**Практическая работа «Интеграция WordPress в сайт, разработанный на языке программирования PHP»**

Целью данной практической работы является изучение методов интеграции CMS WordPress в готовый сайт, разработанный на языке программирования PHP, и их практическое применение.

# 1. Теоретические сведения

Существует несколько особенностей CMS WordPress, которые можно использовать при интеграции его с php-сайтом, или вы можете изменить весь сайт под WordPress.

Для отображения записей в WordPress используется цикл. С его помощью, WordPress обрабатывает каждую из записей для вывода на текущей странице и форматирует ее в соответствии с указанными критериями внутри цикла. Любой HTML или PHP код, расположенный внутри цикла, будет повторен для каждой записи.

В WordPress возможны четыре построения циклов для вывода записей:

1. Стандартный цикл WordPress;
2. Цикл WordPress на основе query\_posts();
3. Цикл WordPress на основе Wp\_Query();
4. Цикл WordPress на основе get\_posts().

**Стандартный цикл WordPress**.

Стандартный цикл WordPress это цикл с предусловием (оператор PHP while).

В стандартном цикле мы не указываем данные для выборки записей. При входе в цикл вычисляется условие и, если его значение отлично от нуля, выполняется тело цикла.

<?php

if ( have\_posts() ){

while ( have\_posts() ){

the\_post();

// код вывода

}

} else {

echo wpautop( 'Постов для вывода не найдено.' );

}

?>

Пример:

<?php if ( have\_posts() ) {

while ( have\_posts() ) { the\_post(); ?>

<!-- Вывод постов -->

<h2><?php the\_title() ?></h2>

<?php } }

else { ?>

<p>Записей нет.</p>

<?php } ?>

При каждом повторении с переменной $post будет считываться информация о посте и выводится на экран, через теги шаблона: the\_title(), the\_content()...

Теги шаблонов при выводе постов:

<?php the\_ID(); ?> // выводит ID текущего поста.

<?php the\_title(); ?> // вывод заголовка поста

<?php the\_title\_attribute(); ?> //Выводит заголовок записи подготовленный для использования в атрибутах html тега

<?php the\_permalink() ?> // ссылку на пост

<?php the\_time('F jS, Y') ?> // выводим дату

<?php the\_author\_posts\_link() ?> // выводим ссылки на другие посты автора

**2. Цикл WordPress на основе query\_posts()**

Функция query\_posts() позволяет изменить базовый запрос и вывести нужный нам вариант записей. query\_posts() не предназначена для создания нового цикла, а только для изменения стандартного цикла.

<?php

// Запрос -The Query

query\_posts( $args );

// The Loop-стандартный цикл

while ( have\_posts() ) : the\_post();

    echo '<li>';

    the\_title();

    echo '</li>';

endwhile;

// Reset Query- сброс (остановка) query

wp\_reset\_query();

?>

Параметры Категорий (рубрик)

Получает посты относящиеся к определенным категориям.

cat(число/строка/массив)

ID категории. Можно указать несколько ID в строке через запятую или в массиве. Чтобы исключить рубрики укажите минус (-) перед ID категории - это исключит и все вложенные рубрики. Чтобы не исключить вложенные, используйте параметр category\_\_not\_in.

category\_name(строка)

Имя категории. Используйте слаг (альтернативное имя), а не само название категории.

category\_\_and(массив)

Получить посты, которые входят одновременно в несколько категорий.

category\_\_in(массив)

Получить посты, которые входят в одну из указанных категорий.

category\_\_not\_in(массив)

Получить посты, которые не входят в указанную категорию.

Примеры использования:

Вывод постов из одной категории (записи из дочерних категорий также будут выбраны):

query\_posts('cat=4');

Вывод постов из одной категории по ярлыку (альтернативному названию категории):

query\_posts('category\_name=staff');

Выведем посты только из категории 4 (дочерние категории не затрагиваются):

query\_posts( 'category\_\_in=4' );

Вывод постов из категорий по ID категорий:

query\_posts('cat=2,6,17,38');

Показать посты, которые находятся сразу в двух категориях:

query\_posts( array( 'category\_\_and' => array(2,6) ) );

Показать посты, которые находятся хотя бы в одной из категорий (дочерние категории не будут учитываться):

query\_posts(array('category\_\_in' => array(2,6)));

Можно исключить несколько категорий, таким образом:

query\_posts( array('category\_\_not\_in' => array(2,6)) );

**3. Цикл WordPress на основе WP\_Query()**

Цикл WordPress на основе класса WP\_Query() используется для вывода записей, которые никак не связаны со страницей или для создания множественных циклов. Выглядят эти циклы аналогично циклам с использование функции query\_posts(). Параметры для WP\_Query() такие же.

Особенностью циклов на основе WP\_Query() является создание нового объекта ($query), который не связан с аналогичным глобальным объектом $wp\_query и поэтому никак не нарушается структура текущей страницы.

Например, мы можем получить посты:

* за определенный промежуток времени;
* из указанной категории, метки;
* свежие посты, случайные посты, популярные посты;
* посты с указанными произвольными полями или набором таких полей.

Пример цикла WP\_query():

<?php

$query = new WP\_Query ( [ 'category\_name' => 'news' ]  );

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

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

$query->the\_post();

the\_title(); // выведем заголовок поста }

} else { // Сообщения не найдены }

/\* Восстановления исходных данных сообщение \*/

wp\_reset\_postdata();

**4. Цикл WordPress на основе get\_posts()**

Получает записи (посты, страницы, вложения) из базы данных по указанным критериям. Можно выбрать любые посты и отсортировать их как угодно.

Цикл WordPress на основе get\_posts() считается самым удобным вариантом вывода нужных записей в нужном порядке. get\_posts() в большинстве случаев полностью заменяет функцию WP\_query(). Например,

если нужно вывести 7 последних постов в сайдбаре или

несколько случайных записей в футере — нет проблем;

нужно вывести все картинки, прикрепленные к посту или

вывести записи с определенным произвольным полем

Функция get\_posts() аналогично с query\_posts() работает на основе класса WP\_query() и поэтому передаваемые параметры одинаковые.

Шаблон использования:

$my\_posts = get\_posts( array(

'numberposts' => 5,

'category'    => 0,

'orderby'     => 'date',

'order'       => 'DESC',

'include'     => array(),

'exclude'     => array(),

'meta\_key'    => '',

'meta\_value'  =>'',

'post\_type'   => 'post',

'suppress\_filters' => true, // подавление работы фильтров изменения SQL запроса

) );

**2.** **Ход выполнения работы**

1. Подключение WordPress к сайту

Зайдем в корневую папку вашего php-сайта на локальном сервере.

Затем создадим новый подкаталог (папку) на вашем сайте и назовём его 'blog' (вы можете использовать другое имя, нежели blog, но вы должны создать этот подкаталог). Так, теперь у вас есть пустой подкаталог в http://localhost/blog/. Теперь скачаем WordPress и загрузим все файлы в эту новую папку, и затем - установим его WordPress.

Для того чтобы трансформировать обычные страницы PHP в те, которые используют WordPress, вам нужно добавить один из следующих фрагментов кода в начало каждой страницы.

<?php

/\* Short and sweet \*/

define('WP\_USE\_THEMES', false);

require('./wp-blog-header.php');

?>

<?php

require('/the/path/to/your/wp-blog-header.php');

?>

1. Вывод постов
   1. В случае, если вы хотите показать десять сообщений в алфавитном порядке, в порядке возрастания на веб-странице, для захвата даты, названия и выдержки из поста вы можете сделать следующее:

<?php

require('/the/path/to/your/wp-blog-header.php');

?>

Выводим все посты:

<?php

$posts = get\_posts('numberposts=10&order=ASC&orderby=post\_title');

foreach ($posts as $post) : setup\_postdata( $post ); ?>

<?php the\_date(); echo "<br />"; ?>

<?php the\_title(); ?>

<?php the\_excerpt(); ?>

<?php

endforeach;

?>

* 1. Выводим последние три поста на веб-странице.

<?php

global $post;

$args = array( 'posts\_per\_page' => 3 );

$myposts = get\_posts( $args );

foreach( $myposts as $post ) : setup\_postdata($post); ?>

<a href="<?php the\_permalink() ?>" rel="bookmark" title="Permanent Link to <?php the\_title\_attribute(); ?>"><?php the\_title(); ?></a><br />

<?php endforeach; ?>

Цикл используется в WordPress для отображения записей. Используя цикл, WordPress обрабатывает каждую из этих записей для вывода на текущей странице и форматирует ее в соответствии с указанными критериями внутри цикла. Любой HTML или PHP код, расположенный внутри цикла, будет повторен для каждой записи.

Например, в Цикле, по умолчанию, для каждой записи выводится следующая информация:

* Заголовок (the\_title())
* Время (the\_time())
* Рубрики (the\_category())

Другую информацию о каждой записи можно отобразить с помощью соответствующих тэгов шаблона или (для продвинутых пользователей) с помощью введения переменной $post, которая содержит информацию текущей записи, пока Цикл выполняется.

Цикл должен быть расположен в файле index.php и в любых других шаблонах, используемых для отображения записей.

Убедитесь, что включили вызов заголовка Шаблона (header.php), в верхней части Шаблонов вашей Темы. Если вы используете Цикл внутри собственного дизайна (и ваш дизайн не является шаблоном), установите WP\_USE\_THEMES в значение false.

<?php define('WP\_USE\_THEMES', false); get\_header(); ?>

Цикл стартует здесь:

Вывод всех постов циклом while:

<?php

$args = array( );

query\_posts( $args );

if ( have\_posts() )

{

    while ( have\_posts() )

    {   the\_post();

    ?>

    <h2><a href="<?php the\_permalink(); ?>"><?php the\_title(); ?></a></h2>

    <?php }

}

else { ?>

    <p><?php \_e('Sorry, no posts matched your criteria.'); ?></p>

<?php } ?>

* 1. Стилизация Записей из различных Рубрик.Этот пример отображает каждую запись с заголовком (который используется в качестве Постоянной ссылки к записи), Рубриками и Содержимым. Это очень простой пример; вероятно ваши Шаблоны будут отображать больше информации, в таком случае стилизацию будет проще сделать с помощью CSS.

Для того чтобы извлечь больше пользы из этого примера, выделим записи из Рубрики с ID равным '3'. Для этого используется Тэг Шаблона in\_category().

Вывод постов с категорией 3 циклом while:

<?php

$args = array( );

query\_posts( $args );

if ( have\_posts() )

{

    while ( have\_posts() )

    {   the\_post();

        if ( in\_category(3)){

    ?>

    <h2><a href="<?php the\_permalink(); ?>"><?php the\_title(); ?></a></h2>

    <?php }

}}

else { ?>

    <p><?php \_e('Sorry, no posts matched your criteria.'); ?></p>

<?php } ?>

* 1. Исключение Записей из некоторых Рубрик

Этот пример может быть использован для исключения определенной Рубрики из обычного отображения. Он основан на предыдущем примере.

Исключение записей с категорией 3

<?php

$args = array( );

query\_posts( $args );

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

 <?php if (in\_category('3')) continue; ?>

  <div class="post">

   <h2><a href="<?php the\_permalink(); ?>"><?php the\_title(); ?></a></h2>

   <small><?php the\_time('F jS, Y'); ?></small>

   <div class="entry">

    <?php the\_content(); ?>

  </div>

  <p class="postmetadata">Posted in <?php the\_category(', '); ?></p>

 </div>

 <?php endwhile; else: ?>

 <p>Sorry, no posts matched your criteria.</p>

 <?php endif; ?>

Если вы используете этот пример для основной страницы, вы должны использовать другой шаблон для архивов рубрик. Иначе, WordPress будет исключать все записи в рубрике 3, каждый раз, когда вы просматриваете архив этой рубрики!

Однако, если вы хотите использовать этот же файл шаблона, вы должны обойти это используя тэг is\_home():

...

<?php if (in\_category('3') && is\_home() ) continue; ?>

...

Благодаря этому записи из рубрики 3 будут исключены только с главной страницы. существуют другие условные теги, которые можно использовать для управления выводом в зависимости от того выполняется ли определенное условие на запрашиваемой странице.

Несмотря на то что запись не отображается, WordPress все равно ее считает, как если бы она отображалась - это значит, что, если вы указали WordPress выводить максимум семь записей, и две, последние из семи находятся в рубрике 3, на главной странице отобразятся только пять записей. Если вы столкнулись с такой проблемой, можно воспользоваться более сложным приемом или вы можете использовать query\_posts, если вам нужно просто исключить одну рубрику из цикла.

**3.** **задания для самостоятельной работы**

Разработайте страницы Books, Movies, Music таким образом, чтобы на них выводились записи с соответствующей меткой.