**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 website bán hàng ( bất động sản )**

**Giảng viên hướng dẫn: ThS. Nguyễn Duy Khương**

**Sinh viên thực hiện: Bùi Quang Trường**

**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 website bán hàng ( bất động sản )**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Mã Sinh Viên | Họ và Tên | Ngày Sinh | Điểm | |
| Bằng Số | Bằng Chữ |
| 1 | 1771020700 | Bùi Quang Trường | 02/01/2005 |  |  |

### 

|  |  |
| --- | --- |
| CÁN BỘ CHẤM THI 1 | CÁN BỘ CHẤM THI 2 |
|
|  |  |

**Hà Nội, 2024**

**LỜI NÓI ĐẦU**

(Nếu có)

Trong thời đại công nghệ số ngày nay, việc có một trang web bán hàng hiệu quả không chỉ là một yêu cầu mà còn là một cơ hội để kinh doanh mở rộng thị trường và tạo ra sự tiện lợi cho khách hàng. Đề tài này tập trung vào việc thiết kế và triển khai một hệ thống phần mềm đa năng, đáp ứng đầy đủ các yêu cầu của một trang web bán hàng hiện đại.

Chúng tôi đã kết hợp sự hiểu biết sâu rộng về kỹ thuật phần mềm, trải nghiệm người dùng, và nhu cầu thị trường để xây dựng một trải nghiệm mua sắm trực tuyến tối ưu. Đồng thời, chúng tôi cũng chú trọng đến việc tạo ra một hệ thống linh hoạt, dễ dàng mở rộng và tùy chỉnh để phù hợp với sự phát triển của doanh nghiệp và sự thay đổi của thị trường.

Bằng việc kết hợp các kỹ thuật tiên tiến như responsive design, hệ thống thanh toán an toàn, và quản lý hàng tồn kho thông minh, chúng tôi tin rằng dự án này sẽ mang lại giá trị lâu dài cho cả doanh nghiệp và khách hàng. Chúng tôi rất vui được chia sẻ kết quả của dự án này và hy vọng nó sẽ đem lại những thành công đáng kinh ngạc.

**MỤC LỤC**

[STT 2](#_Toc160200772)

[Mã Sinh Viên 2](#_Toc160200773)

[Họ và Tên 2](#_Toc160200774)

[Ngày Sinh 2](#_Toc160200775)

[Điểm 2](#_Toc160200776)

[Bằng Số 2](#_Toc160200777)

[Bằng Chữ 2](#_Toc160200778)

[1 2](#_Toc160200779)

[1771020700 2](#_Toc160200780)

[Bùi Quang Trường 2](#_Toc160200781)

[02/01/2005 2](#_Toc160200782)

[CÁN BỘ CHẤM THI 1 2](#_Toc160200783)

[CÁN BỘ CHẤM THI 2 2](#_Toc160200784)

[Trần Thị Huệ 2](#_Toc160200785)

[**LỜI NÓI ĐẦU** 3](#_Toc160200786)

[**MỤC LỤC** 4](#_Toc160200787)

[**MỤC LỤC HÌNH ẢNH** 6](#_Toc160200788)

[**CHƯƠNG I: GIỚI THIỆU VỀ THIẾT KẾ WEB VÀ TRIỂN KHAI HỆ THỐNG PHẦN MỀM** 9](#_Toc160200789)

[**1.1.** **Các khái niệm cơ bản** 9](#_Toc160200790)

[1.1.1 Mạng internet 9](#_Toc160200791)

[a. Mạng Internet là gì? 9](#_Toc160200792)

[d. Lịch sử hình thành và phát triển của mạng Internet 12](#_Toc160200793)

[Internet hoạt động như thế nào? 18](#_Toc160200794)

[1.1.2 Kĩ thuật internet 19](#_Toc160200795)

[Tổng quan về mặt kỹ thuật[ 22](#_Toc160200796)

[WiFi và Ethernet 26](#_Toc160200797)

[Độ trễ của Ethernet thấp như thế nào? 28](#_Toc160200798)

[Vậy khi nào nên sử dụng Ethernet? 30](#_Toc160200799)

[Điểm khác biệt giữa 4G và 5G 33](#_Toc160200800)

[5G đang ứng dụng tại đâu ở Việt Nam? 35](#_Toc160200801)

[5G VinaPhone - Dẫn đầu cuộc cách mạng 4.0 36](#_Toc160200802)

[1.3. Một số thuật ngữ (Hosting, Tên miền-Domain Name-Web Server Name, HomePage, 40](#_Toc160200803)

[Web Site, WebServer, URL-Uniform Resource Location, Browser, …) 40](#_Toc160200804)

[1.3. Một số công cụ dùng thiết kế Web (Visual Studio code, Adobe Dreamweaver, …) 42](#_Toc160200805)

[1.4 Kết luận 44](#_Toc160200806)

[**CHƯƠNG II: HTML & HTML5** 46](#_Toc160200807)

[**(**Hyper Text Markup Language**)** 46](#_Toc160200808)

[2.1. Tổng quan về HTML 46](#_Toc160200809)

[2.2 Cấu trúc tổng quát trang HTML 50](#_Toc160200810)

[2.4. Các thẻ tạo biểu mẫu (form) 63](#_Toc160200811)

[2.5. Một số thẻ HTML đặc biệt 77](#_Toc160200812)

[2.6. HTML5 96](#_Toc160200813)

[2.2. Cập nhật ngữ nghĩa 98](#_Toc160200814)

[**3. Các thành phần HTML5:** 100](#_Toc160200815)

[3.1. Các thẻ gom nhóm: 100](#_Toc160200816)

[3.2. <figure></figure> 111](#_Toc160200817)

[3.3. <time></time> 112](#_Toc160200818)

[3.4. HTML5 Form: 113](#_Toc160200819)

[Hướng dẫn xếp tài liệu tham khảo 127](#_Toc160200820)

**MỤC LỤC HÌNH ẢNH**

**(Nếu có)**

**MỤC LỤC BẢNG**

**(Nếu có)**

**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 |
| 3 | JS | JavaScrip |
| 4 | URL | Uniform Resource Locator |
| 5 | HTTP | HyperText Transfer Protocol |
| 6 | HTTPS | HyperText Transfer Protocol Secure |
| 7 | API | Application Programming Interface |
| 8 | DOM | Document Object Model |
| 9 | SVG | Scalable Vector Graphics |
| 10 | DOCTYPE | Document Type Declaration |
| 11 | XML | eXtensible Markup Language |
|  | XHTML | eXtensible HyperText Markup Language |
|  | W3C | World Wide Web Consortium |
|  | SEO | Search Engine Optimization |
|  | CMS | Content Management system |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

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

* 1. **Các khái niệm cơ bản**
     1. Mạng internet

Ngày nay, mạng Internet có mặt ở khắp mọi nơi và là một phần không thể thiếu trong cuộc sống hiện đại. Nhưng liệu bạn có hiểu hết về hệ thống mạng này? Bài viết sau đây sẽ giới thiệu rõ hơn về lịch sử hình thành, nguyên tắc hoạt động và cách phân biệt với wifi. Điều này sẽ giúp bạn đọc hiểu thêm về mạng Internet và biết cách sử dụng hiệu quả hơn.

## a. Mạng Internet là gì?

Mạng Internet là hệ thống thông tin toàn cầu gồm nhiều mạng máy tính liên kết với nhau, truyền thông tin theo kiểu nối chuyển gói dữ liệu (packet switching) dựa trên một giao thức liên mạng đã được chuẩn hóa (giao thức IP). Hệ thống cho phép người dùng truy cập công cộng miễn phí.



Mạng Internet là hệ thống thông tin toàn cầu giúp kết nối các thiets bị ở khắp mọi nơi trên thế giới.

*b Lợi ích của mạng Internet*

Là hệ thống mạng được phủ sóng toàn cầu, mạng Internet mang lại cho con người nhiều lợi ích như:

1 - Internet là kho tàng kiến thức khổng lồ lưu trữ tri thức của nhân loại: Số lượng kiến thức lưu trữ trên Internet là rất lớn. Khi cần thông tin, bạn không phải tìm kiếm trên sách vở mà có thể tìm kiếm ngay trên mạng Internet. Hệ thống sẽ cung cấp thông tin cho bạn ngay lập tức. Hơn nữa, Internet còn có thể truyền tải thông tin tới nhiều người tìm kiếm cùng một lúc và tất cả đều miễn phí. Nhờ Internet, bạn có thể dễ dàng tìm tòi, nghiên cứu, học hỏi để mở mang kiến thức và nâng cao trình độ bản thân.

2 - Internet là nơi thực hiện hoạt động kinh doanh, buôn bán tiện lợi: Mạng Internet là nơi kết nối người bán hàng với người mua hàng, doanh nghiệp với khách hàng. Đây là nơi người bán dùng để quảng cáo, giới thiệu, đăng bán sản phẩm, dịch vụ; người mua dùng mạng Internet để tìm kiếm, so sánh giá cá, đặt mua sản phẩm, dịch vụ một cách nhanh chóng. Nhờ đó, việc kinh doanh, buôn bán online diễn ra thuận lợi, dễ dàng. Hơn nữa, Internet còn giúp con người thực hiện việc thanh toán online như chuyển khoản, thanh toán hóa đơn điện, nước… trở nên dễ dàng.

3 - Internet là thế giới ảo giải trí đa dạng: Khi sử dụng mạng Internet, người dùng có thể thực hiện nhiều hoạt động giải trí khác nhau như nghe nhạc, xem phim, chơi game, đọc truyện, đọc báo… Những hoạt động này giúp người dùng cảm thấy thoải mái, được thư giãn mà lại tốn ít sức lực, tiết kiệm thời gian và rất thuận tiện.

4 - Giúp kết nối mọi người: Người dùng có thể dễ dàng kết nối với người khác bằng cách nhắn tin, gọi điện, gọi video, gửi email… thông qua các ứng dụng mạng xã hội trên Internet. Không những thế, Internet còn giúp kết nối người dùng ở cách xa nhau trong giảng dạy, học tập, hội họp, làm việc nhóm, thảo luận… Nhờ đó, con người đến gần nhau hơn. Mọi công việc, học tập trở nên dễ dàng. Đặc biệt, trong thời kỳ dịch bệnh Covid kéo dài, con người không thể tiếp xúc trực tiếp với nhau thì việc này càng trở nên có ý nghĩa.



Mạng Internet kết nối mọi người ở khoảng cách xa lại gần nhau hơn thông qua các ứng dụng, trang mạng xã hội, phần mềm…

c. Tác động tiêu cực của Internet

Tuy có nhiều lợi ích nổi bật nhưng mạng Internet không phải là không có những tác động tiêu cực, đó là:

1 - Khiến con người lười vận động hơn: Các hình thức giải trí trên Internet rất phong phú. Vì thế, một số người đã dành quá nhiều thời gian để lướt web, chơi game, xem phim… và trở thành những “con nghiện” Internet. Việc sử dụng nhiều Internet, ít vận động đã gây ra một loại các bệnh về mắt, xương khớp và sức khỏe giảm sút, sức đề kháng của con người cũng kém hơn.

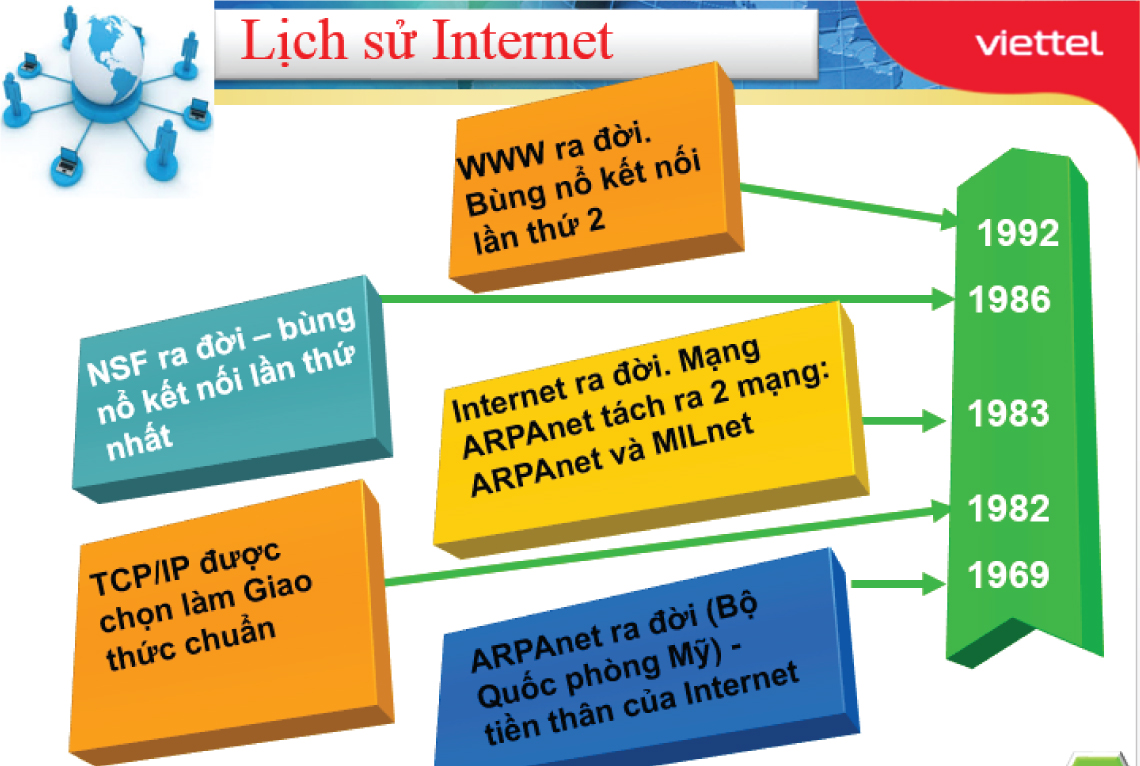
2 - Không kiểm soát được nội dung độc hại: Nhiều nội dung trên mạng Internet không được kiểm duyệt có thể đầu độc tâm hồn thế hệ trẻ gây ra những tư tưởng lệch lạc. Thậm chí, nếu sử dụng không đúng Internet còn là nơi tiếp tay cho nhiều tệ nạn xã hội như đánh bạc, lừa đảo, cho vay nặng lãi, tội phạm an ninh mạng…



Một số người trở thành “con nghiện” Internet và luôn cảm thấy mệt mỏi, sức khỏe giảm sút.

## Lịch sử hình thành và phát triển của mạng Internet

Mạng Internet ra đời từ những năm 1960 với tên gọi là ARPANET. Mạng bắt nguồn từ sự phát triển của chuyển mạch gói và nghiên cứu do Bộ Quốc phòng Hoa Kỳ ủy quyền thực hiện để cho phép chia sẻ thời gian của máy tính. Sự ra đời này đã đánh dấu một bước tiến dài trong lịch sử phát triển của nhân loại và là cuộc cách mạng lớn trong lĩnh vực công nghệ thông tin. Mạng Internet là hệ thống trao đổi thông tin do Paul Baran phát minh. Sau đó, ông kết hợp với Leonard Kleinrock và Donald Davies để hiện thực hóa ý tưởng của mình.



Tóm lược quá trình hình thành và phát triển của mạng Internet.

Dưới đây là những dấu mốc quan trọng đánh dấu sự ra đời và phát triển của mạng Internet.

|  |  |
| --- | --- |
| **Thời gian** | **Sự kiện** |
| Năm 1960 | Cơ quan Dự án Nghiên cứu Tiên tiến (ARPA) của Bộ Quốc phòng Hoa Kỳ  đã tài trợ cho nghiên cứu về việc chia sẻ thời gian của máy tính. Paul Baran  bắt đầu nghiên cứu về chuyển mạch gói, một trong những công nghệ Internet  cơ bản. |
| Năm 1965 | Donald Davies nghiên cứu về chuyển mạch gói. |
| Năm 1967 | Chuyển mạch gói được tích hợp vào thiết kế đề xuất cho ARPANET. |
| Năm 1969 | Cơ quan quản lý dự án nghiên cứu phát triển ARPA (Bộ Quốc Phòng Mỹ) tiếp  tục phát triển ARPANET. Bốn địa điểm đầu tiên là Viện nghiên cứu Stanford,  Đại học California – Los Angeles, Đại học California – Santa Barbara  và Đại học Utah được kết nối bằng mạng liên khu vực (Wide Area Network –  WAN). Lúc này, các máy tính được liên kết với nhau và có khả năng tự định  đường truyền tin ngay sau khi một phần mạng đã được phá hủy. |
| Năm 1970 | ARPANET là xương sống kết nối các mạng lưới học thuật và quân sự khu vực. Các mạng chuyển mạch gói khác như mạng NPL, mạng Merit và CYCLADES được phát triển. |
| Cuối năm 1971 | Mười lăm trang web đã được kết nối với ARPANET. |
| Năm 1972 | Trong một cuộc hội nghị quốc tế về truyền thông máy tính, Bob Kahn đã trình diễn mạng ARPANET liên kết 40 máy thông qua các bộ xử lý giao tiếp giữa những trạm cuối Terminal Interface Processor-TIP. Nhằm đáp ứng nhu cầu thiết lập giao thức bắt tay (agreed-upon), nhóm Internet Working Group (INWG) do Vinton Cerf làm chủ tịch được thành lập. Ray Tomlinson phát minh ra Email để gửi thông điệp trên mạng. |
| Năm 1973 | Hợp tác quốc tế trên mạng ARPANET lần đầu tiên. Các kết nối đã được thực hiện với Mảng địa chấn Na Uy (NORSAR) thông qua một trạm vệ tinh ở Tanum, Thụy Điển và nhóm nghiên cứu của Peter Kirstein tại Đại học College London. Dự án ARPANET và các nhóm làm việc quốc tế kết hợp đã dẫn đến sự phát triển của các giao thức và tiêu chuẩn khác nhau, nhiều mạng riêng biệt có thể trở thành một mạng hoặc "một mạng các mạng". Đại học Harvard, Bob Metcalfe đã hình thành ý tưởng về ethernet (một giao thức trong mạng cục bộ). Vinton Cerf và Bob Kahn đề xuất những vấn đề cơ bản của Internet. Đây chính là những nét chính của giao thức TCP/IP. |
| Năm 1974 | Thuật ngữ Internet được Vint Cerf và Bob Kahn nhắc đến lần đầu tiên như một cách viết tắt cho mạng nội bộ RFC 675 và các RFC. Các nhà cung cấp PTT thương mại đã quan tâm đến việc phát triển mạng dữ liệu công cộng X.25. BBN xây dựng giao thức ứng dụng Telnet cho phép sử dụng máy tính từ xa. |
| Năm 1976 | Phòng thí nghiệm của hãng AT&T phát minh ra dịch vụ truyền tệp cho mạng FTP. |
| Năm 1978 | Tom Truscott và Steve Bellovin thiết lập mạng USENET dành cho những người sử dụng Unix. Mạng USENET là một trong những mạng phát triển sớm nhất và thu hút nhiều người nhất. |
| Năm 1979 | ARPA thành lập ban kiểm soát cấu hình Internet. |
| Năm 1980 | Quyền truy cập vào ARPANET được mở rộng. Quỹ khoa học quốc gia (NSF) tài trợ ARPANET cho Mạng khoa học máy tính (CSNET) và tài trợ tư nhân cho các phần mở rộng thương mại khác. Điều này làm cho toàn thế giới đều tham gia phát triển các công nghệ mạng mới và sát nhập nhiều mạng. |
| Năm 1981 | Mạng CSNET (Computer Science NETwork) ra đời. Hệ thống này cung cấp những dịch vụ mạng cho các nhà khoa học ở trường đại học mà không cần truy cập vào mạng ARPANET. |
| Năm 1982 | DAC và ARPA dùng mạng ARPANET với các giao thức TCP và IP. Sau đó, bộ giao thức Internet (TCP/IP) đã được chuẩn hóa giúp phổ biến các mạng kết nối trên toàn thế giới. |
| Năm 1983 | Giao thức TCP/IP được coi là chuẩn trong ngành quân sự Mỹ. Tất cả các máy tính kết nối với ARPANET đều phải dùng chuẩn mới này. |
| Năm 1984 | ARPANET được chia làm hai thành phần là ARPANET (chuyên về nghiên cứu, phát triển và là mạng dân sự) và MILNET (tích hợp dữ liệu quốc phòng và là mạng phục vụ cho mục đích quân sự). Những ưu điểm của giao thức TCP/IP và các chính sách mở cửa của chính phủ đã thúc đẩy việc nghiên cứu và thương mại qua ARPANET. Đây cũng là tiền đề cho sự hình thành của siêu mạng trong tương lai. Hội đồng các hoạt động Internet ra đời. Sau đó, hội đồng này được đổi tên thành hội đồng kiến trúc Internet. |
| Năm 1985 | Tổ chức khoa học quốc gia Mỹ NSF thành lập NSFNET để liên kết các trung tâm máy tính lớn với nhau. Cơ quan quản lý viễn thông Mỹ mở cửa cho một số băng tần của mạng không dây để cho người dùng sử dụng mà không cần giấy phép của chính phủ. Nhờ đó, các nhà cung cấp thiết bị không dây dùng cho mạng LAN đã phát triển các sản phẩm độc quyền của mình. Tuy nhiên, sản phẩm của các công ty không tương thích với nhau, gây rắc rối cho người dùng. Điều này đòi hỏi phải xác lập một chuẩn không dây chung. |
| Năm 1986 | Truy cập mạng TCP/IP được mở rộng trở lại. Mạng NSFnet chính thức được thiết lập, kết nối năm trung tâm máy tính. Mạng Khoa học Quốc gia (NSFNet) cung cấp quyền truy cập vào các trang web siêu máy tính ở Hoa Kỳ cho các nhà nghiên cứu. Đây là năm có sự bùng nổ kết nối, đặc biệt là ở các trường đại học. Trong thời điểm này, NSF và ARPANET song song tồn tại theo cùng 1 giao thức, có kết nối với nhau. |
| Năm 1988 | NSFNet đã mở rộng thành các tổ chức nghiên cứu và học thuật ở Châu u, Úc, New Zealand và Nhật Bản. Dù các giao thức mạng khác như UUCP đã tiếp cận toàn cầu trước thời điểm này, nhưng đây chính là sự khởi đầu của Internet với tư cách một mạng lưới liên lục địa. |
| Năm 1989 | Các nhà cung cấp dịch vụ Internet thương mại (ISP) xuất hiện tại Hoa Kỳ và Úc. MCI Mail và Compuserve đã thiết lập các kết nối với Internet, cung cấp email và các sản phẩm truy cập công cộng tới nửa triệu người dùng Internet. |
| Năm 1990 | PSInet đề xuất mạng Internet thay thế cho mục đích thương mại. Đây là một trong những mạng được thêm vào cốt lõi của Internet thương mại sau này.  - Liên kết T1 (1,5 Mbit/s) tốc độ cao đầu tiên giữa NSFNET và châu u đã được cài đặt giữa Đại học Cornell và CERN. Liên kết này cho phép khả năng liên lạc hơn nhiều so với khả năng của các vệ tinh.  - Dự án ARPANET chính thức ngừng hoạt động. Tuy nhiên, mạng do NSF và ARPANET tạo ra đã được sử dụng vào mục đích dân dụng. Đây chính là tiền thân của mạng Internet ngày nay. Lúc này, một số hãng lớn bắt đầu tổ chức kinh doanh trên mạng.  - Đối tượng sử dụng Internet chủ yếu trong thời gian này là những nhà nghiên cứu. Dịch vụ phổ biến nhất là email và FTP và Internet là 1 phương tiện đại chúng. Ước tính lưu lượng truy cập trên Internet công cộng tăng 100% mỗi năm, mức tăng trưởng số lượng người dùng Internet hàng năm là từ 20 – 50 %.  - Tim Berners-Lee bắt đầu viết WorldWideWeb, trình duyệt web đầu tiên sau hai năm vận động hành lang ban quản lý CERN. Ông đã xây dựng tất cả các công cụ cần thiết cho một web hoạt động như giao thức truyền tải siêu văn bản (HTTP) 0.9, ngôn ngữ đánh dấu siêu văn bản (HTML), trình duyệt web đầu tiên (cũng là trình soạn thảo HTML và có thể truy cập các nhóm tin Usenet và các tệp FTP), phần mềm máy chủ HTTP đầu tiên (sau này được gọi là CERN httpd), máy chủ web đầu tiên và các trang web đầu tiên mô tả chính dự án. |
| Năm 1991 | - World Wide Web (WWW) dựa trên ý tưởng về siêu văn bản của Ted Nelson đưa ra vào năm 1985 được Tim Berners Lee (thuộc Trung tâm nghiên cứu nguyên tử châu u) công bố. Phát minh này đã mở ra một kỷ nguyên về siêu văn bản, một cuộc cách mạng trên Internet. Từ đây, con người có thể truy cập và trao đổi thông tin dễ dàng, nhanh chóng hơn.  - Commercial Internet eXchange được thành lập, cho phép PSInet giao tiếp với các mạng thương mại khác CERFnet và Alternet.  - NSFnet backbone được nâng cấp đạt tốc độ 44736 Mbps. NSFnet truyền 1 tỉ tỉ byte/tháng và 10 tỷ gói tin/tháng. |
| Năm 1993 | Người ta ước tính Internet chỉ mang theo 1% thông tin truyền qua viễn thông hai chiều. |
| Năm 1994 | - NIST đề nghị thống nhất trong việc dùng giao thức TCP/IP. WWW được sử dụng nhiều thứ hai, chỉ sau FTP. Sau đó, WWW tiếp tục trở thành dịch vụ lưu thông lớn nhất với gói tin truyền và byte truyền nhiều hơn. Các hệ thống quay số trực tuyến truyền thống như CompuServe, AmericanOnline, Prodigy bắt đầu kết nối Internet.  - Liên minh tín dụng liên bang Stanford là tổ chức tài chính đầu tiên cung cấp dịch vụ ngân hàng trực tuyến trên Internet cho tất cả các thành viên của mình.  - Tập đoàn truyền thông Netscape cho ra đời phiên bản beta của trình duyệt Navigator 1.0. Phiên bản này còn cồng kềnh và chạy rất chậm. |
| Năm 1995 | - NSFNET chuyển sang chỉ chuyên về nghiên cứu.  - Internet đã được thương mại hóa hoàn toàn ở Mỹ. Sự liên kết của các mạng thương mại và doanh nghiệp đánh dấu sự khởi đầu của quá trình chuyển đổi sang Internet hiện đại và sự tăng trưởng diễn ra theo cấp số nhân. Internet tiếp tục phát triển và trở thành mạng lớn nhất thế giới, được ưa chuộng trong mọi lĩnh vực của đời sống xã hội, tạo ra kỷ nguyên thương mại điện tử trên Internet. |
| Năm 1996 | - Ngân hàng hợp tác OP Financial Group trở thành ngân hàng trực tuyến thứ hai trên thế giới và đầu tiên ở châu Âu.  - Công ty Hotmail cho ra đời dịch vụ web mail. Sau 18 tháng, dịch vụ đã có 12 triệu người sử dụng. Vì thế, Microsoft đã mua lại web mail với giá 400 triệu USD.  - Triển lãm thế giới đầu tiên trên Internet Internet World Exposition ra đời. |
| Năm 1997 | - Netscape và Microsoft cho ra đời trình duyệt phiên bản 4.0 của riêng mình.  - Một tiểu ban đã tiến hành thương lượng hợp nhất các chuẩn và ban hành chuẩn chính thức IEEE 802.11 |
| Năm 1999 | - Chuẩn không dây chung 802.11b được ban hành.  - Nokia, 3Com, Intersil, Symbol, Aironet và Lucent kết hợp với nhau ra đời liên minh tương thích ethernet không dây VECA. Sau đó một thời gian, thuật ngữ wifi chính thức ra đời và trở thành tên gọi thống nhất của công nghệ kết nối cục bộ không dây được chuẩn hóa. |
| Năm 2000 | - Chuẩn không dây chung 802.11a được ban hành.  - Người ta ước tính Internet mang theo 51% thông tin truyền qua viễn thông hai chiều. |
| Tháng 11/2006 | Internet được đưa vào danh sách Bảy kỳ quan mới của USA Today. |
| Năm 2007 | Người ta ước tính hơn 97% thông tin được điều khiển qua Internet. |
| Ngày 31/3/2011 | Tổng số người dùng Internet ước tính là 2,095 tỷ (30,2% dân số thế giới). |

## Internet hoạt động như thế nào?

Dù thiết kế có sự thay đổi linh hoạt tùy theo từng thời điểm nhưng nhìn chung mạng Internet gồm ba thành phần chính. Đó là các máy tính, các thiết bị mạng đóng vai trò kết nối các máy tính với nhau và phần mềm hỗ trợ kết nối giữa các máy.

Mạng Internet hoạt động theo cách thức sau: Khi bạn truy cập vào một trang web, thiết bị của bạn sẽ gửi yêu cầu qua các dây cáp (cáp quang, dây đồng điện thoại, cáp TV) tới máy chủ. Máy chủ là thiết bị lưu trữ các trang web và hoạt động giống như ổ cứng máy tính. Khi yêu cầu đến, máy chủ sẽ truyền thông tin để thiết lập ISP và chịu trách nhiệm định tuyến yêu cầu này, tìm một miền tương tự như yêu cầu tìm kiếm gọi là DNS. Sau đó, máy chủ truy xuất trang web, gửi dữ liệu được yêu cầu đến thiết bị của bạn một cách chính xác trong vài micro giây.

Lưu ý: Các loại kết nối không dây như wifi, 3G/4G/5G tuy không kết nối qua dây nhưng cũng dựa trên những loại cáp vật lý để truy cập Internet.



Mạng Internet hoạt động theo chu trình khép kín, kết quả trả về chỉ trong vòng vài micro giây.

* + 1. Kĩ thuật internet

Kỹ thuật Internet bao gồm một loạt các công nghệ và giao thức được sử dụng để kết nối và truyền dẫn dữ liệu giữa các thiết bị và mạng trên toàn cầu. Các công nghệ chính bao gồm:

1. **TCP/IP (Transmission Control Protocol/Internet Protocol)**

Là một bộ giao thức chuẩn được sử dụng để quản lý việc truyền dẫn dữ liệu giữa các thiết bị trên mạng Internet. TCP quản lý việc chia nhỏ dữ liệu thành các gói và đảm bảo chúng được truyền dẫn một cách tin cậy, trong khi IP quản lý việc định tuyến các gói dữ liệu qua mạng.

Bộ giao thức IP dùng [sự đóng gói dữ liệu](https://vi.wikipedia.org/w/index.php?title=S%E1%BB%B1_%C4%91%C3%B3ng_g%C3%B3i_d%E1%BB%AF_li%E1%BB%87u_(m%E1%BA%A1ng_truy%E1%BB%81n_th%C3%B4ng)&action=edit&redlink=1" \o "Sự đóng gói dữ liệu (mạng truyền thông) (trang không tồn tại)) hòng trừu tượng hóa (*thu nhỏ lại quan niệm cho dễ hiểu*) các giao thức và các dịch vụ. Nói một cách chung chung, giao thức ở tầng cao hơn dùng giao thức ở tầng thấp hơn để đạt được mục đích của mình. [Chồng giao thức](https://vi.wikipedia.org/wiki/Ch%E1%BB%93ng_giao_th%E1%BB%A9c" \o "Chồng giao thức) Internet gần giống như các tầng cấp trong [mô hình của Bộ quốc phòng Mỹ](https://vi.wikipedia.org/w/index.php?title=M%C3%B4_h%C3%ACnh_c%E1%BB%A7a_B%E1%BB%99_qu%E1%BB%91c_ph%C3%B2ng_M%E1%BB%B9&action=edit&redlink=1" \o "Mô hình của Bộ quốc phòng Mỹ (trang không tồn tại)):

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 4 | Tầng ứng dụng [DNS](https://vi.wikipedia.org/wiki/DNS), [TFTP](https://vi.wikipedia.org/wiki/Trivial_File_Transfer_Protocol), [TLS/SSL](https://vi.wikipedia.org/wiki/Transport_Layer_Security), [FTP](https://vi.wikipedia.org/wiki/FTP), [HTTP](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol), [IMAP](https://vi.wikipedia.org/wiki/Internet_Message_Access_Protocol), [IRC](https://vi.wikipedia.org/wiki/IRC), [NNTP](https://vi.wikipedia.org/wiki/Network_News_Transfer_Protocol), [POP3](https://vi.wikipedia.org/wiki/Post_Office_Protocol), [SIP](https://vi.wikipedia.org/wiki/Session_Initiation_Protocol), [SMTP](https://vi.wikipedia.org/wiki/SMTP), [SNMP](https://vi.wikipedia.org/wiki/SNMP), [SSH](https://vi.wikipedia.org/wiki/SSH), [TELNET](https://vi.wikipedia.org/wiki/Telnet), [ECHO](https://vi.wikipedia.org/w/index.php?title=ECHO_protocol&action=edit&redlink=1), [BitTorrent](https://vi.wikipedia.org/wiki/BitTorrent), [RTP](https://vi.wikipedia.org/wiki/Real-time_Transport_Protocol), [PNRP](https://vi.wikipedia.org/wiki/Peer_Name_Resolution_Protocol), [rlogin](https://vi.wikipedia.org/w/index.php?title=Rlogin&action=edit&redlink=1), [ENRP](https://vi.wikipedia.org/w/index.php?title=Endpoint_Handlespace_Redundancy_Protocol&action=edit&redlink=1), … | | | | |
|  | Các giao thức định tuyến như [BGP](https://vi.wikipedia.org/wiki/BGP) và [RIP](https://vi.wikipedia.org/wiki/Routing_Information_Protocol), vì một số lý do, chạy trên TCP và UDP - theo thứ tự từng cặp: BGP dùng TCP, RIP dùng UDP - còn có thể được coi là một phần của [tầng ứng dụng](https://vi.wikipedia.org/wiki/T%E1%BA%A7ng_%E1%BB%A9ng_d%E1%BB%A5ng" \o "Tầng ứng dụng) hoặc [tầng mạng](https://vi.wikipedia.org/wiki/T%E1%BA%A7ng_m%E1%BA%A1ng" \o "Tầng mạng). | |
| 3 | Tầng giao vận [TCP](https://vi.wikipedia.org/wiki/TCP), [UDP](https://vi.wikipedia.org/wiki/UDP), [DCCP](https://vi.wikipedia.org/w/index.php?title=Datagram_Congestion_Control_Protocol&action=edit&redlink=1), [SCTP](https://vi.wikipedia.org/wiki/Stream_Control_Transmission_Protocol), [IL](https://vi.wikipedia.org/w/index.php?title=IL_Protocol&action=edit&redlink=1), [RUDP](https://vi.wikipedia.org/w/index.php?title=Reliable_User_Datagram_Protocol&action=edit&redlink=1), … | | | |
|  | Các giao thức định tuyến như [OSPF](https://vi.wikipedia.org/wiki/OSPF) (*tuyến ngắn nhất được chọn đầu tiên*), chạy trên IP, cũng có thể được coi là một phần của [tầng giao vận](https://vi.wikipedia.org/wiki/T%E1%BA%A7ng_giao_v%E1%BA%ADn" \o "Tầng giao vận), hoặc [tầng mạng](https://vi.wikipedia.org/wiki/T%E1%BA%A7ng_m%E1%BA%A1ng" \o "Tầng mạng). [ICMP](https://vi.wikipedia.org/wiki/ICMP) (*Internet control message protocol| - tạm dịch là Giao thức điều khiển thông điệp Internet*) và [IGMP](https://vi.wikipedia.org/wiki/IGMP) (*Internet group management protocol - tạm dịch là Giao thức quản lý nhóm Internet*) chạy trên IP, có thể được coi là một phần của [tầng mạng](https://vi.wikipedia.org/wiki/T%E1%BA%A7ng_m%E1%BA%A1ng" \o "Tầng mạng). |
| 2 | Tầng mạng [IP](https://vi.wikipedia.org/wiki/Internet_Protocol" \o "Internet Protocol) ([IPv4](https://vi.wikipedia.org/wiki/IPv4), [IPv6](https://vi.wikipedia.org/wiki/IPv6)) | | | |
|  | [ARP](https://vi.wikipedia.org/wiki/ARP) (*Address Resolution Protocol| - tạm dịch là Giao thức phân giải địa chỉ*) và [RARP](https://vi.wikipedia.org/wiki/RARP) (*Reverse Address Resolution Protocol - tạm dịch là Giao thức tìm địa chỉ ngược lại*) hoạt động ở bên dưới IP nhưng ở trên [tầng liên kết](https://vi.wikipedia.org/w/index.php?title=T%E1%BA%A7ng_li%C3%AAn_k%E1%BA%BFt&action=edit&redlink=1" \o "Tầng liên kết (trang không tồn tại)) (*link layer*), vậy có thể nói là nó nằm ở khoảng trung gian giữa hai tầng. | | |
| 1 | Tầng liên kết [Ethernet](https://vi.wikipedia.org/wiki/Ethernet" \o "Ethernet), [Wi-Fi](https://vi.wikipedia.org/wiki/Wi-Fi), [Token ring](https://vi.wikipedia.org/w/index.php?title=Token_ring&action=edit&redlink=1), [PPP](https://vi.wikipedia.org/wiki/PPP_(giao_th%E1%BB%A9c)), [SLIP](https://vi.wikipedia.org/w/index.php?title=Serial_Line_Internet_Protocol&action=edit&redlink=1), [FDDI](https://vi.wikipedia.org/wiki/Fiber_distributed_data_interface), [ATM](https://vi.wikipedia.org/w/index.php?title=Asynchronous_Transfer_Mode&action=edit&redlink=1), [Frame Relay](https://vi.wikipedia.org/wiki/Frame_Relay), [SMDS](https://vi.wikipedia.org/w/index.php?title=SMDS&action=edit&redlink=1), … | | | | |

Những tầng gần trên nóc gần với người sử dụng hơn, còn những tầng gần đáy gần với thiết bị truyền thông dữ liệu. Mỗi tầng có một [giao thức để phục vụ tầng trên](https://vi.wikipedia.org/w/index.php?title=Giao_th%E1%BB%A9c_c%E1%BB%A7a_t%E1%BA%A7ng_tr%C3%AAn&action=edit&redlink=1" \o "Giao thức của tầng trên (trang không tồn tại)) nó, và một [giao thức để sử dụng dịch vụ của tầng dưới](https://vi.wikipedia.org/w/index.php?title=Giao_th%E1%BB%A9c_c%E1%BB%A7a_t%E1%BA%A7ng_d%C6%B0%E1%BB%9Bi&action=edit&redlink=1" \o "Giao thức của tầng dưới (trang không tồn tại)) nó (ngoại trừ giao thức của tầng đỉnh và tầng đáy).

1. **HTTP (Hypertext Transfer Protocol)**

**HTTP** ([tiếng Anh](https://vi.wikipedia.org/wiki/Ti%E1%BA%BFng_Anh" \o "Tiếng Anh): **HyperText Transfer Protocol** - *Giao thức truyền tải siêu văn bản*) là một giao thức [lớp ứng dụng](https://vi.wikipedia.org/wiki/T%E1%BA%A7ng_%E1%BB%A9ng_d%E1%BB%A5ng" \o "Tầng ứng dụng) nằm trong [bộ giao thức](https://vi.wikipedia.org/wiki/TCP/IP" \o "TCP/IP) dành cho hệ thống thông tin siêu phương tiện phân tán, cộng tác.[[1]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-ietf2616-1) Nó chính là nền tảng dùng để trao đổi và liên lạc dữ liệu với [World Wide Web](https://vi.wikipedia.org/wiki/World_Wide_Web), nơi mà các tập tin tài liệu [siêu văn bản](https://vi.wikipedia.org/wiki/Si%C3%AAu_v%C4%83n_b%E1%BA%A3n" \o "Siêu văn bản) có thể chứa [các siêu liên kết](https://vi.wikipedia.org/wiki/Si%C3%AAu_li%C3%AAn_k%E1%BA%BFt" \o "Siêu liên kết) dẫn đến các tài nguyên số khác mà người dùng có thể dễ dàng truy cập được bằng cách [dùng chuột](https://vi.wikipedia.org/wiki/Chu%E1%BB%99t_(m%C3%A1y_t%C3%ADnh)" \o "Chuột (máy tính)) nhấp vào hoặc dùng ngón tay chạm vào màn hình cảm hứng lúc [duyệt web](https://vi.wikipedia.org/wiki/Tr%C3%ACnh_duy%E1%BB%87t_web" \o "Trình duyệt web). Nhờ đó, HTTP cho phép người sử dụng truy cập và tải về các tài nguyên như văn bản HTML, text, video, ảnh... của các trang web và hiển thị chúng trên [trình duyệt](https://vi.wikipedia.org/wiki/Tr%C3%ACnh_duy%E1%BB%87t_web" \o "Trình duyệt web).

[Tim Berners-Lee](https://vi.wikipedia.org/wiki/Tim_Berners-Lee) là người khởi xướng cho sự hình thành và phát triển của HTTP vào năm 1989 tại [CERN](https://vi.wikipedia.org/wiki/CERN). Nội dung ý tưởng của HTTP được tóm gọn lại trong một tài liệu cơ bản, bên trong mô tả hoạt động tương tác giữa máy khách và máy chủ ở phiên bản HTTP thử nghiệm đầu tiên, có số 0.9.[[2]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-HTTP/0.9-specifications-2) Phiên bản đó một thời gian sau được phát triển, trở thành phiên bản chính thức 1.0.[[3]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-HTTP/1.0-first-unofficial-draft-3)

Quá trình phát triển các [RFC](https://vi.wikipedia.org/wiki/RFC) HTTP ban đầu là nhờ vào sự hợp tác từ phía [Lực lượng Đặc nhiệm Kỹ thuật Internet](https://vi.wikipedia.org/wiki/Internet_Engineering_Task_Force) (IETF) và [World Wide Web Consortium](https://vi.wikipedia.org/wiki/W3C) (W3C), về sau thì chuyển sang do IETF phụ trách.

HTTP/1 được hoàn thiện và xuất bản tài liệu sử dụng (phiên bản 1.0) vào năm 1996. HTTP tiếp tục được phát triển vào năm 1997 (khi này là phiên bản 1.1), và đặc tính kỹ thuật của nó được cải tiến lần lượt vào các năm 1999, 2014 và 2022.

Sau này, phiên bản bảo mật hơn của [HTTP](https://vi.wikipedia.org/wiki/HTTP) là [HTTPS](https://vi.wikipedia.org/wiki/HTTPS) được sử dụng rộng rãi trên hơn 80% trang web.[[4]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-HTTPS-usage-web-servers-4) [HTTP/2](https://vi.wikipedia.org/wiki/HTTP/2) là một giao thức hiệu quả hơn về ngữ nghĩa của HTTP "trên dây" và được đưa vào sử dụng vào năm 2015. Tính đến năm tháng 4 năm 2023, nó được sử dụng trên 39% trang web[[5]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-HTTP2-usage-web-servers-5) và được gần như toàn bộ trình duyệt web hỗ trợ (chiếm 97% người sử dụng).[[6]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-HTTP2-Can-I-Use-6) Ngoài ra, nó cũng được hỗ trợ bởi các [máy chủ web](https://vi.wikipedia.org/wiki/M%C3%A1y_ch%E1%BB%A7_web" \o "Máy chủ web) lớn qua [Bảo mật tầng truyền tải](https://vi.wikipedia.org/wiki/Transport_Layer_Security) (TLS) bằng cách sử dụng tiện ích mở rộng Application-Layer Protocol Negotiation (ALPN) [[7]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-rfc7301-7) nhưng bắt buộc phải sử dụng [TLS 1.2](https://vi.wikipedia.org/wiki/Transport_Layer_Security) hoặc mới hơn.[[8]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-8)[[9]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-9)

[HTTP/3](https://vi.wikipedia.org/w/index.php?title=HTTP/3&action=edit&redlink=1) là phiên bản kế nhiệm của HTTP/2 được triển khai vào năm 2022,[[10]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-10)[[11]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-11). Nó đã được sử dụng trên 26% trang web và được nhiều trình duyệt hỗ trợ. Phiên bản này sử dụng [QUIC](https://vi.wikipedia.org/w/index.php?title=QUIC&action=edit&redlink=1) thay vì [TCP](https://vi.wikipedia.org/wiki/TCP) cho giao thức truyền tải cơ bản. Giống như HTTP/2, nó không gây lỗi thời các phiên bản chính trước đó của giao thức. Hỗ trợ dành cho cho HTTP/3 đã được bổ sung vào [Cloudflare](https://vi.wikipedia.org/wiki/Cloudflare) và [Google Chrome](https://vi.wikipedia.org/wiki/Google_Chrome) vào tháng 9 năm 2019,[[12]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-12)[[13]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-13) và có thể được kích hoạt trong các phiên bản ổn định của [Chrome](https://vi.wikipedia.org/wiki/Google_Chrome) và [Firefox](https://vi.wikipedia.org/wiki/Firefox).[[14]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-14) HTTP/3 có độ trễ thấp hơn khi duyệt các trang web, tải trang nhanh hơn HTTP/2, và thậm chí còn nhanh hơn nhiều so với HTTP/1.1, trong một số trường hợp thì HTTP/3 cho tốc độ nhanh gấp ba lần so với HTTP/1.1.[[15]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-15)

Tổng quan về mặt kỹ thuật

[Tập tin:Internet1.svg](https://vi.wikipedia.org/w/index.php?title=%C4%90%E1%BA%B7c_bi%E1%BB%87t:T%E1%BA%A3i_l%C3%AAn&wpDestFile=Internet1.svg)Một [URL](https://vi.wikipedia.org/wiki/URL) sẽ bắt đầu bằng HTTP, rồi kế đó là hai chấm và hai dấu gạch xiên liên tiếp, rồi đến mác tên miền [WWW](https://vi.wikipedia.org/wiki/World_Wide_Web)

HTTP hoạt động dựa trên giao thức yêu cầu-phản hồi nằm trong [mô hình khách - chủ](https://vi.wikipedia.org/wiki/M%C3%B4_h%C3%ACnh_kh%C3%A1ch_-_ch%E1%BB%A7" \o "Mô hình khách - chủ). *Khách* có thể là một trình duyệt được một người truy cập thực hiện yêu cầu, còn *chủ* có thể là một máy chủ web trên đám mây hoặc lưu trữ cục bộ chạy trên một chiếc máy tính quản lý một hay nhiều trang web. Khách sẽ gửi một yêu cầu HTTP lên cho máy chủ. Còn máy chủ (nơi cung cấp các *tài nguyên* chẳng hạn như tập tin [HTML](https://vi.wikipedia.org/wiki/HTML) và những nội dung, hoặc thực hiện những chức năng khác thay mặt cho khách) sẽ *phản hồi* lại thông điệp cho khách. Thông điệp được phản hồi sẽ chứa thông tin hoàn thành yêu cầu như thế nào, và đôi khi cũng sẽ chứa thông tin được yêu cầu trong phần nội dung của nó.

Trình duyệt web là ví dụ điển hình của [*user agent*](https://vi.wikipedia.org/w/index.php?title=User_agent&action=edit&redlink=1) (UA). Các loại user agent khác có thể kể đến gồm có các công cụ tự động lập chỉ mục được sử dụng bởi các bộ máy tìm kiếm ([web crawler](https://vi.wikipedia.org/w/index.php?title=Web_crawler&action=edit&redlink=1)), duyệt tìm bằng giọng nói, ứng dụng di động hoặc những phần mềm khác có thể truy cập, tải xuống và hiển thị nội dung web.

HTTP được thiết kế nhằm cải thiện các thành phần mạng trung gian cùng với cho phép liên lạc giữa máy khách và máy chủ. Các trang web có lưu lượng truy cập cao thường tận dụng những ưu điểm từ các máy chủ bộ đệm web ([web caching](https://vi.wikipedia.org/wiki/Web_caching)) nhằm cung cấp nội dung thay mặt cho các máy chủ ngược dòng để cải thiện thời gian phản hồi. Trình duyệt web thường sẽ lưu giữ các tài nguyên web đã từng được truy cập trước đó vào bộ nhớ và tái sử dụng chúng bất cứ khi nào có thể để giảm lưu lượng mạng. Máy chủ proxy HTTP ở không gian mạng riêng tư có thể tạo điều kiện liên lạc cho khách mà không cần địa chỉ có thể định tuyến toàn cầu bằng cách chuyển tiếp thông điệp với máy chủ bên ngoài.

Để cho phép các nút HTTP trung gian chẳng hạn như proxy, web cache,... có thể thực hiện được chức năng thì một số tiêu đề HTTP (có trong phần yêu cầu/phản hồi HTTP) sẽ được quản lý theo từng bước (hop-by-hop), trong khi những tiêu đề khác thì được quản lý đầu cuối (tức là chỉ duy nhất một khách đầu nguồn và một máy chủ được chỉ định).

HTTP là một giao thức lớp ứng dụng được thiết kế hoạt động trong khuôn khổ bộ giao thức Internet. Ngoài ra, nó cũng còn được giả định cho là giao thức lớp vận chuyển cơ bản và đáng tin cậy. Trong phiên bản [HTTP/3](https://vi.wikipedia.org/w/index.php?title=HTTP/3&action=edit&redlink=1) mới nhất, [Transmission Control Protocol](https://vi.wikipedia.org/wiki/Transmission_Control_Protocol) (TCP) không còn được sử dụng nữa nhưng các phiên cũ thì vẫn còn sử dụng rất nhiều. HTTP cũng được thiết kế và điều chỉnh để sử dụng các giao thức không đáng tin cậy như [User Datagram Protocol](https://vi.wikipedia.org/wiki/User_Datagram_Protocol) (UDP), và công nghệ mà HTTP/3 gián tiếp xây dựng dựa trên nó, ví dụ như [HTTPU](https://vi.wikipedia.org/w/index.php?title=HTTPU&action=edit&redlink=1) và [Simple Service Discovery Protocol](https://vi.wikipedia.org/w/index.php?title=Simple_Service_Discovery_Protocol&action=edit&redlink=1) (SSDP).

[Tài nguyên HTTP](https://vi.wikipedia.org/wiki/T%C3%A0i_nguy%C3%AAn_web) cùng vị trí của nó được xác định dựa trên đường dẫn URL ([Uniform Resource Locator](https://vi.wikipedia.org/wiki/Uniform_Resource_Locator)), sử dụng bộ URI *http* và [*https*](https://vi.wikipedia.org/wiki/Https). Theo như định nghĩa [RFC](https://vi.wikipedia.org/wiki/RFC) [3986](https://tools.ietf.org/html/rfc3986), các URI sẽ được mã hoá ở dạng siêu liên kết trong các tập tin tư liệu [HTML](https://vi.wikipedia.org/wiki/HTML) để hình thành tập tin tư liệu siêu văn bản liên kết với nhau.

Ở phiên bản HTTP/1.0, một kết nối TCP đến cùng máy chủ sẽ được thực hiện cho mỗi yêu cầu tài nguyên.[[16]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-rfc1945-1.3-16)

Ở phiên bản HTTP/1.1, một kết nối TCP có thể được sử dụng nhiều lần để tạo ra nhiều yêu cầu tài nguyên khác (trang HTML, khung, hình ảnh, mã nguồn,...).[[17]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-rfc9112-9.3-17) Nhờ vậy, liên lạc qua HTTP/1.1 sẽ ít gặp độ trễ hơn, vốn dĩ việc thiết lập các kết nối TCP gây ra chi phí đáng kể, đặc biệt trong điều kiện lưu lượng truy cập cao.[[18]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-18)

[HTTP/2](https://vi.wikipedia.org/wiki/HTTP/2) là phần cải tiến của HTTP/1.1 trước đó nhằm giữ lại mô hình khách-chủ và phương pháp giao thức giống nhau nhưng khác nhau ở chỗ:

* để sử dụng siêu dữ liệu nhị phân nén (tiêu đề HTTP) thay vì dạng văn bản, để tiêu đề yêu cầu ít không gian hơn;
* để sử dụng một kết nối [TCP/IP](https://vi.wikipedia.org/wiki/TCP/IP) duy nhất (thường đã được mã hoá) cho mỗi tên miền máy chủ được truy cập thay vì từ 2 cho tới 8 kết nối TCP/IP;
* để sử dụng một hoặc nhiều luồng stream hai chiều cho mỗi kết nối TCP/IP trong đó các yêu cầu/phản hồi HTTP được chia nhỏ và truyền thành từng gói (packet) nhỏ để gần như giải quyết được vấn đề HOLB ([head-of-line blocking](https://vi.wikipedia.org/w/index.php?title=Head-of-line_blocking&action=edit&redlink=1)).[[note 1]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-19)
* để thêm khả năng "đẩy", cho phép ứng dụng máy chủ gửi dữ liệu đến máy khách bất cứ khi nào có dữ liệu mới (không bắt buộc máy khách phải yêu cầu dữ liệu mới định kỳ đến máy chủ bằng cách sử dụng phương pháp polling).[[19]](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol#cite_note-rfc9113-2-20)

Nhờ đó, liên lạc qua HTTP/2 sẽ giảm độ trễ đi nhiều, và nhiều lúc còn chạy nhanh hơn cả liên lạc qua HTTP/1.1.

[HTTP/3](https://vi.wikipedia.org/w/index.php?title=HTTP/3&action=edit&redlink=1) là bản tinh chỉnh của HTTP/2 nhằm sử dụng giao thức truyền tải [QUIC](https://vi.wikipedia.org/w/index.php?title=QUIC&action=edit&redlink=1) + UDP thay cho TCP. Trước đó, phiên bản kết nối TCP/IP được sử dụng, thì giờ đây chỉ có lớp IP (được xây dựng trên UDP, như TCP) cũng cải thiện một chút tốc độ liên lạc trung bình và để tránh sự cố tắc nghẽn kết nối TCP không thường xuyên (rất hiếm) có thể tạm thời chặn đứng hoặc làm chậm luồng dữ liệu của tất cả các stream của nó (dạng khác của "*head of line blocking*").

1. **DNS (Domain Name System)**

Hệ thống tên miền ( DNS ) là hệ thống đặt tên phân cấp và phân cấp cho [máy tính](https://en.wikipedia.org/wiki/Computer" \o "Máy tính) , dịch vụ và các tài nguyên khác trên Internet hoặc các mạng [Giao thức Internet](https://en.wikipedia.org/wiki/Internet_Protocol) (IP) khác. Nó liên kết nhiều thông tin khác nhau với *[tên miền](https://en.wikipedia.org/wiki/Domain_name" \o "Tên miền)* ( [chuỗi](https://en.wikipedia.org/wiki/String_(computer_science)" \o "Chuỗi (khoa học máy tính)) nhận dạng ) được gán cho từng thực thể được liên kết. Nổi bật nhất, nó dịch các tên miền dễ ghi nhớ sang [các địa chỉ IP](https://en.wikipedia.org/wiki/IP_address" \o "địa chỉ IP) số cần thiết để định vị và xác định các dịch vụ và thiết bị máy tính bằng [các giao thức mạng](https://en.wikipedia.org/wiki/Network_protocol" \o "Giao thức mạng) cơ bản . [[1]](https://en.wikipedia.org/wiki/Domain_Name_System#cite_note-1) Hệ thống tên miền là một thành phần thiết yếu trong chức năng của Internet kể từ năm 1985.

Hệ thống tên miền ủy quyền trách nhiệm gán tên miền và ánh xạ những tên đó tới tài nguyên Internet bằng cách chỉ định [máy chủ tên có thẩm quyền](https://en.wikipedia.org/wiki/Authoritative_name_server" \o "Máy chủ tên có thẩm quyền) cho mỗi tên miền. Quản trị viên mạng có thể ủy quyền [các tên miền phụ](https://en.wikipedia.org/wiki/Subdomain" \o "Tên miền phụ) trong không gian tên được phân bổ của họ cho các máy chủ tên khác. Cơ chế này cung cấp dịch vụ phân tán và [có khả năng chịu lỗi](https://en.wikipedia.org/wiki/Fault-tolerant" \o "Chịu lỗi) và được thiết kế để tránh sử dụng một cơ sở dữ liệu trung tâm lớn. Ngoài ra, DNS chỉ định chức năng kỹ thuật cốt lõi của dịch vụ [cơ sở dữ liệu .](https://en.wikipedia.org/wiki/Database_model" \o "Mô hình cơ sở dữ liệu)Nó xác định giao thức DNS, đặc tả chi tiết về cấu trúc dữ liệu và trao đổi truyền thông dữ liệu được sử dụng trong DNS, như một phần của [bộ giao thức Internet](https://en.wikipedia.org/wiki/Internet_protocol_suite" \o "Bộ giao thức Internet) .

Internet duy trì hai [không gian tên](https://en.wikipedia.org/wiki/Namespace" \o "Không gian tên) chính , phân cấp tên miền và [không gian địa chỉ](https://en.wikipedia.org/wiki/Address_space" \o "Không gian địa chỉ) IP . [[2]](https://en.wikipedia.org/wiki/Domain_Name_System#cite_note-rfc781-2) Hệ thống tên miền duy trì hệ thống phân cấp tên miền và cung cấp dịch vụ dịch thuật giữa nó và không gian địa chỉ. Máy chủ tên Internet và [giao thức truyền thông](https://en.wikipedia.org/wiki/Communication_protocol" \o "Giao thức truyền thông) triển khai Hệ thống tên miền. Máy chủ tên DNS là máy chủ lưu trữ các bản ghi DNS cho một miền; máy chủ tên DNS phản hồi bằng các câu trả lời cho các truy vấn đối với cơ sở dữ liệu của nó.

Các loại bản ghi phổ biến nhất được lưu trữ trong cơ sở dữ liệu DNS là để bắt đầu cấp quyền ( [SOA](https://en.wikipedia.org/wiki/SOA_record) ), địa chỉ IP ( [A](https://en.wikipedia.org/wiki/List_of_DNS_record_types#A) và [AAAA](https://en.wikipedia.org/wiki/AAAA_record) ), [bộ trao đổi thư](https://en.wikipedia.org/wiki/Mail_exchanger" \o "Mail exchanger)[SMTP](https://en.wikipedia.org/wiki/SMTP) (MX), máy chủ định danh (NS), con trỏ để [tra cứu DNS ngược](https://en.wikipedia.org/wiki/Reverse_DNS_lookup" \o "Reverse DNS lookup) (PTR), và [bí danh tên miền](https://en.wikipedia.org/wiki/Domain_name_alias" \o "Domain name alias) (CNAME). Mặc dù không nhằm mục đích trở thành cơ sở dữ liệu cho mục đích chung, DNS đã được mở rộng theo thời gian để lưu trữ bản ghi cho các loại dữ liệu khác nhằm mục đích tra cứu tự động, chẳng hạn như bản ghi [DNSSEC](https://en.wikipedia.org/wiki/DNSSEC) hoặc cho các truy vấn của con người như bản ghi *người chịu trách nhiệm* (RP). Là cơ sở dữ liệu có mục đích chung, DNS cũng đã được sử dụng để chống [email không mong muốn](https://en.wikipedia.org/wiki/Unsolicited_email) (thư rác) bằng cách lưu trữ [danh sách lỗ đen thời gian thực](https://en.wikipedia.org/wiki/Real-time_blackhole_list" \o "Real-time blackhole list) (RBL). Cơ sở dữ liệu DNS theo truyền thống được lưu trữ trong một tệp văn bản có cấu trúc, [tệp vùng](https://en.wikipedia.org/wiki/Zone_file" \o "Zone file) , nhưng các hệ thống cơ sở dữ liệu khác là phổ biến.

Hệ thống tên miền ban đầu sử dụng [Giao thức gói dữ liệu người dùng](https://en.wikipedia.org/wiki/User_Datagram_Protocol) (UDP) làm phương tiện truyền tải qua IP. Những lo ngại về độ tin cậy, bảo mật và quyền riêng tư đã dẫn đến việc sử dụng [Giao thức điều khiển truyền dẫn](https://en.wikipedia.org/wiki/Transmission_Control_Protocol) (TCP) cũng như nhiều sự phát triển giao thức khác.

Chức năng

DNS là nó đóng vai trò như [danh bạ điện thoại](https://en.wikipedia.org/wiki/Telephone_directory) trên Internet bằng cách dịch [tên máy chủ](https://en.wikipedia.org/wiki/Hostname) máy tính thân thiện với con người thành địa chỉ IP. Ví dụ: tên máy chủ www.example.comtrong tên miền [example.com](https://en.wikipedia.org/wiki/Example.com) chuyển thành địa chỉ *93.184.216.34* ( [IPv4](https://en.wikipedia.org/wiki/IPv4) ) và *2606:2800:220:1:248:1893:25c8:1946* ( [IPv6](https://en.wikipedia.org/wiki/IPv6) ). DNS có thể được cập nhật nhanh chóng và minh bạch, cho phép thay đổi vị trí của dịch vụ trên mạng mà không ảnh hưởng đến người dùng cuối, những người tiếp tục sử dụng cùng tên máy chủ. Người dùng tận dụng lợi thế này khi họ sử dụng Bộ định vị tài nguyên thống nhất ( [URL](https://en.wikipedia.org/wiki/URL) ) và [địa chỉ e-mail](https://en.wikipedia.org/wiki/E-mail_address" \o "Địa chỉ email) có ý nghĩa mà không cần phải biết máy tính thực sự định vị các dịch vụ như thế nào.

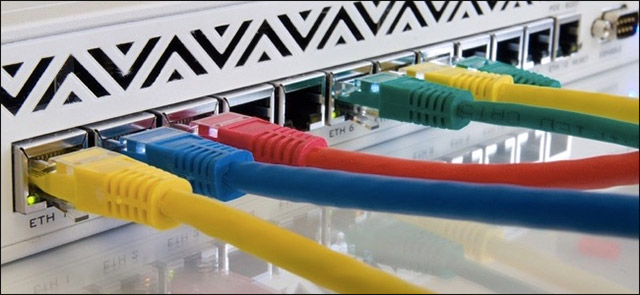
Một chức năng quan trọng và [phổ biến](https://en.wikipedia.org/wiki/Ubiquitous_computing" \o "Điện toán phổ biến) của DNS là vai trò trung tâm của nó trong các dịch vụ Internet phân tán như [dịch vụ đám mây](https://en.wikipedia.org/wiki/Cloud_service" \o "Dịch vụ điện toán đám mây) và [mạng phân phối nội dung](https://en.wikipedia.org/wiki/Content_delivery_network" \o "Mạng phân phối nội dung) . [[3]](https://en.wikipedia.org/wiki/Domain_Name_System#cite_note-3) Khi người dùng truy cập dịch vụ Internet phân tán bằng URL, tên miền của URL [sẽ](https://en.wikipedia.org/wiki/URL" \o "URL) được dịch sang địa chỉ IP của máy chủ gần nhất với người dùng. Chức năng chính của DNS được khai thác ở đây là những người dùng khác nhau có thể *đồng thời* nhận được các bản dịch khác nhau cho *cùng* một tên miền, một điểm khác biệt chính so với chế độ xem danh bạ điện thoại truyền thống của DNS. Quá trình sử dụng DNS để gán máy chủ gần nhất cho người dùng là chìa khóa để cung cấp phản hồi nhanh hơn và đáng tin cậy hơn trên Internet và được sử dụng rộng rãi bởi hầu hết các dịch vụ Internet lớn. [[4]](https://en.wikipedia.org/wiki/Domain_Name_System#cite_note-4)

DNS phản ánh cấu trúc trách nhiệm quản trị trên Internet. [[5]](https://en.wikipedia.org/wiki/Domain_Name_System#cite_note-rfc1035-5) Mỗi tên miền phụ là một [vùng](https://en.wikipedia.org/wiki/DNS_zone" \o "Vùng DNS) tự chủ hành chính được ủy quyền cho người quản lý. Đối với các vùng do cơ [quan đăng ký điều hành](https://en.wikipedia.org/wiki/Domain_name_registry" \o "Cơ quan đăng ký tên miền) , thông tin quản trị thường được bổ sung bởi các dịch vụ [RDAP](https://en.wikipedia.org/wiki/Registration_Data_Access_Protocol) và [WHOIS](https://en.wikipedia.org/wiki/WHOIS) của cơ quan đăng ký . Dữ liệu đó có thể được sử dụng để hiểu rõ hơn và theo dõi trách nhiệm đối với một máy chủ nhất định trên Internet.

1. **Ethernet và Wi-Fi**

[WiFi](https://quantrimang.com/cong-nghe/thu-thuat-wifi) rõ ràng là tiện lợi hơn rất nhiều so với [kết nối Ethernet](https://quantrimang.com/cong-nghe/mang-cuc-bo-lan-ethernet-phan-ii-305" \o "Tìm hiểu về mạng cục bộ - LAN: Ethernet - Phần II) dây dợ loằng ngoằng. Thế nhưng tại sao cho đến nay, mạng Ethernet vẫn là sự lựa chọn hàng đầu đối với các tổ chức, doanh nghiệp và thậm chí là cả những người dùng cá nhân? Đơn giản là vì mạng dây Ethernet vẫn mang trong mình những ưu điểm vượt trội so với mạng không dây WiFi.

Trong thời đại của các thiết bị di động như hiện nay, dĩ nhiên các kết nối không dây cũng đang trở thành xu hướng của thế giới công nghệ. Rõ ràng bạn sẽ chẳng cần cắm sợi dây mạng nào vào điện thoại để có thể truy cập được Internet. Nhưng các thiết bị quan trọng, hay nói cách khác là những cỗ máy để bàn, cố định, thường sử dụng Ethernet, ví dụ như những cỗ máy chơi game, hay những dàn PC phục vụ giải trí đa phương tiện, các thiết bị backup, đầu thu kỹ thuật số… Tại sao lại như vậy? Dưới đây, chúng ta sẽ cùng nhau điểm qua 3 ưu điểm chính của mạng dây Ethernet so với WiFi, đó là tốc độ nhanh hơn, độ trễ thấp hơn và kết nối đáng tin cậy hơn.



## WiFi và Ethernet

WiFi là một công nghệ mạng cho phép các thiết bị di động kết nối Internet không dây hoặc để tạo điều kiện giao tiếp không dây với nhau. Đó là công nghệ sử dụng sóng vô tuyến để cung cấp kết nối Internet tốc độ cao cho các thiết bị dựa trên tiêu chuẩn IEEE 802.11.

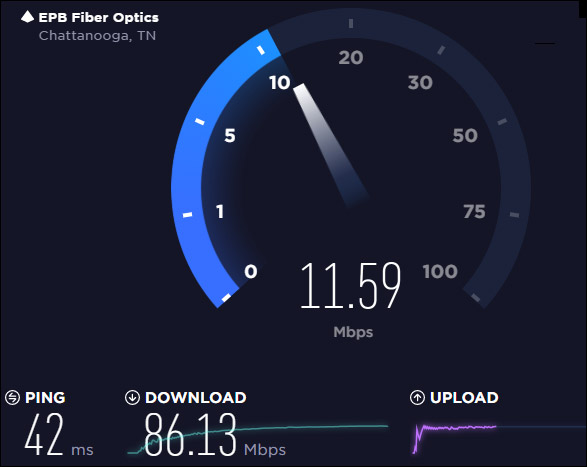
Mặt khác, Ethernet là một giao thức giao tiếp tiêu chuẩn và là công nghệ được sử dụng phổ biến nhất trong các [mạng cục bộ có dây (LAN)](https://quantrimang.com/cong-nghe/mang-cuc-bo-lan-phan-i-304" \o "Mạng cục bộ LAN là mạng gì?). Ethernet liên quan đến cáp vật lý hoặc cáp Internet mà dữ liệu truyền qua đó.

**Tốc độ của Ethernet nhanh như thế nào?**

Có một điều mà các “fan” của WiFi phải chấp nhận đó là Ethernet bao giờ cũng nhanh hơn WiFi, và hiện tại thì không có cách nào để có thể giúp WiFi bắt kịp với tốc độ Ethernet. Tuy nhiên trên thực tế, khoảng cách này không quá lớn như bạn nghĩ. Nhờ các công nghệ như [802.11ac](https://quantrimang.com/cong-nghe/tim-hieu-ve-wifi-chuan-80211ac-118642) và [802.11n](https://quantrimang.com/cong-nghe/cac-chuan-wireless-802-11b-802-11a-802-11g-va-802-11n-47723), với tốc độ tối đa lần lượt đạt 866.7Mb/s và 150Mb/s, WiFi ngày càng bám đuổi Ethernet quyết liệt hơn trong cuộc đua tốc độ. Tuy rằng đó là những con số trên lý thuyết còn về mặt sử dụng thực tế thì chắc chắn không thể được như vậy, thế nhưng WiFi vẫn đủ tốt để đáp ứng mọi nhu cầu thường ngày của bạn.

Mặt khác, một kết nối Ethernet có dây về mặt lý thuyết có thể đạt tốc độ tối đa 10Gb/s nếu dùng loại cáp Cat6. Phải nói thêm rằng tốc độ tối đa của mạng Ethernet sẽ phụ thuộc vào[loại cáp](https://quantrimang.com/cong-nghe/phan-biet-cap-utp-s-utp-ftp-stp-va-cap-sftp-143224" \o "Phân biệt các loại cáp mạng UTP, S/UTP, FTP, STP, SFTP...) mà bạn dùng. Tuy nhiên, ngay cả cáp Cat5e là loại cáp được sử dụng phổ biến hiện nay cũng hỗ trợ tốc độ tối đa tới 1Gb/s. Và quan trọng là tốc độ này vô cùng ổn định chứ không chập chờn giống như WiFi.

Tuy nhiên, Ethernet cũng sẽ ảnh hưởng đến tốc độ giữa các thiết bị trong một mạng. Ví dụ, nếu bạn muốn chuyển các file giữa hai máy tính trong nhà với tốc độ nhanh nhất có thể, trong trường hợp này, dùng mạng dây sẽ nhanh hơn Wifi. Lúc đó, kết nối Internet không tham gia vào quá trình chuyển tải, nên bạn sẽ đạt được tốc độ tối đa mà phần cứng mạng của bạn có thể cung cấp.



Dưới đây là một vài ví dụ điển hình để hiểu hơn về tình huống trên:

* Nếu bạn có nhiều thiết bị để sao lưu dữ liệu vào NAS, ví dụ như máy chủ dự phòng hoặc [ổ đĩa cứng](https://quantrimang.com/cong-nghe/o-cung-la-gi-hien-co-may-loai-140736) chia sẻ chung, thì quá trình sao lưu sẽ diễn ra nhanh hơn nếu dùng Ethernet.
* Nếu bạn đang stream từ một máy chủ media (như Plex hay [Kodi](https://quantrimang.com/cong-nghe/huong-dan-cai-dat-va-su-dung-kodi-tren-may-tinh-140579)) thì Ethernet sẽ cho tốc độ nhanh và chất lượng kết nối ổn định hơn nhiều.
* Nếu bạn tò mò về sự khác biệt của tốc độ truyền file cục bộ, hãy thử chuyển một file có kích thước lớn giữa hai máy tính trong khi cả hai máy tính đều được kết nối với Ethernet, và khi chúng được kết nối WiFi. Bạn sẽ thấy một sự chênh lệch về tốc độ.

## **Độ trễ của Ethernet thấp như thế nào?**

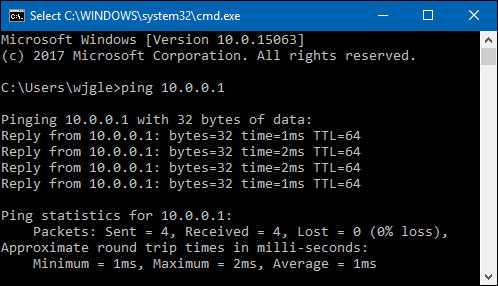
Ngoài những vấn đề về băng thông, tốc độ kết nối còn phụ thuộc vào một yếu tố quan trọng khác là "độ trễ". Trong trường hợp này, độ trễ là khoảng thời gian delay khi dữ liệu truyền từ thiết bị đến đích, game thủ thường gọi nó là [ping](https://quantrimang.com/cong-nghe/lenh-ping-tren-may-tinh-la-gi-130545).

Ví dụ: nếu bạn đang chơi một game trực tuyến và cần thời gian phản hồi càng nhanh càng tốt, bạn sẽ phải tìm cách giảm tối đa độ trễ bằng cách sử dụng kết nối có dây Ethernet để cho độ trễ tốt hơn. Tất nhiên là vẫn sẽ có độ trễ bị gây ra bởi các vấn đề trên đường dẫn Internet giữa thiết bị của bạn và máy chủ trò chơi, nhưng không quá đáng kể.

Còn nếu bạn chỉ xem video hay nghe nhạc, lướt web… thì độ trễ chẳng phải là vấn đề gì to tát.

Bạn có thể kiểm tra độ trễ bằng lệnh "Ping" trong [Command Prompt](https://quantrimang.com/cong-nghe/huong-dan-su-dung-command-prompt-85301) của [Windows](https://quantrimang.com/windows)hay [Terminal trên Linux](https://quantrimang.com/cong-nghe/4-cach-truy-cap-nhanh-toi-mot-terminal-trong-linux-69438) và Mac. Cả khi thiết bị được kết nối qua

WiFi và khi được kết nối qua Ethernet. So sánh kết quả và bạn sẽ thấy rằng nếu dùng Wifi, độ trễ sẽ cao hơn một chút bởi tín hiệu phải đi qua đi lại giữa thiết bị và [router](https://quantrimang.com/cong-nghe/kien-thuc-co-ban-ve-mang-phan-2-router-35172).



Tóm lại, kết nối WiFi sẽ có độ trẽ lớn hơn bởi tín hiệu sẽ phải di chuyển qua lại giữa thiết bị WiFi và router không dây của bạn. Ngược lại, với kết nối Ethernet có dây, độ trễ sẽ thấp hơn nhiều.

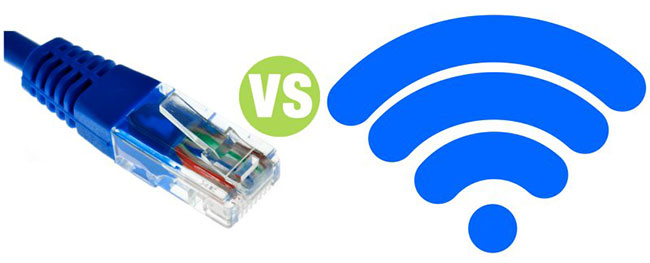
**Mức độ đáng tin cậy của kết nối Ethernet**

Ethernet sẽ cho chúng ta một kết nối đáng tin cậy hơn so với WiFi, đó là điều không phải bàn cãi.

Đơn giản là bởi kết nối không dây sẽ phải chịu nhiều ảnh hưởng hơn so với kết nối có dây. Đơn cử như kết cấu nhà, vật cản sóng, tín hiệu điện từ, hay thậm chí là mạng WiFi của nhà hàng xóm cũng [có thể là yếu tố gây ảnh hưởng đến WiFi](https://quantrimang.com/cong-nghe/6-dieu-anh-huong-toi-toc-do-mang-wifi-79051" \o "7 điều ảnh hưởng tới tốc độ mạng WiFi). Tất cả những yếu tố trên đều góp phần cho WiFi thường bất ổn định và kém tin cậy hơn. Có thể kể đến một số vấn đề như:

* Mất tín hiệu: thỉnh thoảng bạn sẽ bị "rớt mạng" mà phải tiến hành kết nối lại. Đây có thể không phải là một vấn đề lớn đối với việc lướt web hằng ngày hoặc thậm chí phát trực tuyến video (được lưu vào bộ nhớ đệm trên thiết bị cục bộ), bởi vì việc kết nối lại sẽ diễn ra nhanh chóng. Nhưng thử tưởng tượng bạn đang chơi game mà mạng cứ rớt liên tục, chắc chắn sẽ không hề dễ chịu chút nào nếu không muốn nói là phiền toái.
* Độ trễ cao hơn: Lý do tại sao đã được chúng ta đề cập ở phía trên.
* Tốc độ chậm hơn: Càng nhiều yếu tố tác động sẽ dẫn đến chất lượng tín hiệu WiFi càng kém và cuối cùng là tốc độ cũng sẽ giảm theo.

Ethernet bảo mật tốt hơn WiFi

Ethernet bảo mật tốt hơn WiFi

Dữ liệu được gửi qua kết nối Ethernet chỉ có thể được truy cập bởi các thiết bị kết nối vật lý với mạng đó và do đó không có nguy cơ mất dữ liệu hoặc bị tấn công. Các thiết bị này cần sử dụng [tường lửa](https://quantrimang.com/cong-nghe/tong-quan-ve-firewall-84474" \o "Tường lửa (Firewall) là gì? Những kiến thức tổng quan về Firewall) để bảo mật.

Mặt khác, WiFi là một mạng mở nên dữ liệu của nó không an toàn. Khi truyền dữ liệu nhạy cảm, hãy đảm bảo sử dụng mạng WiFi nơi dữ liệu được mã hóa và bảo mật. Phương pháp mã hóa an toàn nhất là WPA2-PSK, còn WEP là kém bảo mật nhất.

* [Nâng cấp bảo mật Wi-Fi từ WEP lên WPA2](https://quantrimang.com/cong-nghe/nang-cap-bao-mat-wi-fi-tu-wep-len-wpa2-76357)

[WiFi công cộng miễn phí không an toàn](https://quantrimang.com/lang-cong-nghe/4-nguy-co-khong-ngo-toi-cua-viec-su-dung-wifi-cong-cong-157775). vì vậy tốt nhất là bạn nên tránh sử dụng. Tìm hiểu thêm về các mối đe dọa đến từ WiFi công cộng qua bài viết: [Dữ liệu của bạn có thể bị đánh cắp khi sử dụng Wifi công cộng?](https://quantrimang.com/cong-nghe/du-lieu-cua-ban-co-the-bi-danh-cap-khi-su-dung-wifi-cong-cong-126650" \o "Dữ liệu của bạn có thể bị đánh cắp khi sử dụng Wifi công cộng?)

* [Những điều cần biết khi dùng WiFi nơi công cộng](https://quantrimang.com/cong-nghe/nhung-dieu-can-biet-khi-dung-wifi-mien-phi-133498)

Ethernet ít bị nhiễu hơn

Mọi người có nhiều thiết bị trong nhà hoặc văn phòng có thể gây nhiễu WiFi và gây ra nhiều sự cố khác nhau.

Những vấn đề này bao gồm:

* Mất tín hiệu
* Độ trễ lớn hơn
* Tốc độ thấp hơn

Do đó, Ethernet đáng tin cậy hơn khi xét về khả năng bị nhiễu.

**Vậy khi nào nên sử dụng Ethernet?**

****

Bài viết hoàn toàn không có ý hạ thấp WiFi, bởi dù sao thì đây cũng là một kết nối khá nhanh, vô cùng tiện lợi, và đáp ứng đủ mọi nhu cầu hằng ngày của phần lớn người dùng phổ thông, đặc biệt là khi bạn sử dụng smartphone hay thường xuyên làm việc bên ngoài văn phòng với laptop. Ngoài ra, cũng có những lúc bạn không thể sử dụng Ethernet. Có thể là do quá khó để chạy cáp đến vị trí bạn muốn. Hoặc có thể chủ nhà của bạn sẽ không cho phép bạn chạy cáp theo cách bạn muốn.

Và đó là lúc tính tiện lợi của WiFi lên tiếng. Nếu bạn thường xuyên di chuyển và chắc chắn là không muốn phải mang theo dây dợ cồng kềnh thì Wifi là lựa chọn hiển nhiên.

Tuy nhiên, nếu bạn dùng máy tính bàn hay máy chủ và chỉ đặt máy ở một nơi duy nhất, thì tốt nhất bạn nên cố gắng sử dụng Ethernet bởi nó cho chất lượng truyền tốt hơn, đặc biệt nếu bạn là game thủ, bạn sẽ thấy được Ethernet đáng giá như thế nào.

Tổng kết

Tóm lại, Ethernet mang đến cho chúng ta tốc độ kết nối tốt hơn, độ trễ thấp, kết nối ổn định và đáng tin cậy hơn. Trong khi đó, WiFi là hiện thân của sự tiện dụng và đáp ứng đủ những nhu cầu cơ bản của đại đa số người dùng. Vậy nên, hãy cân nhắc kĩ về vấn đề sử dụng mạng không dây hay có dây; dù sao đi nữa thì một sợi cáp đơn giản cũng đã mang lại sự khác biệt về tốc độ, nhưng sự khác biệt này có đủ lớn để bạn phải đánh đổi lấy sự tiện dụng hay không mà thôi.

1. **4G/5G**

**Mạng 4G**

Ra mắt cách đây hơn 10 năm, mạng 4G thực sự đã tạo ra những thay đổi rõ rệt trong trải nghiệm sử dụng internet của người dùng trên toàn thế giới.

**Định nghĩa**

4G hay tên đầy đủ Fourth-Generation là mạng không dây thế hệ thứ tư. 4G là đoạn truyền thông di động băng thông rộng thay thế 3G (không dây thế hệ thứ ba) và là tiền thân của 5G (không dây thế hệ thứ năm).

Tiêu chuẩn di động không dây 4G do Liên minh Viễn thông Quốc tế (ITU) xác định và chỉ định các đặc điểm chính của tiêu chuẩn, bao gồm công nghệ truyền dẫn và tốc độ dữ liệu.

**Ứng dụng**

Với tốc độ tải xuống lên tới 1GB của 4G, người dùng có thể truyền phát video và âm thanh độ nét cao. 4G cũng cho phép băng thông rộng không dây, cung cấp cho người dùng cách kết nối internet mà không cần kết nối có dây, cố định từ nhà cung cấp dịch vụ internet (ISP).

**Tốc độ**

Thực nghiệm cho thấy 4G có thể đạt tốc độ lên tới 100 Mbps, trong khi với 3G chỉ tối đa 14 Mbps.



So sánh mạng di động 4G và 5G để thấy được sự khác biệt

Mạng 5G

5G được ví như cuộc cách mạng viễn thông của cả thế giới. 5G lần đầu xuất hiện vào khoảng những năm 2020, tức là chỉ một vài năm trở lại đây, hứa hẹn mang đến rất nhiều ứng dụng đột phá trong trương lai.

**Định nghĩa**

5G là mạng di động thế hệ thứ 5. Đây là tiêu chuẩn không dây toàn cầu mới sau mạng 1G, 2G, 3G và 4G. 5G được thiết kế cho phép vạn vật kết nối với nhau (IoT), giữa người với người và người với thiết bị. Công nghệ không dây 5G nhằm cung cấp tốc độ dữ liệu cao nhất nhiều Gbps, độ trễ cực thấp, độ tin cậy cao hơn, dung lượng mạng lớn, tính khả dụng cao hơn và trải nghiệm thống nhất hơn cho nhiều người dùng hơn.

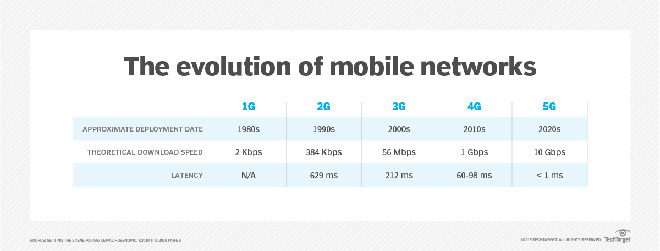
**Ứng dụng**

Nhờ hiệu suất cao hơn và hiệu quả được cải thiện, 5G mang lại trải nghiệm mới cho người dùng và kết nối các ngành công nghiệp mới, thúc đẩy sự phát triển của internet vạn vật. Những ứng dụng thực tế tăng cường (AR) và thực tế ảo (VR), y học, ô tô tự lái… đều được hỗ trợ mạnh mẽ bởi 5G.

Thậm chí, Hiệp hội vận tải hàng không quốc tế IATA còn cho biết người dùng có thể an toàn sử dụng 5G trên máy bay do mạng này sử dụng tần số khác với máy bay. Từ tháng 06/2023, liên minh châu Âu sẽ cho phép gọi điện và truy cập 5G trên máy bay. Đây có lẽ là bước khởi đầu cho xu hướng loại bỏ chế độ máy bay trên điện thoại.

**Tốc độ**

Tốc độ của 5G được đánh giá nhanh hơn 4G tới 10 lần (khoảng 10GB/s). Chúng ta có thể xem thêm bảng dưới đây để thấy 5G đã thực sự phát triển như thế nào so với những mạng di động thế hệ trước như thế nào:



5G đã được phát triển nhanh hơn 10 lần so với 4G và hơn 5 triệu lần so với 1G.

Có thể bạn quan tâm:[Điện thoại 5G có dùng được 4G không? Có nên mua điện thoại 5G ở thời điểm hiện tại?](https://vnpt.com.vn/tu-van/dien-thoai-5g-co-dung-duoc-4g-khong.html)

**Điểm khác biệt giữa 4G và 5G**

Qua bảng ở trên chúng ta đã thấy được tốc độ vượt trội của 5G so với mạng di động tiền nhiệm. Tuy nhiên như vậy chưa đủ để nó trở thành cuộc cách mạng công nghệ viễn thông. Để xứng đáng với danh xưng này, chúng ta cùng đi xem xét 5G ở góc nhìn số liệu toàn diện hơn bằng cách so sánh giữa 4G và 5G khác gì nhau.

**Lưu ý:** đây là số liệu trong môi trường lý tưởng.

|  |  |  |
| --- | --- | --- |
| **Tiêu chí** | **5G** | **4G** |
| Băng tần | Dao động từ 30 GHz tới 300 GHz | Dao động từ 700 MHz tới 2.600 MHz |
| Nguyên lý phát sóng | Rộng do sử dụng trạm HAPS trên không | Giới hạn, do trạm kết nối trên mặt đất |
| Tốc độ mạng | Khoảng 10 GBps | Khoảng 1 GBps |
| Xử lý hoạt động phức tạp (game, phim…) | Không giật lag ngay cả khi dùng di động. Chỉ mất 10s để tải tập phim dài 2h. | Giới hạn, mất tới 7 phút để tải một tập phim dài 2h. |
| Độ trễ khi kết nối | 4ms tới 1ms | khoảng 75ms |
| Kết nối nhiều thiết bị | Gấp 10 đến 100 lần thiết bị kết nối cùng lúc | Kết nối giữa các thiết bị cá nhân ở phạm vi giới hạn |
| Khu vực phủ sóng tại Việt Nam (cập nhật) | Hà Nội, TP HCM và 1 số tình thành lớn. Dự kiến hoàn thành phủ sóng toàn dân trước 2030. | Toàn quốc |
| Tiêu thụ năng lượng | Giảm 90% năng lượng tiêu thụ | Bình thường |

Qua bảng so sánh trên chúng ta có thể thấy các dữ liệu 5G và 4G đều có sự cách biệt lớn cả về số lượng, chất lượng. Ngoài tốc độ mạng nhanh hơn 10 lần, độ trễ của 5G cũng thấp hơn 4G rất nhiều lần trong khi đó phạm vi cho phép kết nối được mở rộng từ 10 đến 100 lần.



5G rất ưu việt so với 4G tuy nhiên khu vực phủ sóng còn hạn chế.

Như vậy, 5G rất ưu việt, tuy nhiên vẫn còn 1 số nhược điểm nhỏ:

* - Độ phủ sóng chưa rộng: Hiện nay 5G mới được triển khai tại Hà Nội, Tp Hồ Chí Minh cùng một số tỉnh thành lớn tại Việt Nam. Điều đó có nghĩa một số địa phương sẽ còn phải chờ một thời gian cho đến khi 5G hoàn thành phủ sóng toàn quốc để được sử dụng.
* - Một số thiết bị phải đáp ứng yêu cầu mới sử dụng được: Để sử dụng 5G, điện thoại của bạn cũng phải đáp ứng yêu cầu về cấu hình, hãng hỗ trợ 5G hay không. Những điện thoại đời thấp dù tài khoản SIM có đăng ký 5G cũng sẽ tự động chuyển sang những mạng thấp hơn như 4G, 3G để phù hợp với cấu hình máy.

Tìm hiểu thêm: [Điều kiện & Cách đăng ký 5G VinaPhone chi tiết nhất](https://vnpt.com.vn/tu-van/cach-dang-ky-5g-vinaphone.html" \t "_blank)

**5G đang ứng dụng tại đâu ở Việt Nam?**

Hiện nay các nhà mạng lớn tại trong nước đang triển khai 5G tại các thành phố lớn như Hà Nội, TP.Hồ Chí Minh và các tỉnh thành lân cận. Việt Nam đặt mục tiêu hoàn thành phủ sóng 5G toàn quốc vào năm 2030.

|  |  |
| --- | --- |
| **Khu vực** | **Tỉnh/Thành phố** |
| Miền Bắc | Hà Nội, Bắc Ninh, Bắc Giang… |
| Miền Trung | Thừa Thiên Huế… |

Miền NamTP. Hồ Chí Minh, Bình Dương, Bình Phước, Bà Rịa - Vũng Tàu, Cần Thơ…



Hình ảnh kỹ sư và ban lãnh đạo VinaPhone thử nghiệm tốc độ mạng vào năm 2020.

**5G VinaPhone - Dẫn đầu cuộc cách mạng 4.0**

Ngay từ khi 5G có trên bản đồ thế giới, VinaPhone là một trong những đơn vị tiên phong tại Việt Nam đưa mạng này vào triển khai thí điểm. Rất nhanh chóng sau đó, vào 2020, nhà mạng tuyên bố thử nghiệm thành công, tốc độ mạng 5G đạt mức 2.2Gbps, nhanh gấp 10 lần so với mạng 4G với độ trễ tiệm cận 0.

Như vậy tốc độ truy cập Internet của VinaPhone 5G đã tương đương với chuẩn 5G của thế giới. Không chỉ hỗ trợ các cuộc gọi online qua Zalo, Zoom, Messenger… thông suốt, mượt hơn mà người dùng VinaPhone 5G cũng được trải nghiệm xem phim, chơi game…với chất lượng hình ảnh rất cao.



VinaPhone đa dạng gói cước 5G đáp ứng mọi nhu cầu người dùng.

Hiện nay mạng 5G VinaPhone đã đi vào hoạt động thương mại được 2 năm, nhà mạng cũng cho ra mắt rất nhiều gói cước ưu đãi dành cho thuê bao 5G. Với kinh nghiệm người dẫn đầu, VinaPhone luôn nỗ lực sáng tạo đổi mới dựa trên nền tảng để đưa tới khách hàng chất lượng dịch vụ tốt nhất.

1.1.3 HTTP, FTP

Tất cả các công nghệ và giao thức này làm việc cùng nhau để tạo ra một hệ thống mạng toàn cầu mạnh mẽ và linh hoạt, cho phép truy cập và trao đổi thông tin trên Internet.

HTTP (Giao thức Truyền Siêu văn bản) và FTP (File Transfer Protocol) chỉ là hai trong số rất nhiều các giao thức đang được sử dụng trên Internet, mỗi chức năng riêng. Mục đích của HTTP là để phục vụ như là một phương tiện để truy cập vào trang web trên toàn thế giới. Trang web được truy cập bằng cách sử dụng http với sự trợ giúp của trình duyệt. FTP, như tên của nó, được sử dụng trong việc chuyển các tập tin từ máy tính này sang máy khác. Đây là một giao thức ít phổ biến hơn do số lượng người thực sự sử dụng FTP và thậm chí ít người biết rằng họ đang sử dụng nó.

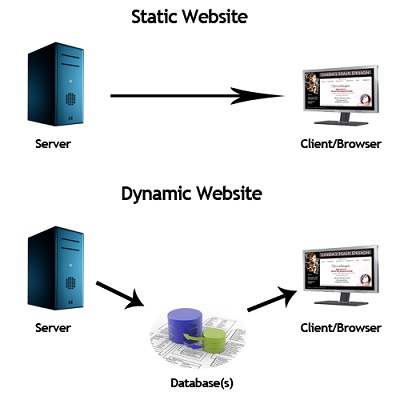
Hầu hết mọi người không thực sự biết, nhưng giao thức HTTP được sử dụng mỗi khi chúng tôi mở một trang web, kiểm tra email của chúng tôi hoặc cập nhật blog của chúng tôi. Điều này có thể dễ dàng kiểm tra bằng cách nhìn vào thanh địa chỉ và nhìn vào vài chữ cái đầu; rất có thể là HTTP của nó. Phần lớn những người sử dụng giao thức FTP là những người duy trì và thường xuyên tải các tệp lên các trang web. FTP cung cấp một phương pháp bảo trì trang web dễ dàng và không phức tạp.

FTP cũng là một lựa chọn tốt cho những người muốn tải các tập tin. Có những máy chủ FTP lưu trữ các tập tin và cho phép mọi người ẩn danh đăng nhập vào trang web của họ và tải các tập tin lớn. Tuy nhiên, tải xuống không phải là một khả năng hoàn toàn được giữ bởi FTP, nó cũng có thể được thực hiện với HTTP. Sự gia tăng số lượt tải xuống trong HTTP chủ yếu là do liên kết với web trên toàn thế giới. Hầu hết các trang web cung cấp nội dung cho tải tệp của họ được lưu trữ trong một máy chủ HTTP để khách truy cập có thể dễ dàng duyệt và chọn tệp của họ.

FTP đã cũ và hầu hết những người dùng mới bắt đầu điều chỉnh các giao thức khác như là một sự thay thế như gopher. Nó sẽ chỉ là một vấn đề thời gian trước khi FTP trở nên hoàn toàn lỗi thời. HTTP, mặt khác, vẫn còn mạnh mẽ và sẽ vẫn là giao thức Internet chiếm ưu thế trong những năm tới. Có nhiều ngôn ngữ đang được phát triển để sử dụng với và để cải tiến giao thức HTTP.

Phân biệt Web tĩnh, Web động

* **Web tĩnh** tức là website không có hệ thống quản lý nội dung, hoặc có nhưng về nội dung website thì không thay đổi được.
* **Web động** tức là website có hệ thông quản lý nội dung nên người dùng có thể thay đổi nội dung được.

****

Web tĩnh là gì?

Web tĩnh là những website không có hệ thống quản lý nội dung và người dùng không thể chỉnh sửa được hoặc có thể chỉnh sửa, thay đổi dữ liệu được. Hiện nay web tĩnh hầu như ít tồn tại hoặc chỉ tồn tại với những công ty chuyên về thiết kế website, họ sử dụng web tĩnh bởi họ có kiến thức và dễ dàng chỉnh sửa nội dung khi cần.

Web tĩnh được làm từ gì?

Về kiến thức cơ bản thì web tĩnh thường được xây dựng từ CSS, HTML, JAVASCRIPT (DHTML), hiện nay có thêm công nghệ HTML5 & CSS3. Có thể bạn nhìn một web nào đó nhìn rất lung linh, đẹp và nội dung có thể thay đổi nhưng chưa chắc nó là web động bởi vì người ta có thể sử dụng DHTML để thay đổi nội dung (tại client).

Nếu bạn chạy trên máy tính của bạn thì với web tĩnh bạn có thể đặt tại đâu cũng chạy được vì bản chất nó chỉ là một file bình thường.

Ưu điểm của web tĩnh:

* Về giao diện Designer có thẻ thiết kế theo kiểu mới lạ
* Tốc độ truy cập nhanh bởi nó chỉ là những file HTML
* Chi phí đầu tư thấp bởi bạn không phải trả tiền nhiều cho Coder
* Thân thiện với bộ máy tìm kiếm bởi bạn có thể đặt tên file tùy ý (ten-file.html, tieu-de-tin-tuc.html)

Nhược điểm của web tĩnh:

* Khó quản lý nội dung
* Khó nâng cấp bảo trì
* Mỗi khi thay đổi phải vào file HTML, CSS hoặc JAVASCRIPT để chỉnh sửa

Khi nào nên sử dụng web tĩnh?

* Nếu bạn là doanh nghiệp muốn tự mình làm website thì bạn có thể học các kiến thức căn bản và tự làm một Web tĩnh cho mình
* Nội dung website ít khi cập nhật và ít nên bạn muốn tiết kiệm chi phí
* Website bạn nhỏ và bạn thuê luôn người chuyên về web để quản trị

Trên thực tế thì mình thấy hiện nay chỉ có các công ty thiết kế web họ mới làm web tĩnh bởi vì họ có khả năng chỉnh sửa được chứ khách hàng thì họ không có khả năng đó. Hiện nay có các trang web ONE PAGE có các hiệu ứng slide khá độc nên cũng có thể họ sử dụng web tĩnh hoặc tĩnh một phần trong đó.

Web động là gì?

Ngược lại với web tĩnh, web động là những website có hệ thống quản lý nội dung và người dùng có thể chỉnh sửa nội dung được. Ví dụ như trang vnexpress.net hay 24h.com.vn thì đó là những web động bởi vì họ có thể đăng tin mới, chỉnh sửa tin mới và chỉnh sửa danh mục menu, ...

Web động được làm từ gì?

Web động sử dụng các công nghệ như HTML,  CSS, HTML5, CSS3, JAVASCRIPT, ... và điều đặc biệt là có sử dụng một ngôn ngữ lập trình server như PHP, một hệ quản trị cơ sở dữ liệu như MySQL, vì vậy web động phải chạy trong máy chủ. Nếu đặt trên máy tính của bạn thì bạn phải cài các Web server ảo như vertrigo, Xampp, wampp, ...

Ưu điểm của web động:

* Dễ dàng quản lý nội dung
* Dễ dàng nâng cấp và bảo trì
* Có thể xây dụng được web lớn
* Thường sử dụng tương tác với người dùng cao
* Hiện nay vẫn SEO tốt bởi ta có thể sử dụng chức năng Rewrite URL để chuyên URL thân thiện

Nhược điểm của web động:

* Chi phí xây dựng cao
* Nếu web lớn có thể cần thêm nhân sự chuyên ngành

Khi nào nên sử dụng web động?

* Bạn làm website thương mại điện tử bán hàng
* Bạn làm web giới thiệu sản phảm công ty
* Bạn làm web tin tức, blog cá nhân
* Web bạn tầm cỡ lớn

Nhưng chung quy lại thì hiện nay đa số bạn nên chọn giải pháp làm web động sẽ tốt hơn.

**# Kết luận**

Dựa vào những ưu điểm và nhược điểm của web động và web tĩnh bạn hãy cân nhắc khi làm web mới nhé. Mình thấy hiện nay chọn giải pháp web động luôn là ưu tiên số 1 vì nó có hệ thống quản lý nội dung giúp bạn dễ dàng thay đổi website hơn. Có những web khách hàng yêu cầu tùy biến luôn giao diện, tuy nhiên giá cả sẽ tốt hơn.

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 và tài nguyên trên máy chủ cho các trang web và ứng dụng web.

Hosting là dịch vụ cung cấp không gian lưu trữ và tài nguyên trên máy chủ cho các trang web và ứng dụng web. Dịch vụ này cho phép người dùng đăng ký một không gian trên máy chủ để lưu trữ các tệp, dữ liệu và ứng dụng của họ, giúp cho trang web hoặc ứng dụng của họ có thể truy cập được từ Internet. Các nhà cung cấp hosting thường cung cấp các dịch vụ như không gian lưu trữ, băng thông, bảo mật, và hỗ trợ kỹ thuật cho các khách hàng của họ.Top of Form

**Tên miền (Domain Name)**

Chuỗi các ký tự dùng để định danh một trang web hoặc tổ chức trên Internet.

Tên miền (Domain Name) là một chuỗi các ký tự được sử dụng để định danh một trang web hoặc một tổ chức trên Internet. Tên miền thường được sử dụng để dễ dàng nhớ và truy cập các trang web thay vì phải sử dụng các địa chỉ IP phức tạp. Ví dụ, "google.com" là một tên miền. Tên miền thường được đăng ký và quản lý thông qua các tổ chức quản lý tên miền như ICANN (Internet Corporation for Assigned Names and Numbers) hoặc các công ty cung cấp dịch vụ đăng ký tên miền

**Web Server Name**

Tên của máy chủ web, nơi lưu trữ và phục vụ các trang web và nội dung web cho người dùng.

Web Server Name là tên của máy chủ web, tức là máy tính hoặc hệ thống phần mềm chịu trách nhiệm lưu trữ và phục vụ các trang web và nội dung web cho người dùng. Mỗi máy chủ web sẽ có một tên duy nhất để xác định nơi lưu trữ và phục vụ các trang web. Ví dụ, trong địa chỉ URL "[http://www.example.com](http://www.example.com/)", phần "[www.example.com](http://www.example.com/)" là tên của máy chủ web, nơi chứa các tệp và dữ liệu của trang web.

**HomePage**

Trang chính được hiển thị khi người dùng truy cập vào một trang web, thường chứa các liên kết và thông tin tổng quan về trang web đó.

HomePage là trang chính được hiển thị khi người dùng truy cập vào một trang web. Đây thường là trang đầu tiên mà người dùng nhìn thấy và chứa các liên kết và thông tin tổng quan về trang web đó. HomePage thường được thiết kế để làm điểm khởi đầu cho việc duyệt web trên trang web cụ thể, cung cấp một bản tóm tắt hoặc hướng dẫn cơ bản về nội dung và tính năng của trang web.Top of Form

**Web Site**

Tập hợp các trang web, tài nguyên, và nội dung trên Internet có chung một chủ đề hoặc mục đích.

Web Site là một tập hợp các trang web, tài nguyên và nội dung trên Internet có chung một chủ đề hoặc mục đích. Một trang web có thể là một phần của một trang web lớn hơn hoặc tồn tại độc lập. Một Web Site có thể bao gồm các trang web, hình ảnh, video, âm nhạc, tài liệu tải xuống và các nội dung khác mà người dùng có thể truy cập và tương tác thông qua trình duyệt web của họ.

**Web Server**

Máy tính hoặc hệ thống phần mềm chịu trách nhiệm lưu trữ và phục vụ các trang web và nội dung web cho người dùng.

Web Server là một máy tính hoặc hệ thống phần mềm chịu trách nhiệm lưu trữ và phục vụ các trang web và nội dung web cho người dùng. Các máy chủ web có khả năng xử lý các yêu cầu HTTP từ trình duyệt web của người dùng và trả về nội dung web tương ứng. Mỗi trang web thường được lưu trữ trên một máy chủ web cụ thể, và thông qua việc truy cập vào máy chủ web đó, người dùng có thể truy cập các trang web và nội dung web mà họ muốn xem.

***URL (Uniform Resource Locator)***

Địa chỉ dùng để xác định một tài nguyên trên Internet, bao gồm tên miền và các thông tin địa chỉ cụ thể.

URL (Uniform Resource Locator) là một chuỗi ký tự dùng để xác định một tài nguyên cụ thể trên Internet. URL thường bao gồm ba phần chính:

* ***Protocol****: Định rõ phương thức truy cập tài nguyên, chẳng hạn như HTTP, HTTPS, FTP, và nhiều phương thức khác.*
* ***Domain Name****: Địa chỉ của máy chủ web hoặc nơi lưu trữ tài nguyên, thường được biết đến dưới dạng tên miền, chẳng hạn như "*[*www.example.com*](http://www.example.com/)*".*
* ***Path****: Đường dẫn đến tài nguyên cụ thể trên máy chủ, bao gồm thư mục và tên tệp, nếu có.*

Ví dụ, URL "https://www.example.com/index.html" sẽ trỏ đến tài nguyên "index.html" được lưu trữ trên máy chủ có tên miền "[www.example.com](http://www.example.com/)", và được truy cập thông qua giao thức HTTPS.

**Browser**

Phần mềm dùng để duyệt và hiển thị các trang web trên Internet, như Chrome, Firefox, hoặc Safari.

Browser, hay trình duyệt web, là một phần mềm được sử dụng để duyệt và hiển thị các trang web trên Internet. Trình duyệt web nhận các yêu cầu từ người dùng, sau đó gửi các yêu cầu này đến các máy chủ web tương ứng và hiển thị nội dung trả về từ các máy chủ đó trên màn hình của người dùng. Một số trình duyệt web phổ biến bao gồm Google Chrome, Mozilla Firefox, Microsoft Edge, Safari và Opera. Trình duyệt web thường hỗ trợ nhiều tính năng như lưu trữ danh sách yêu thích, quản lý mật khẩu, cải thiện bảo mật và hỗ trợ cho các tiện ích mở rộng.

**HTML (Hypertext Markup Language)**

Ngôn ngữ đánh dấu được sử dụng để tạo cấu trúc và định dạng nội dung trên các trang web.  
HTML (Hypertext Markup Language) là ngôn ngữ đánh dấu được sử dụng để tạo cấu trúc và định dạng nội dung trên các trang web. HTML sử dụng các thẻ (tag) để định dạng và hiển thị các phần tử trên trang web, bao gồm văn bản, hình ảnh, video, liên kết, biểu mẫu và nhiều nội dung khác. Các thẻ HTML thường đi kèm với các thuộc tính để cung cấp thông tin bổ sung về cách hiển thị hoặc tính năng của các phần tử đó.

Ví dụ, để hiển thị một đoạn văn bản đơn giản trong HTML, chúng ta có thể sử dụng thẻ **<p>**

<p>hello word</p>

Trong đoạn mã này, thẻ **<p>** được sử dụng để bắt đầu một đoạn văn bản, và **</p>** được sử dụng để kết thúc đoạn văn bản. Nội dung của đoạn văn bản được đặt giữa hai thẻ này và sẽ được hiển thị là một đoạn văn bản trên trang web.

***HTTP (Hypertext Transfer Protocol)***

Giao thức được sử dụng để truyền dẫn các trang web và dữ liệu giữa máy chủ và trình duyệt web.

HTTP (Hypertext Transfer Protocol) là một giao thức được sử dụng để truyền dẫn dữ liệu giữa máy chủ và trình duyệt web. Giao thức này định rõ cách trình duyệt web và máy chủ web giao tiếp với nhau để truyền dẫn các trang web và dữ liệu trên Internet.

Cụ thể, khi người dùng gõ một địa chỉ URL vào trình duyệt, trình duyệt sẽ tạo một yêu cầu HTTP để yêu cầu trang web tương ứng từ máy chủ web. Máy chủ web sau đó sẽ phản hồi bằng cách gửi trang web đó trở lại trình duyệt của người dùng thông qua giao thức HTTP. Giao thức này cũng có thể được mở rộng để hỗ trợ các loại dữ liệu và yêu cầu phức tạp hơn, như HTTP/2 và HTTP/3.

***HTTPS (Hypertext Transfer Protocol Secure)***

Phiên bản bảo mật của HTTP, sử dụng mã hóa để bảo vệ dữ liệu khi truyền dẫn trên Internet

HTTPS (Hypertext Transfer Protocol Secure) là một phiên bản bảo mật của giao thức HTTP (Hypertext Transfer Protocol). HTTPS sử dụng mã hóa để bảo vệ dữ liệu khi truyền dẫn trên Internet, giúp ngăn chặn việc đánh cắp thông tin cá nhân hoặc chiếm đoạt thông tin trên các trang web.

Điều quan trọng nhất là HTTPS sử dụng SSL/TLS (Secure Sockets Layer/Transport Layer Security) để mã hóa dữ liệu giữa trình duyệt web của người dùng và máy chủ web. Khi một trình duyệt web kết nối đến một trang web thông qua HTTPS, nó sẽ thiết lập một kênh an toàn với máy chủ web, đảm bảo rằng dữ liệu được truyền dẫn giữa hai bên được mã hóa và an toàn.

Sự phổ biến của HTTPS ngày càng tăng lên do sự chú ý đến bảo mật trên Internet, và nhiều trình duyệt web đã bắt đầu cảnh báo người dùng khi họ truy cập các trang web không sử dụng HTTPS.

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

*a. WordPress*

WordPress là một phần mềm thiết kế web rất phù hợp cho các nhà thiết kế mới vào nghề. Phần mềm có một số chủ đề và plugin tích hợp sẵn để người dùng cài đặt, chỉnh sửa và tùy chỉnh trên trang web của họ. Nhiều sự lựa chọn có thể giúp bạn tìm thấy một trang web phù hợp với phong cách, sở thích và nhu cầu trang web của bạn.

*b. Adobe Dreamweaver*

Adobe Dreamweaver mang đến cho bạn cơ hội xây dựng trang web của mình từ những bước đầu . Nói cách khác, bạn có thể viết mã trực tiếp thiết kế cụ thể mà bạn muốn cho trang web của mình. Bạn thậm chí có thể tạo các thiết kế đáp ứng, đảm bảo một trang web được tối ưu hóa để xem trên thiết bị di động và cả desktop. Mặc dù Dreamweaver yêu cầu bạn nhập code, nhưng bạn không cần phải có kiến ​​thức sâu rộng về các ngôn ngữ lập trình khác nhau. Phần mềm này chủ yếu hoạt động với HTML và chỉnh sửa trực quan, điều này là tối ưu nếu bạn mới sử dụng Dreamweaver.

Hãy nhớ rằng so với các công cụ khác như WordPress, làm việc trong Dreamweaver mất nhiều thời gian và nỗ lực hơn vì bạn đang tự lập trình trang web. Mặc dù vậy, công cụ này giúp bạn tạo ra thiết kế chính xác mà bạn muốn cho trang web của mình. Nó cũng có thể làm cho bạn cảm thấy hoàn thiện hơn khi biết bạn đã nỗ lực tạo ra trang web từ những điều cơ bản nhất.

*c. InVision Studio*

InVision Studio là một công cụ giao diện người dùng với nhiều tính năng và ứng dụng khác nhau cho các Web Designer khám phá. Với InVision, bạn có thể tạo giao diện trang web thân thiện với người dùng và có tính thẩm mỹ cao. Công cụ thiết kế web này cũng có tính năng tạo mẫu nhanh, cho phép các nhà phát triển web tạo các chuyển đổi web cho các dự án hoạt họa của họ.

*d. Adobe Photoshop*

Các nhà thiết kế web sáng tạo sử dụng Adobe Photoshop để tạo các bản in và mẫu cho các dự án thiết kế của họ. Nó không chỉ có nhiều màu sắc và độ chuyển màu mà bạn còn có thể tạo tác phẩm nghệ thuật kỹ thuật số dựa trên bản vẽ của chính mình. Phù hợp với các nhà thiết kế thành thạo, Photoshop cung cấp cho bạn khả năng tạo các thiết kế ấn tượng khá nhanh chóng.

e. *Phác thảo*

Sketch là một công cụ thiết kế giao diện người dùng vector với kích thước tệp nhỏ hơn so với Photoshop. Nó có một hệ thống lưới tích hợp và có một cảm giác tối thiểu. Cảm giác tối thiểu này mang lại một thiết kế gọn nhẹ giúp công cụ dễ sử dụng hơn. Bạn cũng có thể tìm thấy hàng trăm plugin thông qua cộng đồng Sketch để trợ giúp quy trình làm việc của bạn với tư cách là một Web Designer.

1. *Wix*

Wix là một trình xây dựng trang web hoạt động giống như WordPress. Nó cung cấp cho bạn tùy chọn để xây dựng nhiều loại trang web, chẳng hạn như online store hoặc blog. Wix cung cấp cả gói miễn phí và cao cấp, gói sau cung cấp hỗ trợ Wix thông qua lưu trữ trang web, bảo mật và sao lưu. Nhìn chung, Wix có thể giúp bạn tạo sự hiện diện trên Internet đơn giản.

*g. Google Web Designer*

Google Web Designer sử dụng các ngôn ngữ lập trình như HTML5, CSS và JavaScript để tạo nội dung, nó chủ yếu phục vụ cho việc tạo digital ads nếu bạn chưa quen với công cụ này. Công cụ thiết kế web này có giao diện người dùng đồ họa dễ sử dụng với những thứ như hoạt ảnh 3D và các hình dạng cơ bản. Bạn có thể tùy chỉnh thêm trang web của mình thông qua các màu sắc khác nhau và thư viện với các thành phần bổ sung như hình ảnh và video. Nếu bạn có kinh nghiệm với Google Web Designer, bạn thậm chí có thể chuyển đổi giữa chế độ xem thiết kế và code. Điều này mang lại cho bạn khả năng tạo các thiết kế nâng cao hơn nữa ngoài quảng cáo.

h.Adobe XD

Adobe XD là một công cụ thiết kế vector và tạo wireframe. Nó có các công cụ vẽ, bản xem trước trên thiết bị di động và máy tính để bàn cũng như khả năng chia sẻ thiết kế của bạn để bạn có thể nhận được phản hồi. Bạn cũng có thể chọn kích thước bảng vẽ cụ thể cho một thiết bị cụ thể và nhập bộ giao diện người dùng. Vì đây là một sản phẩm của Adobe, bạn cũng có quyền truy cập vào các thư viện của Adobe, cho phép bạn nhập các thiết kế và nội dung từ phần mềm như Photoshop hoặc Illustrator một cách dễ dàng.

*i.Figma*

Công cụ thiết kế giao diện này cho phép nhiều nhà thiết kế làm việc trên cùng một dự án cùng một lúc. Không giống như nhiều công cụ thiết kế web khác, bạn có thể mở phần mềm này trên trình duyệt. Nhìn chung, Figma cho phép bạn nhanh chóng tạo các thiết kế cho thiết bị di động, máy tính bảng và máy tính để bàn với sự trợ giúp của các nội dung và kiểu dáng có thể tìm kiếm.

## 1.4 Kết luận

Thiết kế web và triển khai hệ thống phần mềm đóng vai trò quan trọng trong việc xây dựng và phát triển các ứng dụng web hiện đại. Qua quá trình thiết kế web, chúng ta có thể tạo ra các trang web hấp dẫn, dễ sử dụng và có hiệu suất cao, cung cấp trải nghiệm người dùng tốt nhất có thể. Sự thiết kế web chủ động và sáng tạo cũng đóng vai trò quan trọng trong việc xây dựng nhận thức thương hiệu và tương tác với khách hàng.

Trong khi đó, việc triển khai hệ thống phần mềm đòi hỏi sự chính xác, kiểm soát và kiến thức chuyên môn vững chắc. Quá trình này bao gồm việc triển khai các mã nguồn mở hoặc tùy chỉnh, quản lý cơ sở dữ liệu, tích hợp các công nghệ và giao thức, và đảm bảo tính ổn định và bảo mật của hệ thống.

Tóm lại, sự kết hợp giữa thiết kế web sáng tạo và triển khai hệ thống phần mềm chính xác và chuyên nghiệp là chìa khóa để tạo ra các ứng dụng web thành công, mang lại giá trị cho người dùng và doanh nghiệp. Qua việc liên tục cập nhật và nâng cấp, chúng ta có thể duy trì và phát triển sự hiện diện trực tuyến một cách bền vững và mang lại trải nghiệm tốt nhất cho người dùng.

Top of Form

**CHƯƠNG II: HTML & HTML5**

**(**Hyper Text Markup Language**)**

2.1. Tổng quan về HTML

2.1.1 Khái quát về HTML

HTML là viết tắt của cụm từ Hypertext Markup Language, là một ngôn ngữ đánh dấu được thiết kế ra để tạo nên các trang web trên World Wide Web.

HTML không phải là ngôn ngữ lập trình, chỉ giống như Microsoft Word, dùng để bố cục và định dạng trang web. Giúp người dùng tạo và cấu trúc các thành phần trong trang web hoặc ứng dụng, phân chia các đoạn văn, heading, links,…

HTML sử dụng cấu trúc đơn giản (tags và attributes) để đánh dấu lên trang web.



Ngôn ngữ HTML là gì?

Để hiểu rõ hơn bạn cần nắm một số ý chính liên quan đến cách thức hoạt động của HTML như thế nào nhé!

Đầu tiên, HTML documents là file kết thúc với đuôi .html hay .htm. Bạn có thể xem chúng bằng cách sử dụng bất kỳ trình duyệt đọc các file HTML này và xuất bản nội dung lên internet sao cho người đọc có thể xem được chúng.

Trung bình một trang web chứa được nhiều trang web HTML (Ví dụ như trang chủ, trang about us, trang liên hệ).

Mỗi trang HTML chứa một bộ các tag (elements) – đây được xem như cách để bạn xem như là việc xây dựng từng khối của một trang web. Bao gồm section, paragraph, heading và những khối nội dung khác.

Bạn có thể xem chúng bằng các trình duyệt web như Google Chrome, Firefox, … Hầu hết các HTML element đều có một tag mở và một tag đóng với cấu trúc <tag></tag>.

2.1.2 Lịch sử hình thành

HTML được tạo ra bởi Tim Berners-Lee, một nhà vật lý học ở Thuỵ Sĩ. HTML đã trở thành một chuẩn Internet được Tổ chức World Wide Web Consortium vận hàng và phát triển.

Phiên bản đầu tiên gồm 18 tag HTML 1.0, xuất hiện năm 1991. Sao đó các nhà phát triển đã thay thế HTML bằng XHTML 2.0 vào năm 2000.

Đến năm 2014, HTML được nâng cấp lên chuẩn HTML5 với nhiều tag được thêm vài markup, nhằm để giúp các nhà phát triển xác định rõ nội dung thuộc loại gì.

Hiện nay, HTML có khoảng 140 tag tuy nhiên có một số tag đã bị tạm ngưng.

2.1.3 Thuật ngữ HTML

Sau đây là 3 thuật ngữ thường gặp nhất trong HTML bạn không thể bỏ qua!

***Elements***

Elements được định nghĩa là các thẻ metadata của HTML. Một phần tử HTML được xác định dựa trên ba thành phần sau:

* Thẻ mở
* Nội dung nằm bên trong nội dung của phần tử
* Thẻ đóng

Tuy nhiên, ở một số trường hợp đặc biệt mà chúng chỉ có một thành phần duy nhất là thẻ mở – chúng được gọi là phần tử rỗng.

Trong một tệp HTML, nếu bạn soạn thảo mã sai quy tắc thì các phần tử sẽ không hiển thị đúng. Nên sau đây là những điều bạn cần lưu ý về elements:

* Đừng bao giờ quên đóng thẻ đóng. Trừ trường hợp đó là phần tử rỗng, vì nếu như thiếu thẻ đóng sẽ làm ảnh hưởng đến các phần tử khác, vô hình các phần tử đó bị xem là nội dung của phần tử thiếu thẻ đóng
* Không được xáo trộn thứ tự giữa các thẻ đóng
* Không nên trình bày mã lệnh một cách ngẫu hứng mà bạn nên tuân theo một tiêu chuẩn chung để đảm bảo tối ưu hoá việc viết mã lệnh
* Tên thẻ không phân biệt chữ hoa và chữ thường. Tuy nhiên các lập trình viên hầu hết dùng chữ thường để viết.

***Tags***

Trong tệp HTML, block-level tags cùng inline tags là các thẻ được dùng chủ yếu nhất. Ở đây chúng ta sẽ tìm hiểu về 2 loại này trước nhé

**#1. Block-level tags**

Đây là tag cấp cao và chiếm phần lớn không gian trang web, và bắt đầu dòng mới của trang.

Có 3 loại block-level tags mà bất kỳ trang HTML nào cũng cần là <html></html>, <head></head> và <body></body>.

**#2. Inline tags**

Tuy chỉ là tag chiếm một phần nhỏ không gian trang web, và không bắt đầu dòng mới của trang.

Tag này được sử dụng để định dạng, thiết lập bố cục cho nội dung có trong block-level tags.

Có nhiều kiểu inline tags trong HTML. Ví dụ, tag <strong></strong> định dạng chữ in đậm, tag <em></em> định dạng chữ in nghiêng.

**Các thẻ cơ bản của HTML**

**Thẻ tiêu đề:** Bất kỳ tài liệu nào cũng bắt đầu với một tiêu đề. HTML có sáu cấp độ tiêu đề là <h1>, <h2>, <h3>, <h4>, <h5> và <h6>

**Thẻ đoạn văn:** Thẻ <p> cung cấp cấu trúc văn bản của bạn thành các đoạn khác nhau. Mỗi đoạn văn bản phải ở giữa một thẻ <p> mở và thẻ đóng </p>

**Thẻ ngắt dòng:** Khi bạn sử dụng <br /> mọi thứ theo sau sẽ bắt đầu từ dòng tiếp theo. Thẻ này phải có khoảng trắng giữa ký tự br và dấu gạch chéo, nếu không có hiển thị dòng sẽ gặp khó khan.

***Attributes***

Nghĩa là thuộc tính phần tử HTML, mỗi thẻ HTML ta có thể xem như một đối tượng. Lúc này đối tượng HTML sẽ có các thuộc tính mô tả cho chúng.

Mỗi thuộc tính sẽ được đại diện bởi một từ khoá bằng tiếng anh. Có 4 thuộc tính cốt lõi được sử dụng thường xuyên như:

* **Id:** có thể được sử dụng để nhận diện bất kỳ yếu tố trong một trang HTML.
* **Title:** là một tiêu đề gợi ý cho phần tử thường được hiển thị dưới dạng một tooltip khi con con trỏ đi qua phần tử
* **Class:** được sử dụng để kết hợp với một phần tử với một style sheet,
* **Style:** Thuộc tính này cho phép bạn chỉ định quy tắc CSS trong phần tử

Ngoài ra còn có 3 thuộc tính quốc tế hoá có sẵn các phần tử XHTML như:

* **Dir:** chỉ cho các trình duyệt về hướng trong văn bản.
* **Lang:** cho phép bạn chỉ ra ngôn ngữ chính được sử dụng trong một tài liệu
* **Xml:lang:** là sự thay thế cho thuộc tính lang trong XHTNL.

***Cấu trúc một bài HTML cơ bản***

Giống như một bài văn thông thường sẽ có cấu trúc 3 phần Mở bài – Thân bài – Kết bài. Viết mã lệnh trong tập tin HTML cũng tương tự như vậy.

 Cấu trúc cơ bản của HTML gồm 3 phần:

<!Doctype>: Phần khai báo chuẩn của html hay xhtml.

<head></head>: Phần khai báo ban đầu, khai báo về [meta](https://hocwebchuan.com/reference/tag/tag_meta.php), [title](https://hocwebchuan.com/reference/tag/tag_title.php), [css](https://hocwebchuan.com/tutorial/tut_css.php), javascript…

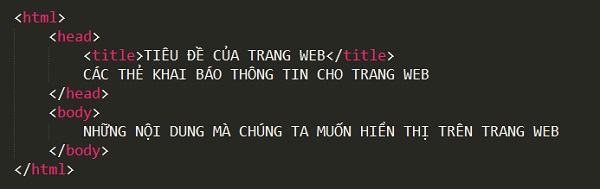
<body></body>: Phần chứa nội dung của trang web, nơi hiển thị nội dung.

Nội dung của một tập tin HTML phải được bắt đầu bằng thẻ <html> và kết thúc bởi thẻ </html>/

Bên trong cặp thẻ <html></html> có 2 phần chính:

**Phần mở đầu:** Được bắt đầu bằng <head> và kết thúc bởi thẻ </head>, nơi đây chứa tiêu đề và các thẻ khai báo thông tin cho trang web.

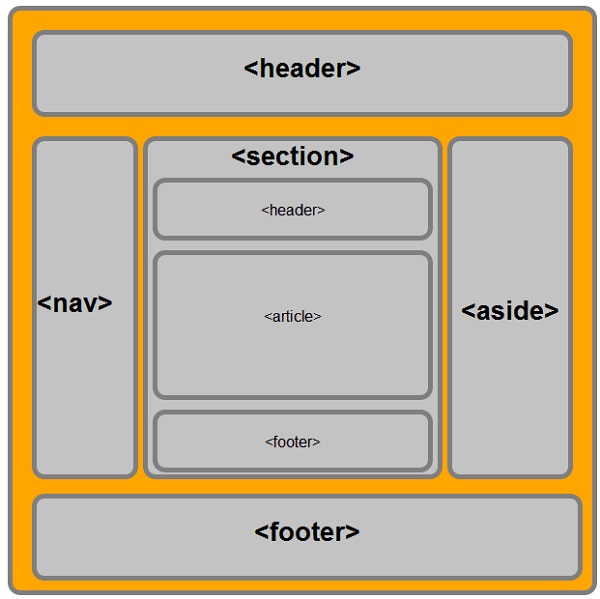
**Phần thân bài viết:** Được bắt đầu bằng thẻ <body> và kết thúc bởi thẻ </body> dùng để chứa những nội dung mà chúng ta muốn hiển thị trên website



Ngôn ngữ lập trình HTML5

Hiện nay HTML5 đang được sử dụng phổ biến, hãy làm quen với bố cục này nhé. Do HTML5 có chứa các thẻ ngữ nghĩa, xây dựng bố cục gồm:

* **Thẻ header:** Định nghĩa phần đầu của trang web
* **Thẻ nav:** thanh điều hướng (menu)
* **Thẻ section:** định nghĩa một phần web
* **Thẻ article:** định nghĩa là nội dung bài viết độc lập
* **Thẻ aside:** phần bên cạnh nội dung
* **Thẻ footer:** Chân trang



Bố cục của một HTML cơ bản

Ưu và nhược điểm của HTML

*Ưu điểm*

* HTML được sử dụng rộng rãi
* Dễ học và dễ sử dụng
* Sử dụng trên hầu hết mọi trình duyệt
* Có mã nguồn mở và hoàn toàn miễn phí
* Markup gọn gàng và thống nhất, thân thiện với người dùng
* Được vận hành bởi W3C
* Dễ dàng tích hợp với các ngôn ngữ backend khác như PHP hay Node.js.

*Nhược điểm*

* Chỉ dùng chủ yếu cho các web tĩnh. Nếu bạn muốn các tính năng động, bạn cần sử dụng JavaScript hoặc ngôn ngữ backend thứ ba như PHP
* Chỉ thực hiện một số logic nhất đinh cho người dùng. Tuy nhiên, hầu hết các trang đều cần được tạo riêng biệt, kể cả khi sử dụng cùng các yếu tố
* Một số trình duyệt chậm hỗ trợ tính năng mới
* Khó kiểm soát quá trình thực hiện của trình duyệt.

2.2 Cấu trúc tổng quát trang HTML

Cấu trúc cơ bản của trang HTML có dạng như sau, thường gồm 3 phần:

* <!Doctype>: Phần khai báo chuẩn của html hay xhtml.
* <head></head>: Phần khai báo ban đầu, khai báo về [meta](https://hocwebchuan.com/reference/tag/tag_meta.php), [title](https://hocwebchuan.com/reference/tag/tag_title.php), [css](https://hocwebchuan.com/tutorial/tut_css.php), javascript…
* <body></body>: Phần chứa nội dung của trang web, nơi hiển thị nội dung.

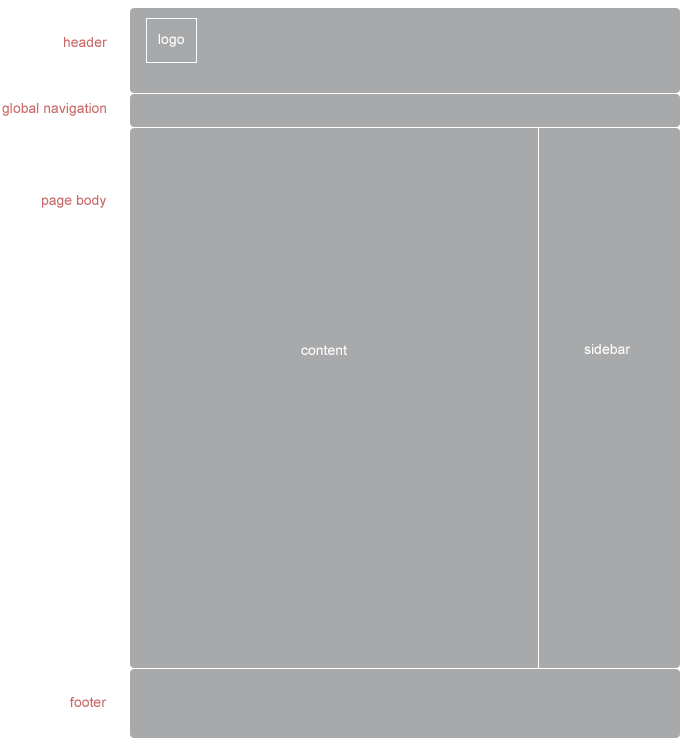
**Cấu trúc cơ bản**

<!DOCTYPE html>  
<html>  
<head>  
<title>Tiêu đề trang web</title>  
</head>  
  
<body>  
...Phần thân viết ở đây...  
</body>  
</html>

Cấu trúc cơ bản của trang web

Mỗi trang web đều có cách thể hiện cấu trúc khác nhau, có trang 1 cột, có trang 2 và cũng có trang chứa nhiều cột, bên dưới đây chúng ta tham khảo một trang đơn giản sử dụng 2 cột để layout.

* **Phần đầu:** header, có thể chứa logo, câu slogan, các liên kết, các banner liên kết, các button, đoạn flash, hoặc các form ngắn như form tìm kiếm,...
* **Phần liên kết toàn cục:** global navigation, dùng để chứa các liên kết đến những trang quan trọng trong toàn bộ trang, trong phần này có thể chứa thêm các liên kết con (sub navigation).
* **Phần thân của trang:** page body, phần này chứa phần nội dung chính (content) và phần nội dung phụ (sidebar).
* **Phần nội dung chính:** content, phần này chứa nội dung chính cần thể hiện cho người dùng xem.
* **Phần nội dung phụ:** sidebar, phần này có thể chứa liên kết phụ của từng trang (local navigation), hoặc các banner chứa liên kết liên quan, hoặc có thể dùng để chứa các liên kết quảng cáo,...
* **Phần cuối trang web:** footer, phần này thường chứa phần liên hệ như: tên công ty, địa chỉ, số điện thoại, mail liên hệ,... và đặc biệt là copyright, hoặc có thể chứa các liên kết toàn trang, các banner liên kết,...



*Cấu trúc của trang web cơ bản HTML*

## 2.3. Các thẻ HTML thông dụng

2.3.1Các thẻ xử lý đoạn văn bản

Bây giờ chúng ta lần lượt đi qua các cặp thẻ hay sử dụng nhất. Xin nói thêm: thẻ có cấu trúc: <tên\_thẻ thuộc\_tính1="gia\_tri1" thuộc\_tính2="gia\_tri2">. Tất nhiên các bạn có thể bỏ một số thuộc tính đi (lúc đó, các thuộc tính bị bỏ đi sẽ được đặt ngầm định bởi trình duyệt, rất khó chịu)  
  
a. Các thẻ xử lý đoạnThẻ phân chia đoạn  
Trong HTML, các đoạn tài liệu, văn bản, hình ảnh… được phân chia bằng cặp thẻ <P>văn bản</P>  
Thẻ <P> có 1 số thuộc tính sau:  
Align: Thuộc tính này sẽ chứa 1 trong 3 giá trị:  
- center: Đoạn tài liệu sẽ được canh chỉnh vào giữa  
- left: Đoạn tài liệu sẽ được canh chỉnh theo lề trái  
- right: Đoạn tài liệu sẽ được canh chỉnh theo lề phải  
- justify: Đoạn tài liệu sẽ được canh chỉnh theo hai bên  
Ví dụ:  
<HTML>  
<BODY>  
<p align ="justify"> Thử một tí</p>  
</BODY>  
</HTML>  
  
Style: Thuộc tính này sẽ quy định khoảng cách lề của đoạn. Trong thuộc tính này lại có các thuộc tính con, tuy nhiên các thuộc tính con này được đặt cách nhau bằng dấu chấm phẩy. Tất nhiên các bạn cũng có thể loại bỏ thuộc tính con:  
- margin-left : x (x là số nguyên chỉ định chiều rộng của lề trái)  
- margin-right : y (y là số nguyên chỉ định chiều rộng của lề phải)  
- margin-top: z (z là số nguyên chỉ định chiều rộng của lề trên)  
- margin-bottom: t(t là số nguyên chỉ định chiều rộng của lề dưới)  
- line-height: u% (u: khoảng cách giữa các dòng tính theo đơn vị 100 %)  
Ví dụ:  
<HTML>  
<BODY>  
<p align ="center" style ="margin-left: 10; margin-right: 5; margin-top: 6; margin-bottom: 6"> Thử hai tí. Tí thứ 2 này xác định đoạn văn bản căn giữa, có lề trái = 10, lề phải bằng 5, lề trên = 6, lề dưới = 6. Hết tí thứ 2.</p>  
<p align ="left" > Thử ba tí. Tí thứ 3 này xác định đoạn văn bản căn trái, các lề đặt theo mặc định của trình duyệt</p>  
<p align ="left" style="line-height: 150%" > Thử ba tí. Tí thứ 3 này xác định đoạn văn bản căn trái, các lề đặt theo mặc định của trình duyệt</p>  
  
</BODY>  
</HTML>  
Thẻ xuống dòng  
Trong HTML, các ký tự xuống dòng không được sử dụng. Để ngắt một dòng nào đó, ta dùng thẻ <BR>. Đây là 1 thẻ đơn:  
<HTML>  
<HEAD>  
<TITLE></TITLE>  
</HEAD>  
<BODY>  
<p align="left" style="margin-left : 30">  
Trung thu là tết thiếu nhi<BR>Tại sao người lớn lại đi chơi nhiều <BR>Chơi nhiều thì sẽ làm liều<BR>Làm liều lại đẻ ra nhiều thiếu nhi<BR></p>  
  
</BODY>  
</HTML>  
  
Chú ý:  
Một cặp thẻ khác cũng được sử dụng để canh chỉnh đoạn tài liệu hay bảng biểu:  
- <center>…</center>: Xác định canh chỉnh một đoạn tài liệu hoặc bảng biểu vào giữa trang.

2.3.2 Các thẻ xử lý font chữ:a. Thẻ Meta:  
Thẻ này có nhiều thuộc tính khác nhau và cũng làm nhiều nhiệm vụ khác nhau. Tuy nhiên ở đây tôi chỉ nói về cách ứng dụng thẻ này để hiển thị các đoạn mã tiếng Việt.  
  
Trước đây các loại font chữ tiếng Việt rất phong phú, điều này khiến cho người sử dụng tiếng Việt trên thế giới phải than trời ầm ỹ cả lên. Rất may là tại thời điểm này, hầu hết các font chữ tiếng Việt thời "đồ đá" không còn được ứng dụng trong thiết kế web nữa, thay vào đó là các chuẩn quốc tế Unicode.  
Ở đây tôi xin giới thiệu 2 chuẩn Unicode tiếng Việt phổ biến và cách sử dụng thẻ meta cho từng trường hợp cụ thể:  
- Mã UTF-8: Là mã font Unicode rút gọn (biểu diễn font chữ theo kiểu 8 bit. Khi ta khai báo:  
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">  
Trình duyệt sẽ hiểu là từ khi đọc được dòng chữ này, nó sẽ giải mã các ký tự theo kiểu UTF-8. Và khi bạn gửi dữ liệu từ mẫu biểu của HTML đi, dữ liệu đó cũng phải được mã hoá theo tiêu chuẩn UTF-8.  
  
Chẳng hạn, để trình duyệt hiển thị đúng bài thơ:  
  
Trung thu là tết thiếu nhi  
Tại sao người lớn lại đi chơi nhiềuChơi nhiều thì sẽ làm liều  
Làm liều lại đẻ ra nhiều thiếu nhi.  
  
Thì đoạn mã HTML do ta sinh ra phải có dạng:  
  
<HTML>  
<BODY>  
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">  
<p align="left" style="margin-left : 30">  
<img src ="bigreen.gif">  
Trung thu lÃ táº¿t thiáº¿u nhi<BR>Táº¡i sao ngÆ°á»i lá»›n láº¡i Ä‘i chÆ¡i nhiá»u <BR>ChÆ¡i nhiá»u thÃ sáº½ lÃ m liá»u<BR>LÃ m liá»u láº¡i Ä‘áº» ra nhiá»u thiáº¿u nhi.<BR></p>  
</BODY>  
</HTML>  
Tất nhiên chúng ta có thể sử dụng chương trình Vietkey hoặc Unikey để gõ tiếng việt theo kiểu UTF-8 (sử dụng bảng mã Unicode UTF-8), nhưng sẽ hơi khó chịu một tí .  
  
- Mã UTF-16 bít: Là mã font Unicode 16 bít. Ta phải khai báo như sau:  
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">  
Trình duyệt sẽ hiểu là từ khi đọc được dòng chữ này, nó sẽ giải mã các ký tự theo kiểu UTF-16 bit. Và khi bạn gửi dữ liệu từ mẫu biểu của HTML đi, dữ liệu đó cũng phải được mã hoá theo tiêu chuẩn UTF- 16 bít trên.  
  
Chẳng hạn, để trình duyệt hiển thị đúng bài thơ:  
  
Trung thu là tết thiếu nhi  
Tại sao người lớn lại đi chơi nhiều  
Chơi nhiều thì sẽ làm liều  
Làm liều lại đẻ ra nhiều thiếu nhi.  
  
Thì đoạn mã HTML do ta sinh ra phải có dạng:  
  
<HTML>  
<BODY>  
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">  
<p align="left" style="margin-left : 30">  
<img src ="bigreen.gif">  
Trung thu l#224; t#7871;t thi#7871;u nhi<BR>T#7841;i sao ng#432;#7901;i l#7899;n l#7841;i #273;i ch#417;i nhi#7873;u.<BR>Ch#417;i nhi#7873;u th#236; s#7869; l#224;m li#7873;u<BR>L#224;m li#7873;u l#7841;i #273;#7867; ra nhi#7873;u thi#7871;u nhi<BR></p>  
</BODY>  
</HTML>  
Tất nhiên chúng ta có thể sử dụng chương trình Vietkey hoặc Unikey để gõ tiếng Việt theo chuẩn này với hệ thống bảng mã Unicode UCS2  
  
**Chú ý:**  
1: Một số chương trình soạn thảo có hỗ trợ chế độ hiển thị mã tiếng Việt trong quá trình soạn thảo. Khi đó nếu các bạn mở chế độ mã HTML mà vẫn đọc được tiếng Việt như thường thì các bạn cứ ung dung gõ tiếng Việt theo kiểu Unicode thông thường là được. Còn nếu nó hiển thị ra các ký tự loằng ngoằng thì các bạn phải đoán mò ra dạng chuẩn mã hoá font của nó và sử dụng chế độ gõ chữ tương ứng.  
2: Trong tài liệu này tôi sẽ đánh tiếng Việt thông thường, các bạn sẽ phải chịu trách nhiệm xử lý mã tiếng Việt trong các đoạn ví dụ.  
  
b) Cặp thẻ <font>…</font>  
Cặp thẻ này quy định tên font, màu sắc, kích cỡ font. Nó có những thuộc tính sau:  
Face: Tên của font, chẳng hạn: Arial, .VnTime, Times New Roman  
Size: Kích cỡ của font, bé nhất là 1.  
Color: Màu sắc cùa font, bao gồm dấu # và 6 chữ số hex tiếp theo quy định mã màu RGB  
Ví dụ:  
<HTML>  
<BODY>  
<p align ="center"><font face="Arial" size="5" color="#800000">Tết trung thu</font></p>  
<p align="left" style="margin-left : 30"><font size ="2" name ="Verdana">  
Trung thu là tết thiếu nhi<BR>Tại sao người lớn lại đi chơi nhiều <BR>Chơi nhiều thì sẽ làm liều<BR>Làm liều lại đẻ ra nhiều thiếu nhi<BR></font></p>  
</body></html>  
  
**c.** Các thẻ định dạng chữ nghiêng, đậm, gạch chân:  
- Định dang chữ đậm:  
Cặp thẻ <B>…</B> xác định đoạn văn bản ở giữa sẽ bị in đậm. Cặp thẻ này không có tham số kèm theo.  
- Định dang chữ nghiêng:  
Cặp thẻ <i>…</i> xác định đoạn văn bản ở giữa sẽ bị in nghiêng. Cặp thẻ này không có tham số kèm theo.  
- Định dang chữ gạch chân:  
Cặp thẻ <u>…</u> xác định đoạn văn bản ở giữa sẽ bị gạch chân. Cặp thẻ này không có tham số kèm theo.  
Ví dụ:  
<HTML>  
<BODY>  
<p align ="center"><font face="Arial" size="5" color="#800000"><B><I>Tết trung thu</font></I></B></p>  
<p align="left" style="margin-left : 30"><font size ="2" name ="Verdana">  
Trung thu là tết thiếu nhi<BR>Tại sao người lớn lại đi chơi nhiều <BR>Chơi nhiều thì sẽ làm liều<BR>Làm liều lại đẻ ra nhiều thiếu nhi<BR></font></p>  
</body></html>

Căn bản về HTML - Các thẻ liên kết

2.3.3 Các thẻ liên kết:  
  
a. Thẻ liên kết với hình ảnh:  
Trong HTML, chúng ta không thể chèn trực tiếp toàn bộ ruột gan của một file hình ảnh, mà ta phải chỉ dẫn đến hình ảnh đặt bên ngoài. Để làm điều này, ta dùng thẻ <img>. Đây là 1 thẻ đơn.  
Thẻ này có một số thuộc tính sau:  
- Src: Xác định địa chỉ URL của hình ảnh:  
- align: Xác định kiểu canh lề:  
- right: Canh theo lề phải  
- left:Canh theo lề trái  
- center:Canh theo lề giữa.  
  
<HTML>  
<HEAD>  
<TITLE></TITLE>  
</HEAD>  
<BODY>  
<p align="left" style="margin-left : 30">  
<img src ="bigreen.gif" align = "right">  
Trung thu là tết thiếu nhi<BR>Tại sao người lớn lại đi chơi nhiều <BR>Chơi nhiều thì sẽ làm liều<BR>Làm liều lại đẻ ra nhiều thiếu nhi<BR></p>  
  
</BODY>  
</HTML>  
- border: Xác định chiều dày của viền bao quanh ảnh.  
<HTML>  
<HEAD>  
<TITLE></TITLE>  
</HEAD>  
<BODY>  
<p align="left" style="margin-left : 30">  
<img src ="bigreen.gif" border ="5">  
Trung thu là tết thiếu nhi<BR>Tại sao người lớn lại đi chơi nhiều <BR>Chơi nhiều thì sẽ làm liều<BR>Làm liều lại đẻ ra nhiều thiếu nhi<BR></p>  
</BODY>  
</HTML>  
  
b. Thẻ liên kết trang web.  
Để tạo một liên kết tới một trang web khác, ta dùng cặp thẻ <a></a>. Cặp thẻ này có các thuộc tính sau:  
Href: Địa chỉ URL của trang web cần liên kết tới. bạn đã có một bookmark trên trang web, bạn có thể trỏ đến vị trí của bookmark bằng cách dùng thuộc tính href với dấu # và tên bookmark.  
name: Xác định tên của bookmark (điểm liên kết trong nội tại trang web).  
Ví dụ, ta có 2 trang web:  
Trang thứ nhất có địa chỉ là "tettrungthu.htm", có nội dung sau:  
<HTML>  
<HEAD>  
<TITLE></TITLE>  
</HEAD>  
<BODY>  
<p align="left" style="margin-left : 30">  
<img src ="bigreen.gif" border ="5">  
Trung thu là tết thiếu nhi<BR>Tại sao người lớn lại đi chơi nhiều <BR>Chơi nhiều thì sẽ làm liều<BR>Làm liều lại đẻ ra nhiều thiếu nhi<BR></p>  
</BODY>  
</HTML>  
  
Trang thứ 2 có tên là danhsachthovui.htm, có nội dung sau:  
<HTML>  
<HEAD>  
<TITLE></TITLE>  
</HEAD>  
<BODY>  
<p align="left" style="margin-left : 30">  
Tet trung thu  
</p>  
</BODY>  
</HTML>  
Giả sử ta muốn thêm một liên kết với file "tettrungthu.htm" vào chữ Tet trung thu trong file danhsachthovui.htm, ta phải chèn cặp thẻ <a>...</a> như sau:  
  
<HTML>  
<HEAD>  
<TITLE></TITLE>  
</HEAD>  
<BODY>  
<p align="left" style="margin-left : 30">  
<a href ="tettrungthu.htm">Tet trung thu</a>  
</p>  
</BODY>  
</HTML>  
Chú ý: cả 2 file này phải được đặt cùng thư mục. Trong trường hợp đặt khác thư mục các bạn phải ghi rõ đường dẫn đến file kia.

Căn bản về HTML - Các thẻ tạo bảng

2.3.4. Các cặp thẻ xử lý bảng:  
Các bảng trong HTML được định nghĩa như sau:  
Định nghĩa 1 bảng bởi cặp thẻ <table></table>  
Trong 1 bảng (table) được chia làm nhiều dòng , mỗi dòng giới hạn bởi 1 cặp thẻ <tr></tr>.  
Trong mỗi dòng lại có các ô , giới hạn bởi cặp thẻ <td></td>  
Chẳng hạn để định nghĩa 1 bảng gồm có 1 dòng và 3 ô, ta làm như sau:  
<table>  
<tr>  
<td>Ô thứ nhất</td>  
<td>Ô thứ 2</td>  
<td>Ô thứ 3</td>  
</tr>  
</table>  
  
Hoặc để định nghĩa một bảng gồm 2 dòng, mỗi dòng 3 ô:  
<table>  
<tr>  
<td>Ô thứ nhất dòng 1</td>  
<td>Ô thứ 2 dòng 1</td>  
<td>Ô thứ 3 dòng 1</td>  
</tr>  
<tr>  
<td>Ô thứ nhất dòng 2</td>  
<td>Ô thứ 2 dòng 2</td>  
<td>Ô thứ 3 dòng 2</td>  
</tr>  
</table>  
Chú ý: Số lượng các ô trong các dòng phải bằng nhau, nếu 1 dòng nào đó có số ô khác với dòng khác, bảng sẽ bị "vỡ kế hoạch". Để khắc phục điều này, ta phải tính trước trong 1 bảng, số lượng tối đa các ô của 1 dòng sẽ là bao nhiêu, sau đó có thể dùng thuộc tính colspan của thẻ <td> để gộp các ô trống trong cùng 1 hàng lại với nhau.  
  
Thuộc tính colspan sẽ chỉ định số lượng các ô được gộp vào nhau trên một hàng.  
Chẳng hạn:  
<table>  
<tr>  
<td colspan = "2" >Chập ô thứ nhất và ô thứ 2 của dòng 1</td>  
<td>Ô thứ 3 dòng 1</td>  
</tr>  
<tr>  
<td>Ô thứ nhất dòng 2</td>  
<td>Ô thứ 2 dòng 2</td>  
<td>Ô thứ 3 dòng 2</td>  
</tr>  
</table>  
  
Một số thuộc tính có liên quan:  
Thẻ Table:  
- border: Xác định độ dày của khung bao quanh bảng  
- bordercolor: Màu của khung bao quanh, viết dưới dạng dấu # và 6 chữ số HEX tiếp theo.  
- cellspacing: Xác định khoảng cách giữa các ô trong bảng.  
- width: Xác định độ rộng của bảng  
- ;td>  
- width: Độ rộng của ô  
- height: Chiều cao của ô  
- colspan: Xác định bao nhiêu ô tính từ ô đó sẽ được chập vào làm 1 (trên cùng 1 dòng)  
- rowspan: Xác định bao nhiêu ô tính từ ô đó sẽ được chập vào làm 1 (trên cùng 1 cột).  
- ;form> và kết thúc bởi thẻ </form>. Giữa 2 cặp thẻ này, các bạn có thể sử dụng các cặp thẻ HTML khác.  
Thẻ form có một số thuộc tính sau:  
- method  
Thuộc tính này có 2 giá trị POST hoặc GET, để xác định dữ liệu gửi lên theo kiểu POST hay GET.  
 Kiểu GET chính là kiểu mà khi nhập dữ liệu lên máy chủ, các dữ liệu này sẽ được hiển thị trên ô Address dưới dạng các căp tên=giá\_tri. Nhược điểm của kiểu này là toàn bộ cái URL và xâu tên=giá\_trị kia sẽ bị giới hạn dưới 255 ký tự (do đặc điểm của trình duyệt). Vì vậy để có thể gửi nhiều dữ liệu hơn, người ta đã sinh ra kiểu POST. Với kiểu này, dữ liệu sẽ không bị giới hạn chiều dài 255 ký tự của chuỗi địa chỉ do không bị gộp vào chuỗi địa chỉ. Kiểu POST cũng thường dùng để truyền các dữ liệu nhạy cảm mà người sử dụng không muốn hiển thị trên ô Address (password chẳng hạn).  
Ví dụ:  
<form method = "post"> Thử một tí  
</form>  
- action: Thuộc tính này sẽ chỉ định form gửi dữ liệu đến trang nào. Trong trường hợp thuộc tính này không được khai báo, form sẽ gửi thẳng dữ liệu và yêu cầu về chính trang hiện hành (sau đó trình duyệt sẽ tải lại nội dung mới).  
Ví dụ:  
<form method = "post" action ="thu2ti.php"> Thử hai tí  
</form>  
Tuy nhiên, 2 ví dụ trên chưa có ý nghĩa gì, vì chúng ta chưa trang bị các thành phần cơ bản của form như ô văn bản, nút bấm...  
 *Các thẻ nhập vào (input)*Thẻ input có một số thuộc tính, mỗi thuộc tính sẽ quy định những chức năng riêng:  
- name: Chỉ định tên cho thẻ. Tên này sẽ được gửi lên máy chủ cùng giá trị nhập vào cho thẻ.  
- Value: Xác định giá trị đặt trước cho thẻ.  
- type:  
Thuộc tính này có một số giá trị sau:  
  
\* submit: Sẽ xác định một nút bấm (submit) mà khi kích chuột vào đó, form sẽ tự động kích hoạt và gửi dữ liệu đi  
\* text: Hiển thị một ô văn bản cho phép người sử dụng nhập văn bản  
\* password: Hiển thị ô văn bản để nhập password.  
\* hidden: Tạo một biến ẩn, ta có thể dùng nó để lưu trữ các dữ liệu "ẩn" khỏi người dùng, tất nhiên nếu họ biết View source lên thì chịu.  
  
Ví dụ: đoạn mã sau sinh ra một form có 1 ô văn bản, một ô nhập mật khẩu và 1 nút bấm:  
  
<form method="POST">  
<p>  
User Name:<input type="text" name="T1" size="20"> </p>  
<p>  
Password:  
<input type="password" name="T2" size="20"><input type="submit" value="Login" name="B1"></p>  
</form>  
  
**Thẻ tạo hộp chọn xổ xuống:**  
Danh sách sổ xuống là một ô danh sách cho phép bạn kích chọn 1 mục trong danh sách sẵn có.  
Để tạo một hộp chọn sổ xuống, ta phải sử dụng 2 thẻ: Thẻ Select và thẻ Option như sau:  
<Select name =\*\*\*>  
<option value = gia\_trí1>nội dung 1</option>  
<option value = gia\_trí2>nội dung 2</option>  
<option value = gia\_trí3>nội dung 3</option>  
.....  
<option value = gia\_trín>nội dung n</option>  
</select>  
Trong đó:  
Thẻ Select với thuộc tính name sẽ xác định tên của hộp danh sách sổ xuống.  
Các thẻ option xác định giá trị của tên biến trong thẻ select nếu được chọn. Giá trị sẽ được gán vào biến nằm trong thuộc tính value của thẻ option.  
  
Ví dụ:  
  
<form method="POST">  
<p>  
User Name:<input type="text" name="T1" size="20"> </p>  
<p>  
Password:  
<input type="password" name="T2" size="20"><input type="submit" value="Login" name="B1"></p>  
<p>\*\*\*: <Select name ="\*\*\*">  
<option value =1>Male </option>  
<option value =0>Female </option>  
</select>  
</p>  
</form>  
  
Khi người dùng kích chọn Male, giá trị 1 sẽ được chuyển vào biến \*\*\* (được định nghĩa trong thẻ Select>).  
  
Các thẻ lựa chọn radio:  
Các thẻ này sẽ cho phép hiển thị ra một danh sách các nút tròn, mà chúng ta sẽ chỉ được phép kích chọn 1 trong các nút đó.  
Để hiển thị các nút radio này, các bạn có thể dùng thẻ input với type là radio. Một nhóm các nút radio này sẽ được thiết lập nếu như thuộc tính name của chúng trùng nhau:  
<input type="radio" value="" name="R1">  
  
Ví dụ dưới đây mô phỏng dòng chọn kiểu gõ (telex, VNI) trên diễn đàn:  
  
<input type=radio name="switcher" value="OFF" >Tắt  
<input type=radio name="switcher" checked value="TELEX" >Telex  
<input type=radio name="switcher" value="VNI"> VNI  
  
Thẻ nhập khối văn bản  
Thẻ input với thuộc tính type = text chỉ cho phép ta hiển thị ra một ô văn bản nhỏ (hiện được 1 dòng). Để hiện ra một ô soạn thảo lớn, có thể hiển thị nhiều dòng đồng thời, ta phải dùng thẻ textarea:  
<textarea name =\*\*\*>Giá trị mặc định của khối văn bản </textarea>  
  
Còn một số cặp thẻ khác, nhưng ít nhất thì tôi cũng không nhớ ra tại thời điểm này. Các bạn có thể vào FrontPage, sau đó sử dụng menu Insert/Form để chèn các thành phần này vào, sau đó các bạn có thể vào xem và thay đổi các thuộc cơ bản của chúng. Còn bây giờ, chúng ta quay lại với việc lấy dữ liệu của PHP:  
  
Để lấy các biến theo kiểu POST, PHP sẽ tự động sinh ra mảng có tên là $HTTP\_POST\_VARS[]. Mảng này có chỉ số chính là tên của các phần tử trong form và giá trị là nội dung giá trị do người sử dụng nhập vào các phần tử có tên tương ứng. Chẳng hạn với mẫu biểu sau:  
  
<form method="POST">  
<p>  
User Name:<input type="text" name="T1" size="20"> </p>  
<p>  
Password:  
<input type="password" name="T2" size="20"></p>  
<p>\*\*\*: <Select name ="\*\*\*">  
<option value =1>Male </option>  
<option value =0>Female </option>  
</select>  
</p>  
<input type="submit" value="Gui di" name="B1">  
</form>

CSS: Cascading Style Sheet - Định kiểu trình bày trang HTML

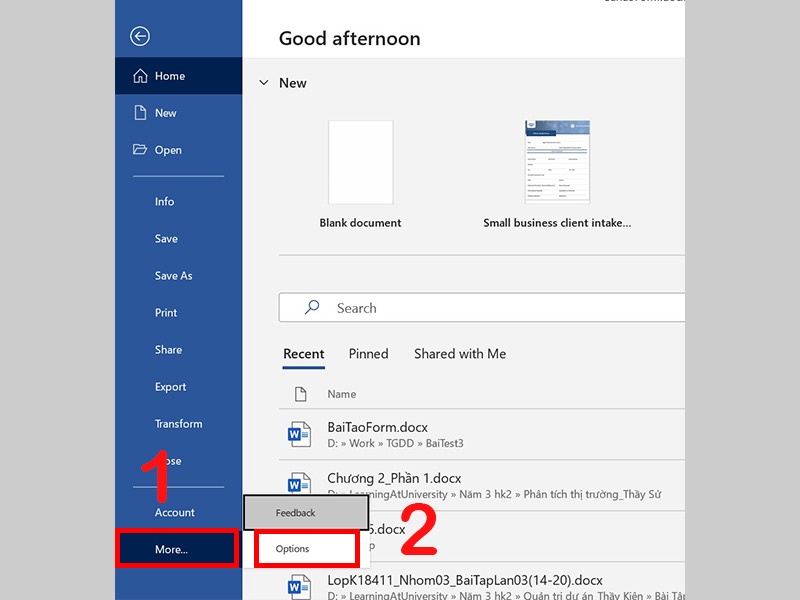
copyright (c) by dvc102 - Faculty of Informatic Technology, Hanoi National University of Education)  
Nội dung chủ yếu của bài được dịch từ http://www.w3schools.com/css/default.asp , phần còn lại từ... một số nguồn khác và trong đầu  
  
Trước khi đọc bài này các bạn cần có kiến thức cơ bản về WWW, HTML cùng những kỹ năng cơ bản nhất để làm một trang web  
  
CSS là gì?  
- CSS: Cascading Style Sheet: Đây là những mẫu để quy định cách thức thể hiện các thẻ HTML. Bạn có thể xem ví dụ về cách thức thể hiện các thẻ này ở http://www.w3schools.com/css/demo\_default.htm  
- CSS có 3 cách sử dụng:  
+ Sử dụng trực tiếp kèm với các thẻ HTML (Inline Style Sheet)  
+ Định nghĩa trong 1 trang web (Internal Style Sheet)  
+ Định nghĩa thành 1 file CSS riêng (External Style Sheet)  
- Style được đưa vào HTML 4.0 để giải quyết một số vấn đề.  
- CSS giúp bạn tiết kiệm được rất nhiều thời gian và công sức cho việc thiết kế web.  
- Bạn có thể định nghĩa nhiều style vào một thẻ HTML  
Style (mẫu định dạng) giải quyết một số vấn đề chung:  
- Ta biết rằng các thẻ HTML chuẩn được thiết kế để định nghĩa nội dung của một văn bản. Đầu tiên các thẻ HTML hỗ trợ cách viết "Đây là tiêu đề", "Đây là đoạn", "Đây là bảng".... bằng cách sử dụng các thẻ <H1>, <P>, <TABLE>... Cách bố trí văn bản này được qui định bởi trình duyệt web và không có bất cứ một thẻ nào để định dạng văn bản.  
- Đến các trình duyệt thế hệ sau đặc biệt là Nescape và Internet Explorer tiếp tục đưa thêm vào các thẻ HTML mới cùng các thuộc tính định dạng riêng của mình (như các thẻ <FONT> và thuộc tính Color...). Do đó ngày càng khó để tạo ra được một web site khi mà nội dung của nó bị tách rời khỏi cách bố trí.  
- Để giải quyết vấn đề này World Wide Web Consortium (W3C) đã tạo ra STYLE để đưa thêm vào trong HTML 4.0  
- Cả hai trình duyệt lớn là Nescape Và Internet Explorer đều hỗ trợ CSS.  
CSS giúp bạn tiết kiệm được rất nhiều thời gian và công sức cho việc thiết kế web.  
- Style trong phiên bản HTML 4.0 (phiên bản chúng ta đang dùng) qui định cách thức thể hiện các thẻ HTML tương tự như thẻ <FONT> hay thuộc tính COLOR trong HTML 3.2. Style thường được lưu trong các file nằm ngoài trang web. Chúng giúp bạn có thể thay đổi cách thức định dạng và cách bố trí các trang web chỉ bằng cách thay đổi riêng file CSS. Chỉ khi bạn muốn thay đổi lại toàn bộ màu sắc, cách định dạng của các tiêu đề, nội dung bạn mới hiểu rõ tác dụng thực sự của CSS.  
- CSS cho phép chúng ta điều khiển cách định dạng và cách bố trí của cùng lúc nhiều trang web với chỉ duy nhất 1 lần thay đổi tại một vị trí. Là một người thiết kế web, bạn có thể định nghĩa 1 file CSS cho các thẻ HTML và áp dụng nó vào nhiều trang web mà bạn muốn. Để thay đổi tổng thể các trang web này bạn chỉ đơn giản là thay đổi file CSS và tất cả các trang đã áp dụng sẽ được thay đổi một cách tự động.  
Bạn có thể định nghĩa nhiều style vào một thẻ HTML  
CSS cho phép bạn đưa các thông tin định nghĩa thẻ thông qua nhiều con đường khác nhau. Style có thể được qui định ở trong chỉ một thẻ HTML, được qui định trong 1 trang web hoặc ở trong một file CSS bên ngoài.  
Thứ tự áp dụng các định dạng  
Như trên đã nói, ta có thể sử dụng nhiều cách khác nhau để làm CSS. Điều gì sẽ xảy ra nếu bạn áp dụng nhiều cách định dạng cho 1 thẻ HTML?  
Theo một cách chung nhất ra có thể nói các style của bạn sẽ được "xếp tầng" (cascade). Việc xếp tầng này tuân theo thứ tự sau: (Số 1 là ưu tiên nhất, số 4 là kém ưu tiên nhất).  
1. Inline Style (Style được qui định trong 1 thẻ HTML cụ thể)  
2. Internal Style (Style được qui định trong phần <HEAD> của 1 trang HTML )  
3. External Style (style được qui định trong file CSS ngoài)  
4. Browser Default (thiết lập mặc định của trình duyệt)  
Như vậy ta thấy các thiết lập trong 1 thẻ HTML có mức ưu tiên cao nhất, Những gì được định nghĩa ở đây sẽ bị bỏ qua tất cả các định nghĩa khác (như trong thẻ <HEAD>, File CSS ngoài,...)

2.4. Các thẻ tạo biểu mẫu (form)

**Hiển thị tab Developer trên thanh Ribbon**

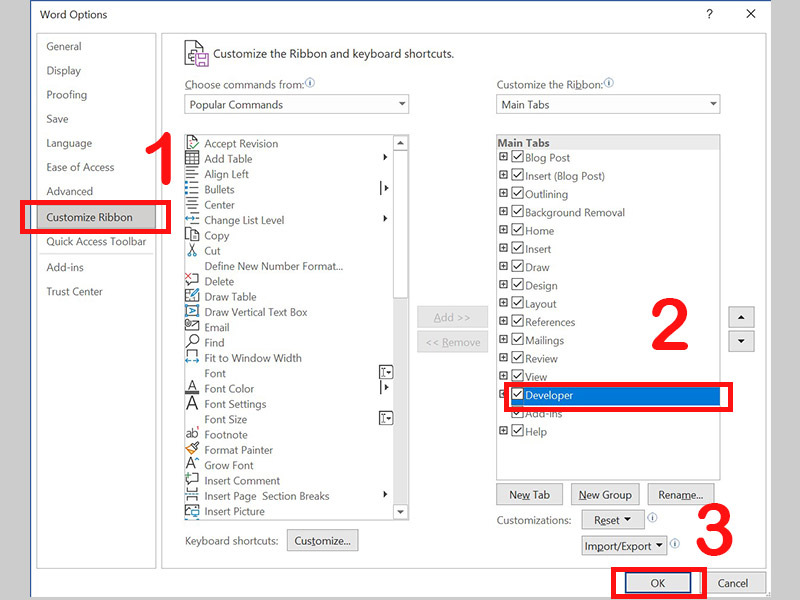
**Bước 1**: Vào **File** > Chọn **Options**.

Nếu không thấy Options thì vào **More** > chọn **Options**.



Vào File để chọn Option

**Bước 2**: Chọn **Customize Ribbon** > Tích chọn **tab** **Developer** > Nhấn **OK**.

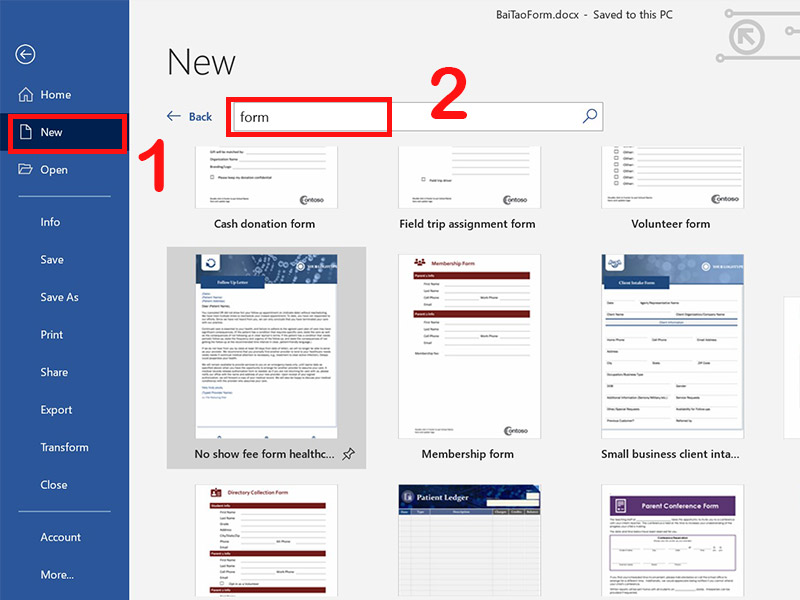


Hiển thị tab Developer trên thanh Ribbon

**Tạo form từ template có sẵn hoặc từ trang trắng**

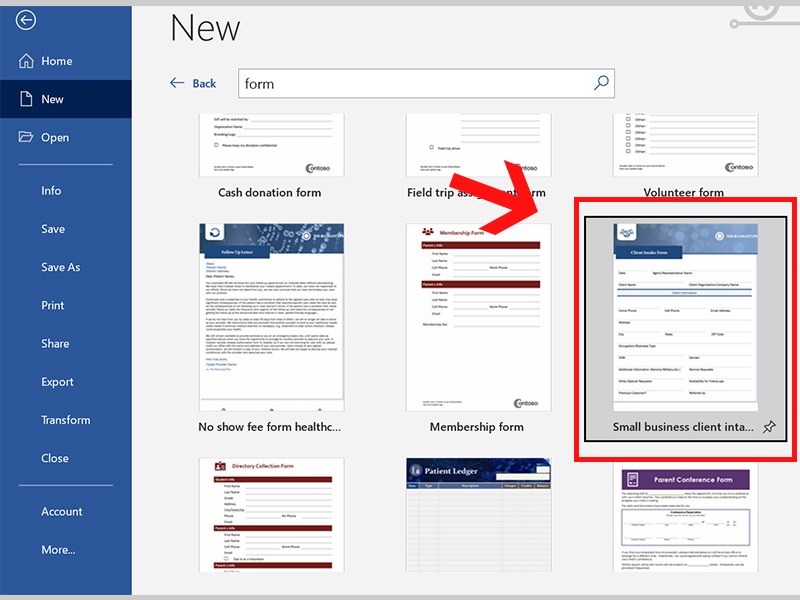
**- Cách 1: Tạo form từ template có sẵn**

**Bước 1**: Vào **File** > Chọn **New** > Nhập “form” vào ô search và nhấn **Enter**.



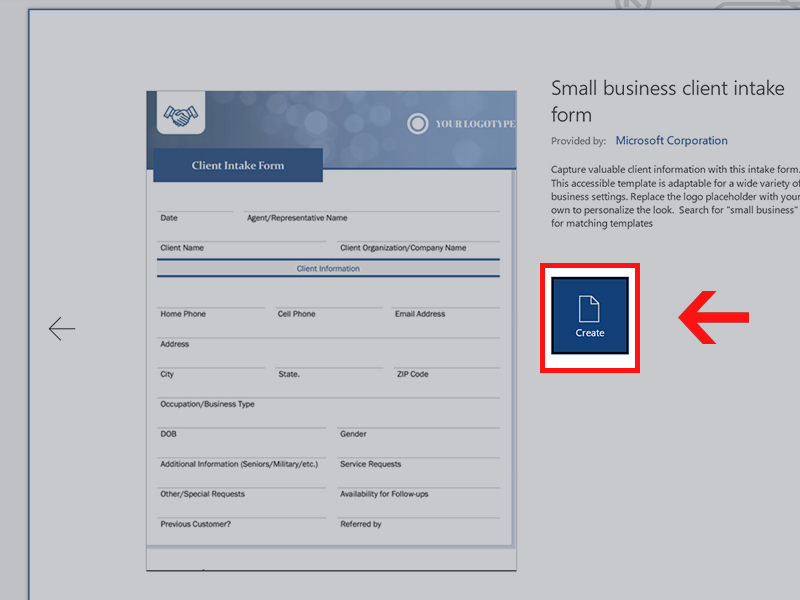
Tìm form template có sẵn

**Bước 2**: Nhấn **chọn 1 template có sẵn**.



Chọn form từ template có sẵn

**Bước 3**: Nhấn **Create**.

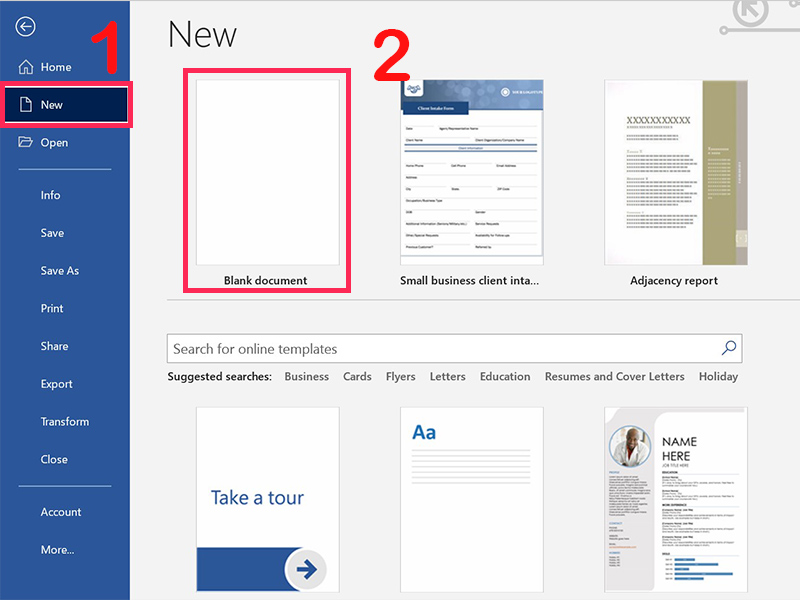


Chọn Create để dùng template có sẵn

**- Cách 2: Tạo form từ trang trắng**

Lấy ví dụ nhỏ là để tạo một form sổ tự bạch, thì trong đó cần tạo khung sườn có các trường là: Tên của tôi, Ảnh, Giới tính, Sinh nhật, Sở thích bằng các thao tác thủ công.

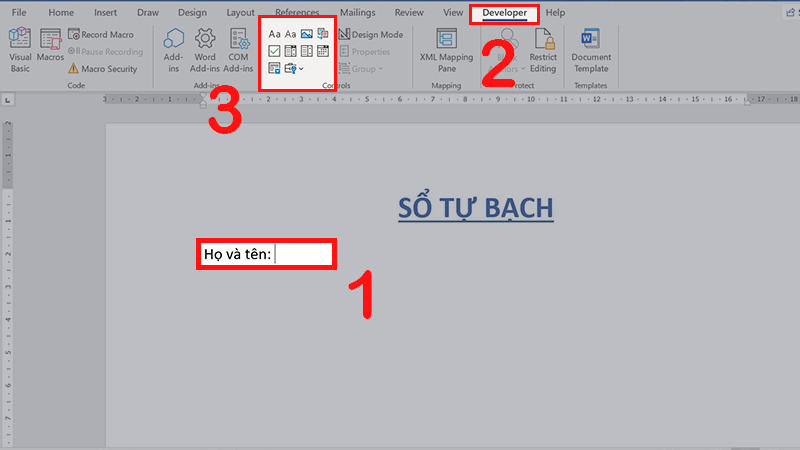
Vào**File** > Chọn **New**> Chọn **Blank Document** > Tạo 1 khung sườn form.



Tạo form từ trang trắng

**Tạo các trường nội dung cho form**

Đặt con trỏ vào vị trí cần thêm > Chọn **tab Developer** > Chọn kiểu trường cần thêm trong nhóm **Controls**.



Bước đầu tạo các trường nội dung cho form

Trong đó:

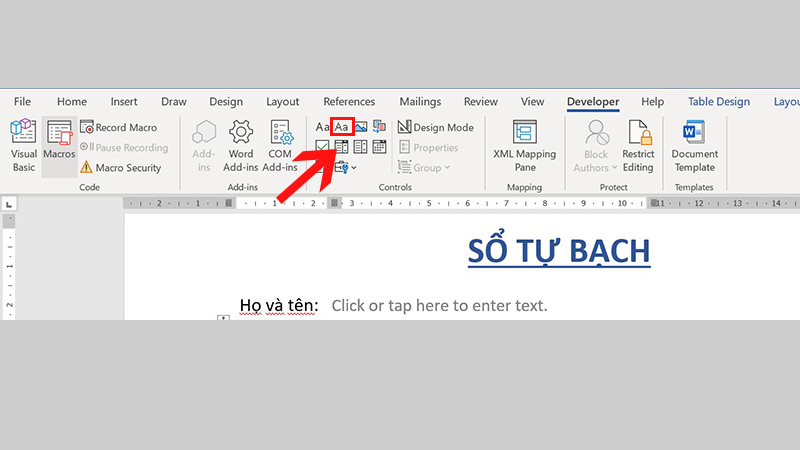
+ Trường dữ liệu **Plain Text**: Đây là trường văn bản, giúp người dùng nhập văn bản vào. Ví dụ như mục Họ và tên thì trường chọn là Plain Text.

**Ghi chú**:

- Text control có 2 loại: Plain Text và Rich Text.

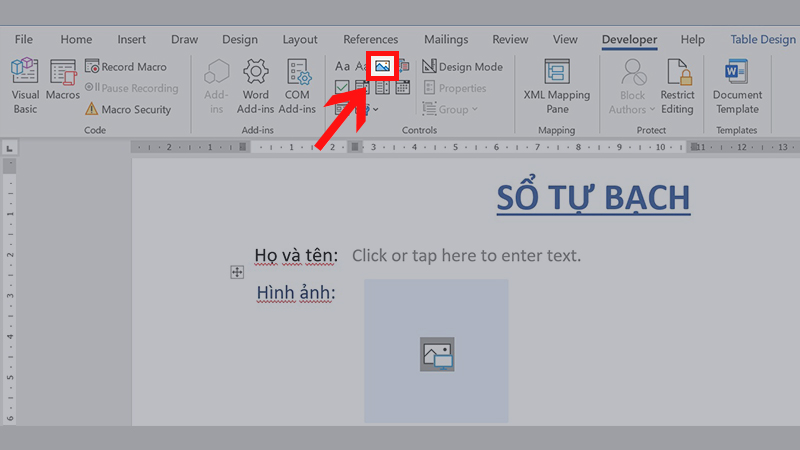
- Rich Text giúp người dùng có thể định dạng chữ in đậm, in nghiêng hoặc gạch chân, cho phép người dùng gõ nhiều đoạn văn bản.

- Để giới hạn định dạng người dùng nhập văn bản dùng Plain text sẽ phù hợp hơn, khi dùng Plain Text, người dùng sẽ chỉ nhập được trên đoạn văn bản.



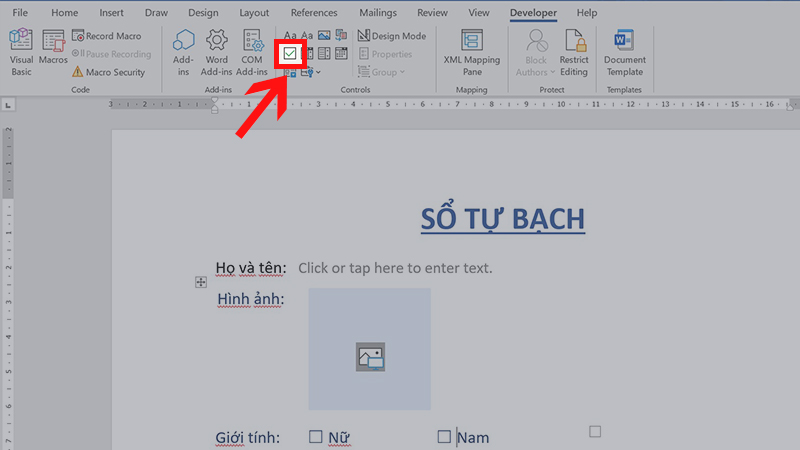
Tạo trường dữ liệu Plain Text

+ Trường dữ liệu **Picture**: Đây là trường giúp người dùng đưa dữ liệu dạng hình ảnh vào, đối với ví dụ trong bài, phần sử dụng dữ liệu Picture là Hình ảnh.



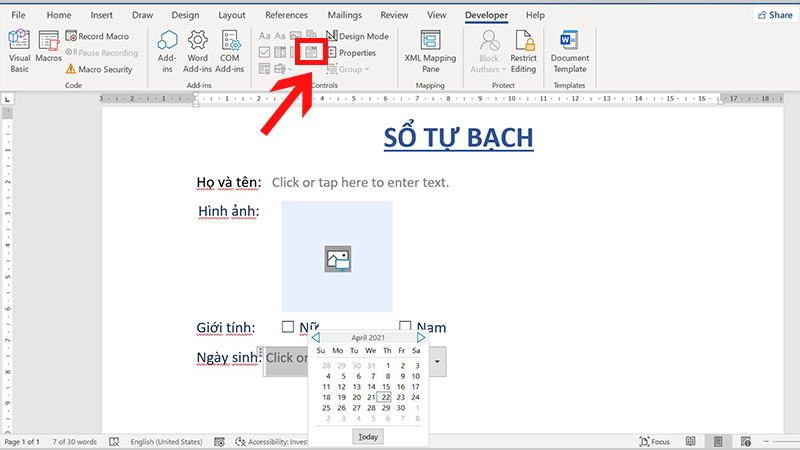
Tạo trường dữ liệu Picture

+ Trường dữ liệu **Check Box**: Đây là dữ liệu cho phép người dùng đánh dấu chọn vào ô, cụ thể hơn là hình vuông trong phần Giới tính được minh họa dưới hình.



Tạo trường dữ liệu Check Box

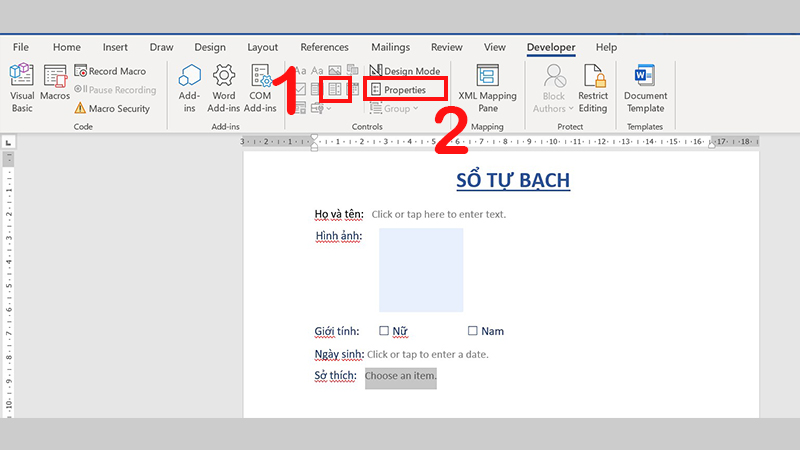
+ Trường dữ liệu **Date Picker**: Đây là hộp kiểm thời gian, dữ liệu cho phép người dùng nhập hoặc chọn ngày, tháng, năm, ví dụ minh họa như phần Ngày sinh trong hình.



Tạo trường dữ liệu Date Picker

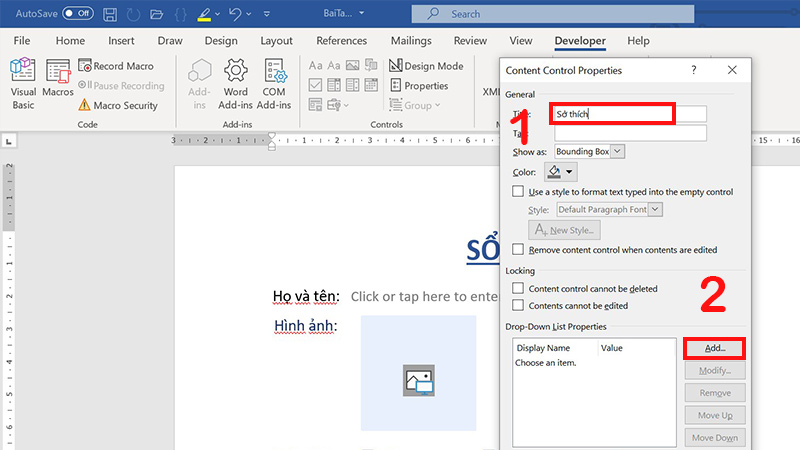
+ Trường dữ liệu **Drop-Down**: Đây là cách tạo một danh sách và cho phép người dùng chọn một trong các danh sách đã cung cấp.

**Bước 1**: Đặt con trỏ tại vị trí muốn chèn **Drop-Down** > Chọn **tab Developer** > Chọn biểu tượng **Drop-Down** > Chọn **Properties**.



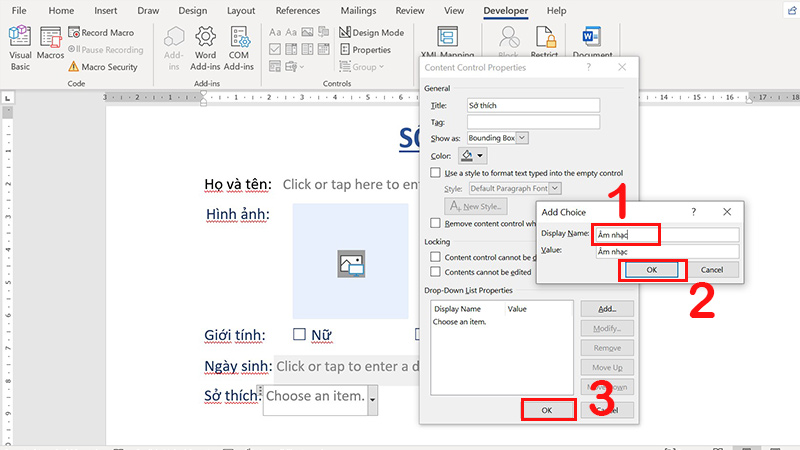
Tạo trường dữ liệu Drop-Down

**Bước 2**: Nhập tên **Title** > Chọn **Add**.



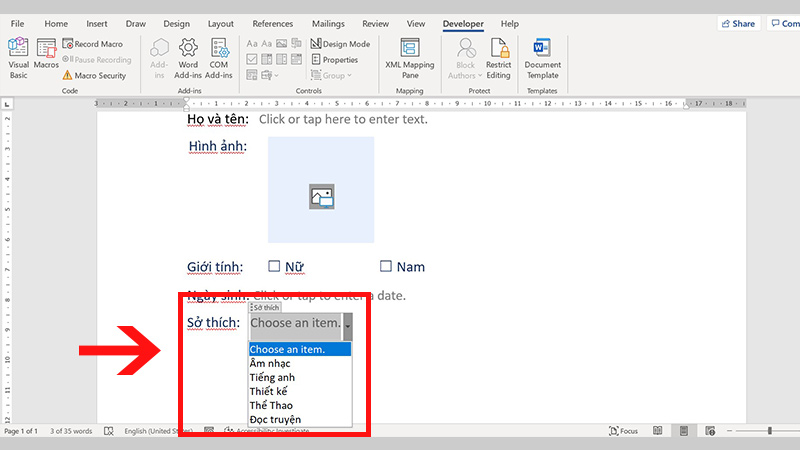
Điền Title cho Drop-Down

**Bước 3**: **Nhập nội dung** để người dùng chọn > Nhấn **OK** > Chọn **OK**.



Thêm nội dung để chọn

Để thêm danh sách nội dung chọn trong phần Drop-Down, bạn cần lặp lại các thao tác như bước 2, bước 3 trên.

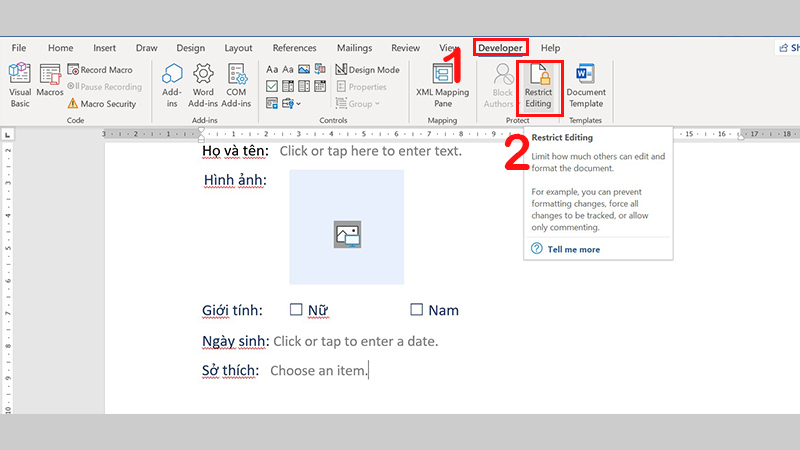


Kết quả sau khi thêm các mục chọn trong phần Drop-Down

**4. Thiết lập quyền bảo vệ cho biểu mẫu**

Dùng lệnh **Restrict Editing** để giới hạn quyền hạn và mức độ có thể chỉnh sửa hoặc định dạng biểu mẫu cho người dùng khác.

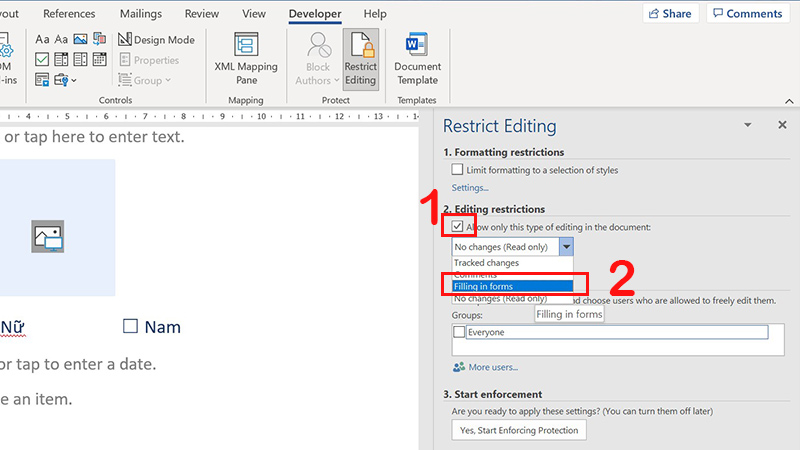
Tại **tab Developer** > Chọn **Restrict Editing**.



Hình chọn Restrict Editing

**+ Tại mục 2. Editing restrictions**

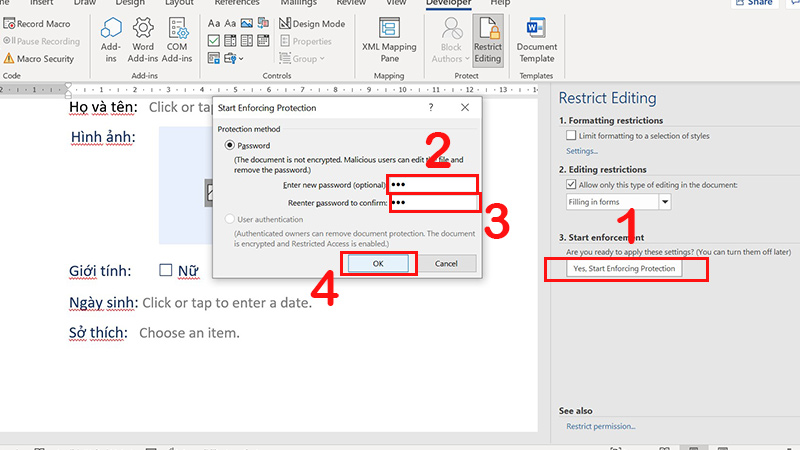
Chọn **Allow only this type of editing in the document** > Chọn **Filling in forms**.



Hình thiết lập Editing restrictions

**+ Tại mục 3. Start enforcement**

Chọn **Yes, Start Enforcing Protection** > Nhập mật khẩu > Nhập xác thực lần nữa > Nhấn **OK**.



Hình thiết lập Start enforcement

## 2.5. Một số thẻ HTML đặc biệt

| **HTML** | **Symbol** | **Numeric** | **Description** | **Hex** | **CSS (ISO)** | **JS (Octal)** |
| --- | --- | --- | --- | --- | --- | --- |
| **Special Characters** | | | | | | |
| &quot; | " | &#34; | quotation mark | u+0022 ISOnum | \0022 | \42 |
| &num; | # | &#35; | number sign | u+0023 ISOnum | \0023 | \43 |
| &dollar; | $ | &#36; | dollar sign | u+0024 ISOnum | \0024 | \44 |
| &percnt; | % | &#37; | percent sign | u+0025 ISOnum | \0025 | \45 |
| &amp; | & | &#38; | ampersand | u+0026 ISOnum | \0026 | \46 |
| &apos; | ' | &#39; | apostrophe | u+0027 ISOnum | \0027 | \47 |
| &lpar; | ( | &#40; | left parenthesis | u+0028 ISOnum | \0028 | \50 |
| &rpar; | ) | &#41; | right parenthesis | u+0029 ISOnum | \0029 | \51 |
| &ast; | \* | &#42; | asterisk | u+002A ISOnum | \002a | \52 |
| &plus; | + | &#43; | plus sign | u+002B ISOnum | \002b | \53 |
| &comma; | , | &#44; | comma | u+002C ISOnum | \002c | \54 |
| &minus; | - | &#45; | hyphen-minus | u+002D ISOnum | \002d | \55 |
| &period; | . | &#46; | full stop; period | u+002E ISOnum | \002e | \56 |
| &sol; | / | &#47; | solidus; slash | u+002F ISOnum | \002f | \57 |
| &colon; | : | &#58; | colon | u+003A ISOnum | \003a | \72 |
| &semi; | ; | &#59; | semicolon | u+003B ISOnum | \003b | \73 |
| &lt; | < | &#60; | less-than | u+003C ISOnum | \003c | \74 |
| &equals; | = | &#61; | equals | u+003D ISOnum | \003d | \75 |
| &gt; | > | &#62; | greater-than sign | u+003E ISOnum | \003e | \76 |
| &quest; | ? | &#63; | question mark | u+003F ISOnum | \003f | \77 |
| &commat; | @ | &#64; | at sign; commercial at | u+0040 ISOnum | \0040 | \100 |
| &lsqb; | [ | &#91; | left square bracket | u+005B ISOnum | \005b | \133 |
| &bsol; | \ | &#92; | backslash | u+005C ISOnum | \005c | \134 |
| &rsqb; | ] | &#93; | right square bracket | u+005D ISOnum | \005d | \135 |
| &Hat; | ^ | &#94; | circumflex accent | u+005E ISOnum | \005e | \136 |
| &lowbar; | \_ | &#95; | low line | u+005F ISOnum | \005f | \137 |
| &grave; | ` | &#96; | grave accent | u+0060 ISOnum | \0060 | \u0060 |
| &lcub; | { | &#123; | left curly bracket | u+007b ISOnum | \007b | \173 |
| &verbar; | | | &#124; | vertical bar | u+007c ISOnum | \007c | \174 |
| &rcub; | } | &#125; | right curly bracket | u+007d ISOnum | \007d | \175 |
|  | ~ | &#125; | tilde | u+007e ISOnum | \007e | \176 |
| **Latin** | | | | | | |
| &nbsp; |  | &#160; | no-break space | %A0 | \00a0 | \240 |
| &iexcl; | ¡ | &#161; | inverted exclamation mark | %A1 | \00a1 | \241 |
| &cent; | ¢ | &#162; | cent sign | %A2 | \00a2 | \242 |
| &pound; | £ | &#163; | pound sterling sign | %A3 | \00a3 | \243 |
| &curren; | ¤ | &#164; | general currency sign | %A4 | \00a4 | \244 |
| &yen; | ¥ | &#165; | yen sign | %A5 | \00a5 | \245 |
| &#x20B9; | ₹ | &#x20B9; | Indian Rupee (INR) sign | U+20B9 | \20B9 | \u20B9 |
| &brvbar; | ¦ | &#166; | broken (vertical) bar | %A6 | \00a6 | \246 |
| &sect; | § | &#167; | section sign | %A7 | \00a7 | \247 |
| &uml; | ¨ | &#168; | umlaut (dieresis) | %A8 | \00a8 | \250 |
| &copy; | © | &#169; | copyright sign | %A9 | \00a9 | \251 |
| &ordf; | ª | &#170; | ordinal indicator, feminine | %AA | \00aa | \252 |
| &laquo; | « | &#171; | angle quotation mark, left | %AB | \00ab | \253 |
| &not; | ¬ | &#172; | not sign | %AC | \00ac | \254 |
| &shy; | ­ | &#173; | soft hyphen | %AD | \00ad | \255 |
| &reg; | ® | &#174; | registered sign | %AE | \00ae | \256 |
| &macr; | ¯ | &#175; | macron | %AF | \00af | \257 |
| &deg; | ° | &#176; | degree sign | %B0 | \00b0 | \260 |
| &plusmn; | ± | &#177; | plus-or-minus sign | %B1 | \00b1 | \261 |
| &sup2; | ² | &#178; | superscript two | %B2 | \00b2 | \262 |
| &sup3; | ³ | &#179; | superscript three | %B3 | \00b3 | \263 |
| &acute; | ´ | &#180; | acute accent | %B4 | \00b4 | \264 |
| &micro; | µ | &#181; | micro sign | %B5 | \00b5 | \265 |
| &para; | ¶ | &#182; | pilcrow (paragraph sign) | %B6 | \00b6 | \266 |
| &middot; | · | &#183; | middle dot | %B7 | \00b7 | \267 |
| &cedil; | ¸ | &#184; | cedilla | %B8 | \00b8 | \270 |
| &sup1; | ¹ | &#185; | superscript one | %B9 | \00b9 | \271 |
| &ordm; | º | &#186; | ordinal indicator, masculine | %BA | \00ba | \272 |
| &raquo; | » | &#187; | angle quotation mark, right | %BB | \00bb | \273 |
| &frac14; | ¼ | &#188; | fraction one-quarter | %BC | \00bc | \274 |
| &frac12; | ½ | &#189; | fraction one-half | %BD | \00bd | \275 |
| &frac34; | ¾ | &#190; | fraction three-quarters | %BE | \00be | \276 |
| &iquest; | ¿ | &#191; | inverted question mark | %BF | \00bf | \277 |
| &Agrave; | À | &#192; | capital A, grave accent | %C0 | \00c0 | \300 |
| &Aacute; | Á | &#193; | capital A, acute accent | %C1 | \00c1 | \301 |
| &Acirc; | Â | &#194 ; | capital A, circumflex accent | %C2 | \00c2 | \302 |
| &Atilde; | Ã | &#195; | capital A, tilde | %C3 | \00c3 | \303 |
| &Auml; | Ä | &#196; | capital A, dieresis or umlaut mark | %C4 | \00c4 | \304 |
| &Aring; | Å | &#197; | capital A, ring | %C5 | \00c5 | \305 |
| &AElig; | Æ | &#198; | capital AE diphthong (ligature) | %C6 | \00c6 | \306 |
| &Ccedil; | Ç | &#199; | capital C, cedilla | %C7 | \00c7 | \307 |
| &Egrave; | È | &#200; | capital E, grave accent | %C8 | \00c8 | \310 |
| &Eacute; | É | &#201; | capital E, acute accent | %C9 | \00c9 | \311 |
| &Ecirc; | Ê | &#202; | capital E, circumflex accent | %CA | \00ca | \312 |
| &Euml; | Ë | &#203; | capital E, dieresis or umlaut mark | %CB | \00cb | \313 |
| &Igrave; | Ì | &#204; | capital I, grave accent | %CC | \00cc | \314 |
| &Iacute; | Í | &#205; | capital I, acute accent | %CD | \00cd | \315 |
| &Icirc; | Î | &#206; | capital I, circumflex accent | %CE | \00ce | \316 |
| &Iuml; | Ï | &#207; | capital I, dieresis or umlaut mark | %CF | \00cf | \317 |
| &ETH; | Ð | &#208; | capital Eth, Icelandic | %D0 | \00d0 | \320 |
| &Ntilde; | Ñ | &#209; | capital N, tilde | %D1 | \00d1 | \321 |
| &Ograve; | Ò | &#210; | capital O, grave accent | %D2 | \00d2 | \322 |
| &Oacute; | Ó | &#211; | capital O, acute accent | %D3 | \00d3 | \323 |
| &Ocirc; | Ô | &#212; | capital O, circumflex accent | %D4 | \00d4 | \324 |
| &Otilde; | Õ | &#213; | capital O, tilde | %D5 | \00d5 | \325 |
| &Ouml; | Ö | &#214; | capital O, dieresis or umlaut mark | %D6 | \00d6 | \326 |
| &times; | × | &#215; | multiply sign | %D7 | \00d7 | \327 |
| &Oslash; | Ø | &#216; | capital O, slash | %D8 | \00d8 | \330 |
| &Ugrave; | Ù | &#217; | capital U, grave accent | %D9 | \00d9 | \331 |
| &Uacute; | Ú | &#218; | capital U, acute accent | %DA | \00da | \332 |
| &Ucirc; | Û | &#219; | capital U, circumflex accent | %DB | \00db | \333 |
| &Uuml; | Ü | &#220; | capital U, dieresis or umlaut mark | %DC | \00dc | \334 |
| &Yacute; | Ý | &#221; | capital Y, acute accent | %DD | \00dd | \335 |
| &THORN; | Þ | &#222; | capital THORN, Icelandic | %DE | \00de | \336 |
| &szlig; | ß | &#223; | small sharp s, German (sz ligature) | %DF | \00df | \337 |
| &agrave; | à | &#224; | small a, grave accent | %E0 | \00e0 | \340 |
| &aacute; | á | &#225; | small a, acute accent | %E1 | \00e1 | \341 |
| &acirc; | â | &#226; | small a, circumflex accent | %E2 | \00e2 | \342 |
| &atilde; | ã | &#227; | small a, tilde | %E3 | \00e3 | \343 |
| &auml; | ä | &#228; | small a, dieresis or umlaut mark | %E4 | \00e4 | \344 |
| &aring; | å | &#229; | small a, ring | %E5 | \00e5 | \345 |
| &aelig; | æ | &#230; | small ae diphthong (ligature) | %E6 | \00e6 | \346 |
| &ccedil; | ç | &#231; | small c, cedilla | %E7 | \00e7 | \347 |
| &egrave; | è | &#232; | small e, grave accent | %E8 | \00e8 | \350 |
| &eacute; | é | &#233; | small e, acute accent | %E9 | \00e9 | \351 |
| &ecirc; | ê | &#234; | small e, circumflex accent | %EA | \00ea | \352 |
| &euml; | ë | &#235; | small e, dieresis or umlaut mark | %EB | \00eb | \353 |
| &igrave; | ì | &#236; | small i, grave accent | %EC | \00ec | \354 |
| &iacute; | í | &#237; | small i, acute accent | %ED | \00ed | \355 |
| &icirc; | î | &#238; | small i, circumflex accent | %EE | \00ee | \356 |
| &iuml; | ï | &#239; | small i, dieresis or umlaut mark | %EF | \00ef | \357 |
| &eth; | ð | &#240; | small eth, Icelandic | %F0 | \00f0 | \360 |
| &ntilde; | ñ | &#241; | small n, tilde | %F1 | \00f1 | \361 |
| &ograve; | ò | &#242; | small o, grave accent | %F2 | \00f2 | \362 |
| &oacute; | ó | &#243; | small o, acute accent | %F3 | \00f3 | \363 |
| &ocirc; | ô | &#244; | small o, circumflex accent | %F4 | \00f4 | \364 |
| &otilde; | õ | &#245; | small o, tilde | %F5 | \00f5 | \365 |
| &ouml; | ö | &#246; | small o, dieresis or umlaut mark | %F6 | \00f6 | \366 |
| &divide; | ÷ | &#247; | divide sign | %F7 | \00f7 | \367 |
| &oslash; | ø | &#248; | small o, slash | %F8 | \00f8 | \370 |
| &ugrave; | ù | &#249; | small u, grave accent | %F9 | \00f9 | \371 |
| &uacute; | ú | &#250; | small u, acute accent | %FA | \00fa | \372 |
| &ucirc; | û | &#251; | small u, circumflex accent | %FB | \00fb | \373 |
| &uuml; | ü | &#252; | small u, dieresis or umlaut mark | %FC | \00fc | \374 |
| &yacute; | ý | &#253; | small y, acute accent | %FD | \00fd | \375 |
| &thorn; | þ | &#254; | small thorn, Icelandic | %FE | \00fe | \376 |
| &yuml; | ÿ | &#255; | small y, dieresis or umlaut mark | %FF | \00ff | \377 |
| **Latin Extended** | | | | | | |
| &OElig; | Œ | &#338; | latin capital ligature oe | u+0152 ISOlat2 | \0152 | \u0152 |
| &oelig; | œ | &#339; | latin small ligature oe | u+0153 ISOlat2 | \0153 | \u0153 |
| &Scaron; | Š | &#352; | latin capital letter s with caron | u+0160 ISOlat2 | \0160 | \u0160 |
| &scaron; | š | &#353; | latin small letter s with caron | u+0161 ISOlat2 | \0161 | \u0161 |
| &Yuml; | Ÿ | &#376; | latin capital letter y with diaeresis | u+0178 ISOlat2 | \0178 | \u0178 |
| &fnof; | ƒ | &#402; | latin small f with hook | u+0192 ISOtech | \0192 | \u0192 |
| **Spacing Modifier Letters** | | | | | | |
| &circ; | ˆ |  | modifier letter circumflex accent | u+02C6 ISOpub | \02c6 | \u02c6 |
| &tilde; | ˜ |  | small tilde | u+02DC ISOdia | \02dc | \u02dc |
| **Greek** | | | | | | |
| &Alpha; | Α | &#913; | greek capital letter alpha | u+0391 | \0391 | \u0391 |
| &Beta; | Β | &#914; | greek capital letter beta | u+0392 | \0392 | \u0392 |
| &Gamma; | Γ | &#915; | greek capital letter gamma | u+0393 ISOgrk3 | \0393 | \u0395 |
| &Delta; | Δ | &#916; | greek capital letter delta | u+0394 ISOgrk3 | \0394 | \u0394 |
| &Epsilon; | Ε | &#917; | greek capital letter epsilon | u+0395 | \0395 | \u0395 |
| &Zeta; | Ζ | &#918; | greek capital letter zeta | u+0396 | \0396 | \u0396 |
| &Eta; | Η | &#919; | greek capital letter eta | u+0397 | \0397 | \u0397 |
| &Theta; | Θ | &#920; | greek capital letter theta | u+0398 ISOgrk3 | \0398 | \u0398 |
| &Iota; | Ι | &#921; | greek capital letter iota | u+0399 | \0399 | \u0399 |
| &Kappa; | Κ | &#922; | greek capital letter kappa | u+039A | \039a | \u039a |
| &Lambda; | Λ | &#923; | greek capital letter lambda | u+039B ISOgrk3 | \039b | \u039b |
| &Mu; | Μ | &#924; | greek capital letter mu | u+039C | \039c | \u039c |
| &Nu; | Ν | &#925; | greek capital letter nu | u+039D | \039d | \u039D |
| &Xi; | Ξ | &#926; | greek capital letter xi | u+039E ISOgrk3 | \039e | \u039e |
| &Omicron; | Ο | &#927; | greek capital letter omicron | u+039F | \039f | \u039f |
| &Pi; | Π | &#928; | greek capital letter pi | u+03A0 ISOgrk3 | \03a0 | \u03a0 |
| &Rho; | Ρ | &#929; | greek capital letter rho | u+03A1 | \03a1 | \u03a1 |
| &Sigma; | Σ | &#931; | greek capital letter sigma | u+03A3 ISOgrk3 | \03a3 | \u03A3 |
| &Tau; | Τ | &#932; | greek capital letter tau | u+03A4 | \03a4 | \u03A4 |
| &Upsilon; | Υ | &#933; | greek capital letter upsilon | u+03A5 ISOgrk3 | \03a5 | \u03A5 |
| &Phi; | Φ | &#934; | greek capital letter phi | u+03A6 ISOgrk3 | \03a6 | \u03A6 |
| &Chi; | Χ | &#935; | greek capital letter chi | u+03A7 | \03a7 | \u03A7 |
| &Psi; | Ψ | &#936; | greek capital letter psi | u+03A8 ISOgrk3 | \03a8 | \u03A8 |
| &Omega; | Ω | &#937; | greek capital letter omega | u+03A9 ISOgrk3 | \03a9 | \u03A9 |
| &alpha; | α | &#945; | greek small letter alpha | u+03B1 ISOgrk3 | \03b1 | \u03b1 |
| &beta; | β | &#946; | greek small letter beta | u+03B2 ISOgrk3 | \03b2 | \u03b2 |
| &gamma; | γ | &#947; | greek small letter gamma | u+03B3 ISOgrk3 | \03b3 | \u03b3 |
| &delta; | δ | &#948; | greek small letter delta | u+03B4 ISOgrk3 | \03b4 | \u03b4 |
| &epsilon; | ε | &#949; | greek small letter epsilon | u+03B5 ISOgrk3 | \03b5 | \u03b5 |
| &zeta; | ζ | &#950; | greek small letter zeta | u+03B6 ISOgrk3 | \03b6 | \u03b6 |
| &eta; | η | &#951; | greek small letter eta | u+03B7 ISOgrk3 | \03b7 | \u03b7 |
| &theta; | θ | &#952; | greek small letter theta | u+03B8 ISOgrk3 | \03b8 | \u03b8 |
| &iota; | ι | &#953; | greek small letter iota | u+03B9 ISOgrk3 | \03b9 | \u03b9 |
| &kappa; | κ | &#954; | greek small letter kappa | u+03BA ISOgrk3 | \03ba | \u03ba |
| &lambda; | λ | &#955; | greek small letter lambda | u+03BB ISOgrk3 | \03bb | \u03bb |
| &mu; | μ | &#956; | greek small letter mu | u+03BC ISOgrk3 | \03bc | \u03bc |
| &nu; | ν | &#957; | greek small letter nu | u+03BD ISOgrk3 | \03bd | \u03bd |
| &xi; | ξ | &#958; | greek small letter xi | u+03BE ISOgrk3 | \03be | \u03be |
| &omicron; | ο | &#959; | greek small letter omicron | u+03BF NEW | \03bf | \u03bf |
| &pi; | π | &#960; | greek small letter pi | u+03C0 ISOgrk3 | \03c0 | \u03c0 |
| &rho; | ρ | &#961; | greek small letter rho | u+03C1 ISOgrk3 | \03c1 | \u03c1 |
| &sigmaf; | ς | &#962; | greek small letter final sigma | u+03C2 ISOgrk3 | \03C2 | \u03c2 |
| &sigma; | σ | &#963; | greek small letter sigma | u+03C3 ISOgrk3 | \03C3 | \u03c3 |
| &tau; | τ | &#964; | greek small letter tau | u+03C4 ISOgrk3 | \03C4 | \u03c4 |
| &upsilon; | υ | &#965; | greek small letter upsilon | u+03C5 ISOgrk3 | \03C5 | \u03c5 |
| &phi; | φ | &#966; | greek small letter phi | u+03C6 ISOgrk3 | \03C6 | \03c6 |
| &chi; | χ | &#967; | greek small letter chi | u+03C7 ISOgrk3 | \03C7 | \u03c7 |
| &psi; | ψ | &#968; | greek small letter psi | u+03C8 ISOgrk3 | \03C8 | \u03c8 |
| &omega; | ω | &#969; | greek small letter omega | u+03C9 ISOgrk3 | \03C9 | \u03c9 |
| &thetasym; | ϑ | &#977; | greek small letter theta symbol | u+03D1 NEW | \03D1 | \u03D1 |
| &upsih; | ϒ | &#978; | greek upsilon with hook symbol | u+03D2 NEW | \03D2 | \u03D2 |
| &piv; | ϖ | &#982; | greek pi symbol | u+03D6 ISOgrk3 | \03D6 | \u03D6 |
| **Punctuation** | | | | | | |
| &ensp; |  | &#8194; | en space | u+2002 ISOpub | \2002 | \u2002 |
| &emsp; |  | &#8195; | em space | u+2003 ISOpub | \2003 | \u2003 |
| &thinsp; |  | &#8201; | thin space | u+2009 ISOpub | \2009 | \u2009 |
| &zwnj; | ‌ | &#8204; | zero width non-joiner | u+200C NEW RFC 2070 | \200C | \u200C |
| &zwj; | ‍ | &#8205; | zero width joiner | u+200D NEW RFC 2070 | \200D | \u200d |
| &lrm; | ‎ | &#8206; | left-to-right mark | u+200E NEW RFC 2070 | \200E | \u200e |
| &rlm; | ‏ | &#8207; | right-to-left mark | u+200F NEW RFC 2070 | \200F | \u200f |
| &ndash; | – | &#8211; | en dash | u+2013 ISOpub | \2013 | \u2013 |
| &mdash; | — | &#8212; | em dash | u+2014 ISOpub | \2014 | \u2014 |
| &lsquo; | ‘ | &#8216; | left single quotation mark | u+2018 ISOnum | \2018 | \u2018 |
| &rsquo; | ’ | &#8217; | right single quotation mark | u+2019 ISOnum | \2019 | \u2019 |
| &sbquo; | ‚ | &#8218; | single low-9 quotation mark | u+201A NEW | \201A | \u201a |
| &ldquo; | “ | &#8220; | left double quotation mark | u+201C ISOnum | \201C | \u201c |
| &rdquo; | ” | &#8221; | right double quotation mark | u+201D ISOnum | \201D | \u201d |
| &bdquo; | „ | &#8222; | double low-9 quotation mark | u+201E NEW | \201E | \u201e |
| &dagger; | † | &#8224; | dagger | u+2020 ISOpub | \2020 | \u2020 |
| &Dagger; | ‡ | &#8225; | double dagger | u+2021 ISOpub | \2021 | \u2021 |
| &permil; | ‰ | &#8240; | per mille sign | u+2030 ISOtech | \2030 | \u2030 |
| &lsaquo; | ‹ | &#8249; | single left-pointing angle quotation mark | u+2039 ISO proposed | \2039 | \u2039 |
| &rsaquo; | › | &#8250; | single right-pointing angle quotation mark | u+203A ISO proposed | \203A | \u203a |
| &bull; | • | &#8226; | bullet, a.k.a. black small circle bullet is NOT the same as bullet operator — u+2219 | u+2022 ISOpub | \2022 | \u2219 |
| &hellip; | … | &#8230; | horizontal ellipsis, a.k.a. three dot leader | u+2026 ISOpub | \2026 | \u2026 |
| &prime; | ′ | &#8242; | prime, a.k.a. minutes, a.k.a. feet | u+2032 ISOtech | \2032 | \u2032 |
| &Prime; | ″ | &#8243; | double prime, a.k.a. seconds, a.k.a. inches | u+2033 ISOtech | \2033 | \u2033 |
| &oline; | ‾ | &#8254; | overline, a.k.a. spacing overscore | u+203E NEW | \203E | \u203e |
| &frasl; | ⁄ | &#8260; | fraction slash | u+2044 NEW | \2044 | \u8260 |
| **Letter-like Symbols** | | | | | | |
| &weierp; | ℘ | &#8472; | script capital P, a.k.a. power set, a.k.a. Weierstrass p | u+2118 ISOamso | \2118 | \u2118 |
| &image; | ℑ | &#8465; | blackletter capital I, a.k.a. imaginary part | u+2111 ISOamso | \2111 | \u2111 |
| &real; | ℜ | &#8476; | blackletter capital R, a.k.a. real part symbol | u+211C ISOamso | \211C | \u211c |
| &trade; | ™ | &#8482; | trade mark sign | u+2122 ISOnum | \2122 | \u2122 |
| &alefsym; | ℵ | &#8501; | alef symbol | u+2135 NEW | \2135 | \u |
| **Arrows** | | | | | | |
| &larr; | ← | &#8592; | leftwards arrow | u+2190 ISOnum | \2190 | \u2190 |
| &uarr; | ↑ | &#8593; | upwards arrow | u+2191 ISOnum | \2191 | \u2191 |
| &rarr; | → | &#8594; | rightwards arrow | u+2192 ISOnum | \2192 | \u2192 |
| &darr; | ↓ | &#8595; | downwards arrow | u+2193 ISOnum | \2193 | \u2193 |
| &harr; | ↔ | &#8596; | left right arrow | u+2194 ISOamsa | \2194 | \u2194 |
| &crarr; | ↵ | &#8629; | downwards arrow with corner leftwards, a.k.a. carriage return | u+21B5 NEW | \21B5 | \u21b5 |
| &lArr; | ⇐ | &#8656; | leftwards double arrow can be used for ‘is implied by’ | u+21D0 ISOtech | \21D0 | \u21d0 |
| &uArr; | ⇑ | &#8657; | upwards double arrow | u+21D1 ISOamsa | \21D1 | \u21d1 |
| &rArr; | ⇒ | &#8658; | rightwards double arrow | u+21D2 ISOtech | \21D2 | \u21d2 |
| &dArr; | ⇓ | &#8659; | downwards double arrow | u+21D3 ISOamsa | \21D3 | \ud1d3 |
| &hArr; | ⇔ | &#8660; | left right double arrow | u+21D4 ISOamsa | \21D4 | \u21d4 |
|  | ◄ | &#9668; | black left pointing triangle | U+25C0 | \25C0 | \u25c0 |
|  | ► | &#9658; | black right pointing triangle | u+25B6 | \25b6 | \u25b6 |
|  | ▲ | &#9650; | black up pointing triangle | u+25B2 | \25b2 | \25b2 |
|  | ▼ | &#9660; | black down pointing triangle | u+25BC | \25bc | \25bc |
| **Mathematical Operators** | | | | | | |
| &forall; | ∀ | &#8704; | for all | u+2200 ISOtech | \2200 | \u2200 |
| &part; | ∂ | &#8706; | partial differential | u+2202 ISOtech | \2202 | \u2202 |
| &exist; | ∃ | &#8707; | there exists | u+2203 ISOtech | \2203 | \u2203 |
| &empty; | ∅ | &#8709; | empty set, a.k.a. null set, a.k.a. diameter | u+2205 ISOamso | \2205 | \u2205 |
| &nabla; | ∇ | &#8711; | nabla, a.k.a. backward difference | u+2207 ISOtech | \2207 | \u2207 |
| &isin; | ∈ | &#8712; | element of | u+2208 ISOtech | \2208 | \u2208 |
| &notin; | ∉ | &#8713; | not an element of | u+2209 ISOtech | \2209 | \u2209 |
| &ni; | ∋ | &#8715; | contains as member | u+220B ISOtech | \220B | \u220b |
| &prod; | ∏ | &#8719; | n-ary product, a.k.a. product sign | u+220F ISOamsb | \220F | \u03a0 |
| &sum; | ∑ | &#8721; | n-ary sumation | u+2211 ISOamsb | \2211 | \u03a3 |
| &minus; | − | &#8722; | minus sign | u+2212 ISOtech | \2212 | \u2212 |
| &lowast; | ∗ | &#8727; | asterisk operator | u+2217 ISOtech | \2217 | \u2217 |
| &radic; | √ | &#8730; | square root, a.k.a. radical sign | u+221A ISOtech | \221A | \u221a |
| &prop; | ∝ | &#8733; | proportional to | u+221D ISOtech | \221D | \u221d |
| &infin; | ∞ | &#8734; | infinity | u+221E ISOtech | \221E | \u221e |
| &ang; | ∠ | &#8736; | angle | u+2220 ISOamso | \2220 | \u2220 |
| &and; | ⊥ | &#8869; | logical and, a.k.a. wedge | u+2227 ISOtech | \2227 | \u2227 |
| &or; | ⊦ | &#8870; | logical or, a.k.a. vee | u+2228 ISOtech | \2228 | \u2228 |
| &cap; | ∩ | &#8745; | intersection, a.k.a. cap | u+2229 ISOtech | \2229 | \u2229 |
| &cup; | ∪ | &#8746; | union, a.k.a. cup | u+222A ISOtech | \222A | \u222a |
| &int; | ∫ | &#8747; | integral | u+222B ISOtech | \222B | \u222b |
| &there4; | ∴ | &#8756; | therefore | u+2234 ISOtech | \2234 | \u2234 |
| &sim; | ∼ | &#8764; | tilde | u+223C ISOtech | \223C | \u223c |
| &cong; | ≅ | &#8773; | approximately equal to | u+2245 ISOtech | \2245 | \u2245 |
| &asymp; | ≈ | &#8776; | almost equal to, a.k.a. asymptotic to | u+2248 ISOamsr | \2248 | \u2248 |
| &ne; | ≠ | &#8800; | not equal to | u+2260 ISOtech | \2260 | \u2260 |
| &equiv; | ≡ | &#8801; | identical to | u+2261 ISOtech | \2261 | \u2261 |
| &le; | ≤ | &#8804; | less-than or equal to | u+2264 ISOtech | \2264 | \u2264 |
| &ge; | ≥ | &#8805; | greater-than or equal to | u+2265 ISOtech | \2265 | \u2265 |
| &sub; | ⊂ | &#8834; | subset of | u+2282 ISOtech | \2282 | \u2282 |
| &sup; | ⊃ | &#8835; | superset of | u+2283 ISOtech | \2283 | \u2283 |
| &nsub; | ⊄ | &#8836; | not a subset of | u+2284 ISOamsn | \2284 | \u2284 |
| &sube; | ⊆ | &#8838; | subset of or equal to | u+2286 ISOtech | \2286 | \u2286 |
| &supe; | ⊇ | &#8839; | superset of or equal to | u+2287 ISOtech | \2287 | \u2287 |
| &oplus; | ⊕ | &#8853; | circled plus, a.k.a. direct sum | u+2295 ISOamsb | \2295 | \u2295 |
| &otimes; | ⊗ | &#8855; | circled times, a.k.a. vector product | u+2297 ISOamsb | \2297 | \u2297 |
| &perp; | ⊥ | &#8869; | up tack | u+22A5 ISOtech | \22A5 | \u22a5 |
| &sdot; | ⋅ | &#8901; | dot operator | u+22C5 ISOamsb | \22C5 | \u22c5 |
| **Technical** | | | | | | |
| &lceil; | ⌈ | &#8968; | left ceiling, a.k.a. apl upstile | u+2308 ISOamsc | \2308 | \u2308 |
| &rceil; | ⌉ | &#8969; | right ceiling | u+2309 ISOamsc | \2309 | \u2309 |
| &lfloor; | ⌊ | &#8970; | left floor, a.k.a. apl downstile | u+230A ISOamsc | \230A | \u230a |
| &rfloor; | ⌋ | &#8971; | right floor | u+230B ISOamsc | \230B | \u230b |
| &lang; | ⟨ | &#9001; | left-pointing angle bracket | u+2329 ISOtech | \2329 | \u2329 |
| &rang; | ⟩ | &#9002; | right-pointing angle bracket | u+232A ISOtech | \232A | \u232a |
| **Miscellaneous** | | | | | | |
| &loz; | ◊ | &#9674; | lozenge | u+25CA ISOpub | \25CA | \u25ca |
| &spades; | ♠ | &#9824; | black spade suit | u+2660 ISOpub | \2660 | \u2660 |
| &clubs; | ♣ | &#9827; | black club suit, a.k.a. shamrock | u+2663 ISOpub | \2663 | \u2663 |
| &hearts; | ♥ | &#9829; | black heart suit, a.k.a. valentine | u+2665 ISOpub | \2665 | \u2665 |
| &diams; | ♦ | &#9830; | black diamond suit | u+2666 ISOpub | \2666 | \u2666 |

## 2.6. HTML5

HTML5 là chuẩn mới nhất của HTML, nó có các đặc điểm:

* Cập nhật một số thành phần HTML
* Có các thành phần và thuộc tính HTML mới (các thành phần HTML5)
* Hỗ trợ đầy đủ CSS3 (chuẩn mới nhất của CSS)
* Có thành phần video và audio
* Hỗ trợ đồ họa 2D/3D
* Lưu trữ cục bộ (**local storage**)
* Cơ sở dữ liệu SQL cục bộ (**local SQL database**)

HTML5 được tạo ra để đáp ứng cho sự phát triển web mạnh mẽ hiện nay, hướng đến một khái niệm web mới – web ngữ nghĩa (sematic web). Nên trọng điểm cải tiến các thành phần HTML ở chuẩn mới này là tạo ra các thẻ mang ngữ nghĩa nhiều hơn để bố cục một tài liệu HTML một cách rõ ràng hơn.

Cập nhật khai báo

<!doctype>

Khai báo <!doctype> mà chúng ta đã được học ở bài trước là mặc định khai báo kiểu tài liệu là HTML5.



Ví dụ một số cách khai báo kiểu tài liệu HTML4.



<meta>

Thẻ <meta> mà ta được học ở bài trước cũng đã được định nghĩa theo cú pháp của HTML5 (đã lược bớt một số thuộc tính không cần thiết). Xem ảnh minh họa để thấy được sự gọn gàng trong việc khai báo thẻ meta ở chuẩn HTML5.



<script>

Tương tự như vậy, trong khai báo thẻ <script> ở HTML5 cũng đã được lược bớt thuộc tính **type** không cần thiết.



<link>

Một lần nữa, HTML5 đã lược bỏ thuộc tính **type** ở thẻ <link>.



Cập nhật ngữ nghĩa

Các thẻ định dạng văn bản: thẻ <i>, <b>, <em>, <strong> được bổ sung thêm một số ngữ nghĩa mới.

<i></i>

Trong HTML4, thẻ <i> (italic) là thành phần định kiểu chữ in nghiêng.

Trong HTML5, thẻ <i> thể hiện nội dung với “alternate voice” hoặc “mood”.

Một số ví dụ trường hợp sử dụng thẻ <i>:

* Tên loài (Taxonomic designation)
* Thuật ngữ kỹ thuật (Technical term)
* Cụm thành ngữ từ ngôn ngữ khác (Idiomatic phrase from another language)
* Phiên dịch (Transliteration)
* Một suy nghĩ (Thought)
* Tên tàu trong các văn bản phương Tây (Ship name in Western texts)

<b></b>

Trong HTML4, thẻ <b> (**bold**) là thành phần định kiểu chữ in đậm.

Trong HTML5, thẻ <b> thể hiện nội dung với “**stylistically offset**”.

Một số ví dụ trường hợp sử dụng thẻ <b>:

* Từ khóa trong phần tổng quan của tài liệu (Key words in a document abstract)
* Các tên sản phẩm trong một bài đánh giá (Product names in a review)
* Các chữ mang hành động trong các phần mềm tương tác định hướng văn bản (Actionable words in interactive text-driven software)
* Phần dẫn dắt bài viết (Article lead)

<em></em>

Trong HTML4, thẻ <em> (**emphasis**) mang nghĩa nhấn mạnh (emphasis).

Trong HTML5, thẻ <em> mang nghĩa mới là đặc biệt nhấn mạnh (stress emphasis).

<strong></strong>

Trong HTML4, thẻ <strong> mang nghĩa nhấn mạnh (strong emphasis).

Trong HTML5, thẻ <strong> mang nghĩa mới là rất quan trọng (strong importance).

*Các thành phần HTML5:*

Các thẻ gom nhóm:

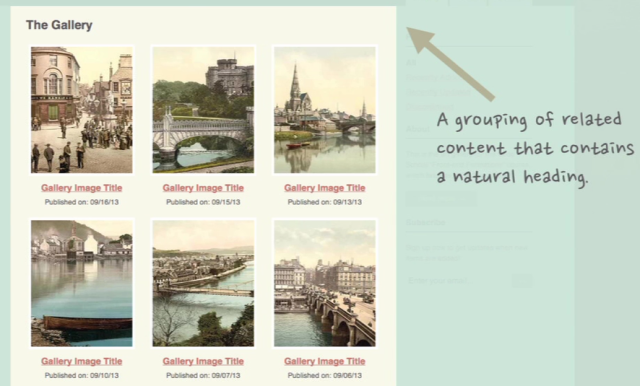
Các thẻ này có chức năng tương tự như thẻ <div> đã nói ở bài trước. Có thể nói những thẻ này là phiên bản nâng cấp của thẻ <div> bởi chúng thực chất là thẻ <div> mang ngữ nghĩa cụ thể giúp ta phân đoạn tài liệu HTML một cách rõ ràng hơn, tiện lợi và đầy ngữ nghĩa để ta dễ dàng hiểu được bố cục khi đọc tài liệu HTML.

<section></section>

Theo định nghĩa của W3C (World Wide Web Consortium) – tổ chức tiêu chuẩn quốc tế chính cho www, thành phần section thể hiện một phân đoạn chung trong một tài liệu hay một ứng dụng.

Hay nói cách khác, thẻ <section> là một thẻ <div> mang ngữ nghĩa, được dùng để nhóm **các nội dung liên quan** (theo chủ đề) lại với nhau.

Hình sau minh họa các nội dung có liên quan với nhau:



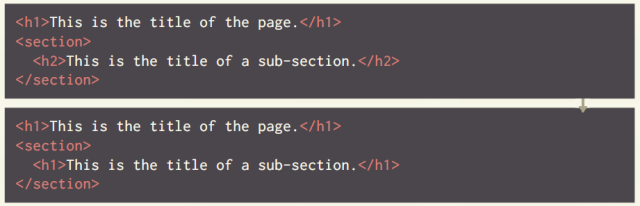
Để gom nhóm các nội dung này ta có thể làm như sau:



Các bạn có thể thấy với sự hỗ trợ của HTML5, việc phân đoạn tài liệu HTML rõ ràng, dễ hiểu hơn với thẻ <section>.

* *Để đảm bảo đúng ngữ nghĩa của thẻ <section>, khi sử dụng thẻ này ta cần cân nhắc xem các****nội dung có liên quan****với nhau không.*
* *Các thành phần HTML5 sau có outline riêng của mình:*
  + *Article*
  + *Aside*
  + *Nav*
  + *Section*

**Ví dụ:**



Cả hai cách khai báo tiêu đề <h1> và <h2> bên trong <section> đều cho ra kết quả như sau:

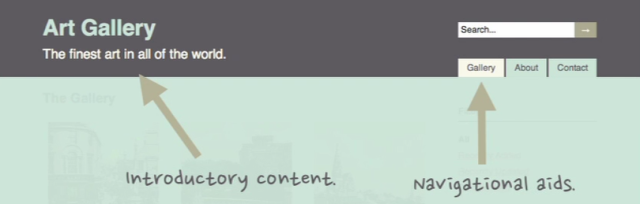
<header></header>

Theo định nghĩa W3C, thành phần header thể hiện một nhóm các hỗ trợ giới thiệu hoặc điều hướng.

Có thể có nhiều header khác nhau trong một trang.

Thường xuất hiện trên đầu tài liệu hoặc phân đoạn, tuy nhiên nó được xác định bởi nội dung của nó hơn là vị trí của nó.

Hình minh họa header của một trang web:



Tài liệu HTML định nghĩa đoạn header:



Ví dụ tài liệu HTML đoạn header trong section:

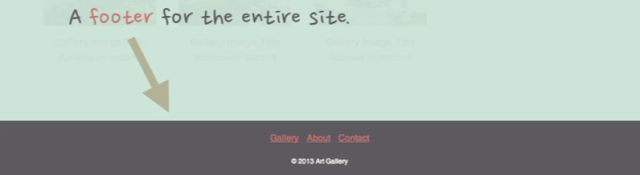


<footer></footer>

Theo định nghĩa của W3C, thành phần footer thể hiện cho một footer của nội dung phân đoạn hoặc thành phần phân đoạn gần nhất.

Giống như header, thành phần footer không phụ thuộc vào vị trí. Nó nên mô tả cho nội dung chưa nó.

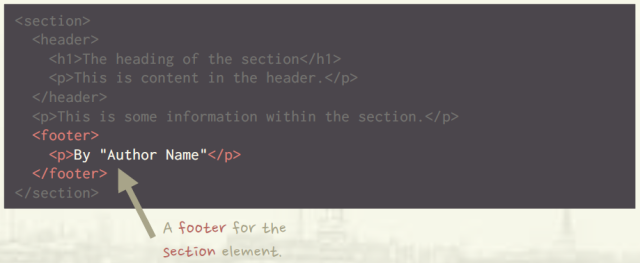
Hình minh họa footer của một trang web:



Tài liệu HTML định nghĩa đoạn footer:



Ví dụ tài liệu HTML đoạn footer trong section:



<aside></aside>

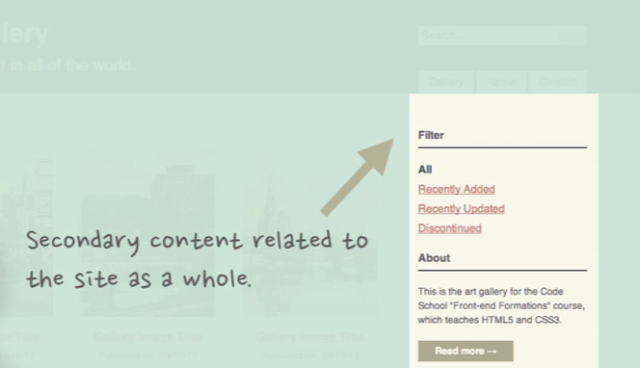
Ban đầu, HTML5 định nghĩa thành phần **aside** là “quan hệ tiếp giáp với các nội dung xung quanh mình”.

Bây giờ thành phần **aside** bao phủ nhiều ngữ cảnh:

* Khi được sử dụng bên trong thành phần article, nội dung của **aside** phải liên quan đến **article** đó.
* Khi được sử dụng bên ngoài thành phần **article**, nội dung aside phải đặc biệt liên quan đến trang web.

Một thành phần aside phù hợp khi nó được dùng để thể hiện nội dung không phải là trọng tâm chính của bài viết hay trang web, nhưng nội dung đó vẫn liên quan đến bài viết hay trang web đó.

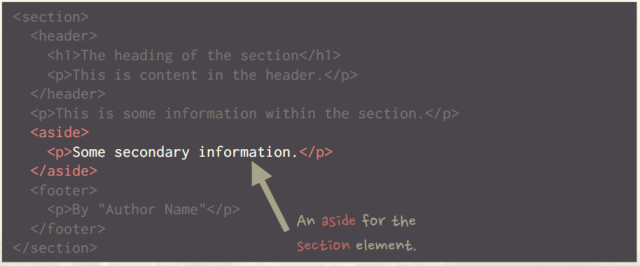
Hình minh họa aside của một trang web:



Tài liệu HTML định nghĩa đoạn aside:



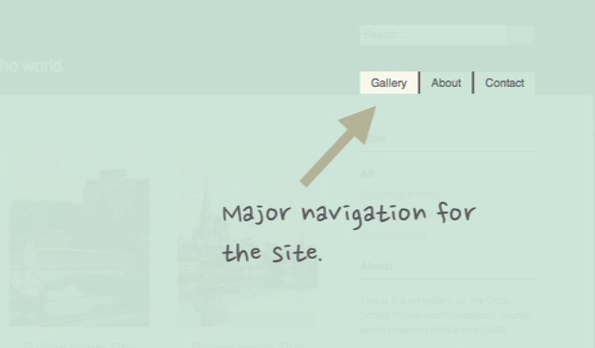
Ví dụ tài liệu HTML đoạn aside trong section:



<nav></nav>

Theo định nghĩa của W3C, thành phần **nav** (**major navigation**) thể hiện một đoạn của một trang mà nó liên kết đến những trang khác hoặc những phần khác trong trang đó: một phân đoạn với các liên kết điều hướng.

Hình minh họa navigation của một trang web:



Tài liệu HTML định nghĩa đoạn **nav**:



<article></article>

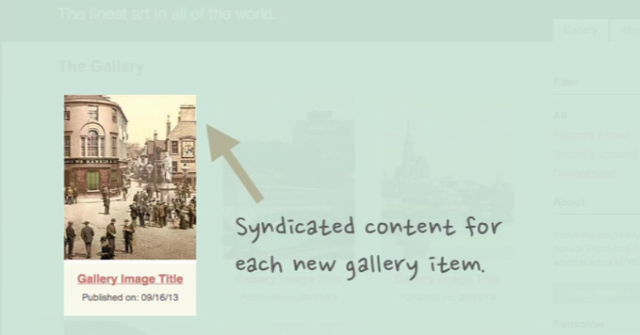
Theo định nghĩa của W3C, thành phần article thể hiện một thành phần hoàn chỉnh (hay khép kín) trong một tài liệu, trang web hay ứng dụng và theo nguyên tắc thì nó có thể được phân phối hoặc tái sử dụng một cách độc lập.

Thành phần article là một loại section. Nó được dùng cho nội dung hoàn chỉnh.

Một số ví dụ trường hợp sử dụng thẻ <article>:

* Một bài đăng trên blog
* Một mẩu tin tức
* Một bình luận trên một bài đăng
* Một bài đánh giá

Hình minh họa một article trong một trang web:



Tài liệu HTML định nghĩa đoạn article:



<main></main>

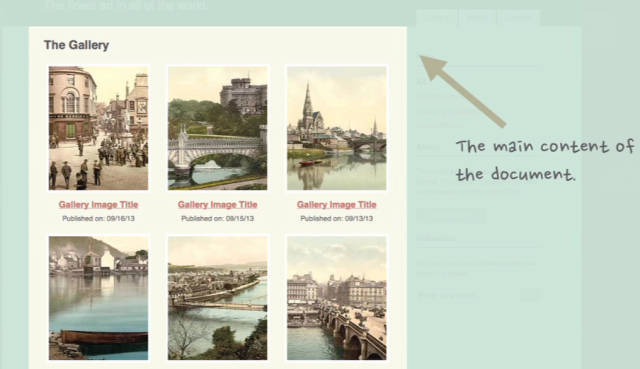
Theo định nghĩa của W3C, thành phần main thể hiện nội dung chính của phần thân của một tài liệu hoặc ứng dụng.

Và cũng theo W3C, vùng nội dung của main bao gồm các nội dung liên quan trực tiếp hoặc mở rộng dựa trên chủ đề chính của một tài liệu hoặc chức năng chính của một ứng dụng.

**Lưu ý:**

* *Không chứa nhiều hơn****một thành phần main****trong****một tài liệu***
* *Không chứa thành phần main trong một thành phần article, aside, footer, header, nav.*

Hình minh họa nội dung main trong một trang web:



Tài liệu HTML định nghĩa đoạn main:



<figure></figure>

Theo W3C, thành phần figure thể hiện một đơn vị của nội dung (có thể có caption) thường được tham chiếm như một đơn vị duy nhất từ mạch tài liệu, và có thể bỏ khỏi mạch tài liệu mà không làm ảnh hưởng đến nghĩa của tài liệu.

Thẻ <figure> thường được dùng cho một ảnh trong một article:



#### **3.2.1 <figcaption></figcaption>**

Theo W3C, thành phần figcaption (**figure caption**) thể hiện cho một caption hoặc legend của một figure.



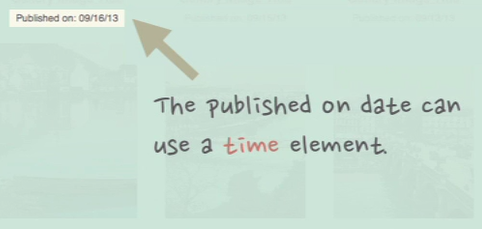
Hình ảnh minh họa của figcaption:



<time></time>

Theo W3C, thành phần time thể hiện hoặc thời gian trên đồng hồ, hoặc ngày tháng trong lịch (có thể có thêm thời gian và múi giờ).

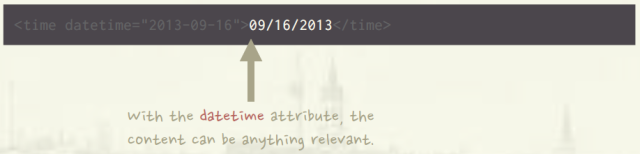
Hình ví dụ sử dụng thành phần time trong trang web:



Định nghĩa thẻ <time> trong tài liệu HTML:



Ta có thể sử dụng thuộc tính **datetime** để định dạng thời gian.



**Lưu ý:** Nếu không sử dụng thuộc tính datetime thì nội dung phải là ngày, giờ hợp lệ hoặc thời gian chính xác.

HTML5 Form:

HTML5 hỗ trợ việc tạo biểu mẫu mạnh mẽ hơn với các loại input mới, thành phần biểu mẫu mới và các thuộc tính mới.

Loại input mới

HTML5 cung cấp nhiều loại input mới:

* search
* email
* url
* tel
* number
* range
* date
* month
* week
* time
* datetime
* datetime-local
* color

**Lưu ý:** Các loại input này không được hỗ trợ đầy đủ ở các trình duyệt khác nhau. Nếu trình duyệt không hỗ trợ loại input thì nó sẽ chuyển về loại mặc định là **text**.

Để biết thêm thông tin về các loại input này và các trình duyệt hỗ trợ chúng, bạn có thể truy cập [tại đây](http://www.w3schools.com/html/html_form_input_types.asp).

Các thành phần mới của biểu mẫu:

HTML5 cung cấp các thành phần biểu mẫu mới:

* Datalist
* Keygen
* Output

Ở đây ta chỉ nói đến thành phần thường được sử dụng là datalist. Bạn có thể tham khảo các thành phần khác của biểu mẫu [tại đây](http://www.w3schools.com/html/html_form_elements.asp).

<datalist></datalist>

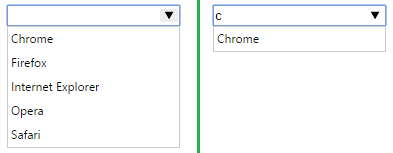
W3C định nghĩa thành phần datalist định nghĩa một tập các thành phần option thể hiện các lựa chọn.

Các lựa chọn được khai báo bằng thẻ **<option>** với giá trị là thuộc tính **value**.

<option> không có thẻ đóng.



Kết quả của đoạn HTML trên sẽ tạo một một input kiểu text có mũi tên ở góc phải, và khi ta nhấn vào thì danh sách các option sẽ hiện ra, hoặc khi ta nhập ký tự vào input thì browser sẽ tìm kiếm trong danh sách option các giá trị khớp với nội dung nhập để hiển thị lên.



Các thuộc tính mới của biểu mẫu

HTML5 cung cấp nhiều thuộc tính của biểu mẫu:

* placeholder
* autofocus
* autocomplete
* required
* pattern
* list
* multiple
* novalidate
* formnovalidate
* form
* formaction
* formenctype
* formmethod
* formtarget

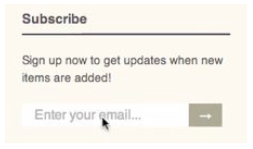
Ở bài này ta chỉ xem xét các thuộc tính placeholder, autofocus, required. Những thuộc tính còn lại bạn có thể tham khảo thêm [tại đây](http://www.w3schools.com/html/html_form_attributes.asp).

Placeholder:

Thuộc tính **placeholder** cho phép bạn xác định một thông điệp hiển thị bên trong trường input, thông điệp ẩn đi khi người dùng bắt đầu nhập, và hiện trở lại nếu trường input vẫn còn trống và bị mất focus (Để con nháy đi chỗ khác).



Kết quả hiển thị placeholder trên trình duyệt:

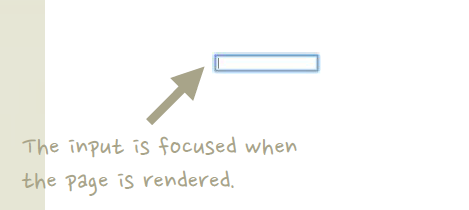


Autofocus

Thuộc tính **autofocus** sẽ giúp tự động focus (đưa con nháy) vào trường input xác định khi trang web được nạp lên trình duyệt.



Trường input được focus khi trang web được hiển thị:

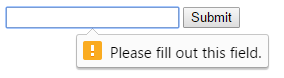


Required

Nếu bạn thêm thuộc tính **required** vào input, khi biểu mẫu được gửi đi (submit), người dùng sẽ nhận được thông báo lỗi nếu trường nhập còn trống.



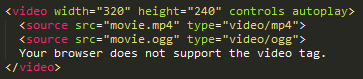
Ảnh thông báo khi submit trường input required còn trống (kết quả trên Chrome 49.0):



HTML5 Video

Trước khi có HTML5, không có một chuẩn nào để hiển thị video trên trang web, video chỉ có thể được hiển thị bằng plug-in (như flash).

Thẻ HTML5 **<video></video>** quy định một phương thức chuẩn để nhúng một video vào một trang web.



Thuộc tính **controls** thêm trình điều khiển video như phát, tạm dừng và âm lượng.

Thuộc tính **autoplay** giúp video tự động phát khi trang web được nạp lên trình duyệt.

Thuộc tính kích thước **width** và **height** nên được thêm vào để trình duyệt biết kích thước của video. Nếu không có 2 thuộc tính này thì khi video được nạp lên thì trang web sẽ bị thay đổi (hay flicker).

Đoạn văn bản nằm giữa cặp thẻ <video> </video> chỉ hiển thị trên trình duyệt không hỗ trợ thành phần video.

Thẻ **<source>** dùng để xác định file video nguồn “**src**” và định dạng video “**type**”.

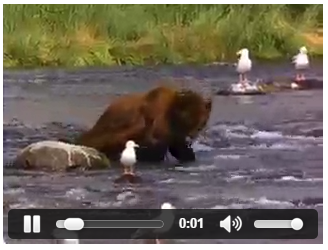
<source> không có thẻ đóng.

Nhiều thành phần source có thể liên kết đến nhiều tập tin video khác nhau. Trình duyệt sẽ sử dụng thẻ có định dạng video được nhận diện đầu tiên.

Hiện tại có 3 định dạng video được hỗ trợ: MP4, WebM và Ogg.

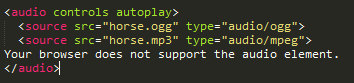


Kết quả hiển thị video trên trình duyệt:



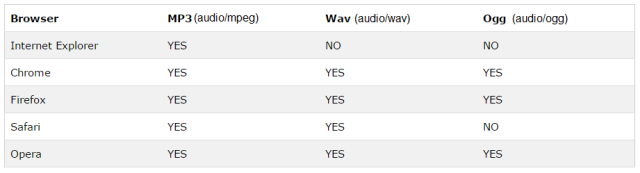
HTML5 Audio

Cũng giống như video, phương pháp chuẩn để nhúng audio vào trang web chỉ có từ HTML5, bằng cách dùng thẻ **<audio></audio>**.



Các thuộc tính và cách hoạt động của thành phần audio tương tự như video.

Hiện nay có 3 định dạng tập tin âm thanh được hỗ trợ là: MP3, Wav, Ogg.



Hiển thị trình điều khiển audio trên trình duyệt:



[H.T.T](https://taivublog.com/2016/03/27/bai-2-1-html5/)

2.7 Kết luận

HTML5 là một bước tiến quan trọng trong lịch sử của ngôn ngữ đánh dấu HTML, mang lại nhiều cải tiến và tính năng mới so với các phiên bản trước đó. Nó không chỉ cung cấp một cách tiếp cận linh hoạt và mạnh mẽ hơn cho việc xây dựng các ứng dụng web, mà còn đem lại trải nghiệm người dùng tốt hơn thông qua việc hỗ trợ đa phương tiện, tích hợp các tính năng địa lý và lưu trữ dữ liệu, cũng như cải thiện khả năng tương thích trên nhiều thiết bị và trình duyệt khác nhau. Với HTML5, nhà phát triển web có thể tạo ra các ứng dụng web phong phú và tương tác, đáp ứng được nhu cầu ngày càng đa dạng của người dùng trên toàn cầu.

**CHƯƠNG III: CSS và CSS3**

 (Cascading Style Sheets)

## 3.1. CSS là gì?

## 3.2. Cú pháp CSS

## 3.3. Áp dụng CSS vào trang HTML

## 3.4. Selectors

## 3.4.1. Universal selector

## 3.4.2. Type selector

## 3.4.3. Identity selector

## 3.4.4. Class selector

## 3.4.5. Descendant selector

## 3.4.6. Child selector

## 3.4.7. Adjacent sibling selector

## 3.4.8. Attribute selector

## 3.4.9. Pseudo class selector

## 3.4.10. Group selector

## 3.5. Đơn vị đo lường CSS

## 3.6. Kế thừa thuộc tính

## 3.7. Các nhóm thuộc tính trong CSS

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

- Thuộc tính định dạng nền (background)

- Mô hình hộp (box model)

- Các thuộc tính định margin, padding, border

- Thuộc tính định dạng cách hiển thị (display)

- Thuộc tính xác định vị trí (position)

## 3.8. Float & Clear

## 3.9. Flex

## 3.10. Grid

## 3.11. CSS3

- CSS3 là gì?

- CSS3 Selectors

- CSS3 Rounded Corners

- CSS3 Border Images

- CSS3 Backgrounds

- CSS Gradients

- CSS3 Shadows

- CSS3 Fonts

- CSS3 2D Transforms

- CSS3 3D Transforms

- CSS3 Transitions

- CSS3 Multiple Columns

- CSS3 Animations

## 3.12. SCSS

## 3.13. SASS

## 3.14 Kết luận

**CHƯƠNG 4: THIẾT KẾ WEBISTE THEO ĐỀ TÀI BẠN CHỌN**

4.1. Ý tưởng của Website

4.2. Xây dựng bố cục của trang Web

4.3. Thiết kế trang Web bằng HTML và CSS

4.5. Kết quả đạt được

Kết luận chương 4

**KẾT LUẬN**

**(Trình bày thành 3 đoạn văn nêu Ưu điểm, nhược điểm và hướng phát triển chủ đề)**

**DANH MỤC TÀI LIỆU THAM KHẢO**

1. Nguyễn Hồng Sơn (2007), Giáo trình hệ thống Mạng máy tính CCNA (Semester 1), NXB Lao động xã hội.
2. Phạm Quốc Hùng (2017), Đề cương bài giảng Mạng máy tính, Đại học SPKT Hưng Yên.
3. James F. Kurose and Keith W. Ross (2013), Computer Networking: A top-down approach sixth Edition, Pearson Education.

https://coccoc.com/search?query=internet+l%C3%A0+g%C3%AC

**QUY ĐỊNH TRÌNH BÀY TRONG BÁO CÁO CÀI TẬP LỚN**

- Bài tập lớn được in trên một mặt giấy trắng khổ A4 (210 x 297mm), dày lớn hơn 30 trang, nhỏ hơn 100 trang, không kể hình vẽ, bảng biểu, đồ thị và danh mục tài liệu tham khảo.

- Phần nội dung trình bày trong bài tập lớn sử dụng Font chữ **Times New Roman** cỡ **13**, hệ soạn thảo Microsoft Word; mật độ chữ bình thường, không được nén hoặc kéo dãn khoảng cách giữa các chữ; dãn dòng đặt ở chế độ **1,5 lines**; lề trên **2,0 cm**; lề dưới **2,0 cm**; lề trái **2,5 cm**, lề phải **2,0 cm**. Số trang được đánh ở giữa, phía dưới trang giấy.

- Cách ghi trích dẫn tài liệu tham khảo: Cuối đoạn trích dẫn đánh số thứ tự tài liệu tham khảo (ví dụ: [1]: tham khảo tài liệu số 1; [3,4,8]: tham khảo 3 tài liệu số 3, 4, 8).

- Tuyệt đối không được tẩy, xoá, sửa chữa trong bài tập lớn.

- Quy cách trình bày nội dung

|  |  |  |  |
| --- | --- | --- | --- |
| **Đề mục** | **Cỡ chữ** | **Định dạng** | **Canh lề trang** |
| Tên chương | 14 | In hoa, đậm | Giữa |
| Tên tiểu mục mức 1 | 13 | Chữ thường, đậm | Trái |
| Tên tiểu mục mức 2 | 13 | Chữ thường, đậm, nghiêng | Trái |
| Tên tiểu mục mức 3 | 13 | Đánh chỉ mục bằng chữ cái thường  a), ...  b), ..... | Trái |
| Nội dung | 13 | Chữ thường (Normal) | Đều hai bên |
| Nội dung bảng (table) | 12 | Normal | Giữa ô |
| Tên bảng | 12 | Chữ thường, nghiêng | Giữa, trên bảng |
| Tên hình | 12 | Chữ thường, nghiêng | Giữa, dưới hình |
| Tài liệu tham khảo | 12 | APA style | Chú thích bên dưới |

**Cách đánh dấu câu**:

Các dấu: : , . ; ) } ] ! ? ” được gõ ngay sau ký tự cuối cùng (không khoảng cách), và gõ 1 phím cách (space) sau chúng. Sau các dấu: “{ ( [ không gõ dấu cách.

**\* Cách đánh số các tiểu đề mục nhiều nhất là 3 mức và không lùi sang phải**

Hướng dẫn xếp tài liệu tham khảo

1. Tài liệu tham khảo xếp theo thứ tự trích dẫn trong bài tập lớn.
2. Tài liệu tham khảo là sách, luận án, báo cáo phải ghi đầy đủ các thông tin sau:

* Tên các tác giả hoặc cơ quan ban hành (không có dấu ngăn cách)
* (năm xuất bản), (đặt trong ngoặc đơn, dấu phẩy sau ngoặc đơn)
* Tên sách, luận án hoặc báo cáo, (in nghiêng, dấu phẩy cuối tên)
* Nhà xuất bản, (dấu phẩy cuối tên nhà xuất bản)
* Nơi xuất bản, (dấu chấm kết thúc tài liệu tham khảo).

Tài liệu tham khảo là bài báo trong tạp chí, bài trong một cuốn sách… ghi đầy đủ các thông tin sau:

* Tên các tác giả (không có dấu ngăn cách);
* (Năm công bố), (đặt trong ngoặc đơn, dấu phẩy sau ngoặc đơn)
* “Tên bài báo”, (đặt trong ngoặc kép, không in nghiên, dấu phẩy cuối tên)
* Tên tạp chí hoặc tên sách, (in nghiên, dấu phẩy cuối tên)
* Tập (không có dấu ngăn cách)
* (Sổ), (đặt trong ngoặc đơn, dấu phẩy sau ngoặc đơn)
* Các số trang, (gạch ngang giữa hai chữ số, dấu chấm kết thúc)

Cần chú ý những chi tiết về trình bày nêu trên. Nếu tài liệu dài hơn một dòng thì nên trình bày sau cho từ dòng thứ hai lùi vào so với dòng thứ nhất 1 cm để phần tài liệu tham khảo được rõ ràng và dễ theo dõi.

Ví dụ:

1. Nguyễn Hồng Sơn (2007), Giáo trình hệ thống Mạng máy tính CCNA (Semester 1), NXB Lao động xã hội.
2. Phạm Quốc Hùng (2017), Đề cương bài giảng Mạng máy tính, Đại học SPKT Hưng Yên.
3. James F. Kurose and Keith W. Ross (2013), Computer Networking: A top-down approach sixth Edition, Pearson Education.