

Ласкаво просимо на перший курс з HTML!

Цей курс розроблено для початківців, які бажають вивчити основи HTML – мови розмітки, що є фундаментом всіх веб-сторінок. Ми крок за кроком розглянемо ключові концепції та елементи, які допоможуть вам створювати власні прості веб-сторінки.

Урок 1: Що таке HTML?

1.1. Вступ до HTML

HTML розшифровується як HyperText Markup Language (Мова розмітки гіпертексту). Це не мова програмування в традиційному сенсі, а скоріше мова для структурування вмісту веб-сторінок. За допомогою HTML ми повідомляємо браузеру, як відображати текст, зображення, посилання та інші елементи на екрані.

1.2. Як працює HTML?

Уявіть, що ви пишете статтю. HTML дозволяє вам сказати: "Це заголовок", "Це абзац", "Це зображення", "Це посилання на іншу статтю" і так далі. Браузер читає цей "інструктаж" і відповідно відображає вміст для користувача.

1.3. Основні терміни:

- * Елемент (Element): Складається з початкового тегу, кінцевого тегу та вмісту між ними. Наприклад, `<p>Це абзац</p>`.

- * Тег (Tag): Мітка, яка вказує браузеру на початок і кінець елемента. Теги зазвичай оточені кутовими дужками (наприклад, `<p>`). Більшість тегів мають відкриваючу (`<p>`) та закриваючу (`</p>`) частини. Деякі теги є "самозакривними" або "порожніми" (наприклад, ``, `
`).

- * Атрибут (Attribute): Додаткова інформація, яка надає тегу специфічні властивості. Атрибути завжди вказуються у відкриваючому тегу і складаються з імені та значення, розділених знаком рівності (наприклад, ``).

1.4. Простий приклад HTML-документа:

```
<!DOCTYPE html>
<html>
<head>
  <title>Моя перша сторінка</title>
</head>
<body>

  <h1>Привіт, світ!</h1>
  <p>Це мій перший абзац HTML.</p>
  <a href="https://www.google.com">Перейти на Google</a>

</body>
</html>
```

Розберемо цей приклад:

- * `<!DOCTYPE html>`: Декларація типу документа. Вона повідомляє браузеру, що це HTML5 документ.

- * `<html>...</html>`: Кореневий елемент, який охоплює весь вміст HTML-сторінки.

- * `<head>...</head>`: Містить метадані про документ (наприклад, заголовок сторінки, кодування символів), яка не відображається безпосередньо на сторінці.

- * `<title>Моя перша сторінка</title>`: Встановлює заголовок сторінки, який відображається у вкладці браузера.

- * `<body>...</body>`: Містить весь видимий вміст веб-сторінки (текст, зображення, посилання тощо).
- * `<h1>Привіт, світ!</h1>`: Заголовок першого рівня. `<h1>` є найбільшим заголовком, `<h6>` – найменшим.
- * `<p>Це мій перший абзац HTML.</p>`: Звичайний абзац тексту.
- * `Перейти на Google`: Посилання на іншу сторінку. `href` – це атрибут, який вказує URL, на який веде посилання.

Урок 2: Базова структура HTML-документа

Кожен HTML-документ має стандартну базову структуру. Розуміння цієї структури є ключовим для правильного створення веб-сторінок.

2.1. Обов'язкові елементи:

- * `<!DOCTYPE html>`: Завжди починає HTML-документ. Це інструкція для браузера про тип документа.
- * `<html>`: Корневий елемент, який є контейнером для всіх інших HTML-елементів на сторінці (крім `<!DOCTYPE html>`).
- * `<head>`: Містить метадані про HTML-документ, які не відображаються безпосередньо на сторінці, але важливі для її функціонування та відображення.
 - * `<title>`: Визначає заголовок документа, який відображається у вкладці браузера або у вікні.
 - * `<meta>`: Використовується для надання метаданих про HTML-документ (наприклад, кодування символів, опис сторінки для пошукових систем). Наприклад, `<meta charset="UTF-8">` вказує кодування символів UTF-8, що є важливим для коректного відображення українських літер.
 - * `<link>`: Використовується для підключення зовнішніх файлів, наприклад, CSS-файлів для стилізації.
 - * `<style>`: Дозволяє вбудовувати CSS-стилі безпосередньо в HTML-документ.
 - * `<script>`: Використовується для вбудовування або підключення JavaScript-коду.
- * `<body>`: Містить весь вміст, який буде відображатися користувачеві в браузері. Це включає текст, зображення, посилання, відео, таблиці, форми тощо.

2.2. Приклад базової структури:

```
<!DOCTYPE html>
<html lang="uk">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Базова HTML-сторінка</title>
  </head>
  <body>

    <h1>Ласкаво просимо!</h1>
    <p>Це приклад базової HTML-сторінки.</p>

  </body>
</html>
```

Пояснення до нового елемента `<meta name="viewport" ...>`:

Цей тег `<meta>` є дуже важливим для адаптивного дизайну (Responsive Web Design). Він вказує браузеру, як масштабувати сторінку на різних пристроях (смартфонах, планшетах, комп'ютерах).

* width=device-width: Встановлює ширину області перегляду рівною ширині пристрою.

* initial-scale=1.0: Встановлює початковий рівень масштабування.

Урок 3: Робота з текстом

HTML надає безліч тегів для форматування тексту.

3.1. Заголовки (Headings):

Використовуються для визначення заголовків та підзаголовків. Існує 6 рівнів заголовків: від <h1> (найбільший) до <h6> (найменший).

<h1>Заголовок першого рівня</h1>

<h2>Заголовок другого рівня</h2>

<h3>Заголовок третього рівня</h3>

<h4>Заголовок четвертого рівня</h4>

<h5>Заголовок п'ятого рівня</h5>

<h6>Заголовок шостого рівня</h6>

Важливо: Використовуйте заголовки для структуризації вмісту, а не для зміни розміру шрифту. Розмір шрифту слід змінювати за допомогою CSS.

3.2. Абзаци (Paragraphs):

Тег <p> використовується для визначення абзаців тексту.

<p>Це мій перший абзац тексту на цій сторінці. Він містить декілька речень.</p>

<p>Це другий абзац. Кожен абзац починається з нового рядка.</p>

3.3. Розриви рядків (Line Breaks):

Тег
 використовується для створення примусового розриву рядка. Це порожній тег (не має закриваючого тегу).

<p>Це текст,
 який розбито
 на декілька рядків.</p>

3.4. Горизонтальні лінії (Horizontal Rules):

Тег <hr> використовується для створення горизонтальної лінії, яка візуально розділяє вміст. Це також порожній тег.

<p>Перший блок тексту.</p>

<hr>

<p>Другий блок тексту, відокремлений горизонтальною лінією.</p>

3.5. Форматування тексту:

* або : Жирний шрифт. має більшу семантичну вагу (вказує на важливість тексту).

* <i> або : Курсив. має більшу семантичну вагу (вказує на наголос).

* <mark>: Маркер (виділення тексту).

* <small>: Зменшений текст.

* : Закреслений текст (видалений).

* <ins>: Підкреслений текст (вставлений).

* <sub>: Нижній індекс.

* <sup>: Верхній індекс.

<!-- end list -->

<p>Цей текст є жирним за допомогою тегу .</p>

<p>Цей текст є важливим за допомогою тегу .</p>

<p>Цей текст є <i>курсивом</i> за допомогою тегу <i>.</p>

<p>Цей текст є наголошеним за допомогою тегу .</p>

```
<p>Цей <mark>виділений</mark> текст.</p>
<p><small>Маленький текст.</small></p>
<p><del>Цей текст видалено.</del></p>
<p><ins>Цей текст вставлено.</ins></p>
<p>Н<sub>2</sub>О (вода)</p>
<p>х<sup>2</sup> (х в квадраті)</p>
```

Урок 4: Списки

HTML дозволяє створювати впорядковані, неупорядковані та списки визначень.

4.1. Невпорядковані списки (Unordered Lists):

Створюються за допомогою тегу (unordered list). Кожен елемент списку позначається тегом (list item). За замовчуванням елементи списку позначаються маркерами.

```
<h2>Мої улюблені фрукти:</h2>
```

```
<ul>
  <li>Яблука</li>
  <li>Банани</li>
  <li>Апельсини</li>
</ul>
```

4.2. Впорядковані списки (Ordered Lists):

Створюються за допомогою тегу (ordered list). Кожен елемент списку також позначається тегом . За замовчуванням елементи списку нумеруються.

```
<h2>Кроки для приготування чаю:</h2>
```

```
<ol>
  <li>Нагрійте воду.</li>
  <li>Покладіть чайний пакетик у чашку.</li>
  <li>Залийте гарячою водою.</li>
  <li>Додайте цукор або молоко за смаком.</li>
</ol>
```

4.3. Атрибути для впорядкованих списків:

* type: Змінює тип нумерації (наприклад, 1 для чисел, а для малих літер, А для великих літер, і для малих римських цифр, І для великих римських цифр).

* start: Встановлює початкове значення нумерації.

```
<!-- end list -->
```

```
<h2>Список з римськими цифрами, що починається з V:</h2>
```

```
<ol type="I" start="5">
  <li>П'ятий елемент</li>
  <li>Шостий елемент</li>
</ol>
```

```
<h2>Список з малими літерами:</h2>
```

```
<ol type="a">
  <li>Перший елемент</li>
  <li>Другий елемент</li>
</ol>
```

4.4. Вкладені списки:

Списки можна вкладати один в одного для створення ієрархічної структури.

`<h2>Меню:</h2>`

``

`Головні страви`

``

`Борщ`

`Вареники`

``

``

`Десерти`

``

`Пампушки`

`Сирники`

``

``

``

Урок 5: Посилання (Links)

Посилання – це основа інтернету, яка дозволяє переходити між веб-сторінками.

5.1. Створення посилань:

Посилання створюються за допомогою тегу `<a>` (anchor). Атрибут `href` (hypertext reference) вказує URL (адресу) сторінки, на яку веде посилання.

`<p>Відвідайте Google.</p>`

`<p>Дізнайтесь більше на Wikipedia.</p>`

5.2. Відкриття посилання в новій вкладці:

Використовуйте атрибут `target="_blank"` для відкриття посилання в новій вкладці або вікні браузера.

`<p>Відкрити YouTube у новій вкладці.</p>`

5.3. Посилання на розділ тієї ж сторінки (Якорі):

Ви можете створювати посилання на певні розділи на тій же сторінці, використовуючи атрибут `id`.

* Створіть якорі: Додайте атрибут `id` до елемента, на який ви хочете посилатися.

`<h2 id="розділ1">Перший розділ</h2>`

`<p>Це зміст першого розділу...</p>`

`<h2 id="розділ2">Другий розділ</h2>`

`<p>Це зміст другого розділу...</p>`

* Створіть посилання на якорі: Використайте `href="#ID_якоря"`.

`<p>Перейти до другого розділу</p>`

5.4. Посилання на email-адресу:

Використовуйте `mailto:` у атрибуті `href` для створення посилання, яке відкриває клієнт електронної пошти.

<p>Напишіть нам: info@example.com</p>

5.5. Посилання на номер телефону:

Використовуйте tel: у атрибуті href для створення посилання, яке дозволяє телефонувати з мобільних пристроїв.

<p>Зателефонуйте нам: +38 (012) 345 67 89</p>

Урок 6: Зображення (Images)

Зображення роблять веб-сторінки більш привабливими та інформативними.

6.1. Додавання зображення:

Зображення додаються за допомогою порожнього тегу . Він має два обов'язкові атрибути:

- * src (source): Вказує шлях до файлу зображення.
- * alt (alternative text): Надає альтернативний текст, який відображається, якщо зображення не завантажилось, або використовується програмами читання з екрану для людей з вадами зору.

<!-- end list -->

6.2. Шляхи до файлів:

* Відносні шляхи: Використовуються, коли зображення знаходиться в тому ж каталозі або в підкаталозі щодо HTML-файлу.

* src="image.jpg": Зображення знаходиться в тому ж каталозі.

* src="images/image.jpg": Зображення знаходиться в підкаталозі images.

* src="../image.jpg": Зображення знаходиться в каталозі на один рівень вище.

* Абсолютні шляхи (URL): Вказують повну веб-адресу зображення.

* src="https://www.example.com/images/logo.png"

6.3. Розмір зображення (width та height):

Атрибути width та height дозволяють вказати ширину та висоту зображення в пікселях.

Важливо: Рекомендується встановлювати розміри зображення за допомогою CSS для кращого контролю та адаптивності, але для швидкого прототипування або коли розміри фіксовані, атрибути width та height є зручними.

6.4. Зображення як посилання:

Ви можете зробити зображення посиланням, помістивши тег всередину тегу <a>.

Практичне завдання 1:

Створіть новий HTML-файл (наприклад, my_first_page.html) і виконайте наступні кроки:

- * Додайте базову структуру HTML (DOCTYPE, html, head, body).
- * Встановіть заголовок сторінки на "Мій перший веб-сайт".
- * У <body> створіть заголовок першого рівня <h1> з текстом "Ласкаво просимо на мій веб-сайт!".

- * Створіть абзац <p> з коротким описом того, чому ви вивчаєте HTML.
- * Додайте горизонтальну лінію <hr>.
- * Створіть неупорядкований список з трьома елементами: "HTML", "CSS", "JavaScript".
- * Створіть посилання на сайт Google, що відкривається в новій вкладці. Текст посилання: "Відвідайте Google".
- * Додайте будь-яке зображення з Інтернету (або використайте placeholder: <https://via.placeholder.com/200>) і встановіть йому альтернативний текст.
- * Збережіть файл і відкрийте його в браузері, щоб переглянути результат.

Наступні кроки:

Ми пройшлися по основам HTML. У наступних уроках ми заглибимося в інші важливі елементи, такі як таблиці, форми, семантичні теги HTML5 та основи CSS для стилізації. Пам'ятайте: Найкращий спосіб вивчити HTML – це практикуватися. Експериментуйте з тегами, змінюйте атрибути, створюйте власні сторінки! Якщо у вас виникнуть питання, не соромтеся їх задавати!