# От BEONMAX Заголовок и логотип

Обновление страницы в браузере со сбросом кеша: **Shift – F5**

В свойствах сайта настраиваем название сайта и описание. Логотипа там нет.

add\_theme\_support( 'custom-logo' ); - регистрирует поддержку новых возможностей темы в WordPress (поддержка миниатюр, форматов записей и т.д.). Если функция вызывается из темы, то она должна находится в файле functions.php

Рассмотрим случай, когда **логотип** – ссылка с картинкой внутри. В header.php удаляем строки с картинкой и текстом логотипа и заменяем их кодом:

<?php the\_custom\_logo(); ?>

После этого в админ панели выбираем в свойствах сайта файл с логотипом и настраиваем его.

В коде header.php исправляем обертку для логотипа – тег a заменяем на div:

<div class="header\_\_logo">

<?php the\_custom\_logo(); ?>

</div>

Другой вариант – получаем только ссылку на логотип:

В коде header.php:

<a href="<?php echo get\_home\_url(); ?>" class="header\_\_logo">

<img src="<?php

$custom\_logo\_\_url = wp\_get\_attachment\_image\_src( get\_theme\_mod( 'custom\_logo' ), 'full' );

echo $custom\_logo\_\_url[0];

?>" alt="Мир детства" class="header\_\_logo-img">

</a>

Код взяли из документации по **get\_custom\_logo**, подставили в адреc картинки и добавили туда прежний класс header\_\_logo-img. Также в ссылке в href подставляем вместо # функцию **get\_home\_url().**

Разбираемся с **заголовком** и **кратким описанием**:

Сделаем их динамически. В header.php

<title><?php bloginfo('name'); echo" | "; bloginfo('description'); ?></title>

Аналогично можно использовать **bloginfo** для вывода информации в любом месте сайта.

# Меню

В function.php активируем:

add\_theme\_support( 'menus' );

Документация: <https://wp-kama.ru/function/wp_nav_menu>

В header.php в строке меню перед ul:

<?php

wp\_nav\_menu( [

'menu' => 'Main', //название

'container' => false, // контейнер

'menu\_class' => 'header\_\_nav', //взять из вёрстки

'echo' => true, // выводить меню в HTML

'fallback\_cb' => 'wp\_page\_menu',

'items\_wrap' => '<ul class="header\_\_nav">%3$s</ul>', // класс взять из вёрстки

'depth' => 1, // уровень вложенности меню ] );

?>

Далее, в верстке удаляем меню <ul>.

В Wordpress *Внешний вид – Настроить – Меню – Создать новое меню*. Называем его, как в коде *Main*. Нажимаем *Далее – Добавить элементы*. Добавляем в нужном порядке. Поставить галочку в *Автоматически добавлять в это меню новые страницы верхнего уровня*. Донастроить страницы.

Добавим класс для пунктов меню. В function.php:

add\_filter('nav\_menu\_link\_attributes', 'filter\_nav\_menu\_link\_attributes', 10, 3);

function filter\_nav\_menu\_link\_attributes($atts, $item, $args) {

if ($args->menu === 'Main') {

$atts['class'] = 'header\_\_nav-item';

if ($item->current) {

$atts['class'] .= ' header\_\_nav-item-active';

};

return $atts;

}

где *'nav\_menu\_link\_attributes'* - название хука, $*atts* - ссылки, $*item* – отдельная ссылка, $*args* – аргументы из *wp\_nav\_menu*.

print\_r($item); - вывести массив ссылок меню.