

Лабораторна робота №1.

Мова розмітки HTML та мова описання представлень CSS

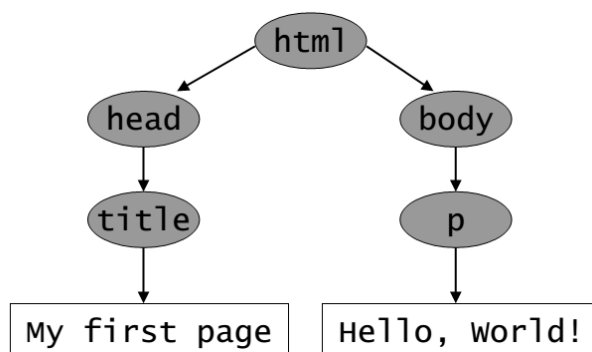
Завдання:

1. Використовуючи мову розмітки HTML, створити шаблон WEB-сайту. Шаблон повинен складатися, як мінімум з трьох сторінок: 1) Титульна сторінка, 2) Звичайна сторінка та 3) Сторінка зворотного зв'язку.
2. Використовуючи можливості CSS, змінити шаблон WEB-сайту, створений в п.1 у відповідності зі стандартом HTML5 (XHTML).
Вимоги до шаблону:
 - a. Шаблон повинен проходити валідацію на ресурсі validator.w3.org
 - b. CSS файл не повинен містити стилів, коментарів, "хаків", які не використовуються в шаблоні
 - c. Верстка шаблону повинна бути семантично вірною

Теоретичні відомості

HTML – це не мова програмування, а «мова розмітки» (HyperText Markup Language). Вона визначає зміст і структуру сторінки, але не зовнішній вигляд. Елементи мови мають структуру дерева (вкладені елементи), пропуски або ігноруються, або замінюються одним пропуском. Вузли дерева являють собою або текст (зміст), або "Структурні елементи", маркуються "тегами" і мають "атрибути".

```
<html>
  <head>
    <title>My first page</title>
  </head>
  <body>
    <p>Hello, world!</p>
  </body>
</html>
```



Структура елементів мови HTML

Елемент, маркований тегом, має таку структуру:

```
<element attr1="value1" attr2="value2"...>  
    внутрішній вміст  
</element>
```

Наприклад, `Next page`

Якщо тег не має *внутрішнього вмісту*, то він може мати спрощену структуру:

```
<element attr1="value1" attr2="value2"... />  
  
  
  
<hr/>
```

Структура сторінки

Правильна сторінка має дві частини – заголовок (інформація про сторінку) та тіло (вміст сторінки)

```
<html>  
  <head>  
    інформація про сторінку  
  </head>  
  <body>  
    внутрішній вміст  
  </body>  
</html>
```

Браузери відображують сторінку, що прийшла до них по запиті. Сторінка може бути написана як на старому HTML, так і на нових XML та XHTML. Ми будемо використовувати більш сучасний XHTML.

Основна відмінність у внутрішньому вмісті документів HTML та XHTML: якщо браузер бачить помилку в HTML-документі, то він зобов'язаний спробувати зрозуміти, що мав на увазі автор документа. Якщо ж помилка виявлена у XHTML документі, то браузер просто повідомляє про неї.

Крім того, форматування елементів XHTML-документа повинно бути оформлене за допомогою CSS-стилів. Існують теги, такі як font, та атрибути, такі як bgcolor і align, що не підтримуються в XHTML.

Вимоги XHTML, які дозволено порушувати у старому HTML

1. Всі елементи повинні бути закриті.
2. Всі обов'язкові атрибути повинні бути присутніми.
3. Вкладеність елементів повинна бути правильною.
4. В тілі документа текст не може бути вкладеним безпосередньо.
5. Атрибути повинні братися в лапки.
6. «Блочні» елементи не можуть бути вкладені в «рядкові».
7. Спецсимволи завжди повинні бути представлені мнемонічними посиланнями.
8. Теги та атрибути записуються лише рядковими літерами.

Блочні та рядкові елементи

Блочні елементи містять фрагменти тексту, які завжди відображаються в окремих блоках. Браузери візуально відокремлюють блочні елементи один від одного.

Приклади: <div>, <p>, , <tr>

Рядкові елементи можуть розташовуватись один за одним в межах одного рядка

Приклади: , <a>, ,

Коментарі вставляються в HTML текст так же, як і звичайні елементи:

```
<!-- This is my coolest page -->
```

Деякі елементи HTML та їхні атрибути

Абзац (параграф) – блочний елемент

Так записується в редакторі:

```
<p>Видимий тут вміст відображується в браузері у вигляді одного абзацу.
```

Пропуски, що повторюються та переходи з рядка на рядок ігноруються.</p>

Так буде відображено:

Видимий тут вміст відображується в браузері у вигляді одного абзацу. Пропуски, що повторюються та переходи з рядка на рядок ігноруються.

Якщо необхідно зробити перехід на наступний рядок всередині абзацу, то це досягається за допомогою елемента

Так записується в редакторі:

```
<p>Це перший рядок параграфа,<br/>а це вже другий</p>
```

```
<p>А це вже наступний абзац.</p>
```

Так буде відображено:

Це перший рядок параграфа,

а це вже другий

А це вже наступний абзац.

Заголовки – блочні елементи

Так записується в редакторі:

```
<h1>Національний Університет Кораблебудування</h1>
```

```
<h2>Кафедра ІУСТ</h2>
```

```
<h3>Розклад занять</h3>
```

Так буде відображено:

Національний Університет Кораблебудування

Кафедра ІУСТ

Розклад занять

Всього може бути до шести рівнів заголовків (от <h1> до <h6>)

Горизонтальна риска (роздільник) – блочний елемент

Так записується в редакторі:

```
<p>Національний Університет Кораблебудування </p><hr/>
```

```
<p>Кафедра ІУСТ</p>
```

Так буде відображено:

Національний Університет Кораблебудування

Кафедра ІУСТ

Гіперпосилання – рядковий елемент

Важливу роль у web-сторінках відіграють гіперпосилання. Розрізняють посилання на інші сторінки та на позиції у поточному документі.

Так записується в редакторі:

```
<p>Користуйтеся пошуком
```

```
<a href = "http://www.google.com" > Google</a> –  
найпоширенішою пошуковою системою у світі!</p>
```

Так буде відображено:

Користуйтеся пошуком Google – найпоширенішою пошуковою системою у світі!

Будьте акуратними із вкладеністю елементів!

```
<p><a href="Page2.html">Це в першому абзаці</p>
```

```
<p>А це вже у другому!</a></p>
```

Тут дві помилки: блочний елемент всередині рядкового і неправильна вкладеність елементів. Проте, HTML-браузер може правильно відобразити ці елементи!

Рисунки та фотографії

Вставка зображень – рядковий елемент.

```
<p></p>
```

Додаткові атрибути

```

```

```

```

Зображення може служити посиланням так само, як і текст:

```
<a href="servers/server.html">

</a>
```

Списки

Список (нумерований або нелінійний) – блочний елемент, що містить всередині себе блочні елементи – члени списку

Так записується в редакторі:

```
<ul>
<li>Перший рядок списку</li>
<li>Другий рядок списку</li>
<li>Третій рядок списку</li>
</ul>
```

Так буде відображено:

- Перший рядок списку
- Другий рядок списку
- Третій рядок списку

Списки можуть бути нумерованими `` та ненумерованими ``. Також можливе вкладення списків один в один.

Виділення фрагментів тексту

Виділення відбувається за допомогою тегів ``, ``, `<kbd>`, `<code>`

Так записується в редакторі:

```
<p>Вивчаючи <code>HTML</code> слід звернути  
<strong>особливу увагу</strong> на різницю між  
<em>нумерованими</em> списками <code>&lt;ol&gt;</code> і  
<em>ненумерованими</em> списками  
<code>&lt;ul&gt;</code>.</p>
```

Так буде відображено:

Вивчаючи HTML слід звернути особливу увагу на різницю між нумерованими списками `` і ненумерованими списками ``.

Елементи заголовка HTML

```
<title>Це моя сторінка</title>
```

Відображується у заголовку сторінки в браузері.

```
<meta name="description" content="Важлива сторінка"/>
```

Описує вміст сторінки.

```
<meta name="keywords" content="мир, труд, май"/>
```

Описує ключові слова (часто використовується пошуковими машинами).

```
<meta name="author" content="Yevhen Berkunskyi"/>
```

Інформація про автора.

```
<meta http-equiv="Content-Language" content="ru">
```

Описує формат і кодування сторінки

```
<meta http-equiv="Content-Type" content="text/html;  
charset=windows -1251">
```

Визначає основну мову, якою написано сторінку.

```
<meta http-equiv="refresh"  
content="5;http://www.google.com/">
```

Описує частоту перезавантаження сторінки (в секундах) і, можливо, робить “redirection” іншу сторінку.

```
<link type="image/jpeg" rel="shortcut icon"  
href="favicon.jpg">
```

Описує іконку, асоційовану зі сторінкою (застарілий варіант: помістити в корінний каталог сайту файл з іменем favicon.ico).

Що таке CSS?

- Це мова описань зовнішнього представлення для вмісту, який описаний в HTML-сторінках;
- Визначає зовнішній вигляд тексту – шрифти, розміри, колір;
- Визначає розташування елементів відносно один одного;
- Описання зовнішнього представлення може бути фізично відокремлене від описання вмісту.

В колишньому стандарті HTML допускалось використання описань зовнішнього представлення за допомогою атрибутів та окремих елементів.

```
<p><font face="Arial">Ласкаво просимо в НУК  
ім.адм.Макарова. Ви отримаєте <b>найповнішу, <i>кращу,  
<u>НАЙКРАЩУ</u></i></b> освіту в Миколаєві з <font  
size="+1" color="red">ПОЧАТКОВИМ </font> багажем  
знань!</font></p>
```

Ласкаво просимо в НУК ім.адм.Макарова. Ви отримаєте **найповнішу, кращу, НАЙКРАЩУ** освіту в Миколаєві з **ПОЧАТКОВИМ** багажем знань!

Проте такий спосіб описання представлення НЕ ПІДТРИМУЄТЬСЯ в «строговому» XHTML!

НЕПРАВИЛЬНО:

```
<p>  
  <font face="Arial">В НУК ви отримаєте  
    <b>найповнішу,  
      <i>кращу,  
        <u>НАЙКРАЩУ</u>  
      </i>  
    </b>освіту в Миколаєві з  
    <font size="+1" color="red">початковим</font>  
    багажем знань!  
  </font>  
</p>
```

ПРАВИЛЬНО:

```
<p style="font-family: Arial;">В НУК ви отримуєте  
  <span style="font-weight: bold;">найповнішу,  
    <span style="font-style: italic;">кращу,  
      <span style="text-decoration:  
underline;">НАЙКРАЩУ</span>  
    </span>  
  </span>освіту в Миколаєві з  
  <span style="font-size: larger; color:  
red;">початковим</span>  
  багажом знань!  
</p>
```

Прив'язування сторінки стилів до документа

Стилі можна вписати прямо в теги HTML-документа. Але частіше їх виносять або на початок документа, або в окремий файл.

Нехай файл *mystyles.css* містить такий текст

```
p { color: white; background-color: black; }  
h1 { font-size: large; font-weight: bold; }  
h2 { font-weight: 500; color: blue; }
```

Тоді сторінка HTML може містити посилання на нього у такому вигляді:

```
<html>  
  <head>  
    <link rel="stylesheet" type="text/css" href=mystyles.css"/>  
  </head>  
  <body>  
    вміст сторінки HTML-документа  
  </body>  
</html>
```

Деякі атрибути та варіанти значень

Атрибути шрифту (font) і тексту (text).

```
font-family: "lucida console", "courier new", sans-serif;  
font-size: small;  
font-size: larger;  
font-size: 10px;  
font-size: 80%;  
font-weight: bold;  
font-weight: 400;  
font-style: italic;
```

```
font: sans-serif bold x-large;
```

```
text-align: center;  
text-align: right;  
text-transform: uppercase;  
text-indent: 2cm;  
text-decoration: underline;  
text-decoration: blink;
```

Атрибути кольору

```
color: red;  
color: rgb(25, 30, 120);  
color: #c0c0c0;
```

```
background-color: yellow;
```

Взаємодія стилів

Якщо стиль визначений таким чином:

```
body { font-family: sans-serif; background-color: yellow; }  
p { color: red; background-color: aqua; }  
a { text-decoration: overline underline; }  
h2 { font-weight: bold; text-align: center; }
```

А сторінка, що підключає його, так:

```
<body>
  <h2>Це заголовок</h2>
  <p>А це абзац із
    <a href="myref.html">посиланням</a>
    всередині
  </p>
</body>
```

Тоді сторінка буде виглядати так

Це заголовок

А це абзац із посиланням всередині

Використання тегів div та span

```
div.style1 { font-family: sans-serif; }
div.style2 { font-family: Times; color: blue; }
.bold { font-weight: bold; }

<body>
  <div class="style1">
    <h2>Це заголовок класа style1</h2>
    <p>Це абзац класу style1</p>
  </div>
  <div class="style2">
    <h2>Це заголовок класу style2</h2>
    <p>Це абзац <span class="bold">класу</span> style2</p>
  </div>
</body>
```

Це заголовок класу style1

Це абзац класу style1

Це заголовок класу style2

Це абзац класу style2

Каскадування стилів

Чим визначається стиль конкретного елемента?

1. Стилем, визначеним браузером «за замовчуванням»
2. Стилем, вказаним в окремій CSS-сторінці, що прив'язана до HTML-документу елементом `<link>` в заголовку:

```
<link rel="stylesheet" type="text/css" href="styles.css"/>
```

3. Стилем, вказаним в заголовку HTML-документа за допомогою елемента

```
<style>:
```

```
<style> body { background-color: yellow; } </style>
```

4. Стилем, вказаним в самому елементі за допомогою атрибута `style`:

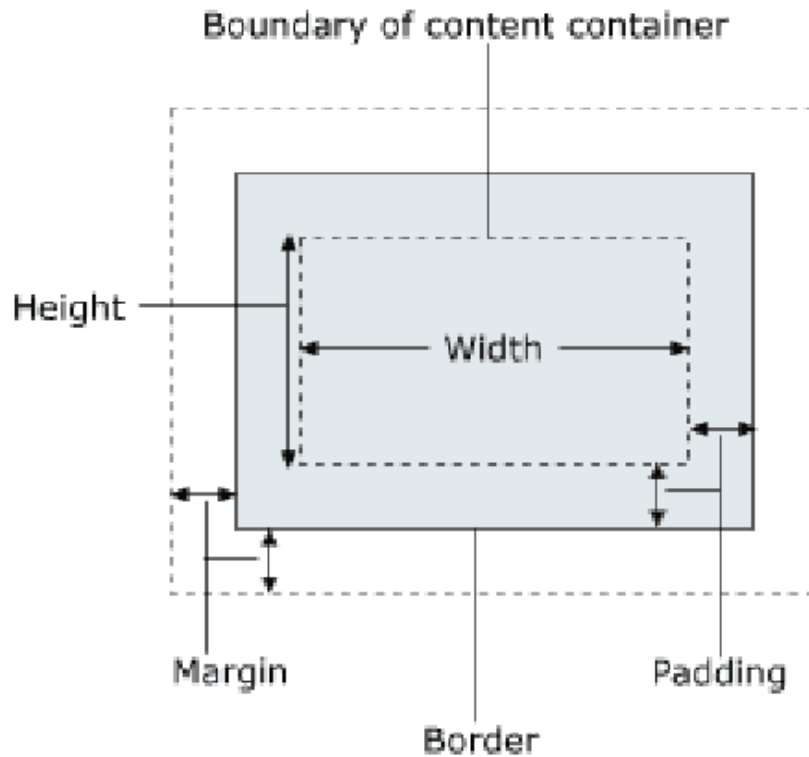
```
<p style="margin: 0.5in;">Відступ в половину дюйма від краю</p>
```

Чим «ближче» визначення стилю до елемента, тим пріоритетнішим він буде у випадку конфлікту параметрів стилю.

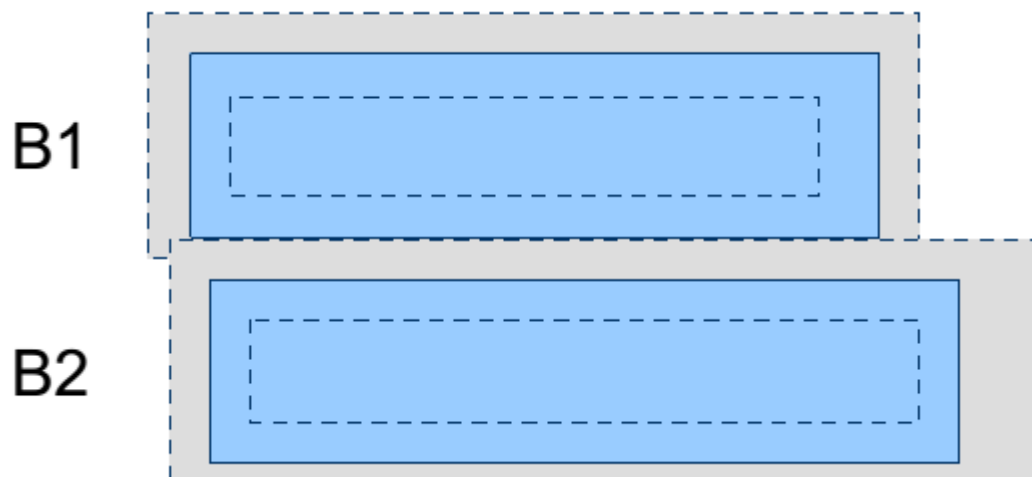
Розміщення фрагментів за допомогою CSS

Блочні елементи мають:

- внутрішній вміст вказаної ширини та висоти (`width, height`);
- прошарок (`padding`);
- межу (`border`);
- поля (`margin`).



За замовчуванням блоки розташовуються вертикально, при цьому поля сусідніх блоків перекриваються (спільне поле двох сусідніх блоків дорівнює по висоті максимальному з двох полів елементів).



Приклад розміщення блоків

```
<style type="text/css">
p { font-family: sans-serif;
    font-size: 16pt;
    border: 2px solid red; }
p.class1 { width: 400px;
```

```

        background-color: yellow;
        padding: 0.5cm;
        margin: 0.5cm; }
p.class2 { width: 500px;
        background-color: green;
        padding: 0.3cm;
        margin: 1cm; }
</style>

<body>
<p class="class1">Первый параграф</p>
<p class="class1">Второй параграф</p>
<p class="class2">Третий параграф</p>
<p class="class2">Четвертый параграф</p>
<p class="class1">Пятый параграф</p>
</body>

```

Розміщення блоків на сторінці

```
{ width: 70%; margin-left: auto; margin-right: auto; }
```

Заданий таким чином, як показано вище, стиль дозволяє розмістити блок по центру сторінки (відповідно ліворуч чи праворуч, якщо задано одне поле).

Розміщення тексту (або інших рядкових елементів) всередині блока задається інакше:

```

{ text-align: left; } (або center, або right)
<h1 style="width: 50%; margin-left: auto;
        background-color: yellow;">Заголовок</h1>
<h2 style="width: 70%; margin-left: auto;
        margin-right: auto; background-color: yellow;
        text-align: right;">ще один заголовок</h2>

```

Таким чином, комбінуючи можливості HTML та CSS можна створити дизайн сторінок на будь-який смак, проте ці технології використовуються винятково на стороні клієнта, якщо ж треба організувати взаємодію з сервером, треба розробляти відповідне серверне програмне забезпечення.