

Означення й застосування стилів, зв'язування аркушів стилів з гіпертекстовими документами.

!!! Увага! Під час роботи з комп'ютером дотримуйтеся правил безпеки та санітарно-гігієнічних норм.!!!

Для роботи можна скористатися будь-яким онлайн редактором для написання html сторінок (наприклад <https://liveweave.com/>), встановленою програмою (блокнот, Notepad++ і т.д.) або записати в зошит.

Виконайте будь-яку одну вправу

Фото виконаного завдання надішліть вчителю інформатики на пошту

Кравчук Тетяна Володимирівна: admin@ck19.ukr.education

Дорда Вікторія Олександрівна: vika@ck19.ukr.education

Використання таблиць каскадних стилів

Ви вже розглянули можливості форматування вмісту веб-сторінок за допомогою засобів мови HTML та візуального редактора веб-сайтів. Проте є й інші засоби оформлення текстів і створення зовнішніх ефектів, які можуть прикрасити веб-сторінку. Одним з них є таблиці стилів. Загальний принцип використання стилів на веб-сторінках той самий, що і для документів, створених у середовищі текстового процесора: користувач визначає набори правил форматування, які потім застосовуються до елементів документа. Проте у веб-дизайні способи застосування стилів різноманітніші.

Поняття про таблиці каскадних стилів

Стиль — це набір правил оформлення та форматування, який можна застосувати до різних елементів веб-документа. У разі використання стандартної мови HTML для надання кільком елементам певних властивостей (наприклад, призначення стилю шрифту) доводиться задавати ці властивості для кожного елемента. Використання таблиць стилів дає змогу уникнути цього, оптимізувавши розробку веб-сайтів. Стили дають змогу позиціювати елементи сторінки (наприклад, тексти і графіку), задаючи координати. Крім того, таблиці стилів часто використовують під час створення так званих динамічних сторінок.

Таблиці каскадних стилів (CSS, Cascading Style Sheets) містять параметри форматування частини або всього тексту веб-сторінки. Якщо таку таблицю підключено, то у тегах можна просто вказувати посилання на неї, а не задавати велику кількість атрибутів. У цьому випадку стилі названі каскадними тому, що в одному документі їх можна описати кілька, і браузер використовуватиме їх каскадом відповідно до їхнього пріоритету.

Таблиці каскадних стилів — це передусім набори параметрів, що змінюють властивості тегів HTML. Такі набори називають ще визначеннями тегів. Наприклад:

```
P {font-size: 40pt; color: green; font-family: "Comic Sans MS" }
```

Тут задано параметри для тегу абзацу <P>, які встановлюють розмір шрифту 40 пунктів, колір шрифту — зелений, гарнітуру — Comic Sans MS. У документі достатньо ввести теги <P>...</P> із текстом абзацу, щоб автоматично надати йому зазначеного оформлення.

Таблиці каскадних стилів дають змогу отримати результати, яких неможливо досягнути звичайними засобами HTML. У наведеному прикладі встановлено розмір шрифту 40 пунктів, хоча в мові HTML за допомогою атрибута SIZE можна задати для шрифту максимальний розмір 7, що відповідає 36 пунктам.

Крім того, таблиці каскадних стилів дають змогу визначити єдиний стиль оформлення для різних сторінок документа і швидко модифікувати його зміною відповідного параметра у таблиці стилів.

Параметрів форматування, які можна задавати за допомогою стилів, досить багато. Це, зокрема:

ПАРАМЕТРИ ФОРМАТУВАННЯ

background — колір тла;
font-family — стиль шрифту (гарнітура);
font-size — розмір шрифту;
font-weight — жирність шрифту;
color — колір шрифту;
text-decoration — оздоблення тексту;
text-align — вирівнювання тексту;
margin-top — відступ від верхнього рядка абзацу;
line-height — міжрядкова відстань.

Застосування каскадних стилів у HTML-документах

є три способи зв'язку каскадних стилів із HTML-документом:

- підключення зовнішньої таблиці стилів;
- розташування опису стилів у розділі HEAD документа;
- задавання властивостей стилів безпосередньо в тегах абзаців чи заголовків.

Підключення зовнішньої таблиці стилів

Зовнішня таблиця стилів (External Style Sheet) — це текстовий файл із розширенням .css. Його підключають до HTML-документа за допомогою тегу <LINK>, який записують у розділі <HEAD>, наприклад:

```
<LINK REL = "stylesheet" TYPE = "text/css"  
      HREF = "mystyle.css">
```

Атрибути REL та TYPE вказують браузеру на те, що сторінка використовує таблиці каскадних стилів. Атрибут HREF задає адресу файлу (mystyle.css).

Вправа 5.1

Оформимо веб-сторінку за допомогою зовнішньої таблиці стилів.

1. Відкрийте текстовий редактор Блокнот і введіть таке визначення тегу <P>:

 mystyle.css - Блокнот

Файл Редагування Формат Вигляд Довідка

```
p{
  color: red;
  font-size: 40px;
  font-family: 'Comic Sans MS';
}
```

2. Збережіть файл з іменем **mystyle.css**

3. Створіть у Блокноті файл такого змісту:

 przyklad1.html - Блокнот

Файл Редагування Формат Вигляд Довідка

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Приклад 1</title>
  <link rel="stylesheet" href="mystyle.css">
</head>
<body>
  <p> В цьому прикладі продемонстровано можливості зміни властивостей
  елементів веб-сторінки з допомогою зовнішньої таблиці стилів!</p>

  <p>Цей текст оформлено відповідно до таблиці стилів, яка міститься у файлі mystyle.css:
  розмір шрифту — 40 пунктів, колір шрифту — червоний, а гарнітура — Comic Sans MS.</p>
</body>
</html>
```

4. Збережіть цей файл з іменем **przyklad1.html** у тій самій папці, що і **mystyle.css**.

5. Відкрийте файл **pryklad1.html** у вікні браузера (рис. 5.1). Як бачите, хоча в цьому документі не задано жодних параметрів для тегу `<P>...</P>`, текст абзацу набув нових властивостей, які описані в зовнішній таблиці стилів **mystyle.css**.

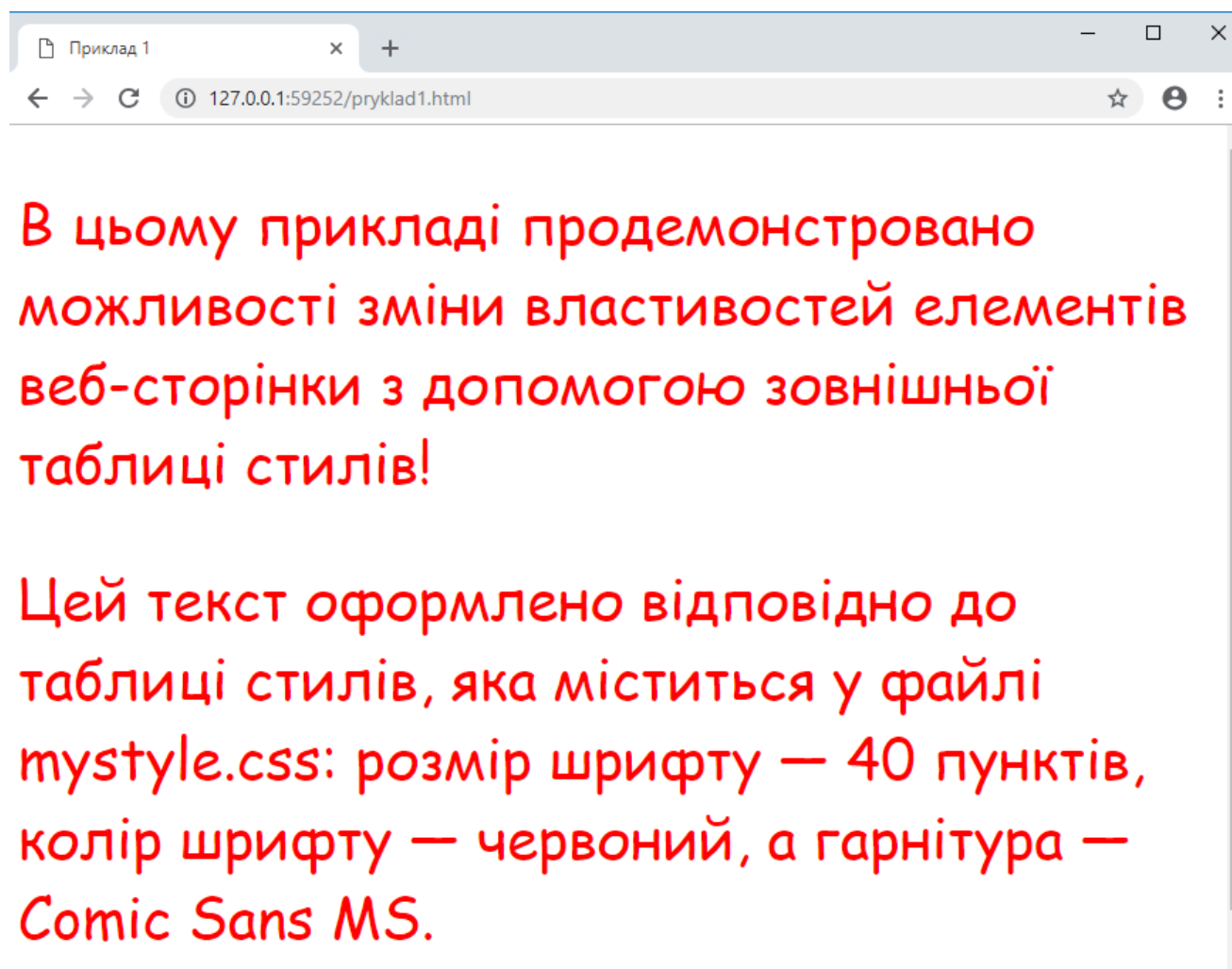


рис. 5.1

Перевага такого методу підключення CSS полягає у тому, що одну таблицю стилів можна підключати до багатьох документів, і всі вони матимуть єдине стильове оформлення.

Використання стилів внутрішньої таблиці

Внутрішню таблицю стилів (Embedded Style Sheet) розміщують безпосередньо в розділі HEAD, у блоці, який обмежений тегами `<STYLE>` та `</STYLE>`.

```
<HEAD>
<STYLE>
  Тег1 {властивість 11: значення11; властивість 12: значення12; ...;
властивість13:значення13;}
  Тег2 {властивість21: значення21; властивість22: значення22; ...;
властивість23: значення23;}
</STYLE>
</HEAD>
```

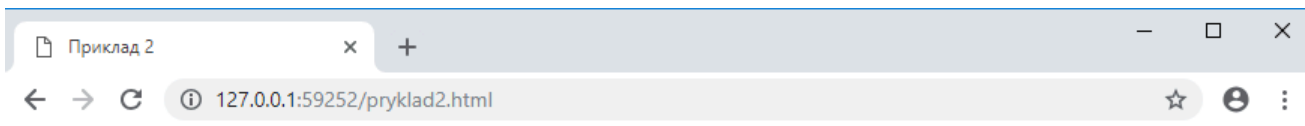
Як видно з коду, найпростіша внутрішня таблиця стилів — це послідовність визначень тегів, кожне з яких записується, як правило, з нового рядка. Визначення тегу містить його ім'я без кутових дужок, за яким у фігурних дужках через крапку з комою перелічують властивості тегів та їхні значення, розділені двокрапками. Розглянемо приклад.

Вправа 5.2

Визначимо стилі заголовків першого та другого рівнів із використанням тегу `<STYLE>`.

```
pryklad2.html - Блокнот
Файл Редагування Формат Вигляд Довідка
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Приклад 2</title>
  <style>
    h1{color: red; font-size: 48px;}
    h2{color: blue; font-size: 20pt;}
  </style>
</head>
<body>
  <h1>Для заголовка першого рівня визначено розмір 48pt, а колір тексту — червоний </h1>
  <h2>Для заголовка другого рівня визначено розмір 20pt, а колір тексту — синій</h2>
  <p>Для цього абзацу стиль не застосовано, для оформлення тексту використано атрибути за умовчанням.</p>
</body>
</html>
```

У цьому описі стилю змінено оформлення заголовків H1 та H2. Зокрема, для першого призначено розмір 48 пунктів і червоний колір, а для другого — розмір 20 пунктів і синій колір. У вікні браузера цей документ матиме вигляд, як на рис. 5.2.



**Для заголовка першого рівня
визначено розмір 48pt, а колір тексту
— червоний**

**Для заголовка другого рівня визначено розмір 20pt, а колір тексту —
синій**

Для цього абзацу стиль не застосовано, для оформлення тексту використано атрибути за умовчанням.

Рис. 5.2. Стилі заголовків першого та другого рівнів

У таблицях можна створювати нові стилі, надаючи їм імена з крапкою перед першим символом. У тегах звертаються до такого стилю за іменем, використовуючи атрибут **CLASS=ім'я_стилю**, де ім'я записують вже без крапки. Для одного тагу можна використовувати декілька стилів форматування.

Вправа 5.3

Застосовуючи атрибут CLASS тегу `</P>`, задамо для двох абзаців різні стилі: першому надамо розмір шрифту 38 пунктів, білий колір символів і оливковий колір тла, другому — розмір шрифту 46 пунктів, фіолетовий колір символів і рожевий колір тла.

pryklad3.html - Блокнот

Файл Редагування Формат Вигляд Довідка

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Приклад 3</title>
  <style>
    .style1{font-size: 38px; color:white; background-color: olive;}
    .style2{font-size: 46px; color:magenta; background-color: mistyrose;}
  </style>

</head>
<body>
  <p class="style1">До цього абзацу застосовано стиль style1</p>
  <p class="style2">До цього абзацу застосовано стиль style2</p>
</body>
</html>
```

Який вигляд має цей документ у вікні браузера, показано на рис. 5.3.

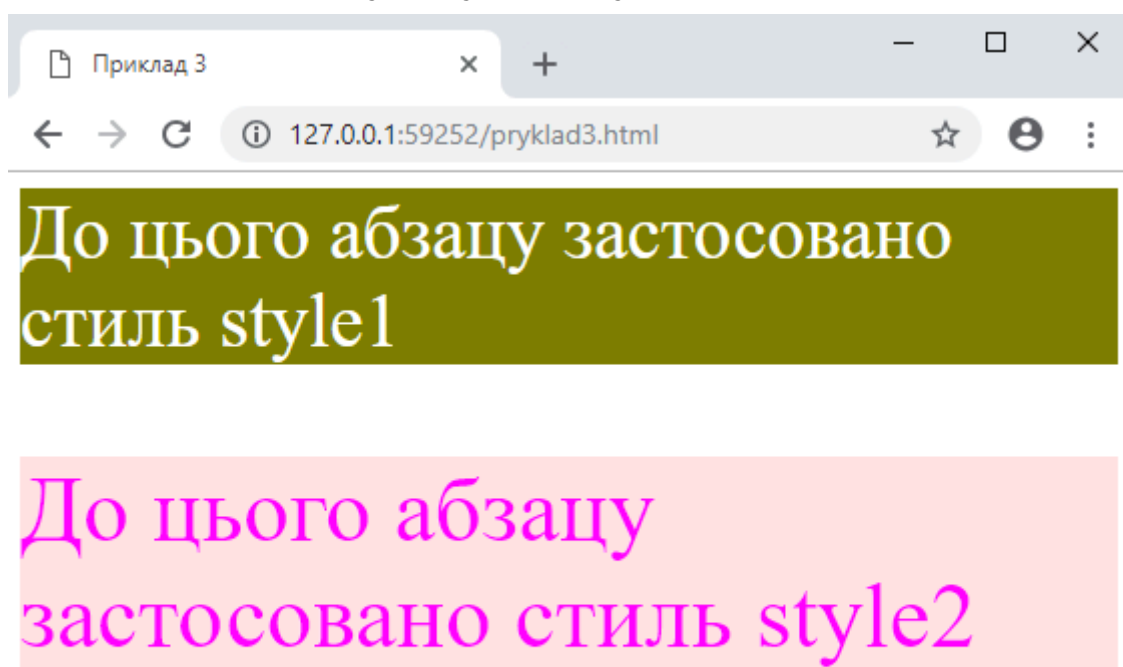


Рис. 5.3. Застосування різних стилів форматування для тегу абзацу

Стилі дають змогу сформувати сторінку з цікавими ефектами, які інакше можна створити лише за допомогою графіки. Це, наприклад, ефект об'ємного тексту або накладання (часткового перекриття) фрагментів сторінки.

Стилі дають змогу сформувати сторінку з цікавими ефектами, які інакше можна створити лише за допомогою графіки. Це, наприклад, ефект об'ємного тексту або накладання (часткового перекриття) фрагментів сторінки.

Вправа 5.4

Створення текстових ефектів за допомогою стилів

Визначимо для тегу <BODY> такий стиль: шрифт **Arial** чорного кольору розміром 16 пунктів. Внаслідок цього весь текст, що міститься між тегам <BODY> та </BODY>, буде автоматично відформатований відповідно до цих властивостей. Тепер визначимо стилі з іменами *тінь*, *основа*, *шар1* і *шар2*. Текст запишемо в шарах, які накладаються. Спочатку відображатиметься шар *тінь*, а на нього накладатимуться шари *основа*, *шар1* та *шар2*. Порядок накладання задано розташуванням фрагментів тексту в HTML-документі.

Буде застосовано такі властивості:

margin-top — відступ згори (за від'ємних значень можна забезпечити накладання фрагментів);

color — колір;

font-size — розмір шрифту;

font-family — сімейство шрифтів (гарнітура);

line-height — висота рядка.

Визначені в документі стилі мають такі параметри:

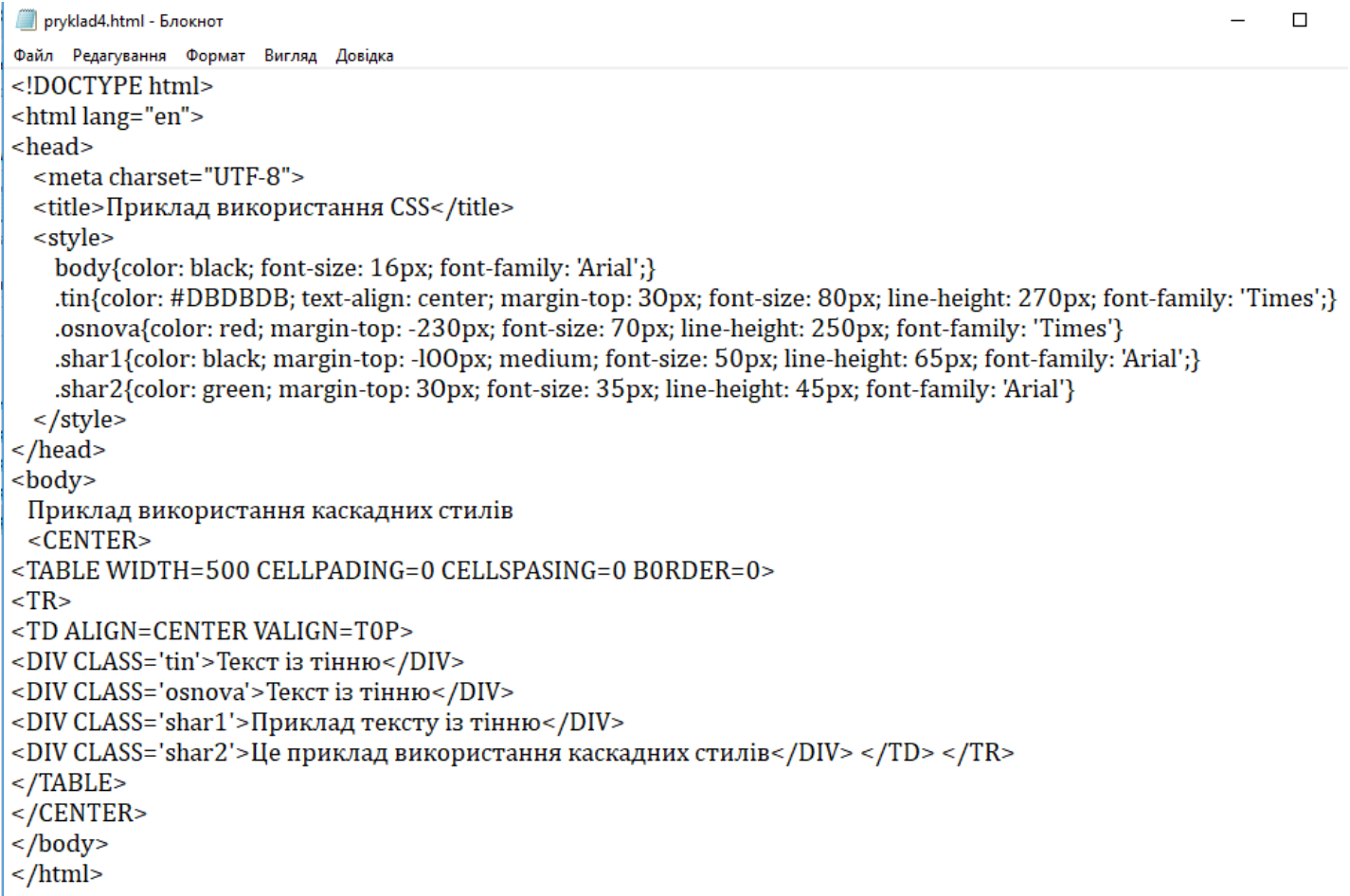
.tin — колір зі значенням #DBDBDB (світло-сірий), вирівнювання по центру, відступ згори — 30 пікселів, розмір шрифту — 80 пікселів, висота рядка — 270 пікселів, гарнітура — Times;

.osnova — червоний колір шрифту, відступ згори — -230 пікселів, розмір — 70 пікселів, висота рядка — 250 пікселів, гарнітура — Times;

.shar1 — чорний колір шрифту, відступ згори----- 100 пікселів, розмір шрифту — 50 пікселів, висота рядка 65 пікселів, гарнітура — Arial;

.shar2 — зелений колір шрифту, відступ згори — 30 пікселів, розмір шрифту — 35 пікселів, висота рядка — 45 пікселів, гарнітура — Arial.

Для виділення частини HTML-документа використовують тег **<DIV>**. Він нічого не форматує, а лише відзначає фрагмент тексту, який виступає як окремий об'єкт. Атрибут **CLASS** цього тегу дає змогу посилатися на стилі внутрішньої таблиці й тим самим задавати стиль подання тексту, розташованого між тегами **<DIV CLASS...>** та **</DIV>**.



```
pryklad4.html - Блокнот
Файл Редагування Формат Вигляд Довідка
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Приклад використання CSS</title>
  <style>
    body{color: black; font-size: 16px; font-family: 'Arial';}
    .tin{color: #DBDBDB; text-align: center; margin-top: 30px; font-size: 80px; line-height: 270px; font-family: 'Times';}
    .osnova{color: red; margin-top: -230px; font-size: 70px; line-height: 250px; font-family: 'Times'}
    .shar1{color: black; margin-top: -100px; medium; font-size: 50px; line-height: 65px; font-family: 'Arial';}
    .shar2{color: green; margin-top: 30px; font-size: 35px; line-height: 45px; font-family: 'Arial'}
  </style>
</head>
<body>
  Приклад використання каскадних стилів
  <CENTER>
<TABLE WIDTH=500 CELLPADDING=0 CELLSPACING=0 BORDER=0>
<TR>
<TD ALIGN=CENTER VALIGN=TOP>
<DIV CLASS='tin'>Текст із тінню</DIV>
<DIV CLASS='osnova'>Текст із тінню</DIV>
<DIV CLASS='shar1'>Приклад тексту із тінню</DIV>
<DIV CLASS='shar2'>Це приклад використання каскадних стилів</DIV> </TD> </TR>
</TABLE>
</CENTER>
</body>
</html>
```

Вигляд цього документа після відкриття його у вікні браузера, показано на рис. 5.4.

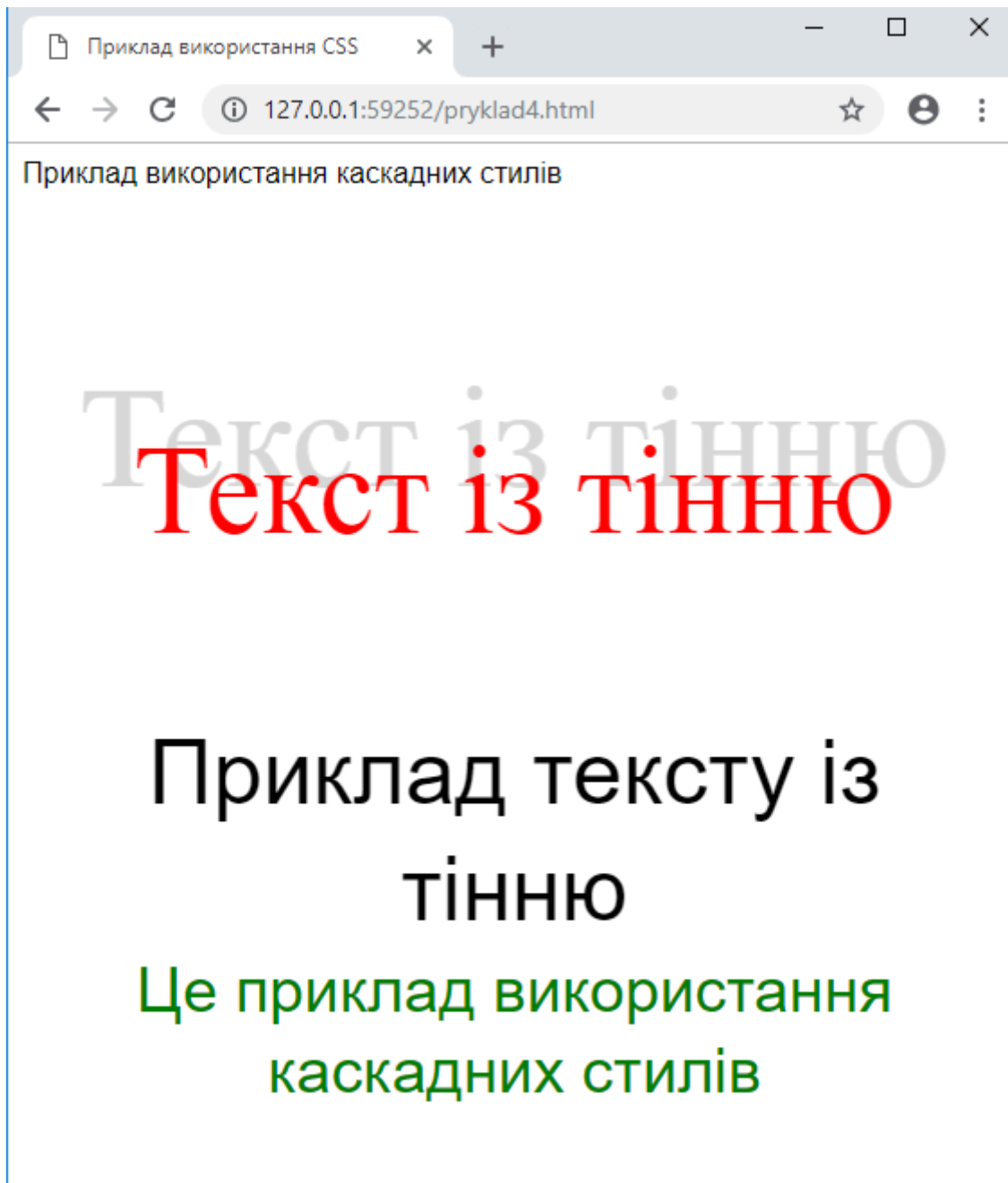


Рис. 5.4. Приклад накладання рядків тексту

Вбудовані стилі

Вбудовані стилі (Inline Styles) вставляють у теги заголовків <H1>... <H6>, абзацу <P>, тіла <BODY>, а також у теги <DIV>, тощо за допомогою атрибута STYLE, в якому перелічують властивості та їх значення. Наприклад:

```
<P STYLE="font-size: 48pt; color: yellow;">
```

Визначені у такий спосіб властивості мають найвищий пріоритет порівняно з іншими, оскільки вони визначені безпосередньо у тегу. Цей підхід використовують для оформлення невеликої кількості елементів.

Приклад використання атрибута STYLE для форматування заголовка другого рівня:

```
<H2 STYLE="font-size: 48pt; font-family: 'Arial'">Текст...</H2>
```

Позиціювання елементів веб-сторінки за допомогою стилів

Розміщувати елементи на сторінці можна не лише за допомогою фреймів і таблиць — каскадні стилі надають для цього додаткові цікаві можливості.

Зокрема, можна обирати один із трьох типів позиціювання елементів на сторінці: статичне, відносне та абсолютне позиціювання. Тег визначається атрибутом POSITION, який може набувати таких значень:

static — статичне; задане за умовчанням і передбачає розміщення чергового об'єкта на вільному місці після попереднього;

relative — відносне; визначає розташування об'єкта відносно того місця, яке б він займав, якби для нього було використане статичне позиціювання;

absolute — абсолютне; передбачає розміщення об'єкта шляхом задавання точного місця його розташування на сторінці.

Для відносного та абсолютного позиціювання використовують чотири властивості — left (ліворуч) та right (праворуч), які задають відступи для розміщення об'єкта по горизонталі, і top (згори) та bottom (знизу) — по вертикалі. Необхідно задавати

по одному значенню позиції по горизонталі та вертикалі. При цьому елементи можуть накладатися один на інший: нижче буде розташований об'єкт, описаний першим, а зверху — той, який описали останнім.

Якщо потрібна зміна стандартного порядку накладання, застосовують шари. Шар об'єкта задають за допомогою властивості Z-index, значенням якої можуть бути лише ціле число та auto (за умовчанням). Об'єкт із більшим значенням z-index розміститься поверх об'єкта з меншим значенням цієї властивості, перекриваючи його. Якщо ж об'єкти матимуть однакові її значення, то зверху розташується об'єкт, описаний нижче за текстом у HTML-документі.

Вправа 5.4

Розглянемо, як позиціювати три об'єкти-зображення і два фрагменти тексту. Координати спеціально вибрані так, щоб об'єкти перекривалися. Зауважте, що слід використовувати атрибут STYLE, а не тег <STYLE>.

У програмі Блокнот введіть такий текст веб-сторінки:

```
<HTML>
<HEAD>
<TITLE>Позиціювання</TITLE>
</HEAD>
<BODY>
<DIV STYLE="position: absolute; top: 0; left: 70; width: 50; height:
100">
<IMG SRC= "fly. jpg"></DIV>
```

```

<DIV STYLE="position: absolute; top: 10; left: 15; width: 600;
height: 100">
  <H1 STYLE="color : yellow">Повій, вітре, до схід сонця, до схід
сонця, край віконця.</H1></DIV>
  <DIV STYLE="position: absolute; top: 60; left: 400; width: 50 ;
height : 100">
    <H1 STYLE = "color: blue">Чайка</H1></DIV>
</BODY>
</HTML>

```

Тег <DIV> тут виконує роль контейнера: елементи, що містяться в ньому, успадковують його властивості, зазначені в атрибуті STYLE. Перший контейнер містить зображення, тип розташування якого є абсолютним, з нульовим відступом згори.

70 пікселів — відступ зліва; його ширина становить 50, висота — 100 пікселів. Другий контейнер містить текст «Повій, вітре, до схід сонця, до схід сонця, край віконця», він зміщений на 10 пікселів від початку сторінки вниз та на 15 вліво, ширина контейнера — 600, висота — 100 пікселів; завдяки використанню тегу H1 із атрибутом !УББ текст має відповідний до заголовка першого рівня розмір та жовтий колір. Третій контейнер зміщений на 60 пікселів униз та 400 ліворуч, ширина контейнера — 50, висота — 100 пікселів. Слово «Чайка» оформлене як заголовок першого рівня синього кольору.

Збережіть цей документ у файлі з іменем pruklad2.html. Скопіюйте у папку з цим документом зображення fly.jpg (можна використати інший файл, але його назва має збігатися з назвою в ИТМБ-документі).

Відкрийте файл pruklad2.html у браузері (рис. 5.5) — контейнери перекриваються

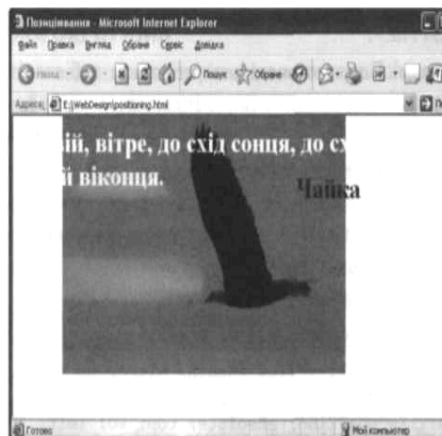


Рис. 5.5. Позиціювання фрагментів тексту і зображення за допомогою стилів

згідно з розташуванням у документі.

Внесіть зміни у текст веб-сторінки, який стосується першого контейнера (із зображенням). Для цього клацніть правою кнопкою миші вільне місце сторінки у вікні браузера і виберіть пункт Перегляд HTML-коду (View Source). У вікні редактора, що з'явиться, внесіть такі зміни:

```

<DIV STYLE="position: absolute; top: 0; left: 70; width: 50; height:
100; z-index: 2">

```

```
<IMG SRC="fly.jpg"></DIV>
```

Решту документа залиште без змін. Хоча в тексті HTML-документа зображення описане першим, завдяки тому, що йому присвоєно індекс із більшим номером, воно відобразиться поверх інших об'єктів.

Збережіть внесені зміни (командою Файл > Зберегти), закрийте текстовий редактор і оновіть веб-сторінку за допомогою команди Вигляд • Оновити або функціональної клавіші F5. Вигляд цього документа у вікні браузера показаний на рис. 5.6.

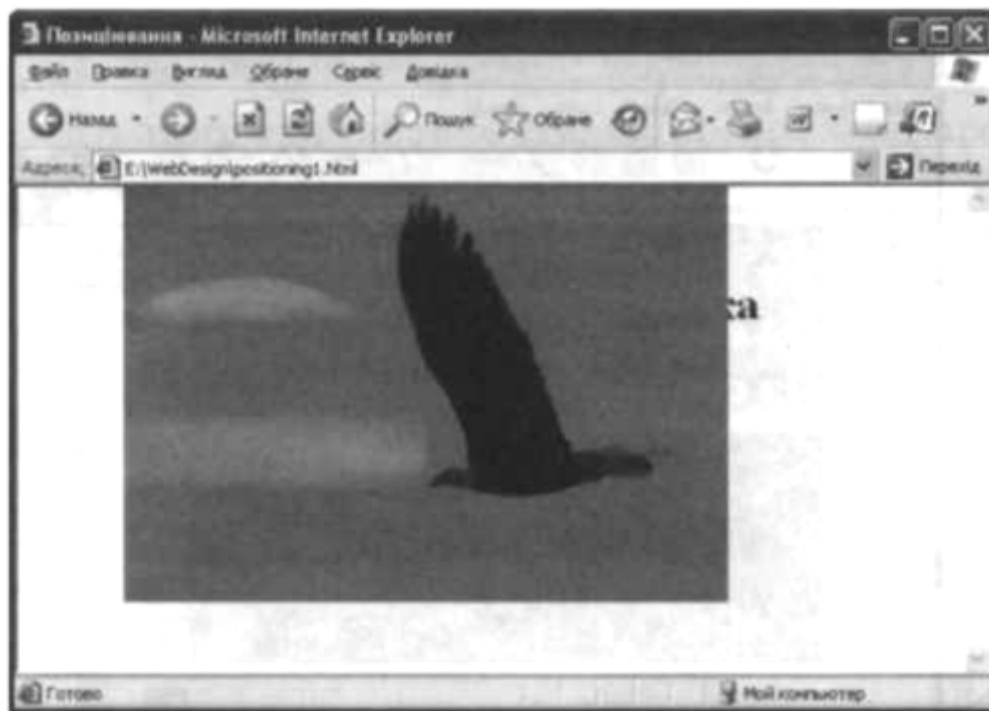


Рис. 5.6. Використання властивості z-index

Отже, використання властивості z-index дає змогу відображати об'єкти незалежно від порядку їх розташування в тексті HTML-документа.

Під час відображення сторінки може статися, що розміри елемента перевищують розміри наданого йому місця. Наприклад, текст і малюнок не вміщуються у виділений для них прямокутник. У таких випадках використовують властивість overflow (переповнення). Вона може мати три значення:

- попе — якщо елемент вийде за межі наданого місця, він все одно буде показаний;
- clip — об'єкти, що виходять за межі, будуть обрізані;
- scroll — буде використано прокручування.

Відкрийте HTML-код сторінки і внесіть у другий контейнер, що містить текст «Повій, вітре, до схід сонця, до схід сонця, край віконця», такі зміни:

```
<DIV STYLE="position: absolute; top: 10; left: 15; width: 220; height: 120; overflow: scroll">
```

```
<H1 STYLE="color: yellow">Повій, вітре, до схід сонця, до схід сонця,  
край віконця </H1></DIV>
```

Збережіть зміни та оновіть веб-сторінку в браузері. Який вигляд вона матиме, показано на рис. 5.7.

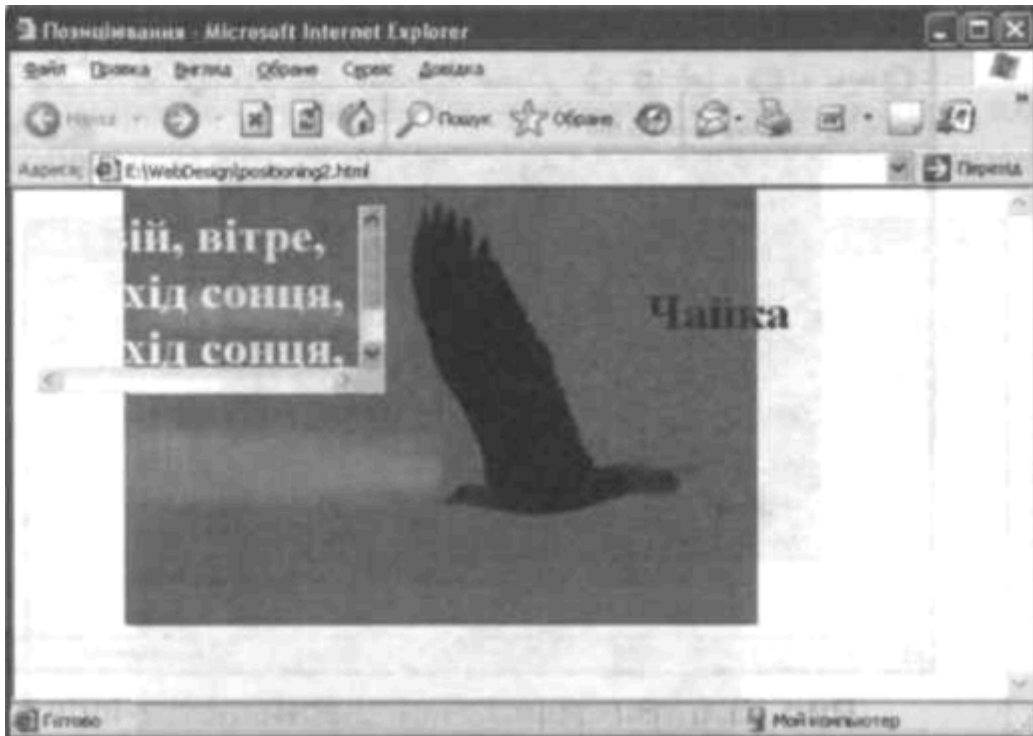


Рис. 5.7. Використання властивості overflow для реалізації прокручування

6. Застосування отриманих знань

Самостійна робота

7. ПІДБИТТЯ ПІДСУМКІВ УРОКУ

8. ДОМАШНЄ ЗАВДАННЯ

Вивчити конспект.

Виконати завдання с.р.