

## Лекція 2.

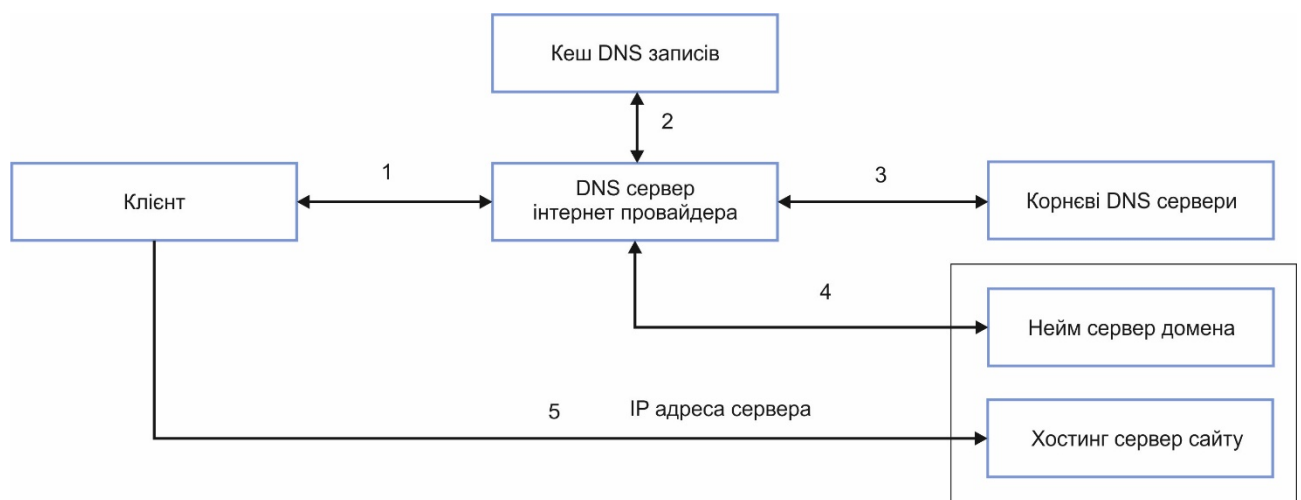
# Основи HTML. Основні принципи побудови та розробки структури веб сайту

### Видача сайту

Комп'ютери в мережі (в тому числі Інтернет) не мають імен, передача даних здійснюється з використанням IP-адрес.

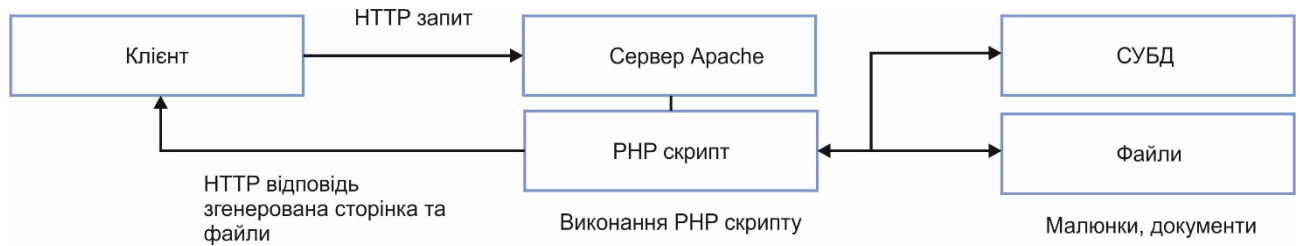
IP-адреса (Internet Protocol Address) - числова адреса в мережі інтернет, має вигляд 123.123.123.123 . IP-адреси важко запам'ятати людині, особливо коли ви відвідуєте десятки сайтів в день з різними IP-адресами.

Аналогічно тому, як ви зберігаєте телефонні номери, ви можете завести записну книжку або телефонний довідник для IP-адрес. Роль телефонного довідника в Інтернеті виконують DNS (Domain Name System), система доменних імен. Коли в інтернет-браузері ви вводите будь-якої домен, він за допомогою DNS перетворюється в IP-адресу, який використовується для доступу до сервера.



DNS-сервери прописуються парами, це робиться для кращої відмово стійкості: якщо один DNS-сервер вийде з ладу, другий залишиться працювати.

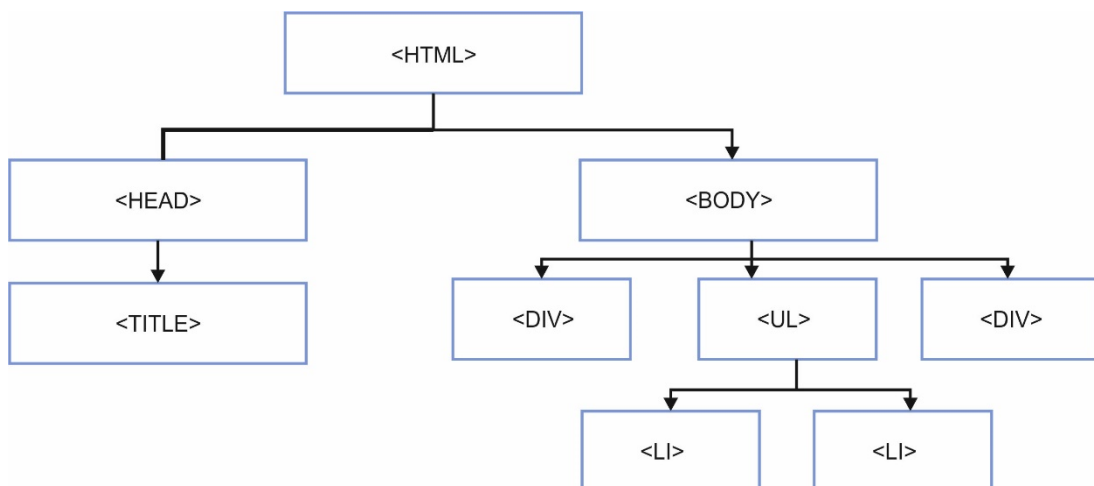
Надалі при запиті сторінки сайту по визначеному IP відбувається звернення до веб-сервера, який запускає інтерпретатор PHP для виконання скрипта. Далі програма виконується, взаємодіє з СУБД і віддає результат виконання клієнту. Крім того, веб-сервер віддає клієнту супутні файли - картинки, документи, css файли і іншу статичну інформацію.



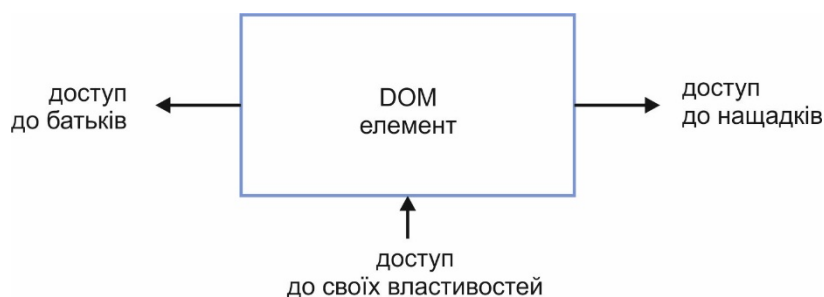
Передача даних клієнтові може уповільнюватися деякими факторами, серед яких: повільні канали передачі даних, продуктивність сервера зокрема PHP, великий час виконання самого скрипта, складні запити та повільний обмін з базою даних, низька продуктивність БД, передача великих не оптимізованих файлів.

### Основна структура сайту

Основним інструментом роботи і динамічних змін на сторінці є DOM (Document Object Model) - об'єктна модель, використовувана для XML / HTML-документів.



Згідно DOM-моделі, документ є ієрархією. Кожен HTML-тег утворює окремий елемент-вузол, кожен фрагмент тексту - текстовий елемент, і т.п. Кожен DOM-елемент є об'єктом і надає властивості для маніпуляції своїм вмістом, для доступу до батьків і нащадкам.



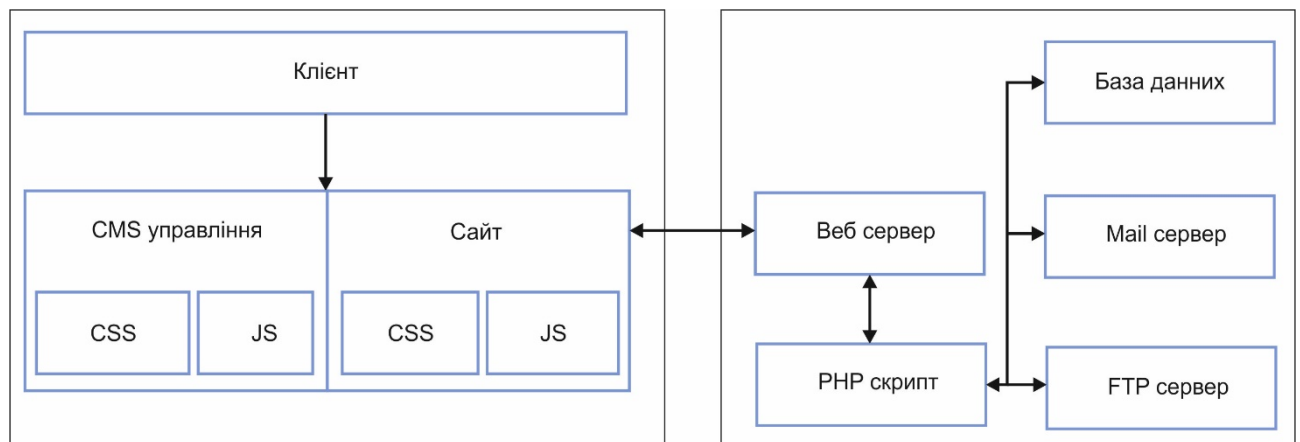
Простіше кажучи, DOM - це уявлення документа у вигляді дерева тегів. Це дерево утворюється за рахунок вкладеної структури тегів плюс текстові фрагменти сторінки, кожен з яких утворює окремий вузол.

### Основа програмування сайтів, етапи

Програмування сайту в основному починається з вибору мови програмування на якій буде він виконаний та структури бази даних для сайту. Виділяють основні етапи проектування сайту: проектування, прототипування, програмування, тестування, оптимізація, просування

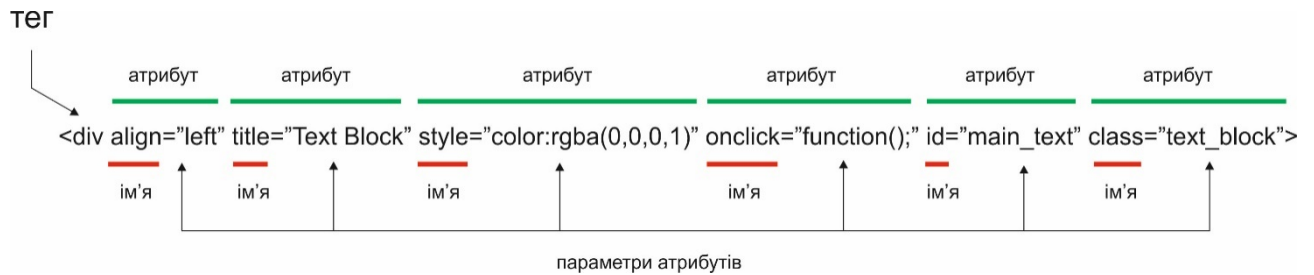
На кожному з цих етапів виконуються відповідні функції результатом яких буде кінцевим продуктом сайт. На етапі **Проектування** виконується аналіз наявних ресурсів, технічних потреб, визначаються із структурою та виконується постановка технічного завдання. На етапі **Прототипування** створюється зовнішній вигляд, інтерфейс взаємодії з користувачем. На етапі **Програмування** виконується реалізація проекту з розробки браузерної та серверної частини сайту. **Тестування** дозволяє визначити проблеми та помилки та усунути їх до здачі проекту. Щодо **оптимізації та просування** дані етапи являються маркетинговими етапами в розробці сайту і стосуються підвищення відвідуваності сайту в мережі інтернет.

Етапи програмування сайту можна розглядати як розробку частин що бачить користувач та серверних технологій які лише генерують результат та складають частини сайту для загальної видачі, як це показано на малюнку.



### Теги та атрибути

На прикладі тега div можна розглянути приклад вказівки атрибутів тегів. Атрибути тега поділяються на ті які відносять тільки до самого тега та ті які є універсальними. Останні свою назву отримати завдяки можливості використання практично у будь якому тегу. До них відносяться теги задання міток id та class, атрибути стилю та подій. Як видно на малюнку атрибути align та title являються основними для тегу div. Більш детальніше значення атрибутів буде розглянуто пізніше.



Сам елемент `<div>` є блоковим елементом і призначений для виділення фрагмента документа з метою зміни виду вмісту. Як правило, вид блоку управляється за допомогою стилів. Щоб не описувати кожен раз стиль всередині тега, можна виділити стиль в зовнішню таблицю стилів, а для тега додати атрибут `class` або `id` з ім'ям селектора.

Як і при використанні інших блочних елементів, вміст тега `<div>` завжди починається з нового рядка. Після нього також додається перенесення рядка.

Універсальні атрибути застосовуються практично до всіх тегам, тому виділені в окрему групу, щоб не повторювати їх для всіх тегів.

**accesskey** Дозволяє отримати доступ до елементу за допомогою заданого поєднання клавіш.

**class** Визначає ім'я класу, яке дозволяє зв'язати тег зі стильовим оформленням.

**contenteditable** Повідомляє, що елемент доступний для редагування користувачем.

**contextmenu** Встановлює контекстне меню для елемента.

**dir** Задає напрямок і відображення тексту - зліва направо або справа наліво.

**hidden** Приховує вміст елемента від перегляду.

**id** Вказує ім'я стильового ідентифікатора.

**lang** Браузер використовує значення параметра для правильного відображення деяких національних символів.

**spellcheck** Вказує браузеру перевіряти чи ні правопис і граматику в тексті.

**style** Застосовується для визначення стилю елемента за допомогою правил CSS.

**tabindex** Встановлює порядок отримання фокусу при переході між елементами за допомогою клавіші Tab.

**title** Описує вміст елемента у вигляді підказки.

**xml: lang** Цей атрибут за своєю дією схожий на `lang`, але застосовується тільки в XHTML-документах і вказує мову всього тексту або його фрагмента.

## Формат графіки

**GIF (Graphics Interchange Format)**- формат графічних файлів, широко застосовуваний при створенні сайтів. GIF використовує 8-бітовий колір і ефективно стискає суцільні кольорові області, при цьому зберігаючи деталі зображення

**JPEG (Joint Photographic Experts Group)**- популярний формат графічних файлів, широко застосовуються при створенні сайтів і зберігання зображень JPEG підтримує 24-бітовий колір і зберігає яскравість і відтінки кольорів у фотографіях незмінними.

PNG-8 (Portable Network Graphics)- формат по своїй дії аналогічний GIF. За запевненням розробників використовує покращений формат стиснення даних, але як показує практика, це не завжди так.

PNG-24- формат, аналогічний PNG-8, але використовує 24-бітову палітру кольору Подібно формату JPEG, зберігає яскравість і відтінки кольорів у фотографіях.

### Формат кольорів

Колір в стилях можна задавати різними способами: по шістнадцятиричним значенням, за назвою, в форматі RGB, RGBA, HSL, HSLA.

Можна визначити колір, використовуючи значення червоної, зеленої та синьої компоненти в десятковому численні. Кожна з трьох компонент кольору приймає значення від 0 до 255.

Також допустимо ставити колір в процентному відношенні, при цьому 100% буде відповідати числу 255. Спочатку вказується ключове слово RGB, а потім в дужках, через кому вказуються компоненти кольору, наприклад rgb (255 , 128, 128) або rgb (100%, 50%, 50%).

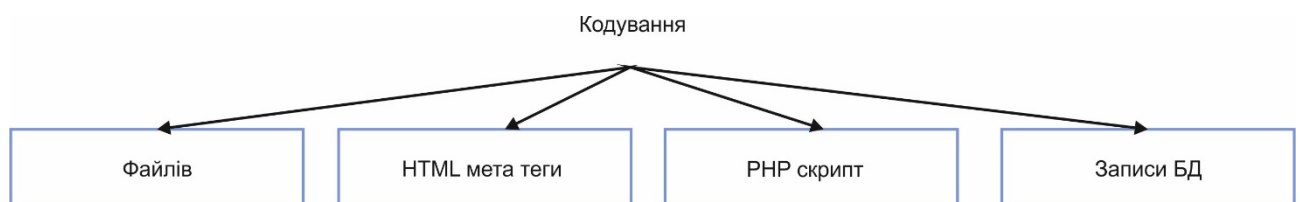
Формат RGBA схожий по синтаксису на RGB, але включає в себе альфа-канал, що задає прозорість елемента. Значення 0 відповідає повній прозорості, 1 - непрозорості, а проміжне значення на зразок 0.5 - напівпрозорості.

Назва формату HSL утворено від сполучення перших букв Hue (відтінок), Saturate (насиченість) і Lightness (світлин).

### Кодування

Windows-1251 – набір символів і кодування, що є стандартним 8-бітним кодуванням для всіх українських та російських версій Microsoft Windows. Користується досить великою популярністю. Windows-1251 вигідно відрізняється від інших 8-бітних кириличних кодувань (таких як CP866, KOI8-R і ISO 8859-5) наявністю практично всіх символів, що використовуються в українській типографії для звичайного тексту; вона також містить всі символи для близьких до української мови мовами: російської, білоруської, сербської та болгарського.

UTF-8 – в даний час поширене кодування, що реалізує уявлення Юнікоду, сумісний з 8-бітовим кодуванням тексту. Знайшло широке застосування в операційних системах і веб-просторі. Текст, що складається тільки з символів Юнікоду з номерами менше 128, при записі в UTF-8 перетворюється на звичайний текст ASCII. Решта символів Юнікоду зображуються послідовностями довжиною від 2 до 6 байт.



Основна відмінність кодувань – це використовуваний набір символів. В UTF-8 набагато більше символів можливо уявити, чим в Windows-1251. Кодування Windows-1251

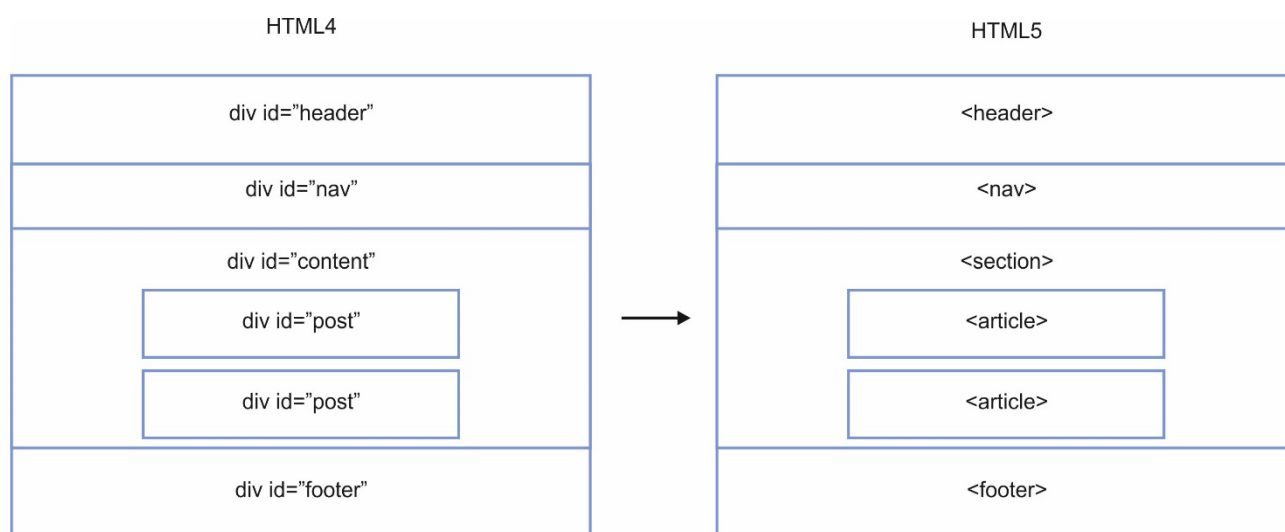
одноболтове, тобто представити в ній можна тільки 255 символів. Для кирилиці, втім, цього цілком достатньо, саме тому одно байтові кодування досі так масово застосовуються.

Символ в кодуванні UTF-8 може кодуватися аж 6 байтами (поки використовується тільки 4 і більше не планується). Для української мови, наприклад, символ займає 2 байта. Усі символи, які є в таблиці символів – підтримуються цим кодуванням. Наприклад, якщо вам потрібен знак копірайту (©), то вам не потрібно шукати особливий шрифт або ж зображати символ в графічному форматі.

### Різниця між HTML та HTML5 (HTML + JavaScript)

Перехід від HTML4 до HTML5 полягає лише в появі додаткових тегів для роботи з інтерфейсом. Прописування типу сторінки замінено лише на `<!DOCTYPE html>`

Додані теги полегшують програмування оскільки замінюють раніше використовувані бібліотеки JavaScript для роботи з даними. Були додані наступні нові елементи верстки `<article>` `<aside>` `<audio>` `<canvas>` `<command>` `<datalist>` `<details>` `<figcaption>` `<figure>` `<footer>` `<header>` `<hgroup>` `<keygen>` `<main>` `<mark>` `<menu>` `<meter>` `<nav>` `<output>` `<progress>` `<rp>` `<rt>` `<ruby>` `<section>` `<source>` `<summary>` `<time>` `<video>` `<wbr>`



Крім цього з'явилися нові типи полів для введення даних `input`. Хоча варто зазначити більшість з них поки що є недоступними у роботі з багатьма браузерами.

HTML4 type: `button` | `checkbox` | `file` | `hidden` | `image` | `password` | `radio` | `reset` | `submit` | `text`

HTML5 type: `date` | `datetime` | `datetime-`

`local` | `email` | `number` | `range` | `search` | `tel` | `time` | `url` | `month` | `week`

### Основні нові теги HTML5

Тег `<article>` задає зміст сайту на кшталт новини, статті, записи блогу, форуму або ін.

Тег `<footer>` задає «підвал» сайту або розділу, в ньому може розташовуватися ім'я автора, дата документа, контактна і правова інформація.

Тег <header> задає «шапку» сайту або розділу, в якій зазвичай розташовується заголовок. Задає розділ документа, може застосовуватися для блоку новин, контактної інформації, глав тексту, вкладок в діалоговому вікні та ін. Розділ зазвичай містить заголовок. Допускається вкладати один тег <section> всередину іншого.

<video> Додає, відтворює і управляє настройками відеоролика на веб-сторінці. Шлях до файлу задається через атрибут src або вкладений тег <source> . Список підтримуваних браузером аудіо та відеокодеків обмежений.

атрибути

autoplay Відео починає відтворюватися автоматично після завантаження сторінки.

controls Додає панель управління до відеоролика.

height Задає висоту області для відтворення відеоролика.

loop Повторює відтворення відео від початку до його завершення.

poster Вказує адресу картинки, яка буде відображатися, поки відео недоступне або не чути.

preload Використовується для завантаження відео разом із завантаженням веб-сторінки.

src Вказує шлях до воспроизводимому відеоролику.

width Задає ширину області для відтворення відеоролика

#### **Джерела:**

<http://htmlbook.ru> - бібліотека, довідник по тегах та атрибутам HTML CSS

<http://javascript.ru> - довідник по мові програмування JavaScript