Создаем папку сайта на WP для OpenServer.

Переходим в папку «Openserver» > «domains» и создаем в ней папку, в которой будет установлен WordPress, например «wpfolder».

Загружаем в нее архив с CMS.

Распаковать архив. При необходимости, распакованные файлы нужно будет перенести в корневую папку (в примере – «wpfolder»), иначе сайт будет открываться не по адресу «wpfolder», а по адресу «wpfolder/wordpress»:

Создаем базу данных и пользователя в PhpMyAdmin.

Открываем PhpMyAdmin:

Откроется окно входа. По умолчанию логин – «root», пароль – отсутствует.

Нажимаем «ОК», попадаем в интерфейс PhpMyAdmin.

Переходим на вкладку «Пользователи», нажимаем «Добавить пользователя»:

В открывшемся окне выполняем следующие действия:

1. Заполняем имя пользователя - wpfolder.
2. Генерируем пароль для него
3. Отмечаем флажками поля «Создать базу данных с именем пользователя в названии и предоставить на нее полные привилегии», «Предоставить полные привилегии на базы данных подпадающие под шаблон (имя пользователя\\_%)».
4. Предоставляем пользователю возможность производить любые действия с этой базой данных (нажимаем на ссылку «Отметить все», после чего в блоках «Данные», «Структура» и «Администрирование» все поля будут отмечены флажками).
5. Сохраняем логин и пароль в текстовом файле у себя на компьютере, затем нажимаем «ОК».

Создалась база данных - wpfolder

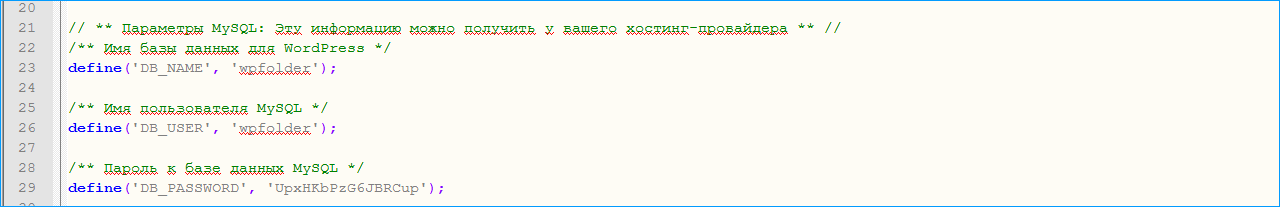
Чтобы установить, переименовываем wp-config-sample.

Следующий шаг – в папке с WordPress переименовываем конфигурационный файл. Из «wp-config-sample.php» в «wp-config.php».

Вставляем БД, логин и пароль в wp-config.

Открываем конфигурационный файл - wp-config.php

Вносим данные - название базы данных и имя пользователя



Сохраняем файл, переходим к установке WordPress.

Запускаем процесс установки WordPress.

Для запуска установки – нужно открыть сайт через OpenServer, либо ввести его в адресной строке браузера.

Откроется окно установки WordPress, где необходимо заполнить следующие поля:

1. Название сайта – SAVE WATER. Это то, что будет показано в шапке сайта.
2. Имя пользователя - Forever.
3. Пароль – Water.
4. Адрес вашей электронной почты.
5. Если флажок не установлен – в файл robots.txt будет прописана директива, запрещающая поисковым системам индексировать сайт. Снимать его нужно только в случае, если Вы знаете, что доработка и наполнение сайта займет 2-3 недели. И только спустя это время Вы готовы показать сайт посетителям.
6. Нажимаем «Установить WordPress»

Проверяем – установился ли ВордПресс.

Мы увидим окно, подтверждающее корректную установку WordPress.

Открываем пользовательскую часть сайта.

Проверяем администраторскую часть (чтобы попасть в админпанель WordPress нужно ввести логин и пароль администратора, после перехода по ссылке названиесайта/wp-login.php) - wpfolder/wp-admin

Установка темы

1. Зайдите на свою страницу администратора WordPress - wpfolder/wp-admin
2. Нажмите на меню «Внешний вид». Выберите раздел «Темы».
3. Нажмите на кнопку «Добавить новую».
4. Выберите или загрузите свою тему.
5. Откроется новая страница с тремя опциями.
   * **Просмотреть**. Эта функция позволит вам посмотреть, как выглядит тема на вашем блоге.
   * **Активировать.** Закончит установку и активирует тему.
   * **Вернуться на страницу тем**. Нажмите для возвращения на предыдущую страницу.

Установка плагина

1. Войдите в админ панель WordPress и откройте вкладку **Плагины — Добавить новый (Plugins**tab**— Add New)**:
2. Вверху нажмите кнопку **Загрузить плагин (Upload**plugin**)**.
3. Кликните **Обзор (Browse)** и выберите .zip архив плагина. Обычно, плагины, которые предоставляются в комплекте с темой CherryFramework WordPress, можно найти в папке **архив шаблона > CherryFramework > Includes > Plugins** и в папке **архив шаблона > Theme > Themexxxxx > Includes > Plugins**. Когда вы нашли необходимый архив плагина, нажмите кнопку **Открыть (Open)**, а затем кликните **Установить сейчас (Install Now)**.
4. Когда плагин будет загружен, активируйте его, кликнув по ссылке **Активировать плагин (Activate**Plugin**)**:

Расширенный синтаксис Contact Form 7

[radio paymethod **id:paynow class:paythis** "VISA" "MASTERCARD" "AMEX"]

Использование введенных данных на сервере без отправки почты

Если вы хотите что-нибудь отличное от отправки по почте, вам нужно использовать следующий код в файл wp-content\plugins\contact-form-7\includes --> functions.php

add\_action( 'wpcf7\_before\_send\_mail', 'my\_custom\_function' );

function my\_custom\_function($cf7) {

Запись в БД

global $wpdb;

$table = $wpdb->prefix.'reg\_user';

$data = array("reg\_name" => $\_POST['your-name'], "reg\_email" => $\_POST['your-email']);

$format = array('%s', '%s');

$wpdb->insert($table, $data, $format);

//$wpdb->show\_errors();

}

Стили

.wpcf7 input[type="text"] {

background-color: #fff;

color: #000;

width: 50%; }

или

[text text-123 id:very-special-field]

#very-special-field

{

color: #f00;

border: 1px solid #f00;

}

### Количество зарегистрированных пользователей

Следующий код в файл wp-content\themes\имя\_темы --> functions.php

// Function to return user count

function wpb\_user\_count() {

$usercount = count\_users();

$result = $usercount['total\_users'];

return $result;

}

// Creating a shortcode to display user count

add\_shortcode('user\_count', 'wpb\_user\_count');

Следующий код в файл wp-content\themes\имя\_темы 🡪 footer.php

<?php echo "<p>Количество <strong>".do\_shortcode( '[user\_count]' )."</strong> зарегистрированных пользователей</p>"; ?>

Количество записей в таблице

в файл wp-content\themes\имя\_темы --> functions.php

function wpb\_reg\_user\_count() {

global $wpdb;

$reg\_user\_count = $wpdb->get\_var( "SELECT COUNT(\*) FROM wp\_reg\_user" );

return $reg\_user\_count;

}

add\_shortcode('reg\_user\_count', 'wpb\_reg\_user\_count');

или на странице

<?php

function get\_reg\_users() {

  global $wpdb;

  return $wpdb->get\_var("select count(\*) from $wpdb-> wp\_reg\_user");

}

print("Число зарегистрированных участников: ".get\_reg\_users());

?>

в файл wp-content\themes\имя\_темы 🡪 footer.php

Обновление страницы

<script type="text/javascript" src="js/jquery.min.js"></script>

<script type="text/javascript">

function mode() {

$.ajax({

url: 'function.php',

success: function(data) {

$('#reg\_user').html(data);

}

});

}

var timeInterval = 100;

setInterval(mode, timeInterval);

</script>

<div id='reg\_user'>

<?php echo "<p>Количество пользователей программы <strong>".do\_shortcode( '[reg\_user\_count]' )."</strong></p>"; ?>

</div>

### Перезагрузка страницы

jQuery('input[type=submit]').click(function() {

setTimeout(function () {

document.location.reload(true);

}, 3000);

})

### Изменить логотип админки

Изначально логотип 84х84

Идем в папку wp-admin -> css -> файл login.css

Заходим в него и ищем строку: .login h1 a

В этой строке ищем 2 свойства: width(ширина) и height(высота). Задаем их такими же, как и размеры логотипа.

Открываем файл functions.php, что находится в папке с нашей текущей темой.

function my\_login\_logo(){

echo '

<style type="text/css">

#login h1 a { background: url("wp-content/themes/images/logo.png") no-repeat 0 0 !important; }

</style>';

}

add\_action('login\_head', 'my\_login\_logo');

add\_filter( 'login\_headerurl', create\_function('', 'return get\_home\_url();') );

add\_filter( 'login\_headertitle', create\_function('', 'return false;') );

### Вывод меню Landing page

Открываем файл functions.php, что находится в папке с нашей текущей темой.

Массив из id страниц меню

function get\_page\_id\_array () {

$locations = get\_nav\_menu\_locations();

$menu = wp\_get\_nav\_menu\_object($locations['primary']);

$menu\_items = wp\_get\_nav\_menu\_items($menu->term\_id);

$page\_id\_array = array();

foreach ( (array) $menu\_items as $key => $menu\_item ) {

array\_push( $page\_id\_array, $menu\_item->object\_id);

}

return $page\_id\_array;

}

В файле wp-content\themes\...\header.php

Вывод заголовков страниц в меню

<?php $query=new WP\_Query(array('post\_type'=>'page', 'post\_\_in'=> get\_page\_id\_array(), 'orderby'=>'post\_\_in'));

while ( $query->have\_posts() ) {

$query->the\_post(); ?>

<a class="navbar-brand" href="#block<?php the\_ID();?>" rel="home">

<?php the\_title(); // выведем заголовок страницы ?>

</a>

<?php }

?>

Плавный переход по якорям

В файле wp-content\themes\...\header.php

<script type="text/javascript" src="https://ajax.googleapis.com/ajax/libs/jquery/1.10.1/jquery.min.js"></script>

<script type="text/javascript">

$(document).ready(function(){

$('a[href^="#"]').bind('click.smoothscroll',function (e) {

e.preventDefault();

var target = this.hash,

$target = $(target);

$('html, body').stop().animate({

'scrollTop': $target.offset().top

}, 600, 'swing', function () {

window.location.hash = target;

});

});

});

</script>

Фиксация меню вверху страницы

<nav class="navbar-fixed-top">

### Вывод содержимого блоков (страниц) LP

В файле wp-content\themes\...\index.php

<?php $query=new WP\_Query(array('post\_type'=>'page', 'post\_\_in'=> get\_page\_id\_array(), 'orderby'=>'post\_\_in'));

while ( $query->have\_posts() ) {

$query->the\_post(); ?>

<div id="block<?php the\_ID();?>"></div>

<?php the\_title(); // выведем заголовок поста

?>

<?php the\_content(); // выводим содержимое ?>

<?php }

?>

jQuery('input[type=submit]').click(function() {

//e.preventDefault();

jQuery.ajax({

// type: "POST",

url: 'function.php',

success: function(data) {

$('#reg\_user').html(data);

}

});

})

<!-- <script type="text/javascript">

function mode() {

$.ajax({

url: 'http://localhost/wpfolder/',

success: function(data) {

$('#reg\_user').html(data);

}

});

}

var timeInterval = 100;

setInterval(mode, timeInterval);

</script> -->

### Изменение темы twentyseventeen

Навигация

В меню Произвольная ссылка 🡪 # block2

Добавить ID в заголовок страницы с помощью редактора WP

<span style='padding-top: 190px' id='block2'>Блок 2</span>

Убираем надпись про WordPress из футера

Открыть файл footer.php в теме и удалить строку:

get\_template\_part( ‘template-parts/footer/site’, ‘info’ );

Меняем картинки

Редактировать страницу – Изображение страницы

Редактируем виджет Мета WordPress

в папке wp-includes\widgets -> class-wp-widget-meta.php

**Создаем Область Для Виджетов**

wp-content\themes\имя\_темы --> functions.php

function wpb\_widgets\_init() {

register\_sidebar( array(

'name' => 'Произвольная область',

'id' => 'custom-header-widget',

'before\_widget' => '<div class="chw-widget">',

'after\_widget' => '</div>',

'before\_title' => '<h2 class="chw-title">',

'after\_title' => '</h2>',

) );

}

add\_action( 'widgets\_init', 'wpb\_widgets\_init' );

wp-content\themes\имя\_темы --> header.php

<?php

if (is\_active\_sidebar( 'custom-header-widget' ) ) : ?>

<div id="header-widget-area" class="chw-widget-area widget-area" role="complementary">

<?php dynamic\_sidebar( 'custom-header-widget' ); ?>

</div>

<?php endif; ?>

### Landind page

В файле wp-content\themes\...\index.php

Вывод всех страниц

<?php $query=new WP\_Query(array('post\_type'=>'page', …));

while ( $query->have\_posts() ) {

$query->the\_post(); ?>

<div id="block<?php the\_ID();?>"></div>

<?php the\_title(); // выведем заголовок поста

?>

<?php the\_content(); // выводим содержимое ?>

<?php }

?>

В файле wp-content\themes\...\header.php

<?php $query=new WP\_Query(array('post\_type'=>'page', …));

while ( $query->have\_posts() ) {

$query->the\_post(); ?>

<a class="navbar-brand" href="#block<?php the\_ID();?>" rel="home">

<?php the\_title(); // выведем заголовок поста ?>

</a>

<?php }

?>

Вывод страниц с определенными номерами

$query=new WP\_Query(array('post\_type'=>'page','post\_\_in'=>array(2,5,12,14,20)));

Вывод только 1 страницы

$id = 1; // id страницы

$post = get\_page($id);

$content = $post->post\_content;

echo $post->post\_content;

Получение id пунктов меню

<?php

$locations = get\_nav\_menu\_locations();

$menu = wp\_get\_nav\_menu\_object($locations['primary']);

$menu\_items = wp\_get\_nav\_menu\_items($menu->term\_id);

foreach ( (array) $menu\_items as $key => $menu\_item ) {

echo $menu\_item->object\_id." "; // ...

}

?>

* $menu\_item->url — адрес страницы, добавленной в меню
* $menu\_item->title — заголовок пункта меню (по умолчанию совпадает с заголовком страницы)
* $menu\_item->object\_id — идентификатор страницы.

\_\_\_\_\_

<?php query\_posts(array('showposts' => 10, 'post\_parent' => 149,

'post\_type' => 'page', 'orderby' => 'menu\_order')); ?>

<?php while (have\_posts()) : the\_post(); ?>

<div>

<?php if ( function\_exists("has\_post\_thumbnail") && has\_post\_thumbnail() )

{ the\_post\_thumbnail('large-thumbnail'); } else {echo ('<img src="' .

get\_bloginfo('template\_directory') . '/images/demo/piecemaker/960x360.gif"

width="960px" height="360px" />'); } ?> //Если миниатюра есть - выводим миниатюру, иначе показываем заглушку

<div><a href="<?php the\_permalink(); ?>"><?php the\_title(); ?></a></div>

</div>

<?php endwhile;?>

<?php wp\_reset\_query(); ?>

Выводит все страницы на экран

<?php query\_posts(array('showposts' => 10, 'post\_parent' => 0, 'post\_type' => 'page', 'orderby' => 'menu\_order')); ?>

<?php while (have\_posts()) : the\_post(); ?>

</div>

<?php get\_template\_part( 'content', get\_post\_format() ); ?>

<?php endwhile;?>

<?php wp\_reset\_query(); ?>

<?php if ( have\_posts() ) : ?>

<?php /\* Start the Loop \*/ ?>

<?php while ( have\_posts() ) : the\_post(); ?>

<?php

/\* Include the Post-Format-specific template for the content.

\* If you want to overload this in a child theme then include a file

\* called content-\_\_\_.php (where \_\_\_ is the Post Format name) and that will be used instead.

\*/

get\_template\_part( 'content', get\_post\_format() );

?>

<?php endwhile; ?>

<?php \_tk\_content\_nav( 'nav-below' ); ?>

<?php else : ?>

<?php get\_template\_part( 'no-results', 'index' ); ?>

<?php endif; ?>