|  |  |  |
| --- | --- | --- |
| Macintosh HD:Users:hoangnguyen:Documents:TDC:fitlog_blue.png | TRƯỜNG CAO ĐẲNG CÔNG NGHỆ THỦ ĐỨC  **Khoa Công Nghệ Thông Tin** | Macintosh HD:Users:hoangnguyen:Documents:TDC:logoTDC_blue.png |
|  |  |  |

**Chuyên đề CMS**

**BÁO CÁO VIẾT THEME SAMPLE**

**Sinh viên thực hiện:**

1. Lê Công Tuấn
2. Trịnh Tấn Lộc
3. Đỗ Thạch Thông

**Backlog**

1. Tạo các thư mục và file cần thiết
2. Kích hoạt theme
3. Hiển thị post tiltle và post text
4. Thêm link cho mỗi post
5. Thêm header và footer
6. Cấu hình file fucntions
7. Định dạng style.css

**Công việc cụ thể**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Họ tên** | **Công việc** | **Thời gian hoàn thành** |
| 1 | Lê Công Tuấn | 6,7 | 5/10/2019 |
| 2 | Trịnh Tấn Lộc | 4,5 | 5/10/2019 |
| 3 | Đỗ Thạch Thông | 1,2,3 | 5/10/2019 |

**Mục lục**

[**Bước 1:** Tạo thư mục customtheme trong thư mục theme của wp-content 1](#_Toc21332459)

[**Bước 2:** Tạo 2 file index.php và style.css (2 file này quan trọng) 1](#_Toc21332460)

[**Bước 3:** Kích hoạt theme trong khu vực quản trị của wordpress 1](#_Toc21332461)

[**Bước 4:** Thêm code để hiển thị thông tin post title và post text 3](#_Toc21332462)

[**Bước 5:** Thêm link cho mỗi post sử dụng hàm **the\_permalink()** 4](#_Toc21332463)

[**Bước 6:** Thêm header và footer cho trang index 5](#_Toc21332464)

[**Bước 7:** Tạo file functions.php, nơi viết code để chỉnh sửa trang Wordpress 10](#_Toc21332465)

[**Bước 8:** Thêm một vài style cho theme 10](#_Toc21332466)

**Danh mục hình**

[Hình 1. Tạo thư mục customtheme 1](#_Toc21331908)

[Hình 2. Tạo 2 file index.php và style.css 1](#_Toc21331909)

[Hình 3. index.php hiển thị ra ngaoif trang web 1](#_Toc21331910)

[Hình 4. style.css để định dạng theme 1](#_Toc21331911)

[Hình 5. Khu vực theme trong trang quản trị 2](#_Toc21331912)

[Hình 6. Kích hoạt customtheme 2](#_Toc21331913)

[Hình 7. Demo customtheme 2](#_Toc21331914)

[Hình 8. Thêm 1 post mới để hiển thị 3](#_Toc21331915)

[Hình 9. Sử dụng vòng lặp để hiển thị các post 3](#_Toc21331916)

[Hình 10. Thêm các hàm the\_title() và the\_content() 4](#_Toc21331917)

[Hình 11. Kết quả demo 4](#_Toc21331918)

[Hình 12. Thêm đường dẫn tới post bằng the\_permalink() 4](#_Toc21331919)

[Hình 13. Demo có đường dẫn post 5](#_Toc21331920)

[Hình 14. Chuyển sang post chi tiết 5](#_Toc21331921)

[Hình 15. Thêm header và footer 6](#_Toc21331922)

[Hình 16. Demo có header và footer mặc định 6](#_Toc21331923)

[Hình 17. Tạo 2 file header.php và footer.php 7](#_Toc21331924)

[Hình 18. Code đơn giản cho file header.php 7](#_Toc21331925)

[Hình 19. Code đơn gainr cho file footer.php 7](#_Toc21331926)

[Hình 20. Kết quả trang có header và footer tự tạo 8](#_Toc21331927)

[Hình 21. Dùng hàm bloginfo() để hiển thị thông tin header 8](#_Toc21331928)

[Hình 22. Demo kết quả có thông tin trang web 9](#_Toc21331929)

[Hình 23. Dùng hàm bloginfo() để hiển thị thông tin footer 9](#_Toc21331930)

[Hình 24.Demo kết quả có thông tin trang web 10](#_Toc21331931)

[Hình 25. File functions.php 10](#_Toc21331932)

[Hình 26. Thêm thẻ div với class container ở header.php 11](#_Toc21331933)

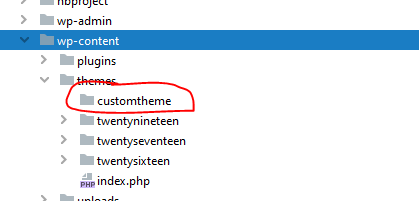
[Hình 27. Đóng thẻ div của class container ở footer.php 11](#_Toc21331934)

[Hình 28. Bao bọc mỗi bài post bằng thẻ <article> 11](#_Toc21331935)

[Hình 29. Thêm đoạn CSS cho file style 12](#_Toc21331936)

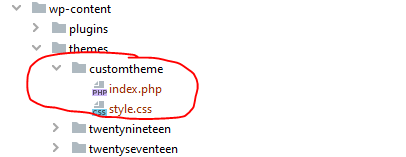
[Hình 30. Kết quả của customtheme 12](#_Toc21331937)

# **Bước 1:** Tạo thư mục customtheme trong thư mục theme của wp-content



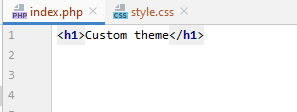
Hình 1. Tạo thư mục customtheme

# **Bước 2:** Tạo 2 file index.php và style.css (2 file này quan trọng)



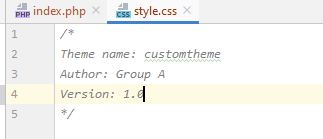
Hình 2. Tạo 2 file index.php và style.css

File index.php: hiển thị ra ngoài trang web



Hình 3. index.php hiển thị ra ngaoif trang web

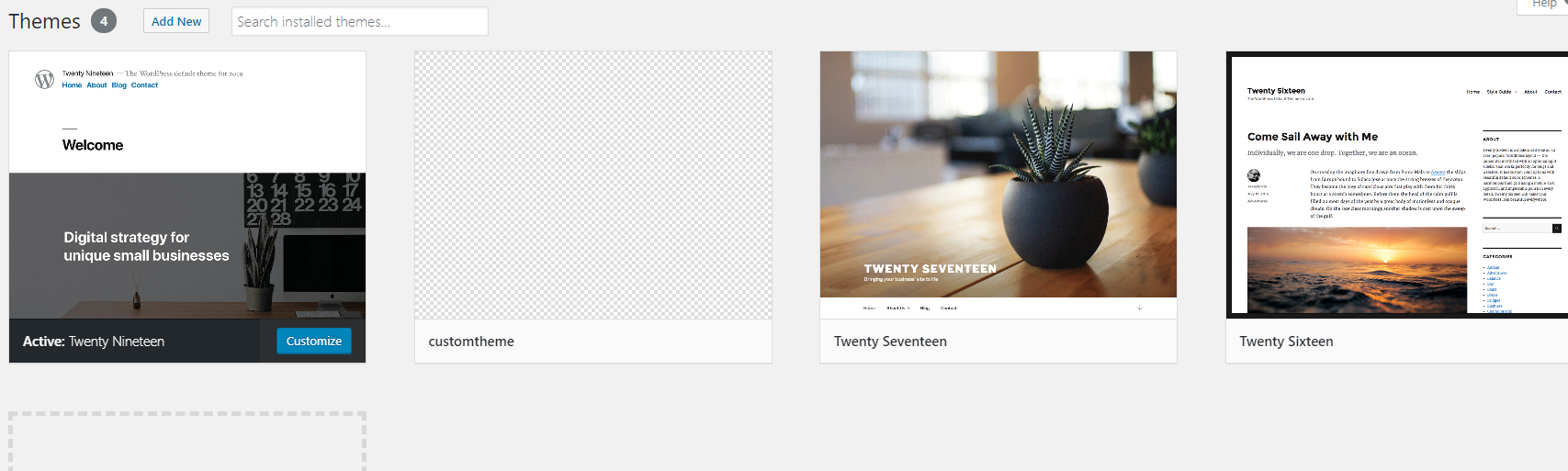
File style.css: Định dạng các thông tin của theme đang tạo



Hình 4. style.css để định dạng theme

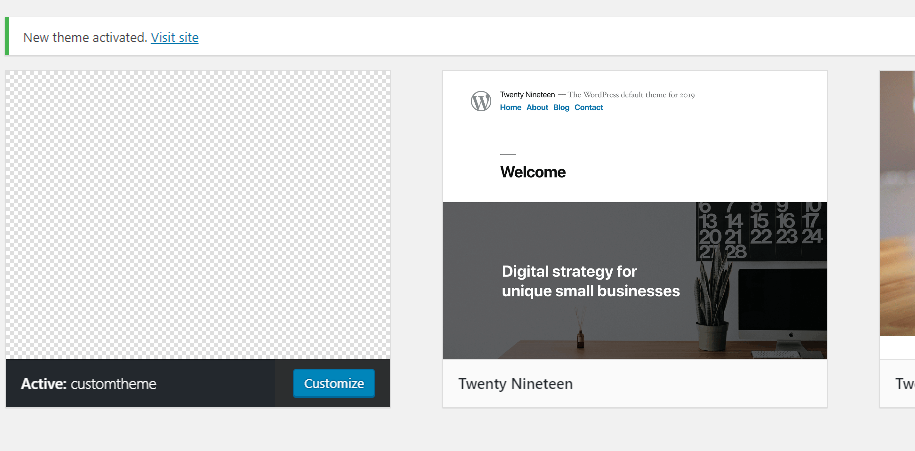
# **Bước 3:** Kích hoạt theme trong khu vực quản trị của wordpress

Dashboard -> Appearance -> Themes



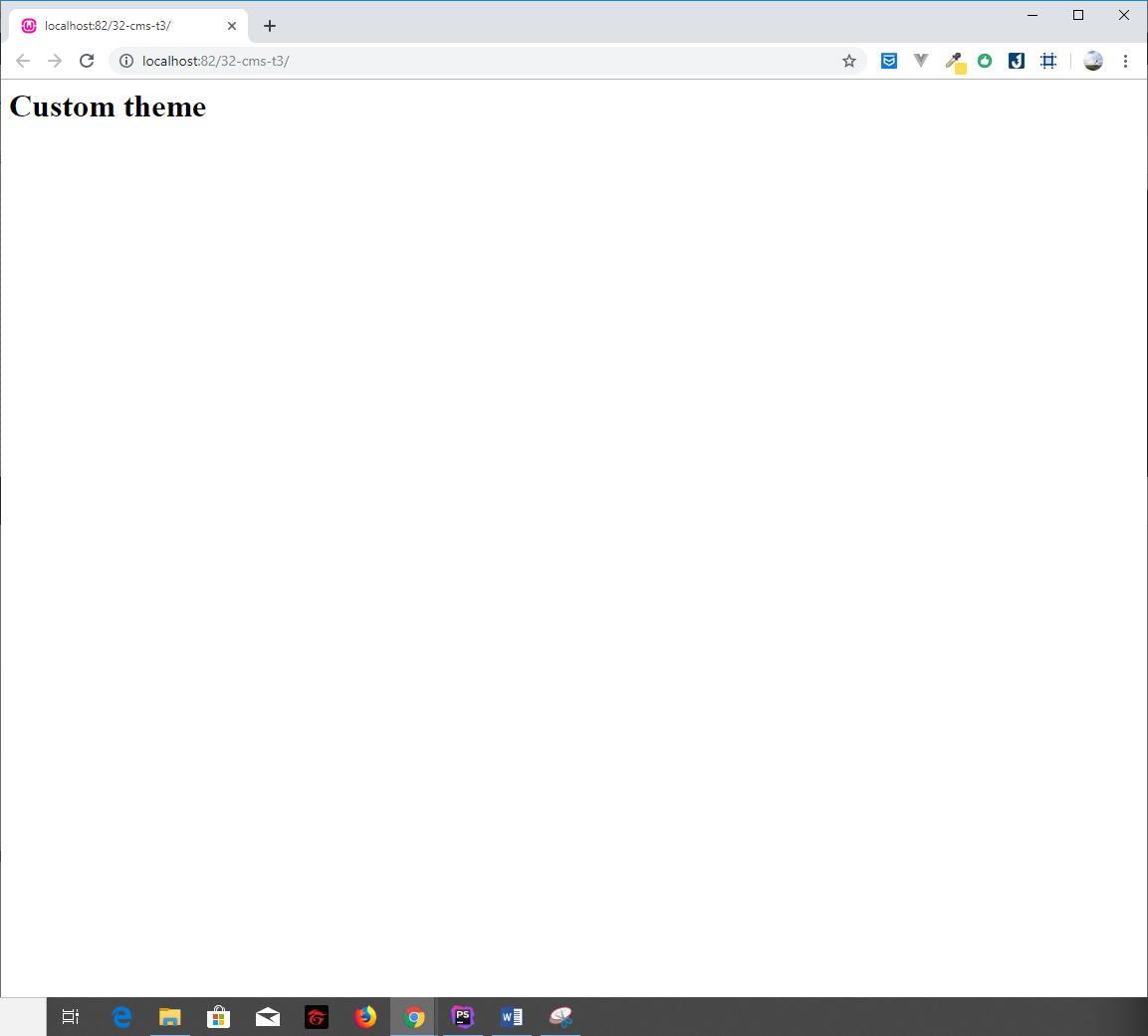
Hình 5. Khu vực theme trong trang quản trị

Chọn customtheme và chọn Activate



Hình 6. Kích hoạt customtheme

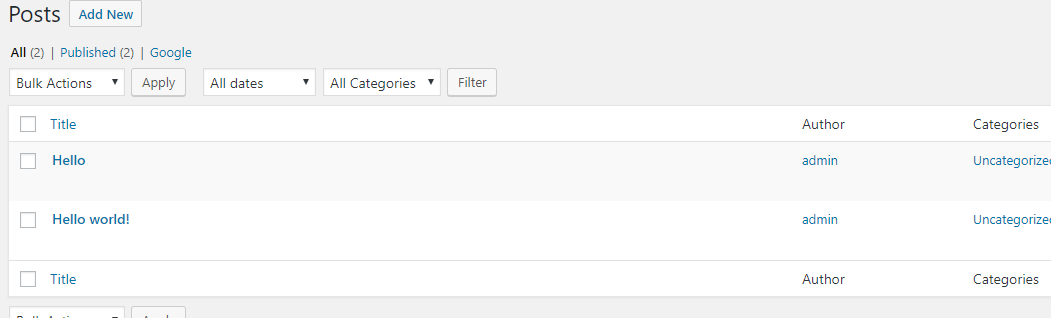
Chạy demo site này



Hình 7. Demo customtheme

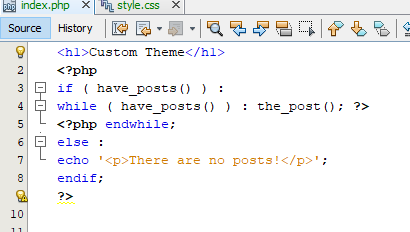
# **Bước 4:** Thêm code để hiển thị thông tin post title và post text

Ta thêm 1 post nhằm có nhiều post để hiển thị



Hình 8. Thêm 1 post mới để hiển thị

Sử dụng vòng lặp để hiển thị các post ra ngoài trang index

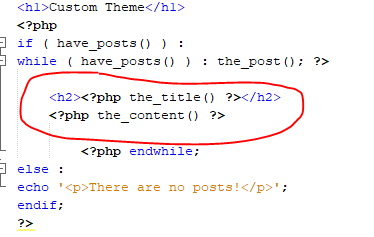


Hình 9. Sử dụng vòng lặp để hiển thị các post

Hàm **have\_post()** để kiểm tra xem có bất kì post nào ở database để lặp hay không

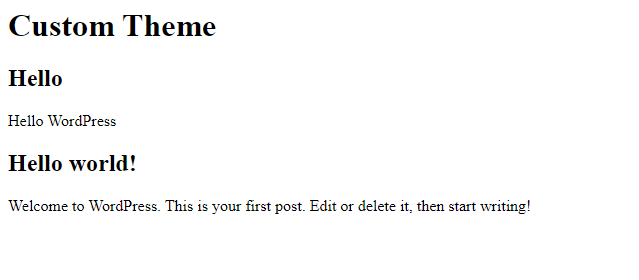
Hàm **the\_post()** để lấy tất cả các post hiển thị ra ngoài

Ta thêm 2 dòng code sử dụng **the\_title()** để hiển thị tiêu đề và **the\_content()** để hiển thị nội dung của bài viết



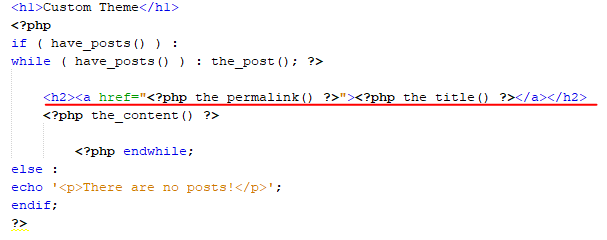
Hình 10. Thêm các hàm the\_title() và the\_content()

Kết quả:



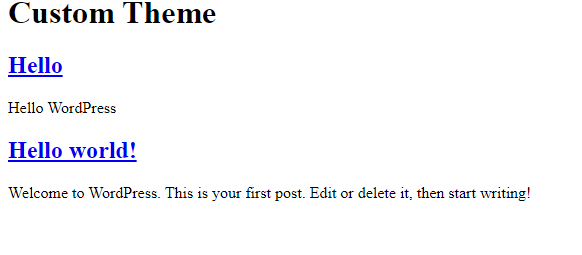
Hình 11. Kết quả demo

# **Bước 5:** Thêm link cho mỗi post sử dụng hàm **the\_permalink()**



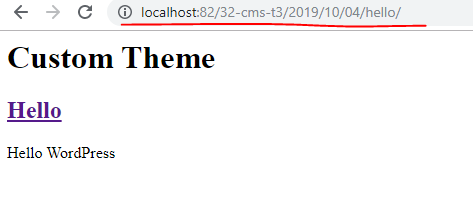
Hình 12. Thêm đường dẫn tới post bằng the\_permalink()

Kết quả:



Hình 13. Demo có đường dẫn post

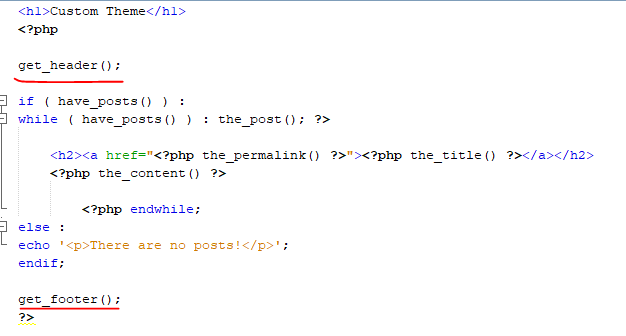
Khi chọn vào post Hello sẽ chuyển sang trang của bài viết đó



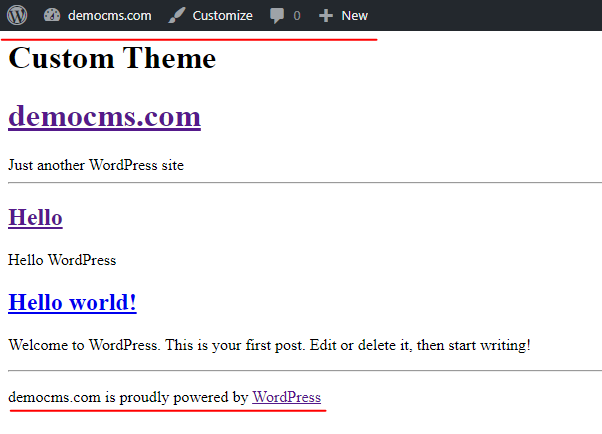
Hình 14. Chuyển sang post chi tiết

# **Bước 6:** Thêm header và footer cho trang index

Header và Footer mặc định của Wordpress sẽ sử dụng hai hàm **get\_header()** và **get\_footer()**

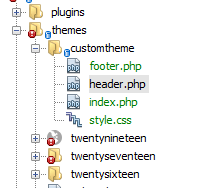


Hình 15. Thêm header và footer

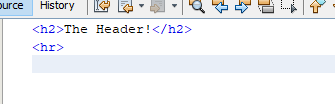


Hình 16. Demo có header và footer mặc định

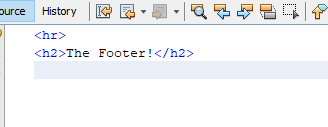
Bây giờ ta sẽ tạo header và footer riêng bằng 2 file header.php và footer.php



Hình 17. Tạo 2 file header.php và footer.php

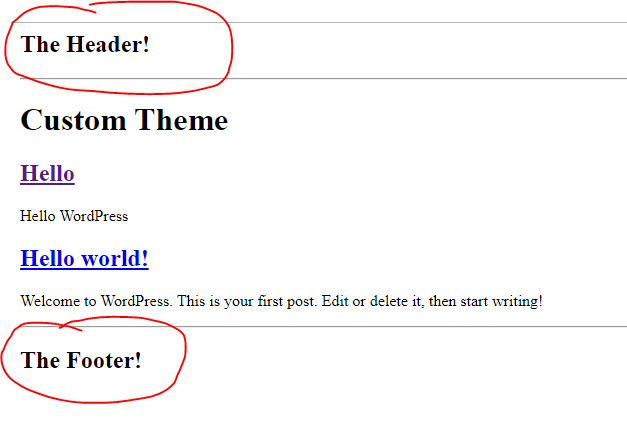


Hình 18. Code đơn giản cho file header.php



Hình 19. Code đơn gainr cho file footer.php

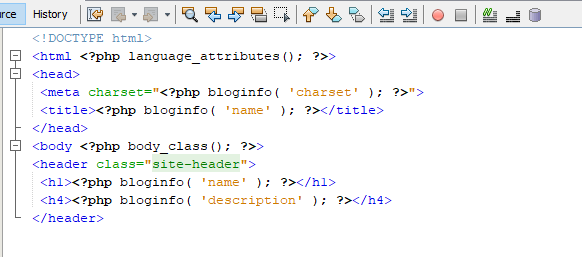
Kết quả:



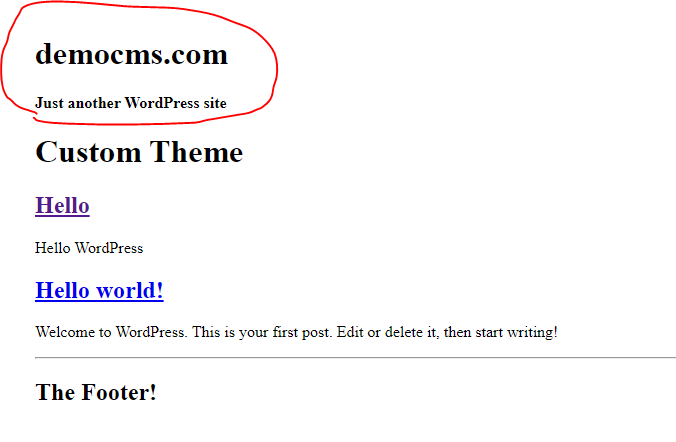
Hình 20. Kết quả trang có header và footer tự tạo

**Chỉnh sửa header.php**

Dùng hàm bloginfo() để hiển thị tên và mô tả của trang web

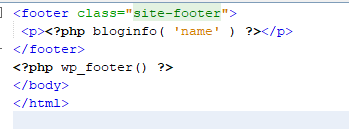


Hình 21. Dùng hàm bloginfo() để hiển thị thông tin header

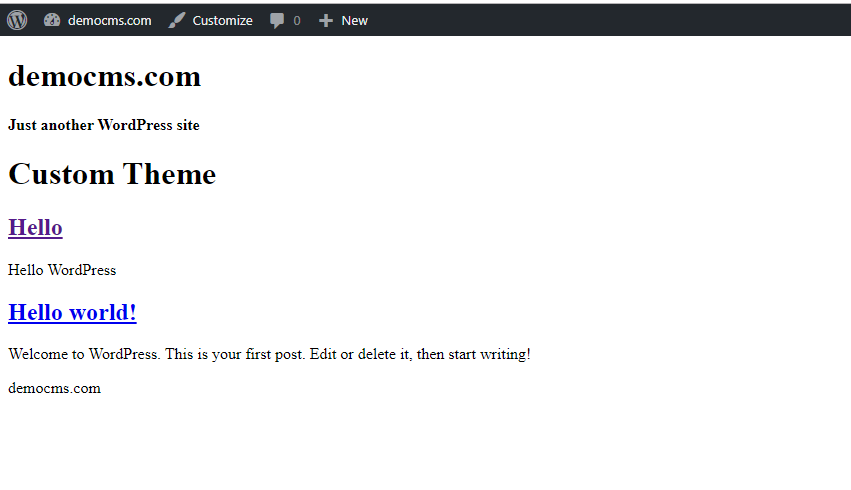


Hình 22. Demo kết quả có thông tin trang web

**Chỉnh sửa footer.php**



Hình 23. Dùng hàm bloginfo() để hiển thị thông tin footer

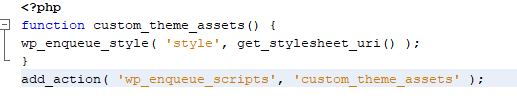


Hình 24.Demo kết quả có thông tin trang web

# **Bước 7:** Tạo file functions.php, nơi viết code để chỉnh sửa trang Wordpress

File functions.php có vài đặc điểm cần nhớ:

* Nằm trong thư mục theme
* Chỉ thực thi khi theme được kích hoạt
* Chỉ áp dụng với theme hiện tại
* Có thể gọi các function của PHP, các function của Wordpress hoặc các function tự tạo

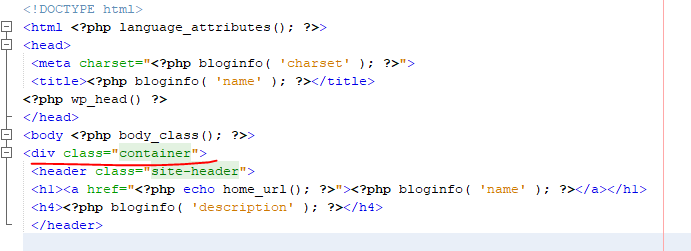


Hình 25. File functions.php

Đoạn code trên dùng để include file CSS hoặc làm cho file CSS hoạt động với theme.

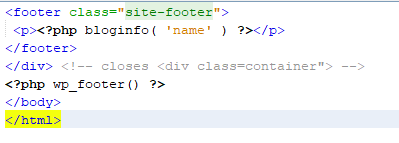
# **Bước 8:** Thêm một vài style cho theme

File header.php: ta thêm thẻ div với class container



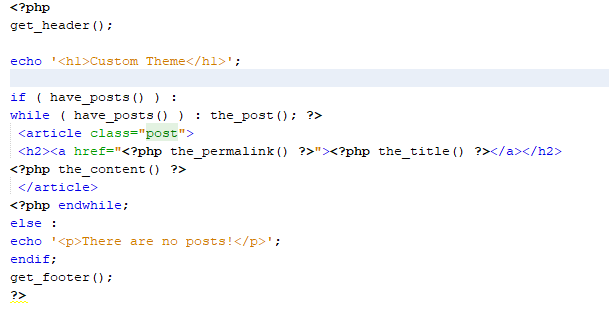
Hình 26. Thêm thẻ div với class container ở header.php

File footer.php: đóng thẻ div của class container ở file header.php



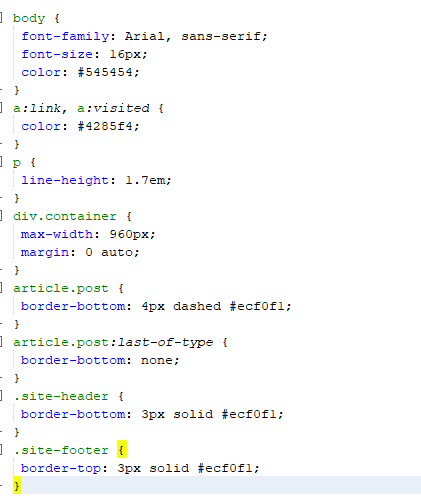
Hình 27. Đóng thẻ div của class container ở footer.php

File index.php: Bao bọc mỗi bài post bằng thẻ <article>



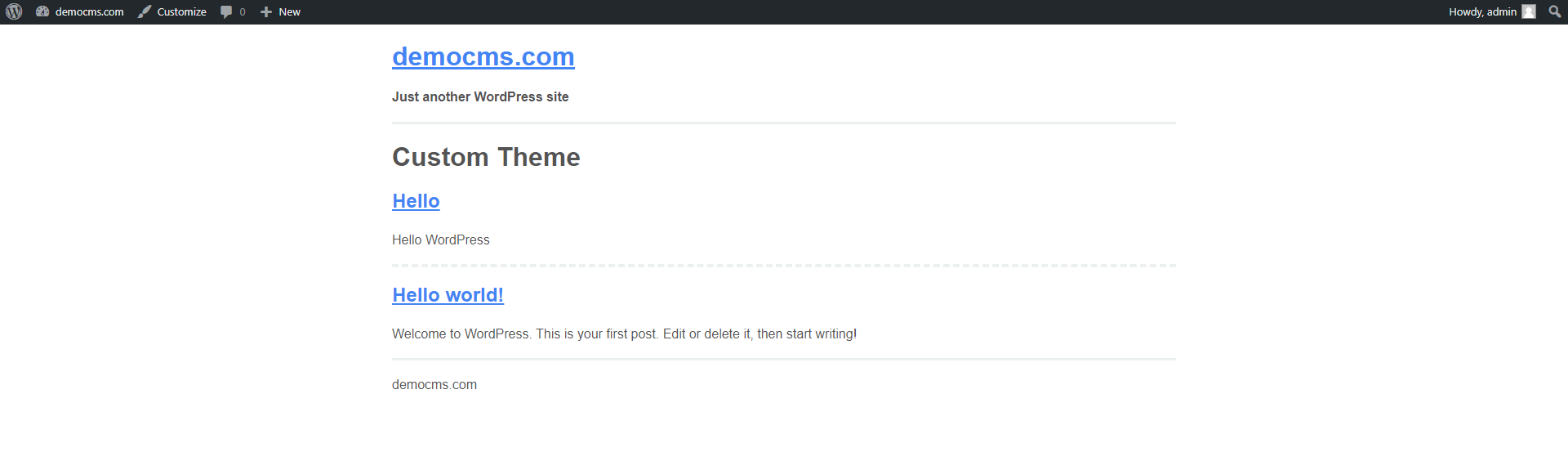
Hình 28. Bao bọc mỗi bài post bằng thẻ <article>

File style.css: thêm đoạn CSS để làm trang web trở nên dễ nhìn hơn



Hình 29. Thêm đoạn CSS cho file style

Kết quả của theme



Hình 30. Kết quả của customtheme