

Лекція 3. МОВА РОЗМІТКИ ГІПЕРТЕКСТУ (HTML)

Веб-сторінки представляють собою каскадне вкладення мультимедійних елементів. Одна веб-сторінка з іншою ділиться текстом, зображеннями, гаджетом, картами, відео, звуковими та мультимедійними ефектами за простого натискання користувача на будь-якому ноутбучі або смартфоні. Веб-сайти є збірками пов'язаних веб-сторінок. Веб-сторінки відображаються веб-браузерами, завдання яких полягає в локалізуванні та відображенні веб-інформації.

Популярним веб-браузером за останні два десятиліття став Microsoft Internet Explorer. Ранні попередники Internet Explorer та інноваційні конкуренти – браузери розробки National Supercomputing Center – Mosaic і Netscape Navigator, на сьогодні не мають всі загального використання. Проте деякі недоліки у розвитку браузера породили веб-оглядачі, які є основними конкурентами Internet Explorer, включаючи Mozilla Firefox, Apple's Safari та Google's Chrome. Інші цікаві ніші браузерів включають Opera, Avant і Flock. На рисунку 3.1 можна знайти піктограми браузерів, які використовуються найчастіше.



Рисунок 3.1 – Піктограми браузерів

HTML та інші інструменти

Визначмо спільні риси браузерів. Всі вони читають і розуміють різні мови Інтернету, починаючи з розробки веб-сторінок мовою Hypertext Markup Language або HTML. HTML — це оригінальний інструмент створення веб-

сторінок. Він дозволяє створювати динамічні веб-сторінки. HTML повідомляє веб-браузери про те, як веб-сторінки повинні виглядати на екрані комп'ютера або смартфона. HTML-теги працюють всюди в Інтернеті і відображають веб-сторінки на комп'ютерах Macintosh, Linux і Windows. Вони навіть працюють на стільникових телефонах з підтримкою веб-сайтів, на пристроях з опальним розміром і на веб-телефонах.

Потужні інструменти HTML, такі як Cascading Style Sheets (CSS), створюють зручні способи визначення стилю (вигляд шрифтів, кольорів та інтервалів) на сторінках веб-сайту. Інструментальні доповнення до HTML, такі як Extensible Markup Language (XML) і Extensible Hypertext Markup Language (XHTML), дають нові можливості даному середовищу.

Існує багато інших мов, що використовуються в Інтернеті, таких як Java, мова програмування, яка широко використовується з інтернет-застосунками; Flash – інструмент для створення мультимедіа; JavaScript – Java-подібна скриптова мова, яка використовується для створення мініпрограм, які називаються альтернативними застосунками.

Тим не менш, HTML є об'єднуючою мовою Всесвітньої павутини. HTML створює основу, на якій можуть бути побудовані інші мови програмування. Це початкова точка для будь-якого онлайн-розробника, з якої слід починати.

HTML (Hypertext Markup Language) – це код, який використовується для визначення структури веб-сторінки та її контенту. Наприклад, контент може бути структурований в межах набору абзаців, списку бюлетенних точок, або за допомогою зображень і таблиць даних.

Вигляд коду для розробки веб-сторінки

На рисунках 3.2А та 3.2В показана одна й та ж сама сторінка, переглянута двома різними способами. На рисунку 3.2А показано, як користувач бачить сторінку в Internet Explorer, а на рисунку 3.2В зображено мітки і код, що створюють візуальний дисплей. Всі слова, зображення, елементи управління та кольори, які надані на рисунку 3.2А організовані тегами, які представлені на рисунку 3.2Б. Сторінка взята з веб-сайту, створеного за допомогою сайтів Google, CSS-подібного засобу розробки, що

дозволяє швидко створювати сайти контенту. На сторінку було додано три гаджети Google, включаючи додаток календаря Google.

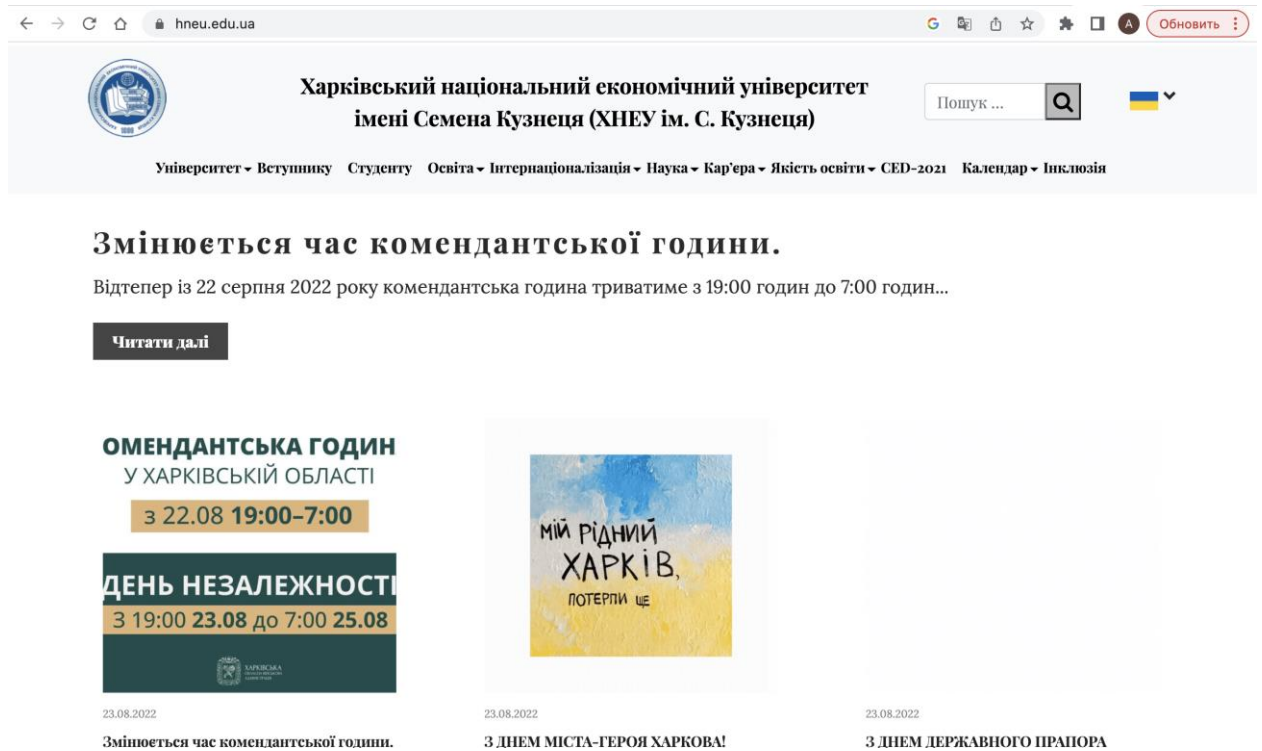


Рисунок 3.2 А – Вигляд веб-сторінки

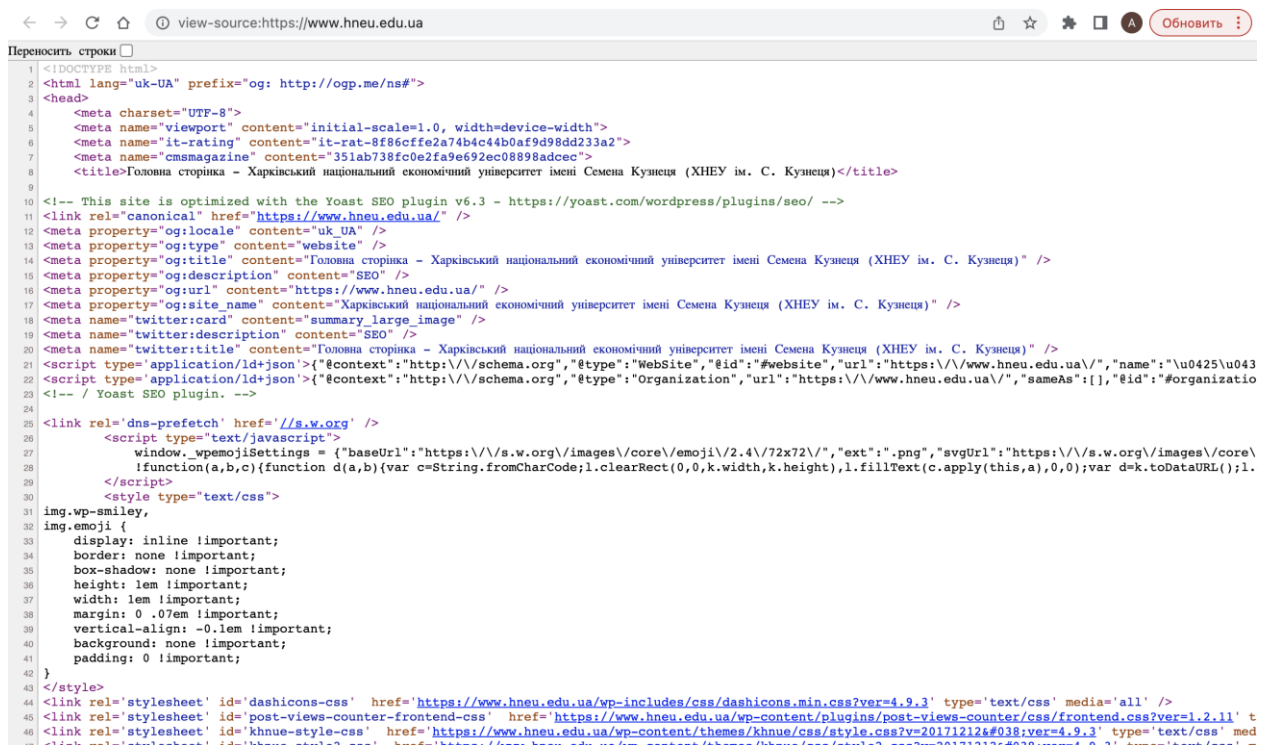
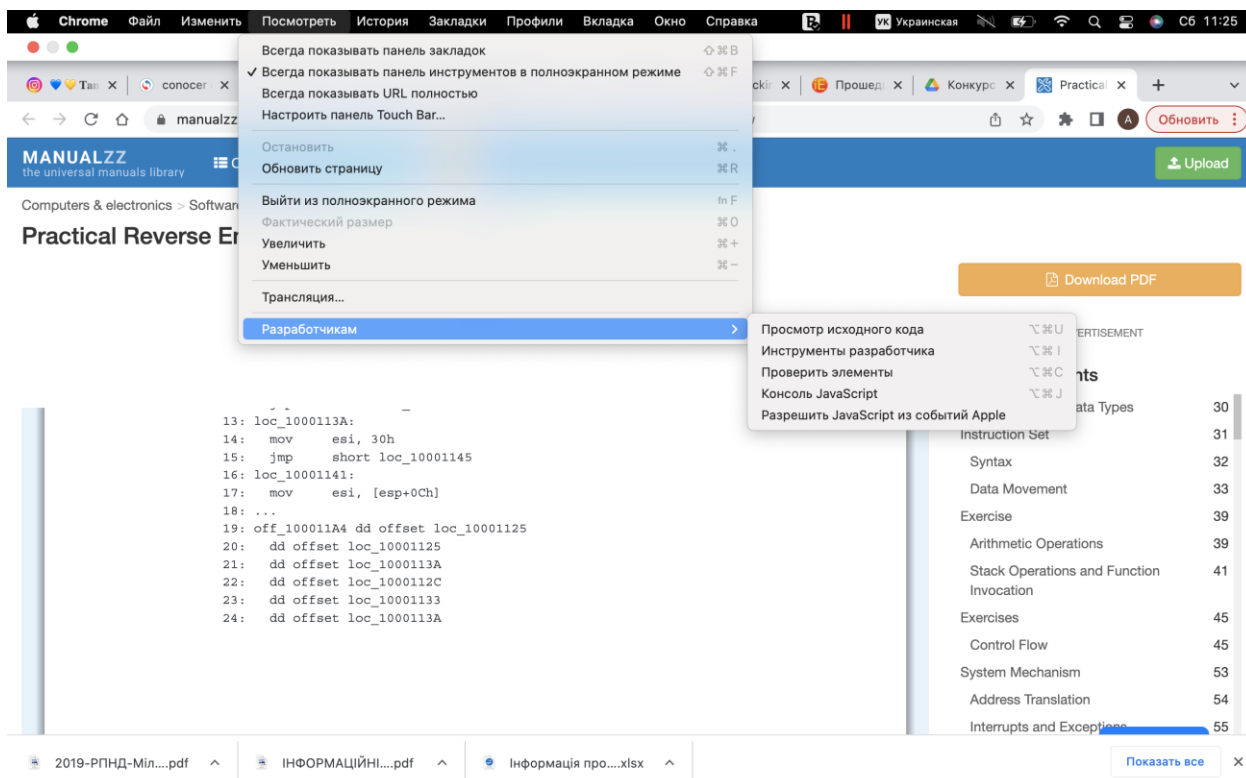


Рисунок 3.2В – HTML код веб-сторінки

Для отримання коду необхідно виконати команду



Структура HTML-тегів

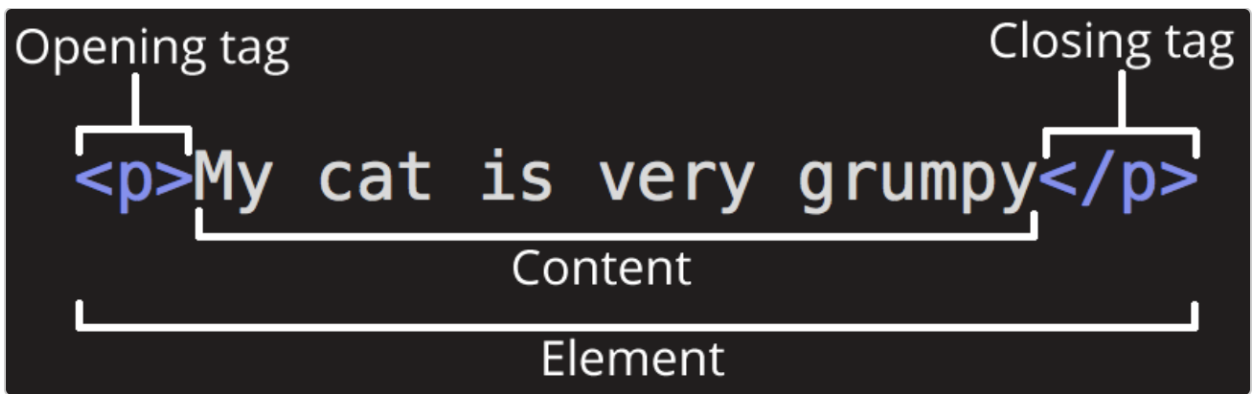
Робота з HTML-тегами є простою, можна швидко вивчити кілька суттєвих міток. Вони зазвичай з'являються парами, розташовані в трикутних дужках $\langle \rangle$.

Щоб більш чітко зрозуміти, як працюють HTML-теги, проаналізуємо наступний приклад. Якщо ви бажаєте розташувати заголовок книги на веб-сторінці, вам слід лише ввести:

`<center>HTML and JavaScript BASICS</center>`

Зауважте, що існує початковий тег, `<center>` і завершальний тег, `</center>`. Єдиною відмінністю між двома тегами є слеш (/), що йде слідом за скобою лівого кута в закритому тезі. Так `<center>` та `</center>` утворюють пару міток, між якими буде текст, який є контентом сторінки.

Наступний рисунок містить схему HTML-елемента.



Основні частини HTML елемента такі:

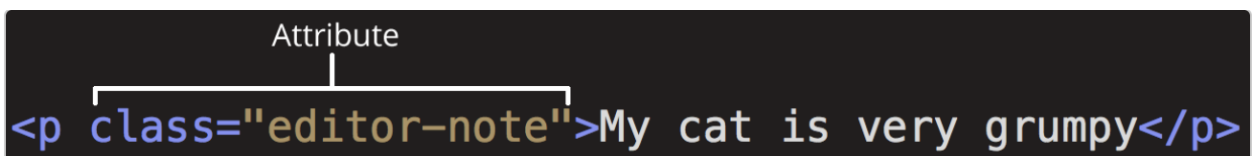
1. Відкриваючий тег: Складається з назви елемента (у цьому випадку `p`), загорнутий у відкриваючий і закриваючий кутові дужки. Це мітка, з якої елемент починає діяти – в цьому випадку, місце з якого починається абзац.

2. Закриваючий тег: має вигляд відкриваючого тега, за винятком того, що він включає в себе похилу риску перед назвою елемента. Це мітка, на якій закінчується елемент – у цьому випадку, місце де закінчується абзац. Не слід забувати додавати завершальний тег – це є однією зі стандартних помилок початківця і може призвести до небажаних результатів.

3. Вміст: контент елемента, який в даному випадку є просто текстом.

4. Елемент: відкриваючий тег, закриваючий тег і вміст разом складають елемент.

Елементи також можуть мати атрибути, які мають вигляд:



Атрибути містять додаткову інформацію про елемент, який не потрібно візуалізувати для користувача. Тут `class` – це ім'я атрибута, а `editor-note` – значення атрибута. Атрибут класу дозволяє надати елементу неунікальний ідентифікатор, який може бути використаний для його призначення (і будь-яких інших елементів з однаковим значенням класу). Елемент за класом може успадкувати інформацію про стилі та інші параметри.

Атрибут завжди повинен мати наступні синтаксичні елементи:

1. Пробіл між ним і назвою елемента (або попереднім атрибутом, якщо елемент вже має один або декілька атрибутів).

2. Назва атрибута, за яким слідує знак рівності.

3. Значення атрибута, розміщене в подвійних лапках.

Зауваження: Прості значення атрибутів, які не містять пробілів ASCII (або будь-якого з символів « » = < >), можна не брати в лапки, але додавання кавичок рекомендується для всіх значень атрибутів, оскільки це робить код більш послідовним та зрозумілим.

Вкладеність елементів

Можна також розмістити елементи всередині інших елементів – це називається вкладеність елементів. Якщо ми хочемо сказати, що наш кіт дуже буркотливий, ми могли б розмістити слово "дуже" в елементі ``, що означає, що слово має бути окремо підкреслено:

```
<p>My cat is <strong>very</strong> grumpy.</p>
```

Однак, потрібно переконатися, що елементи правильно вкладені. У прикладі, наведеному вище, спочатку відкрито елемент `<p>`, потім елемент ``; отже, спочатку слід закрити елемент ``, потім елемент `<p>`. Некоректно:

```
<p>My cat is <strong>very grumpy.</p></strong>
```

Елементи повинні відкриватися і закриватися правильно, щоб вони були явно всередині або зовні один від одного. Якщо вони перекриваються, як показано вище, то ваш веб-браузер спробує зробити найкраще вгадати, що ви намагалися сказати, що може призвести до несподіваних результатів.

Порожні елементи

Деякі елементи не мають вмісту і називаються порожніми елементами. Наприклад елемент ``:

```

```

Цей елемент містить два атрибути, але не існує закриваючого тегу `` і жодного внутрішнього вмісту. Це тому, що елемент зображення не “обгортає” вміст, щоб відформувати його. Метою цього елемента є вбудовування зображення на HTML-сторінці в потрібному місці.

Структура HTML-документа

Розглянемо, як окремі елементи об'єднуються, щоб сформувати цілу сторінку HTML. Перегляньмо наступний код:

```
<!DOCTYPE html>
<html lang="en-US">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width">
    <title>My test page</title>
  </head>
  <body>
    
  </body>
</html>
```

Даний код містить наступні елементи:

`<!DOCTYPE html>` – тип документу. Це обов'язковий початковий елемент. В минулому, коли HTML був новітнім інструментарієм (близько 1991/92), цей елемент виступав як посилання на набір правил, яких повинна була дотримуватися HTML-сторінка, щоб вважатися документом HTML, що означає автоматичну перевірку помилок та застосування інших інструментів налагодження HTML коду.

`<html></html>` – елемент `<html>`. Цей елемент включає весь вміст на всій сторінці. Це кореневий елемент документу. Він також включає атрибут `lang`, який визначає основну мову документа.

`<head></head>` – елемент `<head>`. Цей елемент виконує роль контейнера для всього того, що ви хочете включити на HTML-сторінку, але який не є вмістом, призначеним для візуалізації глядачам сторінки. Цей включає в себе: ключові слова та опис сторінки, який з'являються в результатах пошуку; CSS призначений для моделювання контенту; декларації набору символів та багато іншого.

`<meta charset="utf-8">` – цей елемент встановлює набір символів для вашого документа. Значення цього елемента свідчить про використання таблиці кодів символів UTF-8, яка містить більшість символів з переважної більшості писемних мов. Тепер код може обробляти будь-який текстовий вміст, який ви забажаєте розмістити в ньому. Встановлення цього елемента може допомогти уникнути деяких проблем пізніше.

`<meta name="viewport" content="width=device-width">` – цей елемент зі значенням `"viewport"` гарантує, що сторінка здається на ширину порта перегляду, не допускаючи, щоб мобільні браузері переглядали сторінки ширші за вікно пристрою. Контент сторінки буде масштабуватися згідно з шириною екрану пристрою перегляду (параметр `content="width=device-width"`).

`<title></title>` – це елемент, за допомогою якого можна встановити заголовок сторінки. Такий заголовок з'являється на вкладці переглядача, у який буде завантажено сторінку. Він також використовується для опису сторінки при закладках, якщо сторінка додається до категорії улюблених.

`<body></body>` – це елемент, в якому розміщується весь контент призначений для візуалізації для веб-користувача під час відвідування сторінки. Таким контентом може бути: текст, зображення, відео, ігри, відтворення звукових доріжок тощо.

Step-by-Step 1.1

Now it's your turn. The following steps allow you to open a Web page of your choosing. Viewing the page behind the page is as easy as selecting View Source or the similar command, such as Page Source, from the View menu in your browser (see **Figures 1-3A** and **1-3B**).

1. Open your Web browser by clicking its icon on the Start menu or by double-clicking its icon on the desktop.
2. (Internet Explorer users, read *Extra for Experts* and display the menu bar.) When a page appears, use your mouse to click the **View** menu (Internet Explorer and Firefox) or click the **Page** button on the toolbar (Internet Explorer), as shown in **Figures 1-3A** and **1-3B**.

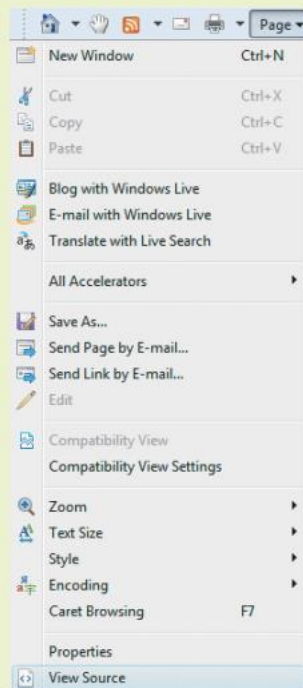


FIGURE 1-3A

The Page (Alternatively: View), View Source command in Internet Explorer

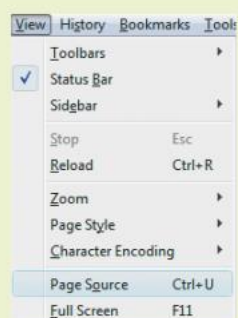


FIGURE 1-3B

The View, Page Source command in Mozilla Firefox

3. If you are using Internet Explorer, click the **Page** button on the toolbar, and then click **View Source** (see **Figure 1–3A**), or if you have unlocked the menu bar, click **View** on the menu bar, and then click **Source**.
If you are using Firefox, click **View** on the menu bar and then click **Page Source**. (See **Figure 1–3B**.)

A new window opens displaying the page beneath the page.

4. Examine the tags that appear on the page beneath the Web page. The tags will look similar to the tags you saw in **Figure 1–2B**.
5. Use your browser to view three or four other Web pages and view the source page for each of these pages. List seven tags that you find repeatedly in these pages. Guess and record what you think they do in **Table 1–1**.

TABLE 1–1 Common tags

NUMBER	TAG	EFFECT IT CREATES ON THE WEB PAGE
Sample	<center></center>	Centers text on a Web page
1		
2		
3		
4		
5		
6		
7		

Загальновживана термінологія

Перед початком введення тегів потрібно визначити терміни, які використовуються для опису сторінок, створених за допомогою тегів HTML. Правда в тому, що ці терміни використовуються настільки часто, що більшість людей абсолютно не знають, що є незначні, але важливі відмінності в їх значеннях.

■ **Web-сторінка (Web page):** у літературі також згадується як веб-документ або HTML-документ, включає в себе будь-яку сторінку, створену в HTML, яка може бути розміщена в інтернет.

■ **Домашня сторінка (Home page):** головна або первинна веб-сторінка для корпорації, організації, або індивідуального сайту. Персональна домашня сторінка - е перша сторінка, яку ви бачите під час запуску веб-браузера. Після

натискання піктограми Home у браузері одразу ж буде відображено початкова домашня сторінка.

- Сторінка привітання (Welcome page): спеціальна сторінка для нових відвідувачів веб-сайту.

- Лендінг сторінка (Landing page): цільова «вітальна» сторінка, що використовується веб-рекламодавцями. Це сторінка, яка з'являється після того, як хтось натискає на веб-оголошення.

- Веб-сайт (Web site): може включати в себе колекцію багатьох взаємопов'язаних веб-сторінок, організованих конкретною компанією, організацією, коледжем або університетом, урядовою агенцією, або окремою особою. Веб-сайти зберігаються на веб-серверах. CSS часто використовуються для створення стандартного вигляду і відчуття для сайту.

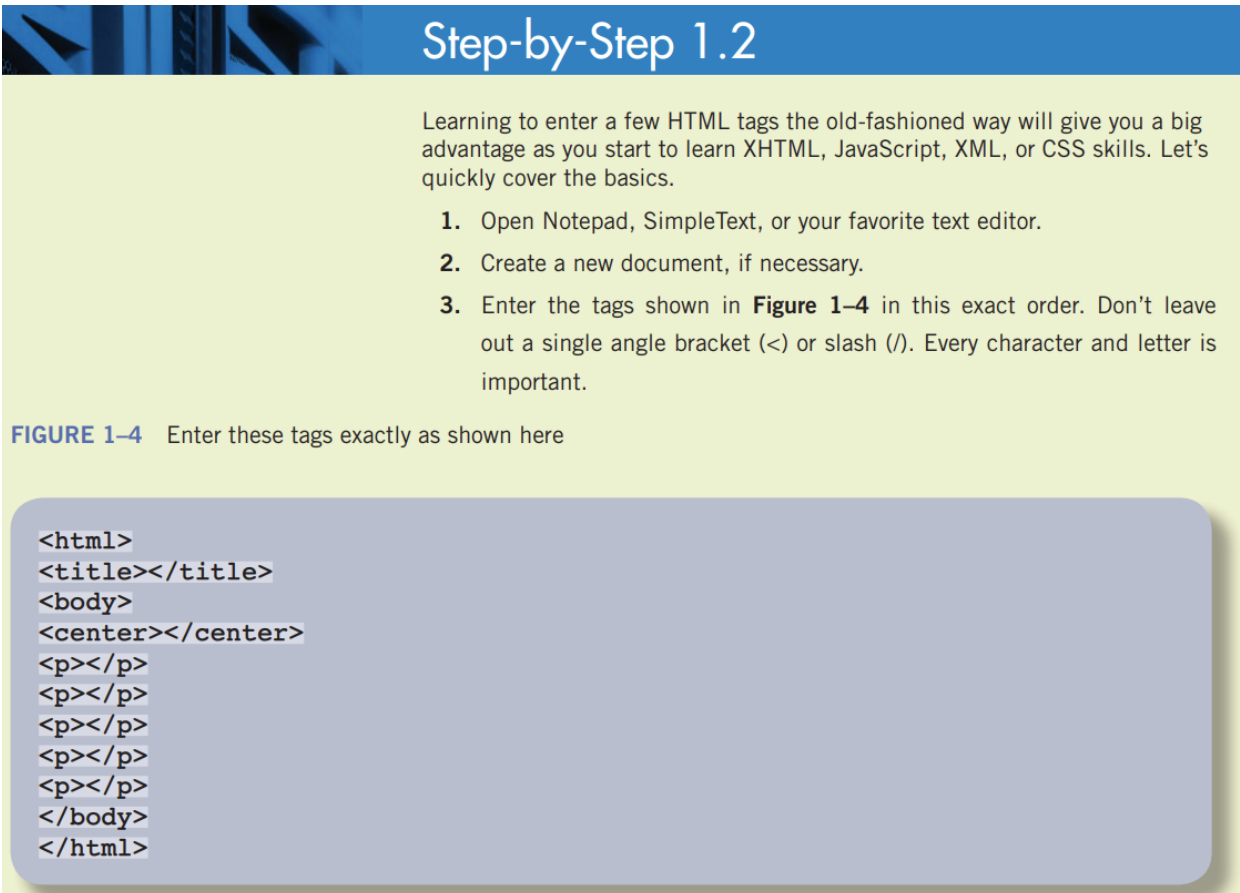
Існує багато способів створення HTML кодів. Для створення складних веб-сторінок можна використовувати спеціалізоване програмне забезпечення, наприклад, Expression Web 2 від Microsoft або Dreamweaver від Adobe. Також можна перейти в Інтернет і використовувати безкоштовні інструменти, такі як сайти від Google (sites.google.com). Ці програми допомагають впорядковувати HTML-сторінки, вводити текст, переміщувати елементи навколо і створювати чудові ефекти веб-сторінок, не вводячи жодного тега HTML. Можна створювати веб-сторінки на основі документів, створених за допомогою застосунків, наприклад Microsoft Office Word, Excel або PowerPoint і Google Docs (docs.google.com). По суті, Google Docs створює веб-документи, електронні таблиці та презентації, які всі знаходяться в Інтернеті в хмарі Інтернету і можуть бути розділені по всьому світу як веб-сторінки.

Є кілька причин, чому важливо вивчити HTML:

- 1 Працюючи з мітками, розвивається більш глибоке розуміння того, як насправді працює HTML.
- 2 Можливість усунути неполадки веб-сторінок, якщо виникне незначна кількість помилок.
- 3 Можливість переглянути інші сторінки та дізнатися, як інші досягли певних ефектів.
- 4 Краще зрозуміння структури файлів та папок, розташованих на веб-серверах.

5 Розуміння того, як HTML та інші інструменти, такі як XHTML, JavaScript, CSS, і XML працюють разом.

Для створення як HTML-тегів, так і JavaScript-коду можна використовувати будь-який текстовий редактор. Це одна з причин, чому HTML і JavaScript настільки популярні. Для створення захоплюючих веб-сторінок не потрібні жодні спеціалізовані програмні засоби. Рекомендація – використовувати найпростіші, основні доступні інструменти. У Windows можна використовувати Блокнот. На Macintosh можна використовувати SimpleText. Ці текстові редактори легко використовуються і доступні майже на кожному комп'ютері.



Step-by-Step 1.2

Learning to enter a few HTML tags the old-fashioned way will give you a big advantage as you start to learn XHTML, JavaScript, XML, or CSS skills. Let's quickly cover the basics.

1. Open Notepad, SimpleText, or your favorite text editor.
2. Create a new document, if necessary.
3. Enter the tags shown in **Figure 1-4** in this exact order. Don't leave out a single angle bracket (<) or slash (/). Every character and letter is important.

FIGURE 1-4 Enter these tags exactly as shown here

```
<html>
<title></title>
<body>
<center></center>
<p></p>
<p></p>
<p></p>
<p></p>
<p></p>
</body>
</html>
```

4. The tags you just entered are called the basic tags. They include a standard set of tags that appear in most Web pages. But your page will look very sad and boring without a little text. Enter text between the tags, as shown in **Figure 1–5**. (Notice that the new text to be entered is shown in bold as a visual cue to help you as you work to create the HTML files in this book.)

FIGURE 1–5 Enter the text between the tags exactly as shown here

```
<html>
<title>HTML and JavaScript</title>
<body>
<center>Creating HTML and JavaScript</center>
<p>Learning to create HTML tags can help you in many ways: </p>
<p>You will develop a deeper understanding of how HTML really works. </p>
<p>You will be able to troubleshoot Web pages when errors occur. </p>
<p>You will be able to view other pages and learn how certain effects
were created. </p>
<p>You will understand how HTML and JavaScript work together. </p>
</body>
</html>
```

5. Leave your text editor open and go on to Step-by-Step 1.3 where you will learn how to save HTML files.

WARNING

In the past, HTML wasn't case sensitive. You could use uppercase <TAGS>, lowercase <tags>, or mixed <TagS>. With HTML 4.01 and XHTML 1.0 standards, new and stricter methods are now being implemented. It is now considered good form to only use lowercase <tags>.

Збереження і перегляд HTML сторінки

HTML-документи – текстові файли. Це означає, що вони заберігаються найпростішим чином. Здебільшого, текстові файли зберігають лише символи, які ви бачите на клавіатурі. Всі складні команди обробки слів приховуються, залишаючи лише символи. Збереження як тексту дозволяє HTML-файлам швидко переміщатися по мережі. Однак проблема з текстовими файлами полягає в тому, що більшість людей не вміють їх зберігати. Перед тим, як зберегти такий файл, є кілька речей, які потрібно знати. Для ідентифікації типу даних, які містить файл (фотографія, музика, документ), комп'ютери додають розширення файлів до назв файлів. Іноді на комп'ютері можна побачити ці розширення, а іноді й ні. Залежно від налаштувань операційної системи

комп'ютера, розширення можуть бути видимими або не бути видимими, але програмне забезпечення на комп'ютері знає, які типи файлів він може відкрити. Розширення ідентифікують типи файлів. У таблиці 1–2 перелічено деякі з цих загальних розширень файлів

TABLE 1–2 Common File Extensions

EXTENSION	FILE TYPE
.doc and .docx	Microsoft Word 2003 or 2007 document
.rtf	Microsoft's Rich Text Format
.txt	Text file
.html	HTML file on some computer systems
.htm	HTML file on some computer systems
.jpg or .jpeg	A popular image format used on the Web

HTML-файли — текстові файли з розширенням .html або .htm. Хоча формат, потрібний для HTML, є текстовим, закінчення або розширення має бути .html (або .htm, якщо використовується деяке програмне забезпечення на основі Windows). Розширення .html або .htm сигналізують веб-браузеру, що це текстовий файл HTML. Розширення .html схоже на те, щоб поставити знак, який каже: "Гей, браузер, прочитайте мене. Я є HTML-документом».

Step-by-Step 1.3

The following steps show you how to save an HTML/text file in Notepad. Saving in other text editors, such as SimpleText, is very similar so you'll be able to use these directions to save your file.

1. Click **File** on the menu bar, and then click **Save As**.
2. Navigate to the folder where you save your documents, and then click the **New Folder** button in the Save As dialog box to create a new folder in which to save your HTML and JavaScript work.

3. Name the folder as you want or as directed by your instructor, and then open the folder into which you want to save your files.
4. Click the **Save as type** arrow, click **All Files (*.*)**, and then type **one.html** to name your file, as shown in **Figure 1–6**. Then click **Save**. If everything saves properly, go on to Step 5. Check with your instructor to make sure you save your file properly.

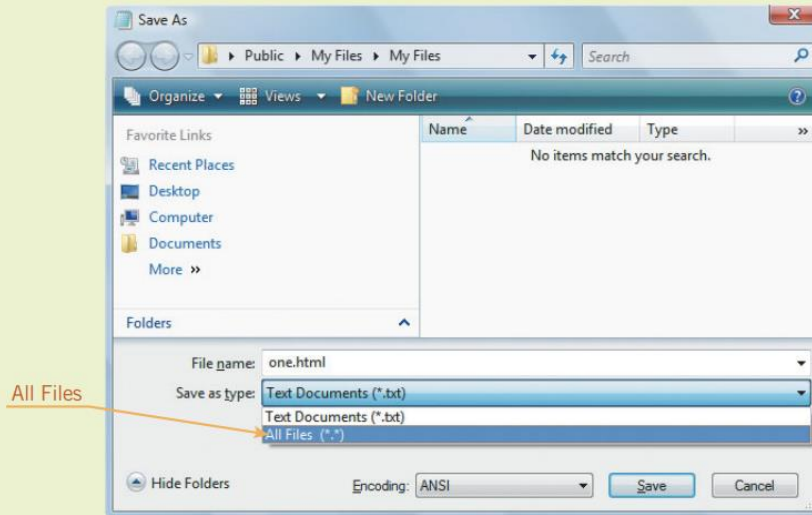


FIGURE 1–6
Name a text file with an .html extension

5. Viewing your HTML page in a Web browser is easy. These steps show you how to do this in Internet Explorer and Firefox.

If you are an **Internet Explorer** user, perform the following steps (the steps for Firefox are listed after these steps):

- a. Start the Internet Explorer Web browser.
- b. Click **Tools** on the toolbar, point to **Toolbars**, and then click **Menu Bar**, as shown in **Figure 1–7**.

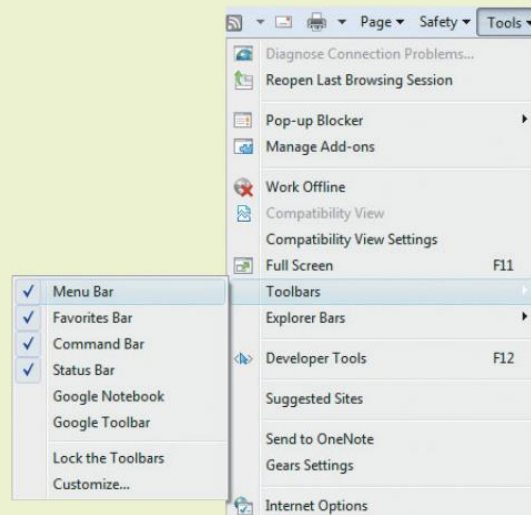
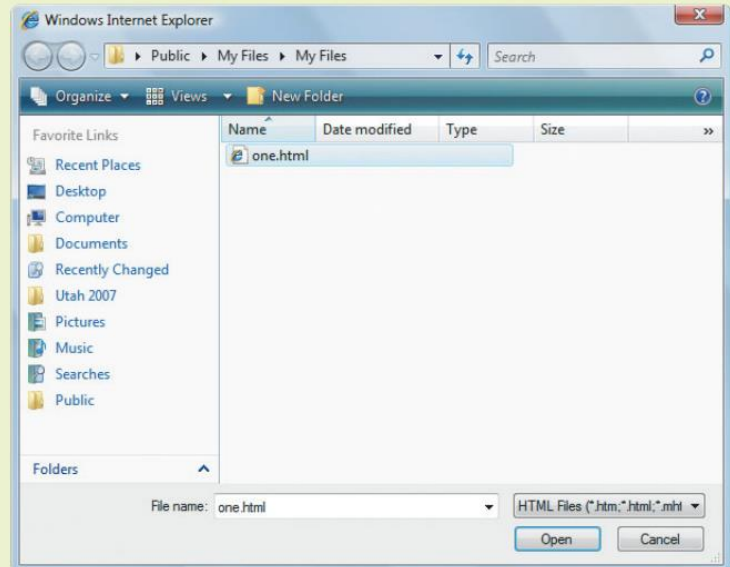


FIGURE 1–7
Open the menu bar to make it easier to use

- c. Click **File** on the menu bar, and then click **Open**.
- d. Click the **Browse** button, and then navigate to the folder where you saved your one.html file.
- e. Click **one.html** in the Open text box, and then click **Open**, as shown in **Figure 1–8**.
- f. In the Open dialog box, click **OK**.

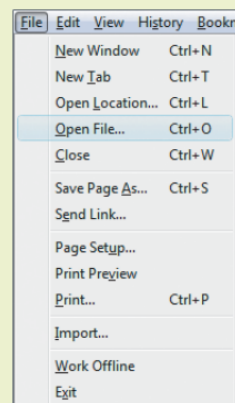
FIGURE 1–8
Find your file in IE



If you are a **Firefox** user, perform the following steps:

- a. Start the Firefox Web browser.
- b. Click **File** on the menu bar, and then click **Open File**, as shown in **Figure 1–9**.

FIGURE 1–9
Open File command in Firefox



- c. Click the **Files of type** arrow, and then click **All Files**, if it is not already selected.
- d. Browse to the folder where you saved the one.html file.
- e. Click the **one.html** file, and then click **Open**, as shown in **Figure 1–10**.

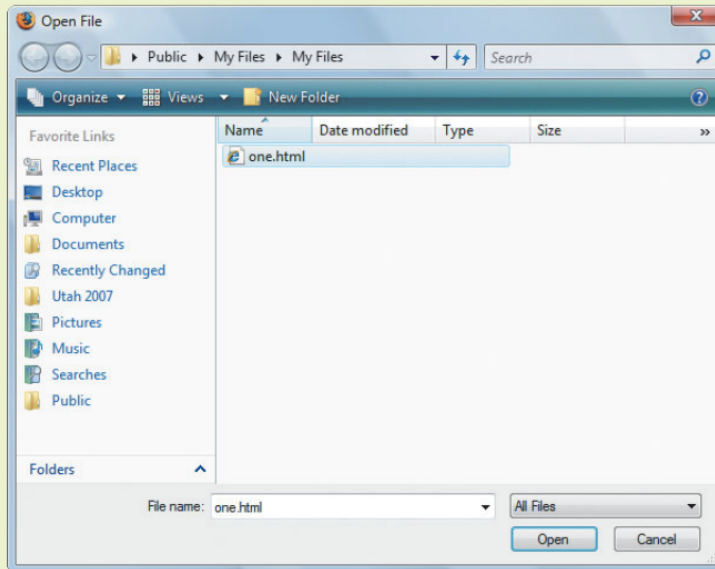


FIGURE 1–10
Find your file in Firefox

6. View your file. It should look like **Figure 1–11**.

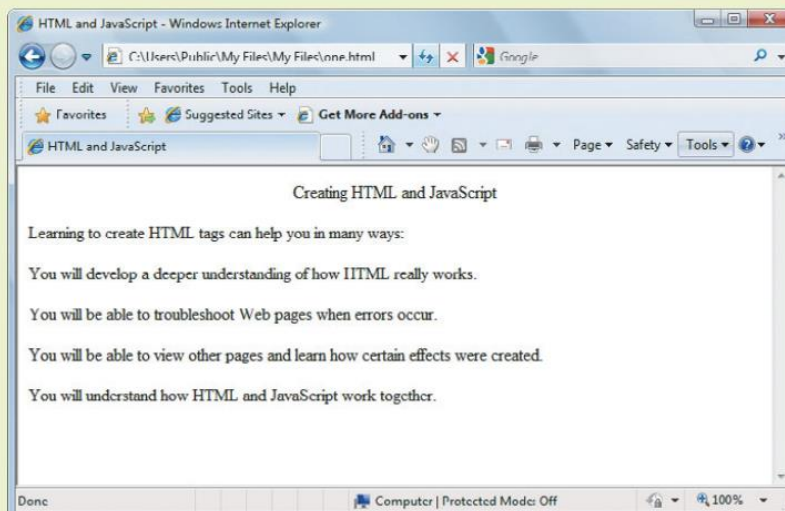


FIGURE 1–11
Congratulations! Your Web page probably looks like this sample

7. How does your Web page look? Make any corrections necessary, save the text file again, return to your browser, and then click the browser **Reload** or **Refresh** button to see the changes you have made.

Використання заголовків

Більшість друкованих документів використовують рубрики, щоб допомогти читачеві знайти важливі частини тексту. Головний заголовок документу зазвичай розташовано зверху і в центрі сторінки. Як правило, від

виокремлений жирним накресленням або все великими літерами. Підзаголовки або заголовки підрозділів та пунктів зазвичай розташовані як основний текст і виокремлюються жирним накресленням.

HTML надає шість стандартних заголовків, або розмірів заголовків, з яких можна вибирати. Теги заголовків забезпечують легкий спосіб встановлення розміру потрібного тексту, таким чином, щоб читач міг чітко переглядати заголовки.

Теги заголовків легко запам'ятати. Вони використовують літеру H з числом від 1 до 6 для позначення рівня заголовка. Номери заголовків вказують на рівень важливості для позначених заголовків, а 1 – найвизначніші, а 6 – найменш помітні:

```
<h1></h1>  
<h2></h2>  
<h3></h3>  
<h4></h4>  
<h5></h5>  
<h6></h6>
```

Будь-що всередині заголовків буде зроблено більшим або меншим, залежно від номера в тезі. Наприклад

```
<h1>Very Big</h1>  
<h3>In the Middle</h3>  
<h6>Very Small</h6>
```

Step-by-Step 1.4

In this step-by-step, you open the HTML file you have been working on and add the heading or title tags.

1. Open your text editor.
2. Open your **one.html** or **one.htm** file, if it is not still open.

If you are using Notepad, click **File** on the menu bar, click **Open**, click the **Files of type** arrow, and then click **All Files**, as shown in **Figure 1–12**.

Otherwise, you will not be able to view your .html or .htm file.

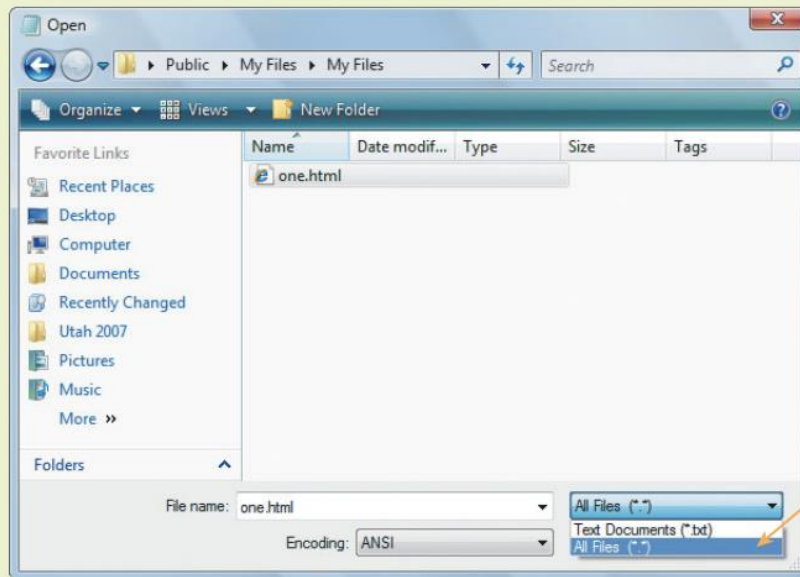


FIGURE 1–12

Open a file in Notepad

3. Enter the heading tags shown in bold in **Figure 1–13**.

FIGURE 1–13 Add the heading tags

```
<html>
<title>HTML and JavaScript</title>
<body>
<center><b>h1</b>Creating HTML and JavaScript</h1></center>
<p><b>h2</b>Learning to create HTML tags can help you in many ways:</h2></p>
<p><b>h3</b>You will develop a deeper understanding of how HTML really
works.</h3>
</p>
<p><b>h4</b>You will be able to troubleshoot Web pages when errors occur.</h4>
</p>
<p><b>h5</b>You will be able to view other pages and learn how certain effects
were created.</h5></p>
<p><b>h6</b>You will understand how HTML and JavaScript work together.</h6></p>
</body>
</html>
```

4. Save your new HTML file as **two.html** or **two.htm** to the same folder that you saved the previous file.
5. Start your Web browser, and then open your **two.html** or **two.htm** file to view it in the browser window. It should look like **Figure 1–14**.

FIGURE 1–14

Various size headings in a Web page

WARNING

Users might see a “Pop-Up Blocked” message or some other error message that might block the opening of the page. Simply click the option to allow the content to be viewed. If you need help, ask your instructor or read the Help files for your specific browser.



Heading tags really change the look of a page. In our example in **Figure 1–14**, there should be only three levels of information:

- `<h1></h1>` The title at the top
- `<h2></h2>` The introductory line followed by a colon (:)
- `<h3></h3>` The list of the reasons to learn HTML tags

6. Return to your text editor and change the heading tags. Use no more than three `<h></h>` tags for this Web page. Think about your tag choices, and then make your Web page comfortable to read, emphasizing the three levels this document dictates. Resave your file to make your changes become effective.

Використання нумерованих та маркірованих списків

Одним з найпотужніших способів організації інформації на веб-сторінці є використання списків. Є кілька видів списків, основні з них це маркірований (не впорядкований) та нумерований (впорядкований):

- `` Unordered (or bulleted) lists
- `` Ordered (or numbered) lists

Step-by-Step 1.5

The unordered lists tags `` create bulleted lists. Start your list with the opening `` unordered lists tag, mark the items to be listed with the list `` tags, and place an `` tag at the end of your list. Try it!

1. Open your **two.html** or **two.htm** file for text editing.
2. Enter the `` tags at the start and at the end of the list to create an unordered list, as shown in **Figure 1-15**.
3. Replace the `<p>` and `</p>` tags with `` and `` tags for each sentence in the list, as shown in **Figure 1-15**.

FIGURE 1-15 Enter the unordered list tags

```
<html>
<title>HTML and JavaScript</title>
<body>
<center><h1>Creating HTML and JavaScript</h1></center>
<p><h2>Learning to create HTML tags can help you in many ways:</h2></p>

<ul>
<li><h3>You will develop a deeper understanding of how HTML really works.
</h3></li>
<li><h3>You will be able to troubleshoot Web pages when errors occur.
</h3></li>
<li><h3>You will be able to view other pages and learn how certain
effects were created.</h3></li>
<li><h3>You will understand how HTML and JavaScript work together.
</h3></li>
</ul>

</body>
</html>
```

4. Save your file as **three.html** or **three.htm** to the folder where you saved the other HTML files for this lesson.

5. View your `three.html` or `three.htm` Web page in a browser to see how it looks. It should be similar to **Figure 1-16**.



FIGURE 1-16
An unordered list

6. Open your `three.html` or `three.htm` file for text editing, if it is not already open.
7. Change the pair of `` tags to `` tags to change your list from an unordered list to an ordered list, as shown in **Figure 1-17**. No other changes are necessary.

WARNING

Don't use a zero; use the letter O for ordered.

FIGURE 1-17 Enter the ordered lists tags

```
<html>
<title>HTML and JavaScript</title>
<body>
<center><h1>Creating HTML and JavaScript</h1></center>
<p><h2>Learning to create HTML tags can help you in many ways: </h2></p>

<ol>
<li><h3>You will develop a deeper understanding of how HTML really works.
</h3></li>
<li><h3>You will be able to troubleshoot Web pages when errors occur.
</h3></li>
<li><h3>You will be able to view other pages and learn how certain
effects were created.
```

```
</h3></li>
<li><h3>You will understand how HTML and JavaScript work together.
</h3></li>
</ol>

</body>
</html>
```

8. Save your file as **four.html** or **four.htm**.
9. View your four.html or four.htm page in a browser to see how it looks. It should look similar to **Figure 1–18**.

FIGURE 1–18
An ordered or numbered list



10. Open your **four.html** or **four.htm** file for text editing, if it is not already open.
11. Add two pairs of **** tags in the middle of the list, as shown in **Figure 1–19**.

FIGURE 1–19 Enter additional unordered lists tags

```
<html>
<title>HTML and JavaScript</title>
<body>
<center><h1>Creating HTML and JavaScript</h1></center>
<p><h2>Learning to create HTML tags can help you in many ways: </h2></p>

<ol>
<li><h3>You will develop a deeper understanding of how HTML really works.
</h3></li>

<ul>
<li><h3>You will be able to troubleshoot Web pages when errors occur. </
h3></li>
<ul>

<li><h3>You will be able to view other pages and learn how certain
effects were created.
</h3></li>
</ul>
</ul>

<li><h3>You will understand how HTML and JavaScript work together.
</h3></li>
</ol>
</body>
</html>
```


12. Save your file as **five.html** or **five.htm**.
13. View your five.html or five.htm Web page in a browser to see how it looks. It should look similar to **Figure 1–20**.

FIGURE 1–20
Embedded and indented lists



14. Exit your browser and text editor and shut down your computer if you are finished for today. Otherwise, continue to the Summary section.

Використання таблиць