

## ПРАКТИЧНА РОБОТА №1

**Тема:** Основні засоби html.

**Мета:** Познайомитися із синтаксисом, основними тегами й атрибутами мови html.

**Знати:** Структуру html-документу, основні теги мови й методи форматування тексту.

**Вміти:** Компонувати текстову інформацію при створенні гіпертекстових документів з використанням простих текстових редакторів. Формувати гіпертекстові посилання й списки й створювати на цій основі зв'язані гіпертекстові сторінки.

### Теоретичні відомості

HTML (англ. HyperText Markup Language) – стандартна мова розмітки гіпертекстових документів (веб-сторінок) в Інтернеті.

Мова HTML інтерпретується браузером. Отриманий в результаті інтерпретації відформатований текст відображається на екрані монітора комп'ютера або мобільного пристрою.

Ми будемо розглядати HTML5, дана версія була опублікована в 2014 році.

Мета розробки HTML5 – поліпшення рівня підтримки мультимедіа-технологій з одночасним збереженням зворотної сумісності, зручності читання коду для людини і простоти аналізу для парсерів.

Розробкою та впровадженням стандартів для мережі Інтернет, в тому числі і стандартів для мови HTML займається організація W3C (англ. World Wide Web Consortium – консорціум Всесвітньої павутини). Переглянути інформацію про HTML5 на їхньому веб-сайті можна за наступним посиланням: <https://www.w3.org/TR/html5/>

Розглянемо структуру html-документів та основні html-теги.

### Синтаксис html

`<назва_тегу> контент </назва_тегу>`

**Приклад 1.1.** Приклад HTML-тегу:

`<p>Мій перший HTML-параграф</p>`

**Приклад 1.2.** Приклад структури HTML5-документу:

`<!DOCTYPE html>`

```
<html>
  <head>
    <title>Назва сторінки</title>
  </head>
  <body>
    <h1>Заголовок статті</h1>
    <p>Абзац тексту</p>
  </body>
</html>
```

Для того, щоб побачити результат роботи даного коду, зробіть наступні кроки:

*Крок 1:* Відкрийте Notepad (або ін. текстовий редактор).

*Крок 2:* Скопіюйте код з Прикладу 2.

*Крок 3:* Збережіть файл з розширенням html.

*Крок 4:* Відкрийте збережений файл у будь-якому вебпереглядачі.

Крім Notepad можна використовувати спеціалізовані текстові редактори, що полегшують роботу з кодом за допомогою додаткових функцій, таких, наприклад, як підсвічування коду. Наведемо деякі безкоштовні редактори для роботи з html-кодом.

#### **Безкоштовні редактори для Windows:**

– Notepad++ – Підтримує FTP та SFTP за допомогою плагіну; підсвічування синтаксису.

– HTML-Kit – Підсвічування синтаксису, підтримка FTP.

– Crimson Editor – Легковагий редактор. Підтримка FTP.

– PSPad – Підтримка FTP; підсвічування синтаксису.

– Вбудований редактор Total Commander.

#### **Безкоштовні редактори для Linux:**

– Gedit – вільний текстовий редактор робочого середовища GNOME, Mac OS X і Microsoft Windows з підтримкою Юнікоду.

– Kate (KDE Advanced Text Editor) – текстовий редактор, який входить у склад середовища робочого столу KDE. Поширюється згідно GNU General Public License.

При виборі необхідно враховувати зручність, функціональність, наскільки свіжа остання версія і чи триває підтримка.

#### **Кросплатформові редактори:**

– Eclipse – Проекти PHPEclipse і PHP Development Tools. З додатковими плагінами підтримує SVN, CVS, моделювання баз даних, доступ по SSH/FTP, навігація по базі даних, інтеграція з Trac, та інше.

– Bluefish – Багатоцільовий редактор з підтримкою синтаксису PHP, вбудованою PHP документацією і т.д. З GVFS підтримує SFTP, FTP, WebDAV і SMB.

– jEdit –Versatile вільний/open source редактор. Підтримує SFTP і FTP.

– Sublime Text – кросплатформовий пропрієтарний текстовий редактор. Підтримує плагіни на мові програмування Python.

**Приклад 1.3.** Приклад структури сайту (стрілки ілюструють один з можливих варіантів посилань сторінок одна на одну):

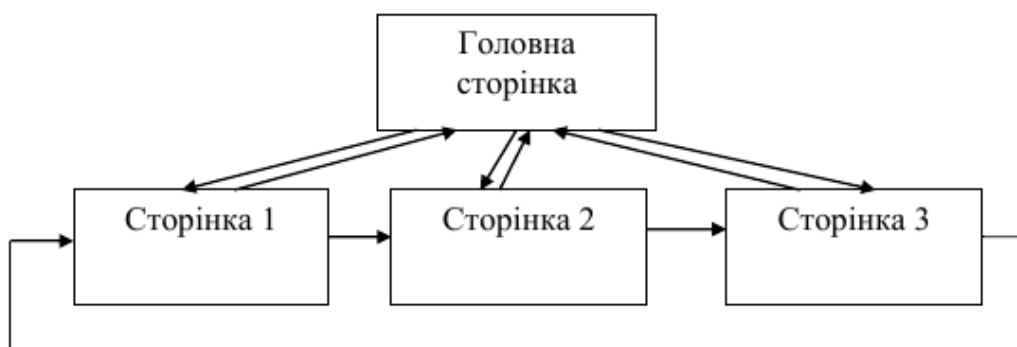


Рисунок 1.1 - Схема перепосилання вебсторінок

Розглянемо детальніше html-код з прикладу 2.

Для початку, зверніть увагу, що теги — це контейнери, які можуть вкладатися один один.

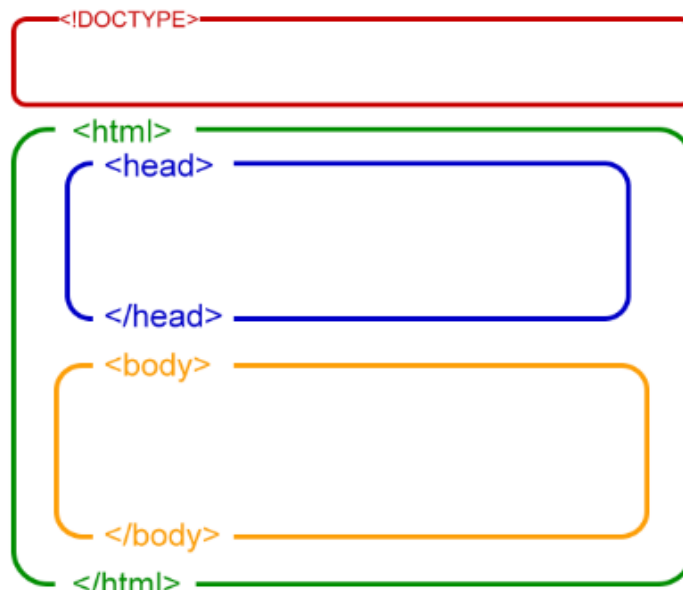


Рисунок 1.2 - Основні контейнери HTML-сторінки

DOCTYPE декларує визначення типу документа HTML.

Для того, щоб вказати вебоглядачу, що далі будемо використовувати HTML саме версії 5, достатньо написати `<!DOCTYPE html>`. Читаючи навчальні матеріали про використання HTML звертайте увагу, на основі якої версії підготовлено матеріал. Попередні версії мають доволі великий, не зручний для запам'ятовування та використання `doctype`. Його вже використовувати не потрібно.

Після оголошення версії й типу документа необхідно позначити його початок і кінець. Це робиться за допомогою тега-контейнера `<HTML>`. Потім, між тегам `<HTML>` і `</HTML>` варто розмістити заголовок і тіло документа.

Текст між `<head>` та `</head>` містить інформацію про поточний документ, таку як заголовок, ключові слова, які можуть використовуватися пошуковими машинами, та інші дані, які не вважаються вмістом документа.

Текст між `<title>` та `</title>` служить для ідентифікації вмісту документа (зовнішнього заголовка документа). Цей текст ви будете бачити у заголовку вікна або вкладки вебоглядача.

Текст між `<body>` і `</body>` описує видимий вміст сторінки.

Текст між `<h1>` і `</h1>` описує заголовок.

Текст між `<p>` і `</p>` визначає параграф.

## **HTML заголовки**

HTML заголовки визначаються тегамі від `<h1>` до `<h6>`, наприклад:

```
<h1>Це Заголовок 1-го рівня</h1>
<h2>Це Заголовок 2-го рівня</h2>
<h3>Це Заголовок 3-го рівня</h3>
```

Цифра визначає розмір заголовку – найбільший заголовок – 1, найменший – 6. Їх використання можна порівняти зі стилями Заголовок 1, Заголовок 2, Заголовок 3 в так далі у текстових процесорах на кшталт MS Word.

## **HTML параграфи**

HTML параграфи визначаються тегом `<p>`, наприклад:

```
<p>Це параграф</p>
<p>Це інший параграф</p>
```

## **HTML Посилання**

HTML посилання визначаються тегом `<a>`, наприклад:

```
<a href="http://www.site.com">Це посилання</a>
```

## HTML Зображення

HTML зображення визначається тегом `<img>`.

Файл зображення (src), альтернативний текст (alt), розмір зображення (width and height) надаються як атрибути, наприклад:

```

```

## HTML Атрибути

HTML елементи можуть мати атрибути.

Атрибути містять додаткову інформацію про елемент, завжди визначаються в початковому тегу. Атрибути є парами ім'я/значення як: `name="value"`. Приклади:

Атрибут мови `<html lang="en-US">`

Атрибут назви `<p title="About site">`

Атрибут посилання `<a href="http://www.site.com">Це посилання</a>`

Атрибути розміру ``

Атрибут альтернативного тексту ``

Таблиця 1.1 - HTML атрибути

Атрибути	Опис
Alt	Вказує альтернативний текст для зображення
Disabled	Вказує, що вхідний елемент повинен бути відключений
href	Вказує URL-адресу (веб-адресу) для зв'язку
Id	Вказує унікальний ідентифікатор для елемента
Src	Вказує URL-адресу (веб-адресу) для зображення
Style	Визначає CSS стиль вбудованого елемента
Title	Визначає додаткову інформацію про елемент (відображається у вигляді підказки)
Value	Визначає значення (текстовий зміст) для вхідного елемента.

**Приклад 4.** Приклад HTML5-документу з використанням css-елементів для вказання стилю тегів:

```
<!DOCTYPE html>
<html>
<body>
    <h2 style="color:red">Я червоний заголовок</h2>
    <h2 style="color:blue">Я синій заголовок</h2>
</body>
</html>
<body style="background-color:lightgrey">
    <h1 style="font-family:verdana">Це заголовок</h1>
    <p style="font-family:courier">Це параграф</p>
    <h1 style="font-size:300%">Це заголовок</h1>
    <p style="font-size:160%">Це параграф</p>
    <h1 style="text-align:center">Це заголовок</h1>
```

Таблиця 1.2 - Html-елементи форматування тексту

Тег	Опис
<b>	Жирний текст
<strong>	Жирний текст, визначає важливість виділеного тексту
<i>	Курсив
<em>	Курсив, визначає важливість виділеного тексту
<small>	Зменшує розмір шрифту на одну умовну одиницю. В HTML розмір шрифту вимірюється в умовних одиницях від 1 до 7. Середній розмір тексту, що використовується за замовчуванням – 3. Допускається застосування вкладених тегів <small>, при цьому розмір шрифту буде менше з кожним вкладеним рівнем, але не може бути менше, ніж 1.
<sub>	Нижній індекс
<sup>	Верхній індекс

Таблиця 1.3 - Вибір кольору шрифта html — сторінки

Найм. коліру	Код	Приклад	
aqua	##00FFFF	HTML-код: <p><font color="#ff0000">червоний колір</font></p>	Вигляд у браузері
black	##000000		червоний колір
blue	##0000FF	<p><font color="blue">Синій колір</font></p>	Синій колір
fuchsia	##FF00FF		
gray	##808080		
green	##008000		
lime	##00FF00		

maroon	##800000	
navy	##000080	
olive	##808000	
purple	##800080	
red	##FF0000	
silver	##C0C0C0	
teal	##008080	
white	##FFFFFF	
yellow	##FFFF00	

Таблиця 1.4 - Html-теги списків

Тег	Опис
<ul>	Ненумерований список
<ol>	Нумерований список
<li>	Елемент списку

### Приклад 5. Приклад ненумерованого списку.

```
<!DOCTYPE html>
<html>
<body>
  <h2>Невідсортований список з типовими позначками</h2>
  <ul>
    <li> Яблука </li>
    <li> Банани </li>
    <li> Лимони </li>
    <li> Мандарини </li>
  </ul>
</body>
</html>
```

Таблиця 1.5 - HTML-теги таблиці

Тег	Опис
<table>	Таблиця
<th>	Комірка заголовку таблиці
<tr>	Рядок таблиці
<td>	Комірка таблиці
<caption>	Заголовок таблиці

**Приклад 6.** Приклад таблиці з використанням css-елементів для вказання стилю її елементів:

```

<!DOCTYPE html>
<html>
<head>
  <style>
    table, th, td {
      border: 1px solid black;
      border-collapse: collapse;
    }
  </style>
</head>
<body>
  <table style="width:100%">
    <tr>
      <td>Jill</td>
      <td>Smith</td>
      <td>50</td>
    </tr>
    <tr>
      <td>Eve</td>
      <td>Jackson</td>
      <td>94</td>
    </tr>
    <tr>
      <td>John</td>
      <td>Doe</td>
      <td>80</td>
    </tr>
  </table>
</body>
</html>

```

### Результат прикладу

Jill	Smith	50
Eve	Jackson	94
John	Doe	80

Таблиця 1.6 - HTML-теги для групування елементів

Тег	Опис
<div>	є блоковим елементом і призначений для виділення фрагмента документа з метою зміни його стилю. Щоб не описувати кожен раз
	стиль всередині тега, можна виділити стиль в зовнішню таблицю
	стилів, а для тегу додати атрибут class або id з ім'ям селектора.



<span>	призначений для визначення стилю малих елементів документа. За допомогою тега <span> можна виділити частину інформації
	всередині інших тегів і встановити для неї свій стиль. Наприклад,
	всередині абзацу <p> можна змінити колір і розмір першої літери,
	якщо додати початковий і кінцевий тег <span> і визначити для
	нього стиль тексту. Щоб не описувати кожен раз стиль всередині тега, можна виділити стиль в зовнішню таблицю стилів, а для тегу
	додати атрибут class або id з ім'ям селектора.

## Елементи розмітки веб-сторінок в HTML5

HTML5 пропонує нові семантичні елементи, які визначають різні частини веб-сторінки, наприклад: <header>, <nav>, <section>, і <footer>.

Семантичні теги - це теги, які призначені для того щоб комп'ютерні програми (пошукові системи, збирачі інформації, мовні браузері і т.д.), розуміли який тип інформації закладений в даних тегах.

Політика специфікації HTML5 полягає в тому, щоб кожен HTML-тег мав свою семантику. Наведемо простий приклад, є теги <s></s> вони роблять текст закресленим. Раніше пошукова система зайшовши на сайт і побачивши цей тег, ніяк не реагувала на нього, тепер вона повинна розуміти, що це не просто закреслений текст, а наприклад що це позначення старого цінника у товару або застарілої (неактуальної) інформації.

Наступний приклад наводить використання семантичних тег у розподілі частин документа.

**Приклад 7.** Макет сайту у кілька колонок:

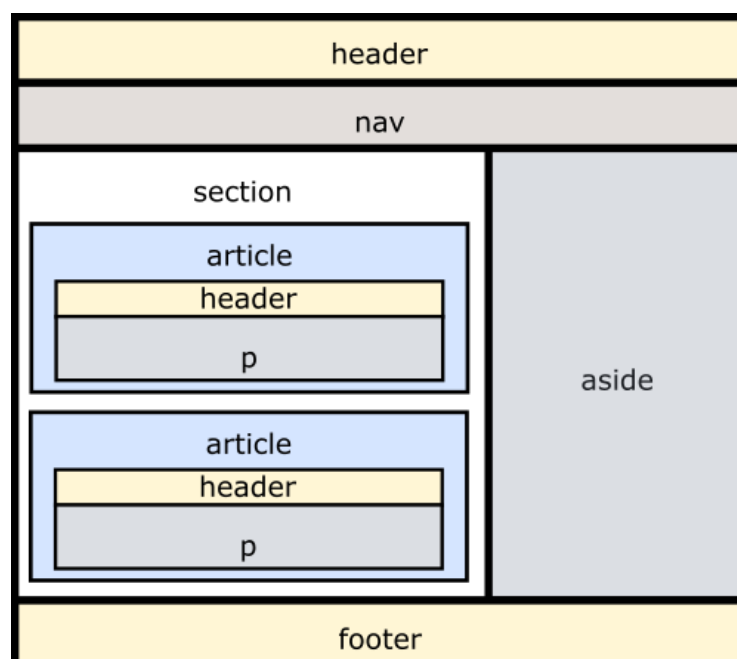


Рисунок 1.3 - Приклад макету сторінки

Таблиця 1.7 - Елементи розмітки веб-сторінок в HTML5

Тег	Опис
Header	Заголовок документа або розділу. Семантичний акцент: пошукова система буде використовувати дані цього контейнера для основного тексту при виведенні сторінки користувачам у результатах пошуку.
Nav	важливі посилання сторінки. Семантичний акцент: пошукова система обов'язково перейде за посиланнями в цьому розділі для індексації наступних сторінок.
Section	семантично окремі одиниці розділів сторінки. Семантичний акцент: пошукова система вважає, що інформація в різних секціях не пов'язана або слабопов'язана між собою.
Article	основний контент сторінки. Семантичний акцент: пошукова система вважає, що інформація розташована між цими тегами, є основною на HTML-сторінці.
Aside	бічна панель (сайдбар) сайту. Семантичний акцент: пошукова система вважає, що інформація розташована між цими тегами, є частиною сайдбару (бічний панелі).
Footer	підвалини сайту. Семантичний акцент: пошукова система вважає, що інформація розташована між цими тегами, є підвалом сайту.

Таблиця 1.8 - HTML-елементи, що містять інформацію про поточний документ

Тег	Опис
<head>	Інформація про документ
<title>	Назва документа
<base>	Адреса за замовчуванням або мета за замовчуванням для всіх посилань на сторінці
<link>	Встановлює зв'язок із зовнішнім документом на зразок файлу зі стилями або з шрифтами.
<meta>	Визначає мета теги, які використовуються для зберігання інформації призначеної для браузерів і пошукових систем. Наприклад, механізми пошукових систем звертаються до метатегів для отримання опису сайту, ключових слів та інших даних.
<script>	Визначає сценарій на стороні клієнта
<style>	Визначає інформацію про стилі для документу

## Завдання

Створити статичний сайт, що містить мінімум 6 web-сторінок з тематики – тлумачний словник з ілюстраціями за тематикою своєї спеціальності.

Обов'язково використати:

- теги форматування тексту (розмір, стиль, колір, фон),
- теги для групування елементів,
- створення списків,
- створення таблиць,
- посилань,
- додавання малюнків,
- елементи розмітки веб-сторінок `<header>`, `<nav>`, `<section>`, і `<footer>`.

Розташування елементів на сторінці зробити за допомогою табличного верстання. Не використовувати CSS!

Загальний приклад вигляду сторінок

ТЛУМАЧНИЙ СЛОВНИК	
Термінологія з ....	
<u>Термін 1</u>	<u>Термін 4</u>
<u>Термін 2</u>	<u>Термін 5</u>
<u>Термін 3</u>	<u>Термін 6</u>
©Всі права застережено. Виконав (ваше прізвище)	

Рисунок 1.4 - Головна сторінка index.html



Рисунок 1.5 - Сторінка терміну 1 term1.html

За аналогією створіть решту сторінок.

### Контрольні питання:

1. Що таке HTML?
2. Яким чином забезпечується форматування тексту в HTML-документі?
3. Яка мета розробки HTML5?
4. Який синтаксис HTML?
5. Структура HTML-документа?
6. Яким чином здійснюється взаємодія між HTML-документом і сервером?
7. Які текстові редактори для роботи з HTML Ви знаєте?
8. Що означає DOCTYPE у HTML-документі?
9. Що таке заголовок HTML-документа?
10. Що таке тіло HTML-документа?
11. Для чого служить текст між тегами <title>?
12. Яким тегом забезпечуються посилання на інші документи HTML і його атрибути?
13. Яким тегом визначаються параграфи?
14. Яким тегом визначаються заголовки?
15. Яким тегом визначається найбільший заголовок?
16. Яким тегом визначається найменший заголовок?

17. Що визначається тегом `<img>`?
18. Перелічити HTML - елементи форматування тексту.
19. Які теги списків ви знаєте?
20. Що визначає тег `<table>`?
21. Яким тегом визначається заголовок таблиці?
22. Для чого використовують тег `<div>`?
23. Для чого використовують тег `<span>`?
24. Що визначають `<header>`, `<nav>`, `<section>`, і `<footer>` елементи?