**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC ĐẠI NAM**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

**TÊN HỌC PHẦN : Thiết kế web**

**ĐỀ TÀI: Thiết kế web – Bài số 10**

**Giáo viên hướng dẫn: Hoàng Xuân Phương**

**Sinh viên thực hiện:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Stt** | **Mã sv** | **Họ và tên** | **Lớp** |
|  | **1671040014** | **Đỗ Lê Mạnh Hùng** | **KHMT 16-01** |
|  | **1671040017** | **Phạm Nam Khánh** | **KHMT 16-01** |
|  | **1671040013** | **Nguyễn Văn Huy** | **KHMT 16-01** |

**Hà Nội, năm 2023**

Mục Lục

[CHƯƠNG 1 - Cơ sở lí thuyết về HTML Và CSS 3](#_Toc136601560)

[1. Tổng quan về HTML 3](#_Toc136601561)

[1.1. Lịch sử về HTML 3](#_Toc136601562)

[1.2. Tác dụng của HTML 3](#_Toc136601563)

[1.3. Đặc điểm của HTMl 4](#_Toc136601564)

[1.4. Cấu trúc của HTML 4](#_Toc136601565)

[1.5. Một số thẻ HTML 6](#_Toc136601566)

[2. Tổng quan về CSS 9](#_Toc136601567)

[2.1. Lịch sử của CSS 9](#_Toc136601568)

[2.2. Bố cục và cấu trúc của CSS 10](#_Toc136601569)

[2.3. Ưu điểm của CSS 12](#_Toc136601570)

[3. Công cụ xây dụng trang (sử dụng Visual Studio Code) 13](#_Toc136601571)

[**CHƯƠNG 2 – PHÂN TÍCH, THIẾT KẾ TRANG WEB BẰNG HTML, CSS, JS** 15](#_Toc136601572)

[4. Ý tưởng thiết kế web 15](#_Toc136601573)

[5. Các bước triển khai thiết kế web 16](#_Toc136601574)

[5.1. Thanh topbar 17](#_Toc136601575)

[5.2. Logo 18](#_Toc136601576)

[5.3. Thanh Menu 22](#_Toc136601577)

[5.4. Slider 24](#_Toc136601578)

[5.5. A 28](#_Toc136601579)

[5.6. Footer 31](#_Toc136601580)

[CHƯƠNG 3 – KẾT QUẢ ĐẠT ĐƯỢC 34](#_Toc136601581)

[6. Topbar 34](#_Toc136601582)

[7. Menu 34](#_Toc136601583)

[8. Slider 34](#_Toc136601584)

[9. A 34](#_Toc136601585)

[10. Footer 34](#_Toc136601586)

# CHƯƠNG 1 - Cơ sở lí thuyết về HTML Và CSS

## Tổng quan về HTML

### Lịch sử về HTML

#### **HTML** được tạo ra bởi Tim Berners-Lee, một nhà vật lý học của trung tâm nghiên cứu CERN ở Thụy Sĩ. Hiện nay, HTML đã trở thành một chuẩn Internet được tổ chức W3C (World Wide Web Consortium) vận hành và phát triển. Bạn có thể tự tìm kiếm tình trạng mới nhất của HTML tại bất kỳ thời điểm nào trên Website của W3C.

#### Phiên bản đầu tiên của **HTML** xuất hiện năm 1991, gồm 18 tag HTML. Phiên bản HTML 4.01 được xuất bản năm 1999. Sau đó, các nhà phát triển đã thay thế HTML bằng XHTML vào năm 2000.

#### Đến năm 2014, HTML được nâng cấp lên chuẩn HTML5 với nhiều tag được thêm vào markup, mục đích là để xác định rõ nội dung thuộc loại là gì (ví dụ như: <article>, <header>, <[footer](https://wiki.matbao.net/footer/)>,…).

#### **Theo Mozilla Developer Network** thì **HTML Element Reference** hiện nay có khoảng **hơn 140 tag**. Tuy nhiên một vài tag trong số đó đã bị tạm ngưng (do không được hỗ trợ bởi các trình duyệt hiện hành).

### Tác dụng của HTML

#### **HTML** là một loại ngôn ngữ đánh dấu siêu văn bản. Như tôi đã đề cập ở trên, nó giúp cấu thành các cấu trúc cơ bản của một Website, làm cho trang Web trở thành một hệ thống hoàn chỉnh. Cụ thể, ngôn ngữ đánh dấu siêu văn bản này giúp bố cục, chia khung sườn các thành phần trang Web. Đồng thời, nó còn hỗ trợ khai báo các File kỹ thuật số như nhạc, Video, hình ảnh,…

#### Nếu muốn Website có cấu trúc tốt, sử dụng nhiều loại yếu tố trong văn bản, bạn sẽ cần đến HTML. Theo quan điểm của tôi, lập trình viên có thể lựa chọn ngôn ngữ lập trình riêng cho Website tùy vào mục đích sử dụng. Nhưng HTML thực chất chứa những yếu tố cần thiết cho mọi thể loại Website. Trang Web của bạn sẽ cần đến ngôn ngữ HTML để hiển thị nội dung cho người truy cập. Điều này đúng dù trang của bạn xây dựng trên bất kỳ nền tảng nào, giao tiếp với bất kỳ ngôn ngữ lập trình nào để xử lý dữ liệu.

#### Theo tôi được biết, trước khi bắt tay vào thiết kế, làm việc trên một trang Web thì nhà phát triển Web, lập trình viên đều phải học HTML như một loại ngôn ngữ cơ bản.

### Đặc điểm của HTML

#### HTML có nhiều thẻ định dạng, do đó bạn có thể trình bày trang Web dễ dàng, hiệu quả với ngôn ngữ đánh dấu siêu văn bản này. Với tôi, nó là một ngôn ngữ đánh dấu dễ dàng và đơn giản để sử dụng. Chúng ta có thể sử dụng nó để thiết kế trang Web cùng với văn bản một cách linh hoạt.

#### Một đặc điểm theo tôi khá thú vị là HTML có thể liên kết đến các trang Web khác. Nhờ ngôn ngữ đánh dấu này, bạn có thể thêm các Video, hình ảnh, âm thanh vào để các Website hấp dẫn, đẹp mắt và dễ tương tác hơn.

#### Đặc biệt, HTML có thể hiển thị trên bất kỳ nền tảng nào khác như Linux, Windows, và Max vì nó là một nền tảng độc lập.

### Cấu trúc của HTML

<!DOCTYPE html>  
<html>  
<head>  
<title>Tiêu đề trang web</title>  
</head>  
  
<body>

<h1>Heading của trang web</h1>

<p>Nội dung văn bản</p>

</body>

* ***<!DOCTYPE html>****: khai báo kiểu dữ liệu hiển thị*
* ***<html>****và****</html>****: cặp thẻ bắt buộc, element cấp cao nhất, có nhiệm vụ đóng gói tất cả nội dung của trang HTML*
* **<head>** và **</head>**: khai báo các thông tin meta của trang web như: tiêu đề trang, charset
* **<title>** và **</title>**: cặp thẻ nằm bên trong thẻ <head>, dùng để khai báo tiêu đề của trang
* **<body>** và **</body>**: cặp thẻ dùng để đóng gói tất cả các nội dung sẽ hiển thị trên trang
* **<h1></h1>, <h2></h2>**: định dạng dữ liệu dạng heading. Thông thường có 6 cấp độ heading trong HTML, trải dài từ <h1> tới <h6>. Trong đó, <h1> là cấp độ heading cao nhất và <h6> là cấp độ heading thấp nhất.
* **<p>** và **</p>**: cặp thẻ chứa các đoạn văn bản của trang web

### Một số thẻ HTML

#### I. Các thẻ HTML thông dụng

#### 1. Thẻ html

<html>: thẻ mở đầu của văn bản HTML.

#### 2. Thẻ head

<head></head>: dùng để chứa những thẻ cung cấp thông tin cho trang web. *Trong chuẩn HTML5, thẻ head cũng có thể được bỏ qua. Theo mặc định, trình duyệt sẽ thêm tất cả các phần tử trước thẻ <body>, đặt mặc định vào phần tử <head>.*

#### 3. Thẻ meta

<meta>: mô tả nội dung trang web

#### 4. Thẻ link

<link>: dùng để nhúng 1 tập tin vào trang web như css ...

#### 5. Thẻ script

<script></script>: thẻ dùng để nhúng tập tin javascript, js.

#### 6. Thẻ style

<style></style>: dùng để bao nội dung css.

#### 7. Thẻ body

<body></body>: thẻ chứa nội dung chính của trang web.

#### 8. Thẻ h1

<h1></h1>: thẻ hiển thị tiêu đề nội dung của bài viết nào đó.

#### 9. Thẻ div

<div></div>: thẻ dùng để chứa nhiều phần tử thường dùng để tạo bố cục cho trang web giúp dể quản lý và code hơn.

#### 10. Thẻ span

<span></span>: thẻ chứa nội dung.

#### 11. Thẻ p

<p></p>: thẻ chứa nội dung thường là 1 đoạn văn bản.

#### 12. Thẻ a

<a></a>: thẻ tạo link.

#### 13. Thẻ ul

<ul></ul>: thẻ mô tả liệt kê theo dạng danh sách.

#### 14. Thẻ img

<img>: thẻ dùng để hiển thị hình ảnh.

#### 15. Thẻ form

<form>: thẻ form kết hợp cùng các thẻ input, legend, fieldset, label dùng để tạo form nhập liệu cho trang web.

* <legend>: thẻ tiêu đề của nhóm các ô nhập liệu
* <fieldset>: thẻ dùng để nhóm các ô nhập liệu
* <label>: tiêu đề của ô nhập liệu
* <input>: thẻ nhập liệu

#### 16. Thẻ br

<br> thẻ xuống hàng

#### 17. Thẻ hr

<hr>: thẻ tạo đường kẻ ngang

#### 18. Thẻ table

<table></table>: thẻ table kết hợp với các thẻ tr, th, td dùng để tạo bảng.

* <tr></tr>: tạo hàng trong bảng
* <th></th>: tạo hàng đầu tiên của bảng
* <td></td>: tạo cột trong hàng của bảng

#### 19 .Thẻ iframe

<iframe></iframe>: thẻ nhúng một nội dung nào đó vào trang web hiện tại.

#### 20. Thẻ b

<b></b> thẻ in đậm nội dung văn bản

#### 21. Thẻ strong

<strong></strong>: thẻ in đậm với mục đích nhấn mạnh ý nghĩa của văn bản hơn

#### 22. Thẻ i

<i></i> thẻ tạo chữ in nghiêng

#### 23. Thẻ u

<u></u>: thẻ tạo chữ gạch dưới

#### 24. Thẻ s

<s></s>: thẻ tạo chữ gạch cắt ngang

#### 25. Thẻ sub và sup

<sub></sub>: thẻ tạo kiểu chữ

#### 26. Thẻ blockquote

<blockquote></blockquote>: thẻ mô tả một đoạn trích dẫn.

#### 27. Thẻ pre

<pre></pre>: thẻ giữ lại định dạng nội dung bên trong văn bản.

## Tổng quan về CSS

### Lịch sử của CSS

#### CSS được đề xuất lần đầu tiên vào ngày 10/10/1994 bởi Håkon Wium Lie. Kể từ đó, các phiên bản của CSS dần được hình thành qua nhiều giai đoạn. Tính từ lúc xuất hiện đến nay, CSS đã có nhiều phiên bản khác nhau. Các phiên bản mới sẽ giúp vá các lỗi của phiên bản cũ và mang đến nhiều cải tiến hơn.

#### Với phiên bản đầu tiên hay còn được gọi là CSS cấp 1 có những đặc điểm cụ thể như: thuộc tính font chữ, màu văn bản, hình nền, các thuộc tính văn bản, căn lề, định vị cho các yếu tố, nhận dạng duy nhất và phân loại chung các nhóm thuộc tính.

#### Phiên bản CSS cấp 2 được W3C phát triển vào tháng 5 năm 1998. Với những cải tiến từ phiên bản CSS cấp 1 và mang đến những cải tiến mới như định vị tuyệt đối, tương đối và cố định các yếu tố chỉ mục z. Khái niệm về các loại phương tiện, hỗ trợ cho các biểu định kiểu âm thanh và văn bản hai chiều. Xuất hiện các kiểu font chữ mới để định dạng văn bản.

#### Ngoài ra, sau phiên bản cấp 2 còn có một sự nâng cấp khác là CSS 2.1 được đề xuất vào tháng 4 năm 2011. Nhằm mục đích sửa lỗi và loại bỏ những tính năng kém hoặc không tương thích cho người dùng.

### Bố cục và cấu trúc của CSS

#### Bố cục của một đoạn CSS thường dựa vào hình hộp, trong đó, mỗi hộp đều có những khoảng trống trên trang với các thuộc tính:

#### Padding: không gian xung quanh nội dung như đoạn văn bản trong một đoạn CSS.

#### Border: đường liền nằm ở ngay bên ngoài của phần đệm CSS.

#### Margin: khoảng cách xung quanh nằm ở ngoài phần tử.

#### Về cấu trúc, một đoạn CSS bao gồm các phần như sau:

vùng chọn {

thuộc tính: giá trị;

thuộc tính: giá trị;

…. }

VD:

<style>

p {

color: blue;

text-weight: bold;

}

<style>

#### Cấu trúc này được khai báo bằng vùng chọn, các giá trị và thuộc tính nằm trong dấu ngoặc nhọn. Mỗi thuộc tính sẽ mang một giá trị riêng ở dạng số hoặc tên gọi trong danh sách có sẵn của CSS. Các thuộc tính và giá trị được cách nhau bằng dấu hai chấm, mỗi dòng khai báo sẽ có dấu phẩy ở cuối. Một vùng chọn không bị giới hạn về thuộc tính.

Trong đó:

#### Selector (bộ chọn): mẫu để chọn phần tử HTML mà người dùng cần định nghĩa phong cách. Thông thường, các selector được sử dụng cho các trường hợp:

#### Toàn bộ phần tử theo một dạng cụ thể nào đó như phần tử tiêu đề H1.

#### Thuộc tính ID và Class của các phần tử.

#### Các phần tử dựa vào mối liên quan với phần tử trong cây phân cấp tài liệu.

#### Declaration (khai báo): chứa một hay nhiều khai báo, phân tách với nhau bằng dấu phẩy. Mỗi khai báo gồm có tên và giá trị đặc tính CSS nằm trong các dấu ngoặc móc và được kết thúc bằng dấu chấm phẩy. Ví dụ: p { color: red; text-align: center;}.

#### Properties (thuộc tính): các cách mà bạn có thể sử dụng để tạo kiểu cho một phần tử HTML, trong đó, color có thể được xem là một thuộc tính của phần tử p. Do đó, với CSS, bạn chỉ cần lựa chọn thuộc tính mà chính bạn muốn tác động nhất trong bộ quy tắc của mình.

#### Giá trị thuộc tính: nằm ở phía bên phải của thuộc tính, sau dấu hai chấm và việc lựa chọn trong số đó phụ thuộc vào số lần xuất hiện của thuộc tính.

#### Muốn xem toàn bộ thuộc tính của CSS, bạn có thể xem trong CSS Reference của Mozilla.

### Ưu điểm của CSS

#### Sự khác biệt giữa site có CSS và không có CSSrất dễ nhận biết.

#### Chắc bạn đã thấy rồi, khi website không load được đầy đủ và chỉ có nền trắng và chữ thì chỉ có màu đen và xanh.

#### Điều này có nghĩa là thành phần CSS của bạn không tải lên được hay nó không tồn tại.

#### Đó một website không có CSS, vậy bạn có muốn website của mình trông như thế không? Hẵn là không rồi phải không.

#### Trước khi sử dụng CSS, tất cả những phong cách của CSS cần được đính kèm vào trong HTML markup. Có nghĩa là bạn cần tách ra để xác định các thành phần như background, font colors, canh hàng, vâng vâng.

#### CSS giúp bnạ định kiểu mọi thứ trên một file khác, bạn có thể tạo phong cách trước rồi sau đó tích hợp file CSS lên trên cùng của file HTML. Việc này giúp HTML markup rõ ràng và dễ quản lý hơn nhiều.

#### Tóm lại,với CSS bạn không cần lặp lại các mô tả cho từng thanh phần. Nó tiết kiệm thời gian, làm code ngắn lại để bạn có thể kiểm soát lỗi dễ dàng hơn.

#### CSS giúp bạn có nhiều styles trên một trang web HTML, vì vậy, khả năng điều chỉnh trang gần như vô hạn?

## Công cụ xây dụng trang web (sử dụng Visual Studio Code)

#### Visual Studio Code (viết tắt là VSCode) là một code editor hiện đại và vô cùng mạnh mẽ, được rất nhiều lập trình viên tin dùng. Ngoài khả năng tương thích với mọi hệ điều hành PC hiện nay, VSCode còn nổi tiếng với khả năng mở rộng cực kỳ linh hoạt nhờ vào hệ thống extension của mình.

#### Visual Studio Code đã hỗ trợ sẵn một số tính năng đối với HTML, CSS và Javascript. Sử dụng những extension để hỗ trợ cho việc thiết kế web. Các mục tiêu hướng đến sẽ là:

1. **Hỗ trợ HTML, CSS, JavaScript:**[Code snippet](https://en.wikipedia.org/wiki/Syntax_highlighting), [IntelliSense](https://docs.microsoft.com/en-us/visualstudio/ide/using-intellisense?view=vs-2019)
2. **LiveServer:** Biến máy tính thành web server nhanh gọn. Xem kết quả trang web nhanh trong trình duyệt hoặc các thiết bị trong mạng LAN.
3. **Emmet:** Code HTML nhanh như một vị thần. Không cần quan tâm đến đóng mở tags, tạo nhiều thẻ (div, span) một lúc, v.v…

#### Visual Studio Code đã hỗ trợ sẵn cho HTML, CSS và JavaScript

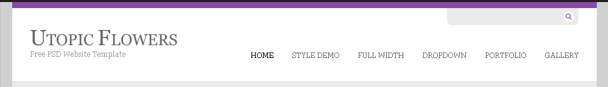
Một trong các tính năng của VSCode hấp dẫn các Web Developer đó là hỗ trợ các ngôn ngữ HTML, CSS và JS mặc định mà không cần phải cài thêm thứ gì. Các tính năng cơ bản có thể nói đến như là: Workspace, Xyntax Highlighting, Language Intellisense, Code Formatting, Emmet,…

* **Workspace:** Là một vùng làm việc mà ta có thể tùy chỉnh những thiết lập đặc biệt, phù hợp với project và ngôn ngữ đang sử dụng. Đơn giản thì nó là một folder chứa code và tài nguyên cho project (ảnh, css, js,…).
* **Xyntax Highlighting:** Làm nổi cú pháp, hay đơn giản là làm code của bạn màu mè dựa theo các từ khóa. Giúp dễ phân biệt các thành phần trong một cục code rối rắm.
* **Language Intellisense:** VSCode sẽ gợi ý các từ khóa phù hợp khi bạn gõ code. Tính năng này giúp ta viết code nhanh và đỡ cực khổ hơn.
* **Code Formatting:** Làm cho code của bạn gọn, đẹp, đúng chuẩn hơn. Đặc biết tính năng này hỗ trợ hầu hết các ngôn ngữ hiện nay. Cách Format code thì các bạn xem hình này nè.

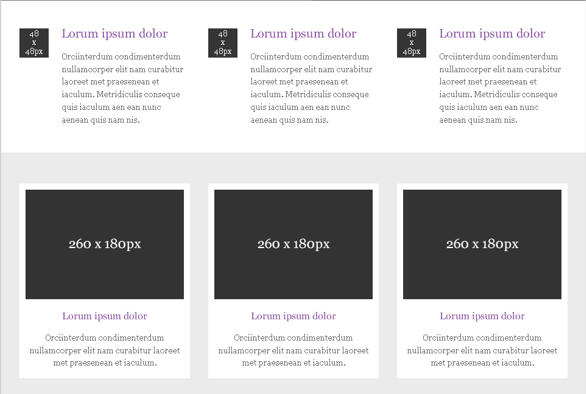
# **CHƯƠNG 2 – PHÂN TÍCH, THIẾT KẾ TRANG WEB BẰNG HTML; CSS; JS**

## Ý tưởng:

* B1: Chia trang web ra thành cấc phần nhỏ:
  + Ở đây trang web đã được chia thành 5 phần gồm :
    - Logo và Navbar



* + - Phần thân 1
    - Slider
    - Phần thân



* + - Footer



* B2: Thực hiện từng phần theo thứ tự từ trên xuống :
  + Làm theo thứ tự đã chia ở trên.
  + Sử dụng công cụ Adobe Photoshop để căn chỉnh đo đạc cho từng phần.

## Các bước triển khai thiết kế web

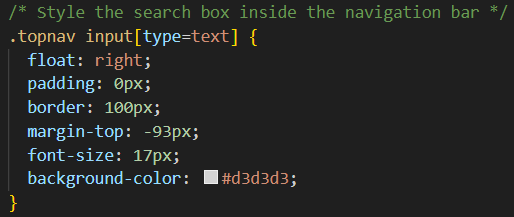
### Logo và Navbar

#### B1: Sử dụng Bootstrap, tạo container, row, column và navbar, thanh tìm kiếm

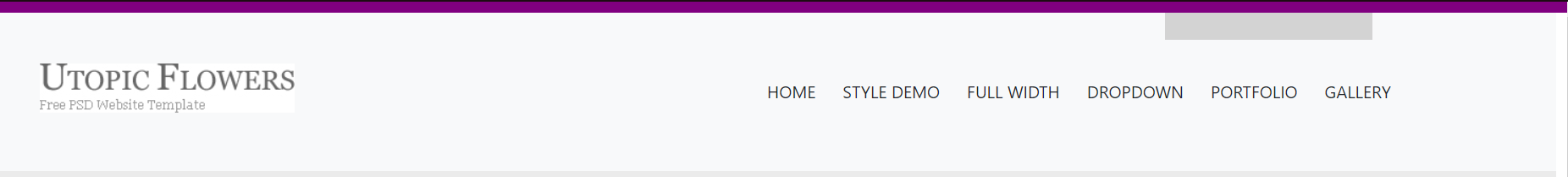
#### HTML:



#### B2 : Ở phần CSS ta cần căn chỉnh chiều cao , màu nền và khoảng cách của thanh tìm kiếm sao cho đúng với mẫu :



#### Kết quả ta thu được là hình bên dưới :

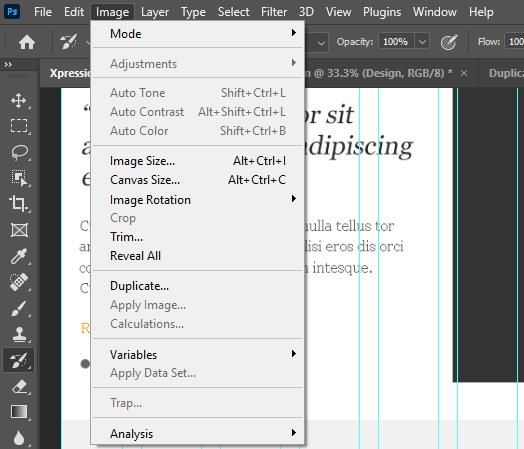


### Logo

#### B1 : Sử dụng công cụ Adobe Photoshop để tiến hành lưu logo từ mẫu

#### Nhần phím Alt đồng thời kết hợp ấn chuột phải vào hình biểu tượng con mắt ở ảnh muốn lưu.

#### Nhấn vào phần Image -> chọn Trim -> Chọn OK



#### Cuối cùng ấn tổ hợp cùng lúc phím Shift + Ctrl + Window và phím S.

#### B2 : Đặt tên và lưu ảnh sau đó đưa ảnh vào chung file với file HTML.

#### B3 : Đưa vào bài như hình dưới đây:

#### HTML :



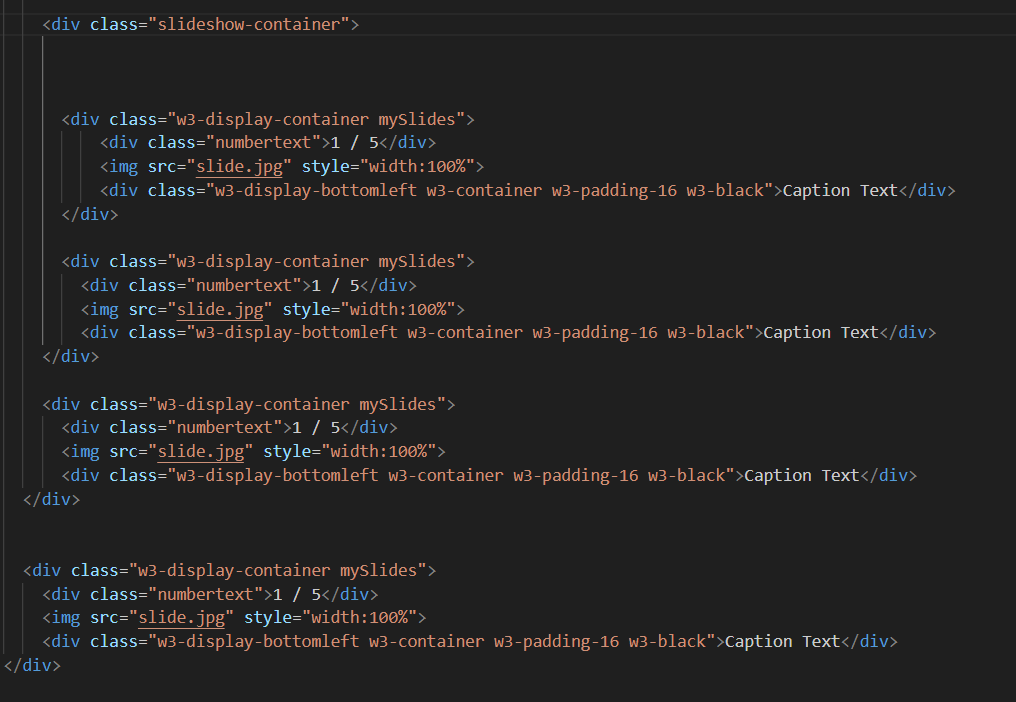
### Slider

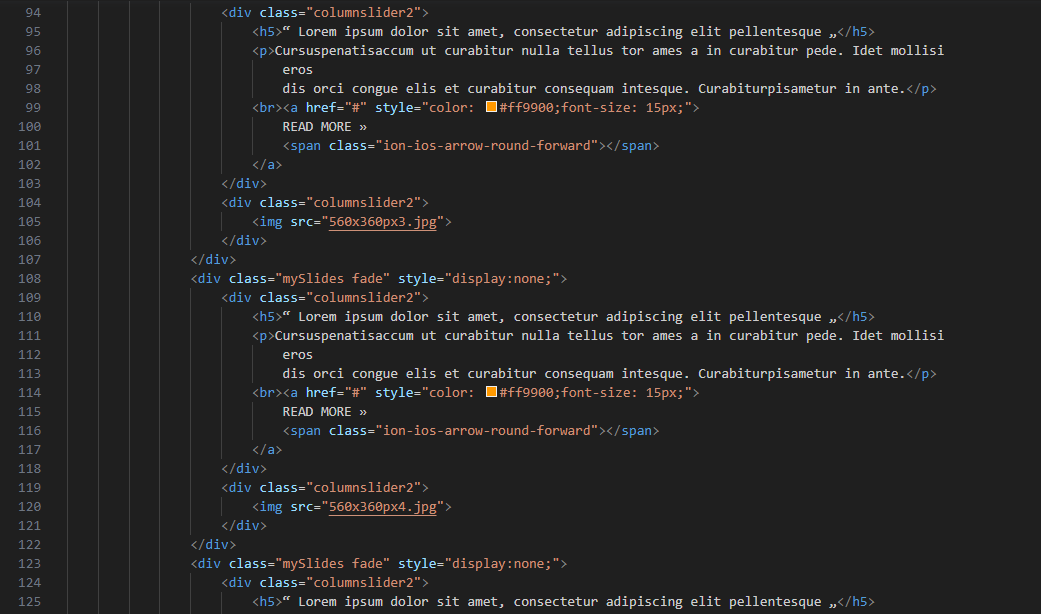
#### B1 : Đặt 1 class tổng cho phần slider

#### B2 : Đặt class column và row để chia slider làm 2 phần ảnh và chữ

#### B3 : Sử dụng thẻ <h> và <p> để viết văn bản vào bên trái cùng với thẻ <img> để chèn ảnh và bên phải

#### HTML :





#### Sử dụng thẻ span đặt class dot cùng với lệnh onclick để tạo ra những chấm chuyển slide như hình dưới đây :

#### 

#### CSS :

#### Chỉnh kích thước, độ rộng, chiều cao của phần column và row sao cho đúng với mẫu:

#### 

#### JS:

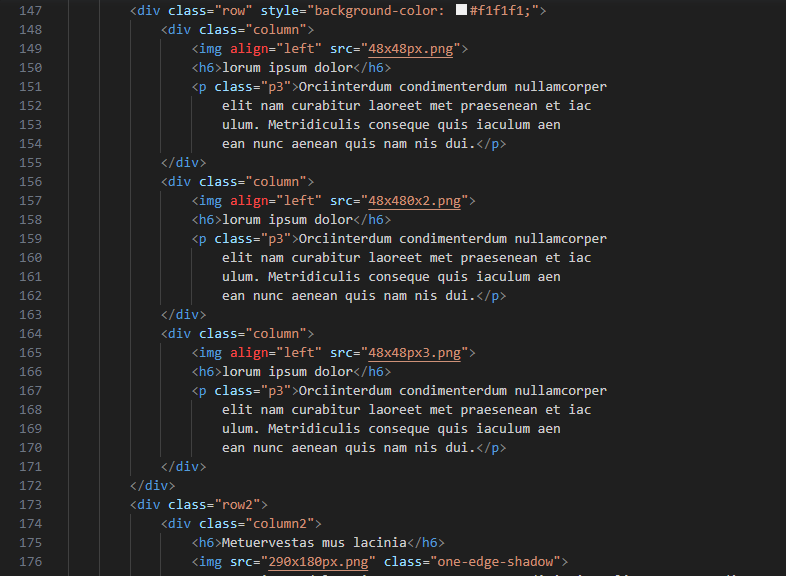
#### 

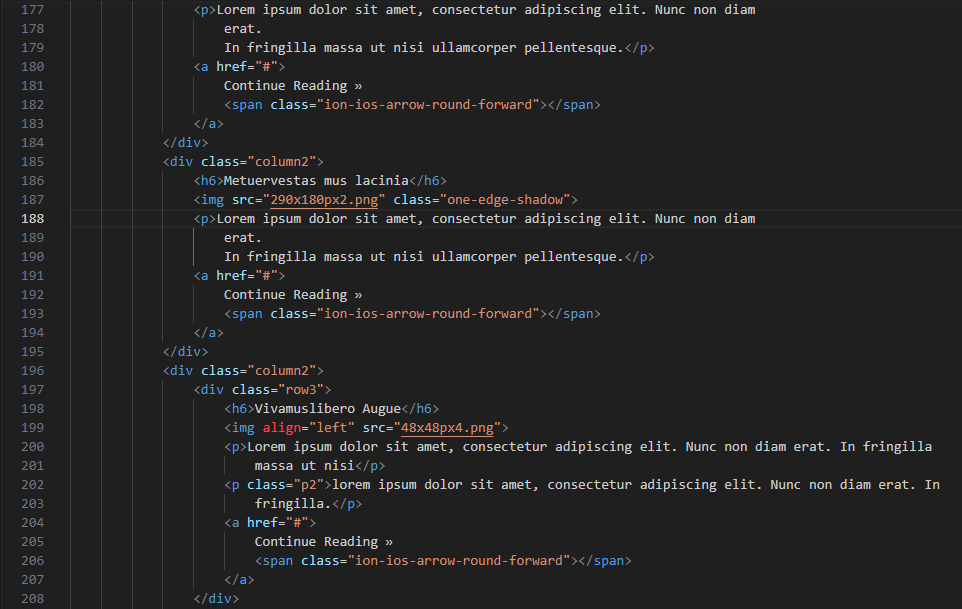
### A

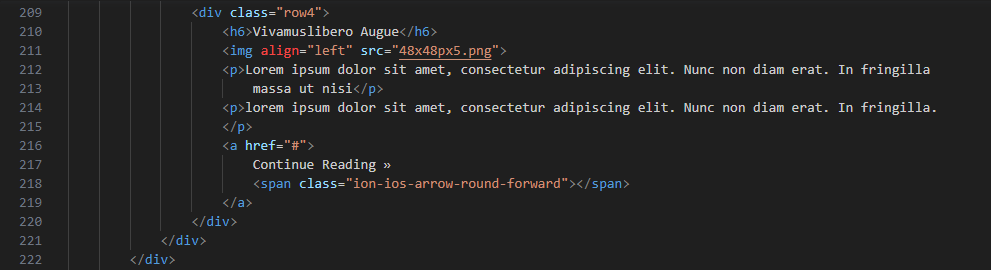
#### B1 : sử dụng class row và column để chia thành 3 cột bằng nhau .

#### B2 : Ở phần row thứ 2 ta vẫn dung column để chia thành 3 cột bằng nhau. Nhưng ở cột thứ 3 ta tiếp tục chia thành 2 row bằng nhau .

#### HTML :







#### CSS :

#### Sử dụng Adobe Photoshop để căn chỉnh các kích thước giống với mẫu

Footer

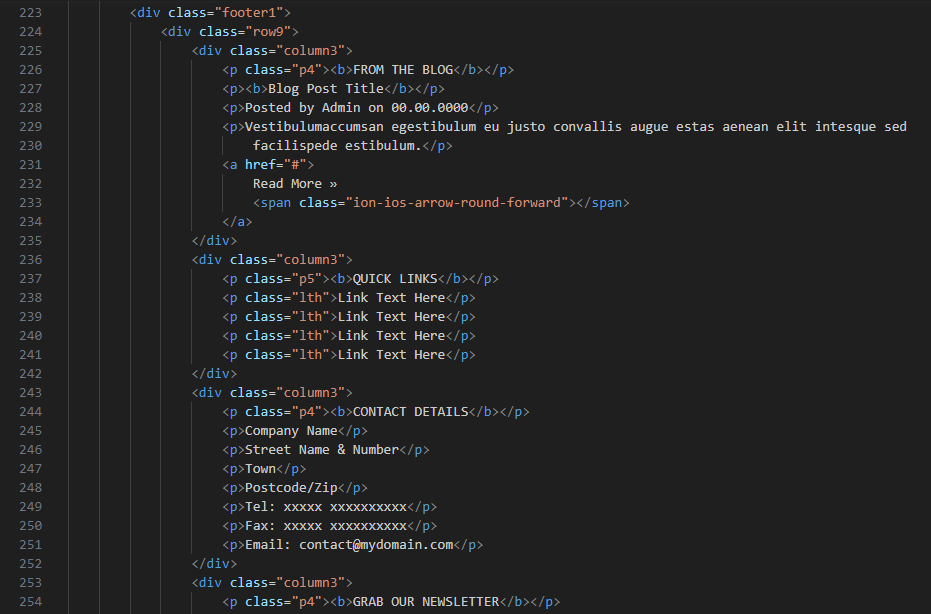
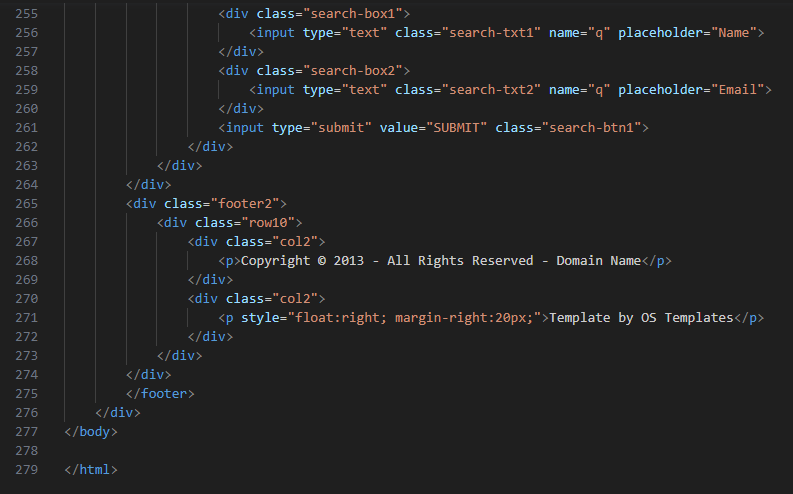
#### B1 : Đặt 1 class “row” sau đó dung class “column” để chia footer thành 4 phần bằng nhau

#### B2 : sử dụng thẻ <p> để viết văn bản

#### B3 : Tạo chữ “Read More” đặc biệt bằng thẻ <span>

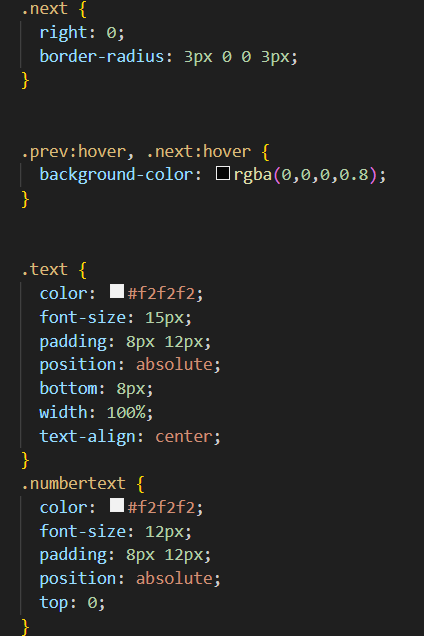
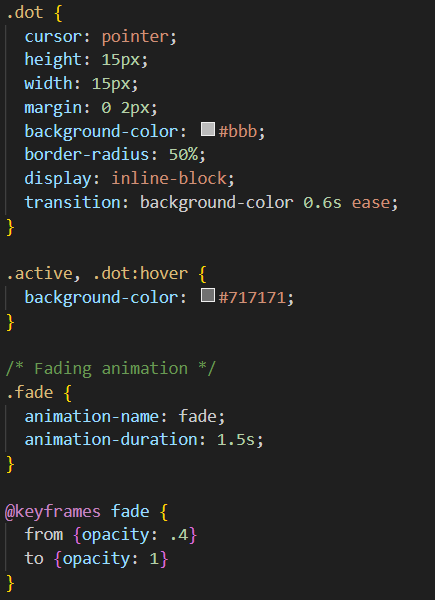
#### B4 : Sử dụng thẻ <input> để tạo thanh tìm kiếm

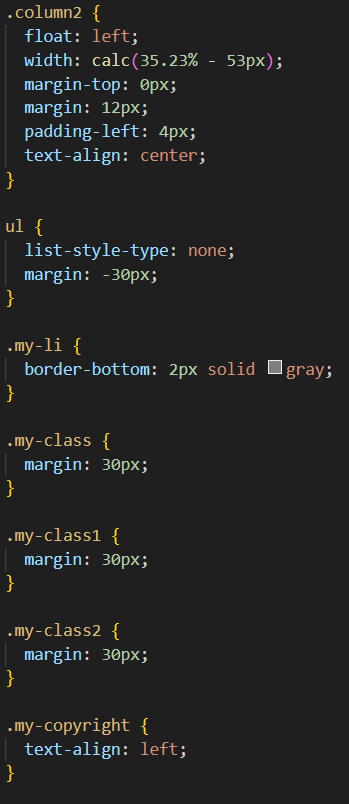
#### HTML :



#### CSS :

#### 

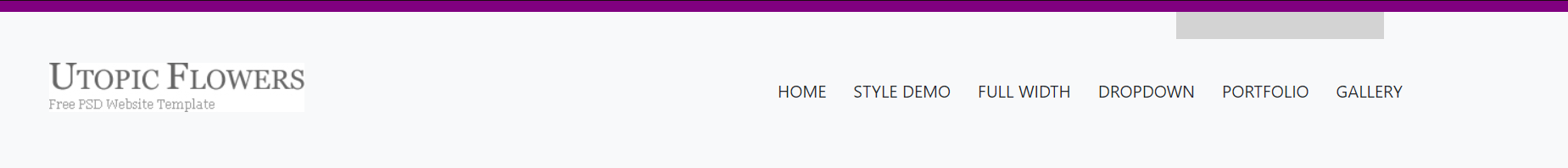
 



# CHƯƠNG 3 – KẾT QUẢ ĐẠT ĐƯỢC

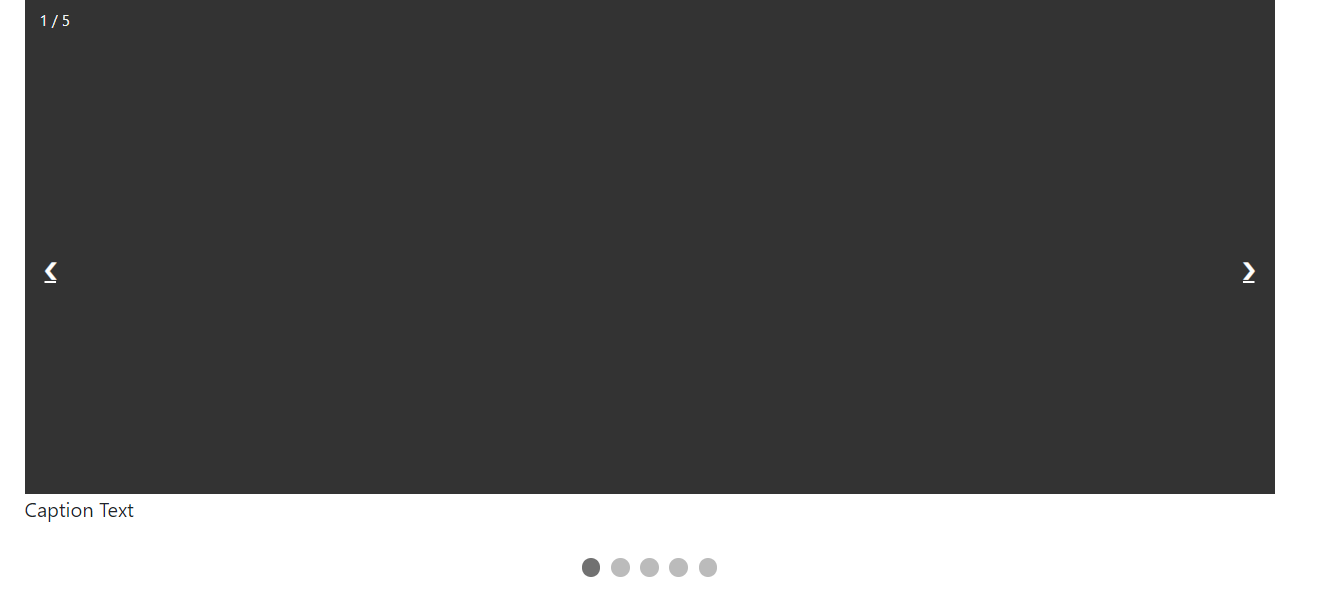
## Topbar, Logo và menu

#### Sau khi thực hiện những bước như trên ta có được sản phẩm như hình bên dưới :



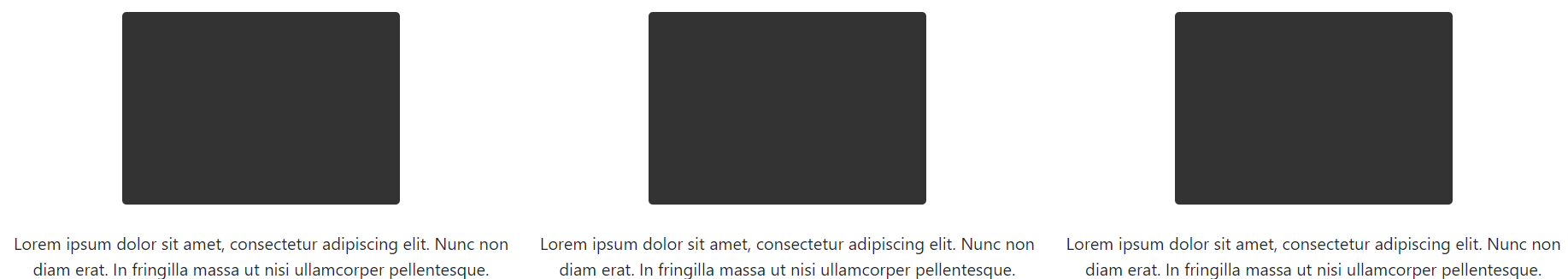
## Slider

#### Sau khi thực hiện các bước như trên ta được 1 slider như hình dưới đây :



## A

#### Sau khi thực hiện các bước như trên ta được 1 phần … như hình dưới đây :



## Footer

#### Sau khi thực hiện các bước như trên ta được phần footer như hình dưới đây :



## Tổng quan trang web

#### 