

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
МІНІСТЕРСТВО КУЛЬТУРИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
КУЛЬТУРИ І МИСТЕЦТВ

І. Л. Бородкіна, Г. О. Бородкін

**WEB-ТЕХНОЛОГІЇ ТА WEB-ДИЗАЙН:
ЗАСТОСУВАННЯ МОВИ HTML ДЛЯ СТВОРЕННЯ
ЕЛЕКТРОННИХ РЕСУРСІВ**

Навчальний посібник
для студентів вищих навчальних закладів

Київ
Видавництво Ліра-К
2020

УДК 004.439 (075.8)
Б833

*Рекомендовано до друку Головною Вченою Радою
Київського національного університету культури і мистецтв
(Протокол № 12 від 8 грудня 2017 р.)*

Рецензенти:

Прокудін Г. С. завідувач кафедри міжнародних перевезень та митного контролю Національного транспортного університету МОН України, Академік Української академії економічної кібернетики, доктор технічних наук, професор;

Бушма О. В. професор кафедри інформаційних технологій та математичних дисциплін Факультету інформаційних технологій та управління Київського університету ім. Бориса Грінченка, доктор технічних наук, професор.

Бородкіна І. Л., Бородкін Г. О.

Б833 Web-технології та Web-дизайн: застосування мови HTML для створення електронних ресурсів : навч. посіб. / І. Л. Бородкіна, Г. О. Бородкін.– Київ: Видавництво Ліра-К, 2020. – 212 с.

ISBN 978-617-7844-14-2

Навчальний посібник призначений для набуття теоретичних знань і практичних навичок створення гіпертекстових інформаційних продуктів із застосуванням мови гіпертекстової розмітки HTML.

В навчальному посібнику надано основний теоретичний матеріал, який стосується різних аспектів використання тегів мови HTML при створенні інформаційних ресурсів, наведено велику кількість прикладів використання основних структурних елементів мови HTML. Для закріплення отриманих знань та набуття необхідних практичних навичок студентам пропонуються завдання для самостійної роботи, варіанти практичних завдань для проведення контрольної роботи, приклади питань для проведення тестового контролю знань та довідкові матеріали, необхідні студентам під час виконання самостійних завдань.

Навчальний посібник розрахований на студентів вищих навчальних закладів, які вивчають технології створення електронних інформаційних ресурсів. Посібник може стати основою для подальшого поглибленого вивчення дисциплін, пов'язаних зі створенням інформаційних ресурсів для мережі Інтернет.

УДК 004.439 (075.8)

ISBN 978-617-7844-14-2

© Бородкіна І. Л., Бородкін Г. О., 2020
© Видавництво-Ліра-К, 2020

ЗМІСТ

| | |
|--|----|
| ВСТУП | 6 |
| РОЗДІЛ 1. ЗАГАЛЬНА ХАРАКТЕРИСТИКА МОВИ HTML | 8 |
| ОСОБЛИВОСТІ ЗАСТОСУВАННЯ МОВИ HTML | 10 |
| <i>РЕГІСТР ЗАПИСУ КОМАНД</i> | 10 |
| <i>ПАРНІСТЬ ТЕГІВ</i> | 10 |
| <i>ESCAPE-ПОСЛІДОВНОСТІ</i> | 11 |
| <i>ПЕРЕВЕДЕННЯ РЯДКА, ПРОПУСКИ, ТАБУЛЯЦІЯ</i> | 13 |
| <i>КОМЕНТАРІ</i> | 14 |
| <i>ЗАГАЛЬНІ ПРАВИЛА ІНТЕРПРЕТАЦІЇ ТЕГІВ БРАУЗЕРАМИ</i> | 14 |
| ТЕХНОЛОГІЯ СТВОРЕННЯ HTML-ДОКУМЕНТІВ | 15 |
| ЗАВДАННЯ | 16 |
| РОЗДІЛ 2. СТРУКТУРА HTML-ДОКУМЕНТІВ | 21 |
| ЗАГАЛЬНА СТРУКТУРА ДОКУМЕНТА | 21 |
| РОЗДІЛ ЗАГОЛОВКА ДОКУМЕНТА HEAD | 22 |
| <i>ЗОВНІШНІЙ ЗАГОЛОВОК ДОКУМЕНТА HEAD</i> | 24 |
| <i>МЕТА-ІНСТРУКЦІЇ В ЗАГОЛОВКУ HTML-ДОКУМЕНТА</i> | 24 |
| РОЗДІЛ ТІЛА ДОКУМЕНТА BODY | 26 |
| АТРИБУТИ ТЕГА <BODY> | 28 |
| <i>АТРИБУТИ ДЛЯ ОФОРМЛЕННЯ ВИГЛЯДУ ДОКУМЕНТА</i> | 28 |
| <i>АТРИБУТИ ДЛЯ ОФОРМЛЕННЯ ВИГЛЯДУ ГІПЕРПОСИЛАНЬ В ДОКУМЕНТІ</i> | 33 |
| <i>АТРИБУТИ ДЛЯ ВСТАНОВЛЕННЯ ПОЛІВ ДОКУМЕНТА</i> | 33 |
| <i>АТРИБУТИ ДЛЯ УПРАВЛІННЯ ПЕРЕГЛЯДОМ ДОКУМЕНТА</i> | 34 |
| ЗАВДАННЯ | 34 |
| РОЗДІЛ 3. РОБОТА З ТЕКСТОМ ЗАСОБАМИ МОВИ HTML | 37 |
| ЛОГІЧНА І ФІЗИЧНА РОЗМІТКА | 37 |
| <i>ФІЗИЧНІ СТИЛІ РОЗМІТКИ ТЕКСТУ</i> | 38 |
| <i>ЛОГІЧНА РОЗМІТКА ТЕКСТУ</i> | 40 |
| ТЕКСТОВІ БЛОКИ | 43 |
| ШРИФТИ В HTML-ДОКУМЕНТІ | 45 |
| ЗАГОЛОВКИ В HTML-ДОКУМЕНТІ | 47 |
| РЯДОК, ЩО РУХАЄТЬСЯ | 49 |
| ЗАВДАННЯ | 50 |
| РОЗДІЛ 4. СПИСКИ | 52 |
| МАРКІРОВАННИЙ СПИСОК | 52 |
| НУМЕРОВАННИЙ СПИСОК | 54 |
| СПИСОК ВИЗНАЧЕНЬ | 58 |
| ВКЛАДЕНІ СПИСКИ | 59 |
| ЗАВДАННЯ | 61 |
| РОЗДІЛ 5. ГІПЕРТЕКСТОВІ ПОСИЛАННЯ | 65 |
| ОРГАНІЗАЦІЯ ПОСИЛАНЬ | 65 |
| ПОСИЛАННЯ НА ІНШІ ДОКУМЕНТИ САЙТУ | 69 |
| ПОСИЛАННЯ НА ДОКУМЕНТИ ІНШИХ САЙТІВ (ЗОВНІШНІ ПОСИЛАННЯ) | 69 |
| ПОСИЛАННЯ НА РОЗДІЛИ В МЕЖАХ ОДНОГО ДОКУМЕНТА | 70 |

| | |
|---|-----|
| Посилання на адресу електронної пошти | 73 |
| Посилання на файлові об'єкти | 74 |
| Завдання | 74 |
| Розділ 6. Графіка в HTML-документах | 75 |
| Загальна характеристика графічних об'єктів в HTML-документах | 75 |
| Вбудовування зображень в HTML-документи | 76 |
| <i>Альтернативний текст</i> | 77 |
| <i>Вирівнювання зображень</i> | 78 |
| <i>Визначення розмірів зображення</i> | 81 |
| <i>Рамки навколо зображень</i> | 82 |
| <i>Відділення зображення від тексту</i> | 82 |
| Способи використання графічних об'єктів в HTML-документах | 83 |
| <i>Горизонтальна лінія</i> | 84 |
| <i>Фонові зображення</i> | 84 |
| <i>Зображення як графічний маркер списку</i> | 85 |
| <i>Графічне зображення як елемент гіперпосилання</i> | 87 |
| <i>Використання графічних зображень для побудови карти сайту</i> | 89 |
| Завдання | 93 |
| Розділ 7. Таблиці в HTML-документах | 94 |
| Організація таблиць в HTML-документах | 94 |
| Управління виглядом таблиці | 97 |
| Успадкування параметрів вирівнювання в таблицях | 10 |
| Управління границями таблиці | 104 |
| Відстань між комірками таблиці та відстань між вмістом комірки і її рамкою | 106 |
| Об'єднання рядків та стовпчиків | 109 |
| Створення в таблицях заголовків та об'єднання стовпчиків та рядків | 111 |
| Визначення ширини таблиці | 116 |
| Вкладені таблиці. Центрування таблиці на сторінці | 118 |
| Завдання | 120 |
| Розділ 8. Альтернативне використання таблиць | 124 |
| Застосування таблиць для форматування вигляду сторінок сайту | 124 |
| Верстання тексту в декілька колонок | 128 |
| Зовнішній вигляд таблиць | 130 |
| Переваги макетування сайту з використанням таблиць | 133 |
| Недолки сайту, змакетованого з використанням таблиць | 134 |
| Завдання | 135 |
| Розділ 9. Фреймові структури | 137 |
| Переваги використання фреймів | 137 |
| Недолки використання фреймів | 138 |
| Синтаксис фреймів | 138 |
| Порядок створення сторінки з фреймами | 140 |
| Посилання у фреймах | 142 |
| Плаваючі фрейми | 143 |
| Завдання | 145 |

| | |
|--|-----|
| Розділ 10. МЕТОДИЧНІ РЕКОМЕНДАЦІЇ ДО ВИКОНАННЯ ПІДСУМКОВОЇ РОБОТИ "СТВОРЕННЯ WEB-САЙТУ" | 146 |
| ЗБИРАННЯ ТА АНАЛІЗ ІНФОРМАЦІЇ | 146 |
| РОЗРОБКА СТРУКТУРИ САЙТУ | 147 |
| <i>ВНУТРІШНЯ СТРУКТУРА САЙТУ</i> | 147 |
| <i>ЗОВНІШНЯ СТРУКТУРА САЙТУ</i> | 148 |
| ОФОРМЛЕННЯ (ДИЗАЙН) САЙТУ | 149 |
| ВЕРСТАННЯ САЙТУ | 150 |
| ТЕСТУВАННЯ І ДООПРАЦЮВАННЯ САЙТУ | 150 |
| ПРОСУВАННЯ І РЕКЛАМА САЙТУ | 151 |
| ПОДАЛЬША ПІДТРИМКА І ОНОВЛЕННЯ САЙТУ | 151 |
| НАЙПОШИРЕНІШІ ПОМИЛКИ В ДИЗАЙНІ САЙТУ | 152 |
| ПОРАДИ СТОСОВНО РОЗРОБКИ HTML-СТОРІНОК | 153 |
| ЗАВДАННЯ | 156 |
| | |
| Розділ 11. ОПТИМІЗАЦІЯ САЙТУ ДЛЯ ПОШУКОВИХ СИСТЕМ | 157 |
| ОПТИМІЗАЦІЯ РОЗДІЛУ HEAD | 157 |
| ТЕГ ЗАГОЛОВКУ СТОРІНКИ TITLE | 157 |
| МЕТА-ТЕГИ | 158 |
| <i>ОПИС МЕТА-ТЕГІВ ГРУПИ NAME</i> | 158 |
| <i>ОПИС МЕТА ТЕГІВ ГРУПИ HTTP-EQUIV</i> | 163 |
| ЗАВДАННЯ | 168 |
| | |
| Додаток 1. ЗАЛКОВІ ТА ЕКЗАМЕНАЦІЙНІ ПИТАННЯ ДЛЯ ПЕРЕВІРКИ ЗНАТЬ ТА НАВИЧОК, ОТРИМАНИХ ПРИ ВИВЧЕННІ МАТЕРІАЛІВ ПОСІБНИКА | 169 |
| Додаток 2. ТЕСТОВІ ПИТАННЯ ДЛЯ ПЕРЕВІРКИ ЗНАТЬ ТА НАВИЧОК, ОТРИМАНИХ ПРИ ВИВЧЕННІ МАТЕРІАЛІВ ПОСІБНИКА | 170 |
| Додаток 3. ВАРІАНТИ ПРАКТИЧНИХ ЗАВДАНЬ ДЛЯ ПРОВЕДЕННЯ КОНТРОЛЬНОЇ РОБОТИ ЗА РЕЗУЛЬТАТАМИ ВИВЧЕННЯ МАТЕРІАЛІВ ПОСІБНИКА | 180 |
| Додаток 4. ТАБЛИЦЯ ОСНОВНИХ ТЕГІВ МОВИ HTML | 194 |
| Додаток 5. ТАБЛИЦЯ ОСНОВНИХ КОЛЬОРІВ ТА ЇХ КОДУВАННЯ В ШІСТНАДЦЯТКОВОМУ ФОРМАТІ | 202 |
| Додаток 6. ГЛОСАРІЙ ТЕРМІНІВ, ЩО ВИКОРИСТОВУЮТЬСЯ ПРИ РОЗРОБЦІ HTML-ДОКУМЕНТІВ | 203 |
| Додаток 7. КОДИ ПРАВИЛЬНИХ ВІДПОВІДЕЙ НА ТЕСТОВІ ПИТАННЯ | 208 |
| | |
| ЛІТЕРАТУРА | 209 |

ВСТУП

Наприкінці ХХ початку ХХІ сторіччя людство перейшло в принципово нову епоху свого існування – епоху постіндустріального суспільства. Цю нову епоху в розвитку людства інколи ще називають «інформаційним суспільством», «суспільством знань» з огляду на ту роль, яку в ньому відіграють знання та інформація.

Сутність інформаційного суспільства визначається кількома взаємопов'язаними процесами:

- інформація і знання стають важливим ресурсом і рушійною силою соціально-економічного, технологічного і культурного розвитку;
- формується ринок інформації і знань як чинник виробництва, поряд з ринками природних ресурсів, праці і капіталу;
- стрімко зростає питома вага галузей, які забезпечують створення, передавання і використання інформації;
- розвинена інформаційна інфраструктура перетворюється на умову, що визначає національну і регіональну конкурентоспроможність;
- розвиток і активне впровадження у всі сфери діяльності нових інформаційно-комунікаційних технологій істотно змінює моделі освіти, праці, суспільного життя і відпочинку.

Розвиток інформаційних технологій, який відбувся за останні 50 років, сформував принципово новий в інформаційному сенсі світ, в якому використання зовнішніх інформаційних ресурсів є природним наслідком глобалізації. Це проявляється перш за все в тому, що пристрої, приєднані до мережі Інтернет, використовуються у всіх областях людського життя. Комп'ютери і Інтернет проникають у всі галузі економіки, створюючи передумови формування і розвитку цифрової економіки. Впровадження цифрових технологій в різні сфери економіки істотно змінює стиль нашого життя, умови роботи і ведення бізнесу.

Інтернет все глибше входить у повсякденне життя вчених, підприємців, окремих громадян, залучає уряди країн до розробки державних документів, які регулюють і підтримують розвиток і використання цієї мережі.

Сьогодні додатки і технології мережі Інтернет не тільки визначають прогрес у наукових дослідженнях та освіті. Вони впливають на політику, економіку й обороноздатність країни, стають важливим чинником її розвитку, сприяють зростанню бізнесу і торгівлі. Розвиток Інтернет вносить великі зміни й у сфери медицини, культури і відпочинку. Багато побутових проблем вирішуються сьогодні за допомогою мережі. Національні сегменти Інтернет увійшли в число основних системоутворюючих компонентів промислового комплексу багатьох країн світу. Великий розвиток набуває електронна комерція. Інтернет застосовується у системах фінансових розрахунків та ЗМІ, стає частиною суспільних відносин та використовується органами державного управління, стає самостійною сферою бізнесу.

Останнім часом Інтернет стає найпопулярнішим засобом міжнародного спілкування. Мільйони людей кожного дня використовують Інтернет

головним чином як постійне джерело новин та інформації. Інтернет зараз отримав статус своєрідного мета-джерела, універсального інформаційного середовища, яке включає у себе всі можливі типи інформаційних джерел.

З широким розвитком мережі Інтернет в світі формується ринок електронних інформаційних ресурсів, до яких можна віднести гіпертекстові електронні видання, електронні публікації. Книжкові видавництва та журналісти створюють електронні версії періодичних видань, бізнесові структури розміщують інформацію про свою діяльність в мережі Інтернет.

Це означає, що цифрові технології, послуги і системи, пов'язані з засобами та технологіями створення електронних інформаційних ресурсів, сьогодні надзвичайно важливі для соціального розвитку. Вони забезпечують створення єдиного цифрового ринку, учасники якого дістануть можливість без перешкод вести економічну діяльність онлайн в умовах чесної і добросовісної конкуренції, самого високого рівня захисту особистих і споживчих даних незалежно від громадянства, національності або місця проживання. В той же час єдиний цифровий ринок вимагатиме спрощеного доступу до послуг і товарів в режимі онлайн як для окремих фізичних осіб, так і для підприємців.

Навчальний посібник "Web-технології та Web-дизайн: Застосування мови HTML для створення електронних ресурсів" призначений для набуття первинних теоретичних знань і практичних навичок створення гіпертекстових інформаційних продуктів із застосуванням мови гіпертекстової розмітки HTML.

В навчальному посібнику надано основний теоретичний матеріал, який стосується різних аспектів використання тегів мови HTML при створенні інформаційних ресурсів, наведено велику кількість прикладів використання основних структурних елементів мови HTML. Для закріплення отриманих знань та набуття необхідних практичних навичок студентам пропонуються завдання для самостійної роботи, варіанти практичних завдань для проведення контрольної роботи, приклади питань для проведення тестового контролю знань та довідкові матеріали, необхідні студентам під час виконання самостійних завдань.

Навчальний посібник розрахований на студентів вищих навчальних закладів, які вивчають технології створення електронних інформаційних ресурсів. Посібник не вичерпує всіх можливостей створення HTML-документів. Він може стати основою для подальшого поглибленого вивчення дисциплін, пов'язаних зі створенням інформаційних ресурсів для мережі Інтернет.

Розділ 1.

Загальна характеристика мови HTML

Всесвітній інформаційний простір World Wide Web (WWW) – це розподілена гіпертекстова база даних, що складається з Web-сторінок, об'єднаних між собою за допомогою гіперпосилань. Для створення Web-сторінок використовується мова розмітки гіпертексту HTML (HyperText Markup Language – мова гіпертекстової розмітки)¹. Розробники цієї мови мали за мету надати розробникам гіпертекстових документів простий інструмент створення документів для мережі Інтернет, причому зробити цей інструмент достатньо потужним.

На цій же мові базуються програмні засоби, за допомогою яких можна суттєво полегшити та візуалізувати процес створення Web-документів. Візуальні засоби створення HTML-документів (наприклад, **MS FrontPage**, який входить до складу розширеного пакету **MS Office**), мають інтерфейс, побудований за тими ж принципами, що і інтерфейс текстових процесорів (наприклад, **MS Word**). Для роботи з візуальними засобами створення HTML-документів можна взагалі не володіти мовою HTML: використовуючи надані йому інтерфейсом можливості, користувач відображає на екрані монітора Web-документ у такому вигляді, як він буде надаватись у подальшому мережею Інтернет; під час збереження цей документ програмними засобами автоматично перетворюється в HTML-файл, який крім текстової та графічної інформації містить команди розмітки документа (команди мови HTML). Недоліком візуальних засобів створення HTML-документів є те, що розмір створюваного ними HTML-файла у декілька разів більший, ніж файла, створеного людиною за допомогою звичайного текстового редактора². Цей недолік суттєво впливає на швидкість завантаження сторінки (і, відповідно, на вартість часу, який на це витрачається).

HTML-документ являє собою звичайний текстовий файл, який зберігається на комп'ютері засобами файлової системи. Такий спосіб представлення HTML-документів має свої переваги: HTML-файл можна створити в будь-якому текстовому редакторі на будь-якій апаратній платформі в середовищі будь-якої операційної системи.³

Таким чином гіпертекстова база даних в концепції WWW – це набір текстових файлів, написаних на мові HTML, яка визначає форму

¹ Мова HTML була запропонована в 1989 Тімом Бернерсом-Лі. Її основою був стандарт мови розмітки друкарських документів SGML (Standard Generalised Markup Language), який спирається на тагову модель опису документів. Ця модель широко застосовується в системах підготовки документів для друку. На момент створення HTML вже існував стандарт мови розмітки друкованих документів – Standard Generalised Markup Language, який і став основою HTML.

² Web-документ, створений засобами MS WORD, у 4 – 10 разів більший, ніж Web-документ аналогічного змісту, створений шляхом безпосереднього застосування команд мови HTML у програмі **Блокнот**).

³ На вибір представлення HTML-документів за допомогою звичайного текстового файла, який зберігається засобами файлової системи операційного середовища комп'ютера, вплинули наступні фактори. По-перше, такий файл може бути створений будь-яким текстовим редактором на будь-якій апаратній платформі в середовищі будь-якої операційної системи. По-друге, на момент розробки HTML вже існував американський стандарт для розробки мережних інформаційних систем, в якому одиницею зберігання визначався простий текстовий файл в кодуванні, яке відповідає ASCII.

представлення інформації (розмітка) і структуру зв'язків цих файлів (гіпертекстові посилання). тобто документи всесвітнього інформаційного середовища WWW – це документи, розмічені за допомогою HTML. Такий підхід передбачає наявність інтерпретатора HTML.

Всі програми-браузери (Internet Explorer, Opera, Mozilla, Mosaic, Netscape Navigator, Arena, Cello, Lynx тощо) фактично є інтерпретаторами мови HTML. Вираз "HTML – мова гіпертекстової розмітки документа" означає, що при створенні виконується розмітка текстового документа, тобто визначається, в якому місці документа слід розташувати малюнок, які розміри він повинен мати, яким шрифтом та розміром надрукувати той чи інший текст, який фоновий колір встановити для документа тощо. Ці вказівки сприймаються браузером, які і здійснюють представлення HTML-документа на екрані монітора.

Вся ідеологія мови HTML полягає у наступному: є обмежений набір елементів, які управляють відображенням тексту, користувач обирає потрібний, а браузер відповідним чином відображає текст. Такі символи, які управляють відображенням тексту і при цьому самі не відображаються на екрані, в мові HTML прийнято називати *тегами* (від англ. *tag* – ярлик, ознака, відмітка)¹. Теги можуть мати атрибути або не мати їх. Наприклад, тег верхнього рівня <HTML> не має атрибутів, а тег створення нової таблиці <TABLE BORDER="1" WIDTH="100"> містить два атрибути. Перший (BORDER="1") визначає товщину границі, а другий (WIDTH="100") – ширину всієї таблиці на сторінці. Щоб відрізнити текст сторінки від символів, які управляють відображенням цього тексту, всі теги мови HTML виділяються символами-обмежувачами (кутові дужки < і >)², між якими записується ідентифікатор (ім'я) тега і, при необхідності, його атрибути.

У загальному вигляді синтаксис запису тега в сукупності з його атрибутами виглядає таким чином:

```
<ТЕГ ім'я_атрибута-1="значення_атрибута-1" ім'я_атрибута-2="значення_атрибута-2" ... ім'я_атрибута-n="значення_атрибута-n">
```

Значення атрибутів слід брати у прямі лапки «"»³.

Специфікація мови HTML дозволяє опускає лапки для наступних видів атрибутів:

- Атрибути, записані тільки рядковими або заголовними символами латинського алфавіту і не включаючи інших символів, наприклад, цифр.
- Атрибути, що складаються тільки з цифр від 0 до 9.
- Атрибути, що позначають проміжки часу.

Наприклад, запис атрибутів BORDER="1" або ALIGN="CENTER" припустимо представити як BORDER=1 та ALIGN=CENTER.

¹ Блокові елементи документа, розміченого за допомогою HTML, прийнято називати контейнерами.

² Єдиним виключенням з цього правила є тег коментаря, обмежувачі якого мають вигляд <!-- i -->.

³ Якщо значення атрибути не містить в середині пробіл, то допускається лапки не використовувати.

Всі об'єкти, не поміщені в кутові дужки "< >", інтерпретатор сприймає як текстові елементи і відображає їх на екрані комп'ютера «як є».

Особливості застосування мови HTML

Регістр запису команд

Мова HTML не чутлива до регістра, в якому записані команди (в даному випадку – теги і імена їх атрибутів). Інтерпретатор браузера однаково обробить теги <HEAD>, <head>, <Head> або <hEaD>. Іншими словами, теги можна записувати як в рядковому, так і в заголовному регістрі. Проте рекомендується все ж таки застосовувати для написання тегів HTML заголовний регістр, перш за все, для поліпшення читаності коду, оскільки в цьому випадку команди мови буде легко відрізнити від інтегрованого в документ інформаційного тексту. Єдиним виключенням з цього правила є два елементи мови HTML, в яких регістр має суттєве значення. Це URL-адреси і escape-послідовності.

Парність тегів

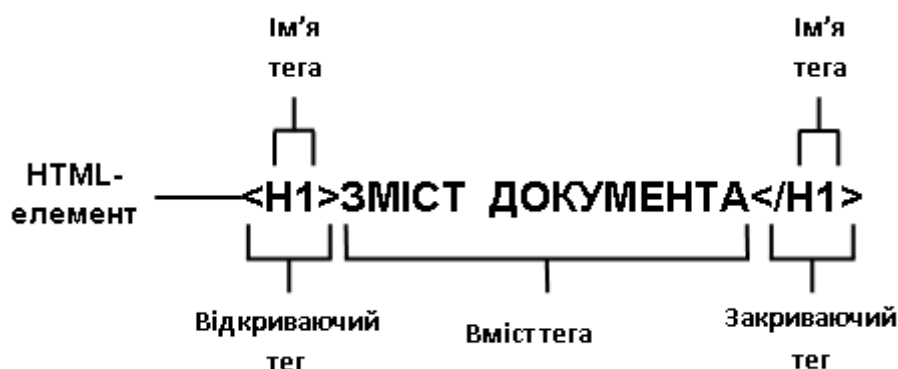


Рис. 1.1. Структура тегів мови HTML

Практично всі теги мови HTML використовуються парно. Така пара складається з «відкриваючого» і «закриваючого» тегів. За правилами мови HTML закриваючий тег записується так само як і відкриваючий тег, але з символом / (прямий слеш) перед іменем тега. Єдиною принциповою відмінністю закриваючих тегів від відповідних відкриваючих є те, що закриваючі теги не містять атрибутів (параметрів). Все, що розташовано між відкриваючим і закриваючим тегами, обробляється інтерпретатором згідно алгоритму, привласненому даному конкретному тегу. Тобто за своїм значенням відкриваючий та закриваючий теги близькі до поняття дужок "begin/end". У загальному вигляді рядок HTML з відкриваючим і закриваючим тегами виглядає так, як це представлено на Рис.1.1.

Все, що записано між відкриваючим і відповідним закриваючим тегами, називатимемо *вмістом тега-контейнера*. Необхідно запам'ятати одне

просте правило: якщо в HTML-документі зустрічається відкриваючий тег, обов'язково повинен бути і закриваючий.

Недотримання цього правила може викликати помилку при обробці такого документа інтерпретатором браузера. Ця властивість мови HTML дозволяє використовувати принцип вкладення одного тега в інший. В цьому випадку оброблюваним значенням одного тега може служити інший тег. Нижче наведено приклад вкладення двох тегів один в один:

<ТЕГ1> <ТЕГ2>Значення, що обробляється</ТЕГ2></ТЕГ1>

Існують теги, які не потребують закриваючих тегів. Прикладами таких тегів є тег вставки зображення , тег примусового початку нового рядка
, тег визначення базового шрифту <BASEFONT>, тег вставки горизонтальної лінії <HR> тощо. Часто з самого призначення тега можна здогадатися, чи повинен він мати закриваючий тег.

Іноді закриваючий тег можна не писати. Наприклад, застосування тегу <P> для початку нового абзацу автоматично відміняє дію попереднього тегу <P>, тобто закриваючий тег </P> застосовувати не обов'язково; застосування наступного тегу для початку нового елемента маркірованого або нумерованого списку означає, що попередній елемент списку вже закінчився, тобто закриваючий тег застосовувати не обов'язково.

Escape-послідовності

Іноді виникає необхідність використовувати в тексті HTML-документа символи, зарезервовані для позначення елементів мови HTML. Це можуть бути кутові дужки (символи "<" і ">"), в яких записуються теги, прямі лапки ("), що вживаються при визначенні значень атрибутів тощо. Іншими словами, необхідно, щоб дані символи відображались у вікні браузера «як є», тобто як текст, інтерпретатор же сприймає їх як частину команд розмітки документа. Для того, щоб уникнути проблем при відображенні подібних елементів, слід застосовувати так звані escape-послідовності, або, як їх ще іноді називають, "нотації символічних об'єктів" (або CER – Character Entity Reference).

Наприклад, створюється HTML-документ, в якому йдеться про елементи цієї мови. Якщо вказати ім'я тега <BODY> просто в документі, браузер сприйме його безпосередньо як тег опису тіла документа. Для того, щоб на екрані з'явилась саме потрібна послідовність символів і використовуються відповідні символам "<" та ">" escape-послідовності.

Щоб представити символ "<" в документі HTML, потрібно замінити його на <, а символ ">" – на >. Тобто, якщо вказати в тексті HTML рядок <BODY>, він виглядатиме на екрані як текст <BODY>.

Запис escape-послідовності передбачає заміну службових символів спеціальними командами, які в процесі інтерпретації документа заміщаються шуканими знаками, що виводяться на екран. Всі escape-послідовності починаються з символу & (амперсанд) і закінчуються крапкою з комою, а

між ними розміщується сама команда, записана в рядковому реєстрі (маленькими літерами).

Escape-послідовності записуються в тексті за тими ж правилами, за якими уживаються символи, що позначаються ними, іншими словами, нотація символічного об'єкту просто вставляється в ту ділянку тексту, де повинен знаходитись відповідний службовий символ.

На відміну від назв тегів HTML, назви escape-послідовностей чутливі до регістра символів. Існує і інший спосіб визначення escape-послідовностей: їх назви можуть задаватися не у вигляді імені, а за допомогою тризначних кодів символів у вигляді **&#nnn;**. В Таблиці 1 приведені найчастіше використовувані escape-послідовності і відповідні їм числові коди¹.

Таблиця 1.1

Escape-послідовності і відповідні їм числові коди

| Числовий код | ESCAPE-послідовність | Символ | Назва символа |
|-------------------|----------------------|--------------|---------------------------------|
| " | " | " | Подвійна лапка |
| & | & | & | Амперсанд |
| < | < | < | Менше |
| > | > | > | Більше |
| ™ ™ | ™ | ™ | Торгівельна марка |
| | | | Нерозривний пробіл |
| ¡ | ¡ | ! | Перевернений знак оклику |
| ¢ | ¢ | ¢ | Цент |
| £ | £ | £ | Фунт стерлінгів |
| ¤ | ¤ | ¤ | Загальний знак валюти |
| ¥ | ¥ | ¥ | Йена |
| § | § | § | Знак параграфу |
| ¨ | ¨ | ¨ | Умляут |
| © | © | © | Авторське право (Копірайт) |
| « | « | « | Ліва кутова лапка |
| ­ | ­ | ¬ | М'який перенос |
| ® | ® | ® | Зареєстрована торгівельна марка |
| ° | ° | ° | знак градуса |
| ± | ± | ± | Плюс або мінус |
| ² | ² | ² | Верхній індекс 2 |
| ³ | ³ | ³ | Верхній індекс 3 |
| ´ | ´ | ´ | Наголос |

¹ Про коди всіх символів ASCII-таблиці можна дізнатись на сайті <http://www.w3.org>, сторінка http://www.w3.org/MarkUp/html-spec/html-spec_13.html

| Числовий код | ESCAPE-послідовність | Символ | Назва символу |
|--------------|----------------------|--------------|-------------------------|
| ¶ | & para; | ¶ | Знак абзаца |
| · | & middot | • | крапка посередині рядка |
| ¹ | & sup1 | ¹ | Верхній індекс один |
| » | & raquo; | » | Права кутова лапка |
| ¼ | & frac14 | ¼ | Одна чверть |
| ½ | & frac12 | ½ | Одна друга |
| ¾ | & frac34 | ¾ | Три чверті |

ПРИКЛАДИ

- Для того, щоб відобразити на екрані тег <HTML>, слід при розмітці документа записати:

<HTML >

- Для того, щоб визначити зареєстровану торговельну марку (MyFirm[®]), розмітка повинна мати вигляд:

MyFirm^{®}

або

MyFirm^{®}

- Для незареєстрованої торговельної марки (MyFirm[™]), розмітка повинна мати вигляд:

MyFirm^{™}

або

MyFirm^{™}

- Авторські права (All rights registered[©]) на екрані позначаються наступною escape-послідовністю

All rights registered^{©}

або

All rights registered^{©}

- Розмітка формули “y<x±1/a” може бути здійснена наступною escape-послідовністю

<P> y<x±1/a </P>

Переведення рядка, пропуски, табуляція

При підготовці HTML-документа обов'язково слід пам'ятати, що будь-яка кількість пробілів між словами, відступ, виконаний за допомогою клавіші «**Tab**», або переведення рядка після натискання клавіші «**Enter**» перетворюються на екрані в один символ пропуску. Тому для розмітки основного тексту HTML-документа слід застосовувати спеціальні теги. Виключенням з цього правила може служити тільки текст, помічений спеціальним тегом

<PRE> – командою попереднього форматування. Мнемоніка запису цього тега виглядає таким чином:

<PRE> *попередньо відформатований текст* **</PRE>**

В цьому випадку все, що розташовано між відкриваючим і закриваючим тегом **<PRE>**, відобразиться у вікні браузера із збереженням всіх введених автором пропусків, переведень рядків і відступів табуляції. Проте зловживати застосуванням даної команди не рекомендується, оскільки через несхожість алгоритмів обробки коду HTML різними браузерами можливе спотворення тексту. Слід також пам'ятати, що всередині тега **PRE** заборонено використовувати інші теги форматування тексту.

Коментарі

Код HTML може містити коментарі, призначені як для самого автора, так і для стороннього користувача, що читає лістинг чужої сторінки. І в тому і в іншому випадку коментарі записують з метою уникнути можливої плутанини при вивченні коду html-документа. Коментарі не обробляються інтерпретаторами браузерів і не відображаються на екрані, вони залишаються у складі коду сторінки «як є». Коментар відкривається символічною послідовністю "**<!--**", закривається ж символами: "**-->**". Іншими словами, в загальному випадку синтаксис запису коментаря виглядає таким чином:

<!--текст коментаря **-->**

Інколи в коментар включається інформація про творця документа, попередження про авторські права або адреса електронної пошти адміністратора сайту. Проте найчастіше даний елемент коду HTML використовується для пояснення призначення тих або інших ділянок лістингу html-документа.

Загальні правила інтерпретації тегів браузерами

1. Мова HTML не реагує на невірний запис тегів. Невірно записаний тег або його параметр просто ігноруються браузером. Наприклад якщо замість **
** було надруковано **<VR>**, то браузер просто не розпочне новий рядок і наступний текст відобразатиметься на екрані в тому ж рядку. Це загальне правило для всіх браузерів. Під дію цього правила підпадають не тільки невірні записані теги, але і теги не розпізнавані даною версією браузера.
2. Теги можуть мати атрибути. Набори допустимих атрибутів індивідуальні для кожного тега. Загальні правила запису атрибутів полягають в наступному.
 - 2.1. Атрибути розташовуються після імені тега.
 - 2.2. Один від одного атрибуту відділяються пропусками.
 - 2.3. Порядок проходження атрибутів тега довільний.
 - 2.4. Багато атрибутів вимагають визначення їх значень, проте деякі атрибути не мають значень або можуть записуватися без них, приймаючи значення за умовчанням.

- 2.5. Якщо атрибут вимагає значення, то воно вказується після назви атрибута через знак "дорівнює". Значення атрибута може записуватися як в лапках, так і без них. Єдиним випадком, в якому лапки є обов'язковими, є випадок, коли в значенні атрибута є пробіли.
- 2.6. У значеннях атрибутів (на відміну від назв тегів і самих атрибутів) іноді важливим є регістр запису.

ПРИКЛАД 1.1

Нижче наведено тег <BODY> з атрибутами BGCOLOR="yellow" (значення yellow визначає жовтий фоновий колір всієї сторінки), LINK="green" (значення green визначає зелений колір гіперпосилань на сторінці), TEXT="blue" (значення blue визначає синій колір тексту на сторінці).

<BODY BGCOLOR="white" LINK="green" TEXT="blue">

3. Теги можуть вкладатися один в одного, наприклад: <i>текст</i>. В цьому випадку важливо дотримувати послідовність їх відкриття і закриття:

```

<тег1>
  <тег2>
    <тег3>
      . . .
    </закриття тега3>
  </закриття тега2>
</закриття тега1>

```


4. HTML не реагує на регістр символів. Наприклад, тег відкриваючий таблицю, може бути написаний як <TABLE>, або <table>, або <tAbLe>. Зайві пропуски і переноси також не враховуються. (Є єдине виключення – тег <PRE>, після якого браузер починає враховувати пропуски і переноси).

Технологія створення HTML-документів

Для створення HTML-документа можна застосувати будь-який текстовий редактор, наприклад БЛОКНОТ (входить до складу стандартного програмного забезпечення системи Windows)¹. Суттєвою перевагою використання текстового редактора є те, що він не додає до тексту HTML-документа ніяких спеціальних команд.

¹ На відміну від текстових процесорів, текстові редактори дозволяють здійснювати з текстом тільки операції редагування, в той час як форматування тексту в текстових редакторах є неможливим. До текстових процесорів відносяться WordPad (група Стандартні системи Windows) та MS Word, що є складовою частиною MS Office.

Загальна технологія створення HTML-документів складається з наступних етапів:

1. Відкрити редактор **Блокнот**, виконавши команду Головного меню WINDOWS **Пуск-Программы-Стандартные-Блокнот**.
2. Ввести документ.
3. Командою меню редактора **Файл-Сохранить как** зберегти створений документ, обов'язково надавши йому розширення **.htm**. Правильно збережений документ матиме піктограму інтернет-браузера, встановленого на комп'ютері.
4. Переглянути створений HTML-документ, для чого завантажити будь-який встановлений на комп'ютері браузер (**Internet Explorer, Opera, Mozilla, Chrome**), виконати команду **Файл-Открыть** та, натиснувши кнопку **Обзор**, знайти у файловій системі комп'ютера щойно створений файл (той самий результат може бути досягнуто подвійним клацанням мишею імені щойно створеного файла).
5. При необхідності внести зміни в HTML-документ. Для внесення змін повернутись до програми **Блокнот**, внести зміни, командою **Файл-Сохранить** зберегти зміни у файлі¹.
6. Активізувати **Internet Explorer**, натиснути кнопку **Обновить** .
7. Процедуру корегування документа повторювати, поки не буде досягнуто потрібного відображення HTML-документа у вікні браузера².

ЗАВДАННЯ


I. Опанувати технологію створення, перегляду та редагування HTML-документів. Для цього:

1. Відкрити редактор **Блокнот**, виконавши команду Головного меню WINDOWS **Пуск-Программы-Стандартные-Блокнот**.
2. Ввести документ відповідно до ЗРАЗКА 1.
3. Командою меню редактора **Файл-Сохранить как** зберегти створений документ, обов'язково надавши йому розширення **.htm**. Правильно збережений документ матиме піктограму встановленого на комп'ютері Internet-браузера (наприклад , , ).
4. Переглянути створений HTML-документ, для чого завантажити встановлений на комп'ютері браузер, виконати команду **Файл-Открыть** та, натиснувши кнопку **Обзор**, знайти у файловій системі комп'ютера щойно створений файл (той самий результат може бути

¹ Більшість Web-браузерів дозволяють переглядати Web-сторінки у форматі HTML. Наприклад, у браузері **Internet Explorer** для перегляду документа у форматі HTML необхідно натиснути праву кнопку мишки і вибрати у контекстному меню пункт – **Просмотр в виде HTML** (або **Просмотр HTML-кода**), або скористатись командою меню вікна браузера **Вид-В виде HTML** (або **Вид-Просмотр HTML-кода**).

² Ні в якому разі не слід зберігати створений документ командами меню вікна браузера, оскільки браузер використовується тільки для перегляду створеного документа. Збереження здійснюється тільки у **Блокноті**.

досягнуто подвійним клацанням мишею імені щойно створеного файла або виконанням відповідної команди контекстного меню).

5. Внести зміни в HTML-документ (здійснити перше редагування документа) відповідно до ЗРАЗКА 2 (вказати власне прізвище та шифр групи). Для внесення змін повернутись до програми **Блокнот**, внести зміни, командою **Файл-Сохранить** зберегти зміни у файлі.
6. Активізувати **Internet Explorer**, натиснути кнопку **Обновить**  .
7. Повернувшись до програми **Блокнот**, здійснити друге редагування створюваного HTML-документа відповідно до ЗРАЗКА 3. Командою **Файл-Сохранить** зберегти зміни у файлі.
8. Активізувати **Internet Explorer**, натиснути кнопку **Обновить**  та здійснити перегляд оригінального документа.
9. Процедуру корегування документа повторювати, поки не буде досягнуто потрібного відображення HTML-документа у вікні браузера.

II. Дослідити структуру створеного HTML-документа. При дослідженні звернути увагу на такі питання:

1. Призначення мови HTML.
2. Технологія створення, редагування та перегляду HTML-документа.
3. Технологія розмітки HTML-документа.
4. Структура тега. Призначення атрибутів тегів та їх синтаксис.
5. На які частини поділяється HTML-документ.
6. Виділити в документі заголовок та тіло документа.
7. Дослідити всі теги, використані при виконанні завдань.

III. Скласти звіт про виконання роботи. В звіті необхідно:

1. Охарактеризувати мову HTML.
2. Описати технологію створення, редагування та перегляду . Описати структуру тега.
3. Описати структуру HTML-документа та теги, які слід застосувати для створення цієї структури.
4. Перелічити всі теги заголовка сторінки. Описати їх.
5. Описати всі теги, використані при розмітці тіла сторінки. Визначити при цьому, що саме виконує кожен тег та які атрибути для цього використані.