**Підключення**

1. В папці C/OSPanel/home створити папку проекту, створити там папки .osp, public
2. Видобути голий WordPress в папку public
3. В PhpMyAdmin потрібно створити базу даних з префіксом wp\_ та кодування utf8\_general\_ci
4. Відкрити через OSPanel веб-сайт проекту, підключитися до бази даних (логін, пароль, хост до бази даних для підключення можна подивитись в файлі PhpMyAdmin/config.inc.php)
5. Ввести необхідну інформацію (Назва сайту, Ім’я користувача, пароль(на даному етапі можна легкий пароль), пошту)

**Створення теми**

Для того, щоб почати створювати нашу власну тему, потрібно перенести всю нашу верстку в папку wp-content/themes, кожна тема WP має містити файли:

1. index.php – основний файл теми
2. style.css – файл зі стилями
3. functions.php – файл з налаштуваннями
4. screenshot.png – прев’ю теми в адмінці WP.

Всі ці файли мають бути в кореневій папці, а також всі файли .html мають бути переіменовані на .php.

Всі файли з ресурсами, такими як додаткові стилі, зображення, шрифти мають знаходитись в папці assets, що знаходиться в кореневій папці проекту.

Також, на початку файла style.css має бути коментар:

/\*

Theme Name: Назва теми

\*/

Після виконання цих кроків, ми вже можемо активувати нашу тему в адмінці WP в вкладці вигляд -> теми і в разі коректного підключення, при переході на сайт ми вже можемо побачити html розмітку нашого веб-сайту.

Далі, ми додаємо до нашого сайту підключаємі футер та хедер:

1. Створюємо в корені теми два файли footer.php header.php;
2. Вирізаємо з файла index.php той код, який буде повторюватись на кожній сторінці сайту, тобто хедер з головним меню і футер з меню та вставляємо код в відповідний файл footer.php чи header.php;
3. В тих місцях файла index.php, де ми вирізали код прописуємо php функції <?php get\_header(); ?>, та <?php get\_footer(); ?> що підключають хедер і футер.

Також, в файлах footer.php header.php потрібно підключити WP меню, для цього, в файлі header.php перед закриваючим тегом </head> потрібно прописати код <?php wp\_head(); ?>, а в файлі footer.php перед закриваючим тегом </body> прописати код <?php wp\_footer(); ?>

**Підключення стилів та скриптів**

Для підключення стилів з головного файлу style.css, в файлі functions.php потрібно прописати код:

<?php

add\_action('wp\_enqueue\_scripts', 'add\_scripts\_and\_styles');

function add\_scripts\_and\_styles() {

    wp\_enqueue\_style('style', get\_stylesheet\_uri());

}

?>

Де:

add\_action – це функція, що дозволяє виконати нашу кастомну функцію в момент події, що відбувається на WordPress (такі події називаються хуками);

wp\_enqueue\_scripts – це назва хука, що виконується коли підключаються стандартні стилі і скрипти WordPress

add\_scripts\_and\_styles – це назва нашої власної функції, яка виконуватиметься після хука

wp\_enqueue\_style – це функція WordPress, що підключає стиль, в цій функції є два параметри, де 'style' - це назва, get\_stylesheet\_uri() - це функція що вказує шлях до файлу style.css не залежно від того де він знаходиться.

Водночас, в файлі header.php можна видалити строку коду, що відповідає за підключення стилів.

Також, для підключення стилів із різних бібліотек, наприклад таких як swiper.js, потрібно файли зі стилями даної бібліотеки, такі як swiper-bundle.min.css, розподілити assets/css, після чого дописати нову функцію wp\_enqueue\_style до нашого хука wp\_enqueue\_scripts:

<?php

add\_action('wp\_enqueue\_scripts', 'add\_scripts\_and\_styles');

function add\_scripts\_and\_styles() {

wp\_enqueue\_style('style', get\_stylesheet\_uri());

wp\_enqueue\_style('swiper', get\_template\_directory\_uri().'/assets/css/swiper-bundle.min.css');

}

?>

Для того щоб підключити власний скрипт, потрібно для початку створити файл main.js, після чого перенести до нього всі скрипти з index.php, після чого розподілити main.js до assets/js, після чого дописати нову функцію wp\_enqueue\_script до нашого хука wp\_enqueue\_scripts:

wp\_enqueue\_script('main', get\_template\_directory\_uri().'/assets/js/main.js', array(), null, true);

Де:

get\_template\_directory\_uri() – це функція, що повертає шлях до директорії теми, після крапки ми прописуємо повний шлях до файлу вже в самій директорії

array() – всередині можна прописати назву скрипта, лише після завантаження якого завантажиться даний скрипт

null - WordPress автоматично поставить версію теми або ти можеш вказати конкретну

true – завантажує скрипт у футері

Також, для підключення скриптів із різних бібліотек, наприклад таких як swiper.js, потрібно перенести всі скрипти, що належать даній бібліотеці з index.php до swiper-bundle.min.js після чого розподілити даний файл до assets/js, після чого дописати нову функцію wp\_enqueue\_script до нашого хука wp\_enqueue\_scripts:

wp\_enqueue\_script('swiper', get\_template\_directory\_uri().'/assets/js/swiper-bundle.min.js', array(), null, true);

**Шаблони сторінок**

Описати як додаються шаблони сторінок, додати головну сторінку в налаштуваннях як головну сторінку

**SCF**

Описати як встановлювати і активовувати плагін, як додавати текст і зображення в верстку