

# Конспект лекцій з дисципліни “Мультиплатформені програмні засоби мультимедіа”

---

## Модуль 1. Мультиплатформені програмні засоби мультимедіа для веб-ресурсів

### Тема 1. Системи управління контентом

#### Лекція 1. Системи управління контентом. Знайомство із Wordpress

Веб-система управління контентом (WCMS) [1] – програмне забезпечення, яке забезпечує сайт авторські роботи, спільні роботи і засоби адміністрування, призначені для забезпечення користувачів з невеликим знанням мережі мова програмування S і мова розмітки и, щоб створювати і керувати вмістом веб-сайту з відносною легкістю. Надійна веб-система управління контентом забезпечує основу для співпраці, пропонуючи користувачам можливість управління документами і вихід для декількох редагування та участі автора.

Більшість систем використовують Content Repository або базу даних до вмісту сторінці магазину, метадані та інших інформаційних ресурсів, які могли б бути необхідні в системі.

Презентація шар (шаблон двигун) відображає вміст на сайт відвідувачів, заснованих на наборі шаблони, які іноді XSLT файли [2].

Більшість систем можна використовувати серверного кешування, щоб підвищити продуктивність. Це працює найкраще, коли WCMS не змінюється часто, але відвідування трапляються регулярно.

Адміністрація також зазвичай робиться через інтерфейси на основі браузера, але деякі системи вимагають використання товстий клієнт.

WCMS дозволяє нетехнічних користувачів вносити зміни в веб-сайт з невеликої підготовки. WCMS як правило, вимагає системного адміністратора та / або веб-розробника для налаштування і додавати нові функції, але це в першу чергу веб-сайт зміст інструментом для проведення технічного персоналу.

Система управління веб-контентом використовується для керування динамічною колекція веб-матеріалу, в тому числі веб-документів, Image S та

інших засобах масової інформації [3]. CMS полегшує контроль документів, аудит, редагування і керування тимчасової шкали. WCMS, як правило, має такі особливості:

#### Автоматизовані шаблони

Створіть стандартні шаблони виводу (зазвичай HTML і XML), які можуть бути автоматично застосовані до новим та існуючим змістом, дозволяючи поява всього змісту повинна бути змінена з одного центрального місця.

Контроль доступу: Деякі групи підтримки користувачів системи WCMS. Групи користувачів дозволяють управляти як зареєстрований користувачі взаємодіють з сайтом. Сторінки на сайті може бути обмежене однією або декількома групами. Це означає, анонімний користувач (хтось не увійшли на), або користувач, не входить до групи сторінках, вони обмежені, буде відмовлено в доступі до сторінки.

#### Масштабування

Доступне в найсучасніші WCMSs є можливість розширити одну реалізацію (одну установку на одному сервері) в декількох областях, в залежності від налаштувань сервера. WCMS сайти можуть бути в змозі створити мікросайту с / веб-порталу в головному сайті.

#### Легко редаговані зміст

Після того, як вміст відділяється від візуального представлення сайту, як правило, стає набагато легше і швидше, щоб редагувати і маніпулювати. Велика частина програмного забезпечення WCMS включає в себе WYSIWYG Інструменти редагування, що дозволяють нетехнічних користувачів для створення і редагування контенту.

Масштабовані набори функцій: Більшість WCMS програмне забезпечення включає в себе плагіни або модулі, які можуть бути легко встановлені для розширення функціональності існуючого сайту.

#### Модернізація веб-стандартів

Активні WCMS програмне забезпечення, як правило, отримує регулярні оновлення, які включають в себе нові набори функцій і підтримувати систему на поточну веб-стандартів.

#### Workflow менеджмент

робочий є процес створення циклів послідовних і паралельних завдань, які повинні бути виконані в CMS. Наприклад, один або кілька творців контенту може уявити історію, але він не опублікований до літературний Редактор не очищає його і редактор головного затверджує його.

#### Взаємодія

CMS програмне забезпечення може виступати як співпраця платформи дозволяючи зміст яких будуть витягнуті і працював на одній або багатьох осіб. Зміни можуть бути відслідковані і дозволено до публікації або ігноруються повертаючись до старої версії. Інші додаткові форми співробітництва дозволяють декільком користувачам змінювати (або коментар) сторінки, в той же час в спільній сесії.

#### Делегування

Деякі програми CMS дозволяє для різних груп користувачів, щоб мати обмежені права порівняно з конкретним змістом на веб-сайті, розводячи відповідальність управління контентом <sup>[6]</sup>

Управління документа: програмне забезпечення CMS може служити засобом спільної управління життєвим циклом документа від первісного часу створення, шляхом перегляду, публікації, архів, та знищення документів.

Зміст віртуалізації: програмне забезпечення CMS може забезпечити засіб, що дозволяє кожному користувачеві працювати у віртуальній копії всього веб-сайту, набір документів, та / або коду. Це дозволяє змін на різних взаємозалежних ресурсів, які розглядатимуться та / або виконаний у-контексті до подачі.

#### Придбання інформації

CMS програмне забезпечення часто допомагає у поширенні контенту шляхом створення RSS і Atom потоки даних в інші системи. Вони можуть також користувачі електронної пошти, якщо оновлення доступні як частина робочого процесу.

#### Багатомовний

Можливість відображення вмісту на декількох мовах.

#### Версіями

Як [система [управління документами]] и, програмне забезпечення CMS може дозволити процес версій, за допомогою якого сторінок перевіряються або з ВЦМП, дозволяє авторизованим редакторів, щоб отримати попередні версії, і продовжити роботу з обрана точка. Управління версіями корисно для утримання, яке змінюється з часом і вимагає оновлення, але це може бути необхідно повернутися або посилатися на попередній копії.

Є три основних типи ВЦМП: автономну обробку, оперативна обробка та гібридні системи. Ці терміни описують картину розгортання для ВЦМП відносно того, коли шаблони подання застосовані, щоб зробити веб-сторінки з структурованим змістом.

Деякі відомі приклади CMS: [12]

- WordPress був найпопулярнішою системою управління контентом до 2014 року <sup>[джерело?]</sup>. Вона виникла як блоги CMS, але була адаптована в повноцінну CMS.
- Textpattern є одним з перших відкритих джерел CMS
- Joomla є популярна система управління контентом
- Drupal виник перед WordPress і Joomla.
- ExpressionEngine - комерційний CMS, зроблені EllisLab.
- MediaWiki повноваження Вікіпедії і проектів, пов'язаних і є одним з найвідоміших прикладів вікі CMS.
- Cascade сервера популярний серед вузів <sup>[13]</sup> і корпоративних масштабних організацій
- EXO Платформа Open Source Соціальна CMS
- Liferay Open Source Portal WCMS
- TWiki Open Source Структуровані Wiki CMS
- Foswiki Open Source Структуровані Wiki CMS

## Лекція 2. Компоненти Wordpress

WordPress – це популярна система керування вмістом сайту (CMS), яка має відкритий вихідний код та поширюється під ліцензією GNU GPL. На даний момент, WordPress найчастіше використовується для створення блогів на її основі. Також на основі даної CMS можна створювати достатньо складні новинні ресурси, а також інтернет-магазини.

WordPress написана на мові програмування PHP та в якості бази даних використовує MySQL. Зараз доступно багато тем та плагінів, з допомогою яких можна створювати практично будь-які проекти. WordPress має досить просту панель адміністрації, яку легко освоїти за лічені години. А сам сайт, на основі WordPress, можна створити за один день.

Серед можливостей WordPress можна виділити підтримку тем, плагінів, RSS, Atom, trackback, pingback, веб-стандартів (XHTML, CSS), SEO-оптимізації, наявність перекладів на різні мови, в т.ч. українську та інші можливості.

Скачати WordPress можна на офіційному сайті: <http://wordpress.org/>

WordPress — це проста у встановленні та використанні система керування вмістом з відкритим кодом, яка широко використовується для створення веб-сайтів. Сфера застосування — від блогів до складних веб-сайтів.

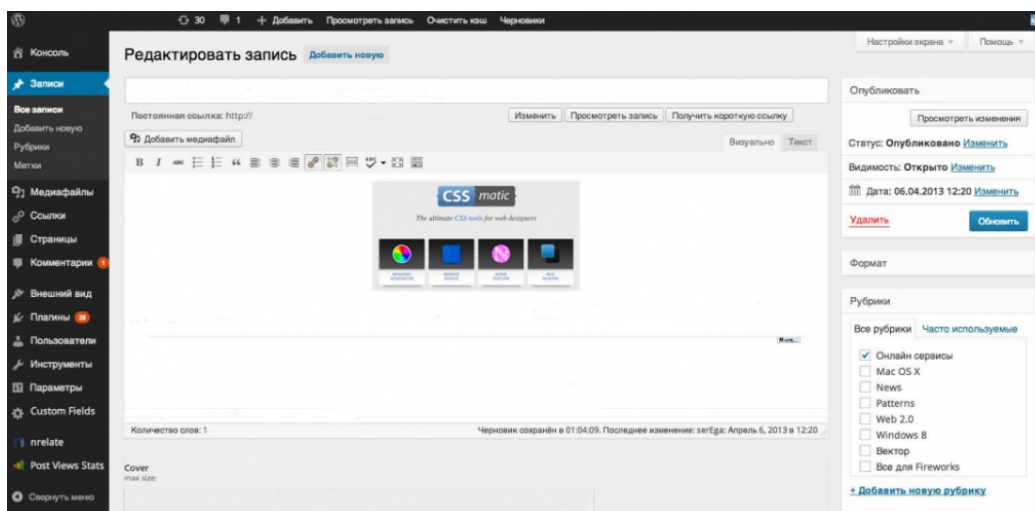
WordPress це система управління вмістом призначена для обслуговування блогів, також підходить як для створення сайтів із малим

контентом, так і для порталів, ґрунтуючись на своїй функціональності та великій кількості плагінів і розширень. WordPress функціонує на основі PHP/MySQL і дуже легкий в установці. Через це вважається однією із найпопулярніших CMS у світі. Втілити будь-який дизайн сайту можна за допомогою вбудованих WordPress-шаблонів або зверставши веб-сайт самостійно із використанням плагінів та модулів системи.

### Переваги WordPress:

- Система управління вмістом WordPress дає можливість конструювання будь-яких проектів завдяки вбудованій системі WordPress-тем та WordPress-плагінів.
- Платформа WordPress ідеальна для публікацій, орієнтована на красу та зручність у використанні.
- Система управління WordPress швидка, проста для замінування коду та достатньо гнучка.
- Оновлення WordPress плагінів, тем та системи відбувається автоматично.
- У середовищі WordPress легко налаштовуються віджети (блоки) та меню, а також є редактор HTML сторінок.
- Розробниками WordPress передбачено ретранслявання тегів з кирилиці у латиницю для правильної індексації сторінок у пошукових системах.

### Вигляд панелі адміністрування:



З часом з'явилися так звані **системи керування контентом, блогом, платформи для публікацій**. По суті - це майже одне й те саме. Вони дозволяють централізовано керувати усім на блозі - записами, категоріями, файлами та інш.

Все робиться з однієї сторінки - **панелі адміністратора** (або по простому - адмінки). В більшості випадків знання якихось **HTML, PHP** та інших мов програмування не потрібно зовсім.

Системи керування, окрім додавання записів, мають ще дуже багато можливостей, які включені в їх стандартну комплектацію, або додаються за допомогою так званих **плагінів** (додаткових підпрограм), які розширяють функціональність цих систем.

WordPress повністю безкоштовна система.

Розповімо трохи докладніше про WordPress.

WordPress є ідеальною платформою для публікації, орієнтованої на естетику, веб-стандарту, а також зручність використання. WordPress є безкоштовною системою.

Простіше кажучи, WordPress — це те, що Ви використовуєте, коли бажаєте працювати з програмним забезпеченням для ведення блогу, а не боротися з ним.

Основні можливості WordPress:

### **Керування та адміністрування**

- WordPress розроблений для інсталяції на Ваш хостинг, що дає повний контроль над блогом, на відміну від безкоштовних блого-сервісів.
- Керування зареєстрованими користувачами
- Особисті профілі користувачів
- Легке встановлення та оновлення. Відома інсталяція за 5 хвилин.
- Динамічне створення сторінок, не має потреби перестворювати усі сторінки блогу при внесенні змін до нього.
- Інтернаціоналізація та локалізація, WordPress "розмовляє" Вашою мовою :)

### **Публікація**

- RSS стрічки, публікація через протоколи RSS 1.0 (aka RDF), RSS 2.0 та ATOM
- Постійні гарні посилання. Вигляд лінків до Ваших записів, категорій, тегів може бути налаштований за Вашим бажанням
- Міжблогові комунікації. У WordPress інтегровані PingBack та TrackBack, дві чудові функції для зв'язку з іншими блогами

### **Дизайн та розширення можливостей**

- Оснований на шаблонах дизайн
- Шаблонні теги, просто додавайте нове у шаблони без особливих знань PHP
- Теми, легко змініть вигляд блогу, просто завантаживши та встановивши нову тему до нього

- Плагіни. Величезна кількість плагінів (додаткових підпрограм), що розширюють функціональність блогу

### Контент

- Захист паролем обраної інформації
- Запланована наперед публікація
- Багатосторінкові записи
- Завантаження файлів та зображень до записів
- Категорії
- Візуальний редактор
- Пошук по блогу
- Коментування

Це лише деякі з багатьох можливостей WordPress.

Завантажити Український WordPress Ви можете [uk.wordpress.org](http://uk.wordpress.org).

## Тема 2. Трансформації, переходи і анімація в мультиплатформених веб-ресурсах

### Лекція 3. Трансформації

Властивість `transform` дозволяє застосовувати 2D або 3D трансформацію до елемента. Ця властивість дозволяє обертати, масштабувати, переміщувати, нахилити елемент, тощо.

Властивість `transform` отримує в якості значень "трансформуючі функції". Ці функції виглядають як `scale()`, `translateX()`, `rotate()` тощо. Вони приймають параметри для визначення ступеня трансформації.

Ви також можете задавати декілька функцій одночасно. Якщо ця властивість приймає значення відмінне від `none` - створюється новий контекст накладення. В цьому випадку об'єкт буде поводитися як контейнер для нащадків з `position: fixed;`.

### Синтаксис

```
transform: none | transform-functions | initial | inherit;
```

Властивість **transform** може отримувати 24 значення:

`none`

Скасовує трансформацію. Без задання.

```
matrix(n, n, n, n, n, n)
```

2D перетворення, за допомогою матрицю з шести значень.

`matrix3d (n, n, n, n, n, n, n, n, n, n, n, n, n, n, n, n)`

3D перетворення, за допомогою матриці 4x4 з 16 значень.

`translate(x, y)`

2D зсув по горизонталі і вертикалі.

`translate3d(x, y, z)`

3D зсув по трьом вісям.

`translateX(x)`

2D зсув по горизонталі (вісь X). Дозволені від'ємні значення. Додатне значення зсовує вправо, від'ємне вліво.

`translateY(y)`

2D зсув по вертикалі (вісь Y). Дозволені від'ємні значення. Додатне значення зсовує вниз, від'ємне вверх.

`translateZ(z)`

3D зсув по вісі Z. Дозволені від'ємні значення. Додатне значення зсовує вправо, від'ємне вліво.

`scale(x, y)`

Масштабування елемента по горизонталі та вертикалі. Значення менше 1 - зменшує елемент, більше - збільшує.

`scale3d(x, y, z)`

Масштабування елемента тривимірному просторі.

`scaleX(x)`

Масштабування елемента по горизонталі (вісь X).

`scaleY(y)`

Масштабування елемента по вертикалі (вісь Y).

`scaleZ(z)`

Маштабування елемента по вісі Z.



`rotate(angle)`

Обертає елемент, на заданий кут, в двовірному просторі.

`rotate3d(x, y, z, angle)`

Обертає елемент в тривірному просторі.

`rotateX(angle)`

3D обертання елемента по вісі X.

`rotateY(angle)`

3D обертання елемента по вісі Y.

`rotateZ(angle)`

3D обертання елемента по вісі Z.

`skew(x-angle, y-angle)`

2D перекіс двум вісям X- та Y ( по горизонталі та вертикалі).

`skewX(angle)`

2D перекіс по горизонталі (вісь X).

`skewY(angle)`

2D перекіс по вертикалі (вісь Y).

`perspective(n)`

Перспективний вид для 3D трансформованого елемента.

`initial`

Встановлює властивість у значення без задання

`inherit`

Вказує на спадковість властивості від свого батьківського елемента (якщо відповідна властивість встановлена).

Динамічний приклад застосування властивості `transform`

<p>Вибери потрібне значення для <code>transform: translate();</code> (у відсотках):</p>

```
<input type="number" value="0" id="field"></input>
<input type="number" value="0" id="field2"></input>
```

```
<div id="block">
  Привіт! :)
</div>
```

```
body {
  padding: 36px;
  text-align: center;
  line-height: 1.45;
  font-size: 17px;
  font-family: sans-serif;
}
```

```
#field,
#field2{
  padding: 8px;
}
```

```
#block {
  width: 100px;
  padding: 30px;
  text-align: center;
  background: gold;
  margin: 36px auto 0 auto;
}
```

```
var block = document.getElementById('block');
var field = document.getElementById('field');
var field2 = document.getElementById('field2')
```

```
field.onchange = field2.onchange = function change() {
  var newVal = getNewValue();
  block.style.transform = 'translate(' + newVal[0] +
'px, ' + newVal[1] + 'px)';
}
```

```
function getNewValue() {
  var result = [];
  result[0] = field.value;
  result[1] = field2.value;
  return result;
}
```

Простий приклад роботи властивості

```
<div class="skew">Я перекошений блок</div>
```

```
div {  
  width: 175px;  
  padding: 20px;  
  background: gold;  
  margin: 125px auto 0 auto;  
}  
  
.skew {  
  transform: skewX(10deg) skewY(35deg);  
}
```

Обертання елемента

```
div {  
-ms-transform: rotate(7deg); /* IE 9 */  
-webkit-transform: rotate(7deg); /* Chrome, Safari,  
Opera */  
transform: rotate(7deg);  
}
```

## Лекція 4. Переходи й анімації

### css властивість transition

CSS властивість встановлює ефект переходу між двома станами елемента. Використовується з псевдокласом `:hover` або `:active` або якщо стан елемента був динамічно змінений за допомогою JavaScript.

Властивість `transition` - це універсальна властивість і дозволяє одночасно задати чотири значення, які розділені комою:

- `transition-property`
- `transition-duration`
- `transition-timing-function`
- `transition-delay`

Завжди вказуйте властивість `transition-duration`, якщо

**Порада:** цього не зробити тривалість переходу буде 0s, тому перехід не матиме ніякого ефекту.

**Нотатка:** Зверніть увагу, що порядок запису значень для властивості має

дуже велике значення.

## Синтаксис

```
transition: property duration timing-function  
delay|initial|inherit;
```

Властивість **transition** може отримувати 6 значень:

**transition-property**

Ім'я властивості, за якою ми будемо слудкувати. І коли значення цієї властивості зміниться - запуститься ефект переходу.

**transition-duration**

Тривалість ефекту переходу.

**transition-timing-function**

Визначає криву (функцію) ефекту переходу.

**transition-delay**

Затримка ефекту переходу.

**initial**

Встановлює властивість у значення без задання

**inherit**

Вказує на спадковість властивості від свого батьківського елемента (якщо відповідна властивість встановлена)

**Значення без задання:** *all 0s ease 0s*

**Наслідує:** Ні

**Анімується:** Так

**JavaScript синтаксис:** `object.style.transition="all 2s"`

## Приклади

Коли елемент `<input type="text">` отримує фокус, він повільно змінює свою ширину з 100px до 250px

```
input[type=text] {
```

```
width: 100px;
-webkit-transition: ease-in-out width .35s ease-in-out;
/* Safari 3.1 to 6.0 */
transition: ease-in-out width .35s ease-in-out;
}
input[type=text]:focus {      width: 250px; }
```

Наведіть курсор миші на елемент <div>, щоб поступово змінити його ширину з 100px до 300px

```
div {      width: 100px;
-webkit-transition: width 2s; /* Safari 3.1 to 6.0 */
transition: width 2s;
}
div:hover {      width: 300px; }
```

### Простий приклад для демонстрації роботи властивості

```
<div class="block">
  <code>transition: width .2s ease-out .15s</code>
</div>
```

```
body {
  padding: 36px;
}

.block {
  width: 350px;
  background-color: gold;
  color: #444;
  font-size: 15px;
  padding: 16px 8px;
  transition: width .2s ease-out .15s
}

.block:hover {
  width: 425px;
}
```

### css властивість **animation**

Універсальна властивість **animation** — скорочена властивість для восьми властивостей анімації:

- *animation-name*
- *animation-duration*
- *animation-timing-function*
- *animation-delay*
- *animation-iteration-count*
- *animation-direction*
- *animation-fill-mode*
- *animation-play-state*

Для створення анімації засобами CSS, використовуйте властивість `animation` разом з правилом `@keyframes`, який встановлює візуальні ефекти для анімації.

Зверніть увагу, що порядок у визначенні властивості має дуже важливий. Перший значення у формі часу, розглядається як `animation-duration`, а другий - як `animation-delay`.

Порядок також важливий для визначення *animation-name* серед інших значень. Під час аналізування правил, ключові слова допустимі лише для властивостей відмінних від `animation-name`, і значення яких не було задані раніше, будуть прийняті саме для них, а не для `animation-name`. Крім того, мають бути задані значення без задання для всіх властивостей, щоб була можливість визначити `animation-name`.

Усі правила задані для елемента можуть бути анімовані; це також не заважає одночасно задіяти для них `transition`.

Завжди вказуйте властивість `animation-duration` (тривалість

**Нотатка:** анімації), в іншому випадку тривалість дорівнює 0, і тому анімація не буде відтворюватися.

## Синтаксис

```
animation: name duration timing-function delay  
iteration-count direction fill-mode play-state;
```

Властивість **animation** може отримувати 10 значень:

`animation-name`

Задає ім'я `@keyframes` для анімації

`animation-duration`

Визначає скільки секунд або мілісекунд анімації потрібно, щоб завершити один цикл

`animation-delay`

Визначає затримку до початку анімації

`animation-iteration-count`

Вказує, скільки разів анімація повинна повторюватись

`animation-direction`

Визначає, чи повинна анімація відтворюватися в зворотному напрямку на альтернативних циклах.

`animation-fill-mode`

Визначає стиль для елемента, коли анімація не відтворюється (коли анімація закінчилася або має затримку)

`animation-play-state`

Визначає, чи буде анімація працювати або буде призупинена

`animation-timing-function`

Визначає криву (функцію) швидкості анімації

`initial`

Встановлює властивість у значення без задання

`inherit`

Вказує на спадковість властивості від свого батьківського елемента (якщо відповідна властивість встановлена)

**Значення без задання:** *none 0 ease 0 1 normal none running*

**Наслідує:** Ні

**Анімується:** Так

**JavaScript синтаксис:** `object.style.animation="mymove 5s infinite"`

Приклад анімації

```
<div id="block">Тицяй, щоб уввімкнути або зупинити анімацію</div>
```

```
#block {  
  font-family: monospace;
```

```

background-color: gold;
color: #222;
font-size: 16px;
line-height: 1.45;
width: 175px;
text-align: center;
padding: 36px 16px;
box-shadow: 1px 1px 1px 1px #000;
cursor: pointer;
transition: box-shadow .15s linear;
position: absolute;
top: 16px;
left: 16px;
animation: move 1.15s ease-in 0s infinite alternate
paused none;
}

#block:hover {
  box-shadow: 1px 1px 1px 1px rgba(0, 0, 0, .45);
}

@keyframes move {
  from {
    top: 16px;
    left: 16px;
  }
  to {
    top: 250px;
    left: 350px;
  }
}

```

### Тема 3. Layers WP для Wordpress.

#### Лекція 5. Layers WP. Особливості

Розглянемо особливості створення теми із використанням Layers WP.

1. Відкрили програму Adobe Photoshop CC та створили новий документ.
2. Намалювали header майбутнього сайту (рис. 1).

Рис. 1. Верхня частина сайту



3. Після цього додали в дизайн лендінгу блок з слайдером головного зображення. На ньому додали заголовок та кнопку (рис. 2).

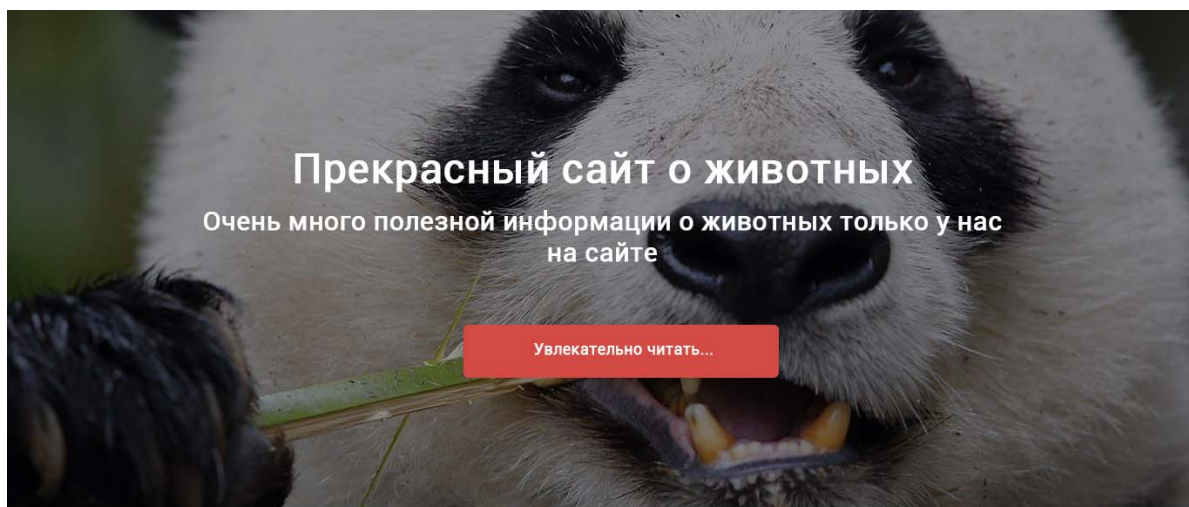


Рис. 2. Статичне зображення з заголовком та кнопкою

4. Наступний блок лендінгу це блок з коротким текстовим описом (рис. 3).

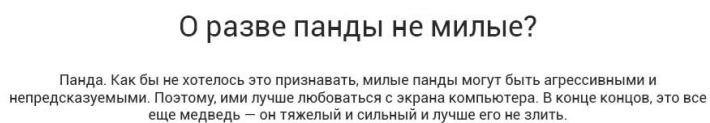


Рис. 3. Блок з текстом

5. Для наступного блоку лендінгу було використано зображення вовка (рис. 4).



Рис. 4. Блок з зображенням

6. Для наступного блоку лендінгу було використано 3 текстових блока з описом (рис. 5).



Рис. 5. Блок з 3 текстовими описом

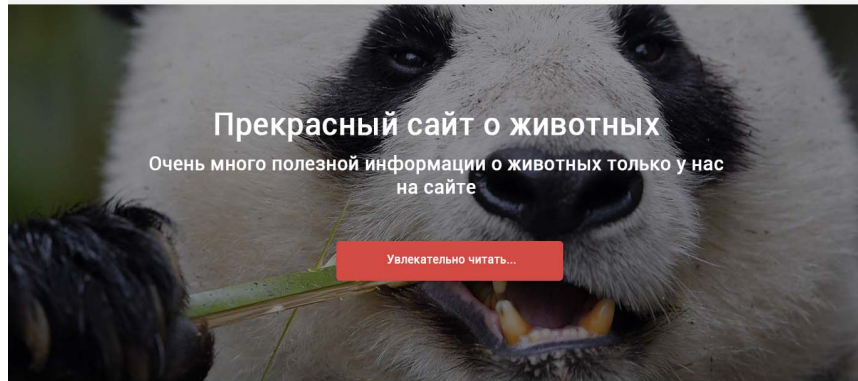
7. Наступний блок лендінгу це блок з коротким текстовим описом (рис. 6).

### Ну может быть еноти милые??

Енот. Выглядят еноты ну очень мило. Неудивительно, что некоторые люди даже заводят их в качестве домашнего питомца. Однако, стоит помнить о том, что еноты могут напасть на человека. Вряд ли он причинит вам серьезные повреждения, однако, этот зверек может принести вам в «подарок» бешенство.

Рис. 6. Блок з текстом

8. Створили дизайн-макет сторінки у Photoshop у вигляді односторінкового дизайну Landing Page (для мобільних пристроїв, для планшетів та екранів моніторів комп'ютерів, тобто три psd-файли). Макет повинен враховувати можливості теми Layers.
9. Подивитися відеоурок та додати статичну Landing Page (головну сторінку), використовуючи Layers WP, зберігши тематику, обрану вами для вашого сайту на вордпрес.
10. Виконати верстку дизайн-макету, використовуючи HTML5 + CSS3. Bootstrap не використовувати. Тільки CSS3.
11. Налаштувати Wordpress для теми Layers та перенести верстку HTML5+CSS3 у цю тему. Тобто зробити так, щоб тема Layers стала виглядати згідно розробленого вами дизайн-макету.
12. Кінцевий результат рис. 7.



## Прекрасный сайт о животных

Очень много полезной информации о животных только у нас на сайте

Увлекательно читать...

### О разве панды не милые?

Панда. Как бы не хотелось это признавать, милые панды могут быть агрессивными и непредсказуемыми. Поэтому, ими лучше любоваться с экрана компьютера. В конце концов, это все еще медведь – он тяжелый и сильный и лучше его не злить.



#### Волк

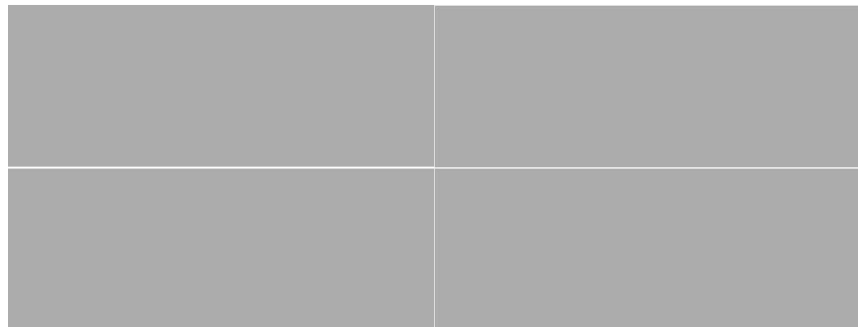
Волк, или серый волк, или обыкновенный волк – вид хищных млекопитающих из семейства псовых. Наряду с койотом, обыкновенным шакалом и ещё несколькими видами составляет род волков

#### Волк

Волк, или серый волк, или обыкновенный волк – вид хищных млекопитающих из семейства псовых. Наряду с койотом, обыкновенным шакалом и ещё несколькими видами составляет род волков

#### Волк

Волк, или серый волк, или обыкновенный волк – вид хищных млекопитающих из семейства псовых. Наряду с койотом, обыкновенным шакалом и ещё несколькими видами составляет род волков



### Ну может быть еноти милые??

Енот. Выглядят еноты ну очень мило. Неудивительно, что некоторые люди даже заводят их в качестве домашнего питомца. Однако, стоит помнить о том, что еноты могут напасть на человека. Вряд ли он причинит вам серьезные повреждения, однако, этот зверек может принести вам в «подарок» бешенство.

Рис. 7. Кінцевий результат

## Лекція 6. Зміна теми оформлення Wordpress

Тема WordPress – це набір файлів, що визначають оформлення блогу. WordPress.com володіє колекцією з [250 різноманітних тем](#).

Щоб активувати тему, в консолі блогу перейдіть на сторінку **Зовнішній вигляд -> Теми**. Тему можна змінювати як завгодно часто.

Приклад теми:



The Coraline Theme  
Just another WordPress.com Theme

HOME | A PARENT PAGE | ABOUT | CONTACT | HTML ELEMENTS | IMAGES | READABILITY TEST

**A Sticky Post**

You can mark a post as sticky by checking the option that says "Stick this post to the front page" in the Publish settings on any post. Sticky posts are great for short but important messages you want to stay prominent.

Posted in [Examples](#) → [Leave a comment](#)  
Tagged [examples](#), [sticky](#)

Sometimes you only want to post something short or a quick link somewhere. In that case, an *Aside* post is the thing for you. Just post in the *Asides* category!

Posted on [July 16, 2010](#) by [Ian Stewart](#) → [Leave a comment](#)

**CALENDAR**

July 2010

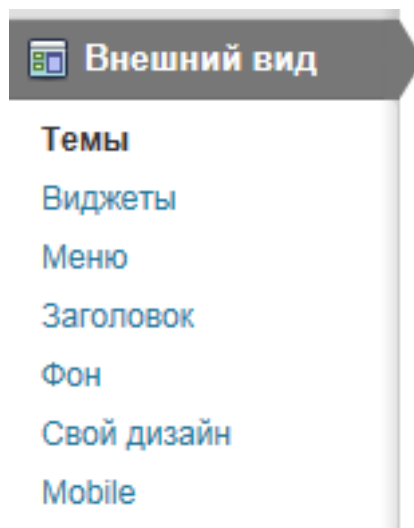
| M  | T  | W  | T  | F  | S  | S  |
|----|----|----|----|----|----|----|
|    |    |    | 1  | 2  | 3  | 4  |
| 5  | 6  | 7  | 8  | 9  | 10 | 11 |
| 12 | 13 | 14 | 15 | 16 | 17 | 18 |
| 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| 26 | 27 | 28 | 29 | 30 | 31 |    |

**ARCHIVES**

- July 2010

**META**

Щоб переглянути всі доступні теми, подивіться **Вітрину тем** чи перейдіть на сторінку **Зовнішній вигляд -> Теми** в консолі блогу. Меню **Зовнішній вигляд** доступний лише адміністраторам блогу.



На сторінці **Зовнішній вигляд** -> **Тем** відображається ваша поточна тема.

Поточна тема блогу не відображається при огляді або пошуку інших тем.

### **Перегляд тем**

Доступно декілька варіантів перегляду тем:

- сортування за алфавітом;
- популярності;
- новизні;
- перегляд платних тем.

### **Пошук тем**

Шукаєте конкретну тему або тему з певними можливостями? Наберіть ключові слова в поле пошуку та натисніть кнопку **Пошук**.

Результати пошуку будуть відсортовані відповідно до режиму перегляду (за алфавітом, популярності та інше).

### **Фільтр характеристик**

Для пошуку тем з конкретними особливостями можна скористатися **Фільтром характеристик**.

#### Фильтры тем

##### Цвета

- |   |                                  |                                     |                                  |                                     |
|---|----------------------------------|-------------------------------------|----------------------------------|-------------------------------------|
| <input type="checkbox"/> Чёрный           | <input type="checkbox"/> Синий   | <input type="checkbox"/> Коричневый | <input type="checkbox"/> Серый   | <input type="checkbox"/> Зелёный    |
| <input type="checkbox"/> Оранжевый        | <input type="checkbox"/> Розовый | <input type="checkbox"/> Пурпурный  | <input type="checkbox"/> Красный | <input type="checkbox"/> Серебряный |
| <input type="checkbox"/> Жёлто-коричневый | <input type="checkbox"/> Белый   | <input type="checkbox"/> Жёлтый     | <input type="checkbox"/> Тёмный  | <input type="checkbox"/> Светлая    |

##### Колонки

- |                                       |                                      |                                      |   |                                     |
|---------------------------------------|--------------------------------------|--------------------------------------|---|-------------------------------------|
| <input type="checkbox"/> Одна колонка | <input type="checkbox"/> Две колонки | <input type="checkbox"/> Три колонки | <input type="checkbox"/> Четыре колонки | <input type="checkbox"/> Меню слева |
| <input type="checkbox"/> Меню справа  |                                      |                                      |   |                                     |

##### Ширина

- |  |                                 |  |
|--|---------------------------------|--|
| <input type="checkbox"/> Фиксированная | <input type="checkbox"/> Гибкая | <input type="checkbox"/> Responsive Layout |
|--|---------------------------------|--|

##### Функции

- |   |  |  |   |   |
|---|--|--|---|---|
| <input type="checkbox"/> Благаватар                               | <input type="checkbox"/> Произвольный фон              | <input type="checkbox"/> Настройка цветов  | <input type="checkbox"/> Произвольный заголовок | <input type="checkbox"/> Произвольное меню    |
| <input type="checkbox"/> Стиль для редактора                      | <input type="checkbox"/> Миниатюры записей в заголовке | <input type="checkbox"/> Миниатюры записей | <input type="checkbox"/> Flexible Header        | <input type="checkbox"/> Публикация с главной |
| <input type="checkbox"/> Одноколоночный шаблон                    | <input type="checkbox"/> Infinite Scroll               | <input type="checkbox"/> Микроформаты      | <input type="checkbox"/> Форматы записей        | <input type="checkbox"/> Post Slider          |
| <input type="checkbox"/> Поддержка языков с письмом справа налево | <input type="checkbox"/> Запись-липучка                | <input type="checkbox"/> Настройки темы    | <input type="checkbox"/> Готова к переводу      |   |

##### Тема

- |                                   |                                   |                                       |
|-----------------------------------|-----------------------------------|---------------------------------------|
| <input type="checkbox"/> Праздник | <input type="checkbox"/> Фотоблог | <input type="checkbox"/> Времена года |
|-----------------------------------|-----------------------------------|---------------------------------------|

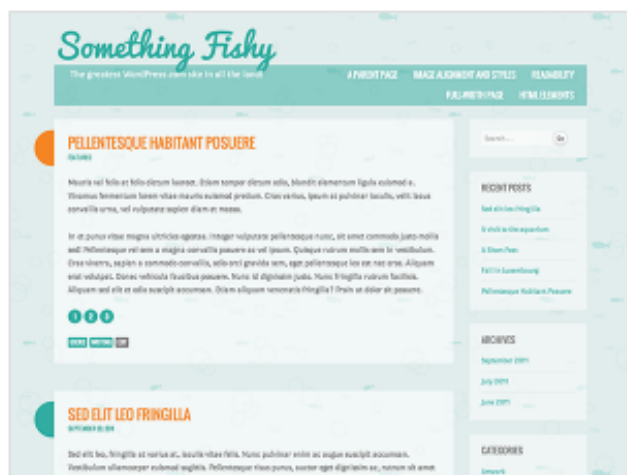
Применить фильтры

[Заккрыть фильтры](#)

Так само, как і при пошуку відфільтровані теми будуть відсортовані відповідно до режиму перегляду (за алфавітом, популярності).

## Зміна/активація теми

Під час вибору теми можна подивитися, як виглядає тема стосовно до вашого блогу. Це можна зробити натисканням на посилання **Перегляд** під зображенням теми.



## Something Fishy

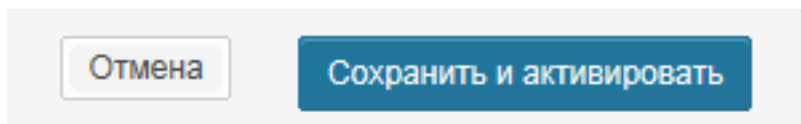
[Активировать](#)

[Просмотр](#)

[Подробности](#)

Якщо ви вже знаєте, що вам потрібна саме ця тема, можна клацнути по посиланню **Активувати**.

Після натискання на **зображення теми** або посилання **Перегляд** відкриється вікно з видом вашого блогу, оформленого відповідно до обраної теми. Якщо вам подобається ця тема, натисніть кнопку **Зберегти і активувати** в лівому верхньому кутку вікна. Якщо ця тема не влаштовує вас, натисніть кнопку **Скасування** в лівому верхньому куті вікна, щоб повернутися до перегляду тем.



## Платні теми

Крім безкоштовних тем, також є можливість обрати платні теми з особливим дизайном, додатковим налаштуванням, а також тех. підтримкою від авторів теми.

## Зображення заголовка

Багато тем WordPress.com дозволяють користувачам завантажувати свої власні зображення заголовка. Ця можливість безкоштовна і не вимагає оплати послуги «Свій дизайн» (за умови, що вона підтримується вашою темою).

## Завантаження зображення заголовка

Перейдіть на сторінку консолі **Зовнішній вигляд -> Тема**. Якщо такої можливості немає, значить, ваша тема не підтримує вибір зображення заголовка.

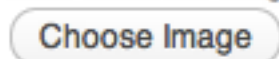
Можна вибрати файл зі свого комп'ютера і натиснути кнопку **Завантажити** ...

Выберите файл на своём компьютере:



... або вибрати зображення з **Бібліотеки файлів**.

Or choose an image from your media library:



**Увага!** Можна використовувати лише файли **.Jpeg**, **.Gif**, і **.Png**. Всі обрізані зображення будуть перетворені у формат **JPEG**. Краще всього використовувати зображення правильного розміру, тобто відповідного для вашої теми. Щоб зберегти прозорість, використовуйте зображення **.Gif** або **.Png** правильного розміру.

Якщо ви завантажите зображення, розміри якого не підходять обраній темі, вам доведеться обрізати його.

Виберіть частину зображення для заголовка, потім натисніть кнопку **Обрізати і опублікувати**.

Після повідомлення про зміну заголовка можна перейти на сайт і подивитися, як він виглядає.

## Тема 4. Дочірні теми для Wordpress.

### Лекція 7. Поглиблене вивчення тем Wordpress. Дочірні теми

Починаючи із третьої версії WP можна створювати дочірні теми. Вони доповнюють або розширюють певну існуючу тему, тобто додають/змінюють щось у темі. Наприклад, ви використовуєте певну тему і хочете там щось змінити/додати, у такому випадку можна створити дочірню тему, яка міститиме в собі лише кілька файлів для того, щоб змінити бажану тему або додати додаткові можливості.

### Структура

Дочірні теми слід розміщувати у теці основних тем, у `/wp-content/themes` Я розгляну створення дочірньої теми для «Twenty Ten», нової стандартної теми WP.

Нижче наведена структура дочірньої теми:

#### **public\_html**

- **wp-content**
  - **themes** (тека, де розміщено теми)
    - **twentyten** (тека нашої теми, Twenty Ten)
    - **twentyten-child** (тека дочірньої теми, можете назвати по-іншому)
      - **style.css** (обов'язковий файл теми; має називатися лише style.css)

Дочірня тема має складатися, як мінімум з одного обов'язкового файлу `style.css`, інші файли є додатковими:



1. style.css (обов'язково)
2. functions.php (додатково)
3. Файли шаблонів (додатково)
4. Інші файли, наприклад JavaScript (додатково)

## Структура обов'язкового файлу style.css

Завдяки цьому файлу WP визначає, що ця тема є дочірньою для певної теми, у нашщому випадку це «Twenty Ten».

1. /\*
2. Theme Name: Twenty Ten Child
3. Theme URI: http://example.com/
4. Description: Child theme for the Twenty Ten theme
5. Author: Your name here
6. Author URI: http://example.com/about/
7. Template: twentyten
8. Version: 0.1.0
9. \*/

Невелике пояснення:

- Theme Name. (обов'язково) Назва дочірньої теми.
- Theme URI. (додатково) Сторінка сайту дочірньої теми.
- Description. (додатково) Опис теми.
- Author URI. (додатково) Сайт автора.
- Author. (додатково) Ім'я автора.
- Template. (обов'язково) тека, де розміщена батьківська тема.
- Version. (додатково) Версія теми, наприклад.: 0.1, 1.0, тощо.

Після \*/ ви вже можете писати свій власний код, можна, наприклад включити стиль із батьківської теми та дописати власний, наприклад так:

1. /\*
2. Theme Name: Twenty Ten Child
3. Description: Child theme for the Twenty Ten theme
4. Author: Your name here
5. Template: twentyten
6. \*/
7. @import url("../twentyten/style.css"); /\* Імпортуємо файл стилів із батьківської теми \*/
8. /\* Пишемо власний стиль \*/
9. #site-title a {
10. color: #009900;
11. }

## Використання файлу `functions.php`

На відміну від файлу `style.css`, який використовується дочірньою темою замість батьківського файлу, файл дочірньої теми `functions.php` додається до файлу батьківського, тобто не замінюється. Він виконується перед аналогічним файлом батьківської теми.

Ми додамо на сайт фавіконку, яка лежить у кореневій теці сайту, пишемо такий код у файлі `functions.php`:

```
1. # Add favicon
2. function cehla_theme_favicon() {
3.     print '<link rel="icon" href="'. get_bloginfo('template_url') .'/i/favicon.ico'
       />'. "n";
4. }
5. add_action('wp_head', 'cehla_theme_favicon');
6. add_action('admin_head', 'cehla_theme_favicon');
7. add_action('login_head', 'cehla_theme_favicon');
8. /**/
```

## Лекція 8. Використання готового html- та css-коду для створення теми Wordpress

Шаблони сторінок в WordPress є чудовим способом змінити структуру сторінок або додати їм функціональності. Створення власного шаблону сторінок може бути не простою справою, особливо для новачків, тому ми постараємося детально висвітлити цей процес. Наявність базових знань HTML, CSS і PHP допоможе вам швидше засвоїти урок, хоча це і не потрібно, якщо ви будете робити все за зразком. Ви зможете створювати і розгортати стільки різних шаблонів, скільки буде потрібно, що відкриває воістину безліч можливостей для тонкої настройки вашого сайту на WordPress.

Власний шаблон буде корисний, якщо ви хочете отримати на виході сторінку, яка виглядає зовсім інакше, ніж решта вашого сайту, але яка як і раніше буде працювати під управлінням WordPress CMS. Це може бути домашня сторінка, landing page (цільова/сторінка-вітрина/рекламна сторінка), сторінка контактів, сторінка тарифних планів і т.д. Створення унікальної шаблонної сторінки дозволить розширити наявний функціонал WordPress і змінити відображення вмісту на сайті на свій розсуд. Наприклад, ви хочете сторінку з іншими шрифтами, шапкою та футером, ніж на інших сторінках – все це можливо з власним шаблоном Вордпрес сторінки.

Користувальницькі шаблони сторінок в WordPress мають спеціальний заголовок, який розміщується на початку файлу, що вказує двигуну своє системне ім'я.

Наприклад,

```
/* Template Name: Custom Page Template */
```

де **Custom Page Template** – системна назва шаблону (і тому вона закоментована – тобто її не бачать звичайні відвідувачі сайту).

Вказавши шаблон «*Custom Page Template*» ви «накажете» Вордпрес виводити контент сторінки саме так, як спроектували в шаблонному файлі. Як спроектувати шаблон? Про це далі.

Тепер, якщо двигун побачить, що сторінці присвоєний цей шаблон – він виведе на екран саме його. За значенням змінної `Template Name`, wordpress дізнається про те, що цей файл потрібно сприймати як шаблон для сторінки.

**Як присвоїти шаблон?** В консолі сайту, коли ви створюєте нову сторінку (Сторінка – Додати нову), в правій бічній панелі «*Атрибути сторінки*» (Page Attributes) ви побачите вкладку “*Шаблон*” (Template), що містить всі доступні шаблони. Вибираючи шаблон – ви привласнюєте його.

## Практика: створюємо новий шаблон сторінки

Є кілька способів:

- створити абсолютно новий шаблон
- взяти за основу наявні шаблони в темі.

Ми підемо другим шляхом – візьмемо за основу шаблонний файл `page.php` з теми Twenty Fifteen, яка йде в комплекті з останньою версією WordPress (4.1 – на момент написання уроку), і трохи його відкоригуємо.

Давайте подивимося на вже готовий шаблон, а потім детально його розберемо. Ось, що у мене вийшло:

```
<?php /* Template Name: Custom Page Template */
get_header(); ?><!-- підключаємо хедер - файл-шаблон
header.php --> <div id="primary" class="content-area">
<!-- ОСНОВНИЙ КОНТЕНТ --> <div class="content">
                                <?php while (have_posts()) :
the_post(); ?>                                <?php
the_content(); ?>                                <?php
endwhile; ?>                                <?php endif; ?>
```

шаблон sidebar.php --> <?php get\_footer(); ?><!--  
підключаємо футер – файл-шаблон footer.php -->

А тепер пройдемося по пунктах.

1. Створюємо порожній файл у текстовому редакторі Notepad++ і вставляємо ім'я шаблону в самий верх:

```
<?php /* Template Name: Custom Page Template */  
get_header(); ?>
```

Ім'я шаблону тут включено в php-код, що викликає шаблон хедера (*header.php*)

Вибираємо кодування: в Notepad++ йдемо *Кодування – Кодувати в UTF-8 (без BOM)*

Зберігаємо сторінку у форматі **.php** (у нашому прикладі *template-custom.php*) в папку з темою за адресою */wp-content/themes/ВАША-ТЕМА/*. Ви можете використовувати свою назву, головне щоб воно відрізнялося від вже наявних у вашій темі.

2. Додаємо html-розмітку для контенту і задаємо умову для виведення контенту:

```
<div id="primary" class="content-area"> <div  
class="content"> <?php if (have_posts()) : ?>  
</div> </div>
```

3. Підключаємо шаблон сайдбару (*sidebar.php*) – якщо нам потрібна бічна колонка на сайті.

```
<?php get_sidebar(); ?>
```

4. Закриваємо шаблон, підключивши футер

```
<?php get_footer(); ?>
```

5. Зберігаємо файл і створюємо сторінку в адмінці сайту, на основі щойно створеного шаблону: йдемо в Консоль – *Сторінки – Створити нову*.

Присвоюємо назву сторінці і заповнюємо область вмісту потрібним контентом. У правій боковій колонці, у вікні «Атрибути сторінки» (Page Attributes) з вкладки «Шаблон» (Template) вибираємо створений шаблон – Custom Page Template.

Публікуємо сторінку і ... милуємося!

## Деякі моменти

- Як бачите, шаблон сторінки – це набір php-коду і html-розмітки, іншими словами – скелет сторінки. За допомогою підключених файлів-стилів CSS ми, потім, додаємо візуальне оформлення сторінці. А за допомогою javascript розширюємо функціональність.
- CSS і Javascript, як правило, підключаються у файлі header.php
- Один шаблон може вміщати в себе і інший/інші шаблони.

Розглянемо корисний плагін WP Maintenance Mode. Він дозволяє встановити тему-заглушку, яка коротко проінформує відвідувачів про те, що на сайті/блогі ведеться реконструкція, залишаючи для адміна повний функціонал щодо інших сторінок і публікацій сайту. У стандартному пакеті плагіну є: кілька шаблонів оформлення; можливість додати таймер зворотнього відліку; можливість підписатись через feedburner або електронну пошту. Також є можливість зробити свою власну сторінку-заглушку і без проблем встановити через настройки плагіну в адмін-панелі WordPress.

Стандартну конструкцію сторінки-заглушки можна оформити з власними шрифтами, фоном і заголовком. Для цього слід просто створити новий CSS-файл Далі необхідно змінити оформлення класів, які використовуються у заглушці, на власні.

Необхідно оформити наступне (для прикладу взято CSS-файл одного з базових шаблонів):

1. \* {
2. margin:0px;
3. padding:0px;
4. border:0px;}
- 5.
6. body {
7. background:#548a32 url('images/bg.jpg') center top no-repeat;
8. font-family:Arial, Verdana;
9. color:#fff;
10. font-size:14px;
11. padding-top:200px;
12. width:700px;
13. margin:0px auto;}
- 14.
15. a { color:#fff; text-decoration:none;}
- 16.
17. a:hover { color:#ead311;}
- 18.
19. #content { position:relative; padding-left:118px;}

```
20.
21.#content p {
22.margin-top:30px;
23.font-size:18px;
24.line-height:28px;}
25.
26.h1 {
27.font-weight:36px;
28.font-family:verdana;
29.font-weight:normal;}
30.
31.h3 {
32.color: #8f1218;}
33.
34.#header { margin-bottom:90px; padding-left:}
35.
36.div.admin a {
37.display:block;
38.width:142px;
39.text-align:center;
40.padding-top:20px;
41.height:32px;
42.background-image:url('images/button.png');
43.position:absolute;
44.right:155px;}
45.
46.#footer { margin-top:75px; font-size:12px;}
47.
48.#footer img { margin-bottom:-2px; width:12px; height:12px;}
```

Клас **body** відповідає за оформлення фону, там ж прописується фонові картинка.

**#content p** - тут ви прописуєте розмір і стиль шрифтів.

**h1** і **h3** - заголовки.

**div.admin** - кнопка для входу в адмін-панель.

**#footer** і **#footer img** - оформлення футера та картинка копірайтів в футері відповідно (їх відображення можна вимкнути у настройках плагіну в полі **Link**).

Картинки, які ви будете використовувати в оформленні, просто завантажте на свій хостинг через вкладку "**Медіафайли**" і пропишіть повне url-посилання

на них в CSS-файлі. Наприклад, я використав картинку "**title.png**", завантажив її у медіафайли.

Після цього я url-посилання на картинку вставляю як background у клас **body**:

1. body {
2. background:#548a32 url('http://notacat.com/wp-content/uploads/title.png') center top no-repeat;
3. }

Завершивши маніпуляції з кодом, збережіть файл як **owntheme.css** і завантажте його також на свій хостинг. В ідеалі, краще за допомогою файл-менеджера завантажити його в папку:

```
/public_html/wp-content/plugins/wp-maintenance-mode/styles
```

але це не принципово, можна також за допомогою завантаження медіафайлів. Далі слід повне url-посилання вставити у відповідне поле в настройках плагіну, а в списку **Theme** вибрати "**Own Style**".

Після цього зберігаємо настройки і оновлюємо сторінку свого сайту. Все готово.

Якщо ж ви хочете змінити шаблон повністю, чи додати свої класи, то просто необхідно змінити файл **site.php**.