* **Nội dung chính**

Chapter 4: The Wordpress Core

Chapter 5: Posts, pages, and custom content

— Tạo, lưu trữ, truy xuất và tương tác với phần lớn nội dung WordPress.

- Vòng loop kiểm soát việc hiển thị và sắp xếp nội dung.

Chapter 7: Metadata — mô tả tags, categories và các thẻ phân loại do user tạo ra.

Chapter 9: Themes — Các functions hỗ trợ làm theme của WordPress

Chapter 8: Các Actions, filters để làm plugins

Chapters 12 and 14: Xác thực Users and authors, quản lý truy cập

-------------------------------------------------

* **Chapter 4: The Wordpress Core**

- functions.php chứa các hàm API WordPress chính. Các hàm này được sử dụng để tương tác với WordPress. Các plugin, theme và core WordPress đều sử dụng các hàm này

- option.php chứa các hàm API để làm việc với các Options trong WordPress

- formatting.php chứa các hàm định dạng của WordPress. Các hàm này định dạng đầu ra theo nhiều cách khác nhau

- plugin.php chứa các hàm để làm Plugin cho WordPress, bao gồm:

➤ add\_filter — Hooks that the WordPress core launches to filter content before displaying on the screen or saving in the database.

➤ add\_action — Hooks that the WordPress core launches at specifc points of execution.

➤ register\_activation\_hook — Hook called when a plugin is activated.

➤ register\_deactivation\_hook — Hook called when a plugin is deactivated.

➤ plugin\_dir\_url — Returns the flesystem directory path for the plugin.

➤ plugin\_dir\_path — Returns the URL for the plugin

- user.php chứa các hàm làm việc với đối tượng User của WordPress

- post.php chứa các hàm làm việc với đối tượng Post của WordPress

- taxonomy.php chứa các hàm được sử dụng bởi WordPress Taxonomy. Các Taxonomy được sử dụng để quản lý các mối quan hệ phân cấp của metadata như category và tag

DON’T HACK THE CORE!

Việc hack core WordPress có thể khiến việc cập nhật lên phiên bản WordPress mới nhất trở nên rất khó khăn

Bất kỳ tính năng hoặc chức năng nào không có trong WordPress đều có thể được thêm vào bằng plugin. Đôi khi, hack core có thể là câu trả lời dễ dàng, nhưng về lâu dài, nó sẽ khiến cuộc sống của bạn khó khăn hơn

-------------------------------------------------

* **Chapter 5: Posts, pages, and custom content**

Loop là cách WordPress xác định nội dung nào (post, page hoặc nội dung tùy chỉnh) sẽ hiển thị trên trang bạn đang truy cập.

Loop có thể hiển thị một mẩu nội dung hoặc một nhóm post và page được chọn và sau đó được hiển thị bằng cách lặp qua nội dung; do đó, nó được gọi là Loop.

Loop chọn các post từ cơ sở dữ liệu MySQL dựa trên một tập hợp các tham số và các tham số từ URL được sử dụng để truy cập trang web WordPress của bạn. Ví dụ: trang chủ có thể hiển thị tất cả các post trên blog theo thứ tự thời gian ngược theo mặc định. Một trang category, được truy cập thông qua một URL như http://example.com/category/zombies, chỉ hiển thị các post trên blog được chỉ định cho category đó, trong trường hợp này là các page được đưa vào category zombies. WordPress ánh xạ gần như mọi tham số về post vào các tham số để select vào MySQL. Rất dễ tùy chỉnh nội dung nào được hiển thị và ở đâu trên trang web của bạn khi hiểu rõ cách Loop dịch URL thành những gì bạn thấy khi truy cập liên kết đó.

* **Các bước xử lý của Loop:**
* Web server bắt đầu bằng cách tải core WordPress thông qua index.php để bắt đầu thiết lập cho Loop.
* WordPress phân tích cú pháp URL thành một tập hợp các tham số truy vấn. Điều này xảy ra trong phương thức parse\_query() trong đối tượng WP\_Query.
* WordPress sau đó chuyển đổi tập hợp các tham số truy vấn thành câu SQL truy vấn cơ sở dữ liệu MySQL để lấy nội dung hiển thị. Điều này xảy ra trong phương thức get\_posts() trong đối tượng WP\_Query.
* Sau khi nội dung được truy xuất, WordPress sẽ thiết lập tất cả các tag is\_ ví dụ is\_home() và is\_page().
* WordPress chọn một template từ theme của bạn dựa trên loại truy vấn và số post được trả về — ví dụ: nếu trả về một post duy nhất nó sẽ dùng template single.html để hiển thị nội dung post.

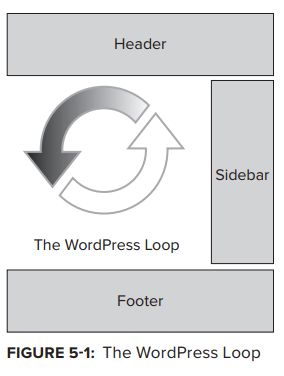
Theo mặc định, WordPress định nghĩa hai loại nội dung: post và page. Tất cả các loại nội dung được lưu trữ trong cùng một bảng MySQL và được phân biệt theo trường "type".

* **Loop là trái tim của một Theme,**

Loop là thứ kiểm soát cách nội dung được hiển thị. Loop kết nối giữa dữ liệu trong cơ sở dữ liệu MySQL và HTML được hiển thị trên trình duyệt.

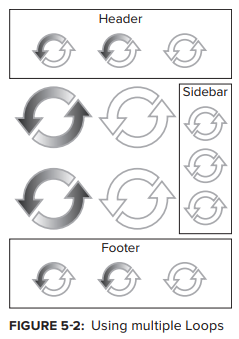
Về cơ bản, bất kỳ nơi nào post hoặc page được hiển thị, WordPress sẽ sử dụng Loop. Đây có thể là một post hoặc page duy nhất, hoặc có thể là một vòng lặp các post, page.

Hầu hết các Theme WordPress đều có header, footer, và sidebar. Hình 5-1 cho thấy cách Loop được đặt trực tiếp vào giữa các phần tử này, tạo ra vùng nội dung trang web của bạn.



Loop được sử dụng trong các file template trong theme của bạn. Có thể tạo Custom Loops ở bất kỳ đâu trong các file template của theme. Hình 5-2 cho thấy Custom Loops cũng được sử dụng trong các Plugin và Widget. Có thể sử dụng Loops ở bất kỳ đâu bên trong WordPress. Có nhiều phương pháp khác nhau để tạo Custom Loops tùy thuộc vào nơi chúng được sử dụng và các tác dụng phụ tiềm ẩn của mỗi cách sẽ khác nhau.

Loop có thể được sử dụng trong toàn bộ các tệp mẫu chủ đề của bạn. Các vòng lặp tùy chỉnh có thể được tạo trong header, sidebars, footer và các khu vực nội dung chính của trang web của bạn.



* **Code ví dụ một Loop tối thiểu:**

<?php

if ( have\_posts() ) :

while ( have\_posts() ) : *//Nếu còn post*

the\_post(); *//Làm giàu dữ liệu cho đối tượng post*

//loop content (template tags, html, etc) *//Hiển thị post*

endwhile;

endif;

?>

Playlist đầy đủ về lập trình Wordpress theme: <https://www.youtube.com/playlist?list=PLpcSpRrAaOaqBzG9rTA66V494iTRySkXq>

Lập trình theme kiểu cổ điển: https://www.youtube.com/watch?v=wUz69qRjN2s&list=PLpcSpRrAaOaqBzG9rTA66V494iTRySkXq

Lập trình theme kiểu mới, theo block: [https://www.youtube.com/watch?v=KBF359\_ZYZ0&list=PLpcSpRrAaOaqBzG9rTA66V494iTRySkXq](https://www.youtube.com/watch?v=KBF359_ZYZ0&list=PLpcSpRrAaOaqBzG9rTA66V494iTRySkXq&index=16&t=6s)