và CSS3Cascading Style Sheets)**

**3.1. CSS là gì?**

CSS là viết tắt của Cascading Style Sheets, là ngôn ngữ lập trình dùng để định dạng và tạo kiểu cho các trang web. CSS giúp kiểm soát cách hiển thị của các phần tử HTML như màu sắc, phông chữ, bố cục, v.v.

Ví dụ:

Thay đổi màu sắc của tiêu đề trang web.

Định dạng phông chữ cho văn bản.

Tạo bố cục hai cột cho trang web.

Thêm viền cho hình ảnh.

Hiển thị ảnh động.

Lợi ích của CSS:

Giúp trang web đẹp mắt và thu hút hơn.

Giúp người dùng dễ đọc và hiểu nội dung trang web.

Giúp kiểm soát bố cục trang web.

Giúp trang web tải nhanh hơn.

Có ba cách để áp dụng CSS vào trang HTML:

Nhúng trực tiếp: Sử dụng thuộc tính style trong thẻ HTML.

CSS nội tuyến: Sử dụng thẻ <style> trong phần <head> của trang HTML.

Tập tin CSS riêng: Tạo tập tin .css riêng và liên kết nó với trang HTML sử dụng thẻ <link>.

**3.2. Cú pháp CSS**

Cú pháp CSS bao gồm:

**Chọn phần tử:**

Sử dụng **selector** để xác định phần tử HTML muốn định dạng. Có nhiều loại selector khác nhau như:

**Universal selector:** Chọn tất cả các phần tử (\*).

**Type selector:** Chọn phần tử dựa trên tên thẻ (h1, p, v.v.).

**Identity selector:** Chọn phần tử dựa trên ID (#id).

**Class selector:** Chọn phần tử dựa trên class (class).

**Descendant selector:** Chọn phần tử con (div p).

**Child selector:** Chọn phần tử con trực tiếp (div > p).

**Adjacent sibling selector:** Chọn phần tử liền kề (h1 + p).

**Attribute selector:** Chọn phần tử dựa trên thuộc tính ([href]).

**Pseudo class selector:** Chọn phần tử dựa trên trạng thái (a:hover).

**Group selector:** Chọn một nhóm phần tử (h1, h2, h3).

**Khai báo thuộc tính:**

Sử dụng tên thuộc tính và giá trị để định dạng phần tử.

**Khối quy tắc:**

Bao gồm selector và các khai báo thuộc tính được bao quanh bởi dấu ngoặc nhọn { }.

**Ví dụ:**

CSS

h1 {

color: red;

font-size: 24px;

Thứ tự của các quy tắc CSS rất quan trọng. Quy tắc được khai báo sau sẽ ghi đè lên quy tắc được khai báo trước.

Có thể sử dụng nhiều selector cho một quy tắc CSS.

Có thể sử dụng dấu phẩy để phân cách các selector trong cùng một quy tắc CSS.

**3.3. Áp dụng CSS vào trang HTML**

Có ba cách để áp dụng CSS vào trang HTML:

**Nhúng trực tiếp:**

Sử dụng thuộc tính style trong thẻ HTML để khai báo các quy tắc CSS trực tiếp cho phần tử đó.

**CSS nội tuyến:**

Sử dụng thẻ <style> trong phần <head> của trang HTML để khai báo các quy tắc CSS cho toàn bộ trang

**Tập tin CSS riêng:**

Tạo tập tin .css riêng để khai báo các quy tắc CSS và liên kết nó với trang HTML sử dụng thẻ <link> trong phần <head> của trang HTML.

Lợi ích của việc sử dụng tập tin CSS riêng:

Giúp code HTML gọn gàng và dễ đọc hơn.

Giúp dễ dàng quản lý và cập nhật các quy tắc CSS.

Giúp tăng hiệu suất tải trang web.

Chú ý

Thứ tự áp dụng CSS: Nhúng trực tiếp > CSS nội tuyến > Tập tin CSS riêng.

Khi sử dụng tập tin CSS riêng, cần lưu ý đến đường dẫn tương đối của tập tin

**3.4. Selectors**

Selectors trong CSS là những công cụ giúp bạn xác định các phần tử HTML mà bạn muốn định dạng. Chúng hoạt động như những con trỏ để xác định các phần tử mà bạn muốn áp dụng quy tắc CSS.

Có nhiều loại selectors khác nhau trong CSS:

Universal Selector (\*): Selector này chọn tất cả các phần tử trên trang. Nó ít được sử dụng trong thực tế vì việc quản lý các styles sẽ khó khăn nếu áp dụng cho mọi thứ.

Type Selector (ví dụ: h1, p, div): Selector này chọn các phần tử dựa trên tên thẻ HTML của chúng. Ví dụ: h1 chọn tất cả các heading level 1, và p chọn tất cả các paragraph.

ID Selector (#id): Selector này chọn một phần tử có ID attribute duy nhất. ID giống như tên được gán cho các phần tử cụ thể, và selector sử dụng ký hiệu # theo sau là tên ID.

Ví dụ: <h1 id="main-title">Đây là tiêu đề chính</h1>

Class Selector (.class): Selector này chọn các phần tử có class attribute cụ thể. Class có thể được gán cho nhiều phần tử, cho phép bạn định dạng chúng tương tự nhau. Selector sử dụng ký hiệu . theo sau là tên class.

Ví dụ: <p class="important">Đây là văn bản quan trọng.</p>

Descendant Selector (ancestor element > descendant element): Selector này chọn các phần tử là con cháu của một phần tử khác. Selector sử dụng ký hiệu > để phân biệt phần tử tổ tiên và phần tử con cháu.

Ví dụ: div > p chọn tất cả các paragraph là con trực tiếp của div.

Child Selector (ancestor element > child element): Selector này tương tự như descendant selector, nhưng nó chỉ chọn con trực tiếp của phần tử tổ tiên được chỉ định.

Ví dụ: #content > p chọn tất cả các paragraph là con trực tiếp của element có ID "content".

Adjacent Sibling Selector (element1 + element2): Selector này chọn một phần tử ngay sau một phần tử khác. Selector sử dụng ký hiệu + để phân biệt hai loại phần tử.

Ví dụ: h1 + p chọn tất cả các paragraph ngay sau một h1.

Attribute Selector ([attribute]): Selector này chọn các phần tử dựa trên sự hiện diện hoặc giá trị của một attribute cụ thể. Selector sử dụng ngoặc vuông [] để bao quanh tên attribute và tùy chọn là một toán tử so sánh và giá trị.

Ví dụ: [href] chọn tất cả các phần tử có attribute href (như liên kết), và [href="#top"] chọn tất cả các phần tử có attribute href liên kết đến element có ID "top".

Pseudo-Class Selector (element:pseudo-class): Selector này chọn các phần tử dựa trên trạng thái hoặc hành vi của chúng. Chúng sử dụng dấu hai chấm : theo sau là tên pseudo-class. Các ví dụ phổ biến bao gồm:

:hover: Áp dụng styles khi người dùng di chuột qua một phần tử.

:active: Áp dụng styles khi người dùng nhấp vào một phần tử.

:focus: Áp dụng styles khi một phần tử được chọn (ví dụ: khi một trường form được chọn).

Group Selector (selector1, selector2, ...): Selector này cho phép bạn chọn nhiều phần tử cùng một lúc, được phân cách bằng dấu phẩy.

Ví dụ: h1, h2, h3 chọn tất cả các heading level 1, 2 và 3.

Bằng cách hiểu và sử dụng hiệu quả các loại selectors khác nhau. có thể chọn chính xác các phần tử mà bạn muốn định dạng trong trang web của mình, tạo ra trải nghiệm trực quan và thân thiện với người dùng.

***3.4.1.Universal selector***

**Universal selector (bộ chọn vạn năng)** trong CSS là ký tự \* dùng để **chọn tất cả các phần tử HTML** trên trang web, bất kể tên thẻ, ID hay class của chúng

Trường hợp sử dụng:

Khôi phục về kiểu mặc định: Universal selector có thể được sử dụng kết hợp với từ khóa initial để đưa tất cả các kiểu trên các phần tử về kiểu mặc định của trình duyệt. Điều này hữu ích để thiết lập nền tảng cho việc tạo kiểu cho tất cả các phần tử.

Kiểu tạm thời: Để kiểm tra nhanh hoặc demo, bạn có thể sử dụng universal selector để áp dụng các kiểu tạm thời cho tất cả các phần tử trên trang.

Tuy nhiên, việc sử dụng quá nhiều universal selector trong code thực tế thường không được khuyến khích. Lý do:

Tính đặc thù (specificity): Universal selector có tính đặc thù rất thấp, nghĩa là nó dễ dàng bị ghi đè bởi các selector có tính đặc thù cao hơn như type selector, ID selector và class selector. Điều này có thể gây khó khăn trong việc quản lý và kiểm soát các kiểu.

Tính bảo trì: Sử dụng universal selector có thể dẫn đến code khó bảo trì hơn, vì khó theo dõi các kiểu được áp dụng ở đâu, đặc biệt trong các dự án lớn.

Hiệu suất: Mặc dù tác động đến hiệu suất thường không đáng kể, nhưng việc sử dụng universal selector quá nhiều có thể làm chậm quá trình render của trình duyệt do cần phải kiểm tra nó với tất cả các phần tử.

Các lựa chọn thay thế cho universal selector:

Group selector: Sử dụng dấu phẩy để chọn một nhóm các type selector cụ thể, ví dụ h1, p, span.

Descendant selector: Kết hợp type selector với descendant selector để chọn các phần tử cụ thể trong cấu trúc, ví dụ div p.

Khôi phục về kiểu mặc định: Sử dụng các framework hoặc thư viện CSS cung cấp các kiểu reset được định sẵn để thiết lập nền tảng cho tất cả các phần tử.

Mặc dù universal selector có thể là một công cụ hữu ích trong các tình huống cụ thể, nhưng nên sử dụng nó một cách tiết kiệm và cân nhắc các phương pháp thay thế để có khả năng bảo trì và kiểm soát tốt hơn các kiểu CSS

***3.4.2. Type selector***

**Type selector** là một bộ chọn cơ bản trong CSS, dùng để chọn các phần tử HTML cụ thể dựa trên loại phần tử của chúng. Đây là một trong những bộ chọn đơn giản và được sử dụng phổ biến nhất.

**Cách thức hoạt động:**

Type selector được viết đơn giản bằng tên của phần tử HTML, ví dụ:

h1 chọn tất cả các phần tử tiêu đề <h1> trên trang.

p chọn tất cả các phần tử đoạn văn <p>.

span chọn tất cả các phần tử span <span>.

Bạn có thể chọn bất kỳ phần tử HTML hợp lệ nào bằng cách sử dụng tên thẻ của nó.

**Ví dụ:**

CSS

/\* Chọn tất cả các phần tử đoạn văn và đặt màu của chúng thành đỏ \*/

p {

color: red;

}

/\* Chọn tất cả các thẻ neo (liên kết) và thêm một số khoảng đệm \*/

a {

padding: 10px;

}

/\* Chọn tất cả các phần tử hình ảnh (img) và đặt chiều rộng của chúng thành 200px \*/

img {

width: 200px;

}Hãy [thận trọng](https://gemini.google.com/faq#coding) khi sử dụng các đoạn mã.

content\_copy

**Ưu điểm của type selector:**

**Đơn giản và dễ sử dụng:** Chúng dễ viết và dễ hiểu.

**Đa năng:** Chúng có thể được sử dụng để chọn nhiều loại phần tử.

**Điểm khởi đầu tốt:** Chúng thường được sử dụng như nền tảng cho các bộ chọn phức tạp hơn kết hợp nhiều loại bộ chọn.

**Nhược điểm của type selector:**

**Thiếu tính đặc thù:** Chúng có thể quá rộng nếu bạn có nhiều phần tử cùng loại và cần định kiểu khác nhau cho chúng.

**Để định kiểu cụ thể hơn, bạn có thể cần kết hợp type selector với các bộ chọn khác như class selector hoặc ID selector.**

**Điểm bổ sung:**

Type selector không phân biệt chữ hoa chữ thường. Ví dụ: h1 giống như H1.

Bạn có thể sử dụng ký tự đại diện với type selector. Ví dụ: \* chọn tất cả các phần tử trên trang và div \* chọn tất cả các phần tử con của div. Tuy nhiên, sử dụng ký tự đại diện quá nhiều có thể dẫn đến định kiểu không mong muốn và thường không được khuyến khích.

**Tóm lại:**

Type selector là một công cụ mạnh mẽ để định kiểu cơ bản trong CSS. Chúng cho phép bạn chọn các phần tử cụ thể theo loại phần tử HTML của chúng. Tuy nhiên, để định kiểu chính xác hơn, bạn thường cần kết hợp chúng với các bộ chọn khác.

***3.4.3. Identity selector***

Identity selector là gì?

**Identity selector**, hay còn gọi là **bộ chọn ID**, là một bộ chọn CSS dùng để chọn một phần tử duy nhất trên trang HTML dựa trên giá trị ID của nó. ID là một thuộc tính đặc biệt của mỗi phần tử HTML, đảm bảo tính duy nhất trên toàn bộ trang.

**Cách thức hoạt động:**

Identity selector được viết bằng ký hiệu # theo sau là giá trị ID của phần tử. Ví dụ:

#header chọn phần tử có ID là header.

#main-content chọn phần tử có ID là main-content.

**Lưu ý:**

Giá trị ID phải được đặt trong ngoặc kép (") nếu nó chứa dấu cách hoặc ký tự đặc biệt.

ID phải là duy nhất trong toàn bộ trang HTML.

**Ví dụ:**

HTML

<h1 id="header">Đây là tiêu đề trang</h1>

<div id="main-content">Đây là nội dung chính của trang</div>

CSS

#header {

color: red;

}

#main-content {

background-color: blue;

}

**Kết quả:**

Tiêu đề trang sẽ có màu đỏ.

Nội dung chính của trang sẽ có nền màu xanh lam.

**Ưu điểm của identity selector:**

**Tính đặc thù cao:** Identity selector đảm bảo chỉ chọn một phần tử duy nhất trên trang.

**Dễ sử dụng:** Việc xác định ID cho phần tử và sử dụng nó trong selector rất đơn giản.

**Nhược điểm của identity selector:**

**Tính linh hoạt thấp:** ID chỉ có thể được sử dụng cho một phần tử duy nhất, hạn chế khả năng áp dụng cho nhiều phần tử.

**Khó bảo trì:** Việc thay đổi ID của một phần tử có thể ảnh hưởng đến nhiều quy tắc CSS, gây khó khăn trong việc bảo trì code.

**Tóm lại:**

Identity selector là một công cụ hữu ích để chọn và định kiểu một phần tử duy nhất trên trang HTML với độ chính xác cao. Tuy nhiên, bạn nên cân nhắc tính linh hoạt và khả năng bảo trì khi sử dụng nó trong dự án của mình.

**Ngoài ra:**

Bạn có thể kết hợp identity selector với các bộ chọn khác để tạo ra các quy tắc CSS phức tạp hơn.

Sử dụng identity selector quá nhiều có thể khiến code CSS trở nên khó đọc và khó bảo trì.

***3.4.4. Class selector***

**Class selector**, hay còn gọi là **bộ chọn lớp**, là một bộ chọn CSS dùng để chọn các phần tử HTML có cùng lớp (class). Lớp là một thuộc tính có thể được áp dụng cho nhiều phần tử trên trang, giúp bạn dễ dàng định kiểu cho các nhóm phần tử liên quan.

**Cách thức hoạt động:**

Class selector được viết bằng ký hiệu . theo sau là tên lớp. Ví dụ:

.button chọn tất cả các phần tử có lớp button.

.menu-item chọn tất cả các phần tử có lớp menu-item.

**Lưu ý:**

Tên lớp không được chứa dấu cách hoặc ký tự đặc biệt.

Có thể sử dụng nhiều lớp cho cùng một phần tử, phân cách nhau bởi dấu cách.

**Ví dụ:**

HTML

<button class="button">Đây là nút</button>

<button class="button important">Đây là nút quan trọng</button>

<li class="menu-item">Đây là mục menu</li>

<li class="menu-item active">Đây là mục menu đang hoạt động</li>.

content\_copy

CSS

.button {

color: white;

background-color: blue;

}

.button.important {

background-color: red;

}

.menu-item {

color: black;

}

.menu-item.active {

background-color: gray;

}

**Kết quả:**

Tất cả các nút sẽ có màu trắng và nền màu xanh lam.

Nút có lớp important sẽ có nền màu đỏ.

Tất cả các mục menu sẽ có màu đen.

Mục menu có lớp active sẽ có nền màu xám.

**Ưu điểm của class selector:**

**Tính linh hoạt cao:** Class selector có thể được áp dụng cho nhiều phần tử, giúp bạn dễ dàng định kiểu cho các nhóm phần tử liên quan.

**Dễ sử dụng:** Việc thêm và sử dụng lớp cho phần tử rất đơn giản.

**Khả năng bảo trì cao:** Việc thay đổi định kiểu cho một lớp sẽ ảnh hưởng đến tất cả các phần tử có lớp đó, giúp bạn dễ dàng bảo trì code.

**Nhược điểm của class selector:**

**Tính đặc thù thấp:** Class selector có thể chọn nhiều phần tử trên trang, gây khó khăn khi cần định kiểu cho một phần tử cụ thể.

**Tóm lại:**

Class selector là một công cụ hữu ích để chọn và định kiểu cho các nhóm phần tử liên quan trên trang HTML với tính linh hoạt và khả năng bảo trì cao. Tuy nhiên, bạn nên cân nhắc tính đặc thù khi sử dụng nó trong dự án của mình.

**Ngoài ra:**

Bạn có thể kết hợp class selector với các bộ chọn khác để tạo ra các quy tắc CSS phức tạp hơn.

Sử dụng class selector quá nhiều có thể khiến code CSS trở nên khó đọc và khó bảo trì

***3.4.5 Descendant selector***  
Descendant selector là gì?

**Descendant selector**, hay còn gọi là **bộ chọn hậu duệ**, là một bộ chọn CSS dùng để chọn tất cả các phần tử con cháu (descendant) của một phần tử được chỉ định.

**Cách thức hoạt động:**

Descendant selector được viết bằng cách sử dụng **khoảng trắng** để phân cách tên của phần tử cha và phần tử con. Ví dụ:

div p chọn tất cả các phần tử <p> nằm bên trong phần tử <div>.

ul li chọn tất cả các phần tử <li> nằm bên trong phần tử <ul>.

**Lưu ý:**

Descendant selector có thể chọn nhiều phần tử trên trang, bao gồm cả các phần tử lồng nhau nhiều cấp.

Descendant selector có tính đặc thù thấp hơn so với child selector.

**Ví dụ:**

HTML

<div>

<p>Đây là đoạn văn đầu tiên</p>

<p>Đây là đoạn văn thứ hai</p>

<ul>

<li>Đây là mục danh sách đầu tiên</li>

<li>Đây là mục danh sách thứ hai</li>

</ul>

</div>

CSS

div p {

color: red;

}

ul li {

color: blue;

}

**Kết quả:**

Tất cả các đoạn văn <p> nằm bên trong phần tử <div> sẽ có màu đỏ.

Tất cả các mục danh sách <li> nằm bên trong phần tử <ul> sẽ có màu xanh lam.

**Ưu điểm của descendant selector:**

**Dễ sử dụng:** Việc sử dụng descendant selector rất đơn giản và dễ hiểu.

**Tính linh hoạt cao:** Descendant selector có thể chọn nhiều phần tử trên trang, giúp bạn dễ dàng định kiểu cho các nhóm phần tử liên quan.

**Nhược điểm của descendant selector:**

**Tính đặc thù thấp:** Descendant selector có thể chọn nhiều phần tử trên trang, gây khó khăn khi cần định kiểu cho một phần tử cụ thể.

**Khó bảo trì:** Việc thay đổi định kiểu cho một phần tử cha có thể ảnh hưởng đến nhiều phần tử con, gây khó khăn trong việc bảo trì code.

**Tóm lại:**

Descendant selector là một công cụ hữu ích để chọn và định kiểu cho các nhóm phần tử liên quan trên trang HTML với tính linh hoạt cao. Tuy nhiên, bạn nên cân nhắc tính đặc thù và khả năng bảo trì khi sử dụng nó trong dự án của mình.

**Ngoài ra:**

Bạn có thể kết hợp descendant selector với các bộ chọn khác để tạo ra các quy tắc CSS phức tạp hơn.

Sử dụng descendant selector quá nhiều có thể khiến code CSS trở nên khó đọc và khó bảo trì.

**3.4.6. Child selector**

**Child selector**, hay còn gọi là **bộ chọn con**, là một bộ chọn CSS dùng để chọn tất cả các phần tử con trực tiếp (direct child) của một phần tử được chỉ định.

**Cách thức hoạt động:**

Child selector được viết bằng cách sử dụng **ký hiệu** > để phân cách tên của phần tử cha và phần tử con. Ví dụ:

div > p chọn tất cả các phần tử <p> là con trực tiếp của phần tử <div>.

ul > li chọn tất cả các phần tử <li> là con trực tiếp của phần tử <ul>.

**Lưu ý:**

Child selector chỉ chọn các phần tử con trực tiếp, không chọn các phần tử con cháu (descendant).

Child selector có tính đặc thù cao hơn so với descendant selector.

**Ví dụ:**

HTML

<div>

<p>Đây là đoạn văn đầu tiên</p>

<p>Đây là đoạn văn thứ hai</p>

<ul>

<li>Đây là mục danh sách đầu tiên</li>

<li>Đây là mục danh sách thứ hai</li>

</ul>

</div>

CSS

div > p {

color: red;

}

ul > li {

color: blue;

}

**Kết quả:**

Chỉ có các đoạn văn <p> là con trực tiếp của phần tử <div> (không bao gồm các đoạn văn nằm bên trong <ul>) sẽ có màu đỏ.

Chỉ có các mục danh sách <li> là con trực tiếp của phần tử <ul> (không bao gồm các mục danh sách nằm bên trong <div>) sẽ có màu xanh lam.

**Ưu điểm của child selector:**

**Tính đặc thù cao:** Child selector chỉ chọn các phần tử con trực tiếp, giúp bạn dễ dàng định kiểu cho các phần tử cụ thể.

**Dễ sử dụng:** Việc sử dụng child selector rất đơn giản và dễ hiểu.

**Nhược điểm của child selector:**

**Tính linh hoạt thấp:** Child selector chỉ chọn các phần tử con trực tiếp, không chọn các phần tử con cháu.

**Tóm lại:**

Child selector là một công cụ hữu ích để chọn và định kiểu cho các phần tử cụ thể trên trang HTML với tính đặc thù cao. Tuy nhiên, bạn nên cân nhắc tính linh hoạt khi sử dụng nó trong dự án của mình.

**Ngoài ra:**

Bạn có thể kết hợp child selector với các bộ chọn khác để tạo ra các quy tắc CSS phức tạp hơn.

Sử dụng child selector quá nhiều có thể khiến code CSS trở nên khó đọc và khó bảo trì.

***3.4.7 Adjacent sibling selector***

Bộ chọn Anh/chị/em kế (adjacent sibling selector) trong CSS được dùng để chọn một phần tử nằm **ngay sau** một phần tử khác **cùng cha mẹ**.

**Cách thức hoạt động:**

Bộ chọn này được viết bằng cách sử dụng dấu **cộng (+)** theo sau bộ chọn của phần tử thứ nhất. Ví dụ:

h1 + p chọn **phần tử p đầu tiên** nằm **ngay sau** một phần tử <h1> trong cùng một khối cha mẹ.

.box + .highlight chọn **phần tử đầu tiên** có class .highlight nằm **ngay sau** một phần tử có class .box trong cùng một khối cha mẹ.

**Lưu ý:**

Bộ chọn Anh/chị/em kế chỉ chọn **phần tử kế ngay sau** phần tử đầu tiên. Nó không chọn các phần tử ở xa hơn.

Cả hai phần tử phải có **cùng một phần tử cha mẹ**.

Khoảng trắng (dấu cách, xuống dòng) giữa các phần tử không ảnh hưởng đến việc chọn.

**Ví dụ:**

HTML

<div>

<h1>Tiêu đề chính</h1>

<p>Đây là đoạn văn đầu tiên.</p>

<p>Đây là đoạn văn thứ hai.</p>

<span class="box">Nội dung trong hộp</span>

<span class="highlight">Nội dung được tô sáng</span>

</div>

CSS

h1 + p {

color: red; /\* Chỉ chọn đoạn văn đầu tiên \*/

}

.box + .highlight {

background-color: yellow; /\* Chỉ chọn nội dung được tô sáng \*/

}

**Kết quả:**

Trong ví dụ này, chỉ đoạn văn đầu tiên theo sau phần tử <h1> sẽ có màu đỏ. Đoạn văn thứ hai không bị ảnh hưởng vì nó không phải phần tử kế ngay sau.

Phần tử có class .highlight chỉ có nền vàng nếu nó nằm ngay sau một phần tử có class .box.

**Ưu điểm của bộ chọn Anh/chị/em kế:**

**Tính đặc thù:** Nó hữu ích để chọn các phần tử cụ thể dựa trên vị trí của chúng so với một phần tử khác.

**Cấu trúc rõ ràng:** Giúp duy trì cấu trúc rõ ràng trong code CSS bằng cách chọn các phần tử dựa trên mối quan hệ của chúng trong HTML.

**Nhược điểm của bộ chọn Anh/chị/em kế:**

**Giới hạn phạm vi:** Nó chỉ chọn phần tử kế ngay sau, không chọn các phần tử xa hơn trong cùng cấp.

**Tiềm ẩn mâu thuẫn:** Sử dụng quá nhiều có thể dẫn đến bộ chọn phức tạp và tiềm ẩn mâu thuẫn với các quy tắc kiểu khác.

**Các lựa chọn khác:**

**Bộ chọn Anh/chị/em tổng hợp (**~**):** Bộ chọn này chọn tất cả các phần tử cùng cấp là Anh/chị/em với phần tử được chỉ định, không chỉ phần tử kế ngay sau. Tuy nhiên, tính đặc thù của nó có thể thấp hơn.

**Bộ chọn con (**>**):** Nếu bạn cần kiểm soát nhiều hơn về thứ bậc, bạn có thể sử dụng bộ chọn con để chọn các phần tử nằm trực tiếp bên trong một phần tử khác.

**Tóm lại,** bộ chọn Anh/chị/em kế là một công cụ hữu ích để chọn các phần tử dựa trên mối quan hệ Anh/chị/em kế cận trong cấu trúc HTML. Sử dụng nó một cách chiến lược để tạo kiểu rõ ràng và cụ thể trong code CSS của bạn.

***3.4.8. Attribute selector***

**Attribute selector**, hay còn gọi là **bộ chọn thuộc tính**, là một bộ chọn CSS dùng để chọn các phần tử HTML dựa trên giá trị của thuộc tính của chúng.

**Cách thức hoạt động:**

Attribute selector được viết bằng cách sử dụng **dấu ngoặc vuông ([])** bao quanh tên thuộc tính, có thể kèm theo **toán tử** và **giá trị**.

**Ví dụ:**

[href] chọn tất cả các phần tử có thuộc tính href.

[type="button"] chọn tất cả các phần tử có thuộc tính type với giá trị là button.

[title~="Hướng dẫn"] chọn tất cả các phần tử có thuộc tính title chứa từ "Hướng dẫn".

**Toán tử:**

=: Giá trị chính xác

!=: Giá trị khác

^=: Bắt đầu bằng

$=: Kết thúc bằng

\*=: Chứa

**Lưu ý:**

Phân biệt chữ hoa chữ thường khi sử dụng tên thuộc tính và giá trị.

Giá trị phải được đặt trong ngoặc kép (") nếu nó chứa dấu cách hoặc ký tự đặc biệt.

**Ví dụ:**

HTML

<a href="https://www.google.com/">Trang chủ Google</a>

<button type="button">Nhấp vào đây</button>

<img src="image.jpg" alt="Hình ảnh" title="Hướng dẫn sử dụng">

CSS

[href] {

color: blue;

}

[type="button"] {

background-color: red;

}

[title~="Hướng dẫn"] {

border: 1px solid green;

}

**Kết quả:**

Tất cả các liên kết (a) sẽ có màu xanh lam.

Tất cả các nút (button) sẽ có nền màu đỏ.

Tất cả các phần tử có thuộc tính title chứa từ "Hướng dẫn" sẽ có viền màu xanh lục.

**Ưu điểm của attribute selector:**

**Tính linh hoạt:** Attribute selector có thể được sử dụng để chọn các phần tử không có ID hoặc class.

**Dễ sử dụng:** Việc sử dụng attribute selector rất đơn giản và dễ hiểu.

**Nhược điểm của attribute selector:**

**Tính đặc thù thấp:** Attribute selector có thể chọn nhiều phần tử trên trang, gây khó khăn khi cần định kiểu cho một phần tử cụ thể.

**Tóm lại:**

Attribute selector là một công cụ hữu ích để chọn các phần tử HTML dựa trên giá trị của thuộc tính của chúng. Việc sử dụng nó có thể giúp bạn dễ dàng định kiểu cho các nhóm phần tử liên quan mà không cần thêm ID hoặc class.

**Ngoài ra:**

Bạn có thể kết hợp attribute selector với các bộ chọn khác để tạo ra các quy tắc CSS phức tạp hơn.

Sử dụng attribute selector quá nhiều có thể khiến code CSS trở nên khó đọc và khó bảo trì.

***3.4.9. Pseudo class selector***

**Pseudo class selector**, hay còn gọi là **bộ chọn trạng thái giả**, là một bộ chọn CSS dùng để chọn các phần tử HTML dựa trên **trạng thái** của chúng, ví dụ như:

**Trạng thái chuột:** Khi người dùng di chuột qua, nhấp chuột vào, hoặc giữ chuột trên phần tử.

**Trạng thái tập trung:** Khi người dùng đang nhập dữ liệu vào phần tử.

**Trạng thái kích hoạt:** Khi người dùng đang kích hoạt một phần tử, ví dụ như nút bấm.

**Trạng thái kiểm tra:** Khi người dùng chọn hoặc bỏ chọn một ô checkbox hoặc radio button.

**Trạng thái hợp lệ/không hợp lệ:** Khi giá trị nhập vào một phần tử hợp lệ hoặc không hợp lệ.

**Cách thức hoạt động:**

Pseudo class selector được viết bằng cách sử dụng **dấu hai chấm (:)** theo sau tên trạng thái giả.

**Ví dụ:**

a:hover chọn tất cả các liên kết (a) khi người dùng di chuột qua.

input:focus chọn tất cả các phần tử nhập liệu (input) khi người dùng đang nhập dữ liệu.

button:active chọn tất cả các nút (button) khi người dùng đang kích hoạt.

**Lưu ý:**

Một số trạng thái giả chỉ áp dụng cho một số loại phần tử nhất định.

Thứ tự của pseudo class selector quan trọng. Ví dụ, a:hover:active sẽ chọn các liên kết (a) khi người dùng di chuột qua **và** đang kích hoạt.

**Ví dụ:**

HTML

<a href="#">Liên kết</a>

<input type="text" name="name">

<button type="button">Nhấp vào đây</button>

CSS

a:hover {

color: red;

}

input:focus {

border-color: blue;

}

button:active {

background-color: green;

}

**Kết quả:**

Khi người dùng di chuột qua liên kết (a), nó sẽ có màu đỏ.

Khi người dùng đang nhập dữ liệu vào phần tử nhập liệu (input), nó sẽ có viền màu xanh lam.

Khi người dùng đang kích hoạt nút (button), nó sẽ có nền màu xanh lá cây.

**Ưu điểm của pseudo class selector:**

**Tính linh hoạt:** Pseudo class selector có thể được sử dụng để chọn các phần tử dựa trên trạng thái của chúng, mà không cần thay đổi cấu trúc HTML.

**Dễ sử dụng:** Việc sử dụng pseudo class selector rất đơn giản và dễ hiểu.

**Nhược điểm của pseudo class selector:**

**Tính đặc thù thấp:** Pseudo class selector có thể chọn nhiều phần tử trên trang, gây khó khăn khi cần định kiểu cho một phần tử cụ thể.

**Tóm lại:**

Pseudo class selector là một công cụ hữu ích để chọn các phần tử HTML dựa trên trạng thái của chúng. Việc sử dụng nó có thể giúp bạn dễ dàng tạo ra các hiệu ứng tương tác và nâng cao trải nghiệm người dùng.

**Ngoài ra:**

Bạn có thể kết hợp pseudo class selector với các bộ chọn khác để tạo ra các quy tắc CSS phức tạp hơn.

Sử dụng pseudo class selector quá nhiều có thể khiến code CSS trở nên khó đọc và khó bảo trì.

***3.4.10. Group selector***

**Group selector**, hay còn gọi là **bộ chọn nhóm**, là một bộ chọn CSS dùng để chọn một nhóm các phần tử HTML dựa trên **mối quan hệ** của chúng trong **cấu trúc DOM**.

**Cách thức hoạt động:**

Group selector được viết bằng cách sử dụng **dấu phẩy (,)** để phân cách các bộ chọn.

**Ví dụ:**

h1, h2, h3 chọn tất cả các thẻ tiêu đề cấp 1, 2, 3.

p, span, a chọn tất cả các đoạn văn (p), thẻ span và liên kết (a).

**Lưu ý:**

Thứ tự của các bộ chọn trong group selector không quan trọng.

Group selector có thể được sử dụng để chọn các phần tử cùng cấp hoặc các phần tử lồng nhau.

**Ví dụ:**

HTML

<h1>Tiêu đề cấp 1</h1>

<h2>Tiêu đề cấp 2</h2>

<h3>Tiêu đề cấp 3</h3>

<p>Đây là đoạn văn.</p>

<span>Đây là thẻ span.</span>

CSS

h1, h2, h3 {

color: red;

}

p, span, a {

font-size: 16px;

}

**Kết quả:**

Tất cả các thẻ tiêu đề cấp 1, 2, 3 sẽ có màu đỏ.

Tất cả các đoạn văn, thẻ span và liên kết sẽ có cỡ chữ 16px.

**Ưu điểm của group selector:**

**Tính linh hoạt:** Group selector có thể được sử dụng để chọn nhiều loại phần tử khác nhau.

**Dễ sử dụng:** Việc sử dụng group selector rất đơn giản và dễ hiểu.

**Nhược điểm của group selector:**

**Tính đặc thù thấp:** Group selector có thể chọn nhiều phần tử trên trang, gây khó khăn khi cần định kiểu cho một phần tử cụ thể.

**Tóm lại:**

Group selector là một công cụ hữu ích để chọn một nhóm các phần tử HTML dựa trên mối quan hệ của chúng trong cấu trúc DOM. Việc sử dụng nó có thể giúp bạn dễ dàng định kiểu cho các nhóm phần tử liên quan một cách hiệu quả.

**Ngoài ra:**

Bạn có thể kết hợp group selector với các bộ chọn khác để tạo ra các quy tắc CSS phức tạp hơn.

Sử dụng group selector quá nhiều có thể khiến code CSS trở nên khó đọc và khó bảo trì.

**3.5. Đơn vị đo lường CSS**

Đơn vị đo lường là yếu tố quan trọng trong việc xác định kích thước, vị trí và các thuộc tính khác của các phần tử HTML trong CSS. Có nhiều loại đơn vị đo lường khác nhau trong CSS, mỗi loại có mục đích sử dụng riêng:

Đơn vị tuyệt đối:

px (pixel): Đơn vị phổ biến nhất, dựa trên mật độ pixel của màn hình.

cm (centimet): Đơn vị dựa trên kích thước thực tế, không phụ thuộc vào màn hình.

mm (milimet): Tương tự như cm, nhưng nhỏ hơn.

in (inch): Đơn vị đo lường inch.

Đơn vị tương đối:

em: Kích thước dựa trên font-size của phần tử cha.

rem: Kích thước dựa trên font-size của element gốc (html).

vh (viewport height): % của chiều cao viewport (phần hiển thị của trang web).

vw (viewport width): % của chiều rộng viewport.

Đơn vị khác:

% (percent): Tỷ lệ phần trăm của một giá trị khác.

deg (degree): Đơn vị đo góc.

s (second): Đơn vị đo thời gian.

Lựa chọn đơn vị đo lường:

Việc lựa chọn đơn vị đo lường phù hợp phụ thuộc vào mục đích sử dụng:

Kích thước cố định: Sử dụng px cho các element có kích thước cố định, không thay đổi theo kích thước màn hình.

Kích thước tương đối: Sử dụng em hoặc rem cho các element có kích thước thay đổi theo font-size.

Responsive design: Sử dụng vh và vw cho các layout responsive, co giãn theo kích thước màn hình.

Lưu ý:

Nên sử dụng đơn vị đo lường phù hợp để đảm bảo tính nhất quán và dễ dàng bảo trì code.

Tránh sử dụng quá nhiều đơn vị đo lường khác nhau trong cùng một trang web.

Có thể sử dụng các công cụ chuyển đổi đơn vị để chuyển đổi giữa các đơn vị đo lường khác nhau.

**3.6. Kế thừa thuộc tính**

Kế thừa thuộc tính là một tính năng quan trọng trong CSS cho phép các thuộc tính được truyền từ phần tử cha sang phần tử con. Nhờ vậy, bạn có thể định dạng một loạt các phần tử một cách dễ dàng và hiệu quả, mà không cần phải định dạng từng phần tử riêng lẻ.

Có hai loại kế thừa thuộc tính:

Kế thừa trực tiếp:

Thuộc tính được truyền từ phần tử cha trực tiếp sang phần tử con.

Ví dụ: nếu bạn đặt font-size: 16px cho element div, thì tất cả các element con của div cũng sẽ có font-size là 16px, trừ khi được định dạng riêng.

Kế thừa gián tiếp:

Thuộc tính được truyền từ phần tử cha gián tiếp qua nhiều cấp.

Ví dụ: nếu bạn đặt font-size: 16px cho element html, thì tất cả các element con của html (bao gồm cả các element con của các element con) đều sẽ có font-size là 16px, trừ khi được định dạng riêng.

Lưu ý:

Không phải tất cả các thuộc tính CSS đều được kế thừa.

Thuộc tính được kế thừa có thể bị ghi đè bởi thuộc tính được định nghĩa riêng cho phần tử con.

Thứ tự ưu tiên: Thuộc tính được định nghĩa cho phần tử con > Thuộc tính được kế thừa từ phần tử cha.

Kế thừa thuộc tính giúp:

Giảm thiểu lượng code cần viết.

Giúp code dễ đọc và dễ bảo trì hơn.

Tạo sự nhất quán trong việc định dạng trang web.

**3.7. Các nhóm thuộc tính trong CSS**

***-Thuộc tính định dạng font chữ, văn bản (font, text)***

font-family: Chọn font chữ cho element.

font-size: Kích thước font chữ.

font-weight: Độ dày font chữ.

font-style: Kiểu font chữ (normal, italic, oblique).

text-align: Căn chỉnh văn bản (left, center, right, justify).

text-decoration: Hiệu ứng trang trí văn bản (underline, overline, line-through).

text-transform: Chuyển đổi chữ hoa/chữ thường (uppercase, lowercase, capitalize).

letter-spacing: Khoảng cách giữa các chữ cái.

word-spacing: Khoảng cách giữa các từ.

line-height: Chiều cao dòng văn bản.

**-*Thuộc tính định dạng nền (background)***

background-color: Màu nền.

background-image: Hình ảnh nền.

background-repeat: Lặp lại hình ảnh nền (repeat, repeat-x, repeat-y, no-repeat).

background-position: Vị trí hình ảnh nền.

background-size: Kích thước hình ảnh nền.

background-attachment: Cố định hình ảnh nền (scroll, fixed).

**-*Mô hình hộp (box model)***

width: Chiều rộng hộp.

height: Chiều cao hộp.

margin: Khoảng cách giữa hộp và các element xung quanh.

padding: Khoảng cách giữa nội dung hộp và viền hộp.

border: Viền hộp.

***-Các thuộc tính định margin, padding, border***

margin-top: Khoảng cách trên của margin.

margin-right: Khoảng cách phải của margin.

margin-bottom: Khoảng cách dưới của margin.

margin-left: Khoảng cách trái của margin.

padding-top: Khoảng cách trên của padding.

padding-right: Khoảng cách phải của padding.

padding-bottom: Khoảng cách dưới của padding.

padding-left: Khoảng cách trái của padding.

border-width: Độ dày của viền.

border-style: Kiểu viền (solid, dotted, dashed, double).

border-color: Màu viền.

***-Thuộc tính định dạng cách hiển thị (display)***

display: Kiểu hiển thị element (block, inline, inline-block, none).

visibility: Hiển thị hoặc ẩn element (visible, hidden).

***-Thuộc tính xác định vị trí (position)***

position: Vị trí element (static, relative, absolute, fixed).

top: Vị trí trên của element.

right: Vị trí phải của element.

bottom: Vị trí dưới của element.

left: Vị trí trái của element.

**3.8. Float & Clear**  
Float & Clear là gì?

**Float** và **Clear** là hai thuộc tính CSS được sử dụng để bố trí các phần tử HTML một cách linh hoạt.

**Float:**

Thuộc tính float cho phép bạn di chuyển một phần tử sang trái hoặc phải container của nó.

**Cách sử dụng:**

float: left;: Di chuyển phần tử sang trái.

float: right;: Di chuyển phần tử sang phải.

float: none;: Bỏ hiệu ứng float.

**Lưu ý:**

Khi sử dụng float, các phần tử khác sẽ tự động điều chỉnh vị trí để phù hợp với phần tử được float.

Float có thể gây ra một số vấn đề về bố trí, ví dụ như phần tử bị che khuất hoặc container bị co lại.

**Clear:**

Thuộc tính clear cho phép bạn loại bỏ các phần tử float khỏi một khu vực cụ thể.

**Cách sử dụng:**

clear: left;: Loại bỏ các phần tử float bên trái khỏi khu vực.

clear: right;: Loại bỏ các phần tử float bên phải khỏi khu vực.

clear: both;: Loại bỏ cả các phần tử float bên trái và bên phải khỏi khu vực.

**Lưu ý:**

Clear thường được sử dụng sau khi float để đảm bảo bố trí chính xác.

**Ví dụ:**

HTML

<div class="container">

<div class="left-column" style="float: left;">

Nội dung cột trái

</div>

<div class="right-column" style="float: right;">

Nội dung cột phải

</div>

<div class="clearfix"></div>

</div>

content\_copy

CSS

.clearfix {

clear: both;

**Kết quả:**

Hai cột left-column và right-column sẽ được di chuyển sang trái và sang phải container.

Phần tử clearfix sẽ loại bỏ các phần tử float khỏi container, đảm bảo phần nội dung bên dưới hiển thị chính xác.

**Ưu điểm của Float & Clear:**

**Tính linh hoạt:** Float & Clear cho phép bạn bố trí các phần tử HTML một cách linh hoạt và sáng tạo.

**Dễ sử dụng:** Việc sử dụng Float & Clear tương đối đơn giản.

**Nhược điểm của Float & Clear:**

**Khó kiểm soát:** Float & Clear có thể gây ra một số vấn đề về bố trí nếu không được sử dụng cẩn thận.

**Lỗi trình duyệt:** Một số trình duyệt có thể hiển thị float & clear khác nhau.

**Tóm lại:**

Float & Clear là hai công cụ hữu ích để bố trí các phần tử HTML. Tuy nhiên, bạn cần sử dụng chúng cẩn thận để tránh các vấn đề về bố trí và đảm bảo khả năng tương thích với các trình duyệt khác nhau.

**Ngoài ra:**

Có nhiều cách bố trí khác trong CSS, ví dụ như flexbox và grid layout.

Sử dụng float & clear quá nhiều có thể khiến code CSS trở nên khó đọc và khó bảo trì.

**3.9.Flex**

Flex là gì?

**Flex** là viết tắt của **Flexible Box Layout Module**, một hệ thống bố trí layout linh hoạt trong CSS giúp bạn sắp xếp các phần tử HTML một cách dễ dàng và hiệu quả.

**Cách thức hoạt động:**

Flex sử dụng khái niệm **container** và **item** để bố trí các phần tử.

**Container:** là phần tử cha chứa các item.

**Item:** là các phần tử con được bố trí bên trong container.

Flex cung cấp cho bạn nhiều thuộc tính để điều khiển cách thức bố trí các item, bao gồm:

**Flex direction:** Xác định hướng bố trí các item (ngang, dọc, hoặc theo hàng).

**Justify content:** Xác định cách sắp xếp các item theo chiều ngang.

**Align items:** Xác định cách sắp xếp các item theo chiều dọc.

**Align content:** Xác định cách sắp xếp các item trong container.

**Ví dụ:**

HTML

<div class="container">

<div class="item">Item 1</div>

<div class="item">Item 2</div>

<div class="item">Item 3</div>

</div>

CSS

.container {

display: flex;

flex-direction: row;

justify-content: space-between;

align-items: center;

}

.item {

background-color: red;

padding: 10px;

margin: 5px;

}

**Kết quả:**

Ba item sẽ được bố trí theo hàng ngang trong container.

Các item sẽ được cách đều nhau theo chiều ngang.

Các item sẽ được căn giữa theo chiều dọc.

**Ưu điểm của Flex:**

**Tính linh hoạt:** Flex cho phép bạn bố trí các phần tử một cách linh hoạt và dễ dàng.

**Dễ sử dụng:** Việc sử dụng Flex tương đối đơn giản.

**Khả năng tương thích:** Flex được hỗ trợ bởi hầu hết các trình duyệt hiện đại.

**Nhược điểm của Flex:**

**Hỗ trợ trình duyệt:** Flex không được hỗ trợ bởi các trình duyệt cũ.

**Độ phức tạp:** Flex có thể trở nên phức tạp khi sử dụng các thuộc tính nâng cao.

**Tóm lại:**

Flex là một hệ thống bố trí layout linh hoạt và hiệu quả trong CSS. Sử dụng Flex giúp bạn dễ dàng tạo ra các giao diện web đẹp mắt và đáp ứng nhiều thiết bị khác nhau.

**3.10. Grid**

Grid là gì?

**Grid** là viết tắt của **CSS Grid Layout**, một hệ thống bố trí layout mới trong CSS giúp bạn sắp xếp các phần tử HTML một cách trực quan và hiệu quả.

**Cách thức hoạt động:**

Grid sử dụng khái niệm **grid** (lưới) để bố trí các phần tử. Lưới này được tạo thành bởi các **hàng** và **cột**.

**Hàng:** là các đường ngang chia lưới thành các phần.

**Cột:** là các đường dọc chia lưới thành các phần.

Bạn có thể đặt các phần tử vào các ô trong lưới. Mỗi ô được xác định bởi vị trí hàng và cột của nó.

Grid cung cấp cho bạn nhiều thuộc tính để điều khiển cách thức bố trí các phần tử, bao gồm:

**Grid-template-columns:** Xác định số lượng và kích thước của các cột.

**Grid-template-rows:** Xác định số lượng và kích thước của các hàng.

**Grid-area:** Xác định vị trí của một phần tử trong lưới.

**Justify-content:** Xác định cách sắp xếp các phần tử trong một cột.

**Align-items:** Xác định cách sắp xếp các phần tử trong một hàng.

**Ví dụ:**

HTML

<div class="container">

<div class="item">Item 1</div>

<div class="item">Item 2</div>

<div class="item">Item 3</div>

</div>

CSS

.container {

display: grid;

grid-template-columns: repeat(3, 1fr);

grid-template-rows: repeat(2, auto);

gap: 10px;

}

.item {

background-color: red;

padding: 10px;

}

**Kết quả:**

Ba item sẽ được bố trí thành 3 cột và 2 hàng trong container.

Các item sẽ có kích thước bằng nhau và được cách đều nhau bởi khoảng cách 10px.

**Ưu điểm của Grid:**

**Tính linh hoạt:** Grid cho phép bạn bố trí các phần tử một cách linh hoạt và dễ dàng.

**Dễ sử dụng:** Việc sử dụng Grid tương đối đơn giản.

**Khả năng tương thích:** Grid được hỗ trợ bởi hầu hết các trình duyệt hiện đại.

**Bố trí phức tạp:** Grid giúp bạn tạo ra các bố trí phức tạp một cách dễ dàng.

**Nhược điểm của Grid:**

**Hỗ trợ trình duyệt:** Grid không được hỗ trợ bởi các trình duyệt cũ.

**Độ phức tạp:** Grid có thể trở nên phức tạp khi sử dụng các thuộc tính nâng cao.

**Tóm lại:**

Grid là một hệ thống bố trí layout mạnh mẽ và linh hoạt trong CSS. Sử dụng Grid giúp bạn dễ dàng tạo ra các giao diện web đẹp mắt và đáp ứng nhiều thiết bị khác nhau.

Flexbox là một hệ thống bố trí layout khác cũng phổ biến, tuy nhiên Grid được xem là giải pháp hiện đại và linh hoạt hơn.

**3.11. CSS3**

***-CSS3 là gì?***

**CSS3** là phiên bản mới nhất của Cascading Style Sheets (CSS), ngôn ngữ được sử dụng để định dạng và trình bày giao diện của trang web. CSS3 bổ sung nhiều tính năng mới và cải thiện so với phiên bản trước đó, CSS2, đồng thời cung cấp nhiều lợi ích vượt trội trong việc thiết kế và trình bày trang web.

**Điểm nổi bật của CSS3:**

**Bố cục mới:** Flexbox và Grid layout giúp bố trí các phần tử HTML một cách linh hoạt và hiệu quả.

**Hiệu ứng**: Cung cấp các hiệu ứng chuyển tiếp, hoạt ảnh và bóng đổ mượt mà.

**Media Queries**: Cho phép điều chỉnh giao diện trang web phù hợp với các thiết bị khác nhau như máy tính, điện thoại thông minh, máy tính bảng.

**Chọn lọc**: Cung cấp bộ chọn thuộc tính và bộ chọn trạng thái giả để chọn các phần tử HTML một cách chính xác.

**Phông chữ**: Hỗ trợ nhiều kiểu phông chữ web và các tính năng nâng cao như định dạng OpenType.

**Màu sắc**: Cung cấp các mô hình màu mới như RGBA và HSL.

**Lợi ích của việc sử dụng CSS3:**

**Tăng khả năng tương thích**: Hỗ trợ hầu hết các trình duyệt hiện đại.

**Cải thiện hiệu suất**: Giảm thời gian tải trang web.

**Tăng khả năng truy cập**: Tạo giao diện dễ tiếp cận cho người dùng có khiếm khuyết.

**Tăng tính thẩm mỹ**: Tạo giao diện đẹp mắt và thu hút.

**Tuy nhiên, CSS3 cũng có một số hạn chế:**

**Hỗ trợ trình duyệt**: Một số tính năng mới có thể không được hỗ trợ bởi các trình duyệt cũ.

**Độ phức tạp**: Việc sử dụng các tính năng nâng cao có thể trở nên phức tạp.

**Tóm lại:**

CSS3 là một công cụ mạnh mẽ giúp bạn tạo ra các trang web đẹp mắt, hiệu quả và tương thích với nhiều thiết bị. Việc sử dụng CSS3 có thể giúp bạn nâng cao trải nghiệm người dùng và cải thiện chất lượng trang web của bạn.

**Ngoài ra:**

Có nhiều tài nguyên online để học thêm về CSS3, ví dụ như W3Schools: <https://www.w3schools.com/css/default.asp>.

Nên sử dụng CSS3 kết hợp với HTML5 để tạo ra các trang web hiện đại và tối ưu.

***-CSS3 Selectors***

Selectors là gì?

**Selectors** là một phần quan trọng trong CSS, giúp bạn chọn các phần tử HTML cụ thể để áp dụng các kiểu định dạng. Nó giống như một công cụ nhắm mục tiêu, cho phép bạn kiểm soát chính xác cách hiển thị các phần tử trên trang web.

**Có nhiều loại Selectors khác nhau, mỗi loại có chức năng riêng:**

**Selectors theo kiểu:**

Chọn các phần tử HTML dựa trên tên thẻ của chúng, ví dụ như <h1>, <p>, <div>, v.v.

Ví dụ: h1 { color: red; } - Lập tất cả các thẻ <h1> có màu đỏ.

**Selectors theo class:**

Chọn các phần tử HTML có class cụ thể được chỉ định bằng thuộc tính class.

Ví dụ: <p class="important">Đây là văn bản quan trọng.</p>. Sau đó trong CSS: .important { font-weight: bold; } - Lập văn bản in đậm trong đoạn văn có class "important".

**Selectors theo ID:**

Chọn một phần tử duy nhất trên trang bằng cách sử dụng thuộc tính id. ID phải là duy nhất trong một tài liệu.

Ví dụ: <h1 id="main-heading">Chào mừng!</h1>. Sau đó trong CSS: #main-heading { color: blue; } - Lập kiểu chỉ cho phần tử có ID "main-heading".

**Selectors theo con cháu:**

Chọn các phần tử dựa trên mối quan hệ của chúng trong cấu trúc HTML.

Ví dụ: div p { text-decoration: underline; } - Gạch chân tất cả các đoạn văn là con của một div.

***-Universal Selector:***

Chọn tất cả các phần tử trên trang (sử dụng cẩn thận để đảm bảo tính đặc biệt).

Ví dụ: \* { margin: 0; padding: 0; } - Khôi phục cài đặt ban đầu để đặt lề và khoảng đệm ban đầu thành 0 cho tất cả các phần tử.

**Group Selectors:**

Chọn nhiều phần tử bằng cách phân cách các Selectors bằng dấu phẩy.

Ví dụ: h1, p { font-family: Arial, sans-serif; } - Cài đặt phông chữ cho cả <h1> và <p>.

**Pseudo-Classes:**

Chọn các phần tử dựa trên trạng thái hoặc điều kiện cụ thể.

Ví dụ phổ biến:

:hover: Áp dụng kiểu khi chuột di chuột qua một phần tử.

:focus: Áp dụng kiểu khi một phần tử được chọn (ví dụ: cho các trường biểu mẫu).

:active: Áp dụng kiểu khi một phần tử được kích hoạt (ví dụ: nhấp chuột vào nút).

**Pseudo-Elements:**

Chọn các phần cụ thể của một phần tử, như trước hoặc sau nội dung.

Ví dụ phổ biến:

::before: Chèn nội dung trước nội dung của phần tử.

::after: Chèn nội dung sau nội dung của phần tử.

**Lựa chọn Selectors phù hợp:**

Khi chọn các phần tử, hãy cân nhắc các yếu tố như tính đặc biệt (các Selectors cụ thể hơn được ưu tiên) và khả năng bảo trì (sử dụng tên class rõ ràng và mô tả).

***-CSS3 Rounded Corners***   
CSS3 Rounded Corners là gì?

**CSS3 Rounded Corners** là kỹ thuật sử dụng thuộc tính border-radius trong CSS3 để tạo các góc bo tròn cho các phần tử HTML. Thuộc tính này giúp bạn tạo ra các thiết kế đẹp mắt và mượt mà hơn cho trang web của mình.

**Cách sử dụng** border-radius**:**

**Cú pháp cơ bản:**

CSS

border-radius: <bán kính> ;

**Ví dụ:**

CSS

div {

border-radius: 10px;

}

**Cấu hình nâng cao:**

**Bán kính riêng cho từng góc:**

CSS

border-radius: <top-left> <top-right> <bottom-right> <bottom-left>;

Hãy [thận trọng](https://gemini.google.com/faq#coding) khi sử dụng các đoạn mã.

content\_copy

**Ví dụ:**

CSS

div {

border-radius: 10px 20px 30px 40px;

}

Hãy [thận trọng](https://gemini.google.com/faq#coding) khi sử dụng các đoạn mã.

content\_copy

**Đơn vị:**

Pixel (px)

Phần trăm (%)

Em (em)

Rem (rem)

**Trình duyệt hỗ trợ:**

Hầu hết các trình duyệt hiện đại đều hỗ trợ border-radius.

Tuy nhiên, một số trình duyệt cũ hơn có thể không hỗ trợ đầy đủ.

**Lợi ích:**

Tạo ra các thiết kế đẹp mắt và mượt mà hơn.

Giúp trang web của bạn trông hiện đại và chuyên nghiệp hơn.

Dễ dàng sử dụng và tùy chỉnh.

**Hạn chế:**

Có thể ảnh hưởng đến hiệu suất trang web nếu sử dụng quá nhiều.

Một số trình duyệt cũ hơn có thể không hỗ trợ đầy đủ.

**Ví dụ sử dụng:**

Tạo các nút bo tròn.

Tạo các thẻ bo tròn.

Tạo các hình ảnh bo tròn.

Tạo các khung bo tròn.

**Tài liệu tham khảo:**

CSS3 border-radius: [đã xoá URL không hợp lệ]

CSS3 Rounded Corners: [đã xoá URL không hợp lệ]

**Kết luận:**

**CSS3 Rounded Corners** là một tính năng hữu ích giúp bạn tạo ra các thiết kế đẹp mắt và mượt mà hơn cho trang web của mình. Hãy sử dụng tính năng này để nâng cao giao diện và trải nghiệm người dùng cho trang web của bạn.

**-*CSS3 Border Images***  
Border Images là gì?

**Border Images** là một tính năng CSS3 cho phép bạn sử dụng hình ảnh để tạo đường viền cho các phần tử HTML. Nó cung cấp nhiều khả năng hơn so với đường viền truyền thống, bao gồm:

**Tính linh hoạt:** Sử dụng bất kỳ hình ảnh nào để tạo đường viền, mang lại sự đa dạng và sáng tạo.

**Tùy chỉnh:** Cắt, chia tỷ lệ và lặp lại hình ảnh để phù hợp với kích thước và vị trí đường viền mong muốn.

**Hiệu ứng độc đáo:** Tạo đường viền với các họa tiết, hoa văn hoặc hiệu ứng đặc biệt.

**Cách sử dụng Border Images:**

Để sử dụng Border Images, bạn cần:

**Hình ảnh:** Chuẩn bị hình ảnh bạn muốn sử dụng làm đường viền.

**Thuộc tính CSS:** Sử dụng thuộc tính border-image trong CSS để xác định hình ảnh, vị trí cắt, cách lặp lại và các thuộc tính khác.

**Cú pháp cơ bản:**

CSS

border-image: url(image.png) [repeat] [slice] [origin];

url(image.png): Đường dẫn đến hình ảnh.

repeat: Xác định cách lặp lại hình ảnh (ví dụ: repeat, repeat-x, repeat-y, no-repeat).

slice: Xác định phần nào của hình ảnh được sử dụng cho các cạnh (ví dụ: 4 4 4 4).

origin: Xác định vị trí bắt đầu của hình ảnh trên đường viền (ví dụ: content-box, border-box).

**Ví dụ:**

HTML

<div class="box">

Nội dung

</div>

CSS

.box {

border-image: url(border.png) repeat round content-box;

}

**Kết quả:**

Hình ảnh border.png được sử dụng để tạo đường viền cho .box.

Hình ảnh được lặp lại trên cả bốn cạnh.

Các phần góc của hình ảnh được bo tròn.

Hình ảnh bắt đầu từ mép ngoài của hộp nội dung.

**Lợi ích của Border Images:**

**Tính thẩm mỹ:** Tạo giao diện đẹp mắt và độc đáo.

**Tính linh hoạt:** Dễ dàng điều chỉnh để phù hợp với thiết kế của bạn.

**Hiệu suất:** Có thể hiệu quả hơn so với các phương pháp tạo đường viền khác.

**Hạn chế của Border Images:**

**Hỗ trợ trình duyệt:** Một số trình duyệt cũ không hỗ trợ Border Images.

**Độ phức tạp:** Việc sử dụng Border Images có thể phức tạp hơn so với đường viền truyền thống.

**Tóm lại:**

Border Images là một tính năng CSS3 mạnh mẽ cho phép bạn tạo ra các đường viền đẹp mắt và độc đáo. Tuy nhiên, nó có thể phức tạp hơn so với đường viền truyền thống và không được hỗ trợ bởi tất cả các trình duyệt.

**-*CSS3 Backgrounds***

## CSS3 Backgrounds là gì?

**CSS3 Backgrounds** là tập hợp các thuộc tính trong CSS3 giúp bạn định dạng nền (background) cho các phần tử HTML. Với CSS3 Backgrounds, bạn có thể:

**Thêm hình ảnh nền:** Sử dụng thuộc tính background-image để thêm hình ảnh vào background.

**Định vị hình ảnh nền:** Sử dụng thuộc tính background-position để xác định vị trí của hình ảnh nền.

**Lặp lại hình ảnh nền:** Sử dụng thuộc tính background-repeat để xác định cách thức lặp lại hình ảnh nền.

**Cài đặt kích thước hình ảnh nền:** Sử dụng thuộc tính background-size để xác định kích thước của hình ảnh nền.

**Thêm màu nền:** Sử dụng thuộc tính background-color để thêm màu nền cho phần tử.

**Tạo hiệu ứng gradient:** Sử dụng thuộc tính background-image với các giá trị gradient để tạo hiệu ứng chuyển đổi màu mượt mà.

**Sử dụng nhiều lớp nền:** Sử dụng nhiều thuộc tính background để tạo ra các lớp nền chồng lên nhau.

**Lợi ích:**

**Tạo ra các thiết kế background đẹp mắt và ấn tượng.**

**Giúp trang web của bạn trông hiện đại và chuyên nghiệp hơn.**

**Dễ dàng sử dụng và tùy chỉnh.**

**Hạn chế:**

**Có thể ảnh hưởng đến hiệu suất trang web nếu sử dụng quá nhiều.**

**Một số trình duyệt cũ hơn có thể không hỗ trợ đầy đủ.**

**Ví dụ sử dụng:**

**Thêm hình ảnh nền cho trang web.**

**Tạo hiệu ứng gradient cho background.**

**Sử dụng nhiều lớp nền để tạo chiều sâu cho thiết kế.**

**Tài liệu tham khảo:**

CSS3 Backgrounds: [đã xoá URL không hợp lệ]

CSS Backgrounds: [đã xoá URL không hợp lệ]

**Kết luận:**

**CSS3 Backgrounds** là một công cụ mạnh mẽ giúp bạn tạo ra các thiết kế background đẹp mắt và ấn tượng cho trang web của mình. Hãy sử dụng công cụ này để nâng cao giao diện và trải nghiệm người dùng cho trang web của bạn.

Backgrounds là gì?

**Backgrounds** trong CSS là một tính năng cho phép bạn định dạng phần nền của một phần tử HTML. Nó bao gồm nhiều khía cạnh như:

**Màu sắc:**

Sử dụng màu đơn sắc hoặc gradient để tạo nền đơn giản hoặc hiệu ứng chuyển tiếp mượt mà.

Ví dụ: background-color: red; hoặc background-image: linear-gradient(to right, red, blue);

**Hình ảnh:**

Sử dụng hình ảnh để tạo nền với các họa tiết, hoa văn hoặc ảnh thực tế.

Ví dụ: background-image: url(background.jpg);

**Vị trí:**

Xác định vị trí của hình ảnh nền trên phần tử.

Ví dụ: background-position: top left; hoặc background-position: center center;

**Kích thước:**

Xác định kích thước của hình ảnh nền.

Ví dụ: background-size: cover; hoặc background-size: 100px 100px;

**Lặp lại:**

Xác định cách hình ảnh nền được lặp lại trên phần tử.

Ví dụ: background-repeat: repeat; hoặc background-repeat: no-repeat;

**Attachment:**

Xác định cách hình ảnh nền di chuyển khi phần tử được cuộn.

Ví dụ: background-attachment: fixed; hoặc background-attachment: scroll;

**Multiple Backgrounds:**

Sử dụng nhiều hình ảnh hoặc màu sắc để tạo nền phức tạp hơn.

Ví dụ: background: linear-gradient(to right, red, blue), url(background.jpg) no-repeat center center;

**Lợi ích của Backgrounds:**

**Tính thẩm mỹ:** Tạo giao diện đẹp mắt và thu hút.

**Tính linh hoạt:** Dễ dàng điều chỉnh để phù hợp với thiết kế của bạn.

**Thông tin:** Cung cấp thêm thông tin hoặc ngữ cảnh cho nội dung.

**Hạn chế của Backgrounds:**

**Độ phức tạp:** Việc sử dụng Backgrounds phức tạp có thể ảnh hưởng đến hiệu suất trang web.

**Hỗ trợ trình duyệt:** Một số trình duyệt cũ không hỗ trợ tất cả các tính năng Backgrounds.

**Tóm lại:**

Backgrounds là một tính năng CSS mạnh mẽ cho phép bạn tạo ra các nền đẹp mắt và thu hút. Tuy nhiên, nó có thể phức tạp hơn so với các phương pháp định dạng truyền thống và không được hỗ trợ bởi tất cả các trình duyệt.

***- CSS Gradients***

## Gradients là gì?

**Gradients** là hiệu ứng chuyển tiếp mượt mà giữa các màu sắc trong CSS. Nó mang đến nhiều lợi ích và được sử dụng rộng rãi để:

**Tạo hiệu ứng thị giác:**

Tăng tính thẩm mỹ và thu hút cho website.

Tạo điểm nhấn cho các phần tử quan trọng.

Mang đến cảm giác chuyển động và mượt mà.

**Phù hợp với nhiều mục đích:**

Sử dụng cho nền, đường viền, bóng đổ, văn bản, v.v.

Tạo hiệu ứng 3D, ánh sáng, phản chiếu, v.v.

Phù hợp với nhiều phong cách thiết kế khác nhau.

**Dễ dàng sử dụng:**

Cú pháp đơn giản với nhiều tùy chỉnh.

Hỗ trợ bởi hầu hết các trình duyệt hiện đại.

**Có hai loại Gradients chính:**

**Linear Gradients:**

Chuyển đổi màu sắc theo đường thẳng.

Xác định điểm bắt đầu và kết thúc, cũng như các màu sắc dọc theo đường chuyển đổi.

Ví dụ: background-image: linear-gradient(to right, red, blue);

**Radial Gradients:**

Chuyển đổi màu sắc theo hình tròn.

Xác định điểm trung tâm, bán kính và các màu sắc dọc theo gradient.

Ví dụ: background-image: radial-gradient(center, circle, red, blue);

**Ngoài ra, bạn có thể:**

Sử dụng nhiều điểm màu để tạo chuyển đổi phức tạp hơn.

Sử dụng các thuộc tính như background-position và background-size để điều chỉnh vị trí và kích thước của gradient.

Kết hợp gradient với các hiệu ứng CSS khác để tạo ra giao diện độc đáo.

**Lợi ích của Gradients:**

**Tính thẩm mỹ:** Tạo giao diện đẹp mắt và thu hút.

**Tính linh hoạt:** Dễ dàng điều chỉnh để phù hợp với thiết kế của bạn.

**Hiệu suất:** Ít tốn tài nguyên hơn so với hình ảnh.

**Hạn chế của Gradients:**

**Hỗ trợ trình duyệt:** Một số trình duyệt cũ không hỗ trợ gradients.

**Độ phức tạp:** Việc sử dụng gradients phức tạp có thể ảnh hưởng đến hiệu suất trang web.

**Tóm lại:**

Gradients là một tính năng CSS mạnh mẽ cho phép bạn tạo ra các hiệu ứng chuyển tiếp mượt mà và đẹp mắt giữa các màu sắc. Tuy nhiên, nó có thể phức tạp hơn so với các phương pháp truyền thống và không được hỗ trợ bởi tất cả các trình duyệt.

***-CSS3 Shadows***

**Shadows** (bóng đổ) là một tính năng CSS3 cho phép bạn tạo ra hiệu ứng bóng đổ cho các phần tử HTML. Bóng đổ có thể giúp:

**Tăng tính thẩm mỹ:** Tạo giao diện đẹp mắt và thu hút.

**Cung cấp chiều sâu:** Giúp các phần tử nổi bật và tạo cảm giác 3D.

**Nhấn mạnh các yếu tố:** Hướng sự chú ý của người dùng đến các phần tử quan trọng.

**Có hai loại bóng đổ chính:**

**Box Shadows:**

Tạo bóng đổ xung quanh các cạnh của một phần tử.

Xác định vị trí, độ lan tỏa, độ mờ và màu sắc của bóng đổ.

Ví dụ: box-shadow: 0 0 10px rgba(0, 0, 0, 0.5);

**Text Shadows:**

Tạo bóng đổ cho văn bản.

Xác định vị trí, độ lan tỏa, độ mờ và màu sắc của bóng đổ.

Ví dụ: text-shadow: 2px 2px 4px rgba(0, 0, 0, 0.5);

**Ngoài ra, bạn có thể:**

Sử dụng nhiều bóng đổ để tạo hiệu ứng phức tạp hơn.

Sử dụng các thuộc tính như box-shadow-inset và text-shadow-inset để tạo bóng đổ bên trong phần tử hoặc văn bản.

Kết hợp bóng đổ với các hiệu ứng CSS khác để tạo ra giao diện độc đáo.

**Lợi ích của Shadows:**

**Tính thẩm mỹ:** Tạo giao diện đẹp mắt và thu hút.

**Tính linh hoạt:** Dễ dàng điều chỉnh để phù hợp với thiết kế của bạn.

**Hiệu suất:** Ít tốn tài nguyên hơn so với hình ảnh.

**Hạn chế của Shadows:**

**Hỗ trợ trình duyệt:** Một số trình duyệt cũ không hỗ trợ shadows.

**Độ phức tạp:** Việc sử dụng shadows phức tạp có thể ảnh hưởng đến hiệu suất trang web.

**Tóm lại:**

Shadows là một tính năng CSS mạnh mẽ cho phép bạn tạo ra các hiệu ứng bóng đổ đẹp mắt và thu hút. Tuy nhiên, nó có thể phức tạp hơn so với các phương pháp truyền thống và không được hỗ trợ bởi tất cả các trình duyệt.

**Ví dụ:**

HTML

<div class="shadow-box">

Nội dung

</div>

CSS

.shadow-box {

box-shadow: 0 0 10px rgba(0, 0, 0, 0.5);

height: 200px;

width: 200px;

}

**Kết quả:**

.shadow-box sẽ có hiệu ứng bóng đổ xung quanh các cạnh.

Kích thước của hộp là 200px x 200px.

**Bạn có thể thử nghiệm với shadows bằng cách sử dụng các công cụ trực tuyến như CSS Shadow Generator:** [đã xoá URL không hợp lệ]

**- *CSS3 Fonts***

## Font là gì?

**Font** (hay còn gọi là **Phông chữ**) là tập hợp hoàn chỉnh các ký tự (character) bao gồm: chữ cái (letter), số, dấu câu và kí tự đặc biệt khác (glyph) được thiết kế theo một phong cách thống nhất. Font được sử dụng để hiển thị văn bản trên màn hình hoặc in ấn.

**Có hai loại font chính:**

**1. Font Raster:**

Được tạo thành từ các pixel, có độ phân giải cố định.

Phù hợp cho hiển thị trên màn hình vì có kích thước nhỏ và dễ tải.

Tuy nhiên, font raster có thể bị vỡ hình khi phóng to.

**2. Font Vector:**

Được tạo thành từ các đường cong và điểm, có thể được phóng to mà không bị vỡ hình.

Phù hợp cho in ấn vì có chất lượng cao và sắc nét.

Tuy nhiên, font vector có kích thước lớn hơn font raster và có thể khó tải hơn.

**Ngoài ra, font còn được phân loại theo nhiều kiểu khác nhau như:**

**Serif:** Chữ có chân ở cuối các nét. Ví dụ: Times New Roman, Garamond.

**Sans-serif:** Chữ không có chân ở cuối các nét. Ví dụ: Arial, Helvetica, Calibri.

**Script:** Chữ được thiết kế để mô phỏng chữ viết tay. Ví dụ: Brush Script MT, Lucida Handwriting.

**Decorative:** Chữ được thiết kế với các hiệu ứng trang trí. Ví dụ: Comic Sans MS, Papyrus.

**Lựa chọn font phù hợp:**

Cần cân nhắc mục đích sử dụng, đối tượng người đọc và phong cách thiết kế chung.

Ví dụ, font serif dễ đọc cho văn bản dài, font sans-serif phù hợp cho văn bản trên màn hình, font script mang lại cảm giác trang trọng, font decorative phù hợp cho các tiêu đề.

**Tìm kiếm và cài đặt font:**

Có nhiều trang web cung cấp font miễn phí và trả phí. Ví dụ: Google Fonts, Adobe Fonts.

Sau khi tải font về, bạn cần cài đặt vào hệ thống để sử dụng.

**Sử dụng font trong CSS:**

Sử dụng thuộc tính font-family để指定font cho phần tử HTML.

Ví dụ: p { font-family: Arial, sans-serif; }

**Kết luận:**

Font là một yếu tố quan trọng trong thiết kế web và in ấn. Lựa chọn font phù hợp sẽ giúp nâng cao tính thẩm mỹ và hiệu quả truyền tải thông tin.

***- CSS3 2D Transforms***

CSS3 2D Transforms là gì?

**CSS3 2D Transforms** là tập hợp các thuộc tính cho phép bạn thao tác với hình ảnh trực quan của các phần tử HTML trên mặt phẳng hai chiều. Chúng cung cấp cách thức nhẹ và linh hoạt để đạt được nhiều hiệu ứng khác nhau mà không cần sửa đổi cấu trúc HTML cơ bản hoặc sử dụng hình ảnh bổ sung.

**Dưới đây là một số 2D Transforms phổ biến:**

**Translate:**

Di chuyển một phần tử dọc theo trục X và Y. Hãy tưởng tượng việc trượt một phần tử trên một mặt phẳng.

Cú pháp: transform: translate(x-axis, y-axis);

Ví dụ: transform: translate(50px, 100px); - Di chuyển phần tử 50 pixel sang phải và 100 pixel xuống.

**Rotate:**

Xoay một phần tử xung quanh một điểm cố định (mặc định là tâm của phần tử). Hãy nghĩ đến việc xoay một phần tử trên một mặt phẳng.

Cú pháp: transform: rotate(angle);

Ví dụ: transform: rotate(45deg); - Xoay phần tử 45 độ theo chiều kim đồng hồ.

**Scale:**

Thay đổi kích thước (tăng hoặc giảm) của một phần tử dọc theo trục X và Y. Hãy tưởng tượng việc kéo giãn hoặc thu nhỏ một phần tử trên một mặt phẳng.

Cú pháp: transform: scale(scaleX, scaleY); (mặc định scaleY bằng scaleX nếu chỉ cung cấp một giá trị)

Ví dụ: transform: scale(2, 0.5); - Thay đổi kích thước phần tử theo chiều ngang lên 200% (gấp đôi kích thước) và theo chiều dọc xuống 50% (nửa kích thước).

**Skew:**

Nghiêng một phần tử trên trục X và Y. Hãy tưởng tượng việc nghiêng một phần tử trên một mặt phẳng.

Cú pháp: transform: skew(skewX, skewY);

Ví dụ: transform: skew(10deg, 5deg); - Nghiêng phần tử 10 độ trên trục X và 5 độ trên trục Y.

**Kết hợp Transforms:**

Điều tuyệt vời của 2D Transforms là bạn có thể kết hợp chúng để tạo ra các hiệu ứng hoạt hình và hiệu ứng phức tạp hơn. Thứ tự khai báo có thể ảnh hưởng đến kết quả biến đổi. Thử nghiệm là chìa khóa!

**Lợi ích của 2D Transforms:**

**Hiệu quả:** Nhẹ và hoạt động tốt trên hầu hết các trình duyệt, phù hợp với nhiều phần tử trang web.

**Linh hoạt:** Cung cấp nhiều hiệu ứng hình ảnh, từ thay đổi vị trí đơn giản đến hoạt hình phức tạp.

**Khả năng tiếp cận:** Có thể được sử dụng để nâng cao trải nghiệm người dùng cho người khuyết tật, chẳng hạn như tạo hiệu ứng di chuột cho nút.

**Trường hợp sử dụng 2D Transforms:**

Tạo hiệu ứng di chuột cho các nút và menu, giúp chúng tương tác hơn.

Hoạt hình đầu vào và đầu ra của phần tử, thêm điểm nhấn cho trang web của bạn.

Xây dựng các phần tử tương tác như thanh trượt và băng chuyền cho thư viện ảnh hoặc giới thiệu sản phẩm.

Vị trí các phần tử một cách linh hoạt dựa trên kích thước màn hình, đảm bảo trải nghiệm người dùng nhất quán trên các thiết bị.

**Tìm hiểu thêm về 2D Transforms:**

Bằng cách sử dụng 2D Transforms hiệu quả, bạn có thể thêm tính năng động và tương tác vào trang web của mình, khiến chúng trở nên hấp dẫn và thu hút người dùng hơn

**-*CSS3 3D Transforms***

CSS3 3D Transforms là gì?

**CSS3 3D Transforms** là tập hợp các thuộc tính CSS cho phép bạn thao tác với vị trí, hình dạng và hướng của các phần tử HTML trong không gian ba chiều. Nó mở ra nhiều khả năng mới cho thiết kế web, cho phép bạn tạo ra các hiệu ứng và giao diện phức tạp và ấn tượng hơn.

**Dưới đây là một số tính năng chính của CSS3 3D Transforms:**

**Di chuyển (Translate):**

Di chuyển phần tử dọc theo trục X, Y và Z.

Cú pháp: transform: translate3d(x, y, z);

**Xoay (Rotate):**

Xoay phần tử xung quanh trục X, Y và Z.

Cú pháp: transform: rotate3d(x, y, z, angle);

**Thay đổi kích thước (Scale):**

Thay đổi kích thước phần tử theo trục X, Y và Z.

Cú pháp: transform: scale3d(x, y, z);

**Nghiêng (Skew):**

Nghiêng phần tử trên mặt phẳng XY và XZ.

Cú pháp: transform: skewX(angle) skewY(angle);

**Perspective:**

Tạo hiệu ứng phối cảnh, giúp các phần tử trông giống như có chiều sâu.

Cú pháp: transform: perspective(distance);

**Phối hợp các biến đổi:**

Kết hợp các biến đổi 3D để tạo ra các hiệu ứng phức tạp và ấn tượng.

**Lợi ích của CSS3 3D Transforms:**

**Tính linh hoạt:** Cho phép bạn tạo ra các hiệu ứng 3D đẹp mắt và độc đáo.

**Hiệu suất:** Hiệu quả hơn so với các phương pháp JavaScript truyền thống để tạo hiệu ứng 3D.

**Khả năng tiếp cận:** Có thể được sử dụng để nâng cao trải nghiệm người dùng cho người khuyết tật.

**Trường hợp sử dụng CSS3 3D Transforms:**

Tạo các hiệu ứng di chuột và chuyển tiếp mượt mà.

Hiển thị các mô hình 3D và đồ họa.

Tạo các trò chơi và ứng dụng web tương tác.

Thiết kế các giao diện web phức tạp và ấn tượng.

**Để tìm hiểu thêm về CSS3 3D Transforms:**

Xem các ví dụ trực quan: [đã xoá URL không hợp lệ]

**Lưu ý:**

Hỗ trợ trình duyệt cho CSS3 3D Transforms khá tốt, nhưng vẫn có một số trình duyệt cũ không hỗ trợ đầy đủ.

Việc sử dụng quá nhiều biến đổi 3D có thể ảnh hưởng đến hiệu suất trang web.

**Kết luận:**

CSS3 3D Transforms là một công cụ mạnh mẽ cho phép bạn tạo ra các hiệu ứng 3D đẹp mắt và ấn tượng trên trang web. Tuy nhiên, nó cũng có thể phức tạp và đòi hỏi kiến thức chuyên môn.

***-CSS3 Transitions***

**CSS3 Transitions** là một công cụ cho phép bạn tạo hiệu ứng chuyển tiếp mượt mà giữa các trạng thái khác nhau của một phần tử HTML. Nó hoạt động bằng cách thay đổi giá trị của các thuộc tính CSS theo thời gian, tạo hiệu ứng chuyển động tự nhiên và đẹp mắt.

**Dưới đây là một số tính năng chính của CSS3 Transitions:**

**Thuộc tính:** transition

**Thành phần:**

transition-property: Xác định thuộc tính nào sẽ được chuyển đổi.

transition-duration: Xác định thời lượng của quá trình chuyển đổi.

transition-timing-function: Xác định tốc độ của quá trình chuyển đổi.

transition-delay: Xác định thời gian chờ trước khi bắt đầu quá trình chuyển đổi.

**Ví dụ:**

CSS

.button {

background-color: red;

transition: background-color 0.5s ease-in-out;

}

.button:hover {

background-color: blue;

}

Trong ví dụ này, khi người dùng di chuột qua nút, màu nền sẽ chuyển đổi từ đỏ sang xanh lam trong 0.5 giây với tốc độ tăng tốc và giảm tốc mượt mà.

**Lợi ích của CSS3 Transitions:**

**Cải thiện trải nghiệm người dùng:** Tạo hiệu ứng chuyển tiếp mượt mà và trực quan, giúp nâng cao trải nghiệm người dùng.

**Tăng tính tương tác:** Giúp trang web trở nên sống động và tương tác hơn.

**Giảm thiểu sử dụng JavaScript:** Có thể tạo ra các hiệu ứng đơn giản mà không cần sử dụng JavaScript.

**Trường hợp sử dụng CSS3 Transitions:**

Tạo hiệu ứng di chuột cho các nút, menu và các phần tử khác.

Thêm hiệu ứng chuyển tiếp cho các hình ảnh, thanh trượt và các nội dung khác.

Tạo các hiệu ứng hoạt hình đơn giản.

**Để tìm hiểu thêm về CSS3 Transitions:**

Xem các ví dụ trực quan: [đã xoá URL không hợp lệ]

Sử dụng các công cụ trực tuyến để thử nghiệm: <https://codepen.io/>, <https://flukeout.github.io/>

**Lưu ý:**

Hỗ trợ trình duyệt cho CSS3 Transitions khá tốt, nhưng vẫn có một số trình duyệt cũ không hỗ trợ đầy đủ.

Việc sử dụng quá nhiều transitions có thể ảnh hưởng đến hiệu suất trang web.

**Kết luận:**

CSS3 Transitions là một công cụ mạnh mẽ và dễ sử dụng cho phép bạn tạo ra các hiệu ứng chuyển tiếp mượt mà và đẹp mắt trên trang web. Hãy sử dụng nó một cách thông minh để nâng cao trải nghiệm người dùng và tạo ra các trang web độc đáo và thu hút.

***- CSS3 Multiple Columns***

**CSS3 Multiple Columns** là một mô-đun cho phép bạn chia nội dung văn bản thành nhiều cột, giống như cách trình bày trong báo chí hoặc tạp chí. Nó cung cấp một cách thức linh hoạt để bố trí nội dung trên trang web, giúp tăng tính dễ đọc và thu hút người dùng.

**Dưới đây là một số tính năng chính của CSS3 Multiple Columns:**

**Thuộc tính:**

column-count: Xác định số lượng cột.

column-width: Xác định độ rộng của mỗi cột.

column-gap: Xác định khoảng cách giữa các cột.

column-rule: Xác định đường viền giữa các cột.

column-span: Xác định số lượng cột mà một phần tử sẽ bao phủ.

**Ví dụ:**

CSS

.content {

column-count: 3;

column-gap: 20px;

}

Trong ví dụ này, nội dung văn bản sẽ được chia thành 3 cột với khoảng cách 20px giữa các cột.

**Lợi ích của CSS3 Multiple Columns:**

**Cải thiện khả năng đọc:** Chia nhỏ nội dung thành nhiều cột giúp người dùng dễ đọc hơn, đặc biệt trên màn hình nhỏ.

**Tăng tính thu hút:** Bố cục nhiều cột giúp trang web trở nên thu hút và bắt mắt hơn.

**Tiết kiệm không gian:** Có thể sử dụng nhiều cột để hiển thị nhiều nội dung hơn trên một trang.

**Trường hợp sử dụng CSS3 Multiple Columns:**

Bố trí bài viết blog và tin tức.

Hiển thị danh sách sản phẩm.

Tạo các bố cục dạng tạp chí.

***-CSS3 Animations***

## CSS3 Animations là gì?

**CSS3 Animations** là một mô-đun cho phép bạn tạo ra các hiệu ứng hoạt hình mượt mà và đẹp mắt cho các phần tử HTML. Nó hoạt động bằng cách thay đổi giá trị của các thuộc tính CSS theo thời gian, tạo ra chuyển động và biến đổi trực quan.

**Dưới đây là một số tính năng chính của CSS3 Animations:**

**Thuộc tính:**

animation-name: Xác định tên của hiệu ứng hoạt hình.

animation-duration: Xác định thời lượng của hiệu ứng hoạt hình.

animation-timing-function: Xác định tốc độ của hiệu ứng hoạt hình.

animation-delay: Xác định thời gian chờ trước khi bắt đầu hiệu ứng hoạt hình.

animation-iteration-count: Xác định số lần lặp lại hiệu ứng hoạt hình.

animation-direction: Xác định chiều hướng của hiệu ứng hoạt hình.

**Ví dụ:**

CSS

@keyframes my-animation {

0% {

background-color: red;

}

50% {

background-color: blue;

}

100% {

background-color: green;

}

}

.button {

animation-name: my-animation;

animation-duration: 2s;

animation-timing-function: ease-in-out;

}

.button:hover {

animation-play-state: paused;

}

Trong ví dụ này, hiệu ứng hoạt hình my-animation sẽ thay đổi màu nền của nút từ đỏ sang xanh lam và sang xanh lá cây trong 2 giây với tốc độ tăng tốc và giảm tốc mượt mà. Khi người dùng di chuột qua nút, hiệu ứng hoạt hình sẽ tạm dừng.

**Lợi ích của CSS3 Animations:**

**Cải thiện trải nghiệm người dùng:** Tạo hiệu ứng hoạt hình mượt mà và trực quan, giúp nâng cao trải nghiệm người dùng.

**Tăng tính tương tác:** Giúp trang web trở nên sống động và tương tác hơn.

**Giảm thiểu sử dụng JavaScript:** Có thể tạo ra các hiệu ứng đơn giản mà không cần sử dụng JavaScript.

**Trường hợp sử dụng CSS3 Animations:**

Tạo hiệu ứng di chuột cho các nút, menu và các phần tử khác.

Thêm hiệu ứng chuyển tiếp cho các hình ảnh, thanh trượt và các nội dung khác.

Tạo các hiệu ứng hoạt hình phức tạp.

**Lưu ý:**

Hỗ trợ trình duyệt cho CSS3 Animations khá tốt, nhưng vẫn có một số trình duyệt cũ không hỗ trợ đầy đủ.

Việc sử dụng quá nhiều animations có thể ảnh hưởng đến hiệu suất trang web.

**Kết luận:**

CSS3 Animations là một công cụ mạnh mẽ và dễ sử dụng cho phép bạn tạo ra các hiệu ứng hoạt hình mượt mà và đẹp mắt trên trang web. Hãy sử dụng nó một cách thông minh để nâng cao trải nghiệm người dùng và tạo ra các trang web độc đáo và thu hút.

**3.12. SCSS**

3.12 không phải là phiên bản chính thức của SCSS. SCSS (Sass Script) là cú pháp tiền xử lý cho CSS, nó liên tục được cập nhật và phát triển theo thời gian, không có hệ thống phiên bản chính thức như phần mềm.

Tuy nhiên, 3.12 có thể đề cập đến:

Phiên bản của một trình biên dịch SCSS cụ thể:

Một số trình biên dịch SCSS phổ biến như Sass, LibSass, Dart Sass có hệ thống phiên bản riêng.

Ví dụ, phiên bản 3.12.0 của node-sass (một công cụ triển khai SASS phổ biến) được phát hành vào năm 2016.

Số tham chiếu nội bộ không liên quan:

Trong một số trường hợp, số 3.12 có thể là mã định danh nội bộ được sử dụng trong các công cụ hoặc tài liệu liên quan đến SCSS.

Để xác định ý nghĩa chính xác của 3.12 trong ngữ cảnh cụ thể:

Cần có thêm thông tin về ngữ cảnh, ví dụ như:

Nơi bạn gặp số 3.12 (tài liệu, công cụ, mã nguồn)?

Có bất kỳ thông tin bổ sung nào đi kèm với số 3.12?

Lưu ý:

Phiên bản SCSS mới nhất (tính đến tháng 11 năm 2023) là SCSS 4.

Nên sử dụng phiên bản mới nhất của SCSS để tận dụng các tính năng mới nhất, sửa lỗi và cải thiện bảo mật.

Tóm lại:

3.12 không phải là phiên bản chính thức của SCSS.

3.12 có thể đề cập đến phiên bản của một trình biên dịch SCSS cụ thể hoặc số tham chiếu nội bộ.

Cần có thêm thông tin để xác định ý nghĩa chính xác của 3.12.

**3.13. SASS**

SASS là viết tắt của **Syntactically Awesome Style Sheets**, là một công cụ phần mềm (cụ thể là một trình biên dịch) giúp bạn viết CSS dễ dàng hơn. SASS sử dụng cú pháp **SCSS** (Sass Script), một ngôn ngữ lập trình được thiết kế riêng cho việc tạo kiểu trang web.

**SASS cung cấp một số tính năng hữu ích:**

**Biến:** Giúp bạn lưu trữ và sử dụng lại các giá trị CSS.

**Nesting:** Giúp bạn tổ chức mã CSS theo cấu trúc logic.

**Mixin:** Giúp bạn tạo các đoạn mã CSS có thể tái sử dụng.

**Functions:** Giúp bạn thực hiện các thao tác logic và toán học trong CSS.

**Lợi ích của việc sử dụng SASS:**

**Giúp viết mã CSS nhanh hơn và dễ dàng hơn.**

**Giúp mã CSS dễ đọc và bảo trì hơn.**

**Giúp tạo ra các trang web có thiết kế đẹp mắt và hiệu ứng ấn tượng.**

**Cách sử dụng SASS:**

**Cài đặt trình biên dịch SASS:** Có nhiều trình biên dịch SASS khác nhau, bạn có thể lựa chọn trình biên dịch phù hợp với nhu cầu của mình.

**Viết mã SCSS:** Sử dụng cú pháp SCSS để viết mã CSS của bạn.

**Biên dịch mã SCSS sang CSS:** Sử dụng trình biên dịch SASS để chuyển đổi mã SCSS sang CSS.

**Để tìm hiểu thêm về SASS:**

**Lưu ý:**

Hỗ trợ trình duyệt cho CSS được tạo từ SASS khá tốt, nhưng vẫn có một số trình duyệt cũ không hỗ trợ đầy đủ.

Việc sử dụng quá nhiều tính năng SASS có thể ảnh hưởng đến hiệu suất trang web.

**Kết luận:**

SASS là một công cụ mạnh mẽ và dễ sử dụng giúp bạn viết CSS nhanh hơn, dễ dàng hơn và đẹp mắt hơn. Hãy sử dụng SASS để nâng cao kỹ năng CSS của bạn và tạo ra các trang web ấn tượng.

**Kết luận chương 3**

CSS là ngôn ngữ lập trình quan trọng để tạo trang web đẹp mắt và thu hút. CSS giúp kiểm soát cách hiển thị của các phần tử HTML như màu sắc, phông chữ, bố cục, v.v.

CSS3 là phiên bản mới nhất của CSS với nhiều tính năng mới giúp tạo trang web hiện đại và phức tạp hơn.

**CHƯƠNG 4: THIẾT KẾ WEBISTE THEO ĐỀ TÀI BẠN CHỌN**

**`**

**4.1. Ý tưởng của Website**

Em được sinh ra và lớn lên tại vùng đất Hà Giang em có một ước mơ là cho vùng đất Hà Giang là thụy sĩ thứ 2 của thế giới ,em làm wed này em muốn quảng bá du lịch ,những danh lam tháng cảnh ở quê hương em.

**4.2. Xây dựng bố cục của trang Web**

1 : Đầu tiên em xác định đối tượng mà em sẽ nhẫm đến (địa điểm).

2 : Trong địa điểm đó sẽ có những nội dung ,ảnh gì đẹp (logo,nội dung).

3 : Tiến hành code với 1 trang đầu để cho người dung lựa chọn những điểm đến

4 : Tiến hành code với html con khi người dùng ấn vào.

5 : Các trang sẽ có ảnh ở bên trên cho có sự quấn hút và nội dung bên dưới.

**4.3. Thiết kế trang Web bằng HTML và CSS**

***4.3.1.Wed trang chủ bằng html và css***

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Hà Giang</title>

    <link rel="stylesheet" href="btl.css">

    <link rel="stylesheet" href="quanbacon.html">

    <link rel="stylesheet" href="dongvan.html">

    <link rel="stylesheet" href="wedcon1.html">

    <link rel="stylesheet" href="wed.html">

    <link rel="stylesheet" href="yenmiinh.html">

    <link rel="stylesheet" href="bacme.html">

    <link rel="stylesheet" href="meovac.html">

    <link rel="stylesheet" href="">

    <link rel="stylesheet" href="">

</head>

<body>

    <header>

        <div class="container">

            <img class="logo" src="12.webp" alt="">

            <input type="text" placeholder="nhập địa điểm"><button><a href="">TÌM KIẾM</a></button>

        </div>

        <ul class="menu">

            <li><a href="">TRANG CHỦ </a> </li>

            <li><a href="">GIỚI THIỆU</a></li>

            <li><a href="">LỊCH TRÌNH ĐI KHÁM PHÁ 1 VÒNG HÀ GIANG</a></li>

            <li><a href="">NHỮNG ĐỊA ĐIỂM NỔI TIẾNG</a></li>

            <li><a href="">PHONG CẢNH ĐẸP NHẤT HÀ GIANG</a></li>

        </ul>

        <h1 class="\*">KHÁM PHÁ HÀ GIANG</h1>

    </header>

    <section class="section">

        <div class="products">

            <div class="product">

                <img src="km-so-0-ha-giang-3.webp" alt="">

                <h3>THÀNH PHỐ HÀ GIANG</h3>

                <button><a href="wedcon1.html">điểm đến</a></button>

            </div>

            <div class="product">

                <img src="nui-doi-quan-ba-chiem-nguong-toa-thien-nhien-quyen-ru-noi-dia-dau-to-quoc-1642419636.webp" alt="">

                <h3>QUẢN BẠ</h3>

                <button><a href="quanbacon.html">điếm đển</a></button>

            </div><div class="product">

                <img src="ha-giang-\_trai-nghiem-cua-bestprice-571ee9f7aa29a-266x150.webp" alt="">

                <h3>YÊN MINH</h3>

                <button><a href="yenmiinh.html">điểm đến</a></button>

            </div>

            </div>

            <div class="products">

            <div class="product">

                <img src="cot-co-lung-cu-1.webp" alt="">

                <h3>ĐỒNG VĂN</h3>

                <button><a href="dongvan.html">điểm đến</a></button>

            </div>

            <div class="product">

                <img src="meo-vac-01\_1687674823.webp" alt="">

                <h3>MÈO VẠC</h3>

                <button><a href="meovac.html">điếm đển</a></button>

            </div>

            <div class="product">

                <img src="512px-MinhNgoc-NaSai'2007.webp" alt="">

                <h3>BẮC MÊ</h3>

                <button><a href="bacme.html">điểm đến</a></button>

            </div>

            </div>

            <a href="wed.html">Xem thêm</a>

            </div>

    </section>

</body>

</html>

Css

\*{

    margin: 0;

    padding: 0;

}

header img{

        width: 100%; /\* Bề ngang là 300 pixel \*/

        height: 200px; /\* Chiều cao tự động điều chỉnh để giữ tỷ lệ khung hình ban đầu \*/

}

input[type="text"] {

    position: absolute; /\* Đặt vị trí của input là absolute \*/

    top:150px; /\* Đặt top ở giữa container \*/

    left: 950px; /\* Đặt left ở giữa container \*/

    transform: translate(-50%, -50%); /\* Dịch chuyển input điều chỉnh để căn giữa \*/

    width: 500px;

    height: 30px;

}

header button{

    position: absolute; /\* Đặt vị trí của input là absolute \*/

    top:150px; /\* Đặt top ở giữa container \*/

    left: 1150px; /\* Đặt left ở giữa container \*/

    transform: translate(-50%, -50%); /\* Dịch chuyển input điều chỉnh để căn giữa \*/

    width: 100px;

    height: 30px;

}

header h1{

    text-align: center;

    color: rgb(184, 10, 10);

    font-size: 30px;

    margin-top: 10px;

    margin-bottom: 10px;

    font-family: Verdana, Geneva, Tahoma, sans-serif;

    padding: 5px;

}

.menu li{

    margin: auto;

    color: white;

    list-style-type: none;

    font-size: 15px;

    padding: 5px;

}

.menu{

    background-color: rgb(186, 197, 25);

    display: flex;

    margin-left: 0px;

    margin-right: 0px;

    margin-top: -4px;

}

main{

    padding: 5px 30px 0 30px;

}

.section{

    margin-left: 100px;

    margin-right: 100px;

    border-color: navajowhite;

    padding: 10px;

    border-radius: 10px;

}

.section p{

    text-align: center;

}

.section h1{

    text-align: center;

    font-size: 30px;

    color: brown;

    text-transform: uppercase;

    font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans Unicode', Geneva, Verdana, sans-serif;

    margin-bottom: auto;

}

.products{

    display: flex;

    flex-wrap: wrap;

    justify-content: space-between;

}

.product{

    width: calc(33.33% - 10px);

    text-align: center;

}

.product img{

    width: 100%;

    border: 1px solid black;

    border-radius: 10px;

}

.product h3{

    font-size: 20px;

}

.product p{

    margin-top: 10px;

    margin-bottom: 10px;

}

.product button{

    background-color: black;

    color: white;

    border: 5px;

    border-radius: 10px;

    padding: 5px 10px 5px 10px;

}

.section a{

    text-decoration: none;

    color: white;

    border: 1px solid black;

    padding: 5px 10px 5px 10px;

    margin: 10px auto;

    background-color: red;

    border-radius: 10px;

}

footer{

    background-color: black;

    color: white;

    padding: 5px;

    text-align: center;

}

.section img {

    width: 380px; /\* Bề ngang là 300px \*/

    height: 390px; /\* Chiều cao tự động để giữ tỷ lệ khung hình ban đầu \*/

    border: none;/\*bo viền\*/

}

body {

    background:rgb(77, 190, 185);

}

***4.3.2. HTML CSS TP Hà Giang***

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>TP HÀ GIANG</title>

    <link rel="stylesheet" href="tp2.css">

    <link rel="stylesheet" href="btl1.html">

</head>

<body>

    <img src="anhphu 111111111.jpg" alt="">

    <div class="sayqua">

        <button><a href="btl1.html">TRANG CHỦ</a></button>

    </div>

    <div class="caoss">

        <h1>THÀNH PHỐ HÀ GIANG</h1>

        <img src="bando.jpg" alt="">

        <p>Hà Giang là tỉnh địa đầu của Tổ quốc, phía Đông giáp tỉnh Cao Bằng, phía Tây giáp tỉnh Yên Bái và Lào Cai, phía Nam giáp tỉnh Tuyên Quang, phía Bắc giáp Trung Quốc. <br> Các cung đường Hà Giang quanh co, cheo leo vốn đầy thách thức cho người lái xe. Thế nhưng khi bạn phóng tầm mắt xung quanh, căng thẳng nhường lại chỗ cho cảm giác thư thái khi chiêm ngưỡng vẻ đẹp cao nguyên.

        </p>

    </div>

    <div class="chode">

        <img src="toancanhhag.webp" alt="">

    </div>

    <div class="loz">

            <h1>HÀ GIANG MÙA NÀO ĐẸP NHẤT</h1><br>

            <img src="tam giac mach.jpg" alt="">

            <p>Là tỉnh miền núi cao nên khí hậu ở Hà Giang lạnh rõ rệt so với vùng thấp và trung du kế cận. Nhiệt độ trung bình năm khoảng 21-23 độ C. Nét nổi bật của khí hậu là độ ẩm cao trong năm, mưa nhiều và kéo dài.

                Bạn có thể du lịch Hà Giang vào bất kỳ mùa nào trong năm. <br>Người ta thường đến Hà Giang vào mùa thu lúa chín vàng. Thời gian đẹp nhất là tháng 10, 11 và 12, khi hoa tam giác mạch hay những cánh đồng cải khoe sắc. Còn mùa xuân, hoa mơ, hoa mận nở trắng rừng khiến bạn như đang bồng bềnh trên mây.</p>

                <p>Tháng 5, những thửa ruộng lấp loáng mùa nước đổ. Tháng 6 và tháng 7, nhiều người bỏ lỡ Hà Giang vì những cơn mưa hè réo rắt bất chợt. Nhưng cũng nhờ sự ẩm ướt này, núi rừng nơi đây lại khoác lên mình một màu xanh mướt say đắm lòng .</p>

    </div>

    <div class="quan">

        <h1>CỘT MỐC KM SỐ 0</h1>

        <img src="cot moc0.jpg" alt="">

        <p>Có thể nói, Hà Giang là một điểm đến khiến bao du khách phải choáng ngợp với cảnh sắc cao nguyên đá hùng vĩ. Khắp các trang mạng xã hội, không khó để bắt gặp hình ảnh các bạn trẻ hào hứng check-in tại các điểm đến ấn tượng ở Hà Giang. Từ những cung đường đèo uốn lượn đến những cánh đồng hoa ngát xanh. Tuy nhiên, chưa cần đến đèo Mã Pì Lèng, chưa cần đi sông Nho Quế hay các bản làng người Mông, một tọa độ khiến hội mê xê dịch phải phát cuồng đó chính là cái tên "Cột mốc số 0" ở thành phố Hà Giang.Cột mốc số 0 hay Km số 0 Hà Giang là một điểm đến "được lòng" rất nhiều dân phượt. Chẳng phải vì cột mốc ấy có hình dáng gì đặc biệt hay nó được đặt tại một địa điểm du lịch nổi tiếng. Đây chỉ là một trụ nhỏ bằng bê tông tương tự như các cột mốc chỉ km tại mọi tuyến quốc lộ ở nước ta.

            Một trụ đá nhỏ nhưng lại chính là điểm bắt đầu của quốc lộ 2, nối giữa 2 tỉnh Hà Giang và Thủ đô Hà Nội. Nghe thì chả có gì đặc biệt nhưng thực chất nếu bạn muốn bắt đầu 1 chuyến du lịch Hà Giang thì bắt buộc phải đi qua cột mốc ấy.

            Dù đã tồn tại qua bao năm tháng, thế nhưng trong vài năm trở lại đây cột mốc Km số 0 Hà Giang bỗng dưng nổi tiếng trên các trang mạng xã hội và trở thành tọa độ check-in "hót hòn họt", không thể bỏ qua của cộng đồng mê xê dịch. Trên thực tế, có đến 2 cột mốc Km số 0 nằm tại trung tâm thành phố Hà Giang. Một cột mốc nằm trong công viên đối diện quảng trường 26 – 3 với kích thước lớn (cột mốc đánh dấu lãnh thổ) và cột mốc nhỏ còn lại nằm trên đường Nguyễn Trãi, quốc lộ 2 (cột mốc chỉ dẫn giao thông).So với nhiều điểm đến ở Hà Giang khác, Km số 0 gần như không có trải nghiệm gì cả. Tuy nhiên đây lại là nơi mang giá trị về mặt tinh thần. Một bức ảnh check in ở đây khẳng định rằng thanh xuân này bạn đã kịp đến Hà Giang, khám phá một vùng đất xinh đẹp, kỳ vĩ và lãng mạn bậc nhất miền núi phía Bắc nước ta.Có thể nói, cột trụ bằng bê tông bé nhỏ để chỉ dẫn giao thông nhưng lại trở thành biểu tượng quen thuộc xuất hiện trong hầu hết các bức hình của du khách mỗi khi đặt chân đến Hà Giang. Nó trở thành điểm khởi đầu của một hành trình đầy ý nghĩa gắn với niềm tự hào về màu cờ Tổ quốc trên cao nguyên đá kỳ vỹ này.

        </p>

    </div>

    <div class="yeu">

        <h1>NÚI CẨM SƠN </h1>

        <img src="nui-cam-ha-giang-4.jpg" alt="">

        <p>Vị trí:<br>

            Núi Cẩm Sơn nằm tại huyện Quản Bạ, tỉnh Hà Giang, Việt Nam.

            Núi Cẩm Sơn có vị trí gần với thị trấn Tam Sơn, nổi tiếng với cảnh quan thiên nhiên đẹp và vùng đất của người H'Mông.

            Đặc điểm:<br>

            Núi Cẩm Sơn là một trong những điểm đến du lịch phổ biến ở Hà Giang, thu hút khách du lịch bởi cảnh quan thiên nhiên hùng vĩ và đẹp mắt.

            Núi Cẩm Sơn thường được du khách ghé thăm để chiêm ngưỡng cảnh sắc hữu tình của núi rừng, thung lũng và ruộng bậc thang.

            Hoạt động du lịch:<br>

            Du khách thường thích tham gia các hoạt động như trekking, leo núi, ngắm cảnh, và khám phá văn hóa của các dân tộc bản địa.

            Có nhiều con đường mòn dẫn đến đỉnh núi và các điểm ngắm cảnh đẹp.

            Lưu ý:<br>

            Trước khi tham gia bất kỳ hoạt động nào tại núi Cẩm Sơn, du khách nên chuẩn bị trang thiết bị cá nhân, đặc biệt là đồ dã ngoại, nước uống và thực phẩm.

            Núi Cẩm Sơn là một trong những điểm đến tuyệt vời để khám phá cảnh quan thiên nhiên và văn hóa ở Hà Giang, mang lại cho du khách trải nghiệm đầy thú vị và ấn tượng về vùng đất núi phía Bắc của Việt Nam.</p>

    </div>

</body>

</html>

***-CSS***

\*{

    padding: 0;

    margin: 0%;

}

img {

    width: 100%;

    height: 250px;

}

p{

    font-size: 25px;

    text-align: left;

}

.sayqua button{

    float: right;

    margin-top: -2px;

    background-color: rgb(34, 14, 211);

    font-size: larger;

}

div h1 {

    text-align:left ;

    color: brown;

}

.caoss h1{

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    text-align: center;

}

.caoss img {

    float: left; /\* Căn ảnh bên trái \*/

    margin-top: -40px;

    width: 100px;

    height: 160px;

}

.chode img{

    height: 230px;

    width: 240px;

    float: right;

    margin-top: 60px;

}

.loz img{

    margin-right: 5px;

    float: left; /\* Căn ảnh bên trái \*/

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    width: 280px;

    height: 230px;

}

.loz h1 {

    margin-left: 100px;

    text-align: center;

    transform: translate(5%, 9%);

}

.loz p{

    font-size: 25px;

    text-align: left;

}

.quan h1{

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    text-align: center;

}

.quan img {

    margin-right: 5px;

    float: left; /\* Căn ảnh bên trái \*/

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    width: 300px;

    height: 460px;

}

.quan p {

    display: flex;

    align-items: center; /\* Căn giữa theo chiều dọc \*/

}

.yeu h1{

    text-align: center;

}

.yeu img{

    float: left;

    width: 500px;

    height: 320px;

}

.yeu p {

    font-size: 25px;

    text-align: left;

}

body p{

    color: aqua;

}

body{

    background: rgb(102, 122, 53);

}

***4.3.3. HTML CSS Huyện Quản Bạ***

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>QUẢN BẠ</title>

    <link rel="stylesheet" href="quanba.css">

</head>

<body>

    <img src="anhquanbaphong.jpg" alt="">

    <div class="sayqua">

        <button><a href="btl1.html">TRANG CHỦ</a></button>

    </div>

    <div class="caoss">

        <h1>    HUYỆN QUẢN BẠ</h1>

        <img src="huyn.jpg" alt="">

        <p>Huyện Quản Bạ nằm ở phía bắc tỉnh Hà Giang, có vị trí địa lý:

            Phía đông giáp huyện Yên Minh

            Phía tây và phía bắc giáp châu Văn Sơn, tỉnh Vân Nam, Trung Quốc

            Phía nam giáp huyện Vị Xuyên.

            Huyện Quản Bạ có diện tích 553,7 km², dân số năm 2018 là 56.840 người, có 22 dân tộc cùng sinh sống, trong đó dân tộc H,Mông chiếm đa số. Huyện lỵ là thị trấn Tam Sơn nằm trên quốc lộ 4C, cách thành phố Hà Giang 46 km về hướng bắc, cách huyện Yên Minh 52 km về phía đông.

        </p>

    </div>

    <div class="chode">

        <h1>CÂY CÔ ĐƠN</h1>

        <img src="cay.webp" alt="">

        <p>Đây là một điểm check-in được nhiều bạn trẻ yêu thích, vị trí cây cô đơn này nằm ở xã Cán Tỷ, trên đường từ Quản Bạ đi Yên Minh. Các bạn chú ý là ngay chỗ cầu Cán Tỷ sẽ có 2 đường đi Yên Minh, đường phía dưới xa hơn nhưng là đường to đẹp hơn, đi xuyên qua rừng thông Yên Minh. Đường phía trên gần hơn sẽ đi qua điểm check-in này.

        </p>

    </div>

    <div class="loz">

            <h1>NÚI ĐÔI</h1>

            <img src="dime.webp" alt="">

            <p>Núi Đôi Quản Bạ, nằm cạnh quốc lộ 4C, cách thành phố Hà Giang 40 km. Giữa núi đá và ruộng bậc thang trùng điệp, nổi lên hai trái núi có hình dáng kỳ lạ, khiến du khách không khỏi ngỡ ngàng trước vẻ đẹp kỳ vĩ của tạo hóa. Hai trái núi này gắn với truyền thuyết Núi Cô Tiên thi vị.

                Truyền thuyết kể rằng ngày xửa ngày xưa, ở vùng đất này có một chàng trai người Mông đẹp trai có tài thổi đàn môi. Tiếng đàn môi của chàng réo rắt như tiếng suối, ríu rít lảnh lót như tiếng chim rừng, lúc lại sâu lắng, trầm bổng, da diết như tiếng gió giữa đêm khuya cứ bay xa, bay xa mãi. Có một nàng tiên trên thượng giới tên là Hoa Đào xinh đẹp tuyệt trần, da trắng như tuyết, đôi môi như nụ đào xuân chúm chím, hai má ửng hồng như trái đào chín. Một hôm nọ nghe được tiếng đàn môi của chàng đã theo gió trốn xuống trần gian tìm người thổi đàn môi mà phải lòng chàng và đã trốn ở lại vùng đất này. Họ nên vợ nên chồng và sinh được một bé trai vô cùng xinh xắn.Lúc này Ngọc Hoàng đã phát hiện ra chuyện nàng bỏ trốn xuống trần gian lấy người phàm trần đã vô cùng giận dữ sai người đi bắt nàng về. Nàng khóc lóc van xin cho nàng được ở lại nuôi con nhưng không ai mủi lòng. Thương chồng một mình nuôi con vất vả, thương con thơ thiếu bầu sữa mẹ, nàng đã bỏ lại đôi nhũ của mình dưới hạ giới cho con bú. Đôi nhũ cứ căng tròn nuôi con nàng ngày càng khôn lớn, sau này đã biến thành hai quả núi dáng hình như bầu vú mẹ, tròn trịa, đều đặn đến lạ thường mà ngày nay vẫn gọi là Núi Đôi hay núi Cô Tiên ở Tam Sơn, Quản Bạ.Tương truyền nhờ dòng sữa của nàng mà vùng đất này có khí hậu vô cùng mát mẻ, các loại hoa trái như Đào, Mận, Lê, Hồng thơm ngon kỳ lạ, rau trái thì luôn xanh tươi, lúa ngô luôn tươi tốt trở thành vùng đất trù phú. Con gái ở đây xinh đẹp có tiếng, hai má lúc nào cùng ửng hồng, mịn màng như trái đào tiên. Và nước mắt của nàng đã biến thành dòng sông Miện xanh ngắt, mơ màng, khảm trôi trên biển đá tai mèo phún sắc, ôm lấy cả rẻo đất phía sau cổng trời huyền thoại.

            </p>

    </div>

    <div class="quan">

        <h1>CHỢ QUẢN BẠ</h1>

        <img src="choqua.webp" alt="">

        <p>Trên chặng đầu tiên của con đường huyền thoại mang tên Con đường hạnh phúc, dốc Bắc Sum chính là con dốc đánh dấu điểm bắt đầu của hành trình xuyên qua công viên Địa chất toàn cầu Cao nguyên đá Đồng Văn – Hà Giang. Ở trên đỉnh con dốc dài uốn lượn ấy, cổng trời Quản Bạ là một địa danh nổi tiếng từ lâu bởi cảnh quan kỳ vĩ và câu chuyện huyền thoại núi Đôi.

            Thế nhưng, có một địa danh nằm ngay dưới chân mỏm núi cổng trời, đó chính là xã Quyết Tiến mộc mạc, đơn sơ bên cổng trời sừng sững quanh năm mây phủ. Ở nơi yên bình ấy, mỗi tuần một lần lại có buổi chợ phiên, khoảng thời gian diễn ra chỉ trong một buổi sáng ngắn ngủi. Chợ Quyết Tiến họp vào mỗi sáng thứ 7 ở ngay sát Quốc lộ 4C, cách cổng trời Quản Bạ khoảng gần chục km.

            Có lẽ nhiều bạn sẽ nhận xét rằng chợ phiên Hà Giang thì chỗ nào cũng vậy, trông cứ na ná nhau, vẫn những bộ quần áo sặc sỡ, bán những mặt hàng thường ngày, thường chỉ họp vào buổi sáng…Nếu là vậy thì Quyết Tiến có thể coi là phiên chợ vùng cao đầu tiên mà bạn găp trên chặng đường chinh phục Hà Giang. Đừng bỏ lỡ, sau khi xuất phát từ Tp Hà Giang chỉ khoảng hơn 1h các bạn sẽ tới Quyết Tiến, gửi xe và làm một vòng chợ, ăn một gói xôi ngũ sắc hay một món bất kỳ rồi vác máy ảnh chạy quanh chợ, bạn sẽ tự tìm ra những điểm riêng của phiên chợ này.

        </p>

    </div>

    <div class="yeu">

        <h1>CỔNG TRỜI QUẢN BẠ </h1>

        <img src="congtroi.webp" alt="">

        <p>Cổng trời Quản Bạ – cao 1500m so với mặt biển, đây là cửa ngõ đầu tiên lên cao nguyên Đồng Văn. Năm 1939, người ta dựng một cánh cửa khổng lỗ bằng gỗ nghiến dày 150cm ở ngay Cổng trời. Một thời, sau cánh cửa gỗ này là một “thế giới” khác – còn gọi là “Vùng tự trị của người Mèo”, gồm các huyện Quản Bạ, Mèo Vạc,Yên Minh và Đồng Văn</p>

    </div>

</body>

</html>

***-CSS***

\*{

    padding: 0;

    margin: 0%;

}

img {

    width: auto;

    height: auto;

}

p{

    font-size: 25px;

    text-align: left;

}

.sayqua button{

    float: right;

    margin-top: -2px;

    background-color: rgb(34, 14, 211);

    font-size: larger;

}

div h1 {

    text-align:left ;

    color: brown;

}

.caoss h1{

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    text-align: center;

}

.caoss img {

    float: left; /\* Căn ảnh bên trái \*/

    margin-top: -0px;

    width: 100px;

    height: 110px;

}

.chode h1 {

    text-align: center;

}

.chode img{

    height: 150px;

    width: 240px;

    float: right;

    margin-top: -20px;

}

.loz img{

    margin-right: 5px;

    float: left; /\* Căn ảnh bên trái \*/

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    width: 280px;

    height: 230px;

}

.loz h1 {

    margin-left: 100px;

    text-align: center;

    transform: translate(5%, 9%);

}

.loz p{

    font-size: 25px;

    text-align: left;

}

.quan h1{

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    text-align: center;

}

.quan img {

    margin-right: 5px;

    float: left; /\* Căn ảnh bên trái \*/

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    width: 300px;

    height: 390px;

}

.quan p {

    display: flex;

    align-items: center; /\* Căn giữa theo chiều dọc \*/

}

.yeu h1{

    text-align: center;

}

.yeu h1{

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    text-align: center;

}

.yeu img {

    float: left; /\* Căn ảnh bên trái \*/

    margin-top: -0px;

    width: 200px;

    height: 110px;

}

body p{

    color: aqua;

}

body{

    background: rgb(102, 122, 53);

}

***4.3.4. HTML Đồng Văn***

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>ĐỒNG VĂN</title>

    <link rel="stylesheet" href="dongvan.css">

</head>

<body>

    <img src="anhphu 111111111.jpg" alt="">

    <div class="sayqua">

        <button><a href="btl1.html">TRANG CHỦ</a></button>

    </div>

    <div class="sao">

        <img src="anh10.jpg" alt="">

        <img src="ảnh2.jpg" alt="">

    </div>

    <div class="caoss">

        <h1>    HUYỆN ĐỒNG VĂN</h1>

        <P>Đồng Văn là huyện vùng cao biên giới của tỉnh Hà Giang, cách thành phố Hà Giang 150 km về phía bắc, là huyện cực bắc của Việt Nam (điểm cực bắc tại núi Rồng, xã Lũng Cú), có tọa độ từ 22°55'B đến 23°23'B; 105°42'Đ, có vị trí địa lý:

            Phía Bắc và Đông bắc giáp Trung Quốc với biên giới dài 52,5 km

            Phía Đông và Đông nam giáp huyện Mèo Vạc

            Phía Tây và Tây Nam giáp huyện Yên Minh.

            Huyện Đồng Văn có diện tích 453,08 km², dân số năm 2021 là 85.662 người[2], mật độ dân số 189 người/km².

            Huyện lỵ trước đây của huyện Đồng Văn đặt tại thị trấn Phố Bảng, sau chuyển về khu phố cổ thuộc xã Đồng Văn, nay là thị trấn Đồng Văn. Đây cũng là một trong những trọng điểm của các cuộc tấn công từ phía Trung Quốc sang Việt Nam năm 1979 và cả trong những năm 1984-1986.

            Huyện Đồng Văn hiện nay có 17 dân tộc sinh sống, gồm: Mông, Tày, Dao, Kinh, Nùng, Giáy, La Chí, Hoa, Pà Thẻn, Cơ Lao, Lô Lô, Bố Y, Phù Lá, Pu Péo, Mường, Sán Chay, Thái. Trong đó dân tộc Mông chiếm phần lớn.

            Nguồn nước mặt trên địa bàn huyện rất khan hiếm, chỉ có các suối nhỏ chảy vào mùa mưa và một số hồ ao khác.

            Huyện Đồng Văn có diện tích đất rừng 23.242,6 ha, chiếm 52,0% diện tích tự nhiên (tính cả diện tích đất đồi núi đá không có rừng cây có khả năng sử dụng vào mục đích lâm nghiệp). Mặc dù đất lâm nghiệp có tỷ lệ khá trong cơ cấu sử dụng đất (23.242,6 ha, chiếm 52,0%) song phần lớn các loại rừng của huyện đều là rừng trồng và rừng tái sinh nên chất lượng rừng và trữ lượng không cao. Việc khai thác và kinh doanh các sản phẩm từ rừng hạn chế, chức năng chính của rừng của huyện là giữ đất và bảo vệ môi trường sinh thái.

            Huyện Đồng Văn cũng có nhiều tài nguyên khoáng sản với trữ lượng lớn; gồm các loại khoáng sản như: mangan, antimon, đá vôi,... Tuy nhiên việc khai thác gặp rất nhiều khó khăn do hạ tầng giao thông còn chưa phát triển, xa trung tâm tỉnh, khoáng sản nằm xen kẽ với nhiều tầng đá.

        </P>

    </div>

    <div class="chode">

        <h1>PHỐ CỔ ĐỒNG VĂN</h1>

        <img src="phoco.jpg" alt="">

        <p>Phố cổ Đồng Văn nằm ở thị trấn Đồng Văn, Đồng Văn, Hà Giang. Khu vực trung tâm thị trấn Đồng Văn xưa thuộc tổng Đông Quan, châu Nguyên Bình, phủ Tường Vân, tỉnh Tuyên Quang và có lịch sử phát triển về kiến trúc, văn hóa hàng trăm năm. Những năm 1880, khi chiếm đóng khu vực này, người Pháp đã có những quy hoạch và để lại những điểm nhấn quan trọng về quy hoạch và kiến trúc, đặc biệt là chợ Đồng Văn (cũ), xây bằng đá trong những năm 1920 gần như còn nguyên vẹn đến ngày nay. <br>

        </p>

    </div>

    <div class="loz">

            <h1>DỐC THẨM MÃ</h1>

            <img src="anh15.jpg" alt="">

            <p>Tương truyền rằng ngày xưa người dân cho ngựa thồ hàng đi từ dưới chân dốc lên, con ngựa nào mà lên đến đỉnh còn khỏe thì người dân sẽ giữ lại nuôi nên đoạn dốc có tên là Thẩm Mã. Đây là đoạn đèo đầu tiên mà các bạn cần chinh phục để bước chân tới mảnh đất Đồng Văn. Sau khi vượt qua con dốc này, các bạn sẽ tới mảnh đất Phố Cáo

            </p>

    </div>

    <div class="quan">

        <h1>DINH VƯƠNG </h1>

        <img src="dinhvuong.jpg" alt="">

        <p>Trước khi khởi công xây dựng, vua Mèo Vương Chính Đức đã sang Trung Quốc tìm thầy phong thủy sang Việt Nam đi qua khu vực 4 huyện nằm trong quyền cai quản của mình để chọn địa thế đất. Cuối cùng, Vương Chính Đức và thầy quyết định dừng chân tại thôn Xà Phìn, địa thế đất nằm giữa thung lũng Sà Phìn. Đặc biệt, ở nơi đây có một khối đất nổi lên cao như hình mui con rùa, tượng trưng cho thần kim quy. Nếu xây dựng dinh thự tại đây thì sự nghiệp của Vương Chính Đức sẽ thành về sau.Sau con đường nằm bên hàng cây sa mộc cao vút, chiếc cổng đá bề thế của dinh thự vua Mèo Vương Chính Đức ở Sà Phìn (Đồng Văn – Hà Giang) hiện ra trên đỉnh đồi.Vương Chính Đức là người đứng đầu dòng họ Vương của người Mông ở Hà Giang 1 thế kỷ trước. Giàu có nhờ hoạt động trồng, chế biến và buôn bán thuốc phiện xuyên biên giới với Trung Quốc, Miến Điện, ông đã thống lĩnh vùng cao nguyên này và xưng vươngPhía sau cổng đá là tòa nhà tiền dinh hoành tráng của tòa dinh thự. Dinh thự họ Vương được xây trong 8 năm, tiêu tốn khoảng 150.000 đồng bạc trắng. Hầu hết thợ xây dựng là người Hồi vùng Vân Nam và những tốp thợ giỏi nhất người Mông.

        </p>

    </div>

    <div class="yeu">

        <h1>CỘT CỜ LŨNG CÚ </h1>

        <img src="11.jpg" alt="">

        <p>Cột cờ Lũng Cú nằm trên đỉnh Lũng Cú có độ cao khoảng 1.700m so với mực nước biển, thuộc xã Lũng Cú, huyện Đồng Văn, tỉnh Hà Giang. Đây là một điểm nhỏ trên đoạn đường biên giới Việt Nam-Trung Quốc. Nếu mô phỏng một cách tương đối hình dạng đường biên giới Việt Nam-Trung Quốc thành một chóp nón thì hai điểm thấp nhất theo vĩ độ là A Pa Chải, Điện Biên và Sa Vĩ, Móng Cái, còn Lũng Cú là đỉnh của chóp nón này cũng là điểm cao nhất của cực Bắc Việt Nam.

            Cột cờ Lũng Cú thực chất chưa phải là Cực Bắc của Việt Nam, điểm cực thực sự này nằm ở dưới dòng sông  Nho Quế, nơi mà phải mất cả ngày đường cùng với sự dẫn dắt của những người am hiểu bạn mới có thể đến. Hiện nay, cách cột cờ Lũng Cú khoảng 3km đã có một mốc cực Bắc mang tính biểu tượng khác được xây dựng nằm tại bản Xéo Lủng.

        </p>

    </div>

</body>

</html>

***-CSS***

\*{

    padding: 0;

    margin: 0%;

}

img {

    width: 100%;

    height: 250px;

}

p{

    font-size: 25px;

    text-align: left;

}

.sayqua button{

    float: right;

    margin-top: -2px;

    background-color: rgb(34, 14, 211);

    font-size: larger;

}

div h1 {

    text-align:left ;

    color: brown;

}

.sao img{

    margin-right: 5px;

    float: left; /\* Căn ảnh bên trái \*/

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    width: 270px;

    height: 300px;

}

.caoss h1{

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    text-align: center;

}

.caoss img {

    float: left; /\* Căn ảnh bên trái \*/

    margin-top: 20px;

    width: 100px;

    height: 110px;

}

.chode h1 {

    text-align: center;

}

.chode img{

    height: 140px;

    width: 300px;

    float: right;

    margin-top: -15px;

}

.loz img{

    margin-top: -25px;

    float: right; /\* Căn ảnh bên trái \*/

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    width: 190px;

    height: 180px;

}

.loz h1 {

    margin-left: 100px;

    text-align: center;

    transform: translate(5%, 9%);

}

.loz p{

    font-size: 25px;

    text-align: left;

}

.quan h1{

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    text-align: center;

}

.quan img {

    float: inline-start; /\* Căn ảnh bên trái \*/

    margin-top: 40px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    width: 300px;

    height: 200px;

}

.quan p {

    display: flex;

    align-items: center; /\* Căn giữa theo chiều dọc \*/

}

.yeu h1{

    text-align: center;

}

.yeu h1{

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    text-align: center;

}

.yeu img {

    float: left; /\* Căn ảnh bên trái \*/

    margin-top: -0px;

    width: 200px;

    height: 210px;

}

body p{

    color: aqua;

}

body{

    background: rgb(102, 122, 53);

}

4.3.5. HTML CSS Yên Minh

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>YÊN MINH</title>

    <link rel="stylesheet" href="YENMING.css">

</head>

<body>

    <img src="rung thong.jpg" alt="">

    <div class="sayqua">

        <button><a href="btl1.html">TRANG CHỦ</a></button>

    </div>

    <div class="sao">

        <img src="anh4.jpg" alt="">

        <img src="yenminhdepvl.jpg" alt="">

    </div>

    <div class="caoss">

        <h1>    HUYỆN YÊN MINH </h1>

        <p>Yên Minh là một trong những huyện nhỏ của Hà Giang nằm ở khu vực Đông Bắc của tỉnh Hà Giang, phía bắc Yên Minh giáp với tỉnh Vân Nam của Trung Quốc phía Đông Giáp với huyện Mèo Vạc, phía Tây giáp với huyện Quản Bạ phía Nam giáp với huyện Bắc Mê. Huyện Yên Minh nằm trên khu vực đường quốc lộ 4C cách với thành phố Hà Giang khoảng 98 km.

            Là một huyện của tỉnh Hà Giang nhưng đây cũng là một trong những huyện có nhiều địa điểm du lịch hấp dẫn đối với rất nhiều khách du lịch đến tham quan nơi đây từ khu vực thành phố Hà Giang bạn phải đi qua huyện Quản Bạ thì mới đến được với vùng đất Yên Minh để khám phá được hết những phong cảnh đẹp của nơi đây.

        </p>

    </div>

    <div class="chode">

        <h1>ĐỘNG ÉN</h1>

        <img src="dongen.jpg" alt="">

        <p>Động Én còn nguyên vẻ hoang sơ nên rất đẹp. Nhiều du khách không khỏi bàng hoàng, sửng sốt khi được trực tiếp chiêm ngưỡng. Đây là một điểm du lịch hấp dẫn của du khách khi đến Yên Minh.</p>

    </div>

    <div class="loz">

            <h1>CHỢ BẠCH ĐÍCH</h1>

            <img src="choyenminh.jpg" alt="">

            <p>Dọc theo đường Quốc lộ 4C từ Tp Hà Giang, sau khoảng 70km, đến địa phận xã Na Khê và cách thị trấn Yên Minh chừng 20 km thì rẽ trái (có biển chỉ dẫn) đi khoảng 20 cây số nữa là đến xã Bạch Đích. Bạch Đích là xã biên giới với 6 thôn bản giáp ranh Trung Quốc, tổng chiều dài đường biên hơn 7 km. Xã có một nét riêng, độc đáo, không nơi nào tại đây có được là có đến 3 phiên chợ trong một tháng, gồm: chợ Bản Muồng, chợ Mốc 358 và chợ trung tâm xã. Các chợ đều họp theo phiên vào ngày Thân và ngày Dần hằng tháng.

                Nói đến chợ Mốc 358 hay còn gọi là chợ Mốc 9, đây vừa là chợ phiên, lại vừa là chợ cửa khẩu; nằm ngay bên chân mốc, đi thêm chục bước chân là chạm tới barier Cửa khẩu Bạch Đích. Chợ họp theo phiên, nếu không đúng phiên, cửa khẩu dù vẫn hoạt động bình thường nhưng khu chợ vắng bóng kẻ mua, người bán. Chợ Mốc 358 được thành lập từ năm 2007, là nơi giao lưu buôn bán và mua sắm hàng hóa giữa nhân dân địa phương với người dân phía bên kia biên giới. Chợ họp từ khoảng 6 giờ đến tầm 12 – 13 giờ trưa, các quầy hàng nằm dọc 2 phía biên giới, cách đường biên chừng 10m.

            </p>

    </div>

    <div class="quan">

        <h1>RƯNG THÔNG </h1>

        <img src="anh16.jpg" alt="">

        <p>Yên Minh cách thành phố Hà Giang khoảng 100km về phía Đông Bắc, men theo quốc lộ 4C chạy từ Cán Tỷ lên trung tậm phố huyện qua ba xã: Bạch Đích, Na Khê, Lao Và Chải có một cung đường đẹp như mơ khiến bạn tựa như đang đứng giữa Đà Lạt mông mơ vậy. Cung đường đẹp rừng thông Yên Minh bắt đầu từ đoạn xã Na Khê cho đến Thị trấn Yên Minh, đoạn quốc lộ 4C với hai bên đường bạt ngàn thông.</p>

    </div>

    <div class="yeu">

        <h1>NÀ LUỒNG </h1>

        <img src="anh13.jpg" alt="">

        <p>Nà Luồng, cách trung tâm thị trấn Yên Minh khoảng 25km, thuộc địa phận thôn Nà Luồng, xã Mậu Long và giáp với xã Lũng Phìn của huyện Đồng Văn mới được tìm thấy năm 2010 được đánh giá là hang đẹp nhất so với các hang đã từng được phát hiện ở Hà Giang

    </p>

    </div>

</body>

</html>

***CSS***

\*{

    padding: 0;

    margin: 0%;

}

img {

    width: 100%;

    height: 250px;

}

p{

    font-size: 25px;

    text-align: left;

}

.sayqua button{

    float: right;

    margin-top: -2px;

    background-color: rgb(34, 14, 211);

    font-size: larger;

}

div h1 {

    text-align:left ;

    color: brown;

}

.caoss h1{

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    text-align: center;

}

.caoss img {

    float: left; /\* Căn ảnh bên trái \*/

    margin-top: -50px;

    width: 160px;

    height: 160px;

}

.chode h1 {

    text-align: center;

}

.chode p{

    margin-left: 10px;

}

.chode img{

    height: 150px;

    width: 240px;

    float: right;

    margin-top: -70px;

}

.loz img{

    margin-right: 5px;

    float: left; /\* Căn ảnh bên trái \*/

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    width: 280px;

    height: 230px;

}

.loz h1 {

    margin-left: 100px;

    text-align: center;

    transform: translate(5%, 9%);

}

.loz p{

    font-size: 25px;

    text-align: left;

}

.quan h1{

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    text-align: center;

}

.quan img {

    margin-right: 5px;

    float: left; /\* Căn ảnh bên trái \*/

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    width: 250px;

    height: 300px;

}

.quan p {

    display: flex;

    align-items: center; /\* Căn giữa theo chiều dọc \*/

}

.yeu h1{

    text-align: center;

}

.yeu h1{

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    text-align: center;

}

.yeu img {

    height: 250px;

    width: 240px;

    float: right;

    margin-top: -70px;

}

.sao img{

    margin-right: 5px;

    float: left; /\* Căn ảnh bên trái \*/

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    width: 270px;

    height: 300px;

}

body p{

    color: aqua;

}

body{

    background: rgb(102, 122, 53);

}

***4.3.6.HTML Mèo Vạc***

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>MÈOVAC</title>

    <link rel="stylesheet" href="meovac.css">

</head>

<body>

    <img src="meo-vac-01\_1687674823.webp" alt="">

    <div class="sayqua">

        <button><a href="btl1.html">TRANG CHỦ</a></button>

    </div>

    <div class="sao">

        <img src="MEO VAC.jpg" alt="">

        <img src="anh3.jpg" alt="">

    </div>

    <div class="caoss">

        <h1>    HUYỆN MÈO VẠC</h1>

        <p>Huyện Mèo Vạc có vị trí địa lý:

            Phía đông và phía bắc giáp với Trung Quốc

            Phía tây giáp huyện Đồng Văn và huyện Yên Minh

            Phía nam giáp huyện Bảo Lâm, tỉnh Cao Bằng.

            Huyện Mèo Vạc có diện tích 574,18 km², dân số năm 2019 là 86.071 người[2], mật độ dân số đạt 150 người/km².

            Huyện Mèo Vạc cùng với các huyện Yên Minh, Đồng Văn, Vị Xuyên là bị thiệt hại nặng trong 2 cuộc chiến tranh biên giới giữa Việt Nam và Trung Quốc. Huyện cũng là điểm sáng trong tinh thần đoàn kết dân tộc khi nhân dân xã Sơn Vĩ (hơn một nửa là dân tộc thiểu số) không sơ tán mà ở lại để chiến đấu và tiếp tế lương thực cho bộ đội.

            Địa hình chủ yếu của huyện là núi đá vôi, có sông Nho Quế chảy qua. Đất nông nghiệp chiếm khoảng 12.100 ha. Sinh hoạt nông nghiệp là trồng trọt và khai thác các loại cây dược liệu, tam thất, hồ đào,... Ngành chăn nuôi có những gia súc bò, dê, ngựa. Có Quốc lộ 4C (con đường hạnh phúc).

        </p>

    </div>

    <div class="chode">

        <h1>ĐÈO MÃ PÌ LÈNG</h1>

        <img src="MÃ PILENG.jpeg" alt="">

        <p>Mã Pí Lèng là tên gọi theo tiếng Quan Hỏa chỉ “sống mũi con ngựa” theo nghĩa đen. Nhưng theo nghĩa bóng tên gọi này chỉ sự hiểm trở bậc nhất của đỉnh núi, nơi những con ngựa cái leo lên đến đỉnh trụy thai mà chết, nơi dốc cao đến mức con ngựa đi qua phải tắt thở, hoặc đỉnh núi dựng đứng như sống mũi con ngựa.Tuy nhiên, theo một số người Hmong bản địa thì tên đúng của đèo là Máo Pì Lèng, nghĩa là “sống mũi con mèo”.

            Đỉnh Mã Pí Lèng thuộc ba xã Pải Lủng, Pả Vi và Xín Cái (Mèo Vạc, Hà Giang) trong cao nguyên đá Đồng Văn có độ cao khoảng 2000m so với mặt nước biển, được tạo nên bởi một loạt trầm tích gồm đá vôi, đá phiến ánh, đá vôi silic chứa các hóa thạch cách đây khoảng 426 triệu năm, bao gồm trong đó nhiều vết trượt và vết nứt do các hoạt động tạo núi gây ra. Cảnh quan khu vực này lởm chởm đá dựng, trong đó vực sâu sông Nho Quế như xẻ đôi một bên là đỉnh Mã Pí Lèng và một bên là Săm Pun (Sam Pun), nơi có cột mốc biên giới và cửa khẩu thông thương từ Xín Cái sang Điền Bồng, Trung Quốc. Các học giả Pháp, từ cả trăm năm trước, đã gọi đỉnh Mã Pí Lèng nói riêng và Công viên địa chất toàn cầu Cao nguyên đá Đồng Văn nói chung là một “Tượng đài Địa chất”. Đi trên đèo Mã Pí Lèng nhìn xuống vực sâu, con sông Nho Quế chỉ bé như một sợi chỉ mà muốn xuống đến mặt nước của sông phải mất hơn một ngày đường.

        </p>

    </div>

    <div class="loz">

            <h1>PARORAMA</h1>

            <img src="par.webp" alt="">

            <p>Vốn là một quán cafe, nhà nghỉ với tầm nhìn hướng thẳng xuống dòng sông Nho Quế. Tuy vậy, sau những ồn ào kéo dài hàng năm trời,  nơi đây giờ chỉ còn là một điểm để ngắm cảnh (có thu phí), không được phép kinh doanh dịch vụ ngủ nghỉ nữa.

            </p>

    </div>

    <div class="quan">

        <h1>SÔNG NHO QUẾ </h1>

        <img src="anh`17.jpg" alt="">

        <P>Là một trong những biểu tượng tươi xanh của Mèo Vạc Hà Giang, dòng sông Nho Quế nằm dưới chân đèo Mã Pí lèng quanh năm êm đềm chảy qua những vách đá, tạo thành một đường ranh giới màu xanh biếc giữa các con đèo ở đây.Con sông hiền hòa ấy trong lành nhất là vào những ngày đầu mùa hạ hoặc thu nhưng thời điểm lý tưởng nhất trong năm vẫn là những ngày tháng 11 khi mà dòng nước chuyển màu xanh như ngọc tưởng chừng như một “Tuyệt Tình Cốc” của miền Bắc vậy.</P>

    </div>

    <div class="yeu">

        <h1>CHỢ TÌNH KHÂU VAI </h1>

        <img src="QUANAODANTOC.jpg" alt="">

       <p>Nhắc đến những trải nghiệm thú vị ở thị trấn Mèo Vạc không thể bỏ qua chợ tình Khâu Vai. Đây là một trong những điểm nhấn độc đáo và chưa bao giờ ngừng hot luôn níu chân du khách mỗi khi có dịp ghé thăm mảnh đất này.Du lịch Hà Giang đâu chỉ có cao nguyên đá Đồng Văn hay cột cờ Lũng Cú, nơi đây còn có một thị trấn Mèo Vạc tuy nhỏ nhưng sở hữu vô vàn những điều thú vị níu chân du khách mỗi lần ghé thăm. Vậy còn chần chừ gì mà không set kèo cùng với hội bạn để khám phá mảnh đất đáng mến này.

        </p>

    </div>

</body>

</html>

***CSS***

\*{

    padding: 0;

    margin: 0%;

}

img {

    width: 100%;

    height: 250px;

}

p{

    font-size: 25px;

    text-align: left;

}

.sayqua button{

    float: right;

    margin-top: -2px;

    background-color: rgb(34, 14, 211);

    font-size: larger;

}

div h1 {

    text-align:left ;

    color: brown;

}

.sao img{

    margin-right: 5px;

    float: left; /\* Căn ảnh bên trái \*/

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    width: 270px;

    height: 380px;

}

.caoss h1{

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    text-align: center;

}

.caoss img {

    float: left; /\* Căn ảnh bên trái \*/

    margin-top: -0px;

    width: 100px;

    height: 190px;

}

.chode h1 {

    text-align: center;

}

.chode img{

    height: 390px;

    width: 370px;

    float: right;

    margin-top: -15px;

}

.loz img{

    margin-right: 5px;

    float: left; /\* Căn ảnh bên trái \*/

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    width: 300px;

    height: 340px;

}

.loz h1 {

    margin-left: 100px;

    text-align: center;

    transform: translate(5%, 9%);

}

.loz p{

    font-size: 25px;

    text-align: left;

}

.quan h1{

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    text-align: center;

}

.quan img {

    float: right; /\* Căn ảnh bên trái \*/

    margin-top: -40px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    width: 300px;

    height: 400px;

}

.quan p {

    display: flex;

    align-items: center; /\* Căn giữa theo chiều dọc \*/

}

.yeu h1{

    text-align: center;

}

.yeu h1{

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    text-align: center;

}

.yeu img {

    float: left; /\* Căn ảnh bên trái \*/

    margin-top: -0px;

    width: 280px;

    height: 270px;

}

body p{

    color: aqua;

}

body{

    background: rgb(102, 122, 53);

}

***4.3.7.HTML CSS Bắc Mê***

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>QUÊ TUI</title>

    <link rel="stylesheet" href="bacme.css">

</head>

<body>

    <img src="512px-MinhNgoc-NaSai'2007.webp" alt="">

    <div class="sayqua">

        <button><a href="btl1.html">TRANG CHỦ</a></button>

    </div>

    <div class="sao">

        <img src="HUYENBACME.jpg" alt="">

    </div>

    <div class="caoss">

        <h1>    HUYỆN BẮC MÊ</h1>

        <P>Bắc Mê là một huyện có 2 kiểu địa hình phổ biến đó là miền núi cao khoảng 900  đến 1400m so với mực nước biển, khu thấp hơn có độ cao chỉ 100 đến 500m so với mức nước biển. Có rất nhiều điểm vui chơi, tham quan đáng lưu ý tại đây.</p>

    </div>

    <div class="chode">

        <h1>HANG BẮC SƠM</h1>

        <img src="dongen.jpg" alt="">

        <p>Hang Bách Sơn  được phát hiện tại khu rừng nguyên sinh có độ cao trên 200m so với mực nước lòng hồ thủy điện Tuyên Quang, thuộc thôn Bách Sơn, xã Thượng Tân, huyện Bắc Mê (Hà Giang) cách trung tâm huyện khoảng 1h đi thuyền

            Hang Bách Sơn nằm trên tuyến du lịch lòng hồ thủy điện Bắc Mê – Nà Hang (Tuyên Quang) – Ba Bể (Bắc Kạn) và nằm ẩn trong khu rừng nguyên sinh của dãy núi đá Bách Sơn. Cửa hang rộng khoảng 20m được che phủ bởi rừng nghiến rậm rạp, lòng hang rộng, sâu, có nhiều nhũ đá với các hình thù kỳ ảo.

        </p>

    </div>

    <div class="loz">

        <h1>CĂNG BẮC MÊ</h1>

            <img src="cang2.webp" alt="">

            <p>Di tích lịch sử Căng Bắc Mê được người Pháp chọn xây dựng có địa thế khá lý tưởng ở xã Yên Cường nhằm kiểm soát toàn bộ tuyến giao thông nối 3 tỉnh Hà Giang, Cao Bằng và Tuyên Quang. Căng được xây dựng gồm hệ thống nhà giam, vọng gác, nhà thông tin của thực dân Pháp, trước kia, đây là đồn binh của thực dân Pháp. Năm 1938, khi phong trào cách mạng ở Việt Nam phát triển lên đỉnh cao mới thì thực dân Pháp biến nơi này thành địa điểm giam giữ cán bộ cách mạng, chúng chuyển một số tù nhân chính trị từ Sơn La, Hoả Lò, Phú Thọ…lên đây giam giữ.

                Từ năm 1938-1942, thực dân Pháp đã 2 lần đưa tù nhân chính trị đến giam tại đây với số lượng khoảng hơn 300 người, trong đó có các đồng chí Xuân Thuỷ, Lê Giản, Nguyễn Văn Ngọc, Nguyên Hồng . Trong số những tù nhân đó, đa số các đồng chí là đảng viên Đảng Cộng sản Đông Dương. Mặc cho gông cùm khổ cực, khí hậu khắc nghiệt nhưng các đồng chí vẫn tìm cách vận động đấu tranh, đòi cải thiện chế độ sinh hoạt trong Căng, dạy văn hoá, bồi dưỡng cho nhau về tinh thần yêu nước và tư tưởng đấu tranh cách mạng. Các đồng chí còn tranh thủ giác ngộ thanh niên, quần chúng trong và ngoài Căng. Tới cuối năm 1942, lo sợ phong trào cách mạng đang lan rộng ở các tỉnh biên giới và đấu tranh của anh em tù chính trị trong Căng, sợ sự ảnh hưởng của anh em tù nhân đến bà con quanh vùng, thực dân Pháp đã phải giải tán Căng Bắc Mê.

            </P>

    </div>

    <div class="quan">

        <h1>SÔNG GÂM </h1>

        <img src="songgam.webp" alt="">

        <P>Sông Gâm chảy qua huyện Bắc Mê được ví như là một dải lụa của tự nhiên ôm lấy miền núi đá. Xung quanh đây là một khung cảnh nên thơ trữ tình với đầy đủ. So với dòng sông khác cũng rất nổi tiếng của Hà Giang đó là sông Nho Quế thì sông Gâm không hiểm trở bằng, mà ngược lại laaij rất yên ả, bằng phẳng, yên giả giữa màu xanh của rừng cây và rêu phong mọc trên núi đá, hình ảnh vốn rất quen thuộc khi đến với Hà Giang. Nhờ sự yên ả của mình mà sông Gâm thường xuyên được chọn để tổ chức những lễ hội đua thuyền hoặc những tour tham quan du lịch trên nước tại Hà Giang, trải nghiệm cực kì thú vị.

        </P>

    </div>

</body>

</html>

***CSS***

\*{

    padding: 0;

    margin: 0%;

}

img {

    width: 100%;

    height: 250px;

}

p{

    font-size: 25px;

    text-align: left;

}

div h1 {

    text-align:left ;

    color: brown;

}

.sao img{

    margin-right: 5px;

    float: left; /\* Căn ảnh bên trái \*/

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    width: 270px;

    height: 330px;

}

.caoss h1{

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    text-align: center;

}

.caoss img {

    float: left; /\* Căn ảnh bên trái \*/

    margin-top: -0px;

    width: 100px;

    height: 110px;

}

.chode h1 {

    text-align: center;

}

.chode img{

    height: 140px;

    width: 200px;

    float: right;

    margin-top: -15px;

}

.loz img{

    margin-right: 5px;

    float: left; /\* Căn ảnh bên trái \*/

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    width: 300px;

    height: 400px;

}

.loz h1 {

    margin-left: 100px;

    text-align: center;

    transform: translate(5%, 9%);

}

.loz p{

    font-size: 25px;

    text-align: left;

}

.quan h1{

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    text-align: center;

}

.quan img {

    float: right; /\* Căn ảnh bên trái \*/

    margin-top: -40px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    width: 300px;

    height: 250px;

}

.quan p {

    display: flex;

    align-items: center; /\* Căn giữa theo chiều dọc \*/

}

.yeu h1{

    text-align: center;

}

.yeu h1{

    margin-left: 10px; /\* Tạo khoảng cách giữa ảnh và nội dung bên trái \*/

    text-align: center;

}

.yeu img {

    float: left; /\* Căn ảnh bên trái \*/

    margin-top: -0px;

    width: 200px;

    height: 210px;

}

body p{

    color: aqua;

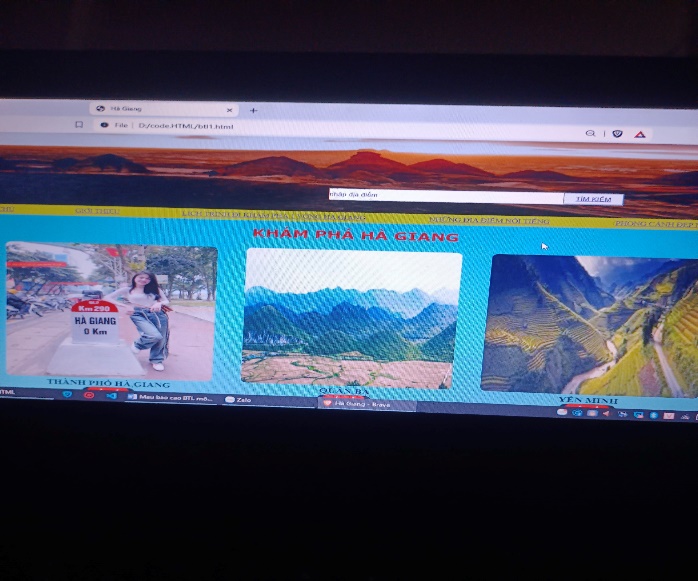
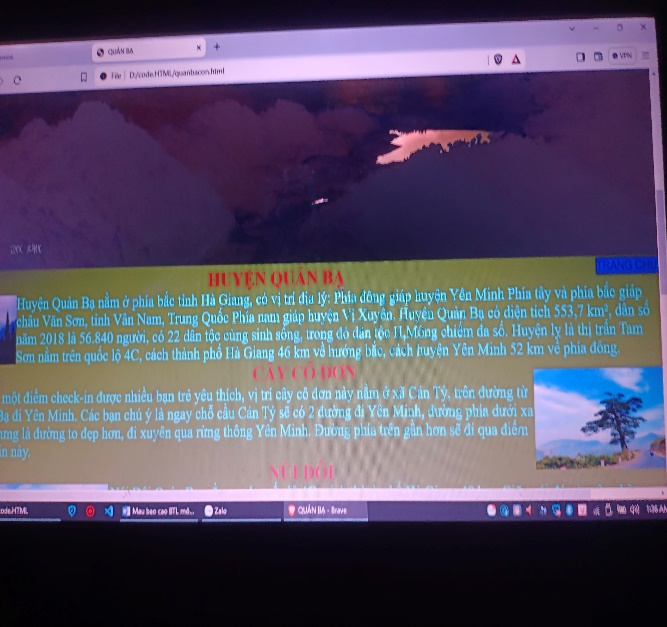
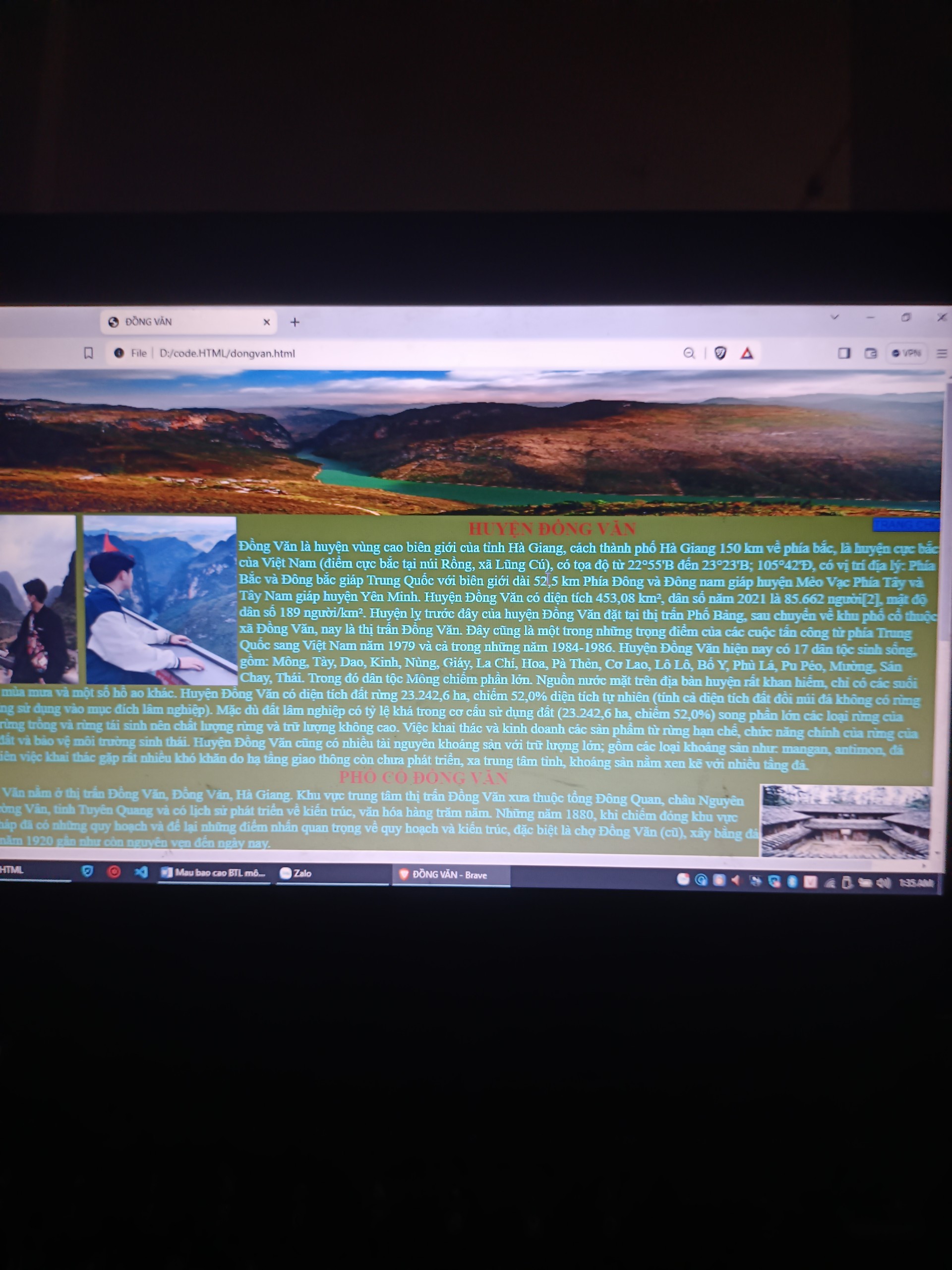
}

body{

    background: rgb(102, 122, 53);

}

**4.5. Kết quả đạt được**

Ta sẽ có 1 wed về tỉnh Hà Giang ,những vùng đất ,phong cảnh đẹp nhất 

**Kết luận chương 4**

Tổng hợp từ chương 1 đến chương 3 đã giúp tôi làm được 1 wed của riêng mình

Làm wed cần một sự kiên nhẫn và một óc sang tạo ,mày mò**KẾT LUẬN**

***(Trình bày thành 3 đoạn văn nêu Ưu điểm, nhược điểm và hướng phát triển chủ đề)***

Bài làm

Hướng phát triển du lịch Hà Giang

Du lịch Hà Giang đang trên đà phát triển mạnh mẽ, thu hút du khách bởi cảnh quan hùng vĩ, hoang sơ và văn hóa độc đáo của các dân tộc thiểu số. Để phát triển du lịch bền vững, Hà Giang cần tập trung vào các hướng sau:

Phát triển du lịch cộng đồng: Khuyến khích người dân địa phương tham gia vào hoạt động du lịch, tạo nguồn thu nhập và nâng cao đời sống.

Bảo vệ môi trường: Giữ gìn cảnh quan thiên nhiên hoang sơ, hùng vĩ của Hà Giang, hạn chế tác động tiêu cực của du lịch.

Phát triển cơ sở hạ tầng: Nâng cấp hệ thống giao thông, đường sá, nhà hàng, khách sạn để đáp ứng nhu cầu của du khách.

Đa dạng hóa sản phẩm du lịch: Phát triển các loại hình du lịch mới như du lịch sinh thái, du lịch văn hóa, du lịch mạo hiểm.

Ưu điểm của du lịch Hà Giang

Cảnh quan thiên nhiên hùng vĩ: Hà Giang sở hữu những cung đường đèo đẹp nhất Việt Nam như Mã Pí Lèng, Cổng Trời Quản Bạ, Núi đôi Cô Tiên.

Văn hóa độc đáo: Hà Giang là nơi sinh sống của nhiều dân tộc thiểu số với những phong tục tập quán độc đáo, thu hút du khách.

Con người thân thiện: Người dân Hà Giang nổi tiếng với sự thân thiện, mến khách, luôn sẵn sàng giúp đỡ du khách.

Ẩm thực phong phú: Hà Giang có nhiều món ăn đặc sản hấp dẫn như bánh cuốn trứng, thắng cố, mèn mén, rượu ngô.

Nhược điểm của du lịch Hà Giang

Cơ sở hạ tầng còn hạn chế: Hệ thống giao thông ở một số khu vực còn khó khăn, nhà hàng, khách sạn chưa đáp ứng đủ nhu cầu của du khách.

Dịch vụ du lịch chưa chuyên nghiệp: Một số điểm du lịch còn thiếu các dịch vụ du lịch cần thiết như hướng dẫn viên, thông tin du lịch.

Môi trường bị tác động: Hoạt động du lịch chưa được quản lý chặt chẽ dẫn đến tình trạng rác thải, ô nhiễm môi trường.

Giá cả dịch vụ cao: Giá cả dịch vụ du lịch tại Hà Giang tương đối cao so với mặt bằng chung.

**DANH MỤC TÀI LIỆU THAM KHẢO**

1. *Nguyễn Thanh Hải, NXB Đại học Quốc gia Hà Nội, 2017*
2. *Nguyễn Minh Châu, NXB Thanh Niên, 2016*
3. *Nguyễn Đức Hùng, NXB Bách Khoa, 2015*
4. *Elisabeth Robson, Kathy Sierra, Bert Bates, 2012*
5. *Andy Budd, 2012*