TRƯỜNG ĐẠI HỌC ĐẠI NAM

**KHOA CÔNG NGHỆ THÔNG TIN**

──────── \* ───────

BÀI TẬP LỚN

**THIẾT KẾ WEB 1**

NGÀNH CÔNG NGHỆ THÔNG TIN

**TÊN ĐỀ TÀI**

Sinh viên thực hiện : **Trần Đình Khiêm**

Lớp: **CNTT14-01**

Giáo viên hướng dẫn: **Ths.Phạm Văn Tiệp**

**HÀ NỘI 06-2020**

**PHẦN 1: ĐẶT VẤN ĐỀ**

**1.1 Lý do chọn đề tài cơ sở thực tiễn của đề tài đã chọn**

Ngày nay, công nghệ thông tin không ngừng phát triển một cách mạnh  
mẽ và hiện đại. Các ứng dụng của công nghệ thông tin ngày càng đi sâu vào  
đời sống con người trở thành một bộ phận không thể thiếu của thế giới văn  
minh. Với xu thế toán càu hóa nền kinh tế thế giới, đặc biệt là nhu càu trao đổi  
hàng hóa của con người ngày càng tăng cả về số lượng và chất lượng, nhu cầu  
sử dụng Internet ngày càng nhiều và các hình thức kinh doanh trên mạng ngày  
càng đa dạng và trở thành một công cụ không thể thiếu. Vì vậy, nhiều cửa  
hàng bán thời trang nói chung và cửa hàng bán thời trang dành cho trẻ em nói riêng đã áp dụng phương pháp bán hàng qua mạng (thương mại điện tử) để khách hàng dễ dạng tìm kiếm thông tin và không cần tốn nhiều thời gian và chi phí.

Vì vậy em đã tạo ra một website có chức năng tương tự để giới thiệu và thời trang dành cho trẻ em, để tiệp cận được nhiều và phục vụ được đúng nhu cầu của khách hàng hơn.

**1.2 Phân tích yêu cầu của đề bài**

- Xây dựng một giao diện web giới thiệu & bán thời trang trẻ em.

- Giao diện website bao gồm phần giới thiệu & phần bán các sản phẩm

- Website hiển thị các sản phẩm đẹp, thân thiện.

- Sử dụng ngôn ngữ HTML, CSS để thiết kế.

**PHẦN 2: NỘI DUNG**

**2.1.Cơ sở lý thuyết**

**2.1.2.Ngôn ngữ HTML**

HTML là chữ viết tắt của Hypertext Markup Language. Nó giúp người dùng tạo và cấu trúc các thành phần trong trang web hoặc ứng dụng, phân chia các đoạn văn, heading, links, blockquotes…..

HTML không phải là ngôn ngữ lập trình, đồng nghĩa với việc nó không thể tạo ra các chức năng “động” được. Nó chỉ giống như Microsoft Word, dùng để bố cục và định dạng trang web.

**2.1.3.Ngôn ngữ CSS**

CSS là ngôn ngữ tạo phong cách cho trang web – Cascading Style Sheet language. Nó dùng để tạo phong cách và định kiểu cho những yếu tố được viết dưới dạng ngôn ngữ đánh dấu, như là [**HTML**](https://www.hostinger.vn/huong-dan/html-la-gi/). Nó có thể điều khiển định dạng của nhiều trang web cùng lúc để tiết kiệm công sức cho người viết web. Nó phân biệt cách hiển thị của trang web với nội dung chính của trang bằng cách điều khiển bố cục, màu sắc, và font chữ.

CSS được phát triển bởi W3C ([**World Wide Web Consortium**](https://www.w3.org/)) vào năm 1996, vì một lý do đơn giản. HTML không được thiết kế để gắn tag để giúp định dạng trang web. Bạn chỉ có thể dùng nó để “đánh dấu” lên site.

Mối tương quan giữa HTML và CSS rất mật thiết. HTML là ngôn ngữ markup (nền tảng của site) và CSS định hình phong cách (tất cả những gì tạo nên giao diện website), chúng là không thể tách rời.

CSS về lý thuyết không có cũng được, nhưng khi đó website sẽ không chỉ là một trang chứa văn bản mà không có gì khác.

**2.1.4.Kiến thức mới**

**a. Flexbox Layout**

**\*Flexbox Layout (hay còn gọi là Flexible Box):** là một kiểu bố cục trang có khả năng tự cân đối kích thước, thay đổi chiều rộng/chiều cao và thứ tự phần tử bên trong để phù hợp với tất cả các loại thiết bị hiển thị và kích thước màn hình.

Với bố cục thông thường, bạn cần phải thiết lập kích thước của phần tử, thiết lập hiển thị dạng block hay inline, cho nó float, còn với Flexbox bạn chỉ cần thiết lập phần hiển thị theo chiều ngang hay chiều dọc, lúc đó các phần tử bên trong có thể hiển thị theo ý muốn..

**\*Lưu ý:** ***Flexbox Layout*** phù hợp nhất để thiết lập bố cục ở quy mô nhỏ, còn thiết lập bố cục với phạm vi lớn hơn thì vẫn nên sử dụng kiểu thông thường là dàn trang theo dạng lưới (grid layout).

**\*Thành phần quan trọng nhất của Flexbox là**

**container:** là thành phần lớn bao quanh các phần tử bên trong, các item bên trong sẽ hiển thị dựa trên thiết lập của container này.

**item:** là phần tử con của container, bạn có thể thiết lập nó sẽ sử dụng bao nhiêu cột trong một container, hoặc thiết lập thứ tự hiển thị của nó.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Flex-container   |  |  |  |  |  | | --- | --- | --- | --- | --- | | Items | items | items | items | items |   Flex-container |

Các item sẽ được bố trí theo trục main axis (bắt đầu từ main-start, kết thúc ở main-end) hoặc theo trục **cross axis** (bắt đầu từ **cross-start**, kết thúc ở **cross-end**).

**main axis:** đây là trục chính để điều khiển hướng mà các item sẽ hiển thị. Lưu ý, main axis không phải lúc nào cũng nằm ngang như sơ đồ trên, bạn có thể sử dụng thuộc tính **flex-direction** để thay đổi hướng của trục và lúc đó các item sẽ hiển thị theo nó.

**main-start | main-end:** khi thiết lập flexbox, các item nằm trong container hiển thị từ điểm bắt đầu gọi là main-start tới điểm kết thúc gọi là main-end.

**main size:** kích thước (chiều rộng hoặc chiều cao) của các item, tùy thuộc vào hướng của main axis.

**cross axis:** cross axis luôn là trục vuông góc của main axis. Hướng của nó phụ thuộc vào hướng của main axis.

**cross-start | cross-end:** có ý nghĩa tương tự nhưng luôn vuông góc với main start, main end.

**cross size:** kích thước (chiều rộng hoặc chiều cao) của các item dựa trên trục cross axis, tùy thuộc vào hướng của main axis.

**\*Bảng các thuộc tính CSS sử dụng với Flexbox Layout**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Giá trị** | **Giải thích** |
| 1 | display | flex  inline-flex | Khai báo một trong 2 giá trị trên để sử dụng flex-box |
| 2 | flex-direction | row  row-reverse column  column-reverse | Xác định hướng của main-axis để container sắp xếp các item. **row**: mặc định, flex item được sắp xếp theo chiều ngang, từ trái qua phải (main axis nằm ngang).  **row-reverse:** flex item được sắp xếp theo chiều ngang, từ phải qua trái (main axis nằm ngang).  **column**: flex item được sắp xếp theo chiều dọc, từ trên xuống dưới (main axis đứng dọc).  **column-reverse:** flex item được sắp xếp theo chiều dọc, từ dưới lên trên (main axis đứng dọc). |
| 3 | flex-wrap | nowrap  wrap  wrap-reverse | Cho phép item tự động xuống dòng khi kích thước container thay đổi. **nowrap:** mặc định, tất cả các item sẽ nằm trhoên một dòng.  **wrap:** khi kích thước container thay đổi và tổng chiều rộng các item cộng lại lớn hơn chiều rộng của container thì item sẽ tự động xuống dòng.  **wrap-reverse:** tương tự như wrap, nhưng thay vì xuống dòng thì item sẽ tự động nhảy lên trên. |
| 4 | flex-flow | flex-flow: <flex-direction> <flex-wrap> | Sử dụng để gộp chung hai thuộc tính flex-direction và flex-wrap. |
| 5 | justify-content | flex-start  flex-end  center  space-between  space-around  space-evenly | **flex-start:** giá trị mặc định, item sẽ bắt đầu từ lề chính main-start của container.  **flex-end:** item sẽ bắt đầu từ lề chính main-end của container (khác với row-reverse là đổi hướng hiển thị).  **center:** item sẽ nằm giữa container.  **space-between:** các item sẽ có khoảng cách giữa các phần tử bằng nhau do container sẽ tự động căn khoảng cách, item đầu tiên sát lề chứa điểm main-start, item cuối cùng sát lề chứa điểm main-end.  **space-around:** tương tự space-between, nhưng khác ở chỗ là mỗi item có khoảng cách 2 bên cạnh và những khoảng cách này bằng nhau.  **space-evenly:** các item được phân phối sao cho khoảng cách giữa hai item bất kỳ, giữa item và các lề là bằng nhau. |
| 6 | align-items | stretch  flex-start  flex-end  center  baseline | Sử dụng để điều chỉnh vị trí bắt đầu và căn chỉnh các item bên trong container theo dọc theo trục cross axis, chiều ngang hoặc chiều dọc tùy thuộc vào flex-direction. **stretch:** giá trị mặc định, các phần tử sẽ được kéo dài để lấp đầy container chứa nó, nhưng sẽ ưu tiên giá trị height/width nếu có, khi đó item sẽ không cao full mà chỉ lấy giá trị height/width mà bạn set. **baseline:** item được căn chỉnh theo đường cơ sở của chúng.  Đường cơ sở là đường mà tất cả các chữ cái sẽ "ngồi lên". Bạn có thể sử dụng kích thước font chữ khác nhau để thấy rằng các mục được căn chỉnh theo đường cơ sở baseline |
| Và còn các thuộc tính khác như order,  flex-basis, flex-shrink, flex-grow…(tìm từ khoá tương ứng trên google để biết thêm thông tin) | | | |

**2.2.Cấu trúc Website**

**2.2.1.**