**Для перевода языка урок 16**

**Esc\_html\_\_(‘text’, ‘переводчк’)** – возвращает переведенный текст

**Esc\_html\_e(‘text’, ‘переводчк’)** – выводит переведенный текст

**Отдельно шаблон с именем single.php и page.php** нужны, вордпресс сам их вставляет где надо

**Have\_posts()** – проверка на наличие постов

**The\_post()** – вывод постов

**The\_permalink()** – в ссылках в атрибут href вписать эту функцию для заголовков, чтоб вели на страницу из цыкла

**The\_content()** – вывод текста записи

**The\_author()** – вывод автора

**The\_date\_xml()** – вывод текущей даты

**The\_title()** – заголовок

**the\_time('j F Y')** – время записи поста

**the\_author\_posts\_link();** - ссылка на страницу автора

**the\_custom\_logo()** – выбранный логотип (если в настройках указать, что логотип будет выбираться)

**the\_post\_thumbnail();** - медиа файл поста

**the\_excerpt();** - обрезанный текст поста ( 55 слов )

**echo wp\_get\_document\_title();** - возвращает название сайта

**wp\_list\_pages()** – вывод страниц сайта ( главная, акции, о нас … )

**wp\_list\_pages('title\_li=')** – вывод тех же сраниц, но обернуых в тег ли и если вписать чтото после =, то этот текст будет выведен перед списком

**register\_sidebar(array('name'=>’…’))** – добавление управления к сайту ( в внешний вид => виджеты )

**Comments\_popup\_link(‘no comm.’, ‘1 comm.’, ‘% comm.’)** – вывод комментариев ( 1- если их нету, 2- если он 1, 3- если их много)

**Стандартные подключения**

**<body <?php body\_class(); ?> >** - чтоб админ меню не наезжало на контент

И в css **body.logged-in .navbar-fixed-top{ top: 30px; }**

**get\_header()** – вставка заголовка

**get\_sidebar()** – вставка сайдбара

**get\_footer()** – вставка футера

**wp\_footer()** – подключить меню wordpress

**wp\_header()** – заголовок со стилями

**get\_search\_form()** – окошко поиска

**Вывод постов**

**If(have\_posts()) { while(have\_posts()) { The\_post(); } }**

**Подключение списка страниц**

**Главная, о нас, блог …**

**wp\_nav\_menu(array(**

**'theme\_location' => 'primary',** // имя локации (указано в register\_nav\_menu )

**'items\_wrap' => '<ul id="%1$s" class="%2$s">%3$s</ul>',** // обертка **'menu\_class' => 'nav navbar-nav',** // классы

**'menu\_id' => '',** // id

**'depth' => 1** // кол-во подменю

**));**

**Подключение скриптов и стилей**

В файле functions.php создать функцию, в которой будут вызваны подключения

стилей и скриптов, после функцию нужно вызвать через

**add\_action('wp\_enqueue\_scripts','название функции');**

**get\_stylesheet\_uri()** – путь к файлу style.css

**wp\_enqueue\_style('…', get\_stylesheet\_uri())** – подключение стилей ( 1- хендлер( уникальное название ) 2- путь к файлу)

**Функции**

**get\_template\_directory\_uri()** – текущая директория

**get\_post\_format()** – формат поста ( подключаеться в Add\_action(‘after\_setup\_theme’,’…’), точнее это типы как картинка, видео , музыка и т.д. )

**bloginfo('template\_url')** – текущий адрес

функция, имя которой будет подключено в **Add\_action(‘after\_setup\_theme’,’…’)** имеет настройки :

* **load\_theme\_textdomain('…', get\_template\_directory() . ‘…’)** – домен для мультиязычности и путь к папке с языками
* **register\_nav\_menu('имя локации','название, которое будет отображаться');** - добавить в раздел внешний вид раздел меню
* **add\_theme\_support('title-tag')** – позволить темам и плагинам изменять тег title
* **add\_theme\_support( 'custom-logo', array('height' => … , 'width' => … , 'flex-height' => true ));** - добавление функции выбора логотипа с указаниями стандартных параметров)
* **add\_theme\_support('post-thumbnails'); + set\_post\_thumbnail\_size(…, …);** - добавить картинку к посту
* **add\_theme\_support('html5', array( 'search\_form', 'comment-form', 'comment-list', 'gallery', 'caption' ));** - добавление форм
* **add\_theme\_support('post-formats', array( 'aside', 'image', 'video', 'gallery' ));** - добавление формаов
* **add\_theme\_support('menus');** - добавить в раздел внешний вид раздел меню

функция, имя которой будет подключено в **Add\_action(‘customize\_register’,’…’)** имеет настройки :

* **$wp\_customize->add\_setting( '…', array(** - добавление секции в определенный пункт меню настроек страницы

**'default' => \_\_('…', ‘…’),** - значенме по умолчанию

**'transport' => 'refresh', ));** - функция обновления при внесении изменений

* **$wp\_customize->add\_control(** - добавление или создания нового меню настроек

**'…',** - название того, что добавляется ( определено в функции выше )

**array(**

**'label' => \_\_( 'Название отображаемого окошка', 'переменная для перевода' ),** - подпись строки в меню

**'section' => 'имя пункта в меню',** - создание нового пункта или добавление в существующий

**'settings' => '…',** - название того, что добавляется ( определено в функции выше )

**'type' => 'text', ) );** - тип

* **echo get\_theme\_mod('…')** – вывод того, ч добавлено выше

функция, имя которой будет подключено в **Add\_action(‘widgets\_init’,’…’)** имеет настройки :

* **function …() {** - добавление виджта напримей sidebar

**register\_sidebar( array(**

**'name' => \_\_( 'имя', 'переводчик' ),** - имя виджета ( в меню вид –> виджеты )

**'id' => '…',** - id с помощью которого будет вызываться функция вывода

**'description' => \_\_( '…', 'переводчик'),**

**'before\_widget' => '<… id="%1$s" class="widget %2$s">',** - основная обертка сайдбара

**'after\_widget' => '</…>',** - закрытие обертки

**'before\_title' => '<h2 class="…">',** - перед заголовком

**'after\_title' => '</h2>',** - после

**) );**

* **function wptuts\_register\_widget() { register\_widget('имя класса'); }** – добавление раздела
* **dynamic\_sidebar('…');** - вывод сайдбара по его id

для перестилизации сайдбара

**function wptuts\_widget\_categories($args){**

**$walker = new Walker\_Categories\_Wptuts();**

**$args = array\_merge($args, array('walker' => $walker));**

**return $args;**

**}**

**add\_filter('widget\_categories\_args', 'wptuts\_widget\_categories');**

**class Walker\_Categories\_Wptuts extends Walker\_Category { скопировать класс и внести изменения }**

**Хуки**

**Add\_action(‘…’,’…’)** – добавить функцию под одну группу ( 1- название группы, 2- функция, которая входит без параметров) 3 параметром может идти параметр приоритетов

**Do\_action(‘…’)** – вызов группы функций

**Apply\_filters(‘…’, …)** – создать группу фильтров для переменной (1- название, 2- переменная) (вызывать выполнение не нужно, оно сразу)

**Add\_filter(‘…’,’…’,)** – добавить функцию в фильтр ( 1- название фильтра, 2- функция, которая входит без параметров)

**Remove\_action(‘…’) or Remove\_action(‘…’,’…’)** – удалить группу или функцию в группе

**Remove\_ filter(‘…’) or Remove\_ filter(‘…’,’…’)** – удалить фильтр ил его часть

**Add\_action(‘…’,’…’)** – имеет константы:

* + - **after\_setup\_theme** – подключение функции с возможностями вордпресса
    - **wp\_enqueue\_scripts** – подключение силей и скриптов
    - **customize\_register** – добавление пункта в меню настроек (в функции принимающийся параметр **$wp\_customize** )
    - **widgets\_init** – добавление пункта в виджетах

**<!--vitrina ярлык столбцы высота строки текста\_в\_заголовке-->** - управление выводом текста в витрине

( главной стр.) и код вводиться в страницах в текстовом поле

**<?php if(!function\_exists('dynamic\_sidebar') || !dynamic\_sidebar('имя рубрики')) : ?>**

**<p>текст при отсутствии</p>**

**<?php endif; ?>**  - вывод категорий

**Home\_url(“/”)** – ссылка на базовую страницу