

О. Г. ПАСІЧНИК, О. В. ПАСІЧНИК, І. В. СТЕЦЕНКО

Основи веб-дизайну

Серія «Інформатика. Профільне навчання»
За загальною редакцією доктора технічних наук, професора,
академіка АПН України А. М. Гуржія

Рекомендовано
Міністерством освіти і науки України

Київ
Видавнича група ВНУ
2009

УДК 004.7
ББК 32.973
П-19

Рецензенти: О. В. Дерев'янченко, доцент Київського національного університету ім. Тараса Шевченка, кандидат фіз.-мат. наук.

Д. І. Кожем'яка, вчитель інформатики Фінансово-правового ліцею Фінансово-правового коледжу КНУ ім. Т. Шевченка.

*Гриф наданий Міністерством освіти і науки України,
лист № 1/11-988 від 31.03.2008 р.*

Пасічник О. Г., Пасічник О. В., Стеценко І. В.

П-19 Основи веб-дизайну / О. Г. Пасічник, О. В. Пасічник, І. В. Стеценко : [Навч. посіб.]. — К.: Вид. група ВНУ. — 2009. — 336 с: іл.
ISBN 978-966-552-228-7

У посібнику в доступній формі описані основні методи та засоби веб-дизайну. Розглядаються мови HTML і JavaScript, автоматизовані та візуальні засоби розробки веб-сайтів, зокрема програма FrontPage, служби хостингу та популяризації сайтів, а також художні аспекти веб-дизайну. Книжка містить теоретичний матеріал, практичні роботи, вправи, завдання для самостійного виконання і тести.

Посібник призначено для учнів та учителів середніх загальноосвітніх шкіл, ліцеїв і гімназій. Він може використовуватися під час вивчення базового курсу інформатики і курсу за вибором. Видання також стане в нагоді всім, хто прагне опанувати веб-дизайн самостійно.

ББК 32.973

Усі права захищено. Жодна частина цієї книжки не може бути відтворена в будь-якій формі будь-якими засобами без письмового дозволу власників авторських прав. Інформація, що міститься в цьому виданні, отримана з надійних джерел і відповідає точці зору видавництва на обговорювані питання на поточний момент. Проте видавництво не може гарантувати абсолютну точність та повноту викладених відомостей і не несе відповідальності за можливі помилки, пов'язані з їхнім використанням. Наведені у книжці назви продуктів або організацій можуть бути товарними знаками відповідних власників.

ISBN 978-966-552-228-7

© Видавнича група ВНУ, 2009

Зміст

Передмова	7
Від видавництва.....	8
Розділ 1. Автоматизоване створення й підтримка веб-ресурсів	9
Основи Інтернету.....	9
Передавання даних в Інтернаті.....	9
Адресація в Інтернеті.....	10
Веб-сторінка та веб-сайт.....	12
Домашні сторінки.....	13
Структура веб-сайтів.....	14
Створення та підтримка веб-ресурсів в автоматизованому режимі.....	16
Безкоштовний хостинг веб-сайту.....	16
Форуми та чати на веб-сайтах.....	20
Створення та ведення блогів.....	24
Практична робота № 1.....	28
Практична робота № 2.....	37
Самостійна робота.....	40
Тест.....	41
Розділ 2. Основи мови HTML	45
Базові конструкції мови HTML.....	45
Поняття тагу.....	46
Структура HTML-документа.....	47
Теги форматування тексту.....	49
Нумеровані й марковані списки.....	54
Текстові гіперпосилання.....	57
Використання таблиць у HTML-документах.....	61
Створення таблиці.....	61
Об'єднання комірок.....	64
Встановлення ширини таблиці та комірок.....	67
Форматування тексту та оформлення рамок таблиці.....	68

Колірне оформлення таблиць.....	69
Використання таблиць для розміщення об'єктів на веб-сторінці.....	70
HTML-документи, які містять фрейми.....	74
Фрейми, їхні теги й атрибути.....	75
Використання посилань у фреймах.....	78
Недоліки використання фреймів.....	80
Практична робота № 3.....	81
Практична робота №4.....	84
Практична робота №5.....	86
Самостійна робота.....	89
Тест.....	91

Розділ 3. Графіка, аудіо- та відеоінформація на веб-сторінках..... 95

Використання зображень у веб-документах.....	95
Формати зображень, які використовують в Інтернеті.....	96
Розміщення і вирівнювання зображень на веб-сторінках.....	97
Створення GIF-анімації.....	109
Карти посилань.....	113
Розміщення та відтворення на веб-сторінках мультимедійних даних.....	119
Формати аудіо- та відеофайлів.....	119
Технології та засоби відтворення мультимедіа.....	121
Використання мультимедіа на веб-сторінках.....	123
Практична робота № 6.....	126
Самостійна робота.....	131
Тест.....	132

Розділ 4. Візуальний редактор веб-сайтів..... 135

Інтерфейс та режими роботи програми.....	135
Вікно програми.....	136
Режими перегляду веб-документа.....	137
Робота з веб-сторінками.....	138
Відкриття та створення сайтів і веб-сторінок.....	138
Форматування тексту.....	140
Розміщення і настроювання зображень.....	141
Створення таблиць.....	145
Створення гіперпосилань.....	147
Створення веб-сайту за допомогою майстра або шаблону.....	150
Використання майстрів та шаблонів для сайтів.....	151
Шаблони веб-сторінок.....	153

Практична робота № 7.....	157
Самостійна робота.....	165
Тест.....	165
Розділ 5. Таблиці каскадних стилів і динамічні веб-сайти.....	169
Використання таблиць каскадних стилів.....	169
Поняття про таблиці каскадних стилів.....	170
Застосування каскадних стилів у HTML-документах.....	171
Позиціонування елементів веб-сторінки за допомогою стилів.....	178
Застосування стилів у програмі FrontPage.....	182
Створення та підключення зовнішньої таблиці стилів.....	183
Внутрішня таблиця стилів та вбудовані стилі.....	186
Поняття об'єктної моделі.....	189
Властивості, методи та події.....	190
Об'єктна модель браузера Internet Explorer.....	193
Сценарії.....	198
Мова програмування JavaScript.....	199
Використання форм.....	204
Створення форм та динамічних ефектів у FrontPage.....	212
Шаблони та майстри для створення форм.....	213
Елементи форм.....	218
Ефекти динамічної HTML у FrontPage.....	222
Практична робота № 8.....	229
Практична робота № 9.....	234
Самостійна робота.....	242
Тест.....	242
Розділ 6. Хостинг та популяризація сайтів.....	247
Розміщення веб-сайту в мережі Інтернет.....	247
Вибір URL-адреси.....	248
Хостинг на безкоштовному сервері.....	249
FTP-З'єднання з хостом.....	256
Рекламування сайту в Інтернеті.....	258
Банеритабанерні мережі.....	258
Реєстрація ресурсу в пошукових системах.....	262
Практична робота № 10.....	268
Самостійна робота.....	272
Тест.....	272

Розділ 7. Дизайн веб-сайтів	276
Типи сайтів та їх особливості	276
Статичні та динамічні сайти	277
Тематична класифікація	277
Візуальна класифікація	279
Планування веб-сайту та етапи роботи над ним	282
Веб-дизайн у контексті художнього дизайну	285
Основні поняття дизайну	285
Ергономіка	288
Основні складові веб-дизайну	289
Проектування структури сайту	291
Вибір інформаційної архітектури	291
Засоби навігації	295
Проектування гіперпосилань	299
Просторовий дизайн веб-сторінок	301
Вибір розмірів веб-сторінок	302
Компонування та визначення набору сторінок сайту	304
Розміщення елементів на сторінці	306
Інформаційне наповнення веб-сторінки	311
Підготовка текстів	311
Вибір шрифтового оформлення	311
Графічне оформлення веб-сторінки	314
Сприйняття людиною форм та фігур	315
Коліру веб-дизайні	318
Текстури у веб-дизайні	322
Типові помилки дизайнерів-початківців	323
Практична робота № 11	326
Самостійна робота	330
Тест	331

Передмова

Уміння створювати якісні й цікаві веб-сайти наразі стає однією з найважливіших складових інформаційної культури людини, адже від того, як вона зможе представити у світовому інформаційному просторі себе, коло своїх професійних та особистих інтересів або ж реалізувати в Інтернеті той чи інший проект, пов'язаний із професійною діяльністю, багато в чому залежить успішність її кар'єри. Такі вміння вже не пов'язані з певною професією, вони необхідні для будь-якого активного члена сучасного суспільства, і тому навчання основам веб-дизайну має стати однією зі складових базового курсу інформатики в середній школі. Проте у більшості діючих програм з інформатики відповідної теми не передбачено, а отже, найпоширенішою формою викладання у школі основ веб-дизайну може стати проведення курсу за вибором. Видання, яке ви тримаєте в руках, містить повний комплект навчально-методичних матеріалів, необхідних для організації такого курсу, хоча може використовуватися і для викладання основ веб-дизайну в будь-якій іншій формі.

Посібник розрахований на проведення 35-годинного навчального курсу, протягом якого учні мають виконати 11 практичних робіт (на кожну з них слід відвести 20-25 хвилин). Крім цих робіт, які підсумовують результати вивчення тієї чи іншої теми, передбачені численні вправи з покроковим описом дій, виконуваних учнями за комп'ютером. Кожний розділ книжки також містить тест із дванадцяти запитань та завдання для самостійного виконання. Отже, посібник є практично спрямованим, але попри це автори прагнули викласти й теоретичний матеріал, в обсязі, достатньому для наповнення лекційної складової курсу та самостійного вивчення предмета.

Слід зазначити, що у навчальній літературі поняття веб-дизайну часто підмінюється поняттям веб-програмування та супутніх технологій. Наш посібник у цьому контексті є винятком, адже

в ньому значну увагу приділено просторовому і колірному оформленню веб-сторінок, дизайну інформаційного наповнення сайту та іншим питанням, пов'язаним із художніми аспектами розробки веб-ресурсів. Звичайно, робота веб-дизайнера неможлива без володіння певним технічним мінімумом. Тому у виданні детально й ґрунтовно висвітлюються такі теми, як основи мови HTML, графіка, подання на веб-сторінках аудіо- та відеоінформації, використання графічного редактора веб-сайтів тощо.

Модульна структура посібника дозволяє поділити матеріал за рівнями складності, гнучко пристосовуючи обсяг та глибину курсу до специфіки навчального закладу. Так, для закладів гуманітарного профілю курс може бути обмежений вивченням 1 і 7 розділів посібника, в яких розглядається процес автоматизованого створення веб-ресурсів та художній дизайн сайтів. Для більшості інших закладів доцільним буде також вивчення розділів 2-4 та 6, де описується мова HTML, використання графіки, звуку та відео на веб-сторінках і робота з Microsoft FrontPage — графічним редактором веб-сайтів. Для закладів інформаційно-технологічного профілю призначено ще й розділ 5, у якому викладено основи динамічного HTML.

Акумулювавши власний та чужий досвід викладання веб-дизайну, спрямувавши значні зусилля на те, щоб зробити матеріал посібника доступним, сучасним і необхідним українській школі, автори водночас усвідомлюють, що як перше видання ця книжка є далекою від досконалості, й тому сподіваються на активну співпрацю з тими, хто використовуватиме посібник у своїй педагогічній роботі та навчальній практиці.

Бажаємо нашим читачам натхнення та успіху в освоєнні курсу «Основи веб-дизайну» і сподіваємося, що наш посібник буде надійною опорою та помічником на нелегкому шляху пізнання.

Від видавництва

За докладною інформацією про видання серії «Інформатика. Профільне навчання» звертайтеся на сайт Видавничої групи ВНУ (<http://www.bhv.kiev.ua>) та на сайт «Онлайнова освіта. Профільне навчання» (<http://osvitaonline.googlepages.com>). На них ви також знайдете навчально-методичні матеріали до посібників серії.

Розділ 1

Автоматизоване створення й підтримка веб-ресурсів

Перш ніж ви почнете вивчати веб-дизайн, тобто проектування та розробку інтернет-ресурсів, слід ознайомитися з деякими загальними поняттями, які стосуються мережі Інтернет та служби Веб. у цьому розділі йтиметься про передавання даних через Інтернет, способи адресації його ресурсів та особливості Веб. Зокрема, ви дізнаєтеся, що таке гіперпосилання, веб-сторінка і веб-сайт. На вас також чекають цікаві практичні заняття, під час яких ви навчитесь використовувати автоматичні засоби створення веб-ресурсів та побачите, як за їх допомогою створити веб-сторінки різних типів і організувати на веб-сайті форум та чат.

Основи Інтернету

Сьогодні у світі є сотні тисяч великих та малих комп'ютерних мереж. Багато з них з'єднані між собою й утворюють єдиний інформаційний простір, який складається з мільйонів комп'ютерів. Цей єдиний віртуальний простір називають *Інтернетом*.

За кількістю комп'ютерів та охоплюваною територією Інтернет є найбільшою у світі мережею. За типом вона належить до клієнт-серверних мереж, тобто в Інтернеті є комп'ютери-сервери, які зберігають інформацію та надають її комп'ютерам-клієнтам.

Передавання даних в Інтернеті

Обмін інформацією між серверами та клієнтами здійснюється за певними правилами, які називають *протоколами*. Всі дані, що циркулюють у глобальному інформаційному полі, розбито на невеликі

блоки і вкладено в пакети. Кожний пакет окрім даних має заголовок, де зберігаються адреса відправника, адреса одержувача та інша інформація, необхідна для збирання пакетів у пункті призначення. Теоретично можливо, що різні пакети одного повідомлення пройдуть різними шляхами, але все одно досягнуть адресата і будуть зібрані в повне повідомлення.

Поділ даних на пакети та їх збирання у пункті призначення здійснюється під керуванням протоколу TCP (Transmission Control Protocol — протокол керування передаванням), а власне передавання пакетів мережею та досягнення ними адресата забезпечує протокол IP (Internet Protocol — міжмережний протокол).

У Інтернеті використовується велика кількість протоколів, завдяки чому існує широкий спектр служб, які надаються та підтримуються за допомогою цієї глобальної мережі.

Найпопулярнішою зі служб є Всесвітня павутина (World Wide Web — WWW), або просто Веб. Це розповсюджена по всьому світу інформаційна мультимедійна система, яка об'єднує в єдиному просторі інформацію різних типів. Робота у Веб подібна до віртуальної подорожі світом з відвідуванням цікавих місць. Ця служба базується на протоколі НТТР (Hyper Text Transfer Protocol — протокол передавання гіпертексту).

Іншою службою, з якою вам доведеться працювати під час вивчення матеріалу посібника, є FTP (File Transfer Protocol — протокол передавання файлів). Як ви, напевно, здогадалися, назва цієї служби збігається з назвою протоколу, який вона використовує. Сервери, що підтримують цей протокол, називають FTP-серверами. Частина дискового простору таких серверів доступна через Інтернет.

Крім того, до служб Інтернету належать електронна пошта, служби миттєвого передавання повідомлень (наприклад, ICQ), служба новин Usenet та інші.

Адресація в Інтернеті

Усі комп'ютери, підключені до Всесвітньої мережі, працюють в автоматичному режимі, без участі людей. Для того щоб такий комп'ютер мав змогу передавати та приймати дані з використанням протоколу IP, він повинен мати унікальну адресу, яку називають

IP-адресою. Вона має такий формат: xxx.xxx.xxx.xxx, де xxx — число від 0 до 255 (наприклад, 193.205.31.47). Призначенням IP-адрес займаються спеціальні організації.

IP-адреса є зручною для комп'ютерів, але людям запам'ятати її важко. Тому серверам присвоюють так звані *доменні імена* — набори розділених крапками послідовностей символів, наприклад zirka.lviv.ua.

Останній, найзагальніший елемент доменного імені — в нашому прикладі *ua* — називають доменом першого рівня, *lviv* — доменом другого рівня і т. д. Кількість доменів у адресі не регламентовано. Домени першого рівня стандартизовані, домени останніх рівнів вибирають за бажанням тих, хто їх встановлює. Імена для доменів першого рівня видає Інформаційний центр Інтернету (InternetNIC), решту імен фіксують ті організації, яким такі права делеговані.

У багатьох країнах домен першого рівня є кодом країни: *ua* — Україна, *ru* — Росія, *fr* — Франція, *se* — Швеція, *ca* — Канада, *jp* — Японія, *au* — Австрія, *uk* — Великобританія.

Домени першого рівня можуть також позначати сферу діяльності: *com* — комерційні компанії, *gov* — урядові організації, *edu* — навчальні заклади, *org* — некомерційні організації, *mil* — військові організації.

Отже, система доменних імен організована у зручний для людини спосіб, оскільки вона має постійну структуру і дає змогу визначити, якій організації належить сервер та в якій країні ця організація розміщена.

Проте, як ви знаєте із власного досвіду, для відкриття будь-якого документа, що зберігається на комп'ютері, необхідно вказати ім'я файлу та повний шлях до нього. Так само і для доступу до інтернет-ресурсу недостатньо знати лише IP-адресу або доменне ім'я комп'ютера, на якому цей ресурс розміщено, — ви маєте вказати також папку та ім'я файлу. Крім того, як зазначалося, в Інтернеті застосовуються різні протоколи, а отже, слід вказати ще й протокол. Адресу, що містить усі зазначені елементи, називають URL (Uniform Resource Locator — єдиний вказівник на ресурс) або *адресою ресурсу*.

Типовий URL має такий вигляд: протокол://адреса, де *протокол* визначає метод доступу до ресурсу, наприклад *http*, *ftp*; *адреса*

описує місце розташування ресурсу і включає назву сервера, шлях до документа і його ім'я.

Часто зустрічаються URL, що містять лише протокол та назву сервера. У цьому випадку завантажується головна сторінка сервера. Наведемо приклади URL-адрес:

- <http://www.syhiv.com/pub/files/school.html> — адреса файлу school.html, розміщеного в каталозі pub/files на сервері www.syhiv.com. Доступ до сервера здійснюється за протоколом HTTP;
- <http://www.syhiv.com> — адреса головної сторінки сервера www.syhiv.com;
- <ftp://ftp.syhiv.com/pub/files/school.txt> — адреса файлу school.txt, розміщеного в каталозі pub/files на сервері [ftp.syhiv.com](ftp://ftp.syhiv.com). Доступ до сервера здійснюється за протоколом FTP.

Веб-сторінка та веб-сайт

Як зазначалося, найвідомішою та найпопулярнішою службою Інтернету є Всесвітня павутина (Веб). Саме після її розповсюдження став можливий масовий доступ користувачів до Всесвітньої мережі. Своєю появою Веб має завдячити Тіму Бернесу-Лі, який винайшов протокол HTTP, адреси URL та мову HTML (з нею ви ознайомитеся у наступному розділі посібника) — технології, на яких ґрунтується Веб.

Служба Веб підтримується сукупністю серверів, які здатні обмінюватися даними за протоколом HTTP. Цих серверів мільйони, й розповсюджені вони по всьому світу. На них містяться *веб-сторінки* — спеціальні документи, створені з використанням мови HTML. Кожна веб-сторінка має адресу URL, за допомогою якої вона може бути знайдена.

Перегляд веб-сторінок здійснюється у спеціальних програмах — браузерях, найпоширенішими з яких є Internet Explorer, Mozilla та Opera. Для відтворення веб-документа у вікні браузера достатньо ввести його URL в поле Адреса і натиснути клавішу Enter.

Основною особливістю та перевагою веб-сторінок є те, що інформація на них організована як *гіпертекст*. Це текст, в який вбудовано спеціальні коди, що керують такими додатковими елементами,

як форматування, ілюстрації, мультимедійні вставки та гіпертекстові посилання.

Гіпертекстове посилання (гіперпосилання, гіперзв'язок чи гіперлінк) — це об'єкт веб-сторінки, що містить інформацію про адресу іншої веб-сторінки або про певне місце на поточній. Таким об'єктом може бути фрагмент тексту (зазвичай виділений кольором та підкресленням) або ілюстрація. У разі наведення на гіперпосилання вказівник миші набуває форми руки з витягнутим вказівним пальцем. Клацнувши лівою кнопкою миші, можна виконати перехід за гіперпосиланням. При цьому браузер завантажує веб-сторінку, яка міститься за адресою, зазначеною в посиланні. Ця веб-сторінка також може містити гіперпосилання, які вказують на інші веб-сторінки. Оскільки веб-сторінки можуть бути зв'язані між собою довільно, такий спосіб їх організації отримав назву Всесвітня павутина.

Процес переходу в інші місця поточної веб-сторінки або до інших веб-сторінок за допомогою гіперпосилань називають *навігацією*. Якщо після низки переходів за гіперпосиланнями необхідно повернутися на попередню сторінку, то користуються кнопкою **Назад** панелі інструментів браузера. Поруч із нею є стрілка для розкриття списку сторінок, що вже були переглянуті в цьому сеансі роботи; у списку можна вибрати потрібну сторінку і перейти до неї.

Сукупність веб-сторінок, що тематично пов'язані між собою й розроблені як єдине ціле, називають *веб-сайтом* або просто *сайтом*. Сторінки веб-сайту розміщуються на одному сервері та мають однакову адресу сайту, наприклад <http://blv.kiev.ua/>.

Веб-сайт може надавати як пасивну інформацію, що читається лише відвідувачем, так і активну, яку відвідувач може додавати або редагувати. Для організації інтерактивної взаємодії відвідувачів використовують гостьові книги, форуми, чати та блоги, які описані в цьому розділі далі.

Домашні сторінки

У 90-х роках минулого століття виник новий вид творчості — створення *домашніх сторінок*. Спочатку цей термін (від англ. home page) означав дім людини в Інтернеті, місце, де вона зберігає

інформацію про себе. Наразі особисті сторінки мають тисячі людей. Одні з них містять лише коротку інформацію про власника, а другі — корисну інформацію з певної тематики, графічний матеріал, фотографії тощо.

Термін «домашня сторінка» не має чіткого визначення, так називають також головну сторінку сайту, що відкривається у разі введення його доменного імені, та сторінку, яка завантажується під час кожного запуску браузера.

Структура веб-сайтів

Зовнішній вигляд кожного сайту є унікальним, проте в усіх сайтів можна знайти спільні за функціональністю частини. На будь-якому сайті першою відкривається *головна сторінка*. Її розробці приділяють особливу увагу, оскільки дослідження показали, що люди не здатні читати інформацію, що відображається на моніторі, так уважно, як книжки або журнали, вони зазвичай лише поверхово переглядають її, наприклад, як рекламу. Якщо головна сторінка містить те, що шукає відвідувач, він читає її далі, а якщо ні — переходить до інших сайтів, яких в Інтернеті дуже багато.

У верхній частині головної сторінки зазвичай розташована так звана шапка, яку дублюють на інших сторінках сайту. Це роблять спеціально, адже ця частина відображається у вікні браузера першою, і відвідувач насамперед звертає увагу на неї.

Щоб забезпечити швидкий перехід до основних тематичних розділів сайту, створюють *меню сайту* — список гіперпосилань на його розділи. Горизонтальне меню зазвичай розташовують у шапці, іноді дублюючи його в нижній частині сторінки, а вертикальне — переважно в лівій частині сторінки, у місці, звідки відвідувач починає її переглядати. Меню є одним із найважливіших компонентів сайту, користувач постійно звертає на нього увагу, і тому вимоги до нього високі. Меню має бути зручним, помітним і зрозумілим, інакше користувач не знатиме, як потрапити в потрібний розділ, і вийде із сайту. Пункти меню мають бути чітко відділені один від одного.

На рис. 1.1 показано головну сторінку сайту гімназії «Сихівська» м. Львова. Меню сайту на ній розміщене ліворуч.

На рис. 1.2 наведено приклад ще однієї головної сторінки — сайту Видавничої групи ВНУ. Меню цього сайту є горизонтальним.

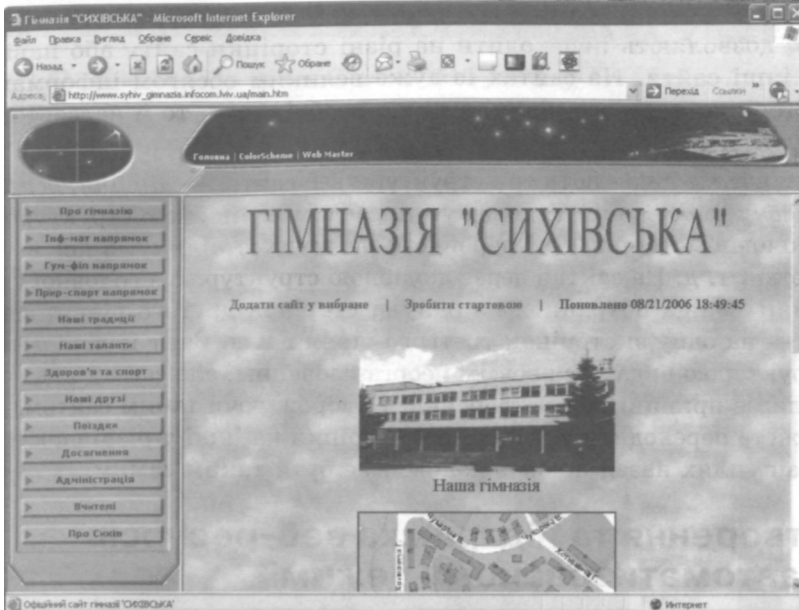


Рис. 1.1. Головна сторінка сайту гімназії «Сихівська»

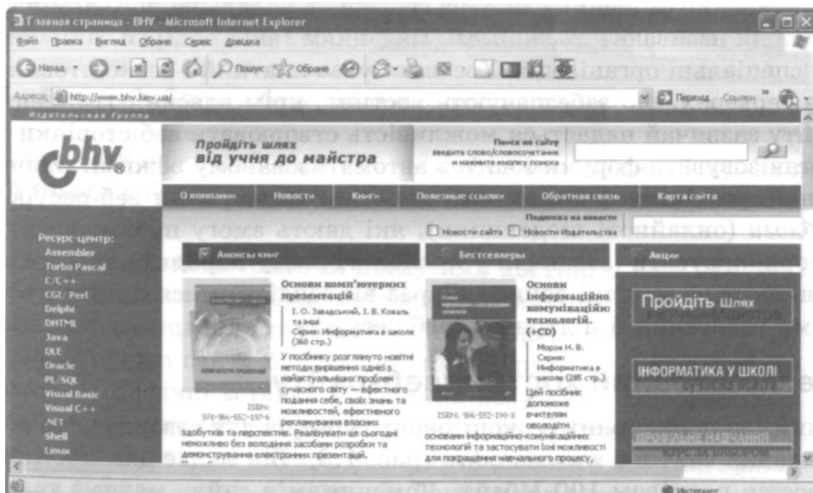


Рис. 1.2. Головна сторінка сайту Видавничої групи BNV

Гіперпосилання, розміщені в тексті чи у вигляді графічних об'єктів, дозволяють переходити на різні сторінки сайту або навіть на інші сайти. На сайтах із дуже великим обсягом інформації є сторінки третього рівня, а якщо необхідно — то й четвертого, п'ятого і т. д.

Загалом же виділяють три структури веб-сайтів — *лінійну*, *деревоподібну* та *довільну*. Подорожуючи сайтом із лінійною структурою, з головної сторінки ви перейдете на другу сторінку, з неї — на третю і т. д. На сайті з деревоподібною структурою з головної сторінки можна потрапити на одну зі сторінок другого рівня, звідти — на одну зі сторінок третього рівня і т. д. Сайт із довільною структурою видається зовсім неорганізованим, але саме у цьому й полягає принцип його створення. Подорожуючи таким сайтом, ви можете переходити з однієї його сторінки на інші у різні способи, і ваш шлях назад не обов'язково має бути таким самим.

Створення та підтримка веб-ресурсів в автоматизованому режимі

Для того щоб сайт став доступним широкому колу відвідувачів, йому необхідно призначити доменне ім'я і розмістити в мережі Інтернат. Розміщення сайту на сервері та подальше його адміністрування називають *хостингом*. Наданням такої послуги займаються спеціальні організації. Хостинг буває платний і безкоштовний. На серверах, що забезпечують хостинг, крім власне розміщення сайту зазвичай надається можливість створювати веб-сторінки та організовувати форуми і чати в автоматизованому режимі. Однією з найпростіших форм автоматизованого створення веб-ресурсів є *блоги* (онлайнві щоденники), які дають змогу публікувати та впорядковувати (зазвичай у хронологічному порядку) на веб-сторінках різноманітні записи. Зараз ви ознайомитеся з цими засобами створення й підтримки веб-ресурсів.

Безкоштовний хостинг веб-сайту

Одним із популярних безкоштовних серверів для створення та розміщення сайтів є www.narod.ru (рис. 1.3). Він надає для хостинга простір розміром 100 Мбайт. Доменне ім'я сайту матиме такий вигляд: назва_сайту.[narod.ru](http://www.narod.ru).

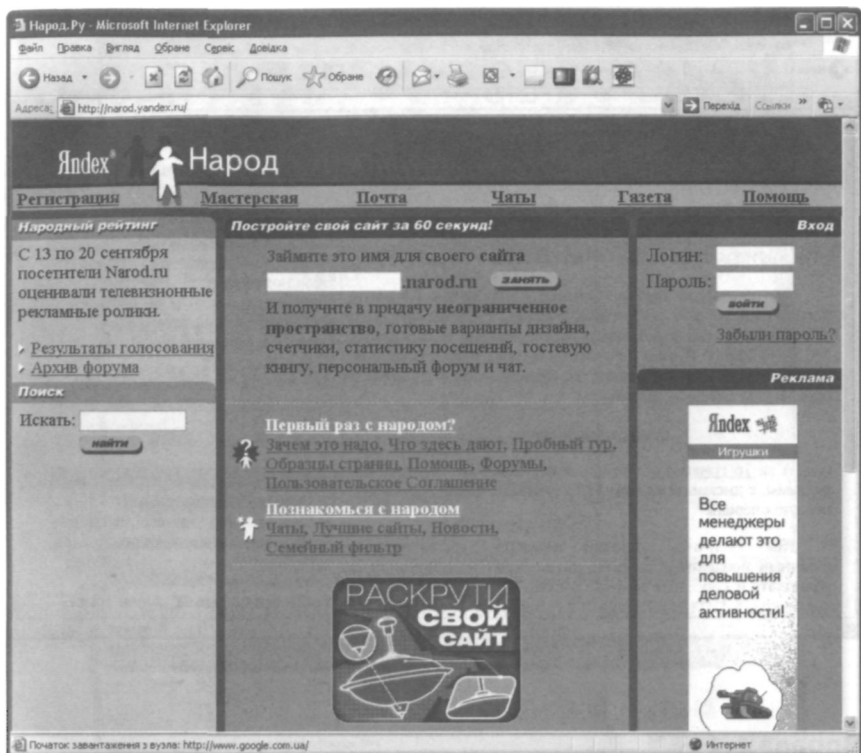


Рис. 1.3. Служба безкоштовного хостинга www.narod.ru

Ще один безкоштовний сервер — www.ho.com.ua (рис. 1.4). Ім'я майбутнього сайту на цьому сервері буде таким: назва_сайту.ho.com.ua.

Користуючись безкоштовними послугами хостинга, слід пам'ятати про те, що ваш сайт буде автоматично використаний як рекламний майданчик для організації, що надала послугу. Під час перегляду такого сайту будуть з'являтися різні спливаючі вікна або банери (графічні вставки невеликих розмірів) з гіперпосиланнями рекламного змісту. Іноді безкоштовний хостинг виявляється таким лише на певний період, по закінченні якого або слід заплатити, або сайт видалять із сервера. Тому перед реєстрацією сайту слід уважно прочитати угоду з користувачем.

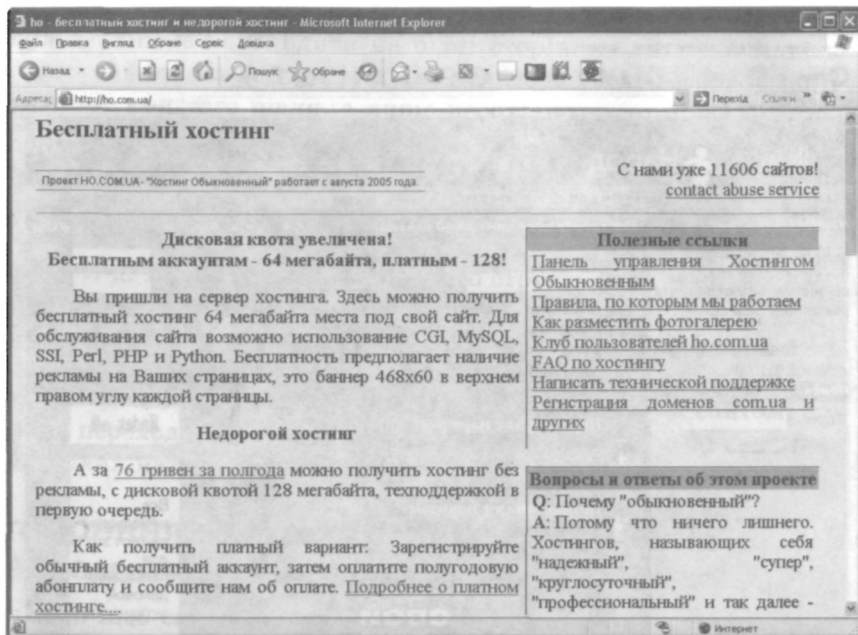


Рис. 1.4. Хостинг на сайті www.ho.com.ua

Вправа 1.1

Переглянемо сайт про тварин — домашніх улюбленців. Цей невеликий проект побудовано засобами автоматичного створення сайтів на безкоштовному сервері www.narod.ru.

1. Запустіть браузер, інстальований на вашому комп'ютері.
2. У рядку Адреса введіть адресу сайту про домашніх улюбленців — www.protvargyn.narod.ru.
3. Відкриється його головна сторінка (рис. 1.5). Розгляньте її уважно.

Головна сторінка поділена на частини: ліворуч міститься інформація про призначення сайту, а в центрі — гіперпосилання на інші його сторінки.

4. Скористайтеся посиланням Ця сторінка про серйозного папугу Річі та розгляньте сторінку про папужку (рис. 1.6).

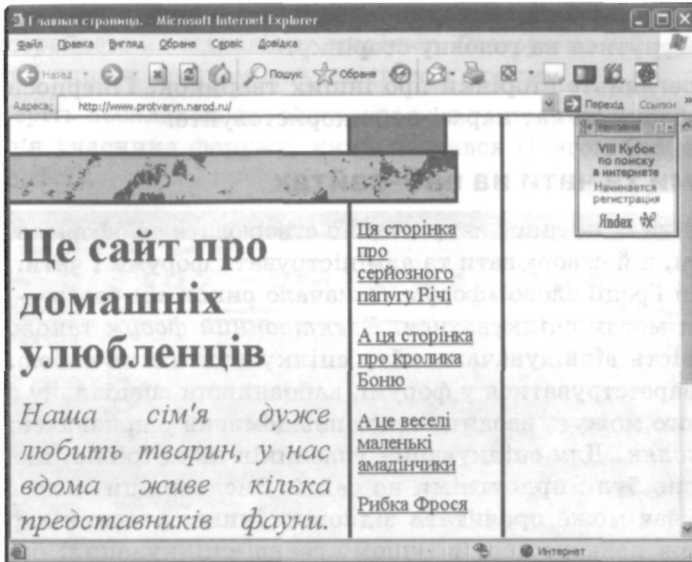


Рис. 1.5. Головна сторінка сайту protvaryn.narod.ru

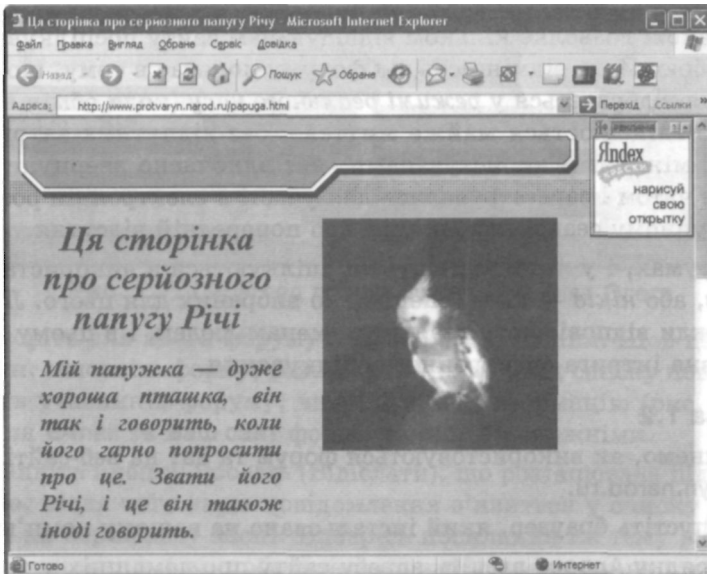


Рис. 1.6. Сторінка сайту, присвячена папужці

Створення та підтримка веб-ресурсів в автоматизованому режимі

5. Клацніть кнопку **Назад** на панелі інструментів браузера, щоб повернутися на головну сторінку.
6. Перегляньте сторінки про інших тваринок. Гіперпосилання на форум та чат наразі не використовуйте.

Форуми та чати на веб-сайтах

Є можливість не лише автоматично створювати та оформляти веб-сторінки, а й створювати та адмініструвати форуми і чати. У стародавній Греції слово «форум» означало ринок або базар — місце, де люди могли спілкуватися. *Електронний форум* також надає можливість відвідувачам сайту спілкуватися між собою. Вони мають зареєструватися у форумі, заповнивши спеціальну форму, після чого можуть вводити текст повідомлень у призначених для цього полях. Для спілкування учасників не потрібно, щоб вони одночасно були присутніми на сайті. Висловивши свою думку, користувач може прочитати відповідь інших учасників форуму упродовж певного часу в іншому сеансі спілкування, оскільки його інформація залишається на сайті й може довго бути доступною для перегляду.

Чат також дозволяє кільком відвідувачам сайту послідувати між собою. Його відмінність від форуму полягає в тому, що спілкування відбувається у *режимі реального часу* (коли обмін інформацією здійснюється майже миттєво, без відчутних затримок) і лише між тими користувачами, які одночасно звернулися до сайту з чатом і зареєструвалися для участі в електронній розмові. У наступному сеансі інформація про попередній відсутня.

І у форумах, і у чатах відвідувачі спілкуються з використанням *логінів*, або *ніків* — імен, спеціально вибраних для цього. Логіни не завжди відповідають реальним іменам людей, і в цьому полягає певна інтрига електронного спілкування.

Вправа 1.2

Розглянемо, як використовуються форум та чат на веб-сайті www.protvaryn.narod.ru.

1. Запустіть браузер, який інстальовано на вашому комп'ютері.
2. У рядку **Адреса** введіть адресу сайту про домашніх улюбленців — www.protvaryn.narod.ru.

На головній сторінці виберіть посилання Форум, після чого ви перейдете на відповідну сторінку. Ліворуч на ній розташований список повідомлень учасників форуму.

Виберіть посилання про кішку і кролика та прочитайте повідомлення учасника форуму, який назвався Олегом Семеновим (рис. 1.7).

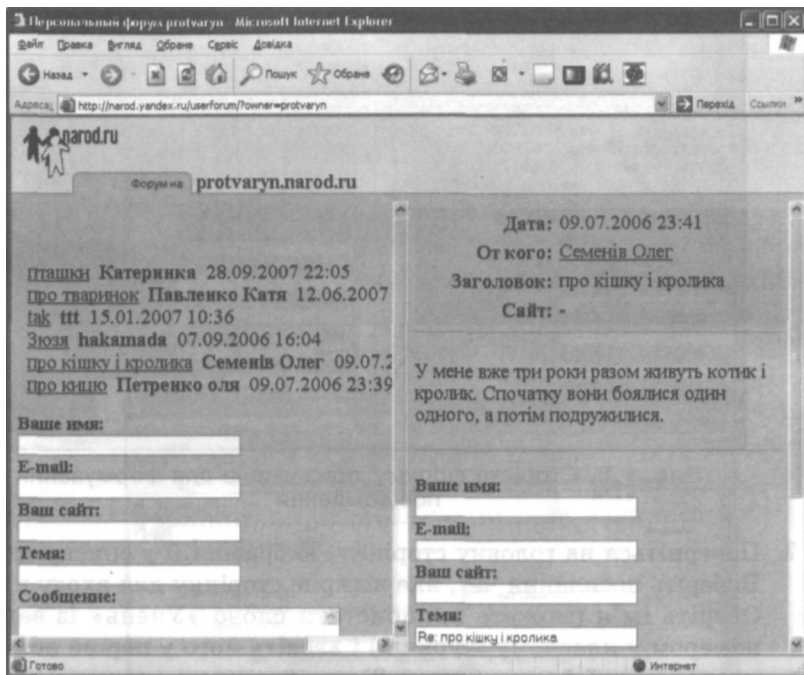


Рис. 1.7. Перегляд повідомлення Семенова Олега

Створіть для цього форуму власне повідомлення. Щоб це зробити, заповніть форму, розташовану нижче від списку повідомлень учасників форуму, ввівши у неї інформацію (рис. 1.8). Поля E-mail та Ваш сайт форми залиште порожніми.

Клацніть кнопку Послать (Відіслати), що розташована під формою, після чого ваше повідомлення з'явиться у списку ліворуч на верхньому місці. Виберіть посилання на тему вашого повідомлення, і ви побачите його зміст праворуч на сторінці (див. рис. 1.8).

Для створення відповіді на це повідомлення потрібно скористатися формою, що розташована під ним.

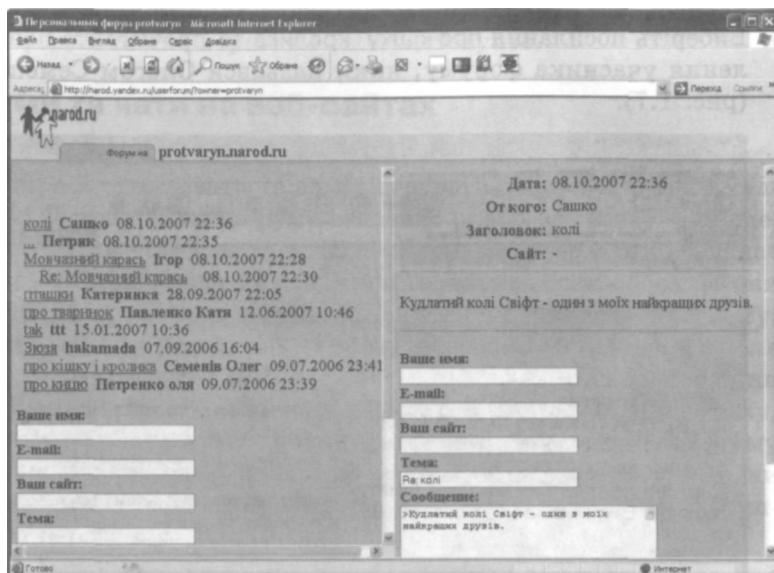


Рис. 1.8. Сторінка форуму, призначена для формування повідомлення

7. Поверніться на головну сторінку, вибравши її у списку адрес. Виберіть посилання Чат, яке відкриє сторінку для входу в чат. Оберіть ім'я (можете використати слово «Учень» із вашим номером у класному журналі) і введіть його у перше поле запропонованої форми (рис. 1.9).

8. Клацніть кнопку Войти в чат (Увійти в чат). На сторінці, що відкриється, праворуч міститиметься список присутніх учасників чату. Для участі у спільній розмові введіть текст (наприклад, «Привіт!») у поле, розташоване в нижній частині сторінки, і клацніть кнопку Отправить (Відіслати). Повідомлення буде відтворене у центральній частині сторінки.

Повідомлення у чат додаються постійно різними учасниками, одні учасники виходять із чату, а інші — долучаються до нього, і зрештою спільна розмова матиме такий вигляд, як на рис. 1.10.

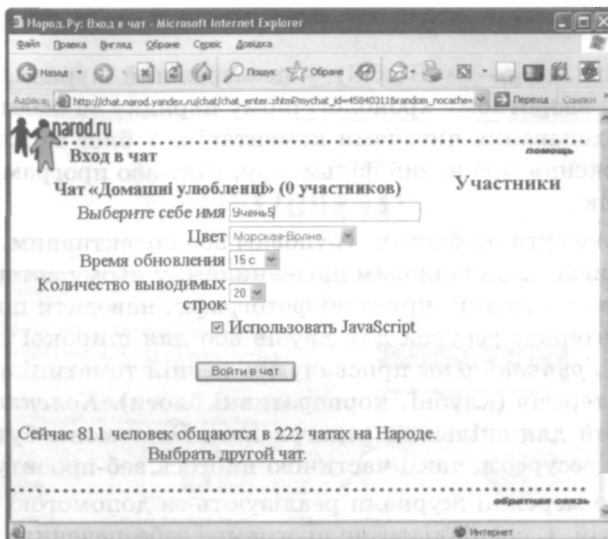


Рис. 1.9. Сторінка для реєстрації учасника чату



Рис. 1.10. Сторінка чату на сайті protvaryn.narod.ru

9. Щоб повернутися на головну сторінку сайту, клацніть кнопку **Выйти** (Вийти).

Створення та ведення блогів

Блог (англ. blog — weB LOG) — це мережний журнал, що містить записи у зворотному хронологічному порядку та забезпечує можливість додавання читачами коментарів. У блог можна записати свої враження про новий фільм, гру, сайт або програму, розповісти історію.

Блог може бути особистим, груповим або колективним. *Особистий блог* називають онлайнним щоденником, у ньому автор може публікувати свої думки, приватні фотографії, наводити посилання на цікаві інтернет-ресурси для друзів або для широкої інтернет-аудиторії. *Груповий блог* присвячують певній тематиці або спільній сфері інтересів (клубні, корпоративні блоги). *Колективний блог* відкритий для спільного редагування. Блог може бути як самостійним ресурсом, так і частиною якогось веб-проекту.

Технічно мережні журнали реалізують за допомогою різних веб-технологій. Є спеціалізоване програмне забезпечення (наприклад, WordPress, Pivot), що дає змогу людині, яка не володіє спеціальними знаннями з розробки веб-ресурсів, спростити створення, обслуговування і ведення блога.

З кінця 90-х років минулого століття все популярнішими стають безкоштовні служби, що надають можливість вести особистий блог за умови реєстрації на сайті, який надає такі послуги. Найвідомішими серед них є www.livejournal.com, www.blogger.com, www.ji.ru, www.rax.ru, а також українська служба blog.net.ua. Ресурс www.livejournal.com (рис. 1.11) надає кожному користувачу можливість вести власний журнал, який інтегрує блоги інших користувачів, організовує спілкування між ними.

Читачі можуть об'єднуватися в спільноти. *Спільнота* — це журнал, до якого пишуть різні користувачі, й усіх їх вважають його власниками. Кожен, хто цікавиться якоюсь конкретною темою, може знайти або створити спільноту, присвячену цій темі. Наприклад, мешканці одного міста можуть створити спільний журнал для обміну інформацією про це місто чи для оголошень про місцеві події.

Інформація блога орієнтована на широку аудиторію. Ще однією його особливістю є те, що вміст журналу унікальний, ніде не повторюється і тому може бути цінним.

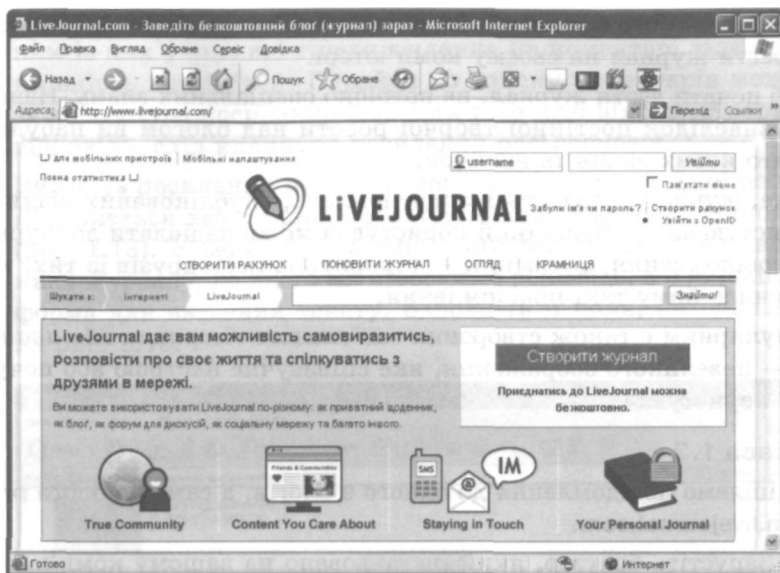


Рис. 1.11. Безкоштовна служба www.livejournal.com

Розміщуючи інформацію в журналі, необхідно дотримуватися певних правил. Зокрема, у блогу не можна:

- Закликати до агресії, ображати за національною, етнічною, географічною, політичною або релігійною ознаками.
- Створювати *флуд* (багато однакових або беззмістовних повідомлень) у журналах, спільнотах та коментарях, розповсюджувати повідомлення рекламного характеру.
- Публікувати зображення, вміст яких не відповідає загальноприйнятим нормам і є неприємним для більшості користувачів.
- Розміщувати у повідомленнях, коментарях чи фотоальбомах зображення, якщо люди, котрі на них зображені, або особи, що мають на них права, не бажають цього.
- Приписувати собі авторство чужих текстів і зображень, вести щоденник від імені іншої особи.

Власник блога може визначати права доступу користувачів до свого журналу та окремих його записів, а також видаляти блог та

переводити його в режим спільноти. Надається також можливість зберегти журнал на своєму комп'ютері.

Щоб почати вести журнал, не потрібно спеціальних знань. Навпаки, внаслідок постійної творчої роботи над блогом ви набудете **багато нових** знань та навичок.

Отже, журнал складається з повідомлень, опублікованих різними користувачами. Будь-який користувач може надіслати до журналу повідомлення, а власник — створити список друзів із тих, хто надсилає йому такі повідомлення.

Популярним є також створення так званого *юзерніка* або *аватара* — невеликого зображення, яке співзвучне настрою або почуттям користувача.

Вправа 1.3

Надішлемо повідомлення до одного з блогів, а саме до блога **protvaryn.livejournal.com**.

1. Запустіть браузер, який інстальовано на вашому комп'ютері.
2. У рядку **Адреса** введіть адресу блога — **http://protvaryn.livejournal.com/**. Ви побачите таку сторінку, як на рис. 1.12.

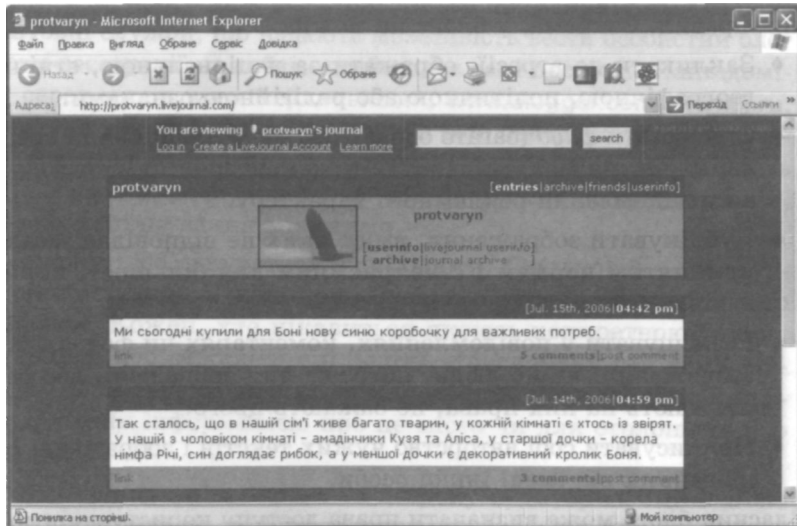


Рис. 1.12. Головна сторінка блога protvaryn.livejournal.com

На ній містяться повідомлення власника журналу; під кожним із них у вигляді гіперпосилання на коментарі зазначено кількість відповідей. Перейти до списку коментарів можна за допомогою посилання link. Для того щоб відразу створити відповідь, слід клацнути посилання post comment.

Виберіть посилання post comment для першого повідомлення. Відкриється веб-сторінка з формою для створення відповіді (рис. 1.13). У верхній її частині міститься повідомлення власника журналу, до якого ми створюємо відповідь, а в нижній — форма для введення тексту. Заповніть її поля, як показано на рис. 1.13.

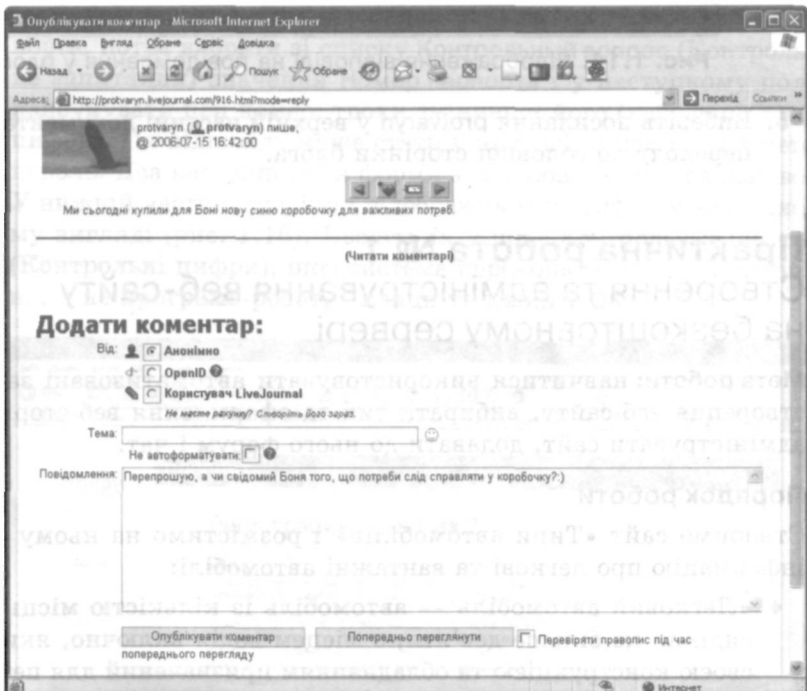


Рис. 1.13. Створення відповіді на повідомлення у блогу

4. Клацніть кнопку Опублікувати коментар. Ваша відповідь з'явиться у списку коментарів після останнього повідомлення (рис. 1.14).

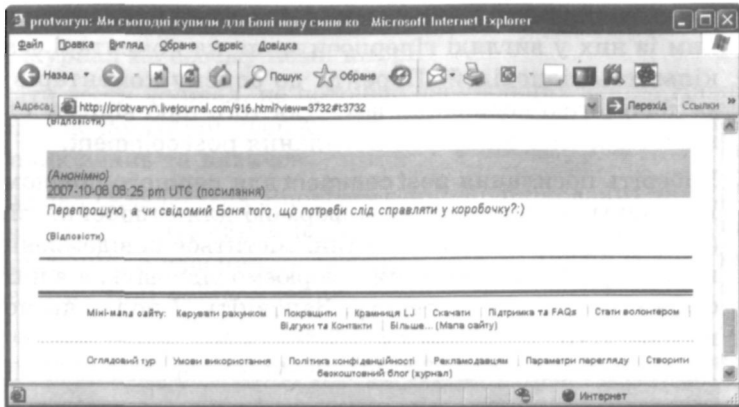


Рис. 1.14. Відображення відповіді на повідомлення у блогу

5. Виберіть посилання protvayun у верхній частині документа для переходу до головної сторінки блога.

Практична робота № 1

Створення та адміністрування веб-сайту на безкоштовному сервері

Мета роботи: навчитися використовувати автоматизовані засоби створення веб-сайту, вибирати тип та оформлення веб-сторінок, адмініструвати сайт, додавати до нього форум і чат.

Порядок роботи

Створимо сайт «Типи автомобілів» і розмістимо на ньому таку інформацію про легкові та вантажні автомобілі:

- «Легковий автомобіль — автомобіль із кількістю місць для сидіння не більше дев'яти, з місцем водія включно, який за своєю конструкцією та обладнанням призначений для перевезення пасажирів та їхнього багажу із забезпеченням необхідного комфорту та безпеки».
- «Вантажний автомобіль — автомобіль, який за своєю конструкцією та обладнанням призначений для перевезення вантажів».

1. Запустіть браузер, який інстальовано на вашому комп'ютері.
2. У рядку Адреса введіть www.narod.ru. Відкриється головна сторінка служби (див. рис. 1.3).
3. У поле Займите это имя для своего сайта (Займіть це ім'я для свого сайту) введіть свій майбутній логін. Його можна визначити так: до назви міста додати перші кілька літер вашого прізвища, наприклад kuivraiv.
4. Клацніть кнопку Занять (Зайняти) та заповніть відповідну форму (рис. 1.15).
5. Клацніть кнопку Дальше (Далі). У наступній формі введіть пароль у два запропонованих рядки (другий раз — для контролю правильності). Пароль можна задати як номер свого паспорта, потім вибрати зі списку Контрольный вопрос (Контрольне запитання) значення Номер паспорта і у наступному полі надрукувати цей номер, трохи змінивши його (набравши самі цифри). Ця відповідь може стати у пригоді, якщо ви забудете пароль. Два наступні поля форми заповнювати не обов'язково. У нижній частині сторінки відображаються цифри у незвичному вигляді (рис. 1.16). Введіть їх у поле Контрольные цифры (Контрольні цифри), щоб система переконалася, що ви людина, а не програма-робот, і клацніть кнопку ОК.

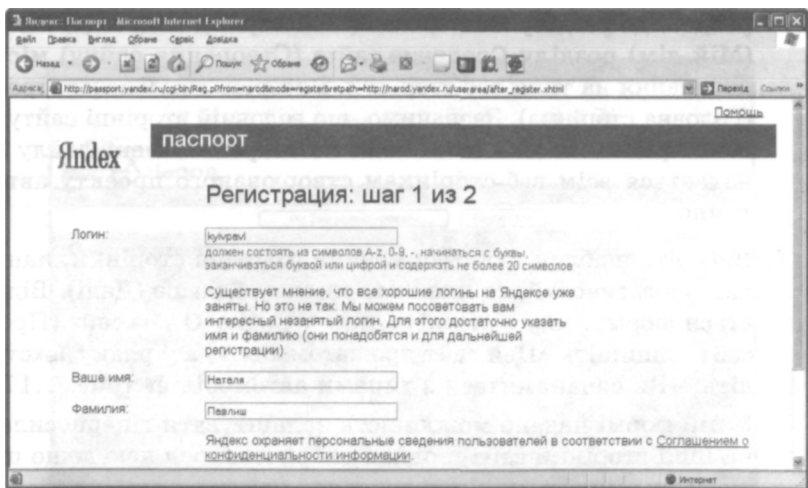


Рис. 1.15. Форма для створення сайту (крок 1)

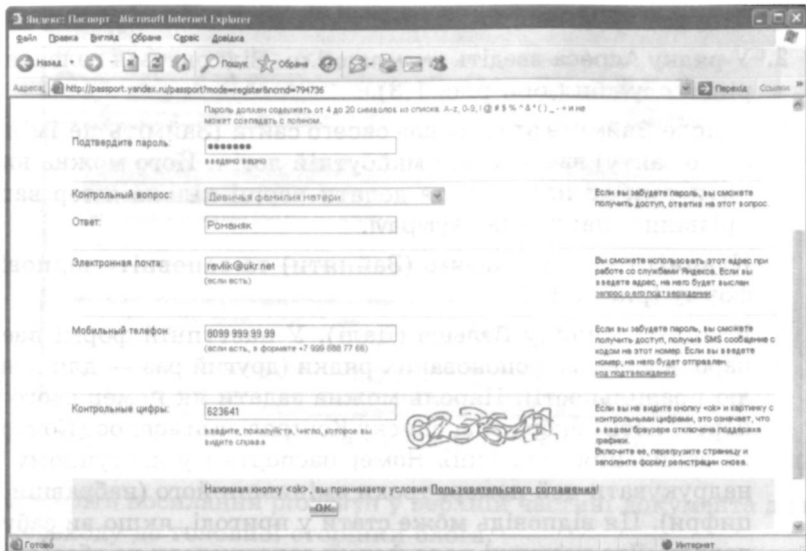


Рис. 1.16. Форма для створення сайту (крок 2)

На наступній сторінці міститься повідомлення, що реєстрація відбулася. Виберіть посилання **Начать пользоваться сервисом** (Почати користуватися сервісом). Після його клацання ви перейдете до розділу **Мастерская** (Майстерня). Підрозділ **Мой дом** (Мій дім) розділу **Создание сайта** (Створення сайту) містить посилання на типи веб-сторінок. Виберіть тип **Главная страница** (Головна сторінка). Зазначимо, що головній сторінці сайту система призначає ім'я `index.html`. Розширення імені файлу `.html` надається всім веб-сторінкам створюваного проекту автоматично.

Виберіть шаблон для оформлення головної сторінки, наприклад **Апельсиновый**, та клацніть кнопку **Дальше** (Далі). Відкриється форма, яку заповніть так: у рядку **О чём сайт** (Про що сайт) напишіть «Цей сайт про автомобілі», а у рядку **Текст введіть**: «Ви ознайомитеся з типами автомобілів» (рис. 1.17).

У цій формі надано можливість налаштувати гіперпосилання на інші сторінки сайту, однак скористаємося нею дещо пізніше, адже інших сторінок у нас ще немає.

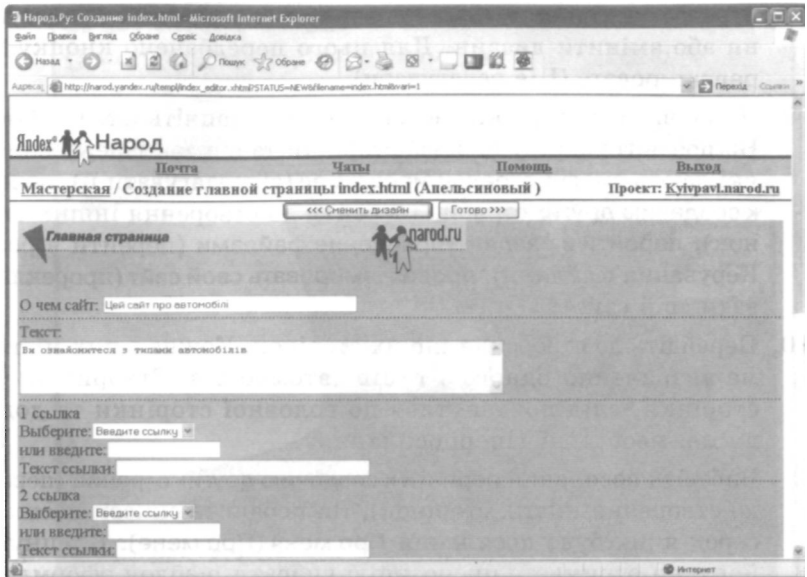


Рис. 1.17. Заповнення інформацією головної сторінки майбутнього сайту

8. Кладніть кнопку Готово, і ви побачите, який вигляд матиме головна сторінка (рис. 1.18).

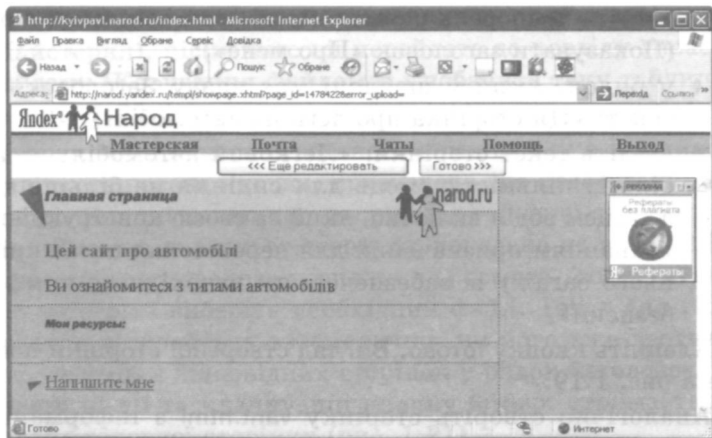


Рис. 1.18. Попередній вигляд головної сторінки

У разі необхідності можна повернутися до режиму редагування або змінити дизайн. Для цього передбачено кнопку Еше редагувати (Ще редагувати).

9. Якщо вигляд сторінки вас влаштовує, клацніть кнопку Готово. Ви побачите адресу цієї веб-сторінки та список гіперпосилань, які дають змогу: отредагувати її (відредагувати її), перейти к созданию других страниц (перейти до створення інших сторінок), перейти в раздел Управление файлами (перейти в розділ Керування файлами), прорекламувати свій сайт (прорекламувати свій сайт).
10. Перейдіть до створення інших сторінок. Кожна з них міститиме визначення одного з типів автомобілів. Створивши такі сторінки, слід повернутися до головної сторінки та додати до неї необхідні гіперпосилання.

Виберіть посилання перейти к созданию других страниц (перейти до створення інших сторінок). Ви побачите список посилань, серед яких буде посилання Про меня (Про мене). Клацнувши його, ви отримаєте пропозицію вибрати шаблон оформлення сторінки. Виберіть Спокойный (Спокійний). Система запропонує назву файлу — simple.html. Можна залишити її, але краще дати свою. Оскільки ця сторінка містить визначення легкового автомобіля, назвемо файл legk.html. Сформууйте сторінку, виконавши такі дії:

- зніміть прапорець у рядку Показывать заголовок «Про меня» (Показувати заголовок «Про мене»);
- у рядок, який у шаблоні має назву Заголовок, введіть текст: «Ця сторінка про легкові автомобілі»;
- введіть текст сторінки: «Легковий автомобіль — автомобіль із кількістю місць для сидіння не більше дев'яти, з місцем водія включно, який за своєю конструкцією та обладнанням призначений для перевезення пасажирів та їхнього багажу із забезпеченням необхідного комфорту та безпеки».

Клацніть кнопку Готово. Вигляд створеної сторінки показаний на рис. 1.19.

11. Аналогічно створіть сторінку vant.html з інформацією про вантажні автомобілі.

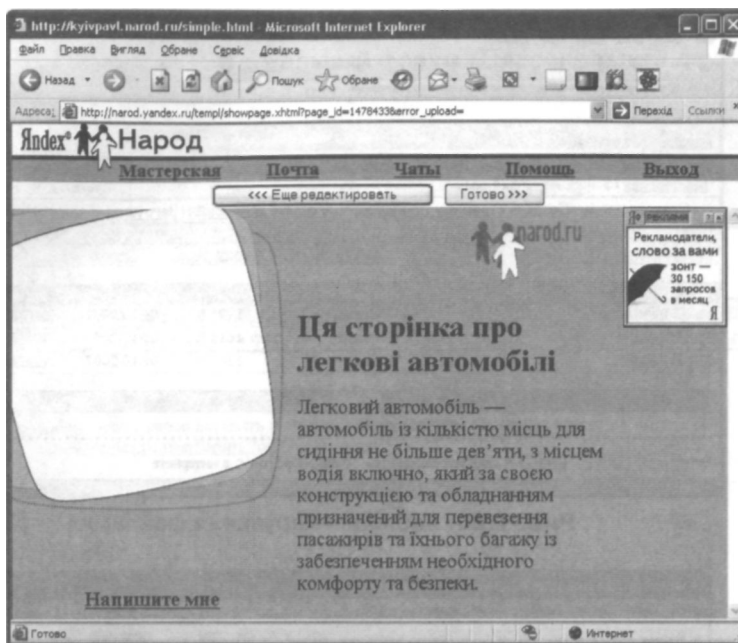


Рис. 1.19. Попередній вигляд сторінки про легкові автомобілі

12. Залишилось відредагувати головну сторінку, яка має назву index.html. До неї слід додати два посилання — на кожну сторінку про машини. Виберіть посилання Мастерская (Майстерня), після чого ви перейдете до розділу Управление файлами (Керування файлами) і побачите таку таблицю, як на рис. 1.20.

Ліворуч від кожного створеного нами файлу з розширенням .html розташована кнопка, призначена для редагування вмісту файлу. Клацніть кнопку ліворуч від головної сторінки index.html. Знову відкриється сторінка, показана на рис. 1.17. Для кожного створюваного посилання (1 ссылка, 2 ссылка) розгорніть список і виберіть необхідний файл. Текст для гіперпосилання з'являється автоматично, ми його вже вводили під час створення відповідних сторінок у рядок Заголовок. Виконавши ці дії та клацнувши кнопку Готово, отримаєте новий вигляд головної сторінки (рис. 1.21).

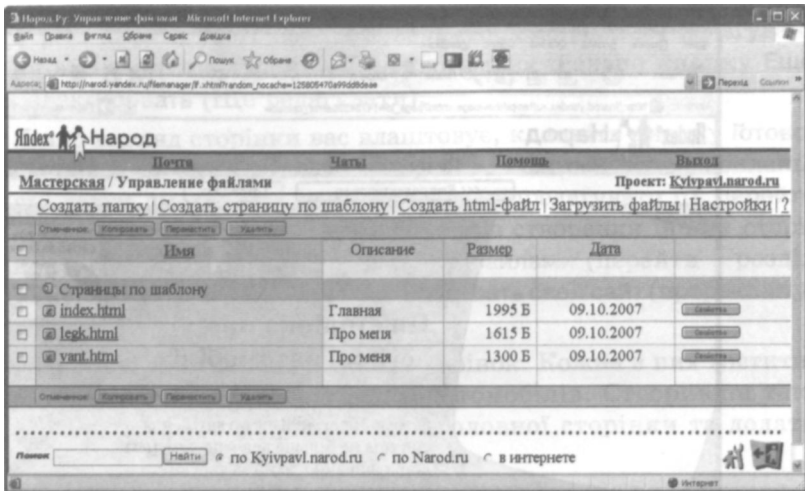


Рис. 1.20. Таблица «Керування файлами»

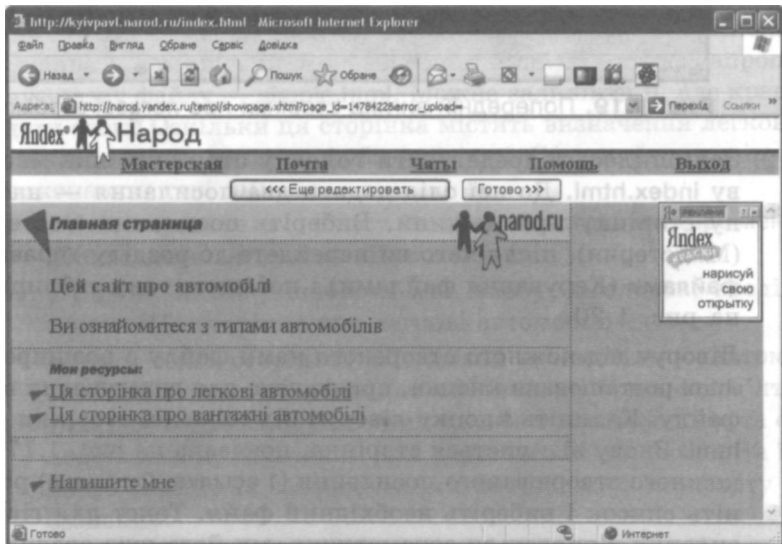


Рис. 1.21. Головна сторінка з гіперпосиланнями

13. Додайте форум до вашого сайту про автомобілі. Для цього увійдіть у «майстерню», скориставшись посиланням Мастер-

ская (Майстерня), і в розділі Общение (Спілкування) виберіть посилання Персональный форум (Персональний форум), після чого відкриється сторінка, показана на рис. 1.22.

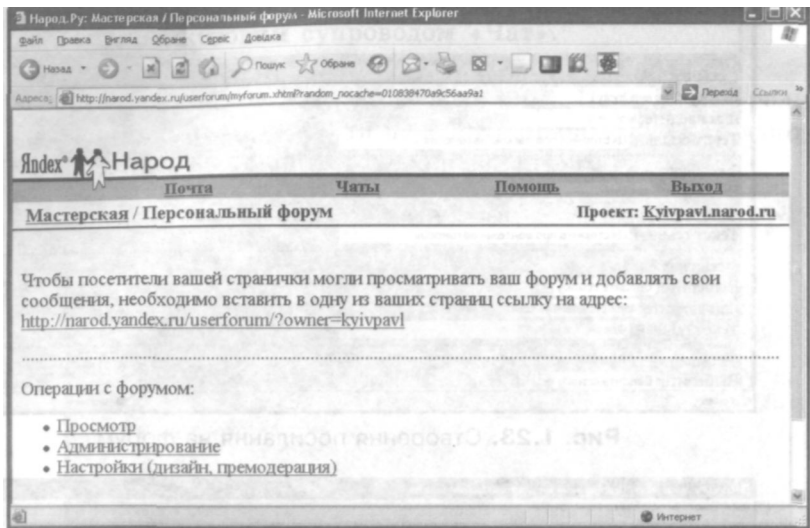


Рис. 1.22. Сторінка для створення форуму

14. Скопіюйте в буфер адресу <http://narod.yandex.ru/userforum/?owner=kyivpravl>, що міститься на цій веб-сторінці.
15. Перейдіть спочатку до розділу Мастерская (Майстерня), а потім у розділі Управление файлами (Керування файлами) виберіть для редагування файл index.html.
16. Виконавши описані у пункті 12 дії, додайте до головної сторінки посилання на зазначену адресу. Для цього вставте скопійовану адресу з буфера обміну в поле или введіть (або введіть) третього гіперпосилання (3 ссылка). Текстовий супровід посилання буде простим — Форум (рис. 1.23).
17. Завершивши редагування, клацніть кнопку Готово.
18. Аналогічно до форумів створюють чати. Увійдіть у «майстерню», скориставшись посиланням Мастерская (Майстерня), у розділі Общение (Спілкування) виберіть посилання Персональный чат (Персональний чат). Після цього відкриється сторінка, показана на рис. 1.24.

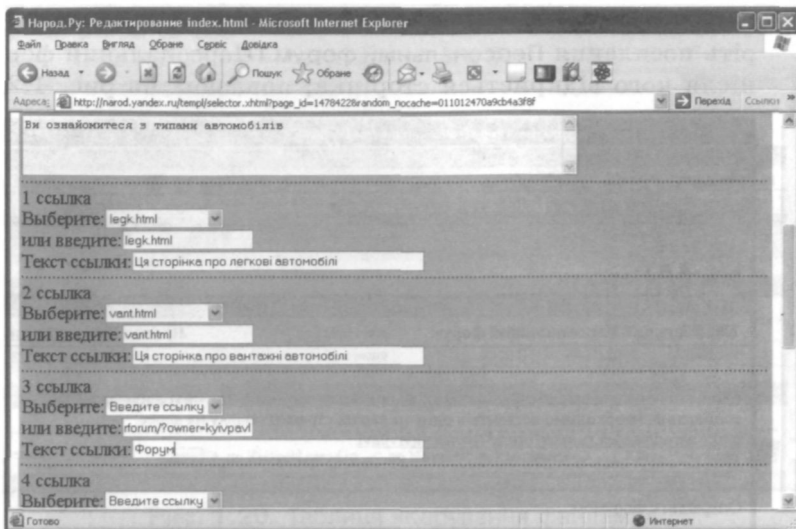


Рис. 1.23. Створення посилання на форум

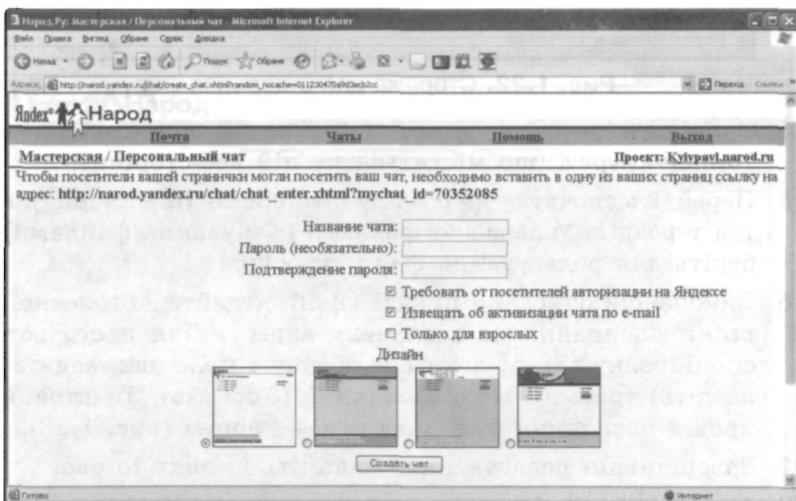


Рис. 1.24. Сторінка для створення чату

19. Скопіюйте в буфер обміну запропоновану адресу (наприклад, http://narod.yandex.ru/chat/chat_enter.shtml?mychatid=70352085), та введіть назву чату — «Про автомобілі».

20. Перейдіть спочатку в розділ Мастерская (Майстерня), а потім у розділ Управление файлами (Керування файлами), виберіть для редагування файл index.html і відредагуйте головну сторінку, додавши до неї четверте посилання на зазначену адресу з текстовим супроводом «Чат».
21. Введіть у рядок Адреса браузера адресу створеного сайту (наприклад, <http://www.kyivpavl.narod.ru/>). Перегляньте головну сторінку (рис. 1.25) та перевірте правильність її роботи, скориставшись різними гіперпосиланнями.

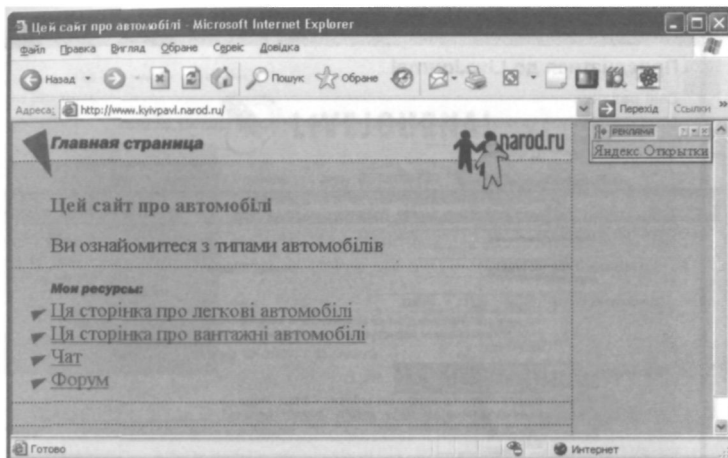


Рис. 1.25. Головна сторінка сайту www.kyivpavl.narod.ru

22. Оформіть висновок до практичної роботи, описавши при цьому створений сайт, його веб-сторінки та можливості форуму і чату.

Практична робота № 2

Створення й ведення блога

Мета роботи: навчитися створювати журнал-блог та розміщувати повідомлення у ньому.

Порядок роботи

Створимо блог за допомогою служби www.livejournal.com.

1. Запустіть браузер, який інстальовано на вашому комп'ютері. У рядку Адреса введіть <http://www.livejournal.com/>. Відкриється

головна сторінка служби безкоштовного ведення блогів LiveJournal (див. рис. 1.11).

2. Клацніть кнопку Створити журнал та заповніть запропоновану форму (рис. 1.26).

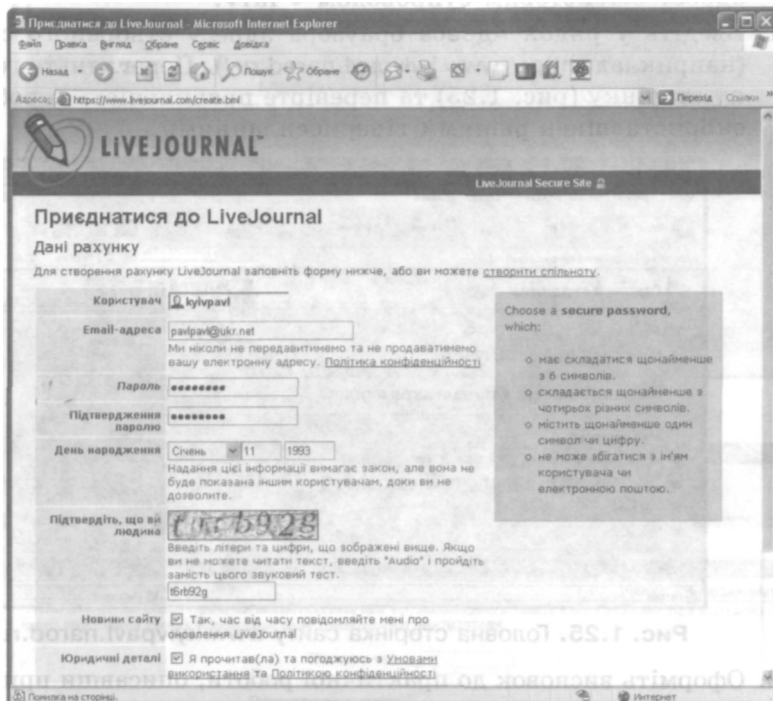


Рис. 1.26. Заповнення форми для створення блога

Клацніть кнопку Create Account (Створити рахунок), оскільки створення блога тут називають *створенням рахунку*. Виберіть рахунок Basic (Основний), клацнувши однойменне посилання. У наступній формі зазначте Gender (Стать), введіть інформацію в поля Bio (Про себе) і Interests (Інтереси) та клацніть кнопку Save and continue (Зберегти та продовжити).

На наступному кроці виконайте пошук друзів за інтересом. Для цього у списку Шукати за виберіть значення Інтерес і у поле нижче введіть слово «автомобілі». Клацніть кнопку Вперед та розгляньте наступну сторінку. На ній містяться список

спільнот, які цікавляться автомобілями, та посилання, що надають можливість переглянути список користувачів, які мають такий самий інтерес, і за бажанням долучитися до нього.

- У верхній частині сторінки є посилання на розділи новоствореного блога. Клацніть посилання Написати та надрукуйте коротке повідомлення у журнал (рис. 1.27).

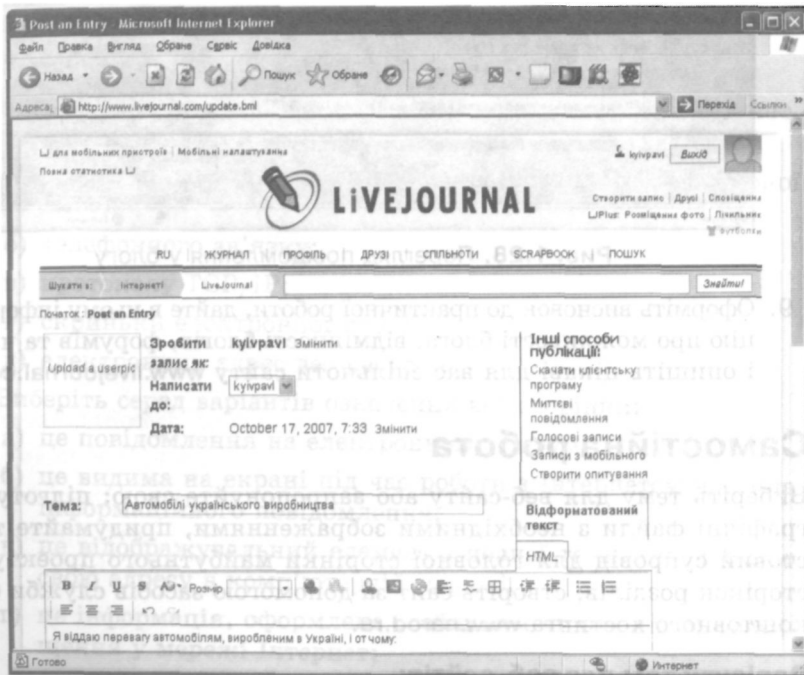


Рис. 1.27. Створення повідомлення

- Клацніть кнопку Опублікувати в ім'я користувача. У меню Журнал виберіть команду Дивитись журнал та перегляньте повідомлення у вашому блогу (рис. 1.28). Його побачить будь-який користувач, що відкриє сторінку за адресою на кшталт такої: <http://kyivpavl.livejournal.com/>.
- Щоб повернутися на попередню сторінку клацніть кнопку Назад на панелі інструментів браузера. Виберіть посилання Вихід та вийдіть із блога.

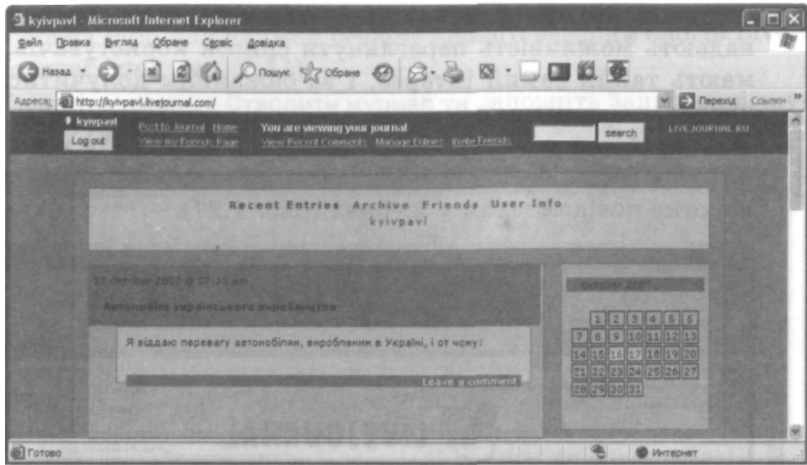


Рис. 1.28. Перегляд повідомлення у блогу

- Оформіть висновок до практичної роботи, дайте в ньому інформацію про можливості блога, відмінності блогів, форумів та чатів і опишіть цікаві для вас спільноти сайту www.livejournal.com.

Самостійна робота

Виберіть тему для веб-сайту або запропонуйте свою; підготуйте графічні файли з необхідними зображеннями, придумайте текстовий супровід для головної сторінки майбутнього проекту та сторінок розділів; створіть сайт за допомогою засобів служби безкоштовного хостинга www.narod.ru.

Варіанти тем для веб-сайтів:

Моделі одягу	Сузір'я	Автомобілі
Моя школа	Туризм	Телефони
Кондитерські вироби	Зачіски	Прапори
Сучасні комп'ютери	Кінофільми	Візерунки
Меблі	Страви	Птахи
Побутова техніка	Комп'ютери	Квіти
Канцелярські товари	Фотоапарати	Годинники
Книжки	Шкільні предмети	

Тест

1. URL-адреса — це:
 - а) адреса ресурсу в Інтернеті;
 - б) назва веб-сайту;
 - в) адреса комп'ютера, що підключений до мережі Інтернет;
 - г) адреса сайту для хостинга;
 - д) адреса електронної пошти.
2. Керування інформацією, що розміщена у мережі Інтернет, здійснюється за допомогою:
 - а) особи, відповідальної за роботу всесвітньої комп'ютерної мережі;
 - б) телефонного зв'язку;
 - в) протоколу TCP/IP;
 - г) скриньки електронної пошти;
 - д) електронних адрес веб-ресурсів.
3. Виберіть серед варіантів означення веб-сторінки:
 - а) це повідомлення на електронному форумі;
 - б) це видима на екрані під час роботи з Інтернетом частина інформаційного повідомлення;
 - в) це відображувальний елемент з певної тематики, що має свою адресу в комп'ютерній мережі;
 - г) це інформація, оформлена особливим чином для її розміщення у мережі Інтернет;
 - д) це повідомлення в електронній скриньці.
4. Гіперпосилання — це:
 - а) відформатований особливим чином текст на веб-сторінці;
 - б) зображення на веб-сторінці;
 - в) об'єкт веб-сторінки, що містить інформацію про адресу іншої веб-сторінки або про певну позицію на поточній веб-сторінці;
 - г) зв'язок між окремими розділами веб-сторінки;
 - д) адреса іншої веб-сторінки, що пов'язана з поточною.

5. Веб-сайт — це:
- а) будь-яка інформація, розміщена в комп'ютерній мережі для вільного доступу до неї;
 - б) оформлена за допомогою спеціальних засобів інформація, що поділена на веб-сторінки та має свою URL-адресу;
 - в) спілкування за допомогою електронної пошти;
 - г) сукупність веб-сторінок, розміщених під однією спільною назвою веб-ресурсу;
 - д) текстова та графічна інформація, до якої можуть отримувати спільний доступ користувачі комп'ютерної мережі.
6. Вкажіть особливості структури веб-сайту:
- а) веб-сторінки розміщені у веб-сайті так само, як у текстовому документі;
 - б) інформація організована за допомогою гіперпосилань таким чином, що перетворюється в ієрархічну систему веб-сторінок;
 - в) всі веб-сторінки з'єднані з головною за зіркоподібною схемою;
 - г) сайт містить одну веб-сторінку;
 - д) структура сайту передбачає використання гіперпосилань для зв'язку між сторінками.
7. Домашня сторінка — це:
- а) веб-сайт про сім'ю;
 - б) веб-сайт, присвячений певній тематиці;
 - в) сторінка зі змістом веб-сайту;
 - г) сторінка сайту, що відкривається першою у разі звертання до веб-ресурсу;
 - д) сторінка, що має назву index.html.
8. Флуд — це:
- а) повідомлення електронної пошти рекламного характеру;
 - б) будь-які повідомлення в чаті;

- в) беззмістовні повідомлення в журналах та чатах;
 - г) один з різновидів блогів;
 - д) текст, авторство якого присвоєне незаконно.
9. Головна сторінка сайту — це:
- а) сторінка, де розташована найбільша кількість гіперпосилань;
 - б) сторінка, на якій подані координати та адреса електронної пошти адміністратора сайту;
 - в) сторінка, яка відкривається першою під час завантаження сайту;
 - г) сторінка, що має назву index.html;
 - д) сторінка з ілюстраціями.
10. Електронний форум — це:
- а) процес обміну повідомленнями з використанням електронної пошти;
 - б) спосіб організації обміну повідомленнями з певної тематики, в якому спілкування може бути розірване в часі;
 - в) спосіб організації спілкування в реальному часі;
 - г) обмін повідомленнями в межах одного сайту різними користувачами у різний час;
 - д) повідомлення на різних сайтах зі спільної тематики.
11. Чат — це:
- а) спосіб організації спілкування зареєстрованих учасників у режимі реального часу;
 - б) створення коментарів до вмісту веб-сторінок;
 - в) спосіб організації обміну повідомленнями, за якого спілкування розірване в часі;
 - г) спілкування двох учасників між собою в межах одного сайту в режимі реального часу;
 - д) спілкування багатьох учасників між собою в межах одного сайту в режимі реального часу.

12. Блог — це:

- а) форум, у якому є один головний дописувач;
- б) мережний журнал, що складається із записів у зворотному хронологічному порядку з можливістю додавання коментарів;
- в) обмін повідомленнями з конкретної тематики в режимі реального часу;
- г) рекламний сайт;
- д) сайт для дистанційного навчання.

Розділ 2

Основи мови HTML

Ви вже переконалися, що за допомогою автоматизованих засобів, які надають у розпорядження користувача служби безкоштовного хостинга, створювати й адмініструвати веб-сайти надзвичайно просто. Крім того, це не потребує особливих знань із веб-дизайну. Однак у такому випадку ваші можливості обмежені, і ви не зможете створити сайт за своїми уподобаннями. Щоб мати змогу це зробити, потрібно опанувати спеціальну мову, призначену для розробки веб-сторінок. Тоді перед вами відкриються захоплюючі можливості щодо створення свого власного віртуального світу!

У цьому розділі ви ознайомитеся із принципами створення веб-сторінок за допомогою мови HTML, навчитесь визначати структуру сторінок, розміщувати і формувати текст, працювати з таблицями та фреймами, пов'язувати створені файли за допомогою гіперпосилань. Виконавши дві запропоновані практичні роботи, ви власноруч створите веб-сторінки — можливо, перші у вашому житті.

Базові конструкції мови HTML

Основною перевагою та особливістю веб-сторінок є те, що інформація, розміщена на них, має вигляд гіпертексту. *Гіпертекст* — це текст, у який вбудовані спеціальні коди, що задають форматування тексту, наявність у ньому ілюстрацій, мультимедійних вставок та гіперпосилань. Ці коди визначені у мові програмування *HTML* (Hyper Text Markup Language — мова гіпертекстової розмітки). Тобто веб-сторінка фактично є документом у форматі HTML. Процес вставлення в текст кодів HTML називають *розміткою*.

Браузери підтримують мову HTML і під час завантаження веб-сторінки читають наявні у ній коди та виконують задані за їх допомогою операції (форматують текст, вставляють у нього зображення

тошо). Таким чином, вигляд сторінки у вікні браузера залежить від того, які коди містяться у її документі.

Поняття тегу

Коди мови HTML, за допомогою яких розмічають вихідний текст, називають *тегами*. Під час відображення документа в браузері самих тегів не видно, але вони впливають на зовнішній вигляд документа. Усі теги починаються із символу < і закінчуються символом > — їх називають *кутовими дужками*. Після відкритої кутової дужки розміщують ключове слово, яке вказує на призначення тегу. Регістр у назвах тегів не має значення, хоча загальноживаними є великі літери. Приклади тегів HTML: <TITLE>, <BODY>, <TABLE>, , , </CENTER>.

Зазвичай тег впливає на певний фрагмент документа, наприклад на абзац. У таких випадках використовують *пару тегів* (відкривальний і закривальний). Перший із них задає ефект, а другий — припиняє його дію. Закривальний тег починається із символу / (слеш). Приклади парних тегів: <HTML></HTML>, , <HEAD></HEAD>, <H3></H3>, <ADDRESS></ADDRESS>, .

Деякі теги задають разовий ефект у місці своєї появи, тоді закривальний тег не потрібен. Приклади одинарних тегів:
, <HR>, <META>, <BASEFONT>, <FRAME>, <INPUT>.

Якщо між кутовими дужками помилково вказане ключове слово, яке відсутнє в мові HTML, то весь тег буде проігноровано.

Відкривальні теги можуть містити атрибути, які впливають на ефект, створюваний тегом. *Атрибути* — це додаткові ключові слова, відокремлені від основного ключового слова тегу та один від одного символами пробілу.

Для деяких атрибутів слід задавати значення, яке відділяють від імені атрибута символом =. Значення атрибута беруть у лапки, але в багатьох випадках їх можна і не ставити. Закривальні теги атрибутів не містять.

Приклади тегів з атрибутами:

```
<BODY BGCOLOR="#000000" TEXT="#FFFFFF"  
BACKGROUND="rain.gif">
```

```
<OPTION SELECTED>  
<FRAME SCR="file.html" NORESIZE>
```

Для того щоб через деякий час ви або інша людина, яка переглядатиме код HTML, могла краще його зрозуміти, у текст вставляють примітки — *коментарі*. Вони починаються зі спеціального тегу `<! —`. Будь-який записаний після нього текст браузер розглядає як коментар і не відображає під час відтворення документа. Закінчують коментар тегом `—>`. Коментар може містити будь-які символи, крім `>`, а отже, не може включати в себе теги.

Структура HTML-документа

HTML-документ складається з основного тексту і тегів розмітки. Фактично це звичайний текстовий файл, для створення і редагування якого можна використати будь-який текстовий редактор. Зазвичай HTML-документи містяться у файлах із розширенням `.htm` або `.html`.

Документ HTML має чітко визначену структуру.

- Починається з тегу `<HTML>` і закінчується відповідним йому тегом `</HTML>`, Така пара тегів повідомляє браузеру, що це HTML-документ.
- Містить два розділи — заголовків і тіла документа (розміщуються саме в такому порядку). Розділ заголовків помічений тегами `<HEAD>` і `</HEAD>` та містить Інформацію про документ загалом. Зокрема, він повинен містити в собі теги `<TITLE>` та `</TITLE>`, між якими розміщено текст, що відобразатиметься в заголовку вікна браузера. Крім цього, у розділі заголовків може міститися тег `<META>`, призначений для технічного опису документа (це інформація для пошукових програм), а також тег `<STYLE>` для опису стилів (наборів параметрів форматування), використаних у документі.
- Сам текст документа міститься в його тілі, яке розташоване між тегами `<BODY>` та `</BODY>`.

Отже, основну структуру HTML-документа визначають чотири парні теги. Їхня наявність передбачена у всіх таких документах;

```
<HTML>  
<HEAD>  
<TITLE>Заголовок документа</TITLE>
```

```
</HEAD>
<BODY>
Текст, що відображається на екрані
</BODY>
</HTML>
```

Для того щоб розбити текст на логічні частини, використовують *заголовки*. Мова HTML підтримує шість рівнів заголовків документів. Вони позначені тегами від `<H1>...</H1>` до `<H6>...</H6>`. На екрані ці заголовки відображаються різними шрифтами (зазвичай напівжирними). Хоча в мові HTML є теги форматування, які можуть змінювати зображення шрифту, користуватися ними для заголовків не рекомендовано.

Текст, що міститься всередині тегу заголовка, відобразатиметься відповідно до його рівня. Найвищий рівень має заголовок H1, найнижчий — H6. Текст заголовка може бути вирівняний по центру, за правим або лівим краєм — для цього використовують атрибут `ALIGN`. Наприклад: `<H2 ALIGN=center \ right \ left>Текст заголовка </H2>` (тут символ `\` означає можливість вибору між значеннями `center`, `right` або `left`).

На рис. 2.1 показано заголовки різних рівнів.

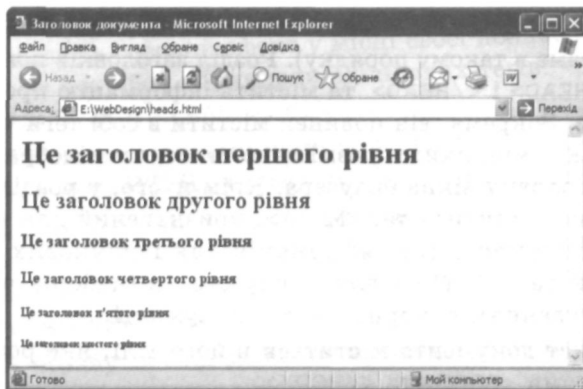


Рис. 2.1. Заголовки різних рівнів

Для визначення *звичайних абзаців* у мові HTML використовують теги `<P>` і `</P>`. Текст, розміщений між ними, власне і буде одним абзацом. При цьому слід пам'ятати, що всі «зайві» пропуски між

словами і переходу на новий рядок під час відтворення HTML-документа браузер ігнорує. Коли браузер відтворює веб-сторінки, абзаци відділяються один від одного за допомогою відступів.

У разі переходу на новий рядок без створення абзацу використовується одинарний тег
. Розділювачами в тексті можуть також бути горизонтальні лінії, які візуально відділяють різні частини документа, — їх створюють за допомогою одинарного тегу <HR>.

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.

Браузер перетворює текст HTML-документа під час його виведення на екран, тобто ігнорує зайві пропуски, символи табуляції, символи кінця рядка, форматує текст відповідно до використаних тегів. Проте іноді виникає потреба у тому, щоб помістити в документ відформатований текст, для якого браузер повинен зберегти всі пропуски, навіть якщо кілька пропусків розміщені один за одним, знаки переносу на інший рядок, табуляції та інші символи форматування. Для цього використовують тег <PRE>Текст</PRE>. Втім, якщо браузер зустріне в такому тексті теги розмітки, вона буде відповідним чином застосована.

Теги форматування тексту

Текст у HTML-документах форматують за допомогою відповідних тегів. У табл. 2.1 наведено теги, які використовують для форматування символів, та заданий ними ефект.

Таблиця 2.1. Теги та задані ними ефекти

Теги	Ефект	Приклад
 та 	Напівжирне накреслення	bold
<I> та </I>	Курсивне накреслення	<i>italic</i>
<U> та </U>	Підкреслене	<u>underlined</u>
<S> та </S>	Перекреслене	stricken
^{та}	Верхній індекс	^{superscript}
_{та}	Нижній індекс	_{subscript}

Окрім цього, можна визначати різні параметри шрифту. Зокрема, за допомогою тегу <BASEFONT> задають гарнітуру (рисунок), розмір і колір шрифту. Цей тег одинарний, діє на весь текст, розміщений нижче, і має такі атрибути.

- FACE="значення" — гарнітура. Якщо зазначеної гарнітури на комп'ютері користувача немає, то текст буде оформлено з використанням гарнітури за умовчанням. В атрибуті FACE іноді записують через кому кілька гарнітур, тоді браузер вибирає гарнітури за списком: за відсутності першої використовує другу і т. д. Значення атрибута слід подавати в лапках.
- FACE=значення — розмір шрифту (ціле число від 1 до 7).

Шрифти різних розмірів мають такий вигляд:

1 2 3 4 5 6 7

- COLOR=значення — колір шрифту. Як значення кольору можна використовувати назви кольорів англійською мовою, такі як red, green, blue тощо, або шістнадцяткові значення (записані в шістнадцятковій системі числення), що починаються з символу # і послідовно задають червоний, зелений та синій кольори, які мають бути змішані для отримання потрібного. Наприклад, #FF0000, #008000, #0000FF тощо.

Для оформлення сайту зазвичай використовують набір кольорів, який називають *безпечною палітрою*. Він складається з 216 елементів. Особливістю безпечних кольорів є те, що вони не змінюються у разі відображення різними браузерами або на різних моніторах, тобто ця палітра забезпечує найточнішу передачу на різних моні-

торах того, що задумав веб-дизайнер. Якщо будь-яка з трьох складових шістнадцяткового значення відрізняється від 00, 33, 66, 99, CC чи FF, то колір не є безпечним. Значення атрибутів SIZE та COLOR можна записувати без лапок.

Гарнітуру, розмір та колір шрифту для фрагмента тексту задають за допомогою тегу Текст. Атрибути цього тегу такі самі, як у тегу <BASEFONT>, за винятком атрибута SIZE: його значення зі знаком мінус (або плюс) означає, що розмір шрифту буде зменшено (чи збільшено) на відповідну величину відносно заданого за умовчанням або у тегу <BASEFONT>. Значення без знака задає абсолютний розмір шрифту.

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

- <BIG>Текст</BIG> — збільшення розміру шрифту. Розмір символів тексту збільшується на одиницю відносно поточного рівня.
- <SMALL>Текст</SMALL> — зменшення розміру шрифту. Розмір символів зменшується на одиницю відносно поточного рівня.
- Текст — виділення важливих фрагментів тексту. Текст відображатиметься курсивом.
- Текст — створення перекресленого тексту. Текст буде перекреслено горизонтальною лінією.

Цікавого оформлення веб-сторінки надає пара тегів <MARQUEE> та </MARQUEE> — вони додають до веб-сторінки «рухомий рядок» із тексту, розташованого між ними.

Вправа 2.1

Створимо HTML-документ за допомогою найпростішого текстового редактора.

1. Відкрийте текстовий редактор Блокнот (Notepad).
2. Для створення HTML-документа наберіть такий текст:

```
<HTML>
<HEAD>
<TITLE>Купівля\продаж</TITLE>
</HEAD>
```

```
<BODY>
<H1>Торговельні об'єкти Львівщини</H1>
<H2>Супермаркети</H2>
<P>На Львівщині діє мережа магазинів Інтермаркет. У кожному районі є торговельний об'єкт.<P>Одним із таких магазинів став нещодавно відкритий супермаркет "АРСЕН".
<HR>
Це торговельний центр із багатьма відділами та зручною системою обслуговування покупців.<BR>Його можна назвати торговельним містечком.
</BODY>
<HTML>
```

3. Збережіть файл на жорсткому диску у створеній задалегідь папці, назвавши його, наприклад, 1.html. Закрийте текстовий редактор.
4. Запустіть браузер Internet Explorer та відкрийте файл 1.html. Перегляньте отриманий результат.
5. Проаналізуйте, як відтворено веб-сторінку. Якщо потрібно внести зміни, виберіть команду Перегляд HTML-коду в меню Вигляд. Збережіть зміни та оновіть веб-сторінку за допомогою клавіші F5 або кнопки Оновити панелі інструментів.

На рис. 2.2 показано, як буде відтворено цей документ у вікні браузера Internet Explorer.

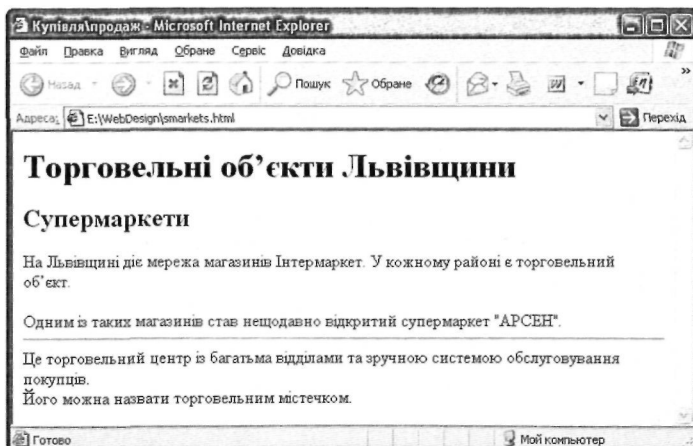


Рис. 2.2. Приклад веб-сторінки

Вправа 2.2

Помістимо на веб-сторінку наведений нижче текст, відформатувавши його за допомогою відповідних тегів.

Троянда сталиста

Граціозні форми, ніжне забарвлення, витончений аромат **троянди** з незапам'ятних часів чарували людей. Тому й створено **безліч легенд** про цю чарівну, ніжну й чудову квітку.

За староіндійською легендою, прекрасна богиня **Лакшмі** народилася з пуп'янка троянди.

У древньому індійському міфі розповідається, що прегарний юнак **Адоніс**, який уособлював прихід весни, постав із куща троянди.

У Персії так шанували царицю квітів, що навіть країна одержала назву Гюлістан — країна троянд.

У Римі цю квітку вважали символом хоробрості, нею нагороджували за доблесть. В Афінах часів **Аристофана** трояндовий вінок одягали на шию переможцям змагань. Гірлянди квітів прикрашали колони і стіни залів у дні тріумфальних святкувань.

Існує припущення, що **русальна неділя** — давнє свято в Україні та Білорусі — пов'язана з **розаріями** — святом римлян на честь цих прекрасних квітів.

*(Мамчур Ф.І., Гладун Я.Д.,
Лікарські рослини на присадибній ділянці)*

1. Введіть HTML-код веб-сторінки, на якій відобразатиметься наведений вище відформатований текст.

У цьому прикладі для деяких фрагментів тексту необхідно застосувати по два теги форматування, наприклад, <I><U>Україні</U></I>. Слід пам'ятати про правило вкладених тегів, згідно з яким їх потрібно закривати у зворотному до відкриття порядку. Тобто якщо спершу відкрили тег курсиву, а потім — тег підкреслення, то спочатку слід закривати тег підкреслення, а потім — курсиву.

2. Збережіть html-документ у файлі з розширенням **.html**. Вигляд створеної веб-сторінки у вікні браузера показаний на рис. 2.3.

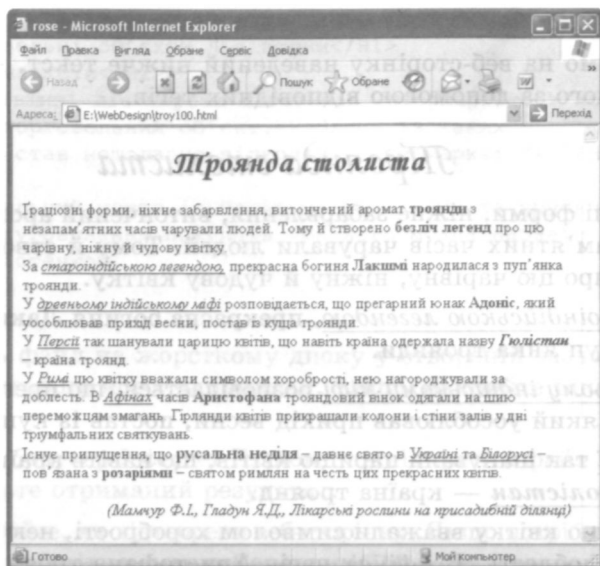


Рис. 2.3. Веб-сторінка про троянду столісту

Нумеровані й марковані списки

Для подання тексту в структурованому вигляді, коли кожен абзац виділено за допомогою позначки або номера, призначені списки. У HTML-документах використовують три види списків:

- неупорядкований (маркований);
- упорядкований (нумерований);
- список визначень.

У *неупорядкованому списку* для виділення елементів використовуються певні позначки (крапки, квадрати тощо). Починається такий список із тегу , а закінчується — . Кожний елемент списку починається з тегу . Тег може мати атрибут TYPE, який визначає форму позначки. Цей атрибут набуває таких значень:

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

В *упорядкованому списку* всі елементи пронумеровані. Починається список із тегу , а закінчується — . Кожний елемент такого списку також починається з тегу . Тег 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>
```

Список визначень (definition list) порівняно з маркованим або нумерованим має дещо складнішу будову. Кожен елемент такого списку складається з двох частин: терміна та його визначення. У такий спосіб можуть бути оформлені словники та термінологічні довідники. Однак цей тип списку використовують рідко, і тому описувати його докладніше не будемо.

Розглянемо приклади використання різних типів списків.

Приклад 2.1. Маркований список, в якому для позначення елементів використовують коло

```
<HTML>
<TITLE>Райдуга</TITLE>
<BODY>
<BASEFONT SIZE=6>
<UL TYPE=circle>
  <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>
</UL>
</BODY>
</HTML>

```

На рис. 2.4 показаний вигляд цього документа у вікні браузера Internet Explorer.

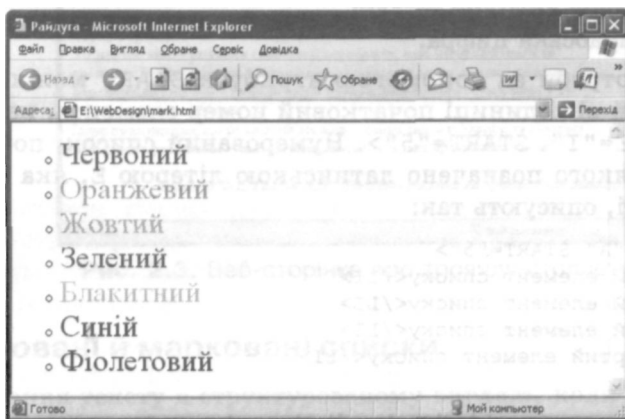


Рис. 2.4. Маркований список

Приклад 2.2. Нумерований список, в якому для позначення елементів використовують малі римські цифри

```

<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>

На рис. 2.5 показаний вигляд цього документа у вікні браузера Internet Explorer.

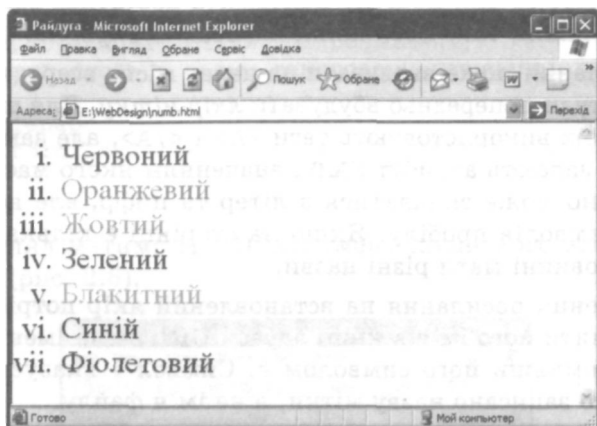


Рис. 2.5. Нумерований список

Зверніть увагу на те, що за допомогою тегу `<BASEFONT>` для всього документа призначено розмір шрифту 6, а за допомогою тегів `` визначено колір шрифту для кожного елемента списку.

Текстові гіперпосилання

Гіперпосилання — це вказівка для браузера, до якого об'єкта у межах або поза межами HTML-документа він має звернутися. За допомогою гіперпосилань користувачі можуть переходити з однієї веб-сторінки на іншу, завантажувати файли тощо. Як гіперпосилання може бути оформлений фрагмент тексту або графічний об'єкт. Коли веб-сторінка відображається у вікні браузера, текстове посилання зазвичай виділено синім кольором і підкресленням. Для переходу в місце, на яке вказує посилання, користувачу достатньо клацнути його текст.

Для створення гіперпосилання необхідно використовувати теги `<A>` та ``, визначивши для тегу `<A>` атрибут `HREF`. Його значенням має бути адреса URL, на яку вказує посилання. Текст посилання розташовують між тегами `<A>` і ``.

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 , 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.

У прикладі, який наводиться нижче, розглянуто застосування гіперпосилань різного типу.

Приклад 2.3. HTML-документ, в якому використовуються гіперпосилання та якорі

```
<HTML>
<HEAD>
<TITLE>Посилання та якорі</TITLE>
</HEAD>
<BODY>
Нові версії стандартних програм операційної системи
та найважливіші драйвери
можна знайти на веб-сайті компанії
<A HREF="http://www.microsoft.com/">Microsoft</A>.
```

```

<P>А тепер можна перейти на <A HREF="main.htm">
мою особисту 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:olena@zirka.lviv.ua">olena@zirka.lviv.ua</A>
</BODY>
</HTML>

```

Такий вигляд матиме HTML-документ після його відтворення браузером (рис. 2.6).

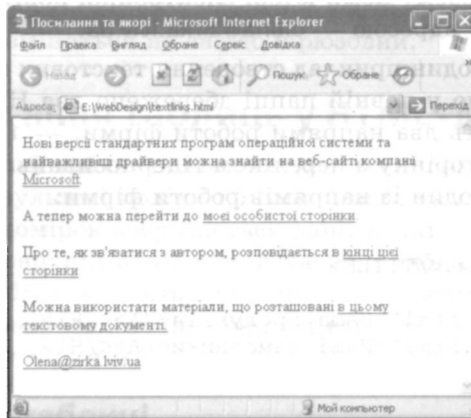


Рис. 2.6. Веб-сторінка з текстовими гіперпосиланнями

У цьому прикладі слово «Microsoft» міститься у тегу зовнішнього гіперпосилання, а текст «мою особисту сторінку» — у тегу внутрішнього посилання. Текст «кінці цієї сторінки» розміщений у тегу посилання на якорь, а «в цьому текстовому документі» — у тегу гіперпосилання з атрибутом HREF, в якому задано зв'язок не з веб-сторінкою, а з текстовим документом, котрий збережений у тій самій папці, що й поточний HTML-документ. Текст «olena@zirka.lviv.ua» вміщено у тег, що описує мітку якоря, а в атрибуті HREF зазначено адресу електронної пошти.

Якщо відвідувач веб-сторінки скористається зовнішнім гіперпосиланням Microsoft, то відкриється сторінка, що міститься за адресою: <http://www.microsoft.com/>. Після клацання гіперпосилання мою особисту сторінку відкриється веб-сторінка, записана у файлі main.htm, який міститься у тій самій папці, що й поточна сторінка. Після клацання внутрішнього посилання в цьому текстовому документі відкриється вікно з текстовим документом text.doc, що міститься у поточній папці. Якщо скористатися посиланням на якір кінці цієї сторінки, то зображення поточної веб-сторінки зміститься так, що текст мітки «[Olena@zirka.lviv.ua](#)», з якою зв'язаний якір, буде розміщений у видимій на екрані частині документа.

Водночас текст «Olena@zirka.lviv.ua» є гіперпосиланням, скориставшись яким, користувач зможе надіслати листа за допомогою настроєної на своєму комп'ютері електронної пошти за вказаною в атрибуті HREF адресою — olena@zirka.lviv.ua.

Розглянемо ще один приклад створення текстових гіперпосилань. Припустимо, що у певній папці збережено два HTML-документи, які описують два напрями роботи фірми, — 1.html та 2.html. Створимо веб-сторінку з переліком гіперпосилань, кожне з яких ілюструватиме один із напрямів роботи фірми:

```
<HTML>
<TITLE>Моя фірма</TITLE>
<BODY>
<H1><A HREF="1.html">Наша продукція</A></H1>
<H1><A HREF="2.html">Наші замовники</A></H1>
</BODY>
</HTML>
```

У цьому прикладі таги гіперпосилань містяться в тегах заголовків першого рівня. Тексти гіперпосилань будуть розташовані в окремих рядках і оформлені як заголовки першого рівня.

За умовчанням текстові гіперпосилання виділені синім кольором і підкреслені, а якщо ними вже скористалися, то темно-червоним. Для зміни цих кольорів додамо відповідні атрибути до тегу <BODY>: <BODY LINK=magenta ALINK=yellow>. Це означає, що після першого завантаження веб-сторінки всі гіперпосилання будуть фіолетового кольору, а якщо відвідувач скористається одним із них, то колір його тексту стане жовтим (рис. 2.7). Це зручно, коли на веб-сторінці є багато посилань, і відвідувач бажає по черзі

переглянути їх: тоді надання іншого кольору відвіданим гіперпосиланням дає змогу систематизувати такий перегляд.

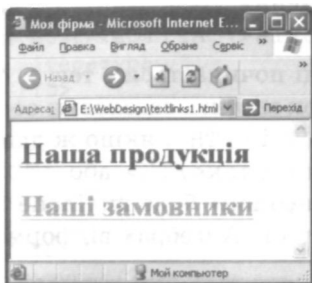


Рис. 2.7. Веб-сторінка з гіперпосиланнями, виконаними як заголовки першого ривня

Якщо додати до тегу `<BODY>` атрибут `BGCOLOR=cyan`, то колір тла веб-сторінки зміниться на світло-бірюзовий.

Використання таблиць у HTML-документах

У найпростішій таблиці інформація розміщена у комірках, утворених у результаті поділу прямокутника на стовпці та рядки. У більшості комірок зберігаються дані, деякі комірки, зазвичай розміщені у верхній або боковій частині таблиці, містять заголовки. У HTML-документах таблицю заповнюють зліва направо, комірку за коміркою, починаючи з лівого верхнього кута і закінчуючи правим нижнім.

Створення таблиці

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>`. Наприклад, `<TH>Понеділок</TH>` або `<TD>1. Алгебра</TD>`. У таблиці слово «Понеділок» буде виділене браузером напівжирним шрифтом, а текст «1. Алгебра» відформатовано стандартним способом.

Приклад 2.4. HTML-документ, що містить таблицю

Визначимо структуру табл. 2.2. Вона містить назви п'яти робочих днів тижня, на кожен з яких припадає 4-5 уроків.

Таблиця 2.2. Розклад уроків

Понеділок	Вівторок	Середа	Четвер	П'ятниця
1. Алгебра	1. Українська мова	1. Зарубіжна література	1. Географія	1. Хімія
2. Іноземна мова	2. Хімія	2. Геометрія	2. Фізична культура	2. Іноземна мова
3. Фізична культура	3. Алгебра	3. Іноземна мова	3. Фізика	3. Геометрія
4. Географія	4. Інформатика	4. Трудове навчання	4. Українська мова	4. Українська література
	5. Фізика	5. Українська література		

HTML-код для створення цієї таблиці матиме такий вигляд:

```
<HTML>
<TITLE>Приклад простої таблиці</TITLE>
<BODY>
<TABLE BORDER="1">
<CAPTION>Розклад уроків</CAPTION>
<TR>
<TH>Понеділок</TH>
<TH>Вівторок</TH>
<TH>Середа</TH>
<TH>Четвер</TH>
```

```

        <TH>П'ятниця</TH>
</TR>
<TR>
    <TD>1. Алгебра</TD>
    <TD>1. Українська мова</TD>
    <TD>1. Зарубіжна література</TD>
    <TD>1. Географія</TD>
    <TD>1. Хімія</TD>
</TR>
<TR>
    <TD>2. Іноземна мова</TD>
    <TD>2. Хімія</TD>
    <TD>2. Геометрія</TD>
    <TD>2. Фізична культура</TD>
    <TD>2. Іноземна мова</TD>
</TR>
<TR>
    <TD>3. Фізична культура</TD>
    <TD>3. Алгебра</TD>
    <TD>3. Іноземна мова</TD>
    <TD>3. Фізика</TD>
    <TD>3. Геометрія</TD>
</TR>
<TR>
    <TD>4. Географія</TD>
    <TD>4. Інформатика</TD>
    <TD>4. Трудове навчання</TD>
    <TD>4. Українська мова</TD>
    <TD>4. Українська література</TD>
</TR>
<TR>
    <TD></TD>
    <TD>5. Фізика</TD>
    <TD>5. Українська література</TD>
    <TD></TD>
    <TD></TD>
</TR>
</TABLE>
</BODY>
</HTML>

```

Який вигляд матиме ця веб-сторінка у вікні браузера, показано на рис. 2.8.

Зверніть увагу на те, що за відсутності даних у комірці таблиці рамка навколо неї не відображається. Для того щоб рамку було

видно, можна ввести спеціальний код ` ` (non-breaking space — нерозривний пробіл). Після цього порожня комірка буде взята в рамку.

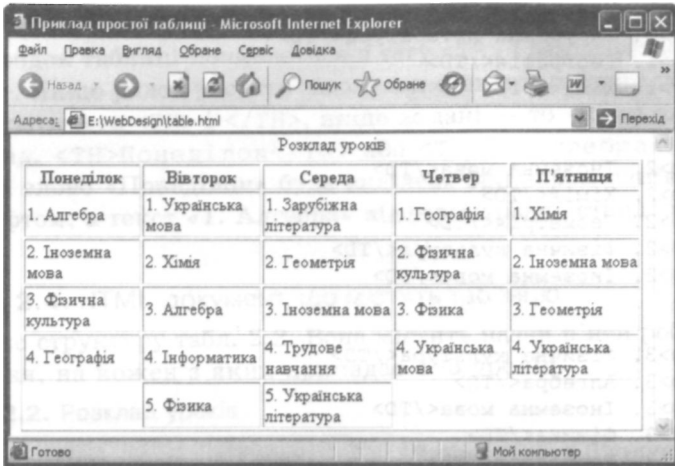


Рис. 2.8. Документ, який містить таблицю

Об'єднання комірок

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 послідовно 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.

Приклад 2.5. HTML-документ, який містить таблицю з об'єднаними комірками

Створимо HTML-документ, що містить дві таблиці, в яких є об'єднані комірки (табл. 2.3, табл. 2.4).

Таблиця 2.3. Таблиця з комірками, об'єднаними по вертикалі

10-A

10-B

Таблиця 2.4. Таблиця з комірками, об'єднаними по горизонталі

Класи

11-A

11-B

11-B

Код цього документа наведено нижче.

```
<HTML>
<TITLE>Приклад ROWSPAN та COLSPAN</TITLE>
<BODY>
<TABLE BORDER="1">
<CAPTION>Приклад ROWSPAN</CAPTION>
  <TR>
    <TD ROWSPAN="2">Класи</TD>
    <TD>10-A</TD>
  </TR>
  <TR>
    <TD>10-B</TD>
  </TR>
</TABLE>
<TABLE BORDER="1">
<CAPTION>Приклад COLSPAN</CAPTION>
  <TR>
    <TD COLSPAN="3">Класи</TD>
  </TR>
  <TR>
    <TD>11-A</TD>
    <TD>11-B</TD>
    <TD>11-B</TD>
  </TR>
</TABLE>
</BODY>
</HTML>
```

На рис. 2.9 показано вигляд цього HTML-документа у вікні браузера. Ширина таблиць добирається автоматично. Їх заголовки таблиць відображено у двох рядках, оскільки вони не можуть виходити за межі таблиць.

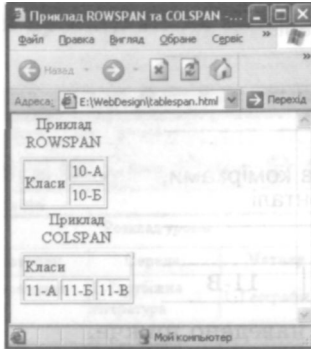


Рис. 2.9. Таблиці з об'єднаними комірками

Вправа 2.3

Створимо веб-сторінку з таблицею, що має таку ж структуру, як і табл. 2.5, в якій одночасно об'єднано і стовпці, і рядки.

Таблиця 2.5. Таблиця, в якій об'єднано кілька рядків і кілька стовпців

		Види спорту			
		Плавання	Гімнастика	Плавання	Гімнастика
11-А	11-Б	11-А	11-Б	11-А	11-Б
15	19	10	5	11	8

1. Створіть HTML-документ, де у відповідних тегах <TD>...</TD> визначені атрибути ROWSPAN та COLSPAN.
2. Збережіть документ із розширенням .html. На рис. 2.10 показано, який вигляд має така таблиця у браузері. Комірка, в якій записано текст «Кількість учнів», об'єднує два рядки і два стовпці.

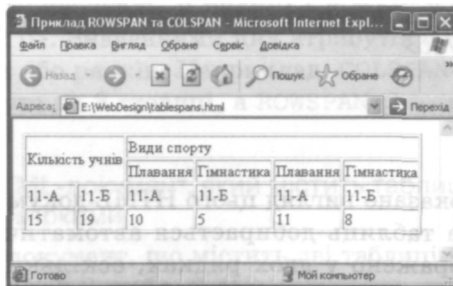


Рис. 2.10. Таблиця, де водночас об'єднано рядки і стовпці

Встановлення ширини таблиці та комірок

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>`, а комірки — за допомогою того самого атрибута у тегу `<TD>` або `<TH>`. Ширину таблиці можна визначати як у пікселях, так і у відсотках від ширини вікна, а комірки — у пікселях або у відсотках від ширини таблиці.

Приклад 2.6. HTML-документ, в якому задана ширина таблиці та деяких комірок

Розглянемо HTML-документ, в якому буде створено таблицю певної ширини, а також задано ширину деяких комірок.

```
<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>
```

На рис. 2.11 показано, який вигляд має цей документ у вікні браузера.

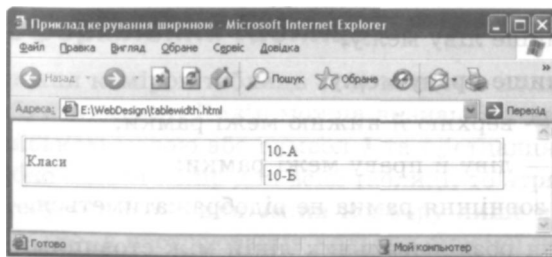


Рис. 2.11. Таблиця, ширину якої задано за допомогою атрибута `WIDTH`

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

Форматування тексту та оформлення рамок таблиці

Текст всередині комірок можна вирівнювати як по горизонталі, так і по вертикалі. Горизонтальне вирівнювання вмісту всього рядка задають за допомогою атрибута `ALIGN` у тегу `<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>`; `<TD ALIGN=center>1. Алгебра</TD>`.

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 рядками таблиці визначають за допомогою атрибута `RULES` тегу `<TABLE>`.

Він може набувати таких значень:

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

Вправа 2.4

Настроїмо зовнішній вигляд таблиці.

1. Створіть HTML-документ, який містить таблицю із такими настройками: товщина рамки таблиці дорівнює 15 пікселам, відображаються всі її чотири сторони і лише горизонтальні розділювальні лінії між рядками. Нехай ширина таблиці становить 300 пікселів.
2. Збережіть документ у файлі з розширенням .html. На рис. 2.12 показано, який вигляд має таблиця у вікні браузера.

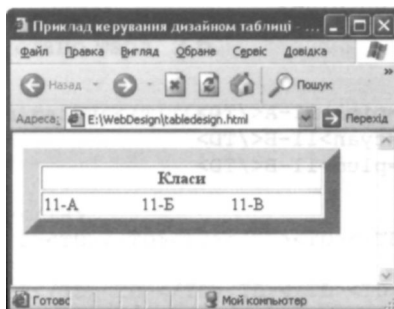


Рис. 2.12. Таблиця із рамкою товщиною 15 пікселів

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 налаштування кольору окремої комірки забезпечує наявність цього атрибута всередині тегу `<TD>` або `<TH>`.

Колір рамки таблиці можна задати за допомогою таких атрибутів:

- **BORDERCOLOR** — колір ВСІЇ 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.

Розглянемо приклад оформлення таблиці за допомогою різних кольорів.

Приклад 2.7. HTML-документ, який містить різнокольорову таблицю

```
<HTML>
<TITLE>Приклад керування кольором</TITLE>
<BODY>
<TABLE BGCOLOR="#F0F0F0" BORDER=10 BORDERCOLOR="#808080"
BORDERCOLORLIGHT="#9 9CCFF" BORDERCOLORDARK="#9900FF">
  <TR BGCOLOR=lightblue>
    <TH COLSPAN="3">Класи</TH>
  </TR>
  <TR>
    <TD BGCOLOR=pink>11-A</TD>
    <TD BGCOLOR=cyan>11-B</TD>
    <TD BGCOLOR=plum>11-B</TD>
  </TR>
</TABLE>
</BODY>
</HTML>
```

Якщо необхідно задати колір шрифту для тексту таблиці, то це можна зробити за допомогою тегу `` всередині відповідних тегів `<TD>`. Наприклад:

```
<TD BGCOLOR=magenta><FONT COLOR=pink>ТеКСТ</FONT></TD>
```

Використання таблиць для розміщення об'єктів на веб-сторінці

Під час відображення веб-сторінок браузер розміщує елементи на екрані один за одним у тому порядку, в якому вони записані у HTML-документі. Для позиціювання елементів розроблено спеціальні засоби мови HTML, проте часто для цього використовують

таблиці. На веб-сторінці створюють таблицю без рамок і у її комірки поміщають різні об'єкти.

Розмістимо назви класів по діагоналі так, як це показано в табл. 2.6, але самої таблиці на веб-сторінці відобразити не будемо.

Таблиця 2.6. Таблиця, в якій інформація розміщується по діагоналі

Старші класи

10-A

10-B

11-A

11-B

11-B

Нижче наведено текст HTML-документа, в якому для позиціонування елементів використано табл. 2.6.

```
<HTML>
<TITLE>Приклад позиціювання</TITLE>
<BODY>
<TABLE>
  <TR>
    <TH>Старші класи</TH>
    <TD></TD><TD></TD><TD></TD><TD></TD><TD></TD><TD></TD><TD></TD></TR>
  <TR>
    <TD></TD><TD>10-A</TD><TD></TD><TD></TD><TD></TD><TD></TD><TD></TD>
  </TR>
  <TR>
    <TD></TD><TD></TD><TD>10-B</TD><TD></TD><TD></TD><TD></TD><TD></TD>
  </TR>
  <TR>
    <TD></TD><TD></TD><TD></TD><TD>11-A</TD><TD></TD><TD></TD><TD></TD>
  </TR>
  <TR>
    <TD></TD><TD></TD><TD></TD><TD></TD><TD>11-B</TD><TD></TD><TD></TD>
  </TR>
  <TR>
    <TD></TD><TD></TD><TD></TD><TD></TD><TD></TD><TD></TD><TD>11-B</TD>
  </TR>
</TABLE>
</BODY>
</HTML>
```

На рис. 2.13 показано, який вигляд має цей HTML-документ у вікні браузера.

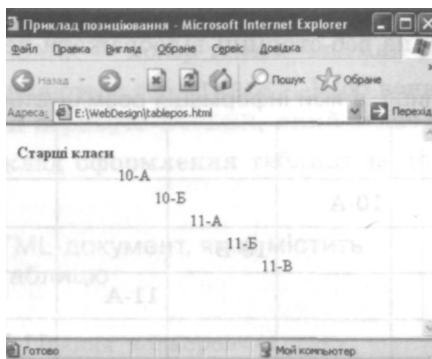


Рис. 2.13. Позиціонування елементів на веб-сторінці за допомогою таблиці

Комірки таблиці можна заповнювати не лише текстом, а й зображеннями, іншими таблицями та рухомими рядками. Текст або зображення всередині комірки може бути гіперпосиланням. Наприклад:

```
<TD><A HREF="lib.html">11-В</A></TD>
```

Вправа 2.5

Розробимо веб-сайт, на головній сторінці якого в кутах уявного трикутника будуть розміщені три гіперпосилання на сторінки про навчальні заклади. Кожне посилання пов'язане з файлом, в якому міститься інформація про навчальний заклад.

1. Визначте структуру головної сторінки сайту за допомогою табл. 2.7.

Таблиця 2.7. Структурна таблиця головної сторінки сайту

Школа «Дивосвіт»	
Гімназія «Сихівська»	Ліцей «Оріяна»

2. Підготуйте текстове наповнення для веб-сторінок, які пов'язані гіперпосиланнями з головною сторінкою.
 - Веб-сторінка про школу «Дивосвіт» — «Цю школу відвідують діти молодших класів, вони додатково вчаться малюванню, музиці та танцям».
 - Веб-сторінка про гімназію «Сихівська» — «Діти відвідують гімназію з п'ятого класу, додатково вивчають іноземні мови, інформатику, відвідують спортивні секції».
 - Веб-сторінка про ліцей «Оріяна» — «У ліцеї основну увагу приділяють вивченню математики та інформатики».
3. Розробіть HTML-код відповідних сторінок, скориставшись текстовим редактором Блокнот (Notepad).
4. Запишіть HTML-код для веб-сторінки про школу «Дивосвіт» і збережіть цей файл як 1.html.

```
<HTML>
<TITLE>Школа «Дивосвіт»</TITLE>
<BODY BGCOLOR=CYAN><BASEFONT SIZE=6 COLOR=olive>
<H1 ALIGN=CENTER><FONT FACE="Monotype Corsiva">
Школа «Дивосвіт»</FONT></H1>
<P>Цю школу відвідують діти молодших класів,
вони додатково вчаться малюванню, музиці та танцям.</P>
</BODY>
</HTML>
```

5. Запишіть HTML-код для веб-сторінки про гімназію «Сихівська» і збережіть цей файл як 2.html.

```
<HTML>
<TITLE>Гімназія «Сихівська»</TITLE>
<BODY BGCOLOR=lightcoral><BASEFONT SIZE=6 COLOR=indigo>
<H1 ALIGN=CENTER><FONT FACE="Monotype Corsiva">Гімназія
«Сихівська»</FONT></H1>
<P>Діти відвідують гімназію з п'ятого класу,
додатково вивчають іноземні мови, інформатику,
відвідують спортивні секції.</P>
</BODY>
</HTML>
```

6. Запишіть HTML-код для веб-сторінки про ліцей «Оріяна» і збережіть цей файл як 3.html.

```
<HTML>
<TITLE>Ліцей «Оріяна»</TITLE>
<BODY BGCOLOR=lavender><BASEFONT SIZE=6 COLOR=dark:cyan>
```

```

<H1 ALIGN=CENTER><FONT FACE="MonotYpe Corsiva">Ліцей «Орія-
На»</FONT></H1>
<P>У ліцеї основну увагу приділяють вивченню
математики та інформатики.</P>
</BODY>
</HTML>

```

7. Запишіть HTML-код для головної сторінки та збережіть цей файл як **index.html**. Для позиціювання гіперпосилань на головній сторінці використайте табл. 2.7. Її межі мають бути невидимі.

На рис. 2.14 показано головну сторінку сайту, на якій міститься інформація про навчальні заклади.

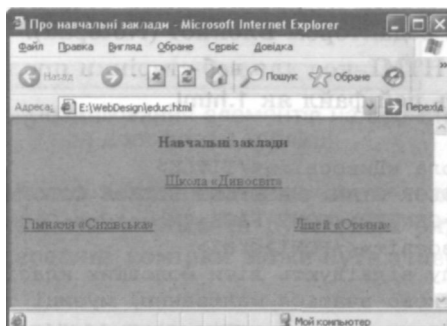


Рис. 2.14. Головна сторінка сайту, на якій міститься інформація про навчальні заклади

8. Відкрийте у вікні браузера головну сторінку створеного веб-сайту, клацніть одне з гіперпосилань, і на екрані відобразиться інформація про вибраний навчальний заклад.
9. Клацніть кнопку **Назад** на панелі інструментів браузера. Після цього знову відобразиться головна сторінка, на якій можна вибрати інше гіперпосилання.

HTML-документи, які містять фрейми

Ви, мабуть, помітили, що користуватися створеним сайтом не дуже зручно, оскільки для переходу на сторінку з головним меню необхідно клацати кнопку **Назад**. Тому в мові HTML є засіб, за допомогою якого можна створити меню, що постійно відображатиметься під час перегляду сайту. Інформація, що відкривається

за допомогою його посилань, буде розташована в іншому наперед визначеному місці.

Список усіх таких посилань, який фактично є змістом веб-сайту і постійно відображається на екрані, називають меню сайту або навігаційним меню. Типове меню складається з графічних зображень у вигляді кнопок або з текстових гіперпосилань, призначених для переходу в основні розділи сайту. Продумане навігаційне меню може легко провести відвідувача сайту всіма основними і додатковими розділами. Постійне відображення меню суттєво спрощує навігацію розділами, тому відвідувачі можуть більше часу присвячувати опрацюванню веб-сайту.

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 < FRAME SET >. 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>`.

Вправа 2.6

Розмістімо на сторінці три документи про навчальні заклади (див. вправу 2.5) так, як це показано в табл. 2.8.

Таблиця 2.8. Схема розташування на сторінці фреймів, які містять інформацію про навчальні заклади

Документ про гімназію
«Сихівська» (2.html)

Документ про ліцей
«Оріяна» (3.html)

Документ про школу «Дивосвіт» (1.html)

Як видно з таблиці, сторінку поділено спочатку на два рядки у співвідношенні приблизно 60 % та 40 %, перший рядок поділено на два стовпці, а в утворених комірках розташовано відповідні веб-сторінки про навчальні заклади. Документи **1.html**, **2.html** та **3.html** було створено під час виконання вправи 2.5.

1. Створіть файл головної сторінки та збережіть його як **main.html** у тій самій папці, що й решту документів.

```
<HTML>
<TITLE>Навчальні заклади</TITLE>
<FRAMESET ROWS="60%, 40%">
<FRAMESET COLS="50%,50%">
<FRAME SRC="2.html">
<FRAME SRC="3.html">
</FRAMESET>
<FRAME SRC="1.html">
</FRAMESET>
</HTML>
```

2. Відкрийте файл **main.html**. Який вигляд має сторінка із поділом на фрейми можна побачити на рис. 2.15.

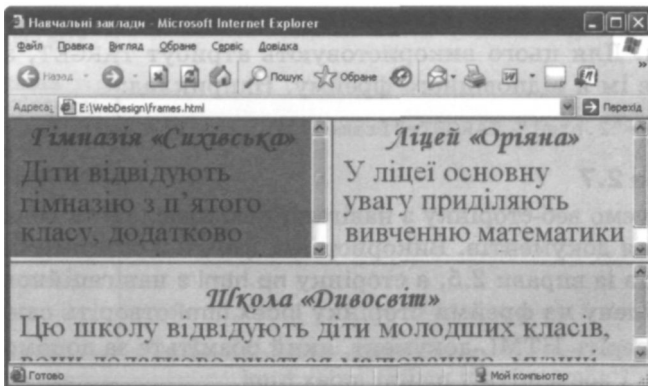


Рис. 2.15. Вікно містить три фрейми, в кожному з яких відображається окремий HTML-документ

Всі файли відображені у вікні браузера одночасно, а оскільки документи не вмістилися у визначені для них області, то кадри мають смуги прокручування.

Використання посилань у фреймах

У вправі 2.6 не використані гіперпосилання — між фреймами немає взаємозв'язку. Проте це не завжди зручно: навчальних закладів може бути багато, а поділ сторінки на велику кількість фреймів призводить до розпорошування уваги відвідувача сайту.

Тому розглянемо веб-сторінку з навігаційною панеллю, яка розміщена у лівій меншій частині екрана, при цьому необхідна інформація відобразатиметься у більшій правій частині так, як це показано в табл. 2.9.

Таблиця 2.9. Схема розміщення фреймів, в якій використовується навігаційна панель

Навігаційна панель	Документ з інформацією про вибраний відвідувачем сайту навчальний заклад
--------------------	--------------------------------------------------------------------------

Якщо заплановано використовувати певний фрейм для відображення інформації, яка змінюватиметься після вибору відвідувачами іншого посилання, то йому надають ім'я. Наприклад, `<FRAME NAME="frame1">`. Веб-сторінка з гіперпосиланнями, що виконуватиме роль навігаційної панелі, складається із сукупності тегів `<A>...`. У них крім адреси файлу, пов'язаного з гіперпосиланням, слід зазначити ім'я фрейму, в якому цей файл відобразатиметься. Для цього використовують атрибут `TARGET`, значенням якого є ім'я відповідного фрейму. Наприклад:

```
<A HREF="2.html" TARGET="frame1">Гімназія «Сихівська»</A>
```

Вправа 2.7

Сформуємо веб-сторінку з навігаційною панеллю та областю відображення документів. Використайте файли з описами навчальних закладів із вправи 2.5, а сторінку `np.html` з навігаційною панеллю та поділену на фрейми сторінку `Index.html` створіть самостійно.

1. Створіть HTML-документ, який розмітьте за допомогою фреймів і збережіть у файлі `index.html`.

```
<HTML>  
<TITLE>Навчальні заклади</TITLE>
```

```

<FRAMESET COLS="20%,*">
<FRAME SRC="np.html">
<FRAME NAME="frame1">
</FRAMESET>
</HTML>

```

- Створіть HTML-документ, який міститиме навігаційну панель, і збережіть його у файлі np.html.

```

<HTML>
<TITLE>Школи, ліцеї, гімназії</TITLE>
<BODY>
<H2><A HREF="1.html" TARGET="frame1">Школа «Дивосвіт»
</A></H2>
<H2><A HREF="2.html" TARGET="frame1">Гімназія
«Сихівська»</A></H2>
<H2><A HREF="3.html" TARGET="frame1">Ліцей «Оріяна»</A>
</H2>
</BODY>
</HTML>

```

- Відкрийте файл index.html у вікні браузера. На рис. 2.16 показано, який вигляд має поділене на фрейми вікно з навігаційною панеллю після першого завантаження веб-сторінки, поки користувач не вибрав гіперпосилання.

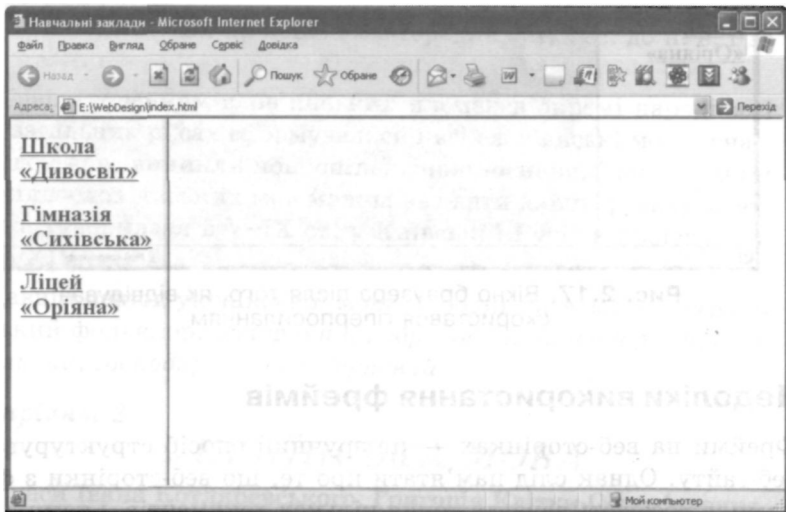


Рис. 2.16. Веб-сторінка index.html після першого завантаження

У документі index.html у п'ятому рядку, в тегу <FRAME>, задано лише один атрибут — NAME, тому після першого завантаження веб-сторінки права частина буде порожньою. Щоб цього уникнути, використовують атрибут SRC для повідомлення адреси файлу, який повинен відкриватись у цьому фреймі під час першого завантаження сторінки. Зазвичай це файл веб-сторінки з графічними вставками та загальною інформацією про сайт.

На рис. 2.17 показано, який вигляд має поділене на фрейми вікно з навігаційною панеллю після того, як відвідувач вибрав гіперпосилання на сторінку школи «Дивосвіт». Як бачите, навігаційне меню завжди в полі зору відвідувача в той час, коли він мандрує сторінками сайту.

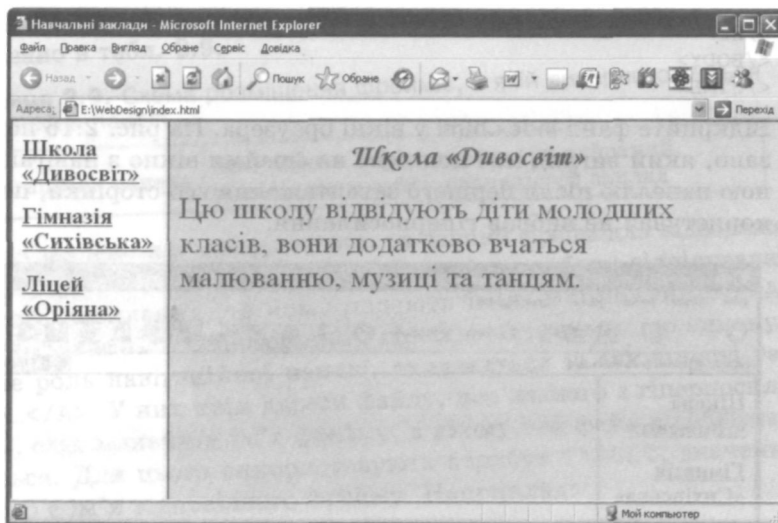


Рис. 2.17. Вікно браузера після того, як відвідувач скористався гіперпосиланням

Недоліки використання фреймів

Фрейми на веб-сторінках — це зручний спосіб структурування веб-сайту. Однак слід пам'ятати про те, що веб-сторінки з фреймами потребують завантаження кількох документів, і тому для їх відтворення браузеру необхідно більше часу, ніж для відтворення

веб-сторінки без фреймів. Окрім цього, екран із кількома веб-сторінками розпорошує увагу користувача. Іноді веб-документи, розміщені на одному екрані, мають різний стиль оформлення, що негативно впливає на загальне враження від такої сторінки.

Практична робота № 3

Розроблення найпростішої веб-сторінки

Мета роботи: навчитися створювати веб-сторінку, визначати її структуру, а також оформляти символи, абзаци і тло.

Порядок роботи

1. Згідно з вибраним варіантом, створіть веб-сторінку, яка на сірому тлі міститиме текст, відформатований у вказаний нижче спосіб.

Варіант 1

ІСТОРИЯ УКРАЇНСЬКОЇ мови

Усі слов'янські мови мають одне джерело — **праслов'янську**, або **спільнослов'янську** основу, яка існувала у слов'янських племінних мовах приблизно із середини III тис. до н. е. і до V ст. н. е.

з перетворенням слов'янських племен в окремі народності в загальних рисах сформувалися і всі слов'янські мови, внаслідок чого виникли нові спільні **мови-основи**. Так, предком східнослов'янських мов можна вважати **давньоруську** мову, яка утворилася в V-IX ст. у Київській Русі та проіснувала до XIV ст.

Давній період української мови породив багатющий український фольклор: *історичний, обрядовий, календарний, соціальний, господарський, побутовий.*

Варіант 2

УКРАЇНСЬКА МОВА

У часи **Івана Котляревського, Григорія Квітки-Основ'яненка, Тараса Шевченка** мова збагатилася народнорозмовним фольклором.

Основоположником нової української мови можна вважати видатного українського поета **Тараса Шевченка**, який синтезував усе найкраще з книжних традицій, з усної творчості та мовлення українців Наддніпрянщини, виробив і закріпив *фонетичні, морфологічні та синтаксичні норми*.

Після **Т. Шевченка** українською мовою майстерно користувалися, постійно її збагачуючи, такі талановиті письменники і діячі культури XIX-XX ст., як **Л. Глібов, Марко Вовчок, Панас Мирний, І. Нечуй-Левицький, Іван Франко, Леся Українка, М. Драгоманов, М. Коцюбинський**.

У розвиток сучасної української мови вагомий внесок внесли українські письменники **В. Сосюра, П. Тичина, М. Рильський** та ін.

Варіант 3

Глобальна мережа Інтернет. Призначення HTML

Інтернет започаткований 1969 року *Агентством з наукових досліджень Міністерства оборони США (ARPA)* як **проект із розроблення надійних цифрових комунікацій** для військових потреб. Спочатку ця мережа існувала для обміну інформацією між комп'ютерами військових (науково-дослідних та навчальних) закладів. У 80-ті роки XX ст. її вже використовували в державних (навчальних, наукових) та комерційних цілях. У 90-ті роки XX ст. мережа Інтернет вийшла за межі США, і до неї приєдналися країни всіх континентів світу.

Варіант 4

Мова HTML

Основою WWW (World Wide Web) є Hyper Text Markup Language (HTML) — мова гіпертекстової розмітки. Мова HTML була запропонована Тімом Бернесом-Лі 1989 року як один із компонентів технології WWW, що призначався для опису вигляду документа.

Стандарт HTML було розроблено організацією W3C (World Wide Web Consortium) на початку 1996 року в кооперації

з такими фірмами, як *IBM, Microsoft, Netscape Communication Corporation, Novell, SoftQuad, Spyglass* та *Sun Microsystems*. Організація W3C спільно зі згаданими фірмами продовжує роботу над розширенням можливостей елементів мови **HTML** та її інтернаціоналізацією.

Варіант 5

ГІПЕРТЕКСТ

Окрім того, комп'ютер дає можливість користуватися не лише *лінійним* поданням матеріалу, як це робиться в традиційній книзі, а й *гіпертекстуальним*. Гіпертекст характеризується **нелінійністю** (користувач сам обирає шлях читання), **різномірністю та мультимедійністю** (застосування всіх засобів впливу на користувача).

Вважають, що *гіпертекстуальна структура викладення матеріалу* схожа на процес нашого мислення, який теж здійснюється нелінійно, і використання такої структури може значно збільшити **інтелектуальний потенціал особистості**.

Варіант 6

Природа і заповідники

Кримські гори на півдні півострова простягнулися трьома рівнобіжними грядами від *Севастополя* до *Феодосії* на 150 кілометрів, їхні південні схили майже стрімкі, а північні переходять у міжгрядові зниження або рівнини.

Дві невисокі гряди (*Зовнішня і Внутрішня*) на півночі складають **Кримське передгір'я**, розрізане на окремі масиви мальовничими річковими долинами. У західній частині

Передгір'я, що майже цілком розташовується в межах **Бахчисарайського району**, є півтора десятка знаменитих печерних міст і монастирів.

Значна кількість гrotів і скельних навісів у вапнякових обривах **Передгір'я**, а також сприятливий мікроклімат відіграли вирішальну роль у формуванні культури кам'яного віку.

Карпати

Велика гірська система **Карпат** починається недалеко від *Братислави (Словаччина)* і закінчується на південному сході поблизу *Залізних воріт (Румунія)*. Довжина **Карпат** становить приблизно 1500 км. Дугою із трьох сторін оточують вони *Середньоевропейську низовину*. У північно-західній частині їхня ширина становить 240 км, у південно-західній — 340, а в північно-східній, де розташовані українські **Карпати**, вона зужується до **100-120** км.

Карпати, а особливо *Косівщина*, — неповторно мальовничий край *високих гір, стрімких потоків і чистого повітря*. Своєрідність краю, краса його природних ландшафтів, багатий тваринний і рослинний світ **Карпат**, унікальна скарбниця природних екосистем, де збереглися рідкісні реліктові **види флори, фауни і, як ніде в Україні, народні звичаї**, традиційна побутова культура ваблять до себе численних туристів.

2. Створіть у текстовому редакторі HTML-сторінку, використовуючи теги, за допомогою яких настроюють відповідний до вибраного варіанта вигляд веб-сторінки.
3. Збережіть документ у файлі з розширенням **.html**.
4. Відкрийте створену веб-сторінку у вікні браузера.
5. Перегляньте веб-сторінку. У разі виявлення помилок у відтворенні тексту відредагуйте HTML-документ.
6. У висновку опишіть використані теги.

Практична робота № 4

Структурування веб-сторінки за допомогою таблиць

Мета роботи: навчитися створювати таблиці та розміщувати в їхніх комірках потрібну інформацію.

Порядок роботи

1. Розробіть HTML-документ, який матиме таблицю, що створена за такою схемою:

коричневий	синьо-фіолетовий
золотистий	
жовтий	
рожевий	

Зі схеми видно, що перший рядок складається з двох комірок: комірки коричневого кольору завширшки 60 % від ширини всієї таблиці, яка містить два стовпці, та синьо-фіолетової комірки, що об'єднує два рядки. Другий рядок складається лише з однієї комірки золотистого кольору. Третій рядок складається з комірки жовтого кольору шириною 30 % від ширини всієї таблиці та пурпурової комірки, що об'єднує два рядки і два стовпці. Четвертий рядок складається з однієї комірки рожевого кольору. Текст у комірках із темним тлом має білий колір.

Збережіть створений документ у файлі з розширенням .html. Відобразіть документ у вікні браузера (рис. 2.18).

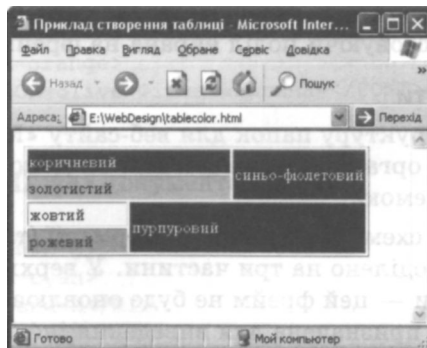


Рис. 2.18. Веб-сторінка, на якій міститься таблиця

4. Відредагуйте HTML-документ: скасуйте призначення меж таблиці та кольорів комірок, змініть розмір шрифту.

На рис. 2.19 показано, який вигляд матиме цей HTML-документ у вікні браузера. Назви кольорів розміщені у різних ділянках екрана.

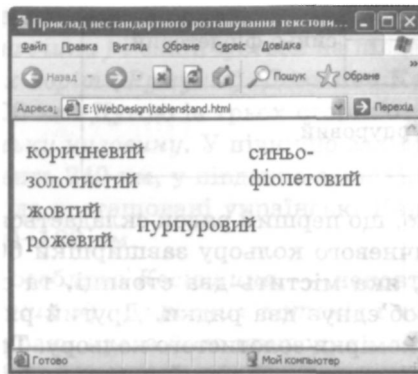


Рис. 2.19. Приклад розміщення тексту за допомогою таблиць

Практична робота № 5

Розроблення веб-сайтів із використанням фреймів

Мета роботи: навчитися створювати веб-сайт із навігаційною панеллю, використовуючи поділ екрана на фрейми.

Порядок роботи

1. Створіть структуру папок для веб-сайту «Моя фірма» з п'яти сторінок з організацією навігації за допомогою фреймів за заданою схемою.

Розгляньте схему розміщення інформації (табл. 2.10). Як бачите, екран поділено на три частини. У верхній лівій міститься назва фірми — цей фрейм не буде оновлюватися. Права верхня частина призначена для виведення інформації про основні напрями роботи фірми. Це найбільший фрейм, він постійно оновлюватиметься залежно від вибору користувачем гіперпосилання. Список посилань розташований у нижній частині екрана, яка буде навігаційною панеллю. Інформація в ній не

оновлюватиметься, відвідувач сайту використовуватиме її гіперпосилання для того, щоб довідатися про роботу фірми.

Таблиця 2.10. Схема поділу екрана на фрейми

Назва фірми	Основна інформація
Навігаційна панель-список текстових гіперпосилань із напрямками роботи певної фірми	

У папці веб-сайту будуть розміщені:

- HTML-документ, в якому записана назва фірми; назвіть його **logo.html**;
- HTML-документи, в яких описується, чим займається фірма; це будуть файли 1.html, 2.html, 3.html;
- HTML-документ — навігаційна панель, що реалізує список гіперпосилань у вигляді тексту, які посилаються на документи **1.html, 2.html, 3.html**; назвіть його **g.html**;
- головний файл веб-сайту — **index.html**, за допомогою якого екран буде поділено на фрейми, при цьому той із них, в якому відобразатимуться документи **1.html, 2.html, 3.html**, має отримати ім'я за допомогою атрибута NAME.

2. Створіть HTML-код документа **logo.html**.

```
<HTML>
<TITLE>Фірма «Калина»</TITLE>
<BODY BGCOLOR=orange>
<H4><FONT COLOR=olive>Фірма «Калина»</ГОБТ></H4>
</BODY>
</HTML>
```

3. Створіть HTML-код документа **1.html**.

```
<HTML>
<TITLE>Замовники фірми «Калина»</TITLE>
<BODY BGCOLOR=cyan>
<H1>Наші замовники</H1>
<H2>Фірма «Карпати»</H2>
<H2>Фірма «Золотий жук»</H2>
<H2>Фірма «Пролісок»</H2>
</BODY>
</HTML>
```

4. Створить HTML-код документа **2.html**.

```
<HTML>
<TITLE>Вироби фірми «Калина»</TITLE>
<BODY BGCOLOR=cyan>
<H1>Наша продукція</H1>
<H2>Шафи для вітальні</H2>
<H2>Жухонні стільці</H2>
<H2>Офісні столи</H2>
</BODY>
</HTML>
```

5. Створить HTML-код документа **3.html**.

```
<HTML>
<TITLE>Перспективи фірми «Калина»</TITLE>
<BODY BGCOLOR=cyan>
<H1>Наша майбутня продукція</H1>
<H2>Жомп'ютерні столи</H2>
<H2>Стилізовані акваріуми</H2>
<H2>Набір меблів для лоджії</H2>
</BODY>
</HTML>
```

6. Створить HTML-код головної сторінки **g.html**.

```
<HTML>
<TITLE>Напрями роботи фірми «Калина»</TITLE>
<BODY BGCOLOR=pink>
<H1>Напрямки роботи фірми «Калина»</H1>
<H3а href="1.html" target="frame1">Наші замовники</а>
</H3>
<H3а href="2.html" target="frame1">Наша продукція</а>
</H3>
<H3а href="3.html" target="frame1">Наша майбутня
продукція</а></H3>
</BODY>
</HTML>
```

7. Створить HTML-код головної сторінки **index.html**.

Зауважте, що значенням атрибута NAME в документі **index.html** буде frame1; те саме значення має бути в атрибута TARGET у документі **g.html**.

На рис. 2.20 показано головну сторінку сайту фірми «Калина», що відображається, коли користувач вибрав гіперпосилання на сторінку з інформацією про майбутню продукцію компанії.

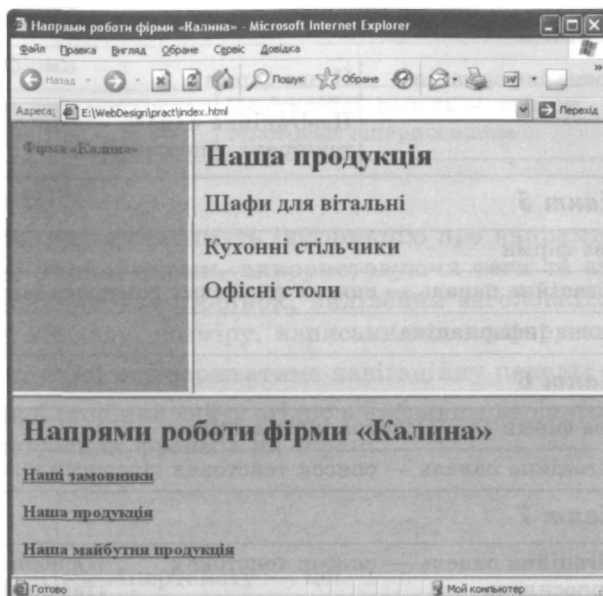


Рис. 2.20. Головна сторінка сайту фірми «Калина»

Самостійна робота

1. Виберіть варіант для схеми поділу сторінки на фрейми.

Варіант 1

Основна інформація	Назва фірми
Навігаційна панель — список текстових гіперпосилань	

Варіант 2

Навігаційна панель — список текстових гіперпосилань	Назва фірми
Основна інформація	

Варіант 3

Навігаційна панель — список текстових гіперпосилань	
Основна інформація	Назва фірми

Варіант 4

Основна інформація	Назва фірми
	Навігаційна панель — список текстових гіперпосилань

Варіант 5

Назва фірми
Навігаційна панель — • список текстових гіперпосилань
Основна інформація

Варіант 6

Назва фірми	Основна інформація
Навігаційна панель — список текстових гіперпосилань	

Варіант 7

Навігаційна панель -- список текстових гіперпосилань	Основна інформація
Назва фірми	

Варіант 8

Навігаційна панель — список текстових гіперпосилань	Основна інформація	Назва фірми
-----------------------------------------------------	--------------------	-------------

Варіант 9

Основна інформація	Назва фірми
Навігаційна панель — список текстових гіперпосилань	

Варіант 10

Назва фірми	Основна інформація
Навігаційна панель -- список текстових гіперпосилань	

Варіант 11

Навігаційна панель -- список текстових гіперпосилань	
Назва фірми	Основна інформація

Варіант 12

Назва фірми	
Основна інформація	Навігаційна панель — список текстових гіперпосилань

2. Створіть HTML-код:

- файлів, які міститимуть інформацію про напрями роботи певної уявної фірми, використовуючи теги та атрибути для задання тла сторінок, виділення заголовків, визначення вигляду, розміру, написання та кольору символів;
- файлу, який відтворюватиме навігаційну панель;
- головної сторінки сайту згідно з вибраним варіантом схеми розташування фреймів на екрані.

Тест

1. Мова розмітки гіпертексту — це:

- а) сукупність команд для настроювання Інтернету;
- б) спеціальні команди для створення зв'язку між певними веб-сторінками;
- в) сукупність та правила розташування спеціальних кодів, які керують форматуванням, вставленням ілюстрацій, таблиць і зв'язком з іншими документами;
- г) спеціальні команди та правила, призначені для створення веб-сторінок;
- д) засіб для відтворення веб-сторінок.

2. Тег — це:

- а) код для визначення структури веб-сторінки;
- б) програма керування виглядом текстових повідомлень та малюнків на веб-сторінці;
- в) будь-яке текстове повідомлення, що розташоване в кутових дужках;
- г) настроювання вигляду частини веб-сторінки за допомогою спеціального коду;
- д) команда у мові HTML.

3. Атрибут — це:
 - а) характеристика тегу, що впливає на створюваний ним ефект;
 - б) характеристика кольорів тексту та тла веб-сторінки;
 - в) характеристика тегу, що визначає місце його розташування в HTML-документі;
 - г) властивість тегу;
 - д) параметри відтворення веб-сторінки на екрані.
4. Навігаційна панель — це:
 - а) HTML-код головної веб-сторінки;
 - б) список гіперпосилань, розташований лише на головній сторінці;
 - в) один із фреймів головної сторінки, що містить гіперпосилання на всі розділи сайту і є постійно видимим для відвідувача;
 - г) зручний засіб перегляду розділів веб-сайту;
 - д) список розділів веб-сайту, кожний елемент якого є гіперпосиланням на відповідний розділ.
5. Тег призначений для:
 - а) настроювання вигляду тла частини веб-сторінки;
 - б) визначення місця розташування заголовка;
 - в) настроювання кольору, накреслення, розміру для групи символів;
 - г) позначення підкреслення для групи символів;
 - д) розмічення групи символів напівжирним шрифтом.
6. Тег <A> призначений для:
 - а) позначення нового абзацу;
 - б) позначення тексту як заголовка;
 - в) створення гіперпосилання;
 - г) позначення нумерованого списку;
 - д) позначення тексту як елемента списку.

7. Тег <BODY> призначений для:
- а) виділення початку та кінця вмісту документа, який буде виведено на екран;
 - б) розмічення важливих фрагментів тексту;
 - в) встановлення кольорів тексту та тла веб-сторінки;
 - г) вставлення нового рядка в таблицю;
 - д) позначення заголовка для вікна відображення документа.
8. Тег <TITLE> призначений для:
- а) визначення заголовка документа, в якому міститься його ключовий опис;
 - б) визначення заголовка вікна, в якому відобразатиметься документ;
 - в) позначення тексту як заголовка;
 - г) вставлення тексту в комірку таблиці;
 - д) позначення нового абзацу.
9. Теги <H1>...<H6> використовують для:
- а) визначення вигляду заголовка вікна, в якому відобразатиметься документ;
 - б) позначення тексту як заголовка;
 - в) позначення нового абзацу;
 - г) позначення заголовка для таблиці;
 - д) розмічення гіперпосилання.
10. Тег <TABLE> призначений для:
- а) опису таблиці;
 - б) вставлення нового рядка в таблицю;
 - в) вставлення тексту в комірку таблиці;
 - г) позначення заголовка для таблиці;
 - д) вставлення таблиці на веб-сторінку.
11. Тег <FRAMESET> призначений для:
- а) поділу сторінки на частини для відображення в них різних HTML-документів;
 - б) позначення того, який саме HTML-документ слід вмістити в поточний фрейм;

- в) позначення назви фрейму;
 - г) позначення заголовка для вікна відображення веб-сторінки;
 - д) налаштування висоти та ширини фрейму.
12. Позичувати об'єкти на веб-сторінці за допомогою таблиці можна:
- а) об'єднанням комірок таблиці по вертикалі та горизонталі;
 - б) вставленням кількох таблиць на веб-сторінку;
 - в) призначенням для різних комірок таблиці різного кольору тла;
 - г) розміщенням даних у комірках таблиці та встановленням межі таблиці завширшки нуль пікселів;
 - д) наданням межам таблиці того самого кольору, що й колір тла.

Розділ 3

Графіка, аудіо- та відеоінформація на веб-сторінках

Отже, ви вже знаєте, як створювати найпростіші веб-документи, що містять заголовки, звичайний текст, таблиці та текстові гіперпосилання. Однак погодьтеся, що сторінки, на яких розміщено лише текст, мають не надто привабливий вигляд і навряд чи викличуть у когось захоплення. Відвідайте кілька популярних сайтів у Всесвітній павутині, і ви переконаєтеся, що якісні, красиві веб-сторінки обов'язково містять графічні зображення, часто приваблюють погляд веселими анімаційними картинками, а інколи пропонують прослухати музику чи переглянути кліп або фільм.

У цьому розділі ви ознайомитеся з форматами зображень, аудіо- та відеофайлів, які використовують в Інтернеті, дізнаєтеся про те, як вставляти на веб-сторінки малюнки і працювати з картою гіперпосилань. На конкретних прикладах навчитеся створювати анімовані зображення та імпортувати їх у веб-документи, а також розміщувати на своїх сайтах мультимедійну інформацію. У практичній роботі, яку необхідно виконати, ви спробуєте оформити тло веб-сторінки у вигляді малюнка, розмістити на ній графічні об'єкти і зробити їх посиланнями.

Використання зображень у веб-документах

У більшості професійно створених сайтів використовують графічне оформлення, що дає змогу яскраво та наочно подати інформацію. Для цього на веб-сторінках розміщують відповідні графічні файли,

які можуть мати різні формати. Зображення мають бути розроблені у такий спосіб, щоб допомогти користувачу сприймати текстову інформацію та доповнювати її.

Розглянемо особливості популярних графічних форматів, а також засоби розміщення та вирівнювання зображень на веб-сторінках.

Формати зображень, які використовують в Інтернеті

Найбільш розповсюдженими в Інтернеті є файли форматів GIF, JPG (JPEG) та PNG. Це найкомпактніші формати збереження графіки.

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 вміст 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.

Розміщення і вирівнювання зображень на веб-сторінках

Ілюстрації відіграють важливу роль в оформленні веб-сторінок: їх використовують як тло, виводять на екран як окремо від тексту, так і разом із ним. Малюнки зберігаються в окремих файлах поза HTML-документами, але відображаються браузером всередині веб-сторінки.

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 **none.jpg**.

```
<HTML>  
<TITLE>Моя фірма</TITLE>
```

```

<BODY BACKGROUND="mone.jpg">
<H1><A HREF="1.html">Наша продукція</AX/H1>
<H1><A HREF="2.html">Наші замовники</AX/H1>
</BODY>
</HTML>

```

На рис. 3.1 показано, який вигляд має ця веб-сторінка у вікні браузера.

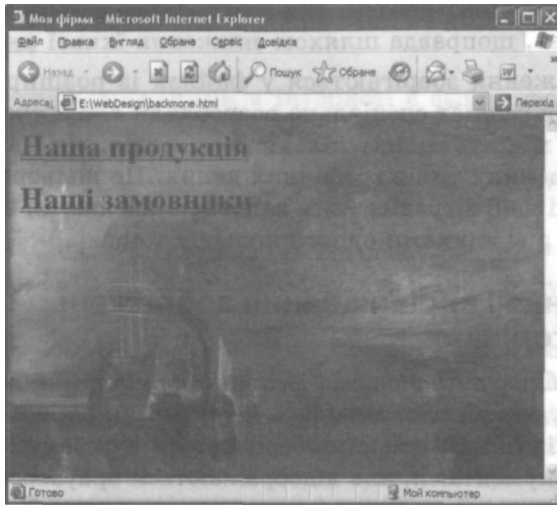


Рис. 3.1. Веб-сторінка, тлом якої є малюнок

Як бачите, малюнок займає цілий екран. Розглянемо інший приклад, задавши в атрибуті `BACKGROUND` зображення невеликого розміру `gorobci.jpg`, яке у вигляді мозаїки заповнить екран. Код такої сторінки наведено нижче, а вигляд її у вікні браузера — на рис. 3.2.

```

<HTML>
<TITLE>Моя фірма</TITLE>
<BODY BACKGROUND="gorobci.jpg">
</BODY>
</HTML>

```

У цьому прикладі на веб-сторінці немає тексту, однак на практиці так не буває. З огляду на це тло рідко роблять у вигляді фотографій, оскільки для них непросто підібрати відповідний колір тексту.

Зображення для тла попередньо готують у графічних редакторах, щоб зробити його неяскравим і неконтрастним.

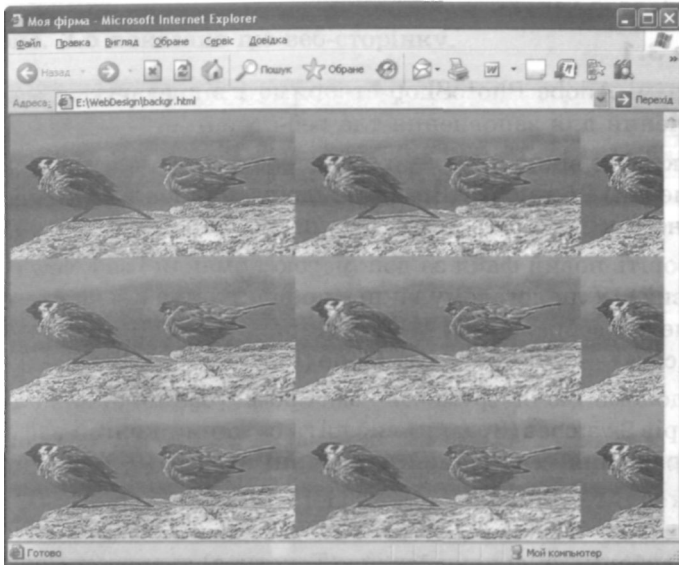


Рис. 3.2. Веб-сторінка, тлом якої слугує малюнок, що заповнив екран у вигляді мозаїки

Можливість відповідної обробки зображень для перетворення їх на фоніві забезпечують багато графічних редакторів, зокрема популярний Adobe PhotoShop. Відкривши у ньому потрібний файл (командою File • Open (Файл • Відкрити)), слід вибрати команду Image • Adjustments • Hue/Saturation (Зображення • Параметри • Тон/Насиченість) і за допомогою шкали Lightness (Освітленість) освітлити малюнок тла. На рис. 3.3 показано малюнок gorobcs1.jpg, який оброблений у такий спосіб.

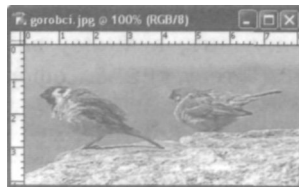


Рис. 3.3. Малюнок gorobcs1.jpg для тла

Але найчастіше підбирають рисунок, який нагадує шпалери, адже текст на тлі має виділятися і легко читатись, тобто привертати увагу відвідувача сайту.

Вправа 3.1

У програмі Adobe PhotoShop створимо і відредагуємо малюнок, призначений для заповнення гла веб-сторінки.

1. Відкрийте вікно програми. Простежте, щоб палітри **Tools** (Інструменти) і **Swatches** (Зразки) відображалися на екрані. Якщо це не так, виберіть назви цих палітр у меню **Window** (Вікно).
2. Створіть новий файл за допомогою команди **File • New** (Файл • Новий). У діалоговому вікні, що відкриється, задайте такі параметри зображення: **Width** (Ширина) — **150** пікселів. **Height** (Висота) — **100** пікселів, **Mode** (Режим) — **RGB Color**.
3. За допомогою інструмента **Paint Bucket** (Заливка) на палітрі кольорів **Swatches** (Зразки) виберіть бажаний колір і зафарбуйте зображення, клацнувши всередині нього.
4. Виконайте команду меню **Filter > Texture > Texturizer** (Фільтр • Текстура • Текстуризатор). У діалоговому вікні, що відкриється, виберіть зі списку **Texture** (Текстура) значення **Canvas** (Полотно) і настройте параметри **Scaling** (Шкала), **Relief** (Рельєф) та **Light Direction** (Напрямок освітлення). Зображення набуде такого вигляду, як на рис. 3.4.

Для корекції отриманого малюнка (надання йому блідого тону) можна скористатися командою **Image • Adjustments > Hue/Saturation** (Зображення • Параметри • Тон/Насиченість) і настроїти освітленість за допомогою шкали **Lightness** (Освітленість), як це описано раніше.

5. Збережіть цей малюнок у файлі з іменем **1.jpg**, розмістивши його у папці з веб-сторінками. Закрийте програму Adobe PhotoShop.
6. Створіть HTML-документ, в якому за допомогою атрибута **BACKGROUND="1.jpg"** теги **<BODY>** оформіть тло сторінки.

```
<HTML>
<TITLE>Моя фірма</TITLE>
<BODY BACKGROUND="1.jpg">
<H1><A HREF="1.html">Наша продукція</A></H1>
```

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

На рис. 3.5 показано цю веб-сторінку.

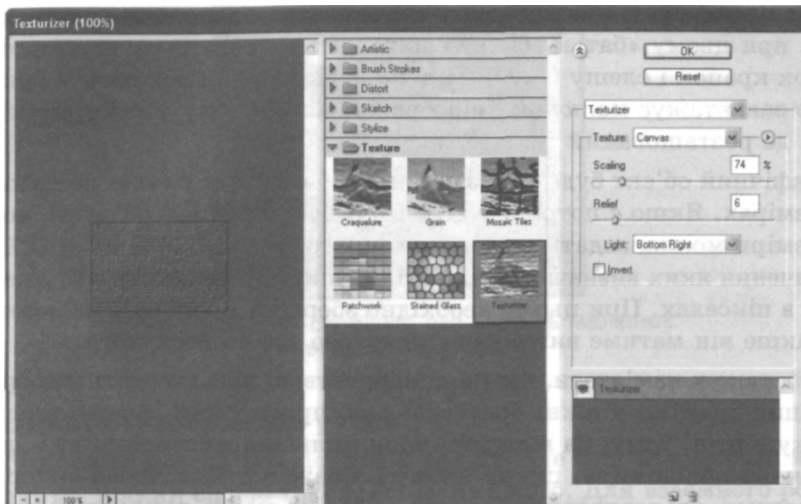


Рис. 3.4. Створення текстури

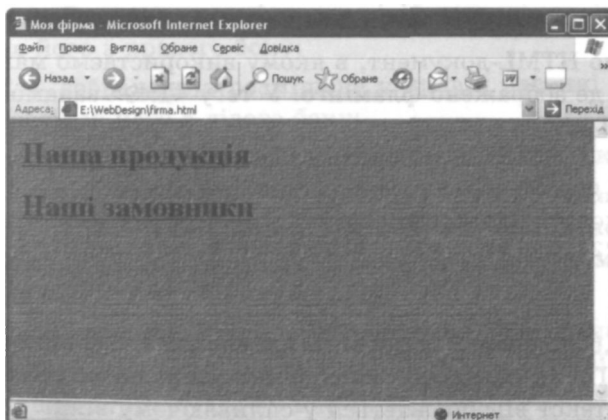


Рис. 3.5. Веб-сторінка, тло якої розроблено за допомогою програми Adobe PhotoShop

Вставлення зображень на веб-сторінку

Для розміщення малюнків у HTML-документі використовують одиницарний тег . 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 .

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 сплюснутого.

Слід також пам'ятати, що іноді користувачі відключають відображення графіки у вікні браузера, щоб прискорити завантаження документів. Тому, на випадок, коли малюнка на сторінці не буде, бажано повідомити, що на ньому зображено. Для цього використовують альтернативний текст — більш-менш докладний опис зображення, який задають у тегу 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 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>
```

На рис. 3.6 показано, який вигляд має ця веб-сторінка у вікні браузера. Після наведення вказівника миші на малюнок альтернативний текст відображається у спливаючому вікні.

У режимі відключення графіки буде показано лише порожню рамку зображення з альтернативним текстом.

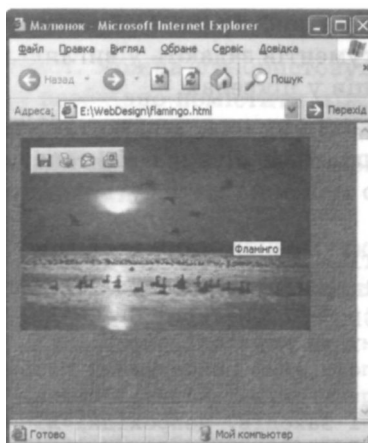


Рис. 3.6. Веб-сторінка, яка містить малюнок і альтернативний текст

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 , 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.

За умовчанням атрибут ALIGN має значення left.

Для того щоб картинка не була розташована впритул до тексту, можна задати відступи. Це роблять за допомогою атрибута HSPACE

для бокових полів та атрибута VSPASE для верхнього і нижнього полів. Значення аргументів задають у вигляді чисел, що визначають розміри відступів у пікселях.

У прикладі, наведеному нижче, задано такі значення атрибутів, що малюнок розміщується ліворуч від тексту, який обтікатиме його. Поля навколо малюнка матимуть розмір 10 пікселів.

```
<HTML>
<TITLE>МАЛЮНОК</TITLE>
<BODY BACKGROUND="1.jpg">
<IMG SRC="flamingo.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°С.
</BODY>
</HTML>
```

На рис. 3.7 показано, який вигляд має ця веб-сторінка у вікні браузера.



Рис. 3.7. Взаємне розміщення тексту і графіки

Графічні гіперпосилання

Зображення, як і текст, можна використовувати як посилання. Для цього тег `` необхідно помістити між тегами `<A>` і ``. Зображення-посилання має синю рамку, а після наведення на нього вказівника миші той набуває такої самої форми, що й у випадку текстового посилання.

Приклад 3.1. Використання зображень як посилань

Розглянемо приклад веб-сторінки, що містить малюнки, які є гіперпосиланнями.

```
<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>
```

Малюнок **gorobci.jpg** пов'язаний із файлом **gorobci.html**, а малюнок **flamingo.jpg** — із файлом **flamingo.html**. На рис. 3.8 показано, який вигляд має ця веб-сторінка у вікні браузера.

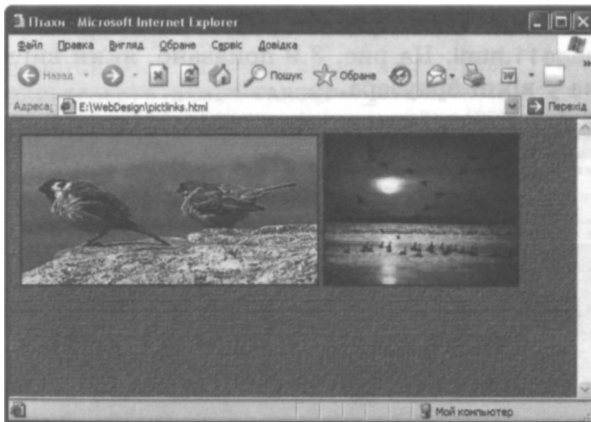


Рис. 3.8. Веб-сторінка, що містить малюнки, які є гіперпосиланнями

Ще одним способом використання малюнків як гіперпосилань є створення графічних кнопок переходу.

Приклад 3.2. Використання кнопок переходу

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>
```

Кнопка Назад пов'язана з файлом page9.html, а кнопка Вперед — з файлом page11.html. На рис. 3.9 показано, який вигляд має ця веб-сторінка у вікні браузера.

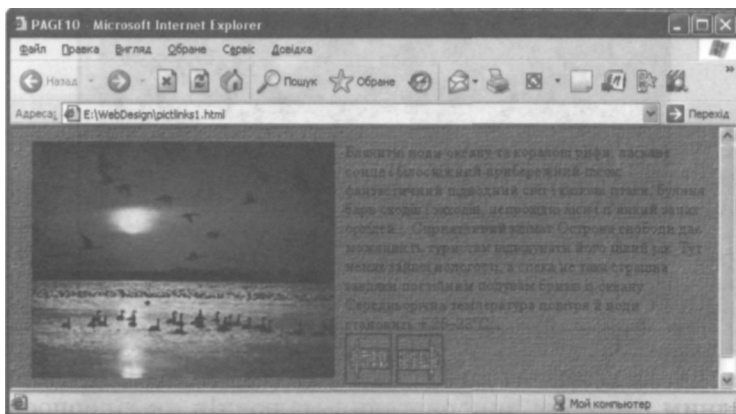


Рис. 3.9. Веб-сторінка, на якій розміщено стрілки переходу

Під час створення сайтів зображення часто вставляють у комірки таблиці. У цьому випадку малюнки будуть розташовані на екрані так, як задумав дизайнер веб-сайту. У такий спосіб створюють, наприклад, фотогалереї: зображення невеликих розмірів вміщують у комірки таблиці та роблять їх гіперпосиланнями, кожне з яких пов'язане з тим самим фото, але більшого розміру.

Приклад 3.3. Створення фотогалереї

Створимо у HTML-документі таблицю, в комірки якої вставимо ілюстрації до веб-сайту навчального закладу.

```
<HTML>
<BODY>
<TABLE>
<TR><TD><A HREF="foto/1.html" TARGET="_blank"><IMG
SRC="foto/1.jpg" WIDTH=180 HEIGHT=150 ALT="Тренажерний зал">
</TD>
<TD><A HREF="foto/2.html" TARGET="_blank"><IMG SRC="foto/2.jpg"
WIDTH=180 HEIGHT=150 ALT="Гимнастика в нашій школі"></TD>
<TD><A HREF="foto/3.html" TARGET="_blank"><IMG SRC="foto/3.jpg"
WIDTH=180 HEIGHT=150 ALT="Нарівні з чемпіонами"></TD>
<TD><A HREF="foto/4.html" TARGET="_blank"><IMG SRC="foto/4.jpg"
WIDTH=180 HEIGHT=150 ALT="КВК"></TD></TR>
<TR><TD><A HREF="foto/5.html"
TARGET="_blank"><IMG SRC="foto/5.jpg"
WIDTH=180 HEIGHT=150 ALT="Шкільний хор"></TD>
<TD><A HREF="foto/6.html" TARGET="_blank"><IMG SRC="foto/6.jpg"
WIDTH=180 HEIGHT=150 ALT="Наші таланти"></TD>
<TD><A HREF="foto/7.html" TARGET="_blank"><IMG SRC="foto/7.jpg"
WIDTH=180 HEIGHT=150 ALT="Відкриття Золотої пекторалі"></TD>
<TD><A HREF="foto/8.html" TARGET="_blank"><IMG SRC="foto/8.jpg"
WIDTH=180 HEIGHT=150 ALT=" Наші таланти"></TD></TR>
</TABLE>
</BODY>
</HTML>
```

Усі зображення розміщені в поточній папці в підкаталозі foto. Кожний малюнок, вставлений у комірку таблиці, є посиланням, що вказує на HTML-документ, тіло якого містить тег вставляння того самого малюнка-гіперпосилання. Наведемо приклад одного з таких документів — 3.html.

```
<HTML>
<TITLE>Нарівні з чемпіонами</TITLE>
```

```
<BODY>
<IMG SRC="3.jpg" WIDTH=900 HEIGHT=750>
</BODY>
</HTML>
```

Отже, малюнок у комірці таблиці та пов'язаному з ним HTML-документі однаковий, але в таблиці його розміри значно менші реальних — 180x150 пікселів. Кожний такий малюнок супроводжується альтернативним текстом.

На рис. 3.10 зображена веб-сторінка, фрагмент коду якої було наведено вище. На ній розташовано фотогалерею веб-сайту гімназії «Сихівська» м. Львова.

HTML-документи, пов'язані з гіперпосиланнями, відкриваються в окремому вікні: цього досягають наданням атрибуту TARGET значення blank. Зображення в них мають суттєво більші розміри, наприклад, ширина — 900 пікселів, а висота — 750. Розміри малюнка, що відкривається в окремому вікні, підбирають так, щоб він мав ширину та висоту, достатні для детальнішого ознайомлення з ним. При цьому слід пильнувати, щоб не відбулося спотворення картинки, тому часто розміри залишають реальними, не використовуючи атрибутів WIDTH та HEIGHT.



Рис. 3.10. Веб-сторінка, яка містить фотогалерею

На рис. 3.11 показано, який вигляд має вікно браузера після того, як користувач вибрав одну з фотографій для детальнішого ознайомлення.



Рис. 3.11. Зображення, відкрите в окремому вікні для детальнішого ознайомлення

Створення GIF-анімацм

В Інтернеті розміщено величезну кількість файлів із готовими анімаційними зображеннями. Проте цікаво навчитися самостійно створювати рухомі картинки для прикрасення ними свого сайту.

Одним із перших графічних форматів, що підтримує анімацію, став GIF. Цей формат дозволяє зберігати у файлі кілька зображень, які можуть відображатися по чергову, як слайд-шоу або невеликий фільм. На відміну від звичайного фільму, в якому швидкість відтворення задана кількістю кадрів за секунду, у GIF-файлі зберігаються параметри, що визначають, у який спосіб і як довго буде демонструватися кожна картинка. GIF-зображення можуть

бути різного розміру, розміщуватись у потрібних позиціях екрана й відтворюватися незалежно від інших.

Розглянемо процес створення GIF-анімації у програмах Adobe PhotoShop та Adobe ImageReady.

Приклад 3.4. Створення простої GIF-анімації

Створимо нескладну анімацію, що складалася б лише з двох кадрів, які імітують політ птаха. На рис. 3.12 зображено перший та другий кадри майбутнього рухомого малюнка, що зберігаються відповідно у файлах 1.jpg та 2.jpg.

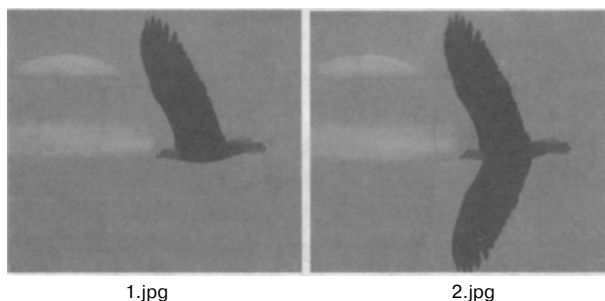


Рис. 3.12. Перший та другий кадри GIF-анімації

Відкрийте програму Adobe PhotoShop. Простежте, щоб відображалися палітри Tools (Інструменти) та Layers (Шари). Якщо їх немає, виберіть назви палітри у меню Window (Вікно). Створіть новий файл за допомогою команди File • New (Файл • Новий). У діалоговому вікні, що відкриється, задайте такі параметри зображення: Width (Ширина) і Height (Висота) — такі самі, як у кадрів 1.jpg та 2.jpg, Mode (Режим) — RGB Color, Background Contents (Заповнення тла) — Transparent (Прозоре).

Виконайте команду File • Open (Файл • Відкрити) і відкрийте файли 1.jpg та 2.jpg.

Використовуючи інструмент Move (Перемістити), почергово перетягніть зображення файлів 1.jpg та 2.jpg у вікно створеного малюнка. Матимете у ньому три шари: Layer 1 (Перший) — початкове зображення. Layer 2 (Другий) — малюнок 1.jpg та Layer 3 (Третій) — 2.jpg. Усі ці шари відображені в палітрі Layers (Шари) (рис. 3.13).

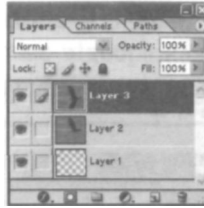


Рис. 3.13. Відображення шарів у палітрі Layers (Шари)

Використовуючи кнопку Edit in ImageReady (Редагувати в ImageReady), розташовану внизу палітри Tools (Інструменти), імпортуйте отримане тришарове зображення у програму Adobe ImageReady. Простежте, щоб були відкриті палітри Tools (Інструменти), Layers (Шари) та Animation (Анімація). У палітрі Animation (Анімація) автоматично з'явиться перший кадр рухомого зображення. За допомогою індикаторів видимості, розміщених ліворуч від кожного шару на палітрі Layers (Шари), задайте відображення лише другого шару (рис. 3.14). Так ви настроїте зображення першого кадру анімації.



Рис. 3.14. Настроювання першого кадру анімації

5. Додайте до анімації новий кадр. Для цього на палітрі Animation (Анімація) клацніть кнопку Duplicates current frame (Дублювати поточний кадр). Буде додано другий кадр — копію першого, який слід настроїти, задавши за допомогою індикаторів

видимості на палітрі Layers (Шари) відображення лише третього шару (рис. 3.15).

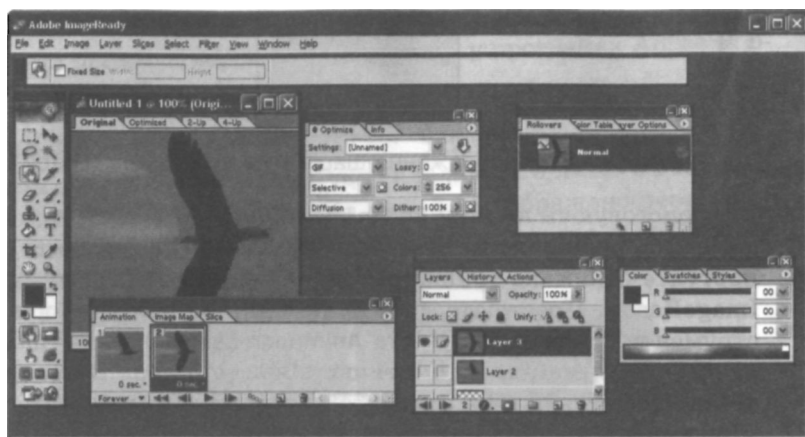


Рис. 3.15. Створення двох кадрів анімації у програмі Adobe ImageReady

Задайте час затримки кадрів за допомогою піктограм, розмішених під кожним із кадрів (рис. 3.16).

Скориставшись кнопкою Plays/stops animation (Програвання/зупинка анімації) на палітрі Animation (Анімація), перегляньте створене рухоме зображення. У разі необхідності відкоригуйте час затримки.

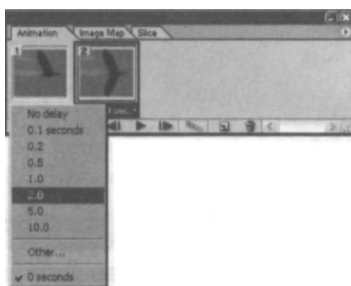


Рис. 3.16. Визначення часу затримки кадрів

7. За допомогою команди File • Save Optimized (Файл • Зберегти з оптимізацією) збережіть зображення як файл GIF-анімації з розширенням .gif.

Якщо таке зображення помістити в HTML-документ, то картинка змінюватиметься, почергово відображаючи кадри, на яких можуть бути не лише малюнки, а й слова.

Карти посилань

Якщо тег `` з ім'ям файлу рисунка розмістити між тегами `<A>` і ``, 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 `` 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.

Приклад визначення карти посилань наведено нижче.

```
<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 зазначити координати всіх кутів. Припусти-

МО, 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.

Пояснимо використання карти гіперпосилань на прикладі.

Вправа 3.2

Створимо сайт «Моя фірма». Його головна сторінка виконана у вигляді графічного зображення, певні частини якого є гарячими областями. Вибравши їх, відвідувач може побачити інформацію про напрями роботи фірми: «Замовники», «Продукція», «Майбутнє».

У папку веб-сайту слід помістити такі файли:

- графічний файл `map.jpg` (рис. 3.17);
- HTML-документи `1.html`, `2.html`, `3.html`, що містять описи напрямів роботи фірми;
- HTML-документ `map.html`, в якому для гарячих областей малюнка `map.jpg` визначені гіперпосилання на документи `1.html`, `2.html`, `3.html`.



Рис. 3.17. Графічний файл `тар.jpg`, де зазначено координати областей

Гарячі області є прямокутними, а отже, кожна з них буде задана за допомогою чотирьох чисел — координат діагоналей уявних прямокутників.

1. За допомогою програми Paint або Adobe PhotoShop створіть файл `тар.jpg`.
2. У редакторі Блокнот створіть документи, де описано напрями роботи фірми, і збережіть їх як файли `1.html`, `2.html`, `3.html`.

```
<HTML>
<TITLE>Замовники фірми "Калина"</TITLE>
<BODY BGCOLOR=cyan>
<H1>Наші замовники</H1>
<H2>Фірма "Карпати"</H2>
<H2>Фірма "Золотий жук"</H2>
<H2>Фірма "Пролісок"</H2>
</BODY>
</HTML>
```

```
<HTML>
<TITLE>Вироби фірми "Калина"</TITLE>
<BODY BGCOLOR=cyan>
<H1>Наша продукція</H1>
<H2>Шафи для вітальні</H2>
<H2>Кухонні стільці</H2>
<H2>Офісні столи</H2>
</BODY>
</HTML>
```

```
<HTML>
<TITLE>Перспективи фірми "Калина"</TITLE>
<BODY BGCOLOR=cyan>
<H1>Наша майбутня продукція</H1>
```

```
<H2>Комп'ютерні столи</H2>
<H2>Стилізовані акваріуми</H2>
<H2>Набір меблів для лоджії</H2>
</BODY>
</HTML>
```

3. Створіть файл map.html, який містить такий код:

```
<HTML>
<TITLE>Напрями роботи фірми "Калина" </TITLE>
<BODY BGCOLOR=pink>
<IMG SRC="map.jpg" USEMAP="#kalyna">
<MAP NAME="kalyna">
<AREA HREF="1.html" ALT="РОЗПОВІДЬ ПРО ЗАМОВНИКІВ"
SHAPE="rect" COORDS="70,250,160,270">
<AREA HREF="2.html" ALT="РОЗПОВІДЬ ПРО ПРОДУКЦІЮ"
SHAPE="rect" COORDS="260,140,345,160">
<AREA HREF="3.html" ALT="РОЗПОВІДЬ ПРО МАЙБУТНЄ"
SHAPE="rect" coords="2,4,98,36">
</MAP>
</BODY>
</HTML>
```

4. Відкрийте файл map.html у вікні браузера. На рис. 3.18 показано карту посилань.

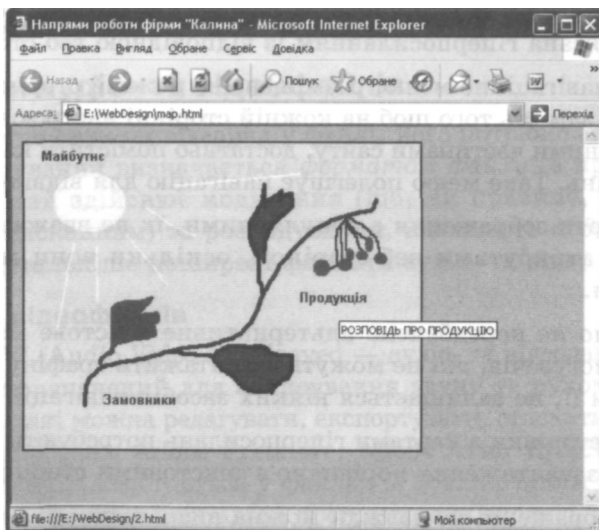


Рис. 3.18. Документ map.html, що містить карту гіперпосилань

На перший погляд, малюнок на веб-сторінці звичайний: про гіперпосилання нагадує лише синя рамка зображення, гарячі області ніяк не виділяються. Тому до карт гіперпосилань потрібно додавати пояснювальний текст до або після малюнка, який дасть змогу відвідувачу зорієнтуватись у подальший діях. Тег із пояснювальним текстом може бути, наприклад, таким:

```
<H3>Виберіть на малюнку напрям роботи фірми</H3>
```

Якщо розмістити його між тегами та <MAP>, то на веб-сторінці під картою посилань відобразиться напис «Виберіть на малюнку напрям роботи фірми». Провівши мишею над малюнком, можна виявити, що над гарячими областями вказівник миші змінює свій вигляд, як над звичайними посиланнями. Якщо ж зупинити його над гарячою областю, побачите спливаючий напис — результат дії атрибута ALT у тегу <AREA>.

Переваги та недоліки карт гіперпосилань

Кarti-зображення доцільно використовувати в таких випадках.

- Для побудови просторових зв'язків, зокрема географічних, які не просто задати окремими кнопками або текстом. Як приклад можна навести карту країни, поділену на області, кожна з яких пов'язана гіперпосиланням із відповідною веб-сторінкою.
- Як навігаційне меню, розмішене на кожній сторінці. Наприклад, замість того щоб на кожній сторінці настроювати зв'язки з різними частинами сайту, достатньо помістити карту гіперпосилань. Таке меню полегшує навігацію для відвідувача сайту.

Хоча карти-зображення є популярними, їх не вважають обов'язковими атрибутами веб-сторінок, оскільки вони мають певні недоліки.

- Якщо не передбачене альтернативне текстове меню, то для користувачів, які не можуть завантажити графіку або відключили її, не залишається ніяких засобів навігації.
- Веб-сторінки з картами гіперпосилань потребують більше часу для завантаження порівняно з текстовими сторінками.
- У зображенні не завжди можна визначити області, які є гарячими.

- У разі використання карт-зображень браузер не має можливості відзначити іншим кольором вже використані графічні посилання так, як це робиться для текстових посилань.

Отже, перш ніж обрати карту посилань як засіб навігації сайтом, слід замислитися над можливими наслідками. Карта є статичною і не змінюється під час перемішування відвідувача сайтом. Оскільки вона займає багато місця, то знижується ефективність використання екрана монітора. З іншого боку, текст, хоч і займає небагато місця, але не так привертає увагу як зображення, і є менш привабливим. Тому, обираючи той чи інший інтерфейс користувача, слід враховувати всі «за» і «проти».

Розміщення та відтворення на веб-сторінках мультимедійних даних

Використання мультимедійних даних на веб-сторінках робить їх наочними, легкими для сприйняття. Крім того, завдяки поданню інформації не у вигляді тексту, який потрібно уважно читати, а у формі аудіозаписів та відеозображень, можна скоротити час перегляду. Завдяки спеціальним модулям, вбудованим у браузер, аудіо- та відеофайли можуть відтворюватися прямо у його вікні.

Формати аудіо- та відеофайлів

Щоб зберегти звук чи відеоряд у файлі, його потрібно закодувати. Спосіб кодування визначається *форматом файлу*, а програмний модуль, який здійснює кодування (що, як правило, супроводжується стисканням) та розкодування, називають *кодеком*. Розглянемо детальніше поширені формати аудіо- та відео файлів.

Формати відеофайлів

Формат AVI (Audio Video Interleaved — аудіо- та відеодані, що чергуються) призначений для записування звуку та рухомих зображень. AVI-дані можна редагувати, експортувати, стискати, використовуючи програми Adobe Premiere, Adobe After Effects та інші. AVI-інформація зберігається у файлах із розширенням .avi. Для її відтворення потрібна спеціальна програма, яка входить у комплект поставки деяких браузерів. Її можна також встановити додатково.

Технологія MOV, або QuickTime, призначена для створення, зберігання та відтворення мультимедійних даних. Вона дає змогу об'єднувати звук, текст, анімацію та відео в одному файлі. MOV-інформація зберігається у файлах із розширенням .mov. Програмне забезпечення для її відтворення також постачається у комплекті з більшістю браузерів.

MPEG (Moving Pictures Experts Group — група експертів з обробки рухомих зображень) розробила стандарт стиснення відео- та аудіоданих. Для перегляду MPEG-даних за допомогою браузера слід використовувати додатковий модуль. Формат має кілька версій, від MPEG-1 до MPEG-4. Відео файли цього формату мають розширення .mpeg.

Формат ASF (Advanced Streaming Format — розширений формат поточкових даних) розроблено корпорацією Майкрософт для файлів, що містять потокове аудіо та відео (поточкову технологію буде розглянуто нижче). Файли цього формату зазвичай мають розширення .asf.

Формати аудіофайлів

У сучасних умовах для записування і відтворення звуку найчастіше використовують формати WAV, AJFF, MIDI, MP3. Розглянемо деякі з них.

Формат WAV (Windows Audio) був створений корпорацією Майкрософт і прийнятий як стандарт для звукового супроводу роботи системи і комп'ютерних ігор. WAV-дані зберігаються у файлах із розширенням .wav у нестиснутому вигляді. Ці файли містять інформацію про кількість доріжок, режим (моно або стерео), швидкість запису.

AIFF (Audio Interchange File Format — формат обміну аудіоданими) — це популярний формат платформи Macintosh, його також використовують для запису аудіоінформації на компакт-диск. Дані AIFF-формату зберігаються у файлах із розширенням .aif, де міститься також інформація про моно- чи стереорежим, швидкість запису.

Формат MP3 (MPEG-1 Audio Layer-3) має високий ступінь стиснення даних і дає змогу створювати файли невеликого розміру (вони мають розширення .mp3). Завдяки цій властивості MP3 сьогодні є найпоширенішим форматом зберігання аудіозаписів у Інтернеті.

Технології та засоби відтворення мультимедіа

Звичайно, для використання звукових та мультимедійних можливостей комп'ютера слід потурбуватися про те, щоб він був обладнаний звуковою картою та колонками. Щодо самого процесу відтворення, то він має низку особливостей. Файли більшості форматів починають відтворюватись лише після завершення їх завантаження. Є й інший спосіб передавання та відтворення файлів мультимедіа — у режимі реального часу. Таку технологію називають *поточною*. Інформацію можна отримувати безпосередньо від джерела даних, зокрема з відеокамери або файлу на сервері. Дані відтворюються в міру їхнього надходження, копія на жорсткому диску комп'ютера користувача не створюється. Передавання поточкових мультимедійних даних схоже на трансляцію телевізійних та радіопередач: користувач може приймати одну передачу, потім переключитися на інший канал або взагалі припинити приймання.

Приймання поточкових мультимедійних даних має кілька переваг порівняно зі звичайним завантаженням файлів з веб-сервера.

- **Негайне відтворення.** У разі приймання інформації в поточковому режимі фрагмент даних відтворюється відразу після його отримання.
- **Можливість передавання поточних подій.** Передавання даних у поточковому режимі зручно використовувати, наприклад, для новин або репортажів зі спортивних змагань.
- **Можливість передавання великих обсягів даних.** У поточковому передаванні даних не діють обмеження на довжину файлу, що передається.

Однак у цьому випадку копії мультимедійного файлу на комп'ютері створено не буде, і для його повторного відтворення потрібно знову зв'язуватися з відповідним веб-сервером.

Наразі поширеними є такі поточкові технології.

- **RealAudio/Video** — це технологія поточкового передавання аудіо-та відеоінформації, розроблена компанією Progressive Networks. Для відтворення даних необхідний додатковий програмний модуль RealPlayer. Файли, призначені для обробки засобами RealAudio/Video, мають розширення **.ra**, **.ram**, **.rm**, **.rmm**, **.rmd**.

- QuickTime Streaming Server — підтримує потокове передавання відео, аудіо, тексту та MIDI-інформації.
- Windows Media Server — це комплект цифрових компонентів для підтримування роботи з мультимедійними даними, що надає користувачам повний набір засобів для роботи з мультимедіа. Вони забезпечують відтворення аудіоінформації, що записана на компакт-диску, дають змогу працювати з аудіо- та відеоданими, які розташовані на веб-сервері, підтримують завантаження таких даних у потоковому режимі, надають низку додаткових можливостей. Окрім формату Windows Media ця технологія підтримує формати WAV, AVI, MIDI, MPEG, VOD, AIFF, MPS.

Однією з найпопулярніших програм відтворення мультимедійних даних є Програвач Windows Media (Windows Media Player). Це універсальний програвач для прослуховування аудіо- та перегляду відеофайлів більшості популярних форматів. Файли можуть міститися як на комп'ютері чи компакт-дисках, так і на веб-сторінках.

Як і у вікнах більшості Windows-програм, у вікні програвача Windows Media (рис. 3.19) є меню, область відображення, панелі задач та керування, рядок стану.



Рис. 3.19. Вікно програвача Windows Media

у центральній частині вікна розміщена область відображення. Програма демонструє у ній відеозапис із Інтернету або локального файлу, виводить візуальний супровід відтворюваного аудіозапису або рекламне оголошення.

Для керування процесом відтворення використовують елементи панелі керування. Показчик смуги пошуку пов'язаний із поточним місцем у записі: пересуваючи його, можна відтворювати будь-який потрібний фрагмент (але це не стосується потокових даних). Нижче розташовані елементи керування звуком і відтворюваними об'єктами, а також рядок стану з допоміжною інформацією.

Використання мультимедіа на веб-сторінках

Internet Explorer може завантажувати та відтворювати фоновий звук, для прослуховування якого не потрібно виконувати жодних дій. Звук зберігається у файлі. Для вставлення фонового звуку використовують тег такого формату:

```
<BGSOUND SRC="URL звукового файлу" LOOP=кількість відтворень>
```

Атрибут LOOP може набувати значень:

- true — повторення звуку доти, доки сторінка відображається на екрані;
- false — відтворення звукового файлу один раз;
- *число* — кількість відтворень.

Наприклад: `<BGSOUND SRC="fonzvuk.au" LOOP=3>`.

У HTML-документах можна також використовувати посилання на звукові та відеофайли, які відтворюватимуться лише у разі вибору цих посилань. Окрім цього, є спеціальний тег для розміщення панелі програвача на сторінці відразу після її завантаження у браузер. Однак слід пам'ятати, що мультимедійні файли можуть бути великими за обсягом, потребувати багато часу для завантаження, тому бажано повідомляти відвідувачів про розміри аудіо- та відео-записів, щоб вони вирішили, чи варто витратити свій час.

Розглянемо, як розміщують посилання на аудіо- та відеофайли. Якщо, наприклад, у поточній папці є файл кліпу school.avi, то посилання на нього можна задати так:

```
<A HREF="school.avi">Відеокліп про школу (600 K)</A>
```

Після клацання мишею гіперпосилання та надання дозволу на відкриття файлу з'явиться вікно програвача для відтворення цього відеокліпу.

Атрибут `DYNSRC` тегу `` дає змогу вбудувати відео у такий спосіб: на веб-сторінці міститься картинка, після наведення на яку вказівника миші починається відтворення відеокліпу. Ось зразок такого тегу:

```
<IMG SRC="1.jpg" DYNSRC="video.avi" START=MOUSEOVER LOOP=7>
```

Розглянемо приклад розміпнення звукового файлу **audio.wav** за допомогою тегу `<EMBED>`, який дає змогу розмістити на веб-сторінці спеціальну панель програвача мультимедійних файлів. Для цього використовують теги `<EMBED SRC=. . .></EMBED>`, наприклад, так:

```
<EMBED SRC="audio.wav"></EMBED>
```

Файл **audio.wav** у цьому прикладі має бути збережений у поточній папці (тій самій, що й HTML-документ).

Тег `<EMBED>` може мати такі атрибути:

- `SRC` (значення — URL-адреса) — адреса кліпу;
- `ALIGN` (набуває значень `left`, `right`, `top`, `middle`, `bottom`) — вирівнювання панелі програвача щодо тексту;
- `WIDTH` (у пікселях) — ширина програвача;
- `HEIGHT` (у пікселях) — висота програвача;
- `AUTOSTART` (набуває значень `true` або `false`) — налаштування автоматичного запуску після завантаження;
- `REPEAT` (значення `true` або `false`) — налаштування повторного програвання;
- `PLAY_LOOP` — кількість повторень;
- `HIDDEN` (значення `true` або `false`) — показати або приховати панель.

Приклад використання тегу `<EMBED>`:

```
<EMBED SRC="filename.avi" WIDTH="300" HEIGHT="160"  
AUTOSTART="true" REPEAT="false" ALIGN="left"></EMBED>
```

На рис. 3.20 показано веб-сторінку, яка містить вбудовану в такий спосіб панель програвача.

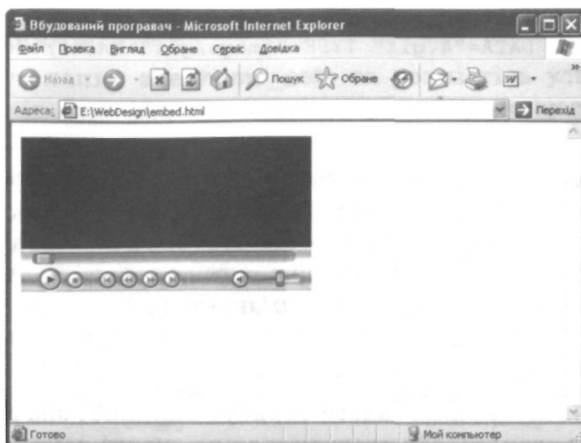


Рис. 3.20. Веб-сторінка із вбудованим програвачем

Інший спосіб розміщення мультимедійного об'єкта на сторінці — це застосування більш універсального тегу `<OBJECT>`. Наприклад:

```
<OBJECT DATA="przyklad.rap3" TYPE="audio/wav"></OBJECT>
```

Атрибут `DATA` задає URL-адресу відтворюваного файлу, атрибут `TYPE` визначає його формат. Для тегу `<OBJECT>` можна використувати ще такі атрибути:

- `ALIGN` — вирівнювання відносно тексту;
- `WIDTH` — ширина;
- `HEIGHT` — висота;
- `HSPACE` — відступ по горизонталі;
- `VSPACE` — відступ по вертикалі.

Як і в попередньому прикладі, об'єкт можна бачити на екрані зазвичай у вигляді вбудованого програвача з елементами керування.

Можна також вкладати кілька елементів `<OBJECT>` один в один. Це приведе до такого результату: якщо у браузері є засіб для перегляду зовнішнього об'єкта, то саме він і буде відображатися, а якщо ні — браузер спробує відобразити внутрішній об'єкт і т. д. Наприклад, можна написати так:

```
<OBJECT DATA="1.mpg" TYPE="video/x-mpeg">  
<OBJECT DATA="2.aiff" TYPE="audio/x-aiff">
```

```
<OBJECT DATA="3.tiff" TYPE="image/tiff">  
  <OBJECT DATA="4.gif" TYPE="image/gif">Кліп</OBJECT>  
</OBJECT>  
</OBJECT>  
</OBJECT>
```

У цьому прикладі браузер спочатку спробує відтворити відеокліп (файл у форматі MPEG). Якщо ця спроба буде вдалою, то все, що міститься всередині зовнішнього тегу <OBJECT>, браузер зігноує, а якщо ні — спробує відтворити файл у форматі AIFF. Якщо відсутній модуль відтворення і для цього файлу, буде спроба відтворити малюнок у форматі TIFF, а в разі невдачі — зображення формату GIF. Зрештою залишиться просто текст: «Кліп».

У тегу <OBJECT> можна задати атрибут STANDBY, значення якого (текстовий рядок) відобразатиметься на екрані доти, доки не завантажиться весь об'єкт. Наприклад, доцільно написати так:

```
<OBJECT DATA="1.wav" TYPE="audio/wav"  
STANDBY="Ifle завантаження. Зачекайте.">
```

Якщо файл 1.wav має великий розмір, відвідувач побачить повідомлення про те, що відбувається завантаження.

Практична робота № 6

Розміщення графічних об'єктів на веб-сторінці

Мета роботи: навчитися оформляти тло веб-сторінки у вигляді малюнка, розмішувати на веб-сторінці графічні об'єкти і налаштовувати їх атрибути, використовувати зображення як посилання.

Створіть веб-сайт, на якому мають міститися графічні зображення як посилання. Головна сторінка сайту повинна складатися з двох частин: у лівій розташовані малюнки-посилання, у правій — місце, де відкриватимуться сторінки відповідно до вибраних гіперпосилань.

Візьміть за зразок сайт «Домашні улюбленці». Для роботи використайте текстове і графічне наповнення веб-сайту, що міститься за адресою <http://protvaryn.narod.ru/>.

Порядок роботи

1. Створіть папку **site_home_pets** та розмістіть у ній файли малюнків про тварин, взяті з веб-сайту protvaryn.narod.ru (рис. 3.21).

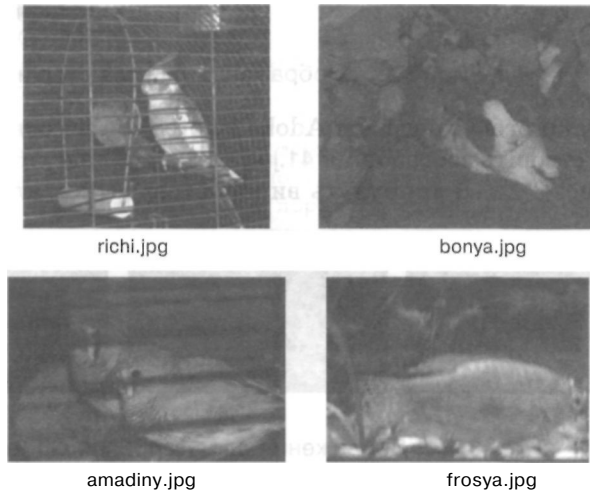


Рис. 3.21. Зображення для веб-сторінок сайту

2. Збережіть у папці **site_home_pets** зображення, показані на рис. 3.22, які будуть використані як гіперпосилання навігаційної панелі.

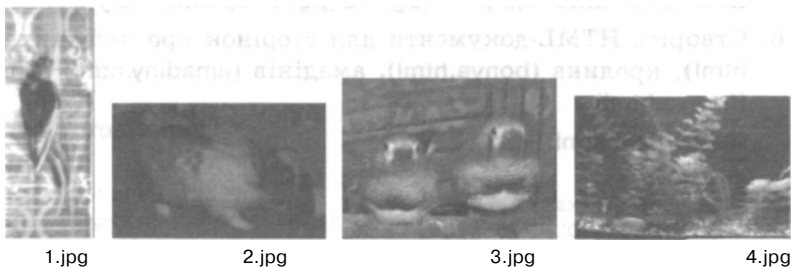


Рис. 3.22. Зображення для навігаційної панелі

3. Відкрийте програму Adobe PhotoShop і створіть п'ять зображень тла — для сторінки навігаційної панелі та кожної сторінки

розділу (рис. 3.23). Збережіть їх у файлах із відповідними іменами.

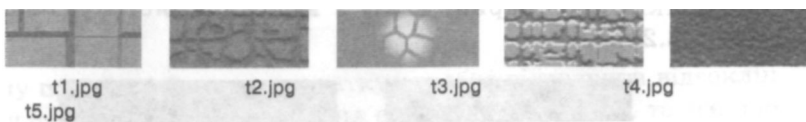


Рис. 3.23. Зображення для тла сторінок

4. За допомогою програми Adobe PhotoShop на основі малюнка 4.jpg створіть зображення 41.jpg та 42.jpg, подібні до наведених на рис. 3.24. Вони будуть використовуватися для анімації.

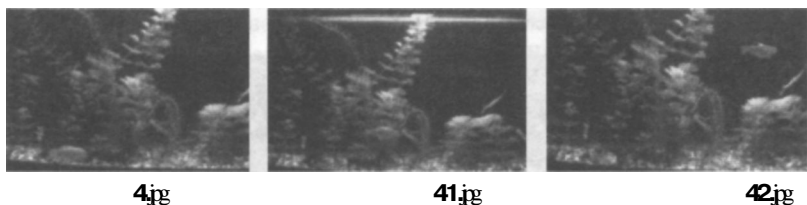


Рис. 3.24. Зображення для створення GIF-анімації

На цих малюнках рибка знаходиться в різних місцях акваріума, завдяки чому можна імітувати її рух, якщо ці зображення змінюватимуть одне одне.

Використовуючи наведені у прикладі 3.4 пояснення, за допомогою програми Adobe PhotoShop створіть GIF-анімацію на основі малюнків 4.jpg, 41.jpg, 42.jpg. Збережіть її у файлі 4.gif. Створіть HTML-документи для сторінок про папужку (richi.html), кролика (bonyu.html), амадінів (amadiny.html) та рибку (frosya.html).

Файл richi.html:

```
<HTML>
<TITLE>Папужка</TITLE>
<BODY BACKGROUND="11.gif">
<IMG SRC="richi.jpg" ALIGN=LEFT HSPACE=10 VSPACE=10>
<FONT SIZE=5><B>
```

Мій папужка – дуже хороша пташка, він так і говорить, коли його добре попросити про це. Його звати Річі, і своє ім'я він також іноді називає. Річінька любить танцювати та співати. Він дуже дружелюбна пташка


```
i має друзів серед інших пташок.</B></FONT>
</BODY>
</HTML>
```

Файл bonya.html:

```
<HTML>
<TITLE>Кролик</TITLE>
<BODY BACKGROUND="t2.gif">
<IMG SRC="bonya.jpg" ALIGN=LEFT HSPACE=10 VSPACE=10>
<FONT SIZE=5><B>
Наш кролик Боня дуже ніжний. У нього гарна біло-полева
шубка. Вуха не дуже довгі, тому що Боня – декоративний
кролик. Він любить гуляти по хаті, заглядати у всі
закутки, наводити там свій порядок. Єдине місце, яке
він завжди оминає, – це кухня. Боня любить лежати
на дивані.
</B></FONT>
</BODY>
</HTML>
```

Файл amadiny.html:

```
<HTML>
<TITLE>Амадіні</TITLE>
<BODY BACKGROUND="t3.gif">
<IMG SRC="amadiny.jpg" ALIGN=LEFT HSPACE=10 VSPACE=10><FONT
SIZE=5><B>
Це двоє братиків – Чіп та Дейл. Вони дружать із Річенькою
та іншими тваринками нашого дому. Чіп – старший братик,
він допомагає Дейлу чиститись навіть тоді, коли сам Дейл
цього робити не хоче. Вони дуже люблять купатись і чомусь
люблять «читати» газети.
</B></FONT>
</BODY>
</HTML>
```

Файл froस्या.html:

```
<HTML>
<TITLE>Рибка</TITLE>
<BODY BACKGROUND="t4.gif">
<IMG SRC="froся.jpg" ALIGN=LEFT HSPACE=10 VSPACE=10><FONT
SIZE=5><B>
У цьому куточку природи/акваріумі дуже часто змінюються
мешканці. Але зараз у ньому живе 4 рибки, найголовніша
серед них – рибка Фрося.
<P>Подивіться, як вона плаває:<BR>
```

```
<IMG SRC="4.gif">
</B></FONT>
</BODY>
</HTML>
```

Для кожної веб-сторінки у тегу `<BODY>` за допомогою атрибута `BACKGROUND` задано своє зображення тла. На сторінці про рибку Фросю розмішена GIF-анімація.

7. Створіть файл `pr.html` для навігаційної панелі, в якій малюнки будуть гіперпосиланнями.

```
<HTML>
<TITLE>Звірята</TITLE>
<BODY BACKGROUND="t5.GIF">
<A HREF="richi.html" TARGET="frame1"><IMG SRC="1.jpg">
</A><BR>
<A HREF="bonya.html" TARGET="frame1"><IMG SRC="2.jpg">
</A><BR>
<A HREF="amadiny.html" TARGET="frame1"><IMG SRC="3.jpg">
</A><BR>
<A HREF="frosya.html" TARGET="frame1"><IMG SRC="4.jpg">
</A>
</BODY>
</HTML>
```

У тегу `<A>` використаний атрибут `TARGET`, значенням якого є ім'я фрейму, в якому відобразатимуться сторінки сайту.

8. Створіть головну сторінку `index.html`, поділену на два фрейми. Після першого завантаження цієї сторінки відвідувач побачить зображення-гіперпосилання у лівому фреймі, а правий буде порожнім доти, доки він не вибере певне посилання.

```
<HTML>
<TITLE>Домашні улюбленці</TITLE>
<FRAMESET COLS="30%,*">
<FRAME SRC="np.html">
<FRAME NAME="frame1">
</FRAMESET>
</HTML>
```

За бажанням для першого завантаження у правий фрейм можна помістити спеціально створену для цього веб-сторінку.

9. Відкрийте у вікні браузера файл `index.html` і перевірте, як діють гіперпосилання. На рис. 3.25 показано головну сторінку

сайту «Домашні улюбленці», коли вибрано посилання на сторінку про кролика.

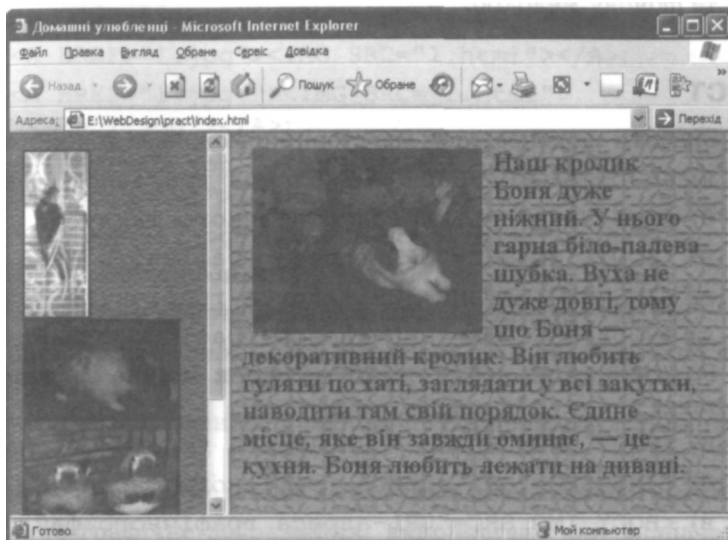


Рис. 3.25. Головна сторінка сайту «Домашні улюбленці»

Самостійна робота

1. Створіть веб-сторінку з таблицею, у комірках якої розмістіть зображення у шаховому порядку. Межі таблиці зробіть невидимими.
2. Створіть такі веб-сторінки:
 - з картою гіперпосилань у вигляді зображення зоряного неба з кількома сузір'ями;
 - з описом кожного сузір'я.

Гарячими областями призначте місця розташування сузір'їв та пов'яжіть ці ділянки зоряного неба з файлами, що містять опис відповідних сузір'їв.

3. Створіть веб-сторінку з описом певної країни, тлом документа зробіть зображення прапора цієї країни, підберіть та настройте фонову мелодію.

Тест

1. Зробити малюнок тлом веб-сторінки можна за допомогою тегу:
- а) `<BODY BACKGROUND="URL-адреса графічного файлу">`;
 - б) ``;
 - в) `<BODY IMG SRC="URL-адреса графічного файлу">`;
 - г) ``;
 - д) `<BODY BGCOLOR="URL-адреса графічного файлу">`.
2. Розмістити на веб-сторінці малюнок можна за допомогою тегу:
- а) `<BODY BACKGROUND="URL-адреса графічного файлу">`;
 - б) ``;
 - в) `<BODY IMG SRC="URL-адреса графічного файлу">`;
 - г) `<BODY BGCOLOR="URL-адреса графічного файлу">`;
 - д) ``.
3. URL-адресу файлу зображення задають за допомогою атрибута:
- а) SRC;
 - б) HREF;
 - в) IMG;
 - г) BACKGROUND;
 - д) BODY.
4. Висоту та ширину графічного об'єкта, розміщеного на веб-сторінці, можна задати:
- а) у дужках після URL-адреси файлу зображення;
 - б) за допомогою атрибутів WIDTH та HEIGHT у тегу IMG;
 - в) за допомогою атрибута ALT;
 - г) за допомогою атрибута SRC;
 - д) за допомогою атрибутів WIDTH та HEIGHT у тегу BODY.

5. Малюнок **1.jpg** можна перетворити на гіперпосилання на веб-сторінку **1.html** за допомогою коду:
- а) ``;
 - б) ``;
 - в) ``;
 - г) ``;
 - д) ``,
6. Графічна карта посилань — це:
- а) зображення карти певної місцевості на веб-сторінці;
 - б) схема розміщення всіх гіперпосилань на сторінках веб-сайту;
 - в) велике зображення, повністю поділене на області-гіперпосилання;
 - г) веб-сторінка, поділена на комірки певної конфігурації;
 - д) одне графічне зображення, використане для визначення кількох гіперпосилань.
7. Атрибут USEMAP у тегу ``:
- а) задає URL-адресу зображення для використання його як карти посилань;
 - б) визначає ім'я карти посилань;
 - в) задає адресу для гіперпосилання, пов'язаного з областю карти;
 - г) визначає тип області карти гіперпосилань;
 - д) має збігатися зі значенням атрибута NAME у тегу `<MAP>`.
8. Гарячі області карти посилань — це:
- а) ділянки визначеної форми у зображенні карти, що є гіперпосиланнями;
 - б) веб-сторінки, пов'язані з картою гіперпосилань;
 - в) пояснювальний текст під картою посилань;
 - г) частини малюнка, пов'язані з іншими веб-сторінками;
 - д) назви інших веб-сторінок, що пов'язані з картою гіперпосилань.

9. Атрибути тегу <AREA> є:
- а) HREF, SHAPE, COORDS;
 - б) RECT, POLYGON, CIRCLE;
 - в) USEMAP, NAME;
 - г) SHAPE, RECT, POLYGON, CIRCLE;
 - д) HREF, USEMAP, NAME.
10. До форматів мультимедійних даних належать:
- а) .avi, .mov, .mpeg, .wav;
 - б) .gif, .jpg, .png;
 - в) .html, .mp3;
 - г) .avi, .mov, .html;
 - д) .ra, .ram, .rm, .rmm, .rmf.
11. Фоновий звук для веб-сторінки можна задати за допомогою коду:
- а) Музика;
 - б) <EMBED SRC="l.html">Мелодія</ EMBED>;
 - в) <BGSOUND SRC="URL-адреса звукового файлу" LOOP=кількість_відтворень>;
 - г) МуЗМКа;
 - д) <BGSOUND HREF="URL-адреса звукового файлу" LOOP=Кількість_відтворень>.
12. Тег :
- а) вставляє малюнок як гіперпосилання;
 - б) виводить картинку, після наведення на яку вказівника миші починається відтворення відеокліпу;
 - в) задає розміри певного зображення;
 - г) дає змогу розмістити панель програвача звукових файлів для відтворення файлу audio.wav;
 - д) визначає відтворення звукового файлу audio.wav зазначену кількість разів після наведення вказівника миші на зображення l.jpg.

Розділ 4

Візуальний редактор веб-сайтів

Тепер, коли ви вмієте наповнювати веб-документи текстовою інформацією та володієте засобами їхнього художнього оформлення, можна подумати про створення першого повноцінного сайту. Та чи приваблює вас перспектива вручну вводити у текстовому редакторі всі теги — від першого до останнього! — і при цьому не бачити відразу результат своєї праці, оскільки його видно лише у браузері? Мабуть, що ні.

Тому вам обов'язково сподобається програма Microsoft FrontPage, яка розглядається у цьому розділі, адже це потужний засіб для візуального створення веб-сторінок та їхньої публікації. Ви ознайомитеся з інтерфейсом програми, засобами форматування тексту, формування гіперпосилань, створення таблиць й розміщення зображень, а також можливостями щодо автоматизованого створення веб-сайтів та окремих сторінок за допомогою майстрів і шаблонів. Набуті знання будуть використані у практичній роботі для створення сайту про рослини.

Інтерфейс та режими роботи програми

Звичайно, було б зручно формувати веб-сторінки так, як у текстових процесорах, коли на екрані документ має приблизно такий самий вигляд, як і після друку. Цей підхід, який називають WYSIWYG (What You See Is What You Get — що бачиш, те й отримаєш), реалізовано у програмі Microsoft FrontPage — візуальному редакторі веб-сторінок. На відміну від більшості текстових редакторів, які виводять на екран лише теги HTML, FrontPage відразу інтерпретує їх, і ви бачите сторінку майже такою, як у вікні

браузера. Завдяки цьому відпадає потреба у постійному переми-
канні між браузером та редактором для перегляду веб-сторінки.
У програмі FrontPage є все необхідне для роботи з веб-сайтом: засіб
для навігації FrontPage Explorer, редактор веб-сторінок FrontPage
Editor, засоби для роботи з графікою та публікації документів.

ВІКНО програми

На рис. 4.1 зображене вікно програми FrontPage 2003. Воно оформлене в єдиному для всіх програм Microsoft Office 2003 сти-
лі. У верхній частині вікна міститься рядок меню і дві панелі
інструментів — Стандартная (Стандартна) та Форматирование (Фор-
матування). Основна частина вікна відведена під робочу область,
в якій можна відкрити один або кілька документів. Внизу розта-
шований рядок стану, що містить довідкову інформацію.

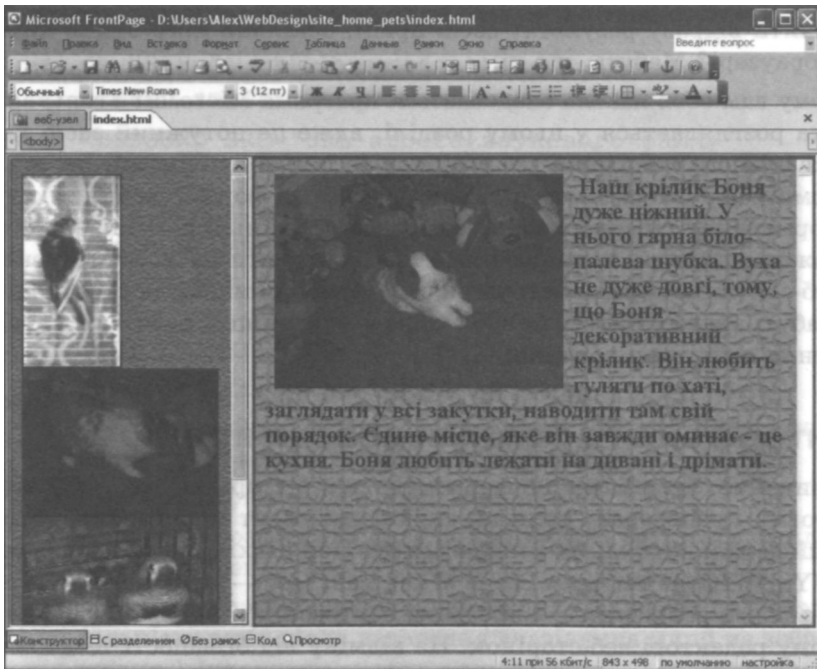









Рис. 4. 1. Вікно програми FrontPage 2003

Програма FrontPage пропонує сім режимів для роботи з веб-документами. Для переходу у потрібний режим використовують команди меню Вид (Вигляд), призначення яких описано у таблиці.

Піктограма команди	Команда	Призначення
	Страница (Сторінка)	Створення, перегляд та редагування веб-сторінок
	Папки (Папки)	Перегляд структури папок створеного веб-сайту
	Удаленный веб-узел (Віддалений веб-вузол)	Робота з віддаленим веб-сайтом, розміщеним на веб-сервері
	Отчеты (Звіти)	Формування та перегляд звітів про веб-сайт
	Переходы (Навігація)	Проектування структури веб-сайту
	Гиперссылки (Гіперпосилання)	Перегляд гіперпосилань веб-сайту
	Задачи(Завдання)	Формування та перегляд завдань, які потрібно виконати під час створення веб-сайту

Поки що для роботи нам буде достатньо лише режиму Страница (Сторінка).

Режими перегляду веб-документа

У режимі Страница (Сторінка) в нижній частині робочої області є чотири вкладки, які дають змогу переглядати веб-сторінку в різних режимах.

- Конструктор (Конструктор) — звичайний режим роботи, в якому створюють або редагують сторінку за допомогою візуальних засобів, розмішуючи на ній текст, лінії, кнопки, посилання та інші об'єкти.

- Код (Код) — режим перегляду і створення сторінки в HTML-кодах. Його можна використовувати як для формування сторінки через уведення HTML-кодів у веб-документ, так і для перегляду, редагування та доповнення кодів сторінки, створеної у звичайному режимі роботи.
- С разделением (3 розподілом) — комбінований режим, у якому робоча область поділена на дві частини: у верхній відображені HTML-коди сторінки, а в нижній — її вигляд у режимі конструктора.
- Просмотр (Перегляд) — попередній перегляд сторінки у тому вигляді, в якому вона відображається у вікні браузера.

Робота з веб-сторінками

Розглянемо, як за допомогою програми FrontPage можна створювати, переглядати та редагувати веб-сайти.

Відкривання та створення сайтів і веб-сторінок

Для відкриття наявного веб-сайту слід виконати такі дії.

1. У меню Файл (Файл) вибрати команду Открыть узел (Відкрити вузол) або на панелі інструментів Стандартная (Стандартна) клацнути стрілку праворуч від кнопки Открыть (Відкрити) і вибрати у меню команду Открыть узел (Відкрити вузол).
2. У вікні Открытие веб-узла (Відкривання веб-вузла) вибрати папку, в якій розташовано веб-сайт. Клацнути кнопку Открыть (Відкрити).

Відкривши сайт, FrontPage не відкриває жодного його файлу, а лише показує вміст папки. Конкретний файл можна відкрити, двічі клацнувши його назву. Який вигляд має вікно програми FrontPage з відкритим сайтом «Домашні улюбленці», що був створений у практичній роботі № 6, показано на рис. 4.2.

Якщо папку зі сторінками сайту в програмі FrontPage ще не відкривали, то з'явиться діалогове вікно із запитом, чи можна додати необхідні технічні каталоги для керування цим сайтом. Після позитивної відповіді така папка стане веб-сайтом у середовищі програми FrontPage. Якщо ж у такий спосіб відкрити порожню папку, то це означатиме створення нового сайту. Значки, якими

позначені веб-сайти, схожі на значки папок, на яких розміщене зображення глобуса.

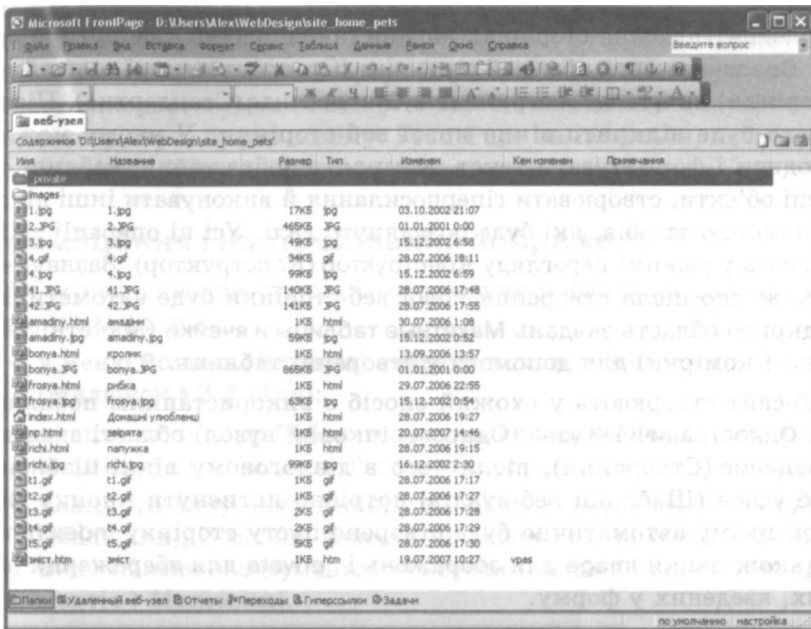


Рис. 4.2. Відкритий веб-сайт «Домашні улюбленці»

За допомогою команди **Файл • Открыть (Файл • Відкрити)** у програмі FrontPage можна відкривати документи, збережені на диску у вигляді окремих файлів, веб-сторінки з Інтернету, а також сторінки локального веб-сайту — в останньому випадку буде відкрито весь відповідний сайт, якщо цього не було зроблено раніше.

Відкрити веб-сторінку можна ще у такі способи:

- на стандартній панелі інструментів клацнути кнопку **Открыть (Відкрити)**;
- клацнути стрілку праворуч від кнопки **Открыть (Відкрити)** і вибрати в меню однойменну команду;
- натиснути клавіші **Ctrl+O**.

У результаті відкривається діалогове вікно **Открытие файла (Відкривання файлу)**, в якому слід вибрати файл веб-сторінки і натиснути кнопку **Открыть (Відкрити)**.

Щоб створити нову звичайну веб-сторінку, необхідно в меню файл (Файл) вибрати команду Создать (Створити) і в області завдань Создание (Створення), що з'явиться, клацнути посилання Пустая страница (Порожня сторінка). Можна також просто клацнути кнопку Создание новой обычной страницы (Створення нової звичайної сторінки) на панелі інструментів Стандартная (Стандартна). Після цього буде відкрито вікно нової веб-сторінки. У ньому можна вводити і форматовувати текст, вставляти зображення, таблиці та інші об'єкти, створювати гіперпосилання й виконувати інші дії за допомогою засобів, які буде розглянуто далі. Усі ці операції здійснюються у режимі перегляду Конструктор (Конструктор). Зазначимо також, що після створення нової веб-сторінки буде автоматично відкрито область завдань Макетные таблицы и ячейки (Макетні таблиці і комірки) для допомоги у створенні таблиць.

Веб-сайт створюють у схожий спосіб з використанням посилання Одностраничный узел (Односторінковий вузол) області завдань Создание (Створення), після чого в діалоговому вікні Шаблоны веб-узлов (Шаблоны веб-вузлів) потрібно натиснути кнопку ОК. При цьому автоматично буде створено пусту сторінку index.html, а також папки image для зображень і _private для збереження даних, введених у форму.

Форматування тексту

Форматування тексту HTML-документа здебільшого виконують за допомогою панелі інструментів Форматирование (Форматування), яка схожа на аналогічну панель програми Word. Але при цьому в текст вставляються відповідні теги HTML. Наприклад, у разі зміни вигляду, розміру, кольору шрифту в документ буде вставлено відповідний тег , виділення курсивом і напівжирним шрифтом забезпечують теги <I> та , а вирівнювання тексту буде задано відповідними атрибутами тегу <P>. Використання кнопок для створення списків приводить до включення у документ відповідних тегів маркованого або нумерованого списків.

Деяких засобів керування шрифтами на панелі інструментів Форматирование (Форматування) немає. Для їх застосування слід вибрати команду Формат • Шрифт (Формат • Шрифт), яка відкриє

діалогове вікно, що дає змогу встановлювати всі параметри символів.

Можна вставляти на сторінку текстові фрагменти, створені в інших програмах Microsoft Office, оскільки FrontPage розпізнає їхні формати даних та конвертує їх у HTML-код. Для перенесення тексту використовують стандартні засоби копіювання (переміщення) тексту через буфер обміну. FrontPage намагається, наскільки це можливо, зберегти форматування такого тексту.

Розміщення і налаштування зображень

Для розміщення на сторінці зображення з файлу слід виконати одну з таких дій:

- у меню Вставка (Вставка) вибрати команду Рисунок • Из файла (Малюнок • З файлу);
- клацнути кнопку Додати рисунок из файла (Додати малюнок із файлу) на стандартній панелі інструментів;
- клацнути таку саму кнопку на панелі інструментів Рисунки (Малюнки); її можна вивести на екран за допомогою команди Вид • Панели инструментов • Рисунки (Вигляд • Панелі інструментів • Малюнки).

У діалоговому вікні, яке відкриється, необхідно знайти потрібний графічний файл і клацнути кнопку ОК. Малюнок буде вставлено в позицію курсору.

Розмістивши зображення на веб-сторінці, можна настроїти його властивості. Для цього використовують діалогове вікно Свойства рисунка (Властивості малюнка). Щоб відкрити його, слід виділити малюнок за допомогою миші та виконати одну з таких дій:

- у меню Формат (Формат) вибрати команду Свойства (Властивості);
- відкрити контекстне меню і вибрати команду Свойства рисунка (Властивості малюнка);
- натиснути клавіші Alt+Enter.

Діалогове вікно Свойства рисунка (Властивості малюнка) має три вкладки: Вид (Вигляд), Общие (Загальні) та Видеозапись (Відеозапис).

Розташування і масштабування зображень

Вкладка Вид (Вигляд) (рис. 4.3) містить елементи керування, за допомогою яких задають розташування малюнка на сторінці. Розглянемо їхнє призначення.

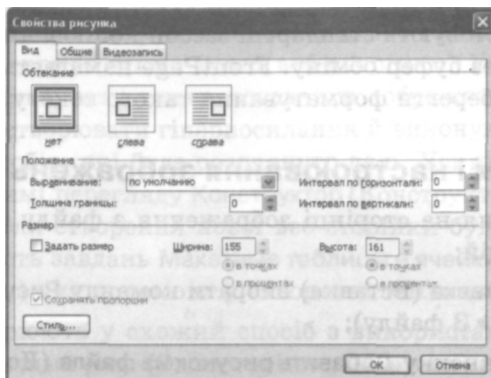


Рис. 4.3. Вкладка Вид (Вигляд) діалогового вікна властивостей малюнка

Список Выравнивание (Вирівнювання) в області Положение (Розташування) містить значення, які визначають розташування зображення відносно тексту, зокрема:

- по умолчанию (за умовчанням) — буде застосовано стандартні параметри вирівнювання;
- по левому краю (за лівим краєм) — зображення розташоване в лівій частині сторінки, текст обтікає його праворуч;
- по правому краю (за правим краєм) — зображення розташоване у правій частині сторінки, текст обтікає його ліворуч;
- по верхнему краю (за верхнім краєм) — обтікання немає, зображення розташоване в тексті, відповідний рядок якого вирівняно за верхньою межею малюнка;
- по середине (по середині) — обтікання немає, зображення розташоване в тексті, відповідний рядок якого вирівняно по середній лінії малюнка;
- по нижнему краю (за нижнім краєм) — обтікання немає, зображення розташоване в тексті, відповідний рядок якого вирівняно за нижньою межею малюнка.

Для вирівнювання зображення з обтіканням його текстом можна також використовувати елементи області Обтекание (Обтікання) і кнопки панелі форматування По левому краю (За лівим краєм) та По правому краю (За правим краєм).

Окрім цього, в області Положение (Розташування) можна задати:

- Толщина границы (Товщина межі) — товщину рамки навколо зображення у пікселях;
- Интервал по горизонтали (Интервал по горизонталі) — відстань (у пікселях) по горизонталі від зображення до тексту, що його оточує;
- Интервал по вертикали (Интервал по вертикалі) — відстань (у пікселях) по вертикалі від зображення до тексту, що його оточує.

Поля Ширина (Ширина) та Высота (Висота) в області Размер (Розмір) призначені для того, щоб задавати розміри зображення у пікселях або у відсотках від розміру вікна браузера. Не забудьте попередньо встановити прапорець Задать размер (Задати розмір). Використовуйте прапорець Сохранять пропорции (Зберігати пропорції) для того, щоб зберегти пропорції зображення. Змінювати ширину і висоту малюнка можна також за допомогою миші аналогічно до того, як це роблять у текстовому процесорі Word.

Загальні параметри зображень

У верхній частині вкладки Общие (Загальні) (рис. 4.4) розташоване поле Рисунок (Малюнок), в якому зазначена URL-адреса малюнка. Тут також розташовані кнопка Обзор (Огляд), за допомогою якої можна вибрати інше зображення для розміщення на веб-сторінці, та кнопка Изменить (Змінити), скориставшись якою можна відредагувати поточний графічний файл.

Кнопка Тип графических файлов (Тип графічних файлів) відкриває діалогове вікно для вибору формату графічного зображення — GIF, JPEG або PNG. Якщо малюнок має інший формат, то програма перетворює його на один із цих трьох.

Область Альтернативные представления (Альтернативні подання) містить поля, за допомогою яких задають параметри, які буде використано у разі перегляду веб-сторінки в режимі відключення

графіки або на моніторі з низькою роздільною здатністю. У рядку Низкое разрешение (Низька роздільна здатність) можна вибрати адресу файлу, який буде завантажуватися замість поточного в режимі з низькою роздільною здатністю. У рядку Текст (Текст) задають короткий опис, що з'являтиметься після наведення на малюнок вказівника миші.

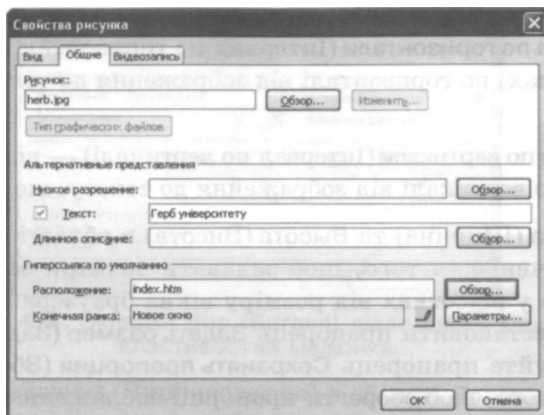




















Рис. 4.4. Вкладка Общие (Загальні) діалогового вікна властивостей малюнка

Зображення можна використовувати як гіперпосилання — для цього призначене поле Расположение (Розташування) в області Гиперссылка по умолчанию (Гіперпосилання за умовчанням). За допомогою кнопки Обзор (Огляд) вибирають веб-сторінку, пов'язану з таким посиланням. Поле Конечная рамка (Цільовий фрейм) визначає фрейм для відображення файлу, пов'язаного з гіперпосиланням. Вкладка Видеозапись (Відеозапис) дає змогу вставити у веб-сторінку аудіо- чи відеозапис; зображення при цьому використовуватиметься як заставка під час завантаження мультимедійного файлу.

Редагування зображень

У програмі FrontPage можна виконувати кілька найтипівіших операцій з редагування зображень, не застосовуючи спеціалізовані

графічні редактори. Для цього слід скористатися панеллю інструментів Рисунки (Малюнки), яка містить, зокрема, такі кнопки:

Кнопка	Операція
	Відкриття діалогового вікна для вибору малюнка та розміщення його на сторінці
	Введення тексту на малюнку
 	Поворот малюнка за або проти годинникової стрілки
 	Дзеркальне відображення малюнка по горизонталі та вертикалі
 	Збільшення та зменшення контрастності малюнка
 	Збільшення та зменшення яскравості малюнка
	Обтинання малюнка
	Задання прозорості одного з кольорів GIF-зображення
	Інструмент для виділення графічних об'єктів
  	Інструменти для створення на зображенні гарячих областей заданої форми та призначення гіперпосилань для карти посилань
	Відображення всіх гарячих областей на карті посилань
	Відновлення початкових параметрів зображення

Створення таблиць

Для створення таблиці слід встановити курсор у місце, де вона має міститися, і в меню Таблица (Таблица) вибрати команду Вставить • Таблица (Вставить • Таблица). Відкриється діалогове вікно

Вставка таблицы (Вставлення таблиці) (рис. 4.5), в якому можна визначити всі параметри таблиці.

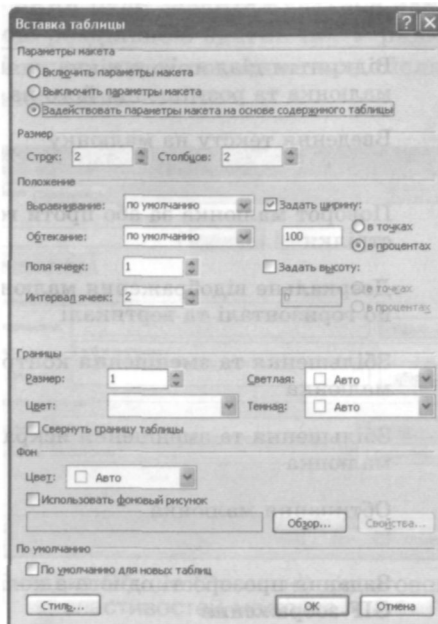


Рис. 4.5. Діалогове вікно для вставлення таблиці

В області **Размер (Розмір)** цього вікна потрібно задати кількість рядків **Строк (Рядків)** та стовпців **Столбцов (Стовпців)**. В області **Положение (Розташування)** слід визначити розташування таблиці на сторінці, вибравши у списку **Выравнивание (Вирівнювання)** одне зі значень:

- по умолчанию (за умовчанням) — до HTML-коду не буде додано тегів, що керують вирівнюванням таблиці;
- по левому краю (за лівим краєм) — вирівнювання за лівим краєм сторінки;
- по правому краю (за правим краєм) — вирівнювання за правим краєм сторінки;
- по центру (по центру) — вирівнювання по центру сторінки.

За допомогою прапорця **Задать ширину** (**Задати ширину**) та поля під ним можна зафіксувати ширину таблиці в пікселях або відсотках від ширини вікна. Введенням потрібного значення у поле **Размер** (**Розмір**) області **Границы** (**Межі**) задають товщину меж таблиці.

Після натискання кнопки **ОК** таблицю буде вставлено на сторінку в позицію курсору. Варто також зазначити, що всі параметри таблиці можна змінити у діалоговому вікні **Свойства таблицы** (**Властивості таблиці**), яке відкривають за допомогою команди **Таблица • Свойства таблицы • Таблица** (**Таблиця • Властивості таблиці • Таблиця**) або однойменної команди контекстного меню таблиці.

Швидко створити таблицю можна за допомогою кнопки **Добавить таблицу** (**Додати таблицю**), яка розміщена на панелі інструментів **Стандартная** (**Стандартна**). Для цього слід натиснути цю кнопку і, не відпускаючи лівої кнопки миші, вибрати необхідну кількість рядків та стовпців таблиці.

Створення гіперпосилань

Щоб створити гіперпосилання, необхідно виділити фрагмент тексту або зображення і виконати одну з таких дій:

- вибрати команду **Гиперссылка** (**Гіперпосилання**) у меню **Вставка** (**Вставка**);
- відкрити контекстне меню і вибрати у ньому команду **Гиперссылка** (**Гіперпосилання**);
- натиснути клавіші **Ctrl+K**;
- клацнути кнопку **Добавление гиперссылки** (**Додання гіперпосилання**) на панелі інструментів **Стандартная** (**Стандартна**).

Відкриється діалогове вікно (рис. 4.6), в якому слід виділити об'єкт, на який вказуватиме посилання, і клацнути кнопку **ОК**.

Гіперпосилання може вказувати, зокрема, на такі об'єкти:

- веб-сторінку поточного сайту — якщо в цей момент у програмі відкритий веб-сайт, то в діалоговому вікні можна вибрати одну з його сторінок;
- веб-сторінку, що міститься за будь-якою URL-адресою — її можна безпосередньо ввести в поле **Адрес** (**Адреса**) або клацнути

кнопку Интернет (Интернет) із зображенням глобуса і знайти сторінку в Інтернаті за допомогою браузера;

- файл, що міститься на диску комп'ютера — необхідно клацнути кнопку Поиск файла (Пошук файлу) із зображенням папки і в діалоговому вікні вибрати потрібний файл;
- адресу електронної пошти — таке посилання дає змогу відвідувачам сайту відіслати електронний лист; для його створення слід клацнути значок електронної поштою (електронною поштою) у лівій частині діалогового вікна і в полі Адрес ел. почты (Адреса електронної пошти) ввести адресу отримувача.
- якорі — якщо на веб-сторінці є якорі (їх можна створити за допомогою команди Закладка (Закладка) меню Вставка (Вставка), попередньо виділивши фрагмент тексту), то гіперпосилання на них формують вибором імені якоря зі списку, що відкривається за допомогою кнопки Закладка (Закладка) або значка местом в документе (місцем у документі) у лівій частині діалогового вікна.

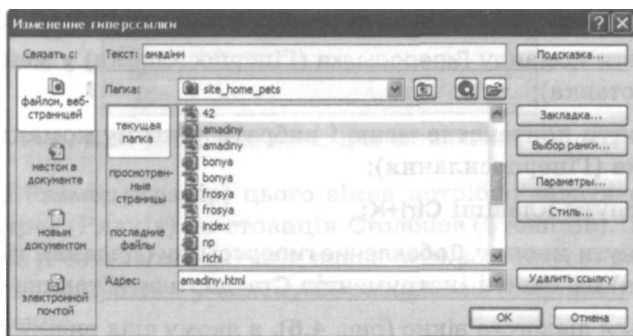


Рис. 4.6. Діалогове вікно для створення гіперпосилань

Якщо посилання потрібно відкрити в іншому фреймі, використовують кнопку Выбор рамки (Вибір фрейму), яка відкриває спеціальне діалогове вікно.

Для видалення наявного гіперпосилання слід виділити його, відкрити описане раніше діалогове вікно і клацнути кнопку Удалить ссылку (Видалити посилання).

Вправа 4.1

Створимо веб-сторінку, що міститиме відформатований текст.

1. Створіть папку, наприклад `my_web`, в якій ви зберігатимете розроблені за допомогою FrontPage веб-сторінки. Відкрийте її у програмі як веб-сайт.
2. Створіть нову веб-сторінку, клацнувши кнопку Создание новой обычной страницы (Створення нової звичайної сторінки) на панелі інструментів Стандартная (Стандартна).
3. Введіть наведений нижче текст у вікно веб-сторінки в режимі Конструктор (Конструктор) (рис. 4.7),

Троянда столиста

Граціозні форми, ніжне забарвлення, витончений аромат троянди з незапам'ятних часів чарували людей. Тому й створено багато легенд про цю чарівну, ніжну й чудову квітку. За староіндійською легендою, прекрасна богиня Лакшмі народилася з пуп'янка троянди.

Зауважте, що після переходу на новий абзац він буде відділений від попереднього відступом.

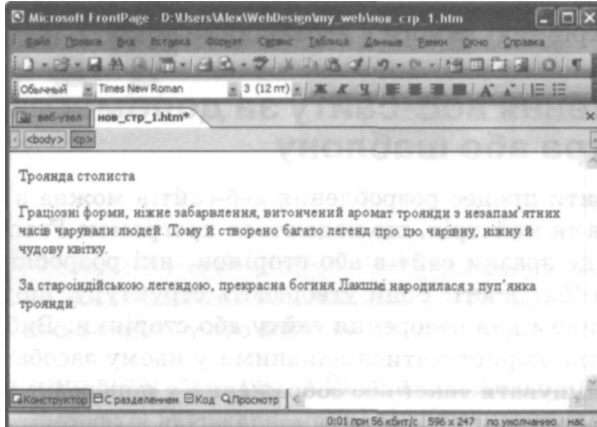


Рис. 4.7. Введений текст поділено на абзаци

4. Користуючись панеллю інструментів Форматирование (Форматування), внесіть зміни, показані на рис. 4.8.

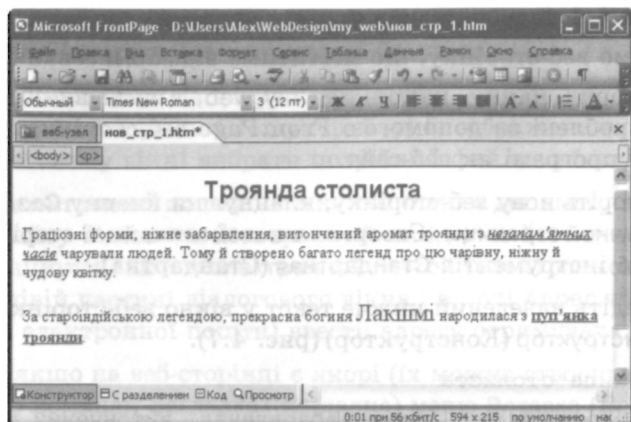


Рис. 4.8. Веб-сторінка з елементами форматування тексту

5. Вибравши команду Файл • Сохранить (Файл • Зберегти), збережіть сторінку, а потім закрийте її вікно. Пам'ятайте, що файлам сайту потрібно надавати короткі назви, записані малими латинськими літерами і цифрами, використовуючи символи «_» замість пробілів. Так ви уникнете проблем під час розміщення сайту у Всесвітній мережі, оскільки для деяких веб-серверів такі вимоги є необхідними.

Створення веб-сайту за допомогою майстра або шаблону

Прискорити процес розроблення веб-сайтів можна за допомогою шаблонів та майстрів, що надаються програмою FrontPage. *Шаблони* — це зразки сайтів або сторінок, які розробник заповнює в міру необхідності. Вони утворюють структуру, що є стартовим майданчиком для створення сайту або сторінки. Вибравши шаблон, можна користуватися наявними у ньому засобами. Наприклад, розміщувати текст або зображення у попередньо визначених областях сторінки або вводити заздалегідь відформатований текст і заголовки.

Майстри у процесі своєї роботи пропонують заповнити низку форм і на основі ваших відповідей створюють веб-сторінки, розміщуючи на них заголовки та частину інформації.

І майстри, і шаблони орієнтовані на розробку невеликих сайтів, вони містять лише кілька елементів, що можуть налаштовуватися. Потім розробник повинен редагувати наявні сторінки, розміщуючи на них зображення, текст і посилання, та додавати до сайту готові веб-сторінки.

Використання майстрів та шаблонів для сайтів

Для запуску шаблону або майстра зі створення веб-сайту необхідно виконати низку дій.

1. У меню Файл (Файл) вибрати команду Создать (Створити) і в області завдань Создание (Створення) клацнути посилання Другие шаблоны веб-узлов (Інші шаблони веб-вузлів). Відкриється діалогове вікно для вибору шаблону або майстра (рис. 4.9). (Майстри позначені піктограмами з чарівними паличками).

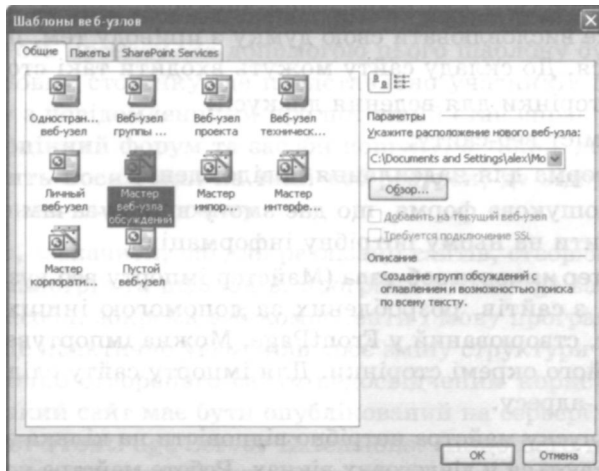


Рис. 4.9. Вибір шаблону або майстра для створення веб-сайту

2. У діалоговому вікні (див. рис. 4.9) виділити значок потрібного майстра. В області Описание (Опис) відобразиться його короткий опис.
3. У полі Укажите расположение нового веб-узла (Зазначте розташування нового веб-вузла) вибрати місцезнаходження сайту.
4. Клацнути кнопку ОК.

Розглянемо призначення деяких майстрів, що надаються програмою FrontPage 2003.

Мастер корпоративного веб-узла (Майстер корпоративного веб-вузла) створює сайт компанії. До його складу можуть входити такі сторінки:

- сторінка для перегляду новин про компанію, інформація про товари або послуги, що надає компанія;
- таблиця зі змістом, яка полегшує відвідувачам навігацію сайтом;
- форма відгуків для клієнтів;
- пошукова форма, що дає змогу відвідувачам сайту знаходити на ньому потрібну інформацію.

Мастер веб-узла обговорений (Майстер веб-вузла обговорень) створює сайт для проведення дискусій, форум, де відвідувач може висловлювати свою думку з приводу тем, що обговорюються. До складу сайту можуть входити такі сторінки:

- сторінки для ведення дискусії;
- зміст веб-сайту;
- форма для надсилання повідомлень;
- пошукова форма, що дає змогу відвідувачам сайту знаходити на ньому потрібну інформацію.

Мастер імпорту веб-узла (Майстер імпорту веб-вузла) імпортує дані з сайтів, розроблених за допомогою інших програм, у сайт, створюваний у FrontPage. Можна імпортувати весь сайт або його окремі сторінки. Для імпорту сайту слід задати його URL-адресу.

Шлях запуску майстра потрібно відповісти на кілька запитань, що відображаються у діалогових вікнах. Робота майстра поділяється на кроки, для переміщення між якими використовують такі кнопки:

- Назад (Назад) — повертає до попереднього кроку для зміни інформації, введеної раніше;
- Далее (Далі) — дає змогу перейти до наступного кроку роботи майстра;
- Отмена (Скасування) — припиняє роботу майстра;
- Готово (Готово) — ініціює останній етап створення сайту на підставі введеної інформації.

По завершенні своєї роботи майстер створює сайт та відкриває його у вікні програми. Якщо кнопкою Готово (Готово) скористатися на проміжному етапі, то FrontPage створить сайт із параметрами за умовчанням для тих складових, які було пропущено.

Крім майстрів як основу для веб-сайту можна використовувати шаблони. Деякі з них описані нижче.

- Веб-узел технической поддержки (Веб-вузол технічної підтримки) — найкраще підходить для фірм, що виробляють програмне забезпечення, однак може бути застосований і для інших напрямів бізнесу.
- Пустой веб-узел (Порожній веб-вузол) — використовують для створення порожнього сайту без сторінок. Розробник потім може доробити його, використовуючи редактор FrontPage.
- Веб-узел проекта (Веб-вузол проекту) — використовують для сайту, який може бути центральним інформаційним форумом для якогось проекту. За допомогою цього шаблону буде створено головну сторінку, де представлено учасників проекту, сторінку з повідомленнями про поточний стан справ у проекті, повноцінний форум та засоби пошуку, а також документ, що містить посилання на інші веб-ресурси, де відбувається обговорення цього проекту.

Варто, однак, зазначити, що для реалізації сайтів, створюваних за допомогою майстрів та шаблонів, програма FrontPage застосовує спеціальні засоби, зокрема веб-компоненти і мову програмування JavaScript. Це практично унеможливує зміну структури та функцій автоматично створеного сайту недосвідченим користувачем. Крім того, такий сайт має бути опублікований на сервері, де встановлено пакет FrontPage Server Extensions. Якщо його немає, то деякі елементи сайту не будуть працювати.

Шаблони веб-сторінок

Крім шаблонів сайтів програма FrontPage пропонує велику кількість шаблонів веб-сторінок — від шаблону звичайної сторінки до шаблону, що містить складні таблиці. Маючи готовий сайт, за допомогою цих шаблонів можна швидко додавати до нього нові сторінки.

Для створення веб-сторінки з шаблону необхідно виконати зазначені нижче дії.

1. У меню **Файл** (**Файл**) вибрати команду **Создать** (**Створити**) і в області завдань клацнути посилання **Другие шаблоны страниц** (**Інші шаблони сторінок**). Ще один спосіб — на панелі інструментів **Стандартная** (**Стандартна**) клацнути стрілку праворуч від кнопки **Создание новой обычной страницы** (**Створення нової звичайної сторінки**) і вибрати у списку команду **Страница** (**Сторінка**).
2. У діалоговому вікні **Шаблоны страниц** (**Шаблони сторінок**), що відкривається, вибрати зі списку шаблон (рис. 4.10).

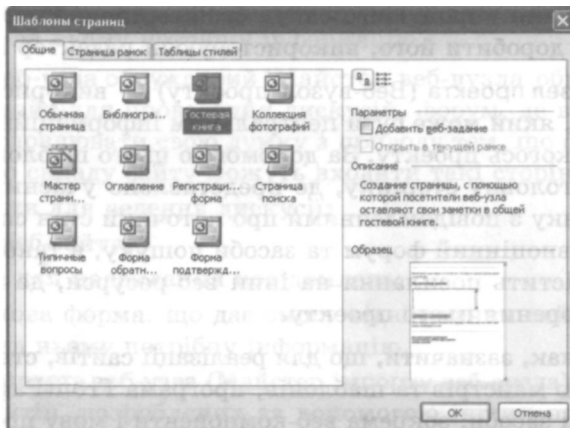


Рис. 4.10. Список шаблонів сторінок

Праворуч у вікні розташовано опис вибраного шаблону та його схематичний вигляд. Вигляд списку шаблонів можна змінювати за допомогою кнопок, що містяться праворуч вгорі. Якщо відрита сторінка з фреймами (див. далі) і потрібно, щоб новий документ відкрився у поточному фреймі, то слід встановити прапорець **Открыть в текущей рамке** (**Відкрити у поточному фреймі**).

На сторінках, створених за допомогою шаблону, розташовані заготовки тексту і заголовків, які слід замінити реальними даними.

У програмі **FrontPage** на сторінці можна розмістити один або кілька фреймів; тоді її називають *сторінкою з фреймами*, або *фреймсе-*

том. Такий приклад ми розглядали під час вивчення мови HTML: екран поділений на частини, ліворуч — список навчальних закладів, оформлений як список гіперпосилань, праворуч — область, в яку виводиться інформація про об'єкт, вибраний відвідувачем сайту. Хоча фрейми є доволі популярними, слід зазначити, що наявність їх на сторінці суттєво сповільнює процес її завантаження; окрім цього, велика кількість фреймів дезорієнтує відвідувача сайту. На вкладці Страница рамок (Сторінка фреймів) діалогового вікна Шаблони страниц (Шаблони сторінок) пропонуються шаблони, кожен з яких містить не більше ніж чотири фрейми у фреймсеті.

Вправа 4.2

Розглянемо приклад використання шаблону сторінки з фреймами.

1. У режимі Конструктор (Конструктор) клацніть стрілку праворуч від кнопки Создание новой обычной страницы (Створення нової звичайної сторінки) і виберіть у списку команду Страница (Сторінка).
2. У вікні, що після цього відкриється, на вкладці Страница рамок (Сторінка фреймів) виберіть шаблон Оглавление (Зміст).

Після клацання кнопки ОК сторінка з фреймами буде відкрита у вікні програми FrontPage (рис. 4.11). Подальші дії полягають у тому, щоб відповідним чином заповнити отриману структуру за допомогою засобів програми FrontPage: відобразити у фреймах уже наявні веб-сторінки, скориставшись кнопкою Задать начальную страницу (Задати початкову сторінку), або створити нові — кнопкою Создать страницу (Створити сторінку).

Яким би великим не був список шаблонів, неможливо задовольнити потреби кожного розробника, тому FrontPage надає можливість створювати та зберігати власні шаблони сторінок. Це може знадобитися тоді, коли необхідно створити кілька однакових за структурою документів, а серед запропонованих шаблонів підходящих немає. У такому випадку розробляють власний шаблон — це заощаджує час розробника й зменшує обсяг рутинної роботи. Окрім цього, можна відкрити готовий шаблон, модифікувати його і зберегти як новий.

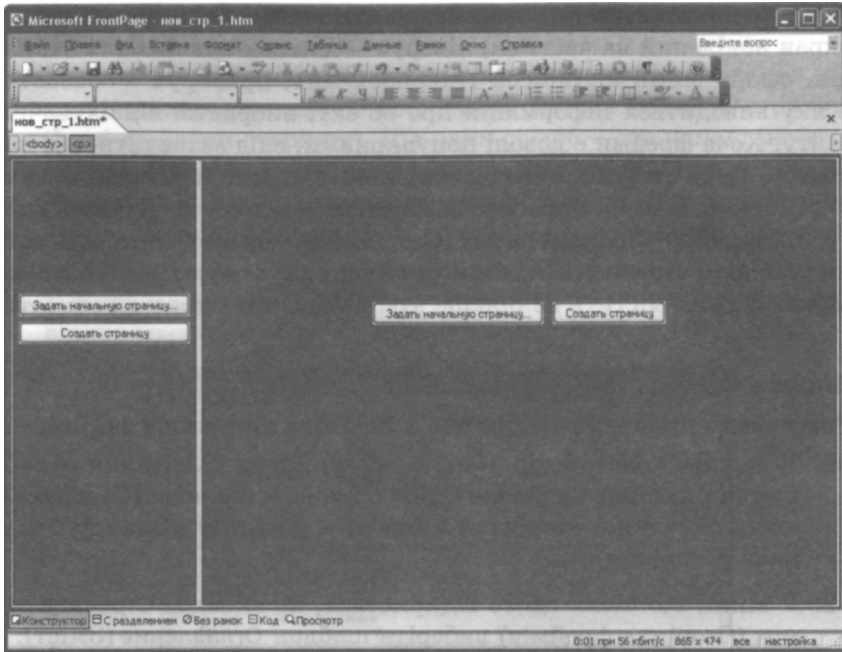


Рис. 4.11. Сторінка з двома фреймами

Вправа 4.3

Створимо новий шаблон.

1. У режимі Конструктор (Конструктор) клацніть кнопку Створення нової звичайної сторінки (Створення нової звичайної сторінки). Буде створено пусту сторінку.
2. Створіть на новій сторінці об'єкти, які необхідно зберегти у шаблоні. Це можуть бути заголовки, логотип, таблиці, списки, зображення тощо.
3. Виберіть команду Файл • Сохранить (Файл • Зберегти). У списку Тип файла (Тип файлу) діалогового вікна, що відкриється, виберіть елемент Шаблоны FrontPage (Шаблони FrontPage) і клацніть кнопку Сохранить (Зберегти).
4. У наступному діалоговому вікні Сохранить как шаблон (Зберегти як шаблон) введіть заголовок, ім'я файлу шаблону та його короткий опис у текстовому полі Описание (Опис). Можна замі-

нити своїм шаблоном якийсь із наявних: для цього потрібно використати кнопку Обзор (Огляд), знайти і вибрати шаблон, який слід замінити.

Після зберігання шаблону він відобразатиметься у списку шаблонів і майстрів, які постачаються з програмою FrontPage.

Практична робота № 7

Створення сайту в середовищі FrontPage

Мета роботи: навчитися використовувати можливості візуального редактора FrontPage для форматування тексту веб-сторінок, створення гіперпосилань, вставлення графіки і таблиць.

Створимо невеликий сайт «Рослини України» із трьома рівнями ієрархії сторінок (рис. 4.12). Головна сторінка має складатися з текстової та графічної інформації, розташованої у комірках таблиці. В одну з комірок помістить зображення, що буде використане як карта посилань. Сторінки розділів мають містити фотографії та текстові описи.



Рис. 4.12. Структура сайту «Рослини України»

Порядок роботи

1. Створіть папку roslyny та розташуйте у ній такі графічні файли: karta.jpg (карта посилань, на якій зображені дерева і квіти), kalyna.jpg (калина), topolya.jpg (тополя), barvinok.jpg (барвінок) та lyon.jpg (льон).
2. Відкрийте вікно програми FrontPage. Виберіть у меню Файл (Файл) команду Открыть узел (Відкрити веб-сайт) та виберіть папку roslyny. Програма повідомить про те, що до цієї папки

будуть записані ще деякі файли, потрібні для роботи. Клацніть кнопку ОК.

3. Створіть сторінки найнижчого ієрархічного рівня — про калину, тополю, барвінок і льон — та збережіть їх у поточній папці у файлах *kalyna.htm*, *topolya.htm*, *barvinok.htm* і *lyon.htm*. Для цього скористайтеся командою Файл • Создать (Файл • Створити) або кнопкою Создание новой обычной страницы (Створення нової звичайної сторінки) стандартної панелі інструментів.
4. Введіть на сторінки опис кожної з рослин.
 - Про калину: «Кущ калини біля материної хати — це не лише окраса, а й символ України, наш духовний світ, наша спадщина. Якщо троянди й виноград, за влучним висловом Максима Рильського, символізують красиве й корисне, то кущ калини, увібравши обидві ознаки, опредметнює й духовний потяг до своєї землі, свого оберіга, своїх традицій».
 - Про тополю: «В українському фольклорі часто зустрічаються згадки про всім відому рослину. Це тополя (*Rorus*) — міфічний образ-тотем давніх українців. Свято тополі проводили наприкінці весни. Обирали найстрункішу дівчину, якій зав'язували над головою руки, на які навішували стрічки, хустки, намисто, водили селом, лугом, полем та співали: «Стояла Тополя край чистого поля, стій, Тополенько, стій не розвивайся, буйному вітроньку не піддавайся...». Особливо свято тополі було поширене в околицях Києва».
 - Про барвінок: «Ще однією рослиною, що зустрічається у багатьох легендах та казках українського народу, є барвінок (*Vinca*). Це трав'яниста багаторічна рослина, яка має видовжене листя, квітки синьо-фіолетового кольору, плід, що складається з двох вільних листків. Барвінок зацвітає в травні-липні. Поширений барвінок в Україні, Білорусі та на Кавказі».
 - Про льон: «У нашій країні льон висівають переважно в зоні Полісся, саме тому його ще називають поліським шовком. Поліщуки добре знають, коли, в яку пору краще сіяти, збирати, слати, тіпати льон, з приводу чого склалося багато народних прикмет. Так, вважали, що сіяти льон

найкраще у п'ятницю чи суботу, несучи насіння в поле у білій скатертині».

Для оформлення сторінок різними темами виконайте для кожної сторінки команду **Формат • Тема (Формат • Тема)**, виберіть в області завдань **Тема (Тема)** одну з тем і призначте її для оформлення сторінки, клацнувши значок теми.

Збережіть сторінки з відповідними назвами за допомогою команди **Файл • Сохранить (Файл • Зберегти)** або однойменної кнопки стандартної панелі інструментів. Оскільки для їх оформлення використані вбудовані теми, під час записування з'явиться вікно з повідомленням, що сторінка буде збережена разом із додатковими файлами. На рис. 4.13 зображено веб-сторінку третього рівня ієрархії, для якої застосовано стандартну тему оформлення.

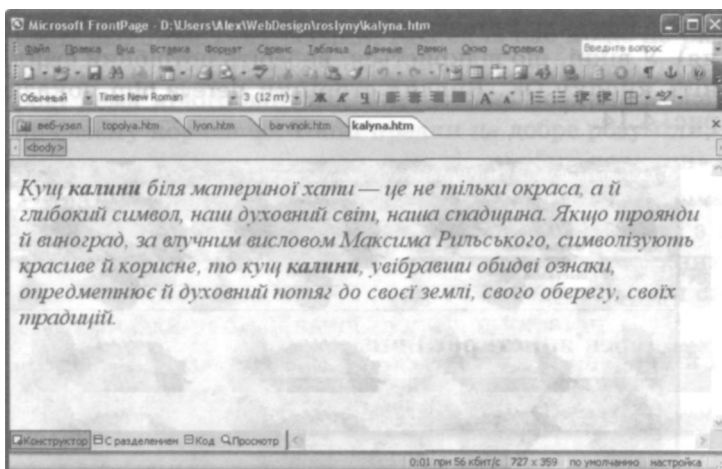


Рис. 4.13. Сторінки третього рівня ієрархії

Створіть сторінки другого рівня ієрархії про дерева та квіти. Використайте зображення як посилання на сторінки третього рівня ієрархії. Збережіть сторінки у файлах з іменами **dereva.htm** **Takvity.htm**.

Для цього спочатку на новій сторінці про дерева сформууйте заголовок, а потім виділіть його текст, введений в окремому абзаці, і застосуйте до нього стиль **Заголовок 3 (Заголовок**

третього рівня), вибравши відповідне значення зі списку Стиль (Стиль) на панелі інструментів Форматирование (Форматування). Потім за допомогою команди Рисунок • Из файла (Малюнок • Из файлу) меню Вставка (Вставка) вставте на сторінку малюнки калини та тополі.

Розмістивши зображення на веб-сторінці, настройте їх розмір у діалоговому вікні Свойства рисунка (Властивості малюнка). Щоб його відкрити, виділіть малюнок за допомогою миші та виберіть у меню Формат (Формат) команду Свойства рисунка (Властивості малюнка). Задайте прийнятний розмір зображення у полях Ширина (Ширина) та Высота (Висота). Для збереження його пропорцій має бути встановлений прапорець Сохранять пропорции (Зберігати пропорції).

Зробіть кожен малюнок посиланням на сторінку з описом відповідної рослини. Для цього для кожного малюнка виконайте команду Вставка • Гиперссылка (Вставка • Гіперпосилання). У вікні, що відкриється, виберіть потрібний файл і клацніть кнопку ОК. Вигляд сторінки dereva.htm показаний на рис. 4.14.

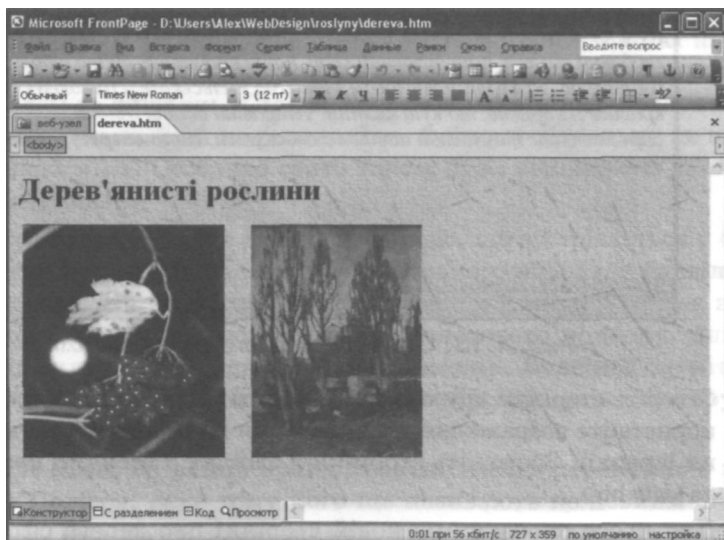


Рис. 4.14. Сторінка другого рівня ієрархії dereva.htm з малюнками-гіперпосиланнями

10. Так само створіть сторінку про квіти.
11. Створіть головну сторінку. Для структурування інформації використайте таблицю без відображення меж. Виберіть команду Таблица • Вставить • Таблица (Таблица • Вставити • Таблица) і задайте для таблиці кількість рядків — 2 та кількість стовпців — 3. У поле Размер (Розмір) області Границы (Межі) введіть 0 (межі відсутні). Решту значень не змінюйте. Після клацання кнопки ОК таблицю буде вставлено на веб-сторінку. Об'єднайте комірки першого рядка, виділивши їх та виконавши команду Таблица • Объединить ячейки (Таблица • Об'єднати комірки).
12. Введіть в об'єднану комірку текст «Рослини України» і призначте для нього стиль Заголовок 1 (Заголовок першого рівня). У комірки другого рядка введіть подані далі тексти.
 - У ліву комірку: «Людство живе в оточенні величезної кількості різноманітних рослин. Кожен із нас може по-різному характеризувати будь-яку рослину, але слід знати, що у природі нема нічого зайвого, тому кожна рослина може бути по-своєму корисною. Наші предки це добре розуміли, тому використовували у лікувальних цілях різні рослини».
 - У праву комірку: «В українському фольклорі зустрічаються згадки про величезну кількість рослин, багато з яких мають лікувальні властивості. Проте лише деякі з них здобули особливу любов українського народу та стали символічними для національної культури українців. Лікарських рослин, які використовують у народній і професійній медицині, дуже багато. Вони бувають:
 - трав'янисті
 - дерев'янисті».
13. У центральну комірку другого рядка вставте малюнок, який буде картою посилань (рис. 4.15). Зауважте, що межі таблиці, які мають нульову ширину, у режимі Конструктор (Конструктор) зображені пунктирними лініями. Після переходу в режим Просмотр (Перегляд) їх не буде видно.
14. Для призначення гарячих областей зображення скористайтеся кнопкою панелі інструментів Рисунки (Малюнки), що означає область гіперпосилання прямокутної форми.

Якщо панель не відкрита, виберіть команду Вид • Панелі інструментов • Рисунки (Вигляд • Панелі інструментів • Малюнки). Виділіть верхню частину малюнка із зображенням дерев і у діалоговому вікні, що відкриється, призначте зв'язок цієї області з файлом *dereva.htm* (рис. 4.16).

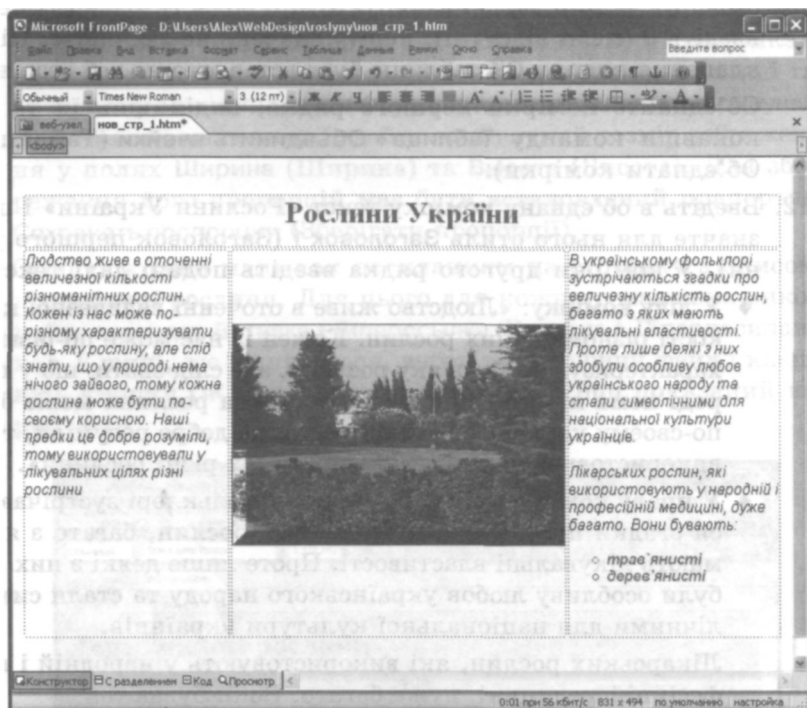


Рис. 4.15. Таблиця на головній сторінці

15. У той самий спосіб зробіть нижню прямокутну область малюнка гіперпосиланням на файл *kvly.htm*.
16. Перейдіть у режим HTML. Знайдіть у HTML-коді рядки, де визначена карта гіперпосилань, та відредагуйте їх, дописавши у тег `<AREA>` атрибут `TITLE`, значенням якого є пояснювальний текст, взятий у лапки. Для верхньої області введіть пояснення «Ознайомтеся з дерев'янистими рослинами», для нижньої — «Ознайомтеся з трав'янистими росли-

нами» (рис. 4.17). Тепер у разі наведення вказівника миші на гарячу область відповідний текст буде показаний у спливаючому прямокутнику. Так можна інформувати відвідувача про призначення гарячої області.

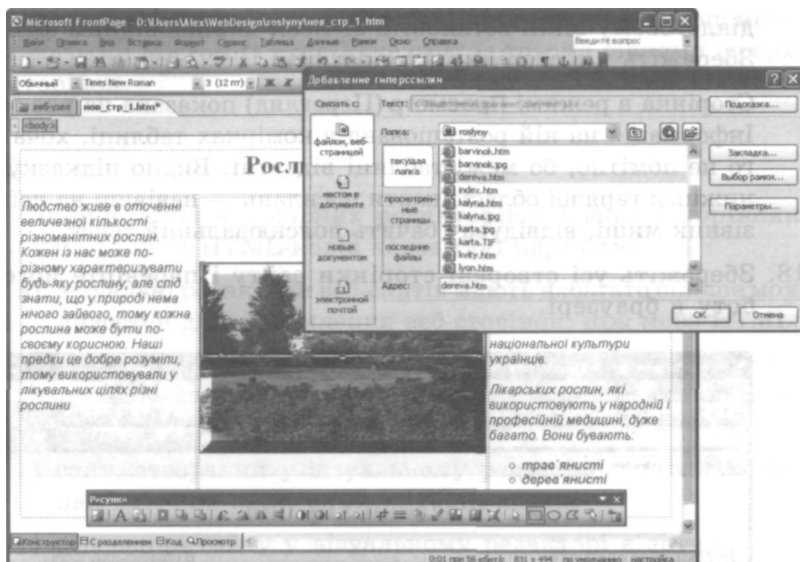


Рис. 4.16. Верхня прямокутна область малюнка є гіперпосиланням на файл dereva.htm

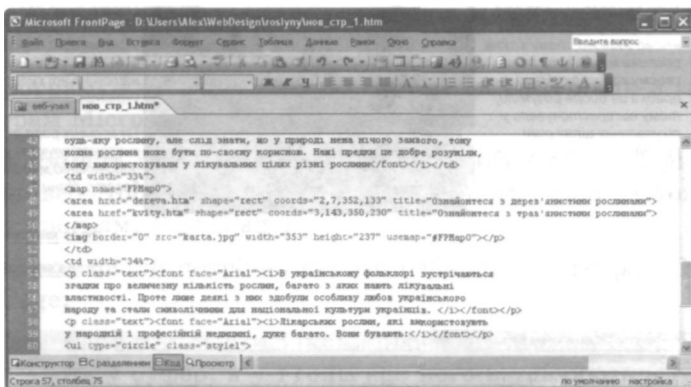


Рис. 4.17. Внесення коректив у HTML-код сторінки

17. Зробіть гіперпосиланнями на ці самі файли, dereva.htm та kvity.htm, елементи списку, який міститься у тексті, що розташований праворуч, — «трав'янисті» та «дерев'янисті». Для цього виділіть потрібне слово, відкрийте контекстне меню, виберіть у ньому команду Гіперссылка (Гіперпосилання) і у діалоговому вікні встановіть зв'язок із відповідним файлом. Збережіть цю сторінку як index.htm.

Сторінка в режимі Просмотр (Перегляд) показана на рис. 4.18. Інформація на ній розташована в комірках таблиці, хоча цього не помітно, бо межі таблиці відсутні. Видно підказку для нижньої гарячої області карти посилань — навівши на неї вказівник миші, відвідувач бачить пояснювальний текст.

18. Збережіть усі створені сторінки сайту і перевірте його роботу в браузері.

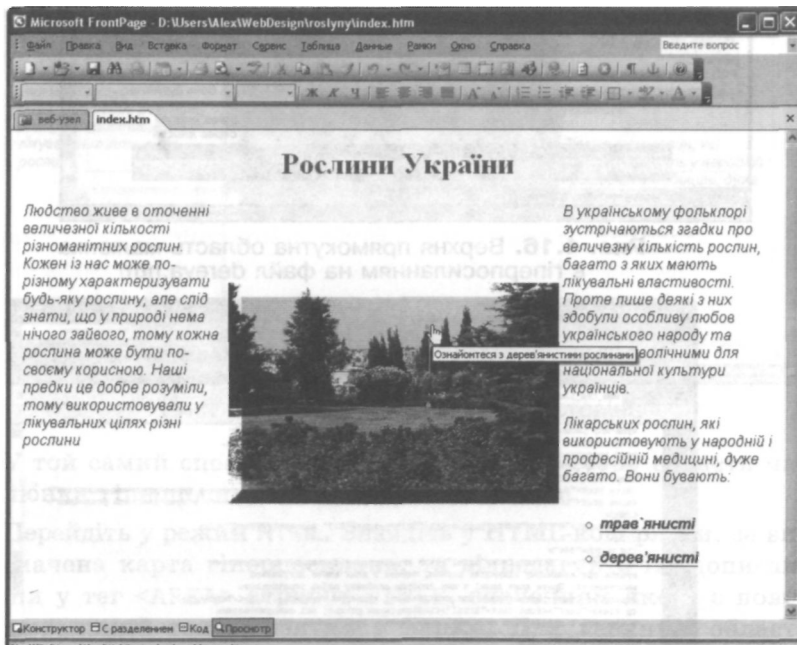


Рис. 4.18. Вигляд сторінки у режимі Просмотр (Перегляд)

Самостійна робота

Виберіть для сайту одну з тем, запропонованих у самостійній роботі до розділу 1, або запропонуйте свою. Підготуйте файли із зображеннями, придумайте текстовий супровід для головної сторінки та сторінок розділів і створіть за допомогою програми Microsoft FrontPage сайт із трьома рівнями ієрархії сторінок.

Тест

1. Використання візуальних редакторів веб-сайтів порівняно з написанням HTML-кодів надає такі переваги:
 - а) візуальні редактори веб-сайтів мають набагато більше можливостей для оформлення веб-сторінок, ніж мова HTML;
 - б) за допомогою візуальних редакторів процес створення сайту відбувається швидше, ніж у разі написання HTML-кодів;
 - в) сайт, створений у візуальному редакторі, займає менше пам'яті, ніж написаний у HTML-кодах;
 - г) створення сайту у візуальному редакторі є зручним, оскільки нагадує роботу в текстовому редакторі та не потребує знання тегів мови HTML;
 - д) на екрані відразу видно, який вигляд матиме веб-сторінка у вікні браузера.
2. У програмі FrontPage використовувати матеріали з інших програм Microsoft Office можна так:
 - а) документ з іншої програми Microsoft Office зберігають як веб-сторінку, яку потім відкривають і використовують у програмі FrontPage;
 - б) ці матеріали потрібно наново набирати у програмі FrontPage;
 - в) фрагмент документа з іншої програми Microsoft Office копіюють у буфер, а у FrontPage його вставляють в HTML-код веб-сторінки зі збереженням форматування;

- г) фрагмент документа з іншої програми Microsoft Office переносять через буфер на веб-сторінку, відображену у FrontPage у звичайному режимі зі збереженням форматування;
 - д) фрагмент документа з іншої програми Microsoft Office переносять через буфер на веб-сторінку у FrontPage у звичайному режимі без збереження форматування.
3. Переглядати веб-сторінки у FrontPage можна у режимах:
- а) Конструктор (Конструктор), Код (Код), Просмотр (Перегляд);
 - б) Конструктор (Конструктор), Папки (Папки), Просмотр (Перегляд);
 - в) Страница (Сторінка), Папки (Папки), Отчеты (Звіти), Переходы (Переходи);
 - г) Редактирование (Редагування) і Просмотр (Перегляд);
 - д) Страница (Сторінка), Папки (Папки), Просмотр (Перегляд).
4. Вкладка Конструктор (Конструктор) у режимі Страница (Сторінка) призначена для:
- а) перегляду списку сторінок, створених для сайту;
 - б) перегляду структури папок сайту;
 - в) створення та редагування веб-сторінки візуальними засобами;
 - г) створення та редагування веб-сторінки засобами мови HTML;
 - д) попереднього перегляду сторінки у тому вигляді, в якому вона відобразатиметься у браузері.
5. Шаблон у FrontPage — це:
- а) зразок сайту або сторінки, який FrontPage пропонує наповнювати вмістом у міру необхідності;
 - б) детальна інструкція щодо створення сайту на задану тему;
 - в) набір малюнків, які використовують для формування тла сторінок;
 - г) документ, де визначено формати шрифтів, абзаців, тла для нових веб-сторінок;
 - д) набір сайтів на одну тему.
6. Майстер FrontPage — це:
- а) список HTML-кодів із поясненням їх можливого використання;

- б) інтерактивний засіб для створення сайту на задану тему;
 - в) зразок сайту або сторінки, який FrontPage пропонує наповнювати вмістом у міру необхідності;
 - г) набір спеціальних форм для створення веб-сторінки на основі наданих відповідей із попереднім розміщенням частини інформації;
 - д) програма, що імпортує веб-сторінки, які створені за допомогою інших програм, у призначену папку для роботи у FrontPage.
7. Фреймсет — це:
- а) сторінка, в яку вставлена таблиця;
 - б) сторінка, текст якої поділено на кілька колонок;
 - в) сторінка, на якій в окремих фреймах розміщено кілька документів;
 - г) сторінка з розміщеною на ній картою гіперпосилань;
 - д) кілька веб-сторінок в одному вікні.
8. Текстову інформацію у програмі FrontPage форматують:
- а) лише за допомогою тегів HTML;
 - б) аналогічно до форматування в текстових процесорах, зокрема у програмі Word;
 - в) текст форматують у текстовому редакторі, а пізніше копіюють у програму FrontPage;
 - г) за допомогою команд головного меню програми; є можливість робити це за допомогою тегів HTML;
 - д) за допомогою спеціальної інструкції програми FrontPage.
9. Щоб розмістити малюнок із файлу на веб-сторінці, потрібно:
- а) у меню Файл (Файл) вибрати команду Открыть (Відкрити);
 - б) у меню Вставка (Вставка) вибрати команду Файл (Файл);
 - в) вибрати команду Вставка • Рисунок • Из файла (Вставка • Малюнок • Из файлу);
 - г) клацнути кнопку для вставлення зображення на панелі інструментів Рисунки (Малюнки);
 - д) клацнути кнопку для вставлення зображення на панелі інструментів Стандартная (Стандартна).

10. Щоб перетворити виділений фрагмент тексту або малюнок на гіперпосилання, необхідно:
 - а) у меню **Формат (Формат)** вибрати команду **Гіперссылка (Гіперпосилання)**;
 - б) у меню **Вставка (Вставка)** вибрати команду **Гіперссылка (Гіперпосилання)**;
 - в) у меню **Файл. (Файл)** вибрати команду **Открыть гиперссылку (Відкрити гіперпосилання)**;
 - г) виконати команду **Гіперссылка (Гіперпосилання)** контекстного меню;
 - д) ввести відповідні теги мови HTML у режимі **Конструктор (Конструктор)**.
11. Для вставлення таблиці на веб-сторінку потрібно:
 - а) у меню **Таблица (Таблица)** вибрати команду **Вставка • Таблица (Вставити • Таблица)**;
 - б) у меню **Вставка (Вставка)** вибрати команду **Таблица (Таблица)**;
 - в) у меню **Формат (Формат)** вибрати команду **Таблица (Таблица)**;
 - г) намалювати таблицю, використовуючи інструменти панелі **Рисунки (Малюнки)**;
 - д) вибрати команду **Файл • Создать • Таблица(Файл • Створити • Таблица)**.
12. Щоб структурувати інформацію на веб-сторінці за допомогою таблиці так, щоб самої таблиці не було видно на екрані, слід:
 - а) об'єднати всі комірки таблиці;
 - б) призначити для меж таблиці колір тла;
 - в) призначити для меж таблиці товщину 0;
 - г) у меню **Формат (Формат)** вибрати команду **Таблица (Таблица)** і настроїти властивості таблиці в діалоговому вікні;
 - д) вставити таблицю, розмістити інформацію у її комірках, після чого скасувати форматування таблиці.

Розділ 5

Таблиці каскадних стилів і динамічні веб-сайти

Знань, одержаних під час вивчення попередніх розділів, достатньо для створення добротних, хоча й простих сайтів. Та чи відмовиться той, хто прагне стати справжнім веб-майстром, від додаткових засобів оформлення сторінок та створення ефектів, яких неможливо досягнути звичайними методами? І чи не мріє він навчитися розробляти сучасні динамічні веб-ресурси, які реагують на дії відвідувачів, відповідають на їхні запитання, обмінюються з ними електронними листами, а може, навіть допомагають робити домашні завдання?

Тому зараз ви дізнаєтеся про ще один засіб форматування текстів і розміщення елементів на веб-сторінках, ознайомитеся з динамічними елементами сайтів та навчитеся їх створювати. Крім того, ви виконаєте дві практичні роботи, присвячені створенню динамічних елементів та форм, а також їхній обробці за допомогою скриптів.

Використання таблиць каскадних стилів

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

Поняття про таблиці каскадних стилів

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

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

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

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

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

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

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

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

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

Застосування каскадних стилів у HTML-документах

є три способи зв'язку каскадних стилів із HTML-документом: підключення зовнішньої таблиці стилів; розташування опису стилів у розділі `HEAD` документа; задавання властивостей стилів безпосередньо в тегах абзаців чи заголовків.

Підключення зовнішньої таблиці стилів

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

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

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

Вправа 5.1

Оформимо веб-сторінку за допомогою зовнішньої таблиці стилів.

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

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

- Збережіть файл з іменем `mystyle.css`.
- Створіть у Блокноті файл такого змісту:

```
<HTML>
<HEAD>
<LINK REL="stylesheet" TYPE="text/css"
  HREF="mystyle.css">
<TITLE>Приклад використання CSS</TITLE>
</HEAD>
<BODY>
<P>Цей текст оформлено відповідно до таблиці стилів, яка
міститься у файлі mystyle.css: розмір шрифту – 40 пунктів,
колір шрифту – зелений, а гарнітура – Comic Sans MS.</P>
</BODY>
</HTML>
```

- Збережіть цей файл з іменем `ryklad1.html` у тій самій папці, що і `mystyle.css`.
- Відкрийте файл `ryklad1.html` у вікні браузера (рис. 5.1). Як бачите, хоча в цьому документі не задано жодних параметрів для тегу `<P>...</P>`, текст абзацу набув нових властивостей, які описані в зовнішній таблиці стилів `mystyle.css`.

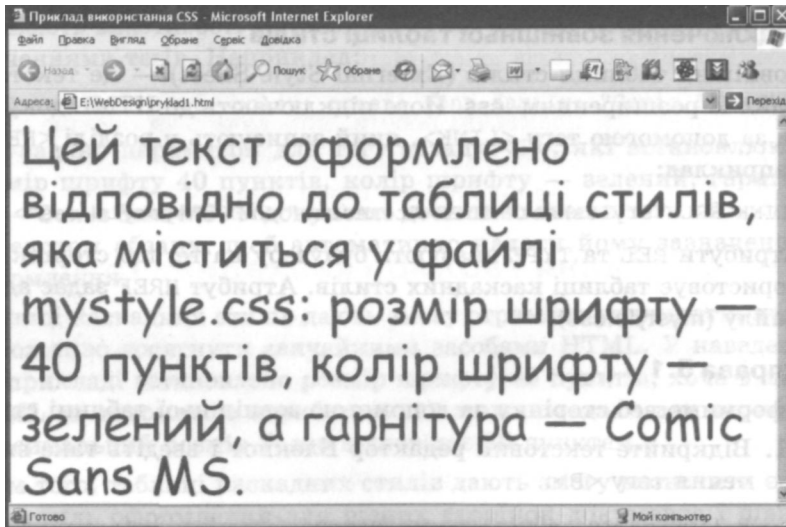


Рис. 5.1. Результат застосування таблиці каскадних стилів

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

Використання стилів внутрішньої таблиці

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

```
<HEAD>
<STYLE>
Тегі {властивість11: значення11; властивість12: значення12; ...;
    властивість1n: значення1n}
Тег2 {властивість21: значення21; властивість22: значення22; ...;
    властивість2m: значення2m}

</STYLE>
</HEAD>
```

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

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

```
<HTML>
<HEAD>
<STYLE>
H1 {font-size: 48pt; color: red}
H2 {font-size: 20pt; color: blue}
</STYLE>
<TITLE>Приклад використання CSS</TITLE>
</HEAD>
<BODY>
<H1>Для заголовка першого рівня визначено розмір 48 pt,
а колір тексту — червоний </H1>
<H2>Для заголовка другого рівня визначено розмір 20 pt,
а колір тексту — синій </H2>
<P>Для цього абзацу стиль не застосовано, для оформлення
тексту використано атрибути за умовчанням.
</BODY>
</HTML>
```

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

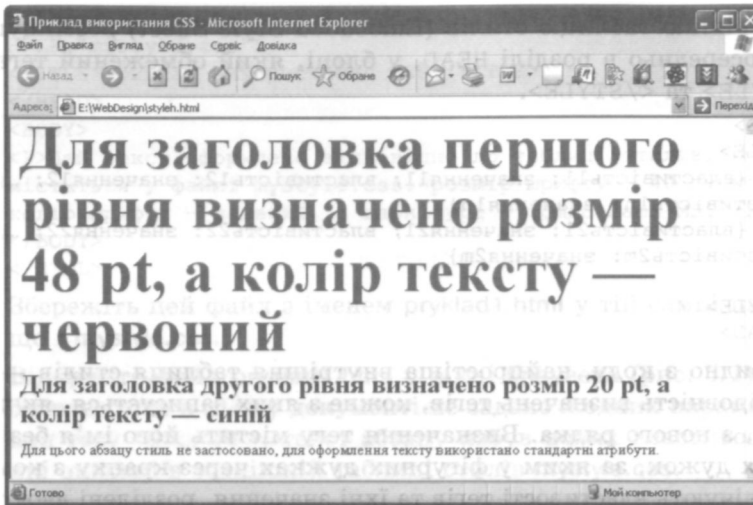


Рис. 5.2. Стилі заголовків першого та другого рівнів

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

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

```
<HTML>
<HEAD>
<STYLE>
.style1 {font-size: 38pt; color: white;
background-color: olive}
.style2 {font-size: 46pt; color: magenta;
background-color: mistyrose}
```

```

</STYLE>
<TITLE>Приклад використання CSS</TITLE>
</HEAD>
<BODY>
<P CLASS=style1>До цього абзацу застосовано стиль style1 </P>
<P CLASS=style2>До цього абзацу застосовано стиль style2 </P>
</BODY>
</HTML>

```

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

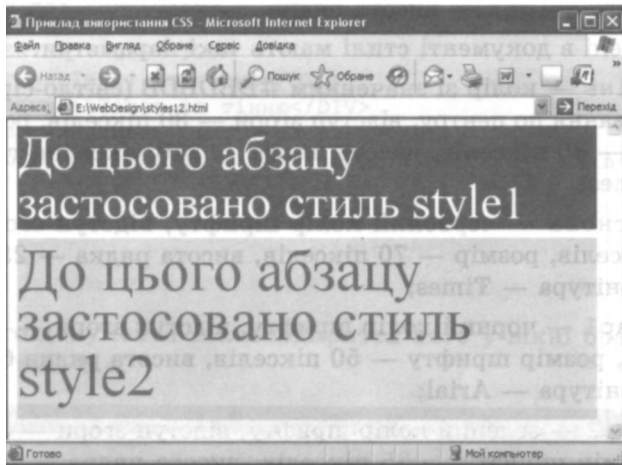


Рис. 5.3. Застосування різних стилів форматування для тегу абзацу

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

Приклад 5.1. Створення текстових ефектів за допомогою стилів

Визначимо для тегу `<BODY>` такий стиль: шрифт Arial чорного кольору розміром 16 пунктів. Внаслідок цього весь текст, що міститься між тегамі `<BODY>` та `</BODY>`, буде автоматично відформатований відповідно до цих властивостей. Тепер визначимо стилі з іменами `тінь`, `основа`, `шар1` і `шар2`. Текст запишемо в шарах, які накладаються. Спочатку відобразатиметься шар `тінь`, а на нього

накладатимуться шари основа, шар1 та шар2. Порядок накладання задано розташуванням фрагментів тексту в HTML-документі.

У прикладі буде застосовано такі властивості:

- `margin-top` — відступ згори (за від'ємних значень можна забезпечити накладання фрагментів);
- `color` — колір;
- `font-size` — розмір шрифту;
- `font-family` — сімейство шрифтів (гарнітура);
- `line-height` — висота рядка.

Визначені в документі стилі мають такі параметри:

- `.тiнь` — колір зі значенням `#DBDBDB` (світло-сірий), вирівнювання по центру, відступ згори — 30 пікселів, розмір шрифту — 80 пікселів, висота рядка — 270 пікселів, гарнітура — Times;
- `.основа` — червоний колір шрифту, відступ згори — -230 пікселів, розмір — 70 пікселів, висота рядка — 250 пікселів, гарнітура — Times;
- `.шар1` — чорний колір шрифту, відступ згори — -100 пікселів, розмір шрифту — 50 пікселів, висота рядка 65 пікселів, гарнітура — Arial;
- `.шар2` — зелений колір шрифту, відступ згори — 30 пікселів, розмір шрифту — 35 пікселів, висота рядка — 45 пікселів, гарнітура — Arial.

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

```
<HTML>
<HEAD>
<TITLE>Приклад використання CSS</TITLE>
<STYLE>
BODY {color: black; font-size: 16px; font-family: Arial}
.тiнь {color: #DBDBDB; text-align: center;; margin-top: 30px;
font-size: 80px; line-height: 270px; font-family: Times}
```



```

.основа {color: red; margin-top: -230px; font-size: 70px;
line-height: 250px; font-family: Times}
.шар1 {color: black; margin-top: -100px; medium; font-size:
50px; line-height: 65px; font-family: Arial}
.шар2 {color: green; margin-top: 30px; font-size: 35px;
line-height: 45px; font-family: Arial}
</STYLE>
</HEAD>
<BODY>
Приклад використання каскадних стилів
<CENTER>
<TABLE WIDTH=500 CELLPADDING=0 CELLSPACING=0 BORDER=0>
<TR>
<TD ALIGN=CENTER VALIGN=TOP>
<DIV CLASS=тінь>Текст із тінню</DIV>
<DIV CLASS=основа>Текст із тінню</DIV>
<DIV CLASS=шар1>Приклад тексту із тінню</DIV>
<DIV CLASS=шар2>Це приклад використання каскадних стилів</DIV>
</TD> </TR>
</TABLE>
</CENTER>
</BODY>
</HTML>

```

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

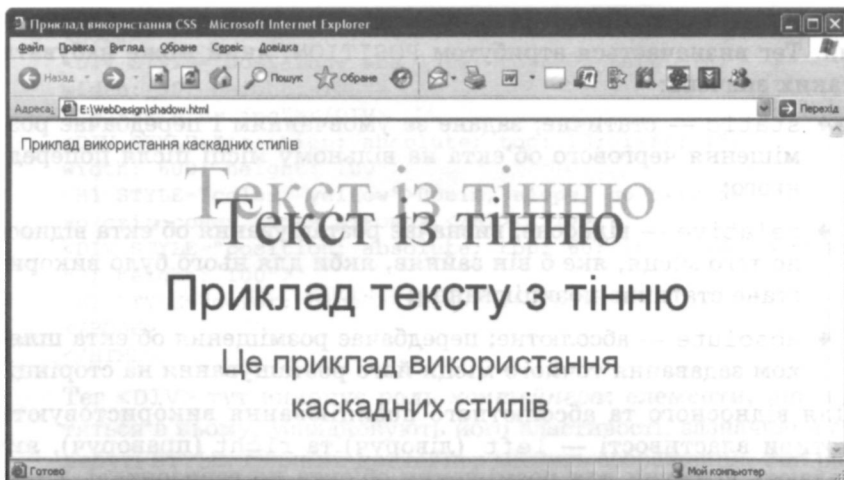


Рис. 5.4. Приклад накладання рядків тексту

Вбудовані стилі

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

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

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

Приклад використання атрибута STYLE для форматування заголовка другого рівня:

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

Позиціювання елементів веб-сторінки за допомогою стилів

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

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

- `static` — статичне; задане за умовчанням і передбачає розміщення чергового об'єкта на вільному місці після попереднього;
- `relative` — відносне; визначає розташування об'єкта відносно того місця, яке б він зайняв, якби для нього було використане статичне позиціювання;
- `absolute` — абсолютне; передбачає розміщення об'єкта шляхом задавання точного місця його розташування на сторінці.

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

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

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

Вправа 5.2

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

1. У програмі Блокнот введіть такий текст веб-сторінки:

```
<HTML>
<HEAD>
<TITLE>Позиціювання</TITLE>
</HEAD>
<BODY>
<DIV STYLE="position: absolute; top: 0; left: 70;
width: 50; height: 100">
<IMG SRC="fly.jpg"></DIV>
<DIV STYLE="position: absolute; top: 10; left: 15;
width: 600; height: 100">
<H1 STYLE="color: yellow">Повій, вітре, до схід сонця,
до схід сонця, край віконця.</H1></DIV>
<DIV STYLE="position: absolute; top: 60; left: 400; width:
50; height: 100">
<H1 STYLE="color: blue">Чайка</H1></DIV>
</BODY>
</HTML>
```

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

- 70 пікселів — відступ зліва; його ширина становить 50, висота — 100 пікселів. Другий контейнер містить текст «Повій, вітре, до схід сонця, до схід сонця, край віконця», він зміщений на 10 пікселів від початку сторінки вниз та на 15 вліво, ширина контейнера — 600, висота — 100 пікселів; завдяки використанню тегу H1 із атрибутом STYLE текст має відповідний до заголовка першого рівня розмір та жовтий колір. Третій контейнер зміщений на 60 пікселів униз та 400 ліворуч, ширина контейнера — 50, висота — 100 пікселів. Слово «Чайка» оформлене як заголовок першого рівня синього кольору.
2. Збережіть цей документ у файлі з іменем `pryklad2.html`. Скопіюйте у папку з цим документом зображення `fly.jpg` (можна використати інший файл, але його назва має збігатися з назвою в HTML-документі).
 3. Відкрийте файл `pryklad2.html` у браузері (рис. 5.5) — контейнери перекриваються згідно з розташуванням у документі.

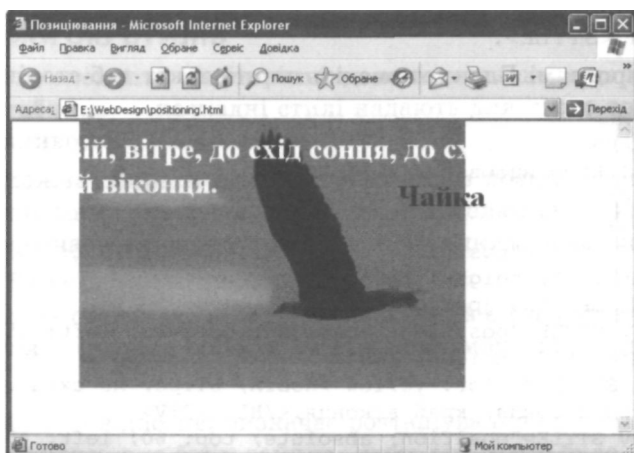


Рис. 5.5. Позиціонування фрагментів тексту і зображення за допомогою стилів

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

```
<DIV STYLE="position: absolute; top: 0; left: 70;  
width: 50; height: 100; z-index: 2">  
<IMG SRC="fly.jpg"></DIV>
```

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

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

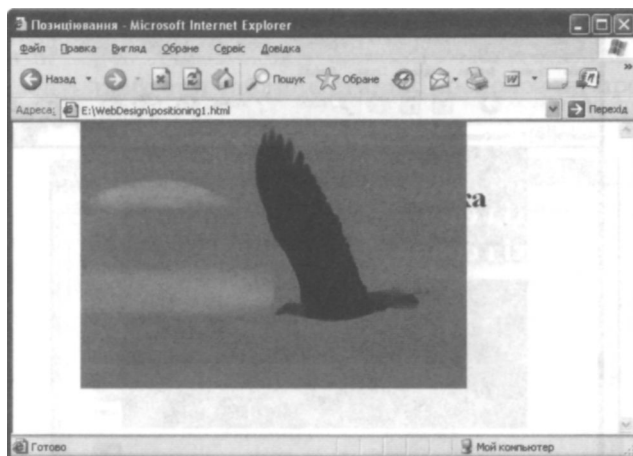


Рис. 5.6. Використання властивості z-index

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

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

- **попе** — якщо елемент вийде за межі наданого місця, він все одно буде показаний;

- clip — об'єкти, що виходять за межі, будуть обрізані;
- scroll — буде використано прокручування.

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

```
<DIV STYLE="position: absolute; top: 10; left: 15; width: 220; height: 120; overflow: scroll">
<H1 STYLE="color: yellow">Повій, вітре, до схід сонця, до схід сонця, край віконця </H1></DIV>
```

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

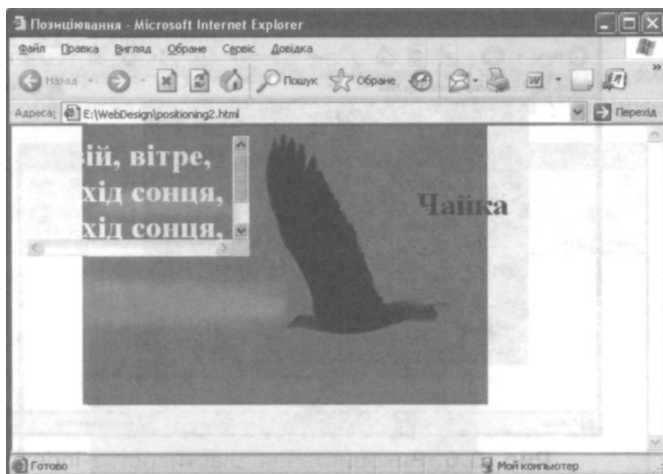


Рис. 5.7. Використання властивості overflow для реалізації прокручування

Застосування стилів у програмі FrontPage

Програма FrontPage дає змогу використовувати зовнішні та внутрішні таблиці стилів, а також вбудовані стилі. Розглянемо роботу із зовнішньою таблицею стилів, яка може бути пов'язана як з окремими, так і з усіма сторінками веб-сайту. Її найзручніше використовувати тоді, коли одні й ті самі стилі застосовуються до багатьох сторінок.

Створення та підключення зовнішньої таблиці стилів

Для створення зовнішньої таблиці стилів необхідно виконати такі дії.

1. Відкрити сайт, для якого потрібно створити цю таблицю. Для прикладу відкриємо веб-сайт roslyny, створений під час виконання практичної роботи № 7.
2. На панелі інструментів Стандартная (Стандартна) клацнути стрілку праворуч від кнопки Создание новой обычной страницы (Створення нової звичайної сторінки), вибрати в меню команду Страница (Сторінка) і у діалоговому вікні, що відкривається, перейти на вкладку Таблицы стилей (Таблиці стилів), яка містить шаблони таблиць стилів, пропоновані програмою FrontPage (рис. 5.8).

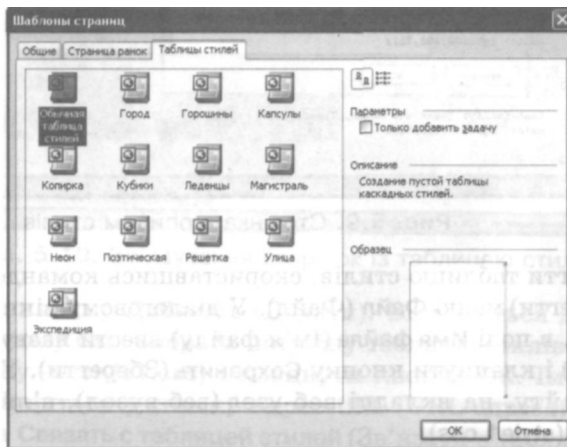


Рис. 5.8. Вибір шаблону таблиці стилів

Вибрати один із шаблонів таблиці стилів, наприклад Горошины (Горошини), і клацнути кнопку ОК. У вікні FrontPage відкриється документ нов_стр_1.CSS. Він міститиме опис стилів, які використовуватимуться для оформлення сторінок (рис. 5.9). Як уже зазначалося, стиль описується у такий спосіб: спочатку йде назва тегу об'єкта веб-сторінки (наприклад, гіперпосилання, заголовок, таблиця тощо) або ім'я створеного користувачем

стиля (з крапкою попереду), а у фігурних дужках — параметри його форматування.

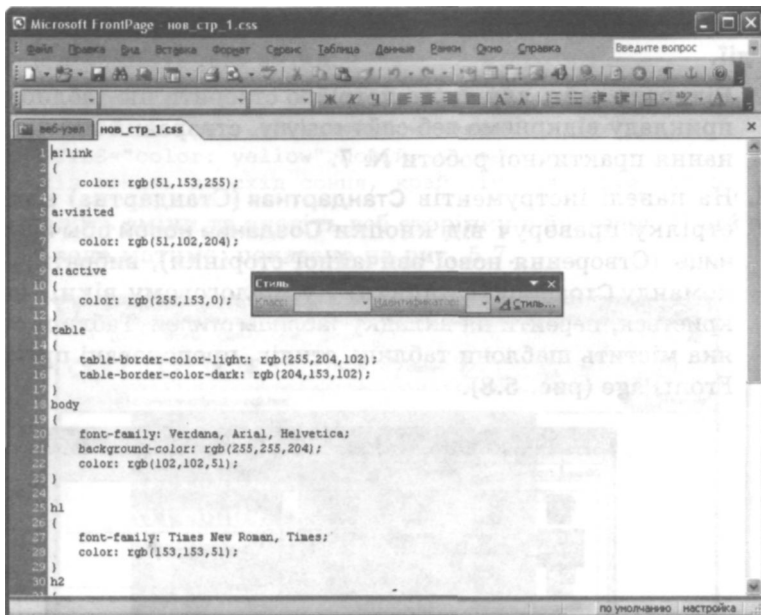


Рис. 5.9. Сторінка з описом стилів

4. Зберегти таблицю стилів, скориставшись командою Сохранить (Зберегти) меню Файл (Файл). У діалоговому вікні, що відкривається, в полі Имя файла (Ім'я файлу) ввести назву (наприклад, style1) і клацнути кнопку Сохранить (Зберегти). У списку файлів сайту, на вкладці веб-узел (веб-вузол), з'явиться новий файл (style1.css).

Після створення таблиці стилів її необхідно зв'язати зі сторінкою сайту. Для цього слід виконати такі дії.

1. Відкрити сторінку веб-сайту.
2. У меню Формат (Формат) вибрати команду Связи с таблицями стилей (Зв'язки з таблицями стилів). У верхній частині діалогового вікна Связать с таблицями стилей (Зв'язати з таблицями стилів), що відкривається, розташовані два перемикачі — все страницы (всі сторінки) та выделенные страницы (виділені сто-

рінки). Слід вибрати один із них залежно від того, потрібно з використанням таблиці стилів оформити всі сторінки сайту, чи лише поточну (рис. 5.10) (на вкладці веб-узел (веб-вузол) можна виділити кілька сторінок). Наприклад, виберемо перший перемикач.

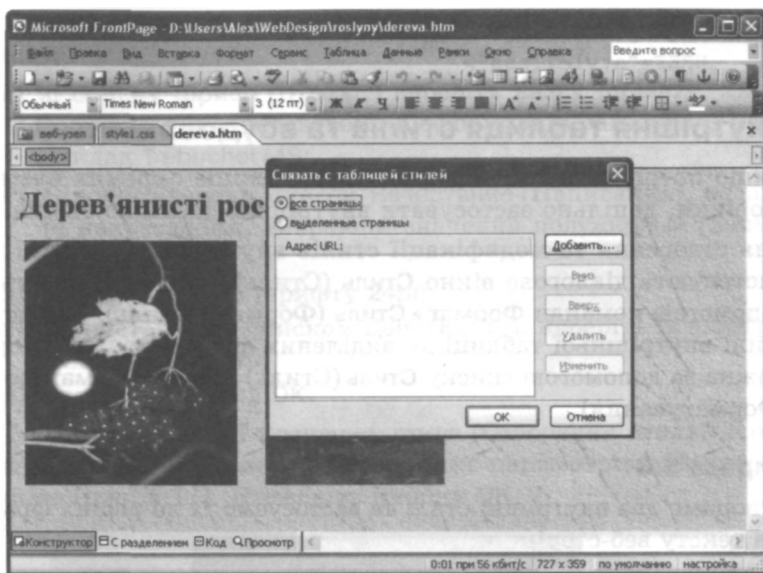


Рис. 5.10. Зв'язування сторінок із таблицею стилів

3. Клацнути кнопку **Добавить (Додати)**. Відкриється діалогове вікно, в якому слід вибрати потрібну таблицю стилів (у нашому випадку — `style1.css`) і клацнути кнопку **OK**. Ім'я файлу таблиці з'явиться у списку **Адрес URL (URL-адреса)** діалогового вікна **Связать с таблицей стилей (Зв'язати з таблицею стилів)**. Після клацання кнопки **OK**, таблицю стилів буде підключено до сторінок сайту відповідно до вибраного за допомогою перемикача режиму (у нашому випадку буде відкрито діалогове вікно з повідомленням про перезаписування всіх посилань на таблиці стилів, де потрібно натиснути кнопку **Продолжить (Продовжити)**). Застосувати стилі таблиці до виділених фрагментів веб-сторінки можна за допомогою списку **Стиль (Стиль)** панелі інструментів **Форматирование (Форматування)**.

Перегляньте сторінки веб-сайту. Оскільки була вибрана опція все сторониці (всі сторінки), до всіх елементів усіх веб-сторінок застосоване форматування із таблиці стилів. Відкрийте будь-який документ веб-сайту в режимі перегляду HTML-кодів. У розділі заголовків ви побачите рядок із тегом <LINK>, в якому записано адресу таблиці каскадних стилів:

```
<LINK REL="stylesheet" TYPE="text/css"  
      HREF="style1.css">
```

Внутрішня таблиця стилів та вбудовані стилі

Якщо потрібно змінити форматування лише окремих елементів сторінки, доцільно застосувати внутрішню таблицю стилів.

Для створення та модифікації стилів внутрішньої таблиці використовують діалогове вікно **Стиль (Стиль)**, яке відкривається за допомогою команди **Формат • Стиль (Формат • Стиль)**. Застосувати стилі внутрішньої таблиці до виділених фрагментів веб-сторінки можна за допомогою списку **Стиль (Стиль)** панелі **Форматування (Форматування)**.

Вправа 5.3

Створимо два внутрішні стилі та застосуємо їх до різних фрагментів тексту веб-сторінки.

1. Відкрийте сторінку веб-сайту, для якої потрібно створити внутрішні стилі. Це може бути, наприклад, головна сторінка веб-сайту «Рослини України».
2. У меню **Формат (Формат)** виберіть команду **Стиль (Стиль)**. Відкриється діалогове вікно, яке у лівій частині містить список тегів і стилів, а у правій — приклади оформлених ними абзаців і символів.
3. Для створення нового стилю клацніть кнопку **Создать (Створити)**. Відкриється діалогове вікно **Создание стиля (Створення стилю)**, де у поле **Имя (Ім'я)** необхідно ввести назву стилю. Введіть назву **style1** — це буде стиль для форматування заголовка оливковим кольором.
4. Клацніть кнопку **Формат (Формат)**, що розташована в нижній частині вікна. Відкриється меню з п'ятьма командами. У разі

їхнього вибору відкриваються діалогові вікна Шрифт (Шрифт), Абзац (Абзац), Граница (Межа), Нумерація (Нумерація) і Положення (Розташування), які дають змогу настроїти відповідні параметри форматування.

Виберіть у меню кнопки **Формат (Формат)** команду **Шрифт (Шрифт)**. У діалоговому вікні, що відкриється, виконайте такі дії:

- а) зі списку **Шрифт (Шрифт)** виберіть назву шрифту, який потрібно використовувати в оформленні заголовка, наприклад **Trebuchet MS**;
- б) використовуючи список **Начертание (Написання)**, задайте накреслення, вибравши значення **полужирный (напівжирний)**;
- в) виберіть розмір шрифту **24pt**;
- г) скориставшись списком **Цвет (Колір)**, виберіть із палітри оливковий колір;
- д) клацніть кнопку **ОК**.

У діалоговому вікні **Создание стиля (Створення стилю)** в області **Описание (Опис)** побачите опис параметрів створеного стилю (рис. 5.11). Натисніть кнопку **ОК**.

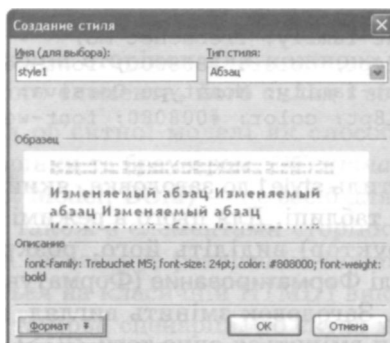


Рис. 5.11. Створення стилю style1

7. Аналогічно створіть стиль style2 з такими параметрами: шрифт — **Monotype Corsiva**, розмір шрифту — **18 пунктів**, написання — **напівжирне**, колір символів — **смарагдовий (бирюзовий)**. У діалоговому вікні **Стиль (Стиль)** з'являться назви обох

створених стилів (рис. 5.12). Зауважте, що крапки перед їхніми назвами додано автоматично.

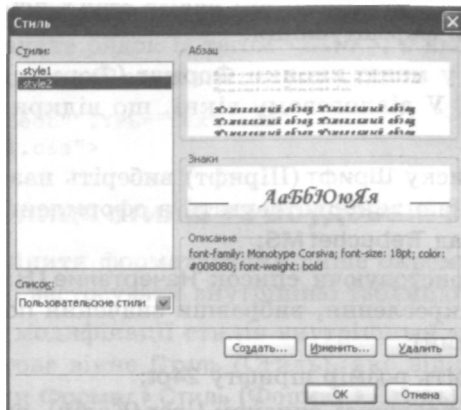


Рис. 5.12. Вікно для перегляду створених стилів style1 і style2

8. Уважно перегляньте сторінку в режимі відображення HTML-кодів. Ви побачите тег `<STYLE>`, в якому описані стилі style1 та style2:

```
<STYLE>
.Style1 {font-family: Trebuchet MS;
font-size: 24pt; color: #808000; font-weight: bold}
.style2 {font-family: Monotype Corsiva;
font-size: 18pt; color: #008080; font-weight: bold}
</STYLE>
```

9. Застосуйте стиль style1 до заголовка, який розміщений у верхній клітинці таблиці. Для цього в режимі перегляду Конструктор (Конструктор) виділіть його, розкрийте список Стиль (Стиль) панелі Форматирование (Форматування) і виберіть значення style1. Заголовок змінить вигляд, при цьому в HTML-коді сторінки зміниться опис тегу `<H1>`:
`<H1 CLASS=style1><I>Рослини України</I> </H1>`
10. В аналогічний спосіб застосуйте стиль style2 до маркованого списку; це буде рівнозначно доданню до кожного елемента списку атрибута `CLASS= style2`.
11. Збережіть зміни, перегляньте сторінку у вікні браузера.

До тексту та об'єктів веб-сторінки у FrontPage можна застосовувати також вбудовані стилі. Для виділених фрагментів тексту це роблять за допомогою діалогових вікон для форматування символів та абзаців Шрифт (Шрифт) і Абзац (Абзац): після вибору потрібних параметрів атрибут STYLE з відповідними властивостями буде автоматично додано до тегів фрагмента.

Для визначення вбудованого стилю виділених об'єктів використовують кнопку Стиль (Стиль) у діалогових вікнах їхніх властивостей, які відкривають за допомогою команди Формат • Свойства (Формат • Властивості). Після клацання цієї кнопки відкривається діалогове вікно Изменение стиля (Змінення стилю), подібне до вікна Создание стиля (Створення стилю) (рис. 5.11), в якому створюють чи змінюють вбудований стиль об'єкта. Для задання його параметрів можна використовувати стилі внутрішньої таблиці цієї веб-сторінки, а також діалогові вікна, що відкриваються після вибору команд меню кнопки Формат (Формат).

Поняття об'єктної моделі

З появою таблиць каскадних стилів у HTML з'явилася можливість будувати логічну структуру документа, а потім визначати формат її відображення. Цей підхід змінив усю технологію проектування сторінок сайту. Тепер можна визначити спочатку типи сторінок, потім логічні структури сторінок для кожного типу і, нарешті, для кожного логічного елемента, його склад і зовнішній вигляд.

Розглянемо поняття об'єктної моделі як способу взаємодії між HTML-кодом веб-сторінки та браузером. *Об'єктна модель документа* (Document Object Model, DOM) — це засіб для роботи зі структурою документа, а також з елементами сторінки в кодах HTML та у сценаріях. Вона забезпечує реалізацію технології динамічної HTML, яка ґрунтується на класичній HTML і використовує таблиці каскадних стилів та мови сценаріїв. Об'єктна модель документа є основою для того, щоб зробити елементи сторінки динамічно керованими під час її відтворення у вікні браузера.

Об'єктна модель описує кожний HTML-документ як набір окремих *об'єктів* — зображень, абзаців, списків і т. д. до найнижчого рівня, навіть до окремих символів. Кожний об'єкт може мати властивості, визначені у вигляді атрибутів. Наприклад, абзац <P>

має атрибут вирівнювання <ALIGN>, який може набувати значень left, right або center. В об'єктній моделі атрибут називають *властивістю* об'єкта. Об'єкт має також свої *методи* і *події*, які можуть відбуватися з ним і впливати на нього. Наприклад, зображення має подію OnMouseOver, яка відбувається тоді, коли користувач розміщує над ним вказівник миші. Можна керувати станом об'єктів, використовуючи методи з деякого набору стандартних методів. Все це й складає концепцію DOM як платформо-незалежного програмного інтерфейсу, який дає змогу програмам та скриптам керувати вмістом HTML-документів, змінювати їх структуру та оформлення.

Ми розглянули об'єкти HTML-документа, зокрема теги з текстовим наповненням. Проте браузер як програма також має свою об'єктну модель, при цьому моделі різних браузерів суттєво відрізняються. Браузер і документи, завантажені в нього, створюють ієрархічно організований набір об'єктів.

Сьогодні є можливість керувати як вмістом HTML-документів, так і браузером. Наприклад, для браузера Internet Explorer за допомогою об'єктно-орієнтованих мов JScript та VBScript можна писати програми, які називають сценаріями (скриптами), і вставляти їх у HTML-код. Такі сценарії розміщують у спеціальних тегах <SCRIPT> і </SCRIPT>.

Властивості, методи та поди

Об'єкти мають фіксовані імена і певні властивості. Наприклад, вікну браузера відповідає об'єкт Window, а HTML-документу, завантаженому в браузер, — об'єкт Document. Звичайні властивості — це змінні з фіксованими іменами, які мають певні значення. Одні властивості можна лише переглядати, інші можна змінювати. Для доступу до властивості об'єкта у мовах сценаріїв використовують такий синтаксис:

об'єкт.властивість

Наприклад, значенням властивості Document.Location є URL-адреса HTML-документа.

Властивістю об'єкта може бути інший об'єкт. При цьому перший об'єкт називають також *батьківським* (parent), а другий — *нащадком* (child). Якщо ми хочемо звернутися до властивості або

методу об'єкта `Object2`, який міститься в об'єкті `Object1`, то слід записати:

```
Object1.object2.властивість  
Object1.Object2.метод()
```

Наприклад, об'єкт `Document` є нащадком об'єкта `Window`. Якщо ми хочемо щось записати в документ, завантажений у поточне вікно, то можемо скористатися для цього методом `Write`.
Наприклад:

```
window.Document.Write ("Текст")
```

Методи — це пов'язані з об'єктами дії, які мають фіксовані імена, можуть мати параметри і повертати значення. Синтаксис застосування методу такий:

```
об'єкт.метод (список_параметрів)
```

Наприклад, метод `Window.Open` `{"www.protvaryn.narod.ru"}` відкриває нове вікно браузера і завантажує у нього сторінку, розташовану за вказаною адресою.

Крім властивостей і методів для кожного об'єкта існує набір *подій*. Події мають наперед визначені назви: натискання кнопки миші позначають `OnClick`, відпускання кнопки миші — `OnMouseUp`, а завантаження документа в браузер — `OnLoad` тощо. Настанням події можна скористатися для ініціювання певних дій.

Окрім об'єктів, у модель входять *колекції* — структури, що складаються з однотипних елементів. Колекція — це групування об'єктів для спрощеного доступу до них за допомогою програмного коду. Наприклад, об'єкт `Document` містить колекцію зображень `Images`. Ми можемо звернутися до зображення або за його іменем, або за порядковим номером:

```
Document.Images ("my_image")  
Document.Images (0)
```

Індекс (порядковий номер) елемента в колекції залежить від його розміщення у тексті HTML-документа. У колекціях об'єктної моделі перший елемент має нульовий індекс.

Колекціями також є:

- `All` — всі теги та елементи, що розташовані на веб-сторінці;
- `Frames` — фрейми;

- Images — зображення;
- Links — посилання та карти посилань.

Приклад 5.2. Об'єктна модель HTML-документа

Розглянемо приклад HTML-документа.

```
<HTML>  
<TITLE>Приклад</TITLE>  
<BODY>  
<P><IMG SRC="fly.jpg" NAME="image1">  
<A HREF="1.html" NAME="link1">Чайка</A></P>  
<P><IMG SRC="ptahy.jpg" NAME="image2">  
<A HREF="2.html" NAME="link2">Фламінго</A></P>  
</BODY>  
</HTML>
```

На рис. 5.13 показано, який вигляд має цей документ у вікні браузера.

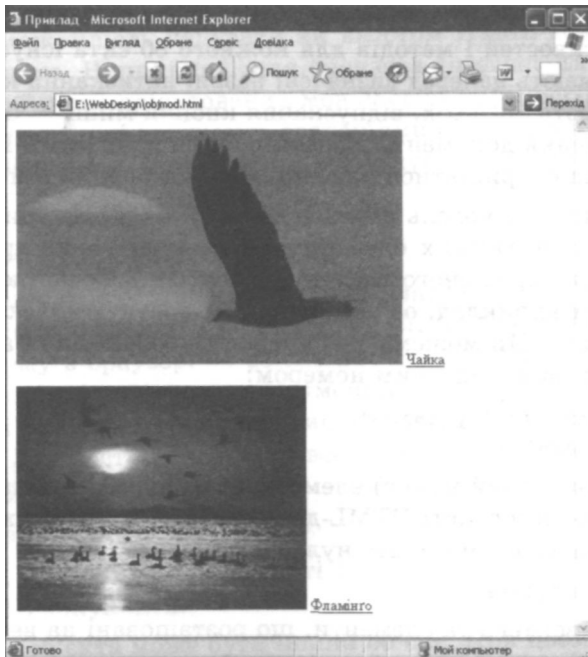


Рис. 5.13. Приклад веб-сторінки

На цій сторінці розташовано два малюнки та два посилання. На рис. 5.14 наведена ієрархічна схема об'єктного подання HTML-документа з цього прикладу.

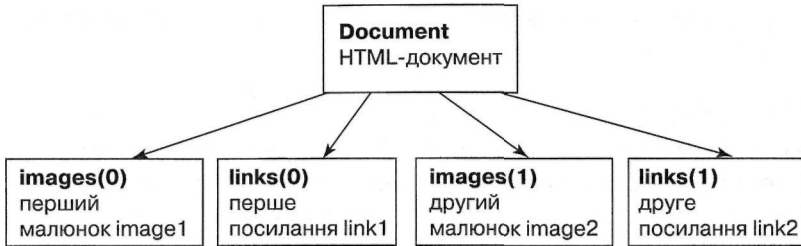


Рис. 5.14. Ієрархічна схема об'єктного подання HTML-документа

Головний об'єкт у цій структурі — Document. Перший елемент HTML-документа — зображення з іменем Image1 (це малюнок із файлу fly.jpg) — воно подане як об'єкт колекції Images (0) (нумерація елементів колекції починається з нуля, тому перший елемент має індекс 0). Другий елемент — посилання з іменем Link1 на файл l.html — нульовий об'єкт колекції Links тощо.

У HTML-документі імена елементів було визначено за допомогою атрибута NAME, і, хоча це не обов'язково, у нашому прикладі таке позначення виявилось корисним, оскільки тепер до елементів можна звертатися за іменами. Наприклад, для доступу до першого елемента сторінки можна записати Document.Images(0) або Document.Image1. Це особливо зручно тоді, коли на сторінці розташовано багато елементів, і важко відстежити, який індекс має той чи інший із них.

Об'єктна модель браузера Internet Explorer

Основні об'єкти моделі браузера Internet Explorer такі:

- Window — об'єкт найвищого рівня, це вікно браузера Internet Explorer;
- Frame — фрейм; усі фрейми утворюють колекцію Frames;
- History — об'єкт-історія, призначений для навігації за списком переглянутих у цьому сеансі сторінок;
- Location — об'єкт-розташування, це URL-адреса поточної сторінки;

- Event — об'єкт-подія, надає інформацію, що пов'язана з деякою подією у сценарії;
- Parent — батьківське вікно;
- Document — документ, область, в яку сценарій виводить інформацію.

Об'єкт Window

Об'єкт window займає в наборі об'єктів особливе, привілейоване місце, оскільки він є основним контейнером, в якому розміщується все те, чим можна керувати за допомогою браузера. Упродовж усього часу, поки вікно браузера відкрите, навіть якщо в ньому не завантажено жодного документа, об'єкт window буде визначений у поточній об'єктній моделі, що зберігається в пам'яті.

Діапазон використання об'єкта Window є широким — від керування вмістом вікна до настроювання його розмірів. Розміри вікна визначають область, де також розміщені смуги прокручування, панелі інструментів, рядок стану і меню — усе, що належить до атрибутів вікна. Кожен фрейм розглядають як окремий об'єкт Window.

У сценарії посилання на властивості та методи об'єкта Window задають безпосереднім звертанням до нього (квадратні дужки у записі означають, що параметри можуть бути відсутні):

```
Window.властивість  
Window.метод([параметри])
```

Оскільки об'єкт window існує завжди, то в посиланні на об'єкти всередині поточного вікна його назву можна не зазначати:

```
властивість  
метод([параметри])
```

Розглянемо деякі властивості та методи об'єкта Window.

Властивість Status

Рядок стану в нижній частині вікна браузера після наведення вказівника миші на будь-яке гіперпосилання зазвичай відображає URL-адресу посилання. Проте можна зробити так, що у певні моменти в рядку стану будуть показані спеціальні повідомлення, що надають корисні для користувача відомості. Наприклад, за-

мість того, щоб відображати адресу посилання, можна вивести короткий опис сторінки, присвоївши його як значення властивості Status.

Метод open()

Метод, за допомогою якого можна генерувати нові вікна, — Window.Open (). У ньому використовують три параметри, що визначають такі характеристики, як URL-адресу завантажуваного документа, назву та розмір вікна.

Наведений нижче рядок сценарію відкриває нове вікно abc заданого розміру (висота — 500 пікселів, ширина — 250) із HTML-документом l.html, розташованим у поточній папці:

```
NewWindow=  
Window.Open ("l.html", "abc", "HEIGHT=500, WIDTH=250")
```

Після цього змінну NewWindow можна використовувати як посилання на це вікно. Наприклад, щоб закрити його, можна скористатися методом Close () :

```
NewWindow.Close ()
```

Метод alert()

Цей метод генерує діалогове вікно-попередження, що відображає текст, заданий як параметр методу. Єдина кнопка ОК, напис якої не можна змінити, призначена для того, щоб користувач міг підтвердити, що він прочитав попередження.

Метод confirm()

У діалоговому вікні, яке відображає цей метод, є дві кнопки та текст, заданий як параметр методу. Для більшості версій браузерів і платформ це кнопки ОК і Cancel (Скасувати). Таке вікно називають діалоговим вікном підтвердження.

Метод ConfirmO повертає значення true (так), якщо користувач клацає кнопку ОК, і false (ні) — якщо кнопку Cancel (Скасувати). Це діалогове вікно і значення, яке воно повертає, можна використовувати для надання користувачу можливості керувати подальшими діями сценарію.

Метод `prompt`

Цей метод генерує діалогове вікно запити. Воно містить повідомлення, задане як перший параметр методу, і текстове поле для введення відповіді з підказкою, заданою другим параметром. Дві кнопки, наявні в діалоговому вікні, — `OK` і `Cancel` — дають змогу користувачу закрити діалогове вікно, повернувши у сценарій значення текстового поля (кнопкою `OK`) або спеціальне значення `null` (кнопкою `Cancel`).

З інформацією, отриманою внаслідок виконання всіх цих методів, можуть далі працювати сценарії: наприклад обробляти відповіді користувача, перевіряти їх правильність, долучати їх до баз даних тощо.

Наведемо приклади ще деяких властивостей, методів та подій об'єкта `Window`.

Властивості об'єкта `Window`

- `Parent` — повертає батьківське вікно;
- `Self` — повертає посилання на поточне вікно;
- `Top` — повертає посилання на головне вікно;
- `Name` — назва вікна.

Методи об'єкта `Window`

- `Open` — відкриває нове вікно браузера;
- `Close` — закриває поточне вікно браузера;
- `Focus` — робить вікно активним;
- `SetInterval` — вказівка процедури виконуватися періодично через задану кількість мілісекунд;
- `SetTimeout` — запускає програму через задану кількість мілісекунд після завантаження сторінки.

Події об'єкта `Window`

- `On Focus` — активізація вікна;
- `OnResize` — змінення користувачем розмірів вікна;
- `OnScroll` — прокручування вікна користувачем;
- `OnLoad` — повне завантаження сторінки.

Об'єкт `Window` має кілька об'єктів-нащадків: `Document`, `History`, `Navigator`, `Location`, `Event` і `Screen`.

Об'єкт Document

Цей об'єкт є центральним в ієрархічній об'єктній моделі й надає всю інформацію про HTML-документ, а також методи та події для роботи з документами. В ньому зберігається весь зміст сторінки. Властивості та методи об'єкта Document впливають здебільшого на вигляд сторінки у вікні. Проте метод Write дає змогу динамічно змінювати зміст у процесі завантаження документа.

Доступ до властивостей і методів об'єкта Document одержують так:

```
[Window.]Document.властивість  
[Window.]Document.метод([параметри])
```

Посилання на вікно Window для доступу до його об'єкта Document є необов'язковим.

Далі подано деякі властивості, методи та події цього об'єкта.

Властивості об'єкта Document

- ALinkColor — колір активних посилань на сторінці;
- BgColor — колір тла;
- LastModified — дата останнього змінення сторінки, доступна як текстовий рядок;
- LinkColor — колір ще не відвіданих гіперпосилань на сторінці;
- Location — повна URL-адреса документа;
- Referer — URL-адреса сторінки, що викликала поточну;
- VLink — колір відвіданих посилань на сторінці.

Методи об'єкта Document

- Clear — очищає виділений фрагмент;
- Close — завершує сеанс запису в поточний документ (для подальших операцій запису документ буде очищено);
- Write — записує текст у документ, який міститься в поточному вікні;
- WriteLn — записує текст у документ, що міститься в поточному вікні, з переведенням курсору на наступний рядок;
- Open — відкриває зазначений як параметр документ.

Події об'єкта Document

- `OnClick` — відбувається, коли користувач клацне кнопкою миші на документі;
- `OnMouseDown` — відбувається, коли користувач натискає кнопку миші і не відпускає її;
- `OnMouseOver` — відбувається, коли вказівник миші розташований на елементі документа;
- `OnMouseMove` — відбувається, коли користувач переміщує мишу;
- `OnDragStart` — відбувається, коли користувач починає перетягувати об'єкт за допомогою миші;
- `OnError` — відбувається, якщо сталася помилка;
- `OnKeyDown` — виникає під час натискання клавіші;
- `OnKeyPress` — виникає, коли користувач натиснув клавішу й утримує її;
- `OnKeyUp` — виникає, коли користувач відпускає клавішу;
- `OnLoad` — виникає після повного завантаження документа.

Якщо якийсь елемент входить у колекцію документа, то звернутися до нього можна, зазначивши його ім'я або номер у колекції через крапку після імені об'єкта. Наприклад, запис `Document.Images(i).Src="1.gif"` означає, що властивості `Src` елемента з номером `i` з колекції `Images` об'єкта `Document` потрібно присвоїти значення `1.gif`; іншими словами, `i`-тий малюнок на цій веб-сторінці буде завантажено з файлу `1.gif`.

Сценарії

Щоб веб-сторінка була інтерактивною, тобто могла взаємодіяти з користувачем, і динамічною, необхідно використовувати скрипти, або сценарії. *Сценарій* (script, скрипт) — це програма, написана спеціальною мовою програмування і вбудована в HTML-документ. Сценарії описують усі можливі дії над елементами HTML-документа під час взаємодії з користувачем (наприклад, реакцію на натискання кнопки миші, зміну вмісту сторінки залежно від певних дій користувача тощо).

Мова програмування JavaScript

Стандартною мовою для веб-скриптів є JavaScript — мова програмування, яка дає змогу вбудовувати виконуваний модулі в документи, написані в кодах HTML. Програму, створену мовою JavaScript, інтерпретує браузер під час завантаження документа, в який вміщено її код. Проте різні браузери сприймають різні її варіанти. Версія мови JavaScript від корпорації Майкрософт, що має назву JScript, є найближчою до стандарту. Браузер Microsoft Internet Explorer підтримує не лише JScript, а й ще одну мову скриптів — Visual Basic Script (VBScript).

За допомогою мови JavaScript, можна, наприклад, зробити так, щоб після клацання зображення лівою кнопкою миші воно змінювало свій вигляд. Її засобами можна реалізувати й складнішу поведінку елементів сторінки, скажімо, змусити їх пересуватися з необхідною швидкістю і за бажаною траєкторією.

За допомогою веб-сценаріїв можна створити принципово новий інтерфейс користувача для своєї сторінки. Всі події, генеровані браузером, такі як клацання кнопок, модифікація полів форм і переміщення між сторінками, можна перехопити й обробити засобами JavaScript. Ця мова придатна для розв'язування рутинних завдань, таких як перевірка достовірності даних, опрацювання форм, виконання дій над текстовими і числовими значеннями, тобто тих завдань, які не можна розв'язати за допомогою стандартних засобів мови HTML.

Основні області застосування мови JavaScript:

- динамічне створення документа HTML за допомогою скриптів;
- перевірка достовірності полів форм HTML до передавання їх на сервер;
- локальне введення інформації для керування програмою;
- надання користувачу можливості вибору операцій, виконуваних браузером;
- виведення повідомлень для користувача у діалогових вікнах;
- локальне опрацювання форм, введення інформації користувачем.

Щоб використовувати мову скриптів ефективно, необхідно орієнтуватися в об'єктній моделі HTML-документа.

Програмний код JavaScript можна помістити в документ HTML у три способи:

- окремі скрипти розмістити в тілі документа, там, де в їхньому використанні є потреба;
- скрипти (функції, оголошення об'єктів) розмістити у заголовній частині документа між тегами <HEAD>...</HEAD>, а використовувати їх у тілі документа;
- зберегти скрипт у файлі (зазвичай із розширенням .js), а в документі дати посилання на нього.

У першому випадку для того, щоб повідомити браузер про використання JavaScript, у тіло HTML-документа потрібно вставити парний тег <SCRIPT> з атрибутом LANGUAGE="JavaScript":

```
<SCRIPT LANGUAGE="JavaScript">  
програма на JavaScript  
</SCRIPT>
```

Оскільки браузер Internet Explorer здатний розпізнавати програму на JavaScript, вміщену між тегами <SCRIPT>...</SCRIPT>, для нього зазначений атрибут задавати необов'язково.

Для відвідувачів сторінки, у яких встановлений браузер, що не підтримує JavaScript, після тегів <SCRIPT>...</SCRIPT> вміщують теги <NOSCRIPT>...</NOSCRIPT>, які описують вміст та вигляд безскриптового варіанта сторінки.

JavaScript, як і будь-яка мова програмування, має набір інструкцій, що описують виконання тих чи інших дій. Синтаксис цих інструкцій схожий на синтаксис операторів у мові Java.

Створимо просту веб-сторінку зі сценарієм. Він виводитиме на екран вікно з повідомленням «!!!». Для цього використаємо метод alert ().

```
<HTML>  
<HEAD>  
<TITLE>приклад</TITLE>  
</HEAD>  
<BODY>  
<SCRIPT>  
alert {"!!!"};  
</SCRIPT>  
</BODY>  
</HTML>
```


Вигляд цієї сторінки у вікні браузера показано на рис. 5.15.



Рис. 5.15. Найпростіша веб-сторінка зі сценарієм

Слід зазначити, що з міркувань безпеки браузер Internet Explorer за умовчанням блокує всі активні елементи веб-сторінок, тому для коректної роботи з ними необхідно додатково підтверджувати запуск скрипту. Коли браузер завантажує сторінку зі скриптом, вгорі вікна з'являється панель безпеки, після клацання на якій необхідно вибрати команду Дозволити заблокований вміст. У вікні, що відкривається (рис. 5.16), слід дозволити браузеру запускати активний вміст. Після цього скрипт буде виконуватися без обмежень.

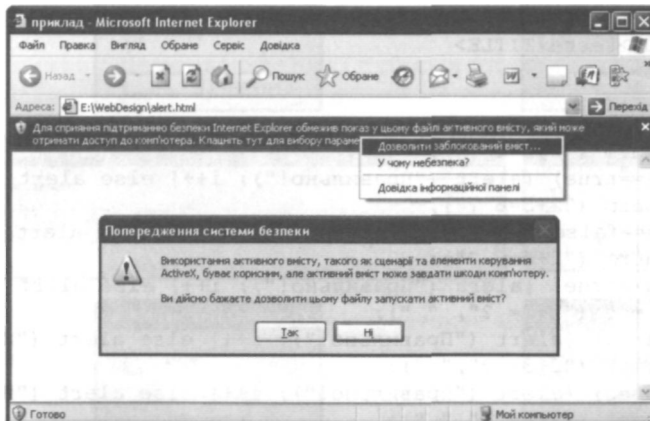


Рис. 5.16. Панель безпеки браузера Internet Explorer та діалогове вікно, що дає змогу запускати активний вміст

Вправа 5.4

Розглянемо простий тест для перевірки навичок із додавання, реалізований за допомогою вбудованого сценарію JavaScript. У ньому використано методи Alert (генерує діалогове вікно-попередження для виведення результатів тесту), Confirm (повертає значення true (істина), якщо користувач клацає кнопку ОК, і false (хибність), якщо користувач клацає кнопку **Cancel** — це буде використано для перевірки правильності виконання завдань) та Prompt, в якому відображається запитання і надається текстове поле для введення відповіді користувачем. У змінній і накопичуватиметься кількість правильних відповідей.

У цьому скрипті використано команди розгалуження, в яких перевіряються умови `s==true` або `s==false` — правильно чи неправильно дано відповідь на запитання тесту. Користувач отримує повідомлення про це, і якщо відповідь була вірною, до змінної і додається 1 (**I++**).

Ще одне розгалуження із вкладеними розгалуженнями аналізує кількість правильних відповідей користувача (що міститься у змінній `i`). Залежно від її числового значення за допомогою методу Alert буде виведена словесна оцінка — «чудово», «добре» або «погано». Останнє повідомлення показує суму балів.

1. У редакторі Блокнот наберіть код веб-сторінки:

```
<HTML>
<TITLE>test</TITLE>
<BODY>
<SCRIPT>
var s, i=0
s=confirm ("2+2=4 ?");
if (s==true) {alert ("Правильно!"); i++;} else alert ("Hi!")
s=confirm ("2+3=6 ?");
if (s==false) {alert ("Правильно!"); i++;} else alert ("Hi!")
s=confirm ("2+4=6 ?");
if (s==true) {alert ("Правильно!"); i++;} else alert ("Hi!")
s=prompt ("2+2= ?", " ");
if (s==4) {alert ("Правильно!"); i++;} else alert ("Hi!")
s=prompt ("2+3= ?", " ");
if (s==5) {alert ("Правильно!"); i++;} else alert ("Hi!")
s=prompt ("2+4= ?", " ");
if (s==6) {alert ("Правильно!"); i++;} else alert ("Hi!")
if (i>=5) alert ("чудово")
```

```
else if (i>=3) alert ("добре")
else alert ("погано")
alert ("Сума балів="+i)
</SCRIPT>
</BODY>
</HTML>
```

- Збережіть документ у файлі **test.html**.
- Відкрийте веб-сторінку **test.html** у вікні браузера. У разі потреби розблокуйте активний вміст документа.
- Дайте відповіді на запитання тесту. Деякі етапи тестування показані на рис. 5.17. Під час тестування викликаються такі методи:
 - перше запитання (2+2=4?) — метод Confirm;
 - повідомлення **про** правильну відповідь праворуч — метод Alert;
 - четверте запитання (2+2=?) — метод Prompt;
 - повідомлення **про** неправильну відповідь праворуч — метод Alert;
 - виведення словесної оцінки — метод Alert;
 - повідомлення про кількість балів праворуч — метод Alert.
- Доповніть тест, щоб у ньому було 8 запитань.

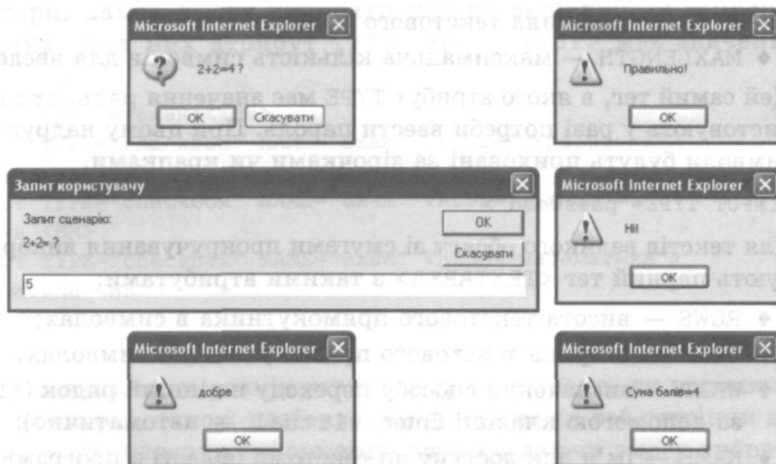


Рис. 5.17. Етапи тестування

Використання форм

Введення інформації можна організувати не лише через діалогові вікна, а й за допомогою *форм* — наборів елементів керування, які забезпечують взаємодію людини з програмою. Користувач може вводити інформацію з клавіатури, а також вибираючи потрібні перемикачі чи прапорці. Його дії потім опрацьовує програма-сценарій.

Елементи форми

Форма може містити поля для введення текстової інформації, списки для вибору заздалегідь визначених відповідей, прапорці, перемикачі, кнопки та інші елементи керування. Розглянемо докладніше деякі з них.

Текстові поля

Для введення текстових даних у спеціальні поля використовують тег `<INPUT>`, в якого атрибут `TYPE` має значення `text`:

```
<INPUT TYPE="text">
```

Інші атрибути тегу `<INPUT>`:

- `NAME` — ім'я змінної, в якій зберігається введене значення;
- `VALUE` — початкове значення;
- `SIZE` — довжина текстового поля;
- `MAXLENGTH` — максимальна кількість символів для введення.

Цей самий тег, в якого атрибут `TYPE` має значення `password` використовують у разі потреби ввести пароль. При цьому надруковані символи будуть приховані за зірочками чи крапками.

```
<INPUT TYPE="password">
```

Для текстів великого обсягу зі смугами прокручування використовують парний тег `<TEXTAREA>` з такими атрибутами:

- `ROWS` — висота текстового прямокутника в символах;
- `COLS` — ширина текстового прямокутника в символах;
- `WRAP` — визначення способу переходу на інший рядок (`off` — за допомогою клавіші `Enter`, `virtual` — автоматично);
- `NAME` — ім'я для доступу до текстової області з програми-сценарію.

Перемикачі

Щоб дізнатися про думку відвідувача з того чи іншого приводу, не змушуючи його вводити інформацію, використовують перемикачі з варіантами відповідей. Відвідувач переглядає їх і вибирає потрібний. Перемикачі дають змогу вибрати лише один із запропонованих варіантів.

Для створення перемикачів використовують тег `<INPUT>`, в якого атрибут `TYPE` (тип) має значення `radio`. Групі перемикачів, що стосуються одного питання, обов'язково присвоюють однакове ім'я (`NAME`). Атрибут `VALUE` позначає відповідний перемикачу варіант відповіді для розробника форми, тоді як напис після тегу `<INPUT>` — для користувача. Для вибору одного з перемикачів за умовчанням використовують атрибут `CHECKED`.

```
<H4>Вибери мову:</H4>
<INPUT TYPE="radio" NAME="lang" VALUE="українська" CHECKED>
Українська <BR>
<INPUT TYPE="radio" NAME="lang" VALUE="російська">
Російська <BR>
<INPUT TYPE="radio" NAME="lang" VALUE="англійська">
Англійська <BR>
```

Прапорці

Прапорці дають змогу вибрати кілька варіантів із запропонованих. Для них атрибут `TYPE` тегу `<INPUT>` має значення `checkbox`.

```
<H4>Вибери екзамен:</H4>
<INPUT TYPE="checkbox" NAME="exam" VALUE="українська">
Українська <BR>
<INPUT TYPE="checkbox" NAME="exam" VALUE="фізика">
Фізика <BR>
<INPUT TYPE="checkbox" NAME="exam" VALUE="математика">
Математика <BR>
```

Списки

Списки, які можна розміщувати у формі так само, як перемикачі та прапорці, також позбавляють відвідувачів веб-сторінки від необхідності вводити інформацію вручну, даючи змогу вибрати відповідь із запропонованих варіантів. Список розміщують між

тегами <SELECT> та </SELECT>, а його елементи визначають за допомогою тегу <OPTION>. Наприклад:

Район:

```
<SELECT>
<OPTION NAME="frank">Франківський
<OPTION NAME="syh">Сихівський
<OPTION NAME="gal">Галицький
</SELECT>
```

Якщо замість <SELECT> записати <SELECT MULTIPLE>, користувач отримає можливість вибрати зі списку кілька варіантів. Атрибут SIZE=N тегу <SELECT> обмежує КІЛЬКІСТЬ показаних елементів списку числом N, після чого буде використано прокручування.

Кнопки

Заповнивши форму текстом та вибравши потрібні елементи керування, користувач повинен мати можливість підтвердити свої дії або скасувати помилково введені дані.

Для підтвердження правильності введення використовують кнопку, яку створює тег <INPUT> з атрибутом TYPE="submit". На цій кнопці буде напис, заданий атрибутом VALUE. Аналогічно створюють кнопку скасування дії: атрибут TYPE у цьому випадку повинен мати значення reset. Звичайна кнопка, з якою можна зв'язати будь-яку дію, має атрибут TYPE="button".

```
<INPUT TYPE="submit" VALUE="Відіслати">
<INPUT TYPE="reset" VALUE="Оновити форму">
<INPUT TYPE="button" VALUE="Обчислити">
```

Для оригінального оформлення кнопки в неї можна вставити малюнок. Для цього використовують такий синтаксис:

```
<INPUT TYPE="image" SRC="..."
WIDTH=... HEIGHT=... ALT="..." VALUE=...>
```

Надсилання форми

Отже, розробник веб-сторінки має змогу отримати відповіді користувача на поставлені запитання, не обмежуючись діалоговими вікнами методів Configm та Pprompt. Форми можна проектувати відповідно до своїх потреб, додавати до них зображення та інші елементи. Після заповнення форми користувач надсилає дані на подальше опрацювання.

Включені в документ HTML елементи на зразок полів введення даних, перемикачів, прапорців та кнопок вміщують у тег форми `<FORM> . . . </FORM>`. Цей тег повинен мати певні атрибути, наприклад:

```
<FORM METHOD="post" ACTION="/bin/serv"  
ENCTYPE="text/plain">
```

Атрибут `METHOD` може мати значення `post` або `get`, які визначають різні методи передавання інформації з форми на URL-адресу сценарію-обробника, що зазначена в атрибуті `ACTION`. У свого провайдера потрібно уточнити, який із методів слід використовувати. Необхідно також знати місце розташування доступних сценаріїв та їхні імена.

Дані форми можна пересилати на певну адресу електронної пошти, якщо задати атрибут `ACTION=mailto:адреса_номТМ`.

Атрибут `ENCTYPE` визначає, в який спосіб дані форми потрібно кодувати перед надсиланням на сервер. За умовчанням використовується кодування, під час якого символи замінюються комбінацією символу «% » та шістнадцяткового коду символу тексту в ASCII-таблиці. Щоб дані форми не перетворювалися на шістнадцяткові числа, слід задавати значення `text/plain`.

Вправа 5.5

У цьому прикладі користувачу запропоновано заповнити форму, дані якої будуть використані для формування статистичної інформації про вступні іспити у деякий навчальний заклад. Запрошення для заповнення форми записане як заголовок четвертого рівня. Для надсилання форми обрано метод `post` та зазначено адресу електронної пошти.

Спочатку відвідувачу пропонують ввести своє прізвище у текстове поле (використано підказку у вигляді спливаючого повідомлення). Після цього користувачу потрібно вибрати район, в якому він проживає, у відповідному списку, що міститься між тегами `<SELECT>` та `</SELECT>`. Для визначення мови, якою відвідувач бажає здавати іспити, використовуються перемикачі. Іспити вибирають за допомогою прапорців, кілька з яких або навіть усі можна встановити одночасно.

Використання кількох груп варіантів відповідей для різних запитань потребує, щоб елементи кожної такої групи мали однакове значення атрибута NAME у тегах INPUT, але воно було різним для різних груп.

Для надсилання форми передбачена стандартна кнопка з написом Відіслати. Для очищення полів форми від введеної інформації створимо кнопку Оновити форму.

1. У текстовому редакторі Блокнот наберіть текст HTML-документа. Замість електронної адреси olena@zirka.lviv.ua у тегу <FORM> введіть власну адресу електронної пошти.

```
<HTML>
<TITLE>Форми</TITLE>
<BODY>
<H4>Заповни форму для складання іспитів</H4>
<FORM METHOD="post" ACTION="mailto:olena@zirka.lviv.ua"
ENCTYPE="text/plain">
Прізвище: <INPUT TYPE="text" NAME="family" size=25
TITLE="Введи тут своє прізвище українською мовою"><BR><BR>
З цього списку вибери район, де ти проживаєш: <BR>
<SELECT>
<OPTION NAME="frank">Франківський
<OPTION NAME="syh">Сихівський
<OPTION NAME="gal">Галицький
</SELECT>
<BR><BR>
Зазнач мову, якою ти складатимеш іспити: <BR>
<INPUT TYPE="radio" NAME="lang"
VALUE="українська">Українська <BR>
<INPUT TYPE="radio" NAME="lang"
VALUE="російська">Російська <BR>
<INPUT TYPE="radio" NAME="lang"
VALUE="англійська">Англійська <BR><BR>
Вибери іспити:<BR>
<INPUT TYPE="checkbox" NAME="subj"
VALUE="українська">Українська <BR>
<INPUT TYPE="checkbox" NAME="subj"
VALUE="фізика">Фізика <BR>
<INPUT TYPE="checkbox" NAME="subj"
VALUE="математика">Математика <BR>
<INPUT TYPE="submit" VALUE="Відіслати">
<INPUT TYPE="reset" VALUE="Оновити форму">
```



```
</FORM>  
</BODY>  
</HTML>
```

2. Збережіть документ із назвою anketa1.html.
3. Відкрийте його у вікні браузера.
4. Заповніть форму, давши відповідь на всі запитання. Після цього веб-сторінка матиме приблизно такий вигляд, як на рис. 5.18. Клацніть кнопку Відіслати.
5. Відкрийте свою електронну скриньку. Перевірте інформацію, що надійшла на неї з форми.

Зауважте, що надіслати інформацію можна лише за умови, що на комп'ютері настроєно поштовий клієнт (наприклад, Microsoft Outlook).

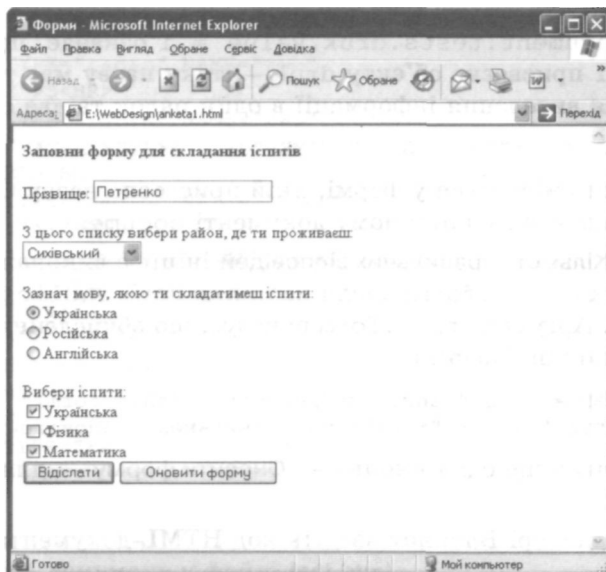


Рис. 5.18. Веб-сторінка із введеною у поля форми інформацією

Вправа 5.6

Розглянемо невелике тестування із вбудованою функцією реє-
вірка, параметром якої є форма з відповідями користувача на

запитання. Використано три запитання, для першого правильна відповідь № 2, для другого — № 1, для третього — № 3.

Відповіді на тестові запитання вибираються у групах перемикачів. Для кожного запитання є своє значення атрибута NAME у тегу INPUT — для першого NAME="one", для другого — NAME="two", для третього — NAME="three". Кожна така група перемикачів інтерпретується у функції перевірки як масив значень.

Функція виконує перевірку такого типу:

```
if (tests.one[1].Checked) {i++;}
```

Цей рядок коду означає, що якщо у формі з іменем tests у групі кнопок з іменем one вибране друге значення (елементи масиву нумеруються від 0), тобто вибрана правильна відповідь, то до значення змінної i додається одиниця.

Вираз document.tests.druk.Value = i означає, що значення змінної i присвоєне об'єкту druk — таку назву має текстова область для виведення інформації в один рядок та два стовпця:

```
<TEXTAREA ROWS="1" COLS="2" NAME="druk"></TEXTAREA>
```

Ця область міститься у формі, якій присвоєно назву tests. Форма розташована у поточному документі document.

Кнопка Кількість правильних відповідей ініціює виконання функції perevirka (this.form) після настання події OnClick (клацання кнопки). Аргумент this.form показує, що обчислення застосовують до поточної форми.

```
<INPUT NAME="check" onclick="perevirka (this.form)"  
TYPE="button" \AБиE="Кількість правильних відповідей">
```

Передбачена ще одна кнопка — Оновити форму — для очищення форми.

1. У редакторі Блокнот введіть код HTML-документа.

```
<HTML>  
<HEAD>  
<TITLE>Тестування</TITLE>  
<BODY>  
<H4>Тестування</H4>  
<SCRIPT>  
function perevirka(tests) {  
var i=0;
```

```

if (tests.one[1].checked) {i++;}
if (tests.two[0].checked) {i++;}
if (tests.three[2].checked) {i++;}
document.tests.druk.value = i;
}
</SCRIPT>
</HEAD>
<FORM METHOD="post" NAME="tests">
2+2=? <BR><BR>
<INPUT TYPE="radio" NAME="one"
VALUE="1">5 <BR>
<INPUT TYPE="radio" NAME="one"
VALUE="2">4 <BR>
<INPUT TYPE="radio" NAME="one"
VALOE="3">3<BR><BR>
3+3=? <BR><BR>
<INPUT TYPE="radio" NAME="two"
VALUE="1">6<BR>
<INPUT TYPE="radio" NAME="two"
VALUE="2">5<BR>
<INPUT TYPE="radio" NAME="two"
VALUE="3">7<BR><BR>
4+4=? <BR><BR>
<INPUT TYPE="radio" NAME="three"
VALUE="1">7<BR>
<INPUT TYPE="radio" NAME="three"
VALUE="2">9<BR>
<INPUT TYPE="radio" NAME="three"
VALUE="3">8<BR><BR>
<INPUT NAME="check" onclick="perevirka (this.form)"
TYPE="button" VALUE="Кількість правильних відповідей">
<TEXTAREA cols="2" NAME="druk" ROWS="1"></TEXTAREA>
<INPUT TYPE="reset" VALUE="ОНОВИТИ форму">
</FORM>
</BODY>
</HTML>

```

2. Збережіть документ у файлі test.html.
3. Відкрийте збережений документ у браузері. У разі потреби розблокуйте активний вміст веб-сторінки (див. рис. 5.16).
4. Заповніть форму, давши відповіді на всі запитання. Клацніть кнопку Кількість правильних відповідей. Після цього веб-сторінка матиме приблизно такий вигляд, як показано на рис. 5.19.



Рис. 5.19. Тестування з використанням форми

Створення форм та динамічних ефектів у FrontPage

Програма FrontPage дає змогу створювати форми, вибираючи команди з меню — знати мову HTML для цього не обов'язково. Але, як зазначалося, створений у FrontPage сайт має бути опублікований на сервері, що підтримує FrontPage Server Extensions. В іншому разі деякі елементи такого сайту працювати не будуть.

Створити форму у FrontPage можна у три способи.

- **За допомогою майстра.** Програма задає низку запитань про те, якою повинна бути форма і, ґрунтуючись на відповідях, будує форму, яку згодом можна відредагувати так, щоб вона повністю відповідала вимогам.
- **Використовуючи шаблон сторінки з формою.** При цьому програма не ставить жодних запитань, а відразу створює форму певного типу, наприклад для гостьової книги. Її також можна редагувати.

- Самостійно зі стандартних елементів. Для цього спочатку створюють сторінку, а потім розміщують на ній форму. FrontPage у формі створить дві кнопки — Отправить (Надіслати) і Сброс (Скинути). Розробнику залишається створити на формі поля та інші елементи.

Шаблони та майстри для створення форм

Щоб переглянути список шаблонів і майстрів форм, які пропонує FrontPage, в меню Файл (Файл) слід вибрати команду Создать (Створити) і в області завдань клацнути посилання Другие шаблоны страниц (Інші шаблони сторінок). У діалоговому вікні, що відкриється, для вибору шаблону або майстра створення форм використовують вкладку Общие (Загальні) (рис. 5.20).

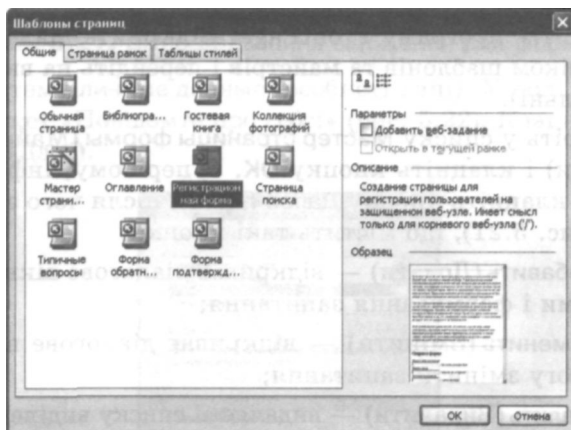


Рис. 5.20. Вибір майстра або шаблону для створення форм

Серед шаблонів та майстрів програми FrontPage варто відзначити такі:

- Форма подтверждения (Форма підтвердження) — шаблон форми для підтвердження отримання інформації;
- Форма обратной связи (Форма зворотного зв'язку) — шаблон форми для введення зауважень стосовно сайту;
- Мастер страницы формы (Майстер сторінки форми) — майстер створення форми для опитування відвідувачів сайту; дає змогу

сформувати список запитань, на які потрібно одержати відповіді;

- Гостевая книга (Гостьова книга) — шаблон гостьової сторінки; ця сторінка дасть змогу відвідувачам сайту залишити свої враження про нього;
- Страница поиска (Сторінка пошуку) — шаблон сторінки у вигляді форми, яку використовують для пошуку інформації на сторінках сайту;
- Регистрационная форма (Реєстраційна форма) — шаблон форми для реєстрації відвідувача сайту.

Вправа 5.7

Розглянемо роботу майстра Мастер страницы формы (Майстер сторінки форми) детально.

1. Запустіть програму FrontPage. Відкрийте діалогове вікно із переліком шаблонів та майстрів і перейдіть на вкладку Общие (Загальні).
2. Виберіть у списку Мастер страницы формы (Майстер сторінки форми) і клацніть кнопку ОК. У першому, інформаційному вікні клацніть кнопку Далее (Далі), після чого з'явиться вікно (рис. 5.21), що містить такі кнопки:
 - Добавить (Додати) — відкриває діалогове вікно для вибору теми і формування запитання;
 - Изменить (Змінити) — відкриває діалогове вікно, що дає змогу змінити запитання;
 - Удалить (Видалити) — видаляє зі списку виділене запитання;
 - Вверх (Вгору) — переміщує виділене у списку запитання на один рівень вгору;
 - Вниз (Донизу) — переміщує виділене у списку запитання на один рівень донизу;
 - Очистить (Очистити) — очищує список, видаляючи з нього всі запитання.
3. Для формування запитання клацніть кнопку Добавить (Додати). У верхній частині діалогового вікна, що відкриється, міститься список тем, відносно яких можна формувати запитання. В області Описание (Опис) є короткий опис полів, які

будуть поміщені у форму за вибраною темою. Запитання слід вводити у поле Измените приглашение для этого вопроса (Змініть запрошення для цього запитання).

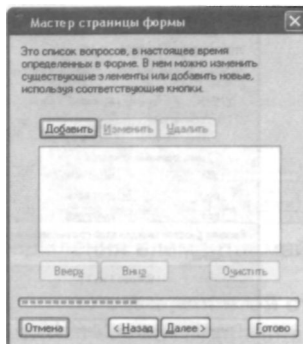


Рис. 5.21. Вікно додавання запитання для форми

4. Виберіть тему личные данные (особисті дані). У полі підказки введіть текст «Повідомте про себе:» (рис. 5.22). Клацніть кнопку Далее (Далі).

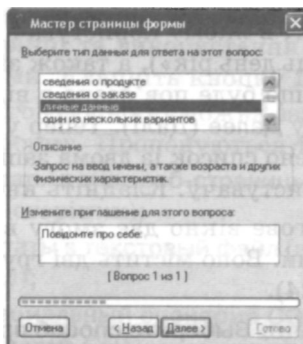


Рис. 5.22. Формування першого запитання на тему «Персональна інформація»

5. У наступному діалоговому вікні виберіть прапорці та перемикачі, що відповідають інформації, яку потрібно отримати від відвідувача веб-сторінки. Встановіть прапорці ФИО (Прізвище, ім'я, по батькові) з властивістю-перемикачем полное имя (повне ім'я), возраст (вік) із властивістю число полных лет

(кількість повних років), пол (стать), цвет волос (колір волосся), цвет глаз (колір очей) (рис. 5.23). Перейдіть до списку запитань за допомогою кнопки Далее (Далі).

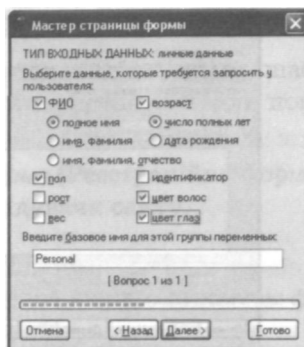


Рис. 5.23. Визначення полів для збирання персональної інформації про відвідувача сайту

- Знову скористайтесь кнопкою **Добавить (Додати)** та виберіть тип інформації **дата (дата)**, у полі підказки наберіть текст «Введіть дату:». Клацніть кнопку **Далее (Далі)**. У наступному вікні визначте формат, в якому користувач повинен вводити дату, `сІсІ.тт.уу` («місяць.день.рік»), а також задайте ім'я змінної (наприклад, `data`), що буде пов'язана з відповіддю користувача. Клацніть кнопку **Далее (Далі)**. Тепер у вікні, зображеному на рис. 5.21, виведено список із двома запитаннями, які заплановано задати користувачу. Клацніть кнопку **Далее (Далі)**.
- Наступне діалогове вікно дає змогу визначити оформлення майбутньої форми. Воно містить дві групи перемикачів та прапорець (рис. 5.24).

Група перемикачів **Выберите способ представления списка вопросов (Виберіть спосіб подання списку запитань)** містить варіанти, що дають змогу визначити спосіб оформлення запитань, розташованих у формі:

- обычные абзацы (звичайні абзаци);
- нумерованный список (нумерований список);
- маркированный список (маркований список);
- список определенных (список визначень).

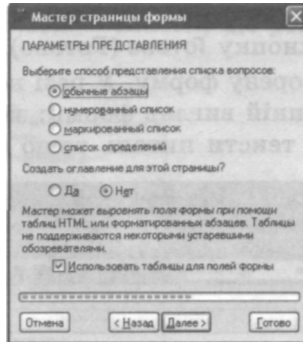


Рис. 5.24. Оформлення елементів майбутньої форми

Перемикач Создать оглавление для этой страницы? (Створити зміст для цієї сторінки?) містить два значення. За умовчанням змісту для сторінки немає (значення Нет). Якщо є потреба його створити, вибирають значення Да. Коли прапорець Использовать таблицы для полей формы (Використовувати таблиці для полів форми) встановлено, вирівнювання полів у формі буде виконано за допомогою таблиць.

8. Виберіть оформлення запитань як звичайних абзаців, відсутність змісту на сторінці та використання таблиць для вирівнювання полів форми. Клацніть кнопку Далее (Далі).
9. У наступному діалоговому вікні можна зазначити, де зберігати дані, введені у форму. Пропонуються такі варіанти:
 - сохранить результаты в виде веб-страницы (зберегти результати у вигляді веб-сторінки);
 - сохранить результаты в текстовый файл (зберегти результати в текстовий файл);
 - использовать специальный сценарий CGI (використовувати спеціальний сценарій CGI).

У разі вибору першого або другого варіанта необхідно ввести назву веб-сторінки або текстового файлу в поле Введите базовое имя для файлов результатов (Введіть базове ім'я для файлів результатів).

10. Виберіть збереження результатів у вигляді веб-сторінки і задайте для файлу результатів назву results. Клацніть кнопку Далее (Далі).

11. Майстер створення форм отримав відповіді на всі запитання. Клацніть кнопку Готово (Готово).
12. Розглянемо створену форму. У разі необхідності можна скорегувати зовнішній вигляд форми: змінити шрифти, кольори, стандартні тексти питань тощо (рис. 5.25).

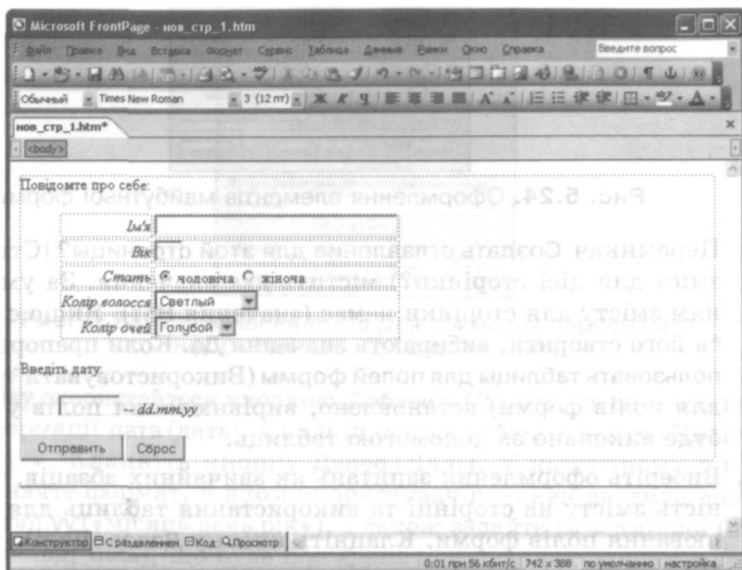


Рис. 5.25. Вигляд форми після внесення коректив

Елементи форм

Програма FrontPage дає змогу самостійно формувати та редагувати форми, виконуючи певні дії з їх окремими елементами, які вставляють на веб-сторінку за допомогою команди Вставка • Форма (Вставка • Форма) (рис. 5.26). При цьому можна вибрати, зокрема, такі елементи форми:

- Поле — однорядкове поле для введення текстових даних;
- Текстовое поле — область із можливістю прокручування для введення текстів великого обсягу;
- Флажок — прапорець;
- Переключатель — перемикач;

- Раскрывающийся список — список, що розкривається;
- Кнопка — кнопка.

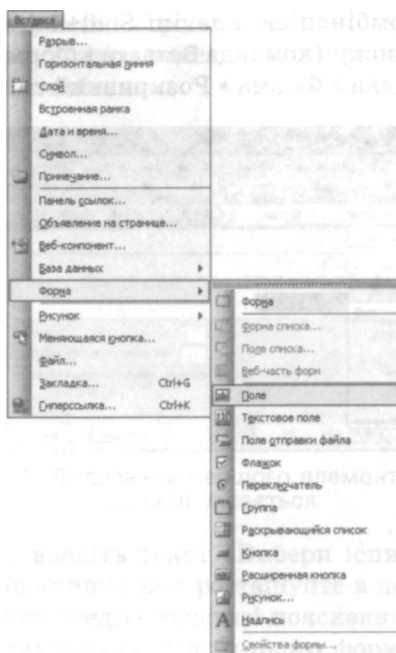


Рис. 5.26. Вставлення форми

Вправа 5.8

Розглянемо процес формування веб-сторінки, призначеної для заповнення форми для складання іспитів.

1. Відкрийте редактор FrontPage. Створіть сторінку за допомогою кнопки **Создание новой обычной страницы** (Створення нової звичайної сторінки) панелі інструментів **Стандартная** (Стандартна).
2. Введіть слова «Заповни форму для складання іспитів» та зробіть цей текст заголовком четвертого рівня (рис. 5.27).
3. У новому абзаці введіть текст «Прізвище:» та додайте поле для введення текстових даних (команда **Вставка • Форма • Поле** (Вставка • Форма • Поле)). Клавішею **Enter** перенесіть кнопки **Отправить** (Надіслати) і **Сброс** (Скинути) у новий абзац.

- Після однорядкового поля перейдіть на новий абзац і введіть текст «З цього списку вибери район, де ти проживаєш!», а потім у новий рядок того самого абзацу (перехід на нього здійснюється комбінацією клавіш Shift+Enter) вставте поле розкривного списку (команда Вставка • Форма • Розкриваючийся список (Вставка • Форма • Розкривний список)).

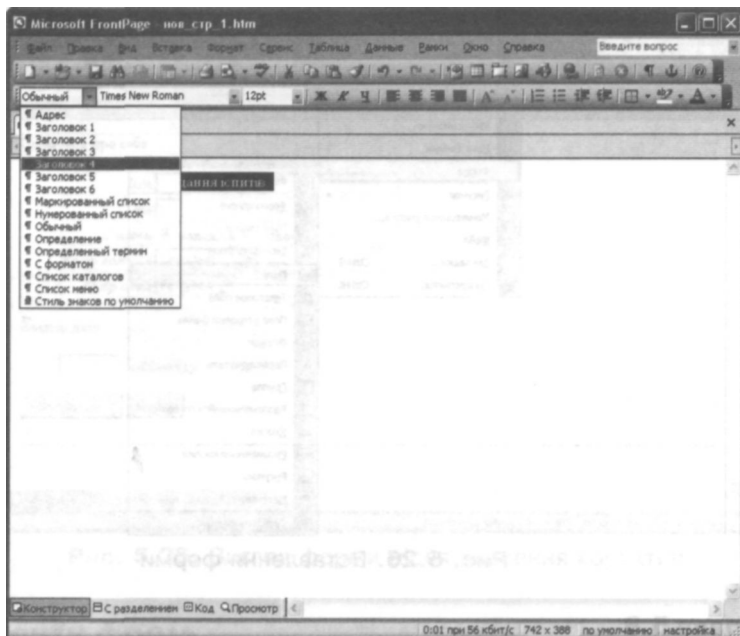


Рис. 5.27. Створення заголовка форми

- Виділіть вставлений елемент і виконайте команду Свойства поля формы (Властивості поля форми) з його контекстного меню. У діалоговому вікні, що відкриється, клацніть кнопку Добавить (Додати), у новому діалоговому вікні в полі Вариант (Варіант) введіть значення першого елемента розкривного списку — «Франківський» — та клацніть ОК. Це значення з'явиться у переліку елементів списку (рис. 5.28).
- Аналогічно додайте у перелік елементів списку пункти «Сихівський» та «Галицький» і клацніть ОК. Розміри поля списку зміняться, і в ньому відобразиться перший елемент.

- З нового абзацу введіть текст «Вибери мову, якою ти складатимеш іспит:» та додайте три перемикачі (за допомогою команди Вставка • Форма • Переключатель (Вставка • Форма • Перемикач)). Всі перемикачі розмістіть в окремих рядках абзацу, поруч із ними введіть текстові пояснення: «Українська», «Російська», «Англійська».

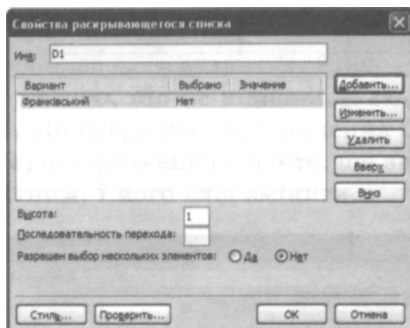


Рис. 5.28. Додавання першого елемента списку, що розкривається

- З нового абзацу введіть текст «Вибери іспити:» та додайте три прапорці. Кожний із них розташуйте в новому рядку, поруч із прапорцями введіть текстові пояснення: «Українська», «Фізика», «Математика». Після цього форма набуде такого вигляду, як на рис. 5.29.
- На цьому етапі всі необхідні об'єкти розміщені на формі. Тепер відкрийте діалогове вікно властивостей форми. Для цього помістіть курсор всередину форми і виберіть команду Вставка • Форма • Свойства формы (Вставка • Форма • Властивості форми) головного меню або однойменну команду контекстного меню. За умовчанням у цьому діалоговому вікні вибрано значення Отправка (Надсилання) перемикача Сохранение результатов (Збереження результатів), а в полі Имя файла (Ім'я файлу) зазначено текстовий файл, в який після клацання кнопки Отправить (Надіслати) будуть записані введені у форму дані. Стандартний формат цих даних (текст, розділений комами) можна змінити у діалоговому вікні, що відкривається кнопкою Параметры (Параметри). Варто, однак, нагадати, що для правильної роботи цієї форми сторінка, яка її містить, має

бути розміщена на сервері, що підтримує FrontPage Server-Extensions.

10. Збережіть документ і перегляньте його у вікні браузера.

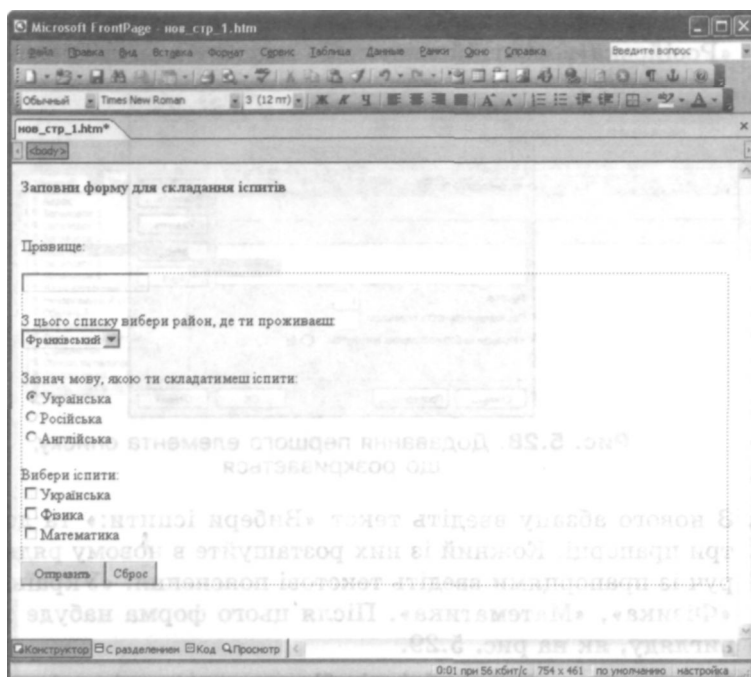


Рис. 5.29. Побудова форми

Перевірка введеної у форму інформації

Програма FrontPage надає можливість перевіряти інформацію, яку вводить користувач у форму, на коректність і відповідність наперед визначеним умовам. Наприклад, можна задати мінімальну або максимальну кількість символів у текстовому полі, зазначити, що відповідь на питання є обов'язковою, тощо.

Щоб визначити правила перевірки значення, введеного в елементі форми, необхідно клацнути його правою кнопкою миші, вибрати у контекстному меню пункт Свойства поля формы (Властивості поля форми) і у діалоговому вікні, що відкриється, натиснути кнопку Проверить (Перевірити).

Розглянемо діалогове вікно, призначене для перевірки текстового поля (рис. 5.30). У ньому можна задати необхідність перевірки типу даних, які вводить користувач, правила їх форматування, мінімальну та максимальну довжину рядка, діапазон допустимих значень.

Якщо для поля задано правила перевірки, то бажано розмістити на сторінці інструкції для користувачів щодо допустимих для цього поля даних, наприклад, повідомити, що текстове поле призначене для введення числового значення. Якщо користувач введе у поле форми інформацію, що не відповідає заданим для цього елемента правилам, він одержить повідомлення **Ошибка проверки (Помилка перевірки)** з інформацією про те, що значення, введене у форму, є некоректним, і його слід змінити.

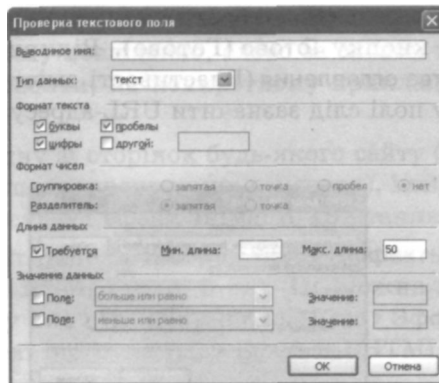


Рис. 5.30. Перевірка текстового поля форми

Ефекти динамічної HTML у FrontPage

У програмі FrontPage є додаткові можливості вдосконалити сайт за допомогою спеціальних компонентів — готових до застосування програмних модулів, для використання яких достатньо настроїти їх властивості. У такий спосіб легше розширити функціональні можливості сайту та прискорити процес його розроблення.

Розглянемо три компоненти, які дозволяють створювати зміст сайту, анімацію тексту за допомогою динамічної HTML та ефект переходу між сторінками сайту.

Створення змісту сайту

Основною перевагою змісту є те, що у разі додавання до сайту нової сторінки зміст змінюється автоматично, на відміну від стандартного меню з гіперпосиланнями, яке в цьому випадку доводиться модифікувати вручну.

Для створення змісту сайту необхідно виконати такі дії.

1. Відкрити у програмі FrontPage наявний сайт (скористайтесь одним із сайтів, розроблених раніше на практичних заняттях) і створити нову сторінку за допомогою кнопки Создание новой обычной страницы (Створення нової звичайної сторінки) панелі інструментів Стандартная (Стандартна).
2. У меню Вставка (Вставка) вибрати команду Веб-компонент (Веб-компонент), у вікні, що з'явиться, у списку Тип компонента (Тип компонента) вибрати значення Оглавление (Зміст) і натиснути кнопку Готово (Готово). Відкриється діалогове вікно Свойства оглавления (Властивості змісту) (рис. 5.31), де у текстовому полі слід зазначити URL-адресу головної сторінки сайту.

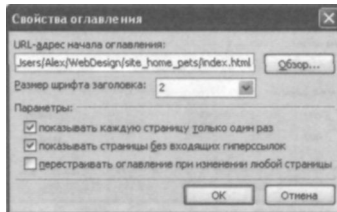


Рис. 5.31. Діалогове вікно для настроювання властивостей змісту

3. Встановити прапорець перестраивать оглавление при изменении любой страницы (переформувувати зміст після змінення будь-якої сторінки) і клацнути кнопку ОК.
4. Клацнути правою кнопкою миші у вільному місці сторінки та у контекстному меню вибрати пункт Свойства страницы (Властивості сторінки). Відкриється відповідне діалогове вікно. У текстове поле Название (Назва) ввести заголовок новоствореної веб-сторінки — слово «Зміст».
5. Зберегти сторінку у файлі з іменем zmist.htm.

6. Відкрити сторінку змісту у вікні браузера; це можна зробити за допомогою команди **Файл • Просмотреть в обозревателе (Файл • Переглянути у браузері)**. У вікні відобразиться перелік усіх сторінок сайту у вигляді гіперпосилань. Відвідувач, клацнувши одне з них, потрапляє на відповідну сторінку. Зміст можна створити ще в один спосіб — використовуючи шаблон сторінки **Оглавление (Зміст)**. Для цього за допомогою команди **Файл • Создать (Файл • Створити)** і посилання **Другие шаблоны страниц (Інші шаблони сторінок)** слід відкрити діалогове вікно з переліком шаблонів та майстрів веб-сторінок і на вкладці **Общие (Загальні)** вибрати шаблон **Оглавление (Зміст)**.

Ефекти анімації тексту

Під час створення ефекту анімації тексту можна вибрати один з кількох її типів: рух по спіралі, виліт із різних точок екрана, виникнення тексту послідовно, слово за словом тощо. Покажемо, як можна це зробити, на конкретному прикладі. Отже, слід виконати такі дії.

1. Відкрити одну зі сторінок будь-якого сайту (наприклад, створеного раніше на практичних заняттях), і виділити текст, для якого застосувати ефект анімації «Влітання».
2. Відобразити панель інструментів **Эффекты DHTML (Ефекти динамічної HTML)** (рис. 5.32). Це можна зробити, вибравши команду **Вид • Панели инструментов • Эффекты DHTML (Вигляд • Панелі інструментів • Ефекти DHTML)**.



Рис. 5.32. Панель інструментів **Эффекты DHTML**

3. У списку **Вкл (Увімкнути)** цієї панелі вибрати пункт **Загрузка страницы (Завантаження сторінки)**. Це означає, що ефект динамічної HTML проявлятиметься тоді, коли користувач завантажить сторінку у вікно браузера.
4. У списку **Применить (Застосувати)** вибрати ефект **Влет (Влітання)**.
5. Виділити якийсь фрагмент тексту — до нього застосуємо ефект «збільшення розміру та зміна кольору».

- У списку Вкл (Увімкнути) панелі інструментів Эфффекты DHTML (Ефекти DHTML) вибрати пункт Наведение мыши (Наведення миші) — ефект динамічного HTML відтворюватиметься тоді, коли користувач наведе вказівник миші на відповідний текст.
- У описку Применить (Застосувати) вибрати елемент Форматирование (Форматування), а правіше у списку Эфффект (Ефект) — елемент Выбрать шрифт (Вибрати шрифт). Відкриється діалогове вікно Шрифт (Шрифт), де слід задати відмінний від поточного розмір шрифту та інший колір символів.
- Зберегти результати у файлі з іншим іменем та перевірити роботу сторінки в браузері.

Спеціальні ефекти зміни сторінок

Ефекти переходів між сторінками визначають, як на екрані зникає попередня сторінка і з'являється нова. Вони можуть базуватися на круговому чи лінійному русі сторінок, їх розчиненні або проявленні тощо. Щоб задати ці ефекти, слід виконати таку послідовність дій.

- Відкрити головну сторінку будь-якого сайту, створеного раніше на практичних заняттях.
- У меню Формат (Формат) вибрати команду Смена страниц (Зміна сторінок). Відкриється відповідне діалогове вікно (рис. 5.33).

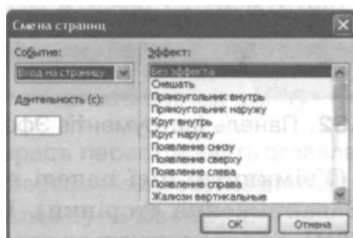


Рис. 5.33. Діалогове вікно, в якому визначається перехід між сторінками

- У списку Событие (Подія) вибрати подію Выход со страницы (Вихід зі сторінки) — ефект виникатиме у разі виходу зі сторінки. У текстовому полі Длительность (с) (Тривалість у секун-

дах) ввести 3 — ефект переходу триватиме 3 секунди. У списку Ефект (Ефект) вибрати відповідне значення, наприклад Появление снизу (Поява знизу).

4. Зберегти результати роботи у файлі з іншим іменем і переглянути роботу сторінки у браузері.

Вправа 5.9

Іноді для економії місця на веб-сторінці або з метою компактного і наочного подання інформації використовують списки, які можна розгортати та згорнути. Розглянемо процедуру формування таких списків у програмі FrontPage.

Створимо списки перших страв, других страв та десертів. Назви категорій страв відображатимуться на веб-сторінці постійно, а назви страв можна буде побачити, клацнувши мишею категорію. Щоб згорнути список страв, необхідно буде знову клацнути категорію. Для елементів списків використовуйте звичайний текст.

1. Створіть у програмі FrontPage нову сторінку.
2. Введіть у стовпчик категорії страв і відповідні страви у них (наприклад, ті, що зображені на рис. 5.34). На цьому етапі слід лише набрати всі назви, кожен в окремому абзаці.

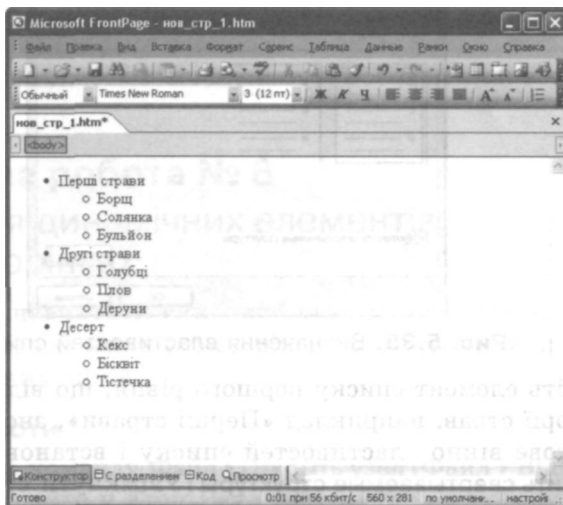


Рис. 5.34. Двохрівневий список

3. Виділіть весь текст і оформіть його як маркований список за допомогою кнопки Маркеры (Маркери) панелі інструментів Форматирование (Форматування).
4. Виділіть страви, які стосуються категорії «Перші страви», та двічі клацніть кнопку Увеличить отступ (Збільшити відступ) тієї самої панелі інструментів. У такий спосіб буде задано відступ для виділеного фрагмента списку.
5. Аналогічні дії виконайте для інших категорій страв. У результаті ви отримаєте дворівневий список, де категорії страв є першим рівнем, а власне страви — другим (рис. 5.34).
6. Виділіть один з елементів списку другого рівня та виберіть команду Список (Список) у меню Формат (Формат) або команду Свойства списка (Властивості списку) в контекстному меню елемента. У діалоговому вікні властивостей списку встановіть прапорець Включить свертываемые структуры (Увімкнути структури, що згортаються) (рис. 5.35) і клацніть ОК.

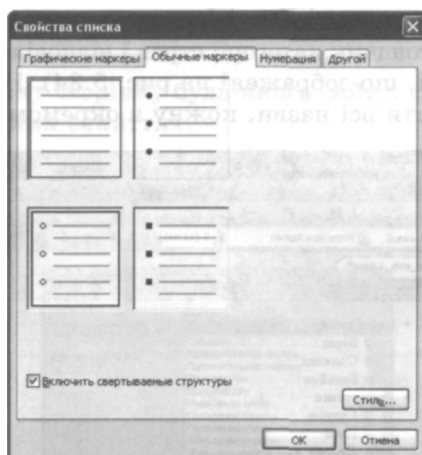


Рис. 5.35. Визначення властивостей списку

Виділіть елемент списку першого рівня, що відповідає назві категорії страв, наприклад «Перші страви», знову відкрийте діалогове вікно властивостей списку і встановіть прапорці Включить свертываемые структуры (Увімкнути структури, що згортаються) та Изначально свернуть (Спочатку згорнути). Клацніть ОК.

Перейдіть в режим попереднього перегляду сторінки, вибравши вкладку Просмотр (Перегляд) у нижній частині робочої області. На екрані відобразяться лише елементи верхнього рівня дворівневого списку, тобто назви категорій страв. Після клацання мишею кожної з них список розкриється, показуючи елементи другого рівня обраної категорії, тобто самі страви (рис. 5.36). Після повторного клацання список згортається.

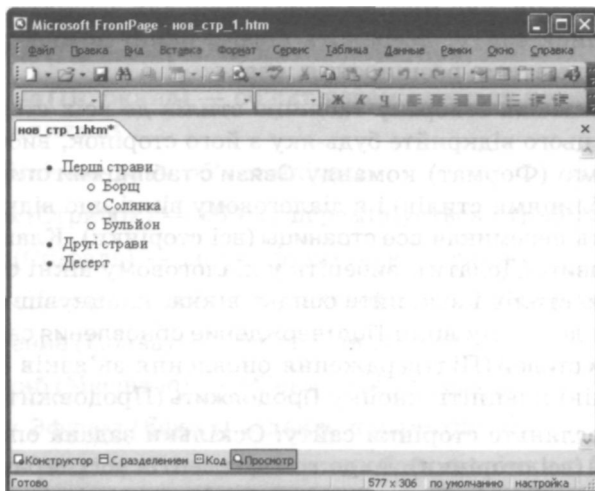


Рис. 5.36. Список зі згорнутими та розгорнутими елементами

Практична робота № 8

Створення динамічних елементів на веб-сторінках

Мета роботи: навчитися використовувати можливості програми FrontPage з оформлення веб-сторінок та вставлення динамічних елементів різних типів.

Порядок роботи

1. Вибравши команду Файл • Открыть узел (Файл • Відкрити вузол), відкрийте у FrontPage сайт goslyny, створений у практичній роботі № 7.

На панелі інструментів Стандартная (Стандартна) клацніть стрілку праворуч від кнопки Создание новой обычной страницы (Створення нової звичайної сторінки), виберіть у меню команду Страница (Сторінка), в діалоговому вікні, що відкриється, перейдіть на вкладку Таблицы стилей (Таблиці стилів) і виберіть на ній один із шаблонів таблиць стилів. Після клацання кнопки ОК буде створено новий документ — таблицю каскадних стилів, яка містить опис стилів для форматування вмісту сторінок сайту. Збережіть його у папці поточного веб-сайту (файл матиме розширення .css).

Підключіть створену таблицю стилів до всіх сторінок сайту. Для цього відкрийте будь-яку з його сторінок, виберіть у меню Формат (Формат) команду Связи с таблицями стилей (Зв'язки з таблицями стилів) і в діалоговому вікні, що відкриється, виберіть перемикач все страницы (всі сторінки). Клацніть кнопку Добавить (Додати), виберіть у діалоговому вікні створену таблицю стилів і закрийте обидва вікна, клацнувши кнопку ОК. У діалоговому вікні Подтверждение обновления связей с таблицями стилей (Підтвердження оновлення зв'язків з таблицями стилів) клацніть кнопку Продолжить (Продовжити).

Перегляньте сторінки сайту. Оскільки задана опція все страницы (всі сторінки), то до всіх елементів усіх веб-сторінок буде застосовано форматування відповідно до стилів таблиці. Відкрийте будь-яку сторінку сайту в режимі перегляду HTML-коду. Уважно переглянувши код, ви побачите рядок із тегом <LINK>, в якому зазначена назва таблиці каскадних стилів. Відкрийте сторінку index.htm і виділіть заголовок або інший текст для застосування ефекту анімації. За допомогою команди Вид • Панели инструментов • Эффекты DHTML (Вигляд • Панелі інструментів • Ефекти DHTML) відобразить панель інструментів Эффекты DHTML (Ефекти DHTML) (див. рис. 5.32).

У списку Вкл (Увімкнути) цієї панелі виберіть подію, після настання якої буде запущено ефект анімації:

- Щелчок (Клацання) — клацання мишею на виділеному фрагменті;
- Двойной щелчок (Подвійне клацання) — подвійне клацання мишею;

- Наведение мыши (Наведення миші) — наведення вказівника миші на виділений фрагмент;
 - Загрузка страницы (Завантаження сторінки) — завантаження веб-сторінки.
7. У списку Применить (Застосувати) задайте анімаційну дію, що відбуватиметься після настання вибраної події. Зокрема, для події Загрузка страницы (Завантаження сторінки) це може бути:
- Падение (Падіння) — об'єкт падає згори;
 - Пружина (Пружина) — об'єкт переміщується з невеликими коливаннями;
 - Влет (Влітання) — об'єкт влітає;
 - Скачок (Стрибок) — об'єкт переміщується стрибками;
 - Виток (Спіраль) — об'єкт рухається по спіралі;
 - Волна (Хвиля) — об'єкт рухається коливними рухами;
 - Появление (Поява) — об'єкт з'являється;
 - Масштаб (Масштаб) — об'єкт змінює розміри.

У списку Эффект (Ефект) задайте параметри дії.

8. Збережіть результати та перевірте роботу сторінки у браузері, вибравши команду Файл • Просмотреть в обозревателе (Файл • Переглянути у браузері).
9. Відкрийте сторінку index.htm. Виконайте команду Формат • Смена страниц (Формат • Зміна сторінок). У діалоговому вікні, що відкриється (див. рис. 5.33), виберіть подію, з настанням якої виконуватимуться анімаційні ефекти переходу між сторінками:
- Вход на страницу (Вхід на сторінку) — під час відкривання сторінки;
 - Выход со страницы (Вихід зі сторінки) — під час виходу зі сторінки;
 - Вход на узел (Вхід на вузол) — під час входу на сайт;
 - Выход с узла (Вихід із вузла) — під час переходу з одного сайту на інший.

10. У полі Длительность (Тривалість) задайте тривалість створюваного ефекту (в секундах). Для вибору ефекту переходу використайте список Эффект (Ефект). Збережіть результати і перевірте роботу сторінки у браузері.
11. На сайті є списки, що мають кілька рівнів. Перетворіть їх на списки, які згортаються та розгортаються. Збережіть результати і перевірте роботу сторінки у браузері.
12. Створіть нову веб-сторінку. Виберіть команду Вставка • Веб-компонент (Вставка • Веб-компонент), у вікні, що з'явиться, у списку Тип компонента (Тип компонента) виберіть значення Оглавление (Зміст) і натисніть кнопку Готово (Готово). У діалоговому вікні, що відкриється (рис. 5.31), зазначте сторінку, що є стартовою для сайту, та встановіть потрібні прапорці. За допомогою команди Свойства страницы (Властивості сторінки) контекстного меню сторінки відкрийте відповідне діалогове вікно і у поле Название (Назва) введіть слово «Зміст». Збережіть сторінку у файлі zmist.htm.
13. Перевірте роботу зі змістом у вікні браузера.
14. Відкрийте будь-яку сторінку сайту і виділіть текст, який потрібно зробити рухомих рядком. У меню Вставка (Вставка) виберіть команду Веб-компонент (Веб-компонент), потім тип компонента Динамические эффекты (Динамічні ефекти) і ефект — Бегущая строка (Рухомий рядок). Після клацання кнопки Готово (Готово) відкриється діалогове вікно, яке призначене для налаштування його параметрів (рис. 5.37).

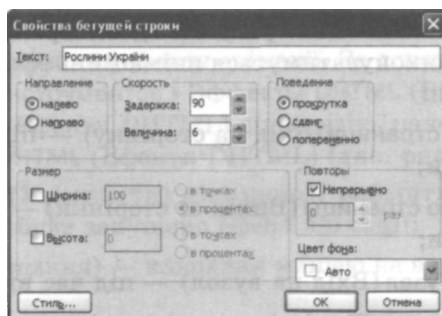


Рис. 5.37. Діалогове вікно властивостей рухомого рядка

15. Використовуючи перемикач Направление (Напрямок), задайте напрямок руху рядка: наліво (ліворуч) або направо (праворуч).
16. Область Скорость (Швидкість) містить два лічильники, що керують швидкістю переміщення рядка:
 - Задержка (Затримка) — задає проміжок часу між двома послідовними переміщеннями рядка (у мілісекундах);
 - Величина (Величина) — визначає відстань між двома сусідніми позиціями рядка (у пікселях).

Варіюючи значення цих параметрів, можна змінювати швидкість руху тексту. Змініть встановлені за умовчанням параметри, але не набагато.

За допомогою параметрів в області Поведение (Поведінка) можна задати спосіб руху рядка:

- прокрутка (прокручування) — рядок з'являється від межі прямокутної області та зникає за протилежною, напрямком переміщення тексту задають значеннями перемикача Направление (Напрямок);
 - сдвиг (зсув) — рядок з'являється від межі прямокутної області та зупиняється перед протилежною, напрямком переміщення тексту задають значеннями перемикача Направление (Напрямок);
 - попеременно (почергово) — рядок рухається від однієї межі прямокутної області до іншої, змінюючи потім напрямок.
17. Виберіть один із трьох запропонованих варіантів.
 18. Використовуючи параметри в області Размер (Розмір), задайте розміри прямокутника, у межах якого переміщатиметься рядок. Ця область містить прапорці Ширина (Ширина) та Высота (Висота), після встановлення яких стають доступними поля, що дають змогу задати ширину та висоту прямокутника в пікселях або у відсотках від розміру екрана.
 19. В області Повторы (Повторення) задайте кількість переміщень рухомого рядка. У разі встановлення прапорця Непрерывно (Безперервно) рядок переміщуватиметься по вебсторінці безупинно.

20. Для визначення кольору тла прямокутної області, в якій переміщується рядок, клацніть кнопку розкритого списку Цвет фона (Колір тла) і виберіть із палітри, що відкриється, потрібний колір.
21. Клацніть кнопку Стиль (Стиль). На екрані з'явиться діалогове вікно Изменение стиля (Змінення стилю), в якому за допомогою меню кнопки Формат (Формат) можна змінювати оформлення рухомого рядка — параметри шрифту (команда Шрифт (Шрифт)), абзацу (Абзац (Абзац)), меж (Граница (Межа)) тощо. Задайте ці параметри на власний розсуд.
22. Завершивши встановлення всіх параметрів рухомого рядка, закрийте діалогове вікно Свойства бегущей отроки (Властивості рухомого рядка).
23. Перегляньте результати своєї роботи, перешовши на вкладку Просмотр (Перегляд).

Практична робота № 9

Обробка форм за допомогою скриптів

Мета роботи: навчитися створювати форми з текстовими полями, перемикачами і прапорцями, вставляти кнопки надсилання форми та очищення її полів, налаштовувати надсилання форми на адресу електронної пошти.

Порядок роботи

Створимо електронний щоденник учня, який даватиме змогу надсилати дані на електронну адресу, наприклад, батьків. Таким чином, вони зможуть оперативно отримувати відомості про успішність дитини, перебуваючи на роботі або у відрядженні. Учень, класний керівник або лаборант можуть відіслати дані форми відразу після останнього уроку ще зі школи.

Головна сторінка має містити прізвище учня та список навчальних днів тижня, які є гіперпосиланнями на відповідні сторінки сайту. Кожна з них містить форму, що включає такі елементи:

- текстове поле для введення дати у довільному форматі;
- група перемикачів Поведінка, які мають значення Незадовільна, Задовільна, Добра, Зразкова;

- списки, що розкриваються, кількість яких відповідає кількості предметів у цей день тижня; імена списків відповідають назві навчального предмета, значення їхніх елементів — Немає оцінки, Відмінно, Добре, Задовільно, Незадовільно; можна ввести не словесну, а кількісну оцінку за дванадцятибальною системою (тоді у кожному списку буде тринадцять значень);
- група прапорців Додатково, за допомогою яких можна показати додаткові заняття учня у цей день; прапорці мають такі значення: Чергування, Відвідування спортивної секції, Відвідування факультативу, Участь у спортивних змаганнях, Участь у предметній олімпіаді;
- кнопки Відіслати та Очистити.

Розглянемо схему такого сайту-щоденника (рис. 5.38).

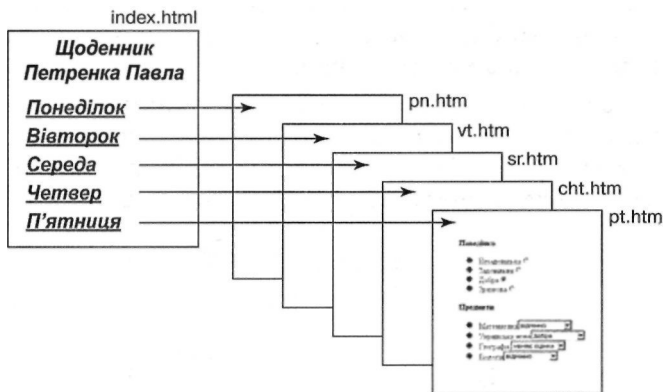


Рис. 5.38. Схема сайтущоденника

Головна сторінка має назву index.htm, а сторінки, що відповідають дням тижня, — pn.htm (понеділок), vt.htm (вівторок), sr.htm (середа), cht.htm (четвер), pt.htm (п'ятниця).

1. Відкрийте FrontPage. Створіть новий сайт за допомогою команди Файл • Создать (Файл • Створити) і посилання Одностраничний веб-узел (Однострінковий веб-вузол) області завдань Создание (Створення).
2. Виберіть за допомогою команди Тема (Тема) меню Формат (Формат) тему для всіх сторінок сайту; для цього потрібно виконати команду Применить как тему по умолчанию (Застосувати

як тему за умовчанням) контекстного меню вибраної в області завдань Тема (Тема) теми. Для сайту на шкільну тематику підійде тема Копирка (Копірка), згідно з якою сторінки сайту нагадуватимуть сторінки зошита.

3. Створіть сторінки, що відповідають дням тижня. Детально пояснимо цей процес для понеділка (файл pn.htm). Решту сторінок, що відповідають іншим дням тижня, створіть в аналогічний спосіб.
4. Введіть слово **Понеділок** і призначте його заголовком, наприклад, третього рівня.
5. У меню **Вставка (Вставка)** виберіть команду **Форма • Поле (Форма • Поле)**, після чого однорядкове текстове поле буде вставлено на сторінку. Введіть перед ним слово «Дата» і оформіть цей текст як заголовок, наприклад, четвертого рівня. Під час заповнення форми користувач вводитиме у це поле дату в довільному форматі (див. рис. 5.44).
6. Виділіть текстове поле і в його контекстному меню виберіть команду **Свойства поля формы (Властивості поля форми)**. Задайте для нього ім'я **Сьогодні** (рис. 5.39), яке відобразиться в електронному листі до батьків. Клацніть **ОК**.

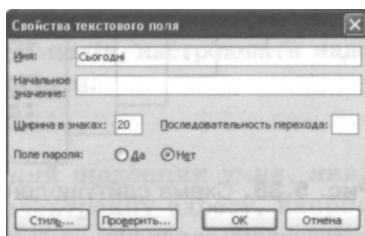


Рис. 5.39. Задавання для текстового поля імені **Сьогодні**

У новому абзаці введіть слово «Поведінка», зробіть його також заголовком четвертого рівня. Нижче створіть список з елементами **Незадовільна**, **Задовільна**, **Добра**, **Зразкова**. Для цього введіть зазначені слова, кожне в окремий абзац, виділіть їх та скористайтеся кнопкою **Маркеры (Маркери)** панелі інструментів **Форматирование (Форматування)**. Праворуч від кожного елемента списку вставте перемикач. Для цього в меню

Вставка (Вставка) виберіть команду Форма • Переключатель (Форма • Перемикач). Приблизний вигляд списку показано на рис. 5.42.

Виділяючи послідовно кожний перемикач, у контекстному меню вибирайте команду Свойства поля формы (Властивості поля форми). Призначте для всіх перемикачів одне групове ім'я (поле Имя группы (Ім'я групи)) «Поведінка», а значенням кожного (поле Значение (Значення)) буде відповідна словесна характеристика поведінки зі списку: Незадовільна, Задовільна, Добра, Зразкова (рис. 5.40). Початковий стан Выбран (Вибраний) задамо лише для першого перемикача. Ім'я групи та значення обраного перемикача відобразатимуться пізніше в електронному листі до батьків.

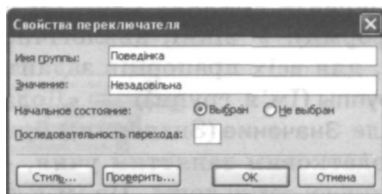


Рис. 5.40. Призначення властивостей для першого перемикача

У новому абзаці введіть слово «Предмети», зробіть його заголовком четвертого рівня. У спосіб, описаний у п. 7, створіть список, що його елементами є назви предметів, які учень вивчає в понеділок, наприклад Математика, Українська мова. Географія, Біологія (рис. 5.42). Праворуч від кожного елемента вставте поле розкривного списку. Скористайтеся для цього командою Форма > Раскрывающийся список (Форма • Розкривний список) меню Вставка (Вставка).

10. До кожного розкривного списку додайте назви елементів, які є словесними характеристиками оцінки учня — Немає оцінки, Відмінно, Добре, Задовільно, Незадовільно. Зробіть це за допомогою діалогового вікна властивостей елемента (див. рис. 5.28), що відкривається командою контекстного меню Свойства поля формы (Властивості поля форми). Кожному розкривному списку слід призначити окреме ім'я, що має збігатися з назвою предмета. Ці імена та значення обраного

елемента відобразатимуться пізніше в електронному листі до батьків.

Отже, для різних списків призначені різні імена, але значення елементів кожного списку однакові, оскільки система оцінювання з усіх предметів однакова.

11. У новому абзаці введіть слова «Додаткові заняття», зробіть їх заголовком четвертого рівня. Створіть список з елементами Чергування, Відвідування спортивної секції, Відвідування факультативу, Участь у спортивних змаганнях, Участь у предметній олімпіаді. Праворуч від кожного елемента списку вставте прапорець, вибравши команду Вставка • Форма (Вставка • Форма) та відповідний елемент керування.
12. Виділяючи по чергово кожне поле-прапорець, у контекстному меню вибирайте команду Свойства поля форми (Властивості поля форми). У вікні, аналогічному до зображеного на рис. 5.40, для всіх прапорців задайте одне групове ім'я (поле Имя группы (Ім'я групи)) — «Додатково». Значенням кожного (поле Значение (Значення)) будуть тексти, що відповідають додатковим заняттям учня, — Чергування, Відвідування спортивної секції тощо. Це ім'я та значення встановлених прапорців відобразатимуться в електронному листі для батьків.
13. Виділивши кнопку Отправить (Відіслати), у контекстному меню виберіть команду Свойства поля формы (Властивості поля форми) і в діалоговому вікні, що відкриється, призначте для цієї кнопки інші ім'я та значення (рис. 5.41).

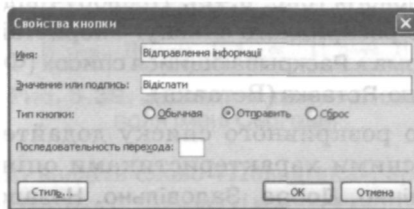


Рис. 5.41. Визначення властивостей кнопки Відіслати

Аналогічно перейменуйте кнопку Сброс на Очистити.

14. Збережіть сторінку у файлі з іменем pn.htm. Вона матиме приблизно такий вигляд, як на рис. 5.42.

У аналогічний спосіб створіть сторінки для інших днів тижня — vt.htm (вівторок), sr.htm (середа), cht.htm (четвер), pt.htm (п'ятниця).

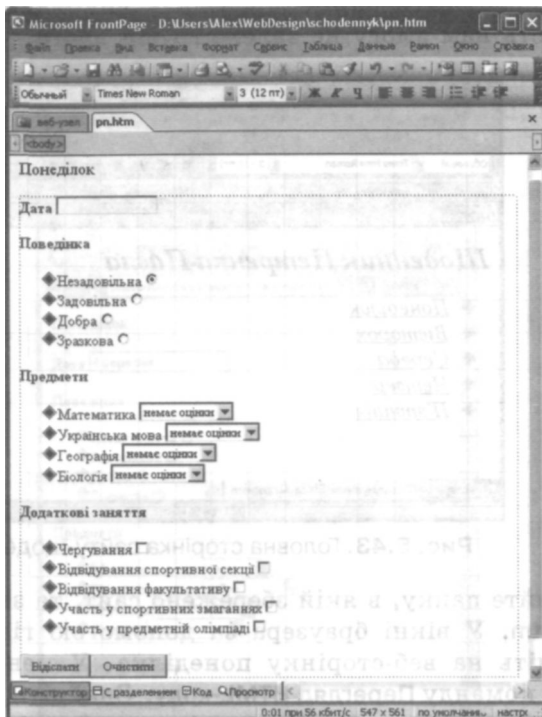


Рис. 5.42. Створення форми сторінки щоденника на понеділок

15. Створіть головну сторінку сайту — index.htm. Введіть прізвище та ім'я власника щоденника (можете додати клас та номер школи). Ця інформація не відобразиться в електронному листі для батьків.
16. Створіть список з елементами Понеділок, Вівторок, Середа, Четвер, П'ятниця, які будуть гіперпосиланнями на відповідні сторінки сайту. Для цього виділіть елемент списку, в меню Вставка (Вставка) виберіть команду Гіперссылка (Гіперпосилання) та в діалоговому вікні настройте зв'язок з відповідним файлом сайту. Збережіть сторінку. Вона матиме приблизно такий вигляд, як на рис. 5.43.

17. Закрийте програму FrontPage. Створений сайт має бути опублікований на сервері з підтримкою FrontPage Server Extensions; якщо цей пакет не інстальовано, деякі елементи сайту працювати не будуть. Однак виконання подальших дій дасть нам змогу не залежати від цього.

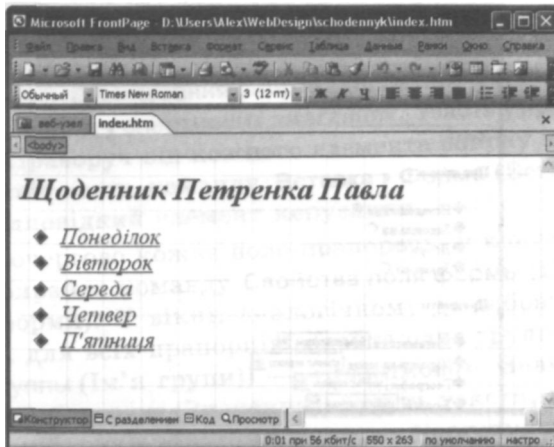


Рис. 5.43. Головна сторінка сайту-щоденника

18. Відкрийте папку, в якій збережено сайт, та запустіть файл index.htm. У вікні браузера за допомогою гіперпосилання перейдіть на веб-сторінку понеділка. У меню Вигляд виберіть команду Перегляд HTML-коду. Знайдіть у HTML-кодї фрагмент, що відповідає тегу відкривання форми; він має приблизно такий вигляд:

```
<FORM METHOD="POST" ACTION="—WEBBOT-SELF—">
<!-- webbot bot="SaveResults"
U-Файл="Файл: ///C: /Documents and Settings/v_pas/
My Documents/My Webs/myweb12/_private/F0RM_results.txt"
S-Формат="TEXT/CSV" S-Label-Fields="TRUE" -->
```

19. Замініть цю частину коду таким тегом відкривання форми:

```
<FORM METHOD="post" ACTION="mailto: ваша електронна
адреса" ENCTYPE="text/plain">
```

Метод post передає дані за заданою в атрибуті ACTION адресою електронної пошти (замість слів «ваша електронна адре-

- са» має бути записана реальна адреса електронної скриньки) у вигляді тексту (це визначає значення атрибута ENCTYPE).
20. Збережіть зміни. Виконайте таку заміну для кожної сторінки з формою. Збережіть виконані зміни та оновіть зображення кожної сторінки.
 21. Перейдіть на сторінку, що відповідає будь-якому дню тижня, та заповніть поля форми (рис. 5.44). Відішліть її, клацнувши кнопку Відіслати.

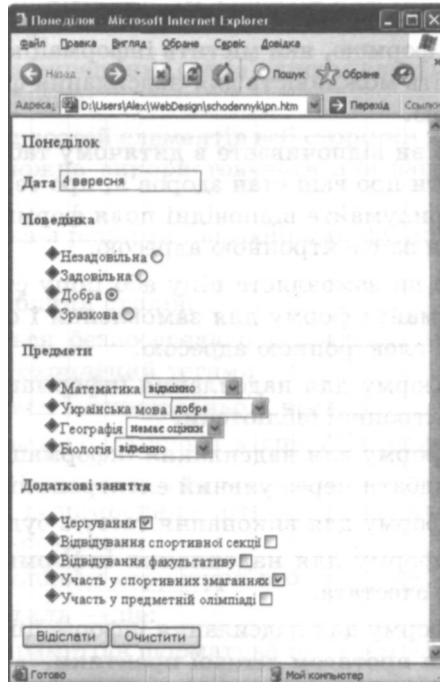


Рис. 5.44. Заповнення полів форми для понеділка

22. Відкрийте електронну скриньку, зазначену в коді форми. В ній міститиметься лист, подібний до наведеного нижче:
 Дата=4 вересня
 Поведінка=Добра
 Математика=відмінно

Українська мова=добре
Географія=немає оцінки
Біологія=відмінно
Додатково=Чергування
Додатково=Участь у спортивних змаганнях
Відправлення інформації=Відіслати.

Самостійна робота

Розробіть такі варіанти веб-сторінок із формами.

- Сторінка з формою, яка містить інформацію про погоду цього дня. Створіть можливість для надсилання форми за електронною адресою.
- Уявіть, що ви відпочиваєте в дитячому таборі, і ваші батьки хочуть знати про ваш стан здоров'я, про те, як ви харчуєтеся та інше; придумайте відповідні поля форми та організуйте її надсилання за електронною адресою.
- Уявіть, що ви замовляєте піцу або іншу страву через Інтернет; придумайте форму для замовлення і організуйте її надсилання за електронною адресою.
- Розробіть форму для надсилання інформації про книжку до уявної електронної бібліотеки.
- Розробіть форму для надсилання інформації про товар, який можна придбати через уявний електронний магазин.
- Розробіть форму для виконання тесту з будь-якого предмета.
- Розробіть форму для надсилання інформації про учня для отримання атестата.
- Розробіть форму для надсилання інформації, що характеризує роботу учня протягом літньої практики.
- Придумайте свій варіант використання форм.

Тест

1. Таблиця каскадних стилів — це:
 - а) файл з описом структури веб-сторінки;
 - б) певна структура з описом властивостей елементів веб-сторінки;

- в) спеціальна структура, в якій задається колір шрифту для елементів веб-сторінки;
 - г) набір правил оформлення та форматування, які застосовують до різних елементів веб-сторінки;
 - д) файл із призначенням кольору тла для веб-сторінки.
2. Зовнішня таблиця стилів — це:
- а) файл із розширенням .html з описом властивостей елементів веб-сторінки;
 - б) файл із розширенням .txt з описом властивостей елементів веб-сторінки;
 - в) файл із розширенням .css із описом властивостей елементів веб-сторінки;
 - г) опис властивостей елементів веб-сторінки в окремому файлі, який можна використовувати для багатьох веб-сторінок;
 - д) веб-сторінка з різними настройками функціональних розділів.
3. Внутрішня таблиця стилів:
- а) розміщується безпосередньо в розділі <HEAD> усередині блоку, що охоплений тегами <STYLE>...</STYLE>;
 - б) розміщується безпосередньо в розділі <HEAD>;
 - в) розміщується у будь-якому місці HTML-документа у фігурних дужках;
 - г) це файл із розширенням .css із описом властивостей елементів веб-сторінки;
 - д) це файл з описом структури веб-сторінки,
4. Вбудований стиль — це:
- а) перелік параметрів форматування елемента веб-сторінки, заданий у його тегу за допомогою атрибута STYLE;
 - б) опис властивостей фрагмента тексту чи об'єкта, розміщений у фігурних дужках у будь-якому місці HTML-документа;
 - в) опис властивостей елемента веб-сторінки в її тілі;
 - г) файл із розширенням .html з описом властивостей елементів веб-сторінки;
 - д) файл з описом структури веб-сторінки.

5. Сценарій — це:
 - а) список гіперпосилань веб-сайту;
 - б) програма, написана спеціальною мовою програмування і вбудована в HTML-документ;
 - в) послідовність відображення веб-сторінок під час перегляду сайту;
 - г) програма, яку використовують для динамічного оновлення веб-сторінки;
 - д) програма будь-якою мовою програмування, розміщена на веб-сторінці.
6. JavaScript — це:
 - а) певна структура, що описує властивості елементів веб-сторінки;
 - б) файл із розширенням .html з описом властивостей елементів веб-сторінки;
 - в) мова програмування, яка дає змогу вбудовувати виконувани модулі в документи, написані мовою HTML;
 - г) програма, що дає змогу імітувати рух зображень на веб-сторінці;
 - д) засіб для створення інтерактивних веб-сторінок.
7. Об'єктна модель — це:
 - а) структура, що дозволяє описати властивості елементів веб-сторінки;
 - б) подання об'єктів, властивостей і подій у вигляді, зручному для роботи з ними в кодах HTML та у сценаріях;
 - в) спосіб взаємодії між HTML-кодом веб-сторінки та користувачем;
 - г) засіб для роботи зі структурою документа;
 - д) основа для створення динамічно керованих веб-сторінок.
8. Об'єкт Window:
 - а) це основний контейнер, в якому розміщується все те, чим можна керувати за допомогою браузера;
 - б) керує інформацією, що міститься у видимій на екрані частині веб-сторінки;

- в) надає всю інформацію про HTML-документ за допомогою колекцій і властивостей;
- г) це об'єкт найвищого ієрархічного рівня, в який завантажуються документи;
- д) надає можливість для керування вмістом та розмірами вікна під час відтворення веб-сторінки у браузері.

9. Об'єкт Document:

- а) керує інформацією, що міститься у видимій на екрані частині веб-сторінки;
- б) надає всю інформацію про HTML-документ за допомогою колекцій і властивостей;
- в) надає всю інформацію про HTML-документ, а також методи та події для роботи з ним;
- г) дає змогу динамічно формувати вміст сторінки у процесі її завантаження;
- д) це основний контейнер, в якому розміщується все те, чим можна керувати за допомогою браузера.

10. Тег <FORM>...</FORM>:

- а) інформує про використання JavaScript або іншої мови веб-сценаріїв у тілі HTML-документа;
- б) визначає атрибути кнопок, розташованих на веб-сторінці;
- в) форматує певну частину тексту;
- г) містить теги полів для введення даних, перемикачів, прапорців, кнопок;
- д) містить атрибути, що надають можливість надсилати дані на сервер для подальшого опрацювання.

11. Тегом кнопки надсилання даних, введених у форму, є:

- а) <INPUT TYPE="reset" ...>;
- б) <INPUT TYPE="submit" ...>;
- в) <INPUT TYPE="goto" ...>;
- г) <INPUT TYPE="select" ...>;
- д) <INPUT TYPE="radio"...>.

12. Переглянути список шаблонів і майстрів форм, пропонованих програмою FrontPage, можна:
- а) за допомогою команди **Файл • Создать (Файл • Створити)** і посилання **Другие шаблоны страниц (Інші шаблони сторінок)**;
 - б) за допомогою команди **Файл • Создать (Файл • Створити)** і посилання **Другие шаблоны веб-узлов (Інші шаблони веб-вузлів)**;
 - в) вибравши в меню **Формат (Формат)** команду **Тема (Тема)**, а потім тему для сторінок із формами;
 - г) вибравши в меню **Вставка (Вставка)** команду **Форма (Форма)**, а потім тип поля форми;
 - д) вибравши команду **Страница (Сторінка)** у меню кнопки **Создание новой обычной страницы (Створення нової звичайної сторінки)**.

Розділ 6

Хостинг та популяризація сайтів

Після того як процес формування сайту завершено — створено всі сторінки, налагоджено гіперпосилання та реалізовано всі заплановані функції, варто подумати про його розміщення у глобальній мережі. Сторінки, малюнки та інші елементи сайту мають зберігатися на веб-сервері — комп'ютері з постійним та швидким доступом до Інтернету, на якому функціонує необхідне програмне забезпечення. Саме через нього відвідувачі зможуть отримати доступ до вашого веб-ресурсу.

У цьому розділі описано, як розмістити сайт у мережі Інтернет і прорекламувати його за допомогою банерних мереж та шляхом реєстрації у пошукових системах.

Розміщення веб-сайту в мережі інтернет

Як ви вже знаєте з розділу 1, є організації, які надають свої комп'ютери для розміщення веб-сторінок. На сервері однієї з них (<http://www.narod.ru>) ви вже розміщували свій сайт, а також користувалися засобами автоматичного створення веб-сторінок.

Нагадаємо, що цю послугу називають *хостингом* (англ. host — господар). Тобто хостинг — це виділення на сервері (хості) розділу у вигляді каталогу, в якому зберігається вся інформація сайту — HTML-документи, графіка, кліпи, скрипти тощо. Сервер має бути потужним комп'ютером, постійно підключеним до мережі Інтернет, на якому функціонує програмне забезпечення, необхідне для правильної роботи розміщених на ньому сайтів.

Вибір URL-адреси

Процедура розміщення сайту в Інтернеті передбачає надання йому адреси. Нижче описані деякі особливості та правила, які слід враховувати під час вибору адреси для свого веб-сайту.

- Домени першого рівня (верхнього або базового) мають наперед визначені імена і поділені на дві групи за тематичними та географічними ознаками. Зокрема, домени ua, ru, uk, us є географічними доменами України, Росії, Великобританії та США. А до тематичних доменів належать такі: com — домени комерційних організацій, org — неурядових, edu — освітніх, net — організацій, що надають мережні послуги, biz — домен бізнес-сайтів та інші. Таким чином, у вас немає великих можливостей щодо вибору цього домену.
- Домени подальших рівнів вибирають, враховуючи назву організації чи тематику веб-сайту. Наприклад, для сайту школи № 89 логічно вибрати домен school89 чи SQschool. Для сайту про домашніх улюбленців, який ви створили і розмістили на одному з безкоштовних хостингів, обрано ім'я protvaqyn. Досить багато імен вже зайняті, тому доводиться проявляти фантазію. Якщо, наприклад, домен firma зайнятий, можна спробувати firmacompany або firmasite. У доменних іменах допускається використання дефісу (-) — firma-company, однак не варто застосовувати у веб-адресах спеціальні символи (наприклад, ~), адже не всі користувачі можуть знайти їх на клавіатурі.
- Адреса сайту має бути короткою, легкою для запам'ятовування, зрозумілою і максимально простою, не викликати непорозумінь під час набору чи диктування.

Як бачите, правила нескладні, а їх виконання сприяє дотриманню відносного порядку в мережі Інтернет та допомагає у пошуку необхідної інформації.

Слід також пам'ятати, що адресу необхідно правильно подавати відвідувачам сайту. Якщо назва домену складається з кількох частин, їх бажано виділяти: livejournal, myspace, euronews, nytimes тощо. У друкованому вигляді можна виокремлювати частини кольором, а у випадку диктування адреси вголос — відділяти інто-

наційно. Синтаксично правильним вважають написання адреси разом із назвою протоколу, а також додавання символу / в кінці адреси. Тобто URL-адреса повинна мати вигляд на кшталт <http://www.astra.lviv.ua/>.

Хостинг на безкоштовному сервері

Сьогодні є дуже багато компаній, які надають послугу хостинга. Багато з них робить це за плату: користувач отримує дисковий простір та інструменти керування ним, лише заплативши певну суму. Такий хостинг переважно пропонує ширші можливості з керування, більше дискового простору, оперативнішу службу підтримки, можливість використання різних доменних імен, додаткових сервісів тощо.

Однак найчастіше обирають безкоштовний хостинг. Такий спосіб розміщення сайту звичайно дуже вигідний економічно, проте за нього теж доведеться розплачуватися. Хост-компанія отримує можливість розміщувати рекламу на сторінках сайтів у вигляді банерів, текстових посилань чи фреймів.

Серед популярних служб безкоштовного хостинга можна виділити такі:

- українські: <http://ho.com.ua/>, <http://www.halyava.in.ua/>;
- російські: <http://www.narod.ru/>, <http://www.boom.ru/>, <http://www.ucoz.ru/>, <http://hut.ru/>, <http://jino-net.ru/>, <http://my.km.ru/>, <http://www.holm.ru/>, <http://ru-board-host.com/>;
- світові: <http://geocities.yahoo.com/>, <http://freewebs.com/>.

Замовлення безкоштовного хостинга

Розглянемо процедуру реєстрації для замовлення послуги безкоштовного хостинга на сервері <http://ho.com.ua/>. Щоб її одержати, необхідно мати свою електронну скриньку.

Вправа 6.1

1. Відкрийте браузер, введіть у поле адреси <http://ho.com.ua/>. Дочекайтеся повного завантаження сторінки.
2. Заповніть поля форми реєстрації, розміщеної внизу сторінки (рис. 6.1).

Зокрема введіть своє ім'я, контактну адресу електронної пошти, логін, домен (якщо він уже є), категорію сайту (наприклад, «Гумор», «Діти», «Закон», «Комп'ютери», «Ігри», «Реферати», «Відпочинок», «Кіно», «Фото», «Розваги» тощо) та тему сайту. Слід прочитати правила хостинга (скориставшись посиланням правила), що визначають інформацію, яку не можна розміщувати на сервері (переважно це стосується музики, хакерських програм, таємних даних тощо), а також описують особливості роботи банерної мережі цієї служби (спосіб розміщення на сайті реклами).

Заказ хостинга	
Ваше имя:	lessonsite
Ваш контактный e-май:	lessonsite@i.ua
Желаемый логин:	lessonsite
Ваши существующие домены:	.ho.com.ua
Категория сайта:	Увлечения и хобби
Тема сайта:	Домашні улюбленці
<input checked="" type="checkbox"/> С правилами ознакомлен и согласен (правила)	
<input type="button" value="Заказать"/>	

Рис. 6.1. Форма замовлення хостинга на ho.com.ua

Отримайте та перегляньте лист (рис. 6.2) з інформацією про надану послугу хостинга, який приходить замовнику на зазначену ним адресу електронної пошти після клацання кнопки **Заказать** (Замовити). У ньому подана детальна інформація про послугу хостинга: назва сервера, на якому розташовано сайт (наприклад, s2.ho.com.ua), квота дискового простору для розміщення файлів сайту (32 Мбайт), логін та пароль користувача для доступу до цих файлів.

Пароль генерується автоматично і тому незручний для запам'ятовування. Його можна змінити у будь-який момент на сайті хостингової компанії, і це варто зробити. Крім того, користувачу надають інформацію про каталоги його сайту, в яких він може розміщувати свої файли. Цей лист містить також усі контактні дані адміністрації хостинга, яка відповідає за його роботу, і до якої можна звертатись у разі виникнення запитань чи проблем.

Здравствуйте, lessonsite!

Ваша заявка отработана.

Сайт protvarup создан.

Администрирование сайта производится по FTP:

```
Login:          lessonsite
Password:       SVRkpy79lm
Hosting server: s2.ho.com.ua
Server Name:    lessonsite.ho.com.ua
Server Alias:   www.lessonsite.ho.com.ua
Quota:         32
```

Для нормального функционирования виртуального сервера необходимо, чтобы домены существовали и имели адрес 62.149.0.85.

Все страницы следует помещать в каталогhtdocs/ Сейчас там находятся образцы документов и скриптов, работу которых можно увидеть:

```
http://lessonsite.ho.com.ua/
http://lessonsite.ho.com.ua/test.html
http://lessonsite.ho.com.ua/test.shtml
http://lessonsite.ho.com.ua/test.cgi
http://lessonsite.ho.com.ua/test.php
```

В каталоге cgi-bin/ сайта содержится интерпретатор php. Он необходим для работы php.

В каталоге logs/ находятся файлы протокола работы сайта. Размер этих файлов учитывается в общей квоте.

Для управления хостингом используйте страницу:

```
https://ho.com.ua/cgi-bin/hosting.cgi
```

Доступ к БД MySQL Вы можете самостоятельно открыть в Панели управления хостингом. Для управления MySQL базой используйте страницу:

```
https://s2.ho.com.ua/phpMyAdmin
```

Не забудьте также стать членом Клуба пользователей ho.com.ua:

```
http://hosting-club.ho.com.ua/
```

В Клубе вы всегда можете узнать много интересного, задать свой вопрос и получить на него ответ.

--

```
С уважением, Support team
http://ho.com.ua
mailto:support@ho.com.ua
```

Рис. 6.2. Лист з інформацією про хостинг

Використання хостинга

Власник веб-сайту має змогу переглядати та змінювати файли, які зберігаються у відповідному розділі на сервері. Скориставшись повідомленою йому адресою для керування файлами і ввівши власні логін та пароль, він отримує доступ до цієї функції. У верхній частині сторінки, що відкриється (рис. 6.3), міститься службова

інформація: логін, назва сервера, тема та категорія сайту, домени, розмір каталогів та загальна квота, контактна інформація, а також дата закриття хостинга.

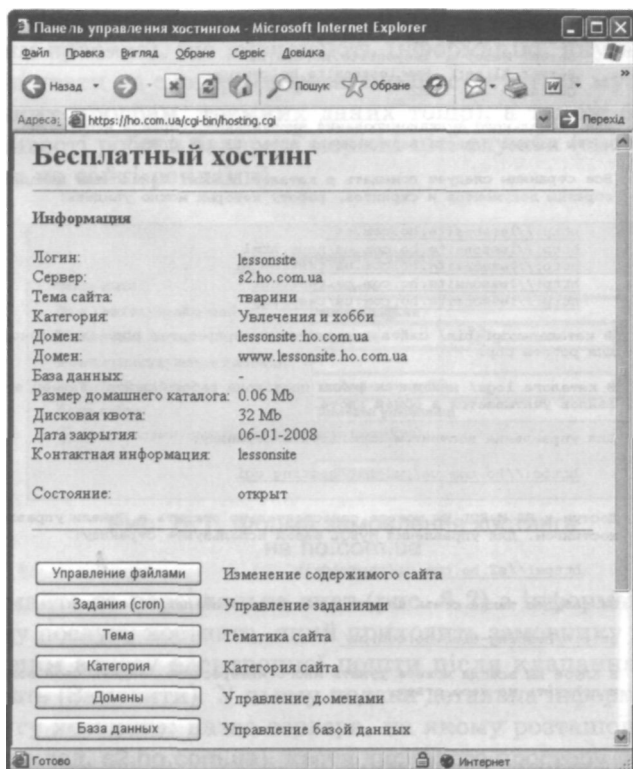


Рис. 6.3. Сторінка для керування хостингом ho.com.ua

За допомогою кнопок цієї сторінки можна виконати низку функцій.

- Управление файлами (Керування файлами) — зміна змісту сайту, завантаження файлів із комп'ютера.
- Задания (стоп) (Завдання (стоп))— визначення завдань, які будуть виконані автоматично у певний час.
- Тема (Тема) — змінення тематики сайту.

- Категорія (Категорія) — змінення категорії сайту.
- Домен (Домен) — керування доменами. Якщо користувач отримує новий домен, то може замінити ним домен ho.com.ua.
- База даних (База даних) — робота з розміщеною на сервері базою даних користувача. Після першого клацання цієї кнопки база даних буде створена, а потім оновлюватиметься.
- Журнальні файли (Журнальні файли) — керування журналом доступу до сайту. Кожна дія на сайті записується у спеціальний файл — журнал. Власник сайту може з ним ознайомлюватися.
- Пароль (Пароль) — змінення пароля користувача. Замість автоматично згенерованого пароля можна задати зручніший.
- Контакти (Контакти) — змінення контактної інформації. Іноді адміністрація хостинга розсилає повідомлення про технічне обслуговування, протягом якого доступ до файлів може бути обмеженим, новини служби чи іншу корисну інформацію. Тому потрібно, щоб чинна контактна адреса електронної пошти користувача була завжди відома адміністрації.
- Продовження (Продовження) — можна в будь-який час продовжити хостинг на 90 днів. Якщо цього не зробити, його буде припинено автоматично після настання дати закриття (див. рис. 6.3).
- Оплата (Оплата) — оскільки хостинг ho.com.ua надає також платні послуги, його користувачі можуть роздрукувати спеціальну квитанцію для їхньої оплати в найближчому відділенні банку.
- Вернутися (Вернутися) — вийти з панелі керування хостингом.

Диспетчер файлів хостинга на ho.com.ua

Основною функцією хостингу, яку використовують на початку роботи, є завантаження файлів сайту на сервер. Для цього призначена кнопка Управление файлами (Керування файлами), яка відкриває диспетчер файлів (рис. 6.4). Він відображає всі файли, які містяться у каталогах, наданих службою хостинга.

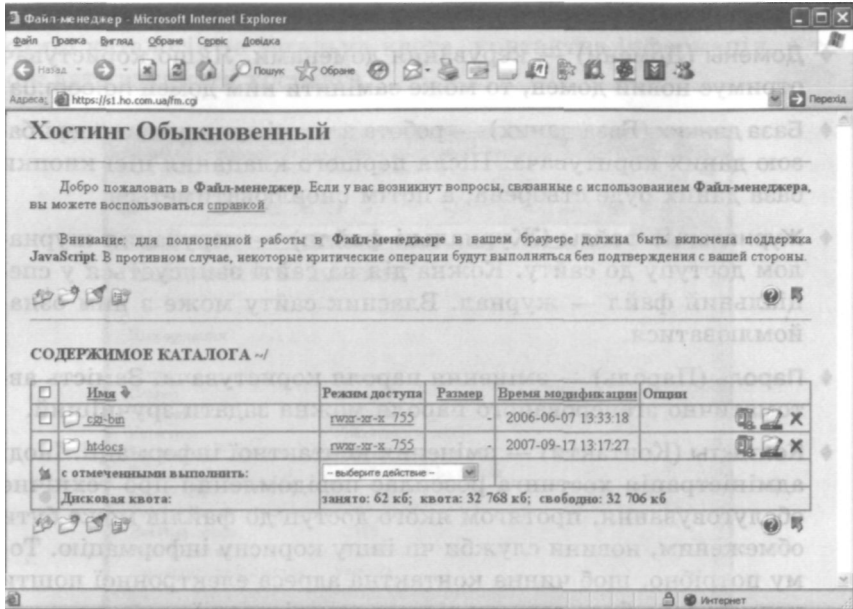








Рис. 6.4. Сторінка диспетчера файлів із вмістом головного каталогу сайту protvatyn.ho.com.ua

Розглянемо основні дії, які можна виконати для керування вмістом веб-сайту.

Вправа 6.2

1. Відкрийте у вікні браузера панель керування хостингом за гіперпосиланням <https://ho.com.ua/cgi-bin/hosting.cgi> та введіть логін і пароль, отримані під час виконання вправи 6.1. Клацніть кнопку Вхід (Вхід).
2. Розгляньте інформацію про обліковий запис хостинга та кнопки керування хостингом.
3. Відкрийте диспетчер файлів хостинга за допомогою кнопки Управление файлами (Керування файлами).
4. Перегляньте кнопки керування головної сторінки диспетчера файлів, які розміщені вище та нижче вмісту поточного каталогу і призначені для роботи з файлами.

Кнопка	Призначення	Кнопка	Призначення
	Завантаження файлів		Редагування настройок каталогу
	Створення нового каталогу		Встановлення доступу до каталогу
	Очищення каталогу		Відновлення файлів службового каталогу (у разі випадкового видалення).

- Перейдіть у каталог `htdocs`, де зберігаються всі файли сайту (рис. 6.5). Для цього необхідно клацнути мишею його назву.

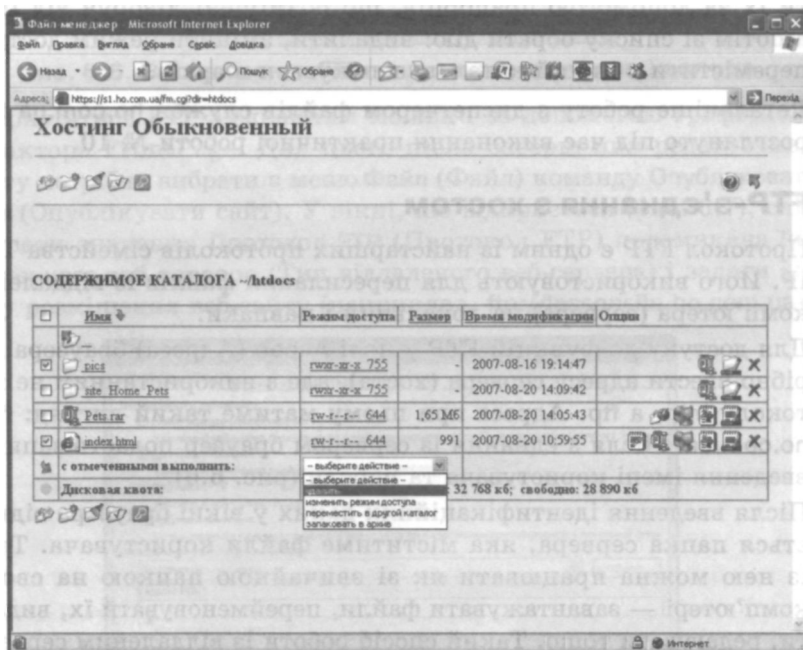






Рис. 6.5. Каталог `htdocs` після завантаження файлів

- Перегляньте кнопки керування файлами сайту. Поруч із назвою кожного файлу відображено режим доступу до нього, розмір, дату і час останніх змін та доступні дії.

Кнопка	Призначення	Кнопка	Призначення
	Редагувати файл		Замінити файл
	Запакувати файл		Перейменувати файл/каталог
	Розпакувати архів		Видалити файл
	Завантажити файл на комп'ютер		

Якщо потрібно виконати дію з кількома файлами, можна вибрати їх за допомогою прапорців, що розміщені ліворуч від імен, а потім зі списку обрати дію: видалити, змінити режим доступу, перемістити в інший каталог, запакувати в архів.

Детальніше роботу з диспетчером файлів служби ho.com.ua буде розглянуто під час виконання практичної роботи № 10.

FTP-з'єднання з хостом

Протокол FTP є одним із найстарших протоколів сімейства TCP/IP. Його використовують для пересилання файлів із віддаленого комп'ютера (сервера) на локальний і навпаки.

Для доступу до функцій FTP у полі **Адрес** (Адреса) браузера потрібно ввести адресу сервера (хоста), але з використанням не протоколу **http**, а **ftp**. Адреса при цьому матиме такий вигляд: **<ftp://ho.com.ua>**. Після з'єднання із сервером браузер подасть запит на введення імені користувача та пароля (рис. 6.6).

Після введення ідентифікаційних даних у вікні браузера відкриється папка сервера, яка міститиме файли користувача. Тепер із нею можна працювати як зі звичайною папкою на своєму комп'ютері — завантажувати файли, перейменовувати їх, видаляти, редагувати тощо. Такий спосіб роботи із віддаленим сервером є досить зручним і простим, не потрібно розбиратися з різними інтерфейсами файл-менеджерів та обмежуватись їхніми не завжди повними функціями.

До файлів сайту також можна отримати доступ, увівши свої логін та пароль як частину адреси: **<ftp://логін:пароль@ho.com.ua>**. При

цьому важливо дотримуватися правил: спершу зазначають логін адміністратора (замість слова логін), потім через двокрапку — реальний пароль адміністратора (а не слово пароль), символ @, а після нього — адресу сервера (ho.com.ua).

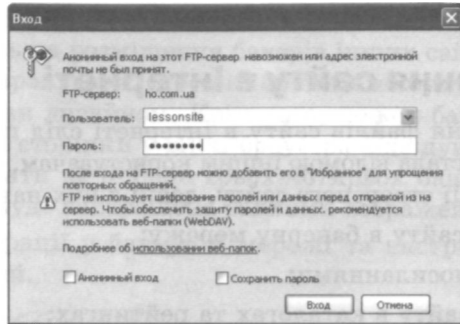


Рис. 6.6. Встановлення FTP-з'єднання із сервером хостинга

Скористатись FTP-з'єднанням можна і за допомогою графічного редактора FrontPage. Для цього після завершення редагування сайту потрібно вибрати з меню Файл (Файл) команду Опублікувати узел (Опублікувати сайт). У вікні, що відкриється (рис. 6.7), слід вибрати значення Протокол FTP (Протокол FTP) перемикача Тип удаленного веб-сервера (Тип віддаленого веб-сервера) і задати адресу розміщення веб-сайту (наприклад, [ftp://lessonsite.ho.com.ua](http://lessonsite.ho.com.ua)).

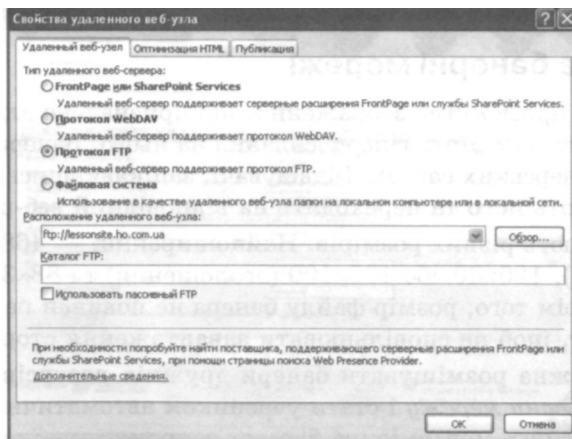


Рис. 6.7. FTP-з'єднання з хостом ho.com.ua засобами FrontPage

Далі програма відкриє діалогове вікно, яке містить запит логіка та пароля доступу до сервера, де потрібно ввести свої дані та клацнути кнопку ОК. Якщо все зроблено правильно, з'єднання із сервером хостинга буде встановлено, і файли буде розміщено за заданою адресою.

Рекламування сайту в Інтернеті

Після розміщення файлів сайту в Інтернеті слід подбати про те, щоб його адреса стала відомою іншим користувачам. Є кілька способів популяризації створеного сайту, деякі з них наведено нижче:

- включення сайту в банерну мережу;
- обмін гіперпосиланнями;
- реєстрація сайту в каталогах та рейтингах;
- реєстрація сайту в пошукових системах;
- створення партнерської програми;
- розміщення гіперпосилань на тематичних форумах і в засобах масової інформації;
- створення поштової розсилки;
- платна реклама.

Розглянемо докладніше використання банерних мереж та пошукових систем.

Банери та банерні мережі

Банер — це прямокутне зображення, що призначене для реклами певного сайту та містить гіперпосилання на нього. Банери розміщують на партнерських сайтах. Відвідувачі, зацікавившись зображенням, клацають його та переходять на відповідний веб-ресурс.

Банери бувають різних розмірів. Найпоширеніші — 468x60 пікселів (смужка), 120x60 або 100x100 (оголошення) та 88x31 пікселів (кнопка). Крім того, розмір файлу банера не повинен перевищувати 15 Кбайт, щоб не сповільнювати завантаження сторінок.

На сайті можна розміщувати банери дружніх ресурсів, а можна увійти в *банерну мережу* і стати учасником автоматичного обміну банерами. Таких мереж існує багато, зокрема українські **uabanner.com** та **www.hitua.net** або російські **www.tbn.ru**, **www.top468.net**.

Окрім загальних банерних мереж, є ще й тематичні, які охоплюють лише сайти певних категорій, забезпечуючи при цьому цільову та ефективну рекламу. Мережі пропонують як платні, так і безкоштовні послуги, тому кожен власник сайту може обрати такий сервіс, який його влаштує.

Погоджуючись на розміщення банерів інших сайтів у себе на сторінках, слід врахувати, що такі об'єкти не будуть гармоніювати з обраним вами дизайном. Крім того, кожен банер збільшує час завантаження сторінки у вікні браузера відвідувача. Тому краще використовувати не більше трьох-чотирьох банерів на сторінці, інакше вона буде захарашена зайвими зображеннями.

Процес реєстрації у банерній мережі та налаштування банерів досить простий.

Реєстрація у банерній мережі

Для початку слід вибрати мережу і зареєструватись у ній. Розглянемо це на прикладі мережі uabanner.com. Внизу її домашньої сторінки міститься гіперпосилання [Регистрация \(Реєстрація\)](#), за допомогою якого відкривається відповідний розділ (рис. 6.8).

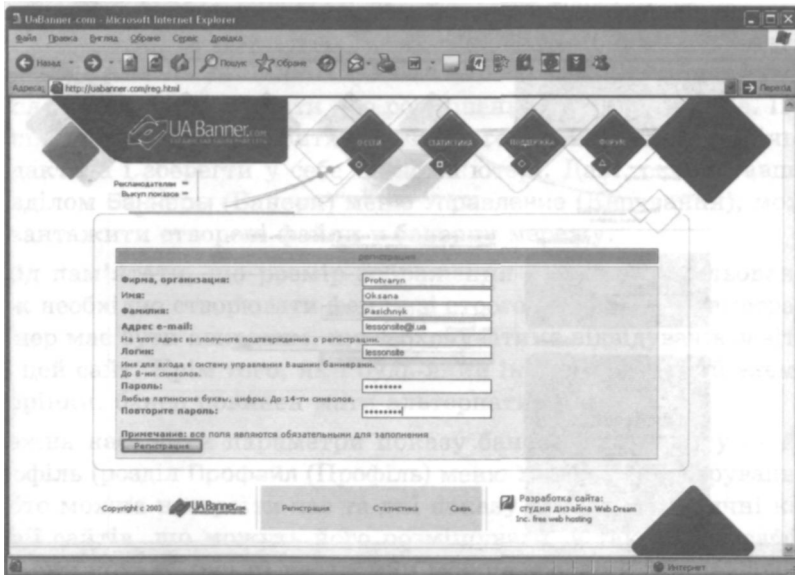


Рис. 6.8. Реєстрація у банерній мережі uabanner.com

Тут слід ввести необхідну інформацію — назву організації, прізвище та ім'я її представника, адресу електронної пошти, бажаний логін та пароль доступу до сайту банерної мережі — і клацнути кнопку **Регистрация (Реєстрація)**.

Далі можна розпочинати безпосередню роботу в мережі.

Настроювання банерних показів

Після реєстрації можна визначити, у роботі якої підмережі братиме участь сайт, виходячи з розміру банера. Для цього необхідно вибрати з меню **Подсети (Підмережі)** відповідний розмір банерів (наприклад, 468x60, 100x100, 120x60 тощо).

Наступним кроком є додавання до мережі свого сайту, який зможе розмістити рекламні банери інших ресурсів (для нього тут використовують термін «майданчик»). Це роблять за допомогою відповідної форми, розташованої праворуч, де треба ввести назву сайту та його адресу, зазначити тематичну категорію, після чого клацнути кнопку **Добавить новую площадку (Додати новий майданчик)** (рис. 6.9).

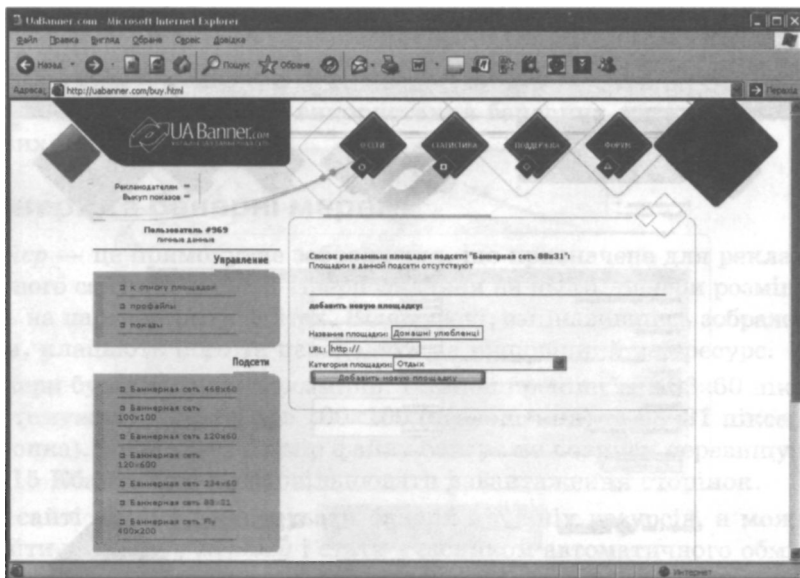


Рис. 6.9. Додавання сайту-майданчика для розміщення банерів

Після реєстрації цього майданчика система видасть спеціальний HTML-код, який слід розмістити на своєму сайті. Це можна зробити на одній або кількох сторінках, враховуючи правила вибраного хостинга та банерної мережі. Код дасть змогу мережі завантажувати банери на сторінку, де він розміщений. Нижче наведено приклад такого коду.

```
<script language='JavaScript'>
// <! —
var uid = 1558;
var rndnum = Math.round(Math.random() * 10000000);
document.write('<center><a href="http://uabanner.com/
bn.php?act=go&id=' + uid + '&n=6Sr=' + rndnum + '
" target="_blank">');
document.write('</a></center>');
// -->
</script>
```

Наступним кроком є визначення сайтів, яких не можна рекламувати на цьому майданчику. Кожен власник сайту може заборонити показ деяких категорій банерів на своїх сторінках.

Після цього варто подбати про розміщення власних банерів. Передусім їх потрібно створити за допомогою доступного графічного редактора і зберегти у себе на комп'ютері. Далі, скориставшись розділом Баннеры (Банери) меню Управление (Керування), можна завантажити створені файли у банерну мережу.

Слід пам'ятати, що розмір зображення жорстко регульований, тож необхідно створювати файли зі строго заданими параметрами. Банер має бути яскравим, що заохочуватиме відвідувачів завітати на цей сайт. Крім того, як і будь-який інший графічний елемент сторінки, банер повинен мати альтернативний опис.

Можна настроїти параметри показу банера свого сайту — його профіль (розділ Профайл (Профіль) меню Управление (Керування)), тобто можна настроїти час та дні показу банера, тематичні категорії сайтів, що можуть його розміщувати, а також географічні регіони показу. Всі ці настройки можна зберегти під певною назвою і застосовувати до різних банерів.

Сайт банерної мережі пропонує статистику показів банерів на зареєстрованому майданчику. Крім демонстрації власних банерів у обмін на розміщення банерів інших ресурсів, можна придбати додаткові покази за певну плату.

Є також спеціальні мережі, за своєю суттю схожі на банерні, тільки обмін у них відбувається не графічними зображеннями, а виключно текстовими посиланнями. Прикладами таких мереж є myic.ru, www.tle.ru, tx3.design.ru. Після реєстрації власник сайту також отримує код, який слід розмістити на домашній сторінці свого ресурсу.

Реєстрація ресурсу в пошукових системах

Щоб отримати доступ до будь-якого інтернет-ресурсу, необхідно знати його мережну адресу. Оскільки Інтернет постійно розвивається і оновлюється, в ньому не може бути єдиного каталогу. Проте є різні інформаційно-пошукові системи, які допомагають користувачам знайти потрібну інформацію. Прикладами таких систем є тематичні каталоги та пошукові сервери.

Більшість відвідувачів переходять на веб-сайти саме з пошукових ресурсів. Користувачі зазвичай переглядають не більше 3-4 перших сторінок із результатами пошуку, тому слід подбати про те, щоб створений сайт потрапляв до них. Для цього необхідно знати принципи функціонування пошукових систем Інтернету.

Тематичні каталоги та пошукові машини

Тематичні каталоги — це інформаційно-довідкові системи, які вручну створюють редактори на основі інформації, зібраної на серверах Інтернету, та відомостей про сайти, внесених самими користувачами. Каталоги мають деревоподібну структуру, за якою всі галузі людської діяльності поділено на розділи: наука, техніка, мистецтво, економіка, політика, спорт тощо, а ті, у свою чергу, — на підрозділи. Наприклад, мистецтво — це література, театр, живопис, музика, архітектура. Далі йде детальніша класифікація: так, до театрального мистецтва належать опера, балет, драма, мюзикл тощо. Користувачі каталогів починають пошук, вибираючи загальну тему чи відповідну тематичну категорію, а потім крок за кроком звужують коло інтересів.

Головною перевагою такого засобу пошуку є чітка відповідність змісту сайтів тематиці, оголошеній для кожного розділу. Тематичні каталоги відзначаються високою якістю, оскільки їх вміст перевіряють люди — редактори. Недоліком каталогів є неповне охоплення наявних у мережі ресурсів, оскільки відстежити всю інформацію в Інтернеті майже нереально.

Якщо відомі конкретна назва, фраза або термін, які потрібно знайти, то для отримання інформації використовують *пошукові машини*, або *пошукові сервери*. Пошук на них виконують за допомогою спеціально розроблених програм. Їхня робота полягає у періодичному відвідуванні веб-сайтів Інтернету, адреси яких внесені у спеціальний реєстр, для створення та оновлення баз даних, наповнених інформацією з веб-сторінок.

Пошукові машини працюють із використанням автоматичних засобів, що дає можливість порівняно швидко опрацьовувати велику кількість веб-документів і створювати бази даних із мільярдів веб-сторінок. З іншого боку, наявність у документі певних слів, за якими користувач виконує пошук, не завжди свідчить про належність цього документа до конкретної тематики. Тому не можна стверджувати, що якість пошуку інформації в таких базах даних найвища.

Робота пошукової машини відбувається у кілька етапів. Спершу програма-робот сканує інформаційний простір, збирає копії веб-ресурсів і передає їх індексувальній програмі, яка заносить кожне слово у спеціальний покажчик-індекс; це називають *індексацією* сайтів. Після певної обробки цих даних пошукова машина може приймати запит від користувача, виконувати пошук у своїх базах даних і видавати веб-сторінку, що міститиме результати пошуку — гіперпосилання на сайти, на яких є шукана фраза.

До найпопулярніших пошукових машин та каталогів сьогодні можна віднести служби, що розміщені за адресами:

- <http://www.google.com/>;
- <http://www.live.com/>;
- <http://www.meta.ua/>;
- <http://www.bigmir.net/>;
- <http://www.yandex.ru/>;

- [http://www.rambler.ru/;](http://www.rambler.ru/)
- [http://www.yahoo.com/;](http://www.yahoo.com/)
- [http://dmoz.org/.](http://dmoz.org/)

Розглянемо процес додавання ресурсів до баз даних пошукових систем — тематичних каталогів та пошукових машин.

Настроювання сайту для використання пошуковими машинами

Передусім слід пам'ятати, що процес реєстрації у пошуковій системі не є миттєвим, оскільки після подання заявки вона потрапляє у чергу опрацювання, яка може розтягнутися на кілька днів чи тижнів. Аби пошукова машина змогла коректно додати створений сайт до своєї бази даних, бажано, щоб він містив певну службову інформацію. Насамперед це стосується визначення ключових слів та опису ресурсу.

Для веб-сайту слід підібрати близько 20 *ключових слів*, які б якомога точніше відображали його вміст, мету функціонування та призначення. Це можуть бути, наприклад, домашні тварини, папуга, амадін, рибка, мишка, корм, улюбленець, навчальний сайт, HTML тощо. Можна також додати інформацію про автора сайту.

Опис сайту — це інформація, яка міститиметься у результатах пошуку біля назви сайту. Зазвичай це одне, максимум два речення, що описують зміст ресурсу. Якщо він створений кількома мовами, то ключові слова та опис варто зробити усіма ними.

Опис сайту та його ключові слова вводять за допомогою тегів <META>. Цей тег має таку структуру:

```
<META NAME="назва параметра" CONTENT="зміст параметра">
```

Наприклад, для сайту про домашніх улюбленців можна ввести такі теги:

```
<META NAME="keywords" CONTENT="домашні тварини, папуга, амадін, рибка, мишка, корм, улюбленець, навчальний сайт, HTML, home pet, parrot, cockatiel, amadin, finch, fish, hamster">
<META NAME="description" CONTENT="сайт про домашніх тварин нашої родини: папугу, амадінів, рибок, site about home pets of one family: parrot, finches, fish, bunny">
```


Отож, для введення ключових слів використовують параметр `keywords`, а для опису сайту — `description`. У такий самий спосіб за допомогою параметра `author` можна вводити інформацію про автора сайту.

Результати пошуку виводяться відповідно до індексу релевантності. *Релевантність* — це відношення кількості слів із введеного пошукового запиту до загальної кількості слів на сторінці, а в широкому значенні — це міра відповідності результатів пошуку поставленому завданню. Тому кращими вважають короткі сторінки з великою кількістю слів із введеного запиту. Проте не всі вони мають однакову вагу для пошукової системи: більш важливими є службові слова, виділені спеціальними тегами.

Найважливішим вважають тег `<TITLE>`. Слова у ньому зазвичай є найвагомішими для пошукового робота. Крім того, саме із цього тегу буде виведено текст біля назви сторінки в результатах пошуку, тож варто зробити його достатньо інформативним, щоб він відображав зміст документа.

Далі програма розглядатиме теги заголовків — `<H1>`. `<H6>`. Саме тому варто виділяти смислові заголовки на сторінках відповідними тегами, а не шрифтами різних розмірів.

Часто більш важливими є слова альтернативного тексту опису зображень ``. Перш за все вони корисні під час пошуку зображень, коли результатом є не сторінки сайтів, а графічні елементи, оскільки їх шукають за альтернативним описом. Такий сервіс реалізовано на більшості пошукових мапін.

Довгий час наявність певної інформації у тегах на сторінках сайту була єдиним способом оцінювання його вмісту для пошукової системи. Проте у 1998 році з'явився новий алгоритм розрахунку релевантності — *PageRank*, або рейтинг сторінки. Було запропоновано брати до уваги надійність поданої інформації, тобто враховувати гіперпосилання, за допомогою яких можна перейти до розглядуваного документа. Якщо їх багато, це означає, що інформація є корисною, і *PageRank* сторінки підвищується. Багато пошукових систем підтримали цю ідею, і поряд із текстовим наповненням сторінки аналізують і її рейтинг.

Точний принцип роботи пошукових систем є комерційною таємницею, і користувачі мережі не можуть знати, як саме певний

пошуковий сайт визначає релевантність знайдених документів. Проте наведені вище принципи опису змісту сайту допоможуть підвищити його рейтинг у результатах пошуку.

Реєстрація сайту в пошукових системах

Оскільки Інтернет постійно оновлюється і змінюється, в пошукових системах не завжди є найновіші та найповніші версії веб-сторінки, відсутні також відомості про новостворені ресурси. Тому більшість із цих служб дають змогу користувачам самим вносити веб-адреси сторінок, які потрібно додати до бази даних. Це не означає, що всі вони будуть додані відразу, проте вони обов'язково будуть враховані під час наступного сеансу роботи відповідної пошукової програми.

Наприклад, додавання сайту до тематичного каталогу системи Мета відбувається за адресою <http://www.meta.ua/ua/webmaster/addurl.asp>. Авторів ресурсу пропонують ввести короткий опис свого сайту, вибрати тематичну категорію, регіон, мову та інші параметри. Така інформація полегшить роботу редакторів, які після її перевірки внесуть ресурс у реєстр пошукової системи.

Вправа 6.3

1. Відкрийте у браузері сторінку, яка міститься за адресою: <http://meta.ua/> і за посиланням додати сайт перейдіть до реєстраційної форми МетаПаспорта.
2. Розгляньте поля реєстраційної форми (рис. 6.10). Введіть такі дані: логін користувача, пароль та його підтвердження, особисту інформацію (ім'я, прізвище, дату народження, стать, місце проживання), додаткову інформацію (адресу електронної пошти, контрольне запитання та відповідь на нього, контрольні цифри на зображенні) і клацніть кнопку Зареєструватися на Метапаспорте (Зареєструватися на МетаПаспорті).

Після цього можна реєструвати власні ресурси у каталозі Мета (рис. 6.11), задаючи адресу сайту в Інтернеті, ключові слова (їх можна вводити українською, російською або англійською мовами), контактну особу та її адресу електронної пошти (тобто свої координати), власника ресурсу, мови, які використовуються на ресурсі, частоту його оновлення, країну та регіон, а також одну або дві категорії, до яких належить сайт.

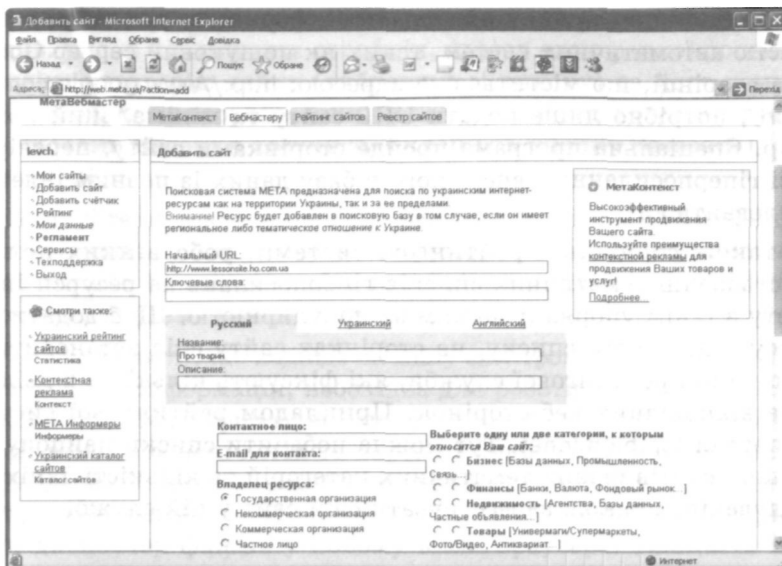


Рис. 6.10. Сторінка реєстрації Метаспорту

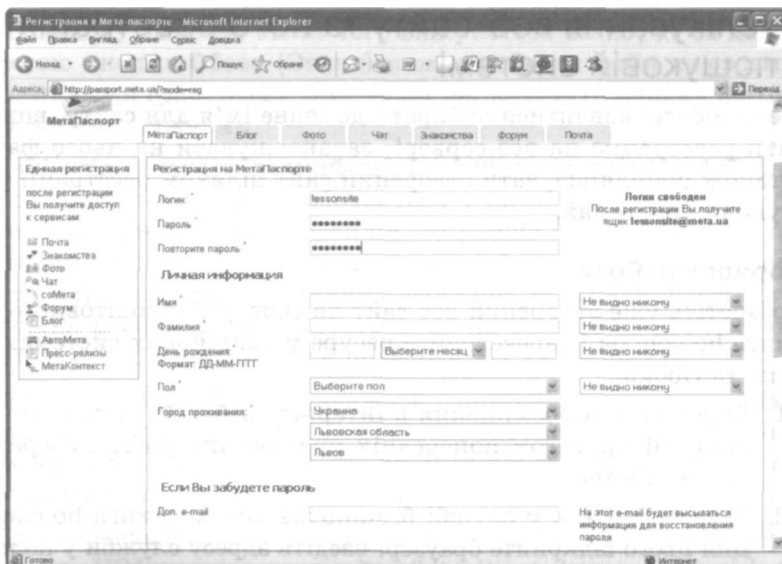


Рис. 6.11. Сторінка реєстрації сайту в каталозі Мета

За дещо іншим принципом додають інформацію про сайти до повністю автоматичних систем, таких як пошуковий сервер Google. На сторінці, що міститься за адресою: <http://www.google.com/ad-durl/>, потрібно лише подати URL сайту та мінімальний коментар. Спеціальна програма пройде сторінками сайту, перевірить усі гіперпосилання і внесе його у базу даних із певним індексом релевантності.

Є також спеціальні рейтингові системи, робота яких полягає у створенні тематичних списків гіперпосилань на ресурси Інтернету з їхнім упорядкуванням за популярністю. Щоб додати свій ресурс до такого списку, на сторінках сайту слід встановити лічильники рейтингової служби, які фіксують кількість відвідувачів відповідних веб-сторінок. Прикладом рейтингової системи є портал top.bigmir.net. Тут можна побачити списки найпопулярніших сайтів різних тематичних категорій та кількість їхніх відвідувачів, а також зареєструвати свій сайт у цій службі.

Практична робота № 10

Публікування веб-сайту та його реєстрація у пошуковій системі

Мета роботи: навчитися вибирати доменне ім'я для сайту, виконувати реєстрацію на веб-сервері, завантажувати на нього файли, а також популяризувати створений сайт шляхом реєстрації у пошукових системах.

Порядок роботи

Розмістіть уже створений веб-сайт на сервері безкоштовного хостинга ho.com.ua і зареєструйте ресурс у пошукових системах Яндекс та Google.

1. Виберіть для розміщення в Інтернеті веб-сайт із тих, що вже створені протягом попередніх практичних робіт і збережені на комп'ютері.
2. Зареєструйтесь у службі безкоштовного хостинга ho.com.ua. Для цього відкрийте браузер, введіть адресу служби у поле Адреса. Дочекайтесь повного завантаження сторінки.

3. Замовте послугу безкоштовного хостинга, заповнивши форму (див. рис. 6.1): зазначте своє ім'я, адресу електронної пошти, логін, категорію сайту (вибравши зі списку) та його тему. Установіть прапорець, який означає, що ви погоджуєтесь з умовами надання хостинга, і клацніть кнопку **Заказать** (Замовити).
4. Активуйте замовлену послугу хостинга: зайдіть у свою поштову скриньку і виконайте інструкції, що містяться у листі від служби ho.com.ua, тобто перейдіть за поданим у листі гіперпосиланням, клацнувши його.
5. У другому листі, що надійде на вашу електронну скриньку, міститься вся інформація, необхідна для роботи із сервером. Перейшовши за гіперпосиланням <https://ho.com.ua/cgi-bin/hosting>, ввівши власні логін та пароль і клацнувши кнопку **Вход** (Вхід), ви отримаєте доступ до керування хостингом. Кнопка **Управление файлами** (Керування файлами) дає змогу відкрити диспетчер файлів у новому вікні браузера. Клацніть її.
6. У диспетчері файлів зайдіть у каталог `htdocs`, де зберігаються всі файли сайту. Для завантаження у каталог файлу клацніть відповідну кнопку, після чого відкриється вікно, де за допомогою кнопки **Обзор** (Огляд) можна вибрати файл на своєму комп'ютері (рис. 6.12). Зробивши це, клацніть кнопку **Загрузить** (Завантажити).

Далі слід зачекати деякий час (залежно від розміру файлу), і обраний документ відобразиться у каталозі. Якщо потрібно завантажити кілька файлів, то краще їх заархівувати на своєму комп'ютері (в одному з таких форматів: `zip`, `rar`, `tar`, `gz`, `bz2`) і завантажити єдиний файл архіву. Після цього його можна розпакувати безпосередньо на сервері за допомогою кнопки, що розміщується праворуч від назви файлу.
7. Закрийте вікно файл-менеджера і поверніться до панелі керування хостингом. Вийдіть звідти, скориставшись кнопкою **Вернуться** (Повернутися), що означатиме завершення режиму редагування параметрів хостинга.
8. Для перегляду результатів роботи у панелі адреси браузера введіть URL розміщеного сайту (<http://логін.го.ком.уа/>) — і на

екрані відобразиться його домашня сторінка, що завантажиться з Інтернету. Саме такою її побачать відвідувачі з різних куточків світу.

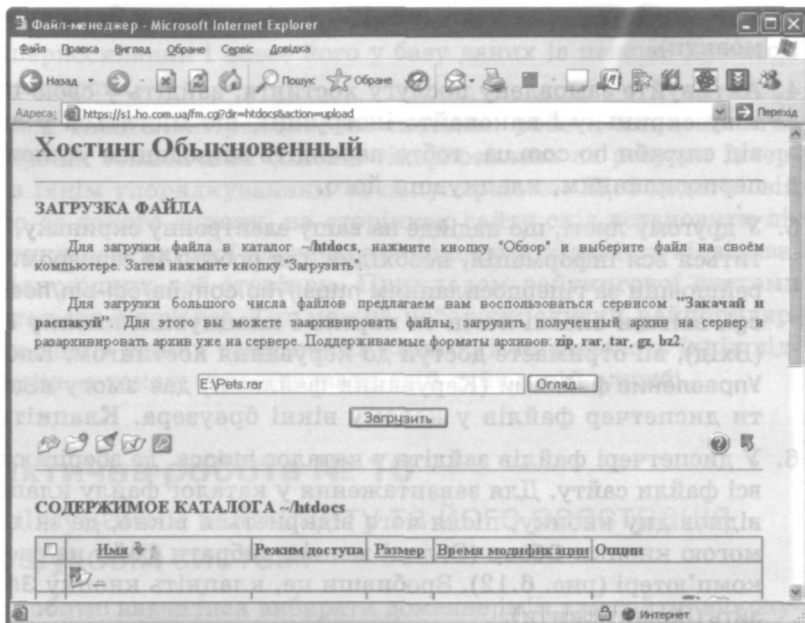


Рис. 6. 12. Панель завантаження файлів із комп'ютера на сервер

9. Якщо виникне потреба у внесенні змін у файли сайту, слід знову зайти на сторінку <https://ho.com.ua/cgi-bin/hosting.odі>, ввести логін та пароль, відкрити файл-менеджер, внести зміни у відповідні файли (скориставшись кнопками, які розміщені праворуч від назв файлів), зберегти їх, а потім перевірити результат роботи. Крім того, файли можна редагувати у себе на комп'ютері (наприклад, за допомогою редактора Microsoft FrontPage), а потім лише замінювати файли на сервері новими версіями.
10. Виберіть 10-15 ключових слів для опису свого сайту, запишіть їх у зошит.

11. Відкрийте у вікні браузера веб-сторінку за адресою http://web-master.yandex.ru/catalogue_free.xml, де можна додати свій сайт до веб-каталогу Яндекс (рис. 6.13). У формі введіть таку інформацію: адресу вашого сайту (<http://логін.го.ком.иа/>), його назву та опис, категорію (тематичну), регіон (Україна), сектор економіки (комерційний, некомерційний), свою адресу електронної пошти, а також символи із наведеного на сторінці зображення для захисту від спам-роботів. Після заповнення всіх полів слід клацнути кнопку Отправить заявку (Відправити заявку).

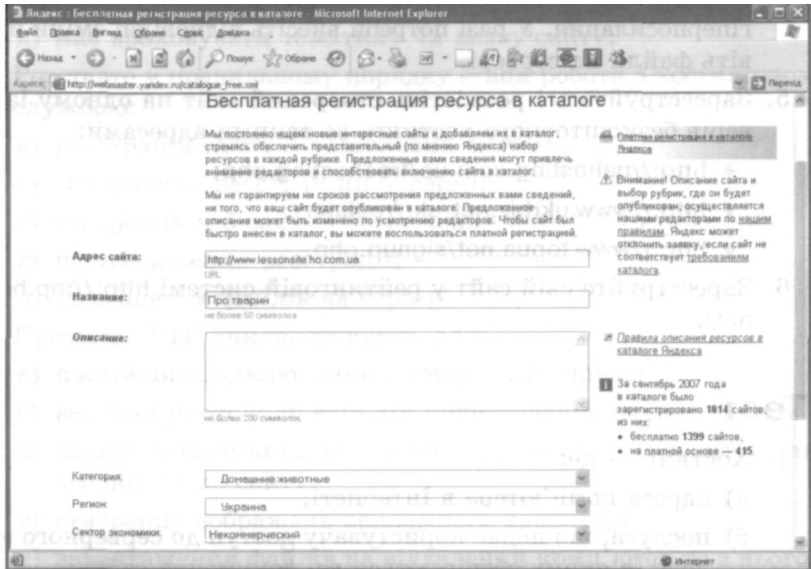


Рис. 6.13. Додавання сайту до веб-каталогу Яндекс

12. Відкрийте у вікні браузера веб-сторінку пошукового сервера Google, що міститься за адресою: <http://www.google.com/addurl/>. Введіть адресу свого сайту і ключові слова, а також сукупність символів із зображення для захисту від спам-роботів. Щоб додати адресу до бази цієї пошукової системи, клацніть кнопку Add URL (Додати URL).

Самостійна робота

1. Створіть для свого сайту сторінку, що міститиме гіперпосилання на веб-ресурси однокласників.
2. Зареєструйтесь у банерній мережі, додайте до неї свій сайт. Задайте параметри банерної системи: географічні, тематичні та регіональні.
3. Створіть банер для свого сайту. Додайте його до банерної мережі.
4. Перевірте відповідність сайту вимогам пошукових систем: чи заповнені теги <TITLE>, , <META>, ЧИ працюють гіперпосилання. У разі потреби внесіть відповідні зміни й оновіть файли на сервері.
5. Зареєструйтесь і розмістіть створений сайт на одному із серверів безкоштовного хостинга за такими адресами:
 - <http://uahosting.com.ua/freehosting.php>;
 - <http://www.ukrbiz.net/ukr/register/>;
 - <http://www.topua.net/signup.php>.
6. Зареєструйте свій сайт у рейтинговій системі <http://top.bigmir.net/>.

Тест

1. Хостинг — це:
 - а) адреса комп'ютера в Інтернеті;
 - б) послуга, що надає користувачу доступ до серверного каталогу, в якому зберігаються всі документи сайту;
 - в) сукупність доступних у мережі Інтернет веб-сторінок, які об'єднані між собою за змістом та навігаційно;
 - г) сервер, який контролює доменні імена в Інтернеті;
 - д) сканування інформаційного простору і збирання копій веб-ресурсів.
2. Веб-сервер — це:
 - а) організація, що контролює доменні імена в Інтернеті;
 - б) спосіб завантаження файлів на хост;

- в) потужний комп'ютер, який постійно підключений до мережі Інтернет;
 - г) один із протоколів Інтернету;
 - д) кілька розділених крапками чисел, що позначають IP-адресу.
3. Доменне ім'я має такі характеристики:
- а) повинно містити символ -;
 - б) має бути якомога коротшим та простішим;
 - в) верхній домен може бути тематичним або географічним;
 - г) повинно містити заголовок стартової сторінки сайту;
 - д) має враховувати тематику чи назву сайту.
4. Розставте у правильному порядку етапи роботи з хостинговою службою:
- а) реєстрація сайту в пошукових системах;
 - б) заповнення реєстраційної форми;
 - в) створення веб-сайту і збереження його на комп'ютері;
 - г) підтвердження реєстрації;
 - д) розміщення файлів на сервері.
5. Протокол FTP використовують для:
- а) постійного підключення до мережі Інтернет;
 - б) ведення протоколу всіх дій користувачів на сайті;
 - в) обліку інформації, що завантажується із сервера після звернення до сайту;
 - г) створення зображень рекламного характеру;
 - д) завантаження файлів на віддалений комп'ютер та з нього.
6. Банер — це:
- а) спосіб завантаження файлів на сервер;
 - б) чотири тризначних числа, розділених крапками, які позначають адресу комп'ютера в Інтернаті;
 - в) прямокутне зображення, яке створене для реклами певного сайту і містить посилання на цей сайт;
 - г) мережний онлайн-журнал;
 - д) організація, що контролює доменні імена в Інтернеті.

7. Вкажіть способи популяризації сайту:
 - а) зазначення автора і дати для зображень;
 - б) реєстрація сайту в пошукових системах;
 - в) обмін гіперпосиланнями;
 - г) вибір FTP-з'єднання із сервером;
 - д) включення сайту в банерну мережу.
8. Пошукова машина виконує такі роботи:
 - а) індексацію сайтів;
 - б) аналіз банерів сторінки;
 - в) видачу результатів пошуку у вигляді гіперпосилань;
 - г) прийом запитів від користувача;
 - д) визначення автора сайту та дати останніх змін.
9. Релевантність — це:
 - а) сканування інформаційного простору і збирання копій веб-ресурсів;
 - б) організація, що контролює доменні імена в Інтернаті;
 - в) відношення кількості слів із пошукового запиту до загальної кількості слів на сторінці;
 - г) кількість тегів на веб-сторінці;
 - д) альтернативний опис зображень.
10. Атрибутами тагу <META> є:
 - а) IMG, SRC;
 - б) NAME, CONTENT;
 - в) TITLE, CONTENT;
 - г) A, HREF;
 - д) NAME, PAGERANK.
11. PageRank — це:
 - а) алгоритм розрахунку релевантності;
 - б) проіндаксований вміст сторінки;
 - в) кількість тагів на сторінці;
 - г) рейтинг сторінки;
 - д) кількість заголовків на сторінці.

12. Покращити якість індексації сайту можна, якщо:
- а) розмістити на ньому банери;
 - б) видалити всі заголовки на сторінці;
 - в) видалити альтернативний опис зображень;
 - г) використовувати теги `<h1>...<h6>` для оформлення заголовків;
 - д) забезпечити альтернативний текст для зображень.

Розділ 7

Дизайн веб-сайтів

Отже, ви вже навчилися не лише користуватися засобами для автоматизованого створення сайтів, а й самостійно розробляти динамічні веб-сайти, сторінки яких містять текст і зображення. Ви також знаєте, як розмістити веб-ресурс на сервері, щоб забезпечити широкий доступ відвідувачів до нього. Однак сайти повинні не лише правильно та швидко функціонувати, а й бути досконалими з естетичного погляду, у жодному разі не втомлювати і не дратувати відвідувачів, а навпаки, дарувати їм насолоду від споглядання та взаємодії з творінням ваших рук. Це стане запорукою популярності, а відтак — і вашої популярності у безмежному віртуальному просторі Всесвітньої павутини.

Останній розділ цього посібника присвячено питанням дизайну веб-сайтів. Ви навчитеся створювати красиві, гармонійні та цікаві веб-сторінки, які будуть приваблювати відвідувачів із різних куточків світу. Ми розглянемо також типові помилки розробників веб-ресурсів для того, аби уникнути їх.

Виконуючи практичну роботу, ви створите веб-сайт, у якому використаєте всі свої знання, отримані упродовж вивчення курсу. Ця робота допоможе узагальнити та повторити вивчене, а також удосконалити набуті вміння та навички зі створення веб-сайтів.

Типи сайтів та їх особливості

Нагадаємо, що *веб-сайт* — це сукупність веб-сторінок, які об'єднані між собою за змістом та навігаційно. Всі сайти, до яких доступ вільний, утворюють Всесвітню павутину — найбільше схвище інформації. Веб-сайтів є дуже багато, і класифікувати їх можна по-різному та за різними критеріями.

Статичні та динамічні сайти

За однією з класифікацій веб-сайти поділяють на статичні та динамічні (рис. 7.1) відповідно до способу наповнення сторінок матеріалом.



Рис. 7. 1. Статичні та динамічні сайти

Статичний сайт — це сайт, вміст якого сталий і не може змінюватися під час відвідування сайту. Він не розрахований на часті оновлення, його легко підтримувати й одній людині, особливо за допомогою програми-редактора. Статичні сайти ще називають сайтами-брошурами.

Динамічний сайт — сайт, що може змінюватися у процесі взаємодії з відвідувачем. Зазвичай це досягається завдяки використанню програмних засобів та наявності бази даних. Отримавши запит на отримання сторінки такого сайту, програмне забезпечення веб-сервера виконує дії відповідно до цього запиту і повертає сторінку клієнту.

Тематична класифікація

Опишемо особливості сайтів різної тематики.

Блог (мережний журнал або щоденник) — сайт, призначений для публічного ведення онлайн-ових щоденників (блогів), може також містити форум. Приклади: www.livejournal.com та www.dnevnik.org.ua.

Бізнес-сайт — містить інформацію про компанію чи послуги, які вона надає. Таких сайтів дуже багато: кожна компанія, навіть маленька, намагається подати відомості про себе у мережі Інтернет. Приклад: www.bhv.kiev.ua.

Сайт електронної комерції — забезпечує функцію електронної торгівлі для купівлі та продажу різних товарів. Наприклад, www.amazon.com.

Громадський сайт — слугує потребам деякої громади, групи людей зі спільними інтересами, які хочуть спілкуватися між собою. Наприклад, сайт www.myspace.com, де міститься інформація, за якою людину можуть знайти однокласники чи друзі через багато років після закінчення школи.

База даних — призначений для пошуку та відображення записів бази даних, наприклад, www.imdb.com чи www.rada.gov.ua (сайт Верховної Ради України, на якому розміщено всі матеріали щодо законодавства нашої держави).

Каталог — сайт, де розміщено посилання, розподілені за категоріями та підкатегоріями; до таких сайтів належать www.yahoo.com та meta.ua.

Сайт матеріалів — призначений для завантаження файлів, таких як програми, ігри, музика, зображення в електронному вигляді тощо. Користуючись цими сайтами, необхідно пам'ятати про авторське право та про особливості його застосування в Інтернеті.

Сайт працевлаштування — дає змогу працевлаштуванню розміщувати інформацію щодо вакансій. Потенційний працівник може її знайти і заповнити анкету чи надіслати резюме, яке відповідає вибраній посаді. Приклад: www.job.ua.

Ігровий сайт — пропонує майданчик для гри. Прикладами можуть бути www.buka.ru, game.km.ru.

Гумористичний сайт — містить анекдоти, жарти, веселі історії чи будь-які інші гумористичні матеріали для розваги (www.kulichki.ru, www.exler.ru).

Інформаційний сайт — призначений для інформування відвідувачів, але необов'язково з комерційною метою. Більшість урядових, освітніх та неприбуткових організацій мають інформаційні сайти, наприклад. Львівська міська рада (www.city-adm.lviv.ua).

Сайт новин — схожий на інформаційний, однак призначений для поширення новин та коментарів до них (www.bbc.co.uk, www.korespondent.net).

Сайт відгуків — на ньому відвідувачі можуть залишити відгуки про різноманітні продукти чи послуги (www.avda.ru — відгуки про відпочинок, www.mrqe.com — рецензії на фільми).

Пошуковий сайт — дає змогу виконувати пошук інформації чи посилань в Інтернаті, наприклад www.google.com.

Особистий сайт — підтримується однією людиною або групою людей (наприклад, сім'єю); містить різну інформацію, яку автори вважають за потрібне розмістити.

Веб-портал — універсальний сайт, через який можна вийти на інші ресурси Інтернету (www.bigmir.net).

Вікі-сайт — сайт, який редагують кілька чи багато людей. Наприклад, вільна енциклопедія uk.wikipedia.org. Кожен відвідувач може зареєструватись у ній і внести інформацію про певний термін, людину, географічний об'єкт, подію тощо.

Деякі сайти можна віднести до кількох категорій одночасно. Наприклад, на сайті електронної комерції може рекламуватися певна продукція, а також він може містити інформаційні документи, звіти тощо.

Візуальна класифікація

За цією класифікацією сайти поділяються на текстові, графічні та імітаційні.

Текстові сайти містять переважно текстову інформацію та відносно мало зображень (рис. 7.2). Вони порівняно невеликі за розміром і мають простий дизайн. Дуже часто сайти інших типів крім стандартної мають ще й текстову версію — для тих відвідувачів, які з різних причин не хочуть завантажувати повні варіанти веб-сторінок.

Графічні сайти окрім певних текстових матеріалів містять багато графіки: спливаючі меню, піктограми, анімаційні зображення тощо (рис. 7.3).

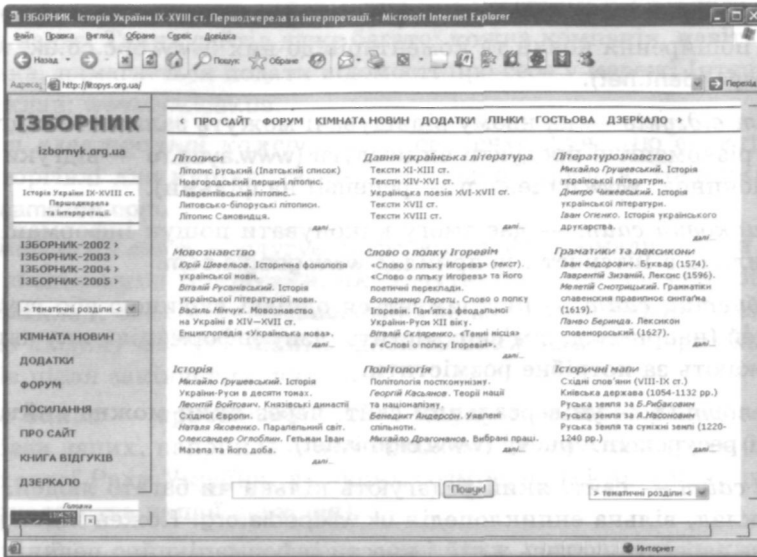


Рис. 7.2. Сайт електронної бібліотеки давньої української літератури litopys.org.ua

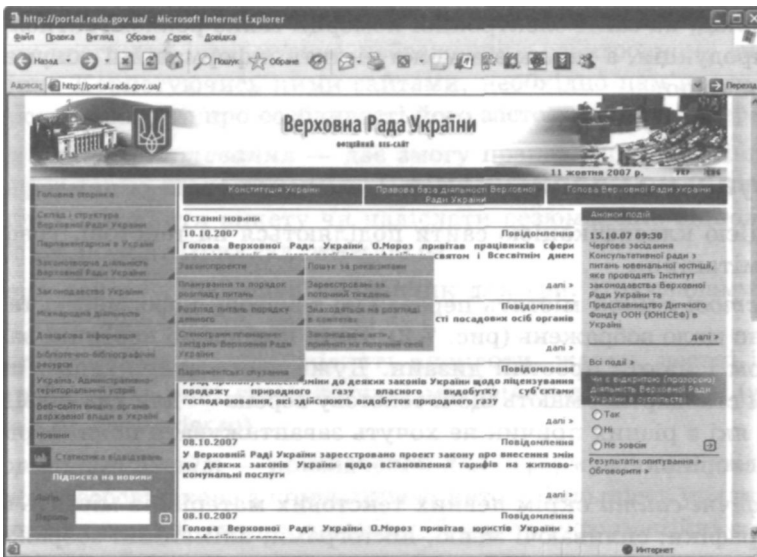


Рис. 7.3. Сайт Верховної Ради України www.racla.gov.ua

Імітаційні сайти запозичують ідею оформлення з реального світу. Так, наприклад, сайт про автомобілі може містити програму, що відображає всі деталі автомобіля на екрані та дає огляд з позиції водія у процесі руху (рис. 7.4).

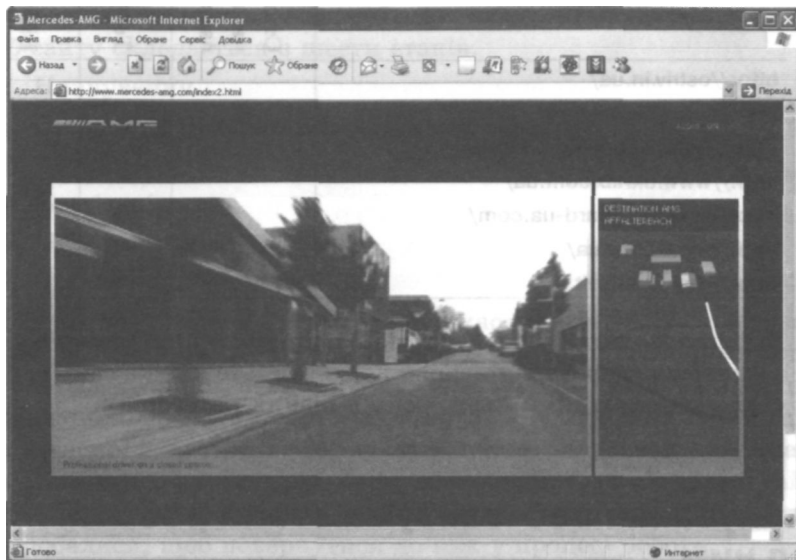


Рис. 7.4. Сайт компанії Mercedes: mercedes-amg.com

Такі сайти найчастіше отримують нагороди Webby Awards за свій нетрадиційний, інноваційний дизайн. Наприклад, сайт LeoBurnett.са став переможцем 2005 року в категорії «Кращий візуальний дизайн».

Як бачимо, класифікація сайтів непроста та неоднозначна. Передусім можна по-різному визначати критерії, за якими класифікують сайти. Крім того, іноді складно визначити, до якої категорії належить той чи інший сайт. Проте така класифікація допомагає зорієнтуватись у величезній кількості ресурсів Інтернету.

Вправа 7.1

1. Відкрийте у вікні браузера сайти за адресами, які наведено в таблиці, та визначте їхню приналежність до однієї з категорій (заповніть таблицю).

Адреса	Статичний чи динамічний сайт	Тематична	Візуальна
http://ostriv.in.ua/ http://www.bigmir.net/ http://www.kinopalace.lviv.ua/ http://www.ukrlib.com.ua/ http://www.postcard-ua.com/ http://www.bmw.ua/ http://www.aerosvit.ua/			

2. Відкрийте сайт Webby Awards за адресою <http://www.webby-awards.com/webbys/current.php> і перегляньте кілька сайтів, що стали переможцями цього року в різних категоріях.

Планування веб-сайту та етапи роботи над ним

Перед тим як створювати сайт, варто визначити, для чого ви це робите, а також спланувати свою роботу. Веб-сайт може мати багато завдань. Якщо, наприклад, головною метою бізнес-сайтів та сайтів електронної комерції є реклама і продаж товарів та послуг, то неприбуткові організації та приватні особи завдяки присутності в Інтернеті можуть легко інформувати людей про себе, залучати кошти, збирати відгуки про певні послуги чи товари, анонсувати події, концерти, збори, надавати простір для проведення дискусій та діалогів, спільної роботи над проектами, пропонувати доступ до різноманітних баз даних, створювати фотогалереї та архіви матеріалів, налагоджувати зв'язки з пресою тощо. З розвитком технологій з'являються й інші способи використання можливостей Інтернету.

З наведеної вище класифікації сайтів видно, що для різних потреб розробникові необхідно створювати різнотипні сайти.

використовуючи різні технологічні рішення. Часто у процесі роботи доводиться змінювати деякі елементи, уточнювати чи розширювати завдання сайту, додавати нові можливості.

Звичайно, кожен веб-проект має свої завдання, під час реалізації яких виникають унікальні проблеми, але загалом процес розробки веб-сайту складається з шести етапів.

1. Планування.
2. Створення інформаційної архітектури.
3. Дизайн.
4. Реалізація.
5. Маркетинг.
6. Оцінювання і підтримка.

Щоб процес розробки сайту не був хаотичним, слід створювати план роботи. При цьому потрібно врахувати всі особливості створення веб-сторінок певного типу, можливі труднощі та шляхи їх подолання. Спочатку варто пошукати сайти схожих організацій чи подібної тематики, сформуванати список складових, матеріалів та елементів дизайну, які автор хоче бачити на власному сайті. Можна до папки Вибране додати адреси ресурсів, які особливо сподобалися, щоб повернутися до них у процесі роботи.

Визначившись із метою сайту і створивши план роботи над ним, можна переходити до наступного етапу — побудови інформаційної архітектури або структури веб-ресурсу. Необхідно детально продумати зміст та організацію сайту, скласти перелік матеріалів, які вже є в електронному вигляді, а також визначити, що ще потрібно оцифрувати. Слід продумати, на які групи можна поділити ці матеріали, розглянути їх як єдине ціле й оцінити, чи немає у цій структурі прогалин.

На етапі дизайну потрібно розробити вигляд кожної окремої сторінки та сайту загалом, створити чи підібрати ілюстрації, фотографії, схеми, логотипи — весь графічний матеріал. Якщо це сайт організації, слід також врахувати, що вона вже може мати певні традиції щодо стильового оформлення рекламних та інформаційних матеріалів, і дизайн сайту має з ними узгоджуватись.

На цьому етапі також виконують роботу з текстовим наповненням сайту — його остаточний підбір, оформлення та редагування.

формування заголовків і назв. Готуючи матеріали для сайту, слід пам'ятати, що в Інтернеті люди читають тексти інакше, ніж у друкованому вигляді. Вони власне не читають веб-сторінки, а переглядають їх. Більшість відвідувачів роблять це, переглядаючи лише окремі слова чи речення, і не звертають уваги на інший вміст. Це означає, що необхідно відповідно оформлювати тексти, призначені для розміщення в Інтернеті: виділяти найголовніше, використовувати заголовки та списки, чітко структурувати текст і дотримуватися тематики, а також завжди розраховувати на у два рази менший обсяг тексту порівняно з друкованим.

Часто сайти створюються за певною структурою, розробленою автором. При цьому зовсім не береться до уваги те, що буде цікавим для відвідувачів. Варто дати їм певне завдання, залучити до анкетування, опитування, розгадування загадок, участі в конкурсі — це не лише підвищить зацікавленість відвідувачів, але й дасть можливість отримати від них певну інформацію для подальших контактів.

Незалежно від способу реалізації сайту — вручну чи за допомогою веб-редакторів — варто створити сторінку-шаблон. Вона буде основою для розробки всіх подальших сторінок сайту. Така підготовка забезпечить єдиний принцип побудови веб-сайту. Крім того, за допомогою шаблону легко буде додавати до проекту нові сторінки, а також залучати до роботи колег-розробників.

Лише після всієї підготовчої роботи можна розпочинати створення веб-сторінок із наявних матеріалів. Завдяки попереднім етапам розробки на цьому етапі непотрібно переробляти графічні меню через те, що забули внести в структуру один із розділів (що найчастіше трапляється з недосвідченими розробниками). Якщо ви знаєте, що отримаєте в результаті роботи, і маєте підготовані матеріали, саме компонування не повинно викликати у вас значних труднощів, хоча й доведеться застосувати знання мови HTML або використати візуальний редактор сайтів.

Після того як веб-сайт буде скомпоновано, необхідно його протестувати — перевірити роботу всіх гіперпосилань, відображення графічних матеріалів, швидкодію сайту, його вигляд у вікнах різних браузерів, з різними налаштуваннями шрифтів та властивостей екрану. Лише після цього сайт можна розміщувати в Інтернеті.

Варто подбати про реєстрацію сайту в пошукових системах. Можна також опублікувати адресу в тематичних форумах чи спільнотах, обмінятися гіперпосиланнями із сайтами колег чи однодумців.

Після опублікування сайту в Інтернеті роботу над ним не можна вважати завершеною. Майже кожен сервер дає змогу стежити за кількістю відвідувачів та їхніми діями на сайті. Аналізуючи цю інформацію, можна вдосконалити сайт, зробити його більш зручним, корисним і популярним. Крім того, розміщена на ньому інформація потребує оновлення, уточнення, змінення чи інших дій, що змушує постійно стежити за станом сайту.

З часом можуть змінюватися дизайн сайту, його структура і навіть призначення. Варто взяти собі за правило зберігати архівні версії сайтів. Зазвичай їх можна створювати автоматично за допомогою спеціальних програм, що містяться на сервері.

Веб-дизайн у контексті художнього дазайну

У попередніх розділах ми розглянули процес створення веб-сайту та його розміщення в мережі Інтернет. Проте цієї інформації недостатньо для організації якісного ресурсу. Зокрема, важливо знати, що веб-сайти поділяються на різні категорії, що впливає на їхнє оформлення та функціонування. Всі ці аспекти необхідно враховувати у веб-дизайні.

Дизайн — це художнє конструювання предметного світу, розробка зразків раціональної побудови предметного середовища; це творча діяльність, спрямована на зовнішній вигляд предметів, а також на їх структурні та функціональні взаємозв'язки, які роблять їх цілісними. Термін *веб-дизайн* означає розроблення та конструювання ресурсів Інтернету.

Основні поняття дизайну

Основними образотворчими засобами дизайну є точка, лінія, фактура, текстура, колір, форма, пропорція тощо.

Точка — базовий елемент, який не має вимірів і характеризується місцем розташування. Сама собою точка може бути вдалим

розділювальним елементом, а поєднання кількох точок формує лінію (рис. 7.5).



Рис. 7.5. Точки: одна; кілька точок, що утворюють лінію; кілька точок, що утворюють трикутник

Лінія — це фігура, яка утворюється рухом точки і може мати як фіксовані, так і змінні напрямки і товщину. Лінії створюють обриси та форми. Їх також використовують для відображення перспективи, а домінуючі лінії надають відчуття єдності композиції. Крім того, лінії можуть поєднуватися для імітації текстури, густини чи об'єму (рис. 7.6).



Рис. 7.6. Лінії: природні; прями; різної товщини

Текстура — характер поверхні певного об'єкта, зумовлений його внутрішньою будовою, структурою, об'єктивними фізичними властивостями (дерево, метал, скло, тканина тощо). Текстура стосується як візуальних, так і тактильних (відчутних на дотик) властивостей об'єкта і найчастіше визначається матеріалом, з якого його виготовлено (рис. 7.7).

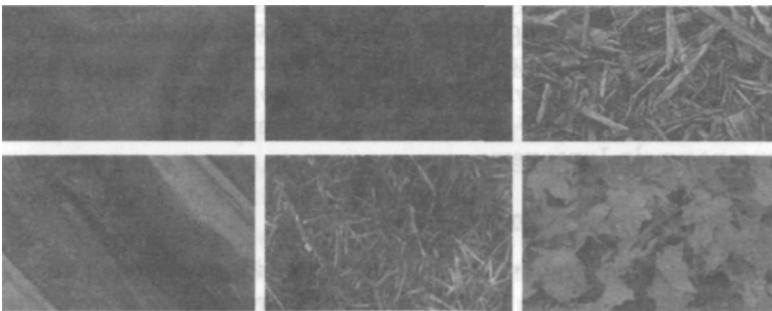


Рис. 7.7. Текстури: шовк, каміння, тирса, дерево, соснові голки, осінні листя

Близькою до цього поняття є *фактура* — зовнішні властивості поверхні об'єкта (гладка, дзеркальна, тьмяна); вона характеризує якість обробки поверхні.

Форма — це зовнішні обриси, вигляд, контур об'єкта. Є три базові форми: коло, прямокутник (квадрат) та трикутник. Форма може бути двовимірною та тривимірною, реалістичною, абстрактною та проміжно (рис. 7.8).

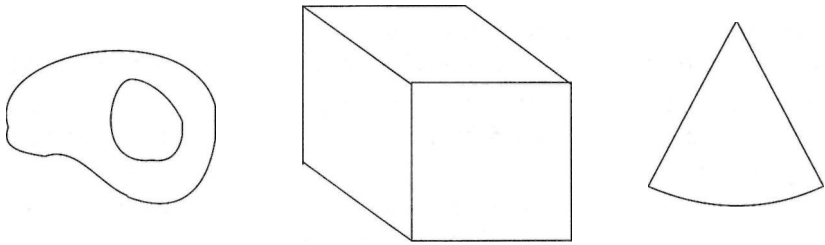


Рис. 7.8. Форма: контур та тривимірні форми

Колір — властивість тіл викликати певне зорове відчуття відповідно до спектрального складу та інтенсивності видимого випромінювання, яке ними відбивається чи поглинається. Основні параметри кольору — тон, насиченість та ясність. Тон визначається спектральним складом світла, насиченість — кількістю сірого тону, а ясність — яскравістю.

Пропорція — це співвідношення складових у межах одного цілого. Розрізняють арифметичну, геометричну та гармонійну пропорції. Найвідомішою дизайнерською пропорцією є так званий «золотий переріз» — поділ відрізка на частини у такому співвідношенні, що менша частина відноситься до більшої так, як більша — до цілого відрізка (приблизно це $1/1,618$, а число $1,618$ називають значенням золотого перерізу).

Перелічені базові елементи творчо поєднуються між собою, ґрунтуючись на основних принципах композиції — симетрії, асиметрії, рівноваги, ритму та динаміки.

Композиція — це об'єднання різних частин (елементів) у єдине ціле відповідно до певної ідеї. Елементи художнього витвору повинні перебувати у гармонійній єдності. Композиція є показником художньої культури дизайнера, його почуття міри та стилю.

Симетрія — таке розташування об'єктів, за якого вони розміщуються дзеркально відносно центральної осі художнього витвору (у нашому випадку — веб-сторінки). Симетричні композиції характеризуються статичністю та стійкістю, ліва і права частина зображення є врівноваженими. Порушення симетрії створює відчуття неспокою, динаміки, складності.

Асиметрія — довільне розташування об'єктів, за якого ліва та права частини художнього витвору стають неврівноваженими.

Рівновага — стан композиції, за якого всі елементи збалансовані між собою. Рівновага викликає відчуття спокою та впевненості.

Динаміка — ілюзія руху, створена за допомогою статичних зображень спеціальними засобами. У веб-дизайні ефекту руху досягають анімацією.

Ритм — чергування різних елементів, яке додає композиції чіткості, стрункості, особливої виразності, динамічності.

Окрім зазначених вище, у дизайні використовують ще й такі поняття, як гармонія, контраст, колорит, перспектива, центр мас.

Гармонія — співмірність всіх частин цілого.

Контраст — різка відмінність елементів композиції, наприклад, протиставлення темного і світлого, високого та низького, легкого та важкого, тонкого та товстого, ліній та плям.

Колорит — гармонійне поєднання, взаємозв'язок, тональне об'єднання різних кольорів.

Перспектива — спосіб зображення тривимірного простору на площині відповідно до видимих змін розмірів та чіткості об'єктів. Завдяки художній перспективі зображення стає об'ємним.

Центр мас — візуальний центр зображення, що міститься дещо вище від його геометричного центру. Це природне місце розташування інформаційного центру композиції.

Ергономіка

Ергономіка (від гр. *ergon* — робота та *nomos* — закон) — наука, яка вивчає робочі процеси з метою створення оптимальних умов праці, що сприяють підвищенню її продуктивності, а також зберігають сили, здоров'я та працездатність людини.

Для системи «людина-комп'ютер» з аналізаторів сприйняття інформації найбільше значення має зоровий. Ергономічний сайт ґрунтується на знаннях про будову та функціонування людського ока, яке, дивлячись на екран, збирає інформацію з обмеженого кута зору. Розглянемо основні елементи ергономічного візуального інтерфейсу.

По-перше, він відображається на екрані монітора і складається з ділянок, частина яких є *найактивнішими*, тобто відразу стають помітними, виділяючи найважливіші елементи сторінки.

По-друге, *колірна схема* елементів інтерфейсу має бути побудована на основі психофізіології кольору. Зокрема, червоний колір — сигнал небезпеки, зелений — сигнал штатної ситуації тощо. По-третє, гарнітуру *шрифту*, його розмір, накреслення, а також варіанти розташування текстових блоків на сторінці необхідно вибирати відповідно до ступеня їхньої важливості для цільової аудиторії.

Нарешті, система *навігації* має бути зручною та інтуїтивно зрозумілою.

І останнє (за порядком, але не за значенням): інтерфейс програми повинен адекватно моделювати *реальні життєві ситуації*, що виникають у тій чи іншій прикладній галузі.

Інтерфейс має бути непомітним настільки, щоб відвідувач зосереджував свою увагу на вмісті веб-сторінки, а не на способі її подання.

Основні складові веб-дизайну

Веб-дизайн — це розділ дизайну, що стосується конструювання веб-ресурсів і поєднує елементи художніх, технологічних, комерційних та інформаційних наук. Термін складається з двох частин: «веб» — скорочена назва Всесвітньої павутини та «дизайн» — слово, яке означає проектування, конструювання, планування, креслення, малювання, процес створення нових предметів так, щоб їхня форма відповідала змісту. Отже, термін «веб-дизайн» означає проектування для Інтернету.

Можна виділити п'ять основних складових поняття веб-дизайну.

- *Зміст* — форма подання та спосіб організації наповнення сайту: стиль написання тексту, його упорядкування, форматування та структурування за допомогою мови HTML.

- *Оформлення* — загальний вигляд сайту, що визначається зокрема графічними елементами для прикраси та навігації.
- *Технологи* — це засоби форматування веб-сторінок та надання їм інтерактивності, наприклад, мови HTML та JavaScript, таблиці CSS, анімація Flash.
- *Подання* — швидкість та надійність відображення сайту в мережі Інтернет.
- *Мета* — причина створення сайту, а також результати, яких він дозволить досягти.

У веб-дизайні можна виділити кілька основних дилем, які постають під час розроблення майже всіх веб-сайтів. Веб-майстрові необхідно знайти хиткий баланс між:

- власними можливостями та потребами користувачів;
- формою та функціями веб-сайту;
- усталеними правилами та творчими рішеннями;
- якістю оформлення веб-сайту і часом на його створення.

Дизайн веб-сайту має відповідати потребам користувачів — це правило є дуже важливим, адже саме для користувачів і створюють цей продукт. Проте досить часто веб-розробники забувають про своїх клієнтів і захоплюються реалізацією власних творчих рішень.

Поширеною проблемою у веб-дизайні є невідповідність між дизайном та функціональністю. Функціональний, але погано оформлений сайт переглядати не цікаво: він не захоплює відвідувача. Те саме можна сказати й про сайт, який має гарне оформлення, але не корисний для відвідувачів. Тому потрібно знайти рівновагу між цими характеристиками, тобто дизайн має відповідати функціональності. Досвідчені веб-розробники підтримують ідею цілісного дизайну: це означає, що першочергова увага приділяється вдалому поєднанню всіх елементів і функцій сайту.

Користувачі комп'ютерів вже звикли до певних елементів, які вони бачать у своїх програмах та на веб-сторінках. Справді, коли створюють сайт, часто дотримуються певних правил: логотип організації розміщують у верхньому лівому кутку, меню розташовують вгорі, зліва чи знизу тощо. Дизайнер повинен дотримуватися

загальноприйнятих правил навігації, колірних гам тощо. Вони не обмежують можливості дизайну, а роблять сайти зручними для відвідувачів.

На сайті не може бути помилок. HTML-код слід писати відповідно до правил та рекомендацій, графічні елементи повинні відображатися правильно. Всі інтерактивні елементи мають працювати без перебоїв і повідомлень про помилки. Гіперпосилання сайту мають забезпечувати перехід до відповідних матеріалів.

Проектування структури сайту

Після визначення мети створення сайту і врахування наведених вище порад можна розпочинати проектування структури сайту, насамперед його системи навігації. Якщо відвідувач не зуміє зорієнтуватись у ній, він просто залишить веб-сайт. Тому варто подумати над схемою гіперпосилань, які поєднують окремі сторінки в одне ціле.

Термін *навігація* означає систему, схему пересування у просторі. Цим терміном можна назвати схему пересування інформаційним простором як усього Інтернету, так і окремого сайту.

Кожен веб-сайт повинен містити відомості про автора, контактну інформацію, а також повідомлення про права інтелектуальної власності на вміст ресурсу. Корисним є повідомлення про дату останнього оновлення веб-сайту: це допомагає відвідувачам оцінити достовірність, надійність та актуальність наведеної інформації.

Вибір інформаційної архітектури

Інформаційна архітектура сайту — це сукупність методів та способів створення структури з веб-сторінок. Можна виділити такі різновиди інформаційних архітектур сайту.

Все в одному. Це найпростіша модель: всю інформацію розміщують на одній сторінці (рис. 7.9). Якщо матеріалу для сайту не дуже багато і його можна розмістити на одній сторінці, то зазвичай так і слід зробити — не варто ускладнювати структуру вигаданими розділами та непотрібними гіперпосиланнями. Проте на такій сторінці обов'язково потрібно розмістити всю необхідну інформацію, зокрема контактну.

Сайт-сторінка

Рис. 7.9. Інформаційна архітектура «все в одному»

Однорівнева модель. Всі сторінки є рівнозначними, і на кожну можна перейти з будь-якої іншої (рис. 7.10). Це найпоширеніша модель для простих сайтів, які містять невелику кількість розділів, наприклад, «Головна сторінка», «Про нас», «Контакти», «Товари».



Рис. 7.10. Однорівнева інформаційна архітектура

Індекс. У цій структурі є головна сторінка з посиланнями на всі інші (рис. 7.11). Вона часто містить список матеріалів, які можна знайти на сайті. За таким принципом можуть бути реалізовані телефонні довідники, енциклопедії чи інші сайти подібного типу.



Рис. 7.11. Інформаційна архітектура «індекс»

Ромашка. Ця модель є корисною для подання чітких лінійних залежностей, коли одна дія чітко має йти за іншою, а всі вони в будь-якому разі починатимуться на головній сторінці (рис. 7.12). Прикладом сайту з такою структурою є сервіс електронної пошти, де відвідувач може створити лист, а у процесі його написання прикріпити до нього файл. Цю дію не можна виконувати на головній сторінці, а лише після переходу до написання листа.

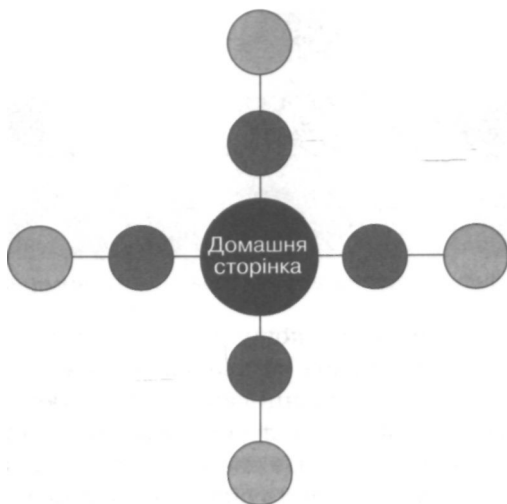


Рис. 7.12. Інформаційна архітектура «ромашка»

Чітка ієрархія. На сторінку нижчого рівня можна потрапити лише через її «батька» (рис. 7.13). Прикладом використання такої архітектури є сайт розподілу офісів певної компанії по містах України, адже кожен офіс може бути розташований лише в одному місті. Так само і гілка обговорень у форумі стосується однієї теми, водночас одна тема може мати кілька гілок обговорення.

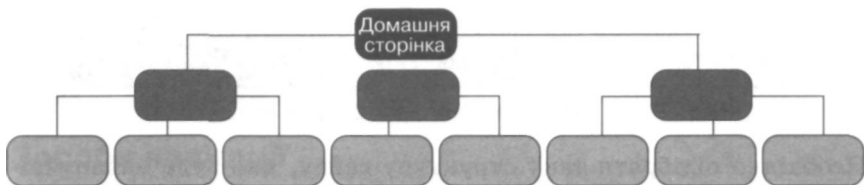


Рис. 7.13. Інформаційна архітектура «чітка ієрархія»

Багатовимірна ієрархія. За такої організації вмісту сайту відвідувач може знайти той самий матеріал різними способами, тобто реалізовано кілька ієрархій, які перетинаються між собою (рис. 7.14). Типовим прикладом такої архітектури є сайт Амазон (<http://www.amazon.com/>), за допомогою якого виконувати пошук книжки можна за жанром, автором чи назвою твору.

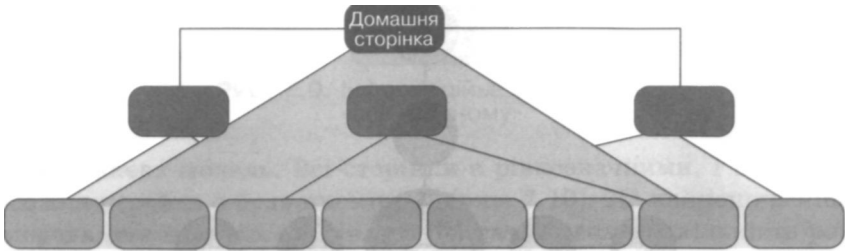


Рис. 7.14. Інформаційна архітектура «багатовимірна ієрархія»

Пошукова система. Функція пошуку часто може бути використана як навігаційна можливість чи елемент архітектури (рис. 7.15). Відвідувачу на головній сторінці сайту пропонують виконати пошук, після чого йому надають перелік сторінок сайту, що відповідають заданим критеріям. Це дає змогу відразу переходити до шуканих матеріалів, не користуючись ієрархіями чи індексами. Проте така функція потребує досить якісного пошукового механізму, який має бути вбудований у сайт.

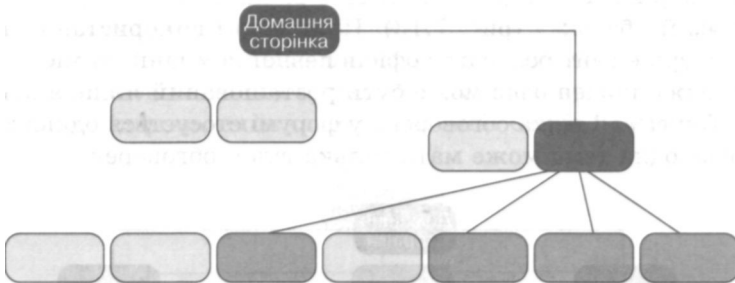


Рис. 7.15. Інформаційна архітектура «пошукова»

Необхідно підібрати таку структуру сайту, яка буде найзручнішою та зрозумілою для відвідувача. Після цього слід визначитися з її проектом, беручи до уваги такі фактори:

- **доступність** — навігація є одним із найважливіших елементів будь-якої сторінки, тому необхідно зробити її доступною в будь-який момент, тобто навігація має бути реалізована (або принаймні дубльована) простими HTML-кодами, а не ефектами Flash, Java, JavaScript, які можуть відтворюватися не на всіх комп'ютерах відвідувачів;

- **ЗМІСТОВНІСТЬ** — **ЗМІСТ** гіперпосилань має відповідати змісту ресурсів, на які вони посилаються, щоб людина, яка вперше потрапила на сайт, розуміла його структуру;
- **видимість** — гіперпосилання повинні виділятися з навколишніх елементів у певний спосіб;
- **цілісність** — можливість навігації слід реалізувати на всіх сторінках сайту (не обов'язково за допомогою однакових меню, але засоби навігації повинні бути впізнавані).

Часто виникає проблема деталізації структури, тобто питання про те, на скільки блоків слід розбивати вміст сайту. Під час її вирішення слід пам'ятати про одну особливість веб-сторінок: *відвідувачі не люблять горизонтальних лінійок прокручування!* Якщо вертикальні лінійки допустимі, то в жодному разі не можна змушувати відвідувачів гортати текст горизонтально. Нормальним вважають розташування матеріалу на 2-3 екранах із вертикальним прокручуванням.

Крім того, слід враховувати й *розміри файлів веб-сторінки*. Відвідувач розраховує на те, що вона завантажуватиметься не довше десяти секунд. Якщо це відбувається упродовж однієї секунди, він не відчуває дискомфорту, якщо до десяти секунд — не відволікається на інші завдання. Коли процес очікування триває довше зазначеного терміну, відвідувач починає відкривати інші сторінки і навряд чи дочекається повного завантаження першої. Це також необхідно враховувати, поділяючи матеріал на частини. Таке обмеження найбільше впливає на мультимедійні елементи — графіку, аудіо- та відео-матеріали, тому варто оптимізувати їхній розмір перед розміщенням в Інтернеті.

Засоби навігації

Інтернет є навігаційною системою: основна взаємодія з ним відвідувачів полягає у використанні гіперпосилань для переміщення в інформаційному просторі. Система ця є дуже складною — кожен сайт містить десятки, а то й сотні гіперпосилань. Одні з них відкривають інші сторінки того самого сайту, другі — забезпечують перехід на інші сайти, а треті визначають переміщення у межах однієї сторінки. Відвідувач завжди повинен знати, де він перебуває, де вже побував і куди може перейти.

Показчик місця перебування. Користувач має чітко ідентифікувати своє місцеперебування на кожній сторінці сайту. Яке б гіперпосилання не вибрав відвідувач, він повинен розуміти, що перебуває на певному сайті і ще не вийшов за його межі. Це означає, що на кожній сторінці варто розмістити логотип чи загальну назву сайту. Крім того, потрібно продумати меню і назви сторінок так, щоб підказати відвідувачу, в якому саме розділі сайту він перебуває.

Відвідані сторінки. Браузери мають кнопку **Назад**, яка завжди може повернути відвідувача на одну із вже відвіданих сторінок, а також забезпечують можливість виділення відвіданих гіперпосилань іншим кольором. Ця проста функція є також дуже цінною, оскільки відвідувач може не заходити на вже переглянуті сторінки, що економить його час та кошти.

Важливе значення також мають **назви веб-сторінок**, вказані у тегу <TITLE>. Саме під ними зберігаються адреси сторінок у папці **Вибране**, вони відображаються у заголовку вікна браузера, а також у меню кнопки **Назад**. Тому доцільно, щоб заголовок точно описував назву, зміст і призначення сторінки.

Подальший перегляд. Щоб відвідувач міг визначитися з тим, куди він бажає перейти з певної сторінки, він повинен розуміти загальну структуру сайту та бачити гіперпосилання, які б відповідали його потребам. Можна виділити кілька навігаційних рішень, які дадуть відвідувачу змогу швидко та ефективно орієнтуватись у структурі сайту.

- *Зміст сторінки* — перелік тем, які містить ця сторінка, переважно розміщений на її початку. Здебільшого назви тем роблять гіперпосиланнями-якорями, які після їх клацання відкривають відповідний розділ.
- *Горизонтальне меню* — відвідувач читає сторінку згори донизу, тож цілком логічно розмістити вгорі сторінки меню, яке дасть змогу орієнтуватись у змісті сайту (рис. 7.16). Створюючи горизонтальні меню, необхідно враховувати те, що не можна перевищувати ширину сторінки. Використання такого меню доречно у випадку, коли невелика кількість його пунктів визначена заздалегідь, і воно не змінюватиметься.
- *Рядок навігації* — відображає весь шлях від домашньої сторінки сайту до тієї, на якій перебуває відвідувач у поточний

момент (рис. 7.17). Так, клацнувши будь-яке гіперпосилання з навігаційного рядка, відвідувач може перейти на певний рівень структури сайту.

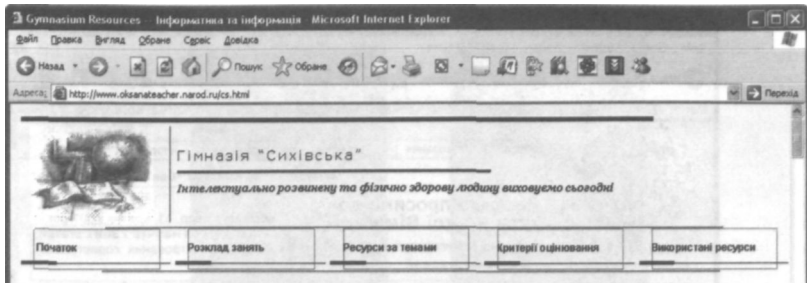


Рис. 7.16. Горизонтальне меню сайту вивчення інформатики у гімназії «Сихівська» (<http://www.astra.lviv.ua/cs/>)

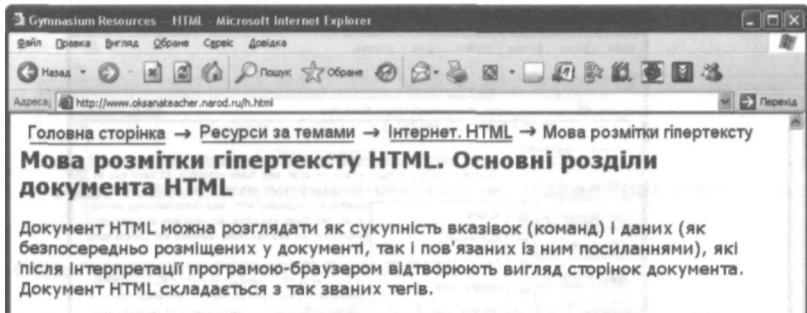


Рис. 7.17. Рядок навігації сайту вивчення інформатики у гімназії «Сихівська»

Закладки — це різновид горизонтального меню (рис. 7.18). Вони вказують на активний у поточний момент розділ сайту і є реалістичнішими порівняно з меню, адже саме закладки використовують у паперових папках для позначення різних тематик матеріалу.

Бокове меню — є найпоширенішим рішенням для сайтів. Здебільшого його використовують тоді, коли навігація має лише один рівень і є достатньо сталою.

Спливаюче меню — принцип його дії полягає в тому, що відвідувач клацає потрібний пункт, після чого з'являється меню

наступного рівня (рис. 7.19). Іноді це відбувається без клацання — лише після наведення вказівника миші. Інколи таке меню може мати кілька рівнів.

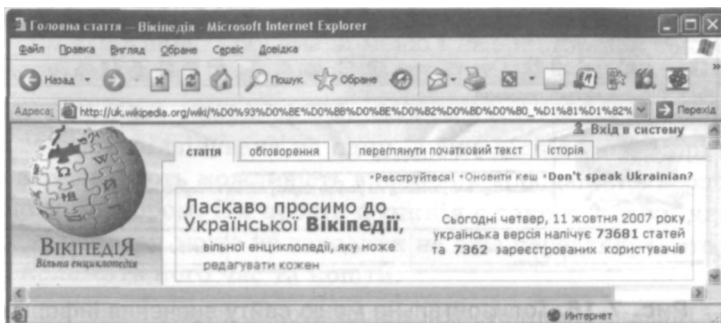


Рис. 7.18. Закладки у дизайні сайту uk.wikipedia.org

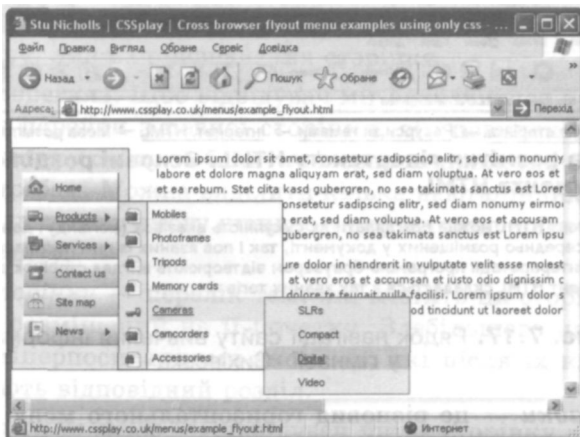


Рис. 7.19. Спливаюче меню на сайті www.cssplay.co.uk

- *Багаторівневе дерево навігації* — схоже на традиційні меню програм; на екрані відображається вся ієрархічна будова сайту, і відвідувач легко може зробити вибір (рис. 7.20). Така реалізація дає змогу досить чітко й зрозуміло відобразити складні структури веб-сайтів.
- *Сторінки* — матеріал поділяють на кілька сторінок (зазвичай це роблять із довгими статтями, індексами, форумами

чи результатами пошуку) (рис. 7.21). Відвідувачу надають можливість пересуватися сторінками, клацаючи їх номери або буквені позначення. Крім того, зазвичай створюють гіпер-посилання на попередню та наступну сторінку.

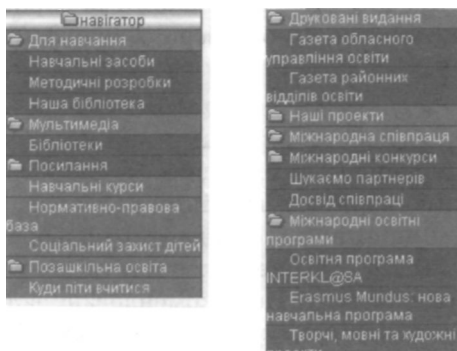


Рис. 7.20. Багаторівневі дерева навігації

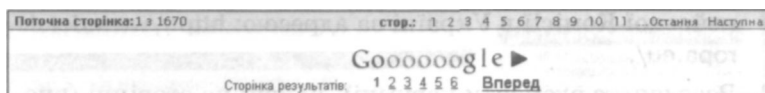


Рис. 7.21. Використання сторінок як навігаційних елементів

Проектування гіперпосилань

Гіперпосилання — це найважливіша частина гіпертексту і «цементуючий матеріал» Інтернету. Посилання має сповіщати відвідувача про те, що той отримає, клацнувши його. Підкреслений текст повинен визначати зміст об'єкта, до якого буде здійснено перехід.

Одне з найстаріших правил веб-дизайну застерігає від використання таких гіперпосилань, як «клацніть тут». Замість фрази:

«Для ознайомлення з інформацією про історію мережі Інтернет клацніть тут»

краще писати:

«Ви можете ознайомитися з додатковою інформацією про історію мережі Інтернет».

Якщо порівняти ці дві фрази, то побачимо, що словосполучення «клацніть тут» не несе жодного змістового навантаження, воно нічого не означає. Краще, щоб текст гіперпосилання інформував про тематику кінцевого об'єкта переходу. Це особливо важливо для відвідувачів, які мають проблеми із зором і користуються голосовими браузерерами. Ці браузери озвучують текст, що відображається на екрані, для тих, хто не може бачити його.

Більшість браузерів використовує два кольори для відображення гіперпосилань — ще не відвідані сторінки за умовчанням виділяють синім кольором, а відвідані — фіолетовим. Дуже часто веб-розробники використовують інші кольорові гами, проте якщо прийнято рішення змінити стандартні кольори, це слід зробити якомога зрозуміліше. Краще використати яскравіший та тьмяніший відтінки одного тону. Не рекомендовано встановлювати однакові кольори для відвіданих та невідвіданих гіперпосилань.

Вправа 7.2

1. Відкрийте у браузері головну сторінку Представництва Європейської Комісії в Україні за адресою: <http://www.delukr.ec.europa.eu/>.
2. Розгляньте елементи навігації, наявні на сторінці (рис. 7.22). Тут міститься одночасно кілька їх різновидів: навігаційний рядок у заголовку сторінки, горизонтальне меню (Головна сторінка, Контакти, Мапа сайту. Посилання, Вакансії, Стажування), спливаюче меню (Про нас, ЄС та Україна, Інформаційні джерела, Конкурси та гранти, Що таке ЄС, Що нового, Актуальна інформація. Зовнішні відносини ЄС). Після вибору будь-якого посилання відбувається перехід на сторінку, що містить відповідну інформацію.
3. Відкрийте по чергово сторінки за такими адресами:
 - <http://www.astra.lviv.ua/cs/>;
 - <http://uk.wikipedia.org/>;
 - <http://www.apple.com/>;
 - <http://www.alex.poninska.pl/>;
 - <http://news.bbc.co.uk/>;
 - <http://ostriv.in.ua/>;
 - <http://leoburnett.ca/>

та занотуйте типи навігаційних елементів і способи оформлення гіперпосилань, які використовують на цих сторінках.

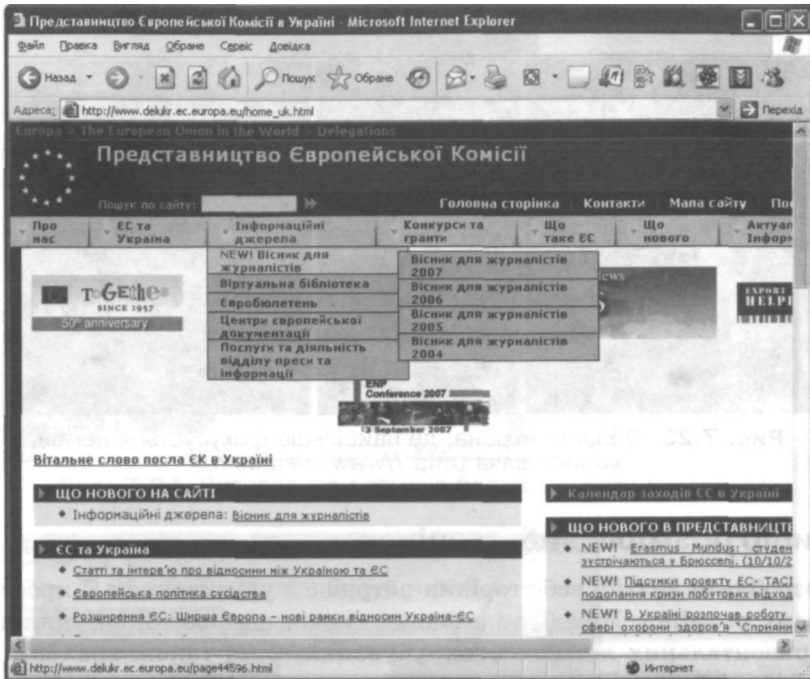


Рис. 7.22. Навігаційне вирішення сайту Представництва Європейської Комісії в Україні

Просторовий дизайн веб-сторінок

Згідно з дослідженнями користувач найбільше часу приділяє розгляду F-подібної області екрана (рис. 7.23), зазвичай зосереджується на виділених світлим кольором областях і майже не звертає уваги на ті, що мають інший колір.

Спочатку користувач розглядає верхню частину екрана по горизонталі, потім переводить свій погляд на текст, що завдовжки дещо менший за верхню частину. Ці дослідження свідчать про те, що відвідувачі сайту не читають веб-сторінок, а лише переглядають їх.

Тому дуже важливо, щоб основна інформація була зосереджена на початку сторінки, а для виділення важливої інформації та привернення уваги відвідувача слід використовувати заголовки, списки та поділ тексту на абзаци.

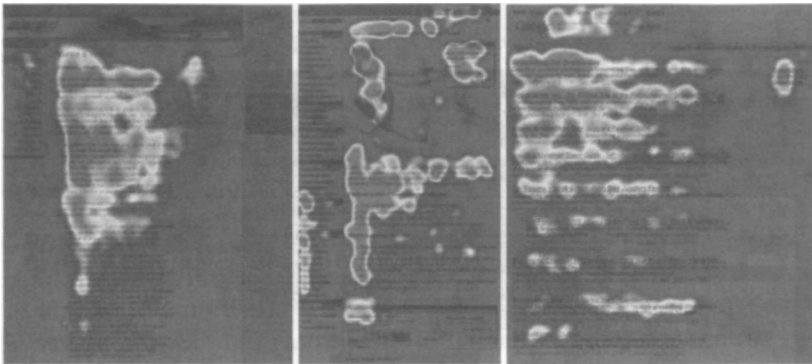


Рис. 7.23. Область екрана, де найбільше фокусується погляд користувача (<http://www.useit.com/>)

Вибір розмірів веб-сторінок

Розробляти дизайн веб-сторінки потрібно з урахуванням *Ti* просторових розмірів. Вже було згадано про те, що необхідно уникати горизонтальних лінійок прокручування. Слід також враховувати, що веб-сторінки можуть відображатися на моніторах із роздільною здатністю 800x600 пікселів. Якщо відняти простір, який займають меню браузера та панелі операційної системи, то для показу сайту залишиться область приблизно 760x410 пікселів. Це безпечний розмір сторінки, у разі використання якого сторінка коректно відобразатиметься на екранах комп'ютерів більшості відвідувачів.

Є різні способи визначення ширини сторінки та її елементів. Передусім можна задавати кількість пікселів, відведених для різних елементів (рис. 7.24).

Можна визначати загальну ширину елементів сторінки відносно ширини вікна, при цьому решту простору слід заповнити певним кольором. Розміри окремих елементів можна також задавати у відсотках відносно загальної ширини (рис. 7.25).

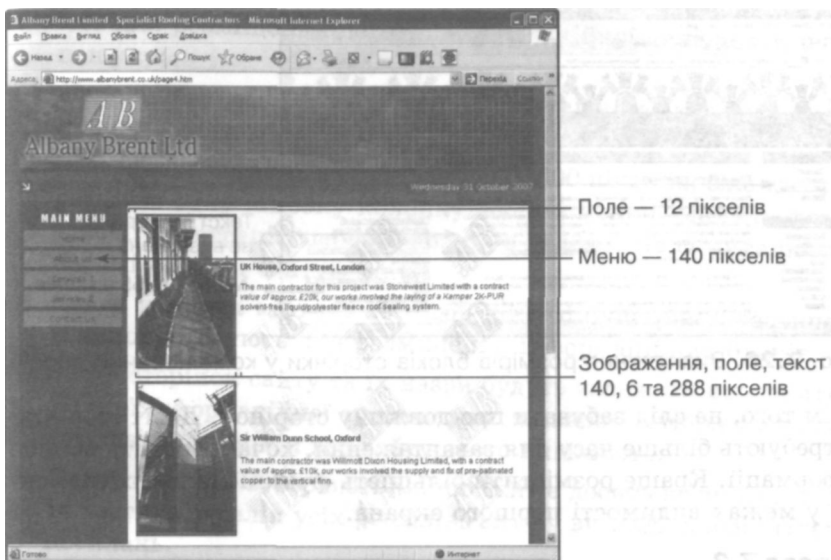


Рис. 7.24. Визначення розмірів блоків сторінки у пікселях

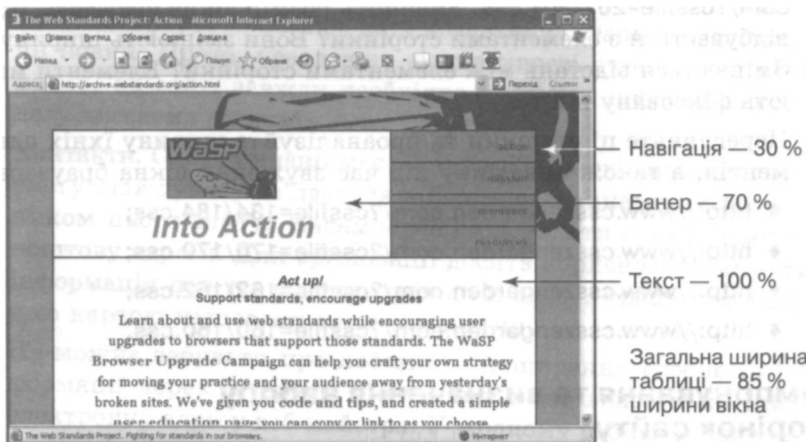


Рис. 7.25. Визначення розмірів блоків сторінки у відсотках

Визначати розміри блоків сторінки можна й у комбінований спосіб (рис. 7.26). Якщо не задавати загальної ширини таблиці, вона займатиме все місце, доступне у відкритому вікні.

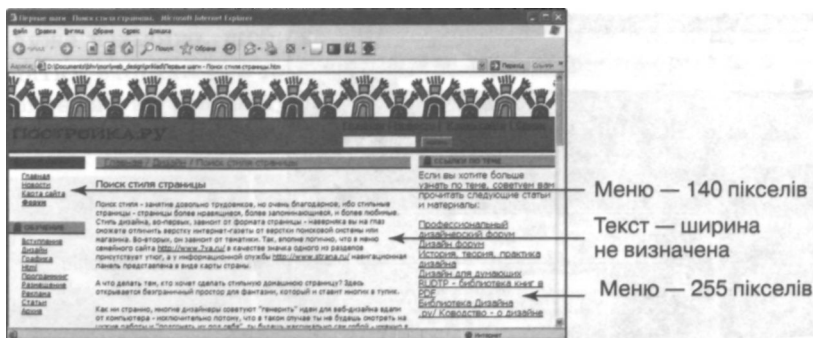


Рис. 7.26. Визначення розмірів блоків сторінки у комбінований спосіб

Крім того, не слід забувати про довжину сторінок. Довгі сторінки потребують більше часу для завантаження, хоча й містять більше інформації. Краще розмістити більшість функціональних елементів у межах видимості першого екрана.

Вправа 7.3

1. Відкрийте веб-сторінку за адресою: <http://www.csszengarden.com/?cssfile=201/201.css>. Зменшіть розмір вікна браузера. Що відбувається з елементами сторінки? Вони змінюють ширину? Змінюється відстань між елементами сторінки? Елементи мають фіксовану ширину?
2. Перегляньте ці сторінки та проаналізуйте ширину їхніх елементів, а також поведінку під час звуження вікна браузера:
 - <http://www.csszengarden.com/?cssfile=184/184.css>;
 - <http://www.csszengarden.com/?cssfile=170/170.css>;
 - <http://www.csszengarden.com/?cssfile=162/162.css>;
 - <http://www.csszengarden.com/?cssfile=160/160.css>.

Компонування та визначення набору сторінок сайту

Більшість головних сторінок сайтів у Інтернеті мають таке компонування:

- **вгорі** розташовується горизонтальна панель, переважно кольорова, заввишки до **100** пікселів, яку зазвичай використовую-

ють для розміщення логотипа, навігаційної панелі та панелі пошуку;

- **ліва колонка** завширшки 200-250 пікселів має кольорове тло, зазвичай вона містить навігаційне меню;
- **центральна частина** має ширину 400-600 пікселів, біле тло, переважно одну широку колонку або дві менші, містить фотографії, ілюстрації, перелік новин, а також гіперпосилання на окремі розділи сайту;
- **права колонка** містить 200-250 пікселів у ширину; зазвичай її використовують для розміщення інформаційних блоків.

Кількість сторінок сайту та їх назви будуть залежати від тематики, обраної інформаційної архітектури, кількості матеріалів тощо. Крім головної сторінки веб-сайту часто виділяють:

- **Карту сайту.** Її створюють, якщо сайт є досить великим. Карта містить перелік усіх розділів сайту у вигляді списку гіперпосилань.
- **Новини.** Практично на кожному сайті матеріали, які він містить, оновлюються. Їх варто вносити у відповідний розділ, щоб відвідувачі знали про дату додавання останніх публікацій чи новин. Проте, незалежно від наявності окремої сторінки новин, на сайті завжди необхідно подавати інформацію про дату внесення останніх змін.
- **Контакти.** Сайт компанії має містити контактні дані, що дасть змогу відвідувачу зв'язатися з її представниками чи розробником цього сайту. Насамперед слід подати електронну та поштову адреси. Для організації досить корисною може бути інформація про те, як дістатися до її офісу, краще з детальною картою місцевості та поясненнями.

Не можна забувати про небезпеку поширення особистої інформації через Інтернет. Достатньо лише вказати на сайті електронну адресу веб-майстра. У жодному разі не можна подавати особисті дані (навіть якщо йдеться про ваш особистий сайт), зокрема домашню адресу чи номер телефону!

- **Поширені запитання.** Це сторінка, що містить запитання, які найчастіше виникають у відвідувачів сайту, та відповіді на них. Оскільки відвідувачі як правило стикаються з однаковими

проблемами, то для того, аби уникнути повторення рекомендацій щодо їх вирішення, і створюють сторінки з поширеними запитаннями.

Якщо веб-розробник хоче, щоб його сайт відвідували люди з різних країн, варто подбати про різномовні версії всіх матеріалів сайту або принаймні їх частини. На головній сторінці в такому разі необхідно розмістити посилання на версії відповідною мовою. Воно може мати вигляд перемикача мов, зображення прапорця країни, в якій користуються певною мовою, чи спеціального меню (рис. 7.27).

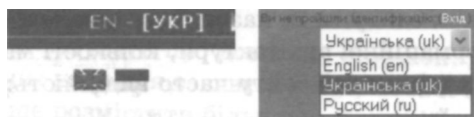


Рис. 7.27. Гіперпосилання для вибору мови у вигляді перемикача, зображення прапорців та меню

Розміщення елементів на сторінці

Спосіб розміщення елементів на сторінці є дуже важливим для її сприйняття та розуміння. Для створення правильних асоціацій застосовують різні методи групування та розділення елементів. Групи створюють за допомогою таких прийомів:

- групування — створення асоціативних зв'язків завдяки розміщенню схожих елементів поряд, розмежування груп лініями або іншими графічними елементами;
- вирівнювання — вирівнювання елементів за видимими чи невидимими лініями;
- охоплення — включення одного елемента в інший; наприклад, категорія «рік» охоплює 12 місяців;
- ритм — створення взаємозв'язку завдяки використанню стилістично однакових виділень.

Групування схожих елементів дає змогу швидко зрозуміти структуру сторінки (рис. 7.28). Людина відразу сприймає згруповані елементи як одне ціле, а елементи з-поза меж групи — як явно відмінні. Навіть на панелі інструментів браузера можна виділи-

ти групи елементів: перші п'ять кнопок відділені від наступних трьох, що означає їхнє різне функціональне навантаження.

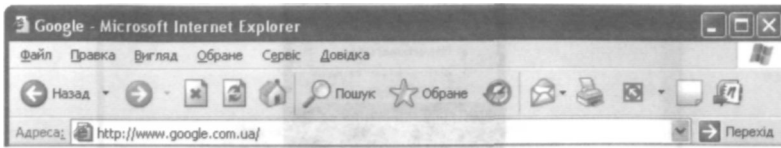


Рис. 7.28. Панель інструментів Internet Explorer: групування елементів

На сторінці може міститися одночасно кілька груп, при цьому їх можна відділяти одну від одної пробілами, кольорами або лініями.

Вирівнювання візуально поєднує кілька елементів. Цей прийом використовують для тексту, зображень, кнопок, гіперпосилань, а також для їх комбінацій. Найсильніший вплив має вирівнювання за верхнім та лівим краєм, оскільки відповідає природному сприйняттю людини. Елементи, вирівняні за правим та нижнім краєм, видаються менш значущими. Лінії вирівнювання, що починаються ближче до верхнього лівого кута екрана, є вагомішими, ніж розташовані нижче та правіше.

Розглянемо фрагмент меню із сайту www.iteach.com.ua (рис. 7.29). Основні розділи вирівняні між собою та за крайньою лівою лінією на сторінці. Гіперпосилання у підменю розміщуються дещо правіше, що робить їх менш важливими.

Охоплення — це один спосіб створення асоціативних зв'язків між елементами сторінки. Наприклад, на сторінці www.roshen.conn.ua (рис. 7.30) виділено назву категорії товарів та кілька позицій із неї. Вони не згруповані між собою, проте чітко належать до однієї групи завдяки особливому виділенню її назви вгорі.

Ритм допомагає пов'язати між собою елементи завдяки використанню стилістично однакових виділень чи розміщень. На понятті ритму, зокрема, ґрунтуються списки. Ритмом вважають і використання однакового стилю для заголовків одного рівня на всій сторінці — це додає їй цілісності.

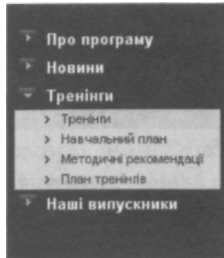


Рис. 7.29. Навігаційне меню сайту www.iteach.com.ua

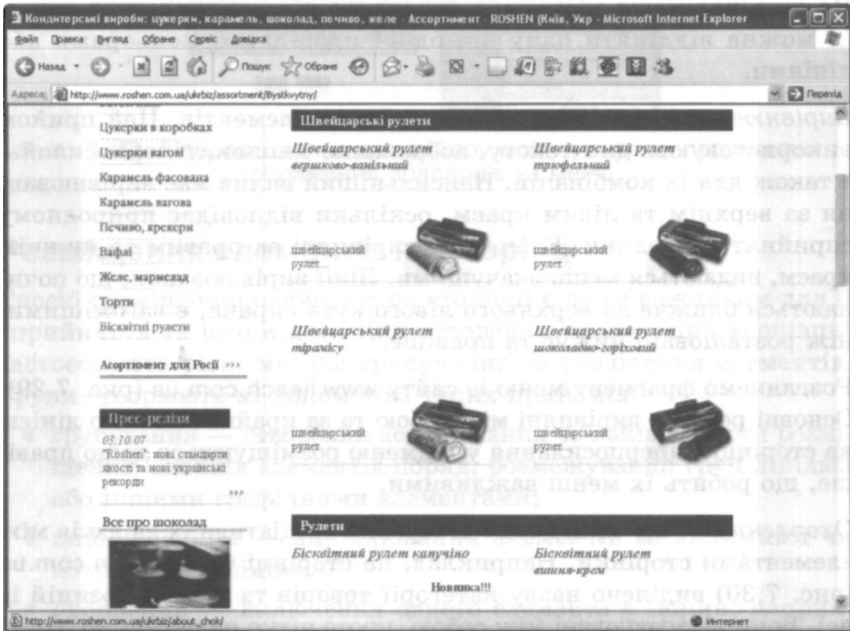


Рис. 7.30. Сторінка сайту www.roshen.com.ua

У прикладі на рис. 7.31 основні навігаційні гіперпосилання на сторінки з іншомовним наповненням не згруповані і не вирівняні. Проте вони чітко сприймаються як рівнозначні однорівневі категорії завдяки використанню однакових виділень.

До принципів, що визначають дизайн сторінки, належать також баланс, пропорція та домінантність.

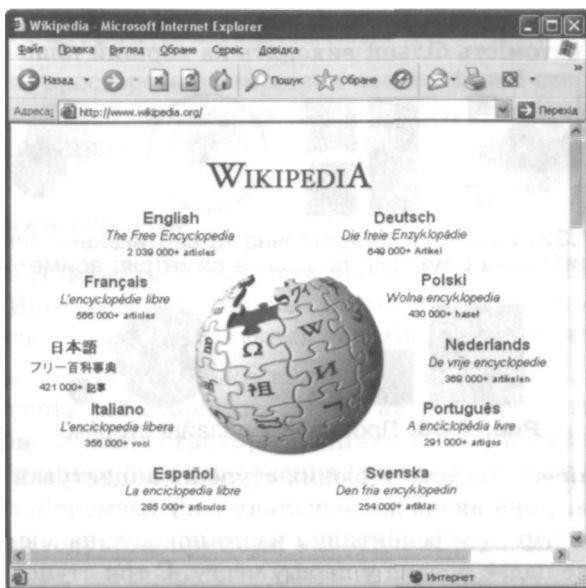


Рис. 7.31. Головна сторінка сайту www.wikipedia.org

Баланс — це рівновага зображення, що досягається розташуванням об'єктів з урахуванням їхньої візуальної ваги у композиції. Баланс буває двох видів — симетричний та асиметричний (рис. 7.32).

- *Симетричний баланс* — вага композиції рівномірно розподілена відносно вертикальної та горизонтальної осей сторінки. Зазвичай це означає наявність однакових елементів по обидва боки від осі. Коли на сторінці містяться не ідентичні, а схожі об'єкти, це називають *приблизною симетрією*. Крім того, рівновагу можна створити за рахунок радіальної (кругової) симетрії.
- *Асиметричний баланс* — вага композиції не розподілена рівномірно відносно центральних осей. Часто при цьому є один домінуючий об'єкт, якому протиставлено кілька менших. Загалом асиметрично збалансовані композиції несуть більше напруження.

Пропорція — це співвідношення між різними елементами або між цілим об'єктом та його частинами. Використання різноманітних пропорцій у композиції може встановити візуальну вагу та глибину.

а також різні форми балансу та симетрії. Менші елементи майже непомітні, натомість більші виходять на перший план (рис. 7.32).



Рис. 7.32. Баланс: горизонтальна та вертикальна симетрія; приблизна симетрія; радіальна симетрія; асиметрія



Рис. 7.33. Пропорції у дизайні сторінки

Домінантність стосується різних ступенів акцентування у дизайні сторінки. Вона визначає візуальну вагу елементів композиції, створює простір та перспективу і часто показує на місце у композиції, на яке падає погляд у першу чергу. Є три ступені доміантності, властиві кожному конкретному об'єкту в межах загальної композиції.

- *Домінантний* — елемент із найбільшим візуальним значенням і першочерговим наголосом, який явно виділяється на тлі композиції.
- *Підпорядкований* — елемент із другорядним наголосом, який міститься на тлі композиції.
- *Підлеглий* — об'єкт із найменшим візуальним значенням/навантаженням, який зливається із тлом композиції.

На рис. 7.34 дерева є доміантним елементом, будинок — підпорядкованим (вторинним), а гори — підлеглими елементами.

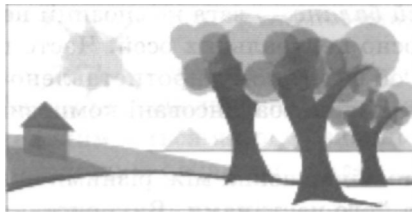


Рис. 7.34. Домінантність у дизайні сторінки

Інформаційне наповнення веб-сторінки

Визначивши просторове розміщення елементів веб-сторінки, можна переходити до її наповнення інформацією. Розглянемо правила підбору та оформлення тексту у веб-просторі.

Підготовка текстів

Текст був і залишається головним джерелом інформації в мережі Інтернет. Його подання впливає на те, як читач сприймає веб-сайт. Основні правила написання текстів для Інтернету такі.

- **Писати менше.** Оскільки відвідувачі не читають, а лише переглядають веб-сторінки, тексту на них має бути небагато. Потрібно писати стисло, лаконічно і тільки найважливіше. Чим чіткішим буде викладення, тим текст краще сприйматиметься.
- **Використовувати принцип «оберненої піраміди».** Він полягає у тому, що спершу подають найважливіші відомості, які згодом пояснюють і деталізують. Ключова інформація має розміщуватись на початку тексту, абзацу, речення.
- **Викладати найважливішу інформацію на початку сторінки.** Цей принцип набуває особливої важливості, якщо матеріал не вміщується на екран, і потрібно використовувати прокручування. Якщо важлива інформація буде розташована нижче від першого екрана, відвідувач може її просто не побачити.
- **Продумати заголовки.** Вміло складені заголовки привертають увагу. Короткий, влучний, привабливий заголовок обов'язково буде помічений відвідувачем сайту. Крім того, заголовки допомагають структурувати матеріал на сторінці, підкреслюючи логіку його побудови.

Вибір шрифтового оформлення

Після того як вибрано текстові матеріали для розміщення на сторінці, перевірено їх на відповідність зазначеним раніше правилам, підбрано влучні заголовки, варто продумати візуальне оформлення тексту. Ознайомимося з термінами, які застосовують у роботі зі шрифтами.

Гарнітура — повний комплект друкарських шрифтів різних накреслень і кеглів, але однакових за характером малюнка.

Найбільш поширеними шрифтами є шрифти із засічками або рублені. Решту можна назвати декоративними, до них належать усілякі стилізації (готичний, староруський шрифти), рукописні шрифти (каліграфічні чи навпаки, недбалі, імітації почерку), шрифти з різноманітними ефектами (тривимірні, з тінями, прикрашені кольорами тощо). Крім того, є шрифти, які замість літер містять спеціальні символи, піктограми, декоративні елементи чи зображення (рис. 7.35).

із засічками	без засічок	фіксованої ширини	декоративні	піктограми
АБВГЕ	АБВГЕ	А Б В Г Е	А В С D E F	☺ ☹ ☹ ☹ ☹ ☹

Рис. 7.35. Види шрифтів: із засічками (Times New Roman), без засічок (Microsoft Sans Serif), фіксованої ширини (Courier), декоративний (Copperplate Gothic) та піктографічний (Windings).

Однією з характеристик шрифту є його *нахил*: прямий, похилий та курсив. Іншою характеристикою накреслення шрифту є його *насиченість*, яка визначається відношенням товщини основних штрихів до висоти прямого знаку. Змінення цього відношення утворює надсвітлий, світлий, нормальний, напівжирний, жирний, наджирний шрифти, які можуть варіюватися. Декоративні шрифти рідко мають варіації насиченості та накреслення.

Шрифти із засічками підходять для тривалого читання (власне тут проявляється роль засічок — вони не дають поглядові збитися з рядка), тому ними варто оформляти основний текст сторінок. *Шрифти без засічок* мають досить лаконічний геометричний малюнок, тому їх легко охопити поглядом. Такі шрифти краще використовувати для заголовків, логотипів, коротких написів. Ці два види шрифтів добре поєднуються і не створюють на сторінці захарашеності чи дисонансу.

Не варто поєднувати різні *декоративні шрифти* на одній сторінці, вони рідко бувають сумісні між собою. Крім того, такими шрифтами складно оформити більш ніж кілька рядків тексту — їх просто незручно читати. Тому в будь-якому разі декоративні шрифти доведеться поєднувати із простішими, нейтральними (рис. 7.36).

Найкраще поєднувати шрифти із засічками та без них, даючи кожному виконувати природну функцію — відображати відповідно текст та заголовки. Шрифти із засічками (в основному тексті) мають гарний вигляд у курсивному накресленні, а шрифт без засічок (у заголовках) — у жирному.

The beginning of the line is written with one font, and this part uses a different one.

The beginning of the line is written with one font, and this part uses a different one.

The beginning of the line is written with one font, and this part uses a different one.

Рис. 7.36. Поєднання шрифтів Times New Roman, French Script, Bradley Handwriting та Arial

Після вибору шрифту для сторінки потрібно визначитися з його *кеглем* (розміром), відстанню між літерами та рядками. Чим більший кегль, тим вагомішим здається набраний ним текст. Не варто робити розмір шрифту надто малим, а тим більше розміщувати текст на тлі з дрібними деталями — його складно буде прочитати.

Вирівнювання тексту також впливає на зручність його читання. Є декілька способів вирівнювання рядків у абзаці. У HTML-документах за умовчанням застосовано вирівнювання за лівим краєм. Інший спосіб — за протилежним, правим краєм, вирізняється своїм незвичним і нетрадиційним виглядом. Вирівнювання по центру найбільш прийнятне для заголовків.

Варто зазначити, що для виділення заголовків ефективно використовувати кольорове оформлення. Іноді можна виокремити одне зі слів заголовка чи девізу, щоб привернути до нього особливу увагу.

Текстове наповнення сторінки можна також розглядати як своєрідну текстуру. Справді, якщо не вдивлятихся у конкретні літери, текстові блоки можна сприймати як графічні елементи із власною текстурою, на яку впливає колір, гарнітура, кегль, накреслення шрифту, міжзнакові інтервали тощо. Це варто враховувати під час розроблення дизайну. Для виділення окремих слів слід користуватися не жирним накресленням, а курсивом — таке оформлення сприяє кращому вигляду тексту.

Вправа 7.4

1. Прочитайте два тексти. Який із них легше запам'ятати?

Звичайний текст

У штаті Небраска знаходяться відомі на увесь світ ландшафти, які незмінно притягують до себе туристів. У 1996 році найвідвідуванішими місцями були Парк форту Робінзона (Fort Robinson State Park) (355 000 відвідувачів), Історичний музей і парк «Арбор Лодж» (Arbor Lodge State Historical Park & Museum) (100 000), Кархендж (Carhenge) (86 598), Музей піонерів прерії (Stuhr Museum of the Prairie Pioneer) (60 000) та Історичний парк «Ранчо Буффало Білла» (Buffalo Bill Ranch State Historical Park) (28 446).

Текст, пристосований до веб-простору

У 1996 році у штаті Небраска виділено шість найвідвідуваніших місць:

- Парк форту Робінзона;
 - Історичний музей;
 - Парк «Арбор Лодж»;
 - Кархендж;
 - Музей піонерів прерії;
 - Історичний парк «Ранчо Буффало Білла».
2. У текстовому редакторі наберіть обидва тексти. Використайте різні шрифти для оформлення окремих слів та рядків. Застосуйте шрифти із засічками, без засічок та декоративні. Які з них легше читати?
 3. Вирівняйте текст за лівим, правим краєм, по центру. Зауважте, що у випадку вирівнювання тексту по центру список втрачає ритмічність, стає розбалансованим.

Графічне оформлення веб-сторінки

Після підбору текстів для веб-сторінки необхідно подбати про її графічне оформлення та ілюстрації. При цьому варто враховувати деякі особливості сприйняття людиною зображень.

Сприйняття людиною форм та фігур

Психологи стверджують, що людське око бачить різні об'єкти по-різному. Щоб правильно зрозуміти властивості зображень, важливо визначити, які деталі мозок сприймає особливо чітко, а які є для нього другорядними. Проведені психологами експерименти зі сприймання різних форм та фігур дають підстави зробити декілька висновків:

- форма об'єкта пізнавана незалежно від його розмірів, кольору і розташування;
- гострий кут сприймається першим;
- прямі контури і кути сприймаються раніше ніж кола/круги.

Зоровий аналізатор сприймає елементи у такій послідовності:

1. **Горизонтальна пряма лінія** визначає орієнтацію зображення, її око вловлює найперше. Достатньо провести лінію на синьому папері, як вийде морський пейзаж.
2. **Вертикальна пряма лінія** також дає змогу відповідним чином зорієнтувати зображення і віднести його до певного смислового ряду.

На рис. 7.37 зображена яхта, що пливе по морю, а праворуч — ліхтар. Якщо придивитися уважніше, стане зрозуміло, що це одна і та сама картинка, тільки повернута на 90°. Орієнтація малюнка й аналіз, виконаний мозком на початковому етапі розгляду, відразу скеровують нас у певному напрямку. Горизонтальна лінія — асоціюється з горизонтом. Трикутник на горизонті — швидше за все, вітрило. Під час аналізу другого малюнка бачимо вертикальну лінію та симетрію — значить, щось рукотворне. Око шукає знайомі контури та асоціації: ніч, вулиця, ліхтар.

3. **Величина (відносний найбільший розмір)**. На дуже ранній стадії аналізу мозок виділяє фігури великого (значного, найбільшого порівняно з іншими спостережуваними об'єктами) розміру. Величина завжди сприймається лише як відношення до інших об'єктів зображення і не заважає упізнанню форми.
4. **Симетрія**. Для демонстрації здатності мозку до аналізу часто наводять прозорий куб-перевертень (рис. 7.38).

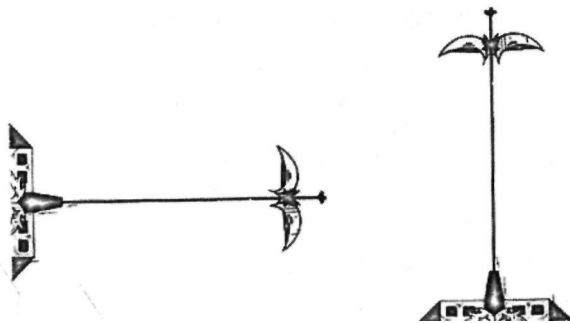


Рис. 7.37. Одне й те саме зображення, повернуте на 90°

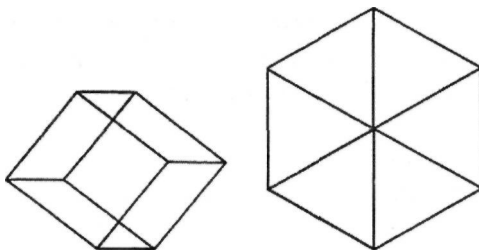


Рис. 7.38. Куб за певного повороту перетворюється на плоский шестигранник

Після зміни точки огляду куб зникає, замість нього стає видно плоский шестигранник: симетрію око схоплює миттєво, і вона пригнічує здатність перебирати інші варіанти. Симетрія лежить в основі всіх символів досконалості.

5. **Періодичність.** Із симетрією пов'язана періодичність структур, яку наш мозок також схильний відшукувати на зображенні.
6. **Похила пряма лінія.** Слід зазначити, що прямі лінії (крім горизонтальної та вертикальної, про які ми вже згадували) у природі зустрічаються вкрай рідко. Саме тому око виділяє пряму лінію у зображенні, бо вона є ознакою порядку, свідомої діяльності людини.
7. **Трикутник і прямокутник.** Найпомітнішим елементом композиції є трикутник із вершиною вгорі, далі — трикутник в інших положеннях, квадрат, прямокутник та багатогранник. *Трикутник* — особливий об'єкт, він має чітку спрямованість

та внутрішню динаміку. Зазвичай у веб-дизайні його використовують для позначення напрямку руху, стрілки, маркера списку чи кнопки.

Прямокутник відіграє особливу роль у комп'ютерному дизайні взагалі та у веб-дизайні зокрема. Цю фігуру використовують найчастіше: майже всі об'єкти екрана мають прямокутну форму — вікна, блоки тексту, зображення. Слід пам'ятати правило: якщо прямокутник зафарбований, то рамку для нього треба зробити або досить широку, або взагалі невидиму — тоді вона не сприйматиметься надлишковим елементом.

8. **Круг** або куля виділяються на початкових стадіях розпізнавання. Сфера, еліпс, овал, дуга мають симетрію, що приваблює око. Проте коло і круг є дуже незручними формами для дизайну: їх складно гармонійно поєднати з іншими елементами оформлення. Тому частіше можна зустріти неповні кола — дуги у різноманітних формах. Крім того, застосовують прийом згладжування прямих кутів: їм надають округлої форми, що сприяє легшому сприйняттю загальної композиції.
9. **Контур**. Лінія — це найчастіше межа між двома об'єктами, край, тінь. Якщо лінія контура пряма, це викликає підвищену увагу. Під час огляду зображення зоровий аналізатор миттєво добудовує лінії, яких бракує (наприклад, обриси букв, криву між паралельними прямими), створюючи ілюзію їхньої видимості (рис. 7.39). Так само миттєво око ловить кут, злам лінії.

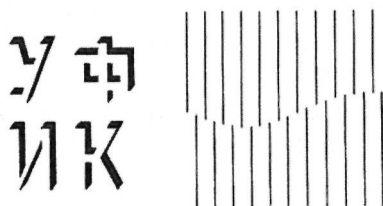


Рис. 7.39. Людина бачить контури літер та криву, хоч їх насправді немає

10. **Текстура**. Це зображення різних поверхонь: гладких, шорстких, переплетених. Досить одного погляду, щоб відчутти, наприклад, м'якість трав'яного килима на узліссі.

11. Образ. Під час розглядання будь-якого зображення слід пам'ятати, що око не міряє — воно порівнює (рис. 7.40). Зоровий аналізатор порівнює два елементи зображення, яскравості фону і об'єкта, а також зображення з його узагальненим еталоном, сформованим у процесі життєдіяльності.

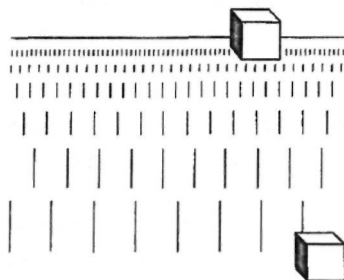


Рис. 7.40. Верхній куб здається більшим, хоча це не так

Колір у веб-дизайні

Вплинути на сприйняття форм можна, використовуючи колір. За певними винятками сприймання колірних поєднань відбувається після аналізу основних форм, розглянутих раніше. А винятком може бути, наприклад, червона пляма на чорно-білому зображенні, яка впаде в око першою.

Правила використання кольорів

Красива і приваблива веб-сторінка потребує використання цілісної колірної схеми оформлення. Щодо застосування кольорів у веб-дизайні можна виділити кілька правил.

Використовуйте достатню кількість кольорів. Колір може підкреслити важливість певних елементів, допомогти їх згрупувати або, навпаки, розділити.

Розділяючи елементи сторінки, не забувайте про поля між ними. Краще, щоб це вільне місце мало колір фону, і бажано, щоб він був світлим.

Краще, коли тло сторінки має світлий колір, а текст — темний. Це звичніше і зручніше для очей та мозку. Водночас яскраві кольори незамінні для привертання уваги.

Враховуйте поєднання кольорів оформлення сторінки та зображень на ній. Зображення будуть губитися на яскравому тлі.

Око людини має рецептори трьох типів, які відповідають за сприйняття кольору і відрізняються своєю чутливістю до електромагнітних коливань хвиль різної довжини. Одні рецептори реагують на фіолетово-синій колір, інші — на зелений, треті — на **оранжево-червоний**. Зоровий апарат людини аналізує світло, визначає у ньому вміст окремих випромінювань, а потім мозок синтезує їх у єдиний колір.

Колірні моделі

У комп'ютерній графіці використовують кілька схем кодування інформації про колір кожного пікселя. Зокрема, це колірні моделі RGB, CMYK та HSV.

RGB є адитивною моделлю, в якій червоний, зелений та синій кольори поєднуються між собою для утворення всіх інших. Базові кольори цієї схеми ще називають первинними. Саме їх сприймають рецептори сітківки людського ока.

Кожен піксель на екрані відтворюється комп'ютером як поєднання червоного, зеленого та синього кольорів за допомогою графічної карти. Якщо вони відсутні, то отримуємо чорний колір, якщо ж усі максимально насичені — білий.

CMYK — колірна модель, яку використовують для друку. Вона працює за принципом поглинання світла і ґрунтується на змішванні блакитного, пурпурового, жовтого та чорного кольорів.

Згідно з моделлю HSV колір складається із трьох складових — відтінку, насиченості (іноді цей параметр ще називають чистотою кольору) та яскравості. Художники віддають перевагу саме цій моделі подання кольору, бо вона найбільш відповідає способу, яким колір відтворює людина. Якщо RGB та CMYK використовують накладання базових кольорів, то HSV містить інформацію про сам колір.

Щоб бути певними, що браузер користувача правильно відобразить кольори, вибрані розробником для дизайну, необхідно дотримуватися гами кольорів, безпечних для браузера. Палітра кольорів, які можна сміливо використовувати для веб-сторінок, складається з 216 елементів. Визначено, що кожен із трьох основних кольорів

(червоний, зелений та синій) може набувати 6 значень: #00, #33, #66, #99, #CC чи #FF (або в іншому поданні — 0 %, 20 %, 40 %, 60 %, 80 %, 100 % інтенсивності). $6^3 = 216$, і це саме та кількість, яка закладена стандартами мови HTML. Сказане означає, що доведеться відмовитись від використання максимальних можливостей монітора і обмежитися тими кольорами, які відображатимуться навіть на найменш потужних дисплеях користувачів.

Гармонія кольорів

Емоційна дія кольорів відома багато століть. Їх традиційно (хоч і досить умовно) поділяють на *холодні* й *теплі*. Холодна гама створює відчуття спокою та рівноваги, тепла — радості, активності, драматизму.

Червоні тони називають гарячими (вогонь), жовті — теплими (сонячне світло), блакитні — прохолодними (небо). Синій (вечір), фіолетовий (захід сонця) і чорний (ніч) — заспокоюють та пригнічують. Водночас темні насичені кольори несуть у собі силу та загадковість, яка змушує хвилюватися. Оранжевий колір створює відчуття благополуччя та веселості, має значну стимулюючу дію, але швидко втомлює. Зелений колір — нейтральний, це колір миру і спокою, до нього око найбільш чутливе і сприймає його передусім. Блакитний, так само як і зелений, має заспокійливий ефект, але у значній кількості може пригнічувати. Червоний колір — теплий і дратівний. Білий утворюється від змішування всіх кольорів, він є втіленням чистоти, миру та холоду, а чорний — це відсутність кольору, він викликає занепокоєння.

Іноді кольори ще поділяють на *легкі* та *важкі*. До легких належать світлі, холодні, малонасичені кольори, особливо синій та блакитний, що нагадують колір неба, простору, далечіні. Важкими називають теплі, темні, насичені кольори — коричневий, оливковий, чорний, темно-сірий, вони асоціюються із землею.

Ще в епоху Відродження почали застосовувати кольори, що виступають та відступають: передній план картини зображали у тепло-коричневій гамі, середній — у нейтрально-зеленій, а дальній — у блакитній. Якщо перед людиною поставити червоний та синій прямокутники, перший сприйматиметься як такий, що розташований ближче.

Не варто допускати використання у невеликому просторі надмірної кількості кольорів. Дизайнери зазвичай обирають для оформлення сторінки не більше чотирьох кольорів, використовуючи кожен із них для розфарбовування кількох елементів. Надмірна яскравість кольору підкреслює дефекти зображення, низька — знижує чутливість до виявлення малих деталей і контрастів.

Один і той самий колір у різному оточенні може здаватися теплим або холодним, тож розподіл кольорів на групи є досить умовним. Так, яскраво-червоний колір порівняно з багряним буде холодним, а поряд із синім — теплим.

Якщо деякий час дивитися на один колір, а потім перевести погляд на інший, то він сприйматиметься спотвореним. Подібні явища називають *контрастом кольорів*. Саме тому сіра фігура на жовтому тлі сприймається синюватою, а на зеленому — дещо рожевою, жовтий колір на зеленому видається оранжевим, рожевий на зеленому стає яскравішим.

Щоб не втомлювалося око відвідувача сайту, необхідно враховувати принципи колірної гармонії. На *колірному колесі* (рис. 7.41) — засоби подання кольору в моделі HSV — кольори змінюються за годинниковою стрілкою за спектром червоний-оранжевий-жовтий-зелений-блакитний-фіолетовий (червоний сектор на колесі розміщений зліва внизу). Є кілька схем колірних гармоній, які можна використати для створення палітри кольорів для своєї роботи.

- *Доповнювальна* — беруть два кольори з протилежних частин колірного колеса (наприклад, червоний та зелений). Коли їх розмістити поряд, вони посилюватимуть інтенсивність одне одного, а якщо змішати між собою, то їхня яскравість зменшиться. Це — контрастна гармонія. Використовують також і часткове доповнення, коли зміщуються два чи більше майже протилежних кольорів.
- *Аналогова* — гармонія кольорів-сусідів на кольоровому колесі. Аналогова гама поєднує близькі тони — жовті (жовтий, жовто-зелений та жовто-оранжевий, третій варіант на рис. 7.41) чи сині (синій, синьо-фіолетовий та синьо-зелений).
- *Триада* — гармонія рівновіддалених між собою кольорів. Зокрема тріаду утворюють червоний, жовтий та синій кольори.

Сприйняття кольору залежить також від площі, яку він заповнює. Слабонасичені чи тьмяні відтінки на малій площі сприймаються гірше, ніж на великій. Це означає, що для менших форм варто використовувати яскравіші кольори, а для розкриття краси слабого кольору краще заповнити ним тло сторінки.



Рис. 7.41. Кольорова гармонія: доповнення, часткове доповнення, аналогова гармонія, тріада

Текстури у веб-дизайні

У комп'ютерній графіці словом «текстура» зазвичай позначають різноманітні імітації реальних поверхонь — мармуру, дерева, цегли тощо, які найчастіше використовуються для заповнення фону. Проте вони є лише одним із класів можливих текстур.

Суцільний колір. Найчастіше для веб-сторінок використовують найпростішу з усіх текстур — звичайне однорідне заповнення кольором. Сучасний дизайн надає перевагу простим, строгим геометричним формам, а для них майже будь-яка текстура здається чужою.

Геометричні структури. Наступним етапом ускладнення текстур є різноманітні геометричні візерунки — смужки, клітинки, ромби тощо. Хоча вони є найближчими родичами суцільного кольору, в дизайні їх використовують порівняно рідко, винятком є горизонтальні або вертикальні вузькі смужки.

Фотографічні текстури. Головним джерелом складніших текстур у сучасному дизайні є фотографія, що містить плавні переходи кольорів і може створити текстуру, яка є нечіткою, неоднорідною, плавною. Вона чудово доповнює прості текстури та кольори і контрастує зі строгими плоскими геометричними формами (літерами тексту і заголовків).

Матеріальні текстури. Імітації матеріальних поверхонь (дерево, метал, скло тощо) професійні веб-дизайнери використовують рід-

ко. Коли ці текстури все ж з'являються на сторінці, важливо обмежитись однією з них, підтримавши її іншими елементами оформлення.

Вправа 7.5

1. Відкрийте почергово приклади веб-сторінок за адресами, які наведено нижче. Визначте, які прийоми застосовувалися для оформлення сторінок:
 - <http://www.csszengarden.com/?cssfile=185/185.css>: «Манхеттен» — ритм та текстура;
 - <http://www.csszengarden.com/?cssfile=181/181.css>: «Красуня в рожевому» — домінантність, пропорція;
 - <http://www.csszengarden.com/?cssfile=152/152.css>: «Метро» — лінії, шрифти;
 - <http://www.csszengarden.com/?cssfile=148/148.css>: «Музей» — пропорційність та лінії;
 - <http://www.csszengarden.com/?cssfile=122/122.css>: «Центральність» — симетрія та колір;
 - <http://www.csszengarden.com/?cssfile=046/046.css>: «Лайм» — точки, ритм та кольорова аналогія;
 - <http://www.csszengarden.com/?cssfile=031/031.css>: «Кутики» — форма і перспектива.
2. Як дизайнерські прийоми впливають на загальне сприйняття сторінки?

Типові помилки дизайнерів-початківців

Звичайно, для того щоб створювати привабливі та інформативні веб-сайти, недостатньо вивчити викладені вище основи дизайну. Досягти цього можна лише шляхом тривалої практики. У цьому підрозділі подано перелік основних порад, які дають досвідчені дизайнери своїм колегам-початківцям. Отже, на «правильних» веб-сторінках:

- кольорова гама сторінки має складатися з одного-двох кольорів;
- у меню не повинно бути більше семи пунктів;

- логотип повинен міститися внизу праворуч чи вгорі ліворуч і мати просту та лаконічну форму;
- дизайн має бути простим, а текст — коротким;
- горизонтальне прокручування вікна неприпустиме.

Крім того, слід взяти до уваги, що:

- бувають неправильні комбінації кольорів;
- чорний колір має негативний (похмурий та гнітючий) підтекст;
- світлий текст на темному тлі читається погано;
- шрифти із засічками читаються легше, ніж без засічок;
- фрейми на сайті небажані.

Кожне з цих правил можна заперечити і довести необхідність та вдалість використання «неправильного варіанта» на конкретному сайті. Тож не варто їх сприймати як безумовні табу. Проте необхідно пам'ятати, що для більшості сайтів усе-таки краще наведених вище рекомендацій дотримуватися.

Варто зазначити, що в Інтернеті досить часто зустрічаються сайти, які є незручними для відвідувачів. Переважно ці незручності мають низку проявів.

- **Повільне завантаження.** Очікування завантаження файлів та завершення процедури пошуку — це марна трата часу. Відвідувач веб-сайту не буде чекати довше десяти секунд, поки завантажиться потрібна сторінка. Важливо подати шукану інформацію якомога швидше та простіше. Для цього слід забезпечити зручну структуру сайту, а також швидке завантаження кожної сторінки.
- **Ненадійність інформації.** Відомостей у мережі Інтернет дуже багато, і відвідувачі часто зустрічаються з неправдивими чи неточними даними. Більшу довіру до сайту відвідувач відчуває, коли чітко зазначено джерела інформації і дату її оновлення. Встановленню довіри також сприяє якісний дизайн.
- **Ілюстрації повинні бути підібрані вдало і виправдано.** Зображення, які не стосуються текстового наповнення, лише відволікають. Слід також використовувати графічні файли малих розмірів для швидшого їх відображення.

- **Суцільний текст, який важко переглядати**, є дуже непривабливим елементом веб-сторінки. Більшість відвідувачів **переглядають** веб-сторінки, і тільки якщо інформація є для них цікавою, вони читають її детальніше. Щоб полегшити цей процес, текст має бути оформлений зручним для читання шрифтом, із виділенням важливих моментів, влучними ілюстраціями тощо.
- **Навігаційні помилки**. На кожній сторінці має чітко зазначитися сайт, якому вона належить, а також міститися гіперпосилання на його головну сторінку. Слід також давати головній сторінці сайту просту і зрозумілу адресу, без використання спеціальних символів.
- **Нестандартне оформлення гіперпосилань**. Відвідані гіперпосилання повинні відрізнятися кольором від невідведаних; гіперпосилання мають також бути змістовними, а не фразою на кшталт «**кляцніть тут**».
- **Нові вікна браузера не можуть відкриватися без попередньої згоди відвідувача чи хоча б попередження**. Винятком є робота з документом, що не є веб-сторінкою (наприклад, із doc- або pdf-файлом).
- **Фіксована ширина сторінки чи елементів на ній**. Відвідувачі сайту повинні мати змогу змінювати шрифт відповідно до своїх потреб за допомогою налаштувань браузера. Тому краще визначати розмір шрифту не абсолютними числами, а відносними величинами.
- **Недоречне використання форм**. З веб-сторінок, що містять форми для заповнення, варто видалити всі запитання, без яких можна обійтись, і не надавати полям статусу обов'язкових без особливої потреби.
- **Невдалі заголовки**, зокрема однакові для всіх сторінок сайту, надто довгі чи неінформативні створюють несприятливе враження про сайт.
- **Ігнорування загальноприйнятих правил, нестандартний інтерфейс**. Стабільність є одним із головних принципів практичності сайту. Якщо однакові дії призводять до однакових наслідків, відвідувач відчуває впевненість у своїх діях. Відвідувачі

сайту вже знають, чого очікувати, де можна знайти меню та елементи керування, тому не варто їх спантеличувати і змушувати закривати вікно браузера через незрозуміле чи незручне подання інформації.

- **Реклама і все, що подібне їй.** Рухомі об'єкти привертають увагу людини, не даючи зосередитися на змісті сторінки. Відвідувачі навчилися не помічати банерів із постійним мерехтінням та яскравими кольорами, рухомих анімацій чи тексту, а також зазвичай закривати всі вікна, відкриті без їхньої згоди. На жаль, при цьому вони можуть не помітити потрібної інформації, розміщеної у такому елементі.
- **Flash та інші сучасні технології** є дуже привабливою можливістю, яка дає змогу дизайнеру повністю реалізувати своє бачення певного сайту. Але варто пам'ятати, що не всі відвідувачі готові чекати кілька хвилин для завантаження ролика перед переглядом сторінки, тому обов'язково слід надати можливість його пропустити. Функції відображення деяких інтерактивних елементів можуть бути вимкнені з міркувань безпеки, тому завжди необхідно передбачати версію сайту з мінімальними вимогами до комп'ютера користувача.
- **Несумісність із браузерами.** Різні браузери можуть по-різному відображати одні й ті самі веб-сторінки. Це слід враховувати і тестувати свій сайт у різних браузерах перед тим, як розміщувати його в Інтернеті. Слід враховувати особливі потреби відвідувачів, зокрема з поганим зором — робити адекватні виділення тегами заголовків, використовувати альтернативні описи зображень тощо.
- **Гіперпосилання зі сторінки на саму себе** є малофункціональними.

Практична робота № 11

Оформлення сайту

Мета роботи: навчитися планувати, створювати та оформляти веб-сайт, враховуючи принципи ергономічного розміщення контенту, дизайну кольору, форми, текстури та шрифтів.

Порядок роботи

Розробіть дизайн статичного комерційного веб-сайту (для прикладу візьміть сайт готелю), запропонувавши певний його стиль та структуру. Крім того, виберіть колірну схему, підберіть шрифти, визначтеся з використанням у проекті різних просторових форм.

Веб-сайт повинен мати кілька сторінок (не менше трьох, не більше п'яти) і бути створений засобами мови HTML або графічного редактора веб-сайтів.

1. У зошиті заповніть таблицю із характеристиками створюваного сайту:

Мета	Матеріали сайту	Інформаційна архітектура	Навігаційна схема
визначте мету створення сайту	вказіть перелік сторінок сайту. а також текстових фрагментів та ілюстрацій, що будуть на них розміщені	визначте тип інформаційної архітектури сайту	вказіть типи елементів навігації, що міститимуться на сторінках сайту

Розробіть дизайн головної сторінки сайту, використавши прийоми групування, вирівнювання, ритму, балансу, пропорцій тощо. Наприклад, головною сторінкою готелю «Затишок» може бути така, як на рис. 7.42.

На цій сторінці елементи вирівняні та згруповані у правій частині екрана. Дотримано ритм та баланс у верхній частині сторінки, де містяться фотографії кімнат. Пропорційно поділено сторінку в нижній частині екрана, що врівноважує її загальний вигляд.

Розробіть дизайн решти сторінок сайту. На кожній сторінці мають бути горизонтальні та бокові меню, а також детальна інформація про вибраний розділ сайту (наприклад, про одну з кімнат готелю, рис. 7.43).

Підберіть текстові матеріали — опис готелю та кімнат із використанням стилю «оберненої піраміди», придумайте влучні заголовки. Виберіть супутні матеріали: як дістатися до готелю, місцеві екскурсії, додаткові послуги тощо. Визначте шрифти для оформлення сайту — гарнітури, кеглі, накреслення.



Рис. 7.42. Головна сторінка сайту готелю



Рис. 7.43. Змістова сторінка сайту

5. Виберіть ілюстрації для сайту, визначте колірну схему з урахуванням принципів гармонійності та балансу кольорів. Наприклад, для сайту готелю з назвою «Затишок» можна використати м'яку коричневу гаму кольорів у аналоговій гармонії: світлі тони застосувати для тла, а насичені темні — для написів.
6. Реалізуйте сторінки сайту за допомогою засобів мови HTML або графічного редактора FrontPage.
7. Проаналізуйте створений сайт на наявність помилок, які могли виникнути під час його оформлення чи виявитися під час навігації.
8. Знайдіть в Інтернеті два веб-сайти зі схожою тематикою.
9. Оцініть кожну з перелічених нижче характеристик за 10-бальною шкалою для сайтів, створених однокласниками. Результати оцінювання запишіть в електронній таблиці:

Назва сайту

URL-адреса сайту

Дата оцінювання

Чи корисна інформація, що міститься на сайті?

Чи зрозуміло викладений матеріал?

Чи наведена контактна інформація?

Скільки розділів має сайт?

Чи є карта сайту?

Чи потрібна карта сайту?

Чи має сайт функцію пошуку?

Сайт має форми для заповнення відвідувачем?

Сторінки сайту мають фіксовану ширину?

Сайт реалізований за допомогою фреймів?

Чи є на сайті недійсні гіперпосилання або сторінки без зв'язку з іншими?

Чи легко прочитати текст на обраному тлі?

Чи можна змінювати розмір шрифту?

Для сайту вибрана доречна колірна гама?

Чи відрізняється колір відвіданих та невідвіданих гіперпосилань?

Чи є альтернативний текст у зображеннях?

Чи є на сайті граматичні або друкарські помилки?

Сайт завантажується достатньо швидко?

10. Обчисліть середню арифметичну оцінку для кожного оцінюваного сайту. В програмі MS Excel це можна зробити за допомогою функції AVERAGE.

Самостійна робота

- Створіть веб-сайт на вибір для однієї з компаній, що:
 - продає одяг;
 - продає книжки;
 - продає автомобілі;
 - продає шкільне приладдя;
 - продає мобільні телефони;
 - продає кондитерські вироби;
 - продає комп'ютерне обладнання;
 - продає меблі;
 - продає квіти;
 - продає картини;
 - займається туризмом;
 - займається прокатом кінофільмів;
 - займається доставкою піци;
 - займається проведенням комп'ютерних курсів;
 - надає косметологічні послуги.
- Оформіть одну й ту саму сторінку розробленого вами сайту за різними колірними та шрифтовими схемами і порівняйте її сприйняття.
- Розгляньте приклади дизайну на сайтах <http://www.templatemonster.com/>, <http://www.freecsstemplates.org/css-templates/>. Скористайтеся ними для оформлення сайту на обрану тематику.

4. За адресами <http://www.freefoto.com/>, <http://flickr.com/explore/>, <http://www.photos.com/> підберіть безкоштовні фотографії та зображення для оформлення свого сайту. Використовуючи фотографії, пам'ятайте про авторські права власників цих знімків.

Тест

1. Веб-дизайн — це:
 - а) міжпредметна галузь, що охоплює планування та розробку веб-сайтів;
 - б) проектування, конструювання, створення вмісту веб-сторінок;
 - в) сукупність веб-сторінок, які об'єднані між собою за змістом та навігаційно;
 - г) стилізація оформлення під ресурси Інтернету;
 - д) проектування структури веб-сайтів.
2. Розрізняють такі типи сайтів:
 - а) текстові, графічні, імітаційні;
 - б) журнальні, шрифтові;
 - в) брошурні, комерційні, особисті;
 - г) фотографічні, матеріальні, геометричні;
 - д) міжнародні, локальні.
3. Розставте у правильному порядку етапи роботи над сайтом:
 - а) визначення мети і планування;
 - б) реалізація;
 - в) оцінювання і підтримка;
 - г) створення інформаційної архітектури;
 - д) дизайн.
4. Навігація — це:
 - а) спосіб організації матеріалів для керування ними;
 - б) процес пересування інформаційним простором Інтернету, а також окремим сайтом;
 - в) схема зв'язків сторінок сайту;

- г) наука, що вивчає спорудження будинків та пересування ними;
 - д) подання інформації в порядку зниження її важливості.
5. Є такі види інформаційної архітектури:
- а) навігаційне меню, зміст сторінки, рядок навігації, закладки;
 - б) все в одному, однорівнева, чітка ієрархія, багатовимірні ієрархія;
 - в) пошук, навігаційне меню, закладки, ромашка;
 - г) індекс, ромашка, пошук, чітка ієрархія;
 - д) брошура, текстова, графічна.
6. Оптимальним розміром сторінки веб-сайту (в пікселях)ють:
- а) 760x410;
 - б) 550x500;
 - в) 1024x768;
 - г) 800x600;
 - д) 480x256.
7. Стил «оберненої піраміди» — це:
- а) подання інформації в порядку зростання її важливості;
 - б) товщина штрихів та співвідношення площі білого і чорного в малюнку літер;
 - в) розміщення ключової інформації на початку речень, абзаців;
 - г) виділення окремих слів кольорами;
 - д) подання інформації в порядку зниження її важливості.
8. Є такі види шрифтів:
- а) підкреслені, курсивні, розріджені;
 - б) фіолетові, блакитні, жовті, чорні;
 - в) суцільні, геометричні, фотографічні, текстові;
 - г) великі, малі;
 - д) із засічками, без засічок, фіксованої ширини, декоративні.

9. Виберіть правильний порядок сприймання елементів дизайну відвідувачем сайту:
- а) горизонтальна пряма лінія, вертикальна пряма лінія, круг;
 - б) круг, вертикальна пряма лінія, симетрія;
 - в) кут, трикутник, квадрат, вертикальна пряма лінія, похила пряма лінія;
 - г) вертикальна пряма лінія, горизонтальна пряма лінія, круг, розмір;
 - д) колір, розмір, трикутник, лінія, симетрія.
10. Текстура — це:
- а) характер поверхні певного об'єкта, зумовлений його фізичними властивостями;
 - б) рівновага зображення, розташування об'єктів та їх візуальна вага у композиції;
 - в) зовнішні обриси, вигляд, контур об'єкта;
 - г) імітація реальної поверхні, яку використовують для заповнення тла сторінки;
 - д) сукупність веб-сторінок, доступних у мережі Інтернет.
11. Вкажіть кількість кольорів, що входять у гаму безпечних кольорів для будь-якого браузера:
- а) 256;
 - б) 760;
 - в) 1024;
 - г) 216;
 - д) 25.
12. Виберіть правильні твердження:
- а) горизонтальне прокручування — вдале рішення для сайту;
 - б) веб-сторінка повинна завантажуватися не довше десяти секунд;
 - в) всім сторінкам сайту слід давати однакові заголовки;
 - г) потрібно призначати однакові кольори для відвіданих та невідвіданих гіперпосилань;
 - д) на сайті слід зазначати контактну інформацію та дату оновлення матеріалів.

Посібники серії «Інформатика. Профільне навчання» для курсів за вибором

Посібники серії відповідають програмам курсів за вибором, що розширюють зміст розділів базового курсу інформатики для 12-річної школи.

Видання адресовані учням середніх загальноосвітніх шкіл, ліцеїв і гімназій, навчально-виховних комплексів, професійно-технічних училищ і коледжів та їхнім вчителям.

Посібники містять теоретичний матеріал, вправи, завдання для самостійного виконання, приклади виконаних завдань, тести тощо. До посібників додаються компакт-диски з електронними навчальними матеріалами. Усі посібники мають гриф «Рекомендовано Міністерством освіти і науки України».

О. М. Левченко та ін. Основи Інтернету

Видавнича група ВНУ, 2008, ISBN 978-966-552-227-0, 320 с.

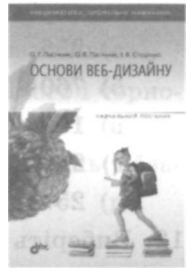
У посібнику розглянуто комплекс питань, пов'язаних із вивченням основ Інтернету та застосуванням набутих знань на практиці. Докладно описано принципи функціонування Вебу, електронної пошти, служб обміну повідомленнями та передавання файлів. Подано засади дотримання інформаційної безпеки, ґрунтовні відомості щодо пошуку інформації та створення онлайн-ових щоденників-блогів.



О. Г. Пасічник та ін. Основи веб-дизайну

Видавнича група ВНУ, 2008, ISBN 978-966-552-228-7, 336 с.

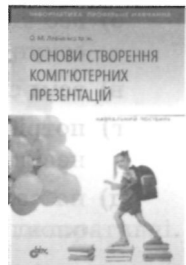
Викладено сучасні підходи до розробки та адміністрування веб-сайтів. Розглянуто базові принципи організації онлайн-ових ресурсів, описано засоби автоматизованого створення веб-сайту на безкоштовному сервері, роботу в середовищі Microsoft FrontPage, використання мови HTML і таблиць каскадних стилів. Важливе значення має розділ, присвячений принципам графічного та стилістичного оформлення веб-сторінок.



О. М. Левченко та ін. Основи створення комп'ютерних презентацій

Видавнича група ВНУ, 2008, ISBN 978-966-552-197-6, 384 с.

Розглянуто новітні методи вирішення однієї з найактуальніших проблем сучасного світу — подання своїх знань, досягнень та можливостей. Сьогодні це не можна реалізувати без засобів розробки та демонстрування електронних презентацій. У посібнику описано технології роботи з двома найпопулярнішими засобами створення мультимедійних презентацій — Microsoft PowerPoint і Flash.



I. О. Завадський та ін. Основи візуального програмування

Видавнича група BHV, 2008, ISBN 978-966-552-226-3, 272 с.

У посібнику розглянуто такі теми: поняття алгоритму, програми, мови програмування; середовище розробки програм Visual Studio .NET; конструювання форм; елементи керування; змінні й присвоювання; введення й виведення даних; арифметичні, логічні та рядкові операції; алгоритми та програми з розгалуженнями; циклічні алгоритми та програми; процедури й функції; одновимірні масиви. Запропоновано методику навчання програмування на прикладах розробки програм у візуальному середовищі. Навчальною мовою програмування обрано Visual Basic .NET.



В. С. Березовський та ін. Основи комп'ютерної графіки

Видавнича група BHV, 2008, ISBN 978-966-552-196-9, 320 с.

Докладно розглянуто основні поняття та засоби комп'ютерної графіки. У першій частині описано переваги й недоліки растрової та векторної графіки, розглянуто колірні моделі, графічні редактори і формати файлів. Виконуючи практичні завдання, учні навчаються працювати із графічними засобами Microsoft Office, графічним редактором Paint, програмою створення анімації та програмами перегляду графічних файлів. У другій частині викладено основи роботи з графічними редакторами Photoshop та CorelDraw, наведено загальні відомості про професійні редактори (3ds Max, AutoCad, Archicad та MAYA).



Замовлення навчальних посібників та додаткових методичних матеріалів

Тел.: +38 (044) 237-08-05

Електронна адреса: osvita@bhv.kiev.ua, market@osvita.info

Поштова адреса: Видавнича група BHV, 03115, м. Київ, а/с 45

Онлайнова форма: <http://osvitaonline.googlepages.com/>

Проект «Онлайнова освіта - профільне навчання»

Запрошуємо вас приєднатися до проекту «Онлайнова освіта - профільне навчання», метою якого є впровадження онлайн-технологій у профільній школі. У рамках реалізації проекту для освіти передбачені майстер-класи, тренінги та сертифікація за профільними курсами та курсом «Онлайнові освітні середовища». Видавнича група BHV здійснюватиме методичну підтримку таких онлайн-середовищ: тематичних середовищ курсів з інформатики, шкільних середовищ «учень-вчитель», а також середовищ для навчання вчителів.

Укласти угоду про співпрацю з нами ви можете, звернувшись за телефоном: +38 (044) 593-74-27 або надіславши запит електронною поштою: pg@bhv.kiev.ua чи звичайною поштою за адресою: Видавнича група BHV, 01032, м. Київ, вул. Старовокзальна, 13.

Стежте за новинами:

<http://www.bhv.kiev.ua>,

<http://osvitaonline.googlepages.com/>



Навчальне видання

**Олена Георгіївна Пасічник
Оксана Володимирівна Пасічник
Ірина Володимирівна Стеценко**

ОСНОВИ ВЕБ-ДИЗАЙНУ

Керівник проекту О. М. Левченко
Редактор В. М. Бабійчук
Коректор В. М. Бабійчук
Комп'ютерна верстка Д. С. Трішенкова

ТОВ «Видавнича група ВНУ»
Свідоцтво про внесення до Державного реєстру
суб'єктів видавничої справи України
серія ДК №175 від 13.09.2000 р.

Підписано до друку 25.08.08. Формат 60x84 1/16. Папір офсетний.
Гарнітура ShoolBook, Pragmatica. Друк офсетний.
Ум. друк. арк. 19,53. Обл.-вид. арк. 17,31.
Наклад 1500 прим. Зам. № 8181.

Віддруковано з готових діапозитивів
на ДП «Державна картографічна фабрика»
21100, м. Вінниця, вул. 600-річчя, 19
Свідоцтво про внесення суб'єкта видавничої справи
до Державного реєстру видавців, виготівників і розповсюджувачів
видавничої продукції Серії ДК № 869 від 26.03.2002 р.

