Національний технічний університет України «КПІ ім. Ігоря Сікорського»

Факультет Інформатики та Обчислювальної Техніки

Кафедра інформаційних систем та технологій

Самостійна робота №1

з дисципліни «WEB»

на тему

СТРУКТУРА HTML-ДОКУМЕНТА. ВИБІР ПРЕДМЕТНОЇ ГАЛУЗІ. РОБОТА З ПОСИЛАННЯМИ, ТАБЛИЦЯМИ, ЗОБРАЖЕННЯМИ, СПИСКАМИ, ФОРМАМИ в HTML-ДОКУМЕНТІ

Виконав:

студент групи ІС-34 Шайдецький Д. В.

Викладач:

доц. Проскура С.Л

Київ 2024

**Мета:** придбати практичні навички роботи з HTML-документом, таблицями, , зображеннями, посиланнями, списками,формами.

**Закріпити теорію та відпрацювати приклади за наступними посиланнями:**

* **Що таке HTML?**

HTML є стандартною мовою розмітки для створення веб-сторінок.

* HTML означає *Hyper Text Markup Language* (мова гіпертекстової розмітки)
* HTML описує структуру веб-сторінок за допомогою розмітки
* HTML-елементи є будівельними блоками сторінок HTML
* Елементи HTML представлені *тегами*
* *Теги* HTML позначають фрагменти змісту, такі як header (заголовок), paragraph (параграф), table (таблиця) та ін.
* Браузери не відображають HTML теги, але використовують їх для відображення змісту web-сторінки
* **HTML Редактори коду та IDE.**

Писати HTML код можна, використовуючи стандартні програми Блокнот (на Windows) та TextEdit (на MacOS). Але зараз існує досить велика кількість різних професійних редакторів коду, за допомогою яких можна писати HTML-код. Хоча для вивчення мови HTML рекомендується спочатку писати код як-раз за допомогою простих текстових редакторів - стандартних Блокнота або TextEdit. А навчившись писати простий HTML код, потім перейти до більш професійних редакторів або IDE.

Виконайте чотири кроки нижче, щоб створити свою першу вебсторінку за допомогою Блокноту або TextEdit.

також можно використовувати інші текстові редактори потипу VS cod та інші.

* **HTML Основні приклади.**

## HTML Документи

Всі HTML документи мають починатися з оголошення типу документа: <!DOCTYPE html>.

Сам HTML документ починається тегом <html> и закінчується тегом </html>.

Видима частина HTML документа знаходиться поміж <body> та </body>.

### Приклад

<!DOCTYPE html>

<html>

<body>

<h1>Мій перший заголовок</h1>

<p>Мій перший параграф.</p>

</body>

</html>

## HTML Заголовки

HTML заголовки визначаються за допомогою тегів від <h1> до <h6>.

Тегом <h1> визначаються найбільш важливі заголовки. Тегом <h6> визначаються найменш важливі заголовки:

<h1>Це заголовок 1</h1>

<h2>Це заголовок 2</h2>

<h3>Це заголовок 3</h3>

## HTML Параграфи

HTML параграфи визначаються за допомогою тега <p>:

### Приклад

<p>Це параграф.</p>

<p>Це інший параграф.</p>

## HTML Посилання

HTML посилання визначаються за допомогою тега <a>:

### Приклад

<a href="https://w3schoolsua.github.io/">Це посилання</a>

## HTML Зображення

HTML зображення визначаються за допомогою тега <img>.

Початковий файл src, альтернативний текст alt, ширина width і висота height надаються в якості атрибутів:

### Приклад

<img src="../images/w3schools.jpg" alt="W3Schools" width="104" height="142">

## HTML Кнопки

HTML кнопки визначаються за допомогою тега <button>:

### Приклад

<button>Натисніть мене</button>

## HTML Списки

HTML списки визначаються за допомогою тегів <ul> (невпорядкований / маркований список) або <ol> (упорядкований / нумерований список), з подальшими тегами <li> (пункти списку):

### Приклад

<h2>Невпорядкований список HTML</h2>

<ul>

<li>Кава</li>

<li>Чай</li>

<li>Молоко</li>

</ul>

<h2>Впорядкований список HTML</h2>

<ol>

<li>Кава</li>

<li>Чай</li>

<li>Молоко</li>

</ol>

* **HTML Елементи**

HTML елемент зазвичай складається із тегу початкового і тегу кінцевого, а зміст вставляється поміж ними:

<tagname>Зміст розміщується тут...</tagname>

HTML елемент - це все від початкового тегу до кінцевого тегу:

<p>Мій перший параграф.</p>

Вкладені HTML елементи

HTML елементи можуть бути вкладеними (елементи можуть містити елементи).

Всі документи HTML складаються з вкладених елементів HTML.

Цей приклад містить чотири елементи HTML:

### Приклад

<!DOCTYPE html>

<html>

<body>

<h1>Мій перший заголовок</h1>

<p>Мій перший параграф.</p>

</body>

</html>

## Порожні HTML елементи

HTML елементи без вмісту називаються порожніми елементами.

Елемент <br> - це порожній елемент без закриваючого тегу (тег <br> визначає обрив рядка):

### Приклад

<p>Це параграф<br> з обривом рядка.</p>

* **HTML Атрибути.**

## HTML Атрибути

* Всі HTML елементи можуть мати атрибути
* Атрибути надають додаткову інформацію про елемент
* Атрибути завжди вказуються в початковому тезі
* Атрибути зазвичай входять в пари ім'я/значення, наприклад: ім'я="значення"

## Атрибут href

HTML посилання визначаються за допомогою тега <a>. Адреса посилання вказується в атрибуті href:

### Приклад

<a href="https://w3schoolsua.github.io/">Це посилання</a>

## Атрибут src

HTML зображення визначаються за допомогою тега <img>.

Ім'я файла джерела зображення вказується в атрибуті src:

### Приклад

<img src="../images/img\_girl.jpg">

## Атрибути width (ширина) та height (висота)

HTML зображення також мають атрибути width (ширина) та height (висота), які визначають ширину та висоту зображення:

### Приклад

<img src="../images/img\_girl.jpg" width="500" height="600">

## Атрибут alt

Атрибут alt вказує альтернативний текст, який буде використовуватись, якщо зображення не може бути відображене.

Значення атрибута alt може бути прочитано програмами читання з екрану (скрінрідерами). Таким чином, можна "прослуховувати" веб-сторінки, наприклад, людина з вадами зору може "чути" елемент, тобто, буде знати, що зображено на картинці.

### Приклад

<img src="../images/img\_girl.jpg" alt="Дівчина в жакеті">

## Атрибут style

Атрибут style використовується для вказівки стилю елемента, такого як колір, шрифт, розмір і т.д.

### Приклад

<p style="color:red">Це параграф червоного кольору.</p>

## Атрибут lang

Мова документу може бути оголошена в тезі <html>.

Мова оголошується за допомогою атрибута lang.

Оголошення мови важливо для додатків з розширеними можливостями (наприклад, засобів читання з екрану - скрінрідерів) і пошукових систем:

<!DOCTYPE html>

<html lang="en-US">

<body>

...

</body>

</html>

## Атрибут title

Тут атрибут title додається до елементу <p>. Значення атрибута title буде відображатися у вигляді підказки при наведенні миші на параграф:

### Приклад

<p title='Я виринаюча підказка'>

Це параграф.

</p>

* **HTML Параграфи**

## HTML Параграфи

HTML елемент <p> визначає параграф:

### Приклад

<p>Це параграф.</p>

<p>Це інший параграф.</p>

## HTML Відображення

Ви не можете знати напевно, де саме буде відображатися ваша вебсторінка, на яких дисплеях або пристроях. Великі або маленькі екрани і розміри вікон показують різні результати. За допомогою HTML-коду ви не можете змінити вихідне відображення, додаючи додаткові пробіли або додаткові рядки в вашому HTML-коді. Під час відображення сторінки веббраузер видаляє будь-які додаткові пробіли і додаткові рядки:

### Приклади

<p>

Цей параграф

містить багато рядків

у вихідному коді,

але браузер

ігнорує їх.

</p>

<p>

Цей параграф

містить багато місця

у вихідному коді

але браузер

ігнорує його.

</p>

## Не забувайте кінцевий тег

Більшість браузерів будуть відображати HTML правильно, навіть якщо ви забудете кінцевий тег:

### Приклад

<p>Це параграф.

<p>Це інший параграф.

## HTML Розриви рядків

HTML елемент <br> визначає розрив рядка.

Використовуйте тег <br> якщо необхідно перервати рядок (створити новий рядок), не починаючи новий параграф:

### Приклад

<p>Цей<br>параграф<br>має розриви рядка.</p>

## HTML елемент <pre>

HTML елемент <pre> визначає форматований текст.

Текст всередині елементу <pre> відображається шрифтом фіксованої ширини (зазвичай Courier), і він зберігає пробіли та розриви рядків:

### Приклад

<pre>

І ось ідем у бою життєвому.

Міцні, тверді, незламні мов граніт,

Бо плач не дав свободи ще нікому,

А хто борець, той здобуває світ.

</pre>

* **HTML Посилання**

## HTML Посилання - Гіперпосилання

HTML посилання - це гіперпосилання.

Ви можете натиснути на посилання і перейти до іншого документу.

При наведенні миші на посилання стрілка миші перетвориться в маленьку стрілку.

## HTML Посилання - Синтаксис

Гіперпосилання визначаються за допомогою HTML тега <a> (від слова anchor - *якір*):

<a href="*url*">*Текст посилання*</a>

### Приклад

<a href="https://www.w3schools.com/html/">HTML уроки для початківців онлайн</a>

## Локальні посилання

У наведеному вище прикладі використовується абсолютний URL (повна вебадреса).

Локальне посилання (посилання на той самий веб-сайт) вказується з відносним URL (без https://www....).

### Приклад

<a href="html\_images.html">HTML Зображення</a>

## HTML Кольори посилань

За замовчуванням посилання буде виглядати так (у всіх браузерах):

* Невідвідуване посилання - підкреслене і синім кольором
* Відвідуване посилання - підкреслене і фіолетовим кольором
* Активне посилання - підкреслене і червоним кольором

Ви можете змінити кольори за замовчуванням, використовуючи CSS:

### Приклад

<style>

a:link {

color: green;

background-color: transparent;

text-decoration: none;

}

a:visited {

color: pink;

background-color: transparent;

text-decoration: none;

}

a:hover {

color: red;

background-color: transparent;

text-decoration: underline;

}

a:active {

color: yellow;

background-color: transparent;

text-decoration: underline;

}

</style>

* **HTML Зображення**

## HTML Синтаксис зображень

В HTML зображення визначаються за допомогою тега <img>.

Тег <img> пустий, містить лише атрибути і не має закриваючого тега.

Атрибут src вказує URL (вебадресу) зображення:

<img src="*url*">

## Атрибут alt

Атрибут alt надає альтернативний текст для зображення, якщо користувач з якоїсь причини не може його переглянути: через повільне підключення до Інтернету, помилки в атрибуті src або якщо користувач використовує програму читання з екрана (скрінрідер).

Значення атрибута alt має описувати зображення, тобто те, що на ньому зображено:

### Приклад

<img src="img\_chania.jpg" alt="Flowers in Chania">

Якщо браузер не може знайти зображення, він відображає значення атрибута alt:

### Приклад

<img src="wrongname.gif" alt="Flowers in Chania">

## Розмір зображення - Width та Height - Ширина та Висота

Ви можете використовувати атрибут style для вказівки ширини та висоти зображення.

### Приклад

<img src="img\_girl.jpg" alt="Girl in a jacket" style="width:500px;height:600px;">

Крім того, ви можете використовувати атрибути width та height:

### Приклад

<img src="img\_girl.jpg" alt="Girl in a jacket" width="500" height="600">

Атрибути width та height завжди визначають ширину і висоту зображення в пікселях.

## Ширина і висота або стиль?

Атрибути width, height та style валідні в HTML.

Однак ми рекомендуємо використовувати атрибут style. Це перешкоджає тому, щоб таблиці стилів змінили розмір зображень:

### Приклад

<!DOCTYPE html>

<html>

<head>

<style>

img {

width: 100%;

}

</style>

</head>

<body>

<img src="html5.gif" alt="HTML5 Icon" width="128" height="128">

<img src="html5.gif" alt="HTML5 Icon" style="width:128px;height:128px;">

</body>

</html>

## Зображення в іншій папці

Якщо не вказано, браузер очікує знайти зображення в тій же папці, в якій знаходиться і вебсторінка.

Однак зображення зазвичай зберігаються в підпапках (інших папках). Потім ви повинні включити ім'я папки в атрибут src:

### Приклад

<img src="/images/html5.gif" alt="HTML5 Icon" style="width:128px;height:128px;">

## Зображення на іншому сервері

Деякі вебсайти зберігають свої зображення на спеціальних серверах зображень.

Насправді ви можете отримати доступ до зображень з будь-якої вебадреси Світової павутини, прописавши повний шлях до зображення:

### Приклад

<img src="https://www.w3schools.com/images/w3schools\_green.jpg" alt="W3Schools.com">

## Анімовані зображення

HTML дозволяє використовувати анімовані GIF-зображення:

### Приклад

<img src="programming.gif" alt="Computer Man" style="width:48px;height:48px;">

## Зображення як посилання

Щоб використовувати зображення в якості посилання, поставте тег <img> всередині тега <a>:

### Приклад

<a href="index.html">

<img src="smiley.gif" alt="HTML tutorial" style="width:42px;height:42px;border:0;">

</a>

## Плаваюче зображення

Використовуйте CSS властивість float, що дозволяє зображенню плавати праворуч або ліворуч від тексту:

### Приклад

<p><img src="smiley.gif" alt="Smiley face" style="float:right;width:42px;height:42px;">

Зображення буде плавати праворуч від тексту.</p>

<p><img src="smiley.gif" alt="Smiley face" style="float:left;width:42px;height:42px;">

Зображення буде плавати ліворуч від тексту.</p>

* **HTML Таблиці**

## Визначення HTML Таблиці

Тег <table> визначає HTML таблицю.

Кожен рядок таблиці визначається тегом <tr>. Кожен заголовок таблиці визначається тегом <th>. Кожна чарунка з даними таблиці визначається тегом <td>.

За замовчуванням текст в елементах <th> виділений жирним шрифтом і відцентрований.

За замовчуванням текст в елементах <td> є звичайним і вирівняним за лівим краєм.

### Приклад

Проста HTML таблиця:

<table style="width:100%">

<tr>

<th>Firstname</th>

<th>Lastname</th>

<th>Age</th>

</tr>

<tr>

<td>Jill</td>

<td>Smith</td>

<td>50</td>

</tr>

<tr>

<td>Eve</td>

<td>Jackson</td>

<td>94</td>

</tr>

</table>

## HTML таблиця. Як додати межу - border

Щоб додати межу до таблиці, використовуйте CSS властивість border:

### Приклад

table, th, td {

border: 1px solid black;

}

## HTML Таблиця. Як згорнути межі - border-collapse

Щоб дозволити межам таблиці згортатися в одну межу, додайте CSS властивість border-collapse:

### Приклад

table, th, td {

border: 1px solid black;

border-collapse: collapse;

}

## HTML Таблиця. Як додати внутрішній відступ в чарунках - padding

Внутрішній відступ чарунки визначає простір між вмістом чарунки та її межами.

Якщо ви не вказали внутрішній відступ, чарунки таблиці відображатимуться без відступу.

Щоб встановити внутрішній відступ, використовуйте CSS властивість padding:

### Приклад

th, td {

padding: 15px;

}

## HTML Таблиця. Як вирівняти заголовки за лівим краєм - text-align

За замовчуванням заголовки таблиць виділені напівжирним шрифтом і відцентровані.

Щоб вирівняти заголовки таблиць за лівим краєм, використовуйте CSS властивість text-align:

### Приклад

th {

text-align: left;

}

## HTML Таблиця. Як додати інтервал між межами - spacing

Інтервал між межами визначає простір між чарунками.

Щоб встановити інтервал між межами таблиці, використовуйте CSS властивість border-spacing:

### Приклад

table {

border-spacing: 5px;

}

## HTML Таблиця - Чарунка, що охоплює кілька стовпців

Щоб зробити чарунку, що охоплює більше одного стовпця, використовуйте атрибут colspan:

### Приклад

<table style="width:100%">

<tr>

<th>Name</th>

<th colspan="2">Telephone</th>

</tr>

<tr>

<td>Bill Gates</td>

<td>55577854</td>

<td>55577855</td>

</tr>

</table>

## HTML Таблиця - Чарунка, яка охоплює багато рядків

Щоб зробити чарунку, яка охоплює більше одного рядка, використовуйте атрибут rowspan:

### Приклад

<table style="width:100%">

<tr>

<th>Name:</th>

<td>Bill Gates</td>

</tr>

<tr>

<th rowspan="2">Telephone:</th>

<td>55577854</td>

</tr>

<tr>

<td>55577855</td>

</tr>

</table>

## HTML Таблиця - Додати підпис

Щоб додати підпис до таблиці, використовуйте тег <caption>:

### Приклад

<table style="width:100%">

<caption>Monthly savings</caption>

<tr>

<th>Month</th>

<th>Savings</th>

</tr>

<tr>

<td>January</td>

<td>$100</td>

</tr>

<tr>

<td>February</td>

<td>$50</td>

</tr>

</table>

## Спеціальний стиль для однієї таблиці

Щоб визначити спеціальний стиль для однієї конкретної таблиці, додайте до таблиці атрибут id:

### Приклад

<table id="t01">

<tr>

<th>Firstname</th>

<th>Lastname</th>

<th>Age</th>

</tr>

<tr>

<td>Eve</td>

<td>Jackson</td>

<td>94</td>

</tr>

</table>

### Тепер ви можете визначити спеціальний стиль для цієї таблиці:

#t01 {

width: 100%;

background-color: #f1f1c1;

}

### додати більше стилів:

#t01 tr:nth-child(even) {

background-color: #eee;

}

#t01 tr:nth-child(odd) {

background-color: #fff;

}

#t01 th {

color: white;

background-color: black;

}

**Самостійно вивчити компонент ФОРМИ. Відпрацювати приклади:**

* **HTML Форми**

## Елемент <form>

HTML елемент <form> визначає форму, яка використовується для збору користувацького вводу:

<form>

.

*елементи форми*

.

</form>

HTML-форма містить елементи форми.

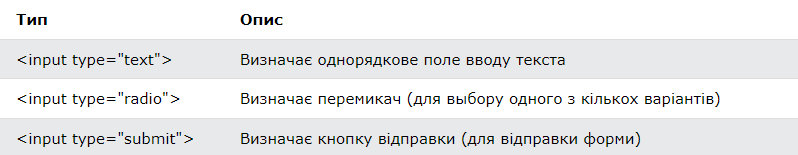
Елементи форми - це різні типи елементів вводу, такі як текстові поля, прапорці, перемикачі, кнопки надсилання тощо.

## Елемент <input>

Елемент <input> є найбільш важливим елементом форми.

Елемент <input> може відображатися кількома способами, в залежності від атрибута type.

Ось кілька прикладів:



## Введення тексту

<input type='text'> визначає однорядкове поле вводу для введення тексту - text input:

### Приклад

<form>

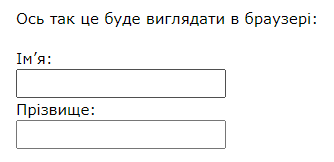
Ім’я:<br>

<input type='text' name='firstname'><br>

Прізвище:<br>

<input type='text' name='lastname'>

</form>



## Радіо кнопка введення

<input type='radio'> визначає радіокнопку.

Перемикачі (радіокнопки) дозволяють користувачу обрати ОДИН з обмеженної кількості варіантів:

### Приклад

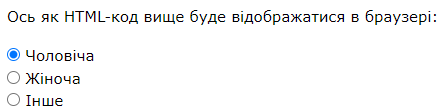
<form>

<input type='radio' name='gender" value='male' checked> Чоловіча<br>

<input type='radio' name='gender' value='female'> Жіноча<br>

<input type='radio' name='gender' value='other'> Інше

</form>



## Кнопка Submit / Надіслати

<input type='submit'> визначає кнопку для надсилання даних форми в обробник форм.

Обробник форм зазвичай являє собою серверну сторінку зі скриптом для обробки вхідних даних.

Обробник форми вказується в атрибуті action форми:

### Приклад

<form action='/action\_page.html'>

Ім’я:<br>

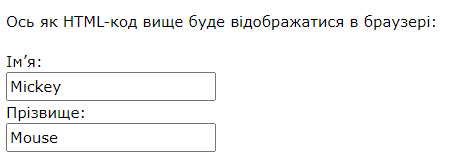
<input type='text' name='firstname' value='Mickey'><br>

Прізвище:<br>

<input type='text' name='lastname' value='Mouse'><br><br>

<input type='submit' value='Надіслати'>

</form>



## Атрибут Action

Атрибут action визначає дію, яку буде виконано при надсиланні форми.

Зазвичай дані форми відправляються на вебсторінку на сервері, коли користувач натискає кнопку надсилання (submit).

У наведеному вище прикладі дані форми відправляються на сторінку на сервері з ім’ям "/action\_page.html". Ця сторінка містить серверний скрипт, який обробляє дані форми:

<form action='/action\_page.html'>

Якщо атрибут action пропущений, дію встановлено на поточну сторінку.

## Атрибут Target

Атрибут target вказує, чи буде надісланий результат відкриватися в новій вкладці браузера, у фреймі або в поточному вікні.

За замовчуванням значення '\_self' означає, що форма буде відправлена в поточному вікні.

Щоб зробити результат форми відкритим в новій вкладці браузера, використовуйте значення '\_blank':

### Приклад

<form action='/action\_page.html' target='\_blank'>

Іншими допустимими значеннями є '\_parent', '\_top' або name, яке представляє ім’я iframe.

## Атрибут Method

Атрибут method вказує метод HTTP (GET або POST), який буде використовуватись при надсиланні даних форми:

### Приклад

<form action='/action\_page.html' method='get'>

або:

<form action='/action\_page.html' method='post'>

## Коли використовувати метод GET?

Метод за замовчуванням при надсиланні даних форми - GET.

Однак при використанні GET відправлені дані форми будуть видимі в полі адреси сторінки (адресному рядку):

## Коли використовувати метод POST?

Завжди використовуйте POST, якщо дані форми містять конфіденційну або особисту інформацію. Метод POST не відображає відправлені дані форми в адресному рядку сторінки.

Примітка по POST:

* POST не має обмежень за розмірами і може використовуватися для надсилання великих обсягів даних
* Надіслані з POST форми можна додати в закладки

## Атрибут Name

Кожне поле вводу повинно мати атрибут name для надсилання.

Якщо атрибут name опущений, дані його поля вводу не будуть відправлені взагалі.

Цей приклад буде відправляти тільки поле вводу "Last name" (Прізвище):

### Приклад

<form action='/action\_page.html'>

First name:<br>

<input type='text' value='Mickey'><br>

Last name:<br>

<input type='text' name='lastname' value='Mouse'><br><br>

<input type='submit' value='Надіслати'>

</form>

## Групування даних форми за допомогою <fieldset>

Елемент <fieldset> використовується для групування пов’язаних даних у формі.

Елемент <legend> визначає заголовок для елемента <fieldset>.

### Приклад

<form action='/action\_page.html'>

<fieldset>

<legend>Особиста інформація:</legend>

Ім’я:<br>

<input type='text' name='firstname' value='Mickey'><br>

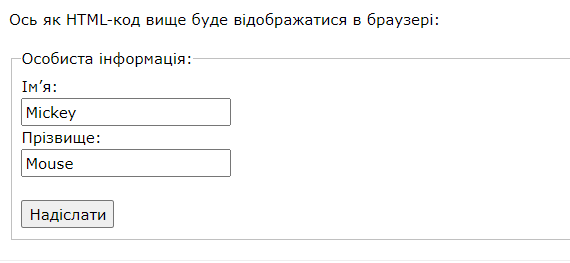
Прізвище:<br>

<input type='text' name='lastname' value='Mouse'><br><br>

<input type='submit' value='Надіслати'>

</fieldset>

</form>

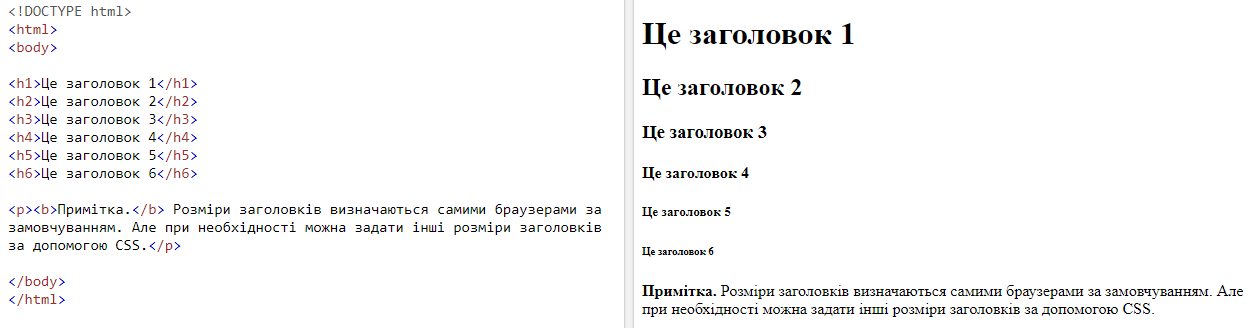


Відпрацювати всі приклади підрозділа «HTML Основи» за посиланням: <https://w3schoolsua.github.io/html/html_examples.html>

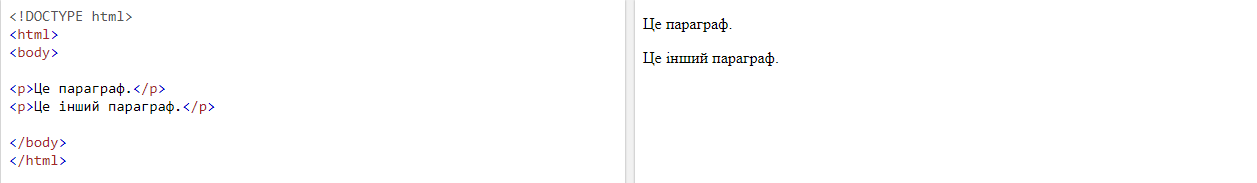
HTML документ:



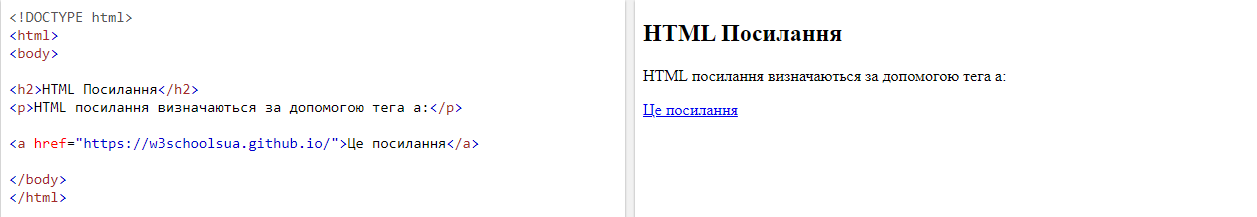
HTML заголовки:



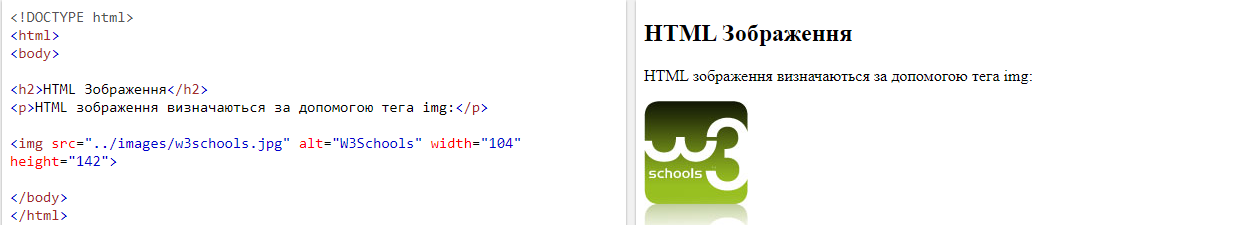
HTML параграфи:



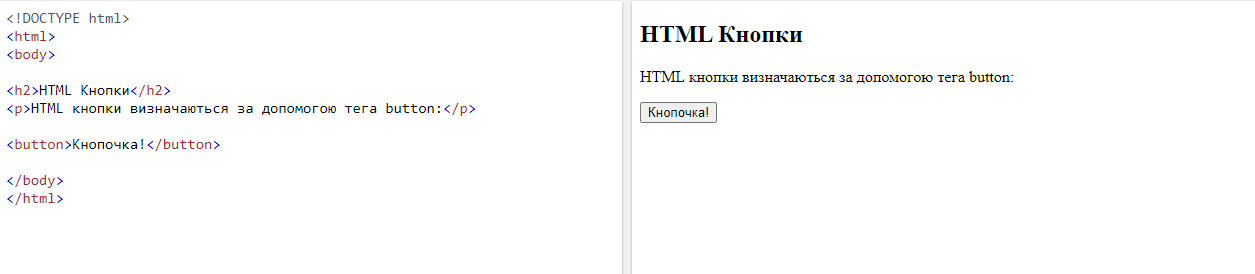
HTML посилання:



HTML зображення:



HTML кнопки:



HTML списки:

