**Cách thức vận hành của mã nguồn WordPress**

1. **Về cơ sở dữ liệu của WordPress.**

Với WordPress các bảng của cơ sở dữ liệu mySQL đã được tự động tạo và thiết lập sẵn một vài giá trị khi chúng ta tiến hành cài đặt website ở bước trên. Tất cả các dữ liệu được WordPress lưu vào database gồm: nội dung các bài viết ta tạo ra, các thiết lập tham số cho website. Rồi sau đó, các tính năng trong mã nguồn sẽ tương tác với các bảng dữ liệu đó và đưa ra ngoài trình duyệt hiển thị dưới dạng HTML thông qua các tập tin đã được thiết lập khuôn mẫu trong theme (ta gọi là template).

1. **Cách thức hoạt động của WordPress khi ta truy cập vào websites WordPress.**
2. **Khởi động mã nguồn.**

* Tập tin index.php trong mã nguồn (không phải của theme hay plugin) được tải ra, sau đó nó sẽ truy xuất các tập tin cốt lõi khác như wp-config.php vốn để kết nối đến cơ sở dữ liệu, wp-settings.php,….
* Ở bước này, WordPress sẽ kết nối đến cơ sở dữ liệu được thiết lập trong wp-config.php, sau đó sẽ tiến hành tải những tệp chức năng trong mã nguồn như /wp-include/functions.php, /wp-include/options.php,…nhằm nạp sẵn các chức năng cần thiết để có thể tiếp nhận những dữ liệu.
* Và cũng trong bước này, toàn bộ các dữ liệu được lưu vào bảng wp\_options có giá trị ở cột autoload là true sẽ được lôi ra hết, dù thiết lập đó có được sử dụng hay không.
* Mục đích của bước này là mang sẵn các tuỳ chọn của website cùng các plugin để hỗ trợ cho các bước sau, đặc biệt là bước tải plugin để nó làm việc chính xác.

1. **Kích hoạt plugin.**

Các plugin đã kích hoạt trong website sẽ được tải ra ngay sau khi mã nguồn WordPress khởi động xong. Bởi vì các tính năng trong WordPress Core được sử dụng trong plugin thường sẽ được gắn vào hookinit (tạm thời hiểu nó là một điểm neo để kích hoạt kịch bản nào đó) nên nó sẽ load ngay sau khi WordPress khởi động.

1. **Thực thi tập tin functions.php trong theme.**

Lúc này, WordPress sẽ tiến hành dò tìm đến tập tin functions.php trong theme mà bạn đang sử dụng cho trang web để tải các tính năng mà người tạo ra theme đã khai báo ở đó.

Vậy làm sao WordPress có thể hiểu được website đang dùng theme nào? Đó là ở bước 1, nó đã kết nối vào database và dựa theo khoá current\_theme trong cột option\_name tại bảng wp\_options. WordPress xác định theme hiện tại thông qua database.

1. **Phân tích truy vấn và khởi tạo truy vấn.**

Đây là bước quan trọng để website của bạn có thể hiển thị nội dung ra bên ngoài, vì các nội dung sẽ được trả về sau khi các truy vấn gửi vào database.

Ban đầu WordPress sẽ chạy hàm wp() được thiết lập trong /wp-include/functions.php vốn để gọi phương thức wp->main() cho mục đích thiết lập truy vấn. Đối tượng wp được tạo ra bởi lớp WP trong /wp-include/class-wp.php. Lúc này, phương thức $$p->parse\_request() sẽ được khởi động cho mục đích phân tích truy vấn dựa theo đường dẫn của website.

*Quá trình phân tích được thực hiện như sau:*

* Từ URL của website phương thức parse\_request() sẽ tách các biến của phương thức GET ra khỏi URL.
* Sử dụng các rewrite rules (luật viết url cho trang web) đang được thiết lập bởi website. Các luật này là tập các mẫu cho các permalink của WordPress. Sẽ có một hành động được sinh ra nếu permalink trùng khớp với các luật.
* Duyệt qua các luật rewrite theo thức tự, nếu các permalink khớp với các luật thì WordPress sẽ trích lấy các thông tin của URL dựa vào từng luật cụ thể. Nếu không có luật nào khớp lỗi 404 sẽ được sinh ra.
* Sau khi trích được các biến truy vấn dựa vào các luật rewrite và URL. WordPress sẽ lưu các giá trị đó vào một mảng truy vấn và sử dụng nó để tạo các câu lệnh truy vấn bằng SQL.

WordPress sẽ tự động sinh truy vấn dựa theo đường dẫn, ví dụ khi chúng ta vào <http://domain.com/?p=123> thì WordPress sẽ gửi một truy vấn vào database để lấy dữ liệu của post mang số ID là 123.

Sau khi truy vấn được phân tích, WordPress sẽ làm việc tiếp theo là thiết lập các hàm điều kiện thông qua phương thức wp\_query->parse\_query().

Sau đó nó sẽ chuyển các truy vấn đã được tạo ra thành truy vấn bằng các lệnh SQL nhằm gửi đến MySQL để lấy dữ liệu bài viết bằng phương thức wp\_query->get\_posts().

Nếu database có dữ liệu, các bài viết sẽ được lấy về sau khi gửi truy vấn và nó sẽ được lưu vào đối tượng $$p\_query để nó sử dụng trong các vòng lặp cho việc hiển thị bài viết.

Trong quá trình gửi truy vấn này, nếu nó không tìm thấy dữ liệu thì sẽ phân tích và hiển thị báo lỗi 404.

1. Thực thi các tập tin khuôn mẫu trong theme.

Sau khi nó đã có dữ liệu bài viết và các dữ liệu liên quan mà nó đã làm ở bước 4, thì nó sẽ tiến hành xử lý các tập tin template của theme thông qua cấu trúc template. Sau đó các nội dung và trang chủ của website sẽ được hiển thị dựa theo các template tags dưới dạng HTML.

1. **Vòng lặp lấy ra các bài viết trong WordPress**

Như trình bày ở trên khi sử dụng phương thức get\_posts() ta sẽ lấy được dữ liệu các bài viết được truy vấn ra theo URL, các dữ lieuj này được lưu vào đối tượng wp\_query và ta sẽ sử dụng vòng lặp để lấy ra các bài viết kết quả. Và chúng ta sẽ lặp thông qua phương thức have\_posts() trong đối tượng wp\_query. Cụ thể đoạn vòng lặp hiển thị các bài viết kết quả như sau:

<?php

if( $ wp\_query->have\_posts()){

while( $wp\_query->have\_posts()){

$wp\_query->the\_posts();

Echo $post->post\_title.’<br>’;

}

}

?>

Đoạn code trên rất đơn giản nó sử dụng phương thức have\_posts() để kiểm tra xem trong kết quả truy vấn trả (được lưu trong đối tượng wp\_query) về có bài viết nào hay không. Nếu có ta sử dụng vòng lặp while để lặp qua kết quả trả về. Trong khi lặp ta sử dụng phương thức the\_post() để thiết đếm chỉ mục để gọi bài kế tiếp trong vòng lặp, nếu không khai báo phương thức này trong vòng lặp thì bài viết đầu tiên sẽ bị lặp đi lặp lại hoài mà không có điểm dừng.

Tiếp đến ta sử dụng các phương thức, thuộc tính của đối tượng post (đối tượng chứa bài viết đang duyệt) để hiển thị các thuộc tính của bài viết, điển hình ở trên là thuộc tính post\_title để hiển thị tiêu đề bài viết.