

Тема 2 HTML та CSS

Як і всі технології, HTML постійно розвивається. Поточна версія є HTML5. Вона запроваджує чимало нових функцій, підтримуючи відтворення відео та аудіо, а також 2D та 3D графіки. Використання HTML5 дозволяє створювати більш легкі вебсторінки, які завантажуються швидше та менше навантажують вебсервери. Багато функцій HTML5 були розроблені з можливістю їх запуску на малопотужних пристроях, як-от смартфони та планшети.

Додаємо відео

За допомогою HTML5 додавати відео на ваші вебсторінки легко. Подивимося, як це зробити.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>Моя перша вебсторінка</title>
5 </head>
6 <body>
7
8 <h1>Це особиста сторінка Оленки</h1>
9 <p>Ласкаво просимо на мою особисту вебсторінку.</p>
10 <p>Ось коло моїх інтересів:</p>
11 <ul>
12 <li>Фото</li>
13 <li>Арт</li>
14 <li>Комп'ютери</li>
15 </ul>
16
17 <h3>Моє відео</h3>
18
19 <!-- Локальне відео -->
20 <video width="640" height="360" controls>
21 <source src="design T-shot.mp4" type="video/mp4">
22 </video>
23
24 <!-- Відео з YouTube -->
25
26 <iframe
27 width="560"
28 height="315"
29 src="https://www.youtube.com/embed/KzKATbGPzAU?si=46bCO6yI1wtGST7A" title="YouTube video player" frameborder="0"
30 allowfullscreen>
31 </iframe>
32
33
34 </video>
35 </body>
36 </html>
37 </body>
38 </html>
```

Вказує ширину та висоту вікна відео

Якщо відеофайл розташований в тій самій теці, що й вебсторінка, просто напишіть ім'я файлу. Якщо відеофайл лежить в іншій теці – вкажіть шлях до цього файлу. Наприклад, `<source src="F:\LEGO\Дизайн.mp4" type="video/mp4">`.

Якщо ваш відеофайл розміщений на сайті в інтернеті, відтворити його можна за допомогою такого коду.

Атрибути тега `<video>`:

- autoplay** – відео починає відтворюватися автоматично;
- loop** – відео відтворюється циклічно;
- muted** – відео відтворюється без звуку.



Каскадні таблиці стилів (CSS) – це мова, яка дозволяє змінювати зовнішній вигляд вебсторінок. CSS співпрацює з HTML. За допомогою CSS можна змінювати колір, розмір, розташування візуальних елементів, додавати відео та анімацію, перетворюючи звичайний текстовий контент у привабливі та функціональні вебсторінки.

Теги `<style type=«text/css»> ... </style>` оточують елементи CSS, які застосовуються до вебсторінки, усередині тегів `<head> ... </head>` HTML-коду. Загальний синтаксис правил CSS наступний:

Перша частина коду – **селектор**. Він є структурою, яку використовують для визначення вигляду певного елемента (або набору елементів) у браузері.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>Моя перша вебсторінка</title>
5 <style type='text/css'>
6   body {
7     background-color: #6FF;
8   }
9 </style>
10 </head>
11 <body>
12
13 <h1>Це особиста сторінка Оленки</h1>
```

Властивості CSS-коду визначають конкретну характеристику елемента, наприклад, властивість `background-color`.

Значення властивості задає параметр вигляду. Наприклад, колір фону.

Звичайно, правила CSS можуть містити багато пар властивостей та їх значень. Кожна з них виконує певне завдання в структурі, щоб надати сторінці потрібний вам вигляд.



Додамо новий CSS-код. Застосуємо до тегу `<p>`, що визначає зовнішній вигляд усіх абзаців на нашій вебсторінці.

```
5 <style type='text/css'>
6   body {
7     background-color: #6FF;
8   }
9   p {
10    background-color: #3CF;
11    padding: 5px;
12    color: #FFFFFF;
13    width: 300px;
14  }
15 </style>
```

Це особиста сторінка Оленки

Ласкаво просимо на мою особисту вебсторінку.

Ось коло моїх інтересів:

- Фото

Кожен елемент HTML міститься в невидимому прямокутнику. Змінивши **background-color** абзацу, цей контейнер забарвлюється, й прямокутник можна побачити. Властивість **color** змінює колір тексту елемента. **Padding** – це відстань від вмісту елемента до внутрішніх стінок прямокутника.

Вказавши значення **width**, встановлюєте точну ширину нашого прямокутника.

```
5 <style type='text/css'>
6   body {
7     back
8   }
9   p {
10    bac
11    pad
12    col
13    wid
14  }
15 </style>
16 </head>
17 <body>
18   background-color: #6FF;
```

Коли починаєте вводити новий CSS-код в редакторі коду, відображається список підказок, що містить назви всіх доступних властивостей, які можна змінити.

Стилі CSS краще розміщувати на початку коду, всередині тегу `<style>`. У такий спосіб браузер відформатує сторінку під час її відкриття. В іншому випадку він спочатку відкриває вебсторінку, а потім застосовує будь-які стилі.



Коли селектором коду CSS є стандартний елемент HTML, наприклад, `<p>` або `<h1>`, це означає, що правило застосовується до всіх екземплярів цього елемента по всій вебсторінці. Однак, якщо ви хочете змінити вигляд окремого елемента чи набору елементів, можна використовувати **id** або **class** як селектор.

У першому правилі створюємо **id** з назвою **video** – це унікальний елемент, який може бути застосований лише до одного елемента на вебсторінці. Правила CSS для ідентифікаторів починаються зі знака хешування `#`.

У другому правилі створюємо клас з назвою **txtCenter** – це тип селектора, який можна застосувати до будь-якої кількості елементів. Правила CSS для класів починаються з крапки `.`.

```
15 #video {
16   width: 400px;
17 }
18 .txtCenter {
19   text-align: center;
20 }
```

Властивостей CSS багато. Більшість назв властивостей можна зрозуміти за їх назвою. Експериментуйте з різними властивостями елементів.

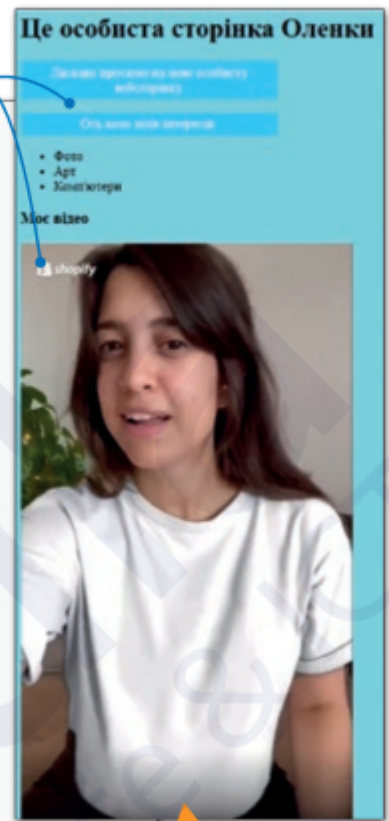
ПІДКАЗКА

Якщо ви зіткнулися з властивістю CSS, функцію якої не можете зрозуміти, або якщо хочете дослідити та дізнатися більше про інші властивості CSS, скористайтеся улюбленою пошуковою системою та надрукуйте **CSS властивості**.

Правила до будь-яких елементів на вебсторінці застосувати дуже легко.

У наведеному прикладі встановили стиль для елемента **video** ідентифікатор **video**, а для другого абзацу – клас **txtCenter**.

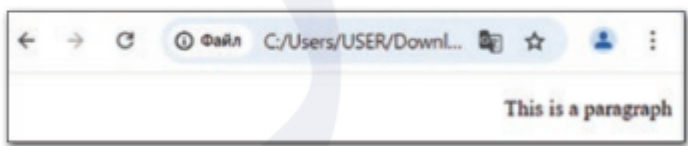
```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>Моя перша вебсторінка</title>
5 <style type='text/css'>
6   body {
7     background-color: #6FF;
8   }
9   p {
10    background-color: #3CF;
11    padding: 5px;
12    color: #FFFFFF;
13    width: 300px;
14  }
15  #video {
16    width: 400px;
17  }
18  .txtCenter {
19    text-align: center;
20  }
21 </style>
22 </head>
23
24 <body>
25 <h1>Це особиста сторінка Оленки</h1>
26 <p class="txtCenter">Ласкаво просимо на мою особисту вебсторінку.</p>
27 <p class="txtCenter">Ось коло моїх інтересів:</p>
28 <ul>
29   <li>Фото</li>
30   <li>Арт</li>
31   <li>Комп'ютери</li>
32 </ul>
33
34 <h3>Моє відео</h3>
35
36 <!-- Локальне відео -->
37 <video id="video" src="design T-shot.mp4"></video>
38 </body>
39 </html>
```



Визначимо колір шрифту – #000 або чорний, текст вирівнюємо по правому краю.

Каскадний ефект CSS

Правила CSS можна успадковувати та перевизначати, створюючи ієрархію та взаємозв'язок стилів.



Тут ми визначаємо два правила для елементів **body**: колір шрифту – #000, або чорний, а текст має бути вирівняний по правому краю. Однак у правилі CSS для елемента параграфа перевизначаємо властивість **text-align** як **right**.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>Моя перша вебсторінка</title>
5 <style>
6   body {
7     color: #000;
8     text-align: center;
9   }
10
11  p {
12    text-align: right;
13  }
14 </style>
15 </head>
16 <body>
17 <p>This is a paragraph</p>
18 </body>
19 </html>
```

Отриманий абзац матиме чорний колір шрифту й буде вирівняний по правому краю. Оскільки елемент `<p>` міститься в елементі `<body>`, він успадковує всі властивості, які не визначені явно в правилі `<p>`, від свого батьківського елемента `<body>`. Так, колір шрифту абзацу успадковується від правила `<body>`. Однак, коли ця ж властивість перевизначається, застосовується правило найбільш специфічного селектора, і саме тому абзац вирівнюється по правому краю.



Зовнішні таблиці стилів

У попередньому прикладі була використана внутрішня таблиця стилів – код CSS розташований в самому файлі `index.html`. Це підходить для невеликих проєктів або в ситуаціях, коли стилі, які ви визначаєте, будуть використовуватися лише на одній сторінці. Буває багато випадків, коли стилі застосовують до багатьох сторінок. У такому випадку копіювання та вставка коду CSS на кожен сторінку – клопітка справа. Окрім того, що сторінки будуть захищені однаковим кодом CSS, доведеться редагувати кожен з цих сторінок, якщо ви захочете змінити стиль.

Створіть окремий файл CSS – зовнішню таблицю стилів, з усіма вашими правилами, а потім зв'яжіть його з вебсторінкою, до якої треба застосувати ці правила. Введіть правила CSS у файл з розширенням `.css`, наприклад `style.css`, збережіть його в тій самій теці, що й ваш файл `index.html`. Не додавайте до файлу жодних тегів, наприклад `<style>`, просто введіть правила. Потім видаліть усі внутрішні визначення таблиці стилів з файлу `index.html` та розмістіть наступний рядок між тегами `<head> ... </head>` вашої сторінки.

```
<link href='style.css' rel='stylesheet' type='text/css'>
```

Ваш код буде виглядати так.

```
1 <!DOCTYPE html>
2 <html lang='uk'>
3 <head>
4   <meta charset="UTF-8">
5   <title>Моя перша вебсторінка</title>
6   <meta name='keywords' content='Персональна сторінка Оленки, Оленка Учениця'>
7   <meta name='description' content='Персональна сторінка Оленки Учениці'>
8   <link href='style.css' rel='stylesheet' type='text/css'>
9 </head>
10
11 <body>
12 <h1>Це особиста сторінка Оленки</h1>
```

Тег `<meta>` повідомляє браузеру користувача, який набір символів і яке кодування необхідно встановити на вебсторінці. Атрибут `charset` задає кодування символів для HTML-документа:

- UTF-8 – кодування символів для Unicode;
- ISO-8859-1 – кодування символів для латинського алфавіту.

Практичне завдання

Завантажте файл-заготовку зі сторінки інтернет-підтримки та виконайте запропоновані завдання.

