

**Поняття про мову розмітки,  
гіпертекстовий документ  
та його елементи.**

**Поняття тегу й атрибуту, теги  
форматування шрифтів і поділу  
тексту на рядки, абзаци.**

**Теги заголовку й тіла веб-сторінки.**

# 1. Базові конструкції мови HTML

**Гіпертекст** — це текст, у який вбудовані спеціальні коди, що задають форматування тексту, наявність у ньому ілюстрацій, мультимедійних вставок та гіперпосилань.

**HTML** (Hyper Text Markup Language – мова гіпертекстової розмітки) – мова тегів, якою пишуться гіпертекстові документи для мережі Інтернет.

## 2. Поняття тегу

Вся інформація про форматування документу міститься в фрагментах розташованих між знаками **< >** — такий фрагмент називають *тегом*.

Відкриваюча дужка **< ім'я тега, атрибут >** закриваюча дужка

Приклади тегів HTML:

**<TITLE>**, **<BODY>**, **<TABLE>**, **</A>**, **<IMG>**, **</CENTER>**.

Теги бувають:

- *Контейнерні (парні)*
- *поодинокі (одинарні)*

Приклади парних тегів:

*<HTML> </HTML>*,  
*<B> </B>*,  
*<HEAD></HEAD>*, *<H3></H3>*,  
*<ADDRESS></ADDRESS>*, *<LI></LI>*.

Приклади одинарних тегів:

*<BR>*,  
*<HR>*,  
*<META>*, *<BASEFONT>*,  
*<FRAME>*, *<INPUT>*.

**Відкривальні теги** (вказують програмі на початок об'єкту) можуть містити атрибути, які впливають на ефект, створюваний тегом.

**Атрибути** — задають значення властивостей об'єкту, поміщеного у контейнер.

Приклад тегу з атрибутами:

```
<BODY BGCOLOR="#000000">
```

### 3. Структура HTML-документа

HTML-документ складається з *основного тексту* і *тегів розмітки*.

HTML-документи містяться у файлах із розширенням *.htm* або *.html*.

*Структура:*

*<HTML>*

*<HEAD>*

*<TITLE>* *Заголовок документа*

*</TITLE>*

*</HEAD>*

*<BODY>*

*Текст, що відображається на екрані*

*</BODY>*

*</HTML>*

**<HTML> </HTML>** - починає та закінчує документ

**<HEAD> </HEAD>** - починає та закінчує заголовок документу

**<TITLE> </TITLE>** - задає назву документу

**<BODY></BODY>** - починає та закінчує тіло документу

# Заголовки

Для того щоб розбити текст на логічні частини, використовують **заголовки**.

(Існує 6 рівнів заголовків документів).

Вони позначені тегами від **<H1>...</H1>** до **<H6>...</H6>**.

найвищий  
рівень

найнищий  
рівень



## Абзаци та рядки

Для визначення звичайних *абзаців* у мові HTML використовують теги `<P>` і `</P>`.

Теги `<H>` та `<P>` можуть додатково містити *атрибут* ***ALIGN***, який додатково використовується для того, щоб вирівняти текст заголовку або абзацу по центру, за правим або лівим краєм.

Наприклад:

```
<H2 ALIGN=center>
```

```
<P ALIGN=right>.
```

## Одинарні теги

**<BR>** - одинарний тег, що використовується для переходу на новий рядок без створення абзацу

**<HR>** - одинарний тег, що використовується для створення горизонтальної лінії, яка візуально відділяє різні частини документа.

**Кольори** та **зображення** для всього документа і його тла задають за допомогою тегу **<BODY>**. Він може мати такі атрибути:

- **BACKGROUND="URL"** — замість URL вказують адресу малюнка, який має бути тлом для сторінки;
- **BGCOLOR=значення** — задає колір, який має бути фоновим для документа;
- **TEXT=значення** — задає колір тексту;
- **LINK=значення** — визначає колір гіперпосилань у документі;
- **ALINK=значення** — задає колір гіперпосилань під час клацання;
- **VLINK=значення** — задає колір переглянутих гіперпосилань.

Гарнітуру, розмір та колір шрифту для фрагмента тексту задають за допомогою теґу

**<FONT АТРИБУТИ> Текст </FONT>.**

Наприклад:

<font size=«7»> текст </font>

.....

<font size=«1»> текст </font>

<font color=darkblue> текст </font> - означає, що літери слова комп'ютер матимуть темно-синій колір.

## 4. Теги форматування символів тексту

<i>Теги</i>	<i>Ефект</i>	<i>Приклад</i>
<code>&lt;B&gt;</code> та <code>&lt;/B&gt;</code>	Напівжирне накреслення	<b>bold</b>
<code>&lt;I&gt;</code> та <code>&lt;/I&gt;</code>	Курсивне накреслення	<i>italic</i>
<code>&lt;U&gt;</code> та <code>&lt;/U&gt;</code>	Підкреслене	<u>underlined</u>
<code>&lt;S&gt;</code> та <code>&lt;/S&gt;</code>	Перекреслене	<del>stricken</del>
<code>&lt;SUP&gt;</code> та <code>&lt;/SUP&gt;</code>	Верхній індекс	<sup>superscript</sup>
<code>&lt;SUB&gt;</code> та <code>&lt;/SUB&gt;</code>	Нижній індекс	<sub>subscript</sub>

Є й інші теги, які можна застосовувати для зміни параметрів тексту.

- **<BIG>** Текст **</BIG>** — збільшення розміру шрифту.
- **<SMALL>** Текст **</SMALL>** — зменшення розміру шрифту.
- **<EM>** Текст **</EM>** — виділення важливих фрагментів тексту. (Текст відображатиметься курсивом)
- **<DEL>** Текст **</DEL>** — створення перекресленого тексту. (Текст буде перекреслено горизонтальною лінією)

# **Тема уроку:**

**“Нумеровані та марковані списки  
на веб-сторінках  
Способи керування структурою  
та розміщенням інформації  
на веб-сторінках”**

У HTML-документах  
використовують три види списків:

- неупорядкований (маркований);
- упорядкований (нумерований);
- СПИСОК ВИЗНАЧЕНЬ.



У *невпорядкованому списку* для виділення елементів використовуються певні позначки (крапки, квадрати тощо). Починається такий список із тегу **<UL>**, а закінчується — **</UL>**.

Кожний елемент списку починається з тегу **<LI>**. Тег **<UL>** може мати атрибут **TYPE**, який визначає форму позначки. Цей атрибут набуває таких значень:

- **disk** — зафарбоване коло;
- **circle** — коло;
- **square** — маленький чорний квадрат.

```
<HTML>
<HEAD>
<TITLE></TITLE>
</HEAD>
<BODY>
<UL TYPE=disk>
<LI> I-курс </LI>
<LI>II-курс </LI>
<LI>III-курс </LI>
</BODY>
</HTML>
```



В *упорядкованому списку* всі елементи пронумеровані. починається список із тегу **<OL>**, а закінчується — **</OL>**. Кожний елемент такого списку також починається з тегу **<LI>**. Тег **<OL>** може мати атрибут **TYPE**, який визначає тип нумерації. Цей атрибут набуває таких значень:

- A — велика літера;
- a — мала літера;
- I — велика римська цифра;
- i — мала римська цифра;
- 1 — арабська цифра.

У разі потреби за допомогою атрибута START можна задати відмінний від одиниці початковий номер елемента, наприклад **<OL TYPE="I" START="5">**, Нумерований список, початковий елемент якого позначено латинською літерою E, яка є п'ятою в алфавіті, описують так:

```
<OL TYPE="A" START="5">
```

```
<LI>Перший елемент списку</LI>
```

```
<LI>Другий елемент списку</LI>
```

```
<LI>Третій елемент списку</LI>
```

```
<LI>Четвертий елемент списку</LI>
```

```
</OL>
```



**<BASEFONT>** для всього документа призначається розмір шрифту.

Наприклад: **<BASEFONT SIZE=6>**

**<FONT>** можна визначити колір шрифту для кожного елемента списку.

Наприклад:

```
<HTML>
<TITLE>Райдуга</TITLE>
<BODY>
<BASEFONT SIZE=6>
<OL TYPE=i>
<LI><FONT COLOR=red>Червоний</FONT></LI>
<LI><FONT COLOR=orange>Оранжевий</FONT></LI>
<LI><FONT COLOR=yellow>Жовтий</FONT></LI>
<LI><FONT COLOR=green>Зелений</FONT></LI>
<LI><FONT COLOR=lightblue>Блакитний</FONT></LI>
<LI><FONT COLOR=blue>Синій</FONT></LI>
<LI><FONT COLOR=darkmagenta>Фіолетовий</FONT></LI>
</OL>
</BODY>
</HTML>
```



## Текстові гіперпосилання

Для створення гіперпосилання необхідно використовувати *теги* **<A>** та **</A>**, визначивши для тегу **<A>** *атрибут* **HREF**.

Його значенням має бути адреса URL, на яку вказує посилання.

Текст посилання розташовують між тегами **<A>** і **</A>**.

Якщо веб-сторінка, на яку вказує посилання, розміщена на іншому веб-сайті, то значенням атрибута HREF має бути повна URL-адреса з назвою протоколу включно; такі посилання називають *зовнішніми*. Якщо ж гіперпосилання вказує на іншу сторінку того самого сайту, то для пошуку документа достатньо задати лише відносний шлях; таке посилання називають *внутрішнім*.

Гіперпосилання може вказувати на певне місце всередині сторінки, якщо туди попередньо вбудувати *якір-мітку*.

Для визначення якоря також використовують теги **<A>** і **</A>**, але замість атрибута **HREF** задають атрибут **NAME**, значенням якого має бути ім'я якоря. *Воно може складатися з літер та цифр, але не повинно містити символів пробілу.* Якщо на сторінці є кілька міток, то всі вони повинні мати різні назви.

Для створення посилання на встановлений якір потрібно у тегу **<A>** зазначити його ім'я в кінці адреси URL після імені документа, відокремивши його символом **#**. Символ **#** вказує на те, що після нього записано назву мітки, а не ім'я файлу. Посилання на мітку всередині поточного документа задають так:

**<A HREF="#назва\_мітки">Текст посилання</A>**.

Якщо в атрибуті HREF задати адресу електронної пошти зі словом **mailto:** перед нею, то після вибору такого посилання можна надіслати електронний лист, де в полі Кому буде записано цю адресу.

...

<BODY>

Житомирський обласний педагогічний ліцей Житомирської обласної ради має свій

<A HREF="http://www.vpedliceum.blogspot.com">блог</A>.

<P> А тепер можна перейти на

<A HREF="II-m.html">

іншу сторінку</A>.

<P>Про те, як зв'язатися з автором, розповідається

в <A HREF="#olenap">

кінці цієї сторінки</A>

<BR><BR>

Можна використати матеріали, що розміщені

<A HREF="text.doc"> в цьому текстовому документі.</A>

<BR><BR>

<A NAME="olenap"

HREF="mailto:pedliceum@mail.ru">pedliceum@mail.ru</A>

</BODY>

...



```
<HTML>
<HEAD>
<TITLE></TITLE>
<HTML>
<BODY LINK=magenta ALINK=yellow>
<H1><A HREF="1.html">Наша
продукція</A></H1>
<H1><A HREF="1.html">Наші
замовники</A></H1>
</BODY>
</HTML>
```



*Тема: Використання  
таблиць у HTML-  
документах*



Для створення таблиці використовують чотири елементи. Таблицю описують за допомогою тегів **<TABLE>** ... **</TABLE>**; вона повинна мати один або кілька рядків **<TR>**...**</TR>**, у кожному з яких може міститися заголовок **<TH>**...**</TH>** або дані **<TD>**...**</TD>**.

За умовчанням таблиця має невидимі межі комірок. Для того щоб зробити їх видимими, використовують атрибут **BORDER**, який має цілочислове значення, що визначає товщину рамки в пікселях, наприклад **<TABLE BORDER=5>**.

Таблиця може мати заголовок, який задається тегами **<CAPTION>** та **</CAPTION>**. Тег **<CAPTION>** може містити атрибут *ALIGN* з одним із значень *top* або *bottom*, які визначають розташування заголовка відповідно перед таблицею або після неї (за умовчанням встановлено значення *top*). Наприклад, **<CAPTION ALIGN=bottom>** Розклад уроків **</CAPTION>**.

Кожний рядок таблиці починається з тегу **<TR>** і закінчується тегом **</TR>**. Якщо рядок містить *заголовки стовпців таблиці*, то використовують теги **<TH>** і **</TH>**, якщо ж *дані* — то **<TD>** і **</TD>**.

Кілька комірок можуть бути об'єднані в одну як по горизонталі, так і по вертикалі. Об'єднання першого типу застосовують тоді, коли потрібно створити для кількох стовпців спільний заголовок.

Коли вміст кількох комірок поспіль у стовпці однаковий, їх об'єднують по вертикалі.

Для об'єднання комірок використовують такі атрибути: **COLSPAN** (об'єднання по горизонталі, у рядку) і **ROWSPAN** (по вертикалі, у стовпці) тегу `<TD>`. Значеннями цих атрибутів є кількість об'єднаних стовпців або рядків. Наприклад, **COLSPAN=3** означає, що комірка розтягнута на 3 стовпці, а **ROWSPAN=2** — що комірка займає 2 рядки.

Розглянемо далі, як настроювати ширину всієї таблиці, а також окремих комірок. Зауважимо, що ширина всіх комірок в одному стовпці однакова, а в одному рядку може бути різною. *Ширину таблиці задають атрибутом **WIDTH** у тегу `<TABLE>`, Ширину таблиці можна комірки — за допомогою того самого атрибута у тегу `<TD>` або `<TH>`.* а визначати як у пікселях, так і у відсотках від ширини вікна, а комірки — у пікселях або у відсотках від ширини таблиці.

```
<HTML>
<TITLE>Приклад встановлення ширини таблиці</TITLE>
<BODY>
<TABLE BORDER="1" WIDTH=500>
<TR>
<TD ROWSPAN="2" WIDTH=50%>Класи</TD>
<TD WIDTH=250>10-A</TD>
</TR>
<TR>
<TD>10-B</TD>
</TR>
</TABLE>
</BODY>
</HTML>
```

Класи	10-A
	10-B

Вся таблиця має ширину 500 пікселів. Ширина комірки з текстом «Класи» становить 50 % від ширини всієї таблиці, а комірки з текстом «10-A» — 250 пікселів. Комірка з текстом «10-B» теж завширшки 250 пікселів (хоча для неї ширину не було задано), оскільки вона розташована в одному стовпці з попередньою.

Текст всередині комірок можна вирівнювати як по горизонталі, так і по вертикалі. Горизонтальне вирівнювання вмісту всього рядка задають за допомогою атрибута **ALIGN** у тегах **<TR>**; цей самий атрибут у тегах **<TD>** або **<TH>** в окремих комірках виконує таку саму роль. *Атрибут **ALIGN** може набувати значень **center**, **right**, **left** або **justify**, які задають вирівнювання вмісту таблиці відповідно по центру, за правим чи лівим краєм або за шириною.*

Приклади тегів **<TR>** та **<TD>** із використанням вирівнювання: **<TR ALIGN=right>**. Вертикальне вирівнювання задають у тегах **<TR>**, **<TD>** або **<TH>** за допомогою атрибута **VALIGN**, який може набувати значень ***top**, **bottom** та **middle***, що визначають вирівнювання відповідно за верхнім та нижнім краєм і по центру.

Для оформлення зовнішньої рамки таблиці можна використовувати атрибут **FRAME** тегу **<TABLE>**. Його значення визначають, що браузер відобразить:

- **box** — всі чотири сторони рамки;
- **above** — лише верхню межу рамки;
- **below** — лише нижню межу;
- **lhs** — лише ліву межу;
- **rhs** — лише праву межу;
- **hsides** — верхню й нижню межі рамки;
- **vsides** — ліву й праву межі рамки;
- **void** — зовнішня рамка не відобразиться.

Відображення розділювальних ліній між стовпцями та рядкам таблиці визначають за допомогою атрибута **RULES** тегу **<TABLE>**.

Він може набувати таких значень:

- **all** — відображати всі вертикальні та горизонтальні лінії;
- **rows** — лише горизонтальні лінії між рядками;
- **cols** — лише вертикальні лінії між стовпцями;
- **none** — не показувати розділювальних ліній.

## Колірне оформлення таблиць

Для оформлення комірок таблиць за допомогою кольорів застосовують атрибут **BGCOLOR**. Колір можна визначати, вказуючи його назву англійською мовою або символ # та шістнадцяткове число.

*Якщо потрібно задати колір для всієї таблиці, то атрибут BGCOLOR задають у тегу <TABLE>, для зміни кольору лише в одному рядку — в тегу <TR>, а настроювання кольору окремої комірки забезпечує наявність цього атрибута всередині тегу <TD> або <TH>.*

Колір рамки таблиці можна задати за допомогою таких атрибутів:

- **BORDERCOLOR** — колір ВСІЄЇ рамки;
- **BORDERCOLORLIGHT** — колір світлої частини рамки;
- **BORDERCOLORDARK** — колір темної частини рамки.

Ці атрибути вставляють у тег <TABLE>. Щоб вони діяли, необхідна наявність ще й атрибута **BORDER**, який задає товщину рамки.

*Фрейми, їхні теги й  
атрибути*

Веб-сторінка сайту може містити кілька блоків або вікон, які називають **фреймами**, або **кадрами**. У кожному з них відображається свій HTML-документ. В одному фреймі може міститися навігаційне меню, а в іншому відкриватися веб-сторінки, на які вказують його пункти.

Для того щоб створити веб-сторінку з фреймами, потрібно кілька HTML-документів. В одному з них задають розмітку екрана, тобто розташовують у вікні браузера фрейми, кожному з яких призначають свої документи.

Сторінка з розміткою, як і звичайна, починається з тегу <HTML> і закінчується тегом </HTML>. Для поділу екрана на кілька фреймів використовують теги **<FRAMESET>** і **</FRAMESET>**. Перший має бути розташований після тегу заголовка, але перед тегом <BODY>.

*Іноді в таких документах зовсім не використовують тег <BODY>.*

Два фрейми можна розташовувати поруч по горизонталі або один над одним. У першому випадку використовують атрибут COLS, а в другому — атрибут ROWS тегу <FRAMESET>. Для поділу вікна на фрейми через кому записують два числа, які визначають розміри фреймів. Для трьох фреймів потрібно три числа. Розміри фреймів вимірюють у пікселях або відсотках від розміру екрана.

Якщо потрібно зазначити, що фрейм займає те місце, яке залишилося, використовують символ \*.



**Наприклад,** тег `<FRAMESET ROWS="150, *">` задає поділ вікна на два горизонтальні фрейми, один з яких має висоту 150 пікселів, а другий займає те місце, що залишилося. Тег `<FRAMESET COLS="20%, 55%, *">` задає поділ вікна на три вертикальні фрейми, один з яких займає 20 % від ширини екрана, другий — 55 %, а третій займає те місце, що залишилося. Можна використовувати одночасно і горизонтальний, і вертикальний поділ вікна на фрейми — це роблять за допомогою вкладення тегів `<FRAMESET>` один в один.

Після поділу екрана на вікна для кожного фрейму слід задати HTML-документ, який відобразатиметься в ньому. Для цього використовують тег `<FRAME>` з атрибутами, що керують властивостями

фреймів:

- **SRC** — задає ім'я файлу, що відобразатиметься у фреймі;
- **NAME** — задає ім'я фрейму;
- **SCROLLING** — визначає наявність (значення `yes`) або відсутність (значення `no`) смуг прокручування у вікні фрейму (за умовчанням — `yes`);
- **NORESIZE** — забороняє користувачу змінювати розміри фрейму;
- **BORDER** — визначає ширину розділювальної смуги між фреймами в пікселях;
- **BORDERCOLOR** — визначає колір розділювальної смуги між фреймами;
- **MARGINHEIGHT** — додає порожнє поле, висота якого визначена в пікселях, між верхньою межею фрейму і початком тексту або графіки;
- **MARGINWIDTH** — додає порожнє поле, ширина якого визначена в пікселях, між боковими межами фрейму і початком тексту або графіки.

Оскільки фрейми підтримують не всі браузери, необхідно помістити тег `<NOFRAME>` перед тегом `<BODY>`, а між `<BODY>` і `</BODY>` записати повідомлення, яке з'являтиметься у вікні, якщо браузер не підтримує фрейми. Після `</BODY>` має йти `</NOFRAME>`.

*Формати зображень, що використовуються в Інтернеті, їхні особливості. Розміщення та вирівнювання зображень на веб-сторінках. Карти посилань.*

## Формат GIF (.gif)

**Формат GIF** (Graphics Interchange Format — формат обміну графічними даними) почали використовувати з 1987 року для обміну малюнками через канали зв'язку глобальної мережі. Він зберігає зображення, що можуть містити не більш ніж 256 кольорів, і не залежить від апаратного забезпечення комп'ютера. Окрім цього, під час завантаження таких файлів на веб-сторінку може бути використаний режим interlaced (рядки малюнка відображаються через один). Ця технологія дає змогу побачити приблизний зміст картинки до її повного відтворення і в разі необхідності скасувати

завантаження.

У GIF-файлах можна зробити один чи більше кольорів прозорими: вони будуть невидимими у вікні браузера та деяких інших програм. Також є можливість зберігати в одному файлі кілька картинок, задавши час показу кожної, тобто застосовувати анімацію. На сьогодні фахівцями розроблено чимало готових анімаційних GIF-файлів, які можна використовувати на своїх веб-сторінках.

## **Формат PNG (.png)**

Формат PNG (Portable Network Graphic — переносні мережні графічні дані) є одним із перспективних форматів графіки для Інтернету, який створено з метою заміни GIF.

Формат PNG забезпечує високу якість графіки та прийнятні розміри файлів. Зображення може зберігатися як у реальних кольорах, так і в 256-колірній GIF-палітрі.

## **Формат JPEG (.jpg)**

Формат JPEG (Joint Photographic Experts Group — об'єднана група експертів у галузі фотографії) був створений для того, щоб позбутись обмежень, властивих формату GIF.

Два попередніх формати орієнтовані на рисовану і креслену графіку. Формат JPEG призначений для збереження повноколірних реалістичних фотозображень. Він має потужні засоби для стиснення зображень, щоправда шляхом зниження їхньої якості. JPEG-зображення зберігаються у файлах із розширенням .jpg.

Оскільки цей формат спеціально розробляли для збереження ілюстрацій, що містять велику кількість кольорів, він є найприйнятнішим для деяких типів графічних даних. Це кольорові фотографії, графічні дані з градієнтним заповненням частин зображення, фотознімки з відтінками одного кольору тощо.

## Створення тла веб-сторінки

Зазвичай для тла вибирають зображення невеликого розміру та неяскравих кольорів, а його файл роблять маленького розміру, що дає змогу швидко завантажувати сторінки. Текст на тлі повинен легко читатись.

Для формування тла у вигляді малюнка у тегу **<BODY>** використовують атрибут **BACKGROUND**, значенням якого є URL-адреса файлу зображення. Наприклад,

Тег **<BODY BACKGROUND="1.jpg">**

визначає, що для фонового заповнення веб-сторінки буде використано файл **1.jpg**.

Далі наведено приклад HTML-коду сторінки, в якій тло оформлене у вигляді малюнка, що міститься у файлі **mone.jpg**.

```
<HTML>
```

```
<TITLE></TITLE>
```

```
<BODY BACKGROUND="mone.jpg">
```

```
<H1><A HREF="1.html">Наша продукція</A></H1>
```

```
<H1><A HREF="2.html">Наші замовники</A></H1>
```

```
</BODY>
```

```
</HTML>
```

## Вставлення зображень на веб-сторінку

Для розміщення малюнків у HTML-документі використовують одинарний тег **<IMG>**. Він має обов'язковий атрибут **SRC**, значенням якого є URL-адреса файлу зображення, записана в абсолютній (повністю) або відносній формі (починаючи від поточного каталогу; при цьому «батьківський» каталог позначають за допомогою двох крапок і слешу (. . /)). Під час відкривання документа браузер завантажує малюнок і відображає його в тому місці документа, де розташований тег **<IMG>**.

Графічний об'єкт буде показаний на веб-сторінці у своїх реальних розмірах. Якщо є потреба у їх зміні (масштабуванні об'єкта), нові розміри можна задати за допомогою атрибутів **WIDTH** та **HEIGHT**, значення яких визначають відповідно ширину та висоту зображення в пікселях. При цьому необхідно зберегти пропорції малюнка, інакше він матиме вигляд розтягнутого або сплюсненого. Слід також пам'ятати, що іноді користувачі відключають відображення графіки у вікні браузера, щоб прискорити завантаження документів. Тому, на випадок, коли малюнок на сторінці не буде, бажано повідомити, що на ньому зображено. Для цього використовують альтернативний текст — більш-менш докладний опис зображення, який задають у тегу **<IMG>** як значення спеціального атрибута **ALT**. Якщо браузер не може показати малюнок, він замість нього виводить цей текст.

Розглянемо HTML-документ, в якому використаємо малюнок `flamingo.jpg`, де зображено фламінго. У тегу **<IMG>** задамо атрибути `WIDTH,HEIGHT` та `ALT`.

```
<HTML>  
<TITLE>Малюнок</TITLE>  
<BODY BACKGROUND="flamingo.jpg">  
<IMG SRC="2.jpg" WIDTH=300 HEIGHT=200 ALT="Фламінго">  
</BODY>  
</HTML>
```

## Розміщення зображень у тексті

Зображення можна розмістити у тексті, але при цьому слід визначити, в який спосіб текст обтікатиме його. Для взаємного розміщення тексту і зображень призначений атрибут `ALIGN` у тегу `<IMG>`, який може набувати, зокрема, таких значень:

- **left** — зображення розміщене в лівій частині сторінки, текст обтікає його з правого боку;
- **right** — зображення розміщене у правій частині сторінки, текст обтікає його з лівого боку;
- **top** — обтікання немає, зображення розміщене в тексті, відповідний рядок якого вирівняно за верхньою межею малюнка;
- **bottom** — обтікання немає, зображення розміщене в тексті, відповідний рядок якого вирівняно за нижньою межею малюнка;
- **middle** — обтікання немає, зображення розміщене в тексті, відповідний рядок якого вирівняно по середній лінії малюнка



```
<HTML>
```

```
<TITLE>МАЛЮНОК</TITLE>
```

```
<BODY BACKGROUND="1.jpg">
```

```
<IMG SRC="flamingo.jpg" ALIGN=left HSPACE=10 VSPASE=10>
```

Блакитні води океану та коралові рифи,  
ласкаве сонце і білосніжний прибережний пісок,  
фантастичний підводний світ і казкові птахи,  
буяння барв сходів і заходів, непрохідні ліси  
і п'янкий запах орхідей...

Сприятливий клімат Острова Свободи дає можливість туристам відвідувати його цілий рік. Тут немає зайвої вологості, а спека не така страшна завдяки постійним подувам бризів із океану. Середньорічна температура повітря й води становить + 26—28°C.

```
</BODY>
```

```
</HTML>
```

## Графічні гіперпосилання

*Зображення, як і текст, можна використовувати як посилання.*

*Для цього тег <IMG> необхідно помістити між тегами <A> і </A>.*

*Зображення-посилання має синю рамку, а після наведення на нього вказівника миші той набуває такої самої форми, що й у випадку текстового посилання.*

### Приклад

```
<HTML>
```

```
<TITLE>Птахи</TITLE>
```

```
<BODY BACKGROUND="1.jpg">
```

```
<A HREF="gorobci.html"><IMG SRC="gorobci.jpg"></A>
```

```
<A HREF="flamingo.html"><IMG SRC="flamingo.jpg"></A>
```

```
</BODY>
```

```
</HTML>
```

*Іноді великий за обсягом текст поділяють на частини, кожна з яких розташована на окремій веб-сторінці. У нижній частині поточних сторінок розміщують зображення двох стрілок: Вперед і Назад. Їх настроюють як гіперпосилання на наступну і попередню веб-сторінки — так, як це показано у прикладі.*

```
<HTML>
```

```
<TITLE>PAGE10</TITLE>
```

```
<BODY BACKGROUND="1.jpg">
```

```
<IMG SRC="2.jpg" ALIGN=left HSPACE=10 VSPACE=10>
```

Блакитні води океану та коралові рифи, ласкаве сонце і білосніжний прибережний пісок, фантастичний підводний світ і казкові птахи, буяння барв сходів і заходів, непрохідні ліси і п'янкий запах орхідей...

Сприятливий клімат Острова Свободи дає можливість туристам відвідувати його цілий рік. Тут немає зайвої вологості, а спека не така страшна завдяки постійним подувам бризів із океану. Середньорічна температура повітря й води становить + 26-28°C. <BR>

```
<A HREF="page9.html"><IMG SRC="nazad.jpg"></A>
```

```
<A HREF="page11.html"><IMG SRC="vpered.gif"></A>
```

```
</BODY>
```

```
</HTML>
```

## Карти посилань

Якщо тег <IMG> з ім'ям файлу рисунка розмістити між тегами <A> і </A>, то з цим рисунком буде пов'язане одне гіперпосилання. Можна зробити і по-іншому: поділити рисунок на області, і кожен з них перетворити на гіперпосилання. Такі області називають *гарячими, а повне зображення — картою посилань*. Вказівник миші після наведення на гарячу область набуває форми руки, як і у випадку текстового посилання. Гарячі області можуть бути прямокутними, багатокутними або круглими. Вибираючи зображення, яке заплановано зробити картою посилань, слід подбати про те, щоб гарячі області не перетиналися. Для того щоб відвідувач сайту зрозумів, що це карта гіперпосилань, а не просто гарна картинка, необхідно дати пояснювальні тексти.

## Формування карти гіперпосилань

Графічну карту посилань створюють за допомогою кількох тегів. У тегу **<IMG>** визначають атрибут SRC, щоб задати зображення, і атрибут USEMAP, значенням якого є ім'я карти (має починатися символом #). Власне карту створюють за допомогою тегу **<MAP>** з атрибутом NAME, який містить ім'я карти (визначене в атрибуті USEMAP, але без символу #) та тегу **</MAP>**. Між ними записують теги **<AREA>**, які задають параметри гарячих областей. При цьому використовують такі атрибути:

- HREF — визначає гіперпосилання, пов'язане з областю;
- SHAPE — визначає форму області, його значеннями можуть бути:

- rect — прямокутник;
- polygone або poly — багатокутник;
- circle — коло;

• COORDS — містить координати області у вигляді взятого в лапки СПИСКУ ЧИСЕЛ, РОЗДІЛЕНИХ КОМОЮ. ДЛЯ ПРЯМОКУТНИКА ЗАДАЮТЬ ЧОТИРИ ЧИСЛА — КООРДИНАТИ ВЕРХНЬОГО ЛІВОГО І ПРАВОГО НИЖНЬОГО КУТІВ, ДЛЯ БАГАТОКУТНИКА — ПОСЛІДОВНІ КООРДИНАТИ КОЖНОГО КУТА, ДЛЯ КОЛА — КООРДИНАТИ ЦЕНТРА І

РАДІУС

```
<IMG SRC="map.bmp" USEMAP="#karta">
```

```
<MAP NAME="karta">
```

```
<AREA HREF="1.html" SHAPE="rect" COORDS="70,250,160,270">
```

```
<AREA HREF="2.html" SHAPE="circle" COORDS="260,140,60">
```

```
</MAP>
```

Слід попрацювати над малюнком карти у графічному редакторі, щоб окреслити необхідні області, які пізніше стануть гіперпосиланнями, та визначити координати, що будуть записані як значення атрибута COORDS.

Місце розташування будь-якої точки на малюнку визначають два числа — її горизонтальна та вертикальна координати. Верхня ліва точка має координати (0;0). Більшим горизонтальним координатам відповідають правіші точки, більшим вертикальним — нижчі.

Наприклад, якщо малюнок має розмір 400x300 пікселів, і необхідно визначити прямокутну область, що займає його верхню ліву частину та має вдвічі меншу ширину і висоту, запис тегу <AREA> буде таким:

```
<AREA SHAPE="rect" COORDS="0,0,200,150">
```

У цьому записі атрибут COORDS визначає прямокутник за допомогою двох точок: лівого верхнього кута з координатами (0;0) і правого нижнього з координатами (200; 150) (центральна точка малюнка, що має розмір 400x300).

Якщо гаряча область є колом, то слід зазначити координати лише однієї точки — центра кола, а також задати його радіус, наприклад:

```
<AREA SHAPE="circle" COORDS="200,150,40">
```

Такий запис визначає круглу гарячу область, розміщену в центрі малюнка розміром 400x300. Координати центральної точки (200;150), радіус кола — 40 пікселів. Найскладніший випадок — багатокутна гаряча область, для якої потрібно послідовно зазначати координати всіх кутів. Припустимо, що на малюнку розміром 400x300 пікселів потрібно визначити гарячу область у формі рівнобедреного трикутника, основа якого проходить точно посередині малюнка, займаючи всю його ширину, а вершина міститься посередині верхньої межі. Цю область визначають так:

```
<AREA SHAPE="poly" COORDS="0,150,400,150,200,0">
```

Шість координат задають три точки — кути трикутника. У цьому випадку їх послідовність неважлива, і ви могли б написати, наприклад, 400;150;0;150;200;0, проте коли кількість кутів є більшою, важливим є порядок з'єднання точок.

Наприклад, записи

```
<AREA SHAPE="poly" COORDS="0,0,200,0,200,150,100,75,0,150">
```

```
<AREA SHAPE="poly" COORDS="0,0,200,0,100,75,200,150,0,150">
```

дадуть різні результати. У першому випадку це прямокутник із вирізаним трикутником знизу, а в другому — з вирізаним трикутником праворуч.

Визначення гарячих областей є дуже клопіткою роботою, оскільки доводиться задавати координати кожної вузлової точки. На реальному малюнку це неможливо зробити точно, тому за допомогою графічних редакторів знаходять усі координати, а потім переносять

їх у HTML-документ.